

# Volume 11: DAF21 Construction Code

**April 2023**

---

**Submitted to:**

Social Security Administration  
Office of Retirement and Disability Policy  
Office of Research, Demonstration, and Employment  
Support  
Washington, DC 20024-2796  
Project Officers: Paul O’Leary and Debra Tidwell-Peters  
Contract Number: 28321322D00060010

**Submitted by:**

Mathematica  
1100 1st Street, NE  
12th Floor  
Washington, DC 20002-4221  
Telephone: (202) 484-9220  
Facsimile: (202) 863-1763  
Project Director: Matthew Urato  
Reference Number: 51390.BY.T02.260.000

**Suggested Citation:** “Disability Analysis File 2021 (DAF21) Documentation: Data from January 1994 through December 2021.” Washington, DC: Mathematica, April 2023.

**This page has been left blank for double-sided copying.**

## Contents

Glossary.....	iv
Overview of DAF Documentation .....	vii
Appendix A: DAF Programs.....	A.1
Appendix B: ADM Code.....	B.1

## Glossary

AB	Accelerated Benefits Demonstration
ADM	Awardee Data Mart
AIME	Average Indexed Monthly Earnings
BEST	Benefits Entitlement Services Team
BFW	Benefits forgone due to work
BIC	Beneficiary Identification Code
BMF	Budget Month Factor
BOAN	Beneficiary's Own Account Number
BOND	Benefit Offset National Demonstration
BOPD	Benefit Offset Pilot Demonstration
CAN	Claim Account Number
CDR	Continuing Disability Review
CER	Characteristics Extract Record 100% Field File
COLA	Cost-of-Living Adjustment
DAC	Disabled Adult Child
DAF	Disability Analysis File (previously known as TRF)
DBAD	Disabled Beneficiary and Dependents Extract
DCF	Disability Control File
DDS	Disability Determination Services
DER	Detailed Earnings Record
DI	Disability Insurance, also referred to as SSDI
DMG	Demographic component of the DAF
DWB	Disabled Widow Beneficiaries
EN	Employment Network (also called a TTW provider)
EPE	Extended Period of Eligibility
EVS	Enumeration Verification System
EXR	Expedited Reinstatement
FBR	Federal Benefit Rate
FCI	Federal Countable Income
FIPS	Federal Information Processing Standards (in reference to U.S. Census standardized codes for uniform identification of geographic entities)
FRA	Full Retirement Age
HI	Hospital Insurance (Medicare Part A)
HOPE	Homeless Outreach Projects and Evaluation Demonstration

HUN	Housed Under Number
ICD-9	International Classification of Diseases Coding Scheme
IPE	Individualized Plan for Employment, developed by State VR Agency
IRS	Internal Revenue Service
IRWE	Impairment-Related Work Expense
LAF	Ledger Account File
LAUS	Local Area Unemployment Statistics
MBR	Master Beneficiary Record
MEF	Master Earnings File
MHTS	Mental Health Treatment Study
MIE	Medical Improvement Expected
MO	Milestone + Outcomes payment system
MPR-EVS	Mathematica's EVS
NBS	National Beneficiary Survey
NSCF	National Survey of SSI Children and Families
NUMIDENT	Numerical Identification File
OIM	Office of Information Management
OO	Outcomes-Only payment system
PAN	Person's Account Number
PASS	Program to Achieve Self-Support
PHUS	Payment History Update System
PIA	Primary Insurance Amount
PIN	Personal Identification Number
POD	Promoting Opportunity Demonstration
POMS	SSA's Program Operations Manual System
PROMISE	Promoting Readiness of Minors in SSI
Provider	Service provider under TTW (also called an EN)
PUF	Public Use File
REMICS	Revised Management Information Counts System
RIB	Retirement Insurance Benefits
RMA	Retrospective Monthly Accounting
RSA	Rehabilitation Services Administration
RSA-911	RSA Case Service Report
SAIPE	Small Area Income and Poverty Estimates
SAS	Statistical Analysis Software, used to produce the DAF

SCWF	SSI Companion Work File
SED	Supported Employment Demonstration
SER	Summary Earnings Record
SGA	Substantial Gainful Activity
SMI	Supplemental Medical Insurance (Medicare Part B)
SNAP	Supplemental Nutrition Assistance Program
SSN	Social Security Number
SSA	Social Security Administration
SSDI	Social Security Disability Insurance (also referred to as DI)
SSI	Supplemental Security Income
SSI-LF	SSI - Longitudinal File
SSR	Supplemental Security Record
STW	Suspension or termination of cash benefits for work
T2	Title II, the SSDI Program
T16	Title XVI, the SSI Program
TANF	Temporary Assistance for Needy Families
TCNEI	Total countable non-earned income
TKT	DAF component containing data related to TTW participation
TRF	Ticket Research File, now called the DAF
TTW	Ticket to Work
TWP	Trial Work Period
VR	Federal/State Vocational Rehabilitation program/agency
VRRMS	Vocational Rehabilitation Reimbursement Management System; data from this system is contained in the Payments component
YTD	Youth Transition Demonstration

## Overview of DAF Documentation

The documentation for the DAF consists of the eleven volumes described below. Questions about these documents should be directed to [ORDES.DAF@ssa.gov](mailto:ORDES.DAF@ssa.gov). All of these documents are available at <https://www.ssa.gov/disabilityresearch/daf.html>.

- **Volume 1: Getting Started with the DAF21.** Provides an overview of the structure and contents of the DAF and related linkable files.
- **Volume 2: Working with the DAF21.** Contains practical suggestions such as how to extract data and interpret blank or missing variables as well as more detailed information on DAF data marts and linkable files.
- **Volume 3: Tips for Conducting Analysis with the DAF21.** Contains suggestions for working with common research concepts in the DAF such as program participation, benefits paid versus benefits due, and constructed measures related to beneficiary work activity resulting in the loss of cash benefits.
- **Volume 4: Lists of DAF21 Variables.** Contains lists of new, changed, and deleted variables, as well as lists of variables by DAF component and analytic category.
- **Volume 5: DAF Variable Detail Pages.** Contains specifications for each DAF variable, including name, definition, data format, identification of the DAF component to which it belongs, data source, availability, and (where applicable) SAS code used to construct the variable.
- **Volume 6: Validating the DAF21 Against Other Sources.** Provides an explanation of validation methods and summary of validation results.
- **Volume 7: DAF21 Development History and Construction Methods.** Describes key changes in DAF construction methodology over time as well as a description of each step in the current year DAF construction process.
- **Volume 8: DAF21 Construction Workflow Charts and Task Tables.** Provides detailed information in both chart and table format on each step in the current year DAF construction process.
- **Volume 9: DAF21 Source File Descriptions.** Describes the administrative source files used to construct the DAF.
- **Volume 10: DAF21 Administrative Source File Documentation.** Contains documentation from SSA or other agencies on the administrative source files described in Volume 9.
- **Volume 11: DAF21 Construction Code.** Contains all SAS code used to construct the DAF.
- **Volume 12: DAF21 RSA Administrative Source File Documentation.** Contains a description of the processing of Rehabilitation Services Administration (RSA) data for linkage to the DAF, along with documentation from RSA on the RSA-911 files.

The following table provides specific locations for common research-related questions and issues.

In order to ...	Refer to ...
Get started with a research task	Volume 2, "Working with the DAF21," for information about selecting beneficiaries using finder files versus selection criteria
Identify what's changed in the latest version of the DAF	Volume 1, "Getting Started with the DAF21"
View lists of DAF variables	Volume 4, "Lists of DAF21 Variables"
Understand individual variable definitions, specifications, and value ranges	Volume 5, "DAF Variable Detail Pages"
Understand the structure of the DAF data files at a high level	Volume 1, "Getting Started with the DAF21"
Identify variables for a specific research task	Volume 4, "Lists of DAF21 Variables," for a list of variables contained within each DAF file and by analytic category
Understand the beneficiaries for which the DAF does and does not contain data	Volume 1, "Getting Started with the DAF21"
Identify administrative data sources for the DAF	Volume 9, "DAF21 Source File Descriptions"
Understand the linkage of the DAF to RSA-911 data and contents of the RSA files	Volume 12, "DAF21 RSA Administrative Source File Documentation"
Generate ideas for using the DAF more efficiently	Volume 1, "Getting Started with the DAF21" and Volume 2, "Working with the DAF21"
Find suggested ways to identify common research concepts in the DAF, such as calculating age of retirement, or disability title	Volume 3, "Tips for Conducting Analysis with the DAF21"
Understand what variables have changed in the most recent DAF	Volume 4, "Lists of DAF21 Variables"
Read about how information in the DAF is validated against other sources	Volume 6, "Validating the DAF21 Against Other Sources"



**Appendix A:**  
**DAF Programs**

## Appendix A.1

JCL/SAS Code: FRACALC

```
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES;
/* THIS PROGRAM ADJUSTS THE BIRTHDAY AND CALCULATES THE RETIRE DATE*/

/* IF BIRTHDAY IS ON JANUARY 1ST, ADJUST THE BIRTHDAY TO 12/31 OF
PREVIOUS YEAR FOR COMPUTING RETIREMENT DATE.
IF BIRTHDAY IS IN FEB. AND AFTER THE 28TH, ADJUST THE BIRTHDAY TO
FEB. 28TH OF THE SAME YEAR.
INPUTS: ORIGINAL BIRTHDAY BEFORE ADJUSTMENT
OUTPUTS: ADJUSTED BIRTHDATE, FRADATE, FRTRAGE*/

IF MONTH(&OLD_DOB.)=1 AND DAY(&OLD_DOB.)=1 THEN
  BIRTHDATE = &OLD_DOB.-1;
ELSE IF MONTH(&OLD_DOB.)=2 AND DAY(&OLD_DOB.)>28 THEN
  BIRTHDATE = MDY(2,28,YEAR(&OLD_DOB.));
ELSE BIRTHDATE = &OLD_DOB.;

/*CALCULATE THE RETIRE DATE*/
IF YEAR(BIRTHDATE) LE 1937 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,(65*12),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) = 1938 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,((65*12)+2),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) = 1939 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,((65*12)+4),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) = 1940 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,((65*12)+6),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) = 1941 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,((65*12)+8),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) = 1942 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,((65*12)+10),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) GE 1943 AND
YEAR(BIRTHDATE) LE 1954 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,(66*12),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) = 1955 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,((66*12)+2),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) = 1956 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,((66*12)+4),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) = 1957 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,((66*12)+6),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) = 1958 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,((66*12)+8),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) = 1959 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,((66*12)+10),"SAME DAY");
ELSE IF YEAR(BIRTHDATE) GE 1960 THEN FRADATE =
  INTNX('MONTH',BIRTHDATE,(67*12),"SAME DAY");
```

Appendix A.1

JCL/SAS Code: FRACALC

```
FRA = INTCK('MONTHS',BIRTHDATE,FRADATE,'C')/12;  
DROP BIRTHDATE;
```

Appendix A.2

JCL/SAS Code: YMMMACR

```
%MACRO ASGYMM;
%GLOBAL TOTMOS;
%GLOBAL MOCNTER;

/* ASSIGN MACROS VARIABLES TO HANDLE TIME SERIES DATA */
/* MACRO REVISED FOR DAF13 IN THIS PROGRAM (DAF14 MAY
   USE IN ALL PROGRAMS)IN ADDED A %PUT SHOWING MAPPING
   OF MONTHS 1 TO N TO EACH ASSOCIATED YMM */
/*INITIATE MONTH COUNTER, MOCNTER, TO 1*/
%LET MOCNTER=1;
/*LOOPING THROUGH EACH YEAR SPECIFIED ABOVE*/
%DO YEAR=&BEGYR. %TO &ENDYR.;
/*SETTING YR MACRO VAR TO LAST TWO DIGITS OF EACH YEAR*/
%LET YR=%SUBSTR(&YEAR.,3,2);

/*LOOP THROUGH MONTHS 1 TO 12 AND INCREMENT MONTH COUNTER*/
%DO M=1 %TO 12;
%GLOBAL YY&MOCNTER;
%LET M=%SYSFUNC(PUTN(&M.,Z2.));
%LET YY&MOCNTER.=%EVAL(&YR.)%EVAL(&M.);
%LET MOCNTER=%EVAL(&MOCNTER.+1);
%END;
%END;

/*SET TOTAL MONTHS, TOTMOS, EQUAL TO MOCNTER - 1 TO GET
TOTAL MONTHS IN TIME SERIES*/
%LET TOTMOS=%EVAL(&MOCNTER.-1);

/*USED TO PRINT MAPPING TO LOG - CHECK DURING CODE REVIEW*/
%DO MM=1 % TO &TOTMOS.;
%PUT &MM. MAPS TO YMM, &&YY&MM.;
%END;
%PUT TOTAL MONTHS = &TOTMOS.;

%MEND ASGYMM;
%ASGYMM;
```

## Appendix A.3

JCL/SAS Code: MSPLIT

```
/* THIS MACRO SPLITS THE MULTI-YEAR DATA INTO YEAR-SPECIFIC FILES;  
INCLUDE IT INTO YOUR PROGRAM;
```

## INSTRUCTIONS:

1. TO RUN THIS MACRO CREATE A PROGRAM WITH JCL NAMES FOR INPUT AND OUTPUT DATA SETS, THEN INCLUDE THIS MACRO;
2. SET YOUR PARAMETERS CORRECTLY;

## PARAMETERS:

1. OLDFILE - YOUR EXISTING MULTI-YEAR DATA (NNNN.NNNN)
2. FILELOC - JCL LOCATION FOR YOUR YEARLY FILES,  
IT SHOULD CONSIST OF NAME STUB + YEAR  
(E.G., FOR NUM99, NUM00, NUM01, ETC. FILELOC = NUM)
3. FILENAME - SAS DATASET NAMES FOR YOUR YEARLY FILES  
(IT SHOULD BE THE SAME FOR ALL NEW FILES)
4. START\_YEAR - THE FIRST YEAR FOR NEW FILES
5. END\_YEAR - THE LAST YEAR FOR NEW FILES
6. REGVARS - THE LIST OF NON-YEAR-SPECIFIC VARIABLES,  
WHICH SHOULD BE KEPT IN EACH OF NEW FILES

## EXAMPLE:

```
<OLD FILE>  
//IN1 DD DSN=OPDR.TG.PRD.ETTW.<*>.SA.V1,DISP=SHR  
<NEW FILES, FROM 1994 TO 2012>  
//OUT94 DD DSN=OPDR.TG.PRD.ETTW.<*>.NUMDAT94.SA.V1,  
// DISP=(NEW,CATLG,DELETE),  
// UNIT=TSILO  
//OUT95 DD DSN=OPDR.TG.PRD.ETTW.<*>.NUMDAT95.SA.V1,  
// DISP=(NEW,CATLG,DELETE),  
// UNIT=TSILO  
//OUT96 DD DSN=OPDR.TG.PRD.ETTW.<*>.NUMDAT96.SA.V1,  
// DISP=(NEW,CATLG,DELETE),  
// UNIT=TSILO  
<***>  
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.<*>.NUMDAT12.SA.V1,  
// DISP=(NEW,CATLG,DELETE),  
// UNIT=TSILO  
<*OTHER NECESSARY JCL STATEMENTS*>  
  
%INCLUDE ...  
  
%OUTP(IN1.ALLYEARS,OUT,YEARF,1994,2012,SSN REC ...);  
*/
```

## Appendix A.3

JCL/SAS Code: MSPLIT

```
OPTIONS NOSYMBOLGEN MPRINT;
%MACRO OUTP (OLDFILE,FILELOC,FILENAME,START_YEAR,END_YEAR,REGVARS);

PROC CONTENTS DATA=&OLDFILE OUT=CONT_A ORDER=VARNUM;RUN;

DATA
  CONT_A;
  SET CONT_A;
  IF STRIP(UPCASE(NAME)) = "SSN"
    THEN DELETE;
  RUN;

%LET CHNUM = "0" "1" "2" "3" "4" "5" "6" "7" "8" "9";

DATA CONT_A1 (KEEP = BODY);
LENGTH BODY $15.;
SET CONT_A (KEEP=NAME);

IF SUBSTR(NAME,LENGTH(NAME),1) IN (&CHNUM.)
& SUBSTR(NAME,LENGTH(NAME)-1,1) IN (&CHNUM.)
THEN _MNTH = INPUT(SUBSTR(NAME,LENGTH(NAME)-1,2),BEST.);

/*CHECK THAT WE HAVE 12 MONTHS*/
RETAIN COUNT_MNTH;
IF _MNTH = 1 THEN COUNT_MNTH = _MNTH;
ELSE IF _MNTH > 1 THEN COUNT_MNTH +1;
ELSE IF _MNTH < 1 THEN COUNT_MNTH = .;

IF COUNT_MNTH = 12 &
SUBSTR(NAME,LENGTH(NAME)-2,1) IN (&CHNUM.)
& SUBSTR(NAME,LENGTH(NAME)-3,1) IN (&CHNUM.)
THEN DO;
IF SUBSTR(NAME,LENGTH(NAME)-3,1) IN ("0" "1") THEN
  _YR = INPUT(SUBSTR(NAME,LENGTH(NAME)-3,2),BEST.) + 2000;
ELSE IF SUBSTR(NAME,LENGTH(NAME)-3,1)="9" THEN
  _YR = INPUT(SUBSTR(NAME,LENGTH(NAME)-3,2),BEST.) + 1900;
IF &START_YEAR. <= _YR <= &END_YEAR. THEN DO;
  BODY = SUBSTR(NAME,1,LENGTH(NAME)-4);
  OUTPUT ;
END;
END;

RUN;

PROC FREQ DATA=CONT_A1;
```

Appendix A.3

JCL/SAS Code: MSPLIT

```
TABLES BODY / NOPRINT OUT=CONT_A2;  
RUN;
```

```
DATA _NULL_;  
SET CONT_A2 END = _A_;  
PUT "ALL 12 MONTHS OF " BODY "AVAILABLE FOR " COUNT "YEARS";  
CALL SYMPUT("BODY"||LEFT(_N_),TRIM(LEFT(BODY)));  
IF _A_ THEN DO;  
CALL SYMPUT("N_BODIES",_N_);  
END;  
RUN;
```

```
/*SPLIT THE DATA*/
```

```
DATA  
%DO I = &START_YEAR %TO &END_YEAR;  
%LET J = %SYSFUNC(SUBSTR(&I.,3,2));  
  
%LET YEARVARS = ;  
%DO B = 1 %TO &N_BODIES.;  
%LET YEARVARS = &YEARVARS &&BODY&B.&J.;;  
%END;  
  
%PUT &I. : &YEARVARS;  
&FILELOC&J.&FILENAME (KEEP = &REGVARS &YEARVARS)  
%END;  
;  
SET &OLDFILE;  
RUN;  
%MEND;
```

Appendix A.4

JCL/SAS Code: TWPA

```
*****. ;  
***** TWPA ***** ;  
*****. ;
```

```
* THIS IS AN INCLUDE FILE FOR YEARLY TWP RATES;  
* FIND TABLE AT HTTPS://WWW.SSA.GOV/OACT/COLA/TWP.HTML ;  
* UPDATED THROUGH 2021 ;
```

```
ARRAY TWPA(1994:2021) TWP1994-TWP2021
```

```
  ( 200 /*1994*/  
    200 /*1995*/  
    200 /*1996*/  
    200 /*1997*/  
    200 /*1998*/  
    200 /*1999*/  
    200 /*2000*/  
    530 /*2001*/  
    560 /*2002*/  
    570 /*2003*/  
    580 /*2004*/  
    590 /*2005*/  
    620 /*2006*/  
    640 /*2007*/  
    670 /*2008*/  
    700 /*2009*/  
    720 /*2010*/  
    720 /*2011*/  
    720 /*2012*/  
    750 /*2013*/  
    770 /*2014*/  
    780 /*2015*/  
    810 /*2016*/  
    840 /*2017*/  
    850 /*2018*/  
    880 /*2019*/  
    910 /*2020*/  
    940 /*2021*/  
  );
```



Appendix A.5

JCL/SAS Code: SGANB

```
*****. ;
***** SGANB ***** ;
*****. ;
```

```
* THIS IS AN INCLUDE FILE FOR YEARLY SGA NON-BLIND RATES;
* FIND TABLE AT HTTPS://WWW.SSA.GOV/OACT/COLA/SGA.HTML ;
* UPDATED THROUGH 2021 ;
```

```
ARRAY SGANB(1994:2021) SGANB1994-SGANB2021
  ( 500 /*1994*/
    500 /*1995*/
    500 /*1996*/
    500 /*1997*/
    500 /*1998*/
    700 /*1999*/
    700 /*2000*/
    740 /*2001*/
    780 /*2002*/
    800 /*2003*/
    810 /*2004*/
    830 /*2005*/
    860 /*2006*/
    900 /*2007*/
    940 /*2008*/
    980 /*2009*/
    1000 /*2010*/
    1000 /*2011*/
    1010 /*2012*/
    1040 /*2013*/
    1070 /*2014*/
    1090 /*2015*/
    1130 /*2016*/
    1170 /*2017*/
    1180 /*2018*/
    1220 /*2019*/
    1260 /*2020*/
    1310 /*2021*/
  );
```

Appendix A.6

JCL/SAS Code: SGAB

```
*****
***** SGAB *****
*****;
```

```
* THIS IS AN INCLUDE FILE FOR YEARLY SGA BLIND RATES;
* FIND TABLE AT HTTPS://WWW.SSA.GOV/OACT/COLA/SGA.HTML ;
* UPDATED THROUGH 2021 ;
```

```
ARRAY SGAB(1994:2021) SGAB1994-SGAB2021
```

```
( 930 /*1994*/
  940 /*1995*/
  960 /*1996*/
 1000 /*1997*/
 1050 /*1998*/
 1110 /*1999*/
 1170 /*2000*/
 1240 /*2001*/
 1300 /*2002*/
 1330 /*2003*/
 1350 /*2004*/
 1380 /*2005*/
 1450 /*2006*/
 1500 /*2007*/
 1570 /*2008*/
 1640 /*2009*/
 1640 /*2010*/
 1640 /*2011*/
 1690 /*2012*/
 1740 /*2013*/
 1800 /*2014*/
 1820 /*2015*/
 1820 /*2016*/
 1950 /*2017*/
 1970 /*2018*/
 2040 /*2019*/
 2110 /*2020*/
 2190 /*2020*/
);
```

Appendix A.7

JCL/SAS Code: FBRIN

```
*****
***** FBRIN *****
*****;
```

```
* THIS IS AN INCLUDE FILE FOR YEARLY FBR RATES FOR INDIVIDUALS;
* FIND TABLE AT HTTPS://WWW.SSA.GOV/OACT/COLA/SSIAMTS.HTML ;
* UPDATED THROUGH 2021 ;
```

```
ARRAY FBRIN(1994:2021) FBRIN1994-FBRIN2021
```

```
(446.00 /* 1994 */
458.00 /* 1995 */
470.00 /* 1996 */
484.00 /* 1997 */
494.00 /* 1998 */
500.00 /* 1999 */
513.00 /* 2000 */
531.00 /* 2001 */
545.00 /* 2002 */
552.00 /* 2003 */
564.00 /* 2004 */
579.00 /* 2005 */
603.00 /* 2006 */
623.00 /* 2007 */
637.00 /* 2008 */
674.00 /* 2009 */
674.00 /* 2010 */
674.00 /* 2011 */
698.00 /* 2012 */
710.00 /* 2013 */
721.00 /* 2014 */
733.00 /* 2015 */
733.00 /* 2016 */
735.00 /* 2017 */
750.00 /* 2018 */
771.00 /* 2019 */
783.00 /* 2020 */
794.00 /* 2021 */
);
```

Appendix A.8

JCL/SAS Code: FBRCP

```
*****
***** FBRCP *****
*****;
```

```
* THIS IS AN INCLUDE FILE FOR YEARLY FBR RATES FOR COUPLES;
* FIND TABLE AT HTTPS://WWW.SSA.GOV/OACT/COLA/SSIAMTS.HTML ;
* UPDATED THROUGH 2021 ;
```

```
ARRAY FBRCP(1994:2021) FBRCP1994-FBRCP2021
```

```
(669.00 /* 1994 */
687.00 /* 1995 */
705.00 /* 1996 */
726.00 /* 1997 */
741.00 /* 1998 */
751.00 /* 1999 */
769.00 /* 2000 */
796.00 /* 2001 */
817.00 /* 2002 */
829.00 /* 2003 */
846.00 /* 2004 */
869.00 /* 2005 */
904.00 /* 2006 */
934.00 /* 2007 */
956.00 /* 2008 */
1011.00 /* 2009 */
1011.00 /* 2010 */
1011.00 /* 2011 */
1048.00 /* 2012 */
1066.00 /* 2013 */
1082.00 /* 2014 */
1100.00 /* 2015 */
1100.00 /* 2016 */
1103.00 /* 2017 */
1125.00 /* 2018 */
1157.00 /* 2019 */
1175.00 /* 2020 */
1191.00 /* 2021 */
);
```

Appendix A.9

JCL/SAS Code: COLA

```
*****  
***** COLA *****  
*****
```

```
* THIS IS AN INCLUDE FILE FOR YEARLY COLA RATES;  
* FIND TABLE AT HTTPS://WWW.SSA.GOV/OACT/COLA/SSIAMTS.HTML ;  
*
```

```
* UPDATED THROUGH 2021 ;  
ARRAY COLA(1994:2021) COLA1994-COLA2021
```

```
    (.026 /* 1994 */  
    .028 /* 1995 */  
    .026 /* 1996 */  
    .029 /* 1997 */  
    .021 /* 1998 */  
    .013 /* 1999 */  
    .025 /* 2000 */  
    .035 /* 2001 */  
    .026 /* 2002 */  
    .014 /* 2003 */  
    .021 /* 2004 */  
    .027 /* 2005 */  
    .041 /* 2006 */  
    .033 /* 2007 */  
    .023 /* 2008 */  
    .058 /* 2009 */  
    0 /* 2010 */  
    0 /* 2011 */  
    .036 /* 2012 */  
    .017 /* 2013 */  
    .015 /* 2014 */  
    .017 /* 2015 */  
    0 /* 2016 */  
    .003 /* 2017 */  
    .020 /* 2018 */  
    .028 /* 2019 */  
    .016 /* 2020 */  
    .013 /* 2021 */  
    );
```

## Appendix A.10

JCL/SAS Code: DBADJCL

```
//#6266DBA JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#7429
//*****
//* PROJECT:   DAF 50214
//* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.FINDER.PRDLIB(DBADJCL)
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   2/15/2021 SERGE LUKASHANETS FOR DAF21
//* PHONE:
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//FINDPROC PROC
//FIND   EXEC SAS9,
//      WORK='120000,60000'
//*
//IN    DD DSN=MTOSI.T2.DBADMBR.D21&M,DISP=SHR
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DBAD21&M..SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.FINDER.PRDLIB(DBADREAD),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//FRUN01 EXEC FINDPROC,M=01
//FRUN02 EXEC FINDPROC,M=02
//FRUN03 EXEC FINDPROC,M=03
//FRUN04 EXEC FINDPROC,M=04
//FRUN05 EXEC FINDPROC,M=05
//FRUN06 EXEC FINDPROC,M=06
//FRUN07 EXEC FINDPROC,M=07
//FRUN08 EXEC FINDPROC,M=08
//FRUN09 EXEC FINDPROC,M=09
//FRUN10 EXEC FINDPROC,M=10
//FRUN11 EXEC FINDPROC,M=11
//FRUN12 EXEC FINDPROC,M=12
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
```

Appendix A.10

JCL/SAS Code: DBADJCL

```
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DBADJCL/DBADREAD IS COMPLETE
DBADJCL/DBADREAD IS COMPLETE.
```

## Appendix A.11

JCL/SAS Code: DBADREAD

```

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES OBS=MAX;
/* LAYOUT FOR DEC 2018*/

DATA TEMP.DBAD;
  INFILE IN TRUNCOVER;
  INPUT
  @0001 SEL_DATE YYMMDD8. /* DATE OF FILE SELECTION */
  @0010 CAN $9. /* CLAIM ACCOUNT NUMBER (SSN) */
  @0376 BIC $2. /* BENE IDENTIFICATION CODE */
  @0378 LAF $2. /* LEDGER ACCOUNT FILE */
  @0380 DOBDBAD YYMMDD8. /* DATE OF BIRTH */
  @1077 DP_OVERPAY_AMT 8.2 /* DP OVERPAYMENT $ (ADDED DAF19) */
  @1085 DP_RECOVERY_DT YYMMN6. /* DP RECOVERY DT (ADDED DAF19) */
  @1091 DP_OVERPAY_TYPE $2. /* DP OVERPAY TYPE (ADDED DAF19) */
  @1093 DP_WITHHOLD_AMT 4. /* DP WITHHOLD AMT $ (ADDED DAF19)*/
  @1097 PROTEST_IND $1. /* PROTEST INDICATOR (ADDED DAF19)*/
  @1113 BOAN $9. /* BENEFICIARY OWN ACCOUNT NUMBER */
  @3297 TOC $1. /* TYPE OF CLAIM */
  @3321 ZIP_DI $5. /* ZIP CODE */
  @3326 ZIP_DIADD $4. /* ZIP ADD ON CODE */
  ;

/* COMPUTE AGE AT THE TIME OF THE EXTRACT */
/* DAF PREFERRED AGE CALCULATION IS NOT USED IN THIS INSTANCE TO
PRESERVE CONTINUITY IN OUR BENE SELECTION ACROSS ITERATIONS OF
THE DAF*/
AGE = (SEL_DATE - DOBDBAD)/365.25;

/* SET THE BIRTHDATE */
%LET OLD_DOB = DOBDBAD;

/* INCLUDE THE PROGRAM TO CALCULATE THE FRADATE */
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(FRACALC)";

/* CALCULATE FRA */
FRA = (FRADATE-&OLD_DOB.)/365.25;

/* CONDITIONS FOR INCLUSION FOR DAF */
IF 18 <= AGE < FRA;
IF ((SUBSTR(BIC,1,1) = 'A' AND TOC IN ('5' '6')) OR
(SUBSTR(BIC,1,1) = 'C' AND TOC IN ('3' '4' '7')) OR
(SUBSTR(BIC,1,1) = 'W'));
IF UPCASE(LAF) IN ('AD' 'AS' 'A9' 'C' 'D' 'DP' 'DW' 'E' 'S' 'SD'
'S0' 'S1' 'C2' 'S2' 'S6' 'S8');

```



Appendix A.11

JCL/SAS Code: DBADREAD

```
/*AUXILIARY BENES ARE CHILDREN (BIC=C) AND WIDOWS (BIC=W) WHO  
RECEIVE SSDI BENEFITS BASED ON ANOTHER PRIMARY WAGE EARNER'S (BIC=A)  
RECORD. AS SUCH, SSA USES THE PRIMARY WAGE EARNER'S SSN (CAN)-NOT THE  
AUXILIARY BENE'S-SSN TO TRACK THE AUXILIARY'S BENEFIT PAYMENTS. SSA  
COMMONLY REFERS TO THE AUXILIARY BENE'S SSN AS BENEFICIARY'S OWN  
ACCOUNT NUMBER (BOAN). IF IT IS A PRIMARY WAGE EARNER'S RECORD, SET  
SSN=CAN. IF IT IS AN AUXILIARY BENE'S RECORD, SET SSN=BOAN*/
```

```
IF SUBSTR(BIC,1,1) = 'A' THEN SSN = CAN;  
ELSE SSN = BOAN;
```

```
IF SSN="" OR SSN="000000000" THEN DELETE;
```

```
DROP SEL_DATE DOBDBAD ;
```

```
RUN;
```

```
/* RETAIN ALL SSN/BIC COMBINATIONS SO THAT NO CAN'S WILL BE LOST */  
PROC SORT DATA=TEMP.DBAD OUT=OUT.DBAD NODUPKEY; BY SSN BIC; RUN;
```

```
PROC CONTENTS DATA=OUT.DBAD;  
RUN;
```

```
/* CHECK DATA */  
TITLE "CHECK OUTPUT DATASET (OBS=20)";  
PROC PRINT DATA=OUT.DBAD(OBS=20);  
RUN;  
TITLE;
```

## Appendix A.12

JCL/SAS Code: DBADCOMB

```

//#6266DBA JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=&SYSUID
//*****
/* PROJECT:   DAF 50214
/* DIRECTOR:  JODY SCHIMMEL HYDE
/* PROGRAM:   OPDR.TG.PRD.ETTW.#7429.DAF19.FINDER.PRDLIB(DBADCOMB)
/* PURPOSE:   COMBINE ALL THE DBAD FILES FOR 2019
/*           CASES IN DBAD FROM JAN 2019 - DEC 2019
/* PROGRAMMER: MIRIAM LOEWENBERG
/* UPDATED:   BRITTANY COLIP 01/31/2020 FOR DAF19
/* E-MAIL:    BCOLIP@MATHEMATICA-MPR.COM
/*           BRITTANY.COLIP@SSA.GOV
//*****
/*
// EXEC SAS9,Y=21,PARM='SYSPARM="'&SYSUID.2021"',
//   WORK='120000,60000'
/*
/*SEE INPUT FILES IN LIBNAME STATEMENTS BELOW
//OUT DD DSN=OPDR.TG.PRD.ETTW.&SYSUID..DAF&Y.P.CMBDBAD.SA.V1,
//   DISP=(MOD,CATLG,DELETE),
//   SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES MPRINT MACROGEN;
%LET Y4 = %SUBSTR(&SYSPARM,6,4);
%LET Y2 = %SUBSTR(&Y4.,3,2);
%LET ID = %SUBSTR(&SYSPARM,1,5);

%LET BEGYR=&Y4.;
%LET ENDYR=&Y4.;

%MACRO START;

/* INCLUDE YYMM MACRO PROGRAM */
%PUT "4-DIGIT - &Y4., 2-DIGIT - &Y2.";
%INCLUDE "OPDR.TG.PRD.ETTW.&ID..DAF&Y2..UTILITY.PRDLIB(YMMPMACR)";

/* INPUT FILES */
%DO SEG = 1 %TO 12;
  %LET SG = %SYSFUNC(PUTN(&SEG.,Z2.));
LIBNAME IN&SEG. "OPDR.TG.PRD.ETTW.&ID..DAF&Y2.P.DBAD&Y2.&SG..SA.V1"
  DISP=SHR;
%END;

```

Appendix A.12

JCL/SAS Code: DBADCOMB

```
/* COMBINE 12 DBAD FILES INTO ONE DBAD FILE */
DATA OUT.DBAD;
  MERGE
  %DO MM = 1 %TO &TOTMOS.;
    IN&MM..DBAD(RENAME=(ZIP_DI=ZIP_DI&&YY&MM
      ZIP_DIADD=ZIP_DIADD&&YY&MM))
  %END;
  ;
  BY SSN BIC;
RUN;

PROC CONTENTS DATA = OUT.DBAD;
RUN;

%MEND START;

%START;
//*****
/* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DBADCOMB IS COMPLETE
```

## Appendix A.13

JCL/SAS Code: CERJCL

```
//#6266CER JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF21.FINDER.PRDLIB(CERJCL)
//* PURPOSE:   READ IN MONTHLY CER DATA - JCL CALLS CERREAD
//* PROGRAMMER: BRITTANY COLIP
//* UPDATED:   2/15/2022 SERGE LUKASHANETS FOR DAF21
//* PHONE:     --
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//CERPROC PROC
//FIND   EXEC SAS9,
//      WORK='120000,60000',
//      PARM='SYSPARM="2021&M"'
//*
//IN    DD DSN=MTOSSI.CER100.FIELD.D21&M.,DISP=SHR
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.CER21&M..SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.FINDER.PRDLIB(CERREAD),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//FRUN01 EXEC CERPROC,M=01
//FRUN02 EXEC CERPROC,M=02
//FRUN03 EXEC CERPROC,M=03
//FRUN04 EXEC CERPROC,M=04
//FRUN05 EXEC CERPROC,M=05
//FRUN06 EXEC CERPROC,M=06
//FRUN07 EXEC CERPROC,M=07
//FRUN08 EXEC CERPROC,M=08
//FRUN09 EXEC CERPROC,M=09
//FRUN10 EXEC CERPROC,M=10
//FRUN11 EXEC CERPROC,M=11
//FRUN12 EXEC CERPROC,M=12
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
```

Appendix A.13

JCL/SAS Code: CERJCL

```
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
DATA  
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
SUBJECT: CERJCL/CERREAD IS COMPLETE
```

## Appendix A.14

JCL/SAS Code: CERREAD

```
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES OBS=MAX;

%LET SEG=%SUBSTR(&SYSPARM.,5,2);

%MACRO READIN;
DATA CER;
INFILE IN TRUNCOVER;
INPUT
@0004 MFT      $2.    /* MASTER FILE TYPE */
@0015 SSN      $9.    /* SSN SOCIAL SECURITY NUMBER */
@;
  IF SSN GT '000000000'; /*DAF13 MOVED TO INPUT PART OF CODE*/

INPUT
@0108 RCD_EST  ?? YYMMDD8. /* RCD-EST-JD RECORD ESTABLISHMENT DATE*/
@0116 DOBCER   ?? YYMMDD8. /* BIRTH-JD DATE OF BIRTH */
@0134 RSEX     $1.    /* SEX */
@0204 DENCDE   $3.    /* DENIAL CODE */
@0223 ERP_IND  $1.    /* ECONOMIC RECOVERY PAYMENT
INDICATOR */
@0326 RDX      $4.    /* PRIMARY DISABILITY DIAG. CODE*/
/*THE VARIABLE PDSCC THAT BEGINS AT POSITION 343 IS LISTED AS 6 CHARS
LONG AND CONTAINS 2 DIGIT STATE, 3 DIGIT COUNTY, AND 1 DIGIT DO.
WE READ ONLY THE FIRST 5 POSITIONS AS TWO VARIABLES PDS2 AND PDS3.*/
@0343 PDS2     $2.    /* PDSCC RESIDENCE STATE CODE */
@0345 PDS3     $3.    /* PDSCC RESIDENCE COUNTY CODE */
@0349 PDZIP1_5 $5.    /* PDZIP1-5 PAYEE'S ZIP CODE */
@0354 PDZIP6_9 $4.    /* PDZIP6-9 PAYEE'S ZIP CODE SUFFIX */
@0361 IET4     $1.    /* EARNED INC. TYPE (SELF EMPLOYMENT)*/
@0368 IEA_NESE 6.2    /* EARNED INC. AMT. SELF EMPLOYMENT */
@0382 IEA_WAGE 6.2    /* EARNED INC. AMT. (WAGES) */
@0396 IEA_BLIND 6.2   /* EARNED INC. AMT. (BLIND W E)*/
@0410 IEA_PASS 6.2    /* EARNED INC. AMT. (PASS)*/
@0424 IEA_IRWE 6.2    /* EARNED INC. AMT. (IRWE)*/
@0438 IEA_LOSS 6.2    /* EARNED INC. AMT. (NET LOSS)*/
@;
POS = 445;
ARRAY UT (6) $ T1-T6; /* UNEARNED INCOME TYPE CODE */
ARRAY UA (6) UN1-UN6; /* UNEARNED INCOME AMOUNT */
DO I = 1 TO 6;
INPUT
@POS UT(I) $2.
@POS+8 UA(I) 6.2
@;
```

## Appendix A.14

JCL/SAS Code: CERREAD

```
POS = POS + 27;
END;

INPUT
@0691 AAZIP1_5 $5. /* AAZIP1-5 APPLICANT'S ADDRESS ZIP*/
@0696 AAZIP6_9 $4. /* AAZIP6-9 APPLICANT'S ZIP CODE SUF*/

%IF &SEG. = 12 %THEN %DO;
  @0767 STATECD_ENDYR $2. /*PAYMENT STATE CODE ADDED FOR*/
%END; /*ONLY FOR DECEMBER OF THE YEAR*/

@0794 CPST $3. /* PSTAT-CURR-PAYMENT STATUS CODE*/
@0811 MEDC $1. /* MEDIC MEDICAID ELIGIBILITY CODE*/
@0853 SEL_DATE ?? YYMMDD8. /* DATE OF SELECTION */
@1003 CIT $1. /* ALIEN ALIEN REFUGE INDICATOR */
@1050 IEA_SEIE 6.2 /* STUDENT EARNED INCOME EXCLUSION
AMOUNT IETYP=B (IEAMT) */
;

DO I = 1 TO 6;
  IF UT(I) = 'A' THEN IUA_SSDI = UA(I); /* TYPE II */
  ELSE IF UT(I) = 'F' THEN IUA_TANF = UA(I); /* TANF */
  ELSE IF UT(I) = 'Q' THEN IUA_WC = UA(I); /* WORKERS COMP */
END;

/* CREATE AGE AT TIME OF EXTRACTION VARIABLE */
/*DAF PREFERRED AGE CALCULATION IS NOT USED IN THIS INSTANCE TO
PRESERVE CONTINUITY IN OUR BENE SELECTION ACROSS ITERATIONS OF
THE DAF*/

AGE = (SEL_DATE - DOBCER)/365.25;

/* SET THE BIRTHDATE */
%LET OLD_DOB = DOBCER;

/* INCLUDE THE PROGRAM TO CALCULATE THE FRADATE */
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(FRACALC)";

/* CALCULATE FRA */
FRA = (FRADATE-&OLD_DOB.)/365.25;

/* CONDITIONS FOR INCLUSION FOR DAF */
IF AGE < FRA;
IF (UPCASE(CPST) IN ('C01' 'E01' 'E02' 'M01' 'M02'
'P01' 'S05' 'S06' 'S07' 'S08' 'S09' 'S21' 'T30'
```

## Appendix A.14

JCL/SAS Code: CERREAD

```
'T32' 'T33')) OR
(DENCDE = ' ' AND /* NEW IN 2004 TO ELIM DENIALS FOR N CODES */
UPCASE(CPST) IN ('N01' 'N02' 'N04' 'N05' 'N06'
'N09' 'N20' 'N22' 'N33' 'N44' 'N54'));
IF UPCASE(MFT) IN ('DI' 'BI' 'DC' 'BC' 'DS' 'BS');

/*CHECK AAZIP1_5 IF IT IS A VALID ZIP CODES-00000 OR 0000A ARE NOT VALID
NOTDIGIT FUNCTION RETURNS THE POSITION OF THE FIRST NON-DIGIT VALUE
IN THE CHAR STRING*/
IF NOTDIGIT(AAZIP1_5)=0 AND AAZIP1_5 NE '00000' THEN DO;
RESZIP = AAZIP1_5;
RZIPS = AAZIP6_9;
END;
ELSE DO;
RESZIP = PDZIP1_5;
RZIPS = PDZIP6_9;
END;

FORMAT
IEA_BLIND IEA_PASS IEA_LOSS IEA_NESE IEA_IRWE
IEA_WAGE IEA_SEIE IUA_SSDI IUA_TANF IUA_WC 11.2
DOBCER RCD_EST YMMDD10.;
;

DROP SEL_DATE I POS UN1-UN6 T1-T6;

RUN;

PROC SORT DATA=CER OUT=TEMP.CER; BY SSN RCD_EST; RUN;

DATA OUT.CER;
SET TEMP.CER;
BY SSN;
IF LAST.SSN; /*KEEP THE MOST RECENT RECORD ESTABLISHMENT DATE*/
RUN;

PROC CONTENTS DATA=OUT.CER; RUN;

/* CHECK OUTPUT */
TITLE "CHECK OUTPUT DATASET (OBS=10)";
PROC PRINT DATA=OUT.CER(OBS=10);
RUN;
TITLE;

%MEND;
```



Appendix A.14

JCL/SAS Code: CERREAD

```
%READIN;
```

Appendix A.15

JCL/SAS Code: CERCOMB

```

//#6266CRM JOB (11710000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF21.FINDER.PRDLIB(CERCOMB)
//* PURPOSE:   COMBINE THE MONTHLY CER EXTRACTS AND OUTPUT ONE
//*            RECORD PER BENE
//* UPDATED:   SERGE LUKASHANETS 02/16/2021 FOR DAF21
//* EMAIL:     SLUKASHANETS@MATHEMATICA-MPR.COM
//*
//*****
//*
// SET REG='64M'
//JS010 EXEC SAS9,
//      WORK='200000,100000',          * SPACE IN BLKS *
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
// *SEE INPUT FILES IN LIBNAME STATEMENTS BELOW
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.COMBCER.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES MPRINT MACROGEN OBS=MAX;

%LET BEGYR=2021; /* CHANGE AS NEEDED */
%LET ENDYR=2021; /* CHANGE AS NEEDED */

%MACRO START;

/* INCLUDE YMM MACRO PROGRAM */
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

/* INPUT FILES */
%DO SEG = 1 %TO 9;
LIBNAME IN&SEG. "OPDR.TG.PRD.ETTW.#6266.DAF21P.CER210&SEG..SA.V1"
      DISP=SHR;
%END;
%DO SEG = 10 %TO 12;
LIBNAME IN&SEG. "OPDR.TG.PRD.ETTW.#6266.DAF21P.CER21&SEG..SA.V1"
      DISP=SHR;
%END;

```

## Appendix A.15

JCL/SAS Code: CERCOMB

```
/* COMBINE 12 CER FILES INTO ONE CER FILE */
DATA OUT.CER;
MERGE
%DO MM=1 %TO &TOTMOS.;
  IN&MM..CER(RENAME=(
CIT=   CIT&&YY&MM
RDX=   RDX&&YY&MM
IET4=  IET4&&YY&MM
IEA_BLIND= IEA_BLIND&&YY&MM
IEA_PASS= IEA_PASS&&YY&MM
IEA_LOSS= IEA_LOSS&&YY&MM
IEA_NESE= IEA_NESE&&YY&MM
IEA_IRWE= IEA_IRWE&&YY&MM
IEA_WAGE= IEA_WAGE&&YY&MM
IUA_SSDI= IUA_SSDI&&YY&MM
IUA_TANF= IUA_TANF&&YY&MM
IUA_WC=  IUA_WC&&YY&MM
MEDC=   MEDC&&YY&MM
PDS2=   PDS2&&YY&MM
PDS3=   PDS3&&YY&MM
CPST=   CPST&&YY&MM
RESZIP = RZIP&&YY&MM
RZIPS =  RZPS&&YY&MM
ERP_IND= ERP_IND&&YY&MM
IEA_SEIE= IEA_SEIE&&YY&MM
  ))
%END;
;
BY SSN;
RUN;

PROC CONTENTS DATA=OUT.CER; RUN;

%MEND START;

%START;
//*****
/* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
```

Appendix A.15

JCL/SAS Code: CERCOMB

```
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
DATA  
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
SUBJECT: CERCOMB IS COMPLETE
```

## Appendix A.16

JCL/SAS Code: MAKEFIND

```
//#6266MKF JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=&SYSUID
// SET REG='64M'
//JOB01 EXEC SAS9,Y=21,PY=20,ID=&SYSUID,
//  WORK='180000,90000',
//  REGION=&REG,
//  PARM='MEMSIZE=&REG'
//*****
//* PROJECT: DAF 50214
//* DIRECTOR: JODY SCHIMMEL HYDE
//* PROGRAM: OPDR.TG.PRD.ETTW.#6266.DAF20.FINDER.PRDLIB(MAKEFIND)
//* PURPOSE: CREATE DI, SSI, MASTER, AND 831/833 FIDNER FILES
//* UPDATE: SERGE LUKASHANETS 03/03/21 FOR DAF20
//* E-MAIL  SLUKASHANETS@MATHEMATICA-MPR.COM
//*        SIARHEI.LUKASHANETS@SSA.GOV
//* NOTE:   IN DAF15 LINKFIND WAS COMBINED INTO MAKEFIND
//* NOTE:   INCLUDING THE HISTORICAL 65-FRA & SSI KIDS IN DAF16
//* NOTE:   INCLUDING SSI KIDS FROM 1996-2004 IN DAF17
//*****
//*
//* INPUT FILES
//* DAF21 DBAD & CER100 FILES
//DBAD DD DSN=OPDR.TG.PRD.ETTW.&ID..DAF&Y.P.CMBDBAD.SA.V1,DISP=SHR
//CER  DD DSN=OPDR.TG.PRD.ETTW.&ID..DAF&Y.P.COMBCER.SA.V1,DISP=SHR
//* DAF20 FINDERS
//PRELDI DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF&PY.P.LINKDI.SA.V1,DISP=SHR
//PRESSI DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF&PY.P.SSIFIND.SA.V1,DISP=SHR
//* OUTPUT FILES
//LDI  DD DSN=OPDR.TG.PRD.ETTW.&ID..DAF&Y.P.LINKDI.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     SPACE=(CYL,(3000,3000),RLSE)
//DI   DD DSN=OPDR.TG.PRD.ETTW.&ID..DAF&Y.P.DIFIND.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     SPACE=(CYL,(3000,3000),RLSE)
//DIFL DD DSN=OPDR.TG.PRD.ETTW.&ID..DAF&Y.P.DIFIND.FL.V1,
//     DISP=(MOD,CATLG,DELETE),
//     SPACE=(CYL,(1000,100),RLSE),
//     DCB=(RECFM=FB,LRECL=80,DSORG=PS)
//SSI  DD DSN=OPDR.TG.PRD.ETTW.&ID..DAF&Y.P.SSIFIND.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     SPACE=(CYL,(3000,3000),RLSE)
//SSIFL DD DSN=OPDR.TG.PRD.ETTW.&ID..DAF&Y.P.SSIFIND.FL.V1,
//     DISP=(MOD,CATLG,DELETE),
//     SPACE=(CYL,(1000,100),RLSE),
```

## Appendix A.16

JCL/SAS Code: MAKEFIND

```

//      DCB=(RECFM=FB,LRECL=80,DSORG=PS)
//MAST DD DSN=OPDR.TG.PRD.ETTW.&ID..DAF&Y.P.MASTFIND.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//MASTFL DD DSN=OPDR.TG.PRD.ETTW.&ID..DAF&Y.P.MASTFIND.FL.V1,
//      DISP=(MOD,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE),
//      DCB=(RECFM=FB,LRECL=80,DSORG=PS)
//SSR DD DSN=OPDR.TG.PRD.ETTW.&ID..DAF&Y.P.FIND831.SA.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TEMP DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES OBS=MAX;

/*****
/*      CREATE LINKDI (NOT SUBMITTED TO SSA)      */
/*****
/* COMBINE NEW DI LINKING FILE WITH OLD DI LINK      */
/* OUTPUT FILE WILL BE THE LINK FOR KEEPING MBR RECORDS      */
/* RETURNED FROM THE FINDER PROCESS      */

DATA LDI.LINKDI;
MERGE PRELDI.LINKDI
      DBAD.DBAD(IN=DBAD KEEP=SSN CAN BIC);
BY SSN BIC;

/* GET RID OF BLANK CAN'S */
IF CAN NE ' ';
RUN;

/*****
/*      CREATE DI FINDER FILE (SUBMITTED TO SSA)      */
/*****

PROC SORT DATA=LDI.LINKDI NODUPKEY
      OUT=DI.DIFIND;
BY CAN;
RUN;

DATA NULL;
SET DI.DIFIND;
BY CAN;

```

Appendix A.16

JCL/SAS Code: MAKEFIND

```

FILE DIFL;
PUT
@1 CAN $9.
;
RUN;

/*****
/*      CREATE SSI FINDER FILE (NOT SUBMITTED TO SSA)   */
/*****
/* COMBINE NEW CER100 FILE WITH PREVIOUS SSIFIND FILE */
DATA SSI.SSIFIND;
  MERGE PRESSI.SSIFIND(IN=PRVSSI)
      CER.CER(IN=CER KEEP=SSN);
  BY SSN;
RUN;

/*****
/*      CREATE SSI FINDER FILE (SUBMITTED TO SSA)       */
/*****
DATA NULL;
  SET SSI.SSIFIND;
  BY SSN;
  FILE SSIFL;
  PUT
  @1 SSN $9.
;
RUN;

/*****
/*      CREATE MASTER FINDER FILE (SUBMITTED TO SSA)   */
/*****

/* RESORT DI DATA DE-DUPING ON SSN FOR MASTER FILE */
PROC SORT DATA=LDI.LINKDI NODUPKEY
  OUT=TEMP.DIFIND;
  BY SSN;
RUN;

/* OUTPUT MASTER FLAT DATA AND SAS DATA SET */
/* FLAT FILE IS SENT TO SSA STAFF AS FINDER - SAS DATA SET IS
  USED TO SCREEN THE RETURNED RECORDS */
DATA MAST.MASTFIND;
  MERGE SSI.SSIFIND(KEEP=SSN)
      TEMP.DIFIND(KEEP=SSN);
  BY SSN;

```

Appendix A.16

JCL/SAS Code: MAKEFIND

RUN;

```
DATA NULL;
  SET MAST.MASTFIND;
  BY SSN;
  FILE MASTFL;
  PUT
  @1 SSN $9.
  ;
RUN;
```

```
/******
/*      CREATE 831 FINDER FILE (NOT SUBMITTED TO SSA)      */
/******
```

```
/* DIFIND CONTAINS CANS FROM THE DBAD AND DEMO DATA DE-DUPLICATED ON
  CAN. IN THE LINKDI FILE THE BIC DISTINGUISHES BETWEEN THEM.
  SSIFIND CONTAINS SSNS FROM NEW CER AND DEMO.
  THE "FIND" FILES ARE FLAT FILES SENT TO SSA AS FINDERS.
  831-33 FILE HAS THE DUPLICATED CAN FOR MBR AND THE PAN FOR SSR.
  TO LINK BACK WE NEED TO MATCH THE CAN AND THE BIC FOR MBR RECORDS
  AND THE PAN FOR SSR. ALL SSR RECORDS HAVE BIC = A IN 831.
  CHILDREN ARE REGISTERED AS PRIMARIES ON SSR-831.
  THE LINK SHOULD FIND ALL THE RECORDS WE NEED. */
```

```
DATA FIND831;
  SET DI.DIFIND(KEEP=CAN RENAME=(CAN=SSN))
  SSI.SSIFIND(KEEP=SSN);
  BY SSN;
RUN;
```

```
PROC SORT NODUPKEY DATA=FIND831 OUT=SSR.FIND831; BY SSN; RUN;
```

```
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
```



Appendix A.16

JCL/SAS Code: MAKEFIND

FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
SUBJECT: MAKEFIND IS COMPLETE

Appendix A.17

JCL/SAS Code: COPYNUM1

```
//#7429CPH JOB (12510000,T715,,SAS,,ITC9FL),COLIP-BRITTANY,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#7429
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETW.#7429.FINDER.PRDLIB(COPYNUM1) |
//*  *-----*
//*
//*  1. COPY NUMIDENT FINDER RETURN TO DAF SPACE
//*
//*****
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETW.DAF21.NUM1.Y22M0330,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETW.#7429.DAF21.NUM1.FL.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
```

## Appendix A.18

JCL/SAS Code: COPYMBR

```
//#7429CPY JOB (12510000,T715,,SAS,,ITC9FL),COLIP-BRITTANY,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#7429
//*****
//* PROJECT:   DAF 51390
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#7429.DAF21.FINDER.PRDLIB(COPYMBR)
//* DESCRIPTION: COPY THE MBR AND PHUS FILES TO DAF SPACE
//*****
//*
//* COPYING FILES
//COPYMBR PROC
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DAF21.&MP.OUT.F&N..R220329,
//      DISP=SHR,UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.&MP..F&N..R220329,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//      PEND
//MBR1 EXEC COPYMBR,MP=MBR,N=1
//PHUS1 EXEC COPYMBR,MP=PHUS,N=1
//MBR2 EXEC COPYMBR,MP=MBR,N=2
//PHUS2 EXEC COPYMBR,MP=PHUS,N=2
//MBR3 EXEC COPYMBR,MP=MBR,N=3
//PHUS3 EXEC COPYMBR,MP=PHUS,N=3
//MBR4 EXEC COPYMBR,MP=MBR,N=4
//PHUS4 EXEC COPYMBR,MP=PHUS,N=4
//MBR5 EXEC COPYMBR,MP=MBR,N=5
//PHUS5 EXEC COPYMBR,MP=PHUS,N=5
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<BCOLIP@MATHEMATICA-MPR.COM>
RCPT TO:<BCOLIP@MATHEMATICA-MPR.COM>
DATA
FROM: <BCOLIP@MATHEMATICA-MPR.COM>
TO: <BCOLIP@MATHEMATICA-MPR.COM>
```

Appendix A.18

JCL/SAS Code: COPYMBR

SUBJECT: DAF21 MBR AND PHUS COPY PROCESSING DONE

## Appendix A.19

JCL/SAS Code: COPYSSR

```
//#7429CPS JOB (12510000,T715,,SAS,,ITC9FL),BRITTANY-COLIP,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#7429
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.#7429.FINDER.PRDLIB(COPYSSR)|
//* *-----*
//*
//* 1. COPY SSR FILES FROM TAPE TO TAPE
//*
//*****
//*
//* WE RECEIVED SSR EARNINGS FILES ON 4/25/2022
//STEPA EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=MTOSSI.MISC.EARN.FRERN21.D2203,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.SSRERN.FL.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//* WE RECEIVED SSR LONGITUDINAL FILES ON 4/25/2022
//* TWO VERSIONS WERE RECEIVED, SO COPYING BOTH
//* VERSION 1 IS A FULL FILE
//STEPA EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=MTOSSI.MISC.LONGTP.FRDAF21.D2203,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.SSRLONG.FL.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//* VERSION 2 IS A TRUNCATED FILE
//STEPB EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=MTOSSI.MISC.LONGFLAT.DAF21.D202203,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.SSRLONG.FL.V2,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
```

## Appendix A.20

JCL/SAS Code: DBAEXTJ

```
//#6266DBA JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#7429
//*****
//* PROJECT:   DAF 50214
//* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.FINDER.PRDLIB(DBADJCL)
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   3/24/2022 SERGE LUKASHANETS FOR DAF21
//* PHONE:
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//FINDPROC PROC
//FIND   EXEC SAS9,
//      WORK='120000,60000'
//*
//IN    DD DSN=MTOSI.T2.DBADMBR.D21&M,DISP=SHR
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,(VOL=,,,5)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DBD21&M.E.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.CERDBAD.PRDLIB(DBAEXT),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//FRUN01 EXEC FINDPROC,M=01
//FRUN02 EXEC FINDPROC,M=02
//FRUN03 EXEC FINDPROC,M=03
//FRUN04 EXEC FINDPROC,M=04
//FRUN05 EXEC FINDPROC,M=05
//FRUN06 EXEC FINDPROC,M=06
//FRUN07 EXEC FINDPROC,M=07
//FRUN08 EXEC FINDPROC,M=08
//FRUN09 EXEC FINDPROC,M=09
//FRUN10 EXEC FINDPROC,M=10
//FRUN11 EXEC FINDPROC,M=11
//FRUN12 EXEC FINDPROC,M=12
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
```

Appendix A.20

JCL/SAS Code: DBAEXTJ

```
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DBADJCL/DBADREAD IS COMPLETE
DBADJCL/DBADREAD IS COMPLETE.
```

## Appendix A.21

JCL/SAS Code: DBADEXT

```
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES OBS=MAX;
/* LAYOUT FOR DEC 2018*/
```

```
DATA TEMP.DBAD;
  INFILE IN TRUNCOVER;
  INPUT
  @0001 SEL_DATE YYMMDD8. /* DATE OF FILE SELECTION */
  @0010 CAN $9. /* CLAIM ACCOUNT NUMBER (SSN) */
  @0376 BIC $2. /* BENE IDENTIFICATION CODE */
  @0378 LAF $2. /* LEDGER ACCOUNT FILE */
  @0380 DOBDBAD YYMMDD8. /* DATE OF BIRTH */
  @1077 DP_OVERPAY_AMT 8.2 /* DP OVERPAYMENT $ (ADDED DAF19) */
  @1085 DP_RECOVERY_DT YYMMN6. /* DP RECOVERY DT (ADDED DAF19) */
  @1091 DP_OVERPAY_TYPE $2. /* DP OVERPAY TYPE (ADDED DAF19) */
  @1093 DP_WITHHOLD_AMT 4. /* DP WITHHOLD AMT $ (ADDED DAF19)*/
  @1097 PROTEST_IND $1. /* PROTEST INDICATOR (ADDED DAF19)*/
  @1113 BOAN $9. /* BENEFICIARY OWN ACCOUNT NUMBER */
  @3297 TOC $1. /* TYPE OF CLAIM */
  @3321 ZIP_DI $5. /* ZIP CODE */
  @3326 ZIP_DIADD $4. /* ZIP ADD ON CODE */
  ;
```

```
/*AUXILIARY BENES ARE CHILDREN (BIC=C) AND WIDOWS (BIC=W) WHO
RECEIVE SSDI BENEFITS BASED ON ANOTHER PRIMARY WAGE EARNER'S (BIC=A)
RECORD. AS SUCH, SSA USES THE PRIMARY WAGE EARNER'S SSN (CAN)-NOT THE
AUXILIARY BENE'S-SSN TO TRACK THE AUXILIARY'S BENEFIT PAYMENTS. SSA
COMMONLY REFERS TO THE AUXILIARY BENE'S SSN AS BENEFICIARY'S OWN
ACCOUNT NUMBER (BOAN). IF IT IS A PRIMARY WAGE EARNER'S RECORD, SET
SSN=CAN. IF IT IS AN AUXILIARY BENE'S RECORD, SET SSN=BOAN*/
```

```
IF SUBSTR(BIC,1,1) = 'A' THEN SSN = CAN;
ELSE SSN = BOAN;
```

```
IF SSN="" OR SSN="000000000" THEN DELETE;
```

```
DROP SEL_DATE DOBDBAD ;
```

```
RUN;
```

```
/* RETAIN ALL SSN/BIC COMBINATIONS SO THAT NO CAN'S WILL BE LOST */
PROC SORT DATA=TEMP.DBAD OUT=OUT.DBAD NODUPKEY; BY SSN BIC; RUN;
```

```
PROC CONTENTS DATA=OUT.DBAD;
RUN;
```



Appendix A.21

JCL/SAS Code: DBADEXT

```
/* CHECK DATA */  
TITLE "CHECK OUTPUT DATASET (OBS=20)";  
PROC PRINT DATA=OUT.DBAD(OBS=20);  
RUN;  
TITLE;
```

## Appendix A.22

JCL/SAS Code: DBADEXTC

```

//#6266DBA JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=&SYSUID
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#7429.DAF19.FINDER.PRDLIB(DBADCOMB)
//* PURPOSE:   COMBINE ALL THE DBAD FILES FOR 2019
//*           CASES IN DBAD FROM JAN 2019 - DEC 2019
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   BRITTANY COLIP 01/31/2020 FOR DAF19
//* E-MAIL:    BCOLIP@MATHEMATICA-MPR.COM
//*           BRITTANY.COLIP@SSA.GOV
//*****
//*
// EXEC SAS9,Y=21,PARM='SYSPARM="&SYSUID.2021"',
//   WORK='120000,60000'
//*
//**SEE INPUT FILES IN LIBNAME STATEMENTS BELOW
//FIN  DD DSN=OPDR.TG.PRD.ETTW.&SYSUID..DAF&Y.P.CMBDBAD.SA.V1,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.&SYSUID..DAF&Y.P.CMBDBDE.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO
//SYSIN DD *

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES MPRINT MACROGEN;
%LET Y4 = %SUBSTR(&SYSPARM,6,4);
%LET Y2 = %SUBSTR(&Y4.,3,2);
%LET ID = %SUBSTR(&SYSPARM,1,5);

%LET BEGYR=&Y4.;
%LET ENDYR=&Y4.;

%MACRO START;

/* INCLUDE YMM MACRO PROGRAM */
%PUT "4-DIGIT - &Y4., 2-DIGIT - &Y2.";
%INCLUDE "OPDR.TG.PRD.ETTW.&ID..DAF&Y2..UTILITY.PRDLIB(YMMMCR)";

/* INPUT FILES */
%DO SEG = 1 %TO 12;
  %LET SG = %SYSFUNC(PUTN(&SEG.,Z2.));
LIBNAME IN&SEG. "OPDR.TG.PRD.ETTW.&ID..DAF&Y2.P.DBD&Y2.&SG.E.SA.V1"
        DISP=SHR;
%END;

```

Appendix A.22

JCL/SAS Code: DBADEXTC

```

/* COMBINE 12 DBAD FILES INTO ONE DBAD FILE */
DATA OUT.DBAD;
  MERGE FIN.DBAD (IN=A KEEP=SSN BIC)
  %DO MM = 1 %TO &TOTMOS.;
    IN&MM..DBAD(RENAME=(ZIP_DI=ZIP_DI&&YY&MM
      ZIP_DIADD=ZIP_DIADD&&YY&MM))
  %END;
;
BY SSN BIC;
IF A;
RUN;

PROC CONTENTS DATA = OUT.DBAD;
RUN;

%MEND START;

%START;
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DBADCOMB IS COMPLETE

```

## Appendix A.23

JCL/SAS Code: CEREXTJ

```
//#6266CER JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF21.FINDER.PRDLIB(CERJCL)
//* PURPOSE:   READ IN MONTHLY CER DATA - JCL CALLS CERREAD
//* PROGRAMMER: BRITTANY COLIP
//* UPDATED:   3/24/2022 SERGE LUKASHANETS FOR DAF21
//* PHONE:     --
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//CERPROC PROC
//FIND   EXEC SAS9,
//      WORK='120000,60000',
//      PARM='SYSPARM="2021&M"'
//*
//IN    DD DSN=MTOSSI.CER100.FIELD.D21&M.,DISP=SHR
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.CER21&M.E.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.CERDBAD.PRDLIB(CEREXT),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//FRUN01 EXEC CERPROC,M=01
//FRUN02 EXEC CERPROC,M=02
//FRUN03 EXEC CERPROC,M=03
//FRUN04 EXEC CERPROC,M=04
//FRUN05 EXEC CERPROC,M=05
//FRUN06 EXEC CERPROC,M=06
//FRUN07 EXEC CERPROC,M=07
//FRUN08 EXEC CERPROC,M=08
//FRUN09 EXEC CERPROC,M=09
//FRUN10 EXEC CERPROC,M=10
//FRUN11 EXEC CERPROC,M=11
//FRUN12 EXEC CERPROC,M=12
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
```

Appendix A.23

JCL/SAS Code: CEREXTJ

```
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
DATA  
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
SUBJECT: CERJCL/CERREAD IS COMPLETE
```

## Appendix A.24

JCL/SAS Code: CEREXT

```
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES OBS=MAX;

%LET SEG=%SUBSTR(&SYSPARM.,5,2);

%MACRO READIN;
DATA CER;
INFILE IN TRUNCOVER;
INPUT
@0004 MFT      $2.    /* MASTER FILE TYPE */
@0015 SSN      $9.    /* SSN SOCIAL SECURITY NUMBER */
@;
  IF SSN GT '000000000'; /*DAF13 MOVED TO INPUT PART OF CODE*/

INPUT
@0108 RCD_EST  ?? YYMMDD8. /* RCD-EST-JD RECORD ESTABLISHMENT DATE*/
@0116 DOBCER  ?? YYMMDD8. /* BIRTH-JD DATE OF BIRTH */
@0134 RSEX    $1.    /* SEX */
@0204 DENCDE  $3.    /* DENIAL CODE */
@0223 ERP_IND $1.    /* ECONOMIC RECOVERY PAYMENT
INDICATOR */
@0326 RDX     $4.    /* PRIMARY DISABILITY DIAG. CODE*/
/*THE VARIABLE PDSCC THAT BEGINS AT POSITION 343 IS LISTED AS 6 CHARS
LONG AND CONTAINS 2 DIGIT STATE, 3 DIGIT COUNTY, AND 1 DIGIT DO.
WE READ ONLY THE FIRST 5 POSITIONS AS TWO VARIABLES PDS2 AND PDS3.*/
@0343 PDS2    $2.    /* PDSCC RESIDENCE STATE CODE */
@0345 PDS3    $3.    /* PDSCC RESIDENCE COUNTY CODE */
@0349 PDZIP1_5 $5.    /* PDZIP1-5 PAYEE'S ZIP CODE */
@0354 PDZIP6_9 $4.    /* PDZIP6-9 PAYEE'S ZIP CODE SUFFIX */
@0361 IET4    $1.    /* EARNED INC. TYPE (SELF EMPLOYMENT)*/
@0368 IEA_NESE 6.2    /* EARNED INC. AMT. SELF EMPLOYMENT */
@0382 IEA_WAGE 6.2    /* EARNED INC. AMT. (WAGES) */
@0396 IEA_BLIND 6.2    /* EARNED INC. AMT. (BLIND W E)*/
@0410 IEA_PASS 6.2    /* EARNED INC. AMT. (PASS)*/
@0424 IEA_IRWE 6.2    /* EARNED INC. AMT. (IRWE)*/
@0438 IEA_LOSS 6.2    /* EARNED INC. AMT. (NET LOSS)*/
@;
POS = 445;
ARRAY UT (6) $ T1-T6; /* UNEARNED INCOME TYPE CODE */
ARRAY UA (6) UN1-UN6; /* UNEARNED INCOME AMOUNT */
DO I = 1 TO 6;
INPUT
@POS UT(I) $2.
@POS+8 UA(I) 6.2
@;
```

## Appendix A.24

JCL/SAS Code: CEREXT

```

POS = POS + 27;
END;

INPUT
@0691 AAZIP1_5 $5. /* AAZIP1-5 APPLICANT'S ADDRESS ZIP*/
@0696 AAZIP6_9 $4. /* AAZIP6-9 APPLICANT'S ZIP CODE SUF*/

%IF &SEG. = 12 %THEN %DO;
  @0767 STATECD_ENDYR $2. /*PAYMENT STATE CODE ADDED FOR*/
%END; /*ONLY FOR DECEMBER OF THE YEAR*/

@0794 CPST $3. /* PSTAT-CURR-PAYMENT STATUS CODE*/
@0811 MEDC $1. /* MEDIC MEDICAID ELIGIBILITY CODE*/
@0853 SEL_DATE ?? YYMMDD8. /* DATE OF SELECTION */
@1003 CIT $1. /* ALIEN ALIEN REFUGE INDICATOR */
@1050 IEA_SEIE 6.2 /* STUDENT EARNED INCOME EXCLUSION
AMOUNT IETYP=B (IEAMT) */
;

DO I = 1 TO 6;
  IF UT(I) = 'A' THEN IUA_SSDI = UA(I); /* TYPE II */
  ELSE IF UT(I) = 'F' THEN IUA_TANF = UA(I); /* TANF */
  ELSE IF UT(I) = 'Q' THEN IUA_WC = UA(I); /* WORKERS COMP */
END;

/*CHECK AAZIP1_5 IF IT IS A VALID ZIP CODES-00000 OR 0000A ARE NOT VALID
NOTDIGIT FUNCTION RETURNS THE POSITION OF THE FIRST NON-DIGIT VALUE
IN THE CHAR STRING*/
IF NOTDIGIT(AAZIP1_5)=0 AND AAZIP1_5 NE '00000' THEN DO;
  RESZIP = AAZIP1_5;
  RZIPS = AAZIP6_9;
END;
ELSE DO;
  RESZIP = PDZIP1_5;
  RZIPS = PDZIP6_9;
END;

FORMAT
  IEA_BLIND IEA_PASS IEA_LOSS IEA_NESE IEA_IRWE
  IEA_WAGE IEA_SEIE IUA_SSDI IUA_TANF IUA_WC 11.2
  DOBCER RCD_EST YYMMDD10.;
;

DROP SEL_DATE I POS UN1-UN6 T1-T6;

```

Appendix A.24

JCL/SAS Code: CEREXT

```
RUN;

PROC SORT DATA=CER OUT=TEMP.CER; BY SSN RCD_EST; RUN;

DATA OUT.CER;
  SET TEMP.CER;
  BY SSN;
  IF LAST.SSN; /*KEEP THE MOST RECENT RECORD ESTABLISHMENT DATE*/
RUN;

PROC CONTENTS DATA=OUT.CER; RUN;

/* CHECK OUTPUT */
TITLE "CHECK OUTPUT DATASET (OBS=10)";
PROC PRINT DATA=OUT.CER(OBS=10);
RUN;
TITLE;

%MEND;
%READIN;
```



## Appendix A.25

JCL/SAS Code: CEREXTC

```
//#6266CRM JOB (11710000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF21.FINDER.PRDLIB(CERCOMB)
//* PURPOSE:   COMBINE THE MONTHLY CER EXTRACTS AND OUTPUT ONE
//*            RECORD PER BENE
//* UPDATED:   SERGE LUKASHANETS 04/07/2022 FOR DAF21
//* EMAIL:     SLUKASHANETS@MATHEMATICA-MPR.COM
//*
//*****
//*
// SET REG='64M'
//JS010 EXEC SAS9,
//      WORK='200000,100000',          * SPACE IN BLKS *
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
// *SEE INPUT FILES IN LIBNAME STATEMENTS BELOW
//FIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.COMBCER.SA.V1,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.COMBCERE.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO
//SYSIN DD *
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES MPRINT MACROGEN OBS=MAX;

%LET BEGYR=2021; /* CHANGE AS NEEDED */
%LET ENDYR=2021; /* CHANGE AS NEEDED */

%MACRO START;

/* INCLUDE YMM MACRO PROGRAM */
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

/* INPUT FILES */
%DO SEG = 1 %TO 9;
LIBNAME IN&SEG. "OPDR.TG.PRD.ETTW.#6266.DAF21P.CER210&SEG.E.SA.V1"
        DISP=SHR;
%END;
%DO SEG = 10 %TO 12;
LIBNAME IN&SEG. "OPDR.TG.PRD.ETTW.#6266.DAF21P.CER21&SEG.E.SA.V1"
        DISP=SHR;
%END;
```

Appendix A.25

JCL/SAS Code: CEREXTC

```

/* COMBINE 12 CER FILES INTO ONE CER FILE */
DATA OUT.CER;
MERGE FIN.CER (IN=A KEEP=SSN)
%DO MM=1 %TO &TOTMOS.;
  IN&MM..CER(RENAME=(
CIT=   CIT&&YY&MM
RDX=   RDX&&YY&MM
IET4=  IET4&&YY&MM
IEA_BLIND= IEA_BLIND&&YY&MM
IEA_PASS= IEA_PASS&&YY&MM
IEA_LOSS= IEA_LOSS&&YY&MM
IEA_NESE= IEA_NESE&&YY&MM
IEA_IRWE= IEA_IRWE&&YY&MM
IEA_WAGE= IEA_WAGE&&YY&MM
IUA_SSDI= IUA_SSDI&&YY&MM
IUA_TANF= IUA_TANF&&YY&MM
IUA_WC=  IUA_WC&&YY&MM
MEDC=   MEDC&&YY&MM
PDS2=   PDS2&&YY&MM
PDS3=   PDS3&&YY&MM
CPST=   CPST&&YY&MM
RESZIP = RZIP&&YY&MM
RZIPS =  RZPS&&YY&MM
PDZIP1_5 = PAYEEZIP&&YY&MM
PDZIP6_9 = PAYEEZIP_EXT&&YY&MM
AAZIP1_5 = APPZIP&&YY&MM
AAZIP6_9 = APPZIP_EXT&&YY&MM
ERP_IND= ERP_IND&&YY&MM
IEA_SEIE= IEA_SEIE&&YY&MM
  ))
%END;
;
BY SSN;
IF A;

RUN;

PROC CONTENTS DATA=OUT.CER; RUN;

%MEND START;

%START;
//*****
/* SENDING EMAIL FROM HOST USING IEBGENER

```

Appendix A.25

JCL/SAS Code: CEREXTC

```
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
DATA  
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
SUBJECT: CERCOMB IS COMPLETE
```

## Appendix A.26

JCL/SAS Code: JCL83N

```

//#6502AL8 JOB (12510000,T715,,SAS,,ITC9FL),ALEX-BRYCE,
// MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.P831.PRDLIB(JCL83N) |
//* | |
//* | THIS JCL RUNS YEARS 1988-THE CURRENT DAF YYYY |
//* *-----*
//* READ IN THE 831/833 FILES
//* JCL CALLS THE APPROPRIATE PROGRAM FOR READING 831 & 833 DATA
//* CONTACT ALEX BRYCE
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//* READ IN PREVIOUS YEARS 831 FILE
//READ831 PROC
//RD831 EXEC SAS9,
// WORK='120000,60000', * SPACE IN BLKS *
// PARM='SYSPARM="&YEAR"'
//*
//P831 DD DSN=AIS.F5750DDB.UNI831.CY&YEAR.,DISP=SHR
//FIND DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.FIND831.SA.V1,DISP=SHR
//TEMP DD DSN=&&TEMP,
// DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(3000,3000),RLSE)
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.P831.Y&YEAR..SA,
// DISP=(NEW,CATLG,DELETE),
// SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.P831.PRDLIB(RD831),
// DISP=(SHR,PASS,KEEP)
// PEND
//*
//RUN1988 EXEC READ831,YEAR=1988
//RUN1989 EXEC READ831,YEAR=1989
//RUN1990 EXEC READ831,YEAR=1990
//RUN1991 EXEC READ831,YEAR=1991
//RUN1992 EXEC READ831,YEAR=1992
//RUN1993 EXEC READ831,YEAR=1993
//RUN1994 EXEC READ831,YEAR=1994
//RUN1995 EXEC READ831,YEAR=1995
//RUN1996 EXEC READ831,YEAR=1996
//RUN1997 EXEC READ831,YEAR=1997
//RUN1998 EXEC READ831,YEAR=1998

```

## Appendix A.26

JCL/SAS Code: JCL83N

```
//RUN1999 EXEC READ831,YEAR=1999
//RUN2000 EXEC READ831,YEAR=2000
//RUN2001 EXEC READ831,YEAR=2001
//RUN2002 EXEC READ831,YEAR=2002
//RUN2003 EXEC READ831,YEAR=2003
//RUN2004 EXEC READ831,YEAR=2004
//RUN2005 EXEC READ831,YEAR=2005
//RUN2006 EXEC READ831,YEAR=2006
//RUN2007 EXEC READ831,YEAR=2007
//RUN2008 EXEC READ831,YEAR=2008
//RUN2009 EXEC READ831,YEAR=2009
//RUN2010 EXEC READ831,YEAR=2010
//RUN2011 EXEC READ831,YEAR=2011
//RUN2012 EXEC READ831,YEAR=2012
//RUN2013 EXEC READ831,YEAR=2013
//RUN2014 EXEC READ831,YEAR=2014
//RUN2015 EXEC READ831,YEAR=2015
//RUN2016 EXEC READ831,YEAR=2016
//RUN2017 EXEC READ831,YEAR=2017
//RUN2018 EXEC READ831,YEAR=2018
//RUN2019 EXEC READ831,YEAR=2019
//RUN2020 EXEC READ831,YEAR=2020
//*
/* READ IN CURRENT YEAR 831 FILE
//RD831 EXEC SAS9,
//  WORK='120000,60000'
/*
//P831 DD DSN=AIS.D5750030.UNI831.MONDAY.WEEK2164,DISP=SHR
//FIND DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.FIND831.SA.V1,DISP=SHR
//TEMP DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.P831.Y2021.SA,
//  DISP=(NEW,CATLG,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.P831.PRDLIB(RD831),
//  DISP=(SHR,PASS,KEEP)
/*
/* READ IN PREVIOUS YEARS 833 FILE
//READ833 PROC
//RD833 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="'&YEAR"'
/*
//P833 DD DSN=AIS.F57500DB.UNI833.CY&YEAR.,DISP=SHR
```

## Appendix A.26

JCL/SAS Code: JCL83N

```
//FIND DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.FIND831.SA.V1,DISP=SHR
//TEMP DD DSN=&&TEMP,
// DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(3000,3000),RLSE)
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.P833.Y&YEAR..SA,
// DISP=(NEW,CATLG,DELETE),
// SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.P831.PRDLIB(RD833),
// DISP=(SHR,PASS,KEEP)
// PEND
//*
//RUN1988 EXEC READ833,YEAR=1988
//RUN1989 EXEC READ833,YEAR=1989
//RUN1990 EXEC READ833,YEAR=1990
//RUN1991 EXEC READ833,YEAR=1991
//RUN1992 EXEC READ833,YEAR=1992
//RUN1993 EXEC READ833,YEAR=1993
//RUN1994 EXEC READ833,YEAR=1994
//RUN1995 EXEC READ833,YEAR=1995
//RUN1996 EXEC READ833,YEAR=1996
//RUN1997 EXEC READ833,YEAR=1997
//RUN1998 EXEC READ833,YEAR=1998
//RUN1999 EXEC READ833,YEAR=1999
//RUN2000 EXEC READ833,YEAR=2000
//RUN2001 EXEC READ833,YEAR=2001
//RUN2002 EXEC READ833,YEAR=2002
//RUN2003 EXEC READ833,YEAR=2003
//RUN2004 EXEC READ833,YEAR=2004
//RUN2005 EXEC READ833,YEAR=2005
//RUN2006 EXEC READ833,YEAR=2006
//RUN2007 EXEC READ833,YEAR=2007
//RUN2008 EXEC READ833,YEAR=2008
//RUN2009 EXEC READ833,YEAR=2009
//RUN2010 EXEC READ833,YEAR=2010
//RUN2011 EXEC READ833,YEAR=2011
//RUN2012 EXEC READ833,YEAR=2012
//RUN2013 EXEC READ833,YEAR=2013
//RUN2014 EXEC READ833,YEAR=2014
//RUN2015 EXEC READ833,YEAR=2015
//RUN2016 EXEC READ833,YEAR=2016
//RUN2017 EXEC READ833,YEAR=2017
//RUN2018 EXEC READ833,YEAR=2018
//RUN2019 EXEC READ833,YEAR=2019
//RUN2020 EXEC READ833,YEAR=2020
//*
```

Appendix A.26

JCL/SAS Code: JCL83N

```
//* READ IN CURRENT YEAR 833 FILE
//RD833 EXEC SAS9,
// WORK='120000,60000'
//*
//P833 DD DSN=AIS.D5750030.UNI833.MONDAY.WEEK2164,DISP=SHR
//FIND DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.FIND831.SA.V1,DISP=SHR
//TEMP DD DSN=&&TEMP,
// DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(3000,3000),RLSE)
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.P833.Y2021.SA,
// DISP=(NEW,CATLG,DELETE),
// SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.P831.PRDLIB(RD833),
// DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: READ 831/833 YEARS 1988-2021 COMPLETE
//*
```

## Appendix A.27

JCL/SAS Code: RD831

```

/*****
* FILENAME: OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.P831.PRDLIB(RD831)
* PROGRAMMER: KATE BARTKUS
* PURPOSE: TO READ IN THE 831 TEXT FILES
* EDITED: NATALIE HAZELWOOD, APR 2012 FOR TRF11, ADDED
*   DO,DD,DT,PD,PDD,VRA
* EDITED: SVETLANA BRONNIKOV, MARCH 2013 FOR DAF12 AGE CALCULATION
* EDITED: JOE MASTRIANNI, MARCH 2014 ADDED VARIABLES (SEE COMMENTS)
* CONTACT: JMASTRIANNI@MATHEMATICA-MPR.COM
*****/
OPTIONS NOCENTER LS=132 PS=60 OBS=MAX;

```

```

DATA WKD831;
INFILE P831 TRUNCOVER;
INPUT
@001 SSN   $char9.
@010 bic   $char2.
@012 AL    $char1.
@013 rid   $char1.
@022 toc   $char1.
@023 dodec ?? yymmdd8.
@023 dodecM ?? yymmN6. /*RE-READING IN MONTHLY VERSION
                        IN CASE DAY OF DODEC IS INVALID*/
@031 rdt   $char1.
@037 rb    $char2.
@039 dob   ?? yymmdd8.
@054 rpx   $char4.
@062 rsdx  $char4.
@066 ddX   $char6.
@072 dt    $char1.
@073 dr    $char1.
@123 ddpb  $char8.
@123 ddpbX $char6. /*RE-READING IN 6 CHARACTER VERSION
                   INCASE 8 CHARACTER DDPB IS INVALID*/
@131 do    $char3.
@140 ed    $char2.
@142 vra   $char1.
@157 dotX  $char6.
@165 PD    $char1.
@168 PDD   $char6.
@229 dpm   $char1.
/*QDD/DSI VARIABLES NEW FOR DAF14*/
@366 QDDIND $CHAR1.
@367 QDRMVDT ?? YMMDD8.

```



## Appendix A.27

JCL/SAS Code: RD831

```
@375 QDRMVRSN $CHAR2.
@377 QDREINDT ?? YMMMD8.
@385 QDSCORE $CHAR1.
@386 DSICLAIM $CHAR1.
;

/* MONTH MAY BE MISSING ON DISABILITY DATE FIELDS
IF SO FILL IN WITH JUNE */
IF SUBSTR(DDPBX,5,2) = '88' THEN SUBSTR(DDPBX,5,2) = '06';
IF SUBSTR(DOTX ,5,2) = '88' THEN SUBSTR(DOTX ,5,2) = '06';
IF SUBSTR(DDX ,5,2) = '88' THEN SUBSTR(DDX ,5,2) = '06';

DDPBM = INPUT(DDPBX,?? YMMMN6.);
DOT = INPUT(DOTX ,?? YMMMN6.);
DD = INPUT(DDX ,?? YMMMN6.);

DECAGE = INTCK('MONTH',DOB,DODEC,'C')/12;

DROP DDPBX DOTX DDX;
RUN;
/* SUBSET RECORDS TO CASES IN FINDER */
DATA TEMP.WKD831;
MERGE WKD831 (IN=WKD )
      FIND.FIND831 (IN=FIND);
BY SSN;
IF WKD AND FIND;
RUN;

PROC SORT DATA=TEMP.WKD831 OUT=OUT.WKD831;
BY SSN BIC DODEC;
RUN;

PROC CONTENTS DATA=OUT.WKD831;
RUN;

PROC PRINT DATA=OUT.WKD831 (OBS=25);
RUN;
```

Appendix A.28

JCL/SAS Code: RD833

```

/*****
* FILENAME: OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.P831.PRDLIB(RD833)
* PROGRAMMER: N. KHAN
* PURPOSE: TO READ IN THE 833 TEXT FILES
* EDITED: NATALIE HAZELWOOD, APRIL 2012 TO ADD BFD, DO, DD
*   EPEDATE EPETERM EPEDENY EPEREIN EPESUSP DT RFC VRA
* EDITED: SVETLANA BRONNIKOV, MARCH 2013 AGE CALCULATION FOR DAF12
* EDITED: JOE MASTRIANNI, MARCH 2014 ADDED VARIABLES (SEE COMMENTS)
* CONTACT:JMASTRIANNI@MATHEMATICA-MPR.COM
*****/
OPTIONS NOCENTER LS=132 PS=60 OBS=MAX;

```

```

DATA WKD833;
INFILE P833 TRUNCOVER;
INPUT
@001 SSN   $char9.
@010 bic   $char2.
@012 AL    $char1.
@013 rid   $char1.
@014 toc   $char1.
@015 dodec ?? yymmdd8.
@015 dodecM ?? yymmN6. /*RE-READING IN MONTHLY VERSION
                        IN CASE DAY OF DODEC IS INVALID*/
@023 rdt   $char1.
@029 rfc   $char2.
@031 bfd   $char1.
@032 wrm   $char2.
@034 dob   ?? yymmdd8.
@049 rpx   $char4.
@057 rsdx  $char4.
@061 ddx   $char6.
@067 dt    $char1.
@068 dr    $char1.
@118 ddpb  $char8.
@118 ddpx  $char6. /*RE-READING IN 6 CHARACTER VERSION
                  INCASE 8 CHARACTER DDPB IS INVALID*/
@126 do    $char3.
@135 ed    $char2.
@137 vra   $char1.
@152 dotx  $char6.
@180 epedatex $char6.
@186 epedeny $char1.
@187 epereinx $char6.
@193 epesusp $char6.

```

## Appendix A.28

JCL/SAS Code: RD833

```
@199 epetermx $char6.
@260 cdt $char2.
@262 DPM $char1.
/*QDD/DSI VARIABLES NEW FOR DAF14*/
@397 QDDIND $CHAR1.
@398 QDRMVDT ?? YMMDD8.
@406 QDRMVRSN $CHAR2.
@408 QDREINDT ?? YMMDD8.
@416 QDScore $CHAR1.
@417 DSICLAIM $CHAR1.
;
/* MONTH MAY BE MISSING ON DISABILITY DATE FIELDS
IF SO FILL IN WITH JUNE */
IF SUBSTR(DDPBX ,5,2) = '88' THEN SUBSTR(DDPBX ,5,2) = '06';
IF SUBSTR(DOTX ,5,2) = '88' THEN SUBSTR(DOTX ,5,2) = '06';
IF SUBSTR(DDX ,5,2) = '88' THEN SUBSTR(DDX ,5,2) = '06';
/*833 ONLY*/
IF SUBSTR(EPEDATEX,5,2) = '88' THEN SUBSTR(EPEDATEX,5,2) = '06';
IF SUBSTR(EPEREINX,5,2) = '88' THEN SUBSTR(EPEREINX,5,2) = '06';
IF SUBSTR(EPESUSPX,5,2) = '88' THEN SUBSTR(EPESUSPX,5,2) = '06';
IF SUBSTR(EPETERMX,5,2) = '88' THEN SUBSTR(EPETERMX,5,2) = '06';

DDPBM = INPUT(DDPBX,?? YMMN6.);
DOT = INPUT(DOTX ,?? YMMN6.);
DD = INPUT(DDX ,?? YMMN6.);

EPEDATE = INPUT(EPEDATEX,?? YMMN6.);
EPEREIN = INPUT(EPEREINX,?? YMMN6.);
EPESUSP = INPUT(EPESUSPX,?? YMMN6.);
EPETERM = INPUT(EPETERMX,?? YMMN6.);

DECAGE = INTCK('MONTH',DOB,DODEC,'C')/12;

DROP DDPBX DOTX DDX EPEDATEX EPEREINX EPESUSPX EPETERMX;
RUN;

/* SUBSET RECORDS TO CASES IN FINDER */
DATA TEMP.WKD833;
MERGE WKD833 (IN=WKD)
      FIND.FIND831 (IN=FIND);
BY SSN;
IF WKD AND FIND;
RUN;

PROC SORT DATA=TEMP.WKD833 OUT=OUT.WKD833;
```

Appendix A.28

JCL/SAS Code: RD833

```
BY SSN BIC DODEC;  
RUN;
```

```
PROC CONTENTS DATA=OUT.WKD833;  
RUN;
```

```
PROC PRINT DATA=OUT.WKD833 (OBS=25);  
RUN;
```

## Appendix A.29

JCL/SAS Code: CMB3133

```

//#6502C3N JOB (12510000,T715,,SAS,,ITC9FL),ALEX-BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.P831.PRDLIB(CMB3133) |
//*  *-----*
//*
//*  1. ALL STEPS ARE STEP RESTARTABLE
//*  2. COMBINING ALL 831-833 FILES
//*
//* CONTACT ALEX BRYCE
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*
//*****
//*
//  SET REG='64M'
//*
//JS010 EXEC SAS9,
//  WORK='200000,100000',
//  REGION=&REG,
//  PARM='MEMSIZE=&REG'
//*
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//TEMP1 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//* ADDITIONAL INPUT DATASETS READ IN AT LINES 50 & 53
//* MAKE SURE TO UPDATE THESE!!
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.CMB3133.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//MEAN  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.P831.MEAN.SA,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES OBS=MAX MPRINT MACROGEN;

%LET DAFYEAR=2021; /*UDPATE TO CURRENT DAF YEAR*/
%LET DAFYY  =%SUBSTR(&DAFYEAR.,3);
/*THE NUMBER OF N VARIABLES WAS RESTRICTED TO 16 FOR COMBINED
AND 10 FOR T16 IN DAF12*/

```

Appendix A.29

JCL/SAS Code: CMB3133

```
%LET T16REST = 10; /*CHECK LAST YEARS LOG TO DETERMINE THIS*/
%LET TOTREST = 16; /*CHECK LAST YEARS LOG TO DETERMINE THIS*/

%MACRO COMB831833;
%DO YEAR=1988 %TO &DAFYEAR.;
LIBNAME INA&YEAR.
  "OPDR.TG.PRD.ETTW.#6502.DAF&DAFY..P831.Y&YEAR..SA"
DISP=SHR;
LIBNAME INB&YEAR.
  "OPDR.TG.PRD.ETTW.#6502.DAF&DAFY..P833.Y&YEAR..SA"
DISP=SHR;
%END;

DATA OUT.COMB3133;
SET
%DO YEAR=1988 %TO &DAFYEAR.;
INA&YEAR..WKD831
INB&YEAR..WKD833
%END;;
BY SSN BIC DODEC;
/* DODEC IS THE DATE OF DECISION. THERE ARE MULTIPLE RECORDS FOR
BENEFICIARIES BECAUSE CDR'S (CONTINUING DISABILITY REVIEWS) ARE
OCCURRING EVERY 3 OR 7 YEARS, CREATING THE MULTIPLE RECORDS

THE COMBINATION OF SSN AND BIC IS THE LINK FOR T2 CASES.
SSN FOR T2 IS THE CAN. ALL T2 RECORDS FOR A GIVEN CAN
HAVE IDENTICAL SSN'S AND CAN BE DISTINGUISHED ONLY BY THE BIC.
THERE IS NO BOAN FIELD IN THE 831-33 FILES.
THE LINK FOR T16 IS SSN - THE PAN - WHICH IS THE SAME AS
SSN IN DAF. THE DATA LISTS ALL T16 BICS AS "A" EVEN THOUGH
BIC IS NOT RELEVANT TO T16 CASES. THE SSN'S ARE UNIQUE THOUGH
THERE MAY BE MULTIPLE RECORDS FOR A GIVEN BENEFICIARY

CREATE A FLAG TO INDICATE DI SSI LINES*/

LINEDI = 0;
LINESSI = 0;

IF RID IN('2','4') THEN LINEDI = 1;
ELSE IF RID IN('R','S') THEN LINESSI = 1;

RUN;

/* COUNT UP TOTAL CDR'S-NUMBER T2 CDRS-NUMBER T16 CDRS.
THESE FIELDS WILL INDEX THE ARRAYS FOR VARIABLES */
```

## Appendix A.29

JCL/SAS Code: CMB3133

```
PROC SQL NOPRINT;
  CREATE TABLE TEMP1.CNTCDR AS
  SELECT COUNT(BIC) AS CDRTOT,
         SUM(LINEDI) AS CDRT2,
         SUM(LINESSI) AS CDRT16
  FROM OUT.COMB3133
  GROUP BY SSN,BIC;
QUIT;

*DETERMINE MAX NUMBER AND P99 OF ADJUDICATIONS FOR USE IN FUTURE STEPS;
PROC MEANS DATA=TEMP1.CNTCDR P99 MAX;
  VAR CDRT16 CDRTOT;
  OUTPUT OUT=MEAN.P831MEAN (DROP = _TYPE_ _FREQ_)
  P99(CDRT16)=P99CDRT16
  P99(CDRTOT)=P99CDRTOT
  MAX(CDRT16)=MAXCDRT16
  MAX(CDRTOT)=MAXCDRTOT;
RUN;

*CHECK WITH PROC FREQ;
PROC FREQ DATA=TEMP1.CNTCDR;
  TABLES CDRT16 CDRTOT / LIST MISSING;
  TITLE "DETERMINE MAX NUM OF ADJUDICATIONS";
RUN;

DATA MEAN.P831MEAN;
  SET MEAN.P831MEAN;

/*THE NUMBER OF N VARIABLES WAS RESTRICTED TO 16 FOR COMBINED
  AND 10 FOR T16 IN TRF12
  RESTRICT THE NUMBER OF N VAIABLES TO MAX OF 99TH PERCENTILE
  AND THE NUMBER FROM LAST YEAR*/

MAXT16_99 = MAX(ROUND(P99CDRT16),&T16REST.);
MAXTOT_99 = MAX(ROUND(P99CDRTOT),&TOTREST.);

RUN;

TITLE "LOOK HERE TO GET THE VALUES OF T16REST AND TOTREST MACRO VARS";
TITLE2 "FOR NEXT DAF YEAR PROCESSING";
PROC PRINT DATA = MEAN.P831MEAN;
RUN;
TITLE; TITLE2;

PROC CONTENTS DATA = OUT.COMB3133;
```

Appendix A.29

JCL/SAS Code: CMB3133

```
RUN;

PROC CONTENTS DATA = MEAN.P831MEAN;
RUN;

PROC PRINT DATA=OUT.COMB3133(OBS=25);
  FORMAT DODECM DOT DDPBM YYMMN6.;
RUN;
%MEND COMB831833;
%COMB831833;
/*
//*****
/* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2  DD SYSOUT=(B,SMTP)
//SYSUT1  DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 CMB3133 COMPLETE
/*
```



## Appendix A.30

JCL/SAS Code: P3133SSI

```

//#6502PSI JOB (12510000,T715,,SAS,,ITC9FL),ALEX-BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.P831.PRDLIB(P3133SSI) |
//*  *-----*
//*
//* PURPOSE:PROCESS 831/833 DATA FOR SSI BENEFICIARIES
//*   CREATE MONTHLY VARS FROM 831-833 FILES FOR TTW
//* CREATED:02/26/05
//* UPDATED:03/22/22
//* CONTACT ALEX BRYCE
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*
//*****
//*
//JS010 EXEC SAS9,
//  WORK='120000,60000'
//*
//CMB  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.CMB3133.SA.V1,
//     DISP=SHR
//MEAN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.P831.MEAN.SA,
//     DISP=SHR
//TEMP DD DSN=&&TEMP,
//     DISP=(NEW,DELETE,DELETE),
//     SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.RSTRCT16.SA.V1,
//     DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//     SPACE=(CYL,(3000,3000),RLSE)
//*
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS =MAX;

/* CREATE A MACRO VARIABLE TO INDEX THE FOLLOWING ARRAYS
IT FLAGS THE MAXIMUM NUMBER OF RECORDS FOR A BENEFICIARY */
*SQL NOW NEEDED TO AVOID INX=    28;
*(CAUSES ISSUES WITH VAR1-VAR&INX SYNTAX);
PROC SQL NOPRINT;
  SELECT MAXCDRT16,
         MAXT16_99,
         MAXCDRT16
  INTO  :INX  SEPARATED BY ' ',
        :REST16 SEPARATED BY ' ',
        :MAXT16 SEPARATED BY ' '

```

## Appendix A.30

JCL/SAS Code: P3133SSI

```
FROM MEAN.P831MEAN;
QUIT;

%PUT INX=&INX.;
%PUT REST16=&REST16.;
%PUT MAXT16=&MAXT16.;

%MACRO P3133SSI;
DATA TEMP.SSICDR(KEEP=SSN AGE18_CDR AGE18REDDT
/*MAX:*/
    T16APPL1 - T16APPL&INX
    T16START1 - T16START&INX
    T16STOP1 - T16STOP&INX
    T16RID1 - T16RID&INX);
    *KEEP T16 CASES;
SET CMB.COMB3133(WHERE = (RID IN ('R','S')));
BY SSN BIC DODEC ; /* BIC IS ALWAYS "A" FOR T16 CLAIMS */

LENGTH DEC1 - DEC&INX
    T16RID1 - T16RID&INX $1;

/* SET UP ARRAYS FOR CREATING VARIABLES */
ARRAY T16APPL (*) T16APPL1 - T16APPL&INX ;
ARRAY T16START (*) T16START1 - T16START&INX;
ARRAY T16STOP (*) T16STOP1 - T16STOP&INX ;
ARRAY DEC (*) $ DEC1 - DEC&INX ;
ARRAY T16RID (*) $ T16RID1 - T16RID&INX ;

RETAIN T16APPL1 - T16APPL&INX
    T16START1 - T16START&INX
    T16STOP1 - T16STOP&INX
    DEC1 - DEC&INX
    T16RID1 - T16RID&INX
    AGE18_CDR AGE18REDDT INDX;

IF FIRST.SSN THEN DO; /* BIC IS ALWAYS "A" FOR SSI RECORDS */

/* INITIALIZE VARS */
DO I = 1 TO DIM(T16APPL);
    T16APPL(I) = .;
    T16START(I) = .;
    T16STOP(I) = .;
    DEC(I) = ' ';
    T16RID(I) = ' ';
END;
```

## Appendix A.30

JCL/SAS Code: P3133SSI

```
AGE18_CDR = 0;
AGE18REDDT = .;
INDX = 0;

END; /* FIRST SSN */

INDX = INDX + 1;

DEC(INDX) = RDT;
T16RID(INDX) = RID;
IF AL IN ('B','C','D','E','F') THEN T16APPL(INDX) = DODECM;
IF DDPBM GT .Z THEN T16START(INDX) = DDPBM;
IF RDT = 'T' THEN T16STOP(INDX) = DOT;

/* DETERMINE IF BENE HAS BEEN RE-DETERMINED AS ADULT */
IF DECA GE 18 AND RDT = 'C' AND CDT = '04' AND WRM = '29'
THEN DO;
AGE18_CDR = 1;
AGE18REDDT = DODEC;
END;

IF LAST.SSN THEN DO;

/* DELETE STOP DATES IF A SUBSEQUENT APPEAL RESULTS IN CONTINUATION */
DO I = 2 TO &INX; /* ONLY IF MORE THAN ONE CLAIM */

IF T16APPL(I) GT .Z AND DEC(I) = 'C' THEN DO; /* CONTINUATION */
IF T16STOP(I-1) NE . THEN T16STOP(I-1) = .;
END;

END; /* DO I = 2 TO &INX */

LABEL AGE18_CDR = "INDICATOR FOR ADULT REDETERMINATION"
AGE18REDDT = "ADULT REDETERMINATION DATE"
%DO I=1 %TO &INX;
T16APPL&I = "T16 APPEAL DATE &I"
T16START&I = "T16 BENEFIT ENTITLEMENT DATE &I"
T16STOP&I = "T16 BENEFIT CESSATION DATE &I"
%END;;
OUTPUT;
END; /* LAST.SSN */
RUN;

PROC FREQ DATA=TEMP.SSICDR; TABLES AGE18REDDT; WHERE AGE18_CDR = 1;
```

## Appendix A.30

JCL/SAS Code: P3133SSI

```
FORMAT AGE18REDDT YMMDD10.;
RUN;
PROC CONTENTS; RUN;

%MEND P3133SSI;
%P3133SSI;

/*START RSTRCT16*/

%MACRO RSTRCT16;
*** PRINT RECORDS PRIOR TO PROCESSING IN ORDER TO CHECK THAT ***;
*** OUTPUT IS WHAT WE EXPECT ***;

PROC PRINT DATA=TEMP.SSICDR(OBS=25);
  VAR SSN T16: ;
  WHERE T16RID11 NE ' ';
  TITLE "CHECK VARIABLES PRIOR TO PROCESSING";
  TITLE2 "RECORDS WITH MORE THAN &REST16 N OCCURRENCES";
RUN;

PROC PRINT DATA=TEMP.SSICDR(OBS=25);
  VAR SSN T16: ;
  TITLE2 " ";
RUN;

DATA OUT.RSTRCT16(DROP=I J RID_START T16RID:);
SET TEMP.SSICDR;

*RESTRICT NUMBER FOR SSI;
ARRAY T16APPL  (*) T16APPL1 - T16APPL&MAXT16;
ARRAY T16RID  (*) $ T16RID1 - T16RID&MAXT16;
ARRAY T16START (*) T16START1 - T16START&MAXT16;
ARRAY T16STOP  (*) T16STOP1 - T16STOP&MAXT16;

*OUTPUT ARRAY;
ARRAY HOLDT16APPL (*) HOLDT16APPL1 - HOLDT16APPL&REST16 ;
ARRAY HOLDT16START(*) HOLDT16START1 - HOLDT16START&REST16;
ARRAY HOLDT16STOP (*) HOLDT16STOP1 - HOLDT16STOP&REST16 ;

RID_INDEX=0;

DO I=&MAXT16 TO 1 BY -1 UNTIL (RID_INDEX>0);
  IF T16RID(I) NE ' ' THEN RID_INDEX=I;
END;
```

## Appendix A.30

JCL/SAS Code: P3133SSI

```
%MACRO REST(VAR=);
*** STEP 1 IS TO CREATE AN ADDITIONAL VARIABLE THAT IS ***;
*** SET TO THE EARLIEST OCCURRENCE OF EACH N VARIABLE ***;
FIRST&VAR.=&VAR.1;

IF RID_INDEX <= &REST16 THEN DO I=1 TO &REST16;
  HOLD&VAR.(I)=&VAR.(I);
END;

*** IF RID_INDEX > 10 THEN SET THE LAST 10 OCCURRENCES TO THE ***;
*** HOLD VARIABLES ***;

IF RID_INDEX > &REST16 THEN DO;
  RID_START=RID_INDEX-(&REST16-1);

  DO I = RID_START TO RID_INDEX;
    J = I-(RID_START-1);
    HOLD&VAR(J) = &VAR.(I);
  END;
END;

DROP &VAR;;
RENAME
%DO I = 1 %TO &REST16;
  HOLD&&VAR.&I=&&VAR.&I
%END;;

%MEND REST;
%REST(VAR=T16APPL );
%REST(VAR=T16START);
%REST(VAR=T16STOP );
RUN;

*** PRINT RECORDS AFTER FINAL PROCESSING IN ORDER TO CHECK THAT ***;
*** OUTPUT IS WHAT WE EXPECT ***;

PROC PRINT DATA=OUT.RSTRCT16(OBS=25);
  VAR SSN T16: ;
  WHERE RID_INDEX > &REST16;
  TITLE "CHECK VARIABLES AFTER PROCESSING";
  TITLE2 "RECORDS WITH MORE THAN &REST16 N OCCURRENCES";
RUN;

PROC PRINT DATA=OUT.RSTRCT16(OBS=25);
  VAR SSN T16: ;
```

Appendix A.30

JCL/SAS Code: P3133SSI

```
TITLE "CHECK VARIABLES AFTER PROCESSING";  
TITLE2 " ";  
RUN;
```

```
PROC CONTENTS DATA=OUT.RSTRCT16;  
  TITLE "PROC CONTENTS RSTRCT16";  
RUN;  
%MEND RSTRCT16;  
%RSTRCT16;
```

## Appendix A.31

JCL/SAS Code: P3133ALL

```

//#6502PAL JOB (12510000,T715,,SAS,,ITC9FL),ALEX-BRYCE,          *****
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6266.DAF17.P831.PRDLIB(P3133ALL) |
//*  *-----*
//*
//* PURPOSE:PROCESS 831/833 DATA FOR ALL BENEFICIARIES
//*   TO CREATE MONTHLY VARS FROM 831-833 FILES
//* CREATED:02/26/05
//* UPDATED:03/22/22
//* CONTACT ALEX BRYCE
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*
//*****
//*
//JS010 EXEC SAS9,
//  WORK='120000,60000'
//TEMP1 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//CMB DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.CMB3133.SA.V1,
//  DISP=SHR
//MEAN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.P831.MEAN.SA,
//  DISP=SHR
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.ALLCDR.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.RSTRCTN.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//*

  OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS = MAX;
%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

/* STEP TO ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA */

%MACRO P3133ALL;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

```

## Appendix A.31

JCL/SAS Code: P3133ALL

```
/* CREATE A MACRO VARIABLE TO INDEX MAX OF RESTRICT N IN ALL FILE  
AND MAX OF TOTAL CRDS; */
```

```
PROC SQL;  
SELECT MAXCDRTOT,  
       MAXTOT_99,  
       MAXCDRTOT  
INTO :IND SEPARATED BY '|',  
     :REST SEPARATED BY '|',  
     :MAXT SEPARATED BY '|'  
FROM MEAN.P831MEAN;  
RUN;
```

```
%PUT IND(MAXCDRTOT)=&IND;  
%PUT REST(MAXTOT_99)=&REST;  
%PUT MAXT(MAXCDRTOT)=&MAXT;
```

```
DATA OUT1.ALLCDR  
(KEEP= SSN BIC PROGRAM /*MAX:*/  
  %DO I = 1 %TO &IND;  
    DODEC&I JUDLVL&I RDT&I RID&I RB&I DPM&I  
    DOC&I DD&I DT&I PD&I PDD&I VRA&I  
    BFD&I EPEDATE&I EPETERM&I EPEDENY&I EPEREIN&I  
    EPESUSP&I RFC&I DR&I  
  
    QDDIND&I QDRMVDT&I QDRMVRSN&I  
    QDREINDT&I QDScore&I  
    DSICLAIM&I  
  %END;  
  %DO MM = 1 %TO &TOTMOS;  
    MIEX&&YY&MM ALX&&YY&MM EDX&&YY&MM  
    DXPRI_SSI&&YY&MM DXPRI_DI831&&YY&MM  
    DXSEC_SSI&&YY&MM DXSEC_DI831&&YY&MM  
  %END;);
```

```
LENGTH  
%DO I = 1 %TO &IND;  
  JUDLVL&I  
  RDT&I  
  RID&I  
  DPM&I  
  DT&I  
  PD&I  
  VRA&I  
  BFD&I
```



## Appendix A.31

JCL/SAS Code: P3133ALL

```
EPEDENY&I
  DR&I
%END;
$1;
SET CMB.COMB3133;
BY SSN BIC DODEC ;

/*CREATE PROGRAM FLAGS FOR DI AND SSI*/
RETAIN PROGRMDI PROGRMSSI;
/*LENGTH PROGRMDI PROGRMSSI 3;*/ /*DAF14 REMOVE?*/

IF FIRST.BIC THEN DO;
  PROGRMDI = 0;
  PROGRMSSI = 0;
END;

  IF LINEDI =1 THEN PROGRMDI = 1; /*NEVER THE CASE WHERE */
  ELSE IF LINESSI=1 THEN PROGRMSSI = 1; /*LINEDI=1 AND LINESSI=1*/

/* SET UP ARRAYS FOR CREATING VARIABLES */

ARRAY DODECA (*) DODEC1 - DODEC&IND;
ARRAY CDRTYP (*) $ JUDLVL1 - JUDLVL&IND;
ARRAY DECIS (*) $ RDT1 - RDT&IND;
ARRAY TERM (*) TRMDTE1 - TRMDTE&IND;
ARRAY PROG (*) $ RID1 - RID&IND;
ARRAY DIS (*) DDPBM1 - DDPBM&IND;
ARRAY REG (*) $ RB1 - RB&IND;
ARRAY CONTIN (*) $ RFC1 - RFC&IND;
ARRAY PRMDIS (*) $ DPM1 - DPM&IND;
ARRAY DSTOFF (*) $ DOC1 - DOC&IND;
ARRAY DIARYDT (*) DD1 - DD&IND;
ARRAY DIARYTP (*) $ DT1 - DT&IND;
ARRAY DISDEC (*) $ PD1 - PD&IND;
ARRAY DECDT (*) $ PDD1 - PDD&IND;
ARRAY VRACT (*) $ VRA1 - VRA&IND;
ARRAY BASDT (*) $ BFD1 - BFD&IND;
ARRAY EPEDT (*) EPEDATE1 - EPEDATE&IND;
ARRAY EPETRM (*) EPETERM1 - EPETERM&IND;
ARRAY EPEDNY (*) $ EPEDENY1 - EPEDENY&IND;
ARRAY EPERIN (*) EPEREIN1 - EPEREIN&IND;
ARRAY EPESUS (*) EPESUSP1 - EPESUSP&IND;
ARRAY DIARYRES (*) DR1 - DR&IND;
```

## Appendix A.31

JCL/SAS Code: P3133ALL

```
ARRAY AQDDIND (*) $ QDDIND1 - QDDIND&IND;
ARRAY AQDRMVDT (*) QDRMVDT1 - QDRMVDT&IND;
ARRAY AQDRMVRSN (*) $ QDRMVRSN1 - QDRMVRSN&IND;
ARRAY AQDREINDT (*) QDREINDT1 - QDREINDT&IND;
ARRAY AQDSCORE (*) $ QDSCORE1 - QDSCORE&IND;
ARRAY ADSICLAIM (*) $ DSICLAIM1 - DSICLAIM&IND;

RETAIN %DO I = 1 %TO &IND;
  DODEC&I JUDLVL&I RDT&I RID&I TRMDTE&I
  DDPBM&I RB&I RFC&I DPM&I DOC&I DD&I
  DT&I PD&I PDD&I VRA&I BFD&I
  EPEDATE&I EPETERM&I EPEDENY&I EPEREIN&I EPESUSP&I DR&I

  QDDIND&I QDRMVDT&I QDRMVRSN&I
  QDREINDT&I QDSCORE&I
  DSICLAIM&I
%END;
INDX;

IF FIRST.BIC THEN DO I = 1 TO DIM(DODECA);
/* INITIALIZE VARIABLES */
  DODECA(I) = . ;
  CDRTYP(I) = ' ';
  DECIS(I) = ' ';
  PROG(I) = ' ';
  TERM(I) = . ;
  DIS(I) = . ;
  REG(I) = ' ';
  CONTIN(I) = ' ';
  PRMDIS(I) = ' ';
  DSTOFF(I) = ' ';
  DIARYDT(I) = . ;
  DIARYTP(I) = ' ';
  DISDEC(I) = ' ';
  DECDT(I) = ' ';
  VRACT(I) = ' ';
  BASDT(I) = ' ';
  EPEDT(I) = . ;
  EPETRM(I) = . ;
  EPEDNY(I) = ' ';
  EPERIN(I) = . ;
  EPESUS(I) = . ;
  DIARYRES(I) = ' ';

  AQDDIND(I) = ' ';
```

## Appendix A.31

JCL/SAS Code: P3133ALL

```
AQDRMVDT(I) = . ;
AQDRMVRSN(I) = ' ' ;
AQDREINDT(I) = . ;
AQDScore(I) = ' ' ;
ADSICLAIM(I) = ' ' ;
```

```
INDX = 0 ; /*UNDERSTOOD THIS IS BEING DONE FOR EACH 'I'*/
END;
```

```
INDX = INDX + 1;
DODECA(INDX) = DODEC;
CDRTYP(INDX) = AL;
DECIS(INDX) = RDT;
TERM(INDX) = DOT;
PROG(INDX) = RID;
DIS(INDX) = DDPBM;
REG(INDX) = RB;
CONTIN(INDX) = RFC;
PRMDIS(INDX) = DPM;
DSTOFF(INDX) = DO;
DIARYDT(INDX) = DD;
DIARYTP(INDX) = DT;
DISDEC(INDX) = PD;
DEC DT(INDX) = PDD;
VRACT(INDX) = VRA;
BASDT(INDX) = BFD;
EPEDT(INDX) = EPEDATE;
EPETRM(INDX) = EPETERM;
EPEDNY(INDX) = EPEDENY;
EPERIN(INDX) = EPEREIN;
EPESUS(INDX) = EPESUSP;
DIARYRES(INDX) = DR;
```

```
AQDDIND(INDX) = QDDIND;
AQDRMVDT(INDX) = QDRMVDT;
AQDRMVRSN(INDX) = QDRMVRSN;
AQDREINDT(INDX) = QDREINDT;
AQDScore(INDX) = QDScore;
ADSICLAIM(INDX) = DSICLAIM;
```

LENGTH

```
%DO MM=1 %TO &TOTMOS;
MIEX&&YY&MM
ALX&&YY&MM $1
EDX&&YY&MM $2
```

## Appendix A.31

JCL/SAS Code: P3133ALL

```

DXPRI_SSI&&YY&MM DXSEC_SSI&&YY&MM
DXPRI_DI831&&YY&MM DXSEC_DI831&&YY&MM $4
%END;;

```

```
/* CREATE ARRAYS FOR MONTHLY VARIABLES FROM 1994 THRU CURRENT YEAR*/
```

```

ARRAY EDX {*} $ %DO MM=1 %TO &TOTMOS; EDX&&YY&MM %END;;
ARRAY ALX {*} $ %DO MM=1 %TO &TOTMOS; ALX&&YY&MM %END;;
ARRAY MIE {*} $ %DO MM=1 %TO &TOTMOS; MIE&&YY&MM %END;;
ARRAY DX1SSI{*} $4 %DO MM=1 %TO &TOTMOS; DXPRI_SSI&&YY&MM %END;;
ARRAY DX2SSI{*} $4 %DO MM=1 %TO &TOTMOS; DXSEC_SSI&&YY&MM %END;;
ARRAY DX1DI{*} $4 %DO MM=1 %TO &TOTMOS; DXPRI_DI831&&YY&MM %END;;
ARRAY DX2DI{*} $4 %DO MM=1 %TO &TOTMOS; DXSEC_DI831&&YY&MM %END;;
ARRAY CYEAR (*) $ %DO MM=1 %TO &TOTMOS; C&&YY&MM %END;;
ARRAY NYEAR (*) %DO MM=1 %TO &TOTMOS; N&&YY&MM %END;;

```

```
RETAIN
```

```

%DO MM=1 %TO &TOTMOS;
    EDX&&YY&MM MIE&&YY&MM ALX&&YY&MM
    DXPRI_SSI&&YY&MM DXSEC_SSI&&YY&MM
    DXPRI_DI831&&YY&MM DXSEC_DI831&&YY&MM
%END;
TERMIND1 TERMIND2;

```

```
IF FIRST.BIC THEN DO I = 1 TO &TOTMOS;
```

```

    EDX(I) = ' ';
    MIE(I) = '0';
    ALX(I) = ' ';
    DX1SSI(I) = ' ';
    DX2SSI(I) = ' ';
    DX1DI(I) = ' ';
    DX2DI(I) = ' ';

```

```
    TERMIND1 = 0; /*UNDERSTOOD THESE ARE BEING SET TO 0 FOR EACH I*/
```

```
    TERMIND2 = 0;
```

```
END; /* FIRST BIC */
```

```

/* THE FOLLOWING ROUTINE CREATES MONTHLY REFERENCE DATES FROM
   JAN 1994 TO DEC CURRENT YEAR - EACH RECORD WILL BE EVALUATED.
   FOR BENEFICIARIES WITH MULTIPLE RECORDS THE FIELDS ARE RETAINED
   FROM RECORD TO RECORD */

```

```

J=1;
Y=&BEGYR;

```

## Appendix A.31

JCL/SAS Code: P3133ALL

```
DO I = 1 TO DIM(CYEAR); /* LOOP THROUGH ALL MONTH/YEAR REFERENCES */

CYEAR(I) = PUT(Y,4.) || (PUT(J,Z2.));

IF INT(I/12)=I/12 THEN DO; /*FINDS YEAR END*/
  J=1;
  Y=Y+1;
END;
ELSE J=J+1;

NYEAR(I)=INPUT(SUBSTR(CYEAR(I),1,4) || SUBSTR(CYEAR(I),5,2),YMMMN6.);

/* IF THE DISAB ENTIT MONTH EXISTS USE IT. ELSE USE DODEC(ADJUD DATE)
  THE DISAB ENTIT MONTH USUALLY PREDATES THE DODEC,
  BUT IS OFTEN MISSING.
  COMPARE TO EACH MONTHLY REFERENCE VARIABLE TO SET MONTHLY VARS */

/* IF THERE IS A TERMINATION DATE - END MONTHLIES THEN */
/* LOOK AT SEPARATE TITLE TERMINATIONS          */
/* FOR MULTIPLE TERMINATIONS ON THE SAME PROGRAM THE LAST
  WILL BE KEPT.
  SET TERMINATION INDEXES USING THE ARRAY REFERENCE I WHEN
  A TERMINATION DATE (DOT) IS IN THAT ARRAY MONTH */

  IF DOT = NYEAR(I) AND RID IN ('2' '4') THEN TERMIND1 = I;
  ELSE IF DOT = NYEAR(I) AND RID IN ('R' 'S') THEN TERMIND2 = I;

IF ( DDPBM > .Z AND DDPBM <= NYEAR(I)) OR
  (DODECM > .Z AND DODECM <= NYEAR(I)) THEN DO;

  EDX(I) = ED;
  /* CHANGE FOR 2006 - ALX EVALUATED FOR ACCEPTED AND DENIED CLAIMS */
  ALX(I) = AL;

IF (RID IN ('2' 'R') AND RDT = 'A') OR
  (RID IN ('4' 'S') AND RDT = 'C') THEN DO; /* ACCEPTED CLAIMS */

  /* NEW DEFINITION OF MIE */
  IF DR IN ('1','3','9') THEN MIE(I)='P';
  ELSE IF DR IN ('5','7') THEN MIE(I)='N';
  ELSE IF DR IN ('E','F','G','I','J','K','L',
    'M','N','V','W','X','Y','Z') THEN DO;
    IF DPM='P' THEN MIE(I)='N';
    ELSE MIE(I)='P';
```

## Appendix A.31

JCL/SAS Code: P3133ALL

```
END;

ELSE IF 'A'<=DR<='Z' THEN MIE(I)='E';

END; /* ACCEPTED CLAIMS */

/* SELECT ACCEPTED CLAIMS FOR SSI*/
IF (RID = 'R' AND RDT = 'A') OR
  (RID = 'S' AND RDT = 'C') THEN DO;
  DX1SSI(I) = RPD;
  DX2SSI(I) = RSD;
END; /* ACCEPTED CLAIMS FOR SSI*/

/* SELECT ACCEPTED CLAIMS FOR DI*/
IF (RID = '2' AND RDT = 'A') OR
  (RID = '4' AND RDT = 'C') THEN DO;
  DX1DI(I) = RPD;
  DX2DI(I) = RSD;
END; /* ACCEPTED CLAIMS FOR DI*/

END; /* DATE FALLS WITHIN MONTH */

END; /* LOOP THROUGH ALL MONTH-YEAR REFERENCES */

IF LAST.BIC THEN DO;

  /* TAKE THE LATEST TERMINATION TO BLANK OUT VARIABLES */
  /* IF FOR A TITLE THERE IS NO TERMINATION THEN DO NOT BLANK */
  /* IF BOTH TITLES THEN TAKE THE LATEST */

  /*CREATE TERMINATION */
  TERMIND=MAX(TERMIND1,TERMIND2);

  /* BLANK OUT THE VARIABLES AFTER TERMINATION DATE */
  /* ONLY IF THE LAST CLAIM IS TERMINATED */

  IF DOT NE . THEN DO I = 1 TO &TOTMOS;

    IF TERMIND = I THEN DO J = I TO &TOTMOS;
      ALX(J) = ' ';
      MIE(J) = '0';
    END;
    /*FOR DI BENES*/
    IF TERMIND1 = I THEN DO J = I TO &TOTMOS;
      DX1DI(J) = ' ';
```

## Appendix A.31

JCL/SAS Code: P3133ALL

```
DX2DI(J) = ' ';
END;
/*FOR SSI BENES*/
IF TERMIND2 = I THEN DO J = I TO &TOTMOS;
DX1SSI(J) = ' ';
DX2SSI(J) = ' ';
END;

END; /* DOT NE . */

/*LENGTH PROGRAM 3;*/ /*DAF14 CAN REMOVE?*/

/* LAST BIC - IF BIC = A AND HE IS NOT A CONCURRENT
HE WILL BE ASSIGNED TO DI AS A CAN.
HOWEVER FOR DI PRIMARIES (BIC = A) WHO ARE CONCURRENTS
RECORDS MAY APPEAR IN THE SSI SEGMENT. THE INFORMATION ON THESE
RECORDS MUST BE COMBINED WITH THE DI INFORMATION IN GET2SSN
BEFORE THEY ARE DE-DUPED.
A WIDOW OR A DAC WHO HAS A CAN WITH A BIC OF C OR W WILL
BE OUTPUT IN PROGRAM 1 AS A CAN. BUT THERE COULD BE AN SSI RECORD
FOR THE PRIMARY UNDER A PAN AND A BIC = A AND THIS CASE
WOULD BE OUTPUT AS PROGRAM 2 SSI. THESE RECORDS WILL BE SCREENED
OUT IN THE GET2SSN PROGRAM WHERE ONLY SSI RECORDS IN THE FINDER
FOR SSI WILL BE KEPT */

/*T2 ONLY CASES*/
IF PROGRMDI = 1 AND PROGRMSSI = 0 THEN PROGRAM = 1;
/* CONCURRENTS */
ELSE IF PROGRMDI = 1 AND PROGRMSSI = 1 THEN PROGRAM = 2;
/*T16 ONLY CASES*/
ELSE IF PROGRMSSI = 1 THEN PROGRAM = 3;

LABEL
%DO MM=1 %TO &TOTMOS;
MIEX&&YY&MM = "&&YY&MM MEDICAL IMPROVEMENT EXPECTED"
ALX&&YY&MM = "&&YY&MM LEVEL OF ADJUDICATION"
EDX&&YY&MM = "&&YY&MM YEARS OF EDUCATION"
DXPRI_SSI&&YY&MM = "&&YY&MM SSI PRIMARY DIAGNOSIS"
DXSEC_SSI&&YY&MM = "&&YY&MM SSI SECONDARY DIAGNOSIS"
DXPRI_DI831&&YY&MM = "&&YY&MM DI PRIMARY DIAGNOSIS"
DXSEC_DI831&&YY&MM = "&&YY&MM DI SECONDARY DIAGNOSIS"
%END;
%DO I=1 %TO &IND;
DODEC&I = "CDR DATE &I"
JUDLVL&I = "LEVEL OF ADJUDICATION &I"
```

## Appendix A.31

JCL/SAS Code: P3133ALL

```
RDT&I = "RESULT OF DETERMINATION &I"
RID&I = "PROGRAM IDENTIFICATION &I"
RB&I = "REGULATION BASIS CODE &I"
RFC&I = "REASON FOR CONTINUATION-CESSATION CODE &I"
DPM&I = "PERMANENT DISABILITY CODE &I"
DOC&I = "DISTRICT OFFICE CODE &I"
DD&I = "DIARY DATE &I"
DT&I = "DIARY TYPE &I"
PD&I = "PRESUMPTIVE DISABILITY DECISION &I"
PDD&I = "PRESUMPTIVE DECISION DATE &I"
VRA&I = "VR ACTION &I"
BFD&I = "BASIS FOR DETERMINATION &I"
EPEDATE&I = "EPE BEGIN DATE &I"
EPETERM&I = "EPE BENEFIT TERMINATION &I"
EPEDENY&I = "EPE REINSTATEMENT DENIED &I"
EPEREIN&I = "EPE REINSTATEMENT ALLOWED &I"
EPESUSP&I = "EPE SUSPENDED AFTER REINSTATEMENT &I"
DR&I = "DIARY REASON &I"

QDDIND&I = "QDD INDICATOR &I"
QDRMVDT&I = "QDD REMOVAL DATE &I"
QDRMVRSN&I = "QDD REMOVAL REASON &I"
QDREINDT&I = "QDD REINSTATEMENT DATE &I"
QDSCORE&I = "QDD SCORING CODE &I"
DSICLAIM&I = "DISABILITY SERVICES IMPROVEMENT CLAIM &I"
%END;
;
OUTPUT OUT1.ALLCDR;

END; /* LAST BIC */
RUN;

PROC FREQ DATA=OUT1.ALLCDR;
TABLES PROGRAM * MIEX0001
PROGRAM * MIEX0101
PROGRAM * MIEX0201
PROGRAM * MIEX0301
PROGRAM * MIEX0401
PROGRAM * MIEX0501
PROGRAM * MIEX0601
;
RUN;

TITLE "PROC CONTENTS ALLCDR";
PROC CONTENTS DATA=OUT1.ALLCDR;
```



## Appendix A.31

JCL/SAS Code: P3133ALL

```
RUN;

PROC PRINT DATA=OUT1.ALLCDR (OBS=100);
TITLE 'CDR DATA ALL';RUN;
PROC CONTENTS; RUN;

/***** START RSTRCTN *****/

*** PRINT RECORDS PRIOR TO PROCESSING IN ORDER TO CHECK THAT ***;
*** OUTPUT IS WHAT WE EXPECT ***;

PROC PRINT DATA=OUT1.ALLCDR(OBS=25);
VAR SSN BFD: DT: PD: DODEC: ;
WHERE DODEC18 NE .;
TITLE "CHECK VARIABLES PRIOR TO PROCESSING";
TITLE2 "RECORDS WITH MORE THAN &REST N OCCURRENCES";
RUN;

PROC PRINT DATA=OUT1.ALLCDR(OBS=25);
VAR SSN BFD: DT: PD: DODEC: ;
TITLE2 " ";
RUN;

DATA OUT2.RSTRCTN(DROP=I J DODEC_START);
SET OUT1.ALLCDR;

LENGTH
%DO I = 1 %TO &REST;
HOLDJUDLVL&I
HOLDRDT&I
HOLDRID&I
HOLDDPM&I
HOLDDT&I
HOLDPD&I
HOLDVRA&I
HOLDBFD&I
HOLDEPEDENY&I
HOLDDR&I
%END; $1;

*INPUT ARRAY;
ARRAY BFD (*) $ BFD1 - BFD&MAXT;
ARRAY DD (*) DD1 - DD&MAXT;
ARRAY DOC (*) $ DOC1 - DOC&MAXT;
```

## Appendix A.31

JCL/SAS Code: P3133ALL

```

ARRAY DPM (*)$ DPM1 - DPM&MAXT;
ARRAY DT (*)$ DT1 - DT&MAXT;
ARRAY EPEDATE (*) EPEDATE1 - EPEDATE&MAXT;
ARRAY EPEDENY (*)$ EPEDENY1 - EPEDENY&MAXT;
ARRAY EPEREIN (*) EPEREIN1 - EPEREIN&MAXT;
ARRAY EPESUSP (*) EPESUSP1 - EPESUSP&MAXT;
ARRAY EPETERM (*) EPETERM1 - EPETERM&MAXT;
ARRAY JUDLVL (*)$ JUDLVL1 - JUDLVL&MAXT;
ARRAY PD (*)$ PD1 - PD&MAXT;
ARRAY PDD (*)$ PDD1 - PDD&MAXT;
ARRAY RB (*)$ RB1 - RB&MAXT;
ARRAY RDT (*)$ RDT1 - RDT&MAXT;
ARRAY RFC (*)$ RFC1 - RFC&MAXT;
ARRAY RID (*)$ RID1 - RID&MAXT;
ARRAY VRA (*)$ VRA1 - VRA&MAXT;
ARRAY DODEC (*) DODEC1 - DODEC&MAXT;
ARRAY DR (*)$ DR1 - DR&MAXT;

ARRAY QDDIND (*)$ QDDIND1 - QDDIND&MAXT;
ARRAY QDRMVDT (*) QDRMVDT1 - QDRMVDT&MAXT;
ARRAY QDRMVRSN (*)$ QDRMVRSN1 - QDRMVRSN&MAXT;
ARRAY QDREINDT (*) QDREINDT1 - QDREINDT&MAXT;
ARRAY QDSCORE (*)$ QDSCORE1 - QDSCORE&MAXT;
ARRAY DSICLAIM (*)$ DSICLAIM1 - DSICLAIM&MAXT;

*OUTPUT ARRAY;
ARRAY HOLDBFD (*)$ HOLDBFD1 - HOLDBFD&REST ;
ARRAY HOLDDD (*) HOLDDD1 - HOLDDD&REST ;
ARRAY HOLDDOC (*)$ HOLDDOC1 - HOLDDOC&REST ;
ARRAY HOLDDPM (*)$ HOLDDPM1 - HOLDDPM&REST ;
ARRAY HOLDDT (*)$ HOLDDT1 - HOLDDT&REST ;
ARRAY HOLDEPEDATE (*) HOLDEPEDATE1 - HOLDEPEDATE&REST ;
ARRAY HOLDEPEDENY (*)$ HOLDEPEDENY1 - HOLDEPEDENY&REST ;
ARRAY HOLDEPEREIN (*) HOLDEPEREIN1 - HOLDEPEREIN&REST ;
ARRAY HOLDEPESUSP (*) HOLDEPESUSP1 - HOLDEPESUSP&REST ;
ARRAY HOLDEPETERM (*) HOLDEPETERM1 - HOLDEPETERM&REST ;
ARRAY HOLDJUDLVL (*)$ HOLDJUDLVL1 - HOLDJUDLVL&REST ;
ARRAY HOLDPD (*)$ HOLDPD1 - HOLDPD&REST ;
ARRAY HOLDPDD (*)$ HOLDPDD1 - HOLDPDD&REST ;
ARRAY HOLDRB (*)$ HOLDRB1 - HOLDRB&REST ;
ARRAY HOLDRDT (*)$ HOLDRDT1 - HOLDRDT&REST ;
ARRAY HOLDRFC (*)$ HOLDRFC1 - HOLDRFC&REST ;
ARRAY HOLDRID (*)$ HOLDRID1 - HOLDRID&REST ;
ARRAY HOLDVRA (*)$ HOLDVRA1 - HOLDVRA&REST ;
ARRAY HOLDDODEC (*) HOLDDODEC1 - HOLDDODEC&REST ;

```

## Appendix A.31

JCL/SAS Code: P3133ALL

```
ARRAY HOLDDR    (*) $    HOLDDR1 - HOLDDR&REST    ;

ARRAY HOLDQDDIND (*) $    HOLDQDDIND1 - HOLDQDDIND&REST ;
ARRAY HOLDQDRMVDT (*)    HOLDQDRMVDT1 - HOLDQDRMVDT&REST ;
ARRAY HOLDQDRMVRSN (*) $    HOLDQDRMVRSN1 - HOLDQDRMVRSN&REST;
ARRAY HOLDQDREINDT (*)    HOLDQDREINDT1 - HOLDQDREINDT&REST;
ARRAY HOLDQDSCORE (*) $    HOLDQDSCORE1 - HOLDQDSCORE&REST ;
ARRAY HOLDDSICLAIM (*) $    HOLDDSICLAIM1 - HOLDDSICLAIM&REST;

DODEC_INDEX=0;
DO I=&MAXT TO 1 BY -1 UNTIL (DODEC_INDEX>0);
  IF DODEC(I) NE '' THEN DODEC_INDEX=I;
END;

%MACRO REST(VAR = );

*STEP 1 IS TO CREATE AN ADDITIONAL VARIABLE THAT IS SET TO THE EARLIEST
OCCURRENCE OF EACH N VARIABLE ;
FIRST&VAR.=&VAR.1;

IF DODEC_INDEX <= &REST THEN DO I=1 TO &REST;
  HOLD&VAR.(I)=&VAR.(I);
END;

* IF DODEC_INDEX > THE RESTRICTION NUMBER THEN SET THE LAST REST NUMBER
OCCURRENCES TO THE HOLD VARIABLES;

IF DODEC_INDEX > &REST THEN DO;
  DODEC_START=DODEC_INDEX-(&REST-1);

  DO I = DODEC_START TO DODEC_INDEX;
    J = I-(DODEC_START-1);
    HOLD&VAR(J) = &VAR.(I);
  END;
END;

DROP
%DO I = 1 %TO &MAXT;
  &&VAR.&I
%END;;

RENAME
%DO I = 1 %TO &REST;
  %PUT RENAMING HOLD&&VAR.&I TO &&VAR.&I;
  HOLD&&VAR.&I=&&VAR.&I
```

## Appendix A.31

JCL/SAS Code: P3133ALL

```
%END;;

%MEND REST;
%REST(VAR=BFD );
%REST(VAR=DD );
%REST(VAR=DOC );
%REST(VAR=DPM );
%REST(VAR=DT );
%REST(VAR=EPEDATE );
%REST(VAR=EPEDENY );
%REST(VAR=EPEREIN );
%REST(VAR=EPESUSP );
%REST(VAR=EPETERM );
%REST(VAR=JUDLVL );
%REST(VAR=PD );
%REST(VAR=PDD );
%REST(VAR=RB );
%REST(VAR=RDT );
%REST(VAR=RFC );
%REST(VAR=RID );
%REST(VAR=VRA );
%REST(VAR=DODEC );
%REST(VAR=DR );

%REST(VAR=QDDIND );
%REST(VAR=QDRMVDT );
%REST(VAR=QDRMVRSN);
%REST(VAR=QDREINDT);
%REST(VAR=QDScore );
%REST(VAR=DSICLAIM);

RUN;

*** PRINT RECORDS AFTER FINAL PROCESSING IN ORDER TO CHECK THAT ***;
*** OUTPUT IS WHAT WE EXPECT ***;

PROC PRINT DATA=OUT2.RSTRCTN(OBS=25);
  VAR SSN BFD: DT: PD: DODEC: QD: DSI;;
  WHERE DODEC_INDEX > &REST;
  TITLE "CHECK VARIABLES AFTER PROCESSING";
  TITLE2 "RECORDS WITH MORE THAN &REST N OCCURRENCES";
RUN;

PROC PRINT DATA=OUT2.RSTRCTN(OBS=25);
  VAR SSN BFD: DT: PD: DODEC: QD: DSI;;
```

Appendix A.31

JCL/SAS Code: P3133ALL

```
TITLE "CHECK VARIABLES AFTER PROCESSING";
TITLE2 " ";
RUN;

PROC CONTENTS DATA=OUT2.RSTRCTN;
  TITLE "PROC CONTENTS RSTRCTN";
RUN;

%MEND P3133ALL;
%P3133ALL;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2  DD SYSOUT=(B,SMTP)
//SYSUT1  DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: SUBMISSION DAF21 TASK05 P3133ALL IS COMPLETE
SUBJECT: SUBMISSION DAF21 TASK05 P3133ALL IS COMPLETE
DAF21 TASK05 P3133ALL SUBMITTED ON <DATE> AT <TIME> IS COMPLETE.
```

## Appendix A.32

JCL/SAS Code: GET2SSI

```

//#6502GSI JOB (11710000,T715,,SAS,,ITC9FL),ALEX-BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#6502
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF21.P831.PRDLIB(GET2SSI)  |
//*  *-----*
//* ATTACH SSN TO ALLCDR RECORDS PRELIMINARY TO MERGING WITH DAF
//* THIS PROGRAM SPLITS THE RESTRCTN FILE BY PROGRAM FOR SSI BENES
//*
//* DATE: 4/14/10
//* UPDATED: 3/23/2022
//* CONTACT ALEX BRYCE
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
// SET REG='64M'
//JS010 EXEC SAS9,
//  WORK='200000,100000'      * SPACE IN BLKS *
//*
//RSTN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.RSTRCTN.SA.V1,DISP=SHR
//SSIF DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIFIND.FL.V1,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.ALLCDSSI.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//SYSIN DD *
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES OBS=MAX;

/* SUBSET SSI DATA TO SSN'S IN SSI FINDER. */
DATA FINDER;
  INFILE SSIF TRUNCOVER;
  INPUT @001 SSN $9.;
RUN;

DATA OUT.SSI;
  MERGE RSTN.RSTRCTN(IN=A WHERE =(PROGRAM IN (2,3)))
    FINDER(IN=F);
  BY SSN;
  IF F AND A;
  IF FIRST.SSN;
RUN;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER

```

Appendix A.32

JCL/SAS Code: GET2SSI

```
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>  
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>  
DATA  
FROM: <ABRYCE@MATHEMATICA-MPR.COM>  
TO: <ABRYCE@MATHEMATICA-MPR.COM>  
SUBJECT: SUBMISSION DAF21 TASK05 GET2SSI IS COMPLETE  
SUBJECT: SUBMISSION DAF21 TASK05 GET2SSI IS COMPLETE  
DAF21 TASK05 GET2SSI SUBMITTED ON 3/23/22 IS COMPLETE.
```

## Appendix A.33

JCL/SAS Code: GET2DI

```

//#6502GDI JOB (11710000,T715,,SAS,,ITC9FL),ALEX-BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF21.P831.PRDLIB(GET2DI) |
//*  *-----*
//* RETRIEVE THE SSN FROM CAN/BIC COMBO FOR T2 RECORDS
//* ATTACH SSN TO ALLCDR RECORDS PRELIMINARY TO MERGING WITH TRF
//* THIS PROGRAM SPLITS THE RSTRCTN FILE BY PROGRAM FOR DIBENES
//*
//* DATE: 4/14/10
//* UPDATED: 3/23/2022
//* CONTACT ALEX BRYCE
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='200000,100000'          * SPACE IN BLKS *
//*
//RSTN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.RSTRCTN.SA.V1,DISP=SHR
//LKDI DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.LINKDI.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.ALLCDRDI.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD *
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES OBS=MAX;

PROC SORT DATA=LKDI.LINKDI OUT=LINKDI;
  BY CAN BIC;
RUN;

/* SPLIT ALLCDR FILE BY PROGRAM */
/* OUTPUT THE CONCURRENTS ON BOTH SSI AND DI DATA SEGMENTS */
/* SUBSET DI DATA TO THE CAN/BIC COMBINATIONS IN DI LINKING FILE */
DATA TEMP1.DI ;
  MERGE RSTN.RSTRCTN(IN=A WHERE = ( PROGRAM IN (1 2))
    RENAME= (SSN=CAN))

```



## Appendix A.33

JCL/SAS Code: GET2DI

```

LINKDI(IN=C);
BY CAN BIC;
IF C AND A;
RUN;

```

```

/* NEW DATA STEP TO ELIMINATE THE DUPLICATE SSN'S PULLED BY CAN DATA
   IN THE 831/33 FILES - THE CAN AND BIC ARE THE IDENTIFIERS IN 831.
   OUR LINKING FILE HAS EACH CAN SSN AND BIC COMBINATIONS WHICH MAY
   REPEAT THE SSN FOR DUAL ELIGIBLES (EG. BENE WITH BIC OF A ON HER
   PRIMARY ACCOUNT WHO HAS A BIC OF W ON HER HUSBAND'S ACCOUNT WHERE
   SHE GETS WIDOW'S BENEFITS. WE WILL TAKE THE FIRST RECORD WHICH
   WILL BE THE PRIMARY IF THERE IS A BIC = A RECORD. IF NOT THEN
   WE WILL TAKE THE FIRST AUXILIARY RECORD. THIS IS THE BEST WE CAN
   DO FOR DUALS */

```

```

/*DIVIDE TEMP.DI IN ORDER TO SUCCEED IN SORTING IT */
%MACRO SORTDAT(BEGIN = ,END = ,RUN = );
PROC SORT DATA = TEMP1.DI(FIRSTOBS = &BEGIN. OBS=&END.)
    OUT=TEMP&RUN..DI;
    BY SSN BIC;
RUN;
%MEND SORTDAT;
%SORTDAT(BEGIN = 1 ,END = 9000000 ,RUN = 2);
%SORTDAT(BEGIN = 9000001 ,END = MAX ,RUN = 3);

```

```

DATA OUT.DI;
    SET TEMP2.DI TEMP3.DI;
    BY SSN BIC;
    /* IF THE CAN = A KEEP THE RECORD - OTHERWISE KEEP THE FIRST
       AUXILIARY */
    IF FIRST.SSN;
RUN;

```

```

//*****
/* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>

```

Appendix A.33

JCL/SAS Code: GET2DI

```
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: SUBMISSION DAF21 TASK05 GET2DI IS COMPLETE
SUBJECT: SUBMISSION DAF21 TASK05 GET2DI IS COMPLETE
DAF21 TASK05 GET2DI SUBMITTED ON 3/23/22 IS COMPLETE.
```

## Appendix A.34

JCL/SAS Code: GET2COM

```

//#6502GCM JOB (11710000,T715,,SAS,,ITC9FL),ALEX-BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.#6502.DAF21.P831.PRDLIB(GET2COM) |
//* *-----*
//* RETRIEVE THE SSN FROM CAN/BIC COMBO FOR T2 RECORDS
//* ATTACH SSN TO ALLCDR RECORDS PRELIMINARY TO MERGING WITH DAF
//* COMBINE SSI AND DI SEGMENTS FROM GET2DI AND GET2SSI
//* DATE: 3/22/17
//* UPDATED: 3/23/2022 BY ALEX BRYCE
//* CONTACT ALEX BRYCE
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
// SET REG='64M'
//JS010 EXEC SAS9,
//      WORK='200000,100000',      * SPACE IN BLKS *
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
//ALDI DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.ALLCDRDI.SA.V1,DISP=SHR
//ALSI DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.ALLCDSSI.SA.V1,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.ALLCDCOM.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD *
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES;
/* COMBINE SSI AND DI SEGMENTS */

DATA OUT.ALLCDCOM (DROP=MIEX0805-MIEX0812 MIEX09: MIEX1:);
MERGE ALSI.SSI ALDI.DI;
BY SSN;
/* THERE ARE RECORDS FOR SOME DAC'S AND WIDOWS IN THE SSI SEGMENT -
   DUPPLICATES OF RECORDS ON THE DI SIDE. IT SEEMS THAT AT SOME
   POINT A RECORD HAS BEEN CREATED IN SSI FOR SOME OF THESE BENES.
   THIS SHOWS UP AS A DUPLICATE BECAUSE THE BIC IS W OR C IN THE
   DI PROGRAM ON 831 AND A IN THE SSI PROGRAM. THESE DUP RECORDS
   HAVE BEEN SCRUTINIZED. IN MANY CASE THEY HAVE DUPLICATE VALUES
   FOR THE VARIABLES. WHEN THESE ARE NOT DUPLICATES THE DI RECORD
   USUALLY HAS MORE COMPLETE DATA. THEREFORE WE HAVE CHANGED THE
   DATA STEP FROM "SET" TO "MERGE" PLACING THE SSI RECORD FIRST
   TO BE OVERWRITTEN BY THE DI RECORD (SECOND) WHEN THE SSN

```

Appendix A.34

JCL/SAS Code: GET2COM

```
    APPEARS IN BOTH FILES. */
RUN;

PROC CONTENTS DATA=OUT.ALLCDCOM; RUN;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2  DD SYSOUT=(B,SMTP)
//SYSUT1  DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: SUBMISSION DAF11 TASK05 GET2COM IS COMPLETE
SUBJECT: SUBMISSION DAF11 TASK05 GET2COM IS COMPLETE
DAF21 TASK05 GET2COM SUBMITTED ON 3/23/21 IS COMPLETE.
```

## Appendix A.35

JCL/SAS Code: SP831833

```
//#6502SPL JOB (12510000,T715,,SAS,,ITC9FL),ALEX-BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//*
//*  *-----*
//*  | TEST THE MACRO SPLIT PROGRAM          |
//*  *-----*
//*
//*
//* CREATED BY: BRITTANY COLIP 4/06/2021
//* MPR PHONE
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*****
// SET REG='64M'
//*
//JS010 EXEC SAS9,
//      WORK='200000,100000', F10
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
//IN1   DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.ALLCDCOM.SA.V1,DISP=SHR
//OUT94 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM94.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT95 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM95.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT96 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM96.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT97 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM97.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT98 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM98.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT99 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM99.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT00 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM00.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT01 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM01.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
```

## Appendix A.35

JCL/SAS Code: SP831833

```
//      UNIT=TSILO
//OUT02 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM02.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT03 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM03.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT04 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM04.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT05 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM05.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT06 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM06.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT07 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM07.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT08 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM08.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT09 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM09.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM10.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT11 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM11.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM12.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT13 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM13.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT14 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM14.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT15 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM15.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM16.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
```

## Appendix A.35

JCL/SAS Code: SP831833

```

//      UNIT=TSILO
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM17.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM18.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM19.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM20.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM21.SA.V1,
//      DISP=(NEW,CATLG,KEEP),
//      UNIT=TSILO
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2021;

* GET VARIABLE LIST FROM INPUT DATASET *;
PROC CONTENTS DATA=IN1.ALLCDCOM NOPRINT
    OUT=TEMP.CONTS(KEEP=NAME);
RUN;

%MACRO SPLITV;

* KEEP ONLY VARIABLES WITHOUT YYYY *;
DATA TEMP.CONTS(DROP=YR MO YYYY);
    SET TEMP.CONTS;
    DO YR=&BEGYR. TO &ENDYR.;
        DO MO=1 TO 12;
            IF MO<=9 THEN YYYY=SUBSTR(STRIIP(YR),3,2) || "0" || STRIP(MO);
            ELSE YYYY=SUBSTR(STRIIP(YR),3,2) || STRIP(MO);

            IF LENGTH(NAME)>=4 THEN DO;
                IF REVERSE(SUBSTR(REVERSE(STRIIP(NAME)),1,4))

```

Appendix A.35

JCL/SAS Code: SP831833

```

        EQ YMMM THEN DELETE;
    END;

    END;
    END;
    RUN;

PROC SORT DATA=TEMP.CONTS NODUPKEY OUT=TEMP1.CONTS;
  BY NAME;
    RUN;

* CREATE MACRO VARIABLE WHICH CONTAINS ALL COMMON VARIABLES *;
PROC SQL;
  SELECT NAME INTO :COMMVAR SEPARATED BY ' '
  FROM TEMP1.CONTS;
QUIT;

%PUT COMMON VARIABLES=&COMMVAR;

%INCLUDE "OPDR.TG.PRD.ETTW.#6502.DAF21.P831.PRDLIB(MSPLIT)";

%OUTP(IN1.ALLCDCOM,OUT,ALLCDCOM,&BEGYR,2008,&COMMVAR);
%OUTP(IN1.ALLCDCOM (DROP=MIEX:),OUT,ALLCDCOM,2009,&ENDYR,&COMMVAR);

%DO YEAR=&BEGYR %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR,3,2);
TITLE "CONTENTS OF &YEAR P831/833 DATA";
PROC CONTENTS DATA=OUT&YR..ALLCDCOM;
    RUN;
    TITLE;
%END;

%MEND SPLITV;
%SPLITV;
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2  DD SYSOUT=(B,SMTP)
//SYSUT1  DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>

```



Appendix A.35

JCL/SAS Code: SP831833

```
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: SUBMISSION DAF21 MACROSPLIT IS COMPLETE
SUBJECT: SUBMISSION DAF21 MACROSPLIT IS COMPLETE
DAF21 TASK MACROSPLIT SUBMITTED ON MM/DD/YY IS COMPLETE.
```

## Appendix A.36

JCL/SAS Code: JCLLOAD

```
//#8047NUM JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR: LAURA KOSA
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DFA20.NUM.PRDLIB(JCLLOAD)
//* PURPOSE: LOAD RAW NUMIDENT DATA INTO SAS
//* CREATED: SERGE LUKASHANETS 5/22/2018
//* UPDATE: MIRIAM CHAPPELKA 4/15/2022
//*
//* E-MAIL: MCHAPPELKA@MATHEMATICA-MPR.COM
//*      SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//MRGPROC PROC
//JS010 EXEC SAS9,
//      WORK='200000,100000',
//      PARM='SYSPARM="'&SEG"'
//*
//***RAW NUMIDENT FINDER RETURN
//IN1   DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.NUMI.FL.V1,DISP=SHR
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//***OUTPUT FILE
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.NUMLOAD&SEG..SA.V1,
//      DISP=(OLD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.NUM.PRDLIB(LOADNUM),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//RUN1  EXEC MRGPROC,SEG=1
//RUN2  EXEC MRGPROC,SEG=2
//RUN3  EXEC MRGPROC,SEG=3
//RUN4  EXEC MRGPROC,SEG=4
//RUN5  EXEC MRGPROC,SEG=5
//RUN6  EXEC MRGPROC,SEG=6
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
```

Appendix A.36

JCL/SAS Code: JCLLOAD

```
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCAHPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: JCLLOAD IS COMPLETE
```

## Appendix A.37

JCL/SAS Code: LOADNUM

```
* EXECUTED BY OPDR.TG.PRD.ETTW.#8047.DAF21.NUM.PRDLIB(JCLLOAD);
* THIS PROGRAM UPLOAD NUMIDENT FLAT FILE INTO MULTIPLE SAS SEGMENTS;

OPTIONS NOCENTER COMPRESS=YES MPRINT MACROGEN OBS=MAX;

%LET SEGOBS=24000000; /*SET 24M FOR EACH SEGMENT*/
%LET LASTOBS=%EVAL(&SEGOBS.*&SYSPARM.);
%LET FIRSTOBS=%EVAL(&LASTOBS.-&SEGOBS.+1);
%PUT PROCESSING SEGMENT &SYSPARM., OBS FROM &FIRSTOBS TO &LASTOBS.;
%MACRO DOIT;

  /* READ IN NUMIDENT DATA */

DATA TEMP.NUM;
  INFILE IN1 TRUNCOVER FIRSTOBS=&FIRSTOBS. OBS=&LASTOBS.;

  /* READ IN ALL RECORDS TYPE */
  INPUT @013 SSN $9.
  @;
  /* REMOVE RECORDS WITH MISSING SSN */
  IF SSN NE ' ';
  INPUT @022 CITIZENSHIP_CODE $1. @;
  INPUT @027 REC $1.
  @;

  /* REMOVE RECORDS WITH OTHER REC TYPE */
  IF REC IN ('G' 'J' 'K' 'O' '1' '2' '4' '5' '9'
            'P' 'S' 'W' 'V' 'D' 'L' 'T' 'E');

  /* READ COMMON DATA FOR CERTAIN RECORD TYPES */
  INPUT @027 ENTLLST $1. /* VAR RENAMED FROM ENTCD */
        @029 CYCDTLST $8. /* VAR RENAMED FROM CYCDTE */
        @037 FRSTNAME $15. /* VAR RENAMED FROM FRSTNAM */
        @053 MIDNAME $15. /* VAR RENAMED FROM MIDNAM */
        @069 LASTNAME $20. /* VAR RENAMED FROM LASTNAM */
        @090 NAMESUF $4. /* VAR RENAMED FROM NAMESUF */
        @097 DOBNUM $8. /* VAR RENAMED FROM DOBR */
  @;

  /* READ NOT COMMON DATA FROM EACH RECORD TYPE */
  IF REC IN ('G' 'J' 'K' 'O' '1' '2' '4' '5' '9'
            'P' 'S' 'W' 'V') THEN DO;
    INPUT @105 NSEX $1. /* VAR RENAMED FROM SEXR */
          @106 NRACE $1. /* VAR RENAMED FROM RACER */
```

## Appendix A.37

JCL/SAS Code: LOADNUM

```
@;
END;
ELSE IF REC IN ('D' 'L') THEN DO;
  INPUT @105 NSEX $1. /* VAR RENAMED FROM SEXR */
  @298 DOD $8. /* VAR RENAMED FROM DODR */
  @;
END;
ELSE IF REC = 'T' THEN DO;
  INPUT @105 NSEX $1. /* VAR RENAMED FROM SEXR */
  @145 DOD $8. /* VAR RENAMED FROM DODR */
  @;
END;
/* RECORD TYPE E HAS NO ADDITIONAL VARS READ IN */

/* NEW VARIABLES IN DAF18 */
INPUT @234 POBSC $2. /*POB-STATE-CONTRY*/
  @236 POBFORID $1. /*POB-FOREIGN-IND*/
;
RUN;

***ORDER THE RECORDS BY SSN FOR SORT LATER;
PROC SORT DATA=TEMP.NUM
  OUT=OUT.NUMLOAD&SYSPARM.
  (RENAME=(CITIZENSHIP_CODE=CITIZEN_NUM));
  BY SSN CYCDTLST;
RUN;

****CHECK CATEGORICAL VALUES;
PROC FREQ DATA=OUT.NUMLOAD&SYSPARM.;
TABLES REC ENTLST NSEX NRACE / LIST MISSING;
TABLES CITIZEN_NUM / LIST MISSING;
RUN;

***PRINT FIRST 50 RECORDS;
PROC PRINT DATA=OUT.NUMLOAD&SYSPARM. (OBS=50);
  TITLE "AFTER SSN SORT NUMLOAD&SYSPARM.";
RUN;

%MEND DOIT;
%DOIT;

ENDSAS;
```

## Appendix A.38

JCL/SAS Code: COMBNUM

```
//#8047COM JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR: LAURA KOSA
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.NUM.PRDLIB(COMBNUM)
//* PURPOSE: COMBINE NUMIDENT FILES FOR ONE MASTER NUMIDENT FILE
//* CREATED: SERGE LUKASHANETS 4/26/2016
//* UPDATE: MIRIAM CHAPPELKA 4/18/2022
//*
//* E-MAIL: MCHAPPELKA@MATHEMATICA-MPR.COM
//*      MIRIAM.CHAPPELKA@SSA.GOV
//*****
// SET REG='64M'
//*
//JS010 EXEC SAS9,
//      WORK='200000,100000',
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
//*SEE INPUT FILES IN THE LIBNAME STATEMENTS
//*** TEMPFILES
//TEMP1 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,15)
//TEMP2 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE),VOL=(,,15)
//*** OUTPUT FILE
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.NUMCOMB.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE),VOL=(,,15)
//*
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT MACROGEN OBS=MAX;
%LET TOTAL_SEG = 6; /* UPDATE THE TOTAL NUMBER OF SEGMENTS */

*READ IN INPUT FILES;
%MACRO DOIT;
  %DO SEG = 1 %TO &TOTAL_SEG.;
    LIBNAME IN&SEG. "OPDR.TG.PRD.ETTW.#8047.DAF21.NUMLOAD&SEG..SA.V1"
      DISP=SHR;
  %END;
```

## Appendix A.38

JCL/SAS Code: COMBNUM

```
*CREATE THE TRANSACTION FILE CONTAINS THE NEW INFORMATION;
*DATA IS ORDERED BY EACH SSN AND THEIR CYCLE DATES;
DATA TEMP1.TRANNUM;
  SET
  %DO SEG = 1 %TO &TOTAL_SEG.;
    IN&SEG..NUMLOAD&SEG.(DROP = REC)
  %END;
;
  BY SSN CYCDTLST;
  IF NRACE = '0' THEN NRACE = ' ';
RUN;
%MEND DOIT;
%DOIT;

*CREATE THE MASTER DATA SET THAT CONTAINS THE LIST OF SSNS;
PROC SORT DATA = TEMP1.TRANNUM OUT = TEMP2.MASTNUM NODUPKEY;
BY SSN;
RUN;

*TEMP1.TRANNUM (TRANSACTION FILE) IS SORTED BY SSN AND CYCLE DATE;
*USE THE UPDATE STATEMENT TO READ OBS FROM THE TRANSACTION DATA SET;
*AND UPDATE CORRESPONDING OBS FROM THE MASTER DATA SET WITH;
*THE MOST RECENT POPULATED RECORDS;

DATA OUT.NUMCOMB;
  UPDATE TEMP2.MASTNUM
  TEMP1.TRANNUM
;
  BY SSN;
RUN;

PROC PRINT DATA=OUT.NUMCOMB (OBS=50); RUN;

PROC CONTENTS DATA=OUT.NUMCOMB; RUN;
PROC FREQ DATA=OUT.NUMCOMB;
TABLES CITIZEN_NUM
/ LIST MISSING;
RUN;
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
```

Appendix A.38

JCL/SAS Code: COMBNUM

```
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: COMBNUM IS COMPLETE
```



## Appendix A.39

JCL/SAS Code: JCLSSR1

```

//#6266SR1 JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*   RESTART=JS020.SAS9
//*****
//*
//*   *-----*
//*   ! OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(JCLSSR1)|
//*   *-----*
//*
//*   1. READ IN EACH SEGMENT OF SSI LONGITUDINAL FILE
//*   2. SORT BY SSN AND RCD_EST AND OUTPUT STEP 1
//* JCL CALLS THE PROGRAM FOR STEP 1 RLONGST1
//* CONTACT SERGE LUKASHANETS
//* E-MAIL SLUKASHANETS@MATHEMATICA-MPR.COM
//*
//* POS PARAMETER ALLOWS PROGRAM TO SKIP YEARS 74-93
//* POS PARAMETER SHOULD BE CALCULATED BASED ON LAYOUT:
//* - POSITION OF FIRST MONTHLY VAR + NUM MONTHLY VARS =
//*   LENGTH OF EACH MONTH
//* - USING KNOWN LENGTH OF MONTH, CALCULATE POSITION OF
//*   JAN OF DAF YEAR
//* - POS OF JAN OF DAF YEAR - POS OF FIRST MONTHLY VAR =
//*   THE &POS PARAMETER
//* POS PARAMETER WILL BE ADDED TO POSITION OF EACH VAR LISTED
//* IN THE LAYOUT WHEN READING IN EACH VAR
//*****
//*
//MRGPROC PROC
//STEP1 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="&SEG&BY&EY&POS"'
//*
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.SSRLONG.FL.V1,
//     DISP=SHR
//IN2  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIFIND.SA.V1,
//     DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO
//OUT1  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.RLONG&SEG..SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(SSR1),
//     DISP=(SHR,PASS,KEEP)
//*****

```

Appendix A.39

JCL/SAS Code: JCLSSR1

```

/* SENDING EMAIL FROM HOST USING IEBGENER IF PROGRAM ABENDS
/******
// IF (ABEND=TRUE | RC > 0 ) THEN
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 JCLSSR1 IS ABENDED
// ENDIF
// PEND
/* *****
//RUN1 EXEC MRGPROC,SEG=01,POS=11040,BY=1994,EY=2021
//RUN2 EXEC MRGPROC,SEG=02,POS=11040,BY=1994,EY=2021
//RUN3 EXEC MRGPROC,SEG=03,POS=11040,BY=1994,EY=2021
//RUN4 EXEC MRGPROC,SEG=04,POS=11040,BY=1994,EY=2021
//RUN5 EXEC MRGPROC,SEG=05,POS=11040,BY=1994,EY=2021
//RUN6 EXEC MRGPROC,SEG=06,POS=11040,BY=1994,EY=2021
//RUN7 EXEC MRGPROC,SEG=07,POS=11040,BY=1994,EY=2021
//RUN8 EXEC MRGPROC,SEG=08,POS=11040,BY=1994,EY=2021
//RUN9 EXEC MRGPROC,SEG=09,POS=11040,BY=1994,EY=2021
//RUN10 EXEC MRGPROC,SEG=10,POS=11040,BY=1994,EY=2021
//RUN11 EXEC MRGPROC,SEG=11,POS=11040,BY=1994,EY=2021
//RUN12 EXEC MRGPROC,SEG=12,POS=11040,BY=1994,EY=2021
//RUN13 EXEC MRGPROC,SEG=13,POS=11040,BY=1994,EY=2021
//RUN14 EXEC MRGPROC,SEG=14,POS=11040,BY=1994,EY=2021
//RUN15 EXEC MRGPROC,SEG=15,POS=11040,BY=1994,EY=2021
//RUN16 EXEC MRGPROC,SEG=16,POS=11040,BY=1994,EY=2021
//RUN17 EXEC MRGPROC,SEG=17,POS=11040,BY=1994,EY=2021
//RUN18 EXEC MRGPROC,SEG=18,POS=11040,BY=1994,EY=2021
//RUN19 EXEC MRGPROC,SEG=19,POS=11040,BY=1994,EY=2021
/******
/* SENDING EMAIL FROM HOST USING IEBGENER
/******
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *

```

Appendix A.39

JCL/SAS Code: JCLSSR1

```
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 SSR1 IS COMPLETE
DAF21 SSR1 FOR READING IN SSR SEGMENTS HAS ENDED
```

Appendix A.40

JCL/SAS Code: SSR1

```

*****;
* FILENAME: OPDR.TG.PRD.ETTW.#7429.DAF18.SSR.PRDLIB(SSR1);
* PROGRAMMER: BRITTANY COLIP;
* PURPOSE: TO READ IN THE LONGITUDINAL FILE '
* CREATED: 3/11/13;
* UPDATED: 4/19/17 - ADELE RIZZUTO;
* UPDATED: 4/30/18 - ALEX BRYCE ;
* UPDATED: 3/27/19 - BRITTANY COLIP;
*****;
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES OBS=MAX MPRINT MACROGEN;

* EXTRACT INFO FOR MACRO VARIABLES FROM THE SYSPARM SET IN JCL;
* FIRST 2 POSITIONS ARE SEGMENT;
%LET SASSEG=%SUBSTR(&SYSPARM.,1,2);

* NEXT 4 ARE BEGINNING YEAR;
%LET BEGYR=%SUBSTR(&SYSPARM.,3,4);

* NEXT 4 ARE END YEAR;
%LET ENDYR=%SUBSTR(&SYSPARM.,7,4);

* REMAINING IS POSITION;
%LET SASPOS=%SUBSTR(&SYSPARM.,11);

* READ INTO 4M FOR EACH SEGEMENT;
%LET OBS=%EVAL(4000000*&SASSEG.);

* FIRST OBS TO BE READ FOR SEG;
%LET FIRSTOBS=%EVAL(&OBS.-4000000+1);

%PUT PROCESSING SEGMENT &SASSEG., OBS &FIRSTOBS. TO &OBS.;
%PUT USING POSITION &SASPOS. TO READ IN YMMM VARIABLES;

* ASSIGN MACROS VARIABLES TO HANDLE TIME SERIES DATA;
* MACRO REVISED FOR DAF13 IN THIS PROGRAM (DAF14 MAY;
* USE IN ALL PROGRAMS)IN ADDED A PUT SHOWING MAPPING;
* OF MONTHS 1 TO N TO EACH ASSOCIATED YMMM ;
%MACRO GETSSILF;

*INITIATE MONTH COUNTER, MOCNTER, TO 1;
%LET MOCNTER=1;

```

## Appendix A.40

JCL/SAS Code: SSR1

```

* LOOPING THROUGH EACH YEAR SPECIFIED ABOVE;
%DO YEAR=&BEGYR. %TO &ENDYR.;
* SETTING YR MACRO VAR TO LAST TWO DIGITS OF EACH YEAR;
%LET YR=%SUBSTR(&YEAR.,3,2);

* LOOP THROUGH MONTHS 1 TO 12 AND INCREMENT MONTH COUNTER;
%DO M=1 %TO 12;
  %LET M=%SYSFUNC(PUTN(&M.,Z2.));
  %LET YY&MOCNTER.=%EVAL(&YR.)%EVAL(&M.);
  %LET MOCNTER=%EVAL(&MOCNTER.+1);
%END;
%END;

* SET TOTAL MONTHS, TOTMOS, EQUAL TO MOCNTER - 1 TO GET;
* TOTAL MONTHS IN TIME SERIES;
%LET TOTMOS=%EVAL(&MOCNTER.-1);

* USED TO PRINT MAPPING TO LOG - CHECK DURING CODE REVIEW;
%DO MM=1 % TO &TOTMOS.;
  %PUT &MM. MAPS TO YYMM, &&YY&MM.;
%END;
%PUT TOTAL MONTHS = &TOTMOS.;

*****
* READ IN EVERY RECORD IN THIS SECTION OF THE FILE ;
* AFTER ALL SECTIONS ARE READ IN THEN PROCESS THE MULTIPLE RECORDS;
* SORT EACH SECTION BY SSN AND FILE SELECTION DATE TO ENSURE;
* PROPER GROUPING OF MULTIPLE RECORDS IN THE FINAL COMBINE STEP;
*****

DATA TEMP1.LONG;

INFILE IN1 TRUNCOVER FIRSTOBS=&FIRSTOBS. OBS=&OBS.;

INPUT
  /*READING IN FIXED VARIABLES (NON-YYMM)*/
  @010 SSN   $9.   /* PAN SOCIAL SECURITY NUMBER */
  @023 MFT   $2.   /* MFT MASTER FILE TYPE CODE */
  @;
  /* MARY BARBOUR INSTRUCTION - IF MFT = X THEN STOP READING RECORDS*/

IF SSN = "000000000" OR SUBSTR(MFT,1,1)= 'X' THEN DELETE;

/* CONTINUE READING RECORDS */

```

## Appendix A.40

JCL/SAS Code: SSR1

```

INPUT
@001 HUN   $9.   /* HOUSED UNDER NUMBER */
@019 TOA   $2.   /* TOA TYPE OF ACTION */
@021 COMP_STA $2. /* COMP-STAT-TOA COMP. STATUS TYPE OF ACTION */
@025 START_RD ?? YMMN6. /* START-RD EARLIEST COMPUTATION DATE */
@041 RCD_EST ?? YMMDD8. /* RCD-EST-JD RECORD ESTABLISHMENT DATE */
@049 DOBSSR ?? YMMDD8. /* BIRTH-JD DATE OF BIRTH */
@057 DODSSR ?? YMMDD8. /* DEATH-JD DATE OF DEATH */
@065 LAF    $2.   /* LAF MBR LEDGER ACCOUNT FILE CODE */
@067 CURSTAT $3. /* CURSTAT CURRENT PAYMENT STATUS */
@070 SEXSSR $1.   /* SEX SEX */
@071 RACE   $1.   /* RACE RACE */
@072 ELG_RD ?? YMMN6. /* ELG-RD DATE OF CURRENT ELIGIBILITY */
@078 APP_RD ?? YMMDD8. /* APPL-JD APPLICATION DATE (CURRENT) */
@086 _8080_JD ?? YMMDD8. /* 8080-JD 8080 DATE */
@112 CLM_FIL ?? YMMDD8. /* CLM-FIL-JD DATE CLAIM FILED */
@120 DISPAYCD $1. /* DISABILITY PAYMENT CODE */
@136 DIB_DIG $4. /* DIB-DIG PRIMARY DISABILITY DIAGNOSTIC CODE */
@140 DIB_DIG2 $4. /* DIB-DIG2 SECONDARY DISABILITY DIAGNOSTIC CODE */
@144 DIB_MDR $1. /* DIB-MDR MEDICAL DIARY REASON */
@145 DIB_DPM $1. /* PERMANENT DISABILITY INDICATOR */
@146 PDSCC $6. /* PDSCC RESIDENCE STATE/COUNTY/DO CODE */
@152 PDZIP $5. /* PDZIP PAYEE'S ZIP CODE */
@157 PDZIP6_9 $4. /* PDZIP6-9 PAYEE'S ZIP CODE SUFFIX */
@161 DOC_SSI_MR $3. /* DO DISTRICT OFFICE CODE */
@172 REPPAYTP $3. /* REPPAYTYP TYPE OF PAYEE CODE */
@191 SEL_DATE ?? YMMN6. /* DATE OF SELECTION */
@197 FIRST_PA ?? YMMDD8. /* FIRST-PAY-DTE FIRST PAYMENT DATE (THIS RECORD)*/
@217 STOP_RD ?? YMMN6. /* STOP-RD STOP DATE */
@223 LANGSSR $2. /* LANGUAGE PREFERENCE WRITTEN */
@;

/* READING IN YYMM VARIABLES*/
POS=&SASPOS.;
/* LOOP THROUGH MONTHS */
%DO MM=1 %TO &TOTMOS.;
INPUT /*POS:VARNAME DESCRIPTION (NEW) */
@POS+301 CMTHRD&&YY&MM. ?? YMMN6. /*CMTH-RD MONTH OF COMPUTATION*/
@POS+307 PSTA&&YY&MM. $3. /*PSTAT PAYMENT STATUS CODE */
@POS+310 LIVF&&YY&MM. $1. /*LIVF LIVING ARRANGEMENT CODE*/
@POS+311 SCON&&YY&MM. $1. /*STCONCATM STATE CONCURR ELIG IND*/
@POS+313 EICM&&YY&MM. 4. /*EINCM CHARGEABLE EARNED */
/* INCOME AMOUNT */
@POS+317 UINC&&YY&MM. 4. /*UINCM CHARGEABLE UNEARNED */
/* INCOME AMOUNT */

```

Appendix A.40

JCL/SAS Code: SSR1

```

@POS+321  FAMT&&YY&MM. 5.  /*FEDAMT FEDERAL ASSISTANCE */
                /* AMOUNT */
@POS+326  SAMT&&YY&MM. 5.  /*SUPAMT CURRENT AMOUNT OF */
                /* STATE SUPPLEMENT*/
@POS+331  MTST&&YY&MM. $1. /*MEDTEST MEDICAL AND SOCIAL */
                /* SERVICE INCOME TEST*/
@POS+332  CURCOMP&&YY&MM. $1. /*CUR-COMP CURRENT COMPOSITION*/
                /* CODE (NEW DAF13) */
@POS+333  FPMT&&YY&MM. 6.  /*FEDPMT FEDERAL MONEY AMOUNT */
                /* (PAYMENT) */
@POS+339  SPMT&&YY&MM. 6.  /*STATPMT STATE SUPPL AMOUNT */
                /*(PAYMENT) */

@;
POS=POS+46;
%END;

DROP POS;
RUN;

* SORT EACH SEGMENT BY SSN AND RECORD ESTAB DATE FOR COMBINING;
PROC SORT DATA=TEMP1.LONG OUT=TEMP2.LONG; BY SSN RCD_EST; RUN;

DATA OUT1.LONG;
MERGE TEMP2.LONG (IN=T)
      IN2.SSIFIND (IN=F WHERE=(SSN NE '000000000'))
;
BY SSN;
IF T AND F;
RUN;

PROC CONTENTS DATA = OUT1.LONG; RUN;
%MEND GETSSILF;
%GETSSILF;

```

## Appendix A.41

JCL/SAS Code: JCLSSR1B

```

//#6266S1B JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*   RESTART=JS020.SAS9
//*****
//*
//*   *-----*
//*   ! OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(JCLSSR1B)|
//*   *-----*
//*
//*   1. READ IN EACH SEGMENT OF SSI LONGITUDINAL FILE (74-93)
//*   2. SORT BY SSN AND RCD_EST AND OUTPUT STEP 1
//* JCL CALLS THE PROGRAM FOR STEP 1 RLONGST1
//* CONTACT SERGE LUKASHANETS
//* E-MAIL SLUKASHANETS@MATHEMATICA-MPR.COM
//*
//* POS PARAMETER ALLOWS PROGRAM TO READ YEARS 74-93
//* POS PARAMETER SHOULD BE 0 FOR READING IN YEARS 74-93
//*****
//*
//MRGPROC PROC
//STEP1 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="&SEG&BY&EY&POS"'
//*
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.SSRLONG.FL.V1,
//      DISP=SHR
//IN2  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIFIND.SA.V1,
//      DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO
//OUT1  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.RLONGB&SEG..SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(SSR1),
//      DISP=(SHR,PASS,KEEP)
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER IF PROGRAM ABENDS
//*****
//  IF (ABEND=TRUE | RC > 0 ) THEN
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *

```



Appendix A.41

JCL/SAS Code: JCLSSR1B

```

HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 JCLSSR1B IS ABENDED
SUBJECT: DAF21 JCLSSR1B IS ABENDED
// ENDIF
// PEND
//*****
//*RUN1 EXEC MRGPROC,SEG=01,POS=0000,BY=1974,EY=1993
//*RUN2 EXEC MRGPROC,SEG=02,POS=0000,BY=1974,EY=1993
//*RUN3 EXEC MRGPROC,SEG=03,POS=0000,BY=1974,EY=1993
//*RUN4 EXEC MRGPROC,SEG=04,POS=0000,BY=1974,EY=1993
//*RUN5 EXEC MRGPROC,SEG=05,POS=0000,BY=1974,EY=1993
//*RUN6 EXEC MRGPROC,SEG=06,POS=0000,BY=1974,EY=1993
//*RUN7 EXEC MRGPROC,SEG=07,POS=0000,BY=1974,EY=1993
//*RUN8 EXEC MRGPROC,SEG=08,POS=0000,BY=1974,EY=1993
//*RUN9 EXEC MRGPROC,SEG=09,POS=0000,BY=1974,EY=1993
//*RUN10 EXEC MRGPROC,SEG=10,POS=0000,BY=1974,EY=1993
//*RUN11 EXEC MRGPROC,SEG=11,POS=0000,BY=1974,EY=1993
//*RUN12 EXEC MRGPROC,SEG=12,POS=0000,BY=1974,EY=1993
//*RUN13 EXEC MRGPROC,SEG=13,POS=0000,BY=1974,EY=1993
//RUN14 EXEC MRGPROC,SEG=14,POS=0000,BY=1974,EY=1993
//RUN15 EXEC MRGPROC,SEG=15,POS=0000,BY=1974,EY=1993
//RUN16 EXEC MRGPROC,SEG=16,POS=0000,BY=1974,EY=1993
//RUN17 EXEC MRGPROC,SEG=17,POS=0000,BY=1974,EY=1993
//RUN18 EXEC MRGPROC,SEG=18,POS=0000,BY=1974,EY=1993
//RUN19 EXEC MRGPROC,SEG=19,POS=0000,BY=1974,EY=1993
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 SSR1 FOR 74-93 IS COMPLETE

```

Appendix A.41

JCL/SAS Code: JCLSSR1B

DAF21 SSR1 FOR READING IN SSR SEGMENTS FOR YEARS 74-93 HAS ENDED

## Appendix A.42

JCL/SAS Code: JCLSSR2

```

//#6266SS2 JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(JCLSSR2) |
//*  *-----*
//*
//*  1. COMBINE AND PROCESS LONGITUDINAL FILE
//*  2. OUTPUT UPDATED STEP2 FILE
//*
//* SEGMENTS WERE PRODUCED IN STEP 1
//* THIS IS A NEW PROGRAM COMBINING JCL STATEMENTS FOR ALL SEGMENTS
//* 1974-1993 VS. 1994-DAFYR WILL HAVE DIFFERENT INCLUDE FILE
//*****
//*
//JS010 EXEC SAS9,
//  WORK='120000,60000',
//* MAKE UPDATES TO THIS SYSPARM
//* THE FORMAT IS: BEGYR, ENDYR, YOURID, NUMSEG, NODE, DAFYR
//* DO NOT UPDATE BEGYR
//* UPDATE ENDYR TO THIS DAFYR
//* UPDATE YOUR ID
//* UPDATE NUMSEG (THE NUMBER OF SEGMENTS IN SSR1 PROGRAMS)
//* DO NOT UPDATE NODE (THIS IS A NODE FOR THE INPUT FILES)
//* UPDATE THE TWO-DIGIT DAFYR (USED IN THE NODE FOR INPUT FILES)
//  PARM='SYSPARM="1994,2021,#6266,19,RLONG,21"'
//*
//LONGF DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,,20)
//LONGV DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,,10),
//  UNIT=TSILO
//LF1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,,10),
//  UNIT=TSILO
//LF2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,,20)
//LV1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,,10),
//  UNIT=TSILO
//LV2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,,10),
//  UNIT=TSILO
//LV3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,,10),
//  UNIT=TSILO
//LFUPD DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,,20)

```

Appendix A.42

JCL/SAS Code: JCLSSR2

```
//LVUPD DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//      UNIT=TSILO
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(SSR2),
//      DISP=(SHR,PASS,KEEP)
//
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 SSR2 (PART 1) IS COMPLETE
DAF21 SSR2 (PART 1) HAS FINISHED RUNNING
```

## Appendix A.43

JCL/SAS Code: JCLSSR2B

```

//#6266SS2 JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(JCLSSR2) |
//*  *-----*
//*
//*  1. COMBINE AND PROCESS LONGITUDINAL FILE
//*  2. OUTPUT UPDATED STEP2 FILE
//*
//* SEGMENTS WERE PRODUCED IN STEP 1
//* THIS IS A NEW PROGRAM COMBINING JCL STATEMENTS FOR ALL SEGMENTS
//* 1974-1993 VS. 1994-DAFYR WILL HAVE DIFFERENT INCLUDE FILE
//*****
//*
//JS010 EXEC SAS9,
//  WORK='120000,60000',
//* MAKE UPDATES TO THIS SYSPARM
//* THE FORMAT IS: BEGYR, ENDYR, YOURID, NUMSEG, NODE, DAFYR
//* DO NOT UPDATE BEGYR
//* UPDATE ENDYR TO THIS DAFYR
//* UPDATE YOUR ID
//* UPDATE NUMSEG (THE NUMBER OF SEGMENTS IN SSR1 PROGRAMS)
//* DO NOT UPDATE NODE (THIS IS A NODE FOR THE INPUT FILES)
//* UPDATE THE TWO-DIGIT DAFYR (USED IN THE NODE FOR INPUT FILES)
//*  PARM='SYSPARM="1994,2020,#6502,19,RLONG,20"'
//*
//*LONGF DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//*  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*LONGV DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//*  UNIT=TSILO
//*LF1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//*  UNIT=TSILO
//*LF2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//*  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*LV1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//*  UNIT=TSILO
//*LV2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//*  UNIT=TSILO
//*LV3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//*  UNIT=TSILO
//*LFUPD DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//*  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)

```

## Appendix A.43

JCL/SAS Code: JCLSSR2B

```
//*LVUPD DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//*    UNIT=TSILO
//*OUT1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.SSR2.SA.V1,
//*    DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*    UNIT=TSILO
//*SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20.SSR.PRDLIB(SSR2),
//*    DISP=(SHR,PASS,KEEP)
//*
//*JS020 EXEC SAS9,
//*    WORK='120000,60000',
//* MAKE UPDATES TO THIS SYSPARM
//* THE FORMAT IS: BEGYR, ENDYR, YOURID, NUMSEG, NODE, DAFYR
//* DO NOT UPDATE BEGYR
//* DO NOT UPDATE ENDYR
//* UPDATE YOUR ID
//* UPDATE NUMSEG (THE NUMBER OF SEGMENTS IN SSR1 PROGRAMS)
//* DO NOT UPDATE NODE (THIS IS A NODE FOR THE INPUT FILES)
//* UPDATE THE TWO-DIGIT DAFYR (USED IN THE NODE FOR INPUT FILES)
//    PARM='SYSPARM="1974,1993,#6266,19,RLONGB,21"'
//*
//LONGF DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//    SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//LONGV DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//    UNIT=TSILO
//LF1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//    UNIT=TSILO
//LF2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//    SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//LV1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//    UNIT=TSILO
//LV2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//    UNIT=TSILO
//LV3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//    UNIT=TSILO
//LFUPD DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//    SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//LVUPD DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//    UNIT=TSILO
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2B.SA.V1,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//    UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(SSR2),
//    DISP=(SHR,PASS,KEEP)
//*
```

Appendix A.43

JCL/SAS Code: JCLSSR2B

```
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
DATA  
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 SSR2 (PART 2) IS COMPLETE  
DAF21 SSR2 (PART 2) HAS FINISHED RUNNING
```

## Appendix A.44

JCL/SAS Code: SSR2

```
* SSR2 - THIS IS AN INCLUDE FILE TO PROCESS 1994 - CURRENT DAF FILES;
%LET BEGYR=%SCAN(%QUOTE(&SYSPARM),1,%STR(,));
%LET ENDYR=%SCAN(%QUOTE(&SYSPARM),2,%STR(,));
%LET YOURID=%SCAN(%QUOTE(&SYSPARM),3,%STR(,));
%LET NUMSEG=%SCAN(%QUOTE(&SYSPARM),4,%STR(,));
%LET NODE=%SCAN(%QUOTE(&SYSPARM),5,%STR(,));
%LET DAFYR=%SCAN(%QUOTE(&SYSPARM),6,%STR(,));

* BEGIN PROCESSING;
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY;
OPTIONS OBS=MAX MPRINT SYMBOLGEN;
/* NOTE: CHANGE THE ENDYR AND ENDMN PARAMETER BEFORE EACH RUN */

/* ASSIGN MACROS VARIABLES TO HANDLE TIME SERIES DATA */

%MACRO PROCSSILF;
/*INITIATE MONTH COUNTER, MOCNTER, TO 1*/
%LET MOCNTER=1;
/*LOOPING THROUGH EACH YEAR SPECIFIED ABOVE*/
%DO YEAR=&BEGYR. %TO &ENDYR.;
/*SETTING YR MACRO VAR TO LAST TWO DIGITS OF EACH YEAR*/
%LET YR=%SUBSTR(&YEAR.,3,2);

/*LOOP THROUGH MONTHS 1 TO 12 AND INCREMENT MONTH COUNTER*/
%DO M=1 %TO 12;
%LET M=%SYSFUNC(PUTN(&M.,Z2.));
%LET YY&MOCNTER=%EVAL(&YR)%EVAL(&M);
%LET MOCNTER=%EVAL(&MOCNTER+1);
%END;
%END;
/*SET TOTAL MONTHS, TOTMOS, EQUAL TO MOCNTER - 1 TO GET
TOTAL MONTHS IN TIME SERIES*/
%LET TOTMOS=%EVAL(&MOCNTER-1);

/*USED TO PRING MAPPING TO LOG - CHECK DURING CODE REVIEW*/
%DO MM=1 %TO &TOTMOS;
%PUT &MM. MAPS TO YYMM, &&YY&MM.;
%END;
%PUT TOTAL MONTHS = &TOTMOS.;
```



## Appendix A.44

JCL/SAS Code: SSR2

```

/*****
WE NEED TO READ THE SSI LONG FILES TO CAPTURE THE HISTORY FIELDS
WE NEED TO SCAN ALL THE RECORDS FOR VALID VALUES INSTEAD OF JUST
KEEPING THE LAST RECORD AS WE DID PREVIOUSLY.
THE LAST RECORD MAY HAVE HISTORY FIELDS BLANKED OUT
*****/

/*SPLIT THE VARIABLE FILE TO UPDATE RECORDS WITH NONMISSING DATA */

%DO SEG = 1 %TO &NUMSEG; /*UPDATE THE TOTAL NUMBER OF SEGMENTS */
  %LET SEG=%SYSFUNC(PUTN(&SEG,Z2.));
  LIBNAME IN&SEG.
  "OPDR.TG.PRD.ETTW.&YOURID..DAF&DAFYR.P.&NODE.&SEG..SA.V1"
  DISP=SHR;
%END;

DATA LONGF.ALLF(DROP=
  %DO MM=1 %TO &TOTMOS;
  PSTA&&YY&MM.  LIVF&&YY&MM.  SCON&&YY&MM.  EICM&&YY&MM.
  UINC&&YY&MM.  FAMT&&YY&MM.  SAMT&&YY&MM.  MTST&&YY&MM.
  FPMT&&YY&MM.  SPMT&&YY&MM.
  CMTHRD&&YY&MM.  CURCOMP&&YY&MM.
  %END;)
LONGV.ALLV(KEEP=SSN RCD_EST /* VARIABLE PART */
%DO MM=1 %TO &TOTMOS;
  PSTA&&YY&MM.  LIVF&&YY&MM.  SCON&&YY&MM.  EICM&&YY&MM.
  UINC&&YY&MM.  FAMT&&YY&MM.  SAMT&&YY&MM.  MTST&&YY&MM.
  FPMT&&YY&MM.  SPMT&&YY&MM.
  CMTHRD&&YY&MM.  CURCOMP&&YY&MM.
%END;);
SET

%DO SEG = 1 %TO &NUMSEG; /* UPDATE THE TOTAL NUMBER OF SEGMENTS */
  %LET SEG=%SYSFUNC(PUTN(&SEG,Z2.));
  IN&SEG..LONG
%END;;
BY SSN RCD_EST;
RUN;

PROC PRINT DATA=LONGF.ALLF (OBS=20); TITLE 'BEFORE UPDATE'; RUN;
PROC PRINT DATA=LONGV.ALLV (OBS=50); RUN;

/*CREATE MIN AND MAX VALUES FROM SOME OF THE FIXED PART VARIABLES */

```

## Appendix A.44

JCL/SAS Code: SSR2

```
PROC MEANS NOPRINT DATA=LONGF.ALLF;
VAR START_RD RCD_EST ELG_RD APP_RD _8080_JD CLM_FIL FIRST_PA STOP_RD;
BY SSN;
OUTPUT OUT=LONGF.MAXMIN(DROP=_:);
    MIN=MINSTRRD MINRCDST SSI_MIN_ELG MINAPPRD MIN_80JD MINCLMFL
    MINFRSTP MINSTPRD
    MAX=MAXSTRRD MAXRCDST MAXELGRD MAXAPPRD MAX_80JD MAXCLMFL
    MAXFRSTP MAXSTPRD
;
RUN;

/*SPLIT THE FIXED PART FILE TO UPDATE RECS WITH NONMISSING DATA */
/*FOR FIXED DATA WE DO NOT HAVE TO EXAMINE INTERMEDIATE RECORDS */
/*WE WANT THE CURRENT VALUE IN THE FIELDS          */

DATA LF1.FIXEDA LF2.FIXEDB;
SET LONGF.ALLF;
BY SSN RCD_EST;
IF FIRST.SSN THEN OUTPUT LF1.FIXEDA;
ELSE OUTPUT LF2.FIXEDB;
RUN;

/* RUN THE UPDATE PROCEDURE */

DATA LFUPD.LONGF;
UPDATE LF1.FIXEDA LF2.FIXEDB;
BY SSN;
RUN;

/* CONVERSATIONS WITH MARY BARBOUR HAVE REVEALED THAT SSA WILL
TRY TO SAVE SPACE BY BLANKING AND ZEROING OUT HISTORY FIELDS
WHEN A NEW PERIOD OF ELIGIBILITY IS ESTABLISHED FOLLOWING THE
END OF AN OLD PERIOD OF ELIGIBILITY (CREATING A NEW RECORD).
THEREFORE WE CAN NOT TAKE THE HISTORY VALUES FROM THE LAST RECORD
AS PREVIOUSLY BUT MUST SCAN ALL THE RECORDS USING AN UPDATE
PROCEDURE. FOR CHARACTER VALUES, A BLANK ON A SUBSEQUENT RECORD
(WHERE THERE HAD BEEN A VALUE ON A PREVIOUS RECORD) WILL NOT
OVERWRITE THE VALUE USING THE UPDATE PROCEDURE. HOWEVER, FOR
NUMERIC VALUES, SSA WILL ZERO OUT THE FIELD AND UPDATE WILL
OVERWRITE A VALUE WITH THE ZERO. AS A RESULT WE WILL BLANK OUT
THESE ZEROS SO AS NOT TO OVERWRITE THE FIELD.
WITH REFERENCE TO THE ACTUAL STATE AND FEDERAL PAYMENT FIELDS
MARY HAS ALSO INDICATED THAT THERE MAY BE DOLLAR ADJUSTMENTS TO A
GIVEN MONTHLY PAYMENT ON A SUBSEQUENT RECORD. FOR EXAMPLE,
```

## Appendix A.44

JCL/SAS Code: SSR2

FDP0309 (FED PAYMENT IN SEPT 2003) MAY SHOW PAYMENTS ON 2 RECORDS  
THE TOTAL ACTUAL PAYMENT BEING THE SUM OF THESE. THIS IS HANDLED  
IN THE FOLLOWING PROCEDURE \*/

```
DATA LV1.VARA LV2.VARB;
SET LONGV.ALLV; BY SSN RCD_EST;

/* ADD UP THE PAYMENT AMOUNTS ACROSS RECORDS */
/* IN THE CASE WHERE A FIELD IS ADJUSTED, THE INCREMENT
WILL APPEAR ACROSS RECORDS AND THE UPDATE PROCEDURE
WILL TAKE THE SUM INSTEAD OF OVERWRITING ONE AMOUNT
WITH A SUBSEQUENT AMOUNT */

ARRAY FPMT (*) %DO MM=1 %TO &TOTMOS; FPMT&&YY&MM. %END;;
ARRAY SPMT (*) %DO MM=1 %TO &TOTMOS; SPMT&&YY&MM. %END;;
/*CREATING TEMPORARY VARIABLES, IMPLIED BY 'T'*/
ARRAY TFPMT (*) %DO MM=1 %TO &TOTMOS; TFPMT&&YY&MM. %END;;
ARRAY TSPMT (*) %DO MM=1 %TO &TOTMOS; TSPMT&&YY&MM. %END;;

RETAIN %DO MM=1 %TO &TOTMOS; TFPMT&&YY&MM. %END;;
RETAIN %DO MM=1 %TO &TOTMOS; TSPMT&&YY&MM. %END;;

IF FIRST.SSN THEN DO I=1 TO DIM(FPMT);
TFPMT(I) = FPMT(I);
TSPMT(I) = SPMT(I);
END;
ELSE DO I=1 TO DIM(FPMT);
IF FPMT(I) >= 0 THEN TFPMT(I) + FPMT(I);
IF SPMT(I) >= 0 THEN TSPMT(I) + SPMT(I);
END;

DROP FPMT: SPMT;;

/* RENAME PAYMENT VARIABLES BACK TO ORIGINAL NAME */

%DO MM = 1 %TO &TOTMOS;
RENAME TFPMT&&YY&MM. = FPMT&&YY&MM.;
RENAME TSPMT&&YY&MM. = SPMT&&YY&MM.;
%END;

IF FIRST.SSN THEN OUTPUT LV1.VARA;
OUTPUT LV2.VARB;
RUN;

DATA LV3.VARB2;
```

## Appendix A.44

JCL/SAS Code: SSR2

```
SET LV2.VARB; BY SSN RCD_EST;
```

```
/* ON THE LAST RECORD IF IT IS A TERMINATION RECORD THE WHOLE  
HISTORY MAY BE BLANKED AND ZEROED */
```

```
ARRAY FAMT (*) %DO MM=1 %TO &TOTMOS; FAMT&&YY&MM. %END;;  
ARRAY SAMT (*) %DO MM=1 %TO &TOTMOS; SAMT&&YY&MM. %END;;  
ARRAY EICM (*) %DO MM=1 %TO &TOTMOS; EICM&&YY&MM. %END;;  
ARRAY UINC (*) %DO MM=1 %TO &TOTMOS; UINC&&YY&MM. %END;;  
ARRAY FPMT (*) %DO MM=1 %TO &TOTMOS; FPMT&&YY&MM. %END;;  
ARRAY SPMT (*) %DO MM=1 %TO &TOTMOS; SPMT&&YY&MM. %END;;  
ARRAY CURCOMP (*) %DO MM=1 %TO &TOTMOS; CURCOMP&&YY&MM. %END;;
```

```
/*NEW DAF13-CURCOMPYYMM RANGES FROM 0 TO 8. 0 IS INVALID SO SET TO  
MISSING. WITH 0'S RECODING TO MISSING, DO NOT HAVE TO WORRY ABOUT  
0'S OVERWRITING VALID 1-8 VALUES DURING THE UPDATE PROCEDURE BELOW*/
```

```
DO I=1 TO DIM(CURCOMP);  
IF CURCOMP(I)="0" THEN CURCOMP(I)=" ";  
END; DROP I;
```

```
/* FLAG THE NUMERIC FIELDS IF GT 0 ON ANY RECORD -  
NEVER SET THEM TO ZERO IF THEY ARE ZEROED OUT SUBSEQUENTLY */  
/*DELETED '3' WHICH SET NUMERIC LENGTH TO 3*/
```

```
ARRAY FAF (&TOTMOS) _TEMPORARY_;  
ARRAY SAF (&TOTMOS) _TEMPORARY_;  
ARRAY EIF (&TOTMOS) _TEMPORARY_;  
ARRAY UIF (&TOTMOS) _TEMPORARY_;  
ARRAY FPF (&TOTMOS) _TEMPORARY_;  
ARRAY SPF (&TOTMOS) _TEMPORARY_;
```

```
/* INITIALIZE FLAGS TO ZERO */
```

```
IF FIRST.SSN THEN DO I = 1 TO DIM(FAF);
```

```
FAF(I) = 0;  
SAF(I) = 0;  
EIF(I) = 0;  
UIF(I) = 0;  
FPF(I) = 0;  
SPF(I) = 0;  
END;
```

```
DO I = 1 TO DIM(FAMT);  
IF FAMT(I) > 0 THEN FAF(I) = 1;
```

## Appendix A.44

JCL/SAS Code: SSR2

```
IF SAMT(I) > 0 THEN SAF(I) = 1;
IF EICM(I) > 0 THEN EIF(I) = 1;
IF UINC(I) > 0 THEN UIF(I) = 1;
IF FPMT(I) > 0 THEN FPF(I) = 1;
IF SPMT(I) > 0 THEN SPF(I) = 1;
END;

/* IF THE AMOUNTS HAVE BEEN ZEROED OUT ON SUBSEQUENT RECORDS
WITH A NEW START DATE -
SET THE ZERO TO MISSING IF THE MONTHLY AMOUNT HAS BEEN
FLAGGED AS EVER BEING > 0. THIS PREVENTS THE UPDATE
PROCEDURE FROM OVERWRITING A RECORD WITH ZERO.
FOR THE ALPHANUMERIC FIELDS A BLANK WILL NOT OVERWRITE
A PREVIOUS VALID VALUE */

DO I = 1 TO DIM(FAMT);
IF FAMT(I) = 0 AND FAF(I) = 1 THEN FAMT(I) = .;
IF SAMT(I) = 0 AND SAF(I) = 1 THEN SAMT(I) = .;
IF EICM(I) = 0 AND EIF(I) = 1 THEN EICM(I) = .;
IF UINC(I) = 0 AND UIF(I) = 1 THEN UINC(I) = .;
IF FPMT(I) = 0 AND FPF(I) = 1 THEN FPMT(I) = .;
IF SPMT(I) = 0 AND SPF(I) = 1 THEN SPMT(I) = .;
END;
DROP I;
RUN;

/* RUN THE UPDATE PROCEDURE */

DATA LVUPD.LONGV;
UPDATE LV1.VARA LV3.VARB2;
BY SSN;
/*CAN WE MOVE THIS DROP STATEMENT TO THE DATASTEP ABOVE?*/

ARRAY CURCOMP (*) %DO MM=1 %TO &TOTMOS; CURCOMP&&YY&MM. %END;;

/*NEW DAF13-REVERSING RECODING PERFORMED ABOVE*/
DO I=1 TO DIM(CURCOMP);
IF CURCOMP(I)=" " THEN CURCOMP(I)="0";
END;
DROP I;
RUN;

DATA OUT1.LONG;
MERGE LFUPD.LONGF(IN=A)
LONGF.MAXMIN
```

## Appendix A.44

JCL/SAS Code: SSR2

```
LVUPD.LONGV
;
BY SSN;
IF A;
/*ADDING SSIYY*/
/*TWO DIMENSIONAL ARRAY - YEAR BY MONTH*/
ARRAY PSTAYMM(&BEGYR.:&ENDYR.,1:12) %DO YEAR=&BEGYR. %TO &ENDYR.;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      PSTA&YR.01-PSTA&YR.12
      %END;;
ARRAY SSIYY (&BEGYR.:&ENDYR.) %DO YEAR=&BEGYR. %TO &ENDYR.;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      SSI&YR.
      %END;;

DO YR=&BEGYR. TO &ENDYR.;
  SSIYY(YR) = 0; *SET INITIAL SSIYY VALUE TO BE 0;
  DO MO=1 TO 12 UNTIL(SSIIYY(YR)=1);
    IF PSTAYMM(YR,MO) IN ("C01","M01","M02") THEN SSIYY(YR)=1;
  END;
END;

DROP MO;
RUN;

PROC PRINT DATA=OUT1.LONG (OBS=20); TITLE 'AFTER UPDATE'; RUN;
RUN;
PROC CONTENTS;
RUN;

%MEND PROCSSILF;
%PROCSSILF;
```

## Appendix A.45

JCL/SAS Code: STKSTUDM

```
//#6266STK JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT: DAF20 CONSTRUCTION
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(STKSTUDM)
//*
//* PURPOSE: STACK CORRECTED SSR ADDITIONAL VARIABLES FILE
//*
//* PROGRAMMER: SERGE LUKASHANETS 5/23/2022
//* E-MAIL SLUKASHANETS@MATHEMATICA-MPR.COM
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//IN1  DD DSN=AIS.P1252.ADVRAS.AVR21.D2203.FILE1A,DISP=SHR
//IN2  DD DSN=AIS.P1252.ADVRAS.AVR21.D2203.FILE2A,DISP=SHR
//IN3  DD DSN=AIS.P1252.ADVRAS.AVR21.D2203.FILE3A,DISP=SHR
//IN4  DD DSN=AIS.P1252.ADVRAS.AVR21.D2203.FILE4A,DISP=SHR
//*TEMP DD DSN=&&TEMP,
//*  DISP=(NEW,DELETE,DELETE),
//*  UNIT=TSILO
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.STKSTD.SA.V1,
//  DISP=(MOD,CATLG,DELETE),
//  UNIT=TSILO
//SYSIN DD *

%LET ENDYR=2021;

OPTIONS OBS=MAX MACROGEN MPRINT;

DATA OUT.SSIEXT;

**** STACK THE DATA FILES ****;
SET IN1.ADVR IN2.ADVR IN3.ADVR IN4.ADVR;

* KEEP ONLY COMPUTATION MONTHS IN THE DAF;
IF '01JAN1994'D<=CMTH_CCYMM<="31DEC&ENDYR."D;

**** REMOVE FORMATS ****;
FORMAT _ALL_;

**** DROP TWO VARIABLES THAT TERSALEE CREATED ****;
```

Appendix A.45

JCL/SAS Code: STKSTUDM

```
DROP RCD_EST_CCYY CMTH_CCYY;

**** RENAME VARIABLES TO BE CONSISTENT WITH DAF ****;
RENAME
  PAN=SSN
  RCDNO=SEQ
  RCD_EST_CCYYMMDD=RCD_EST
  CMTH_CCYYMM=CMTHMO
;

RUN;

PROC CONTENTS DATA=OUT.SSIEXT; RUN;

PROC FREQ DATA=OUT.SSIEXT;
TABLES STUDM RCD_EST DEEMCD/LIST MISSING;
RUN;

PROC PRINT DATA=OUT.SSIEXT (OBS=10);
WHERE RANUNI(1100) < 0.1;
RUN;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 STKSTUDM IS COMPLETE
DAF21 STKSTUDM PROGRAM HAS ENDED
```



## Appendix A.46

JCL/SAS Code: SORTSSR

```

//#6266SRT JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(SORTSSR)
//*  *-----*
//*
//* 1. DIVIDE SSIEXT INTO SEGMENTS FOR SORTING
//* CONTACT SERGE LUKASHANETS
//* E-MAIL SLUKASHANETS@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000'          * SPACE IN BLKS *
//*
//*SSIEXT DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIEXT.SA.V1,DISP=SHR
//SSIEXT DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.STKSTD.SA.V1,DISP=SHR
//*
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP5 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP6 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP7 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP8 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP9 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP10 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP11 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP12 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP13 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP14 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIEXTS.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//*
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=BINARY OBS=MAX;

%MACRO SORTDAT(BEGOBS= ,ENDOBS= ,RUN= );
PROC SORT DATA = SSIEXT.SSIEXT(FIRSTOBS=&BEGOBS. OBS=&ENDOBS.)
  OUT = TEMP&RUN..SSIEXTS;
  BY SSN RCD_EST CMTHMO;

```

## Appendix A.46

JCL/SAS Code: SORTSSR

```
RUN;
%MEND SORTDAT;

%SORTDAT(BEGOBS = 1,      ENDOBS = 200000000, RUN = 1);
%SORTDAT(BEGOBS = 200000001, ENDOBS = 400000000, RUN = 2);
%SORTDAT(BEGOBS = 400000001, ENDOBS = 600000000, RUN = 3);
%SORTDAT(BEGOBS = 600000001, ENDOBS = 800000000, RUN = 4);
%SORTDAT(BEGOBS = 800000001, ENDOBS = 1000000000, RUN = 5);
%SORTDAT(BEGOBS = 1000000001, ENDOBS = 1200000000, RUN = 6);
%SORTDAT(BEGOBS = 1200000001, ENDOBS = 1400000000, RUN = 7);
%SORTDAT(BEGOBS = 1400000001, ENDOBS = 1600000000, RUN = 8);
%SORTDAT(BEGOBS = 1600000001, ENDOBS = 1800000000, RUN = 9);
%SORTDAT(BEGOBS = 1800000001, ENDOBS = 2000000000, RUN = 10);
%SORTDAT(BEGOBS = 2000000001, ENDOBS = 2200000000, RUN = 11);
%SORTDAT(BEGOBS = 2200000001, ENDOBS = 2400000000, RUN = 12);
%SORTDAT(BEGOBS = 2400000001, ENDOBS = 2600000000, RUN = 13);
%SORTDAT(BEGOBS = 2600000001, ENDOBS = MAX,      RUN = 14);

DATA OUT.SSIEXTS;
  SET TEMP1.SSIEXTS
    TEMP2.SSIEXTS
    TEMP3.SSIEXTS
    TEMP4.SSIEXTS
    TEMP5.SSIEXTS
    TEMP6.SSIEXTS
    TEMP7.SSIEXTS
    TEMP8.SSIEXTS
    TEMP9.SSIEXTS
    TEMP10.SSIEXTS
    TEMP11.SSIEXTS
    TEMP12.SSIEXTS
    TEMP13.SSIEXTS
    TEMP14.SSIEXTS
  ;
  BY SSN RCD_EST CMTHMO;
RUN;

PROC CONTENTS DATA=OUT.SSIEXTS; RUN;

PROC PRINT DATA=OUT.SSIEXTS (OBS = 20);
WHERE RANUNI(1100) < 0.1;
RUN;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
```

Appendix A.46

JCL/SAS Code: SORTSSR

```
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
DATA  
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 SORTSSR ARE COMPLETE
```

Appendix A.47

JCL/SAS Code: D2PRSSR

```
//#6266D2P JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT: DAF20 CONSTRUCTION
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(D2PRSSR)
//*
//* PURPOSE: READ SSR EXTRA VARIABLES FILE
//*
//* PROGRAMMER: SERGE LUKASHANETS
//* E-MAIL SLUKASHANETS@MATHEMATICA-MPR.COM
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//SSIEXTS DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIEXTS.SA.V1,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSREXT.SA.V1,
//  DISP=(MOD,CATLG,DELETE),UNIT=TSILO
//SYSIN DD *

OPTIONS OBS=MAX MACROGEN MPRINT;

%LET ENDYR=2021;/*CHANGE AS NEEDED*/

%MACRO DOIT;

DATA OUT.SSREXT;
  SET SSIEXTS.SSIEXTS;
  BY SSN;

  * SET UP OUTPUT ARRAYS;
  ARRAY DEEMINC (1994:&ENDYR,1:12) %DO YEAR=1994 %TO &ENDYR;
          %LET YR=%SUBSTR(&YEAR.,3,2);
          DEEMINC&YR.01-DEEMINC&YR.12
  %END;;
  ARRAY A_DEEMCD (1994:&ENDYR,1:12) $ %DO YEAR=1994 %TO &ENDYR;
          %LET YR=%SUBSTR(&YEAR.,3,2);
          DEEMCD&YR.01-DEEMCD&YR.12
  %END;;
  ARRAY A_BMF (1994:&ENDYR,1:12) %DO YEAR=1994 %TO &ENDYR;
          %LET YR=%SUBSTR(&YEAR.,3,2);
          BMF&YR.01-BMF&YR.12
  %END;;
```

## Appendix A.47

JCL/SAS Code: D2PRSSR

```
ARRAY A_FCI (1994:&ENDYR,1:12) %DO YEAR=1994 %TO &ENDYR;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      FCI&YR.01-FCI&YR.12
      %END;;
ARRAY A_PROFAC (1994:&ENDYR,1:12) %DO YEAR=1994 %TO &ENDYR;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      PROFAC&YR.01-PROFAC&YR.12
      %END;;
ARRAY A_STUDM (1994:&ENDYR.,1:12) $ %DO YEAR=1994 %TO &ENDYR;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      STUDM&YR.01-STUDM&YR.12
      %END;;
```

```
* KEEP YYMM VARIABLES ACROSS OBS WITH THE SAME SSN;
RETAIN DEEMINC: DEEMCD: BMF: FCI: PROFAC: STUDM;;
```

```
* INITIALIZE RETAINED VARIABLES WHEN AT THE FIRST OB WITH THIS SSN;
```

```
ARRAY NUMVARS (*) %DO YEAR=1994 %TO &ENDYR;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      DEEMINC&YR.01-DEEMINC&YR.12
      BMF&YR.01-BMF&YR.12
      FCI&YR.01-FCI&YR.12
      PROFAC&YR.01-PROFAC&YR.12
      %END;;
ARRAY CHARVARS ($) %DO YEAR=1994 %TO &ENDYR;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      DEEMCD&YR.01-DEEMCD&YR.12
      STUDM&YR.01-STUDM&YR.12
      %END;;
```

```
IF FIRST.SSN THEN DO;
  DO I=1 TO DIM(NUMVARS);
    NUMVARS(I)=.;
  END;
  DO I=1 TO DIM(CHARVARS);
    CHARVARS(I)='';
  END;
END;
```

```
* SET YYMM VARIABLES;
```

```
* EACH OB REPRESENTS AN SSN, RECORD ESTABLISHMENT DATE, AND
COMPUTATION MONTH COMBINATION UNIQUELY. SO WE CAN USE THE VALUE
OF COMPUTATION MONTH TO ASSIGN THE YYMM VALUES. WHEN TWO
COMPUTATION MONTHS OVERLAP FROM ONE SSN-RECORD ESTABLISHMENT DATE
COMBINATION THE DATA FROM THE LATER ESTABLISHED RECORDS OVERLAYS
THAT FROM THE EARLIER ESTABLISHED RECORDS BECAUSE OF THE RETAIN
```

Appendix A.47

JCL/SAS Code: D2PRSSR

```
STATEMENT USED ABOVE;
```

```
IF CMTHMO NE . THEN DO;  
  DEEMINC(YEAR(CMTHMO),MONTH(CMTHMO))=DEEMAMT;  
  A_DEEMCD(YEAR(CMTHMO),MONTH(CMTHMO))=DEEMCD;  
  A_BMF(YEAR(CMTHMO),MONTH(CMTHMO))=BMF;  
  A_FCI(YEAR(CMTHMO),MONTH(CMTHMO))=FCI;  
  A_PROFAC(YEAR(CMTHMO),MONTH(CMTHMO))=PROFAC;  
  A_STUDM(YEAR(CMTHMO),MONTH(CMTHMO))=STUDM;  
END;
```

```
* LAST OB FOR THIS SSN THEN OUTPUT;  
IF LAST.SSN THEN OUTPUT;
```

```
RUN;
```

```
PROC CONTENTS DATA=OUT.SSREXT;  
RUN;
```

```
%MEND DOIT;  
%DOIT;
```

```
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>  
DATA  
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 D2PRSSR ARE COMPLETE
```

## Appendix A.48

JCL/SAS Code: SSIELIG

```

//#6266ELG JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(SSIELIG)
//*  *-----*
//*
//* PURPOSE:TO CREATE TWO NEW VARIABLES FOR EARLIEST SSI
//*   ELIGIBILITY DATE AND EARLIEST ADULT SSI
//*   ELIGIBILITY DATE AND MERGE ON ADDITIONAL VARS
//*   AND BUILD PSTA_MR AND PSTA_MR_DATE
//* CONTACT SERGE LUKASHANETS
//* E-MAIL SLUKASHANETS@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//      WORK='120000,60000'          * SPACE IN BLKS *
//*
//SSR  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2.SA.V1,DISP=SHR
//SSRB DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2B.SA.V1,DISP=SHR
//EXT  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSREXT.SA.V1,DISP=SHR
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIELIG.SA.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TEMP DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//      VOL=(,,10),UNIT=TSILO
//SYSIN DD *

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS = MAX MPRINT
MACROGEN ;

%LET DAFYR=2021; /*CURRENT DAF YEAR*/
%LET BEGYR=1994;
%LET ENDYR=2021; /* CURRENT DAF YEAR */

%MACRO DOIT;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

DATA TEMP.SSIELIG (DROP = PSTA7: PSTA8:
                  PSTA90: PSTA91: PSTA92: PSTA93: );

MERGE SSRB.LONG (KEEP=SSN PSTA: DOBSSR DODSSR)

```

## Appendix A.48

JCL/SAS Code: SSIELIG

```

        SSR.LONG (IN=A)
        EXT.SSREXT(IN=B);

BY SSN;
IF A;

ARRAY PSTA (1974:&DAFYR.,1:12) %DO YEAR=1974 %TO &DAFYR.;
        %LET YR=%SUBSTR(&YEAR.,3,2);
        PSTA&YR.01-PSTA&YR.12
        %END;;

* SET EARLIEST SSI ELIGIBILITY DATE;

STARTLOOP=MAX("01JAN1974"D ,DOBSSR);
STOPLOOP=MIN("31DEC&DAFYR."D,DODSSR);

DO YR=YEAR(STARTLOOP) TO YEAR(STOPLOOP) UNTIL(SSIELIG_FIRST NE .);
  IF YR=YEAR(STARTLOOP) THEN STARTMO=MONTH(STARTLOOP);
  ELSE STARTMO=1;
  IF YR=YEAR(STOPLOOP) THEN ENDMO=MONTH(STOPLOOP);
  ELSE ENDMO=12;

  DO MO=STARTMO TO ENDMO UNTIL(SSIELIG_FIRST NE .);
    IF PSTA(YR,MO) IN ('C01','M01','M02')
      THEN SSIELIG_FIRST=MDY(MO,1,YR);
  END;
END;

* SET EARLIEST SSI ADULT ELIGIBILITY DATE;
_18YR = SUM(YEAR(DOBSSR),18);
_18MO = MONTH(DOBSSR);
_18BDAY = MDY(_18MO,1,_18YR);
STARTSALOOP=MAX(_18BDAY, SSIELIG_FIRST, "01JAN1974"D);

* ONLY COMPUTE IF BENE BECOMES AN ADULT BEFORE END OF DAF YEAR;
IF _18BDAY LE MDY(12,31,&DAFYR.) THEN DO;
DO YR=YEAR(STARTSALOOP) TO YEAR(STOPLOOP) UNTIL(SSIELIG_ADULT NE .);
  IF YR=YEAR(STARTSALOOP) THEN STARTMO=MONTH(STARTSALOOP);
  ELSE STARTMO=1;
  IF YR=YEAR(STOPLOOP) THEN ENDMO=MONTH(STOPLOOP);
  ELSE ENDMO=12;

  DO MO=STARTMO TO ENDMO UNTIL(SSIELIG_ADULT NE .);

  IF PSTA(YR,MO) IN ('C01','M01','M02') AND

```



## Appendix A.48

JCL/SAS Code: SSIELIG

```
        YRDIF(DOBSSR,MDY(MO,1,YR),'AGE')>=18
        THEN SSIELIG_ADULT=MDY(MO,1,YR);

    END;
END;
END;

IF SSIELIG_ADULT AND SSIELIG_FIRST THEN
DATE_ERROR=(SSIELIG_ADULT < SSIELIG_FIRST);

*BUILD PSTA_MR ANALOGOUS TO TASK 8 LAF_MR;
ARRAY PSTA_ALL (*) $ %DO MM=1 %TO &TOTMOS; PSTA&&YY&MM. %END;;
LENGTH PSTA_MR $3. _NM $4.;
PSTA_MR = "";

DO I = DIM(PSTA_ALL) TO 1 BY -1 WHILE (PSTA_MR="");
    IF PSTA_ALL(I) NE "" THEN DO;
        PSTA_MR = PSTA_ALL(I);
        _NM = SUBSTR(VNAME(PSTA_ALL(I)),5,4);
        _YR = INPUT(SUBSTR(_NM,1,2),BEST.);
        IF _YR > 90 THEN _YR1=1900 + _YR;
        ELSE IF _YR > .Z THEN _YR1=2000 + _YR;
        _MN = INPUT(SUBSTR(_NM,3,2),BEST.);
        PSTA_MR_DATE = MDY(_MN,1,_YR1);
    END;
END;

RUN;

*CHECK THAT SSIELIG_ADULT IS NEVER BEFORE SSIELIG_FIRST;
PROC FREQ DATA=TEMP.SSIELIG;
    TABLES DATE_ERROR/MISSING;
RUN;

*CHECK PSTA_MR CONSTRUCTION;
PROC FREQ DATA=TEMP.SSIELIG;
    TABLES PSTA_MR / LIST MISSING;
    TABLES PSTA_MR_DATE / LIST MISSING;
    FORMAT PSTA_MR_DATE YYMMD.;
RUN;

PROC PRINT DATA=TEMP.SSIELIG (OBS=100); WHERE _YR = 18;
    VAR PSTA18: PSTA17: PSTA_;;
RUN;
```

Appendix A.48

JCL/SAS Code: SSIELIG

```
*CHECK DATE CONSTRUCTION;
PROC PRINT DATA=TEMP.SSIELIG (OBS=100);
VAR DOBSSR _18YR _18MO _18BDAY SSIELIG_FIRST STARTSALOOP
    SSIELIG_ADULT;
FORMAT DOBSSR _18BDAY SSIELIG_FIRST STARTSALOOP SSIELIG_ADULT
    YMMMD.;
RUN;

DATA OUT1.SSIELIG (DROP= STARTLOOP STARTMO STARTSALOOP STOPLOOP
    I_MN_NM_YR_YR1 MO YR _18YR _18MO _18BDAY);
SET TEMP.SSIELIG;
LABEL
    PSTA_MR="SSI PAYMENT STATUS, MOST RECENT"
    PSTA_MR_DATE="DATE OF MOST RECENT SSI PAYMENT STATUS"
;
RUN;

PROC CONTENTS DATA=OUT1.SSIELIG;
RUN;

%MEND DOIT;
%DOIT;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: SLUKASHANETS@MATHEMATICA-MPR.COM
TO: SLUKASHANETS@MATHEMATICA-MPR.COM
SUBJECT: SSIELIG (DAF21)
SSIELIG HAS FINISHED RUNNING
```

## Appendix A.49

JCL/SAS Code: JCLYRSP

```
//#6266MSP JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//*
//*  *-----*
//*  | CALL THE MACRO SPLIT PROGRAM          |
//*  *-----*
//*
//*
//* CREATED BY: SERGE LUKASHANETS, 5/26/2016
//* MODIFIED BY: SERGE LUKASHANETS 05/21/2022
//* E-MAIL SLUKASHANETS@MATHEMATICA-MPR.COM
//*****
// SET REG='64M'
//*
//JS010 EXEC SAS9,
//      WORK='200000,100000', F10
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
//***DATA FOR SPLIT
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIELIG.SA.V1,DISP=SHR
//***OUT YEARLY FILES
//OUT94 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1994.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT95 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1995.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT96 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1996.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT97 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1997.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT98 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1998.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT99 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1999.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT00 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2000.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
```

## Appendix A.49

JCL/SAS Code: JCLYRSP

```
//OUT01 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2001.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT02 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2002.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT03 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2003.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT04 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2004.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT05 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2005.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT06 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2006.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT07 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2007.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT08 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2008.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT09 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2009.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2010.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT11 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2011.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2012.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT13 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2013.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT14 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2014.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT15 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2015.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
```

## Appendix A.49

JCL/SAS Code: JCLYRSP

```
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2016.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2017.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2018.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2019.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2020.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2021.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT OBS=MAX;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(MSPLIT)";

%OUTP(IN1.SSIELIG,OUT,SSR,1994,2021,
      SSN MFT SSI: PSTA_MR PSTA_MR_DATE);

PROC CONTENTS DATA=OUT94.SSR;RUN;
PROC CONTENTS DATA=OUT95.SSR;RUN;
PROC CONTENTS DATA=OUT96.SSR;RUN;
PROC CONTENTS DATA=OUT97.SSR;RUN;
PROC CONTENTS DATA=OUT98.SSR;RUN;
PROC CONTENTS DATA=OUT99.SSR;RUN;
PROC CONTENTS DATA=OUT00.SSR;RUN;
PROC CONTENTS DATA=OUT01.SSR;RUN;
PROC CONTENTS DATA=OUT02.SSR;RUN;
PROC CONTENTS DATA=OUT03.SSR;RUN;
PROC CONTENTS DATA=OUT04.SSR;RUN;
PROC CONTENTS DATA=OUT05.SSR;RUN;
PROC CONTENTS DATA=OUT06.SSR;RUN;
PROC CONTENTS DATA=OUT07.SSR;RUN;
PROC CONTENTS DATA=OUT08.SSR;RUN;
PROC CONTENTS DATA=OUT09.SSR;RUN;
PROC CONTENTS DATA=OUT10.SSR;RUN;
PROC CONTENTS DATA=OUT11.SSR;RUN;
PROC CONTENTS DATA=OUT12.SSR;RUN;
```

Appendix A.49

JCL/SAS Code: JCLYRSP

```
PROC CONTENTS DATA=OUT13.SSR;RUN;
PROC CONTENTS DATA=OUT14.SSR;RUN;
PROC CONTENTS DATA=OUT15.SSR;RUN;
PROC CONTENTS DATA=OUT16.SSR;RUN;
PROC CONTENTS DATA=OUT17.SSR;RUN;
PROC CONTENTS DATA=OUT18.SSR;RUN;
PROC CONTENTS DATA=OUT19.SSR;RUN;
PROC CONTENTS DATA=OUT20.SSR;RUN;
PROC CONTENTS DATA=OUT21.SSR;RUN;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 JCLYRSP FOR SSR IS COMPLETE
SSR SPLIT TO ANNUAL SSRS ARE COMPLETE.
```

## Appendix A.50

JCL/SAS Code: SSRERNLD

```
//#6266SEL JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT: DAF19 CONSTRUCTION (50214)
//* PROJECT DIRECTOR: JODY SCHIMMEL HYDE
//* PROGRAM: OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(SSRERNLD)
//*
//* PURPOSE: READ SSR EARNINGS FOR DCF CORRECTIONS
//*
//* PROGRAMMER: SERGE LUKASHANETS 05/26/2022
//* E-MAIL SLUKASHANETS@MATHEMATICA-MPR.COM
//* MODIFIED:
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//EARN DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.SSRERN.FL.V1,DISP=SHR
//TEMP DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP5 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSREARN1.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *

OPTIONS OBS=MAX MACROGEN MPRINT;

%MACRO DOIT;

*READ IN SSR EARNINGS USING SSR RECORD LAYOUT;
DATA TEMP.SSREARN (COMPRESS=YES);
  INFILE EARN;
  INPUT
    @01 HUN $CHAR9.
    @10 PAN $CHAR9.
    @19 MFT $CHAR2.
    @21 TOA $CHAR2.
    @23 RIC $CHAR1.
    @24 RCDNO IBR1.
    @25 RCD_EST_CCYYMMDD ?? YYMMDD8.
```

## Appendix A.50

JCL/SAS Code: SSRERNLD

```
@33 IESTART ?? $CHAR6.
@39 IESTOP ?? $CHAR6.
@45 IETYP $CHAR1.
@46 IEAMT ?? S370FPD4.3
@50 IEFRQ $CHAR1.
@51 IEVAR $CHAR1.
;
FORMAT
RCD_EST_CCYYMMDD MMDDYY10.
IEAMT DOLLAR11.2
RCDNO $HEX2.
;
RCD_EST_CCYY = YEAR(RCD_EST_CCYYMMDD);
LAPSE = IESTOP - IESTART;
RUN;

*QA;
PROC FREQ DATA = TEMP.SSREARN;
TITLE 'SSREARN QA ON IETYP IESTART IESTOP RCD_EST_CCYY';
TABLES RCD_EST_CCYY / LIST MISSING;
TABLES IETYP / LIST MISSING;
TABLES IESTART / LIST MISSING;
TABLES IESTOP / LIST MISSING;
RUN;

PROC FREQ DATA=TEMP.SSREARN;
TITLE 'FREQ OF IEA LAPSE AND IEFRQ';
TABLES LAPSE * IEFRQ / LIST MISSING;
RUN;

PROC FORMAT;
VALUE AMT .='MISSING'
      0='0 IEAMT'
      LOW-<0='NEGATIVE IEAMT'
      0<-HIGH='POSITIVE IEAMT'
      OTHER='OTHER';
RUN;

PROC MEANS DATA=TEMP.SSREARN;
TITLE 'MEANS OF IEAMT';
VAR IEAMT;
RUN;

PROC FREQ DATA=TEMP.SSREARN;
TITLE 'FREQ OF IEAMT AND IETYP';
```



## Appendix A.50

JCL/SAS Code: SSRERNLD

```
TABLES IEAMT * IETYP / LIST MISSING;
FORMAT IEAMT AMT.;
RUN;

*CONFIRM NUMBER OF DISTINCT SSNS;
PROC SQL;
CREATE TABLE NEW AS
SELECT COUNT(DISTINCT(PAN)) AS SSN_COUNT
FROM TEMP.SSREARN;
QUIT;

PROC PRINT; TITLE 'NUMBER OF DISTINCT SSNS'; RUN;

%MEND DOIT;
%DOIT;

/*SORT SSREARN DATA*/
%MACRO SORTDAT(BEGOBS= ,ENDOBS= ,RUN= );
PROC SORT DATA = TEMP.SSREARN (FIRSTOBS=&BEGOBS. OBS=&ENDOBS.)
    OUT = TEMP&RUN..SSREXTS;
    BY PAN RCD_EST_CCYYMMDD;
RUN;
%MEND SORTDAT;

%SORTDAT(BEGOBS = 1,      ENDOBS = 50000000, RUN = 1);
%SORTDAT(BEGOBS = 50000001, ENDOBS = 100000000, RUN = 2);
%SORTDAT(BEGOBS = 1000000001, ENDOBS = 150000000, RUN = 3);
%SORTDAT(BEGOBS = 1500000001, ENDOBS = 200000000, RUN = 4);
%SORTDAT(BEGOBS = 2000000001, ENDOBS = MAX, RUN = 5);

DATA OUT.SSREARN;
    SET TEMP1.SSREXTS (RENAME=(PAN=SSN RCD_EST_CCYYMMDD = RCD_EST))
        TEMP2.SSREXTS (RENAME=(PAN=SSN RCD_EST_CCYYMMDD = RCD_EST))
        TEMP3.SSREXTS (RENAME=(PAN=SSN RCD_EST_CCYYMMDD = RCD_EST))
        TEMP4.SSREXTS (RENAME=(PAN=SSN RCD_EST_CCYYMMDD = RCD_EST))
        TEMP5.SSREXTS (RENAME=(PAN=SSN RCD_EST_CCYYMMDD = RCD_EST))
        ;
    BY SSN RCD_EST;
RUN;

PROC CONTENTS DATA=OUT.SSREARN; RUN;
TITLE 'CONTENTS OF SORTED SSREARN FILE';
RUN;

PROC PRINT DATA=OUT.SSREARN (OBS=10);
```

Appendix A.50

JCL/SAS Code: SSRERNLD

```
TITLE 'PRINT 10 RANDOM OBS FOR CHECK';  
WHERE RANUNI(1100)<0.1;  
RUN;
```

## Appendix A.51

JCL/SAS Code: JCLSSR21

```

//#6266SS2 JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(JCLSSR2) |
//*  *-----*
//*
//*  1. COMBINE AND PROCESS LONGITUDINAL FILE
//*  2. OUTPUT UPDATED STEP2 FILE
//*
//* SEGMENTS WERE PRODUCED IN STEP 1
//* THIS IS A NEW PROGRAM COMBINING JCL STATEMENTS FOR ALL SEGMENTS
//* 1974-1993 VS. 1994-DAFYR WILL HAVE DIFFERENT INCLUDE FILE
//*****
//*
//JS010 EXEC SAS9,
//  WORK='120000,60000',
//* MAKE UPDATES TO THIS SYSPARM
//* THE FORMAT IS: BEGYR, ENDYR, YOURID, NUMSEG, NODE, DAFYR
//* DO NOT UPDATE BEGYR
//* UPDATE ENDYR TO THIS DAFYR
//* UPDATE YOUR ID
//* UPDATE NUMSEG (THE NUMBER OF SEGMENTS IN SSR1 PROGRAMS)
//* DO NOT UPDATE NODE (THIS IS A NODE FOR THE INPUT FILES)
//* UPDATE THE TWO-DIGIT DAFYR (USED IN THE NODE FOR INPUT FILES)
//  PARM='SYSPARM="1994,2021,#6266,19,RLONG,21"'
//* OUTPUT FILE 1: SSR LONG SORT BY SSN AND ASCENDING RCD_EST (ALL VARS)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2P1.SA.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,,10),
//      UNIT=TSILO
//* OUTPUT FILE 2: SSR LONG SORT BY SSN AND DECENDING RCD_EST (2 VARS)
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2P1ST.SA.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,,10),
//      UNIT=TSILO
//* TEMP FILES FOR SORTING - NEED SAME NUMBER OF TEMPFILES AS NUMSEGS
//TEMP01 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,,10)
//TEMP02 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,,10)
//TEMP03 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,,10)
//TEMP04 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,,10)
//TEMP05 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,,10)
//TEMP06 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,,10)
//TEMP07 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,,10)
//TEMP08 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,,10)

```

## Appendix A.51

JCL/SAS Code: JCLSSR21

```
//TEMP09 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP10 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP11 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP12 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP13 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP14 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP15 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP16 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP17 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP18 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP19 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//* INCLUDE FILE
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(SSR2P1),
// DISP=(SHR,PASS,KEEP)
//*
//*JS020 EXEC SAS9,
//* WORK='120000,60000',
//* MAKE UPDATES TO THIS SYSPARM
//* THE FORMAT IS: BEGYR, ENDYR, YOURID, NUMSEG, NODE, DAFYR
//* DO NOT UPDATE BEGYR
//* DO NOT UPDATE ENDYR
//* UPDATE YOUR ID
//* UPDATE NUMSEG (THE NUMBER OF SEGMENTS IN SSR1 PROGRAMS)
//* DO NOT UPDATE NODE (THIS IS A NODE FOR THE INPUT FILES)
//* UPDATE THE TWO-DIGIT DAFYR (USED IN THE NODE FOR INPUT FILES)
//* PARM='SYSPARM="1974,1993,#7429,19,RLONGB,19"'
//*
//*OUT1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF19P.SSR2BP1.SA.V1,
//* DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//* UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF19.SSR.PRDLIB(SSR2P1),
// DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBCGENER
//*****
//SENDNOTE EXEC PGM=IEBCGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
```

Appendix A.51

JCL/SAS Code: JCLSSR21

FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 SSR2 IS COMPLETE  
DAF21 SSR2 HAS FINISHED RUNNING

## Appendix A.52

JCL/SAS Code: SSR2P1

```

* SSR2 - THIS IS AN INCLUDE FILE TO PROCESS 1994 - CURRENT DAF FILES;
%LET BEGYR=%SCAN(%QUOTE(&SYSPARM),1,%STR(,));
%LET ENDYR=%SCAN(%QUOTE(&SYSPARM),2,%STR(,));
%LET YOURID=%SCAN(%QUOTE(&SYSPARM),3,%STR(,));
%LET NUMSEG=%SCAN(%QUOTE(&SYSPARM),4,%STR(,));
%LET NODE=%SCAN(%QUOTE(&SYSPARM),5,%STR(,));
%LET DAFYR=%SCAN(%QUOTE(&SYSPARM),6,%STR(,));

* BEGIN PROCESSING;
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY;
OPTIONS OBS=MAX MPRINT SYMBOLGEN;
/* NOTE: CHANGE THE ENDYR AND ENDMN PARAMETER BEFORE EACH RUN */

/* ASSIGN MACROS VARIABLES TO HANDLE TIME SERIES DATA */

%MACRO PROCSSILF;
/*INITIATE MONTH COUNTER, MOCNTER, TO 1*/
%LET MOCNTER=1;
/*LOOPING THROUGH EACH YEAR SPECIFIED ABOVE*/
%DO YEAR=&BEGYR. %TO &ENDYR.;
/*SETTING YR MACRO VAR TO LAST TWO DIGITS OF EACH YEAR*/
%LET YR=%SUBSTR(&YEAR.,3,2);

/*LOOP THROUGH MONTHS 1 TO 12 AND INCREMENT MONTH COUNTER*/
%DO M=1 %TO 12;
%LET M=%SYSFUNC(PUTN(&M.,Z2.));
%LET YY&MOCNTER=%EVAL(&YR)%EVAL(&M);
%LET MOCNTER=%EVAL(&MOCNTER+1);
%END;
%END;
/*SET TOTAL MONTHS, TOTMOS, EQUAL TO MOCNTER - 1 TO GET
TOTAL MONTHS IN TIME SERIES*/
%LET TOTMOS=%EVAL(&MOCNTER-1);

/*USED TO PRING MAPPING TO LOG - CHECK DURING CODE REVIEW*/
%DO MM=1 %TO &TOTMOS;
%PUT &MM. MAPS TO YYMM, &&YY&MM.;
%END;
%PUT TOTAL MONTHS = &TOTMOS.;

/*****
WE NEED TO READ THE SSI LONG FILES TO CAPTURE THE HISTORY FIELDS
WE NEED TO SCAN ALL THE RECORDS FOR VALID VALUES INSTEAD OF JUST

```

Appendix A.52

JCL/SAS Code: SSR2P1

```

KEEPING THE LAST RECORD AS WE DID PREVIOUSLY.
THE LAST RECORD MAY HAVE HISTORY FIELDS BLANKED OUT
*****/

/*SPLIT THE VARIABLE FILE TO UPDATE RECORDS WITH NONMISSING DATA */

%DO SEG = 1 %TO &NUMSEG; /*UPDATE THE TOTAL NUMBER OF SEGMENTS */
  %LET SEG=%SYSFUNC(PUTN(&SEG,Z2.));
  LIBNAME IN&SEG.
  "OPDR.TG.PRD.ETTW.&YOURID..DAF&DAFYR.P.&NODE.&SEG..SA.V1"
  DISP=SHR;

  PROC SORT DATA=IN&SEG..LONG
    OUT=TEMP&SEG..LONG (KEEP=SSN RCD_EST);
  BY SSN DESCENDING RCD_EST;
  RUN;

%END;

DATA OUT1.SSR2P1;
  SET
  %DO SEG = 1 %TO &NUMSEG; /* UPDATE THE TOTAL NUMBER OF SEGMENTS */
    %LET SEG=%SYSFUNC(PUTN(&SEG,Z2.));
    IN&SEG..LONG
  %END;;
  BY SSN RCD_EST;
  RUN;

PROC CONTENTS DATA=OUT1.SSR2P1;
TITLE1 "CONTENTS OF SSR LONG FILE SORTED BY SSN AND ASCENDING RCD_EST";
TITLE2 "SHOULD CONTAIN ALL VARIABLES";
RUN;

DATA OUT2.SSR2P1ST;
  SET
  %DO SEG = 1 %TO &NUMSEG; /* UPDATE THE TOTAL NUMBER OF SEGMENTS */
    %LET SEG=%SYSFUNC(PUTN(&SEG,Z2.));
    TEMP&SEG..LONG
  %END;;
  BY SSN DESCENDING RCD_EST;
  IF FIRST.RCD_EST;
  RUN;

PROC CONTENTS DATA=OUT2.SSR2P1ST;
TITLE1 "CONTENTS OF SSR LONG FILE SORTED BY SSN AND DESCENDING RCD_EST";

```

Appendix A.52

JCL/SAS Code: SSR2P1

```
TITLE2 "SHOULD ONLY HAVE 2 VARIABLES: SSN AND RCD_EST";  
RUN;
```

```
%MEND PROCSSILF;  
%PROCSSILF;
```



## Appendix A.53

JCL/SAS Code: SSRERNCR

```
//#6266SRE JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT: DAF19 CONSTRUCTION
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(SSRERNCR)
//*
//* PURPOSE: GENERATE EARNYMM VARIABLES FROM SSITXT
//*
//* PROGRAMMER: SERGE LUKASHANETS
//*      SLUKASHANETS@MATHEMATICA-MPR.COM
//*
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000'      * SPACE IN BLKS *
//*
//SSRERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSREARN1.V1,DISP=SHR
//SSILF DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2P1ST.SA.V1,DISP=SHR
//*
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP5 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP6 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP7 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP8 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP9 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP10 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//*
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSREARN.SA.V1,
//  DISP=(MOD,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//*
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MACROGEN OBS=MAX;

%LET ENDYR=2021; /*CHANGE AS NEEDED*/

%MACRO DOIT;
```

## Appendix A.53

JCL/SAS Code: SSRERNCR

```
/*WE SUBSET THE SSREARN FILE USING IESTART AND IESTOP.
WE DO THIS IN A SEPARATE DATA SET SINCE WE NEED TO TRANSFORM
 IESTART AND IESTOP INTO NUMERIC DATE VARIABLES FIRST*/
DATA TEMP1.SSREARN;
  SET SSRERN.SSREARN;
  BY SSN;

  *TRANSFORM IESTART AND IESTOP INTO NUMERIC VARIABLES;
  IESTART_DATE = MDY(SUBSTR(IESTART,5,2), 1, SUBSTR(IESTART,1,4));

  IF IESTOP = '000000' THEN IESTOP_DATE = MDY(12,1,&ENDYR.);
  ELSE IESTOP_DATE = MDY(SUBSTR(IESTOP,5,2), 1, SUBSTR(IESTOP,1,4));

  *DROP IESTOPS BEFORE JAN 1994;
  IF NOT MISSING(IESTOP_DATE) AND IESTOP_DATE GE '01JAN1994'D;

  *DROP IESTARTS AFTER DEC OF LAST DAF YEAR;
  IF IESTART_DATE LE "31DEC&ENDYR."D;

  *RECODE START AND END DATES;
  IF IESTART_DATE < '01JAN1994'D THEN IESTART_DATE='01JAN1994'D;
  IF IESTOP_DATE > "01DEC&ENDYR."D THEN IESTOP_DATE="01DEC&ENDYR."D;

RUN;
  /*
PROC FREQ DATA=TEMP1.SSREARN;
TITLE 'QA IE VARIABLES BEFORE RESHAPING';
TABLES IESTART * IESTART_DATE / LIST MISSING;
TABLES IESTOP * IESTOP_DATE / LIST MISSING;
TABLES IETYP / LIST MISSING;
RUN;
  */

DATA TEMP2.SSREARN;
  SET TEMP1.SSREARN;
  BY SSN RCD_EST;

  * SET UP OUTPUT ARRAYS;
  ARRAY T16EXLAMT (1994:&ENDYR.,1:12)
    %DO YEAR=1994 %TO &ENDYR;
    %LET YR=%SUBSTR(&YEAR.,3,2);
    T16EXLAMT&YR.01-T16EXLAMT&YR.12
    %END;;
  ARRAY T16EXPAMT (1994:&ENDYR,1:12)
    %DO YEAR=1994 %TO &ENDYR;
```

## Appendix A.53

JCL/SAS Code: SSRERNCR

```
%LET YR=%SUBSTR(&YEAR.,3,2);
T16EXPAMT&YR.01-T16EXPAMT&YR.12
%END;;
ARRAY T16BEXPAMT (1994:&ENDYR,1:12)
%DO YEAR=1994 %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
T16BEXPAMT&YR.01-T16BEXPAMT&YR.12
%END;;
ARRAY T16GRSAMT (1994:&ENDYR,1:12)
%DO YEAR=1994 %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
T16GRSAMT&YR.01-T16GRSAMT&YR.12
%END;;
ARRAY T16SEAMT (1994:&ENDYR,1:12)
%DO YEAR=1994 %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
T16SEAMT&YR.01-T16SEAMT&YR.12
%END;;
ARRAY T16NETAMT (1994:&ENDYR,1:12)
%DO YEAR=1994 %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
T16NETAMT&YR.01-T16NETAMT&YR.12
%END;;
ARRAY T16PASAMT (1994:&ENDYR,1:12)
%DO YEAR=1994 %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
T16PASAMT&YR.01-T16PASAMT&YR.12
%END;;

ARRAY T16VERCD (1994:&ENDYR.,1:12)
%DO YEAR=1994 %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
T16VERCD&YR.01-T16VERCD&YR.12
%END;;

* KEEP YMM VARIABLES ACROSS OBS WITH THE SAME SSN;
RETAIN T16EXLAMT: T16EXPAMT: T16BEXPAMT: T16GRSAMT:
T16SEAMT: T16NETAMT: T16PASAMT: T16VERCD;;

* INITIALIZE RETAINED VARIABLES WHEN AT THE FIRST OB WITH THIS SSN;
IF FIRST.SSN THEN DO;
DO YR=1994 TO &ENDYR.;
DO MO = 1 TO 12;
T16EXLAMT(YR,MO)=.;
T16EXPAMT(YR,MO)=.;
```

## Appendix A.53

JCL/SAS Code: SSRERNCR

```
T16BEXPAMT(YR,MO)=.;
T16GRSAMT(YR,MO)=.;
T16SEAMT(YR,MO)=.;
T16NETAMT(YR,MO)=.;
T16PASAMT(YR,MO)=.;
T16VERCD(YR,MO)=";
END;
END;
END;

* SET YMMM VARIABLES;
* EACH OB REPRESENTS AN SSN, RECORD ESTABLISHMENT DATE, AND
COMPUTATION MONTH COMBINATION UNIQUELY. SO WE CAN USE THE VALUE
OF COMPUTATION MONTH TO ASSIGN THE YMMM VALUES. WHEN TWO
COMPUTATION MONTHS OVERLAP FROM ONE SSN-RECORD ESTABLISHMENT DATE
COMBINATION THE DATA FROM THE LATER ESTABLISHED RECORDS OVERLAYS
THAT FROM THE EARLIER ESTABLISHED RECORDS BECAUSE OF THE RETAIN
STATEMENT USED ABOVE;

*SET START AND END POINTS;
START_YR = YEAR(IESTART_DATE); START_MO = MONTH(IESTART_DATE);
END_YR = YEAR(IESTOP_DATE); END_MO = MONTH(IESTOP_DATE);

DO YR=START_YR TO END_YR;

IF YR=START_YR THEN FIRSTMO = START_MO;
ELSE FIRSTMO = 1;
IF YR=END_YR THEN LASTMO = END_MO;
ELSE LASTMO = 12;

DO MO = FIRSTMO TO LASTMO;
SELECT(IETYP);
WHEN('W') DO;
IF IEFRQ NE 'C' THEN DO CYR=YR TO &ENDYR.;
IF CYR=YR THEN CFIRSTMO=MO;
ELSE CFIRSTMO=1;
DO CMO=CFIRSTMO TO 12;
T16GRSAMT(CYR,CMO)=.;
T16VERCD(CYR,CMO)=";
END;
END;
T16GRSAMT(YR, MO) = IEAMT;
T16VERCD(YR,MO) = IEVAR;
END;
WHEN('B') DO;
```

## Appendix A.53

JCL/SAS Code: SSRERNCR

```
IF IEFQR NE 'C' THEN DO CYR=YR TO &ENDYR.;
  IF CYR=YR THEN CFIRSTMO=MO;
  ELSE CFIRSTMO=1;
  DO CMO=CFIRSTMO TO 12;
    T16EXLAMT(CYR,CMO)=.;
  END;
END;
T16EXLAMT(YR, MO) = IEAMT;
END;
WHEN('C') DO;
  IF IEFQR NE 'C' THEN DO CYR=YR TO &ENDYR.;
  IF CYR=YR THEN CFIRSTMO=MO;
  ELSE CFIRSTMO=1;
  DO CMO=CFIRSTMO TO 12;
    T16BEXPAMT(CYR,CMO)=.;
  END;
END;
T16BEXPAMT(YR, MO) = IEAMT;
END;
WHEN('T') DO;
  IF IEFQR NE 'C' THEN DO CYR=YR TO &ENDYR.;
  IF CYR=YR THEN CFIRSTMO=MO;
  ELSE CFIRSTMO=1;
  DO CMO=CFIRSTMO TO 12;
    T16EXPAMT(CYR,CMO)=.;
  END;
END;
T16EXPAMT(YR, MO) = IEAMT;
END;
WHEN('S') DO;
  IF IEFQR NE 'C' THEN DO CYR=YR TO &ENDYR.;
  IF CYR=YR THEN CFIRSTMO=MO;
  ELSE CFIRSTMO=1;
  DO CMO=CFIRSTMO TO 12;
    T16SEAMT(CYR,CMO)=.;
  END;
END;
T16SEAMT(YR, MO) = IEAMT;
END;
WHEN('N') DO;
  IF IEFQR NE 'C' THEN DO CYR=YR TO &ENDYR.;
  IF CYR=YR THEN CFIRSTMO=MO;
  ELSE CFIRSTMO=1;
  DO CMO=CFIRSTMO TO 12;
    T16NETAMT(CYR,CMO)=.;
```

## Appendix A.53

JCL/SAS Code: SSRERNCR

```
        END;
    END;
    T16NETAMT(YR, MO) = IEAMT;
END;
WHEN('D') DO;
    IF IEFREQ NE 'C' THEN DO CYR=YR TO &ENDYR.;
        IF CYR=YR THEN CFIRSTMO=MO;
        ELSE CFIRSTMO=1;
        DO CMO=CFIRSTMO TO 12;
            T16PASAMT(CYR,CMO)=.;
        END;
    END;
    T16PASAMT(YR, MO) = IEAMT;
END;
OTHERWISE;
END;
END;
END;

* LAST OBS FOR THIS SSN THEN OUTPUT;
IF LAST.RCD_EST THEN OUTPUT;

RUN;

%MACRO SORTDAT(BEGOBS= ,ENDOBS= ,RUN= );
PROC SORT DATA=TEMP2.SSREARN(FIRSTOBS=&BEGOBS. OBS=&ENDOBS.)
    OUT = TEMP&RUN..SSREARN;
    BY SSN DESCENDING RCD_EST;
RUN;
%MEND SORTDAT;
%SORTDAT(BEGOBS = 1, ENDOBS = 5000000, RUN = 3);
%SORTDAT(BEGOBS = 5000001, ENDOBS = 10000000, RUN = 4);
%SORTDAT(BEGOBS = 10000000, ENDOBS = MAX, RUN = 5);

DATA TEMP6.SSREARN;
    SET TEMP3.SSREARN
        TEMP4.SSREARN
        TEMP5.SSREARN
    ;
    BY SSN DESCENDING RCD_EST;
RUN;

DATA TEMP7.SSREARN;
    MERGE TEMP6.SSREARN(IN=A WHERE=(RCD_EST<="31DEC&ENDYR."D))
```

## Appendix A.53

JCL/SAS Code: SSRERNCR

```
        SSILF.SSR2P1ST(IN=B KEEP=SSN RCD_EST)
        ;
        BY SSN DESCENDING RCD_EST;

        RETAIN PREV_RCD_EST;

        IF FIRST.SSN THEN PREV_RCD_EST=.;

* SET UP OUTPUT ARRAYS;
ARRAY T16EXLAMT (1994:&ENDYR.,1:12)
        %DO YEAR=1994 %TO &ENDYR;
        %LET YR=%SUBSTR(&YEAR.,3,2);
        T16EXLAMT&YR.01-T16EXLAMT&YR.12
        %END;;
ARRAY T16EXPAMT (1994:&ENDYR,1:12)
        %DO YEAR=1994 %TO &ENDYR;
        %LET YR=%SUBSTR(&YEAR.,3,2);
        T16EXPAMT&YR.01-T16EXPAMT&YR.12
        %END;;
ARRAY T16BEXPAMT (1994:&ENDYR,1:12)
        %DO YEAR=1994 %TO &ENDYR;
        %LET YR=%SUBSTR(&YEAR.,3,2);
        T16BEXPAMT&YR.01-T16BEXPAMT&YR.12
        %END;;
ARRAY T16GRSAMT (1994:&ENDYR,1:12)
        %DO YEAR=1994 %TO &ENDYR;
        %LET YR=%SUBSTR(&YEAR.,3,2);
        T16GRSAMT&YR.01-T16GRSAMT&YR.12
        %END;;
ARRAY T16SEAMT (1994:&ENDYR,1:12)
        %DO YEAR=1994 %TO &ENDYR;
        %LET YR=%SUBSTR(&YEAR.,3,2);
        T16SEAMT&YR.01-T16SEAMT&YR.12
        %END;;
ARRAY T16NETAMT (1994:&ENDYR,1:12)
        %DO YEAR=1994 %TO &ENDYR;
        %LET YR=%SUBSTR(&YEAR.,3,2);
        T16NETAMT&YR.01-T16NETAMT&YR.12
        %END;;
ARRAY T16PASAMT (1994:&ENDYR,1:12)
        %DO YEAR=1994 %TO &ENDYR;
        %LET YR=%SUBSTR(&YEAR.,3,2);
        T16PASAMT&YR.01-T16PASAMT&YR.12
        %END;;
ARRAY T16VERCD (1994:&ENDYR.,1:12)
```

## Appendix A.53

JCL/SAS Code: SSRERNCR

```
%DO YEAR=1994 %TO &ENDYR;  
%LET YR=%SUBSTR(&YEAR.,3,2);  
T16VERCD&YR.01-T16VERCD&YR.12  
%END;;
```

```
* SET DATE OF SUBSEQUENT RECORD ESTABLISHMENT FOR USE AS START DATE  
FOR BACKING OUT CARRIED FORWARD EARNINGS;  
STARTYR=MAX('01JAN1994'D,MIN(PREV_RCD_EST,"31DEC&ENDYR."D));
```

```
IF NOT FIRST.SSN THEN  
DO YR=YEAR(STARTYR) TO &ENDYR.;  
IF YR=YEAR(STARTYR) THEN STARTMO=MONTH(STARTYR);  
ELSE STARTMO=1;  
DO MO=STARTMO TO 12;  
T16GRSAMT(YR,MO) = .;  
T16EXLAMT(YR,MO) = .;  
T16BEXPAMT(YR,MO) = .;  
T16EXPAMT(YR,MO) = .;  
T16SEAMT(YR,MO) = .;  
T16NETAMT(YR,MO) = .;  
T16PASAMT(YR,MO) = .;  
T16VERCD(YR,MO) = "  
END;  
END;
```

```
PREV_RCD_EST=RCD_EST;
```

```
IF A THEN OUTPUT;  
RUN;
```

```
%MACRO SORTDAT(BEGOBS= ,ENDOBS= ,RUN= );  
PROC SORT DATA=TEMP7.SSREARN(FIRSTOBS=&BEGOBS. OBS=&ENDOBS.)  
OUT = TEMP&RUN..SSREARN;  
BY SSN RCD_EST;  
RUN;  
%MEND SORTDAT;  
%SORTDAT(BEGOBS = 1, ENDOBS = 5000000, RUN = 8);  
%SORTDAT(BEGOBS = 5000001, ENDOBS = 10000000, RUN = 9);  
%SORTDAT(BEGOBS = 10000000, ENDOBS = MAX, RUN = 10);
```

```
DATA OUT.SSREARN;  
SET TEMP8.SSREARN  
TEMP9.SSREARN  
TEMP10.SSREARN  
;
```



Appendix A.53

JCL/SAS Code: SSRERNCR

```

    BY SSN RCD_EST;
    RUN;

    PROC CONTENTS DATA = OUT.SSREARN;
    TITLE 'CONTENTS OF FINAL SSIEXT FILE';
    RUN;

    PROC PRINT DATA=TEMP6.SSREARN (OBS=20);
    RUN;

    PROC PRINT DATA=OUT.SSREARN (OBS=20);
    RUN;
    /*
    PROC PRINT DATA = OUT.SSREARN (OBS=10);
    TITLE 'PRINT 10 RANDOM OBS FOR CHECK';
    WHERE RANUNI(1100)<0.1;
    RUN;
    */
    %MEND;
    %DOIT;

    //*****
    /* SENDING EMAIL FROM HOST USING IEBGENER
    //*****
    //SENDNOTE EXEC PGM=IEBGENER
    //SYSIN DD DUMMY
    //SYSPRINT DD SYSOUT=*
    //SYSUT2 DD SYSOUT=(B,SMTP)
    //SYSUT1 DD *
    HELO MVSHOST
    MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
    RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
    DATA
    FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
    TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
    SUBJECT: DAF21 SSRERNCR COMPLETED
    PROGRAM SSRERNCR IS COMPLETE.

```

## Appendix A.54

JCL/SAS Code: JCLSSR22

```

//#6266SS2 JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.$4671.DAF18.TEARN.PRDLIB(JCLSSR22)|
//*  *-----*
//*
//*  1. COMBINE AND PROCESS LONGITUDINAL FILE
//*  2. OUTPUT UPDATED STEP2 FILE
//*
//* SEGMENTS WERE PRODUCED IN STEP 1
//* THIS IS A NEW PROGRAM COMBINING JCL STATEMENTS FOR ALL SEGMENTS
//* 1974-1993 VS. 1994-DAFYR WILL HAVE DIFFERENT INCLUDE FILE
//*****
//*
//JS010 EXEC SAS9,
//  WORK='120000,60000',
//  PARM='SYSPARM="1994,2021,#6266,19,RLONG,21"'
//* MAKE UPDATES TO THIS SYSPARM
//* THE FORMAT IS: BEGYR, ENDYR, YOURID, NUMSEG, NODE, DAFYR
//* DO NOT UPDATE BEGYR
//* UPDATE ENDYR TO THIS DAFYR
//* UPDATE YOUR ID
//* UPDATE NUMSEG (THE NUMBER OF SEGMENTS IN SSR1 PROGRAMS)
//* DO NOT UPDATE NODE (THIS IS A NODE FOR THE INPUT FILES)
//* UPDATE THE TWO-DIGIT DAFYR (USED IN THE NODE FOR INPUT FILES)
//*
//SSILF DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2P1.SA.V1,DISP=SHR
//EARN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSREARN.SA.V1,DISP=SHR
//LONGF DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//LONGV DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//LF1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//LF2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//LV1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//LV2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//LV3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO

```

## Appendix A.54

JCL/SAS Code: JCLSSR22

```

//LFUPD DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//      UNIT=TSILO
//LVUPD DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//      UNIT=TSILO
//OUT1  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2P2.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.SSR.PRDLIB(SSR2P2),
//      DISP=(SHR,PASS,KEEP)
//*
//*JS020 EXEC SAS9,
//*  WORK='120000,60000',
//* MAKE UPDATES TO THIS SYSPARM
//* THE FORMAT IS: BEGYR, ENDYR, YOURID, NUMSEG, NODE, DAFYR
//* DO NOT UPDATE BEGYR
//* DO NOT UPDATE ENDYR
//* UPDATE YOUR ID
//* UPDATE NUMSEG (THE NUMBER OF SEGMENTS IN SSR1 PROGRAMS)
//* DO NOT UPDATE NODE (THIS IS A NODE FOR THE INPUT FILES)
//* UPDATE THE TWO-DIGIT DAFYR (USED IN THE NODE FOR INPUT FILES)
//*  PARM='SYSPARM="1974,1993,#7429,18,RLONGB,18"'
//*
//*SSILF DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF18A.SSR2BP1.SA.V1,DISP=SHR
//*EARN  DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF18A.SSREARN.SA.V1,DISP=SHR
//*OUT   DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF18A.SSR2BP2.SA.V1,
//*     DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*     UNIT=TSILO
//*SYSIN DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF18.ALTSSR.PRDLIB(SSR2P2),
//*     DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 SSR2 IS COMPLETE

```

Appendix A.54

JCL/SAS Code: JCLSSR22

DAF21 SSR2 HAS FINISHED RUNNING

## Appendix A.55

JCL/SAS Code: SSR2P2

```
* SSR2 - THIS IS AN INCLUDE FILE TO PROCESS 1994 - CURRENT DAF FILES;
%LET BEGYR=%SCAN(%QUOTE(&SYSPARM),1,%STR(,));
%LET ENDYR=%SCAN(%QUOTE(&SYSPARM),2,%STR(,));
%LET YOURID=%SCAN(%QUOTE(&SYSPARM),3,%STR(,));
%LET NUMSEG=%SCAN(%QUOTE(&SYSPARM),4,%STR(,));
%LET NODE=%SCAN(%QUOTE(&SYSPARM),5,%STR(,));
%LET DAFYR=%SCAN(%QUOTE(&SYSPARM),6,%STR(,));

* BEGIN PROCESSING;
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY;
OPTIONS OBS=MAX MPRINT;
/* NOTE: CHANGE THE ENDYR AND ENDMN PARAMETER BEFORE EACH RUN */

/* ASSIGN MACROS VARIABLES TO HANDLE TIME SERIES DATA */

%MACRO PROCSSILF;
/*INITIATE MONTH COUNTER, MOCNTER, TO 1*/
%LET MOCNTER=1;
/*LOOPING THROUGH EACH YEAR SPECIFIED ABOVE*/
%DO YEAR=&BEGYR. %TO &ENDYR.;
/*SETTING YR MACRO VAR TO LAST TWO DIGITS OF EACH YEAR*/
%LET YR=%SUBSTR(&YEAR.,3,2);

/*LOOP THROUGH MONTHS 1 TO 12 AND INCREMENT MONTH COUNTER*/
%DO M=1 %TO 12;
%LET M=%SYSFUNC(PUTN(&M.,Z2.));
%LET YY&MOCNTER=%EVAL(&YR)%EVAL(&M);
%LET MOCNTER=%EVAL(&MOCNTER+1);
%END;
%END;
/*SET TOTAL MONTHS, TOTMOS, EQUAL TO MOCNTER - 1 TO GET
TOTAL MONTHS IN TIME SERIES*/
%LET TOTMOS=%EVAL(&MOCNTER-1);

/*USED TO PRING MAPPING TO LOG - CHECK DURING CODE REVIEW*/
%DO MM=1 %TO &TOTMOS;
%PUT &MM. MAPS TO YYMM, &&YY&MM.;
%END;
%PUT TOTAL MONTHS = &TOTMOS.;

/*****
WE NEED TO READ THE SSI LONG FILES TO CAPTURE THE HISTORY FIELDS
WE NEED TO SCAN ALL THE RECORDS FOR VALID VALUES INSTEAD OF JUST
```

## Appendix A.55

JCL/SAS Code: SSR2P2

```

KEEPING THE LAST RECORD AS WE DID PREVIOUSLY.
THE LAST RECORD MAY HAVE HISTORY FIELDS BLANKED OUT
*****/

/*SPLIT THE VARIABLE FILE TO UPDATE RECORDS WITH NONMISSING DATA */

DATA LONGF.ALLF(DROP=
  %DO MM=1 %TO &TOTMOS;
    PSTA&&YY&MM.  LIVF&&YY&MM.  SCON&&YY&MM.  EICM&&YY&MM.
    UINC&&YY&MM.  FAMT&&YY&MM.  SAMT&&YY&MM.  MTST&&YY&MM.
    FPMT&&YY&MM.  SPMT&&YY&MM.
    CMTHRD&&YY&MM.  CURCOMP&&YY&MM.
    T16GRSAMT&&YY&MM.
    T16EXLAMT&&YY&MM.
    T16EXPAMT&&YY&MM.
    T16BEXPAMT&&YY&MM.
    T16SEAMT&&YY&MM.
    T16NETAMT&&YY&MM.
    T16PASAMT&&YY&MM.
    T16VERCD&&YY&MM.
  %END;);
LONGV.ALLV(KEEP=SSN RCD_EST /* VARIABLE PART */
%DO MM=1 %TO &TOTMOS;
  PSTA&&YY&MM.  LIVF&&YY&MM.  SCON&&YY&MM.  EICM&&YY&MM.
  UINC&&YY&MM.  FAMT&&YY&MM.  SAMT&&YY&MM.  MTST&&YY&MM.
  FPMT&&YY&MM.  SPMT&&YY&MM.
  CMTHRD&&YY&MM.  CURCOMP&&YY&MM.
  T16GRSAMT&&YY&MM.
  T16EXLAMT&&YY&MM.
  T16EXPAMT&&YY&MM.
  T16BEXPAMT&&YY&MM.
  T16SEAMT&&YY&MM.
  T16NETAMT&&YY&MM.
  T16PASAMT&&YY&MM.
  T16VERCD&&YY&MM.
%END;);
MERGE SSILF.SSR2P1(IN=SSILF)
      EARN.SSREARN(IN=SSREARN)
      ;
BY SSN RCD_EST;
IF SSREARN;
RUN;

PROC PRINT DATA=LONGF.ALLF (OBS=20); TITLE 'BEFORE UPDATE'; RUN;
PROC PRINT DATA=LONGV.ALLV (OBS=50); RUN;

```

## Appendix A.55

JCL/SAS Code: SSR2P2

```
/*CREATE MIN AND MAX VALUES FROM SOME OF THE FIXED PART VARIABLES */

PROC MEANS NOPRINT DATA=LONGF.ALLF;
VAR START_RD RCD_EST ELG_RD APP_RD _8080_JD CLM_FIL FIRST_PA STOP_RD;
BY SSN;
OUTPUT OUT=LONGF.MAXMIN(DROP=_:);
    MIN=MINSTRRD MINRCDST SSI_MIN_ELG MINAPPRD MIN_80JD MINCLMFL
    MINFRSTP MINSTPRD
    MAX=MAXSTRRD MAXRCDST MAXELGRD MAXAPPRD MAX_80JD MAXCLMFL
    MAXFRSTP MAXSTPRD
;
RUN;

/*SPLIT THE FIXED PART FILE TO UPDATE RECS WITH NONMISSING DATA */
/*FOR FIXED DATA WE DO NOT HAVE TO EXAMINE INTERMEDIATE RECORDS */
/*WE WANT THE CURRENT VALUE IN THE FIELDS */

DATA LF1.FIXEDA LF2.FIXEDB;
SET LONGF.ALLF;
BY SSN RCD_EST;
IF FIRST.SSN THEN OUTPUT LF1.FIXEDA;
ELSE OUTPUT LF2.FIXEDB;
RUN;

/* RUN THE UPDATE PROCEDURE */

DATA LFUPD.LONGF;
UPDATE LF1.FIXEDA LF2.FIXEDB;
BY SSN;
RUN;

/* CONVERSATIONS WITH MARY BARBOUR HAVE REVEALED THAT SSA WILL
TRY TO SAVE SPACE BY BLANKING AND ZEROING OUT HISTORY FIELDS
WHEN A NEW PERIOD OF ELIGIBILITY IS ESTABLISHED FOLLOWING THE
END OF AN OLD PERIOD OF ELIGIBILITY (CREATING A NEW RECORD).
THEREFORE WE CAN NOT TAKE THE HISTORY VALUES FROM THE LAST RECORD
AS PREVIOUSLY BUT MUST SCAN ALL THE RECORDS USING AN UPDATE
PROCEDURE. FOR CHARACTER VALUES, A BLANK ON A SUBSEQUENT RECORD
(WHERE THERE HAD BEEN A VALUE ON A PREVIOUS RECORD) WILL NOT
OVERWRITE THE VALUE USING THE UPDATE PROCEDURE. HOWEVER, FOR
NUMERIC VALUES, SSA WILL ZERO OUT THE FIELD AND UPDATE WILL
OVERWRITE A VALUE WITH THE ZERO. AS A RESULT WE WILL BLANK OUT
THESE ZEROS SO AS NOT TO OVERWRITE THE FIELD.
WITH REFERENCE TO THE ACTUAL STATE AND FEDERAL PAYMENT FIELDS
```

## Appendix A.55

JCL/SAS Code: SSR2P2

MARY HAS ALSO INDICATED THAT THERE MAY BE DOLLAR ADJUSTMENTS TO A GIVEN MONTHLY PAYMENT ON A SUBSEQUENT RECORD. FOR EXAMPLE, FDP0309 (FED PAYMENT IN SEPT 2003) MAY SHOW PAYMENTS ON 2 RECORDS THE TOTAL ACTUAL PAYMENT BEING THE SUM OF THESE. THIS IS HANDLED IN THE FOLLOWING PROCEDURE \*/

```

DATA LV1.VARA LV2.VARB;
SET LONGV.ALLV; BY SSN RCD_EST;

/* ADD UP THE PAYMENT AMOUNTS ACROSS RECORDS */
/* IN THE CASE WHERE A FIELD IS ADJUSTED, THE INCREMENT
WILL APPEAR ACROSS RECORDS AND THE UPDATE PROCEDURE
WILL TAKE THE SUM INSTEAD OF OVERWRITING ONE AMOUNT
WITH A SUBSEQUENT AMOUNT */

ARRAY FPMT (*) %DO MM=1 %TO &TOTMOS; FPMT&&YY&MM. %END;;
ARRAY SPMT (*) %DO MM=1 %TO &TOTMOS; SPMT&&YY&MM. %END;;
/*CREATING TEMPORARY VARIABLES, IMPLIED BY 'T'*/
ARRAY TFPMT (*) %DO MM=1 %TO &TOTMOS; TFPMT&&YY&MM. %END;;
ARRAY TSPMT (*) %DO MM=1 %TO &TOTMOS; TSPMT&&YY&MM. %END;;

RETAIN %DO MM=1 %TO &TOTMOS; TFPMT&&YY&MM. %END;;
RETAIN %DO MM=1 %TO &TOTMOS; TSPMT&&YY&MM. %END;;

IF FIRST.SSN THEN DO I=1 TO DIM(FPMT);
  TFPMT(I) = FPMT(I);
  TSPMT(I) = SPMT(I);
END;
ELSE DO I=1 TO DIM(FPMT);
  IF FPMT(I) >= 0 THEN TFPMT(I) + FPMT(I);
  IF SPMT(I) >= 0 THEN TSPMT(I) + SPMT(I);
END;

* SET EARNINGS VALUES TO MISSING WHEN THE STATUS CODE IS BLANK. THIS
WILL CURTAIL ANY CONTINUIOUS EARNINGS VALUES SO THEY WILL NOT BE
REPORTED ON LATER RECORDS;
ARRAY PSTA (*) %DO MM=1 %TO &TOTMOS;
  PSTA&&YY&MM.
  %END;;
* IN THIS ARRAY THE FIRST DIMENSION IS THE MONTH AND THE SECOND
IS THE NUMBER OF EARNINGS VARIABLES;
ARRAY ERNVARS (&TOTMOS.,7) %DO MM=1 %TO &TOTMOS.;
  T16GRSAMT&&YY&MM.
  T16EXLAMT&&YY&MM.

```



## Appendix A.55

JCL/SAS Code: SSR2P2

```
T16EXPAMT&&YY&MM.
T16BEXPAMT&&YY&MM.
T16SEAMT&&YY&MM.
T16NETAMT&&YY&MM.
T16PASAMT&&YY&MM.
%END;;
ARRAY ERNVER (&TOTMOS.) %DO MM=1 %TO &TOTMOS.;
T16VERCD&&YY&MM.
%END;;

* WHEN CURRENT MONTH STATUS CODE IS MISSING SET ALL EARNINGS VARS TO
MISSING AS ANY CONTINUOUS VALUE SHOULD STOP AT THIS POINT. IF THE
PRIOR MONTH WAS TERMINATED AND THE CURRENT MONTH IS NOT TERMINATED
SET ALL EARNINGS VARS TO MISSING AS ANY CONTINUOUS VALUE SHOULD
STOP AT TERMINATION.;
DO I=1 TO DIM(PSTA);
  IF PSTA(I)=" " THEN DO;
    ERNVER(I)=" ";
    DO J=1 TO 7;
      ERNVAR(I,J)=.;
    END;
  END;
  IF I>1 AND SUBSTR(PSTA(I-1),1,1)='T'
  AND SUBSTR(PSTA(I),1,1) NE 'T' THEN DO;
    ERNVER(I)=" ";
    DO J=1 TO 7;
      ERNVAR(I,J)=.;
    END;
  END;
END;

DROP FPMT: SPMT: I J;
/* RENAME PAYMENT VARIABLES BACK TO ORIGINAL NAME */

%DO MM = 1 %TO &TOTMOS;
  RENAME TFPMT&&YY&MM. = FPMT&&YY&MM.;
  RENAME TSPMT&&YY&MM. = SPMT&&YY&MM.;
%END;

IF FIRST.SSN THEN OUTPUT LV1.VARA;
OUTPUT LV2.VARB;
RUN;

DATA LV3.VARB2;
SET LV2.VARB; BY SSN RCD_EST;
```

## Appendix A.55

JCL/SAS Code: SSR2P2

```
/* ON THE LAST RECORD IF IT IS A TERMINATION RECORD THE WHOLE  
HISTORY MAY BE BLANKED AND ZEROED */
```

```
ARRAY FAMT (*) %DO MM=1 %TO &TOTMOS; FAMT&&YY&MM. %END;;  
ARRAY SAMT (*) %DO MM=1 %TO &TOTMOS; SAMT&&YY&MM. %END;;  
ARRAY EICM (*) %DO MM=1 %TO &TOTMOS; EICM&&YY&MM. %END;;  
ARRAY UINC (*) %DO MM=1 %TO &TOTMOS; UINC&&YY&MM. %END;;  
ARRAY FPMT (*) %DO MM=1 %TO &TOTMOS; FPMT&&YY&MM. %END;;  
ARRAY SPMT (*) %DO MM=1 %TO &TOTMOS; SPMT&&YY&MM. %END;;  
ARRAY CURCOMP (*) %DO MM=1 %TO &TOTMOS; CURCOMP&&YY&MM. %END;;
```

```
/*NEW DAF13-CURCOMPYYMM RANGES FROM 0 TO 8. 0 IS INVALID SO SET TO  
MISSING. WITH 0'S RECODING TO MISSING, DO NOT HAVE TO WORRY ABOUT  
0'S OVERWRITING VALID 1-8 VALUES DURING THE UPDATE PROCEDURE BELOW*/  
DO I=1 TO DIM(CURCOMP);  
IF CURCOMP(I)="0" THEN CURCOMP(I)=" ";  
END; DROP I;
```

```
/* FLAG THE NUMERIC FIELDS IF GT 0 ON ANY RECORD -  
NEVER SET THEM TO ZERO IF THEY ARE ZEROED OUT SUBSEQUENTLY */
```

```
/*DELETED '3' WHICH SET NUMERIC LENGTH TO 3*/  
ARRAY FAF (&TOTMOS) _TEMPORARY_;  
ARRAY SAF (&TOTMOS) _TEMPORARY_;  
ARRAY EIF (&TOTMOS) _TEMPORARY_;  
ARRAY UIF (&TOTMOS) _TEMPORARY_;  
ARRAY FPF (&TOTMOS) _TEMPORARY_;  
ARRAY SPF (&TOTMOS) _TEMPORARY_;
```

```
/* INITIALIZE FLAGS TO ZERO */  
IF FIRST.SSN THEN DO I = 1 TO DIM(FAF);
```

```
FAF(I) = 0;  
SAF(I) = 0;  
EIF(I) = 0;  
UIF(I) = 0;  
FPF(I) = 0;  
SPF(I) = 0;  
END;
```

```
DO I = 1 TO DIM(FAMT);  
IF FAMT(I) > 0 THEN FAF(I) = 1;  
IF SAMT(I) > 0 THEN SAF(I) = 1;
```

## Appendix A.55

JCL/SAS Code: SSR2P2

```
IF EICM(I) > 0 THEN EIF(I) = 1;
IF UINC(I) > 0 THEN UIF(I) = 1;
IF FPMT(I) > 0 THEN FPF(I) = 1;
IF SPMT(I) > 0 THEN SPF(I) = 1;
END;

/* IF THE AMOUNTS HAVE BEEN ZEROED OUT ON SUBSEQUENT RECORDS
WITH A NEW START DATE -
SET THE ZERO TO MISSING IF THE MONTHLY AMOUNT HAS BEEN
FLAGGED AS EVER BEING > 0. THIS PREVENTS THE UPDATE
PROCEDURE FROM OVERWRITING A RECORD WITH ZERO.
FOR THE ALPHANUMERIC FIELDS A BLANK WILL NOT OVERWRITE
A PREVIOUS VALID VALUE */

DO I = 1 TO DIM(FAMT);
IF FAMT(I) = 0 AND FAF(I) = 1 THEN FAMT(I) = .;
IF SAMT(I) = 0 AND SAF(I) = 1 THEN SAMT(I) = .;
IF EICM(I) = 0 AND EIF(I) = 1 THEN EICM(I) = .;
IF UINC(I) = 0 AND UIF(I) = 1 THEN UINC(I) = .;
IF FPMT(I) = 0 AND FPF(I) = 1 THEN FPMT(I) = .;
IF SPMT(I) = 0 AND SPF(I) = 1 THEN SPMT(I) = .;
END;
DROP I;
RUN;

/* RUN THE UPDATE PROCEDURE */

DATA LVUPD.LONGV;
UPDATE LV1.VARA LV3.VARB2;
BY SSN;
/*CAN WE MOVE THIS DROP STATEMENT TO THE DATASTEP ABOVE?*/

ARRAY CURCOMP (*) %DO MM=1 %TO &TOTMOS; CURCOMP&&YY&MM. %END;;

/*NEW DAF13-REVERSING RECODING PERFORMED ABOVE*/
DO I=1 TO DIM(CURCOMP);
IF CURCOMP(I)=" " THEN CURCOMP(I)="0";
END;
DROP I;
RUN;

DATA OUT1.LONG (KEEP = SSN T16GRSAMT:T16EXLAMT:T16EXPAMT:T16BEXPAMT:
T16SEAMT: T16NETAMT: T16PASAMT: T16VERCD: );
MERGE LFUPD.LONGF(IN=A)
LONGF.MAXMIN
```

## Appendix A.55

JCL/SAS Code: SSR2P2

```
LVUPD.LONGV
;
BY SSN;
IF A;
/*ADDING SSIYY*/
/*TWO DIMENSIONAL ARRAY - YEAR BY MONTH*/
ARRAY PSTAYMM(&BEGYR.:&ENDYR.,1:12) %DO YEAR=&BEGYR. %TO &ENDYR.;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      PSTA&YR.01-PSTA&YR.12
      %END;;
ARRAY SSIYY (&BEGYR.:&ENDYR.) %DO YEAR=&BEGYR. %TO &ENDYR.;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      SSI&YR.
      %END;;

DO YR=&BEGYR. TO &ENDYR.;
  SSIYY(YR) = 0; *SET INITIAL SSIYY VALUE TO BE 0;
  DO MO=1 TO 12 UNTIL(SSIIYY(YR)=1);
    IF PSTAYMM(YR,MO) IN ("C01","M01","M02") THEN SSIYY(YR)=1;
  END;
END;

DROP MO;
RUN;

PROC PRINT DATA=OUT1.LONG (OBS=20); TITLE 'AFTER UPDATE'; RUN;
RUN;
PROC CONTENTS;
RUN;

%MEND PROCSSILF;
%PROCSSILF;
```

## Appendix A.56

JCL/SAS Code: MBRLOADA

```

//#6502LDA JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   50214 DAF
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.PRDLIB(MBRLOADA)
//* DESCRIPTION: CHECK THE CURRENT MBR DATA LAYOUT
//*           THIS PROGRAM LOADS MBR RECORDS RETURNED FROM
//*           MBR FINDER PRELIMINARY TO ADDING UP BENEFITS
//*           OUTPUT FILE WILL BE PROCESSED TO ADD UP DEPENDENT
//*           BENEFIT AMOUNTS SORTED BY CAN AND BIC AND OUTPUT
//*           COMBINED FILE.
//* PROGRAMMER: SERGE LUKASHANETS 4/7/2021 FOR DAF21
//* EMAIL:      ABRYCE@MATHEMATICA-MPR.COM
//*           ALEX.BRYCE@SSA.GOV
//*****
//JS010 EXEC SAS9,
//  WORK='200000,100000'          * SPACE IN BLKS *
//*
//IN1 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.MBR.F1.R220329,DISP=SHR
//  DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.MBR.F2.R220329,DISP=SHR
//  DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.MBR.F3.R220329,DISP=SHR
//  DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.MBR.F4.R220329,DISP=SHR
//  DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.MBR.F5.R220329,DISP=SHR
//TEMP DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP1 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP3 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP4 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBRLOADA.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//*
//SYSIN DD *
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE END YEAR AS NEEDED */

```

Appendix A.56

JCL/SAS Code: MBRLOADA

```
%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

/* EXTRACT THE PAYMENT FIELDS FOR USE IN COMPUTING DEPENDENT AMTS */
/* THE DATA WILL CONTAIN ALL AUXILIARIES */

DATA TEMP.MBR;

INFILE IN1 TRUNCOVER;
INPUT
  @001 CAN    $9.
  @061 BIC    $2.
  @305 BOAN   $9.
  @;

POS = 7784;
%DO MM=1 %TO &TOTMOS.;
  INPUT
    @POS MBC&&YY&MM 5.1 /* Monthly Benefit Credited */
    @;
  POS=POS+5;
%END;

POS = 12488;
%DO MM=1 %TO &TOTMOS.;
  INPUT
    @POS BPD&&YY&MM $1./* BENEFIT PAYMENT DESIGNATION */
    @;
  POS=POS+1;
%END;
;

/*CREATE SSN*/
IF SUBSTR(BIC,1,1) IN ('A') THEN SSN = CAN;
ELSE SSN = BOAN;

DROP POS;
RUN;

/* CHECK THAT NO DUPLICATE RECORDS ARE IN THE FILE */

%MACRO SORTD(BEGIN=,END=,RUN=);
PROC SORT DATA=TEMP.MBR (FIRSTOBS=&BEGIN. OBS=&END.)
```

Appendix A.56

JCL/SAS Code: MBRLOADA

```

OUT=TEMP&RUN..MBR;
BY CAN BIC;
RUN;
%MEND SORTD;
%SORTD(BEGIN=1,END=12000000,RUN=1);
%SORTD(BEGIN=12000001,END=24000000,RUN=2);
%SORTD(BEGIN=24000001,END=36000000,RUN=3);
%SORTD(BEGIN=36000001,END=MAX,RUN=4);

```

```

DATA OUT.MBR;
  SET TEMP1.MBR
    TEMP2.MBR
    TEMP3.MBR
    TEMP4.MBR;
  BY CAN BIC;
  IF LAST.BIC;
RUN;

```

```

PROC PRINT DATA=OUT.MBR (OBS=25);
  TITLE 'MBRLOADA';
RUN;

```

```

PROC FREQ DATA=OUT.MBR ; TABLES BIC; RUN;
RUN;

```

```
%MEND;
```

```
%START;
```

```

/*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>

```

Appendix A.56

JCL/SAS Code: MBRLOADA

SUBJECT: DAF21 MBRLOADA IS COMPLETE



## Appendix A.57

JCL/SAS Code: PHUSLOAD

```
//#6502PHU JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:  50214 DAF
//* DIRECTOR:  MATT URATO
//* PROGRAM:  OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.PRDLIB(PHUSLOAD)
//* DESCRIPTION: CHECK THE CURRENT PHUS DATA LAYOUT
//*          THIS PROGRAM LOADS PHUS RECORDS RETURNED FROM
//*          MBR FINDERPRELIMINARY TO ADDING UP BENEFITS
//*          OUTPUT FILE WILL BE PROCESSED TO ADD UP DEPENDENT
//*          BENEFIT AMOUNTS SORTED BY CAN AND BIC AND OUTPUT
//*          COMBINED FILE.
//* PROGRAMMER: ALEX BRYCE 4/7/2022 FOR DAF21
//* EMAIL:  ABRYCE@MATHEMATICA-MPR.COM
//*          ALEX.BRYCE@SSA.GOV
//*****
//*
//JS010 EXEC SAS9,
//  WORK='200000,100000'          * SPACE IN BLKS *
//*
//IN1 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.PHUS.F1.R220329,DISP=SHR
//  DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.PHUS.F2.R220329,DISP=SHR
//  DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.PHUS.F3.R220329,DISP=SHR
//  DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.PHUS.F4.R220329,DISP=SHR
//  DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.PHUS.F5.R220329,DISP=SHR
//TEMP DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP1 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP3 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP4 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP5 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP6 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP7 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP8 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PHUSLOAD.SA.V1,
```

## Appendix A.57

JCL/SAS Code: PHUSLOAD

```
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO
//*
//SYSIN DD *
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS=MAX;
%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE END YEAR AS NEEDED */
*START YEAR FOR PHUS RECORDS IS 1984;
%LET STARTPOS=%EVAL((&BEGYR-1984)*12);

%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

/* THE DATA WILL CONTAIN ALL AUXILIARIES */

DATA TEMP.PHUS;

INFILE IN1 TRUNCOVER;
INPUT
  @01 CAN $9.
  @10 BIC $2.
  @;
  POS = 39+(&STARTPOS.*7);
  %DO MM=1 %TO &TOTMOS.;
    INPUT
      @POS DIRPAY&&YY&MM ZD7.1 /* Direct Pay */
      @;
      POS=POS+7;
    %END;
    POS = 3231+(&STARTPOS.*5);
    %DO MM=1 %TO &TOTMOS.;
      INPUT
        @POS MEDPREM&&YY&MM ZD5.1/* Medicare Premium */
        @;
        POS=POS+5;
      %END;
    ;
  DROP POS MM;

  /* SUM DIRECT PAY AND MEDICARE PREMIUM TO GET THE TOTAL AMT PAID */
  ARRAY PAYD (*) %DO MM=1 %TO &TOTMOS.; PAYD&&YY&MM %END;;
  ARRAY DIRPAY (*) %DO MM=1 %TO &TOTMOS.; DIRPAY&&YY&MM %END;;
  ARRAY MEDPREM (*) %DO MM=1 %TO &TOTMOS.; MEDPREM&&YY&MM %END;;
```

## Appendix A.57

JCL/SAS Code: PHUSLOAD

```
DO MM = 1 TO &TOTMOS.;
  PAYD(MM) = SUM(DIRPAY(MM),MEDPREM(MM));
END;
RUN;

%MACRO SORTD(BEGIN=,END=,RUN=);
PROC SORT DATA=TEMP.PHUS(FIRSTOBS=&BEGIN. OBS=&END.)
OUT=TEMP&RUN..PHUS;
BY CAN BIC;
RUN;
%MEND SORTD;
%SORTD(BEGIN=1,      END=6000000,RUN=1);
%SORTD(BEGIN= 6000001,END=12000000,RUN=2);
%SORTD(BEGIN=12000001,END=18000000,RUN=3);
%SORTD(BEGIN=18000001,END=24000000,RUN=4);
%SORTD(BEGIN=24000001,END=30000000,RUN=5);
%SORTD(BEGIN=30000001,END=36000000,RUN=6);
%SORTD(BEGIN=36000001,END=40000000,RUN=7);
%SORTD(BEGIN=40000001,END=MAX,RUN=8);

DATA OUT.PHUS;
  SET TEMP1.PHUS
      TEMP2.PHUS
      TEMP3.PHUS
      TEMP4.PHUS
      TEMP5.PHUS
      TEMP6.PHUS
      TEMP7.PHUS
      TEMP8.PHUS;
  BY CAN BIC;
  IF LAST.BIC;
RUN;

PROC PRINT DATA=OUT.PHUS (OBS=25);
  TITLE 'PHUSLOAD';
RUN;
PROC FREQ DATA=OUT.PHUS ; TABLES BIC; RUN;
RUN;

%MEND;
%START;

/*
//*****
```

Appendix A.57

JCL/SAS Code: PHUSLOAD

```
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 PHUSLOAD IS COMPLETE
```

## Appendix A.58

JCL/SAS Code: MRGPHUS

```
//#6502MRG JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:  50214 DAF
//* DIRECTOR:  MATT URATO
//* PROGRAM:  OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.PRDLIB(MRGPHUS)
//* DESCRIPTION:
//*  1. MERGE PHUS DATA TO RESULT OF MBRLOAD1 BY CAN AND BIC
//*  2. THIS WILL ADD THE BOAN TO PHUS PRELIMINARY TO DOING
//*     MBRADDUP AND MBR SECOND READ - ADD TO CASES IN FINDER
//*  3. OUTPUT FULL FILE FOR ADDUP AND LINKED FILE FOR MBR
//* PROGRAMMER: ALEX BRYCE 4/07/2022 FOR DAF21
//* EMAIL:  ABRYCE@MATHEMATICA-MPR.COM
//*        ALEX.BRYCE@SSA.GOV
//* NOTE: IN DAF14 MRGPHUS1 AND MRGPHUS2 ARE COMBINED INTO MRGPHUS
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000'          * SPACE IN BLKS *
//*
//IN1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBRLOADA.SA.V1,DISP=SHR
//IN2 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PHUSLOAD.SA.V1,DISP=SHR
//LNK DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.LINKDI.SA.V1,DISP=SHR
//*
//TEMP DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP1 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO
//TEMP2 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO
//TEMP3 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO
//TEMP4 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO
//TEMP5 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO
//TEMP6 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
```

## Appendix A.58

JCL/SAS Code: MRGPHUS

```
// UNIT=TSILO
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBRPHUS.SA.V1,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO,VOL=(,,10)
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PHUSSSN.SA.V1,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO,VOL=(,,10)
//OUT3 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PHUSLINK.SA.V1,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO
//*
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=BINARY OBS=MAX;

/* MERGE PHUS ONTO MBRLOAD1 DATA AND OUTPUT MBRADDUP FILE AND LINKED
FILE TO MERGE ONTO 814 DATA */

DATA OUT1.MBRPHUS (DROP=BOAN)
  OUT2.PHUSSSN (KEEP=SSN BIC PAYD: DIRPAY: MEDPREM:);
MERGE IN1.MBR (IN=M)
  IN2.PHUS (IN=P);
BY CAN BIC;

LENGTH HASPHUS $1;
HASPHUS = 'N';
IF P THEN HASPHUS = 'Y';

IF M THEN DO;
  OUTPUT OUT1.MBRPHUS;
  IF SSN NE ' ' THEN OUTPUT OUT2.PHUSSSN;
END;
RUN;

PROC FREQ DATA=OUT1.MBRPHUS; TABLES HASPHUS; TITLE'MERGED PHUS & MBR';
RUN;
PROC PRINT DATA=OUT1.MBRPHUS (OBS=20);RUN;
PROC PRINT DATA=OUT2.PHUSSSN (OBS=20);RUN;

/*DIVIDE TEMP.PHUS IN ORDER TO SUCCEED IN SORTING IT */
%MACRO SORTDAT(BEGIN = ,END = ,RUN = );
PROC SORT DATA = OUT2.PHUSSSN(FIRSTOBS = &BEGIN. OBS=&END.)
  OUT = TEMP&RUN..PHUS;
  BY SSN BIC;
RUN;
%MEND SORTDAT;
```

## Appendix A.58

JCL/SAS Code: MRGPHUS

```
%SORTDAT(BEGIN = 1 ,    END = 10000000 ,RUN = 1);
%SORTDAT(BEGIN = 10000001 ,END = 20000000 ,RUN = 2);
%SORTDAT(BEGIN = 20000001 ,END = 30000000 ,RUN = 3);
%SORTDAT(BEGIN = 30000001 ,END = 40000000 ,RUN = 4);
%SORTDAT(BEGIN = 40000001 ,END = MAX ,    RUN = 5);
```

```
DATA TEMP6.PHUS;
  SET TEMP1.PHUS
      TEMP2.PHUS
      TEMP3.PHUS
      TEMP4.PHUS
      TEMP5.PHUS;
BY SSN BIC;
RUN;
```

```
PROC SORT DATA=LNK.LINKDI(DROP=CAN) OUT=LINKDI NODUPKEY;
BY SSN BIC;
RUN;
```

```
DATA OUT3.LINKPHUS;
MERGE TEMP6.PHUS(IN=P)
      LINKDI (IN=L);
BY SSN BIC;
IF L AND P;
RUN;
```

```
PROC PRINT DATA=OUT3.LINKPHUS (OBS=20);TITLE 'FINAL LINKED PHUS';
RUN;
```

```
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2  DD SYSOUT=(B,SMTP)
//SYSUT1  DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 MRGPHUS IS COMPLETE
```

## Appendix A.59

JCL/SAS Code: DUALPHUS

```
//#6502PHU JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT: DAF21 50214
//* DIRECTOR: MATT URATO
//* PROGRAM: OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.PRDLIB(DUALPHUS)
//* DESCRIPTION:
//*  1. OUTPUT SINGLE RECORDS IN SEPARATE FILE
//*  2. PROCESS DUAL RECORDS AND OUTPUT SUMMED DUALS FILE
//* CONTACT ALEX BRYCE 4/8/2022 FOR DAF21
//* E-MAIL: ABRYCE@MATHEMATICA-MPR.COM
//*  ALEX.BRYCE@SSA.GOV
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000'          * SPACE IN BLKS *
//*
//LNK DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PHUSLINK.SA.V1,DISP=SHR
//TEMP DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PHUSSING.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PHUSDUAL.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//*
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2021;

%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

DATA OUT1.LINKPHUS
  TEMP.PHUSDUAL;
SET LNK.LINKPHUS;
BY SSN;
IF FIRST.SSN AND LAST.SSN THEN OUTPUT OUT1.LINKPHUS;
ELSE OUTPUT TEMP.PHUSDUAL;
```



## Appendix A.59

JCL/SAS Code: DUALPHUS

```
RUN;

PROC PRINT DATA=TEMP.PHUSDUAL (OBS=100);
TITLE 'PHUSDUAL BEFORE PROCESSING';
RUN;

DATA OUT2.PHUSDUAL;
SET TEMP.PHUSDUAL
(RENAMED=(
%DO MM=1 %TO &TOTMOS.;
    PAYD&&YY&MM=TPAYD&&YY&MM
    MEDPREM&&YY&MM=TMEDPREM&&YY&MM
    DIRPAY&&YY&MM=TDIRPAY&&YY&MM
%END;))
;
BY SSN;

/* INPUT ARRAYS - RENAMED VARS */
ARRAY TPAYD(*) %DO MM=1 %TO &TOTMOS.; TPAYD&&YY&MM %END;;
ARRAY TMEDPREM(*) %DO MM=1 %TO &TOTMOS.; TMEDPREM&&YY&MM %END;;
ARRAY TDIRPAY(*) %DO MM=1 %TO &TOTMOS.; TDIRPAY&&YY&MM %END;;

/* OUTPUT ARRAYS - SUMMED PAYMENTS - ORIGINAL NAME */
ARRAY PAYD(*) %DO MM=1 %TO &TOTMOS.; PAYD&&YY&MM %END;;
ARRAY MEDPREM(*) %DO MM=1 %TO &TOTMOS.; MEDPREM&&YY&MM %END;;
ARRAY DIRPAY(*) %DO MM=1 %TO &TOTMOS.; DIRPAY&&YY&MM %END;;

RETAIN %DO MM=1 %TO &TOTMOS.;
    PAYD&&YY&MM
    MEDPREM&&YY&MM
    DIRPAY&&YY&MM
%END;;

DO MM = 1 TO DIM(PAYD);
/* INITIALIZE */
IF FIRST.SSN THEN DO;
    PAYD(MM) = 0;
    MEDPREM(MM) = 0;
    DIRPAY(MM) = 0;
END;
/* SUM THE AMOUNTS ON SUBSEQUENT RECORDS */
PAYD(MM)=SUM(PAYD(MM),TPAYD(MM));
MEDPREM(MM)=SUM(MEDPREM(MM),TMEDPREM(MM));
DIRPAY(MM)=SUM(DIRPAY(MM),TDIRPAY(MM));
END;
```

Appendix A.59

JCL/SAS Code: DUALPHUS

```
DROP TPAYD: TMEDPREM: TDIRPAY: MM;
IF LAST.SSN;

RUN;

PROC PRINT DATA=OUT2.PHUSDUAL (OBS=20);
TITLE 'PHUSDUAL AFTER PROCESSING';
RUN;

PROC PRINT DATA=OUT1.LINKPHUS (OBS=20);
TITLE 'PHUSSING AFTER PROCESSING';
RUN;
%MEND;
%START;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 DUALPHUS IS COMPLETE
```

## Appendix A.60

JCL/SAS Code: MBRADDUP

```
//#6502MAD JOB (11710000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   50214 DAF
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.PRDLIB(MBRADDUP)
//* DESCRIPTION: MBR RECORDS WERE MERGED WITH PHUS RECORDS TO
//*            CREATE MBRPHUS.
//*            ON MBRPHUS RECORDS - ADD UP NUMBER OF DEPENDENTS
//*            ADD UP BENEFIT AMOUNTS FOR TOTAL DEPENDENT AMOUNT
//*            ADD UP TOTAL PAYMENT AMOUNTS FOR DEPENDENTS (PHUS)
//*            FILE SORTED BY CAN BIC FOR LINKING TO MBR
//*            ORIGINAL RECORDS
//* PROGRAMMER: ALEX BRYCE 4/8/2022 FOR DAF21
//* EMAIL:     ABRYCE@MATHEMATICA-MPR.COM
//*           ALEX.BRYCE@SSA.GOV
//*****
//*
// SET REG='128M'
//JS010 EXEC SAS9,
//      WORK='200000,100000',          * SPACE IN BLKS *
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
//IN1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBRPHUS.SA.V1,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.SUMREC.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     SPACE=(CYL,(1000,100),RLSE),VOL=(,,5)
//TEMP1 DD DSN=&&TEMP,
//     DISP=(NEW,DELETE,DELETE),
//     UNIT=TSILO,(VOL=,,5)
//TEMP2 DD DSN=&&TEMP,
//     DISP=(NEW,DELETE,DELETE),
//     UNIT=TSILO,(VOL=,,5)
//SYSIN DD *

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS = MAX;
%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE END YEAR AS NEEDED */

%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";
```

## Appendix A.60

JCL/SAS Code: MBRADDUP

```
DATA TEMP1.MBR (KEEP=CAN
    %DO MM=1 %TO &TOTMOS.;DUEO&&YY&MM
        PAYO&&YY&MM
        DPEN&&YY&MM %END;)
    TEMP2.IDENT (KEEP=CAN SSN BIC);
/* THE FILE IS SORTED WITH THE PRIMARY (BIC=A) FIRST */
/* SUBSEQUENT RECORDS FOR A GIVEN CAN ARE DEPENDENTS */
/* OUTPUT FILE WILL HOLD DEPENDENT COUNTS AND AMOUNTS*/
/* THESE WILL BE ATTACHED TO THE PRIMARIES IN DAF */
/* IF FIRST BICS ARE NOT = 'A' THEY ARE DACS OR WIDOWS
    AND WILL NOT BE PROCESSED IN THE ADDUP PROCEDURE */

SET IN1.MBRPHUS; BY CAN BIC;

RETAIN %DO MM=1 %TO &TOTMOS.;
    DUEO&&YY&MM DPEN&&YY&MM PAYO&&YY&MM %END;;

ARRAY DUEO (*) %DO MM=1 %TO &TOTMOS.; DUEO&&YY&MM %END;;
ARRAY PAYO (*) %DO MM=1 %TO &TOTMOS.; PAYO&&YY&MM %END;;
ARRAY DPEN (*) %DO MM=1 %TO &TOTMOS.; DPEN&&YY&MM %END;;
ARRAY BPD (*) %DO MM=1 %TO &TOTMOS.; BPD&&YY&MM %END;;
ARRAY MBC (*) %DO MM=1 %TO &TOTMOS.; MBC&&YY&MM %END;;
ARRAY PAYD (*) %DO MM=1 %TO &TOTMOS.; PAYD&&YY&MM %END;;

IF FIRST.CAN THEN DO;

    DO MM = 1 TO DIM(DUEO);

        /* INITIALIZE THE FIELDS ON THE PRIMARY RECORD */

        DUEO(MM) = 0; /* SUMMED BENEFIT AMOUNT */
        PAYO(MM) = 0; /* SUMMED PAYED AMOUNT */
        DPEN(MM) = 0; /* SUMMED NUMBER OF DEPENDENTS */
    END;

    /* IF THE FIRST RECORD IS THE ONLY RECORD THE DEPENDENT VARS
        WILL = 0 */

    /* THE TEMP2.IDENT FILE WILL HAVE RECORDS ONLY FOR PRIMARIES
        IF THE FIRST BIC IS NOT A PRIMARY - AN ODD OCCURRENCE- THE
        RECORD IS NOT OUTPUT */

    IF SUBSTR(BIC,1,1) EQ 'A' THEN OUTPUT TEMP2.IDENT;
```

Appendix A.60

JCL/SAS Code: MBRADDUP

```

END;

ELSE DO; /* ADD UP THE BENEFITS FOR THE AUXILIARIES */

DO MM = 1 TO DIM(DUEO);
IF BPD(MM) = '1' THEN DO; /* BENEFIT PAID INDICATOR */
DUEO(MM) = SUM(DUEO(MM),MBC(MM));
DPEN(MM) = SUM(DPEN(MM),1);
PAYO(MM) = SUM(PAYO(MM),PAYD(MM));
END;
END;
END;
IF LAST.CAN THEN OUTPUT TEMP1.MBR;
RUN;

DATA OUT.MBR;
MERGE TEMP1.MBR TEMP2.IDENT (IN=I);
BY CAN;
/* ONLY PRIMARY RECORDS OUTPUT IN STEP 1 ARE KEPT AND MERGED WITH
DEPENDENT AMOUNTS */
IF I;
NOSSN = 0;
IF SSN = ' ' THEN NOSSN = 1;
RUN;

PROC FREQ DATA=OUT.MBR; TABLES NOSSN ; RUN;

%MEND;

%START;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 MBRADDUP IS COMPLETE

```

## Appendix A.61

JCL/SAS Code: MBRLOADB

```

//#6502LDB JOB (11710000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   50214 DAF
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF20.MBR.PRDLIB(MBRLOADB)
//* DESCRIPTION: SAS LOAD 5 SECTIONS OF MBR DATA
//*      ALL MBR FIELDS LOADED
//* DATE:      XIAO BARRY
//* UPDATE:    ALEX BRYCE 4/8/2022
//* EMAIL:     ABRYCE@MATHEMATICA-MPR.COM
//*      ALEX.BRYCE@SSA.GOV
//*****
//*
//MRGPROC PROC
//JS010 EXEC SAS9,
//      WORK='200000,100000',          * SPACE IN BLKS *
//      PARM='SYSPARM="&SEG"'
//*
//IN1 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF21.MBR.F&SEG..R220329,DISP=SHR
//IN2 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.LINKDI.SA.V1,DISP=SHR
//TEMP DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//TEMP1 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//TEMP2 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//TEMP3 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//TEMP4 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//TEMP5 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR&SEG..SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.PRDLIB(MBRREAD),
//      DISP=(SHR,PASS,KEEP)

```

Appendix A.61

JCL/SAS Code: MBRLOADB

```
// PEND
//*
//RUN1 EXEC MRGPROC,SEG=1
//RUN2 EXEC MRGPROC,SEG=2
//RUN3 EXEC MRGPROC,SEG=3
//RUN4 EXEC MRGPROC,SEG=4
//RUN5 EXEC MRGPROC,SEG=5
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 MBRLOADB SEGMENT ? IS COMPLETE
```

## Appendix A.62

JCL/SAS Code: MBRREAD

```

/*****
* FILENAME: MBRREAD.SAS
* PROGRAMMER: MIRIAM LOEWENBERG
* PURPOSE: TO READ IN THE MBR DATA
* EDITED: 06/08/16 BY XIAO BARRY
* THE INPUT FILE IS IN 5 SECTIONS
* THERE ARE DUAL ELIGIBLES AMONG THE RECORDS AND THERE WILL
* BE DUPLICATE SSN/BIC COMBINATIONS - THESE WILL BE PROCESSED
* SEPARATELY AS OUTPUT FROM PROGRAM LOAD2COM
* UPDATE: ALEX BRYCE 04/08/2021
*****/
OPTIONS NOCENTER COMPRESS=BINARY OBS=MAX;
%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE END YEAR AS NEEDED */

%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";
DATA TEMP.MBR;
INFILE IN1 TRUNCOVER;
INPUT
  @001 CAN $9.
  @010 TAC $1. /* TYPE OF AWARD CODE */
  @011 RP $1. /* RACE OF PRIMARY */
  @012 LSPA 5.2 /* LUMP SUM PAYMENT AMOUNT */
  @019 PNOB $2. /* NUMBER OF BENES IN PAYMENT */
  @029 DOC_SSDI_MR $3. /* DISTRICT OFFICE CODE */
  @032 STATE $2.
  @034 COUNTY $3.
  @042 LSSC_DSALLW $1. /* LUMP SUM STATUS CODE -DISALLOWED */
  @043 LSSC_FUNR $1. /* LUMP SUM STAT CODE -FUNERAL HOME */
  @044 LSSC_BURL $1. /* LUMP SUM STAT CODE -BURIAL EXP */
  @045 LSSC_LVSPS $1. /* LUMP SUM STAT CODE -LIVING W/SPOUSE*/
  @046 LSSC_AUTPMT $1. /* LUMP SUM STAT CODE -AUTHORIZED PYMT*/
  @047 LSSC_WID $1. /* LUMP SUM STAT CODE -ENTITLED WIDOW*/
  @048 LSSC_CHLD $1. /* LUMP SUM STAT CODE -ENTITLED CHILD*/
  @049 LSFDX $6. /* LUMP SUM FILING DATE */
  @055 LSAP 5.2 /* LUMP SUM AWARD AMOUNT */
  @060 LSDC $1. /* LUMP SUM DISALLOWANCE CODE */
  @061 BIC $2. /* BENE IDENTIFICATION CODE */
  @063 LAF $2. /* LEDGER ACCOUNT FILE */
  @065 DOBMBR ??MMDDYY8. /* DATE OF BIRTH */
  @073 BGN $15. /* BENE GIVEN NAME */
  @088 BLN $20. /* BENE LAST NAME */

```



## Appendix A.62

JCL/SAS Code: MBRREAD

```
@108 DOEIX $6. /* EARLIEST MONTH ENTITLED TO BENEFITS */
@114 DOECX $6. /* DATE OF CURRENT ENTITLEMENT */
@120 CEC $1. /* CURRENT ENTITLEMENT CODE */
@129 DOSTX $6. /* DATE OF SUSPENSION/TERMINATION */
@135 MBRSEX $1. /* GENDER */
@136 RELATC $1. /* RELATIONSHIP CODE */
@137 MBP 5.1 /* MONTHLY BENEFIT PAID */
@142 LANG $1. /* WRITTEN LANGUAGE */
@143 TOC_NUM 2. /* NUMBER OF TOC OCCURRENCES */
@;
POS=145;
%DO I=1 %TO 20;
INPUT
@POS TOC&I $1. /* TYPE OF CLAIM */
@;
POS=POS+1;
%END;
POS=165;
%DO I=1 %TO 20;
INPUT
@POS TOC_STARX&I $6. /* TOC START DATE */
@;
POS=POS+6;
%END;
INPUT @285 RDD $3. /* REASON FOR DISALLOWANCE/DENIAL */
@288 TOP $1. /* TYPE OF PAYEE */
@289 CC $1. /* TYPE OF PAYEE */
@290 GS $1. /* GUARDIAN STATUS */
@291 RZIP $5. /* RESIDENCE ZIP CODE */
@296 SIFT $1. /* SECURITY INCOME FILE TYPE */
@297 SIEDX $6. /* SSI TERMINATION DATE */
@303 SLAC $1. /* SSI LIVING ARRANG CODE */
@304 SISC $1. /* SSI INCOME STATUS CODE */
@305 BOAN $9. /* BENE OWN ACCOUNT NUMBER */
@314 DODMBR ??MMDDYY8. /* DATE OF DEATH */
@322 NDOF 2. /* NUM OCCURRENCES DATE IF FILING */
@;
POS=324;
%DO I=1 %TO 25;
INPUT
@POS BDOF&I ??MMDDYY8. /* BENEFICIARY DATE OF FILING */
@;
POS=POS+8;
%END;
```

Appendix A.62

JCL/SAS Code: MBRREAD

```
POS=524;
%DO I=1 %TO 25;
  INPUT
  @POS BDOE_STARTX&I $6. /* BDOE START DATE */
  @;
  POS=POS+6;
%END;
```

```
POS=674;
%DO I=1 %TO 25;
  INPUT
  @POS BDOE_TERM X&I $6. /* BDOE TERM DATE */
  @;
  POS=POS+6;
%END;
```

```
POS=824;
%DO I=1 %TO 25;
  INPUT
  @POS HBIC&I $2. /* HISTORIC BIC */
  @;
  POS=POS+2;
%END;
```

```
POS=874;
%DO I=1 %TO 25;
  INPUT
  @POS HCEC&I $1. /* HISTORIC CEC */
  @;
  POS=POS+1;
%END;
```

```
INPUT @899 NHI 2. /* NUMBER OF HI OCCURENCE */
@;
POS=901;
%DO I=1 %TO 10;
  INPUT
  @POS HI_STARTX&I $6. /* HI START DATE */
  @;
  POS=POS+6;
%END;
```

```
POS=961;
%DO I=1 %TO 10;
  INPUT
```

Appendix A.62

JCL/SAS Code: MBRREAD

```
@POS HI_TERM&I $6. /* DATE OF HI TERMINATION */
@;
POS=POS+6;
%END;

INPUT @1021 NSMI 2. /* NUMBER OF SMI OCCURENCES */
@;

POS=1023;
%DO I=1 %TO 20;
INPUT
@POS SMI_STAX&I $6. /* SMI ENROLLMENT START DATE */
@;
POS=POS+6;
%END;

POS=1143;
%DO I=1 %TO 20;
INPUT
@POS SMI_TERX&I $6. /* DATE OF SMI TERMINATION */
@;
POS=POS+6;
%END;

INPUT @1263 NPIA 3.
@;

IF SUBSTR(BIC,1,1) IN ('A') THEN SSN=CAN;
ELSE SSN=BOAN;

POS=1266;
%DO I=1 %TO 150;
INPUT
@POS PIEDX&I $6. /* PIA EFFECTIVE DATE */
@;
POS=POS+6;
%END;

POS=2166;
%DO I=1 %TO 150;
INPUT
@POS PIA&I 5.1 /* PIA AMOUNT */
@;
POS=POS+5;
%END;
```

Appendix A.62

JCL/SAS Code: MBRREAD

```
POS=2916;
%DO I=1 %TO 150;
  INPUT
  @POS IME&I 4. /* AVERAGE INDEXED MONTHLY EARNINGS */
  @;
  POS=POS+4;
%END;
```

```
POS=3516;
%DO I=1 %TO 150;
  INPUT
  @POS PIARFC&I $1. /* REASON FOR CHANGE IN PIA */
  @;
  POS=POS+1;
%END;
```

```
INPUT @5016 DLMX $6. /* DATE LAST INSURED FOR DISABILITY */
  @5022 NODF 2.
  @;
POS=5024;
%DO I=1 %TO 12;
  INPUT
  @POS DDO&I MMDDYY8. /* DATE OF DISABILITY ONSET */
  @;
  POS=POS+8;
%END;
```

```
POS=5120;
%DO I=1 %TO 12;
  INPUT
  @POS DAC&I $1. /* DISABILITY AWARD CODE */
  @;
  POS=POS+1;
%END;
```

```
POS=5132;
%DO I=1 %TO 12;
  INPUT
  @POS LOD&I $1. /* LEVEL OF DENIAL CODE */
  @;
  POS=POS+1;
%END;
```

```
POS=5144;
```

Appendix A.62

JCL/SAS Code: MBRREAD

```
%DO I=1 %TO 12;
  INPUT
  @POS DOEDX&I $6. /* DATE OF ENTITLEMENT TO DIB */
  @;
  POS=POS+6;
%END;

POS=5216;
%DO I=1 %TO 12;
  INPUT
  @POS T2_DOEDX&I $6. /* T2 DATE OF ENTITLEMENT TO DIB */
  @;
  POS=POS+6;
%END;

POS=5288;
%DO I=1 %TO 12;
  INPUT
  @POS DDBCX&I $6. /* DATE OF DISAB BENEFIT CESSATION */
  @;
  POS=POS+6;
%END;

POS=5360;
%DO I=1 %TO 12;
  INPUT
  @POS DSDX&I $6. /* DISAB ADJUDICATION DATE */
  @;
  POS=POS+6;
%END;

POS=5432;
%DO I=1 %TO 12;
  INPUT
  @POS HDDX&I $6. /* HEARING DECISION DATE */
  @;
  POS=POS+6;
%END;

POS=5504;
%DO I=1 %TO 12;
  INPUT
  @POS SDSX&I $6. /* SGA DISAB CESSATION */
  @;
  POS=POS+6;
```

Appendix A.62

JCL/SAS Code: MBRREAD

```
%END;
```

```
POS=5576;  
%DO I=1 %TO 12;  
  INPUT  
  @POS  ADCX&I  $6. /* APPLICANTS DISAB CESSATION */  
  @;  
  POS=POS+6;  
%END;
```

```
POS=5648;  
%DO I=1 %TO 12;  
  INPUT  
  @POS  APSX&I  $6. /* APPEALS */  
  @;  
  POS=POS+6;  
%END;
```

```
POS=5720;  
%DO I=1 %TO 12;  
  INPUT  
  @POS  EBDX&I  $6. /* EPE BEGIN DATE */  
  @;  
  POS=POS+6;  
%END;
```

```
POS=5792;  
%DO I=1 %TO 12;  
  INPUT  
  @POS  BDC&I  $2. /* BASIS FOR DENIAL CODE */  
  @;  
  POS=POS+2;  
%END;
```

```
POS=5816;  
%DO I=1 %TO 12;  
  INPUT  
  @POS  CDR&I  $1. /* CESSATION DISAB REASON */  
  @;  
  POS=POS+1;  
%END;
```

```
POS=5828;  
%DO I=1 %TO 12;
```

## Appendix A.62

JCL/SAS Code: MBRREAD

```
INPUT
@POS CSA&I $1. /* CURRENT SGA */
@;
POS=POS+1;
%END;

POS=5864;
%DO I=1 %TO 12;
INPUT
@POS TDXPRI_MBR&I $10. /* PRIMARY DIAGNOSIS */
@;
POS=POS+10;
%END;

POS=5984;
%DO I=1 %TO 12;
INPUT
@POS TDXSEC_MBR&I $10. /* SECONDARY DIAGNOSIS */
@;
POS=POS+10;
%END;

POS=6104;
%DO MM=1 %TO &TOTMOS.;
INPUT
@POS MBA&&YY&MM 5.1 /* MONTHLY BENEFIT AMOUNT */
@;
POS=POS+5;
%END;

POS=7784;
%DO MM=1 %TO &TOTMOS.;
INPUT
@POS DUED&&YY&MM 5.1 /* MONTHLY BENEFIT CREDITED */
@;
POS=POS+5;
%END;

POS=9464;
%DO MM=1 %TO &TOTMOS.;
INPUT
@POS MBP&&YY&MM 5.1 /* MONTHLY BENEFIT PAID */
@;
POS=POS+5;
%END;
```

Appendix A.62

JCL/SAS Code: MBRREAD

```
POS=11144;
%DO MM=1 %TO &TOTMOS.;
  INPUT
  @POS HSA&&YY&MM 4.1 /* HISTORICAL SMI AMOUNT */
  @;
  POS=POS+4;
%END;
```

```
POS=12488;
%DO MM=1 %TO &TOTMOS.;
  INPUT
  @POS BPD&&YY&MM $1. /* BENEFIT PAID DESIGNATION */
  @;
  POS=POS+1;
%END;
```

```
POS=12824;
%DO MM=1 %TO &TOTMOS.;
  INPUT
  @POS RFD&&YY&MM $1. /* REASON FOR DEDUCTION */
  @;
  POS=POS+1;
%END;
```

```
POS=13160;
%DO MM=1 %TO &TOTMOS.;
  INPUT
  @POS WIC&&YY&MM $1. /* WORK INDICATION CODE */
  @;
  POS=POS+1;
%END;
```

```
POS=13496;
%DO MM=1 %TO &TOTMOS.;
  INPUT
  @POS RFST&&YY&MM $6. /* REASON FOR SUSPENSION */
  @;
  POS=POS+6;
%END;
```

```
POS=15512;
%DO MM=1 %TO &TOTMOS.;
  INPUT
  @POS LAF&&YY&MM $2. /* MONTHLY LAF STATUS */
```



Appendix A.62

JCL/SAS Code: MBRREAD

```
@;
  POS=POS+2;
%END;

INPUT @16184 NHDEMO 2. /* NUMBER OF DEMONSTR. PROJECT OCCURENCE */
  @;

  POS=16186;
  %DO I=1 %TO 10;
    INPUT
      @POS DPI&I $2. /* DEMONSTRATION PROJECT INDICATOR */
      @;
      POS=POS+2;
  %END;

  POS=16206;
  %DO I=1 %TO 10;
    INPUT
      @POS DPI_STARX&I $6. /* DPI START DATE */
      @;
      POS=POS+6;
  %END;

  POS=16266;
  %DO I=1 %TO 10;
    INPUT
      @POS DPI_STOPX&I $6. /* DPI STOP DATE */
      @;
      POS=POS+6;
  %END;

  INPUT
    @16326 NINSND 2. /* NUMBER OF INSURED STATUS DATA OCCURENCES*/
    @;
    POS=16328;
    %DO I=1 %TO 15;
      INPUT
        @POS DIBREQ&I $2. /* DIB QUARTERS OF COVERAGE REQUIRED */
        @;
        POS=POS+2;
      %END;

    POS=16358;
    %DO I=1 %TO 15;
      INPUT
```

## Appendix A.62

JCL/SAS Code: MBRREAD

```

@POS DIBHAS&I $2. /* DIB QUARTERS OF COVERAGE POSSESSED */
@;
POS=POS+2;
%END;
;
/* CHANGE THE MBR14 DATES FROM CHARACTER TO NUMERIC */

ARRAY OLD LSFDX DOEDX1-DOEDX12 SIEDX
      DDBCX1-DDBCX12 DSDX1-DSDX12 HDDX1-HDDX12 SDSX1-SDSX12
      ADCX1-ADCX12 APSX1-APSX12 EBDX1-EBDX12
      DOEIX DOECX
      DOSTX PIEDX1-PIEDX150
      HI_STARTX1-HI_STARTX10 HI_TERM1-HI_TERM10
      SMI_STAX1-SMI_STAX20 SMI_TERX1-SMI_TERX20
      TOC_STARX1-TOC_STARX20 DLMX T2_DOEDX1-T2_DOEDX12
      DPI_STARX1-DPI_STARX10 DPI_STOPX1-DPI_STOPX10
      BDOE_STARTX1-BDOE_STARTX25 BDOE_TERM1-BDOE_TERM25;

ARRAY NEW LSFDF ENTDAT1-ENTDAT12 SIED
      DDBC1-DDBC12 DSD1-DSD12 HDD1-HDD12 SDS1-SDS12
      ADC1-ADC12 APS1-APS12 EBD1-EBD12
      DOI DOEC
      DOST PIED1-PIED150
      HI_START1-HI_START10 HI_TERM1-HI_TERM10
      SMI_START1-SMI_START20 SMI_TERM1-SMI_TERM20
      TOC_START1-TOC_START20 DLM T2_DOED1-T2_DOED12
      DPI_START1-DPI_START10 DPI_STOP1-DPI_STOP10
      BDOE_START1-BDOE_START25 BDOE_TERM1-BDOE_TERM25;

DO I=1 TO DIM(OLD);
  NEW(I)=INPUT('01' || OLD(I),??DDMMYY8.);
END;

/*CHANGE DIAGNOSIS LENGTH TO 4*/

ARRAY DIAGOLD (*) $ TDXPRI_MBR1-TDXPRI_MBR12 TDXSEC_MBR1-TDXSEC_MBR12;
ARRAY DIAGNEW (*) $4 DXPRI_MBR1-DXPRI_MBR12 DXSEC_MBR1-DXSEC_MBR12;

DO I=1 TO DIM(DIAGOLD);
  DIAGNEW(I) =SUBSTRN(LEFT(TRIM(COMPRESS(DIAGOLD(I)))),1,4);
END;

DROP LSFDF DOEDX1-DOEDX12 SIEDX
      DOEIX DOECX I POS
      DDBCX1-DDBCX12 DSDX1-DSDX12 HDDX1-HDDX12 SDSX1-SDSX12

```

## Appendix A.62

JCL/SAS Code: MBRREAD

```
HI_STARTX1-HI_STARTX10 HI_TERM1-HI_TERM10
SMI_STAX1-SMI_STAX20 SMI_TER1-SMI_TER20
ADCX1-ADCX12 APSX1-APSX12 EBDX1-EBDX12
DOSTX PIEDX1-PIEDX150
TOC_STARX1-TOC_STARX20 DLMX T2_DOEDX1-T2_DOEDX12
DPI_STARX1-DPI_STARX10 DPI_STOPX1-DPI_STOPX10
TDXPRI_MBR1-TDXPRI_MBR12 TDXSEC_MBR1-TDXSEC_MBR12
BDOE_STARTX1-BDOE_STARTX25 BDOE_TERM1-BDOE_TERM25;
RUN;

%MACRO SORTD(BEGIN=,END=,RUN=);
PROC SORT DATA=TEMP.MBR(FIRSTOBS=&BEGIN. OBS=&END.)
OUT=TEMP&RUN..MBR;
BY SSN BIC;
RUN;
%MEND SORTD;
%SORTD(BEGIN=1,END=2500000,RUN=1);
%SORTD(BEGIN=2500001,END=5000000,RUN=2);
%SORTD(BEGIN=5000001,END=MAX,RUN=3);

DATA TEMP4.MBR;
  SET TEMP1.MBR
    TEMP2.MBR
    TEMP3.MBR;
BY SSN BIC;
RUN;

/* SSN STAFF INDICATED THAT 2% OF RECORDS MAY HAVE THE WRONG
BOANS */

PROC SORT DATA=IN2.LINKDI NODUPKEY
OUT=TEMP5.LINKDI;
BY SSN BIC;
RUN;

DATA OUT.MBR;
MERGE TEMP4.MBR (IN=M) TEMP5.LINKDI (KEEP=SSN BIC IN=LINK);
BY SSN BIC;
IF M AND LINK;
RUN;

PROC CONTENTS DATA=OUT.MBR; RUN;

/* CHANGE FOR TRF07 - WE ARE GOING TO PROCESS THE DUAL ELIGIBLES
AFTER THE SEGMENTS HAVE BEEN COMBINED. THERE WILL BE DUPLICATE
```

Appendix A.62

JCL/SAS Code: MBRREAD

SSN/BIC COMBINATIONS IN EACH SEGMENT. THESE DUPLICATE RECORDS  
WILL BE PROCESSED SEPARATELY AND THE INFORMATION COMBINED INTO  
1 RECORD. \*/

```
PROC FREQ DATA=OUT.MBR ;TABLES BIC CEC TOC1 LAF NPJA NHDEMO; RUN;
PROC PRINT DATA=OUT.MBR (OBS=10 );
FORMAT ENTDAT1-ENTDAT12 LSF D BDOF1-BDOF25 DOBMBR DODMBR DDO1-DDO12
      DDBC1-DDBC12 DSD1-DSD12 HDD1-HDD12 SDS1-SDS12
      ADC1-ADC12 APS1-APS12 EBD1-EBD12
      DOEI DOEC SIED
      DOST PIED1-PIED150 DLM TOC_START1-TOC_START20 T2_DOED1-T2_DOED12
      HI_START1-HI_START10 HI_TERM1-HI_TERM10
      DPI_START1-DPI_START10 DPI_STOP1-DPI_STOP10
      SMI_START1-SMI_START20 SMI_TERM1-SMI_TERM20 YYMMDD10.;
RUN;

%MEND;
%START;
```

## Appendix A.63

JCL/SAS Code: LOADBCOM

```

//#6502LD2 JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:  50214 DAF
//* DIRECTOR:  MATT URATO
//* PROGRAM:  OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.PRDLIB(LOADBCOM)
//* DESCRIPTION:
//*  1. COMBINE EACH SEGMENT OF MBR LOAD2 FILE (READMBR)
//*  2. OUTPUT SINGLE RECORDS IN 1 FILE AND DUAL ELIG IN ANOTHER
//*  2. SORTED BY SSN-BIC
//*  3. READ814 SEGMENTS WERE SORTED BY SSN/BIC
//* PROGRAMMER:  ALEX BRYCE 4/10/2022 FOR DAF21
//* EMAIL:  ABRYCE@MATHEMATICA-MPR.COM
//*  ALEX.BRYCE@SSA.GOV
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000'          * SPACE IN BLKS *
//*
//IN1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR1.SA.V1,DISP=SHR
//IN2 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR2.SA.V1,DISP=SHR
//IN3 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR3.SA.V1,DISP=SHR
//IN4 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR4.SA.V1,DISP=SHR
//IN5 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR5.SA.V1,DISP=SHR
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.SING.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.DUALS.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=YES;
DATA OUT1.MBR OUT2.DUALS;
SET IN1.MBR
  IN2.MBR
  IN3.MBR
  IN4.MBR
  IN5.MBR;
  BY SSN BIC;
  WHERE SSN NE ' ';
  IF FIRST.SSN AND LAST.SSN THEN OUTPUT OUT1.MBR;
  ELSE OUTPUT OUT2.DUALS;
RUN;

```

Appendix A.63

JCL/SAS Code: LOADBCOM

```
PROC PRINT DATA=OUT1.MBR (OBS=25); TITLE 'SINGLE RECORDS';RUN;
PROC PRINT DATA=OUT2.DUALS (OBS=25); TITLE 'DUAL RECORDS'; RUN;
PROC FREQ DATA=OUT1.MBR; TABLES BIC TOC1; RUN;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 LOADBCOM IS COMPLETE
```

## Appendix A.64

JCL/SAS Code: PROC DUAL

```
//#6502DUA JOB (11710000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*      RESTART=JS020.SAS9
//*****
//* PROJECT:  50214 DAF
//* DIRECTOR:  MATT URATO
//* PROGRAM:  OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.PRDLIB(PROCDUAL)
//* DECRPTION:  PROCESS DUAL ELIGIBLE RECORDS
//*      ALL MBR FIELDS LOADED
//* PROGRAMMER:  ALEX BRYCE 4/10/2021 FOR DAF21
//*      ABRYCE@MATHEMATICA-MPR.COM
//*      ALEX.BRYCE@SSA.GOV
//*****
//*
//JS010 EXEC SAS9,
//      WORK='200000,100000'          * SPACE IN BLKS *
//*
//IN  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.DUALS.SA.V1,DISP=SHR
//TEMP DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP1 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP2 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,,10)
//TEMP31 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP32 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP33 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,,10)
//TEMP5 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,,10)
//TEMP6 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP7 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
```

## Appendix A.64

JCL/SAS Code: PROC DUAL

```
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP8 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP9 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP10 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP11 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP12 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//TEMP13 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(5000,5000),RLSE),VOL=(,,,10)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PROCDUAL.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=BINARY OBS = MAX;
%LET BEGYR=1994;
%LET ENDYR=2021;

%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

PROC PRINT DATA=IN.DUALS (OBS=20); TITLE 'BEFORE PROCESSING';
VAR SSN CAN BIC NODF DOEI DOEC ENTDAT1-ENTDAT3 PIED1 DUED: BPD;;
RUN;

/* DELETE CASES WITH NO DISABILITY DATA.
   PUT ASIDE NEWLY CREATED SINGLE RECORDS AFTER DELETIONS
   ADD THESE TO THE FINAL PROCESSED FILE */
DATA TEMP1.FINAL SINGLE ;
SET IN.DUALS(WHERE = (NODF NE 0));
BY SSN BIC;
IF FIRST.SSN AND LAST.SSN THEN OUTPUT SINGLE;
ELSE OUTPUT TEMP1.FINAL;
RUN;
```



## Appendix A.64

JCL/SAS Code: PROC DUAL

```
/* TAKE THE MOST RECENT RECORD (LATEST DOEC) - THE OTHER RECORDS ARE
   CONSIDERED SUPPLEMENTARY. CERTAIN VARIABLES ARE RETAINED
   FROM THE SUPPLEMENTARY RECORDS AND RENAMED ENDING IN D.
   FOR THE CASES WHERE DOEC IS THE SAME ON BOTH RECORDS USE THE
   MOST RECENT (LATEST) FIRST ENTDATE.
   FOR THE CASES WHERE ENTDATE IS THE SAME ON BOTH RECORDS USE THE
   RECORD WITH LATEST PIED1.
   FOR THE REMAINDER SELECT THE RECORD WITH THE BIC = A.
   FOR THE FEW RECORDS REMAINING SORT BY BIC, PROCESS THE BENEFIT
   AMOUNTS, AND OUTPUT THE LAST RECORD.
   */

DATA TEMP1.FINAL FLAGDATA (KEEP=SSN DOECFLAG ENTFLAG PIEDFLAG BICAFLAG);
SET TEMP1.FINAL;
BY SSN BIC;

/* GET THE FIRST ENTDATE - SOMETIMES THE FIRST IS BLANK AND A
   SUCCESSION ONE IS POPULATED */
FIRSTENT = .;
ARRAY ENT(*) ENTDAT1-ENTDAT12;
DO I = 1 TO DIM(ENT) UNTIL (ENT(I) GT .Z);
  FIRSTENT = ENT(I);
END;

/* CREATE VARIABLE FOR GROUPING BIC CODES AND SORTING LATER */
BICINDEX = 0;
IF SUBSTR(BIC,1,1) = 'A' THEN BICINDEX = 1;

/* RETAIN TEMPORARY VARIABLES FOR COMPARISONS OF 1ST TO LAST RECORD */
RETAIN CURRELIG ENT1 PIEDD BICC;

/* CREATE FLAGS TO SHOW WHEN DATES ARE NOT EQUAL ON ALL RECORDS */
ARRAY FLGS(*) DOECFLAG ENTFLAG PIEDFLAG BICAFLAG;

IF FIRST.SSN THEN DO;
  CURRELIG = DOEC;
  ENT1 = FIRSTENT;
  PIEDD = PIED1;
  BICC = SUBSTR(BIC,1,1);
  DO I = 1 TO DIM(FLGS);
    FLGS(I) = 0;
  END;
END;
/* CHECK ALL SUCCESSIVE RECORDS - COMPARISON FIELDS ARE RETAINED */
ELSE DO;
```

## Appendix A.64

JCL/SAS Code: PROC DUAL

```
IF CURRELIG NE DOEC THEN DOECFLAG = 1;
ELSE IF ENT1 NE FIRSTENT THEN ENTFLAG = 1;
ELSE IF PIEDD NE PIED1 THEN PIEDFLAG = 1;
ELSE IF BICC = 'A' OR SUBSTR(BIC,1,1) = 'A' THEN BICAFLAG = 1;
END;
/* OUTPUT THE RECORDS */
IF LAST.SSN THEN DO;

    OUTPUT FLAGDATA;

END; /* LAST SSN */

OUTPUT TEMP1.FINAL;

RUN;

/* MERGE THE FLAGS BACK ONTO THE DATA */

DATA TEMP3.DIFFELIG DIFFENT DIFFPIED DIFFA SAMEALL;
MERGE TEMP1.FINAL
(DROP=DOECFLAG ENTFLAG PIEDFLAG BICAFLAG) FLAGDATA;
BY SSN;

/* USE THE LOGICAL SEQUENCE NOTED ABOVE TO OUTPUT SEPARATE DATA SETS*/

IF DOECFLAG THEN OUTPUT TEMP3.DIFFELIG;
ELSE IF ENTFLAG THEN OUTPUT DIFFENT;
ELSE IF PIEDFLAG THEN OUTPUT DIFFPIED;
ELSE IF BICAFLAG THEN OUTPUT DIFFA;
ELSE OUTPUT SAMEALL;
RUN;

/*THIS MACRO SELECTS THE KEPT RECORD FROM EACH OF THE DATA SETS CREATED
ABOVE, SORTING THE DATA BY THE APPROPRIATE FLAGGED VARIABLE */

%MACRO CHOOSIT (FILE,SORTBY,SFILE,FFILE);
%IF &FILE=TEMP3.DIFFELIG %THEN %DO;
    %MACRO SORTD(BEGIN=,END=,RUN=);
        PROC SORT DATA=TEMP3.DIFFELIG(FIRSTOBS=&BEGIN. OBS=&END.)
            OUT=TEMP3&RUN..DIFFELIG;
        BY SSN &SORTBY;
        RUN;
    %MEND SORTD;
    %SORTD(BEGIN=1,END=300000,RUN=1);
    %SORTD(BEGIN=300001,END=600000,RUN=2);
```

## Appendix A.64

JCL/SAS Code: PROC DUAL

```
%SORTD(BEGIN=600001,END=MAX,RUN=3);

DATA TEMP4.DIFFELIG;
  SET TEMP31.DIFFELIG
      TEMP32.DIFFELIG
      TEMP33.DIFFELIG;
  BY SSN &SORTBY;
  RUN;
%END;
%ELSE %DO;
  PROC SORT DATA=&FILE OUT=&SFILE; BY SSN &SORTBY; RUN;
%END;

DATA &FFILE;
*KEEP THE SAME LENGTH FOR CONSTR. VARIABLES FOR COMBINING DATA SETS;
LENGTH
%DO MM=1 %TO &TOTMOS.;
  BPD&&YY&MM
%END;
$ 1;
SET &SFILE
  (RENAME=(
    %DO MM=1 %TO &TOTMOS.;
      BPD&&YY&MM=BPDA&&YY&MM
      DUED&&YY&MM=DUEDA&&YY&MM
      MBA&&YY&MM=MBAA&&YY&MM
      MBP&&YY&MM=MBPA&&YY&MM
    %END;))
  ;
  BY SSN &SORTBY;
  RETAIN DOECT DOEIT;
  IF FIRST.SSN THEN DO;
    DOECT = .;
    DOEIT = .;
    DOECT = DOEC;
    DOEIT = DOEI;
  END;
  ELSE IF NOT LAST.SSN THEN DO;
    DOECT = DOEC;
    DOEIT = DOEI;
  END;
  /* SUM UP THE PAYMENT VARIABLES ON MULTIPLE RECORDS WHERE BENEFIT PAID
    DESIGNATOR (BPD) = 1 */
  /* INPUT VARIABLES */
```

## Appendix A.64

JCL/SAS Code: PROC DUAL

```
ARRAY BPDA(*) $ %DO MM=1 %TO &TOTMOS; BPDA&&YY&MM %END;;
ARRAY DUEDA(*) %DO MM=1 %TO &TOTMOS; DUEDA&&YY&MM %END;;
ARRAY MBAA(*) %DO MM=1 %TO &TOTMOS; MBAA&&YY&MM %END;;
ARRAY MBPA(*) %DO MM=1 %TO &TOTMOS; MBPA&&YY&MM %END;;
/*TEMPORARY VARIABLES */
ARRAY MBAT(*) %DO MM=1 %TO &TOTMOS; MBAT&&YY&MM %END;;
ARRAY DUEDT(*) %DO MM=1 %TO &TOTMOS; DUEDT&&YY&MM %END;;
ARRAY MBPT(*) %DO MM=1 %TO &TOTMOS; MBPT&&YY&MM %END;;
ARRAY BPDT(*) $ %DO MM=1 %TO &TOTMOS; BPDT&&YY&MM %END;;
/*OUTPUT VARIABLES */
ARRAY BPD(*) $ %DO MM=1 %TO &TOTMOS; BPD&&YY&MM %END;;
ARRAY MBA(*) %DO MM=1 %TO &TOTMOS; MBA&&YY&MM %END;;
ARRAY DUED(*) %DO MM=1 %TO &TOTMOS; DUED&&YY&MM %END;;
ARRAY MBPO(*) %DO MM=1 %TO &TOTMOS; MBP&&YY&MM %END;;

/* PROCESS THE BENEFIT AMOUNTS-DETERMINE WHETHER TO ADD THE AMOUNT
ON BOTH OF THE DUAL RECORDS */

RETAIN %DO MM=1 %TO &TOTMOS;
DUEDT&&YY&MM MBAT&&YY&MM MBPT&&YY&MM BPDT&&YY&MM
%END;;

DO MM = 1 TO &TOTMOS;
IF FIRST.SSN THEN DO;
  MBAT(MM) = 0;
  DUEDT(MM) = 0;
  MBPT(MM) = 0;
  BPDT(MM) = '0';

/* THESE TEMPORARY VARS WILL BE COMPUTED ONLY IF BPD = 1 ON EITHER
OR BOTH RECORDS.
IF BPD IS NEVER = 1 ON ANY OF THE RECORDS WE
STILL WANT TO RETAIN THE VALUES ON THE LAST RECORD ACCORDING TO
THE SORT - AMOUNT VALUES ARE ON THE RECORDS EVEN IF PAYMENTS
ARE IN SUSPENSION - SEE PROCESSING FOR LAST.SSN */

IF BPDA(MM) = '1' THEN DO;
  BPDT(MM) = '1';
  MBAT(MM) = MBAA(MM);
  DUEDT(MM) = DUEDA(MM);
  MBPT(MM) = MBPA(MM);
END;
END;
ELSE IF NOT FIRST.SSN THEN DO; /* SUBSEQUENT RECORDS */
/* THIS WILL SUM WITH WHAT IS ON THE PREVIOUS RECORD */
```

## Appendix A.64

JCL/SAS Code: PROC DUAL

```
IF BPDA(MM) = '1' THEN DO;
  BPDT(MM) = '1';
  MBAT(MM) = SUM(MBAT(MM),MBAA(MM));
  DUEDT(MM) = SUM(DUEDT(MM),DUEDA(MM));
  MBPT(MM) = SUM(MBPT(MM),MBPA(MM));
END;
END;

END; /* WHOLE ARRAY */

/* ARRAYS FOR THE DISABILITY FIELDS - WE CHOOSE THE LAST POPULATED */
ARRAY DDO (*) DDO1-DDO12;
ARRAY DAC (*) $ DAC1-DAC12;
ARRAY LOD (*) $ LOD1-LOD12;
ARRAY ENT (*) ENTDAT1-ENTDAT12;
ARRAY DDBC (*) DDBC1-DDBC12;
ARRAY DSD (*) DSD1-DSD12;
ARRAY HDD (*) HDD1-HDD12;
ARRAY SDS (*) SDS1-SDS12;
ARRAY ADC (*) ADC1-ADC12;
ARRAY APS (*) APS1-APS12;
ARRAY EBD (*) EBD1-EBD12;
ARRAY BDC (*) $ BDC1-BDC12;
ARRAY CDR (*) $ CDR1-CDR12;
ARRAY CSA (*) $ CSA1-CSA12;
ARRAY DXPRI_MBR (*) $ DXPRI_MBR1-DXPRI_MBR12;
ARRAY DXSEC_MBR (*) $ DXSEC_MBR1-DXSEC_MBR12;

/* THE FOLLOWING MACROS CREATE THE NEW VARIABLE WE ARE GOING TO
KEEP FROM THE (PREVIOUS) SUPPLEMENTARY RECORD ON THE KEPT RECORD.
THE DATA IS SORTED BY THE VARIABLE WHOSE LATEST VALUE WE
HAVE CHOSEN AS FLAGGING THE RECORD WE WANT TO RETAIN.
SINCE WE RETAIN THE NEW VARIABLE IN PROCESSING ALL THE RECORDS
EXCEPT THE LAST ONE, THE PREVIOUS RETAINED VALUE WILL BE KEPT
ON THE LAST (KEPT) RECORD. EACH DATA SET OUTPUT USING THE FLAGS
ABOVE IS PROCESSED SEPARATELY */

%MACRO LASTIT (ARR,VAR); /* FOR NUMERIC VARIABLES */

/* INITIALIZE THE VARIABLES BECAUSE THEY ARE ALL MISSING ON MANY
RECORDS */
RETAIN &VAR;
IF FIRST.SSN THEN &VAR = .;
IF NOT LAST.SSN THEN DO;
  DO I = 1 TO DIM(&ARR);
```

## Appendix A.64

JCL/SAS Code: PROC DUAL

```
/* WE HAVE TO LOOP THRU ALL VARS BECAUSE EARLIER VAR MAY BE MISSING
   AND LATER VAR POPULATED */
IF &ARR(I) GT .Z THEN &VAR = &ARR(I);
END;
END;

%MEND LASTIT;
%LASTIT(DDO,DDODT);
%LASTIT(ENT,ENTDATDT);
%LASTIT(DDBC,DDBC DT);
%LASTIT(DSD,DSDDT);
%LASTIT(HDD,HDDDT);
%LASTIT(SDS,SDSDT);
%LASTIT(ADC,ADCDT);
%LASTIT(APS,APSDT);
%LASTIT(EBD,EBDDT);

%MACRO LASTIT1(ARR,VAR); /* FOR CHARACTER VARIABLES */
LENGTH BDCDT $2 DXPRI_MBRDT DXSEC_MBRDT $4;

/* INITIALIZE THE VARIABLES BECAUSE THEY ARE ALL MISSING ON MANY
   RECORDS */
RETAIN &VAR;
IF FIRST.SSN THEN &VAR = ' ';
IF NOT LAST.SSN THEN DO;
DO I = 1 TO DIM(&ARR);
IF &ARR(I) NE ' ' THEN &VAR = &ARR(I);
END;
END;

%MEND LASTIT1;
%LASTIT1(DAC,DACDT);
%LASTIT1(LOD,LODDT);
%LASTIT1(BDC,BDCDT);
%LASTIT1(CDR,CDRDT);
%LASTIT1(CSA,CSADT);
%LASTIT1(DXPRI_MBR,DXPRI_MBRDT);
%LASTIT1(DXSEC_MBR,DXSEC_MBRDT);

/* NOW PROCESS THE CHOSEN RECORD */
IF LAST.SSN THEN DO;
/* TAKE THE OUTCOMES OF THE AMOUNT VARS AND BPD.
   IF THE AMOUNTS HAVE BEEN SUMMED INTO THE TEMP VARS THEN TAKE
   THE SUMMED VALUE AS THE FINAL AMOUNT. ELSE TAKE THE ORIGINAL
   AMOUNT ON THE RECORD WHICH BECOMES THE LAST RECORD IN THE SORT */
```

## Appendix A.64

JCL/SAS Code: PROC DUAL

```
DO MM = 1 TO &TOTMOS;
  BPD(MM) = BPDT(MM);
  IF BPDT(MM) = '' THEN BPD(MM) = BPDA(MM);
  IF MBAT(MM) GT 0 THEN MBA(MM) = MBAT(MM);
  ELSE MBA(MM) = MBAA(MM);
  IF DUEDT(MM) GT 0 THEN DUED(MM) = DUEDT(MM);
  ELSE DUED(MM) = DUEDA(MM);
  IF MBPT(MM) GT 0 THEN MBPO(MM) = MBPT(MM);
  ELSE MBPO(MM) = MBPA(MM);
END;

/* TAKE RETAINED PREVIOUS VALUE FOR SUPPLEMENTARY VARIABLES */
LENGTH BDCD $2 DXPRI_DUAL DXSEC_DUAL $4;
DOECD = DOECT;
DOEID = DOEIT;
DDOD = DDODT;
DACD = DACDT;
LODD = LODDT;
ENTDATD = ENTDATDT;
DDBCD = DDBCDT;
DSDD = DSDDT;
HDDD = HDDDT;
SDSD = SDSDT;
ADCD = ADCDT;
APSD = APSDT;
EBDD = EBDDT;
BDCD = BDCDT;
CDRD = CDRDT;
CSAD = CSADT;
DXPRI_DUAL = DXPRI_MBRDT;
DXSEC_DUAL = DXSEC_MBRDT;

DROP DOECFLAG ENTFLAG PIEDFLAG BICAFLAG I
  MBAA: DUEDA: MBPA: BPDA:
  MBAT: DUEDT: MBPT: BPDT:
  PIEDD
  CURRELIG
  ENT1 FIRSTENT
  DOECT
  DOEIT
  DDODT
  DACDT
  LODDT
  ENTDATDT
```

Appendix A.64

JCL/SAS Code: PROC DUAL

```

DDBC DT
DSDDT
HDDDT
SDSDT
ADCDT
APSDT
EBDDT
BDCDT
CDRDT
CSADT
DXPRI_MBRDT
DXSEC_MBRDT
BICINDEX;

OUTPUT;

END;

RUN;

TITLE "AFTER PROCESSING - &FFILE RECORDS";
PROC PRINT DATA=&FFILE (OBS=20);
  VAR SSN BIC DOEC: DOEI: ENTDAT1-ENTDAT3
  ENTDATD DDO1-DDO3 DDOD DAC1-DAC3 DACD LOD1-LOD3 LODD DUED: BPD;;
RUN;

%MEND CHOOSIT;

%CHOOSIT(TEMP3.DIFFELIG,DOEC, TEMP4.DIFFELIG,TEMP5.DIFFELIG);
%CHOOSIT(DIFFENT, FIRSTENT,TEMP6.DIFFENT, TEMP7.DIFFENT);
%CHOOSIT(DIFFPIED, PIED1, TEMP8.DIFFPIED,TEMP9.DIFFPIED);
%CHOOSIT(DIFFA, BICINDEX, TEMP10.DIFFA, TEMP11.DIFFA);
%CHOOSIT(SAMEALL, BIC, TEMP12.SAMEALL,TEMP13.SAMEALL);

/* COMBINE ALL FILES FOR FINAL DATASET */
DATA OUT.DUALS;
  SET SINGLE TEMP5.DIFFELIG TEMP7.DIFFENT
  TEMP9.DIFFPIED TEMP11.DIFFA TEMP13.SAMEALL;
  BY SSN;
RUN;
TITLE 'FINAL SELECTED RECORDS';
PROC CONTENTS DATA=OUT.DUALS; RUN;
PROC PRINT DATA=OUT.DUALS (OBS=20); RUN;
%MEND;
%START;

```



Appendix A.64

JCL/SAS Code: PROC DUAL

```
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>  
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>  
DATA  
FROM: <ABRYCE@MATHEMATICA-MPR.COM>  
TO: <ABRYCE@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 PROC DUAL IS COMPLETE
```

## Appendix A.65

JCL/SAS Code: MRGDUALS

```

//#6502MRG JOB (11710000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   50214 DAF
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.PRDLIB(MRGDUALS)
//* PURPOSE:   PROCESS DUAL ELIGIBLE RECORDS
//*           ALL MBR FIELDS LOADED
//* PROGRAMMER: ALEX BRYCE 4/10/2021 FOR DAF21
//* E-MAIL:    ABRYCE@MATHEMATICA-MPR.COM
//*           ALEX.BRYCE@SSA.GOV
//*****
//*
//JS010 EXEC SAS9,
//      WORK='200000,100000'          * SPACE IN BLKS *
//*
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.SING.SA.V1,DISP=SHR
//IN2  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PROCDUAL.SA.V1,DISP=SHR
//IN3  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PHUSSING.SA.V1,DISP=SHR
//IN4  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PHUSDUAL.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//TEMP2 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.COMB.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES OBS = MAX;
DATA TEMP1.MBR;
MERGE IN1.MBR (IN=S) IN2.DUALS (IN=D DROP=BICC);
BY SSN;
/* BUILD DUAL ELIGIBLE FLAG */
IF D THEN DUALELIG = 1;ELSE DUALELIG = 0;
RUN;

DATA TEMP2.PHUS;
MERGE IN3.LINKPHUS(IN=S) IN4.PHUSDUAL (IN=D);
BY SSN;
/* BUILD DUAL ELIGIBLE FLAG */
IF D THEN DUALELIG = 1;ELSE DUALELIG = 0;
RUN;

```

Appendix A.65

JCL/SAS Code: MRGDUALS

```
DATA OUT.MBR;
MERGE TEMP1.MBR(IN=M) TEMP2.PHUS;
  BY SSN;
  IF M;
RUN;

TITLE 'FINAL MERGED MBR DATA WITH DUALS AND PHUS';
PROC FREQ DATA=OUT.MBR; TABLES DUALELIG; RUN;
PROC CONTENTS DATA=OUT.MBR; RUN;
PROC PRINT DATA=OUT.MBR (OBS=20); RUN;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 MRGDUALS IS COMPLETE
```

## Appendix A.66

JCL/SAS Code: PROCMBR

```
//#6502PRM JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   50214 DAF
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.PRDLIB(PROCMBR)
//* PURPOSE:   TO CREATE MONTHLY VARS FROM MBR DATA FOR TTW
//*           ADD NEW VARIABLES TO OUTPUT FILE
//* WRITTEN BY: MIRIAM LOEWENBERG
//* UPDATED:   XIAO BARRY
//* UPDATED:   4/11/2022 ALEX BRYCE FOR DAF21
//* E-MAIL:    ABRYCE@MATHEMATICA-MPR.COM
//*           ALEX.BRYCE@SSA.GOV
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//IN1   DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.COMB.SA.V1,DISP=SHR
//IN2   DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.SUMREC.SA.V1,DISP=SHR
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PROCMBR.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//*
//SYSIN DD *
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES OBS = MAX;

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */

%MACRO START;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

DATA TEMP.MBR;
SET IN1.MBR;
  BY SSN;

/* SET UP ARRAYS FOR CREATING VARIABLES */
/* CREATE ARRAYS FOR MONTHLY VARIABLES */
/* NEW VARIABLES */
ARRAY DX1{*} $4 %DO MM=1 %TO &TOTMOS; DXPRI_DIMBR&&YY&MM %END;;
```

## Appendix A.66

JCL/SAS Code: PROCMBR

```

ARRAY DX2{*} $4 %DO MM=1 %TO &TOTMOS; DXSEC_DIMBR&&YY&MM %END;;
ARRAY CDRN {*} $1 %DO MM=1 %TO &TOTMOS; CDR&&YY&MM %END;;
ARRAY CYEAR (*) $ %DO MM=1 %TO &TOTMOS; C&&YY&MM %END;;
ARRAY NYEAR (*) %DO MM=1 %TO &TOTMOS; N&&YY&MM %END;;

/* OLD VARIABLES */
ARRAY DIAG1{*} $ %DO I=1 %TO 12; DXPRI_MBR&I %END;;
ARRAY DIAG2{*} $ %DO I=1 %TO 12; DXSEC_MBR&I %END;;
ARRAY DDO (*) %DO I = 1 %TO 12; DDO&I %END;; /* ONSET DATES */
ARRAY ENTDAT (*) %DO I = 1 %TO 12; ENTDAT&I %END;;
        /* DISABILITY ENTITLE DATE */
ARRAY DDBC (*) %DO I = 1 %TO 12; DDBC&I %END;;
        /* DISABILITY CESS DATE */
ARRAY APS (*) %DO I = 1 %TO 12; APS&I %END;; /* APPEAL DATES */
ARRAY BDC (*) $ %DO I = 1 %TO 12; BDC&I %END;; /* DENIAL CODES */
ARRAY CDR (*) $ %DO I = 1 %TO 12; CDR&I %END;; /* DIS CESS REASON */

/* INITIALIZE MONTHLY VARS TO MISSING */
DO I = 1 TO &TOTMOS;
    DX1(I) = ' ';
    DX2(I) = ' ';
    CDRN(I) = ' ';
END;

/* LOOP THROUGH THE NUMBER (NODF) OF DISABILITY FIELDS */
DO K = 1 TO NODF;

/* THE FOLLOWING ROUTINE CREATES MONTHLY REFERENCE DATES FROM
   JAN 1994 TO CURRENT YEAR */
J=1;
Y=&BEGYR;

/* IF THERE IS A TERMINATION DATE IN LATEST ENTRY - END MONTHLIES */
DO I = 1 TO DIM(CYEAR); /* LOOP THROUGH ALL MONTH/YEAR REFERENCES */

CYEAR(I) = PUT(Y,4.) || (PUT(J,Z2.));
IF INT(I/12)=I/12 THEN DO;
    J=1;
    Y=Y+1;
END;
ELSE J=J+1;

NYEAR(I)=INPUT(SUBSTR(CYEAR(I),1,4) || SUBSTR(CYEAR(I),5,2),Y YMMN6.);

/* REVISION - IF THERE IS A DISABILITY CESSATION DATE THEN USE IT

```

## Appendix A.66

JCL/SAS Code: PROCMBR

```
    TO LIMIT THE RANGE OF THE VARIABLES.
    ELSE ALLOW THE RANGE TO EXTEND THE ENTIRE PERIOD OF DISABILITY */

IF ENTDAT(K) > .Z THEN DO;
  IF ENTDAT(K) LE NYEAR(I) AND DDBC(K) GT NYEAR(I) THEN DO;
    DX1(I) = DIAG1(K);
    DX2(I) = DIAG2(K);
    CDRN(I) = CDR(K);
  END;
  ELSE IF DDBC(K) = . AND ENTDAT(K) LE NYEAR(I) THEN DO;
    DX1(I) = DIAG1(K);
    DX2(I) = DIAG2(K);
    CDRN(I) = CDR(K);
  END;

END; /* DATE FALLS WITHIN REFERENCE MONTH */
END; /* MONTH/YEAR REFERENCE VARIABLES */
END; /* DISABILITY FIELDS (DO K= 1 TO NODF) */

DROP I J K Y
%DO MM = 1 %TO &TOTMOS; C&&YY&MM N&&YY&MM %END;;

RUN;

DATA OUT.MBR;
MERGE TEMP.MBR(IN=M) IN2.MBR
    ;
BY SSN ;
IF M;

*SET TO MISSING INSURED DATA FOR DUAL ELIGIBLE RECORDS;
ARRAY INSV (*) $ DIBREQ1-DIBREQ15 DIBHAS1-DIBHAS15;

IF DUALELIG = 1 THEN DO;
  NINSD = . ;
  DO I = 1 TO DIM(INSV);
    INSV(I) = ' ';
  END;
END;

*BUILD THE SSDI COMMON VARIABLE FOR DAF;
ARRAY LAFYMM(&BEGYR.:&ENDYR.,1:12) %DO YEAR=&BEGYR. %TO &ENDYR.;
    %LET YR=%SUBSTR(&YEAR.,3,2);
    LAF&YR.01-LAF&YR.12
    %END;;
```

## Appendix A.66

JCL/SAS Code: PROCMBR

```
ARRAY SSDIYY (&BEGYR.:&ENDYR.) %DO YEAR=&BEGYR. %TO &ENDYR.;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      SSDI&YR.
      %END;;
DO YR=&BEGYR. TO &ENDYR.;
  SSDIYY(YR) = 0; *SET INITIAL SSDIYY VALUE TO BE 0;
  DO MO=1 TO 12 UNTIL(SSDIYY(YR)=1);
    IF LAFYMM(YR,MO) IN ("C","E") THEN SSDIYY(YR)=1;
  END;
END;

SSDI_ONSET=MIN(OF DDO1-DDO12,DDOD);
SSDI_ONSET_LAST =MAX(OF DDO1-DDO12,DDOD);
RUN;
```

```
PROC CONTENTS; RUN;
%MEND;
%START;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 PROCMBR IS COMPLETE
```

## Appendix A.67

JCL/SAS Code: RSTRCTN

```

//#6502RST JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT: 50214 DAF
//* DIRECTOR: MATT URATO
//* PROGRAM: OPDR.TG.PRD.ETTW.#6502.DAF21.MBR.PRDLIB(RSTRCTN)
//* PURPOSE: RESTRICT NUMBER OF N VARIABLES TO 50 FOR PIA,PIED,
//*      IME,PIARFC VARIABLES IN MBR DATA
//* CONTACT: XIAO BARRY
//* UPDATED: 4/11/2022 SERGE LUKASHANETS FOR DAF20
//* E-MAIL: ABRYCE@MATHEMATICA-MPR.COM
//*      ALEX.BRYCE@SSA.GOV
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//TEMP DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//IN1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PROCMBR.SA.V1,
//      DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBRREST.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//*

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS=MAX MPRINT;

%LET MAXT=150; /*NUMBER OF OCCURRENCES FROM MBR DATA*/
%LET OLDREST = 50; /*NUMBER OF OCCURENCES FROM PREEVIOUS DAF */

*DETERMINE MAX NUM OF ADJUDICATIONS FOR CURRENT DAF;
PROC MEANS DATA=IN1.MBR P99;
  VAR NPIA;
  OUTPUT OUT=STATIS (DROP = _TYPE_ _FREQ_)
  P99(NPIA)=P99NPIA;
RUN;

%MACRO START;
* CREATE A MACRO VARIABLE TO INDEX MAX OF RESTRICT N IN ALL FILE*;
*THE RESTRICT NUMBER OF N VAIABLES IS MAX OF 99TH PERCENTILE
AND THE NUMBER FROM LAST YEAR*;

DATA _NULL_;

```



## Appendix A.67

JCL/SAS Code: RSTRCTN

```
SET STATIS;
  CALL SYMPUT('REST',MAX(ROUND(P99NPPIA),&OLDREST.));
STOP;
RUN;

*** PRINT RECORDS PRIOR TO PROCESSING IN ORDER TO CHECK THAT ***;
*** OUTPUT IS WHAT WE EXPECT ***;

PROC PRINT DATA=IN1.MBR(OBS=25);
  VAR SSN NPPIA PIA: PIED: IME: ;
  WHERE NPPIA GT &REST.;
  TITLE "CHECK VARIABLES PRIOR TO PROCESSING";
  TITLE2 "RECORDS WITH MORE THAN &REST. OCCURRENCES";
RUN;

PROC PRINT DATA=IN1.MBR(OBS=25);
  VAR SSN NPPIA PIA: PIED: IME: ;
  TITLE2 " ";
RUN;

DATA OUT.MBR(DROP=I J NPPIA_START);
  SET IN1.MBR;

  LENGTH
  %DO I = 1 %TO &REST;
  HOLDPIARFC&I %END; $1;

  *** STEP 1 IS TO CREATE AN ADDITIONAL VARIABLE THAT IS ***;
  *** SET TO THE EARLIEST OCCURRENCE OF EACH N VARIABLE ***;

  *INPUT ARRAY;
  ARRAY IME(*) %DO I=1 %TO &MAXT; IME&I %END;;
  ARRAY PIED(*) %DO I=1 %TO &MAXT; PIED&I %END;;
  ARRAY PIA(*) %DO I=1 %TO &MAXT; PIA&I %END;;
  ARRAY PIARFC(*) $ %DO I=1 %TO &MAXT; PIARFC&I %END;;

  *OUTPUT ARRAY;
  ARRAY HOLDIME(*) %DO I=1 %TO &REST; HOLDIME&I %END;;
  ARRAY HOLDPIED(*) %DO I=1 %TO &REST; HOLDPIED&I %END;;
  ARRAY HOLDPIA(*) %DO I=1 %TO &REST; HOLDPIA&I %END;;
  ARRAY HOLDPIARFC(*) $ %DO I=1 %TO &REST; HOLDPIARFC&I %END;;

  %MACRO HOLDVAR(AR=);

  *** STEP 1 IS TO CREATE AN ADDITIONAL VARIABLE THAT IS ***;
```

## Appendix A.67

JCL/SAS Code: RSTRCTN

```
*** SET TO THE EARLIEST OCCURRENCE OF EACH N VARIABLE ***;
FIRST&AR.=&AR.1;

*** IF NPI <= REST NUM, SET FIRST N OCCURRENCES TO THE ***;
*** HOLD VARIABLES ***;
IF NPIA <= &REST THEN DO I=1 TO &REST;
  HOLD&AR.(I)=&AR.(I);
END;

*** IF NPIA > REST NUM THEN SET THE LAST N OCCURRENCES TO THE ***;
*** HOLD VARIABLES ***;

IF NPIA > &REST THEN DO;
  NPIA_START=NPIA-(&REST-1);

  DO I = NPIA_START TO NPIA;
    J = I-(NPIA_START-1);
    HOLD&AR(J) = &AR.(I);
  END;
END;

DROP
%DO I = 1 %TO &MAXT;
  &&AR.&I
%END;;

RENAME
%DO I = 1 %TO &REST;
  HOLD&&AR.&I=&&AR.&I
%END;;

%MEND HOLDVAR;
%HOLDVAR(AR=IME);
%HOLDVAR(AR=PIED);
%HOLDVAR(AR=PIA);
%HOLDVAR(AR=PIARFC);

NPIA_ORIG = NPIA;
NPIA_NEW = NPIA;
IF NPIA GT &REST. THEN NPIA_NEW = &REST.;
DROP NPIA;

RUN;

*** PRINT RECORDS AFTER FINAL PROCESSING IN ORDER TO CHECK THAT ***;
```

Appendix A.67

JCL/SAS Code: RSTRCTN

```
*** OUTPUT IS WHAT WE EXPECT ***;

PROC PRINT DATA=OUT.MBR(OBS=25);
  VAR SSN NPIA: PIA: PIED: IME: FIRST;;
  WHERE NPIA_ORIG > &REST;
  TITLE "CHECK VARIABLES AFTER FINAL PROCESSING";
  TITLE2 "RECORDS WITH MORE THAN 50 N OCCURRENCES";
RUN;

PROC PRINT DATA=OUT.MBR(OBS=25);
  VAR SSN NPIA: PIA: PIED: IME: FIRST;;
  TITLE "CHECK VARIABLES AFTER FINAL PROCESSING";
  TITLE2 " ";
RUN;
%MEND START;
%START;

PROC CONTENTS DATA=OUT.MBR;
  TITLE "CONTENTS OF RSTRCTN";
RUN;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 RSTRCTN IS COMPLETE
```

## Appendix A.68

JCL/SAS Code: JCLYRSP

```
//#6502MSP JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//*
//*  *-----*
//*  ! THE MACRO SPLIT PROGRAM          !
//*  *-----*
//*
//* CREATED BY: SERGE LUKASHANETS, 5/26/2016
//* MODIFIED BY: MIAOMIAO SHEN 6/23/2017
//* UPDATED: ALEX BRYCE 4/11/2021 FOR DAF21
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*****
// SET REG='64M'
//*
//JS010 EXEC SAS9,
//  WORK='200000,100000',
//  REGION=&REG,
//  PARM='MEMSIZE=&REG'
//*
//***DATA FOR SPLIT
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBRREST.SA.V1,DISP=SHR
//***OUT YEARLY FILES
//OUT94 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1994.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT95 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1995.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT96 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1996.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT97 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1997.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT98 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1998.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT99 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1999.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT00 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2000.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
```

## Appendix A.68

JCL/SAS Code: JCLYRSP

```
//OUT01 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2001.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT02 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2002.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT03 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2003.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT04 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2004.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT05 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2005.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT06 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2006.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT07 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2007.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT08 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2008.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT09 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2009.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2010.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT11 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2011.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2012.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT13 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2013.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT14 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2014.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
//OUT15 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2015.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,,10)
```

## Appendix A.68

JCL/SAS Code: JCLYRSP

```
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2016.SA.V1,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO,VOL=(,,10)
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2017.SA.V1,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO,VOL=(,,10)
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2018.SA.V1,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO,VOL=(,,10)
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2019.SA.V1,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO,VOL=(,,10)
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2020.SA.V1,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO,VOL=(,,10)
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2021.SA.V1,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT OBS=MAX;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(MSPLIT)";

%OUTP(IN1.MBR,OUT,MBR,1994,2021,
      SSN HI_START: HI_TERM: SMI_STAR: SMI_TERM: SSDI:);

PROC CONTENTS DATA=OUT94.MBR;RUN;
PROC CONTENTS DATA=OUT95.MBR;RUN;
PROC CONTENTS DATA=OUT96.MBR;RUN;
PROC CONTENTS DATA=OUT97.MBR;RUN;
PROC CONTENTS DATA=OUT98.MBR;RUN;
PROC CONTENTS DATA=OUT99.MBR;RUN;
PROC CONTENTS DATA=OUT00.MBR;RUN;
PROC CONTENTS DATA=OUT01.MBR;RUN;
PROC CONTENTS DATA=OUT02.MBR;RUN;
PROC CONTENTS DATA=OUT03.MBR;RUN;
PROC CONTENTS DATA=OUT04.MBR;RUN;
PROC CONTENTS DATA=OUT05.MBR;RUN;
PROC CONTENTS DATA=OUT06.MBR;RUN;
PROC CONTENTS DATA=OUT07.MBR;RUN;
PROC CONTENTS DATA=OUT08.MBR;RUN;
PROC CONTENTS DATA=OUT09.MBR;RUN;
PROC CONTENTS DATA=OUT10.MBR;RUN;
PROC CONTENTS DATA=OUT11.MBR;RUN;
PROC CONTENTS DATA=OUT12.MBR;RUN;
```

Appendix A.68

JCL/SAS Code: JCLYRSP

```
PROC CONTENTS DATA=OUT13.MBR;RUN;
PROC CONTENTS DATA=OUT14.MBR;RUN;
PROC CONTENTS DATA=OUT15.MBR;RUN;
PROC CONTENTS DATA=OUT16.MBR;RUN;
PROC CONTENTS DATA=OUT17.MBR;RUN;
PROC CONTENTS DATA=OUT18.MBR;RUN;
PROC CONTENTS DATA=OUT19.MBR;RUN;
PROC CONTENTS DATA=OUT20.MBR;RUN;
PROC CONTENTS DATA=OUT21.MBR;RUN;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM
TO: <ABRYCE@MATHEMATICA-MPR.COM
SUBJECT: DAF21 JCLYRSP IS COMPLETE
```

## Appendix A.69

JCL/SAS Code: LAFMR

```

//#6502LAF JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:  50214 DAF
//* DIRECTOR:
//* PROGRAM:  OPDR.TG.PRD.ETTW.#6501.DAF21.MBR.PRDLIB(LAFMR)
//* PURPOSE:  CREATE THE MOST RECENT VALUE OF LAF
//* CONTACT:  ALEX BRYCE 4/20/2022 FOR DAF21
//* MPR PHONE:
//* E-MAIL:   ABRYCE@MATHEMATICA-MPR.COM
//*          ALEX.BRYCE@SSA.GOV
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000'
//*
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBRREST.SA.V1,DISP=SHR
//TEMP DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.LAFMR.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//SYSIN DD *

OPTIONS NOCENTER LS=142 PS=60 COMPRESS=BINARY OBS=MAX;
/* UPDATE CURRENT DAF YEAR AND PREVIOUS DAF YEAR */
%LET CURRYR=21; /* CHANGE AS NEEDED */
%LET PREVYR=20; /* CHANGE AS NEEDED */

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */

%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

DATA TEMP.LAFMR;
SET IN1.MBR (KEEP = SSN %DO MM=1 %TO &TOTMOS; LAF&&YY&MM %END;);

ARRAY LAF (*) $ %DO MM=1 %TO &TOTMOS; LAF&&YY&MM %END;;

LENGTH LAF_MR $2.;

LAF_MR = "";

```



## Appendix A.69

JCL/SAS Code: LAFMR

```
DO I = DIM(LAF) TO 1 BY -1 WHILE (LAF_MR="");
  IF LAF(I) NE "" THEN DO;
    LAF_MR = LAF(I);
    _MN=SUBSTR(VNAME(LAF(I)),6,2);
    _YR=SUBSTR(VNAME(LAF(I)),4,2);
    LAF_MR_DATE = MDY(_MN,1,_YR);
  END;
END;
RUN;

PROC PRINT DATA=TEMP.LAFMR (OBS=100);
  WHERE _YR = "&CURRYR.";
  VAR LAF&CURRYR.: LAF&PREVYR.: LAF_;
RUN;
PROC PRINT DATA=TEMP.LAFMR (OBS=100); RUN;
PROC FREQ DATA=TEMP.LAFMR;
  TABLES LAF_MR / LIST MISSING;
RUN;

DATA OUT1.LAFMR;
SET TEMP.LAFMR (KEEP=SSN LAF_);
LABEL
LAF_MR="LEDGER ACCOUNT FILE STATUS, MOST RECENT"
LAF_MR_DATE="DATE OF MOST RECENT LEDGER ACCOUNT FILE STATUS"
;
RUN;

PROC CONTENTS DATA=OUT1.LAFMR;
RUN;

%MEND;
%START;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
```

Appendix A.69

JCL/SAS Code: LAFMR

```
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 LAFMR IS COMPLETE
```

## Appendix A.70

JCL/SAS Code: DCFDMG

```
//#6502DCF JOB (12510000,T715,,SAS,,ITC9TH),BRYCE,CLASS=T,
//  MSGCLASS=1,MSGLEVEL=(1,1),REGION=4096K,NOTIFY=#6502
/*SSAMAIN SASBASIC
//*****
/* PROJECT:   BUILD DAF21 (50214)
/* DIRECTOR:  MATT URATO
/* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DEMO.PRDLIB(DCFDMG)
/* DESCRIPTION: CREATES EXTRACT OF DCF CLAIM FILE
/*           TO OBTAIN THE TWP COMPLETION MONTHS FOR DEMO
/* CONTACT:   ALEX BRYCE
/* UPDATED:   ALEX BRYCE 4/22/2022 FOR DAF21
/* EMAIL:     ABRYCE@MATHEMATICA-MPR.COM
/*           ALEX.BRYCE@SSA.GOV
//*****
//S1 EXEC SAS9,LOAD='DBP8.DB2.SDSNLOAD',SORT='15000',
//  WORK='15000,15000',COND=EVEN
//CLM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFCLM.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=TSILO,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HCLM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFHCLM.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=AFF=CLM,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFMED.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=AFF=CLM,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HMED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFHMED.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=AFF=CLM,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//SYSIN DD *

OPTIONS NOCENTER;

LIBNAME CLM TAPE;
LIBNAME HCLM TAPE;
LIBNAME MDCF DB2 SSID=DBP8 SCHEMA=MDCF ;
```

Appendix A.70

JCL/SAS Code: DCFDMG

```
* CREATE SAS EXTRACT OF CLAIM GROUP DATA;
PROC SQL;
CREATE TABLE CLM.CLM AS
SELECT * FROM MDCF.CLM;
QUIT;
RUN;

* CREATE SAS EXTRACT OF HISTORY CLAIM GROUP DATA;
PROC SQL;
CREATE TABLE HCLM.HCLM AS
SELECT * FROM MDCF.HCLM;
QUIT;
RUN;

PROC CONTENTS DATA=CLM.CLM ;
RUN;
PROC CONTENTS DATA=HCLM.HCLM ;
RUN;

LIBNAME MED TAPE;
LIBNAME HMED TAPE;
LIBNAME MDCF DB2 SSID=DBP8 SCHEMA=MDCF ;

* CREATE SAS EXTRACT OF CLAIM MEDICAL DATA;
PROC SQL;
CREATE TABLE MED.MED AS
SELECT * FROM MDCF.MEDICAL;
QUIT;
RUN;

* CREATE SAS EXTRACT OF HISTORY MEDICAL GROUP DATA;
PROC SQL;
CREATE TABLE HMED.HMED AS
SELECT * FROM MDCF.HMEDICAL;
QUIT;
RUN;

PROC CONTENTS DATA=MED.MED;
RUN;
PROC CONTENTS DATA=HMED.HMED;
RUN;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
```

Appendix A.70

JCL/SAS Code: DCFDMG

```
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 DCFDMG IS COMPLETE
```

## Appendix A.71

JCL/SAS Code: TWPCMPL

```
//#6502TWP JOB (12510000,T715,,SAS,,ITC9FL),BRYCE-ALEX,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   BUILD DAF21 (50214)
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DEMO.PRDLIB(TWPCMPL)
//* DESCRIPTION: GET TWP_CMPL_MNTH VARIABLES, BY SSN, ORDERED FROM
//*             OLDEST TO NEWEST
//*             (OBTAIN TRIAL WORK PERIOD COMPLETION MONTH
//*             INFORMATION FROM THE DCF CLAIM TABLE)
//* CONTACT:   ALEX BRYCE
//*             ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//SAS EXEC SAS9,
//  WORK='120000,60000'
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFCLM.SA.V1,DISP=SHR
//FIND DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.MASTFIND.SA.V1,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.TWPCMPL.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *
```

```
OPTIONS NOCENTER OBS=MAX;
```

```
* MERGE DCF CLAIM FILE WITH DEMO SSN LINK FILE TO ELIMINATE SSNS
  THAT ARE NOT IN THE DAF;
```

```
PROC SORT DATA=IN1.CLM(KEEP=COSSN TWP_CMPL_MDT
  WHERE=(TWP_CMPL_MDT NE .))
```

```
  OUT=TEMP1.CLM;
  BY COSSN;
RUN;
```

```
DATA TEMP1.TRFCCLAIM;
  MERGE TEMP1.CLM(IN=CLAIM RENAME=(COSSN=SSN))
  FIND.MASTFIND(IN=FIND KEEP=SSN)
```

```
  ;
  BY SSN;
  IF CLAIM AND FIND;
RUN;
```

```
PROC SORT DATA=TEMP1.TRFCCLAIM;
```

## Appendix A.71

JCL/SAS Code: TWPCMPL

```
    BY SSN TWP_CMPL_MDT;
RUN;
```

```
DATA TEMP1.TRFLCLAIM;
  SET TEMP1.TRFLCLAIM;
  BY SSN TWP_CMPL_MDT;
  IF FIRST.TWP_CMPL_MDT THEN OUTPUT;
RUN;
```

```
PROC TRANSPOSE DATA=TEMP1.TRFLCLAIM OUT=OUT.TWPCMPL PREFIX=TWPCMPLMNTH;
  BY SSN;
  VAR TWP_CMPL_MDT;
RUN;
```

```
PROC CONTENTS DATA=OUT.TWPCMPL;
RUN;
```

```
PROC FORMAT;
VALUE MISSFMT .='MISSING' OTHER='NOT MISSING';
RUN;
```

```
TITLE "CHECK FREQUENCY OF TWPCMPLMNTH FOR MISSING OR NOT MISSING";
PROC FREQ DATA=OUT.TWPCMPL;
FORMAT _NUMERIC_ MISSFMT.;
TABLES TWPCMPLMNTH:/MISSING;
RUN;
TITLE;
```

```
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 TWPCMPL IS COMPLETE
```

## Appendix A.72

JCL/SAS Code: DCFMED

```
//#6502MED JOB (12510000,T715,,SAS,,ITC9FL),ALEX-BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   DAF21 CONSTRUCTION
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DEMO.PRDLIB(DCFMED)
//* DESCRIPTION: GET BLND_ONST_DT VARIABLES ORDERED BY SSN AND
//*             GET MIE RELATED DATA AVAILABLE ON THE DCF MEDICAL
//*             TABLE, CREATE MONTHLY IMPROVEMENT CODE (MEDEXYYMM)
//* DATE:      04/22/2022 FOR DAF21
//* CONTACT:   ALEX BRYCE
//* MPR EMAIL: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//* INPUT
//MED  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFMED.SA.V1,DISP=SHR
//FIND DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.MASTFIND.SA.V1,DISP=SHR
//* OUTPUT
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIEDCF.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP3 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP4 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMPS1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN NOCENTER OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */

%MACRO START;
* MERGE DCF MED FILE WITH DEMO SSN LINK FILE TO ELIMINATE SSNS
  THAT ARE NOT IN THE DAF *;
PROC SORT DATA=MED.MED(KEEP=COSSN BLND_ONST_DT
```



Appendix A.72

JCL/SAS Code: DCFMED

```

        WHERE=(BLND_ONST_DT GT .Z)) NODUPKEY
        OUT=TEMP1.BLIND;
        BY COSSN BLND_ONST_DT;
RUN;

*KEEP THE EARLIEST BLIND DATE RECORDS FOR SSNS;
DATA TEMP1.BLIND_FINAL;
SET TEMP1.BLIND;
BY COSSN BLND_ONST_DT;
IF FIRST.COSSN THEN OUTPUT;
RUN;

*SELECT SUBSET OF DCFMED WE NEED TO REDUCE STRAIN ON SQL PROCESSING;
DATA TEMPS1.MED_SUBSET
  (KEEP=COSSN CTL_DT DETN_SEQ_NUM MED_STDT LU_TS MDR DRY_PERM_CD)
  ;
SET MED.MED
  (WHERE=(LU_PGM_NM NE 'CDCNVCLM' AND
          CID = '00'      AND
          MED_STDT NE .))
  ;
RUN;

PROC SORT DATA=TEMPS1.MED_SUBSET;
BY COSSN;
RUN;

* MERGE MED WITH MASTFIND BY SSN *;
PROC SQL;
CREATE TABLE TEMP2.DCF AS
SELECT A.SSN,
       B.CTL_DT,
       B.DETN_SEQ_NUM,
       B.MED_STDT,
       B.LU_TS,
       B.MDR,
       B.DRY_PERM_CD,
       C.BLINDDT
FROM FIND.MASTFIND AS A
LEFT JOIN
  (SELECT COSSN AS SSN,
          CTL_DT,
          DETN_SEQ_NUM,
          MED_STDT,
          LU_TS,

```

## Appendix A.72

JCL/SAS Code: DCFMED

```
        MDR,
        DRY_PERM_CD
    FROM TEMPS1.MED_SUBSET) AS B
ON B.SSN=A.SSN
LEFT JOIN
    (SELECT COSSN AS SSN,
        BLND_ONST_DT AS BLINDDT
    FROM TEMP1.BLIND_FINAL) AS C
ON C.SSN=A.SSN
ORDER BY A.SSN, B.CTL_DT, B.DETN_SEQ_NUM, B.MED_STDT, B.LU_TS
;
QUIT;
```

```
* FIND THE MOST RECENT LU_TS *;
* FOR EACH SSN-CTL_DT-DETN_SEQ_NUM-MED_STDT GROUP;
```

```
PROC SQL;
CREATE TABLE TEMP3.DCF AS
SELECT *,
    MAX(LU_TS) AS LU_TS_MAX
FROM TEMP2.DCF
GROUP BY SSN,CTL_DT,DETN_SEQ_NUM,MED_STDT;
QUIT;
```

```
* OUTPUT DATASET WITH MOST LU_TS *;
DATA TEMP3.DCF(DROP=LU_TS_MAX);
SET TEMP3.DCF;
```

```
* RESET MED_STDT TO THE FIRST DAY OF THE MONTH *;
MED_STDT_FIRST=MDY(MONTH(MED_STDT),01,YEAR(MED_STDT));
```

```
IF LU_TS=LU_TS_MAX THEN OUTPUT;
RUN;
```

```
PROC SORT DATA=TEMP3.DCF;
BY SSN CTL_DT DETN_SEQ_NUM MED_STDT_FIRST LU_TS;
RUN;
```

```
* CREATE MEDEX AND MONTHLY MEDEX VARIABLES *;
DATA TEMP4.MIEDCF;
SET TEMP3.DCF;
```

```
BY SSN /*CTL_DT DETN_SEQ_NUM MED_STDT_FIRST LU_TS*/;
```

```
* CREATE MEDEX VARIABLE *;
```

## Appendix A.72

JCL/SAS Code: DCFMED

```
IF MDR='3' THEN MEDEX='P';
ELSE IF MDR IN ('V','W','X') THEN DO;
  IF DRY_PERM_CD='P' THEN MEDEX='N';
  ELSE IF DRY_PERM_CD NE 'P' THEN MEDEX='P';
END;
ELSE IF MDR IN ('5','7') THEN MEDEX='N';
ELSE IF MDR IN ('A','B','C','D','E','F','G','H','I','J',
  'K','L','M','N','O','P','Q','R','S','T',
  'U','Y','Z') THEN MEDEX='E';
ELSE MEDEX='0';

* CREATE MONTHLY MEDEX *;
* SET ARRAY FOR ASSIGNING THE MEDEXYYMM FOR THE FIRST MONTH OF MEDEX
  CHANGES. IN OTHER WORDS NOT THE CARRYFORWARD FROM START MONTH TO
  SUBSEQUENT START MONTH;
ARRAY MEDEXA (&BEGYR.:&ENDYR.,1:12) $
  %DO YEAR=&BEGYR %TO &ENDYR;
  %LET YR=%SUBSTR(&YEAR.,3,2);
  MEDEX&YR.01-MEDEX&YR.12
  %END;;

IF MED_STDT_FIRST NE . THEN DO;
  IF MED_STDT_FIRST<"01JAN&BEGYR."D THEN MEDEXA(1994,1)=MEDEX;
  ELSE IF MED_STDT_FIRST<="31DEC&ENDYR."D THEN
    MEDEXA(YEAR(MED_STDT_FIRST),MONTH(MED_STDT_FIRST))=MEDEX;
END;

* SET ARRAY FOR ASSIGNING THE MEDEXYYMM BETWEEN ON START MONTH TO
  THE NEXT START MONTH OF MEDEX;
ARRAY MEDEXB (*) $
  %DO YEAR=&BEGYR %TO &ENDYR;
  %LET YR=%SUBSTR(&YEAR.,3,2);
  MEDEX&YR.01-MEDEX&YR.12
  %END;;

DO I=2 TO DIM(MEDEXB);
  IF MEDEXB(I)="" THEN MEDEXB(I)=MEDEXB(I-1);
END;

RUN;

* CHECK CREATION OF MEDEX *;
TITLE "CHECK CREATION OF MEDEX";
PROC FREQ DATA=TEMP4.MIEDCF;
  TABLES MDR*DRY_PERM_CD*MEDEX / LIST MISSING;
```

Appendix A.72

JCL/SAS Code: DCFMED

```
RUN;
TITLE;

* COLLAPSE ALL DATA INTO ONE RECORDS PER EACH BENE *;
DATA OUT.MIEDCF;
  UPDATE TEMP4.MIEDCF(OBS=0 KEEP=SSN BLINDDT MEDEX:)
    TEMP4.MIEDCF(KEEP=SSN BLINDDT MEDEX:);
  BY SSN;
RUN;

TITLE "CONTENTS FOR OUT MIEDCF";
PROC CONTENTS DATA=OUT.MIEDCF;
RUN;
TITLE;

%MEND;
%START;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DCFMED IS COMPLETE
```

## Appendix A.73

JCL/SAS Code: MEDEX

```
//#6502MED JOB (12510000,T715,,SAS,,ITC9FL),ALEX-BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   DAF21 CONSTRUCTION
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DEMO.PRDLIB(MEDEXD)
//* DESCRIPTION: CREATE PATCH FILE FOR MEDEX
//*           MERGE 831/832&833 AND DCFMED FILES
//*           PROPAGATE VALUES FORWARD
//* DATE:      04/22/2022 FOR DAF21
//* CONTACT:   ALEX BRYCE
//* MPR EMAIL: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//DCF  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIEDCF.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.ALLCDCOM.SA.V1,DISP=SHR
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PATMED.SA.V1,
//  DISP=(MOD,CATLG,DELETE),
//  UNIT=TSILO
//SYSIN DD *
```

```
OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN NOCENTER OBS=MAX;
```

```
%LET BEGYR=1994;
```

```
%LET ENDYR=2021; /* CHANGE AS NEEDED */
```

```
%MACRO START;
```

```
* MERGE 831 & 832/833 DATA WITH MEDICAL DCF TABLE;
```

```
* FROM THE 831 & 832/833 KEEP MIEX FROM 1994-2007 AND JAN 2008-APR 2008;
```

```
* FROM THE DCF KEEP MEDEX MAY 2008-DEC OF DAF YEAR;
```

```
DATA OUT.PATMED;
```

```
  MERGE T831.ALLCDCOM(IN=P831 KEEP=%DO YEAR=&BEGYR. %TO 2007;
```

```
    %LET YR=%SUBSTR(&YEAR.,3,2);
```

```
    MIEX&YR.01-MIEX&YR.12
```

```
  %END;
```

```
  MIEX0801-MIEX0804 SSN
```

```
  DODEC: RID: RDT:
```

```
  RENAME=(%DO YEAR=&BEGYR. %TO 2007;
```

```
    %LET YR=%SUBSTR(&YEAR.,3,2);
```

```
  MIEX&YR.01-MIEX&YR.12=MEDEX&YR.01-MEDEX&YR.12
```

Appendix A.73

JCL/SAS Code: MEDEX

```

                %END;
                MIEX0801-MIEX0804=MEDEX0801-MEDEX0804))
    DCF.MIEDCF(IN=DCF KEEP=MEDEX0805-MEDEX0812
                %DO YEAR=2009 %TO &ENDYR.;
                %LET YR=%SUBSTR(&YEAR.,3,2);
                MEDEX&YR.01-MEDEX&YR.12
                %END; SSN)
                ;
    BY SSN;

    * STRING MEDEX VARIABLE BETWEEN POPULATED MONTHS;
    ARRAY MEDEX (*) %DO YEAR=&BEGYR %TO &ENDYR;
                %LET YR=%SUBSTR(&YEAR.,3,2);
                MEDEX&YR.01-MEDEX&YR.12
                %END;;

    DO I=2 TO DIM(MEDEX);
        IF MEDEX(I)="" THEN MEDEX(I)=MEDEX(I-1);
    END;

    RUN;

    PROC FREQ DATA=OUT.PATMED;
        TABLE MEDEX: / MISSING;
    RUN;

    %MEND START;
    %START;

    //*****
    //* SENDING EMAIL FROM HOST USING IEBGENER
    //*****
    //SENDNOTE EXEC PGM=IEBGENER
    //SYSIN DD DUMMY
    //SYSPRINT DD SYSOUT=*
    //SYSUT2 DD SYSOUT=(B,SMTP)
    //SYSUT1 DD *
    HELO MVSHOST
    MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
    RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
    DATA
    FROM: <ABRYCE@MATHEMATICA-MPR.COM>
    TO: <ABRYCE@MATHEMATICA-MPR.COM>
    SUBJECT: MEDEX IS COMPLETE

```

## Appendix A.74

JCL/SAS Code: SPMIE

```
//#6502MIE JOB (12510000,T715,,SAS,,ITC9FL),ALEX-BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   BUILD DAF21 (50214)
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DEMO.PRDLIB(SPMIE)
//* DESCRIPTION: SPLIT FILE INTO ANNUAL FILES
//* PROGRAMMER: ALEX BRYCE
//* EMAIL:     ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
// SET REG='64M'
//*
//JS010 EXEC SAS9,
//      WORK='200000,100000', F10
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
//***DATA FOR SPLIT
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PATMED.SA.V1,DISP=SHR
//***OUT YEARLY FILES
//OUT94 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1994.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT95 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1995.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT96 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1996.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT97 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1997.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT98 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1998.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT99 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1999.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT00 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2000.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT01 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2001.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
```

## Appendix A.74

JCL/SAS Code: SPMIE

```
//      UNIT=TSILO,VOL=(,,10)
//OUT02 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2002.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT03 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2003.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT04 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2004.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT05 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2005.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT06 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2006.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT07 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2007.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT08 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2008.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT09 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2009.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2010.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT11 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2011.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2012.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT13 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2013.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT14 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2014.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT15 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2015.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2016.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
```



## Appendix A.74

JCL/SAS Code: SPMIE

```
//      UNIT=TSILO,VOL=(,,10)
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2017.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2018.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2019.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2020.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2021.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT OBS=MAX;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(MSPLIT)";

%OUTP(IN1.PATMED,OUT,PATMED,1994,2021,SSN);

PROC CONTENTS DATA=OUT94.PATMED;RUN;
PROC CONTENTS DATA=OUT95.PATMED;RUN;
PROC CONTENTS DATA=OUT96.PATMED;RUN;
PROC CONTENTS DATA=OUT97.PATMED;RUN;
PROC CONTENTS DATA=OUT98.PATMED;RUN;
PROC CONTENTS DATA=OUT99.PATMED;RUN;
PROC CONTENTS DATA=OUT00.PATMED;RUN;
PROC CONTENTS DATA=OUT01.PATMED;RUN;
PROC CONTENTS DATA=OUT02.PATMED;RUN;
PROC CONTENTS DATA=OUT03.PATMED;RUN;
PROC CONTENTS DATA=OUT04.PATMED;RUN;
PROC CONTENTS DATA=OUT05.PATMED;RUN;
PROC CONTENTS DATA=OUT06.PATMED;RUN;
PROC CONTENTS DATA=OUT07.PATMED;RUN;
PROC CONTENTS DATA=OUT08.PATMED;RUN;
PROC CONTENTS DATA=OUT09.PATMED;RUN;
PROC CONTENTS DATA=OUT10.PATMED;RUN;
PROC CONTENTS DATA=OUT11.PATMED;RUN;
PROC CONTENTS DATA=OUT12.PATMED;RUN;
PROC CONTENTS DATA=OUT13.PATMED;RUN;
PROC CONTENTS DATA=OUT14.PATMED;RUN;
PROC CONTENTS DATA=OUT15.PATMED;RUN;
```

Appendix A.74

JCL/SAS Code: SPMIE

```
PROC CONTENTS DATA=OUT16.PATMED;RUN;
PROC CONTENTS DATA=OUT17.PATMED;RUN;
PROC CONTENTS DATA=OUT18.PATMED;RUN;
PROC CONTENTS DATA=OUT19.PATMED;RUN;
PROC CONTENTS DATA=OUT20.PATMED;RUN;
PROC CONTENTS DATA=OUT21.PATMED;RUN;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
TO: <ABRYCE@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 SPMIE COMPLETE
```

## Appendix A.75

JCL/SAS Code: DXPROC

```
//#6502DXP JOB (12510000,T715,,SAS,,ITC9FL),BRYCE-ALEX,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//JOB01 EXEC SAS9,
//      WORK='180000,90000'
//*****
//* PROJECT:   BUILD DAF21 (50214)
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DEMO.PRDLIB(DXPROC)
//* DESCRIPTION: TO CREATE DIAGNOSIS VARIABLES FOR DAF
//* CONTACT:   ALEX BRYCE
//* EMAIL:     ABRYCE@MATHEMATICA-MPR.COM
//*           ALEX.BRYCE@SSA.GOV
//*****
//*
//IN1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.ALLCDCOM.SA.V1,DISP=SHR
//IN2 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBRREST.SA.V1,DISP=SHR
//IN3 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIELIG.SA.V1,DISP=SHR
//TEMP DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPROC.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
```

```
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS=MAX;
```

```
%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */
```

```
/* STEP TO ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA */
%MACRO START;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";
```

```
DATA OUT.DXPROC;
MERGE
  IN1.ALLCDCOM(IN = IN1 KEEP = SSN DX:)
  IN2.MBR (IN=IN2 KEEP=SSN DX:)
  IN3.SSIELIG (IN=IN3 KEEP=SSN DIB_DIG DIB_DIG2)
;
BY SSN;
```

```
LENGTH
%DO MM = 1 %TO &TOTMOS;
```

## Appendix A.75

JCL/SAS Code: DXPROC

```

DXPRI_DI&&YY&MM DXSEC_DI&&YY&MM
DXPRIBEST&&YY&MM DXSECBEST&&YY&MM
%END;
DXPRI_DI DXSEC_DI
DXPRI_SSI DXSEC_SSI
DXPRIBEST DXSECBEST
$4;

/*INPUT ARRAYS*/
ARRAY ARPRI_DI831(*) $ %DO MM = 1 %TO &TOTMOS; DXPRI_DI831&&YY&MM %END;;
ARRAY ARSEC_DI831(*) $ %DO MM = 1 %TO &TOTMOS; DXSEC_DI831&&YY&MM %END;;
ARRAY ARPRI_DIMBR(*) $ %DO MM = 1 %TO &TOTMOS; DXPRI_DIMBR&&YY&MM %END;;
ARRAY ARSEC_DIMBR(*) $ %DO MM = 1 %TO &TOTMOS; DXSEC_DIMBR&&YY&MM %END;;
ARRAY ARPRI_SSI(*) $ %DO MM = 1 %TO &TOTMOS; DXPRI_SSI&&YY&MM %END;;
ARRAY ARSEC_SSI(*) $ %DO MM = 1 %TO &TOTMOS; DXSEC_SSI&&YY&MM %END;;

/* NEW ARRAYS */
ARRAY ARPRI_DI(*) $ %DO MM = 1 %TO &TOTMOS; DXPRI_DI&&YY&MM %END;;
ARRAY ARSEC_DI(*) $ %DO MM = 1 %TO &TOTMOS; DXSEC_DI&&YY&MM %END;;
ARRAY ARPRIBEST(*) $ %DO MM = 1 %TO &TOTMOS; DXPRIBEST&&YY&MM %END;;
ARRAY ARSECBEST(*) $ %DO MM = 1 %TO &TOTMOS; DXSECBEST&&YY&MM %END;;

DO I = 1 TO &TOTMOS;
  ARPRI_DI(I) = ' ';
  ARSEC_DI(I) = ' ';
  ARPRIBEST(I) = ' ';
  ARSECBEST(I) = ' ';
END;

*CREATE DI MONTHLY DIAGNOSIS VARIABLES DXPRI_DIIYMM AND DXSEC_DIIYMM;
*POPULATE USING 831 CODES FIRST;
DO I = 1 TO DIM(ARPRI_DI831);
  IF ARPRI_DI831(I) NE ' ' THEN DO;
    ARPRI_DI(I) = ARPRI_DI831(I);
    ARSEC_DI(I) = ARSEC_DI831(I);
  END;
  ELSE DO;
    ARPRI_DI(I) = ARPRI_DIMBR(I);
    ARSEC_DI(I) = ARSEC_DIMBR(I);
  END;
END;

*CREATE BEST MONTHLY DIAGNOSIS VARIABLES;
*POPULATE FIRST USING DI CODES, IF DI CODE IS MISSING THEN POPULATE;
*SSI CODE;

```

## Appendix A.75

JCL/SAS Code: DXPROC

```
DO I = 1 TO DIM(ARPRI_DI);
  IF ARPRI_DI(I) NE ' ' THEN DO;
    ARPRIBEST(I) = ARPRI_DI(I);
    ARSECBEST(I) = ARSEC_DI(I);
  END;
  ELSE DO;
    ARPRIBEST(I) = ARPRI_SSI(I);
    ARSECBEST(I) = ARSEC_SSI(I);
  END;
END;

/*CREATE THE MOST RECENT DIAGNOSIS CODES*/
%MACRO FINDMR(VAR = );

DO I = &TOTMOS TO 1 BY -1 UNTIL(DX&VAR. NE "");
DX&VAR. = AR&VAR.(I);
END;

%MEND FINDMR;
%FINDMR(VAR = PRI_DI);
%FINDMR(VAR = SEC_DI);
%FINDMR(VAR = PRI_SSI);
%FINDMR(VAR = SEC_SSI);
%FINDMR(VAR = PRIBEST);
%FINDMR(VAR = SECBEST);

IF DXPRI_SSI="" THEN DXPRI_SSI=DIB_DIG;
IF DXSEC_SSI="" THEN DXSEC_SSI=DIB_DIG2;
IF DXPRIBEST="" THEN DXPRIBEST=DIB_DIG;
IF DXSECBEST="" THEN DXSECBEST=DIB_DIG2;

RUN;
%MEND;
%START;

PROC CONTENTS; RUN;

//*****
/* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
```

Appendix A.75

JCL/SAS Code: DXPROC

```
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>  
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>  
DATA  
FROM: <ABRYCE@MATHEMATICA-MPR.COM>  
TO: <ABRYCE@MATHEMATICA-MPR.COM>  
SUBJECT: DXPROC IS COMPLETE
```

## Appendix A.76

JCL/SAS Code: SPLIT2AN

```
//#6502SPL JOB (12510000,T715,,SAS,,ITC9FL),BRYCE-ALEX,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   BUILD DAF21 (50214)
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DEMO.PRDLIB(SPLIT2AN)
//* PURPOSE:   SPLIT DXPROC INTO ANNUAL FILES
//* CREATED:   SERGE LUKASHANETS 5/26/2016
//* E-MAIL:    ABRYCE@MATHEMATICA-MPR.COM
//*           ALEX.BRYCE@SSA.GOV
//*****
// SET REG='64M'
//*
//JS010 EXEC SAS9,
//      WORK='200000,100000', F10
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
//IN1   DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPROC.SA.V1,DISP=SHR
//OUT94 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY94.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT95 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY95.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT96 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY96.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT97 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY97.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT98 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY98.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT99 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY99.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT00 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY00.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT01 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY01.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT02 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY02.SA.V1,
```

## Appendix A.76

JCL/SAS Code: SPLIT2AN

```
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT03 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY03.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT04 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY04.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT05 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY05.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT06 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY06.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT07 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY07.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT08 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY08.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT09 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY09.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY10.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT11 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY11.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY12.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT13 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY13.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT14 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY14.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT15 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY15.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY16.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY17.SA.V1,
```



## Appendix A.76

JCL/SAS Code: SPLIT2AN

```
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY18.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY19.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY20.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY21.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2021;

* GET VARIABLE LIST FROM INPUT DATASET *;
PROC CONTENTS DATA=IN1.DXPROC NOPRINT
      OUT=TEMP.CONTS(KEEP=NAME);
RUN;

%MACRO SPLITV;

* KEEP ONLY VARIABLES WITHOUT YMMM *;
DATA TEMP.CONTS(DROP=YR MO YMMM);
  SET TEMP.CONTS;
  DO YR=&BEGYR. TO &ENDYR.;
    DO MO=1 TO 12;
      IF MO<=9 THEN YMMM=SUBSTR(STRIIP(YR),3,2) || "0" || STRIP(MO);
      ELSE YMMM=SUBSTR(STRIIP(YR),3,2) || STRIP(MO);

      IF LENGTH(NAME)>=4 THEN DO;
        IF REVERSE(SUBSTR(REVERSE(STRIIP(NAME)),1,4))
          EQ YMMM THEN DELETE;
      END;
```

## Appendix A.76

JCL/SAS Code: SPLIT2AN

```
        END; /*END OF MO LOOP*/
        END; /*END OF YR LOOP*/
RUN;

PROC SORT DATA=TEMP.CONTS NODUPKEY OUT=TEMP1.CONTS;
  BY NAME;
RUN;

* CREATE MACRO VARIABLE WHICH CONTAINS ALL COMMON VARIABLES *;
PROC SQL;
  SELECT NAME INTO :COMMVAR SEPARATED BY ' '
  FROM TEMP1.CONTS;
QUIT;

%PUT COMMON VARIABLES=&COMMVAR;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(MSPLIT)";

%OUTP(IN1.DXPROC,OUT,DXPROC,&BEGYR,&ENDYR,&COMMVAR);

%DO YEAR=&BEGYR %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR,3,2);
TITLE "CONTENTS OF &YEAR DXPROC DATA";
PROC CONTENTS DATA=OUT&YR..DXPROC;
RUN;
TITLE;
%END;

%MEND SPLITV;
%SPLITV;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2  DD SYSOUT=(B,SMTP)
//SYSUT1  DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: <ABRYCE@MATHEMATICA-MPR.COM>
```

Appendix A.76

JCL/SAS Code: SPLIT2AN

TO: <ABRYCE@MATHEMATICA-MPR.COM>  
SUBJECT: SPLIT2AN IS COMPLETE

## Appendix A.77

JCL/SAS Code: CRDMG

```
//#8047DMG JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//JOB01 EXEC SAS9,
//      WORK='180000,90000'
//*****
//* PROJECT:  DAF 50214
//* DIRECTOR: JODY SCHIMMEL HYDE
//* PROGRAM:  OPDR.TG.PRD.ETTW.#8047.DAF21.DEMO.PRDLIB(CRDMG)
//* PURPOSE:  TO CREATE DEMOGRAPHIC VARIABLES FILE FOR DAF
//* CONTACT:  MIRIAM CHAPPELKA
//* MPR PHONE:
//* E-MAIL:   MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//* PREVIOUS YEAR DEMO FILE (USE THE FINAL VERSION)
//DEM  DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.DMG,DISP=SHR
//*
//SSR  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIELIG.SA.V1,DISP=SHR
//MBR  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBRREST.SA.V1,DISP=SHR
//NUM  DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.NUMCOMB.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.ALLCDCOM.SA.V1,DISP=SHR
//CER  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.COMBCERE.SA.V1,DISP=SHR
//T833 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.RSTRCT16.SA.V1,DISP=SHR
//FIN  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.MASTFIND.SA.V1,DISP=SHR
//TWP  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.TWPCMPL.SA.V1,DISP=SHR
//* MIEDCF IS THE SOURCE FOR THE BLINDDT
//MED  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIEDCF.SA.V1,DISP=SHR
//* PATCHED MEDEX FILE - THIS IS THE SOURCE FOR MEDEX
//MEDX DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PATMED.SA.V1,DISP=SHR
//DXP  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPROC.SA.V1,DISP=SHR
//TEMP DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.DMG.SA.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *
```

```
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS=MAX
      MPRINT MACROGEN;
```

```
%LET BEGYR=1994;
```

## Appendix A.77

JCL/SAS Code: CRDMG

```
%LET ENDDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

%LET DAFYR=21; /* CHANGE AS NEEDED */

/* OLD_DOB IS THE MACRO VARIABLE NEEDED BY THE FRACALC PROGRAM
   THAT CALCULATED THE FRADATE AND THE FULL RETIREMENT AGE (FRA
   VARIABLES */
%LET OLD_DOB=DOBBEST; /* CHANGE AS NEEDED */

%MACRO START;

/* STEP TO ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA */
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

/* MACRO VAR TO SET STARTING POINT FOR SORD PROCESSING. WE ONLY
   WE ONLY PROCESS THE CURRENT SORD YEAR IN EACH ITERATION OF
   DAF. ADJUST THIS EACH NEW ITERATION */

%LET SORDSTART=%EVAL((&ENDDYR - &BEGYR)*12+1);

/* CONVERT CHARACTER VARIABLES TO NUMERIC IN ALLCDCOME
   BEFORE MERGE. */
DATA TEMP.ALLCDCOM (DROP=FIRSTPDDOLD PDDOLD:);
  SET T831.ALLCDCOM (KEEP=SSN DODEC: JUDLV: RDT: RID: RB:
    DPM: BFD: DOC: EPE: DD: DT: PD: RFC:
    VRA: DR: FIRST:QD: DSICLAIM:
    RENAME=(PDD1-PDD16 = PDDOLD1-PDDOLD16
      FIRSTPDD = FIRSTPDDOLD
    ));

  ARRAY OLD FIRSTPDDOLD PDDOLD1-PDDOLD16;
  ARRAY NEW FIRSTPDD PDD1-PDD16;

  DO I = 1 TO DIM(OLD);
    IF OLD(I) NE . THEN
      NEW(I) = MDY(SUBSTR(OLD(I),5,2),1,SUBSTR(OLD(I),1,4));
  END;
RUN;

DATA OUT1.DMG
  OUT2.DMSTW(KEEP = SSN CAN DOBBEST DODBEST TWPCMPLMNT1
    TOA MFT FRA FRADATE SEX DOEI STATE_MR
```

Appendix A.77

JCL/SAS Code: CRDMG

```

        SSDI_ONSET SSIELIG_FIRST SSIELIG_ADULT);
    LENGTH AGE18_CDR DOBCER 8;

```

MERGE

```

    DEM.DMG (IN=IND RENAME = (SEX=OSEX))
    FIN.MASTFIND (IN=MAST)
    SSR.SSIELIG(IN=INL KEEP = SSN HUN TOA COMP_STA MFT SEXSSR
        START_RD DOBSSR DODSSR LAF CURSTAT
        ELG_RD APP_RD_8080_JD CLM_FIL DISPAYCD
        DIB_MDR DIB_DPM PDSCC PDZIP PDZIP6_9
        REPPAYTP STOP_RD MAX_80JD MAXAPPRD MAXCLMFL
        MAXELGRD MAXFRSTP MAXSTPRD MAXSTRRD LANGSSR
        MIN_80JD MINAPPRD MINCLMFL
        MINFRSTP MINSTPRD MINSTRRD DOC_SSI_MR
        PSTA_MR PSTA_MR_DATE
        SSI%SUBSTR(&BEGYR.,3,2)-SSI99
        SSI00-SSI%SUBSTR(&ENDYR.,3,2)
        SSIELIG_FIRST SSIELIG_ADULT
        RENAME=(
            REPPAYTP = PAYEE_TYP_SSI
            PDSCC = STATECOUNTY_SSR
            PDZIP6_9 = ZIP_EXT_REPPAY_SSR
            PDZIP = ZIP_REPPAY_SSR)
    )

```

```

T833.RSTRCT16(KEEP=SSN AGE18_CDR AGE18REDDT
    T16APPL: T16START: T16STOP:)

```

```

    MBR.MBR (IN=INM KEEP=SSN CAN BGN BIC TAC DUALELIG BLN COUNTY
        ADC: APS: BDC: BDOE: BDOF: DAC: DDBC: DDO:
        DSD: EBD: ENTDATD: HBIC: HDD: IME: LOD: PIA:
        PIED: SDS: SMI: HI: TOC: NHI NSMI FIRST:
        LANG DOEID DOECD MBRSEX STATE TOP
        NPJA: NODF LSAP LSDC LSF D LSPA SIFT SISC
        DOBMBR DOC_SSDI_MR LAF PNOB RDD RP RZIP SLAC
        DOEI DOEC DOST DODMBR CDR1-CDR12 CDRD DPI: NHDEMO
        DXPRI_MBR1-DXPRI_MBR12
        DXSEC_MBR1-DXSEC_MBR12
        DXPRI_DUAL DXSEC_DUAL
        NINSD DIBREQ: DIBHAS:
        SSDI%SUBSTR(&BEGYR.,3,2)-SSDI99
        SSDI00-SSDI%SUBSTR(&ENDYR.,3,2)
        SSDI_ONSET
        RENAME = (DACD=ACD_SSDI_DUAL
            DAC1 = ACD_SSDI1

```

Appendix A.77

JCL/SAS Code: CRDMG

```
DAC2 = ACD_SSDI2
DAC3 = ACD_SSDI3
DAC4 = ACD_SSDI4
DAC5 = ACD_SSDI5
DAC6 = ACD_SSDI6
DAC7 = ACD_SSDI7
DAC8 = ACD_SSDI8
DAC9 = ACD_SSDI9
DAC10 = ACD_SSDI10
DAC11 = ACD_SSDI11
DAC12 = ACD_SSDI12
ADCD = ADC_SSDI_DUAL
APSD = APS_SSDI_DUAL
BDCD = BDC_SSDI_DUAL
BGN = BENE_FRSTNAME_MBR
BLN = BENE_LASTNAME_MBR
COUNTY = COUNTY_RES_MBR
CDRD = DCR_SSDI_DUAL
CDR1 = DCR_SSDI1
CDR2 = DCR_SSDI2
CDR3 = DCR_SSDI3
CDR4 = DCR_SSDI4
CDR5 = DCR_SSDI5
CDR6 = DCR_SSDI6
CDR7 = DCR_SSDI7
CDR8 = DCR_SSDI8
CDR9 = DCR_SSDI9
CDR10= DCR_SSDI10
CDR11= DCR_SSDI11
CDR12= DCR_SSDI12
DDBCD = DDBC_DUAL
DDOD = DDO_DUAL
DSDD = DSD_DUAL
EBDD = EBD_DUAL
ENTDATD = ENTDAT_DUAL
HDDD = HDD_DUAL
LANG = LANG_MBR
LODD = LOD_DUAL
TOP = PAYEE_TYP_SSDI
SDSD = SDS_DUAL
STATE=STATE_MBR
RZIP=ZIP_RES_MBR
))
NUM.NUMCOMB(IN=INU KEEP=SSN DOD FRSTNAME LASTNAME
NAMESUFFIX DOBNUM NSEX NRACE
```

## Appendix A.77

JCL/SAS Code: CRDMG

```
        POBFORID POBSC CITIZEN_NUM
        RENAME=(DOD=DODNUM
        POBFORID=FOREIGN_BORN
        POBSC=COUNTRY_OF_BIRTH
        FRSTNAME = BENE_FRSTNAME_NUMI
        LASTNAME = BENE_LASTNAME_NUMI
        NAMESUFIX = BENE_SUFIXNAME_NUMI
        CITIZEN_NUM = CITIZEN_NUMI
        DOBNUM = DOBNUMI
        DODNUM = DODNUMI
    )
)
TEMP.ALLCDCOM(IN=INC)
CER.CER(IN=INR KEEP=SSN DOBCER RSEX STATECD_ENDYR
/*START WITH JAN OF DAF YEAR*/
%DO MM=&SORDSTART %TO &TOTMOS;
CIT&&YY&MM RDX&&YY&MM
%END;)

TWP.TWPCMPL(IN=INTW KEEP=SSN TWPCMPLMNTNTH:)
MED.MIEDCF(KEEP=SSN BLINDDT)
DXP.DXPROC(IN=INDX KEEP=SSN DXPRI_SSI DXSEC_SSI
DXPRI_DI DXSEC_DI DXPRIBEST DXSECBEST)
/* MEDEX PATCH FOR DAF19 */
MEDX.PATMED
;
BY SSN;

/* SUBSET TO VALID SSN'S IN THE MASTER FINDER FILE */
IF MAST AND SSN GT '000000000';

/* SCREEN OUT CASES WHICH HAVE THE WRONG DATA BECAUSE
2% OF MBR RECORDS HAVE THE WRONG BOAN. WE WILL CHECK
MBR BIRTH YEAR AND FIRST NAME AGAINST NUMIDENT BIRTH
YEAR AND FIRST NAME.WHERE THE FIRST NAME IS DIFFERENT
AND THE BIRTH YEARS ARE MORE THAN 2 YEARS APART WE WILL
DELETE THE RECORD. THE CHECK ON FIRST NAME MAKES SENSE
BECAUSE THESE CASES WILL MOST OFTEN OCCUR FOR DEPENDENTS
WHERE THE PRIMARY HAS THE SAME LAST NAME */

IF DOBMBR NE . THEN MBRYR = YEAR(DOBMBR);

/*ERROR MESSAGE IF DOBNUM HAS MISSING YEAR 042319XX*/
/*AND NUMYR WILL BE MISSING */
NUMYR = INPUT(SUBSTR(DOBNUMI,5,4),8.);
```



## Appendix A.77

JCL/SAS Code: CRDMG

```
IF MBRYR GT .Z AND NUMYR GT .Z THEN DO;
IF BENE_FRSTNAME_MBR NE BENE_FRSTNAME_NUMI
  AND ABS(MBRYR-NUMYR)>2 THEN DELETE;
END;

/* CREATE THE BEST DATE OF BIRTH - DOBBEST
THE NUMIDENT DATA OF BIRTH IS A CHARACTER VARIABLE.
IF NUMIDENT HAS VALID DATE VALUE, THEN DOBBEST=DOBNUM.
IF NUMIDENT HAS MISSING DAY/MONTH AND VALID YEAR, THEN
COMPARE THE YEAR TO DOBSSR, DOBMBR AND DOBCER IN ORDER.
WHERE AN EQUAL YEAR IS FOUND FILL IN THE MONTH AND DAY.
IF NO MATCHED YEAR FOUND, THEN SELECT THE FIRST
NONMISSING DATE VALUE FROM DOBSSR, DOBMBR AND DOBCER
IN ORDER. AT THE END OF THE PROCESS, IF DOBBEST IS
STILL MISSING AND NUMIDENT YEAR IS VALID, SET JAN 1 AS
THE MONTH AND DAY, CREATE DOBFLAG TO MARK THOSE CASES. */

/*ERROR MESSAGE IF DOBNUM HAS INVALID MONTH/DAY VALUE*/
/*NUMMO AND NUMDAY WILL BE MISSING, DOBBEST WILL BE MISSING*/
NUMMO = INPUT(SUBSTR(DOBNUMI,1,2),8.);
NUMDAY = INPUT(SUBSTR(DOBNUMI,3,2),8.);
DOBBEST = MDY(NUMMO,NUMDAY,NUMYR);

DOBFLAG = 0;

IF DOBBEST = . THEN DO;
  IF NUMYR NE . THEN DO;
    IF YEAR(DOBSSR) = NUMYR THEN DOBBEST = DOBSSR;
    ELSE IF YEAR(DOBMBR) = NUMYR THEN DOBBEST = DOBMBR;
    ELSE IF YEAR(DOBCER) = NUMYR THEN DOBBEST = DOBCER;
  END;
  /*IF NO MATCH IS FOUND SELECT THE FIRST VALID DOB FROM
  THE THREE OTHER DATES*/
  ELSE DOBBEST=COALESCE(DOBSSR,DOBMBR,DOBCER);
END;
/*IF AT THE END OF THE PROCESS, NUMIDENT YEAR IS THE
ONLY INFO AVAILABLE, CERATE DUMMY MONTH AND DAY, USE
THE NUMYR FOR DOBBEST. CREATE A FLAG FOR THESE CASES.*/
IF DOBBEST = . AND NUMYR NE . THEN DO;
  IF NUMMO NE . THEN DOBBEST = MDY(NUMMO,1,NUMYR);
  ELSE DOBBEST = MDY(1,1,NUMYR);
  DOBFLAG = 1;
END;
```

## Appendix A.77

JCL/SAS Code: CRDMG

```
/* CREATE THE BEST DATE OF DEATH - DODBEST
THE NUMIDENT DATA OF DEATH IS A CHARACTER VARIABLE.
IF NUMIDENT HAS VALIE DATE VALUE, THEN DODBEST=DODNUM.
IF NUMIDENT HAS MISSING DAY/MONTH AND VALID YEAR, THEN
COMPARE THE YEAR TO DODSSR AND DODMBR IN ORDER. WHERE
AN EQUAL YEAR IS FOUND FILL IN THE MONTH AND DAY.
IF NO MATCHED YEAR FOUND, THEN SELECT THE FIRST NONMISSING
DATE VALUE FROM DODSSR AND DODMBR IN ORDER.
AT THE END OF THE PROCESS, ID DODBEST IS TILL MISSING
AND NUMIDENT YEAR IS VALID, SET JAN 1 AS THE MONTH AND
DAY, CREATE DODFLAG TO MARK THOSE CASES.*/

/*ERROR MESSAGE IF DODNUM HAS MISSING YEAR 042319XX*/
/*AND NUMYRD,NUMMOD AND NUMDAYD WILL BE MISSING*/
/*DODBEST WILL BE MISSING*/
NUMYRD = INPUT(SUBSTR(DODNUMI,5,4),8.);
NUMMOD = INPUT(SUBSTR(DODNUMI,1,2),8.);
NUMDAYD = INPUT(SUBSTR(DODNUMI,3,2),8.);
DODBEST = MDY(NUMMOD,NUMDAYD,NUMYRD);

DODFLAG = 0;

IF DODBEST = . THEN DO;
  IF NUMYRD NE . THEN DO;
    IF YEAR(DODSSR) = NUMYRD THEN DODBEST = DODSSR;
    ELSE IF YEAR(DODMBR) = NUMYRD THEN DODBEST = DODMBR;
  END;
  /*IF NO MATCH IS FOUND SELECT THE FIRST VALID DODNUM FROM
  THE TWO OTHER DATES */
  ELSE DODBEST = COALESCE(DODSSR,DODMBR);
END;
/* IF AT THE END OF THE PROCESS, NUMIDENT YEAR IS THE ONLY
INFO AVAILABLE, CREATE DUMMY MONTH AND DAY AND USE THE
YEAR FOR DODBEST.CREATE A FLAG FOR THESE CASES */
IF DODBEST = . AND NUMYRD NE . THEN DO;
  IF NUMMOD NE . THEN DODBEST = MDY(NUMMOD,1,NUMYRD);
  ELSE DODBEST = MDY(1,1,NUMYRD);
  DODFLAG = 1;
END;

/* CALCULATE FULL RETIREMENT AGE */
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF20.UTILITY.PRDLIB(FRACALC)";

/* CREATE JUDGEMENT INDICATOR (FROM DODEC) VARIABLES
FROM 831 DATA */
```

## Appendix A.77

JCL/SAS Code: CRDMG

```
ARRAY DOD1(*) DODEC1-DODEC16;
ARRAY LEV(*) JUDLVL1-JUDLVL16;
ARRAY RDT(*) RDT1-RDT16;
ARRAY RID(*) RID1-RID16;

DO I = DIM(DOD1) TO 1 BY -1 UNTIL(DOD1(I) GT .Z);
  LAST_DODEC = DOD1(I);
END;

/* ESTABLISH THE DATE OF THE FIRST MIE */
/* USE THE VNAME FUNCTION TO CREATE THE VALUE FOR YEAR
  AND MONTH */
ARRAY MEDEXA (&BEGYR.:&ENDYR.,1:12) $
  %DO YEAR=&BEGYR %TO &ENDYR;
  %LET YR=%SUBSTR(&YEAR.,3,2);
  MEDEX&YR.01-MEDEX&YR.12
  %END;;

LENGTH MIENAME $8;
LENGTH MIE_BEFORE_CDR $1.;

MIENAME = ' ';

/* INITIALIZE INDICATOR VARIABLE FOR MIE TO BLANK.
  IF THERE IS A RECORD IN THE ALLCDR FILE THEN
  INITIALIZE TO 'N'. */
MIE_BEFORE_CDR = ' ';
IF INC THEN MIE_BEFORE_CDR = 'N';

DO YR=&BEGYR. TO &ENDYR. UNTIL (MIENAME NE ' ');
  DO MO=1 TO 12 UNTIL (MIENAME NE ' ');
    IF MEDEXA(YR,MO)='E' THEN DO;
      MIENAME = VNAME(MEDEXA(YR,MO));
      FIRSTMIE=MDY(MO,01,YR);
    END;
  END;
END;

/* DETERMINE IF MIE BEFORE THE FIRST CONTINUING DISABILITY
  REVIEW. THE MIE MONTHS ARE SET FROM THE DATE OF ENTITLEMENT
  TO DISABILTY OR THE DODEC FIELD IF THE FIRST IS MISSING.
  THE DATE OF ENTITLEMENTIS SET AFTER THE JUDGMENT AFFIRMS
  DISABILITY AND IS MOST OFTEN EARLIER THAN THE JUDGMENT DATE
  SINCE IT TAKES TIME FOR A JUDGMENT TO BE MADE. THEREFORE
  WE CHECK WHETHER THE FIRST MIE OCCURS EARLIER OR ON THE
```

## Appendix A.77

JCL/SAS Code: CRDMG

```

SAME DATE AS ANY ADJUDICATION EVENT ON THE 833 FILE. THERE
MAY BE MANY DENIALS BEFORE A CLAIM IS APPROVED SO WE NEED
TO CHECK THE RESULT OF DETERMINATION FIELD (RDT). THE DAY
PORTION OF THE FIRST MIE FIELD IS SET TO "1" SO IT WILL
ALWAYS BE LESS THAN OR = TO DODECN. BUT WE HAVE MONTHLY
MIE'S ONLY FROM 1994 - IF THE FIRST DODEC IS BEFORE THEN
WE ARE LIMITED IN ESTABLISHING MIE BEFORE CDR. */

```

```

IF FIRSTMIE NE . THEN DO;
/* STOP LOOPING WHEN AN 833 EVENT OCCURS */
DO I = 1 TO DIM(DOD1) WHILE (RID(I) IN ('2' 'R'));
  IF FIRSTMIE LE DOD1(I) AND RDT(I) = 'A'
    THEN MIE_BEFORE_CDR = 'Y';
  /* IF MIE DATE IS JAN 1 1994 WE CANNOT DETERMINE ITS
  RELATIONSHIP TO A CDR DATE AND SO WE SET THE VARIABLE
  TO MISSING */
  IF FIRSTMIE EQ '01JAN1994'D THEN MIE_BEFORE_CDR = '';
END;
END;

```

```

DROP MIENAME;

```

```

/* CREATE REPRESENTATIVE PAYEE INDICATOR FOR SSI */
IF PAYEE_TYP_SSI IN (' ','SEL') THEN REPPYSSI = 0;
ELSE IF PAYEE_TYP_SSI > '' THEN REPPYSSI = 1;
ELSE REPPYSSI = .;

```

```

/* CREATE REPRESENTATIVE PAYEE INDICATOR */
IF PAYEE_TYP_SSDI IN (' ','A') THEN REPPYSSD = 0;
ELSE IF PAYEE_TYP_SSDI > '' THEN REPPYSSD = 1;
ELSE REPPYSSD = .;

```

```

LENGTH REPPYSSI REPPYSSD 3;

```

```

/* TRIAL WORK PERIOD INDICATOR MOVED TO DCF */
IF MAXSTRRD = . AND MAXELGRD NE . THEN MAXSTRRD = MAXELGRD;
ELSE IF MAXSTRRD = . AND MAXELGRD = . THEN MAXSTRRD = .;

```

```

/* PROCESS SORD VARIABLES */
/* ONLY FOR CURRENT ONGOING AND NEW CASES */
IF INR THEN DO;

```

```

  ARRAY REMDX{*} $ %DO MM=&SORDSTART %TO &TOTMOS;
    RDX&&YY&MM
  %END;;
  ARRAY REMCT{*} $ %DO MM=&SORDSTART %TO &TOTMOS;

```

## Appendix A.77

JCL/SAS Code: CRDMG

```
CIT&&YY&MM
%END;;

DO I=1 TO DIM(REMDX);
  IF REMDX{I} NE '' THEN REMDIG1N=REMDX{I};
  IF REMCT{I} NE '' THEN CITIZENN=REMCT{I};
END;

DROP %DO MM=&SORDSTART %TO &TOTMOS;
  RDX&&YY&MM CIT&&YY&MM
%END;;
END; /* INR */

/* SET THE SORD VARIABLES - DO NOT OVERWRITE WITH BLANK */
IF CITIZEN_CER = '' THEN CITIZEN_CER = CITIZENN;
IF REMDIG1 = '' THEN REMDIG1 = REMDIG1N;
DROP CITIZENN REMDIG1N ;

/* RECODE NUMIDENT SEX */
IF NSEX = '1' THEN NUM_SEX = 'M';
ELSE IF NSEX = '2' THEN NUM_SEX = 'F';

/* GET SEX AS THE FIRST NONMISSING VALUE FROM NUMIDENT,
  SEXSSR, MBRSEX, RSEX AND OSEX IN ORDER. */
LENGTH SEX $1.;
SEX = COALESCEC(NUM_SEX,SEXSSR,MBRSEX,RSEX,OSEX);

DROP SEXSSR MBRSEX NSEX RSEX NUM_SEX OSEX;

/* CREATE INDICATOR VARIABLE FOR MALE */
IF SEX = 'M' THEN MALE = 1;
ELSE IF SEX = 'F' THEN MALE = 0;
ELSE      MALE = .;

/* CREATE INDICATOR VARIABLE FOR MISSING SEX */
SEXMISS = 0;
IF SEX = '' THEN SEXMISS = 1;

/*CREATE BOND PARTICIPATION INDICATOR*/
ARRAY DPI(*) $ DPI1 - DPI10;

BOND = .;
DO I = 1 TO DIM(DPI) UNTIL (BOND NE .);
  IF DPI(I) = "DD" THEN BOND = 1;
END;
```

## Appendix A.77

JCL/SAS Code: CRDMG

```
IF INM = 1 AND BOND NE 1 THEN BOND = 0;

/*CREATE MOST RECENT DATES FOR HI AND SMI*/
HI_START_MR = MAX(OF HI_START1-HI_START10);
HI_TERM_MR = MAX(OF HI_TERM1-HI_TERM10);
SMI_START_MR = MAX(OF SMI_START1-SMI_START20);
SMI_TERM_MR = MAX(OF SMI_TERM1-SMI_TERM20);

/*CREATE THE MOST RECENT MBR DIG1 AND DIG2 VALUES*/
LENGTH DXPRI_MBR_MR DXSEC_MBR_MR $4;
ARRAY DIGAR1(12) DXPRI_MBR1-DXPRI_MBR12;
ARRAY DIGAR2(12) DXSEC_MBR1-DXSEC_MBR12;

DO I = 12 TO 1 BY -1 UNTIL(DIGAR1(I) NE ' ');
  DXPRI_MBR_MR = DIGAR1(I);
END;

DO I = 12 TO 1 BY -1 UNTIL(DIGAR2(I) NE ' ');
  DXSEC_MBR_MR = DIGAR2(I);
END;

*RACE CONSTRUCTION;
IF NRACE="0" THEN RACE='U';
ELSE IF NRACE="1" THEN RACE='W';
ELSE IF NRACE="2" THEN RACE='B';
ELSE IF NRACE="3" THEN RACE='O';
ELSE IF NRACE="4" THEN RACE='A';
ELSE IF NRACE="5" THEN RACE='H';
ELSE IF NRACE="6" THEN RACE='I';

DROP MBRYR NUMYR NUMDAY NUMMO NUMYRD NUMDAYD NUMMOD I NRACE
  %DO MM=1 %TO &TOTMOS; MEDEX&&YY&MM %END;;

RUN;
%MEND;
%START;
PROC CONTENTS DATA=OUT1.DMG;
TITLE "DEMO FOR DAF&DAFYR.";
RUN;

TITLE 'DEMO FREQS';
PROC FREQ DATA=OUT1.DMG;
  TABLES SEX: CITIZEN: LANG_MBR LANGSSR
    FIRSTMIE MIE_BEFORE_CDR DUALELIG
```

Appendix A.77

JCL/SAS Code: CRDMG

```
        DOBBEST DOBFLAG DODBEST DODFLAG
        DPI1 BOND FIRSTPDD PDD1-PDD16
        SSIELIG_FIRST SSIELIG_ADULT RACE;
FORMAT DOBBEST DODBEST FIRSTMIE
        SSIELIG_FIRST SSIELIG_ADULT YYMMN6.;
RUN;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2  DD SYSOUT=(B,SMTP)
//SYSUT1  DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: SUBMISSION DAF21 CRDMG IS COMPLETE
CRDMG IS COMPLETE.
```

Appendix A.78

JCL/SAS Code: DEMOLBLS

```
%MACRO GETLB;
LABEL
RACE = "RACE OR ETHNICITY"
ADC_SSDI_DUAL="SSDI DUAL ELIG-DISAB CESSATION DATE-SUPPLEMENTARY RECORD"
ADC1="SSDI DISABILITY CESSATION DATE 1"
ADC10="SSDI DISABILITY CESSATION DATE 10"
ADC11="SSDI DISABILITY CESSATION DATE 11"
ADC12="SSDI DISABILITY CESSATION DATE 12"
ADC2="SSDI DISABILITY CESSATION DATE 2"
ADC3="SSDI DISABILITY CESSATION DATE 3"
ADC4="SSDI DISABILITY CESSATION DATE 4"
ADC5="SSDI DISABILITY CESSATION DATE 5"
ADC6="SSDI DISABILITY CESSATION DATE 6"
ADC7="SSDI DISABILITY CESSATION DATE 7"
ADC8="SSDI DISABILITY CESSATION DATE 8"
ADC9="SSDI DISABILITY CESSATION DATE 9"
AGE18_CDR="SSI ADULT REDETERMINATION INDICATOR"
AGE18REDDT="SSI FAVORABLE AGE 18 REDETERMINATION DATE"
APP_RD="SSI APPLICATION DATE (MOST RECENT)"
APS_SSDI_DUAL="SSDI DUAL ELIG-APPEALS DATE- SUPPLEMENTARY RECORD"
APS1="SSDI APPEALS DATE 1"
APS10="SSDI APPEALS DATE 10"
APS11="SSDI APPEALS DATE 11"
APS12="SSDI APPEALS DATE 12"
APS2="SSDI APPEALS DATE 2"
APS3="SSDI APPEALS DATE 3"
APS4="SSDI APPEALS DATE 4"
APS5="SSDI APPEALS DATE 5"
APS6="SSDI APPEALS DATE 6"
APS7="SSDI APPEALS DATE 7"
APS8="SSDI APPEALS DATE 8"
APS9="SSDI APPEALS DATE 9"
BDC_SSDI_DUAL=
"SSDI DUAL ELIG-BASIS FOR DENIAL CODE-SUPPLEMENTARY RECORD"
BDC1="SSDI BASIS FOR DENIAL CODE 1"
BDC10="SSDI BASIS FOR DENIAL CODE 10"
BDC11="SSDI BASIS FOR DENIAL CODE 11"
BDC12="SSDI BASIS FOR DENIAL CODE 12"
BDC2="SSDI BASIS FOR DENIAL CODE 2"
BDC3="SSDI BASIS FOR DENIAL CODE 3"
BDC4="SSDI BASIS FOR DENIAL CODE 4"
BDC5="SSDI BASIS FOR DENIAL CODE 5"
BDC6="SSDI BASIS FOR DENIAL CODE 6"
BDC7="SSDI BASIS FOR DENIAL CODE 7"
```



Appendix A.78

JCL/SAS Code: DEMOLBLS

BDC8="SSDI BASIS FOR DENIAL CODE 8"  
BDC9="SSDI BASIS FOR DENIAL CODE 9"  
DODMBR="SSDI DATE OF DEATH - MBR"  
BDOE\_START1="SSDI DATE OF ENTITLEMENT START 1"  
BDOE\_START10="SSDI DATE OF ENTITLEMENT START 10"  
BDOE\_START11="SSDI DATE OF ENTITLEMENT START 11"  
BDOE\_START12="SSDI DATE OF ENTITLEMENT START 12"  
BDOE\_START13="SSDI DATE OF ENTITLEMENT START 13"  
BDOE\_START14="SSDI DATE OF ENTITLEMENT START 14"  
BDOE\_START15="SSDI DATE OF ENTITLEMENT START 15"  
BDOE\_START16="SSDI DATE OF ENTITLEMENT START 16"  
BDOE\_START17="SSDI DATE OF ENTITLEMENT START 17"  
BDOE\_START18="SSDI DATE OF ENTITLEMENT START 18"  
BDOE\_START19="SSDI DATE OF ENTITLEMENT START 19"  
BDOE\_START2="SSDI DATE OF ENTITLEMENT START 2"  
BDOE\_START20="SSDI DATE OF ENTITLEMENT START 20"  
BDOE\_START21="SSDI DATE OF ENTITLEMENT START 21"  
BDOE\_START22="SSDI DATE OF ENTITLEMENT START 22"  
BDOE\_START23="SSDI DATE OF ENTITLEMENT START 23"  
BDOE\_START24="SSDI DATE OF ENTITLEMENT START 24"  
BDOE\_START25="SSDI DATE OF ENTITLEMENT START 25"  
BDOE\_START3="SSDI DATE OF ENTITLEMENT START 3"  
BDOE\_START4="SSDI DATE OF ENTITLEMENT START 4"  
BDOE\_START5="SSDI DATE OF ENTITLEMENT START 5"  
BDOE\_START6="SSDI DATE OF ENTITLEMENT START 6"  
BDOE\_START7="SSDI DATE OF ENTITLEMENT START 7"  
BDOE\_START8="SSDI DATE OF ENTITLEMENT START 8"  
BDOE\_START9="SSDI DATE OF ENTITLEMENT START 9"  
BDOE\_TERM1="SSDI DATE OF ENTITLEMENT TERMINATION 1"  
BDOE\_TERM10="SSDI DATE OF ENTITLEMENT TERMINATION 10"  
BDOE\_TERM11="SSDI DATE OF ENTITLEMENT TERMINATION 11"  
BDOE\_TERM12="SSDI DATE OF ENTITLEMENT TERMINATION 12"  
BDOE\_TERM13="SSDI DATE OF ENTITLEMENT TERMINATION 13"  
BDOE\_TERM14="SSDI DATE OF ENTITLEMENT TERMINATION 14"  
BDOE\_TERM15="SSDI DATE OF ENTITLEMENT TERMINATION 15"  
BDOE\_TERM16="SSDI DATE OF ENTITLEMENT TERMINATION 16"  
BDOE\_TERM17="SSDI DATE OF ENTITLEMENT TERMINATION 17"  
BDOE\_TERM18="SSDI DATE OF ENTITLEMENT TERMINATION 18"  
BDOE\_TERM19="SSDI DATE OF ENTITLEMENT TERMINATION 19"  
BDOE\_TERM2="SSDI DATE OF ENTITLEMENT TERMINATION 2"  
BDOE\_TERM20="SSDI DATE OF ENTITLEMENT TERMINATION 20"  
BDOE\_TERM21="SSDI DATE OF ENTITLEMENT TERMINATION 21"  
BDOE\_TERM22="SSDI DATE OF ENTITLEMENT TERMINATION 22"  
BDOE\_TERM23="SSDI DATE OF ENTITLEMENT TERMINATION 23"  
BDOE\_TERM24="SSDI DATE OF ENTITLEMENT TERMINATION 24"

Appendix A.78

JCL/SAS Code: DEMOLBLS

BDOE\_TERM25="SSDI DATE OF ENTITLEMENT TERMINATION 25"  
BDOE\_TERM3="SSDI DATE OF ENTITLEMENT TERMINATION 3"  
BDOE\_TERM4="SSDI DATE OF ENTITLEMENT TERMINATION 4"  
BDOE\_TERM5="SSDI DATE OF ENTITLEMENT TERMINATION 5"  
BDOE\_TERM6="SSDI DATE OF ENTITLEMENT TERMINATION 6"  
BDOE\_TERM7="SSDI DATE OF ENTITLEMENT TERMINATION 7"  
BDOE\_TERM8="SSDI DATE OF ENTITLEMENT TERMINATION 8"  
BDOE\_TERM9="SSDI DATE OF ENTITLEMENT TERMINATION 9"  
BDOF1="SSDI DATE OF FILING ENTRY 1"  
BDOF10="SSDI DATE OF FILING ENTRY 10"  
BDOF11="SSDI DATE OF FILING ENTRY 11"  
BDOF12="SSDI DATE OF FILING ENTRY 12"  
BDOF13="SSDI DATE OF FILING ENTRY 13"  
BDOF14="SSDI DATE OF FILING ENTRY 14"  
BDOF15="SSDI DATE OF FILING ENTRY 15"  
BDOF16="SSDI DATE OF FILING ENTRY 16"  
BDOF17="SSDI DATE OF FILING ENTRY 17"  
BDOF18="SSDI DATE OF FILING ENTRY 18"  
BDOF19="SSDI DATE OF FILING ENTRY 19"  
BDOF2="SSDI DATE OF FILING ENTRY 2"  
BDOF20="SSDI DATE OF FILING ENTRY 20"  
BDOF21="SSDI DATE OF FILING ENTRY 21"  
BDOF22="SSDI DATE OF FILING ENTRY 22"  
BDOF23="SSDI DATE OF FILING ENTRY 23"  
BDOF24="SSDI DATE OF FILING ENTRY 24"  
BDOF25="SSDI DATE OF FILING ENTRY 25"  
BDOF3="SSDI DATE OF FILING ENTRY 3"  
BDOF4="SSDI DATE OF FILING ENTRY 4"  
BDOF5="SSDI DATE OF FILING ENTRY 5"  
BDOF6="SSDI DATE OF FILING ENTRY 6"  
BDOF7="SSDI DATE OF FILING ENTRY 7"  
BDOF8="SSDI DATE OF FILING ENTRY 8"  
BDOF9="SSDI DATE OF FILING ENTRY 9"  
BFD1="BASIS FOR DETERMINATION 1"  
BFD10="BASIS FOR DETERMINATION 10"  
BFD11="BASIS FOR DETERMINATION 11"  
BFD12="BASIS FOR DETERMINATION 12"  
BFD13="BASIS FOR DETERMINATION 13"  
BFD14="BASIS FOR DETERMINATION 14"  
BFD15="BASIS FOR DETERMINATION 15"  
BFD16="BASIS FOR DETERMINATION 16"  
BFD2="BASIS FOR DETERMINATION 2"  
BFD3="BASIS FOR DETERMINATION 3"  
BFD4="BASIS FOR DETERMINATION 4"  
BFD5="BASIS FOR DETERMINATION 5"

Appendix A.78

JCL/SAS Code: DEMOLBLS

BFD6="BASIS FOR DETERMINATION 6"  
BFD7="BASIS FOR DETERMINATION 7"  
BFD8="BASIS FOR DETERMINATION 8"  
BFD9="BASIS FOR DETERMINATION 9"  
BENE\_FRSTNAME\_MBR="SSDI BENEFICIARY GIVEN NAME"  
BIC="SSDI BENEFICIARY IDENTIFICATION CODE (BIC)"  
BIC\_HIST1="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 1"  
BIC\_HIST2="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 2"  
BIC\_HIST3="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 3"  
BIC\_HIST4="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 4"  
BIC\_HIST5="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 5"  
BIC\_HIST6="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 6"  
BIC\_HIST7="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 7"  
BIC\_HIST8="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 8"  
BIC\_HIST9="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 9"  
BIC\_HIST10="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 10"  
BIC\_HIST11="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 11"  
BIC\_HIST12="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 12"  
BIC\_HIST13="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 13"  
BIC\_HIST14="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 14"  
BIC\_HIST15="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 15"  
BIC\_HIST16="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 16"  
BIC\_HIST17="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 17"  
BIC\_HIST18="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 18"  
BIC\_HIST19="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 19"  
BIC\_HIST20="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 20"  
BIC\_HIST21="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 21"  
BIC\_HIST22="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 22"  
BIC\_HIST23="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 23"  
BIC\_HIST24="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 24"  
BIC\_HIST25="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 25"  
BIC\_HIST26="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 26"  
BIC\_HIST27="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 27"  
BIC\_HIST28="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 28"  
BIC\_HIST29="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 29"  
BIC\_HIST30="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 30"  
BIC\_HIST31="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 31"  
BIC\_HIST32="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 32"  
BIC\_HIST33="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 33"  
BIC\_HIST34="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 34"  
BIC\_HIST35="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 35"  
BIC\_HIST36="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 36"  
BIC\_HIST37="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 37"  
BIC\_HIST38="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 38"  
BIC\_HIST39="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 39"

Appendix A.78

JCL/SAS Code: DEMOLBLS

BIC\_HIST40="SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 40"  
BIC\_HIST\_NUM=  
"SSDI # OF HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) OCCURRENCES"  
BIC\_HIST\_START1=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 1"  
BIC\_HIST\_START2=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 2"  
BIC\_HIST\_START3=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 3"  
BIC\_HIST\_START4=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 4"  
BIC\_HIST\_START5=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 5"  
BIC\_HIST\_START6=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 6"  
BIC\_HIST\_START7=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 7"  
BIC\_HIST\_START8=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 8"  
BIC\_HIST\_START9=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 9"  
BIC\_HIST\_START10=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 10"  
BIC\_HIST\_START11=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 11"  
BIC\_HIST\_START12=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 12"  
BIC\_HIST\_START13=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 13"  
BIC\_HIST\_START14=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 14"  
BIC\_HIST\_START15=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 15"  
BIC\_HIST\_START16=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 16"  
BIC\_HIST\_START17=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 17"  
BIC\_HIST\_START18=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 18"  
BIC\_HIST\_START19=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 19"  
BIC\_HIST\_START20=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 20"  
BIC\_HIST\_START21=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 21"

Appendix A.78

JCL/SAS Code: DEMOLBLS

BIC\_HIST\_START22=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 22"  
BIC\_HIST\_START23=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 23"  
BIC\_HIST\_START24=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 24"  
BIC\_HIST\_START25=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 25"  
BIC\_HIST\_START26=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 26"  
BIC\_HIST\_START27=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 27"  
BIC\_HIST\_START28=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 28"  
BIC\_HIST\_START29=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 29"  
BIC\_HIST\_START30=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 30"  
BIC\_HIST\_START31=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 31"  
BIC\_HIST\_START32=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 32"  
BIC\_HIST\_START33=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 33"  
BIC\_HIST\_START34=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 34"  
BIC\_HIST\_START35=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 35"  
BIC\_HIST\_START36=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 36"  
BIC\_HIST\_START37=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 37"  
BIC\_HIST\_START38=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 38"  
BIC\_HIST\_START39=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 39"  
BIC\_HIST\_START40=  
"SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 40"  
BLINDDT="DATE OF BLINDNESS ONSET"  
BENE\_LASTNAME\_MBR="SSDI BENEFICIARY LAST NAME"  
BMI="SSDI BENEFICIARY MIDDLE INITIAL"  
CAN="SSDI CLAIM ACCOUNT NUMBER"  
DCR\_SSDI\_DUAL=  
"SSDI DUAL ELIG- DISABILITY CESSATION REASON-SUPPLEMENTARY RECORD"  
DCR\_SSDI1="SSDI DISABILITY CESSATION REASON 1"

Appendix A.78

JCL/SAS Code: DEMOLBLS

DCR\_SSDI2="SSDI DISABILITY CESSATION REASON 2"  
DCR\_SSDI3="SSDI DISABILITY CESSATION REASON 3"  
DCR\_SSDI4="SSDI DISABILITY CESSATION REASON 4"  
DCR\_SSDI5="SSDI DISABILITY CESSATION REASON 5"  
DCR\_SSDI6="SSDI DISABILITY CESSATION REASON 6"  
DCR\_SSDI7="SSDI DISABILITY CESSATION REASON 7"  
DCR\_SSDI8="SSDI DISABILITY CESSATION REASON 8"  
DCR\_SSDI9="SSDI DISABILITY CESSATION REASON 9"  
DCR\_SSDI10="SSDI DISABILITY CESSATION REASON 10"  
DCR\_SSDI11="SSDI DISABILITY CESSATION REASON 11"  
DCR\_SSDI12="SSDI DISABILITY CESSATION REASON 12"  
CITIZEN\_CER="SSI US CITIZENSHIP INDICATOR FROM THE CER"  
CITIZEN\_NUMI="US CITIZENSHIP INDICATOR FROM THE NUMIDENT"  
CLM\_FIL="SSI CLAIM FILED DATE"  
COMP\_STA="SSI STATUS TYPE OF ACTION"  
COUNTY\_RES\_MBR="SSDI SSA COUNTY CODE OF RESIDENCE"  
COUNTRY\_OF\_BIRTH="COUNTRY OF BIRTH"  
CURSTAT="SSI MOST RECENT PAYMENT STATUS"  
ACD\_SSDI\_DUAL=  
    "SSDI DUAL ELIG-DISABILITY AWARD CODE-SUPPLEMENTARY RECORD"  
ACD\_SSDI1="SSDI DISABILITY AWARD CODE 1"  
ACD\_SSDI2="SSDI DISABILITY AWARD CODE 2"  
ACD\_SSDI3="SSDI DISABILITY AWARD CODE 3"  
ACD\_SSDI4="SSDI DISABILITY AWARD CODE 4"  
ACD\_SSDI5="SSDI DISABILITY AWARD CODE 5"  
ACD\_SSDI6="SSDI DISABILITY AWARD CODE 6"  
ACD\_SSDI7="SSDI DISABILITY AWARD CODE 7"  
ACD\_SSDI8="SSDI DISABILITY AWARD CODE 8"  
ACD\_SSDI9="SSDI DISABILITY AWARD CODE 9"  
ACD\_SSDI10="SSDI DISABILITY AWARD CODE 10"  
ACD\_SSDI11="SSDI DISABILITY AWARD CODE 11"  
ACD\_SSDI12="SSDI DISABILITY AWARD CODE 12"  
DDBC\_DUAL=  
    "SSDI DUAL ELIG-DATE OF BENEFIT CESSATION-SUPPLEMENTARY RECORD"  
DDBC1="SSDI DISABILITY CESSATION DATE 1"  
DDBC10="SSDI DISABILITY CESSATION DATE 10"  
DDBC11="SSDI DISABILITY CESSATION DATE 11"  
DDBC12="SSDI DISABILITY CESSATION DATE 12"  
DDBC2="SSDI DISABILITY CESSATION DATE 2"  
DDBC3="SSDI DISABILITY CESSATION DATE 3"  
DDBC4="SSDI DISABILITY CESSATION DATE 4"  
DDBC5="SSDI DISABILITY CESSATION DATE 5"  
DDBC6="SSDI DISABILITY CESSATION DATE 6"  
DDBC7="SSDI DISABILITY CESSATION DATE 7"  
DDBC8="SSDI DISABILITY CESSATION DATE 8"

Appendix A.78

JCL/SAS Code: DEMOLBLS

DDBC9="SSDI DISABILITY CESSATION DATE 9"  
DDO\_DUAL="SSDI DUAL ELIG-DISABILITY ONSET DATE-SUPPLEMENTARY RECORD"  
DDO1="SSDI DISABILITY ONSET DATE 1"  
DDO10="SSDI DISABILITY ONSET DATE 10"  
DDO11="SSDI DISABILITY ONSET DATE 11"  
DDO12="SSDI DISABILITY ONSET DATE 12"  
DDO2="SSDI DISABILITY ONSET DATE 2"  
DDO3="SSDI DISABILITY ONSET DATE 3"  
DDO4="SSDI DISABILITY ONSET DATE 4"  
DDO5="SSDI DISABILITY ONSET DATE 5"  
DDO6="SSDI DISABILITY ONSET DATE 6"  
DDO7="SSDI DISABILITY ONSET DATE 7"  
DDO8="SSDI DISABILITY ONSET DATE 8"  
DDO9="SSDI DISABILITY ONSET DATE 9"  
DD1="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 1"  
DD10="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 10"  
DD11="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 11"  
DD12="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 12"  
DD13="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 13"  
DD14="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 14"  
DD15="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 15"  
DD16="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 16"  
DD2="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 2"  
DD3="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 3"  
DD4="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 4"  
DD5="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 5"  
DD6="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 6"  
DD7="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 7"  
DD8="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 8"  
DD9="MEDICAL CONTINUING DISABILITY REVIEW DIARY DATE 9"  
DODSSR="SSI DATE OF DEATH - FROM SSR"  
DIB\_DPM="SSI PERMANENT DISABILITY INDICATOR"  
DIB\_MDR="SSI MEDICAL DIARY REASON"  
DIBHAS1="DIB QUARTERS OF COVERAGE POSSESSED 1"  
DIBHAS10="DIB QUARTERS OF COVERAGE POSSESSED 10"  
DIBHAS11="DIB QUARTERS OF COVERAGE POSSESSED 11"  
DIBHAS12="DIB QUARTERS OF COVERAGE POSSESSED 12"  
DIBHAS13="DIB QUARTERS OF COVERAGE POSSESSED 13"  
DIBHAS14="DIB QUARTERS OF COVERAGE POSSESSED 14"  
DIBHAS15="DIB QUARTERS OF COVERAGE POSSESSED 15"  
DIBHAS2="DIB QUARTERS OF COVERAGE POSSESSED 2"  
DIBHAS3="DIB QUARTERS OF COVERAGE POSSESSED 3"  
DIBHAS4="DIB QUARTERS OF COVERAGE POSSESSED 4"  
DIBHAS5="DIB QUARTERS OF COVERAGE POSSESSED 5"  
DIBHAS6="DIB QUARTERS OF COVERAGE POSSESSED 6"

Appendix A.78

JCL/SAS Code: DEMOLBLS

DIBHAS7="DIB QUARTERS OF COVERAGE POSSESSED 7"  
DIBHAS8="DIB QUARTERS OF COVERAGE POSSESSED 8"  
DIBHAS9="DIB QUARTERS OF COVERAGE POSSESSED 9"  
DIBREQ1="DIB QUARTERS OF COVERAGE REQUIRED 1"  
DIBREQ10="DIB QUARTERS OF COVERAGE REQUIRED 10"  
DIBREQ11="DIB QUARTERS OF COVERAGE REQUIRED 11"  
DIBREQ12="DIB QUARTERS OF COVERAGE REQUIRED 12"  
DIBREQ13="DIB QUARTERS OF COVERAGE REQUIRED 13"  
DIBREQ14="DIB QUARTERS OF COVERAGE REQUIRED 14"  
DIBREQ15="DIB QUARTERS OF COVERAGE REQUIRED 15"  
DIBREQ2="DIB QUARTERS OF COVERAGE REQUIRED 2"  
DIBREQ3="DIB QUARTERS OF COVERAGE REQUIRED 3"  
DIBREQ4="DIB QUARTERS OF COVERAGE REQUIRED 4"  
DIBREQ5="DIB QUARTERS OF COVERAGE REQUIRED 5"  
DIBREQ6="DIB QUARTERS OF COVERAGE REQUIRED 6"  
DIBREQ7="DIB QUARTERS OF COVERAGE REQUIRED 7"  
DIBREQ8="DIB QUARTERS OF COVERAGE REQUIRED 8"  
DIBREQ9="DIB QUARTERS OF COVERAGE REQUIRED 9"  
DXPRI\_MBR\_MR="SSDI PRIMARY DIAGNOSIS CODE, MOST RECENT"  
DISPAYCD="SSI DISABILITY PAYMENT CODE"  
DOBBEST="BEST DATE OF BIRTH"  
DOBFLAG="JANUARY 1 IMPUTED FOR BEST DATE OF BIRTH"  
DOBSSR="DATE OF BIRTH - SSR"  
DOBMBR="DATE OF BIRTH - MBR"  
DOBNUMI="DATE OF BIRTH - NUMIDENT"  
DOBCER="DATE OF BIRTH - CER100% FIELD FILE"  
DOC\_SSDI\_MR="SSDI DISTRICT OFFICE CODE, MOST RECENT"  
DOC\_SSI\_MR="SSI DISTRICT OFFICE CODE, MOST RECENT"  
DOC1="DISTRICT OFFICE CODE 1"  
DOC10="DISTRICT OFFICE CODE 10"  
DOC11="DISTRICT OFFICE CODE 11"  
DOC12="DISTRICT OFFICE CODE 12"  
DOC13="DISTRICT OFFICE CODE 13"  
DOC14="DISTRICT OFFICE CODE 14"  
DOC15="DISTRICT OFFICE CODE 15"  
DOC16="DISTRICT OFFICE CODE 16"  
DOC2="DISTRICT OFFICE CODE 2"  
DOC3="DISTRICT OFFICE CODE 3"  
DOC4="DISTRICT OFFICE CODE 4"  
DOC5="DISTRICT OFFICE CODE 5"  
DOC6="DISTRICT OFFICE CODE 6"  
DOC7="DISTRICT OFFICE CODE 7"  
DOC8="DISTRICT OFFICE CODE 8"  
DOC9="DISTRICT OFFICE CODE 9"  
DODNUMI="DATE OF DEATH - NUMIDENT"



Appendix A.78

JCL/SAS Code: DEMOLBLS

DODBEST="BEST DATE OF DEATH"  
DODEC1="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 1"  
DODEC10="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 10"  
DODEC11="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 11"  
DODEC12="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 12"  
DODEC13="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 13"  
DODEC14="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 14"  
DODEC15="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 15"  
DODEC16="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 16"  
DODEC2="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 2"  
DODEC3="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 3"  
DODEC4="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 4"  
DODEC5="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 5"  
DODEC6="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 6"  
DODEC7="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 7"  
DODEC8="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 8"  
DODEC9="DATE OF MEDICAL DECISION (SSA OR STATE DDS) 9"  
DODFLAG="JANUARY 1 IMPUTED FOR BEST DATE OF DEATH"  
DOEC="SSDI MOST RECENT ENTITLEMENT DATE"  
DOEC\_DOEI\_SAME=  
"HISTORICAL TOC/BIC BUILT FROM RECORD WITH SAME DOEC AND DOEI"  
DOEC\_SET\_TO\_DOEI=  
"HISTORICAL TOC/BIC BUILT FROM RECORD WHERE A MISSING DOEC WAS SET TO  
THE DOEI"  
DOECD=  
"SSDI DUAL ELIG-SSDI MOST RECENT ENTITLEMENT DATE-SUPPLEMENTARY RECORD"  
DOEI="SSDI INITIAL ENTITLEMENT DATE"  
DOEID=  
"SSDI DUAL ELIG-SSDI INITIAL ENTITLEMENT DATE-SUPPLEMENTARY RECORD"  
DOST="SSDI SUSPENSION OR TERMINATION DATE"  
DPI\_START1="SSDI DEMONSTRATION PROJECT START DATE 1"  
DPI\_START10="SSDI DEMONSTRATION PROJECT START DATE 10"  
DPI\_START2="SSDI DEMONSTRATION PROJECT START DATE 2"  
DPI\_START3="SSDI DEMONSTRATION PROJECT START DATE 3"  
DPI\_START4="SSDI DEMONSTRATION PROJECT START DATE 4"  
DPI\_START5="SSDI DEMONSTRATION PROJECT START DATE 5"  
DPI\_START6="SSDI DEMONSTRATION PROJECT START DATE 6"  
DPI\_START7="SSDI DEMONSTRATION PROJECT START DATE 7"  
DPI\_START8="SSDI DEMONSTRATION PROJECT START DATE 8"  
DPI\_START9="SSDI DEMONSTRATION PROJECT START DATE 9"  
DPI\_STOP1="SSDI DEMONSTRATION PROJECT END DATE 1"  
DPI\_STOP10="SSDI DEMONSTRATION PROJECT END DATE 10"  
DPI\_STOP2="SSDI DEMONSTRATION PROJECT END DATE 2"  
DPI\_STOP3="SSDI DEMONSTRATION PROJECT END DATE 3"  
DPI\_STOP4="SSDI DEMONSTRATION PROJECT END DATE 4"

Appendix A.78

JCL/SAS Code: DEMOLBLS

DPI\_STOP5="SSDI DEMONSTRATION PROJECT END DATE 5"  
DPI\_STOP6="SSDI DEMONSTRATION PROJECT END DATE 6"  
DPI\_STOP7="SSDI DEMONSTRATION PROJECT END DATE 7"  
DPI\_STOP8="SSDI DEMONSTRATION PROJECT END DATE 8"  
DPI\_STOP9="SSDI DEMONSTRATION PROJECT END DATE 9"  
DPI1="SSDI DEMONSTRATION PROJECT CODE 1"  
DPI10="SSDI DEMONSTRATION PROJECT CODE 10"  
DPI2="SSDI DEMONSTRATION PROJECT CODE 2"  
DPI3="SSDI DEMONSTRATION PROJECT CODE 3"  
DPI4="SSDI DEMONSTRATION PROJECT CODE 4"  
DPI5="SSDI DEMONSTRATION PROJECT CODE 5"  
DPI6="SSDI DEMONSTRATION PROJECT CODE 6"  
DPI7="SSDI DEMONSTRATION PROJECT CODE 7"  
DPI8="SSDI DEMONSTRATION PROJECT CODE 8"  
DPI9="SSDI DEMONSTRATION PROJECT CODE 9"  
DPM1="PERMANENT DISABILITY CODE ENTRY 1"  
DPM10="PERMANENT DISABILITY CODE ENTRY 10"  
DPM11="PERMANENT DISABILITY CODE ENTRY 11"  
DPM12="PERMANENT DISABILITY CODE ENTRY 12"  
DPM13="PERMANENT DISABILITY CODE ENTRY 13"  
DPM14="PERMANENT DISABILITY CODE ENTRY 14"  
DPM15="PERMANENT DISABILITY CODE ENTRY 15"  
DPM16="PERMANENT DISABILITY CODE ENTRY 16"  
DPM2="PERMANENT DISABILITY CODE ENTRY 2"  
DPM3="PERMANENT DISABILITY CODE ENTRY 3"  
DPM4="PERMANENT DISABILITY CODE ENTRY 4"  
DPM5="PERMANENT DISABILITY CODE ENTRY 5"  
DPM6="PERMANENT DISABILITY CODE ENTRY 6"  
DPM7="PERMANENT DISABILITY CODE ENTRY 7"  
DPM8="PERMANENT DISABILITY CODE ENTRY 8"  
DPM9="PERMANENT DISABILITY CODE ENTRY 9"  
DR1="MEDICAL CONTINUING DISABILITY REVIEW REASON 1"  
DR10="MEDICAL CONTINUING DISABILITY REVIEW REASON 10"  
DR11="MEDICAL CONTINUING DISABILITY REVIEW REASON 11"  
DR12="MEDICAL CONTINUING DISABILITY REVIEW REASON 12"  
DR13="MEDICAL CONTINUING DISABILITY REVIEW REASON 13"  
DR14="MEDICAL CONTINUING DISABILITY REVIEW REASON 14"  
DR15="MEDICAL CONTINUING DISABILITY REVIEW REASON 15"  
DR16="MEDICAL CONTINUING DISABILITY REVIEW REASON 16"  
DR2="MEDICAL CONTINUING DISABILITY REVIEW REASON 2"  
DR3="MEDICAL CONTINUING DISABILITY REVIEW REASON 3"  
DR4="MEDICAL CONTINUING DISABILITY REVIEW REASON 4"  
DR5="MEDICAL CONTINUING DISABILITY REVIEW REASON 5"  
DR6="MEDICAL CONTINUING DISABILITY REVIEW REASON 6"  
DR7="MEDICAL CONTINUING DISABILITY REVIEW REASON 7"

Appendix A.78

JCL/SAS Code: DEMOLBLS

DR8="MEDICAL CONTINUING DISABILITY REVIEW REASON 8"  
DR9="MEDICAL CONTINUING DISABILITY REVIEW REASON 9"  
DSD\_DUAL=  
"SSDI DUAL ELIG-SSDI DISABILITY ADJUDICATION DATE-SUPPLEMENTARY RECORD"  
DSD1="SSDI DISABILITY ADJUDICATION DATE 1"  
DSD10="SSDI DISABILITY ADJUDICATION DATE 10"  
DSD11="SSDI DISABILITY ADJUDICATION DATE 11"  
DSD12="SSDI DISABILITY ADJUDICATION DATE 12"  
DSD2="SSDI DISABILITY ADJUDICATION DATE 2"  
DSD3="SSDI DISABILITY ADJUDICATION DATE 3"  
DSD4="SSDI DISABILITY ADJUDICATION DATE 4"  
DSD5="SSDI DISABILITY ADJUDICATION DATE 5"  
DSD6="SSDI DISABILITY ADJUDICATION DATE 6"  
DSD7="SSDI DISABILITY ADJUDICATION DATE 7"  
DSD8="SSDI DISABILITY ADJUDICATION DATE 8"  
DSD9="SSDI DISABILITY ADJUDICATION DATE 9"  
DSICLAIM1="DISABILITY SERVICES IMPROVEMENT CLAIM 1"  
DSICLAIM2="DISABILITY SERVICES IMPROVEMENT CLAIM 2"  
DSICLAIM3="DISABILITY SERVICES IMPROVEMENT CLAIM 3"  
DSICLAIM4="DISABILITY SERVICES IMPROVEMENT CLAIM 4"  
DSICLAIM5="DISABILITY SERVICES IMPROVEMENT CLAIM 5"  
DSICLAIM6="DISABILITY SERVICES IMPROVEMENT CLAIM 6"  
DSICLAIM7="DISABILITY SERVICES IMPROVEMENT CLAIM 7"  
DSICLAIM8="DISABILITY SERVICES IMPROVEMENT CLAIM 8"  
DSICLAIM9="DISABILITY SERVICES IMPROVEMENT CLAIM 9"  
DSICLAIM10="DISABILITY SERVICES IMPROVEMENT CLAIM 10"  
DSICLAIM11="DISABILITY SERVICES IMPROVEMENT CLAIM 11"  
DSICLAIM12="DISABILITY SERVICES IMPROVEMENT CLAIM 12"  
DSICLAIM13="DISABILITY SERVICES IMPROVEMENT CLAIM 13"  
DSICLAIM14="DISABILITY SERVICES IMPROVEMENT CLAIM 14"  
DSICLAIM15="DISABILITY SERVICES IMPROVEMENT CLAIM 15"  
DSICLAIM16="DISABILITY SERVICES IMPROVEMENT CLAIM 16"  
DT1="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 1"  
DT10="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 10"  
DT11="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 11"  
DT12="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 12"  
DT13="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 13"  
DT14="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 14"  
DT15="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 15"  
DT16="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 16"  
DT2="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 2"  
DT3="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 3"  
DT4="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 4"  
DT5="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 5"  
DT6="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 6"

Appendix A.78

JCL/SAS Code: DEMOLBLS

DT7="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 7"  
DT8="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 8"  
DT9="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE 9"  
DUALELIG="SSDI DUAL ELIG-BENEFICIARY FLAG"  
DXPRI\_DI=  
"SSDI PRIMARY DIAGNOSIS CODE MOST RECENT, BEST FROM ALL SOURCES"  
DXPRI\_DUAL="SSDI DUAL ELIG-PRIMARY DIAGNOSIS-SUPPLEMENTARY RECORD"  
DXPRI\_MBR1="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 1"  
DXPRI\_MBR10="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 10"  
DXPRI\_MBR11="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 11"  
DXPRI\_MBR12="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 12"  
DXPRI\_MBR2="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 2"  
DXPRI\_MBR3="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 3"  
DXPRI\_MBR4="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 4"  
DXPRI\_MBR5="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 5"  
DXPRI\_MBR6="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 6"  
DXPRI\_MBR7="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 7"  
DXPRI\_MBR8="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 8"  
DXPRI\_MBR9="SSDI PRIMARY DIAGNOSIS CODE FROM MBR 9"  
DXPRI\_SSI="SSI PRIMARY DIAGNOSIS CODE BEST FROM ALL SOURCES"  
DXPRIBEST="PRIMARY DIAGNOSIS CODE MOST RECENT, BEST FROM ALL SOURCES"  
DXSEC\_DI=  
"SSDI SECONDARY DIAGNOSIS CODE MOST RECENT, BEST FROM ALL SOURCES"  
DXSEC\_DUAL="SSDI DUAL ELIG-SECONDARY DIAGNOSIS-SUPPLEMENTARY RECORD"  
DXSEC\_MBR1="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 1"  
DXSEC\_MBR10="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 10"  
DXSEC\_MBR11="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 11"  
DXSEC\_MBR12="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 12"  
DXSEC\_MBR2="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 2"  
DXSEC\_MBR3="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 3"  
DXSEC\_MBR4="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 4"  
DXSEC\_MBR5="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 5"  
DXSEC\_MBR6="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 6"  
DXSEC\_MBR7="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 7"  
DXSEC\_MBR8="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 8"  
DXSEC\_MBR9="SSDI SECONDARY DIAGNOSIS CODE FROM MBR 9"  
DXSEC\_SSI=  
"SSI SECONDARY DIAGNOSIS CODE MOST RECENT, BEST FROM ALL SOURCES"  
DXSECBEST="SECONDARY DIAGNOSIS CODE MOST RECENT, BEST FROM ALL SOURCES"  
EBD\_DUAL="SSDI DUAL ELIG-EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE"  
EBD1="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 1"  
EBD10="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 10"  
EBD11="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 11"  
EBD12="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 12"  
EBD2="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 2"

Appendix A.78

JCL/SAS Code: DEMOLBLS

EBD3="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 3"  
EBD4="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 4"  
EBD5="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 5"  
EBD6="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 6"  
EBD7="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 7"  
EBD8="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 8"  
EBD9="EXTENDED PERIOD OF ELIGIBILITY BEGIN DATE 9"  
ELG\_RD="SSI DATE OF MOST RECENT ELIGIBILITY"  
ENTDAT\_DUAL="SSDI DUAL ELIG-SSDI ENTITLEMENT DATE"  
ENTDAT1="SSDI ENTITLEMENT DATE 1"  
ENTDAT10="SSDI ENTITLEMENT DATE 10"  
ENTDAT11="SSDI ENTITLEMENT DATE 11"  
ENTDAT12="SSDI ENTITLEMENT DATE 12"  
ENTDAT2="SSDI ENTITLEMENT DATE 2"  
ENTDAT3="SSDI ENTITLEMENT DATE 3"  
ENTDAT4="SSDI ENTITLEMENT DATE 4"  
ENTDAT5="SSDI ENTITLEMENT DATE 5"  
ENTDAT6="SSDI ENTITLEMENT DATE 6"  
ENTDAT7="SSDI ENTITLEMENT DATE 7"  
ENTDAT8="SSDI ENTITLEMENT DATE 8"  
ENTDAT9="SSDI ENTITLEMENT DATE 9"  
EPEDATE1="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 1"  
EPEDATE10="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 10"  
EPEDATE11="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 11"  
EPEDATE12="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 12"  
EPEDATE13="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 13"  
EPEDATE14="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 14"  
EPEDATE15="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 15"  
EPEDATE16="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 16"  
EPEDATE2="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 2"  
EPEDATE3="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 3"  
EPEDATE4="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 4"  
EPEDATE5="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 5"  
EPEDATE6="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 6"  
EPEDATE7="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 7"  
EPEDATE8="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 8"  
EPEDATE9="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE 9"  
EPEDENY1=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 1"  
EPEDENY10=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 10"  
EPEDENY11=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 11"  
EPEDENY12=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 12"

Appendix A.78

JCL/SAS Code: DEMOLBLS

EPEDENY13=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 13"  
EPEDENY14=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 14"  
EPEDENY15=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 15"  
EPEDENY16=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 16"  
EPEDENY2=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 2"  
EPEDENY3=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 3"  
EPEDENY4=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 4"  
EPEDENY5=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 5"  
EPEDENY6=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 6"  
EPEDENY7=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 7"  
EPEDENY8=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 8"  
EPEDENY9=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENIED FLAG 9"  
EPEREIN1=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 1"  
EPEREIN10=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 10"  
EPEREIN11=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 11"  
EPEREIN12=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 12"  
EPEREIN13=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 13"  
EPEREIN14=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 14"  
EPEREIN15=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 15"  
EPEREIN16=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 16"  
EPEREIN2=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 2"  
EPEREIN3=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 3"  
EPEREIN4=

Appendix A.78

JCL/SAS Code: DEMOLBLS

"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 4"  
EPEREIN5=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 5"  
EPEREIN6=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 6"  
EPEREIN7=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 7"  
EPEREIN8=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 8"  
EPEREIN9=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOWED DATE 9"  
EPESUSP1=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 1"  
EPESUSP10=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 10"  
EPESUSP11=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 11"  
EPESUSP12=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 12"  
EPESUSP13=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 13"  
EPESUSP14=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 14"  
EPESUSP15=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 15"  
EPESUSP16=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 16"  
EPESUSP2=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 2"  
EPESUSP3=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 3"  
EPESUSP4=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 4"  
EPESUSP5=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 5"  
EPESUSP6=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 6"  
EPESUSP7=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 7"  
EPESUSP8=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 8"  
EPESUSP9=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSPENDED AFTER REINSTATE DATE 9"  
EPETERM1=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 1"

Appendix A.78

JCL/SAS Code: DEMOLBLS

EPETERM10=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 10"  
EPETERM11=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 11"  
EPETERM12=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 12"  
EPETERM13=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 13"  
EPETERM14=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 14"  
EPETERM15=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 15"  
EPETERM16=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 16"  
EPETERM2=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 2"  
EPETERM3=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 3"  
EPETERM4=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 4"  
EPETERM5=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 5"  
EPETERM6=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 6"  
EPETERM7=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 7"  
EPETERM8=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 8"  
EPETERM9=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE 9"  
FIRSTBFD="BASIS FOR DETERMINATION, EARLIEST"  
FIRSTDD="MEDICAL CONTINUING DISABILITY REVIEW DATE, EARLIEST"  
FIRSTDOC="DISTRICT OFFICE CODE, EARLIEST"  
FIRSTDODEC="DATE OF MEDICAL DECISION (SSA OR DDS), EARLIEST"  
FIRSTDPM="PERMANENT DISABILITY CODE ENTRY, EARLIEST"  
FIRSTDR="MEDICAL CONTINUING DISABILITY REVIEW REASON, EARLIEST"  
FIRSTDSICLAIM="DISABILITY SERVICES IMPROVEMENT CLAIM, EARLIEST"  
FIRSTDT="MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE, EARLIEST"  
FIRSTEPEDATE="SSDI EXTENDED PERIOD OF ELIGIBILITY START DATE, EARLIEST"  
FIRSTEPEDENY=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT DENY FLAG, EARLIEST"  
FIRSTEPEREIN=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY REINSTATEMENT ALLOW DATE, EARLIEST"  
FIRSTEPESUSP=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY SUSP. AFTER REINS. DATE, EARLIEST"



Appendix A.78

JCL/SAS Code: DEMOLBLS

FIRSTSTEPETERM=  
"SSDI EXTENDED PERIOD OF ELIGIBILITY BENEFIT TERMINATION DATE, EARLIEST"  
FIRSTIME="SSDI INDEXED MONTHLY EARNINGS FOR EARLIEST PERIOD"  
FIRSTJUDLVL="LEVEL OF ADJUDICATION, EARLIEST"  
FIRSTPD="PRESUMPTIVE DISABILITY DECISION CODE, EARLIEST"  
FIRSTPDD="PRESUMPTIVE DECISION DATE, EARLIEST"  
FIRSTPIA="PRIMARY INSURANCE AMOUNT, EARLIEST"  
FIRSTPIARFC="REASON FOR PRIMARY INSURANCE AMOUNT CHANGE, EARLIEST"  
FIRSTPIED="PRIMARY INSURANCE DATE, EARLIEST"  
FIRSTQDDIND="QUICK DISABILITY DETERMINATION (QDD) INDICATOR, EARLIEST"  
FIRSTQDREINDT="QDD REINSTATEMENT DATE, EARLIEST"  
FIRSTQDRMVDT="QDD REMOVAL DATE, EARLIEST"  
FIRSTQDRMVRSN="QDD REMOVAL REASON, EARLIEST"  
FIRSTQDScore="QDD SCORING CODE, EARLIEST"  
FIRSTRB="REGULATION BASIS CODE, EARLIEST"  
FIRSTRDT="RESULT OF DETERMINATION DATE, EARLIEST"  
FIRSTRFC="REASON FOR CONTINUANCE/CESSATION CODE, EARLIEST"  
FIRSTRID="PROGRAM IDENTIFICATION, EARLIEST"  
FIRSTVRA="VOCATIONAL REHABILITATION REFERRAL CODE, EARLIEST"  
FRA="BENEFICIARY'S FULL RETIREMENT AGE"  
FRADATE = "FULL RETIREMENT DATE"  
FIRSTMIE="DATE MIE FIRST FLAGGED"  
FOREIGN\_BORN="FOREIGN BORN INDICATOR"  
BENE\_FRSTNAME\_NUMI="NUMIDENT BENEFICIARY FIRST NAME"  
HBIC1="HISTORICAL BIC 1"  
HBIC10="HISTORICAL BIC 10"  
HBIC11="HISTORICAL BIC 11"  
HBIC12="HISTORICAL BIC 12"  
HBIC13="HISTORICAL BIC 13"  
HBIC14="HISTORICAL BIC 14"  
HBIC15="HISTORICAL BIC 15"  
HBIC16="HISTORICAL BIC 16"  
HBIC17="HISTORICAL BIC 17"  
HBIC18="HISTORICAL BIC 18"  
HBIC19="HISTORICAL BIC 19"  
HBIC2="HISTORICAL BIC 2"  
HBIC20="HISTORICAL BIC 20"  
HBIC21="HISTORICAL BIC 21"  
HBIC22="HISTORICAL BIC 22"  
HBIC23="HISTORICAL BIC 23"  
HBIC24="HISTORICAL BIC 24"  
HBIC25="HISTORICAL BIC 25"  
HBIC3="HISTORICAL BIC 3"  
HBIC4="HISTORICAL BIC 4"  
HBIC5="HISTORICAL BIC 5"

Appendix A.78

JCL/SAS Code: DEMOLBLS

HBIC6="HISTORICAL BIC 6"  
HBIC7="HISTORICAL BIC 7"  
HBIC8="HISTORICAL BIC 8"  
HBIC9="HISTORICAL BIC 9"  
HDD\_DUAL="SSDI DUAL ELIG-HEARING DECISION DATE"  
HDD1="SSDI HEARING DECISION DATE 1"  
HDD10="SSDI HEARING DECISION DATE 10"  
HDD11="SSDI HEARING DECISION DATE 11"  
HDD12="SSDI HEARING DECISION DATE 12"  
HDD2="SSDI HEARING DECISION DATE 2"  
HDD3="SSDI HEARING DECISION DATE 3"  
HDD4="SSDI HEARING DECISION DATE 4"  
HDD5="SSDI HEARING DECISION DATE 5"  
HDD6="SSDI HEARING DECISION DATE 6"  
HDD7="SSDI HEARING DECISION DATE 7"  
HDD8="SSDI HEARING DECISION DATE 8"  
HDD9="SSDI HEARING DECISION DATE 9"  
HI\_START\_MR=  
"SSDI HOSPITAL INS. (HI) START DATE, MEDICARE PART A MOST RECENT"  
HI\_START1="SSDI HOSPITAL INS. (HI) START DATE, MEDICARE PART A 1"  
HI\_START10="SSDI HOSPITAL INS. (HI) START DATE, MEDICARE PART A 10"  
HI\_START2="SSDI HOSPITAL INS. (HI) START DATE, MEDICARE PART A 2"  
HI\_START3="SSDI HOSPITAL INS. (HI) START DATE, MEDICARE PART A 3"  
HI\_START4="SSDI HOSPITAL INS. (HI) START DATE, MEDICARE PART A 4"  
HI\_START5="SSDI HOSPITAL INS. (HI) START DATE, MEDICARE PART A 5"  
HI\_START6="SSDI HOSPITAL INS. (HI) START DATE, MEDICARE PART A 6"  
HI\_START7="SSDI HOSPITAL INS. (HI) START DATE, MEDICARE PART A 7"  
HI\_START8="SSDI HOSPITAL INS. (HI) START DATE, MEDICARE PART A 8"  
HI\_START9="SSDI HOSPITAL INS. (HI) START DATE, MEDICARE PART A 9"  
HI\_TERM\_MR=  
"SSDI HOSPITAL INS. (HI) END DATE, MEDICARE PART A MOST RECENT"  
HI\_TERM1="SSDI HOSPITAL INS. (HI) END DATE, MEDICARE PART A 1"  
HI\_TERM10="SSDI HOSPITAL INS. (HI) END DATE, MEDICARE PART A 10"  
HI\_TERM2="SSDI HOSPITAL INS. (HI) END DATE, MEDICARE PART A 2"  
HI\_TERM3="SSDI HOSPITAL INS. (HI) END DATE, MEDICARE PART A 3"  
HI\_TERM4="SSDI HOSPITAL INS. (HI) END DATE, MEDICARE PART A 4"  
HI\_TERM5="SSDI HOSPITAL INS. (HI) END DATE, MEDICARE PART A 5"  
HI\_TERM6="SSDI HOSPITAL INS. (HI) END DATE, MEDICARE PART A 6"  
HI\_TERM7="SSDI HOSPITAL INS. (HI) END DATE, MEDICARE PART A 7"  
HI\_TERM8="SSDI HOSPITAL INS. (HI) END DATE, MEDICARE PART A 8"  
HI\_TERM9="SSDI HOSPITAL INS. (HI) END DATE, MEDICARE PART A 9"  
HUN="SSI HOUSED UNDER NUMBER (HUN)"  
IME1="SSDI INDEXED MONTHLY EARNINGS 1"  
IME10="SSDI INDEXED MONTHLY EARNINGS 10"  
IME11="SSDI INDEXED MONTHLY EARNINGS 11"

Appendix A.78

JCL/SAS Code: DEMOLBLS

IME12="SSDI INDEXED MONTHLY EARNINGS 12"  
IME13="SSDI INDEXED MONTHLY EARNINGS 13"  
IME14="SSDI INDEXED MONTHLY EARNINGS 14"  
IME15="SSDI INDEXED MONTHLY EARNINGS 15"  
IME16="SSDI INDEXED MONTHLY EARNINGS 16"  
IME17="SSDI INDEXED MONTHLY EARNINGS 17"  
IME18="SSDI INDEXED MONTHLY EARNINGS 18"  
IME19="SSDI INDEXED MONTHLY EARNINGS 19"  
IME2="SSDI INDEXED MONTHLY EARNINGS 2"  
IME20="SSDI INDEXED MONTHLY EARNINGS 20"  
IME21="SSDI INDEXED MONTHLY EARNINGS 21"  
IME22="SSDI INDEXED MONTHLY EARNINGS 22"  
IME23="SSDI INDEXED MONTHLY EARNINGS 23"  
IME24="SSDI INDEXED MONTHLY EARNINGS 24"  
IME25="SSDI INDEXED MONTHLY EARNINGS 25"  
IME26="SSDI INDEXED MONTHLY EARNINGS 26"  
IME27="SSDI INDEXED MONTHLY EARNINGS 27"  
IME28="SSDI INDEXED MONTHLY EARNINGS 28"  
IME29="SSDI INDEXED MONTHLY EARNINGS 29"  
IME3="SSDI INDEXED MONTHLY EARNINGS 3"  
IME30="SSDI INDEXED MONTHLY EARNINGS 30"  
IME31="SSDI INDEXED MONTHLY EARNINGS 31"  
IME32="SSDI INDEXED MONTHLY EARNINGS 32"  
IME33="SSDI INDEXED MONTHLY EARNINGS 33"  
IME34="SSDI INDEXED MONTHLY EARNINGS 34"  
IME35="SSDI INDEXED MONTHLY EARNINGS 35"  
IME36="SSDI INDEXED MONTHLY EARNINGS 36"  
IME37="SSDI INDEXED MONTHLY EARNINGS 37"  
IME38="SSDI INDEXED MONTHLY EARNINGS 38"  
IME39="SSDI INDEXED MONTHLY EARNINGS 39"  
IME4="SSDI INDEXED MONTHLY EARNINGS 4"  
IME40="SSDI INDEXED MONTHLY EARNINGS 40"  
IME41="SSDI INDEXED MONTHLY EARNINGS 41"  
IME42="SSDI INDEXED MONTHLY EARNINGS 42"  
IME43="SSDI INDEXED MONTHLY EARNINGS 43"  
IME44="SSDI INDEXED MONTHLY EARNINGS 44"  
IME45="SSDI INDEXED MONTHLY EARNINGS 45"  
IME46="SSDI INDEXED MONTHLY EARNINGS 46"  
IME47="SSDI INDEXED MONTHLY EARNINGS 47"  
IME48="SSDI INDEXED MONTHLY EARNINGS 48"  
IME49="SSDI INDEXED MONTHLY EARNINGS 49"  
IME5="SSDI INDEXED MONTHLY EARNINGS 5"  
IME50="SSDI INDEXED MONTHLY EARNINGS 50"  
IME6="SSDI INDEXED MONTHLY EARNINGS 6"  
IME7="SSDI INDEXED MONTHLY EARNINGS 7"

Appendix A.78

JCL/SAS Code: DEMOLBLS

IME8="SSDI INDEXED MONTHLY EARNINGS 8"  
IME9="SSDI INDEXED MONTHLY EARNINGS 9"  
JUDLVL1="LEVEL OF ADJUDICATION 1"  
JUDLVL10="LEVEL OF ADJUDICATION 10"  
JUDLVL11="LEVEL OF ADJUDICATION 11"  
JUDLVL12="LEVEL OF ADJUDICATION 12"  
JUDLVL13="LEVEL OF ADJUDICATION 13"  
JUDLVL14="LEVEL OF ADJUDICATION 14"  
JUDLVL15="LEVEL OF ADJUDICATION 15"  
JUDLVL16="LEVEL OF ADJUDICATION 16"  
JUDLVL2="LEVEL OF ADJUDICATION 2"  
JUDLVL3="LEVEL OF ADJUDICATION 3"  
JUDLVL4="LEVEL OF ADJUDICATION 4"  
JUDLVL5="LEVEL OF ADJUDICATION 5"  
JUDLVL6="LEVEL OF ADJUDICATION 6"  
JUDLVL7="LEVEL OF ADJUDICATION 7"  
JUDLVL8="LEVEL OF ADJUDICATION 8"  
JUDLVL9="LEVEL OF ADJUDICATION 9"  
LAF\_MR="SSDI LEDGER ACCOUNT FILE STATUS, MOST RECENT"  
LAF\_MR\_DATE="DATE OF MOST RECENT SSDI LEDGER ACCOUNT FILE STATUS"  
LANG\_MBR="SSDI WRITTEN LANGUAGE INDICATOR"  
LAST\_DODEC="MOST RECENT ADJUDICATION DATE"  
BENE\_LASTNAME\_NUMI="NUMIDENT BENEFICIARY LAST NAME"  
LOD\_DUAL="SSDI DUAL ELIG-LEVEL OF DENIAL CODE"  
LOD1="SSDI LEVEL OF DENIAL CODE 1"  
LOD10="SSDI LEVEL OF DENIAL CODE 10"  
LOD11="SSDI LEVEL OF DENIAL CODE 11"  
LOD12="SSDI LEVEL OF DENIAL CODE 12"  
LOD2="SSDI LEVEL OF DENIAL CODE 2"  
LOD3="SSDI LEVEL OF DENIAL CODE 3"  
LOD4="SSDI LEVEL OF DENIAL CODE 4"  
LOD5="SSDI LEVEL OF DENIAL CODE 5"  
LOD6="SSDI LEVEL OF DENIAL CODE 6"  
LOD7="SSDI LEVEL OF DENIAL CODE 7"  
LOD8="SSDI LEVEL OF DENIAL CODE 8"  
LOD9="SSDI LEVEL OF DENIAL CODE 9"  
LSAP="SSDI LUMP SUM AWARD AMOUNT"  
LSDC="SSDI LUMP SUM DISALLOWANCE CODE"  
LSFD="SSDI LUMP SUM FILING DATE"  
LSPA="SSDI LUMP SUM PAYMENT AMOUNT"  
MALE="MALE INDICATOR"  
MAX\_80JD="SSI MOST RECENT DECISION DATE"  
MAXAPPRD="SSI MOST RECENT APPLICATION DATE"  
MAXCLMFL="SSI MOST RECENT APPLICATION RECEIPT DATE"  
MAXELGRD="SSI MOST RECENT SSI ELIGIBILITY DATE"

Appendix A.78

JCL/SAS Code: DEMOLBLS

MAXFRSTP="SSI MOST RECENT APPLICATION, FIRST SSI PAYMENT DATE"  
MAXSTPRD="SSI MOST RECENT APPLICATION, ELIGIBILITY PERIOD END DATE"  
MAXSTRRD="SSI MOST RECENT START DATE OF SSI PAYMENTS"  
MFT="SSI MASTER FILE TYPE"  
MIE\_BEFORE\_CDR="MIE BEFORE FIRST CDR"  
MIN\_80JD="SSI EARLIEST AVAILABLE DECISION DATE"  
MINAPPRD="SSI EARLIEST AVAILABLE APPLICATION DATE"  
MINCLMFL="SSI EARLIEST AVAILABLE APPLICATION RECEIPT DATE"  
MINFRSTP="SSI EARLIEST AVAILABLE APPLICATION, FIRST SSI PAYMENT DATE"  
MINSTPRD=  
"SSI EARLIEST AVAILABLE APPLICATION ELIGIBILITY PERIOD END DATE"  
MINSTRRD="SSI EARLIEST AVAILABLE APPLICATION COMPUTATION DATE"  
BENE\_SUFNAME\_NUMI="NUMIDENT BENEFICIARY NAME SUFFIX"  
NDOF="NUMBER OF MBR DATE OF FILING FIELDS"  
NHDEMO="NUMBER OF DEMONSTRATION PROJECT OCCURRENCES"  
NHI=  
"NUMBER OF SSDI HOSPITAL INS. (HI) MEDICARE PART A FIELD OCCURRENCES"  
NINSD="NUMBER OF INSURED STATUS DATA OCCURRENCES"  
NODF="NUMBER OF MBR DISABILITY FIELDS"  
NPPIA\_NEW="NUMBER OF MBR PIA ENTRIES NEW"  
NPPIA\_ORIG="NUMBER OF MBR PIA ENTRIES ORIGINAL"  
NSMI=  
"NUMBER OF SSDI SUP. MED. INS. (SMI) MEDICARE PART B FIELD OCCURRENCES"  
PDD1="PRESUMPTIVE DECISION DATE 1"  
PDD10="PRESUMPTIVE DECISION DATE 10"  
PDD11="PRESUMPTIVE DECISION DATE 11"  
PDD12="PRESUMPTIVE DECISION DATE 12"  
PDD13="PRESUMPTIVE DECISION DATE 13"  
PDD14="PRESUMPTIVE DECISION DATE 14"  
PDD15="PRESUMPTIVE DECISION DATE 15"  
PDD16="PRESUMPTIVE DECISION DATE 16"  
PDD2="PRESUMPTIVE DECISION DATE 2"  
PDD3="PRESUMPTIVE DECISION DATE 3"  
PDD4="PRESUMPTIVE DECISION DATE 4"  
PDD5="PRESUMPTIVE DECISION DATE 5"  
PDD6="PRESUMPTIVE DECISION DATE 6"  
PDD7="PRESUMPTIVE DECISION DATE 7"  
PDD8="PRESUMPTIVE DECISION DATE 8"  
PDD9="PRESUMPTIVE DECISION DATE 9"  
STATECOUNTY\_SSR="SSI STATE & COUNTY SSA CODE"  
ZIP\_REPPAY\_SSR="SSI PAYEE ZIP CODE"  
ZIP\_EXT\_REPPAY\_SSR="SSI PAYEE ZIP SUFFIX"  
PD1="PRESUMPTIVE DISABILITY DECISION CODE 1"  
PD10="PRESUMPTIVE DISABILITY DECISION CODE 10"  
PD11="PRESUMPTIVE DISABILITY DECISION CODE 11"

Appendix A.78

JCL/SAS Code: DEMOLBLS

PD12="PRESUMPTIVE DISABILITY DECISION CODE 12"  
PD13="PRESUMPTIVE DISABILITY DECISION CODE 13"  
PD14="PRESUMPTIVE DISABILITY DECISION CODE 14"  
PD15="PRESUMPTIVE DISABILITY DECISION CODE 15"  
PD16="PRESUMPTIVE DISABILITY DECISION CODE 16"  
PD2="PRESUMPTIVE DISABILITY DECISION CODE 2"  
PD3="PRESUMPTIVE DISABILITY DECISION CODE 3"  
PD4="PRESUMPTIVE DISABILITY DECISION CODE 4"  
PD5="PRESUMPTIVE DISABILITY DECISION CODE 5"  
PD6="PRESUMPTIVE DISABILITY DECISION CODE 6"  
PD7="PRESUMPTIVE DISABILITY DECISION CODE 7"  
PD8="PRESUMPTIVE DISABILITY DECISION CODE 8"  
PD9="PRESUMPTIVE DISABILITY DECISION CODE 9"  
PIARFC1="SSDI REASON FOR CHANGE IN PIA 1"  
PIARFC10="SSDI REASON FOR CHANGE IN PIA 10"  
PIARFC11="SSDI REASON FOR CHANGE IN PIA 11"  
PIARFC12="SSDI REASON FOR CHANGE IN PIA 12"  
PIARFC13="SSDI REASON FOR CHANGE IN PIA 13"  
PIARFC14="SSDI REASON FOR CHANGE IN PIA 14"  
PIARFC15="SSDI REASON FOR CHANGE IN PIA 15"  
PIARFC16="SSDI REASON FOR CHANGE IN PIA 16"  
PIARFC17="SSDI REASON FOR CHANGE IN PIA 17"  
PIARFC18="SSDI REASON FOR CHANGE IN PIA 18"  
PIARFC19="SSDI REASON FOR CHANGE IN PIA 19"  
PIARFC2="SSDI REASON FOR CHANGE IN PIA 2"  
PIARFC20="SSDI REASON FOR CHANGE IN PIA 20"  
PIARFC21="SSDI REASON FOR CHANGE IN PIA 21"  
PIARFC22="SSDI REASON FOR CHANGE IN PIA 22"  
PIARFC23="SSDI REASON FOR CHANGE IN PIA 23"  
PIARFC24="SSDI REASON FOR CHANGE IN PIA 24"  
PIARFC25="SSDI REASON FOR CHANGE IN PIA 25"  
PIARFC26="SSDI REASON FOR CHANGE IN PIA 26"  
PIARFC27="SSDI REASON FOR CHANGE IN PIA 27"  
PIARFC28="SSDI REASON FOR CHANGE IN PIA 28"  
PIARFC29="SSDI REASON FOR CHANGE IN PIA 29"  
PIARFC3="SSDI REASON FOR CHANGE IN PIA 3"  
PIARFC30="SSDI REASON FOR CHANGE IN PIA 30"  
PIARFC31="SSDI REASON FOR CHANGE IN PIA 31"  
PIARFC32="SSDI REASON FOR CHANGE IN PIA 32"  
PIARFC33="SSDI REASON FOR CHANGE IN PIA 33"  
PIARFC34="SSDI REASON FOR CHANGE IN PIA 34"  
PIARFC35="SSDI REASON FOR CHANGE IN PIA 35"  
PIARFC36="SSDI REASON FOR CHANGE IN PIA 36"  
PIARFC37="SSDI REASON FOR CHANGE IN PIA 37"  
PIARFC38="SSDI REASON FOR CHANGE IN PIA 38"

Appendix A.78

JCL/SAS Code: DEMOLBLS

PIARFC39="SSDI REASON FOR CHANGE IN PIA 39"  
PIARFC4="SSDI REASON FOR CHANGE IN PIA 4"  
PIARFC40="SSDI REASON FOR CHANGE IN PIA 40"  
PIARFC41="SSDI REASON FOR CHANGE IN PIA 41"  
PIARFC42="SSDI REASON FOR CHANGE IN PIA 42"  
PIARFC43="SSDI REASON FOR CHANGE IN PIA 43"  
PIARFC44="SSDI REASON FOR CHANGE IN PIA 44"  
PIARFC45="SSDI REASON FOR CHANGE IN PIA 45"  
PIARFC46="SSDI REASON FOR CHANGE IN PIA 46"  
PIARFC47="SSDI REASON FOR CHANGE IN PIA 47"  
PIARFC48="SSDI REASON FOR CHANGE IN PIA 48"  
PIARFC49="SSDI REASON FOR CHANGE IN PIA 49"  
PIARFC5="SSDI REASON FOR CHANGE IN PIA 5"  
PIARFC50="SSDI REASON FOR CHANGE IN PIA 50"  
PIARFC6="SSDI REASON FOR CHANGE IN PIA 6"  
PIARFC7="SSDI REASON FOR CHANGE IN PIA 7"  
PIARFC8="SSDI REASON FOR CHANGE IN PIA 8"  
PIARFC9="SSDI REASON FOR CHANGE IN PIA 9"  
PIA1="SSDI PRIMARY INS. AMOUNT 1"  
PIA10="SSDI PRIMARY INS. AMOUNT 10"  
PIA11="SSDI PRIMARY INS. AMOUNT 11"  
PIA12="SSDI PRIMARY INS. AMOUNT 12"  
PIA13="SSDI PRIMARY INS. AMOUNT 13"  
PIA14="SSDI PRIMARY INS. AMOUNT 14"  
PIA15="SSDI PRIMARY INS. AMOUNT 15"  
PIA16="SSDI PRIMARY INS. AMOUNT 16"  
PIA17="SSDI PRIMARY INS. AMOUNT 17"  
PIA18="SSDI PRIMARY INS. AMOUNT 18"  
PIA19="SSDI PRIMARY INS. AMOUNT 19"  
PIA2="SSDI PRIMARY INS. AMOUNT 2"  
PIA20="SSDI PRIMARY INS. AMOUNT 20"  
PIA21="SSDI PRIMARY INS. AMOUNT 21"  
PIA22="SSDI PRIMARY INS. AMOUNT 22"  
PIA23="SSDI PRIMARY INS. AMOUNT 23"  
PIA24="SSDI PRIMARY INS. AMOUNT 24"  
PIA25="SSDI PRIMARY INS. AMOUNT 25"  
PIA26="SSDI PRIMARY INS. AMOUNT 26"  
PIA27="SSDI PRIMARY INS. AMOUNT 27"  
PIA28="SSDI PRIMARY INS. AMOUNT 28"  
PIA29="SSDI PRIMARY INS. AMOUNT 29"  
PIA3="SSDI PRIMARY INS. AMOUNT 3"  
PIA30="SSDI PRIMARY INS. AMOUNT 30"  
PIA31="SSDI PRIMARY INS. AMOUNT 31"  
PIA32="SSDI PRIMARY INS. AMOUNT 32"  
PIA33="SSDI PRIMARY INS. AMOUNT 33"

Appendix A.78

JCL/SAS Code: DEMOLBLS

PIA34="SSDI PRIMARY INS. AMOUNT 34"  
PIA35="SSDI PRIMARY INS. AMOUNT 35"  
PIA36="SSDI PRIMARY INS. AMOUNT 36"  
PIA37="SSDI PRIMARY INS. AMOUNT 37"  
PIA38="SSDI PRIMARY INS. AMOUNT 38"  
PIA39="SSDI PRIMARY INS. AMOUNT 39"  
PIA4="SSDI PRIMARY INS. AMOUNT 4"  
PIA40="SSDI PRIMARY INS. AMOUNT 40"  
PIA41="SSDI PRIMARY INS. AMOUNT 41"  
PIA42="SSDI PRIMARY INS. AMOUNT 42"  
PIA43="SSDI PRIMARY INS. AMOUNT 43"  
PIA44="SSDI PRIMARY INS. AMOUNT 44"  
PIA45="SSDI PRIMARY INS. AMOUNT 45"  
PIA46="SSDI PRIMARY INS. AMOUNT 46"  
PIA47="SSDI PRIMARY INS. AMOUNT 47"  
PIA48="SSDI PRIMARY INS. AMOUNT 48"  
PIA49="SSDI PRIMARY INS. AMOUNT 49"  
PIA5="SSDI PRIMARY INS. AMOUNT 5"  
PIA50="SSDI PRIMARY INS. AMOUNT 50"  
PIA6="SSDI PRIMARY INS. AMOUNT 6"  
PIA7="SSDI PRIMARY INS. AMOUNT 7"  
PIA8="SSDI PRIMARY INS. AMOUNT 8"  
PIA9="SSDI PRIMARY INS. AMOUNT 9"  
PIED1="SSDI PIA EFFECTIVE DATE 1"  
PIED10="SSDI PIA EFFECTIVE DATE 10"  
PIED11="SSDI PIA EFFECTIVE DATE 11"  
PIED12="SSDI PIA EFFECTIVE DATE 12"  
PIED13="SSDI PIA EFFECTIVE DATE 13"  
PIED14="SSDI PIA EFFECTIVE DATE 14"  
PIED15="SSDI PIA EFFECTIVE DATE 15"  
PIED16="SSDI PIA EFFECTIVE DATE 16"  
PIED17="SSDI PIA EFFECTIVE DATE 17"  
PIED18="SSDI PIA EFFECTIVE DATE 18"  
PIED19="SSDI PIA EFFECTIVE DATE 19"  
PIED2="SSDI PIA EFFECTIVE DATE 2"  
PIED20="SSDI PIA EFFECTIVE DATE 20"  
PIED21="SSDI PIA EFFECTIVE DATE 21"  
PIED22="SSDI PIA EFFECTIVE DATE 22"  
PIED23="SSDI PIA EFFECTIVE DATE 23"  
PIED24="SSDI PIA EFFECTIVE DATE 24"  
PIED25="SSDI PIA EFFECTIVE DATE 25"  
PIED26="SSDI PIA EFFECTIVE DATE 26"  
PIED27="SSDI PIA EFFECTIVE DATE 27"  
PIED28="SSDI PIA EFFECTIVE DATE 28"  
PIED29="SSDI PIA EFFECTIVE DATE 29"



Appendix A.78

JCL/SAS Code: DEMOLBLS

PIED3="SSDI PIA EFFECTIVE DATE 3"  
PIED30="SSDI PIA EFFECTIVE DATE 30"  
PIED31="SSDI PIA EFFECTIVE DATE 31"  
PIED32="SSDI PIA EFFECTIVE DATE 32"  
PIED33="SSDI PIA EFFECTIVE DATE 33"  
PIED34="SSDI PIA EFFECTIVE DATE 34"  
PIED35="SSDI PIA EFFECTIVE DATE 35"  
PIED36="SSDI PIA EFFECTIVE DATE 36"  
PIED37="SSDI PIA EFFECTIVE DATE 37"  
PIED38="SSDI PIA EFFECTIVE DATE 38"  
PIED39="SSDI PIA EFFECTIVE DATE 39"  
PIED4="SSDI PIA EFFECTIVE DATE 4"  
PIED40="SSDI PIA EFFECTIVE DATE 40"  
PIED41="SSDI PIA EFFECTIVE DATE 41"  
PIED42="SSDI PIA EFFECTIVE DATE 42"  
PIED43="SSDI PIA EFFECTIVE DATE 43"  
PIED44="SSDI PIA EFFECTIVE DATE 44"  
PIED45="SSDI PIA EFFECTIVE DATE 45"  
PIED46="SSDI PIA EFFECTIVE DATE 46"  
PIED47="SSDI PIA EFFECTIVE DATE 47"  
PIED48="SSDI PIA EFFECTIVE DATE 48"  
PIED49="SSDI PIA EFFECTIVE DATE 49"  
PIED5="SSDI PIA EFFECTIVE DATE 5"  
PIED50="SSDI PIA EFFECTIVE DATE 50"  
PIED6="SSDI PIA EFFECTIVE DATE 6"  
PIED7="SSDI PIA EFFECTIVE DATE 7"  
PIED8="SSDI PIA EFFECTIVE DATE 8"  
PIED9="SSDI PIA EFFECTIVE DATE 9"  
PNOB="SSDI NUMBER OF DEPENDENT BENEFICIARIES INCLUDED IN PAYMENT"  
STATE\_MR="BENEFICIARY'S MOST RECENT AVAILABLE STATE OF RESIDENCE"  
PSTA\_MR="SSI STATUS CODE, MOST RECENT"  
PSTA\_MR\_DATE="DATE OF MOST RECENT SSI STATUS CODE"  
QDDIND1="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 1"  
QDDIND2="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 2"  
QDDIND3="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 3"  
QDDIND4="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 4"  
QDDIND5="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 5"  
QDDIND6="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 6"  
QDDIND7="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 7"  
QDDIND8="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 8"  
QDDIND9="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 9"  
QDDIND10="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 10"  
QDDIND11="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 11"  
QDDIND12="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 12"  
QDDIND13="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 13"

Appendix A.78

JCL/SAS Code: DEMOLBLS

QDDIND14="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 14"  
QDDIND15="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 15"  
QDDIND16="QUICK DISABILITY DETERMINATION (QDD) INDICATOR 16"  
QDREINDT1="QDD REINSTATEMENT DATE 1"  
QDREINDT2="QDD REINSTATEMENT DATE 2"  
QDREINDT3="QDD REINSTATEMENT DATE 3"  
QDREINDT4="QDD REINSTATEMENT DATE 4"  
QDREINDT5="QDD REINSTATEMENT DATE 5"  
QDREINDT6="QDD REINSTATEMENT DATE 6"  
QDREINDT7="QDD REINSTATEMENT DATE 7"  
QDREINDT8="QDD REINSTATEMENT DATE 8"  
QDREINDT9="QDD REINSTATEMENT DATE 9"  
QDREINDT10="QDD REINSTATEMENT DATE 10"  
QDREINDT11="QDD REINSTATEMENT DATE 11"  
QDREINDT12="QDD REINSTATEMENT DATE 12"  
QDREINDT13="QDD REINSTATEMENT DATE 13"  
QDREINDT14="QDD REINSTATEMENT DATE 14"  
QDREINDT15="QDD REINSTATEMENT DATE 15"  
QDREINDT16="QDD REINSTATEMENT DATE 16"  
QDRMVDT1="QDD REMOVAL DATE 1"  
QDRMVDT2="QDD REMOVAL DATE 2"  
QDRMVDT3="QDD REMOVAL DATE 3"  
QDRMVDT4="QDD REMOVAL DATE 4"  
QDRMVDT5="QDD REMOVAL DATE 5"  
QDRMVDT6="QDD REMOVAL DATE 6"  
QDRMVDT7="QDD REMOVAL DATE 7"  
QDRMVDT8="QDD REMOVAL DATE 8"  
QDRMVDT9="QDD REMOVAL DATE 9"  
QDRMVDT10="QDD REMOVAL DATE 10"  
QDRMVDT11="QDD REMOVAL DATE 11"  
QDRMVDT12="QDD REMOVAL DATE 12"  
QDRMVDT13="QDD REMOVAL DATE 13"  
QDRMVDT14="QDD REMOVAL DATE 14"  
QDRMVDT15="QDD REMOVAL DATE 15"  
QDRMVDT16="QDD REMOVAL DATE 16"  
QDRMVRSN1="QDD REMOVAL REASON 1"  
QDRMVRSN2="QDD REMOVAL REASON 2"  
QDRMVRSN3="QDD REMOVAL REASON 3"  
QDRMVRSN4="QDD REMOVAL REASON 4"  
QDRMVRSN5="QDD REMOVAL REASON 5"  
QDRMVRSN6="QDD REMOVAL REASON 6"  
QDRMVRSN7="QDD REMOVAL REASON 7"  
QDRMVRSN8="QDD REMOVAL REASON 8"  
QDRMVRSN9="QDD REMOVAL REASON 9"  
QDRMVRSN10="QDD REMOVAL REASON 10"

## Appendix A.78

JCL/SAS Code: DEMOLBLS

```
QDRMVRSN11="QDD REMOVAL REASON 11"  
QDRMVRSN12="QDD REMOVAL REASON 12"  
QDRMVRSN13="QDD REMOVAL REASON 13"  
QDRMVRSN14="QDD REMOVAL REASON 14"  
QDRMVRSN15="QDD REMOVAL REASON 15"  
QDRMVRSN16="QDD REMOVAL REASON 16"  
QDScore1="QDD SCORING CODE 1"  
QDScore2="QDD SCORING CODE 2"  
QDScore3="QDD SCORING CODE 3"  
QDScore4="QDD SCORING CODE 4"  
QDScore5="QDD SCORING CODE 5"  
QDScore6="QDD SCORING CODE 6"  
QDScore7="QDD SCORING CODE 7"  
QDScore8="QDD SCORING CODE 8"  
QDScore9="QDD SCORING CODE 9"  
QDScore10="QDD SCORING CODE 10"  
QDScore11="QDD SCORING CODE 11"  
QDScore12="QDD SCORING CODE 12"  
QDScore13="QDD SCORING CODE 13"  
QDScore14="QDD SCORING CODE 14"  
QDScore15="QDD SCORING CODE 15"  
QDScore16="QDD SCORING CODE 16"  
/*RACE="RACE OR ETHNICITY"  
RACE_ASIAN="RACE: ASIAN"  
RACE_BLACK="RACE: AFRICAN-AMERICAN"  
RACE_HISPANIC="RACE: HISPANIC"  
RACE_NAINDIAN="RACE: NATIVE AMERICAN"  
RACE_OTHER="RACE: OTHER"  
RACE_UNKNOWN="RACE: UNKNOWN"  
RACE_WHITE="RACE: WHITE"  
RACEMISS="RACE: MISSING"*/  
RB1="REGULATION BASIS CODE 1"  
RB10="REGULATION BASIS CODE 10"  
RB11="REGULATION BASIS CODE 11"  
RB12="REGULATION BASIS CODE 12"  
RB13="REGULATION BASIS CODE 13"  
RB14="REGULATION BASIS CODE 14"  
RB15="REGULATION BASIS CODE 15"  
RB16="REGULATION BASIS CODE 16"  
RB2="REGULATION BASIS CODE 2"  
RB3="REGULATION BASIS CODE 3"  
RB4="REGULATION BASIS CODE 4"  
RB5="REGULATION BASIS CODE 5"  
RB6="REGULATION BASIS CODE 6"  
RB7="REGULATION BASIS CODE 7"
```

Appendix A.78

JCL/SAS Code: DEMOLBLS

```
RB8="REGULATION BASIS CODE 8"  
RB9="REGULATION BASIS CODE 9"  
RDD="SSDI REASON FOR DISALLOWANCE OR DENIAL"  
RDT1="RESULT OF DISABILITY DETERMINATION 1"  
RDT10="RESULT OF DISABILITY DETERMINATION 10"  
RDT11="RESULT OF DISABILITY DETERMINATION 11"  
RDT12="RESULT OF DISABILITY DETERMINATION 12"  
RDT13="RESULT OF DISABILITY DETERMINATION 13"  
RDT14="RESULT OF DISABILITY DETERMINATION 14"  
RDT15="RESULT OF DISABILITY DETERMINATION 15"  
RDT16="RESULT OF DISABILITY DETERMINATION 16"  
RDT2="RESULT OF DISABILITY DETERMINATION 2"  
RDT3="RESULT OF DISABILITY DETERMINATION 3"  
RDT4="RESULT OF DISABILITY DETERMINATION 4"  
RDT5="RESULT OF DISABILITY DETERMINATION 5"  
RDT6="RESULT OF DISABILITY DETERMINATION 6"  
RDT7="RESULT OF DISABILITY DETERMINATION 7"  
RDT8="RESULT OF DISABILITY DETERMINATION 8"  
RDT9="RESULT OF DISABILITY DETERMINATION 9"  
/*REMDIG1="PRIMARY 4-DIGIT DIAGNOSIS - CER100% FIELD FILE"*/  
PAYEE_TYP_SSI="SSI TYPE OF PAYEE CODE"  
REPPYSSD="SSDI REPRESENTATIVE PAYEE INDICATOR"  
REPPYSSI="SSI REPRESENTATIVE PAYEE INDICATOR"  
/*RETIREDATE="DATE BENEFICIARY REACHES FULL RETIREMENT AGE"*/  
RFC1="REASON FOR CONTINUANCE/CESSATION CODE 1"  
RFC10="REASON FOR CONTINUANCE/CESSATION CODE 10"  
RFC11="REASON FOR CONTINUANCE/CESSATION CODE 11"  
RFC12="REASON FOR CONTINUANCE/CESSATION CODE 12"  
RFC13="REASON FOR CONTINUANCE/CESSATION CODE 13"  
RFC14="REASON FOR CONTINUANCE/CESSATION CODE 14"  
RFC15="REASON FOR CONTINUANCE/CESSATION CODE 15"  
RFC16="REASON FOR CONTINUANCE/CESSATION CODE 16"  
RFC2="REASON FOR CONTINUANCE/CESSATION CODE 2"  
RFC3="REASON FOR CONTINUANCE/CESSATION CODE 3"  
RFC4="REASON FOR CONTINUANCE/CESSATION CODE 4"  
RFC5="REASON FOR CONTINUANCE/CESSATION CODE 5"  
RFC6="REASON FOR CONTINUANCE/CESSATION CODE 6"  
RFC7="REASON FOR CONTINUANCE/CESSATION CODE 7"  
RFC8="REASON FOR CONTINUANCE/CESSATION CODE 8"  
RFC9="REASON FOR CONTINUANCE/CESSATION CODE 9"  
RID1="PROGRAM IDENTIFICATION 1"  
RID10="PROGRAM IDENTIFICATION 10"  
RID11="PROGRAM IDENTIFICATION 11"  
RID12="PROGRAM IDENTIFICATION 12"  
RID13="PROGRAM IDENTIFICATION 13"
```

Appendix A.78

JCL/SAS Code: DEMOLBLS

RID14="PROGRAM IDENTIFICATION 14"  
RID15="PROGRAM IDENTIFICATION 15"  
RID16="PROGRAM IDENTIFICATION 16"  
RID2="PROGRAM IDENTIFICATION 2"  
RID3="PROGRAM IDENTIFICATION 3"  
RID4="PROGRAM IDENTIFICATION 4"  
RID5="PROGRAM IDENTIFICATION 5"  
RID6="PROGRAM IDENTIFICATION 6"  
RID7="PROGRAM IDENTIFICATION 7"  
RID8="PROGRAM IDENTIFICATION 8"  
RID9="PROGRAM IDENTIFICATION 9"  
RP="RACE OF THE PRIMARY"  
ZIP\_RES\_MBR="SSDI RESIDENCE ZIP CODE"  
DXSEC\_MBR\_MR="SSDI SECONDARY DIAGNOSIS CODE, MOST RECENT"  
SDS\_DUAL="SSDI DUAL ELIG-SGA DISABILITY CESSATION DATE"  
SDS1="SSDI SGA DISABILITY CESSATION 1"  
SDS10="SSDI SGA DISABILITY CESSATION 10"  
SDS11="SSDI SGA DISABILITY CESSATION 11"  
SDS12="SSDI SGA DISABILITY CESSATION 12"  
SDS2="SSDI SGA DISABILITY CESSATION 2"  
SDS3="SSDI SGA DISABILITY CESSATION 3"  
SDS4="SSDI SGA DISABILITY CESSATION 4"  
SDS5="SSDI SGA DISABILITY CESSATION 5"  
SDS6="SSDI SGA DISABILITY CESSATION 6"  
SDS7="SSDI SGA DISABILITY CESSATION 7"  
SDS8="SSDI SGA DISABILITY CESSATION 8"  
SDS9="SSDI SGA DISABILITY CESSATION 9"  
SEX="BENEFICIARY SEX"  
SEXMISS="INDICATOR FOR MISSING SEX DATA"  
SIFT="SSDI SECURITY INCOME FILE TYPE"  
SISC="SSI INCOME STATUS CODE"  
SLAC="SSI LIVING ARRANGEMENT CODE"  
SMI\_START\_MR=  
"SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B MOST RECENT"  
SMI\_START1="SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 1"  
SMI\_START10=  
"SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 10"  
SMI\_START11=  
"SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 11"  
SMI\_START12=  
"SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 12"  
SMI\_START13=  
"SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 13"  
SMI\_START14=  
"SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 14"

Appendix A.78

JCL/SAS Code: DEMOLBLS

```
SMI_START15=
"SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 15"
SMI_START16=
"SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 16"
SMI_START17=
"SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 17"
SMI_START18=
"SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 18"
SMI_START19=
"SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 19"
SMI_START20="SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 20"
SMI_START2="SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 2"
SMI_START3="SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 3"
SMI_START4="SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 4"
SMI_START5="SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 5"
SMI_START6="SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 6"
SMI_START7="SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 7"
SMI_START8="SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 8"
SMI_START9="SSDI SUPP. MEDICAL INS. (SMI) START DATE, MEDICARE PART B 9"
SMI_TERM_MR=
"SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B MOST RECENT"
SMI_TERM1="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 1"
SMI_TERM10="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 10"
SMI_TERM11="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 11"
SMI_TERM12="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 12"
SMI_TERM13="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 13"
SMI_TERM14="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 14"
SMI_TERM15="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 15"
SMI_TERM16="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 16"
SMI_TERM17="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 17"
SMI_TERM18="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 18"
SMI_TERM19="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 19"
SMI_TERM2="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 2"
SMI_TERM20="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 20"
SMI_TERM3="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 3"
SMI_TERM4="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 4"
SMI_TERM5="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 5"
SMI_TERM6="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 6"
SMI_TERM7="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 7"
SMI_TERM8="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 8"
SMI_TERM9="SSDI SUPP. MEDICAL INS. (SMI) END DATE, MEDICARE PART B 9"
SSDI_ONSET="SSDI DISABILITY ONSET DATE EARLIEST PERIOD OF ELIGIBILITY"
```

```
%DO YEAR=&BEGYR. %TO &ENDYR.;
```

Appendix A.78

JCL/SAS Code: DEMOLBLS

```
%LET YR=%SUBSTR(&YEAR.,3,2);
SSDI&YR.="SSDI CURRENT PAY STATUS IN &YEAR."
SSI&YR.="SSI CURRENT PAY STATUS IN &YEAR."
%END;
```

```
SSIELIG_ADULT="SSI EARLIEST ADULT ELIGIBILITY DATE (CONSTRUCTED)"
SSIELIG_FIRST="SSI EARLIEST ELIGIBILITY DATE (CONSTRUCTED)"
SSN="SOCIAL SECURITY NUMBER"
START_RD="SSI EARLIEST COMPUTATION DATE"
STATE_MBR="SSDI SSA STATE CODE OF RESIDENCE"
/*STATECD_ENDYR="PAYMENT STATE CODE"*/
STOP_RD="SSI END DATE FOR ELIGIBILITY PERIOD"
TAC="SSDI TYPE OF AWARD CODE"
```

```
%DO YEAR=2002 %TO &ENDYR.;
%LET YR=%SUBSTR(&YEAR.,3,2);
TKT&YR.="TICKET ACTIVE IN &YEAR."
%END;
```

```
TOA="SSI TYPE OF ACTION CODE"
TOC_HIST1="SSDI HISTORICAL TYPE OF CLAIM (TOC) 1"
TOC_HIST2="SSDI HISTORICAL TYPE OF CLAIM (TOC) 2"
TOC_HIST3="SSDI HISTORICAL TYPE OF CLAIM (TOC) 3"
TOC_HIST4="SSDI HISTORICAL TYPE OF CLAIM (TOC) 4"
TOC_HIST5="SSDI HISTORICAL TYPE OF CLAIM (TOC) 5"
TOC_HIST6="SSDI HISTORICAL TYPE OF CLAIM (TOC) 6"
TOC_HIST7="SSDI HISTORICAL TYPE OF CLAIM (TOC) 7"
TOC_HIST8="SSDI HISTORICAL TYPE OF CLAIM (TOC) 8"
TOC_HIST9="SSDI HISTORICAL TYPE OF CLAIM (TOC) 9"
TOC_HIST10="SSDI HISTORICAL TYPE OF CLAIM (TOC) 10"
TOC_HIST11="SSDI HISTORICAL TYPE OF CLAIM (TOC) 11"
TOC_HIST12="SSDI HISTORICAL TYPE OF CLAIM (TOC) 12"
TOC_HIST13="SSDI HISTORICAL TYPE OF CLAIM (TOC) 13"
TOC_HIST14="SSDI HISTORICAL TYPE OF CLAIM (TOC) 14"
TOC_HIST15="SSDI HISTORICAL TYPE OF CLAIM (TOC) 15"
TOC_HIST16="SSDI HISTORICAL TYPE OF CLAIM (TOC) 16"
TOC_HIST17="SSDI HISTORICAL TYPE OF CLAIM (TOC) 17"
TOC_HIST18="SSDI HISTORICAL TYPE OF CLAIM (TOC) 18"
TOC_HIST19="SSDI HISTORICAL TYPE OF CLAIM (TOC) 19"
TOC_HIST20="SSDI HISTORICAL TYPE OF CLAIM (TOC) 20"
TOC_HIST21="SSDI HISTORICAL TYPE OF CLAIM (TOC) 21"
TOC_HIST22="SSDI HISTORICAL TYPE OF CLAIM (TOC) 22"
TOC_HIST23="SSDI HISTORICAL TYPE OF CLAIM (TOC) 23"
TOC_HIST24="SSDI HISTORICAL TYPE OF CLAIM (TOC) 24"
TOC_HIST25="SSDI HISTORICAL TYPE OF CLAIM (TOC) 25"
```

Appendix A.78

JCL/SAS Code: DEMOLBLS

TOC\_HIST26="SSDI HISTORICAL TYPE OF CLAIM (TOC) 26"  
TOC\_HIST27="SSDI HISTORICAL TYPE OF CLAIM (TOC) 27"  
TOC\_HIST28="SSDI HISTORICAL TYPE OF CLAIM (TOC) 28"  
TOC\_HIST29="SSDI HISTORICAL TYPE OF CLAIM (TOC) 29"  
TOC\_HIST30="SSDI HISTORICAL TYPE OF CLAIM (TOC) 30"  
TOC\_HIST31="SSDI HISTORICAL TYPE OF CLAIM (TOC) 31"  
TOC\_HIST32="SSDI HISTORICAL TYPE OF CLAIM (TOC) 32"  
TOC\_HIST33="SSDI HISTORICAL TYPE OF CLAIM (TOC) 33"  
TOC\_HIST34="SSDI HISTORICAL TYPE OF CLAIM (TOC) 34"  
TOC\_HIST35="SSDI HISTORICAL TYPE OF CLAIM (TOC) 35"  
TOC\_HIST36="SSDI HISTORICAL TYPE OF CLAIM (TOC) 36"  
TOC\_HIST37="SSDI HISTORICAL TYPE OF CLAIM (TOC) 37"  
TOC\_HIST38="SSDI HISTORICAL TYPE OF CLAIM (TOC) 38"  
TOC\_HIST39="SSDI HISTORICAL TYPE OF CLAIM (TOC) 39"  
TOC\_HIST40="SSDI HISTORICAL TYPE OF CLAIM (TOC) 40"  
TOC\_HIST\_NUM="SSDI NUMBER OF HISTORICAL TYPE OF CLAIM (TOC) OCCURRENCES"  
TOC\_HIST\_START1="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 1"  
TOC\_HIST\_START2="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 2"  
TOC\_HIST\_START3="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 3"  
TOC\_HIST\_START4="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 4"  
TOC\_HIST\_START5="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 5"  
TOC\_HIST\_START6="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 6"  
TOC\_HIST\_START7="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 7"  
TOC\_HIST\_START8="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 8"  
TOC\_HIST\_START9="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 9"  
TOC\_HIST\_START10="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 10"  
TOC\_HIST\_START11="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 11"  
TOC\_HIST\_START12="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 12"  
TOC\_HIST\_START13="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 13"  
TOC\_HIST\_START14="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 14"  
TOC\_HIST\_START15="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 15"  
TOC\_HIST\_START16="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 16"  
TOC\_HIST\_START17="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 17"  
TOC\_HIST\_START18="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 18"  
TOC\_HIST\_START19="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 19"  
TOC\_HIST\_START20="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 20"  
TOC\_HIST\_START21="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 21"  
TOC\_HIST\_START22="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 22"  
TOC\_HIST\_START23="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 23"  
TOC\_HIST\_START24="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 24"  
TOC\_HIST\_START25="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 25"  
TOC\_HIST\_START26="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 26"  
TOC\_HIST\_START27="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 27"  
TOC\_HIST\_START28="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 28"  
TOC\_HIST\_START29="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 29"



Appendix A.78

JCL/SAS Code: DEMOLBLS

TOC\_HIST\_START30="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 30"  
TOC\_HIST\_START31="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 31"  
TOC\_HIST\_START32="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 32"  
TOC\_HIST\_START33="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 33"  
TOC\_HIST\_START34="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 34"  
TOC\_HIST\_START35="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 35"  
TOC\_HIST\_START36="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 36"  
TOC\_HIST\_START37="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 37"  
TOC\_HIST\_START38="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 38"  
TOC\_HIST\_START39="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 39"  
TOC\_HIST\_START40="SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 40"  
TOC\_NUM="SSDI NUMBER OF TYPE OF CLAIM (TOC) OCCURRENCES"  
TOC\_START1="SSDI TYPE OF CLAIM (TOC) START DATE 1"  
TOC\_START10="SSDI TYPE OF CLAIM (TOC) START DATE 10"  
TOC\_START11="SSDI TYPE OF CLAIM (TOC) START DATE 11"  
TOC\_START12="SSDI TYPE OF CLAIM (TOC) START DATE 12"  
TOC\_START13="SSDI TYPE OF CLAIM (TOC) START DATE 13"  
TOC\_START14="SSDI TYPE OF CLAIM (TOC) START DATE 14"  
TOC\_START15="SSDI TYPE OF CLAIM (TOC) START DATE 15"  
TOC\_START16="SSDI TYPE OF CLAIM (TOC) START DATE 16"  
TOC\_START17="SSDI TYPE OF CLAIM (TOC) START DATE 17"  
TOC\_START18="SSDI TYPE OF CLAIM (TOC) START DATE 18"  
TOC\_START19="SSDI TYPE OF CLAIM (TOC) START DATE 19"  
TOC\_START2="SSDI TYPE OF CLAIM (TOC) START DATE 2"  
TOC\_START20="SSDI TYPE OF CLAIM (TOC) START DATE 20"  
TOC\_START3="SSDI TYPE OF CLAIM (TOC) START DATE 3"  
TOC\_START4="SSDI TYPE OF CLAIM (TOC) START DATE 4"  
TOC\_START5="SSDI TYPE OF CLAIM (TOC) START DATE 5"  
TOC\_START6="SSDI TYPE OF CLAIM (TOC) START DATE 6"  
TOC\_START7="SSDI TYPE OF CLAIM (TOC) START DATE 7"  
TOC\_START8="SSDI TYPE OF CLAIM (TOC) START DATE 8"  
TOC\_START9="SSDI TYPE OF CLAIM (TOC) START DATE 9"  
TOC1="SSDI TYPE OF CLAIM 1"  
TOC10="SSDI TYPE OF CLAIM 10"  
TOC11="SSDI TYPE OF CLAIM 11"  
TOC12="SSDI TYPE OF CLAIM 12"  
TOC13="SSDI TYPE OF CLAIM 13"  
TOC14="SSDI TYPE OF CLAIM 14"  
TOC15="SSDI TYPE OF CLAIM 15"  
TOC16="SSDI TYPE OF CLAIM 16"  
TOC17="SSDI TYPE OF CLAIM 17"  
TOC18="SSDI TYPE OF CLAIM 18"  
TOC19="SSDI TYPE OF CLAIM 19"  
TOC2="SSDI TYPE OF CLAIM 2"  
TOC20="SSDI TYPE OF CLAIM 20"

Appendix A.78

JCL/SAS Code: DEMOLBLS

TOC3="SSDI TYPE OF CLAIM 3"  
TOC4="SSDI TYPE OF CLAIM 4"  
TOC5="SSDI TYPE OF CLAIM 5"  
TOC6="SSDI TYPE OF CLAIM 6"  
TOC7="SSDI TYPE OF CLAIM 7"  
TOC8="SSDI TYPE OF CLAIM 8"  
TOC9="SSDI TYPE OF CLAIM 9"  
PAYEE\_TYP\_SSDI="SSDI TYPE OF PAYEE"  
TWPCMPLMNT1="SSDI TRIAL WORK PERIOD (TWP) COMPLETION MONTH 1"  
TWPCMPLMNT2="SSDI TRIAL WORK PERIOD (TWP) COMPLETION MONTH 2"  
TWPCMPLMNT3="SSDI TRIAL WORK PERIOD (TWP) COMPLETION MONTH 3"  
TWPCMPLMNT4="SSDI TRIAL WORK PERIOD (TWP) COMPLETION MONTH 4"  
TWPCMPLMNT5="SSDI TRIAL WORK PERIOD (TWP) COMPLETION MONTH 5"  
T16APPL1="SSI MEDICAL DECISION APPEAL DATE 1"  
T16APPL10="SSI MEDICAL DECISION APPEAL DATE 10"  
T16APPL2="SSI MEDICAL DECISION APPEAL DATE 2"  
T16APPL3="SSI MEDICAL DECISION APPEAL DATE 3"  
T16APPL4="SSI MEDICAL DECISION APPEAL DATE 4"  
T16APPL5="SSI MEDICAL DECISION APPEAL DATE 5"  
T16APPL6="SSI MEDICAL DECISION APPEAL DATE 6"  
T16APPL7="SSI MEDICAL DECISION APPEAL DATE 7"  
T16APPL8="SSI MEDICAL DECISION APPEAL DATE 8"  
T16APPL9="SSI MEDICAL DECISION APPEAL DATE 9"  
LANGSSR="SSI LANGUAGE"  
T16START1="SSI BENEFIT ENTITLEMENT DATE 1"  
T16START10="SSI BENEFIT ENTITLEMENT DATE 10"  
T16START2="SSI BENEFIT ENTITLEMENT DATE 2"  
T16START3="SSI BENEFIT ENTITLEMENT DATE 3"  
T16START4="SSI BENEFIT ENTITLEMENT DATE 4"  
T16START5="SSI BENEFIT ENTITLEMENT DATE 5"  
T16START6="SSI BENEFIT ENTITLEMENT DATE 6"  
T16START7="SSI BENEFIT ENTITLEMENT DATE 7"  
T16START8="SSI BENEFIT ENTITLEMENT DATE 8"  
T16START9="SSI BENEFIT ENTITLEMENT DATE 9"  
T16STOP1="SSI BENEFIT CESSATION DATE 1"  
T16STOP10="SSI BENEFIT CESSATION DATE 10"  
T16STOP2="SSI BENEFIT CESSATION DATE 2"  
T16STOP3="SSI BENEFIT CESSATION DATE 3"  
T16STOP4="SSI BENEFIT CESSATION DATE 4"  
T16STOP5="SSI BENEFIT CESSATION DATE 5"  
T16STOP6="SSI BENEFIT CESSATION DATE 6"  
T16STOP7="SSI BENEFIT CESSATION DATE 7"  
T16STOP8="SSI BENEFIT CESSATION DATE 8"  
T16STOP9="SSI BENEFIT CESSATION DATE 9"  
VRA1="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 1"

Appendix A.78

JCL/SAS Code: DEMOLBLS

```
VRA10="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 10"  
VRA11="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 11"  
VRA12="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 12"  
VRA13="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 13"  
VRA14="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 14"  
VRA15="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 15"  
VRA16="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 16"  
VRA2="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 2"  
VRA3="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 3"  
VRA4="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 4"  
VRA5="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 5"  
VRA6="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 6"  
VRA7="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 7"  
VRA8="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 8"  
VRA9="VOCATIONAL REHABILITATION REFERRAL ACTION CODE 9"  
;  
%MEND GETLB;  
%GETLB;
```

## Appendix A.79

JCL/SAS Code: FINDMG

```
//#8047FDM JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//JOB01 EXEC SAS9,
//      WORK='180000,90000'
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  JODY SCHIMMEL-HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#7429.DAF18.DEMO.PRDLIB(FINDEMO)
//* DESCRIPTION: FINALIZE DMG FILE, CREATE 10% FILE
//* DATE:
//* UPDATE:   11/01/2019 MIAOMIAO SHEN FOR DAF18
//*****
//*
//DEM DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.DMG.SA.V1,DISP=SHR
//COM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//* GET PST_MR FROM PSTMR DATASET CREATED DURING INTERMEDIATE
//* ANNUALS PROCESSING
//PST DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.PSTMR.SA.V2,DISP=SHR
//LAF DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.LAFMR.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.DMG,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMGCON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUT3 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10DMG,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUT4 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.P10DCON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS=MAX
      MPRINT MACROGEN;

%LET BEGYR=1994; /*CHANGE AS NEEDED*/
%LET ENDYR=2021; /*CHANGE AS NEEDED*/

%MACRO START;

PROC CONTENTS DATA=DEM.DMG; TITLE "CHECK DEMO BEFORE FINALIZATION"; RUN;
DATA OUT1.DMG;
```

## Appendix A.79

JCL/SAS Code: FINDMG

```
MERGE DEM.DMG ( IN=INFIN

      DROP= BOND LAF
            REMDIG1 STATECD_ENDYR
            WHERE=(YEAR(DODBEST)>=1996 OR
                   YEAR(DODBEST)=.)
      COM.COMVARS
      PST.PSTMTR /* THIS ONLY HAS SSN AND PST_MR VARS */
                (KEEP=SSN STATE_MR

      LAF.LAFMR
      ;
      BY SSN;

      %INCLUDE "OPDR.TG.PRD.ETTW.#8047.DAF21.DEMO.PRDLIB(DEMOLBLS)";

      IF INFIN;

      /* COLLAPSE GROUPS OF VALUES */
      ARRAY RB(*)$ FIRSTRB RB1-RB16;

      DO I=1 TO DIM(RB);
        IF RB(I) IN ("K1","L1","M3","M5","M7","Z1") THEN RB(I)="W1";
        ELSE IF RB(I) IN ("K2","L2","M4","M6","M8","Z2") THEN RB(I)="W2";
        ELSE IF RB(I) IN ("36","37","38","39") THEN RB(I)="99";
      END;

      IF RDD IN ("0L1","0L2","0M5","0M6","0K1","0K2","0M3","0M4",
                "0M7","0M8","0Z1","0Z2","0X3") THEN RDD="OTH";

      ARRAY BDC(*)$ BDC1 - BDC12;

      DO J=1 TO DIM(BDC);
        IF BDC(J) IN ("L1","L2","M5","M6","K1","K2","M3","M4",
                    "M7","M8","Z1","Z2","0X3") THEN BDC(J)="TH";
      END;

      IF CURSTAT IN ("N10","N11","N12","N17","N18","N20","N36",
                   "N37","N38","N39","N47","N48","N49","N50")
        THEN CURSTAT="W01";

      ARRAY CDR(*)$ DCR_SSDI_DUAL DCR_SSDI1 - DCR_SSDI12;

      DO K=1 TO DIM(CDR);
```

Appendix A.79

JCL/SAS Code: FINDMG

```

    IF CDR(K) IN ("F","N","W") THEN CDR(K)="X";
END;

DROP I J K
    DODEC_INDEX_8080_JD MAXAPPRD RP MO YR
;

RENAME APP_RD=MAXAPPRD;
RUN;

TITLE "CONTENTS OF DMG FILE";
PROC CONTENTS DATA=OUT1.DMG OUT=OUT2.DMGCON;
RUN;
TITLE;

/* CREATE THE SIMPLE 10% EXTRACT DATA MART FOR DEMO */
DATA OUT3.P10DMG;
SET OUT1.DMG;
IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9')) AND
    (SUBSTR(SSN,9,1) IN ('0', '5'));
RUN;

TITLE "CONTENTS OF 10% DMG FILE";
PROC CONTENTS DATA=OUT3.P10DMG OUT=OUT4.P10DCON;
RUN;
TITLE;

%MEND START;
%START;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>

```

Appendix A.79

JCL/SAS Code: FINDMG

SUBJECT: DMG FINALIZATION COMPLETE

Appendix A.80

JCL/SAS Code: Upload Provider Data

```

*****
* PROJECT: Evaluation of the Ticket to Work Program-Part A
*   Evaluation Implementation () DAF30
* PROJECT
* DIRECTOR:  MATT URATO
*
* PROGRAM:   M:\DAF21\TASK 11 Create Ticket Component\Programs\Upload Provider Data.sas
*
*
* DESCRIPTION: UPLOADS PROVIDER DATA
*
* DATE:      07/07/2022 ALEX BRYCE
***** ;

LIBNAME OUT 'M:\DAF21\TASK 11 Create Ticket Component\Provider File';

/*set macro vars*/

%let fname = DAF_EN_Provider_File_062822;
%let datafile = M:\DAF21\TASK 11 Create Ticket Component\Provider File\EN Provider File 062822.xlsx;
%let libloc = M:\DAF21\TASK 11 Create Ticket Component\Programs;
%let outloc = M:\DAF21\TASK 11 Create Ticket Component\Provider File;

/*THIS FILE IS OBTAINED FROM THE HYPERION SYSTEM BY KATIE STRIEBINGER VIA EMAIL REQUEST.
THE FILE CONTAINS DATA RELATED PROVIDERS (EN & VR) FOR THE TTW PROGRAM*/

PROC IMPORT OUT= OUT.&fname.
    DATAFILE= "&datafile."
    DBMS=XLSX REPLACE;
    SHEET='DAF_EN_Provider';
    GETNAMES=YES;/*
    MIXED=YES;
    USEDATE=YES;
    SCANTIME=YES;*/
RUN;
/*
proc compare base=out.En_provider_file_060619
              compare=out.En_provider_file_062719;

run;
*/

PROC PRINT DATA = OUT.&fname.;
RUN;

```



Appendix A.80

JCL/SAS Code: Upload Provider Data

```
DATA OUT.TRUNC_PROVIDER
  (KEEP = DUNS Vr_Sw);
  SET OUT.&fname.;
RUN;

PROC PRINT DATA = OUT.TRUNC_PROVIDER;
RUN;

PROC SORT DATA = OUT.TRUNC_PROVIDER (RENAME = (Vr_Sw=Vocational_Rehabilitation_Switch))
  OUT = OUT.PROVIDER
  NODUP;
  BY DUNS;
RUN;

filename a "&outloc.\provider.txt";
data _null_;
set OUT.PROVIDER;
file a;
put DUNS Vocational_Rehabilitation_Switch;
run;

endsas;
```

## Appendix A.81

JCL/SAS Code: PROVFMT

```
//#6502PVF JOB (12510000,T715,,SAS,,ITC9FL),BRYCE-ALEX,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#7429
//*****
//* PROJECT: BUILD DAF21
//* PROJECT
//* DIRECTOR:  MATT URATO
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.TKT.PRDLIB(PROVFMT)
//*
//* DESCRIPTION: CREATE PROVIDER FORMAT FROM PROVIDER FILE
//*
//* DATE:      07/07/22 ALEX BRYCE
//* EMAIL:     ABRYCE@MATHEMATICA-MPR.COM
//*****
//STEP1 EXEC SAS9
//WORK  DD SPACE=(6160,(99000,99000),,,ROUND)
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,
//      VOL=(,,10)
//PROV  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PROVIDER.FL.V1,DISP=SHR
//LIBRARY DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.TKT.FMTLIB,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *

* READ PROVIDER FILE;
DATA PROVIDER;
INFILE PROV TRUNCOVER;
INPUT
@01 DUNS $9.
@11 Vocational_Rehabilitation_Switch $1.
;
IF DUNS="" THEN DELETE;
RUN;

PROC PRINT DATA=PROVIDER (OBS=10);
RUN;

DATA PROVIDER(KEEP=START LABEL FMTNAME);
  SET PROVIDER (RENAME=(DUNS=START
    Vocational_Rehabilitation_Switch=LABEL));
  * IN ORDER TO CREATE THE FORMAT THE VARIABLE CONTAINING THE VALUE
  TO BE FORMATTED MUST BE NAMED "START" AND THE VARIABLE CONTAINING
  THE FORMATTED VALUE MUST BE NAME "LABEL";
```

Appendix A.81

JCL/SAS Code: PROVFM

```
IF LABEL='Y' THEN LABEL='V';  
ELSE IF LABEL='N' THEN LABEL='E';
```

```
* A VARIABLE NAMED FMTNAME MUST BE CREATED TO STORE THE NAME OF THE  
  FORMAT --- THE NAME USED TO REFERENCE THE FORMAT;  
FMTNAME='$PROVTYPE';
```

```
OUTPUT;
```

```
*THE 'IF _N_=1' CONDITION IS NECESSARY SINCE WE WANT TO ADD ONE LINE  
  FOR ALL 'OTHER' VALUES WHEREVER THEY APPEAR, THE TOTAL N OF  
  RECORDS IN THE OUTPUT DATA SET IS N+1;
```

```
IF _N_=1 THEN DO;  
  START='OTHER';  
  LABEL=' ';  
  OUTPUT;  
END;
```

```
RUN;
```

```
* CREATE THE FORMAT FROM THE FILE CREATED IN PREVIOUS DATASTEP;  
PROC FORMAT CNTLIN=PROVIDER LIBRARY=LIBRARY FMTLIB;  
RUN;
```

```
* CHECK THAT THE FORMAT IS WORKING CORRECTLY BY PRINTING THE FORMATED  
  VALUE OF "START" FROM THE CNTLIN FILE AND THE VALUE OF "LABEL". IF  
  THE FORMAT CREATION WAS SUCCESSFULL THESE VALUES WILL BE THE SAME;  
PROC PRINT DATA=PROVIDER;  
  VAR START LABEL;  
  FORMAT START $PROVTYPE. ;  
RUN;
```

## Appendix A.82

JCL/SAS Code: DCFTKT

```
//#6502DCF JOB (12510000,T715,,SAS,,ITC9TH),BRYCE-A,CLASS=T,
// MSGCLASS=1,MSGLEVEL=(1,1),REGION=4096K,NOTIFY=#6502
/*SSAMAIN SASBASIC
//*****
//* PROJECT: EVALUATION OF THE TICKET TO WORK PROGRAM-PART A
//*   EVALUATION IMPLEMENTATION (06979) DAF
//* PROJECT
//* DIRECTOR:  MATT URATO
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.TKT.PRDLIB(DCFEXT)
//*
//* DESCRIPTION: CREATE DCF EXTRACTS FOR DAF TICKET FILE
//*
//* DATE:     05/27/22 BY ALEX BRYCE
//*
//* EMAIL:    ABRYCE@MATHEMATICA-MPR.COM
//* PHONE:    202-552-6454
//*****
//S1 EXEC SAS9,LOAD='DBP8.DB2.SDSNLOAD',SORT='15000',
// WORK='15000,15000',COND=EVEN
//TKTASGN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFTKTA.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=TSILO,
// VOL=(,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HTKTASGN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFHTKTA.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=AFF=TKTASGN,
// VOL=(,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//TKT  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFTKT.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=AFF=TKTASGN,
// VOL=(,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HTKT  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFHTKT.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=AFF=TKTASGN,
// VOL=(,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//TKTTITLE DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFTKTT.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=AFF=TKTASGN,
// VOL=(,,255),
```

## Appendix A.82

JCL/SAS Code: DCFTKT

```
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HTKTTITL DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFHTKTT.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=AFF=TKTASGN,
// VOL=(,,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//TKTMNTH DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFTKTM.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=AFF=TKTASGN,
// VOL=(,,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HTKTMNTH DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFHTKTM.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=AFF=TKTASGN,
// VOL=(,,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//CLNT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFCLNT.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=AFF=TKTASGN,
// VOL=(,,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HCLNT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFHCLNT.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=AFF=TKTASGN,
// VOL=(,,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//TKTST DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFTKTST.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=AFF=TKTASGN,
// VOL=(,,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//SYSIN DD *
/*NOTE: DATASETS PREFIXED WITH AN "H" ARE HISTORICAL FILES
AND ARE CREATED FOR INTERNAL DOCUMENTATION PURPOSES ONLY
HTKTASGN, HTKT, HTKTTITL, HTKTMNTH, HCLNT*/
```

```
OPTIONS NOCENTER;
OPTIONS OBS=MAX;
```

```
*THE LIBNAME STATEMENT BELOW REFERS TO THE DCF DB2 TABLES,
INPUT DATA IS SPECIFIED IN THAT LIBNAME STATEMENT;
```

```
LIBNAME MDCF DB2 SSID=DBP8 SCHEMA=MDCF;
```

```
*WE ARE CREATING SNAPSHOTS OF THIS DATA SO WE DO NOT WANT TO DO ANYTHING
```

## Appendix A.82

JCL/SAS Code: DCFTKT

THAT CHANGES THE NUMBER OF OBS OR VARS CONTAINED IN THE DATASET  
(I.E. DROP, KEEP, WHERE, ETC.);

PROC SQL;

```
CREATE TABLE TKTASGN.TKTASGN AS
SELECT * FROM MDCF.TKTASGN
/*ORDER BY REPLACES SORT INITIAL SORT IN TKTMLASG*/
ORDER BY COSSN, TKT_NUM;
```

```
CREATE TABLE HTKTASGN.HTKTASGN AS
SELECT * FROM MDCF.HTKTASGN;
```

```
CREATE TABLE TKT.TKT AS
SELECT * FROM MDCF.TKT
/*ORDER BY REPLACES SORT INITIAL SORT IN TKTMLASG*/
ORDER BY COSSN, TKT_NUM;
```

```
CREATE TABLE HTKT.HTKT AS
SELECT * FROM MDCF.HTKT;
```

```
CREATE TABLE TKTTITLE.TKTTITLE AS
SELECT * FROM MDCF.TKTTITLE;
```

```
CREATE TABLE HTKTTITL.HTKTTITL AS
SELECT * FROM MDCF.HTKTTITL;
```

```
CREATE TABLE TKTMNTH.TKTMNTH AS
SELECT * FROM MDCF.TKTMNTH
/*ORDER BY REPLACES SORT INITIAL SORT IN TKTTITLE*/
ORDER BY COSSN, TKT_STMDT;
```

```
CREATE TABLE HTKTMNTH.HTKTMNTH AS
SELECT * FROM MDCF.HTKTMNTH;
```

```
CREATE TABLE CLNT.CLNT AS
SELECT * FROM MDCF.CLNT;
```

```
CREATE TABLE HCLNT.HCLNT AS
SELECT * FROM MDCF.HCLNT;
```

```
CREATE TABLE TKTST.TKTSENT AS
```

Appendix A.82

JCL/SAS Code: DCFTKT

```
SELECT * FROM MDCF.TKTSENT;

QUIT;

PROC CONTENTS DATA=TKTST.TKTSENT;
PROC CONTENTS DATA=TKTASGN.TKTASGN;
PROC CONTENTS DATA=HTKTASGN.HTKTASGN;
PROC CONTENTS DATA=TKT.TKT;
PROC CONTENTS DATA=HTKT.HTKT;
PROC CONTENTS DATA=TKTTITLE.TKTTITLE;
PROC CONTENTS DATA=HTKTTITL.HTKTTITL;
PROC CONTENTS DATA=TKTMNTH.TKTMNTH;
PROC CONTENTS DATA=HTKTMNTH.HTKTMNTH;
PROC CONTENTS DATA=CLNT.CLNT;
PROC CONTENTS DATA=HCLNT.HCLNT;
RUN;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: ABRYCE@MATHEMATICA-MPR.COM
TO: ABRYCE@MATHEMATICA-MPR.COM
SUBJECT: DCFEXT (DAF21)

DCFEXT HAS FINISHED RUNNING
```

## Appendix A.83

JCL/SAS Code: TKTTITLE

```

//#6502TTL JOB (12510000,T715,,SAS,,ITC9FL),BRYCE-A,
//  MSGCLASS=1,MSGLEVEL=(1,1)
//*****
//* PROJECT: BUILD DAF (50214)
//* PROJECT
//* DIRECTOR:  MATT URATO
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.TKT.PRDLIB(TKTTITLE)
//*
//* DESCRIPTION: CREATE TICKET TITLE & IN-USE DATA
//*           THIS PROGRAM MODIFIES THE ORIGINAL TKTTITLE PROGRAM
//*           BY ADDING MACRO VARIABLE PROCESSING
//*           (DOES NOT CREATE A MACRO)
//* DATE:     06/03/21 BY BRITTANY COLIP
//* EMAIL:    ABRYCE@MATHEMATICA-MPR.COM
//*****
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFTKTM.SA.V1,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.TKTTITLE.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;
%GLOBAL BEGYR ENDYR;
%LET BEGYR=2002;
%LET ENDYR=2021; * CHANGE WHEN NEEDED;

DATA TEMP1.TKTMNTH(KEEP=COSSN TKT_STMDT SSACT_PRTCPN_CD YYMM
TKT_INUSE_S:) TEMP1.TKTDUP (KEEP=COSSN TKT_STMDT);/*TKTDUP FOR DUPS ONLY*/
  SET IN1.TKTMNTH (RENAME=(TKT_INUSE_SW=TKT_INUSE_SWC));
  WHERE SSACT_PRTCPN_CD NE '0';
  BY COSSN TKT_STMDT;

  * CREATE VARIABLE CONTAINING THE YYMM OF THE EFFECTIVE DATE;
  * THE YYMM VARIABLE IS USED AS AN ID IN THE TRANSPOSE PROCEDURE

  LENGTH YYMM $4.;
  YYMM=PUT(TKT_STMDT,YYMMDD4.);

```



## Appendix A.83

JCL/SAS Code: TKTTITLE

```
*CONVER TO NUMERIC;
IF TKT_INUSE_SWC='Y' THEN TKT_INUSE_SW=1;
ELSE IF TKT_INUSE_SWC='N' THEN TKT_INUSE_SW=0;

* KEEP ALL UNIQUE COMBINATIONS OF COSSN AND TKT_STMDT;
* IF NOT A UNIQUE COMBO THEN KEEP THE FIRST OBS;
IF FIRST.TKT_STMDT THEN OUTPUT TEMP1.TKTMNTH;
ELSE OUTPUT TEMP1.TKTDUP;
RUN;
PROC SORT DATA=TEMP1.TKTMNTH;
  BY COSSN YYMM;
RUN;

PROC FREQ DATA=TEMP1.TKTMNTH;
TABLES TKT_INUSE_SW*TKT_INUSE_SWC/LIST MISSING;
RUN;

* CREATE ONE RECORD PER COSSN WITH MONTHLY TITLE VARIABLES;
PROC TRANSPOSE DATA=TEMP1.TKTMNTH OUT=TEMP1.TKTTITLE PREFIX=TITLE_MO;
  BY COSSN;
  ID YYMM;
  VAR SSACT_PRTCPN_CD;
RUN;

* CREATE ONE RECORD PER COSSN WITH MONTHLY INUSE VARIABLES;
PROC TRANSPOSE DATA=TEMP1.TKTMNTH OUT=TEMP1.TKTINUSE PREFIX=INUSE_MO;
  BY COSSN;
  ID YYMM;
  VAR TKT_INUSE_SW;
RUN;

* CREATE MACRO TKTTITLE SO THAT %DO PROCESSING WILL EXECUTE;

%MACRO TKTTITLE();
* POPULATE ALL MONTHLY TITLE & IN-USE VARIABLES;
DATA OUT.TKTTITLE;
  MERGE TEMP1.TKTTITLE(IN=A)
        TEMP1.TKTINUSE(IN=B)
  ;
  BY COSSN;

  /*TITLE_MO AND CINUSE_MO ARE INPUT ARRAYS*/
  /*WE'RE NOT GOING TO USE YYMM MACRO TO KEEP IT SIMPLE*/
```

## Appendix A.83

JCL/SAS Code: TKTTITLE

```
ARRAY TITLE_MO $
      %DO YEAR=&BEGYR %TO &ENDYR;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      TITLE_MO&YR.01-TITLE_MO&YR.12
      %END;
;

ARRAY CINUSE_MO
      %DO YEAR=&BEGYR %TO &ENDYR;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      INUSE_MO&YR.01-INUSE_MO&YR.12
      %END;
;

* IF THE CURRENT MONTH IS NOT BLANK AND THE SUBSEQUENT
  MONTH IS BLANK THEN POPULATE THE SUBSEQUENT MONTH
  WITH THE CURRENT MONTH. THIS WILL POPULATE ALL MONTHS
  FROM THE FIRST MONTH OF EFFECTIVE DATE UNTIL THE
  MONTH THE VARIABLE CHANGES.;
DO I=1 TO DIM(TITLE_MO)-1;

  IF TITLE_MO(I) NE " AND TITLE_MO(I+1)="
  THEN TITLE_MO(I+1)=TITLE_MO(I);

  IF CINUSE_MO(I) NE . AND CINUSE_MO(I+1)=.
  THEN CINUSE_MO(I+1)=CINUSE_MO(I);

END;

RUN;

PROC CONTENTS DATA=OUT.TKTTITLE;
RUN;

PROC MEANS DATA=OUT.TKTTITLE N MIN MEAN MAX;
VAR INUSE_MO;;
RUN;

%MEND;

%TKTTITLE();
ENDSAS;
//*****
```

Appendix A.83

JCL/SAS Code: TKTTITLE

```
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: ABRYCE@MATHEMATICA-MPR.COM
TO: ABRYCE@MATHEMATICA-MPR.COM
SUBJECT: TKTTITLE

TKTTITLE HAS FINISHED RUNNING
```

## Appendix A.84

JCL/SAS Code: TKTMLASG

```
//#6502TKM JOB (12510000,T715,,SAS,,ITC9FL),BRYCE-A,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT: BUILD DAF21 (50214)
//* PROJECT
//* DIRECTOR:  MATT URATO
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.TKT.PRDLIB(TKTMLASG)
//*
//* DESCRIPTION: CREATE TICKET MAILINGS & ASSIGNMENT PORTIONS OF DAF
//*
//* DATE: 07/07/2022 ALEX BRYCE
//* EMAIL: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//TKTML EXEC SAS9,
//  WORK='120000,60000'
//*
//TKT  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFTKT.SA.V1,DISP=SHR
//TKTA DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFTKTA.SA.V1,DISP=SHR
//CLNT DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFCLNT.SA.V1,DISP=SHR
//TKTST DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFTKTST.SA.V1,DISP=SHR
//MLAS DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.TKTMLASG.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//NOPROV DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.NOPROV.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  SPACE=(CYL,(500,10),RLSE)
//BPRE  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.BLDPRETK.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//LIBRARY DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.TKT.FMTLIB,DISP=SHR
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP3 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP4 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP5 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *
```

Appendix A.84

JCL/SAS Code: TKTMLASG

```
OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;
```

```
PROC SORT DATA=TKT.TKT OUT=TEMP1.TKT;  
  BY COSSN TKT_NUM MAILD_DT TKT_STUS_DT;  
RUN;
```

```
PROC SORT DATA=TKTA.TKTASGN OUT=TEMP2.TKTASGN;  
  BY COSSN TKT_NUM ASGND_DT;  
RUN;
```

```
PROC SORT DATA=TKTST.TKTSENT OUT=TEMP3.TKTSENT;  
  BY COSSN TKT_NUM NTC_VER_NUM DESCENDING LU_TS;  
  WHERE NTC_VER_NUM NE 2001;  
RUN;
```

```
PROC SORT DATA=TKTST.TKTSENT OUT=TEMP3.TKTSENT1;  
  BY COSSN TKT_NUM NTC_VER_NUM NTCDT;  
  WHERE NTC_VER_NUM = 2001;  
RUN;
```

```
PROC SORT DATA=TEMP3.TKTSENT NODUPKEYS;  
  BY COSSN TKT_NUM NTC_VER_NUM;  
RUN;
```

```
PROC SORT DATA=TEMP3.TKTSENT1 NODUPKEYS;  
  BY COSSN TKT_NUM NTC_VER_NUM;  
RUN;
```

```
PROC MEANS DATA=TEMP3.TKTSENT MIN MAX MEAN;  
VAR NTCDT;  
CLASS NTC_VER_NUM;  
RUN;
```

```
PROC MEANS DATA=TEMP3.TKTSENT1 MIN MAX MEAN;  
VAR NTCDT;  
CLASS NTC_VER_NUM;  
RUN;
```

```
PROC TRANSPOSE DATA=TEMP3.TKTSENT OUT=TEMP3.TKSENTA;  
ID NTC_VER_NUM;  
BY COSSN TKT_NUM;  
VAR NTCDT;
```

## Appendix A.84

JCL/SAS Code: TKTMLASG

```
RUN;

PROC TRANSPOSE DATA=TEMP3.TKTSNT1 OUT=TEMP3.TKTSNTB;
ID NTC_VER_NUM;
BY COSSN TKT_NUM;
VAR NTCDT;
RUN;

PROC CONTENTS DATA=TEMP3.TKTSNTA;RUN;
PROC CONTENTS DATA=TEMP3.TKTSNTB;RUN;

DATA TEMP4.TKTMLASG;
  MERGE TEMP1.TKT (IN=A)
        TEMP2.TKTASGN
        TEMP3.TKTSNTA (IN=B DROP=_NAME__LABEL_)
        TEMP3.TKTSNTB (IN=C DROP=_NAME__LABEL_)
  ;
  BY COSSN TKT_NUM;
  IF A;
  IF A THEN IN_TKT=1;
  IF B THEN IN_TKS=1;
  IF C THEN IN_TKS2=1;

  IF _1>.Z THEN MAIL_DT=_1;
  IF _5001>.Z THEN MAIL_DT=_5001;

RUN;

PROC FREQ DATA=TEMP4.TKTMLASG;
TABLES IN_TKT*IN_TKS*IN_TKS2 / LIST MISSING;
RUN;

PROC MEANS DATA=TEMP4.TKTMLASG;
VAR MAIL_DT _1 _5001;
WHERE _1>.Z OR _5001>.Z;
RUN;

/*SELECT THE NUMBER OF ARRAY MEMBERS, BASED ON N OF RECS PER SSN*/

PROC SQL NOPRINT;
  SELECT MAX(TKT_COUNT) INTO :MAXTKTN
  FROM (SELECT COUNT(COSSN) AS TKT_COUNT
        FROM TEMP4.TKTMLASG
```

Appendix A.84

JCL/SAS Code: TKTMLASG

```

GROUP BY COSSN);
QUIT;
%LET MAXTKTNUM = %LEFT(&MAXTKTN.);
%PUT FOUND &MAXTKTNUM AS THE MAXIMUM NUMBER OF RECS PER SSN;

DATA MLAS.TKTMLASG(KEEP=TKTMAILDDT: TKTTERMDDT: TKTASGNDDT: TKTUNASGDT:
    TKTSELDT: DUNS: PROVTYPE: PMTTYE: NOE COSSN
    NUMBER: TKTINTMLDT: TKTADHMLDT: TKT1YRMLDT:
    TKT3YRMLDT: TKTCUPMLDT;)

    /*STORE RECORDS WITH DUNS AND NO PROVIDER TYPE IN NOPROV
    DATASET*/
NOPROV.NOPROV(KEEP=DUNS: PROVTYPE: MAX_NOTYPE FLAG_NOTYPE:);

RETAIN TKTMAILDDT1-TKTMAILDDT&MAXTKTNUM

    TKTINTMLDT1-TKTINTMLDT&MAXTKTNUM
    TKTADHMLDT1-TKTADHMLDT&MAXTKTNUM
    TKT1YRMLDT1-TKT1YRMLDT&MAXTKTNUM
    TKT3YRMLDT1-TKT3YRMLDT&MAXTKTNUM
    TKTCUPMLDT1-TKTCUPMLDT&MAXTKTNUM

    TKTTERMDDT1-TKTTERMDDT&MAXTKTNUM
    TKTASGNDDT1-TKTASGNDDT&MAXTKTNUM
    TKTUNASGDT1-TKTUNASGDT&MAXTKTNUM
    TKTSELDT1-TKTSELDT&MAXTKTNUM
    DUNS1-DUNS&MAXTKTNUM
    PROVTYPE1-PROVTYPE&MAXTKTNUM
    FLAG_NOTYPE1-FLAG_NOTYPE&MAXTKTNUM
    MAX_NOTYPE
    PMTTYE1-PMTTYE&MAXTKTNUM
    NOE
    NUMBER1-NUMBER&MAXTKTNUM
;

SET TEMP4.TKTMLASG
;
BY COSSN;

* OUTPUT ARRAYS;
ARRAY TKTMAILDDT  TKTMAILDDT1- TKTMAILDDT&MAXTKTNUM;

ARRAY TKTINTMLDT  TKTINTMLDT1- TKTINTMLDT&MAXTKTNUM;
ARRAY TKTADHMLDT  TKTADHMLDT1- TKTADHMLDT&MAXTKTNUM;

```

## Appendix A.84

JCL/SAS Code: TKTMLASG

```
ARRAY TKT1YRMLDT TKT1YRMLDT1- TKT1YRMLDT&MAXTKTNUM;  
ARRAY TKT3YRMLDT TKT3YRMLDT1- TKT3YRMLDT&MAXTKTNUM;  
ARRAY TKTCUPMLDT TKTCUPMLDT1- TKTCUPMLDT&MAXTKTNUM;
```

```
ARRAY TKTTERMDDT TKTTERMDDT1- TKTTERMDDT&MAXTKTNUM;  
ARRAY TKTASGNDDT TKTASGNDDT1- TKTASGNDDT&MAXTKTNUM;  
ARRAY TKTUNASGDT TKTUNASGDT1- TKTUNASGDT&MAXTKTNUM;  
ARRAY TKTSELDT TKTSELDT1- TKTSELDT&MAXTKTNUM;  
ARRAY DUNSA $9. DUNS1- DUNS&MAXTKTNUM;  
ARRAY PMTTYPER $ PMTTYPER1- PMTTYPER&MAXTKTNUM;  
ARRAY PROVTYPE $ PROVTYPE1- PROVTYPE&MAXTKTNUM;  
ARRAY FLAG_NOTYPE FLAG_NOTYPE1-FLAG_NOTYPE&MAXTKTNUM;  
ARRAY NUMBER NUMBER1- NUMBER&MAXTKTNUM;
```

```
* INITIALIZE VARIABLES WHEN ENCOUNTERING A NEW SSN;
```

```
IF FIRST.COSSN THEN DO;
```

```
  DO I=1 TO &MAXTKTNUM;
```

```
    TKTMAILDDT(I)=.;
```

```
    TKTINTMLDT(I)=.;
```

```
    TKTADHMLDT(I)=.;
```

```
    TKT1YRMLDT(I)=.;
```

```
    TKT3YRMLDT(I)=.;
```

```
    TKTCUPMLDT(I)=.;
```

```
    TKTTERMDDT(I)=.;
```

```
    TKTASGNDDT(I)=.;
```

```
    TKTUNASGDT(I)=.;
```

```
    TKTSELDT(I)=.;
```

```
    DUNSA(I)='';
```

```
    PMTTYPER(I)='';
```

```
    PROVTYPE(I)='';
```

```
    FLAG_NOTYPE(I)=.;
```

```
    NUMBER(I)=.;
```

```
  END;
```

```
  NOE=0;
```

```
END;
```

```
* SET NUMBER OF ENTRIES (NOE) VARIABLE;
```

```
NOE=NOE+1;
```

```
* POPULATE THE TICKET ACTIVE VARIABLES;
```

```
TKTMAILDDT(NOE)=MAILD_DT;
```



## Appendix A.84

JCL/SAS Code: TKTMLASG

```
TKTINTMLDT(NOE)=_1;
TKTADHMLDT(NOE)=_2001;
TKT1YRMLDT(NOE)=_3001;
TKT3YRMLDT(NOE)=_4001;
TKTCUPMLDT(NOE)=_5001;

TKTASGNDDT(NOE)=ASGND_DT;
TKTUNASGDT(NOE)=UNASGND_DT;
TKTSELDT(NOE)=SLTD_DT;
  DUNSA(NOE)=DUNS;
  PMTTYPER(NOE)=TKT_PMT_TYP;
  PROVTYPE(NOE)=PUT(DUNSA(NOE),$PROVTYPE.);
  IF DUNSA(NOE) NE "" AND PROVTYPE(NOE)="" THEN FLAG_NOTYPE(NOE)=1;
IF TKT_STUS_CD='T' THEN TKTTERMDDT(NOE)=TKT_STUS_DT;
NUMBER(NOE)=TKT_NUM;

MAX_NOTYPE=MAX(OF FLAG_NOTYPE:);
IF MAX_NOTYPE = 1 THEN OUTPUT NOPROV.NOPROV;

  *IF THE LAST RECORD FOR THIS SSN THEN OUTPUT;
IF LAST.COSSN THEN OUTPUT MLAS.TKTMLASG;

  DROP DUNS; * ADDED FOR DAF12 (PER ASSESS DATA NEEDS);
RUN;

PROC CONTENTS DATA=MLAS.TKTMLASG;
RUN;

PROC FREQ DATA=NOPROV.NOPROV;
TITLE1 'RECORDS WITH NON-MISSING DUNS AND MISSING PROVIDER TYPE';
TABLES FLAG_NOTYPE: ;
RUN;

PROC PRINT DATA=NOPROV.NOPROV (OBS=50);
TITLE1 'SAMPLE PRINT (50 OBS)';
TITLE2 'RECORDS WITH NON-MISSING DUNS AND MISSING PROVIDER TYPE';
RUN;
TITLE2;

TITLE1 'SAMPLE PRINT:TKTMLASG(OBS=50)';
PROC PRINT DATA=MLAS.TKTMLASG(OBS=50);
  FORMAT TKTMAILDDT: TKTASGNDDT: TKTUNASGDT: TKTTERMDDT:
    TKTINTMLDT: TKTADHMLDT: TKT1YRMLDT: TKT3YRMLDT: TKTCUPMLDT:
    YYMMDD10.;
RUN;
```

## Appendix A.84

JCL/SAS Code: TKTMLASG

```
*****;
* END TKTMLASG, BEGIN BLDPRETK;
*****;

* SORT DCF CLIENT DATASET AND KEEP ONLY WHERE THE COSSN IS THE
BENEFICIARIES BOAN;
  /*THIS WHERE SYNTAX DOES NOT WORK ON PC SAS9.3... */
PROC SORT DATA=CLNT.CLNT NODUPKEYS OUT=TEMP2.CLNT
  (KEEP=T16_DOB T2_DOB T16_DOD T2_DOD COSSN);
  WHERE CID='00'; /* COSSN IS THE BOAN */
  BY COSSN;
RUN;

* COMBINE TICKET MAIL, ASSIGNMENT, & TITLE FILE WITH CLNT FILE TO
OBTAIN THE DOB & DOD;
DATA BPRE.BLDPRETK(DROP=T16_DOB T2_DOB T16_DOD T2_DOD);
  MERGE MLAS.TKTMLASG(IN=TKT)
    TEMP2.CLNT(IN=CLNT);
  BY COSSN;
  IF TKT;

  IF CLNT THEN INCLNT=1;
  ELSE INCLNT=0;

* DETERMINE DOBTKT;
DOBTKT=T16_DOB;
IF DOBTKT=. THEN DOBTKT=T2_DOB;

* DETERMINE DODTKT;
DODTKT=T16_DOD;
IF DODTKT=. THEN DODTKT=T2_DOD;

***** CALCULATE FULL RETIREMENT AGE *****;

* ADJUST RETIREMENT DATE FOR BENEFICIARIES BORN ON JAN 1;
IF MONTH(DOBTKT)=1 AND DAY(DOBTKT)=1 THEN
  DOBRETIRE=INTNX('DAY',DOBTKT,-1);
ELSE DOBRETIRE=DOBTKT;

* SET FRA;
  IF YEAR(DOBRETIRE)<=1937 THEN FRA=780/12; /* 65 */
  ELSE IF YEAR(DOBRETIRE) =1938 THEN FRA=782/12; /* 65 & 2 M */
  ELSE IF YEAR(DOBRETIRE) =1939 THEN FRA=784/12; /* 65 & 4 M */
  ELSE IF YEAR(DOBRETIRE) =1940 THEN FRA=786/12; /* 65 & 6 M */
```

## Appendix A.84

JCL/SAS Code: TKTMLASG

```

ELSE IF YEAR(DOBRETIRE) =1941 THEN FRA=788/12; /* 65 & 8 M */
ELSE IF YEAR(DOBRETIRE) =1942 THEN FRA=790/12; /* 65 & 10 M */
ELSE IF 1943<=YEAR(DOBRETIRE)<=1954 THEN FRA=792/12; /* 66 */
ELSE IF YEAR(DOBRETIRE) =1955 THEN FRA=794/12; /* 66 & 2 M */
ELSE IF YEAR(DOBRETIRE) =1956 THEN FRA=796/12; /* 66 & 4 M */
ELSE IF YEAR(DOBRETIRE) =1957 THEN FRA=798/12; /* 66 & 6 M */
ELSE IF YEAR(DOBRETIRE) =1958 THEN FRA=800/12; /* 66 & 8 M */
ELSE IF YEAR(DOBRETIRE) =1959 THEN FRA=802/12; /* 66 & 10 M */
ELSE IF YEAR(DOBRETIRE)>=1960 THEN FRA=804/12; /* 67 */

RUN;

TITLE1;
TITLE2;
PROC CONTENTS DATA=BPRES.BLDPRETK;
RUN;
TITLE1 'SAMPLE PRINT:TKTSELDT WHERE TKTSELDT2 NE . (OBS=100)';
PROC PRINT DATA=BPRES.BLDPRETK(OBS=50);
  WHERE TKTSELDT2 NE .;
  VAR TKTMAILDDT1-TKTMAILDDT2 TKTSELDT1-TKTSELDT2;
  FORMAT TKTMAILDDT: TKTSELDT: YMMDD10.;
RUN;

TITLE1 'SAMPLE PRINT:TKTSELDT WHERE TKTSELDT5 NE . (OBS=100)';
PROC PRINT DATA=BPRES.BLDPRETK(OBS=50);
  WHERE TKTSELDT5 NE .;
  VAR TKTMAILDDT1-TKTMAILDDT5 TKTSELDT1-TKTSELDT5;
  FORMAT TKTMAILDDT: TKTSELDT: YMMDD10.;
RUN;

TITLE1 'SAMPLE PRINT:BLDPRETK (OBS=50)';
PROC PRINT DATA=BPRES.BLDPRETK(OBS=50);
  FORMAT TKTMAILDDT: TKTASGNDDT: TKTUNASGDT: TKTERMDDT: DOBTKT
  DODTKT TKTSELDT: YMMDD10.;
RUN;

ENDSAS;
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *

```

Appendix A.84

JCL/SAS Code: TKTMLASG

```
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: ABRYCE@MATHEMATICA-MPR.COM
TO: ABRYCE@MATHEMATICA-MPR.COM
SUBJECT: TKTMLASG
```

TKTMLASG HAS FINISHED RUNNING

## Appendix A.85

JCL/SAS Code: CRTKTFLG

```

//#6502CTK JOB (12510000,T715,,SAS,,ITC9FL),BRYCE-A,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#7429
//*****
/* PROJECT: BUILD DAF (50214)
/*
/* PROJECT
/* DIRECTOR:  MATT URATO
/*
/* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.TKT.PRDLIB(CRTKTFLG)
/*
/* DESCRIPTION: CREATE TICKET PORTION OF TRF
/*
/* DATE:      06/15/21 ALEX BRYCE
/* EMAIL:     ABRYCE@MATHEMATICA-MPR.COM
/*
/* PURPOSE:   CREATE TICKET PORTION OF THE TICKET RESEARCH FILE.
/* PROGRAM CREATES MONTHLY FLAGS (ELIGIBLE, ACTIVE, PMTTYPE,
/* PROVTYPE, ETC. - SEE SPECS "VARIABLES FOR TICKET MASTER
/* FILE, TICKET PORTION ONLY") BASED ON DCF TKT DATA (TKTNUM,
/* TKTMAILDDT, TKTASGNDT, TKTUNASGDT, TKTERMDT, ETC.).
//*****
/*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
/*
//IN1  DD  DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.BLDPRETK.SA.V1,DISP=SHR
//OUT  DD  DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CRTKTFLG.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN  OBS=MAX;

TITLE1 "BUILD DAF21";
TITLE2 "TKT FLAG CREATION";
TITLE3;

/* change endyr and endmmn for each run */
%LET BEGYR=2002;
%LET ENDYR=2021; /* change as needed */

/* step to assign macro variables to handle time series data */

```

## Appendix A.85

JCL/SAS Code: CRTKTFLG

```

%MACRO TICKETFLAG;
%LET MOCNTER=1;
/*LOOPING THROUGH EACH YEAR SPECIFIED ABOVE*/
%DO YEAR=&BEGYR. %TO &ENDYR.;
/*SETTING YR MACRO VAR TO LAST TWO DIGITS OF EACH YEAR*/
%LET YR=%SUBSTR(&YEAR.,3,2);

/*LOOP THROUGH MONTHS 1 TO 12 AND INCREMENT MONTH COUNTER*/
%DO M=1 %TO 12;
%LET M=%SYSFUNC(PUTN(&M.,Z2.));
%LET YY&MOCNTER.=%EVAL(&YR.)%EVAL(&M.);
%LET MOCNTER=%EVAL(&MOCNTER.+1);
%END;
%END;
/*SET TOTAL MONTHS, TOTMOS, EQUAL TO MOCNTER - 1 TO GET
TOTAL MONTHS IN TIME SERIES*/
%LET TOT=%EVAL(&MOCNTER.-1);

/*USED TO PRINT MAPPING TO LOG - CHECK DURING CODE REVIEW*/
/*USING TOT HERE INSTEAD OF TOTMOS TO ACCOMODATE VERY LONG
ARRAY NAMES BELOW...*/
%DO MM=1 % TO &TOT.;
%PUT &MM. MAPS TO YYMM, &&YY&MM.;
%END;
%PUT TOTAL MONTHS = &TOT.;
/*GETTING MAXIMUM NUMBER OF ENTRIES*/
PROC SQL;
SELECT MAX(NOE) INTO :MAXNOE
SEPARATED BY ' '
FROM IN1.BLDPRETK;
QUIT;

%PUT &MAXNOE = MAXIMUM NUMBER OF ENTRIES;

DATA OUT.CRTKTFLG;
SET IN1.BLDPRETK;

ARRAY ELIGIBLE_MO (*) %DO MM=1 %TO &TOT; ELIGIBLE_MO&&YY&MM %END;;
ARRAY ACTIVE_MO (*) %DO MM=1 %TO &TOT; ACTIVE_MO&&YY&MM %END;;
ARRAY PMTTYPE_MO (*) $ %DO MM=1 %TO &TOT; PMTTYPE_MO&&YY&MM %END;;
ARRAY PROVTYPE_MO (*) $ %DO MM=1 %TO &TOT; PROVTYPE_MO&&YY&MM %END;;
ARRAY ELIGIBLE_EV (*) %DO MM=1 %TO &TOT; ELIGIBLE_EV&&YY&MM %END;;
ARRAY ACTIVE_EV (*) %DO MM=1 %TO &TOT; ACTIVE_EV&&YY&MM %END;;
ARRAY VR_EV (*) %DO MM=1 %TO &TOT; VR_EV&&YY&MM %END;;
ARRAY EN_EV (*) %DO MM=1 %TO &TOT; EN_EV&&YY&MM %END;;

```

## Appendix A.85

JCL/SAS Code: CRTKTFLG

```
ARRAY MAIL_MO (*) %DO MM=1 %TO &TOT; MAIL_MO&&YY&MM %END;;
ARRAY FIRST_ASGN_MO (*) %DO MM=1 %TO &TOT; FIRST_ASGN_MO&&YY&MM %END;;
ARRAY DEACTIVE_MO (*) %DO MM=1 %TO &TOT; DEACTIVE_MO&&YY&MM %END;;
ARRAY REASSIGN_MO (*) %DO MM=1 %TO &TOT; REASSIGN_MO&&YY&MM %END;;
ARRAY
  FIRST_ASGN_VR_MO (*) %DO MM=1 %TO &TOT;
    FIRST_ASGN_VR_MO&&YY&MM
  %END;;
ARRAY
  DEACTIVE_VR_MO (*) %DO MM=1 %TO &TOT;
    DEACTIVE_VR_MO&&YY&MM
  %END;;
ARRAY
  REASSIGN_VR_MO (*) %DO MM=1 %TO &TOT;
    REASSIGN_VR_MO&&YY&MM
  %END;;
ARRAY
  FIRST_ASGN_EN_MO (*) %DO MM=1 %TO &TOT;
    FIRST_ASGN_EN_MO&&YY&MM
  %END;;
ARRAY
  DEACTIVE_EN_MO (*) %DO MM=1 %TO &TOT;
    DEACTIVE_EN_MO&&YY&MM
  %END;;
ARRAY
  REASSIGN_EN_MO (*) %DO MM=1 %TO &TOT;
    REASSIGN_EN_MO&&YY&MM
  %END;;

ARRAY VR_TR_MO (*) %DO MM=1 %TO &TOT; VR_TR_MO&&YY&MM %END;;
ARRAY VR_MO_MO (*) %DO MM=1 %TO &TOT; VR_MO_MO&&YY&MM %END;;
ARRAY VR_OO_MO (*) %DO MM=1 %TO &TOT; VR_OO_MO&&YY&MM %END;;
ARRAY EN_MO_MO (*) %DO MM=1 %TO &TOT; EN_MO_MO&&YY&MM %END;;
ARRAY EN_OO_MO (*) %DO MM=1 %TO &TOT; EN_OO_MO&&YY&MM %END;;
ARRAY VR_TR_EV (*) %DO MM=1 %TO &TOT; VR_TR_EV&&YY&MM %END;;
ARRAY VR_MO_EV (*) %DO MM=1 %TO &TOT; VR_MO_EV&&YY&MM %END;;
ARRAY VR_OO_EV (*) %DO MM=1 %TO &TOT; VR_OO_EV&&YY&MM %END;;
ARRAY EN_MO_EV (*) %DO MM=1 %TO &TOT; EN_MO_EV&&YY&MM %END;;
ARRAY EN_OO_EV (*) %DO MM=1 %TO &TOT; EN_OO_EV&&YY&MM %END;;

ARRAY TKTNUM NUMBER1-NUMBER&MAXNOE;
ARRAY MAILDT TKTMAILDDT1-TKTMAILDDT&MAXNOE;
ARRAY TERMDT TKTTERMDDT1-TKTTERMDDT&MAXNOE;
ARRAY ASGNDT TKTASGNDDT1-TKTASGNDDT&MAXNOE;
ARRAY UNASSGDT TKTUNASGDT1-TKTUNASGDT&MAXNOE;
```

## Appendix A.85

JCL/SAS Code: CRTKTFLG

```
ARRAY PMTTY1 PMTTY1-PMTTY1&MAXNOE;
ARRAY PROVTYPE1 PROVTYPE1-PROVTYPE1&MAXNOE;

* SET BEGINNING AND ENDING DATES FOR EACH REPORTING MONTH;
* I.E. BEGDATE1=20020101 ENDDATE1=20020131;

FORMAT BEGDATE: ENDDATE: MMDDYY10.;

ARRAY BEGDATE (*) BEGDATE1-BEGDATE&TOT.;
ARRAY ENDDATE (*) ENDDATE1-ENDDATE&TOT.;
%LET ENDMN=12;

DO YR=&BEGYR. TO &ENDYR.;
  IF YR NE &ENDYR. THEN DO MO=1 TO 12;
    BEGDATE(((YR-&BEGYR.)*12)+MO)=MDY(MO,1,YR);
    ENDDATE(((YR-&BEGYR.)*12)+MO)
      =INTNX('MONTH',BEGDATE(((YR-&BEGYR.)*12)+MO),0,'END');
  END;
  ELSE DO MO=1 TO &ENDMN.;
    BEGDATE(((YR-&BEGYR.)*12)+MO)=MDY(MO,1,YR);
    ENDDATE(((YR-&BEGYR.)*12)+MO)
      =INTNX('MONTH',BEGDATE(((YR-&BEGYR.)*12)+MO),0,'END');
  END;
END;

* INITIALIZE EVER VARIABLES;
EVER_ELIG=0;
EVER_ACTIVE=0;
EVER_VR=0;
EVER_EN=0;
EVER_VR_TR=0;
EVER_VR_MO=0;
EVER_VR_OO=0;
EVER_EN_MO=0;
EVER_EN_OO=0;

*** LOOP THROUGH ONCE FOR EACH REPORTING MONTH;
DO RPTMO=1 TO DIM(BEGDATE);

  *** CALCULATE AGE FOR REPORTING MONTH;
  * OLD CALCULATION: AGE=YRDIF(DOBTKT,ENDDATE(RPTMO),'ACT/ACT');
  AGE=INTCK('MONTH',DOBTKT,ENDDATE(RPTMO),'C')/12;
```



## Appendix A.85

JCL/SAS Code: CRTKTFLG

```
*** INITIALIZE VARIABLES;
ELIGIBLE_MO(RPTMO)=0;
ACTIVE_MO(RPTMO)=0;
PMTTYPE_MO(RPTMO)=' ';
PROVTYPE_MO(RPTMO)=' ';
ELIGIBLE_EV(RPTMO)=0;
ACTIVE_EV(RPTMO)=0;
MAIL_MO(RPTMO)=0;
VR_EV(RPTMO)=0;
EN_EV(RPTMO)=0;
VR_TR_MO(RPTMO)=0;
VR_MO_MO(RPTMO)=0;
VR_OO_MO(RPTMO)=0;
EN_MO_MO(RPTMO)=0;
EN_OO_MO(RPTMO)=0;
FIRST_ASGN_MO(RPTMO)=0;
FIRST_ASGN_VR_MO(RPTMO)=0;
FIRST_ASGN_EN_MO(RPTMO)=0;
DEACTIVE_MO(RPTMO)=0;
DEACTIVE_VR_MO(RPTMO)=0;
DEACTIVE_EN_MO(RPTMO)=0;
REASSIGN_MO(RPTMO)=0;
REASSIGN_VR_MO(RPTMO)=0;
REASSIGN_EN_MO(RPTMO)=0;

*** FLAG TICKET MAILED IN REPORTING MONTH;

*** TICKET HAS BEEN MAILED IF ALL OF THE FOLLOWING ARE TRUE:
  1. THE MAIL DATE IS ON OR AFTER THE FIRST DAY OF THE REPORTING
      MONTH OR ON OR BEFORE THE LAST DAY OF THE REPORTING MONTH
  2. THE TICKET TERMINATION DATE IS MISSING OR IS AFTER THE MAIL
      DATE
  4. THE MAIL DATE IS NOT MISSING
;
/* TKTEVT=1; */
*** LOOP THRU TICKETS UNTIL MAILED TICKET IS FOUND OR UNTIL OUT OF
TICKETS STARTING WITH TICKET #1;
DO TKTEVT=1 TO NOE;
  IF BEGDATE(RPTMO)<=MAILDT(TKTEVT)<=ENDDATE(RPTMO)
    AND (TERMDT(TKTEVT)=. OR TERMDT(TKTEVT)>MAILDT(TKTEVT))
  THEN MAIL_MO(RPTMO)=1;
END;
```

Appendix A.85

JCL/SAS Code: CRTKTFLG

```
*** FLAG TICKET ELIGIBLE FOR REPORTING MONTH STARTING WITH TICKET
#1;
*** BENE IS "TICKET ELIGIBLE" IN REPORTING MONTH IF ALL OF THE
FOLLOWING CONDITIONS ARE MEET:
1. TICKET MAILED ON OR BEFORE LAST DAY OF REPORTING MONTH
2. TICKET HAD EITHER NOT BEEN TERMINATED OR, IF TERMINATED,
   TERMINATION DATE WAS LATER THAN THE MAIL DATE & ON OR AFTER
   THE FIRST DAY OF THE REPORTING MONTH
3. BENE WAS OLDER THAN EXACTLY 18 & YOUNGER THAN FRA ON THE
   LAST DAY OF REPORTING MONTH (SSA CONSIDERS A BENE TO BE AT
   RETIREMENT AGE IN THE MONTH THEY REACH FRA)
4. BENES DATE OF DEATH WAS MISSING OR, IF NOT MISSING, IT WAS
   AFTER THE LAST DAY OF THE REPORTING MONTH
;

*** LOOP THRU TICKETS UNTIL ELIGIBLE IS FOUND OR UNTIL OUT OF
TICKETS;
DO TKTEVT=1 TO NOE;

IF MAILDT(TKTEVT)<=ENDDATE(RPTMO)
  AND MAILDT(TKTEVT) NE .
  AND 18<AGE<FRA
  AND (DODTKT EQ . OR DODTKT>=BEGDATE(RPTMO))
  AND ((TERMDT(TKTEVT)>MAILDT(TKTEVT)
    AND TERMDT(TKTEVT)>=BEGDATE(RPTMO))
    OR TERMDT(TKTEVT)=.)
THEN DO;
  ELIGTKTEVT=TKTNUM(TKTEVT);
  ELIGIBLE_MO(RPTMO)=1;
  EVER_ELIG=1;
END;
END;
*** END ELIGIBLE LOOP;

*** FLAG ACTIVE TICKET IN REPORTING MONTH;
*** BENES HAVE AN "ACTIVE TICKET" IF THE BENE WAS "TICKET ELIGIBLE"
IN REPORTING MONTH & EITHER OF THE FOLLOWING CONDITIONS ARE
MEET:
1. TICKET WAS ASSIGNED BEFORE THE 1ST DAY OF REPORTING MONTH
  & IT EITHER HAD NOT BEEN UNASSIGNED OR TERMINATED, OR IF
  UNASSIGNED OR TERMINATED, IT WAS ON OR AFTER THE FIRST DAY
  OF THE MONTH
2. TICKET WAS ASSIGNED ON ANY DAY DURING THE MONTH
;
```

## Appendix A.85

JCL/SAS Code: CRTKFLG

```
*** IF ELIGIBLE TICKET THIS REPORTING MONTH START ACTIVE LOOP;
IF ELIGIBLE_MO(RPTMO)=1 THEN DO;
  *** LOOP THROUGH ELIGIBLE ASSIGNMENTS;
  DO ASGN=ELIGTKTEVT TO NOE UNTIL(ACTIVE_MO(RPTMO)=1);
  IF((ASGNDT(ASGN)<BEGDATE(RPTMO) AND ASGNDT(ASGN) NE .)
    AND (UNASSGDT(ASGN)>=BEGDATE(RPTMO) OR UNASSGDT(ASGN)=.)
    AND (TERMDT(ASGN)>=BEGDATE(RPTMO) OR TERMDT(ASGN)=.)) OR
    BEGDATE(RPTMO)<=ASGNDT(ASGN)<=ENDDATE(RPTMO) THEN DO;
    ACTIVE_MO(RPTMO)=1;
    EVER_ACTIVE=1;
  END;
END;
*** END ASSIGNMENT LOOP;
END;
*** END ACTIVE LOOP;

*** DETERMINE PROVIDER AND PAYMENT TYPES FOR ACTIVE TICKETS IN
REPORTING MONTH;
IF ACTIVE_MO(RPTMO)=1 THEN DO;

  *** FLAG PAYMENT TYPE FOR ACTIVE TICKET;
  PMTTYE_MO(RPTMO)=PMTTYP(ASGN);

  *** FLAG PROVIDER TYPE;
  PROVTYPE_MO(RPTMO)=PROVTYPE(ASGN);

  *** SVRAS;
  IF PROVTYPE_MO(RPTMO)='V' THEN DO;
    EVER_VR=1;

  *** FLAG PROVIDER/PAYMENT TYPE COMBOS;
  SELECT(PMTTYE_MO(RPTMO));
  WHEN ('V') DO; VR_TR_MO(RPTMO)=1; EVER_VR_TR=1; END;
  WHEN ('M') DO; VR_MO_MO(RPTMO)=1; EVER_VR_MO=1; END;
  WHEN ('O') DO; VR_OO_MO(RPTMO)=1; EVER_VR_OO=1; END;
  OTHERWISE DO;
    VR_TR_MO(RPTMO)=9;
    VR_MO_MO(RPTMO)=9;
    VR_OO_MO(RPTMO)=9;
  END;
END;

END;

*** ENS;
```

## Appendix A.85

JCL/SAS Code: CRTKTFLG

```
ELSE IF PROVTYPE_MO(RPTMO)='E' THEN DO;
  EVER_EN=1;

  *** FLAG PROVIDER/PAYMENT TYPE COMBOS;
  SELECT(PMTTYPE_MO(RPTMO));
  WHEN('M') DO; EN_MO_MO(RPTMO)=1; EVER_EN_MO=1; END;
  WHEN('O') DO; EN_OO_MO(RPTMO)=1; EVER_EN_OO=1; END;
  OTHERWISE DO;
    EN_MO_MO(RPTMO)=9;
    EN_OO_MO(RPTMO)=9;
  END;
END;
END;

ELSE PROVTYPE_MO(RPTMO)='X';
END;
*** END PROVIDER & PAYMENT TYPE LOOP;

*** SET MONTHLY EVER FLAGS;
IF EVER_ELIG=1 THEN ELIGIBLE_EV(RPTMO)=1;
ELSE ELIGIBLE_EV(RPTMO)=0;

IF EVER_ACTIVE=1 THEN ACTIVE_EV(RPTMO)=1;
ELSE ACTIVE_EV(RPTMO)=0;

IF EVER_VR=1 THEN VR_EV(RPTMO)=1;
ELSE VR_EV(RPTMO)=0;

IF EVER_VR_TR=1 THEN VR_TR_EV(RPTMO)=1;
ELSE VR_TR_EV(RPTMO)=0;

IF EVER_VR_MO=1 THEN VR_MO_EV(RPTMO)=1;
ELSE VR_MO_EV(RPTMO)=0;

IF EVER_VR_OO=1 THEN VR_OO_EV(RPTMO)=1;
ELSE VR_OO_EV(RPTMO)=0;

IF EVER_EN=1 THEN EN_EV(RPTMO)=1;
ELSE EN_EV(RPTMO)=0;

IF EVER_EN_MO=1 THEN EN_MO_EV(RPTMO)=1;
ELSE EN_MO_EV(RPTMO)=0;

IF EVER_EN_OO=1 THEN EN_OO_EV(RPTMO)=1;
ELSE EN_OO_EV(RPTMO)=0;
```

## Appendix A.85

JCL/SAS Code: CRTKTFLG

```
*** FLAG BENES FIRST ASSIGNMENTS;
* FOR 1ST REPORTING MONTH -- JAN 02;
IF RPTMO=1 AND ACTIVE_EV(RPTMO)=1 THEN DO;
  FIRST_ASGN_MO(RPTMO)=1;
  IF VR_TR_MO(RPTMO)=1 OR VR_MO_MO(RPTMO)=1 OR VR_OO_MO(RPTMO)=1
    THEN FIRST_ASGN_VR_MO(RPTMO)=1;
  ELSE IF EN_MO_MO(RPTMO)=1 OR EN_OO_MO(RPTMO)=1
    THEN FIRST_ASGN_EN_MO(RPTMO)=1;

  *** FLAGS FOR REASSIGNMENT PROCESSING;
  FIRSTASGNDT=ASGNDT(ASGN);
  FIRSTASGNMO=RPTMO;
END;
* FOR SUBSEQUENT REPORTING MONTHS -- AFTER JAN 02;
ELSE IF ACTIVE_EV(RPTMO)=1 AND ACTIVE_EV(RPTMO-1)=0 THEN DO;
  FIRST_ASGN_MO(RPTMO)=1;
  IF VR_TR_MO(RPTMO)=1 OR VR_MO_MO(RPTMO)=1 OR VR_OO_MO(RPTMO)=1
    THEN FIRST_ASGN_VR_MO(RPTMO)=1;
  ELSE IF EN_MO_MO(RPTMO)=1 OR EN_OO_MO(RPTMO)=1
    THEN FIRST_ASGN_EN_MO(RPTMO)=1;

  *** FLAGS FOR REASSIGNMENT PROCESSING;
  FIRSTASGNDT=ASGNDT(ASGN);
  FIRSTASGNMO=RPTMO;
END;
*** END FIRST ASSIGNMENT LOOP;

*** COUNT DEACTIVATIONS;
*** A TICKET HAS BEEN DEACTIVATED WHEN BENE HAS HAD AN ASSIGNMENT
    IN OR BEFORE REPORTING MONTH & AN UNASSIGNED DATE IN THE
    REPORTING MONTH OR IF UNASSIGNED DATE IS MISSING HAS A
    TERMINATION DATE IN THE REPORTING MONTH;

IF ACTIVE_EV(RPTMO)=1 THEN DO UNASGN=1 TO NOE;
  IF (UNASSGDT(UNASGN) NE .
    AND BEGDATE(RPTMO)<=UNASSGDT(UNASGN)<=ENDDATE(RPTMO))
    OR (UNASSGDT(UNASGN)=. AND TERMDT(UNASGN) NE .
    AND BEGDATE(RPTMO)<=TERMDT(UNASGN)<=ENDDATE(RPTMO))
  THEN DO;
    DEACTIVE_MO(RPTMO)=DEACTIVE_MO(RPTMO)+1;
    IF VR_TR_MO(RPTMO)=1 OR VR_MO_MO(RPTMO)=1
      OR VR_OO_MO(RPTMO)=1 THEN
      DEACTIVE_VR_MO(RPTMO)=DEACTIVE_VR_MO(RPTMO)+1;
    ELSE IF EN_MO_MO(RPTMO)=1 OR EN_OO_MO(RPTMO)=1 THEN
```

## Appendix A.85

JCL/SAS Code: CRTKTFLG

```

        DEACTIVE_EN_MO(RPTMO)=DEACTIVE_EN_MO(RPTMO)+1;
    END;
END;
*** END DEACTIVATIONS LOOP;

*** COUNT REASSIGNMENTS;
*** A TICKET HAS BEEN REASSIGNED WHEN BENE HAS HAD AN ASSIGNMENT IN
    OR BEFORE REPORTING MONTH & A DEACTIVATION IN OR BEFORE
    REPORTING MONTH & HAS A SUBSEQUENT VALID ASSIGNMENT DATE
    WITHIN REPORTING MONTH THAT IS NOT EQUAL TO BENES FIRST
    ASSIGNMENT DATE;

IF ACTIVE_EV(RPTMO)=1 THEN DO I=FIRSTASGNMO TO RPTMO;
IF DEACTIVE_MO(I)=1 THEN DO J=1 TO NOE;
    IF ASGNDT(J)>FIRSTASGNDT AND ASGNDT(J) NE .
        AND BEGDATE(RPTMO)<=ASGNDT(J)<=ENDDATE(RPTMO)
        AND (UNASSGDT(J)>BEGDATE(RPTMO) OR UNASSGDT(J)=.)
        AND (TERMDT(J)>BEGDATE(RPTMO) OR TERMDT(J)=.)
    THEN DO;
        REASSIGN_MO(RPTMO)=REASSIGN_MO(RPTMO)+1;
        IF VR_TR_MO(RPTMO)=1 OR VR_MO_MO(RPTMO)=1
        OR VR_OO_MO(RPTMO)=1 THEN
            REASSIGN_VR_MO(RPTMO)=REASSIGN_VR_MO(RPTMO)+1;
        ELSE IF EN_MO_MO(RPTMO)=1 OR EN_OO_MO(RPTMO)=1 THEN
            REASSIGN_EN_MO(RPTMO)=REASSIGN_EN_MO(RPTMO)+1;
    END;
END;
*** END REASSIGNMENTS LOOP;

END;
* END REPORTING MONTH LOOP;

* SET THE MINIMUM (FIRST) MAIL DATE;
MINMAIL=MIN(OFTKTMAILDDT1-TKTMAILDDT&MAXNOE);

DROP I J RPTMO TKTEVT ASGN UNASGN FRA; * DROP FRA ADDED PER DAF12;
    * ASSESS DATA NEEDS ;

/*NEW FOR DAF13 - ADDING TKTY COMMON VARIABLE FROM BLDVAR2*/

ARRAY ACTMOYMM(&BEGYR.:&ENDYR.,1:12)
    %DO YYYY=&BEGYR. %TO &ENDYR.;
    %LET YY=%SUBSTR(&YYYY.,3,2);
    ACTIVE_MO&YY.01-ACTIVE_MO&YY.12

```

Appendix A.85

JCL/SAS Code: CRTKTFLG

```

%END;;

ARRAY TKTY (&BEGYR.:&ENDYR.)

%DO YYYY=&BEGYR. %TO &ENDYR.;
  %LET YY=%SUBSTR(&YYYY.,3,2);
  TKT&YY.
%END;;

DO YY=&BEGYR. TO &ENDYR.;
  TKTY(YY) = 0; *SET INITIAL TKTY VALUE TO BE 0;
  DO MO=1 TO 12 UNTIL(TKTY(YY)=1);
    IF ACTMOYMM(YY,MO) = 1 THEN TKTY(YY)=1;
  END;
END;

RUN;

PROC CONTENTS DATA=OUT.CRTKTFLG POSITION;
RUN;

*** ADDED IN TRF10 - THE PRINT CODE IS REMOVED FROM THIS ***
*** PROGRAM IN ORDER TO AVOID INADVERTANT LOSS OF PII. ***;
*** THE CODE BELOW HAS BEEN MADE INTO A SEPARATE PROGRAM ***;
*PROC PRINT DATA=OUT.CRTKTFLG (OBS=25);
*RUN;

%MEND TICKETFLAG;

%TICKETFLAG;

ENDSAS;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA

```

Appendix A.85

JCL/SAS Code: CRTKTFLG

FROM: ABRYCE@MATHEMATICA-MPR.COM  
TO: ABRYCE@MATHEMATICA-MPR.COM  
SUBJECT: CRTKTFLG

CRTKTFLG HAS FINISHED RUNNING



## Appendix A.86

JCL/SAS Code: FINTKT

```
//#6502TSP JOB (12510000,T715,,SAS,,ITC9FL),BRYCE-ALEX,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT: BUILD DAF 50214
//* PROJECT
//* DIRECTOR:  MATT URATO
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.TKT.PRDLIB(FINTKT)
//*
//* DESCRIPTION: CREATE FINAL TICKET FILES (BASE AND ANNUALS)
//*
//* DATE:     09/28/21 ALEX BRYCE
//* EMAIL:    ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CRTKTFLG.SA.V1,DISP=SHR
//IN2  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.TKTTITLE.SA.V1,DISP=SHR
//DEMO DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.DMG.SA.V1,DISP=SHR
//CON  DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TKTCON,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TKTBS DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKETBS,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TKTFL02 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET02,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TKTFL03 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET03,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TKTFL04 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET04,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TKTFL05 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET05,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TKTFL06 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET06,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TKTFL07 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET07,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
```

## Appendix A.86

JCL/SAS Code: FINTKT

```
//      UNIT=TSILO
//TKTFL08 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET08,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL09 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET09,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET10,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL11 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET11,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL12 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET12,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL13 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET13,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL14 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET14,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL15 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET15,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL16 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET16,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL17 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET17,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL18 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET18,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL19 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET19,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL20 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET20,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//TKTFL21 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET21,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//*10% DATA MARTS
//TKTBSM DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10KTBS,
```

## Appendix A.86

JCL/SAS Code: FINTKT

```
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM02 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT02,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM03 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT03,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM04 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT04,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM05 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT05,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM06 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT06,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM07 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT07,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM08 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT08,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM09 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT09,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM10 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT10,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM11 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT11,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM12 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT12,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM13 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT13,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM14 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT14,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM15 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT15,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM16 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT16,
```

## Appendix A.86

JCL/SAS Code: FINTKT

```
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM17 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT17,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM18 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT18,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM19 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT19,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM20 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT20,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TKTDM21 DD  DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT21,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//*COMMON VARIABLES
//COMV DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX
        DKROCOND=WARN;
```

```
TITLE1 "Ticket to Work";
TITLE2 "DAF TKTSPLIT";
TITLE3;
```

```
/* change endyr and endmmn for each run */
%LET BEGYR=2002;
%LET BEGY=%SUBSTR(&BEGYR.,3); *FOR TKTY COMMON VARS;
%LET ENDYR=2021; /* change as needed */
%LET ENDY=%SUBSTR(&ENDYR.,3); *FOR TKTY COMMON VARS;
%PUT &BEGYR -> &BEGY., &ENDYR -> &ENDY;
```

```
%LET DROP_TITLE=22; /*SET TO YEAR AFTER CURRENT DAF YY*/
```

```
/* step to assign macro variables to handle time series data */
%MACRO SPLITTKT;
```

```
DATA TKTBS.TICKETBS(DROP=ACTIVE_EV: ACTIVE_MNTH: DEACTIVE_EN_MNTH:
        DEACTIVE_MNTH: DEACTIVE_SVRA_MNTH: ELIGIBLE_EV:
```

## Appendix A.86

JCL/SAS Code: FINTKT

```

        ELIGIBLE_MNTH: EN_EV: EN_MO_EV: EN_MO_MNTH:
        EN_OO_EV: EN_OO_MNTH: FIRST_ASGN_EN_MNTH:
        FIRST_ASGN_MNTH: FIRST_ASGN_SVRA_MNTH: MAIL_MNTH:
        PMTTYE_MNTH: PROVTYPE_MNTH: REASSIGN_EN_MNTH:
        REASSIGN_MNTH: REASSIGN_VR_MNTH: TITLE_MNTH:
SVRA_EV: SVRA_MO_EV: SVRA_MO_MNTH: SVRA_OO_EV: SVRA_OO_MNTH:
        SVRA_TR_EV: SVRA_TR_MNTH:
        INUSE_MNTH:)
%DO YYYY=&BEGYR. %TO &ENDYR.;
  %LET YY=%SUBSTR(&YYYY,3,2);
TKTFL&YY..TICKET&YY.(KEEP=SSN
  ACTIVE_EV&YY: ACTIVE_MNTH&YY: DEACTIVE_EN_MNTH&YY:
  DEACTIVE_MNTH&YY: DEACTIVE_SVRA_MNTH&YY: ELIGIBLE_EV&YY:
  ELIGIBLE_MNTH&YY: EN_EV&YY: EN_MO_EV&YY:
  EN_MO_MNTH&YY: EN_OO_EV&YY: EN_OO_MNTH&YY:
  FIRST_ASGN_MNTH&YY:
  MAIL_MNTH&YY: PMTTYE_MNTH&YY: PROVTYPE_MNTH&YY:
  TITLE_MNTH&YY: SVRA_EV&YY: SVRA_MO_EV&YY:
  SVRA_MO_MNTH&YY: SVRA_OO_EV&YY: SVRA_OO_MNTH&YY:
  SVRA_TR_EV&YY: SVRA_TR_MNTH&YY: INUSE_MNTH&YY:
  TKT&BEGYY.-TKT&ENDYY.
  DOBBEST DODBEST DOEI SEX
  SSDI_ONSET
  SSDI94-SSDI99 SSDI00-SSDI&ENDYY
  SSI94- SSI99 SSI00- SSI&ENDYY
  DXPRIBEST FRADATE
  SSIELIG_FIRST)
%END;
COMV.COMVARS (KEEP=SSN DOBBEST DODBEST FRADATE SSIELIG_FIRST DOEI
  SSDI94-SSDI99 SSDI00-SSDI&ENDYY SSI94-SSI99 SEX
  SSI00-SSI&ENDYY TKT02-TKT&ENDYY DXPRIBEST SSDI_ONSET)
;
MERGE IN1.CRTKTFLG(IN=CRTKTFLG
  RENAME=(COSSN=SSN
  EVER_VR =EVER_SVRA
  EVER_VR_MO =EVER_SVRA_MO
  EVER_VR_OO =EVER_SVRA_OO
  EVER_VR_TR =EVER_SVRA_TR
  NOE = TKT_NOE
  %DO YYYY=&BEGYR. %TO &ENDYR.;
    %LET YR=%SUBSTR(&YYYY,3,2);
    %DO P=1 %TO 12;
    %LET MN = %SYSFUNC(PUTN(&P.,Z2.));
    DEACTIVE_VR_MO&YR.&MN.=DEACTIVE_SVRA_MNTH&YR.&MN.

```

Appendix A.86

JCL/SAS Code: FINTKT

```

DEACTIVE_EN_MO&YR.&MN.=DEACTIVE_EN_MNTH&YR.&MN.
DEACTIVE_MO&YR.&MN.=DEACTIVE_MNTH&YR.&MN.
ACTIVE_MO&YR.&MN.=ACTIVE_MNTH&YR.&MN.
ELIGIBLE_MO&YR.&MN.=ELIGIBLE_MNTH&YR.&MN.

```

```

VR_EV&YR.&MN.= SVRA_EV&YR.&MN.
VR_MO_EV&YR.&MN.= SVRA_MO_EV&YR.&MN.
VR_MO_MO&YR.&MN.= SVRA_MO_MNTH&YR.&MN.
VR_OO_EV&YR.&MN.= SVRA_OO_EV&YR.&MN.
VR_OO_MO&YR.&MN.= SVRA_OO_MNTH&YR.&MN.
VR_TR_EV&YR.&MN.= SVRA_TR_EV&YR.&MN.
VR_TR_MO&YR.&MN.= SVRA_TR_MNTH&YR.&MN.

```

```

FIRST_ASGN_VR_MO&YR.&MN.= FIRST_ASGN_SVRA_MNTH&YR.&MN.
FIRST_ASGN_EN_MO&YR.&MN.= FIRST_ASGN_EN_MNTH&YR.&MN.
FIRST_ASGN_MO&YR.&MN.= FIRST_ASGN_MNTH&YR.&MN.

```

```

EN_MO_MO&YR.&MN.=EN_MO_MNTH&YR.&MN.
EN_OO_MO&YR.&MN.=EN_OO_MNTH&YR.&MN.
MAIL_MO&YR.&MN.=MAIL_MNTH&YR.&MN.
PMTTYPE_MO&YR.&MN.=PMTTYPE_MNTH&YR.&MN.
PROVTYPE_MO&YR.&MN.=PROVTYPE_MNTH&YR.&MN.

```

```

REASSIGN_EN_MO&YR.&MN.=REASSIGN_EN_MNTH&YR.&MN.
REASSIGN_VR_MO&YR.&MN.=REASSIGN_VR_MNTH&YR.&MN.
REASSIGN_MO&YR.&MN.=REASSIGN_MNTH&YR.&MN.

```

```

    %END;
%END;
))

```

```

IN2.TKTTITLE(RENAME=(COSSN=SSN

```

```

    %DO YYYY=&BEGYR. %TO &ENDYR.;
    %LET YR=%SUBSTR(&YYYY,3,2);
    %DO P=1 %TO 12;
    %LET MN = %SYSFUNC(PUTN(&P.,Z2.));
    INUSE_MO&YR.&MN.=INUSE_MNTH&YR.&MN.
    TITLE_MO&YR.&MN.=TITLE_MNTH&YR.&MN.
    %end;
    %end;
))

```

```

DEMO.DMG (KEEP=SSN DOBBEST DODBEST DOEI SEX FRADATE
SSDI_ONSET

```

## Appendix A.86

JCL/SAS Code: FINTKT

```
        SSDI94-SSDI99 SSDI00-SSDI&ENDYY
        SSI94- SSI99 SSI00- SSI&ENDYY
        DXPRIBEST SSIELIG_FIRST
        WHERE=(YEAR(DODBEST)>=1994 OR YEAR(DODBEST)=.)
        IN=INDEMO);
```

```
BY SSN;
```

```
DROP _LABEL_
    _NAME_
    AGE
    BEGDATE:
    DOBRETIRE
    ELIGTKTEVT
    ENDDATE:
    FIRSTASGNDT
    FIRSTASGNMO
    INCLNT
    MO
    NUMBER:
    /*DROPS TITLE DATA BEFORE 2002, AFTER DAFYY, AND SOME SLOP*/
    TITLE_MO01:
    TITLE_MO&DROP_TITLE.:
    INUSE_MO01:
    INUSE_MO&DROP_TITLE.:
    YR
    YY
    I;
```

```
    %INCLUDE 'OPDR.TG.PRD.ETTW.#6502.DAF21.TKT.PRDLIB(LABTKT)';
%DO YYYY=&BEGYR. %TO &ENDYR.;
    %LET YR=%SUBSTR(&YYYY,3,2);
    %INCLUDE 'OPDR.TG.PRD.ETTW.#6502.DAF21.TKT.PRDLIB(LABELSAN)';
%END;
```

```
/*OUTPUT*/
IF CRTKTF LG THEN OUTPUT TKTBS.TICKETBS
    %DO YYYY=&BEGYR. %TO &ENDYR.;
    %LET YY=%SUBSTR(&YYYY,3,2);
    TKTFL&YY..TICKET&YY.
    %END;
;
if INDEMO THEN OUTPUT COMV.COMVARS;
RUN;
```

Appendix A.86

JCL/SAS Code: FINTKT

```

DATA TKTBSM.P10TKTB;
SET TKTBS.TICKETBS;
IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9'))
AND (SUBSTR(SSN,9,1) IN ('0', '5'));
RUN;

%DO YYYY=&BEGYR. %TO &ENDYR.;
%LET YY=%SUBSTR(&YYYY,3,2);
DATA TKTDM&YY..P10TKT&YY.;
SET TKTFL&YY..TICKET&YY.;
IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9'))
AND (SUBSTR(SSN,9,1) IN ('0', '5'));
RUN;
%END;

PROC CONTENTS DATA=TKTBS.TICKETBS OUT=CON.TICKETBS; RUN;
PROC CONTENTS DATA=TKTBSM.P10TKTB OUT=CON.P10TKTB NOPRINT; RUN;

%DO YYYY=&BEGYR %TO &ENDYR;
%LET YY=%SUBSTR(&YYYY.,3,2);
PROC CONTENTS DATA=TKTFL&YY..TICKET&YY. OUT=CON.TICKET&YY;
RUN;
PROC CONTENTS DATA=TKTDM&YY..P10TKT&YY. OUT=CON.P10TKT&YY. NOPRINT;
RUN;
%END;

proc contents data=COMV.COMVARS; RUN;

%MEND SPLITTKT;

%SPLITTKT;

ENDSAS;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST

```



Appendix A.86

JCL/SAS Code: FINTKT

```
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: ABRYCE@MATHEMATICA-MPR.COM
TO: ABRYCE@MATHEMATICA-MPR.COM
SUBJECT: FINTKT
```

FINTKT HAS FINISHED RUNNING

Appendix A.87

JCL/SAS Code: LABTKT

LABEL

%MACRO LBL;

%LET MAXTKTNUM=31;

%DO I = 1 %TO &MAXTKTNUM;

TKTINTMLDT&I.="TICKET INITIAL MAIL DATE &I."

TKTADHMLDT&I.="TICKET AD-HOC RE-MAIL DATE &I."

TKT1YRMLDT&I.="TICKET ONE YEAR ANNIVERSARY NOTICE DATE &I."

TKT3YRMLDT&I.="TICKET THREE YEARS ANNIVERSARY NOTICE DATE &I."

TKTCUPMLDT&I.="TICKET CATCH-UP NOTICE DATE &I."

DUNS&I. ="EN DUNS &I."

PMTTYPE&I.="TICKET PAYMENT TYPE &I."

PROVTYPE&I.="TICKET PROVIDER TYPE &I."

TKTASGNDT&I.="TICKET ASSIGNMENT DATE &I."

TKTMAILDDT&I.="TICKET MAIL DATE &I."

TKTSELDT&I.="TICKET SELECTION DATE &I."

TKTTERMDT&I.="TICKET TERMINATION DATE &I."

TKTUNASGDT&I.="TICKET UNASSIGNMENT DATE &I."

%END;

%MEND;

%LBL

DOBTKT ="DATE OF BIRTH ON TKT FILE"

DODTKT ="DATE OF DEATH ON TKT FILE"

EVER\_ACTIVE ="EVER ASSIGNED TICKET AS OF DECEMBER OF DAF YEAR"

EVER\_ELIG ="EVER ELIGIBLE AS OF DECEMBER OF DAF YEAR"

EVER\_EN ="EVER ASSIGNED TICKET TO EN AS OF DECEMBER OF DAF YEAR"

EVER\_EN\_MO ="EVER ASSIGNED TICKET TO EN: MO AS OF DECEMBER OF DAF YEAR"

EVER\_EN\_OO ="EVER ASSIGNED TICKET TO EN: OO AS OF DECEMBER OF DAF YEAR"

EVER\_SVRA ="EVER ASSIGNED TICKET TO SVR AGENCY AS OF DECEMBER OF DAF YEAR"

EVER\_SVRA\_MO ="EVER ASSIGNED TICKET TO SVR AGENCY: MO AS OF DECEMBER OF DAF YEAR"

EVER\_SVRA\_OO ="EVER ASSIGNED TICKET TO SVR AGENCY: OO AS OF DECEMBER OF DAF YEAR"

EVER\_SVRA\_TR ="EVER ASSIGNED TICKET TO SVR AGENCY: TR AS OF DECEMBER OF DAF YEAR"

MINMAIL ="EARLIEST TICKET MAIL DATE"

TKT\_NOE ="TICKET NUMBER OF ENTRIES"

SSN ="SOCIAL SECURITY NUMBER"

DOBBEST ="BEST DATE OF BIRTH"

DODBEST ="BEST DATE OF DEATH"

DOEI ="SSDI INITIAL ENTITLEMENT DATE"

SEX ="SEX"

SSDI\_ONSET ="SSDI DISABILITY ONSET DATE EARLIEST PERIOD OF ELIGIBILITY"

Appendix A.87

JCL/SAS Code: LABTKT

SSDI00="SSDI CURRENT PAY STATUS IN 2000"  
SSDI01="SSDI CURRENT PAY STATUS IN 2001"  
SSDI02="SSDI CURRENT PAY STATUS IN 2002"  
SSDI03="SSDI CURRENT PAY STATUS IN 2003"  
SSDI04="SSDI CURRENT PAY STATUS IN 2004"  
SSDI05="SSDI CURRENT PAY STATUS IN 2005"  
SSDI06="SSDI CURRENT PAY STATUS IN 2006"  
SSDI07="SSDI CURRENT PAY STATUS IN 2007"  
SSDI08="SSDI CURRENT PAY STATUS IN 2008"  
SSDI09="SSDI CURRENT PAY STATUS IN 2009"  
SSDI10="SSDI CURRENT PAY STATUS IN 2010"  
SSDI11="SSDI CURRENT PAY STATUS IN 2011"  
SSDI12="SSDI CURRENT PAY STATUS IN 2012"  
SSDI13="SSDI CURRENT PAY STATUS IN 2013"  
SSDI14="SSDI CURRENT PAY STATUS IN 2014"  
SSDI15="SSDI CURRENT PAY STATUS IN 2015"  
SSDI16="SSDI CURRENT PAY STATUS IN 2016"  
SSDI17="SSDI CURRENT PAY STATUS IN 2017"  
SSDI18="SSDI CURRENT PAY STATUS IN 2018"  
SSDI19="SSDI CURRENT PAY STATUS IN 2019"  
SSDI20="SSDI CURRENT PAY STATUS IN 2020"  
SSDI21="SSDI CURRENT PAY STATUS IN 2021"  
SSDI94="SSDI CURRENT PAY STATUS IN 1994"  
SSDI95="SSDI CURRENT PAY STATUS IN 1995"  
SSDI96="SSDI CURRENT PAY STATUS IN 1996"  
SSDI97="SSDI CURRENT PAY STATUS IN 1997"  
SSDI98="SSDI CURRENT PAY STATUS IN 1998"  
SSDI99="SSDI CURRENT PAY STATUS IN 1999"  
SSI00="SSI CURRENT PAY STATUS IN 2000"  
SSI01="SSI CURRENT PAY STATUS IN 2001"  
SSI02="SSI CURRENT PAY STATUS IN 2002"  
SSI03="SSI CURRENT PAY STATUS IN 2003"  
SSI04="SSI CURRENT PAY STATUS IN 2004"  
SSI05="SSI CURRENT PAY STATUS IN 2005"  
SSI06="SSI CURRENT PAY STATUS IN 2006"  
SSI07="SSI CURRENT PAY STATUS IN 2007"  
SSI08="SSI CURRENT PAY STATUS IN 2008"  
SSI09="SSI CURRENT PAY STATUS IN 2009"  
SSI10="SSI CURRENT PAY STATUS IN 2010"  
SSI11="SSI CURRENT PAY STATUS IN 2011"  
SSI12="SSI CURRENT PAY STATUS IN 2012"  
SSI13="SSI CURRENT PAY STATUS IN 2013"  
SSI14="SSI CURRENT PAY STATUS IN 2014"  
SSI15="SSI CURRENT PAY STATUS IN 2015"  
SSI16="SSI CURRENT PAY STATUS IN 2016"

Appendix A.87

JCL/SAS Code: LABTKT

```
SSI17="SSI CURRENT PAY STATUS IN 2017"  
SSI18="SSI CURRENT PAY STATUS IN 2018"  
SSI19="SSI CURRENT PAY STATUS IN 2019"  
SSI20="SSI CURRENT PAY STATUS IN 2020"  
SSI21="SSI CURRENT PAY STATUS IN 2021"  
SSI94="SSI CURRENT PAY STATUS IN 1994"  
SSI95="SSI CURRENT PAY STATUS IN 1995"  
SSI96="SSI CURRENT PAY STATUS IN 1996"  
SSI97="SSI CURRENT PAY STATUS IN 1997"  
SSI98="SSI CURRENT PAY STATUS IN 1998"  
SSI99="SSI CURRENT PAY STATUS IN 1999"  
TKT02="TICKET ACTIVE IN 2002"  
TKT03="TICKET ACTIVE IN 2003"  
TKT04="TICKET ACTIVE IN 2004"  
TKT05="TICKET ACTIVE IN 2005"  
TKT06="TICKET ACTIVE IN 2006"  
TKT07="TICKET ACTIVE IN 2007"  
TKT08="TICKET ACTIVE IN 2008"  
TKT09="TICKET ACTIVE IN 2009"  
TKT10="TICKET ACTIVE IN 2010"  
TKT11="TICKET ACTIVE IN 2011"  
TKT12="TICKET ACTIVE IN 2012"  
TKT13="TICKET ACTIVE IN 2013"  
TKT14="TICKET ACTIVE IN 2014"  
TKT15="TICKET ACTIVE IN 2015"  
TKT16="TICKET ACTIVE IN 2016"  
TKT17="TICKET ACTIVE IN 2017"  
TKT18="TICKET ACTIVE IN 2018"  
TKT19="TICKET ACTIVE IN 2019"  
TKT20="TICKET ACTIVE IN 2020"  
TKT21="TICKET ACTIVE IN 2021"  
DXPRIBEST="PRIMARY DIAGNOSIS CODE MOST RECENT, BEST FROM ALL SOURCES"  
SSIELIG_FIRST="SSI EARLIEST ELIGIBILITY DATE (CONSTRUCTED)"  
FRADATE = "FULL RETIREMENT DATE"  
;
```

Appendix A.88

JCL/SAS Code: LABELSAN

LABEL

ACTIVE\_EV&YR.01="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.01"  
ACTIVE\_EV&YR.02="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.02"  
ACTIVE\_EV&YR.03="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.03"  
ACTIVE\_EV&YR.04="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.04"  
ACTIVE\_EV&YR.05="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.05"  
ACTIVE\_EV&YR.06="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.06"  
ACTIVE\_EV&YR.07="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.07"  
ACTIVE\_EV&YR.08="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.08"  
ACTIVE\_EV&YR.09="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.09"  
ACTIVE\_EV&YR.10="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.10"  
ACTIVE\_EV&YR.11="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.11"  
ACTIVE\_EV&YR.12="EVER ASSIGNED OR IN-USE TICKET AS OF MONTH &YR.12"  
ACTIVE\_MNTH&YR.01="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.01"  
ACTIVE\_MNTH&YR.02="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.02"  
ACTIVE\_MNTH&YR.03="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.03"  
ACTIVE\_MNTH&YR.04="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.04"  
ACTIVE\_MNTH&YR.05="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.05"  
ACTIVE\_MNTH&YR.06="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.06"  
ACTIVE\_MNTH&YR.07="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.07"  
ACTIVE\_MNTH&YR.08="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.08"  
ACTIVE\_MNTH&YR.09="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.09"  
ACTIVE\_MNTH&YR.10="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.10"  
ACTIVE\_MNTH&YR.11="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.11"  
ACTIVE\_MNTH&YR.12="TICKET ACTIVE (ASSIGNED OR IN-USE) IN MONTH &YR.12"  
DEACTIVE\_EN\_MNTH&YR.01=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.01"  
DEACTIVE\_EN\_MNTH&YR.02=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.02"  
DEACTIVE\_EN\_MNTH&YR.03=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.03"  
DEACTIVE\_EN\_MNTH&YR.04=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.04"  
DEACTIVE\_EN\_MNTH&YR.05=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.05"  
DEACTIVE\_EN\_MNTH&YR.06=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.06"  
DEACTIVE\_EN\_MNTH&YR.07=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.07"  
DEACTIVE\_EN\_MNTH&YR.08=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.08"  
DEACTIVE\_EN\_MNTH&YR.09=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.09"  
DEACTIVE\_EN\_MNTH&YR.10=

Appendix A.88

JCL/SAS Code: LABELSAN

"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.10"  
DEACTIVE\_EN\_MNTH&YR.11=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.11"  
DEACTIVE\_EN\_MNTH&YR.12=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM EN IN MONTH &YR.12"  
DEACTIVE\_MNTH&YR.01=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.01"  
DEACTIVE\_MNTH&YR.02=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.02"  
DEACTIVE\_MNTH&YR.03=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.03"  
DEACTIVE\_MNTH&YR.04=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.04"  
DEACTIVE\_MNTH&YR.05=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.05"  
DEACTIVE\_MNTH&YR.06=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.06"  
DEACTIVE\_MNTH&YR.07=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.07"  
DEACTIVE\_MNTH&YR.08=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.08"  
DEACTIVE\_MNTH&YR.09=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.09"  
DEACTIVE\_MNTH&YR.10=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.10"  
DEACTIVE\_MNTH&YR.11=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.11"  
DEACTIVE\_MNTH&YR.12=  
"TICKET UNASSIGNED OR NO LONGER IN-USE IN MONTH &YR.12"  
DEACTIVE\_SVRA\_MNTH&YR.01=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.01"  
DEACTIVE\_SVRA\_MNTH&YR.02=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.02"  
DEACTIVE\_SVRA\_MNTH&YR.03=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.03"  
DEACTIVE\_SVRA\_MNTH&YR.04=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.04"  
DEACTIVE\_SVRA\_MNTH&YR.05=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.05"  
DEACTIVE\_SVRA\_MNTH&YR.06=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.06"  
DEACTIVE\_SVRA\_MNTH&YR.07=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.07"  
DEACTIVE\_SVRA\_MNTH&YR.08=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.08"

Appendix A.88

JCL/SAS Code: LABELSAN

DEACTIVE\_SVRA\_MNTH&YR.09=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.09"  
DEACTIVE\_SVRA\_MNTH&YR.10=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.10"  
DEACTIVE\_SVRA\_MNTH&YR.11=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.11"  
DEACTIVE\_SVRA\_MNTH&YR.12=  
"TICKET UNASSIGNED OR NO LONGER IN-USE FROM SVR AGENCY IN MONTH &YR.12"  
ELIGIBLE\_EV&YR.01="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.01"  
ELIGIBLE\_EV&YR.02="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.02"  
ELIGIBLE\_EV&YR.03="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.03"  
ELIGIBLE\_EV&YR.04="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.04"  
ELIGIBLE\_EV&YR.05="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.05"  
ELIGIBLE\_EV&YR.06="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.06"  
ELIGIBLE\_EV&YR.07="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.07"  
ELIGIBLE\_EV&YR.08="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.08"  
ELIGIBLE\_EV&YR.09="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.09"  
ELIGIBLE\_EV&YR.10="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.10"  
ELIGIBLE\_EV&YR.11="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.11"  
ELIGIBLE\_EV&YR.12="EVER ELIGIBLE FOR A TICKET AS OF MONTH &YR.12"  
ELIGIBLE\_MNTH&YR.01="ELIGIBLE FOR A TICKET IN MONTH &YR.01"  
ELIGIBLE\_MNTH&YR.02="ELIGIBLE FOR A TICKET IN MONTH &YR.02"  
ELIGIBLE\_MNTH&YR.03="ELIGIBLE FOR A TICKET IN MONTH &YR.03"  
ELIGIBLE\_MNTH&YR.04="ELIGIBLE FOR A TICKET IN MONTH &YR.04"  
ELIGIBLE\_MNTH&YR.05="ELIGIBLE FOR A TICKET IN MONTH &YR.05"  
ELIGIBLE\_MNTH&YR.06="ELIGIBLE FOR A TICKET IN MONTH &YR.06"  
ELIGIBLE\_MNTH&YR.07="ELIGIBLE FOR A TICKET IN MONTH &YR.07"  
ELIGIBLE\_MNTH&YR.08="ELIGIBLE FOR A TICKET IN MONTH &YR.08"  
ELIGIBLE\_MNTH&YR.09="ELIGIBLE FOR A TICKET IN MONTH &YR.09"  
ELIGIBLE\_MNTH&YR.10="ELIGIBLE FOR A TICKET IN MONTH &YR.10"  
ELIGIBLE\_MNTH&YR.11="ELIGIBLE FOR A TICKET IN MONTH &YR.11"  
ELIGIBLE\_MNTH&YR.12="ELIGIBLE FOR A TICKET IN MONTH &YR.12"  
EN\_EV&YR.01="EVER ASSIGNED TO EN AS OF MONTH &YR.01"  
EN\_EV&YR.02="EVER ASSIGNED TO EN AS OF MONTH &YR.02"  
EN\_EV&YR.03="EVER ASSIGNED TICKET TO EN AS OF MONTH &YR.03"  
EN\_EV&YR.04="EVER ASSIGNED TICKET TO EN AS OF MONTH &YR.04"  
EN\_EV&YR.05="EVER ASSIGNED TICKET TO EN AS OF MONTH &YR.05"  
EN\_EV&YR.06="EVER ASSIGNED TICKET TO EN AS OF MONTH &YR.06"  
EN\_EV&YR.07="EVER ASSIGNED TICKET TO EN AS OF MONTH &YR.07"  
EN\_EV&YR.08="EVER ASSIGNED TICKET TO EN AS OF MONTH &YR.08"  
EN\_EV&YR.09="EVER ASSIGNED TICKET TO EN AS OF MONTH &YR.09"  
EN\_EV&YR.10="EVER ASSIGNED TICKET TO EN AS OF MONTH &YR.10"  
EN\_EV&YR.11="EVER ASSIGNED TICKET TO EN AS OF MONTH &YR.11"  
EN\_EV&YR.12="EVER ASSIGNED TICKET TO EN AS OF MONTH &YR.12"  
EN\_MO\_EV&YR.01="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.01"

Appendix A.88

JCL/SAS Code: LABELSAN

EN\_MO\_EV&YR.02="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.02"  
EN\_MO\_EV&YR.03="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.03"  
EN\_MO\_EV&YR.04="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.04"  
EN\_MO\_EV&YR.05="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.05"  
EN\_MO\_EV&YR.06="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.06"  
EN\_MO\_EV&YR.07="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.07"  
EN\_MO\_EV&YR.08="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.08"  
EN\_MO\_EV&YR.09="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.09"  
EN\_MO\_EV&YR.10="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.10"  
EN\_MO\_EV&YR.11="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.11"  
EN\_MO\_EV&YR.12="EVER ASSIGNED TICKET TO EN:MO AS OF MONTH &YR.12"  
EN\_MO\_MNTH&YR.01="TICKET ASSIGNED TO EN:MO IN MONTH &YR.01"  
EN\_MO\_MNTH&YR.02="TICKET ASSIGNED TO EN:MO IN MONTH &YR.02"  
EN\_MO\_MNTH&YR.03="TICKET ASSIGNED TO EN:MO IN MONTH &YR.03"  
EN\_MO\_MNTH&YR.04="TICKET ASSIGNED TO EN:MO IN MONTH &YR.04"  
EN\_MO\_MNTH&YR.05="TICKET ASSIGNED TO EN:MO IN MONTH &YR.05"  
EN\_MO\_MNTH&YR.06="TICKET ASSIGNED TO EN:MO IN MONTH &YR.06"  
EN\_MO\_MNTH&YR.07="TICKET ASSIGNED TO EN:MO IN MONTH &YR.07"  
EN\_MO\_MNTH&YR.08="TICKET ASSIGNED TO EN:MO IN MONTH &YR.08"  
EN\_MO\_MNTH&YR.09="TICKET ASSIGNED TO EN:MO IN MONTH &YR.09"  
EN\_MO\_MNTH&YR.10="TICKET ASSIGNED TO EN:MO IN MONTH &YR.10"  
EN\_MO\_MNTH&YR.11="TICKET ASSIGNED TO EN:MO IN MONTH &YR.11"  
EN\_MO\_MNTH&YR.12="TICKET ASSIGNED TO EN:MO IN MONTH &YR.12"  
EN\_OO\_EV&YR.01="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.01"  
EN\_OO\_EV&YR.02="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.02"  
EN\_OO\_EV&YR.03="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.03"  
EN\_OO\_EV&YR.04="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.04"  
EN\_OO\_EV&YR.05="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.05"  
EN\_OO\_EV&YR.06="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.06"  
EN\_OO\_EV&YR.07="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.07"  
EN\_OO\_EV&YR.08="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.08"  
EN\_OO\_EV&YR.09="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.09"  
EN\_OO\_EV&YR.10="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.10"  
EN\_OO\_EV&YR.11="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.11"  
EN\_OO\_EV&YR.12="EVER ASSIGNED TICKET TO EN:OO AS OF MONTH &YR.12"  
EN\_OO\_MNTH&YR.01="TICKET ASSIGNED TO EN:OO IN MONTH &YR.01"  
EN\_OO\_MNTH&YR.02="TICKET ASSIGNED TO EN:OO IN MONTH &YR.02"  
EN\_OO\_MNTH&YR.03="TICKET ASSIGNED TO EN:OO IN MONTH &YR.03"  
EN\_OO\_MNTH&YR.04="TICKET ASSIGNED TO EN:OO IN MONTH &YR.04"  
EN\_OO\_MNTH&YR.05="TICKET ASSIGNED TO EN:OO IN MONTH &YR.05"  
EN\_OO\_MNTH&YR.06="TICKET ASSIGNED TO EN:OO IN MONTH &YR.06"  
EN\_OO\_MNTH&YR.07="TICKET ASSIGNED TO EN:OO IN MONTH &YR.07"  
EN\_OO\_MNTH&YR.08="TICKET ASSIGNED TO EN:OO IN MONTH &YR.08"  
EN\_OO\_MNTH&YR.09="TICKET ASSIGNED TO EN:OO IN MONTH &YR.09"  
EN\_OO\_MNTH&YR.10="TICKET ASSIGNED TO EN:OO IN MONTH &YR.10"



Appendix A.88

JCL/SAS Code: LABELSAN

EN\_OO\_MNTH&YR.11="TICKET ASSIGNED TO EN:OO IN MONTH &YR.11"  
EN\_OO\_MNTH&YR.12="TICKET ASSIGNED TO EN:OO IN MONTH &YR.12"  
FIRST\_ASGN\_MNTH&YR.01="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.01"  
FIRST\_ASGN\_MNTH&YR.02="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.02"  
FIRST\_ASGN\_MNTH&YR.03="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.03"  
FIRST\_ASGN\_MNTH&YR.04="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.04"  
FIRST\_ASGN\_MNTH&YR.05="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.05"  
FIRST\_ASGN\_MNTH&YR.06="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.06"  
FIRST\_ASGN\_MNTH&YR.07="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.07"  
FIRST\_ASGN\_MNTH&YR.08="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.08"  
FIRST\_ASGN\_MNTH&YR.09="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.09"  
FIRST\_ASGN\_MNTH&YR.10="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.10"  
FIRST\_ASGN\_MNTH&YR.11="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.11"  
FIRST\_ASGN\_MNTH&YR.12="TICKET FIRST ASSIGNED OR IN-USE IN MONTH &YR.12"  
INUSE\_MNTH&YR.01="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.01"  
INUSE\_MNTH&YR.02="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.02"  
INUSE\_MNTH&YR.03="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.03"  
INUSE\_MNTH&YR.04="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.04"  
INUSE\_MNTH&YR.05="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.05"  
INUSE\_MNTH&YR.06="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.06"  
INUSE\_MNTH&YR.07="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.07"  
INUSE\_MNTH&YR.08="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.08"  
INUSE\_MNTH&YR.09="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.09"  
INUSE\_MNTH&YR.10="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.10"  
INUSE\_MNTH&YR.11="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.11"  
INUSE\_MNTH&YR.12="TICKET IN-USE (YYMM = YEAR/MONTH) &YR.12"  
MAIL\_MNTH&YR.01="TICKET MAILED IN CALENDAR MONTH &YR.01"  
MAIL\_MNTH&YR.02="TICKET MAILED IN CALENDAR MONTH &YR.02"  
MAIL\_MNTH&YR.03="TICKET MAILED IN CALENDAR MONTH &YR.03"  
MAIL\_MNTH&YR.04="TICKET MAILED IN CALENDAR MONTH &YR.04"  
MAIL\_MNTH&YR.05="TICKET MAILED IN CALENDAR MONTH &YR.05"  
MAIL\_MNTH&YR.06="TICKET MAILED IN CALENDAR MONTH &YR.06"  
MAIL\_MNTH&YR.07="TICKET MAILED IN CALENDAR MONTH &YR.07"  
MAIL\_MNTH&YR.08="TICKET MAILED IN CALENDAR MONTH &YR.08"  
MAIL\_MNTH&YR.09="TICKET MAILED IN CALENDAR MONTH &YR.09"  
MAIL\_MNTH&YR.10="TICKET MAILED IN CALENDAR MONTH &YR.10"  
MAIL\_MNTH&YR.11="TICKET MAILED IN CALENDAR MONTH &YR.11"  
MAIL\_MNTH&YR.12="TICKET MAILED IN CALENDAR MONTH &YR.12"  
PMTTYPE\_MNTH&YR.01="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.01"  
PMTTYPE\_MNTH&YR.02="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.02"  
PMTTYPE\_MNTH&YR.03="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.03"  
PMTTYPE\_MNTH&YR.04="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.04"  
PMTTYPE\_MNTH&YR.05="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.05"  
PMTTYPE\_MNTH&YR.06="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.06"  
PMTTYPE\_MNTH&YR.07="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.07"

Appendix A.88

JCL/SAS Code: LABELSAN

PMTTYPE\_MNTH&YR.08="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.08"  
PMTTYPE\_MNTH&YR.09="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.09"  
PMTTYPE\_MNTH&YR.10="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.10"  
PMTTYPE\_MNTH&YR.11="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.11"  
PMTTYPE\_MNTH&YR.12="PAYMENT TYPE FOR ACTIVE TICKET IN MONTH &YR.12"  
PROVTYPE\_MNTH&YR.01="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.01"  
PROVTYPE\_MNTH&YR.02="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.02"  
PROVTYPE\_MNTH&YR.03="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.03"  
PROVTYPE\_MNTH&YR.04="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.04"  
PROVTYPE\_MNTH&YR.05="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.05"  
PROVTYPE\_MNTH&YR.06="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.06"  
PROVTYPE\_MNTH&YR.07="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.07"  
PROVTYPE\_MNTH&YR.08="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.08"  
PROVTYPE\_MNTH&YR.09="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.09"  
PROVTYPE\_MNTH&YR.10="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.10"  
PROVTYPE\_MNTH&YR.11="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.11"  
PROVTYPE\_MNTH&YR.12="PROVIDER TYPE FOR ACTIVE TICKET IN MONTH &YR.12"  
TITLE\_MNTH&YR.01="TICKET TITLE (PROGRAM) IN MONTH &YR.01"  
TITLE\_MNTH&YR.02="TICKET TITLE (PROGRAM) IN MONTH &YR.02"  
TITLE\_MNTH&YR.03="TICKET TITLE (PROGRAM) IN MONTH &YR.03"  
TITLE\_MNTH&YR.04="TICKET TITLE (PROGRAM) IN MONTH &YR.04"  
TITLE\_MNTH&YR.05="TICKET TITLE (PROGRAM) IN MONTH &YR.05"  
TITLE\_MNTH&YR.06="TICKET TITLE (PROGRAM) IN MONTH &YR.06"  
TITLE\_MNTH&YR.07="TICKET TITLE (PROGRAM) IN MONTH &YR.07"  
TITLE\_MNTH&YR.08="TICKET TITLE (PROGRAM) IN MONTH &YR.08"  
TITLE\_MNTH&YR.09="TICKET TITLE (PROGRAM) IN MONTH &YR.09"  
TITLE\_MNTH&YR.10="TICKET TITLE (PROGRAM) IN MONTH &YR.10"  
TITLE\_MNTH&YR.11="TICKET TITLE (PROGRAM) IN MONTH &YR.11"  
TITLE\_MNTH&YR.12="TICKET TITLE (PROGRAM) IN MONTH &YR.12"  
SVRA\_EV&YR.01= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.01"  
SVRA\_EV&YR.02= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.02"  
SVRA\_EV&YR.03= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.03"  
SVRA\_EV&YR.04= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.04"  
SVRA\_EV&YR.05= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.05"  
SVRA\_EV&YR.06= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.06"  
SVRA\_EV&YR.07= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.07"  
SVRA\_EV&YR.08= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.08"  
SVRA\_EV&YR.09= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.09"  
SVRA\_EV&YR.10= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.10"  
SVRA\_EV&YR.11= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.11"  
SVRA\_EV&YR.12= "EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY AS OF MONTH&YR.12"  
SVRA\_MO\_EV&YR.01=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.01"  
SVRA\_MO\_EV&YR.02=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.02"

Appendix A.88

JCL/SAS Code: LABELSAN

SVRA\_MO\_EV&YR.03=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.03"  
SVRA\_MO\_EV&YR.04=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.04"  
SVRA\_MO\_EV&YR.05=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.05"  
SVRA\_MO\_EV&YR.06=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.06"  
SVRA\_MO\_EV&YR.07=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.07"  
SVRA\_MO\_EV&YR.08=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.08"  
SVRA\_MO\_EV&YR.09=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.09"  
SVRA\_MO\_EV&YR.10=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.10"  
SVRA\_MO\_EV&YR.11=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.11"  
SVRA\_MO\_EV&YR.12=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:MO AS OF MONTH &YR.12"  
SVRA\_MO\_MNTH&YR.01="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.01"  
SVRA\_MO\_MNTH&YR.02="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.02"  
SVRA\_MO\_MNTH&YR.03="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.03"  
SVRA\_MO\_MNTH&YR.04="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.04"  
SVRA\_MO\_MNTH&YR.05="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.05"  
SVRA\_MO\_MNTH&YR.06="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.06"  
SVRA\_MO\_MNTH&YR.07="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.07"  
SVRA\_MO\_MNTH&YR.08="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.08"  
SVRA\_MO\_MNTH&YR.09="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.09"  
SVRA\_MO\_MNTH&YR.10="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.10"  
SVRA\_MO\_MNTH&YR.11="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.11"  
SVRA\_MO\_MNTH&YR.12="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:MO IN MONTH &YR.12"  
SVRA\_OO\_EV&YR.01=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.01"  
SVRA\_OO\_EV&YR.02=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.02"  
SVRA\_OO\_EV&YR.03=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.03"  
SVRA\_OO\_EV&YR.04=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.04"  
SVRA\_OO\_EV&YR.05=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.05"  
SVRA\_OO\_EV&YR.06=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.06"  
SVRA\_OO\_EV&YR.07=

Appendix A.88

JCL/SAS Code: LABELSAN

"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.07"  
SVRA\_OO\_EV&YR.08=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.08"  
SVRA\_OO\_EV&YR.09=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.09"  
SVRA\_OO\_EV&YR.10=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.10"  
SVRA\_OO\_EV&YR.11=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.11"  
SVRA\_OO\_EV&YR.12=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:OO AS OF MONTH &YR.12"  
SVRA\_OO\_MNTH&YR.01="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.01"  
SVRA\_OO\_MNTH&YR.02="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.02"  
SVRA\_OO\_MNTH&YR.03="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.03"  
SVRA\_OO\_MNTH&YR.04="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.04"  
SVRA\_OO\_MNTH&YR.05="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.05"  
SVRA\_OO\_MNTH&YR.06="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.06"  
SVRA\_OO\_MNTH&YR.07="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.07"  
SVRA\_OO\_MNTH&YR.08="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.08"  
SVRA\_OO\_MNTH&YR.09="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.09"  
SVRA\_OO\_MNTH&YR.10="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.10"  
SVRA\_OO\_MNTH&YR.11="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.11"  
SVRA\_OO\_MNTH&YR.12="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:OO IN MONTH &YR.12"  
SVRA\_TR\_EV&YR.01=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.01"  
SVRA\_TR\_EV&YR.02=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.02"  
SVRA\_TR\_EV&YR.03=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.03"  
SVRA\_TR\_EV&YR.04=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.04"  
SVRA\_TR\_EV&YR.05=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.05"  
SVRA\_TR\_EV&YR.06=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.06"  
SVRA\_TR\_EV&YR.07=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.07"  
SVRA\_TR\_EV&YR.08=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.08"  
SVRA\_TR\_EV&YR.09=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.09"  
SVRA\_TR\_EV&YR.10=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.10"  
SVRA\_TR\_EV&YR.11=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.11"

Appendix A.88

JCL/SAS Code: LABELSAN

```
SVRA_TR_EV&YR.12=  
"EVER ASSIGNED OR IN-USE TICKET TO SVR AGENCY:TR AS OF MONTH &YR.12"  
SVRA_TR_MNTH&YR.01="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.01"  
SVRA_TR_MNTH&YR.02="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.02"  
SVRA_TR_MNTH&YR.03="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.03"  
SVRA_TR_MNTH&YR.04="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.04"  
SVRA_TR_MNTH&YR.05="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.05"  
SVRA_TR_MNTH&YR.06="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.06"  
SVRA_TR_MNTH&YR.07="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.07"  
SVRA_TR_MNTH&YR.08="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.08"  
SVRA_TR_MNTH&YR.09="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.09"  
SVRA_TR_MNTH&YR.10="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.10"  
SVRA_TR_MNTH&YR.11="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.11"  
SVRA_TR_MNTH&YR.12="TICKET ASSIGNED OR IN-USE TO SVR AGENCY:TR IN MONTH &YR.12"  
;
```

## Appendix A.89

JCL/SAS Code: DCFERNEX

```
//#6266DCF JOB (12510000,T715,,SAS,,ITC9TH),LUKASHANETS,CLASS=T,
// MSGCLASS=1,MSGLEVEL=(1,1),REGION=4096K,NOTIFY=#6266
//*****
//* PROJECT:   BUILD DAF21 (51390)
//* PROJECT
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF21.ANN.PRDLIB(DCFERNEX)
//* DESCRIPTION: CREATES EXTRACT OF DCF EARNINGS TABLES
//*           TO OBTAIN THE EARNINGS DATA FOR THE ANNUAL FILES
//* PROGRAMMER: SERGE LUKASHANETS 06/20/2022
//*****
//S1 EXEC SAS9,LOAD='DBP8.DB2.SDSNLOAD',SORT='15000',
//  WORK='15000,15000',COND=EVEN
//ALE DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DCFALERN.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=TSILO,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HALE DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DCFHALER.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=AFF=ALE,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//T16E DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DCFT16ER.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=AFF=ALE,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HT16 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DCFHT16E.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=AFF=ALE,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//T2ER DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DCFT2ER.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=AFF=ALE,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HT2E DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DCFHT2E.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=AFF=ALE,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//T2WK DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DCFT2WKD.SA.V1,
```

## Appendix A.89

JCL/SAS Code: DCFERNEX

```
// DISP=(,CATLG,DELETE),
// UNIT=AFF=ALE,
// VOL=(,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HT2W DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DCFHT2WK.SA.V1,
// DISP=(,CATLG,DELETE),
// UNIT=AFF=ALE,
// VOL=(,,255),
// DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//SYSIN DD *
```

```
OPTION NOCENTER;
```

```
LIBNAME ALE TAPE;
LIBNAME HALE TAPE;
LIBNAME T16E TAPE;
LIBNAME HT16 TAPE;
LIBNAME T2ER TAPE;
LIBNAME HT2E TAPE;
LIBNAME T2WK TAPE;
LIBNAME HT2W TAPE;
LIBNAME MDCF DB2 SSID=DBP8 SCHEMA=MDCF ;
```

```
* CREATE SAS EXTRACT OF ALLEGED EARNINGS GROUP DATA;
PROC SQL;
CREATE TABLE ALE.ALLGERNG AS
SELECT * FROM MDCF.ALLGERNG;
QUIT;
RUN;
```

```
* CREATE SAS EXTRACT OF HISTORY ALLEGED EARNINGS GROUP DATA;
PROC SQL;
CREATE TABLE HALE.HALLGERNG AS
SELECT * FROM MDCF.HALLGERNG;
QUIT;
RUN;
```

```
* CREATE SAS EXTRACT OF T16 EARNINGS GROUP DATA;
PROC SQL;
CREATE TABLE T16E.T16ERNNGS AS
SELECT * FROM MDCF.T16ERNNGS;
QUIT;
RUN;
```

```
* CREATE SAS EXTRACT OF HISTORY T16 EARNINGS GROUP DATA;
```

Appendix A.89

JCL/SAS Code: DCFERNEX

```
PROC SQL;  
CREATE TABLE HT16.HT16ERNG AS  
SELECT * FROM MDCF.HT16ERNG;  
QUIT;
```

```
* CREATE SAS EXTRACT OF T2 EARNINGS GROUP DATA;  
PROC SQL;  
CREATE TABLE T2ER.T2ERNGS AS  
SELECT * FROM MDCF.T2ERNGS;  
QUIT;  
RUN;
```

```
* CREATE SAS EXTRACT OF HISTORY T2 EARNINGS GROUP DATA;  
PROC SQL;  
CREATE TABLE HT2E.HT2ERNGS AS  
SELECT * FROM MDCF.HT2ERNGS;  
QUIT;  
RUN;
```

```
* CREATE SAS EXTRACT OF T2 WORK GROUP DATA;  
PROC SQL;  
CREATE TABLE T2WK.T2WKDET AS  
SELECT * FROM MDCF.T2WKDET  
QUIT;  
RUN;
```

```
* CREATE SAS EXTRACT OF HISTORY T2 WORK GROUP DATA;  
PROC SQL;  
CREATE TABLE HT2W.HT2WKDET AS  
SELECT * FROM MDCF.HT2WKDET;  
QUIT;  
RUN;
```

```
PROC CONTENTS DATA=ALE.ALLGERNG;  
RUN;  
PROC CONTENTS DATA=HALE.HALLGERN;  
RUN;  
PROC CONTENTS DATA=T16E.T16ERNGS;  
RUN;  
PROC CONTENTS DATA=HT16.HT16ERNG;  
RUN;  
PROC CONTENTS DATA=T2ER.T2ERNGS;  
RUN;  
PROC CONTENTS DATA=HT2E.HT2ERNGS;  
RUN;
```



Appendix A.89

JCL/SAS Code: DCFERNEX

```
PROC CONTENTS DATA=T2WK.T2WKDETN;
RUN;
PROC CONTENTS DATA=HT2W.HT2WKDET;
RUN;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 DCFERNEX IS COMPLETE
```

## Appendix A.90

JCL/SAS Code: ALGEARN

```

//#6266ALG JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   BUILD DAF21 51390
//* PROJECT
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF21.ANN.PRDLIB(ALGEARN)
//* DESCRIPTION: BUILDS THE YYMM EARNINGS VARIABLES FROM THE DCF
//*             ALLEGED EARNINGS TABLE
//* PROGRAMMER: SERGE LUKASHANETS 06/23/2022
//*****
//*
//SAS   EXEC SAS9,
//     WORK='120000,60000'
//*
//*
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DCFALERN.SA.V1,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ALGEARN.SA.V1,
//     DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//     SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//     SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//     SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN OBS=MAX;

/* CHANGE ENDYR AND ENDMMN FOR EACH RUN */
%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

/* STEP TO ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA */
%LET NUMYRS=%EVAL(&ENDYR.-&BEGYR.+1);

%MACRO START;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

***** ALLEGED EARNINGS PROCESSING *****,

DATA ALLERN;
  SET IN1.ALLGERNG;

```

## Appendix A.90

JCL/SAS Code: ALGEARN

```
* IF THE EARNINGS DATE IS IN A REPORTING YEAR THEN KEEP;
WHERE
  (ERNGS_DT>="01JAN&BEGYR."D) AND
  (ERNGS_DT<="31DEC&ENDYR."D) AND
  (CID='00');

* ATTACH FORMAT TO THE ERNGS_DT FIELD SO IT WILL APPEAR AS YYYYMM;
ATTRIB ERNGS_DT FORMAT=YYYYMMN4.;

RUN;

PROC SORT DATA=ALLERN;
  BY COSSN;
RUN;

* CREATE ONE OBSERVATION PER SSN WITH YYYYMM VARIABLES CONTAINING THE
  ALLEGED EARNINGS AMOUNT;
PROC TRANSPOSE DATA=ALLERN OUT=TEMP1.ALLERN(DROP=_NAME_ _LABEL_)
  PREFIX=ALLGAMT;
  BY COSSN;
  ID ERNGS_DT;
  VAR ALLGD_AMT;
RUN;

DATA TEMP2.ALGEARN;
  SET TEMP1.ALLERN;
  * THIS ARRAY WILL ENSURE THAT THERE IS A VARIABLE FOR EVERY
    EARNINGS TYPE FOR EVERY YYYYMM COMBINATION.  ALTHOUGH UNLIKELY
    IT IS POSSIBLE THAT THERE WILL BE NO BENES WHO HAD A CERTAIN
    EARNINGS TYPE IN A GIVEN MONTH -- IF THIS WERE TO HAPPEN THEN
    THERE WOULD BE NO YYYYMM VARIABLE FOR THAT EARNING TYPE SINCE THE
    PROC TRANSPOSE IN THE PREVIOUS STEP ONLY CREATES VARIABLES FOR
    DATA IT ENCOUNTERS;
  ARRAY OVARS {1,&NUMYRS.,12}
    %DO MM=1 %TO &TOTMOS; ALLGAMT&&YY&MM %END;;
RUN;

* SORT THE YYYYMM VARIABLES INTO THE RIGHT ORDER;
PROC CONTENTS DATA = TEMP2.ALGEARN OUT = TEMP2.VARS(KEEP = NAME);
RUN;

PROC SQL;
  SELECT NAME INTO :SORTED_COLS SEPARATED BY ' '
  FROM TEMP2.VARS
  ORDER BY NAME;
```

Appendix A.90

JCL/SAS Code: ALGEARN

```
QUIT;
```

```
DATA OUT.ALGEARN;  
RETAIN &SORTED_COLS;  
SET TEMP2.ALGEARN;  
RUN;
```

```
PROC CONTENTS DATA=OUT.ALGEARN;  
RUN;
```

```
%MEND START;  
%START;  
ENDSAS;
```

## Appendix A.91

JCL/SAS Code: T2WKDETN

```

//#6266T2W JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   BUILD DAF21 51390
//* PROJECT
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF21.ANN.PRDLIB(T2WKDETN)
//* DESCRIPTION: BUILDS THE YYMM EARNINGS VARIABLES FROM THE DCF
//*            T2WKDETN TABLE
//* PROGRAMMER: SERGE LUKASHANETS 06/23/2022
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//*
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DCFT2WKD.SA.V1,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.T2WKDETN.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP3 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN OBS=MAX;

/* CHANGE ENDYR AND ENDMN FOR EACH RUN */
%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

/* STEP TO ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA */
%LET NUMYRS=%EVAL(&ENDYR.-&BEGYR.+1);

%MACRO START;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

***** T2 WORK PROCESSING *****
DATA TEMP1.T2WKDETN;

```

## Appendix A.91

JCL/SAS Code: T2WKDET

```
SET IN1.T2WKDET;
```

\* IF THE EARNINGS DATE IS IN A REPORTING YEAR THEN KEEP;  
WHERE  
 (ERNGS\_DT>="01JAN&BEGYR."D) AND  
 (ERNGS\_DT<="31DEC&ENDYR."D) AND  
 (CID='00');

\* ATTACH FORMAT TO THE ERNGS\_DT FIELD SO IT WILL APPEAR AS YMMM;  
ATTRIB ERNGS\_DT FORMAT=YMMN4.;

```
RUN;
```

```
PROC SORT DATA=TEMP1.T2WKDET;  
  BY COSSN ERNGS_DT DESCENDING LU_TS;  
RUN;
```

```
PROC SORT DATA=TEMP1.T2WKDET NODUPKEYS;  
  BY COSSN ERNGS_DT;  
RUN;
```

\* CREATE ONE OBSERVATION PER SSN WITH YMMM VARIABLES CONTAINING EACH  
OR THE EARNINGS AMOUNTS;  
%MACRO T2WKD(ERNVAR,ERNPREF);  
PROC TRANSPOSE DATA=TEMP1.T2WKDET  
 OUT=TEMP2.&ERNPREF.(DROP = \_NAME\_ \_LABEL\_)  
 PREFIX=&ERNPREF.;  
 BY COSSN;  
 ID ERNGS\_DT;  
 VAR &ERNVAR.;

```
RUN;
```

```
%MEND T2WKD;
```

```
%T2WKD(SPCL_CDNS_AMT,T2CDNAMT);  
%T2WKD(WRK_EXP_AMT,T2EXPAMT);  
%T2WKD(SBDY_AMT,T2SBDYAMT);  
%T2WKD(SE_UNBIZEXP_AMT,T2UBEAMT);  
%T2WKD(WRK_DET_N_CD,TWPDATA);  
%T2WKD(FRAUD_VRFD_SW,T2FRAUDVER);
```

```
DATA TEMP3.T2WKDET;  
  MERGE TEMP2.T2CDNAMT  
        TEMP2.T2EXPAMT  
        TEMP2.T2SBDYAMT
```

Appendix A.91

JCL/SAS Code: T2WKDETN

```

        TEMP2.T2UBEAMT
        TEMP2.TWpdata
        TEMP2.T2FRAUDVER
    ;
    BY COSSN;
RUN;

* SORT THE YYMM VARIABLES INTO THE RIGHT ORDER;
PROC CONTENTS DATA = TEMP3.T2WKDETN OUT = TEMP3.VARS(KEEP = NAME);
RUN;

PROC SQL;
SELECT NAME INTO :SORTED_COLS SEPARATED BY ' '
FROM TEMP3.VARS
ORDER BY NAME;
QUIT;

DATA OUT.T2WKDETN;
RETAIN &SORTED_COLS;
SET TEMP3.T2WKDETN;
RUN;

PROC CONTENTS DATA=OUT.T2WKDETN;
RUN;

%MEND START;
%START;
ENDSAS;

//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 T2WKDETN IS COMPLETE

```

## Appendix A.92

JCL/SAS Code: T2EARN

```

//#6266T2 JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   BUILD DAF21 (51390)
//* PROJECT
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF21.ANN.PRDLIB(T2EARN)
//* DESCRIPTION: BUILDS THE YYMM EARNINGS VARIABLES FROM THE DCF
//*            T2EARN TABLE
//* PROGRAMMER: SERGE LUKASHANETS 06/23/2022
//*****
//*
//SAS   EXEC SAS9,
//     WORK='120000,60000'
//*
//
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.DCFT2ER.SA.V1,DISP=SHR
//BLD  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIEDCF.SA.V1,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.T2EARN.SA.V1,
//     DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//     UNIT=TSILO
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//     SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//     SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP3 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//     SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP4 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//     SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP5 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//     SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN OBS=MAX;

/* CHANGE ENDYR AND ENDMMN FOR EACH RUN */
%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

```

```

/* STEP TO ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA */
%LET NUMYRS=%EVAL(&ENDYR.-&BEGYR.+1);

```

```

%MACRO START;

```



## Appendix A.92

JCL/SAS Code: T2EARN

```
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

***** T2 EARNINGS PROCESSING *****,

DATA TEMP1.T2ERN;
  SET IN1.T2ERNGS;

  * IF THE EARNINGS DATE IS IN A REPORTING YEAR THEN KEEP;
  WHERE
    (ERNGS_DT>="01JAN&BEGYR."D) AND
    (ERNGS_DT<="31DEC&ENDYR."D) AND
    (CID='00');

  * ATTACH FORMAT TO THE ERNGS_DT FIELD SO IT WILL APPEAR AS YMM;
  ATTRIB ERNGS_DT FORMAT=YMMN4.;

RUN;

PROC SORT DATA=TEMP1.T2ERN;
  BY COSSN;
RUN;

* CREATE ONE OBSERVATION PER SSN WITH YMM VARIABLES CONTAINING EACH
  OR THE EARNINGS AMOUNTS;
%MACRO T2EARN(ERNVAR,ERNPREF);
PROC TRANSPOSE DATA=TEMP1.T2ERN
  OUT=TEMP2.&ERNPREF.(DROP = _NAME__LABEL_)
  PREFIX=&ERNPREF.;
  BY COSSN;
  ID ERNGS_DT;
  VAR &ERNVAR.;
RUN;

%MEND T2EARN;

%T2EARN(GRS_AMT,T2GRSAMT);
%T2EARN(SE_NET_AMT,T2NETAMT);
%T2EARN(SE_HRS,T2SEHRS);
%T2EARN(VRFD_IND,T2VERIND);
%T2EARN(SE_VRFD_IND,T2SEVERIND);

DATA TEMP3.T2EARN;
  MERGE TEMP2.T2GRSAMT
        TEMP2.T2NETAMT
        TEMP2.T2SEHRS
```

## Appendix A.92

JCL/SAS Code: T2EARN

```
        TEMP2.T2VERIND
        TEMP2.T2SEVERIND
    ;
    BY COSSN;
RUN;

* SORT THE YYMM VARIABLES INTO THE RIGHT ORDER;
PROC CONTENTS DATA = TEMP3.T2EARN OUT = TEMP3.VARS(KEEP = NAME);
RUN;

PROC SQL;
SELECT NAME INTO :SORTED_COLS SEPARATED BY ' '
FROM TEMP3.VARS
ORDER BY NAME;
QUIT;

DATA TEMP4.T2EARN;
RETAIN &SORTED_COLS;
SET TEMP3.T2EARN;
RUN;

PROC CONTENTS DATA=TEMP4.T2EARN;
RUN;

***** BUILD DIERNLVL *****;
DATA TEMP5.T2EARN;
MERGE TEMP4.T2EARN (IN=DCF)
      BLD.MIEDCF (KEEP = SSN BLINDDT RENAME=(SSN=COSSN));
BY COSSN;
IF DCF;

* SET INPUT ARRAY *;
ARRAY T2GRSAMT (&BEGYR.:&ENDYR.,1:12)
      %DO YEAR=&BEGYR %TO &ENDYR;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      T2GRSAMT&YR.01-T2GRSAMT&YR.12
      %END;;

ARRAY T2NETAMT (&BEGYR.:&ENDYR.,1:12)
      %DO YEAR=&BEGYR %TO &ENDYR;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      T2NETAMT&YR.01-T2NETAMT&YR.12
      %END;;

ARRAY T2EXPAMT (&BEGYR.:&ENDYR.,1:12)
```

## Appendix A.92

JCL/SAS Code: T2EARN

```
%DO YEAR=&BEGYR %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
T2EXPAMT&YR.01-T2EXPAMT&YR.12
%END;;
```

```
ARRAY T2SBDYAMT (&BEGYR.:&ENDYR.,1:12)
%DO YEAR=&BEGYR %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
T2SBDYAMT&YR.01-T2SBDYAMT&YR.12
%END;;
```

```
ARRAY T2CDNAMT (&BEGYR.:&ENDYR.,1:12)
%DO YEAR=&BEGYR %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
T2CDNAMT&YR.01-T2CDNAMT&YR.12
%END;;
```

```
* SET TWP VALUES *;
* VALUES FROM HTTPS://WWW.SSA.GOV/OACT/COLA/TWP.HTML *;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(TWPA)";
```

```
* SET SGA VALUES - NON-BLIND *;
* VALUES FROM HTTPS://WWW.SSA.GOV/OACT/COLA/SGA.HTML *;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(SGANB)";
```

```
* SET SGA VALUES - BLIND *;
* VALUES FROM HTTPS://WWW.SSA.GOV/OACT/COLA/SGA.HTML *;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(SGAB)";
```

```
* SET UP OUTPUT ARRAY *;
ARRAY DIERNLVL (&BEGYR.:&ENDYR.,1:12) $
%DO YEAR=&BEGYR %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
DIERNLVL&YR.01-DIERNLVL&YR.12
%END;;
```

```
DO YR=&BEGYR. TO &ENDYR.;
DO MO = 1 TO 12;
* MONTHLY TOTAL SSDI GROSS EARNINGS AMOUNT *;
T2TENAMT = SUM(T2GRSAMT(YR,MO),T2NETAMT(YR,MO));
```

```
* MONTHLY TOTAL SSDI INCOME EXCLUSIONS *;
T2INCEXL = SUM(T2EXPAMT(YR,MO),
T2SBDYAMT(YR,MO),
T2CDNAMT(YR,MO));
```

## Appendix A.92

JCL/SAS Code: T2EARN

```

* MONTHLY SSDI COUNTABLE EARNINGS *;
T2CNTERN = SUM(T2TENAMT,T2INCEXL*-1);

* CREATE DIERNLVL *;
IF T2TENAMT = 0 THEN DIERNLVL(YR,MO) = "0";
ELSE IF 0 < T2TENAMT < TWPA(YR)
  THEN DIERNLVL(YR,MO) = "L";

ELSE IF MONTH(BLINDDT) NE . AND
  MDY(MO,1,YR) >= BLINDDT THEN DO;
  IF TWPA(YR) <= T2TENAMT < SGAB(YR)
    THEN DIERNLVL(YR,MO) = "T";
  ELSE IF T2TENAMT >= SGAB(YR) AND
    T2CNTERN < SGAB(YR)
    THEN DIERNLVL(YR,MO) = "G";
  ELSE IF T2CNTERN >= SGAB(YR)
    THEN DIERNLVL(YR,MO) = "S";
END;

ELSE IF MDY(MO,1,YR) < BLINDDT
  OR MONTH(BLINDDT)=. THEN DO;
  IF TWPA(YR) <= T2TENAMT < SGANB(YR)
    THEN DIERNLVL(YR,MO) = "T";
  ELSE IF T2TENAMT >= SGANB(YR) AND
    T2CNTERN < SGANB(YR)
    THEN DIERNLVL(YR,MO) = "G";
  ELSE IF T2CNTERN >= SGANB(YR)
    THEN DIERNLVL(YR,MO) = "S";
END;
END; /* END OF MO LOOP */
END; /* END OF YR LOOP */
RUN;

DATA OUT.T2EARN;
  SET TEMP5.T2EARN;
RUN;

PROC FORMAT;
  VALUE AMTV
  0          = "$0"
  0-<100     = "$0 - LESS THAN $100"
  %DO I = 1 %TO 300;
  %EVAL(100*&I)-<%EVAL(100*(&I+1))
  = "$ %EVAL(100*&I) - LESS THAN $ %EVAL(100*(&I+1))"

```

Appendix A.92

JCL/SAS Code: T2EARN

```

%END;;
RUN;

* CHECK FREQ OF DIERNLVL VARIABLE *;
TITLE1 "CROSSTAB OF DIERNLVL AND TOTAL SSDI GROSS EARNINGS AMOUNT";
TITLE2 "FOR MARCH,JUNE,SEPTEMBER AND DECEMBER";
%DO YEAR=&BEGYR. %TO &ENDYR.;
%LET YR=%SUBSTR(&YEAR.,3,2);
%LET MO=03 06 09 12;
%DO I=1 %TO 4;
%LET VAR=%SCAN(&MO,&I);
PROC FREQ DATA=OUT.T2EARN;
  TABLES DIERNLVL&YR.&VAR*T2TENAMT / LIST MISSING;
  FORMAT T2TENAMT AMTV.;
RUN;
%END; /*END OF MO LOOP*/
%END; /*END OF YEAR LOOP*/
TITLE;

%MEND;
%START;
ENDSAS;

//*****
/* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: T2EARN IS COMPLETE

```

## Appendix A.93

JCL/SAS Code: SSIERNLV

```
//#6266SSI JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT: BUILD DAF21 (51390)
//* PROJECT
//* DIRECTOR:
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF21.ANN.PRDLIB(SSIERNLV)
//*
//* DESCRIPTION: BUILDS THE YMM EARNINGS VARIABLES FROM THE SSR
//*      EARNINGS FILE
//*
//* UPDATED:   SERGE LUKASHANETS 6/30/2022 FOR DAF21
//*****
//*
//SAS   EXEC SAS9,
//      WORK='120000,60000'
//*
//*
//IN1   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2P2.SA.V1,DISP=SHR
//BLD   DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIEDCF.SA.V1,DISP=SHR
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSRERNLV.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP3 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP4 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

/* ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA */
%LET NUMYRS=%EVAL(&ENDYR.-&BEGYR.+1);

%MACRO START;
```

## Appendix A.93

JCL/SAS Code: SSIERNLV

```
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

***** BUILD SSIERNLV *****;

DATA OUT.SSREARN;
  MERGE IN1.LONG (IN=SSR)
        BLD.MIEDCF (IN=BLD KEEP = SSN BLINDDT);
  BY SSN;

  IF SSR;

  * SET INPUT ARRAY *;
  * IETYP = W;
  ARRAY T16GRSAMT (&BEGYR.:&ENDYR.,1:12)
         %DO YEAR=&BEGYR %TO &ENDYR;
         %LET YR=%SUBSTR(&YEAR.,3,2);
         T16GRSAMT&YR.01-T16GRSAMT&YR.12
         %END;;
  * IETYP = N;
  ARRAY T16NETAMT (&BEGYR.:&ENDYR.,1:12)
         %DO YEAR=&BEGYR %TO &ENDYR;
         %LET YR=%SUBSTR(&YEAR.,3,2);
         T16NETAMT&YR.01-T16NETAMT&YR.12
         %END;;
  * IETYP = T;
  ARRAY T16EXPAMT (&BEGYR.:&ENDYR.,1:12)
         %DO YEAR=&BEGYR %TO &ENDYR;
         %LET YR=%SUBSTR(&YEAR.,3,2);
         T16EXPAMT&YR.01-T16EXPAMT&YR.12
         %END;;
  * IETYP = D;
  ARRAY T16PASAMT (&BEGYR.:&ENDYR.,1:12)
         %DO YEAR=&BEGYR %TO &ENDYR;
         %LET YR=%SUBSTR(&YEAR.,3,2);
         T16PASAMT&YR.01-T16PASAMT&YR.12
         %END;;
  * IETYP = B;
  ARRAY T16EXLAMT (&BEGYR.:&ENDYR.,1:12)
         %DO YEAR=&BEGYR %TO &ENDYR;
         %LET YR=%SUBSTR(&YEAR.,3,2);
         T16EXLAMT&YR.01-T16EXLAMT&YR.12
         %END;;
  * IETYP = C;
  ARRAY T16BEXPAMT (&BEGYR.:&ENDYR.,1:12)
         %DO YEAR=&BEGYR %TO &ENDYR;
```

## Appendix A.93

JCL/SAS Code: SSIERNLV

```
%LET YR=%SUBSTR(&YEAR.,3,2);
T16BEXPAMT&YR.01-T16BEXPAMT&YR.12
%END;;

* IETYP = S;
ARRAY T16SEAMT (&BEGYR.:&ENDYR.,1:12)
%DO YEAR=&BEGYR %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
T16SEAMT&YR.01-T16SEAMT&YR.12
%END;;

* SET FBR VALUES *;
* VALUES FROM HTTPS://WWW.SSA.GOV/OACT/COLA/SSIAMTS.HTML *;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(FBRIN)";

* SET SGA VALUES - NON-BLIND *;
* VALUES FROM HTTPS://WWW.SSA.GOV/OACT/COLA/SGA.HTML *;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(SGANB)";

* SET SGA VALUES - BLIND *;
* VALUES FROM HTTPS://WWW.SSA.GOV/OACT/COLA/SGA.HTML *;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(SGAB)";

* SET UP OUTPUT ARRAY *;
ARRAY SSIERNLVL (&BEGYR.:&ENDYR.,1:12) $
%DO YEAR=&BEGYR %TO &ENDYR;
%LET YR=%SUBSTR(&YEAR.,3,2);
SSIERNLVL&YR.01-SSIERNLVL&YR.12
%END;;

DO YR=&BEGYR. TO &ENDYR.;
DO MO = 1 TO 12;
* MONTHLY TOTAL SSI GROSS EARNINGS AMOUNT *;
T16TENAMT = SUM(T16GRSAMT(YR,MO),
T16SEAMT(YR,MO),
T16NETAMT(YR,MO)*-1);

* MONTHLY TOTAL SSI INCOME EXCLUSIONS *;
T16INCEXL = SUM(T16EXLAMT(YR,MO),
T16BEXPAMT(YR,MO),
T16PASAMT(YR,MO),
T16EXPAMT(YR,MO));

* MONTHLY SSI COUNTABLE EARNINGS *;
T16CNTERN = SUM(T16TENAMT,T16INCEXL*-1);
```



## Appendix A.93

JCL/SAS Code: SSIERNLV

```
* CREATE SSIERNLVL *;
IF T16TENAMT = 0 THEN SSIERNLVL(YR,MO) = "0";
ELSE IF 0 < T16TENAMT < FBRIN(YR)
  THEN SSIERNLVL(YR,MO) = "L";

ELSE IF MONTH(BLINDDT) NE . AND
  MDY(MO,1,YR) >= BLINDDT THEN DO;
IF FBRIN(YR) <= T16TENAMT < SGAB(YR) AND
  FBRIN(YR) > T16CNTERN
  THEN SSIERNLVL(YR,MO) = "F";
ELSE IF FBRIN(YR) <= T16TENAMT < SGAB(YR) AND
  FBRIN(YR) <= T16CNTERN
  THEN SSIERNLVL(YR,MO) = "B";
ELSE IF T16TENAMT >= SGAB(YR) AND
  T16CNTERN < FBRIN(YR)
  THEN SSIERNLVL(YR,MO) = "A";
ELSE IF T16TENAMT >= SGAB(YR) AND
  FBRIN(YR) <= T16CNTERN < SGAB(YR)
  THEN SSIERNLVL(YR,MO) = "G";
ELSE IF T16CNTERN >= SGAB(YR)
  THEN SSIERNLVL(YR,MO) = "S";
END;

ELSE IF MDY(MO,1,YR) < BLINDDT
  OR MONTH(BLINDDT)=. THEN DO;
IF FBRIN(YR) <= T16TENAMT < SGANB(YR) AND
  FBRIN(YR) > T16CNTERN
  THEN SSIERNLVL(YR,MO) = "F";
ELSE IF FBRIN(YR) <= T16TENAMT < SGANB(YR) AND
  FBRIN(YR) <= T16CNTERN
  THEN SSIERNLVL(YR,MO) = "B";
ELSE IF T16TENAMT >= SGANB(YR) AND
  T16CNTERN < FBRIN(YR)
  THEN SSIERNLVL(YR,MO) = "A";
ELSE IF T16TENAMT >= SGANB(YR) AND
  FBRIN(YR) <= T16CNTERN < SGANB(YR)
  THEN SSIERNLVL(YR,MO) = "G";
ELSE IF T16CNTERN >= SGANB(YR)
  THEN SSIERNLVL(YR,MO) = "S";
END;
END; /* END OF MO LOOP */
END; /* END OF YR LOOP */
RUN;

%MEND START;
```

Appendix A.93

JCL/SAS Code: SSIERNLV

```
%START;
ENDSAS;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 SSIERNLV IS COMPLETE
```

## Appendix A.94

JCL/SAS Code: ALLEARN

```
//#6266EAR JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   BUILD DAF21 (51390)
//* PROJECT
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF21.ANN.PRDLIB(ALLEARN)
//* PURPOSE:   COMBINE DCF EARNINGS DATA
//* PROGRAMMER: LUKASHANETS 6/30/2022 FOR DAF21
//* E-MAIL     SLUKASHANETS@MATHEMATICA-MPR.COM
//*****
//JS030 EXEC SAS9,
//      WORK='120000,60000'          * SPACE IN BLKS *
//*
//T2E  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.T2EARN.SA.V1,DISP=SHR
//T2W  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.T2WKDET.N.SA.V1,DISP=SHR
//SSI  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSRERNLV.SA.V1,DISP=SHR
//ALG  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ALGEARN.SA.V1,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ALLEARN.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY MPRINT OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

/* STEP TO ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA */
%MACRO START;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

DATA OUT.ALLEARN (RENAME=(COSSN=SSN));
MERGE
T2E.T2EARN
( KEEP=COSSN
%DO MM=1 %TO &TOTMOS;
  T2GRSAMT&&YY&MM
  T2NETAMT&&YY&MM
  T2SEHRS&&YY&MM
  T2SEVERIND&&YY&MM
  T2VERIND&&YY&MM
  DIERNLVL&&YY&MM
%END;
```

Appendix A.94

JCL/SAS Code: ALLEARN

```

)
T2W.T2WKDETN
( KEEP=COSSN
%DO MM=1 %TO &TOTMOS;
  TWPDATA&&YY&MM
  T2CDNAMT&&YY&MM
  T2EXPAMT&&YY&MM
  T2FRAUDVER&&YY&MM
  T2SBDYAMT&&YY&MM
  T2UBEAMT&&YY&MM
%END;
)
SSI.SSREARN
( KEEP=SSN
%DO MM=1 %TO &TOTMOS;
  T16BEXPAMT&&YY&MM
  T16EXLAMT&&YY&MM
  T16EXPAMT&&YY&MM
  T16GRSAMT&&YY&MM
  T16NETAMT&&YY&MM
  T16PASAMT&&YY&MM
  T16SEAMT&&YY&MM
  SSIERNLVL&&YY&MM
%END;
  RENAME=(SSN=COSSN)
)
ALG.ALGEARN
( KEEP=COSSN
%DO MM=1 %TO &TOTMOS;
  ALLGAMT&&YY&MM
%END;
)
;
  BY COSSN;
RUN;
%MEND;
%START;

PROC CONTENTS DATA=OUT.ALLEARN;
RUN;

//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*

```

Appendix A.94

JCL/SAS Code: ALLEARN

```
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 ALLEARN ARE COMPLETE
```

## Appendix A.95

JCL/SAS Code: SPALLERN

```
//#7429MSP JOB (12510000,T715,,SAS,,ITC9FL),COLIP,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#7429
//*****
//*
//* *-----*
//*  ! THE MACRO SPLIT PROGRAM          !
//* *-----*
//*
//*
//* CREATED BY: SERGE LUKASHANETS, 5/26/2016
//* MODIFIED BY: COLIP 11/08/2021 FOR DAF20 REPROCESSING
//* E-MAIL   BCOLIP@MATHEMATICA-MPR.COM
//*****
// SET REG='64M'
//*
//JS010 EXEC SAS9,
//      WORK='200000,100000', F10
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
//***DATA FOR SPLIT
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ALLEARN.SA.V1,DISP=SHR
//***OUT YEARLY FILES
//OUT94 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN1994.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT95 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN1995.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT96 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN1996.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT97 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN1997.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT98 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN1998.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT99 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN1999.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT00 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2000.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
```

## Appendix A.95

JCL/SAS Code: SPALLERN

```
//OUT01 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2001.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT02 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2002.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT03 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2003.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT04 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2004.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT05 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2005.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT06 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2006.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT07 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2007.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT08 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2008.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT09 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2009.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2010.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT11 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2011.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2012.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT13 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2013.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT14 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2014.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT15 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2015.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
```

## Appendix A.95

JCL/SAS Code: SPALLERN

```
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2016.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2017.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2018.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2019.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.#7429.DAF20T.ERN2020.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT OBS=MAX;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF20.UTILITY.PRDLIB(MSPLIT)";

%OUTP(IN1.ALLEARN,OUT,ERN,1994,2020,SSN);

PROC CONTENTS DATA=OUT94.ERN;RUN;
PROC CONTENTS DATA=OUT95.ERN;RUN;
PROC CONTENTS DATA=OUT96.ERN;RUN;
PROC CONTENTS DATA=OUT97.ERN;RUN;
PROC CONTENTS DATA=OUT98.ERN;RUN;
PROC CONTENTS DATA=OUT99.ERN;RUN;
PROC CONTENTS DATA=OUT00.ERN;RUN;
PROC CONTENTS DATA=OUT01.ERN;RUN;
PROC CONTENTS DATA=OUT02.ERN;RUN;
PROC CONTENTS DATA=OUT03.ERN;RUN;
PROC CONTENTS DATA=OUT04.ERN;RUN;
PROC CONTENTS DATA=OUT05.ERN;RUN;
PROC CONTENTS DATA=OUT06.ERN;RUN;
PROC CONTENTS DATA=OUT07.ERN;RUN;
PROC CONTENTS DATA=OUT08.ERN;RUN;
PROC CONTENTS DATA=OUT09.ERN;RUN;
PROC CONTENTS DATA=OUT10.ERN;RUN;
PROC CONTENTS DATA=OUT11.ERN;RUN;
PROC CONTENTS DATA=OUT12.ERN;RUN;
PROC CONTENTS DATA=OUT13.ERN;RUN;
PROC CONTENTS DATA=OUT14.ERN;RUN;
PROC CONTENTS DATA=OUT15.ERN;RUN;
PROC CONTENTS DATA=OUT16.ERN;RUN;
```



Appendix A.95

JCL/SAS Code: SPALLERN

```
PROC CONTENTS DATA=OUT17.ERN;RUN;
PROC CONTENTS DATA=OUT18.ERN;RUN;
PROC CONTENTS DATA=OUT19.ERN;RUN;
PROC CONTENTS DATA=OUT20.ERN;RUN;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<BCOLIP@MATHEMATICA-MPR.COM>
RCPT TO:<BCOLIP@MATHEMATICA-MPR.COM>
DATA
FROM: <BCOLIP@MATHEMATICA-MPR.COM>
TO: <BCOLIP@MATHEMATICA-MPR.COM>
SUBJECT: DAF20 SPALLERN IS COMPLETE
```

## Appendix A.96

JCL/SAS Code: JCLYRSP

```
//#6266MSP JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//*
//* *-----*
//* | THE MACRO SPLIT PROGRAM           |
//* *-----*
//*
//*
//* CREATED BY: SERGE LUKASHANETS, 5/26/2016
//* MODIFIED BY: LUKASANETS 06/30/2022 FOR DAF21 (RERUN)
//* E-MAIL   SLUKASHANETS@MATHEMATICA-MPR.COM
//*****
// SET REG='64M'
//*
//JS010 EXEC SAS9,
//      WORK='200000,100000', F10
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
//***DATA FOR SPLIT
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ALLEARN.SA.V1,DISP=SHR
//***OUT YEARLY FILES
//OUT94 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1994.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT95 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1995.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT96 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1996.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT97 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1997.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT98 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1998.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT99 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1999.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT00 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2000.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
```

## Appendix A.96

JCL/SAS Code: JCLYRSP

```
//OUT01 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2001.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT02 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2002.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT03 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2003.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT04 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2004.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT05 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2005.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT06 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2006.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT07 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2007.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT08 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2008.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT09 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2009.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2010.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT11 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2011.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2012.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT13 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2013.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT14 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2014.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
//OUT15 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2015.SA.V1,
//     DISP=(NEW,CATLG,DELETE),
//     UNIT=TSILO,VOL=(,,10)
```

## Appendix A.96

JCL/SAS Code: JCLYRSP

```
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2016.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2017.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2018.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2019.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2020.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2021.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT OBS=MAX;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(MSPLIT)";

%OUTP(IN1.ALLEARN,OUT,ERN,1994,2021,SSN);

PROC CONTENTS DATA=OUT94.ERN;RUN;
PROC CONTENTS DATA=OUT95.ERN;RUN;
PROC CONTENTS DATA=OUT96.ERN;RUN;
PROC CONTENTS DATA=OUT97.ERN;RUN;
PROC CONTENTS DATA=OUT98.ERN;RUN;
PROC CONTENTS DATA=OUT99.ERN;RUN;
PROC CONTENTS DATA=OUT00.ERN;RUN;
PROC CONTENTS DATA=OUT01.ERN;RUN;
PROC CONTENTS DATA=OUT02.ERN;RUN;
PROC CONTENTS DATA=OUT03.ERN;RUN;
PROC CONTENTS DATA=OUT04.ERN;RUN;
PROC CONTENTS DATA=OUT05.ERN;RUN;
PROC CONTENTS DATA=OUT06.ERN;RUN;
PROC CONTENTS DATA=OUT07.ERN;RUN;
PROC CONTENTS DATA=OUT08.ERN;RUN;
PROC CONTENTS DATA=OUT09.ERN;RUN;
PROC CONTENTS DATA=OUT10.ERN;RUN;
PROC CONTENTS DATA=OUT11.ERN;RUN;
PROC CONTENTS DATA=OUT12.ERN;RUN;
PROC CONTENTS DATA=OUT13.ERN;RUN;
```

Appendix A.96

JCL/SAS Code: JCLYRSP

```
PROC CONTENTS DATA=OUT14.ERN;RUN;
PROC CONTENTS DATA=OUT15.ERN;RUN;
PROC CONTENTS DATA=OUT16.ERN;RUN;
PROC CONTENTS DATA=OUT17.ERN;RUN;
PROC CONTENTS DATA=OUT18.ERN;RUN;
PROC CONTENTS DATA=OUT19.ERN;RUN;
PROC CONTENTS DATA=OUT20.ERN;RUN;
PROC CONTENTS DATA=OUT21.ERN;RUN;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SLUKASHANETS@MATHEMATICA-MPR.COM>
RCPT TO:<SLUKASHANETS@MATHEMATICA-MPR.COM>
DATA
FROM: <SLUKASHANETS@MATHEMATICA-MPR.COM>
TO: <SLUKASHANETS@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 JCLYRSP IS COMPLETE
```

## Appendix A.97

JCL/SAS Code: FIPS\_FORMAT

```

/*****
*   PROJECT NAME: DAF21                               *
*   SUBTASK: CREATE FLAT FILE OF ZIP-FIPS CROSSWALK   *
* PROJECT-TASK NUMBER: 50214                           *
*   PROGRAMMER: MIRIAM LOEWENBERG                     *
*   DATE WRITTEN: 19 AUG 2022                         *
*   INPUT FILE(S): MCZ.SAS7BDAT                       *
*                                                     *
* THIS PROGRAM READS SAS FILE AND OUTPUTS FLAT FILE for ZIP-FIP CONV *
* FILE IS FROM INTRANET, SEARCH FOR PZLAFMT AND DOWNLOADED ONE      *
* WITH MOST RECENT DATE                                           *
*****/
OPTIONS PS=60 LS=132 NOCENTER FORMCHAR="|----|+|---+=|-\<>*" ;

%let inpath = M:\DAF21\TASK 12 Annuals Pre-Processing\fips;

proc import
  datafile = "&inpath.\mcz.csv"
  out = mcz
  dbms = csv
  replace;
run;
/*
proc contents data=mcz;
run;

proc freq data=mcz;
  table zipcode statefips countyfips/list missing nocum;
run;*/

/* Convert zipcode, statefips, and countyfips to character and add leading 0s */
data mcz_codes;
  set mcz;
  zipcode_char = put(zipcode, z5.);
  statefips_char = put(statefips, z2.);
  countyfips_char = put(countyfips, z3.);
run;
/*
proc freq data=mcz_codes;
  table zipcode_char statefips_char countyfips_char/list missing nocum;
run;*/

data mcz_recoded;
  set mcz_codes;

```

## Appendix A.97

JCL/SAS Code: FIPS\_FORMAT

```
drop zipcode statefips countyfips;
rename zipcode_char = zipcode;
rename statefips_char = statefips;
rename countyfips_char = countyfips;
run;

proc contents data=mcz_recoded; run;
proc freq data=mcz_recoded;
  table zipcode statefips countyfips/list missing nocum;
run;

proc sort data=mcz_recoded;
  by zipcode
     descending percent;
run;
DATA ZIP (KEEP=ZIPCODE FIPSCODE);
  SET mcz_recoded;
  FIPSCODE=CATS(STATEFIPS,COUNTYFIPS);
  LABEL ZIPCODE="ZIPCODE"
        FIPSCODE="FIPSCODE";
RUN;

* ZIP CODES CAN SPAN MORE THAN ONE FIPS CODE;
* WE KEEP THE ZIP CODE APPEARING FIRST IN THE DATA;

PROC SORT DATA=ZIP NODUPKEY; BY ZIPCODE; RUN;

options comamid=tcp; %let SY56=SY56.sspf.ssa.gov; %let tcpsec=_prompt_; options remote=sy56.spawner; run;

* UPLOAD DATA TO MAINFRAME *;
SIGNON;
RSUBMIT;

LIBNAME FIPS 'OPDR.TG.PRD.ETTW.#6266.DAF21.FIPSRMT.SA.V1';
OPTION COMPRESS=BINARY REMOTE=session_id;

PROC UPLOAD DATA=ZIP
  OUT=FIPS.FIPSRMT(COMPRESS=BINARY);
RUN;

ENDRSUBMIT;
SIGNOFF;
```

## Appendix A.98

JCL/SAS Code: FIPSF0RM

```
//#6266PVF JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT: BUILD DAF21 (50214)
//* PROJECT
//* DIRECTOR:  CASSANDRA PICKENS JEWELL
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF21.ANN.PRDLIB(FIPSF0RM)
//*
//* DESCRIPTION: CREATE FIPS FORMATS FROM PROVIDER ZIP/FIPS CROSSWALK
//*
//* PROGRAMMER: LUKASHANETS 09/07/2022
//*****
//STEP1  EXEC SAS9
//TEMP1  DD DSN=*&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,
//        VOL=(,,10)
//FIPS   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.FIPSF0RMT.SA.V1,DISP=SHR
//LIBRARY DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.FIPS.FMTLIB,
//        DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//        SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES;
/* THIS PROGRAM READS SORD CREATE FORMAT STATEMENT
   FROM UPDATED ZIP-FIPS FILE */
PROC PRINT DATA=FIPS.FIPSF0RMT (OBS=25); RUN;

*CREATES ZIP-FIPS FORMAT USING DATA SET CROSSWALK;
*"IF _N_ = 1" SECTION HANDLES ZIP CODES THAT DO NOT APPEAR IN CROSSWALK;

DATA TEMP1.FIPS;
SET FIPS.FIPSF0RMT(RENAME=(ZIPCODE=START FIPSCODE=LABEL));
FMTNAME='$FIPS';
OUTPUT;
IF _N_ = 1 THEN DO;
START='OTHER';
LABEL = ' ';
OUTPUT;
END;
RUN;

PROC FORMAT CNTLIN=TEMP1.FIPS LIBRARY=LIBRARY FMTLIB;
RUN;
```



## Appendix A.99

JCL/SAS Code: ANNCURR

```
//#8047ANN JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNCURR)
//* PURPOSE: CREATE YEARLY FILES FOR CURRENT YEAR WITH MONTHLY
//*      FIELDS FROM JAN TO DEC
//* UPDATE: MIRIAM CHAPPELKA
//* EMAIL: MCHAPPELKA@MATHEMATICA-MPR.COM
//*      MIRIAM.CHAPPELKA@SSA.GOV
//*****
//JS030 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2021"'
//*
//* SSI LONGITUDINAL FILE
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIELIG.SA.V1,DISP=SHR
//* MBR DATA
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBRREST.SA.V1,DISP=SHR
//* 831-833 DATA
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.ALLCDCOM.SA.V1,DISP=SHR
//* DCF DATA
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ALLEARN.SA.V1,DISP=SHR
//* CER FILE
//CER DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.COMBCERE.SA.V1,DISP=SHR
//* DX FILE
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPROC.SA.V1,DISP=SHR
//* DBAD FILE
//DBAD DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.CMBDBDE.SA.V1,DISP=SHR
//* DEMO FILE
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//* PREVIOUS YEAR FILE
//OLD DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2020E,DISP=SHR
//* FIPS FORMAT LIBRARY
//LIBRARY DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21.FIPS.FMTLIB,DISP=SHR
//* MIECDF
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2021.SA.V1,DISP=SHR
//*
//TEMP DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2021.SA.V1,
//      DISP=(OLD,CATLG,DELETE),
```

## Appendix A.99

JCL/SAS Code: ANNCURR

```
// UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2021CON.SA.V1,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//OUT3 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.PSTM.R.SA.V1,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY MPRINT OBS=MAX
      FMTSEARCH=(LIBRARY);

*** CHANGE ENDYR AND ENDDMMN FOR EACH RUN;

%LET YR = SUBSTR(&SYSPARM,3,2);
%LET PYR = %SUBSTR(%EVAL(&SYSPARM-1),3,2);

%LET BEGYR=&SYSPARM;
%LET ENDYR=&SYSPARM; /* CHANGE AS NEEDED */
%LET ENDDMMN=12; /* CHANGE AS NEEDED */

* CHECK AND CHANGE AS NEEDED NUBER OF HI AND SMI OCCURRENCES;
* FIND IT FROM THE MBR SOURCE LAYOUT - MBRREAD PROGRAM;
%LET NUMHI = 10;
%LET NUMSMI = 20;

/* STEP TO ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA */
%MACRO START;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

/* DELETE DUPLICATE SSNs IN DBAD FILE */
PROC SORT DATA = DBAD.DBAD OUT=TEMP.DBAD NODUPKEY;
BY SSN;
RUN;

DATA OUT1.Y&SYSPARM
      OUT3.PSTM.R(KEEP=SSN PST_MR);
MERGE
      FIN.DMSTW (IN=INFIN KEEP=SSN DODBEST DOBBEST STATE_MR
                RENAME=(STATE_MR=STATE_MR_DM))
      SSR.SSIELIG(KEEP=SSN MFT SSI:
                %DO MM=1 %TO &TOTMOS;
                EICM&&YY&MM FAMT&&YY&MM FPMT&&YY&MM
                SPMT&&YY&MM SAMT&&YY&MM UINC&&YY&MM
```

## Appendix A.99

JCL/SAS Code: ANNCURR

```
LIVF&&YY&MM MTST&&YY&MM PSTA&&YY&MM
SCON&&YY&MM CURCOMP&&YY&MM STUDEM&&YY&MM
%END;)
MBR.MBR(KEEP=SSN HI_START: HI_TERM: SMI_STAR: SMI_TERM: SSDI:
%DO MM=1 %TO &TOTMOS;
  DUED&&YY&MM LAF&&YY&MM WIC&&YY&MM
  RFD&&YY&MM RFST&&YY&MM BPD&&YY&MM
  DPEN&&YY&MM DUEO&&YY&MM PAYD&&YY&MM
  PAYO&&YY&MM MBA&&YY&MM MBP&&YY&MM
  CDR&&YY&MM MEDPREM&&YY&MM DIRPAY&&YY&MM
%END;)
T831.ALLCDCOM(KEEP=SSN
%DO MM=1 %TO &TOTMOS;
  ALX&&YY&MM EDX&&YY&MM
%END;)
ERN.ALLEARN(KEEP=SSN
%DO MM=1 %TO &TOTMOS;
  T2GRSAMT&&YY&MM T2NETAMT&&YY&MM
  T2SEHRS&&YY&MM T2SEVERIND&&YY&MM
  T2VERIND&&YY&MM DIERNLVL&&YY&MM
  TWPDATA&&YY&MM T2CDNAMT&&YY&MM
  T2EXPAMT&&YY&MM T2FRAUDVER&&YY&MM
  T2SBDYAMT&&YY&MM T2UBEAMT&&YY&MM
  T16BEXPAMT&&YY&MM T16EXLAMT&&YY&MM
  T16EXPAMT&&YY&MM T16GRSAMT&&YY&MM
  T16NETAMT&&YY&MM T16PASAMT&&YY&MM
  T16SEAMT&&YY&MM SSIERNLVL&&YY&MM
  ALLGAMT&&YY&MM
%END;)
CER.CER(KEEP=SSN
%DO MM=1 %TO &TOTMOS;
  IEA_BLIND&&YY&MM IEA_PASS&&YY&MM IEA_LOSS&&YY&MM
  IEA_NESE&&YY&MM IEA_IRWE&&YY&MM IEA_WAGE&&YY&MM
  IEA_SEIE&&YY&MM
  IET4&&YY&MM MEDC&&YY&MM IUA_SSDI&&YY&MM
  IUA_WC&&YY&MM IUA_TANF&&YY&MM PAYEEZIP&&YY&MM
%END;
  RENAME=(%DO MM=1 %TO &TOTMOS;
    PAYEEZIP&&YY&MM=ZIP_SSI&&YY&MM
  %END;))
DXP.DXPROC(KEEP=SSN
%DO MM=1 %TO &TOTMOS;
  DXPRIBEST&&YY&MM DXSECBEST&&YY&MM
  DXPRI_SSI&&YY&MM DXSEC_SSI&&YY&MM
  DXPRI_DI&&YY&MM DXSEC_DI&&YY&MM
```

## Appendix A.99

JCL/SAS Code: ANNCURR

```

        %END;)
TEMP.DBAD(KEEP=SSN
        %DO MM=1 %TO &TOTMOS; ZIP_DI&&YY&MM %END;)
MED.PATMED(KEEP=SSN
        %DO MM=1 %TO &TOTMOS; MEDEX&&YY&MM %END;)
OLD.Y20&PYR. (KEEP = SSN ZIP_DI&PYR.12)
;
BY SSN;

/* DELETE CASES NOT IN THE MASTER FINDER */
IF INFIN;

/* TO REDUCE THE SIZE OF THE ANNUALS FILES CRETE
   FLAGS FOR DECEASED INDIVIDUALS AND UNBORNED */
IF YEAR(DODBEST) NE . AND YEAR(DODBEST) < &ENDYR THEN FL_DEATH=1;
ELSE FL_DEATH=0;
IF YEAR(DOBEST) NE . AND YEAR(DOBEST) > &ENDYR THEN FL_UNBORN=1;
ELSE FL_UNBORN=0;

/* NEW IN DAF15: REMOVE INDIVIDUALS WHO DIED BEFORE 1996 */
IF YEAR(DODBEST) NE . AND YEAR(DODBEST) < 1996 THEN DELETE;

ARRAY FPMT(*) %DO MM=1 %TO &TOTMOS; FPMT&&YY&MM %END;;
ARRAY FAMT(*) %DO MM=1 %TO &TOTMOS; FAMT&&YY&MM %END;;
ARRAY PSTA(*) %DO MM=1 %TO &TOTMOS; PSTA&&YY&MM %END;;
ARRAY MTST(*) %DO MM=1 %TO &TOTMOS; MTST&&YY&MM %END;;
ARRAY SPMT(*) %DO MM=1 %TO &TOTMOS; SPMT&&YY&MM %END;;
ARRAY SAMT(*) %DO MM=1 %TO &TOTMOS; SAMT&&YY&MM %END;;
ARRAY SCON(*) %DO MM=1 %TO &TOTMOS; SCON&&YY&MM %END;;
ARRAY DUED(*) %DO MM=1 %TO &TOTMOS; DUED&&YY&MM %END;;
ARRAY LAFCODE (*) %DO MM=1 %TO &TOTMOS; LAF&&YY&MM %END;;

ARRAY PAYS (*) %DO MM=1 %TO &TOTMOS; PAYS&&YY&MM %END;;
ARRAY DUES (*) %DO MM=1 %TO &TOTMOS; DUES&&YY&MM %END;;

/* REVISION 2009- TAKE THE ZERO OUT OF THE CODE */

DO I = 1 TO DIM(FPMT);
    PAYS(I) = SUM(OF FPMT(I),SPMT(I));
    DUES(I) = SUM(OF FAMT(I),SAMT(I));
END;

/* CONCURRENT BENEFICIARY INDICATOR */
/* REVISION 2008 - INDICATOR WILL FLAG BENEFICIARIES RECEIVING
   A PAYMENT (IN CURRENT PAY) IN BOTH PROGRAMS, SSI AND SSDI */

```

## Appendix A.99

JCL/SAS Code: ANNCURR

```
ARRAY CONC (*) %DO MM=1 %TO &TOTMOS; CONC&&YY&MM %END;;
```

```
DO I = 1 TO DIM(DUED);  
  IF (DUES(I) > 0 AND PSTA(I) IN ('C01' 'M01' 'M02')) AND  
    (DUED(I) > 0 AND SUBSTR(LAFCODE(I),1,1) IN ('C' 'E'))  
  THEN CONC(I) = 1;  
  ELSE CONC(I) = 0;  
END;
```

```
/* CREATE MONTHLY INDICATORS FOR 1619A-1619B*/  
ARRAY PROA(*) %DO MM=1 %TO &TOTMOS; PROA&&YY&MM %END;;  
ARRAY PROB(*) %DO MM=1 %TO &TOTMOS; PROB&&YY&MM %END;;  
ARRAY EICM(*) %DO MM=1 %TO &TOTMOS; EICM&&YY&MM %END;;
```

```
/* PROB */  
/* REVISION 2008 - CREATE ONLY FOR SSI */  
DO I = 1 TO DIM(MTST);  
  IF MTST(I) IN ('A','B','F') AND  
    PSTA(I) IN ('N01', 'P01', 'E01') AND  
    EICM(I) > 0  
  THEN PROB(I) = 1;  
  ELSE IF PSTA(I) NE ' ' THEN PROB(I) = 0;  
END;
```

```
/* FOR PROA (THIS SHOULD YIELD NO CASES WHERE PROB = 1  
AND PROA = 1) */  
DO I = 1 TO DIM(SCON);  
  IF SCON(I) IN ('D','E','F','G') AND  
    PSTA(I) IN ('C01', 'M01','M02') AND  
    DUES(I) > 0 AND  
    EICM(I) > 0  
  THEN PROA(I) = 1;  
  ELSE IF PSTA(I) NE ' ' THEN PROA(I) = 0;  
END;
```

```
/* CREATE A MEDICARE INDICATOR */  
/* INPUT ARRAYS */  
ARRAY HI_START(*) HI_START1-HI_START&NUMHI.;  
ARRAY HI_TERM(*) HI_TERM1-HI_TERM&NUMHI.;  
ARRAY SMI_START(*) SMI_STAR1-SMI_STAR&NUMSMI.;  
ARRAY SMI_TERM(*) SMI_TERM1-SMI_TERM&NUMSMI.;
```

```
/* OUTPUT ARRAYS */  
ARRAY MEDR (*) %LET YR=%SUBSTR(&BEGYR.,3,2);  
  MEDR&YR.01-MEDR&YR.12;
```

## Appendix A.99

JCL/SAS Code: ANNCURR

```

/* FIND THE EARLIEST HI START DATE */
HI_START_EARL = MIN(OF HI_START1-HI_START&NUMHI.);

/* IF ALL HI AND SMI DATES ARE MISSING THEN MEDICARE
INDICATOR IS MISSING */
/* IF HI AND SMI DATES ARE PRESENT THEN SET MEDICARE
INDICATOR TO 0 */

DO K=1 TO 12;
  IF N(OF HI_START(*))=0 AND N(OF HI_TERM(*))=0 AND
    N(OF SMI_START(*))=0 AND N(OF SMI_TERM(*))=0
    THEN MEDR(K) = .;
  ELSE MEDR(K) = 0;

DO M = 1 TO &NUMSMI UNTIL(MEDR(K) = 1);
  IF (HI_START_EARL NE . AND
    HI_START_EARL <= INTNX('MONTH',MDY(K,1,&BEGYR.),0,'END')) OR
    (SMI_START(M) NE . AND SMI_TERM(M) NE . AND
    SMI_START(M) <= INTNX('MONTH',MDY(K,1,&BEGYR.),0,'END')
    <= SMI_TERM(M)) OR
    (SMI_START(M) NE . AND SMI_TERM(M) = . AND
    SMI_START(M) <= INTNX('MONTH',MDY(K,1,&BEGYR.),0,'END'))
  THEN MEDR(K) = 1;
END; /* END OF M LOOP */

/* SET TO MISSING IF BENE DIES */
IF INTNX('MONTH',MDY(K,1,&BEGYR.),0,'END') > DODBEST AND
  DODBEST NE . THEN MEDR(K) = .;
END; /* END OF K LOOP */

DROP HI_START: HI_TERM: SMI_STAR: SMI_TERM: M K;

/* FILL IN MISSING ZIP WITH DATA AVAILABLE FROM LAST
MONTH PRIOR TO S7 SUSPENSION */
ARRAY ZIP_DI(*) $ %DO MM=1 %TO &TOTMOS; ZIP_DI&&YY&MM %END;;
IF UPCASE(LAF&YR.01) = 'S7' THEN ZIP_DI&YR.01=ZIP_DI&PYR.12;
DO MM=2 TO &TOTMOS;
  IF UPCASE(LAF&CODE(MM)) = 'S7' THEN ZIP_DI(MM) = ZIP_DI(MM-1);
END;
DROP ZIP_DI&PYR.12;

/* CREATE MONTHLY STATE INDICATORS FOR PHASE CONSTRUCTION */
LENGTH %DO MM=1 %TO &TOTMOS; PST&&YY&MM %END; $2;
ARRAY ZIP_SSI(*) %DO MM=1 %TO &TOTMOS; ZIP_SSI&&YY&MM %END;;

```

## Appendix A.99

JCL/SAS Code: ANNCURR

```

ARRAY PST  (*) $ %DO MM=1 %TO &TOTMOS; PST&&YY&MM  %END;;

DO I = 1 TO DIM(ZIP_SSI);
  IF ZIP_SSI(I) NE ' ' AND ZIP_SSI(I) NE '00000'
    THEN PST(I) = ZIPSTATE(ZIP_SSI(I));
END;
/* IF NO VALUE FOR SSI USE DI */
DO I = 1 TO DIM(ZIP_DI);
  IF PST(I)=' ' AND ZIP_DI(I) NE ' ' AND ZIP_DI(I) NE '00000'
    THEN PST(I)=ZIPSTATE(ZIP_DI(I));
END;

/* CREATE SINGLE STATE VARIABLE PST_MR - NEW FOR DAF16 */
LENGTH PST_MR $2.;
DO I = DIM(PST) TO 1 BY -1 UNTIL (NOT MISSING(PST_MR));
  IF NOT MISSING(PST(I)) AND
    PST(I) NE "--"
    THEN PST_MR=PST(I);
END;
IF PST_MR="" THEN PST_MR=STATE_MR_DM;

/* CREATE FIPS_SSIymm AND FIPS_DIymm -
  IT WILL BE DONE IN ANOTHER PROCESSING NEXT YEAR */
ARRAY FIPS_SSI (*) $5 %DO MM=1 %TO &TOTMOS;
  FIPS_SSI&&YY&MM  %END;;
ARRAY FIPS_DI  (*) $5 %DO MM=1 %TO &TOTMOS;
  FIPS_DI&&YY&MM  %END;;
ARRAY FIPS_BEST(*) $5 %DO MM=1 %TO &TOTMOS;
  FIPS_BEST&&YY&MM %END;;

DO I = 1 TO DIM(ZIP_SSI);
  FIPS_SSI(I) = PUT(ZIP_SSI(I), $FIPS.);
  FIPS_DI(I)  = PUT(ZIP_DI(I) , $FIPS.);

  /* CREATE BEST FIPS */
  IF FIPS_SSI(I) NE ' ' THEN FIPS_BEST(I) = FIPS_SSI(I);
  ELSE FIPS_BEST(I) = FIPS_DI(I);
END;

/* CREATE MONTHLY INDICATORS FOR IRWE PASS AND
  SELF-EMPLOY - REMICS */
ARRAY IET4(*) $ %DO MM=1 %TO &TOTMOS; IET4&&YY&MM %END;;
ARRAY SIND(*) %DO MM=1 %TO &TOTMOS; SIND&&YY&MM %END;;

DO I = 1 TO DIM(IET4);

```

## Appendix A.99

JCL/SAS Code: ANNCURR

```

    IF IET4(I) = 'S' THEN SIND(I) = 1;
    ELSE SIND(I) = 0;
END;

/* TRIAL WORK PERIOD INDICATOR MOVED TO DCF */
ARRAY EARN  (*) %DO MM=1 %TO &TOTMOS; EARN&&YY&MM  %END;;
ARRAY IEA_NESE (*) %DO MM=1 %TO &TOTMOS; IEA_NESE&&YY&MM %END;;
ARRAY IEA_WAGE (*) %DO MM=1 %TO &TOTMOS; IEA_WAGE&&YY&MM %END;;
ARRAY EIND  (*) %DO MM=1 %TO &TOTMOS; EIND&&YY&MM  %END;;
ARRAY SGA   (*) %DO MM=1 %TO &TOTMOS; SSIERNSGA&&YY&MM %END;;

/*INCLUDE BLIND AND NONBLIND SGA VALUES FROM UTILITY LIBRARY*/
/*USE ENDYR TO REFERENCE TO SGA VALUES AS NEEDED*/
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(SGAB)";
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(SGANB)";

DO I = 1 TO DIM(EARN);
    EARN(I) = .;
    IF IEA_NESE(I) > .Z OR IEA_WAGE(I) > .Z THEN DO;
        EARN(I) = SUM(OF IEA_NESE(I),IEA_WAGE(I));
    END;

    IF EARN(I) > 0 THEN EIND(I) = 1;
    ELSE IF EARN(I) = 0 THEN EIND(I) = 0;
    ELSE EIND(I) = .;

/* SGA IS DIFFERENT FOR BLIND PEOPLE */
/* REVISED CODE UPDATED EVERY YEAR - THIS IS COMPUTED
   ONLY FOR THE NEW YEAR OF DATA COLLECTION. UPDATED
   EARNINGS CEILINGS ARE INCLUDED. EARLIER YEARS ALREADY
   HAVE THE VARIABLE AND IT IS MERGED ON FROM THE OLD FILE */
IF SUBSTR(MFT,1,1) = 'B' THEN DO; /* BLIND PEOPLE */
    IF EARN(I) >= SGAB(&ENDYR.) THEN SGA(I) = 1;
    ELSE IF 0 <= EARN(I) < SGAB(&ENDYR.) THEN SGA(I) = 0;
    ELSE SGA(I) = .;
END;
ELSE DO;
    IF EARN(I) >= SGANB(&ENDYR.) THEN SGA(I) = 1;
    ELSE IF 0 <= EARN(I) < SGANB(&ENDYR.) THEN SGA(I) = 0;
    ELSE SGA(I) = .;
END;
END; /* LOOP THROUGH EARNINGS */

/*CREATE AN INDICATOR FOR 1619(B) */
ARRAY MEDC (*) $ %DO MM=1 %TO &TOTMOS; MEDC&&YY&MM %END;;

```



Appendix A.99

JCL/SAS Code: ANNCURR

```

ARRAY MEDE (*) %DO MM=1 %TO &TOTMOS; MEDE&&YY&MM %END;;

DO I = 1 TO DIM(MEDC);
  IF MEDC(I) IN ('C','W','Y') THEN MEDE(I) = 1;
  ELSE IF MEDC(I) > '' THEN MEDE(I) = 0;
  ELSE MEDE(I) = .;
END;

DROP I MFT IET4;;
RUN;
%MEND;
%START;

TITLE "CONTENTS OF YEAR Y&SYSPARM";
PROC CONTENTS DATA=OUT1.Y&SYSPARM
  OUT=OUT2.ANN&SYSPARM._CON;
RUN;
TITLE;

PROC FREQ DATA=OUT1.Y&SYSPARM;
  TABLES MEDR: LAF: PSTA: MEDEX: / LIST MISSING;
  TABLES PST_MR PST_MR*STATE_MR_DM/LIST MISSING;
RUN;

PROC MEANS DATA=OUT1.Y&SYSPARM;
  VAR DUED:
    PAYD:
    PAYO:
    DUEO:
    FAMT:
    FPMT:
    SAMT:
    PAYS:
    DUES:
    SPMT:
    UINC:
    EICM:
    PROA:
    PROB:
    DIRPAY:
    MEDPREM:
  ;
RUN;

TITLE "PRINT OF YEAR Y&SYSPARM";

```

Appendix A.99

JCL/SAS Code: ANNCURR

```
PROC PRINT DATA=OUT1.Y&SYSPARM(FIRSTOBS=10000 OBS=10025);  
RUN;
```

```
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 ANNCURR IS COMPLETE
```

## Appendix A.100

JCL/SAS Code: Y94JCL

```
//#8047Y94 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y94JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 1994-DEC 1994
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* CREATED BY: MIRIAM CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="1994"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y1994E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y1994NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1994.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1994.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM94.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1994.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY94.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1994.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1994.SA.V1,
//      DISP=(OLD,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1994CON.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.100

JCL/SAS Code: Y94JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y94 IS COMPLETE.
```

## Appendix A.101

JCL/SAS Code: Y95JCL

```
//#8047Y95 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y95JCL)
//* PURPOSE: CREATE YEARLY FILES FOR 2020
//*      MONTHLY FIELDS FROM JAN 1995-DEC 1995
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: MIRIAM CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="1995"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y1995E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y1995NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1995.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1995.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM95.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1995.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY95.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1995.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1995.SA.V1,
//      DISP=(OLD,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1995CON.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.101

JCL/SAS Code: Y95JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y95 IS COMPLETE.
```

## Appendix A.102

JCL/SAS Code: Y96JCL

```
//#8047Y96 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y96JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 1996-DEC 1996
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: MIRIAM CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="1996"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y1996E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y1996NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1996.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1996.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM96.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1996.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY96.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1996.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1996.SA.V1,
//      DISP=(OLD,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1996CON.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.102

JCL/SAS Code: Y96JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y96 IS COMPLETE.
```



## Appendix A.103

JCL/SAS Code: Y97JCL

```
//#8047Y97 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y97JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 1997-DEC 1997
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="1997"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y1997E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y1997NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1997.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1997.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM97.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1997.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY97.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1997.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1997.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1997CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.103

JCL/SAS Code: Y97JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y97 IS COMPLETE.
```

## Appendix A.104

JCL/SAS Code: Y98JCL

```
//#8047Y98 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y98JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 1998-DEC 1998
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="1998"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y1998E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y1998NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1998.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1998.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM98.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1998.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY98.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1998.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1998.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1998CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.104

JCL/SAS Code: Y98JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y98 IS COMPLETE.
```

## Appendix A.105

JCL/SAS Code: Y99JCL

```
//#8047Y99 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y99JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF20
//*      MONTHLY FIELDS FROM JAN 1999-DEC 1999
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="1999"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y1999E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y1999NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR1999.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR1999.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM99.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN1999.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY99.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE1999.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1999.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y1999CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.105

JCL/SAS Code: Y99JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 99 IS COMPLETE.
```

## Appendix A.106

JCL/SAS Code: Y00JCL

```

//#8047Y00 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y00JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2000-DEC 2000
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2000"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2000E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2000NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2000.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2000.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM00.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2000.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY00.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2000.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2000.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2000CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *

```

Appendix A.106

JCL/SAS Code: Y00JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y00 IS COMPLETE.
```



## Appendix A.107

JCL/SAS Code: Y01JCL

```
//#8047Y01 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.SSRF.PRDLIB(Y01JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2001-DEC 2001
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2001"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2001E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2001NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2001.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2001.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM01.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2001.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY01.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2001.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2001.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2001CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.107

JCL/SAS Code: Y01JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y01 IS COMPLETE.
```

## Appendix A.108

JCL/SAS Code: Y02JCL

```
//#8047Y02 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.SSRF.PRDLIB(Y02JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2002-DEC 2002
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2002"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2002E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2002NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2002.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2002.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM02.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2002.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY02.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2002.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2002.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2002CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.108

JCL/SAS Code: Y02JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y02 IS COMPLETE.
```

## Appendix A.109

JCL/SAS Code: Y03JCL

```
//#8047Y03 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.SSRF.PRDLIB(Y03JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2003-DEC 2003
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2003"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2003E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2003NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2003.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2003.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM03.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2003.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY03.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2003.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2003.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2003CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.109

JCL/SAS Code: Y03JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y03 IS COMPLETE.
```

## Appendix A.110

JCL/SAS Code: Y04JCL

```
//#8047Y04 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y04JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2004-DEC 2004
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2004"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2004E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2004NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2004.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2004.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM04.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2004.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY04.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2004.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2004.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2004CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.110

JCL/SAS Code: Y04JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y04 IS COMPLETE.
```



## Appendix A.111

JCL/SAS Code: Y05JCL

```
//#8047Y05 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y05JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2005-DEC 2005
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: ALEX BRYCE
//* MPR PHONE 202 554-7556
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2005"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2005E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2005NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2005.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2005.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM05.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2005.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY05.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2005.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2005.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2005CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
```

Appendix A.111

JCL/SAS Code: Y05JCL

```
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 Y05 IS COMPLETE.
```

## Appendix A.112

JCL/SAS Code: Y06JCL

```
//#8047Y06 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y06JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2006-DEC 2006
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2006"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2006E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2006NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2006.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2006.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM06.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2006.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY06.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2006.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2006.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2006CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.112

JCL/SAS Code: Y06JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y06 IS COMPLETE.
```

## Appendix A.113

JCL/SAS Code: Y07JCL

```
//#8047Y07 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y07JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2007-DEC 2007
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2007"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2007E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2007NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2007.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2007.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM07.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2007.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY07.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2007.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2007.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2007CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.113

JCL/SAS Code: Y07JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y07 IS COMPLETE.
```

## Appendix A.114

JCL/SAS Code: Y08JCL

```
//#8047Y08 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y08JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2008-DEC 2008
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: MIRIAM CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2008"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2008E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2008NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2008.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2008.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM08.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2008.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY08.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2008.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2008.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2008CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.114

JCL/SAS Code: Y08JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y08 IS COMPLETE.
```



## Appendix A.115

JCL/SAS Code: Y09JCL

```
//#8047Y09 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y09JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2009-DEC 2009
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2009"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2009E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2009NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2009.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2009.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM09.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2009.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY09.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2009.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2009.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2009CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.115

JCL/SAS Code: Y09JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y09 IS COMPLETE.
```

## Appendix A.116

JCL/SAS Code: Y10JCL

```
//#8047Y10 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y10JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2010-DEC 2010
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2010"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2010E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2010NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2010.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2010.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM10.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2010.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY10.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2010.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2010.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2010CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.116

JCL/SAS Code: Y10JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y10 IS COMPLETE.
```

## Appendix A.117

JCL/SAS Code: Y11JCL

```
//#8047Y11 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y11JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2011-DEC 2011
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2011"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2011E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2011NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2011.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2011.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM11.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2011.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY11.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2011.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2011.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2011CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.117

JCL/SAS Code: Y11JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y11 IS COMPLETE.
```

## Appendix A.118

JCL/SAS Code: Y12JCL

```
//#8047Y12 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y12JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2012-DEC 2012
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: ALEX BRYCE
//* MPR PHONE 202 554-7556
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2012"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2012E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2012NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2012.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2012.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM12.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2012.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY12.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2012.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2012.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2012CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
```

Appendix A.118

JCL/SAS Code: Y12JCL

```
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 Y12 IS COMPLETE.
```



## Appendix A.119

JCL/SAS Code: Y13JCL

```
//#8047Y13 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y13JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2013-DEC 2013
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: ALEX BRYCE
//* MPR PHONE 202 554-7556
//* E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2013"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2013E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2013NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2013.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2013.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM13.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2013.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY13.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2013.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2013.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2013CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
```

Appendix A.119

JCL/SAS Code: Y13JCL

```
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 Y13 IS COMPLETE.
```

## Appendix A.120

JCL/SAS Code: Y14JCL

```
//#8047Y14 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y14JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2014-DEC 2014
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2014"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2014E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2014NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2014.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2014.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM14.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2014.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY14.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2014.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2014.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2014CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.120

JCL/SAS Code: Y14JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y14 IS COMPLETE.
```

## Appendix A.121

JCL/SAS Code: Y15JCL

```
//#8047Y15 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y15JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2015-DEC 2015
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2015"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2015E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2015NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2015.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2015.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM15.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2015.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY15.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2015.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2015.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2015CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.121

JCL/SAS Code: Y15JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y15 IS COMPLETE.
```

## Appendix A.122

JCL/SAS Code: Y16JCL

```
//#8047Y16 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y16JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2016-DEC 2016
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2016"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2016E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2016NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2016.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2016.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM16.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2016.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY16.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2016.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2016.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2016CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.122

JCL/SAS Code: Y16JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y16 IS COMPLETE.
```



## Appendix A.123

JCL/SAS Code: Y17JCL

```
//#8047Y17 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y17JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2017-DEC 2017
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2017"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2017E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2017NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2017.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2017.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM17.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2017.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY17.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2017.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2017.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2017CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.123

JCL/SAS Code: Y17JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y17 IS COMPLETE.
```

## Appendix A.124

JCL/SAS Code: Y18JCL

```
//#8047Y18 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y18JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2018-DEC 2018
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER MIRIAM CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2018"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2018E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2018NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2018.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2018.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM18.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2018.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY18.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2018.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2018.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2018CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.124

JCL/SAS Code: Y18JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y18 IS COMPLETE.
```

## Appendix A.125

JCL/SAS Code: Y19JCL

```
//#8047Y19 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF20.ANN.PRDLIB(Y19JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2018-DEC 2018
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: MIRIAM CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2019"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2019E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2019NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2019.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2019.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM19.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2019.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY19.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2019.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2019.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2019CON.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.125

JCL/SAS Code: Y19JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 Y19 IS COMPLETE.
```

## Appendix A.126

JCL/SAS Code: Y20JCL

```
//#8047Y20 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT: DAF 50214
//* DIRECTOR:
//* PROGRAM: OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(Y20JCL)
//* PURPOSE: CREATE YEARLY FILES FOR DAF21
//*      MONTHLY FIELDS FROM JAN 2018-DEC 2018
//*      DCF VARIABLES NOT INCLUDED BEFORE 2000
//* PROGRAMMER: CHAPPELKA
//* E-MAIL MCHAPPELKA@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="2020"'
//* INPUTS
//OLDE DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.Y2020E,DISP=SHR
//OLDNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF20P.Y2020NE,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2020.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBR2020.SA.V1,DISP=SHR
//T831 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDCOM20.SA.V1,DISP=SHR
//ERN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.ERN2020.SA.V1,DISP=SHR
//DXP DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DXPY20.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MED DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MIE2020.SA.V1,DISP=SHR
//* OUTPUTS
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2020.SA.V1,
//      DISP=(OLD,CATLG,DELETE),
//      UNIT=TSILO
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2020CON.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(ANNPREV),
//      DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.126

JCL/SAS Code: Y20JCL

```
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 20 IS COMPLETE.
```



Appendix A.127

JCL/SAS Code: ANNPREV

```

/*****/
/* FILENAME: ANNPREV          */
/* PURPOSE: TO CREATE YEARLY FILE FOR 1994-2018      */
/* CREATED: 07/13/06 BY MIRIAM LOEWENBERG          */
/* MODIFIED: 09/19/22 BY MIRIAM CHAPPELKA         */
/*****/
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS=MAX;

%LET YR = SUBSTR(&SYSPARM,3,2);

%LET BEGYR=&SYSPARM;
%LET ENDYR=&SYSPARM; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

/* CHECK AND CHANGE AS NEEDED NUMBER OF HI AND SMI OCCURRENCES */
/* FIND IT FROM THE MBR SOURCES */
%LET NUMHI = 10;
%LET NUMSMI = 20;

/* STEP TO ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA */
%MACRO START;
%INCLUDE "OPDR.TG.PRD.ETW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

DATA OUT1.Y&SYSPARM;
LENGTH MEDX&YR.01 - MEDX&YR.12 $8;
MERGE
  OLDE.Y&SYSPARM (DROP=DOBBEST DODBEST DOEI DXPRIBEST: FRADATE
    SEX SSDI9: SSDI0: SSDI1: SSDI_ONSET STW: BFW:
    SSI9: SSI0: SSI1: SSIELIG_FIRST TKT0: TKT1:
    MEDX:)
  OLDNE.Y&SYSPARM.NE (DROP=DOBBEST DODBEST DOEI DXPRIBEST: FRADATE
    SEX SSDI9: SSDI0: SSDI1: SSDI_ONSET STW: BFW:
    SSI9: SSI0: SSI1: SSIELIG_FIRST TKT0: TKT1:
    MEDX:)
  FIN.DMSTW (IN=INFIN KEEP=SSN DODBEST DOBBEST)
  SSR.SSR(KEEP=SSN SSI:
    %DO MM=1 %TO &TOTMOS;
    EICM&&YY&MM FAMT&&YY&MM
    FPMT&&YY&MM SPMT&&YY&MM
    SAMT&&YY&MM UINC&&YY&MM
    LIVF&&YY&MM MTST&&YY&MM
    PSTA&&YY&MM SCON&&YY&MM

```

## Appendix A.127

JCL/SAS Code: ANNPREV

```
    CURCOMP&&YY&MM STUDEM&&YY&MM
    %END;)
MBR.MBR (KEEP=SSN HI_START: HI_TERM: SMI_STAR: SMI_TERM: SSDI:
    %DO MM=1 %TO &TOTMOS;
    DUE&&YY&MM LAF&&YY&MM
    WIC&&YY&MM RFD&&YY&MM
    RFST&&YY&MM BPD&&YY&MM
    DPEN&&YY&MM DUEO&&YY&MM
    PAYD&&YY&MM PAYO&&YY&MM
    MBA&&YY&MM MBP&&YY&MM
    CDR&&YY&MM MEDPREM&&YY&MM
    DIRPAY&&YY&MM
    %END;)
T831.ALLCDCOM (KEEP=SSN
    %DO MM=1 %TO &TOTMOS;
    ALX&&YY&MM
    EDX&&YY&MM
    %END;)
ERN.ERN (KEEP=SSN
    %DO MM=1 %TO &TOTMOS;
    T2GRSAMT&&YY&MM T2NETAMT&&YY&MM
    T2SEHRS&&YY&MM T2SEVERIND&&YY&MM
    T2VERIND&&YY&MM DIERNLVL&&YY&MM
    TWPDATA&&YY&MM T2CDNAMT&&YY&MM
    T2EXPAMT&&YY&MM T2FRAUDVER&&YY&MM
    T2SBDYAMT&&YY&MM T2UBEAMT&&YY&MM
    T16BEXPAMT&&YY&MM T16EXLAMT&&YY&MM
    T16EXPAMT&&YY&MM T16GRSAMT&&YY&MM
    T16NETAMT&&YY&MM T16PASAMT&&YY&MM
    T16SEAMT&&YY&MM SSIERNLVL&&YY&MM
    ALLGAMT&&YY&MM
    %END;)
DXP.DXPROC(KEEP=SSN
    %DO MM=1 %TO &TOTMOS;
    DXPRIBEST&&YY&MM DXSECBEST&&YY&MM
    DXPRI_SSI&&YY&MM DXSEC_SSI&&YY&MM
    DXPRI_DI&&YY&MM DXSEC_DI&&YY&MM
    %END;)
MED.PATMED(KEEP=SSN
    %DO MM=1 %TO &TOTMOS; MEDEX&&YY&MM %END;
    RENAME=(%DO MM = 1 %TO &TOTMOS;
    MEDEX&&YY&MM = MEDX&&YY&MM %END;));

BY SSN;
```

## Appendix A.127

JCL/SAS Code: ANNPREV

```
/* KEEP RECORDS ONLY IN THE DEMO FILE */
IF INFIN ;

/*NEW IN DAF15: REMOVE INDIVIDUALS WHO DIED BEFORE 1996 */
IF YEAR(DODBEST) NE . AND YEAR(DODBEST) < 1996 THEN DELETE;

/* INPUT ARRAY */
ARRAY FPMT(*) %DO MM=1 %TO &TOTMOS; FPMT&&YY&MM %END;;
ARRAY FAMT(*) %DO MM=1 %TO &TOTMOS; FAMT&&YY&MM %END;;
ARRAY PSTA(*) %DO MM=1 %TO &TOTMOS; PSTA&&YY&MM %END;;
ARRAY MTST(*) %DO MM=1 %TO &TOTMOS; MTST&&YY&MM %END;;
ARRAY SPMT(*) %DO MM=1 %TO &TOTMOS; SPMT&&YY&MM %END;;
ARRAY SAMT(*) %DO MM=1 %TO &TOTMOS; SAMT&&YY&MM %END;;
ARRAY SCON(*) %DO MM=1 %TO &TOTMOS; SCON&&YY&MM %END;;
ARRAY DUED(*) %DO MM=1 %TO &TOTMOS; DUED&&YY&MM %END;;
ARRAY LAFCODE(*) %DO MM=1 %TO &TOTMOS; LAF&&YY&MM %END;;

/* OUTPUT ARRAY */
ARRAY PAYS(*) %DO MM=1 %TO &TOTMOS; PAYS&&YY&MM %END;;
ARRAY DUES(*) %DO MM=1 %TO &TOTMOS; DUES&&YY&MM %END;;

DO I = 1 TO DIM(FPMT);
  PAYS(I) = SUM(OF FPMT(I),SPMT(I));
  DUES(I) = SUM(OF FAMT(I),SAMT(I));
END;

/* CONCURRENT BENEFICIARY INDICATOR */
/* INDICATES BENEFICIARIES WHO RECEIVING A
PAYMENT (IN CURRENT PAY) IN BOTH SSI AND SSDI */
ARRAY CONC(*) %DO MM=1 %TO &TOTMOS; CONC&&YY&MM %END;;

DO I = 1 TO DIM(DUED);
  IF (DUES(I) > 0 AND PSTA(I) IN ('C01' 'M01' 'M02')) AND
    (DUED(I) > 0 AND SUBSTR(LAFCODE(I),1,1) IN ('C' 'E'))
    THEN CONC(I) = 1;
  ELSE CONC(I) = 0;
END;

/* CREATE MONTHLY INDICATORS FOR 1619A-1619B*/
ARRAY PROA(*) %DO MM=1 %TO &TOTMOS; PROA&&YY&MM %END;;
ARRAY PROB(*) %DO MM=1 %TO &TOTMOS; PROB&&YY&MM %END;;
ARRAY EICM(*) %DO MM=1 %TO &TOTMOS; EICM&&YY&MM %END;;

/* PROB */
/* CREATE ONLY FOR SSI */
```

## Appendix A.127

JCL/SAS Code: ANNPREV

```
DO I = 1 TO DIM(MTST);
  IF MTST(I) IN ('A','B','F') AND
    PSTA(I) IN ('N01', 'P01', 'E01') AND
    EICM(I) > 0
    THEN PROB(I) = 1;
  ELSE IF PSTA(I) NE ' ' THEN PROB(I) = 0;
END;

/* PROA */
/* THIS SHOULD YIELD NO CASES WHERE PROB=1 AND PROA=1 */
DO I = 1 TO DIM(SCON);
  IF SCON(I) IN ('D','E','F','G') AND
    PSTA(I) IN ('C01', 'M01','M02') AND
    DUES(I) > 0 AND
    EICM(I) > 0
    THEN PROA(I) = 1;
  ELSE IF PSTA(I) NE ' ' THEN PROA(I) = 0;
END;

/* CREATE A MEDICARE INDICATOR */
/* INPUT ARRAYS */
ARRAY HI_START(*) HI_START1-HI_START&NUMHI.;
ARRAY HI_TERM(*) HI_TERM1-HI_TERM&NUMHI.;
ARRAY SMI_START(*) SMI_STAR1-SMI_STAR&NUMSMI.;
ARRAY SMI_TERM(*) SMI_TERM1-SMI_TERM&NUMSMI.;

/* OUTPUT ARRAYS */
ARRAY MEDR(*) MEDR&YR.01-MEDR&YR.12;

/* FIND THE EARLIEST HI START DATE */
HI_START_EARL = MIN(OF HI_START1-HI_START&NUMHI.);

/* IF ALL HI AND SMI DATES ARE MISSING THEN
  MEDICARE INDICATOR IS MISSING */
/* IF HI AND SMI DATES ARE PRESENT THEN SET
  MEDICARE INDICATOR TO 0 */
DO K =1 TO 12;
  IF N(OF HI_START(*))=0 AND N(OF HI_TERM(*))=0 AND
    N(OF SMI_START(*))=0 AND N(OF SMI_TERM(*))=0
    THEN MEDR(K) = .;
  ELSE MEDR(K) = 0;

  DO M = 1 TO &NUMSMI UNTIL(MEDR(K) = 1);
    IF (HI_START_EARL NE . AND
      HI_START_EARL <=
```

## Appendix A.127

JCL/SAS Code: ANNPREV

```
        INTNX('MONTH',MDY(K,1,&BEGYR.),0,'END')) OR
        (SMI_START(M) NE . AND SMI_TERM(M) NE . AND
        SMI_START(M) <= INTNX('MONTH',MDY(K,1,&BEGYR.),0,'END')
        <= SMI_TERM(M)) OR
        (SMI_START(M) NE . AND SMI_TERM(M) = . AND
        SMI_START(M) <= INTNX('MONTH',MDY(K,1,&BEGYR.),0,'END'))
        THEN MEDR(K) = 1;
END;

/* SET TO MISSING IF BENE DIES */
IF INTNX('MONTH',MDY(K,1,&BEGYR.),0,'END') > DODBEST AND
   DODBEST NE . THEN MEDR(K) = .;
END;

DROP HI_START: HI_TERM: SMI_STAR: SMI_TERM:
   M K I SSDI_ONSET_LAST;
RUN;
%MEND;
%START;

TITLE "CONTENTS OF YEAR &SYSPARM";
PROC CONTENTS DATA=OUT1.Y&SYSPARM
   OUT=OUT2.ANN&SYSPARM._CON;
RUN;

TITLE "CHECK FREQS OF YEAR &SYSPARM";
PROC FREQ DATA=OUT1.Y&SYSPARM;
   TABLES MEDR: LAF: PSTA: MEDX: / LIST MISSING;
RUN;

TITLE "CHECK STATISTICAL FOR YEAR &SYSPARM";
PROC MEANS DATA=OUT1.Y&SYSPARM;
   VAR DUED: PAYD: PAYO: DUEO: FAMT: FPMT: SAMT:
   PAYS: DUES: SPMT: UINC: EICM: PROA: PROB:
   DIRPAY: MEDPREM;;
RUN;

TITLE "PRINT 25 OBS FOR YEAR &SYSPARM";
PROC PRINT DATA=OUT1.Y&SYSPARM(FIRSTOBS=10000 OBS=10025);
RUN;
TITLE;
```

## Appendix A.128

JCL/SAS Code: FIN1994

```
//#8047F94 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.ANN.PRDLIB(FIN1994)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//      DISP=(SHR,PASS,KEEP)
//      PEND
```

Appendix A.128

JCL/SAS Code: FIN1994

```
//*
//RUN EXEC STACK,YYYY=1994,YY=94
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: FIN1994 COMPLETE
```

## Appendix A.129

JCL/SAS Code: FIN1995

```
//#8047F95 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN1995)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="'&YYYY'"
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//      DISP=(SHR,PASS,KEEP)
//      PEND
```



Appendix A.129

JCL/SAS Code: FIN1995

```
//*  
//RUN EXEC STACK,YYYY=1995,YY=95  
//*  
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: FIN1995 COMPLETE
```

## Appendix A.130

JCL/SAS Code: FIN1996

```
//#8047F94 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN1996)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```

Appendix A.130

JCL/SAS Code: FIN1996

```
//*
//RUN EXEC STACK,YYYY=1996,YY=96
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: FIN1996 COMPLETE
```

## Appendix A.131

JCL/SAS Code: FIN1997

```
//#8047F97 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN1997)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//      DISP=(SHR,PASS,KEEP)
//      PEND
```

Appendix A.131

JCL/SAS Code: FIN1997

```
//*  
//RUN EXEC STACK,YYYY=1997,YY=97  
//*  
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: FIN1997 COMPLETE
```

## Appendix A.132

JCL/SAS Code: FIN1998

```
//#8047F98 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN1998)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//      DISP=(SHR,PASS,KEEP)
//      PEND
```

Appendix A.132

JCL/SAS Code: FIN1998

```
//*  
//RUN EXEC STACK,YYYY=1998,YY=98  
//*  
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: FIN1998 COMPLETE
```

## Appendix A.133

JCL/SAS Code: FIN1999

```
//#8047F98 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN1999)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```



Appendix A.133

JCL/SAS Code: FIN1999

```
//*  
//RUN EXEC STACK,YYYY=1999,YY=99  
//*  
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: FIN1999 COMPLETE
```

## Appendix A.134

JCL/SAS Code: FIN2000

```
//#8047F98 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8407.DAF21.ANN.PRDLIB(FIN2000)
//* DESCRIPTION: FINALIZE ANNUALS
//*          SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```

Appendix A.134

JCL/SAS Code: FIN2000

```
//*  
//RUN EXEC STACK,YYYY=2000,YY=00  
//*  
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: FIN2000 COMPLETE
```

## Appendix A.135

JCL/SAS Code: FIN2001

```
//#8047F01 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN2001)
//* DESCRIPTION: FINALIZE ANNUALS
//*          SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```

Appendix A.135

JCL/SAS Code: FIN2001

```
//*  
//RUN EXEC STACK,YYYY=2001,YY=01  
//*  
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: FIN2001 COMPLETE
```

## Appendix A.136

JCL/SAS Code: FIN2002

```
//#8047F02 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN2002)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="'&YYYY'"
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//      DISP=(SHR,PASS,KEEP)
//      PEND
```

Appendix A.136

JCL/SAS Code: FIN2002

```
//*
//RUN EXEC STACK,YYYY=2002,YY=02
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: FIN2002 COMPLETE
```

## Appendix A.137

JCL/SAS Code: FIN2003

```
//#8047F02 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN2003)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```



Appendix A.137

JCL/SAS Code: FIN2003

```
//*
//RUN EXEC STACK,YYYY=2003,YY=03
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: FIN2003 COMPLETE
```

## Appendix A.138

JCL/SAS Code: FIN2004

```
//#8047F04 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN2004)
//* DESCRIPTION: FINALIZE ANNUALS
//*          SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//      DISP=(SHR,PASS,KEEP)
//      PEND
```

Appendix A.138

JCL/SAS Code: FIN2004

```
//*  
//RUN EXEC STACK,YYYY=2004,YY=04  
//*  
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: FIN2004 COMPLETE
```

## Appendix A.139

JCL/SAS Code: FIN2005

```
//#8047F02 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN2005)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY'"
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```

Appendix A.139

JCL/SAS Code: FIN2005

```
//*
//RUN EXEC STACK,YYYY=2005,YY=05
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: FIN2005 COMPLETE
```

## Appendix A.140

JCL/SAS Code: FIN2006

```
//#8047F06 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN2006)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```

Appendix A.140

JCL/SAS Code: FIN2006

```
//*  
//RUN EXEC STACK,YYYY=2006,YY=06  
//*  
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: FIN2006 COMPLETE
```

## Appendix A.141

JCL/SAS Code: FIN2007

```
//#8047F06 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FIN2007)
//* DESCRIPTION: FINALIZE ANNUALS
//*          SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```



Appendix A.141

JCL/SAS Code: FIN2007

```
//*
//RUN EXEC STACK,YYYY=2007,YY=07
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: FIN2007 COMPLETE
```

## Appendix A.142

JCL/SAS Code: FIN2008

```
//#8047F08 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ANN.PRDLIB(FIN2008)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```

Appendix A.142

JCL/SAS Code: FIN2008

```
//*
//RUN EXEC STACK,YYYY=2008,YY=08
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: FIN2008 COMPLETE
```

## Appendix A.143

JCL/SAS Code: FIN2009

```
//#8047F06 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ANN.PRDLIB(FIN2009)
//* DESCRIPTION: FINALIZE ANNUALS
//*          SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```

Appendix A.143

JCL/SAS Code: FIN2009

```
//*
//RUN EXEC STACK,YYYY=2009,YY=09
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: FIN2009 COMPLETE
```

## Appendix A.144

JCL/SAS Code: FIN2010

```
//#8047F10 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ANN.PRDLIB(FIN2010)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//      DISP=(SHR,PASS,KEEP)
//      PEND
```

Appendix A.144

JCL/SAS Code: FIN2010

```
//*
//RUN EXEC STACK,YYYY=2010,YY=10
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: FIN2010 COMPLETE
```

## Appendix A.145

JCL/SAS Code: FIN2011

```
//#8047F10 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ANN.PRDLIB(FIN2011)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//      DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//      DISP=(SHR,PASS,KEEP)
//      PEND
```



Appendix A.145

JCL/SAS Code: FIN2011

```
//*  
//RUN EXEC STACK,YYYY=2011,YY=11  
//*  
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: FIN2011 COMPLETE
```

## Appendix A.146

JCL/SAS Code: FIN2012

```
//#8047F12 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ANN.PRDLIB(FIN2012)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```

Appendix A.146

JCL/SAS Code: FIN2012

```
//*  
//RUN EXEC STACK,YYYY=2012,YY=12  
//*  
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: FIN2012 COMPLETE
```

## Appendix A.147

JCL/SAS Code: FIN2013

```
//#8047F13 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ANN.PRDLIB(FIN2013)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```

Appendix A.147

JCL/SAS Code: FIN2013

```
//*
//RUN EXEC STACK,YYYY=2013,YY=13
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: FIN2013 COMPLETE
```

## Appendix A.148

JCL/SAS Code: FIN2014

```
//#8047F14 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ANN.PRDLIB(FIN2014)
//* DESCRIPTION: FINALIZE ANNUALS
//*          SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```

Appendix A.148

JCL/SAS Code: FIN2014

```
//*
//RUN EXEC STACK,YYYY=2014,YY=14
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: FIN2014 COMPLETE
```

## Appendix A.149

JCL/SAS Code: FIN2015

```
//#8047F14 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ANN.PRDLIB(FIN2015)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(FINANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
```



Appendix A.149

JCL/SAS Code: FIN2015

```
//*  
//RUN EXEC STACK,YYYY=2015,YY=15  
//*  
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: FIN2015 COMPLETE
```

## Appendix A.150

JCL/SAS Code: FIN2016

```
// $4671F16 JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ANN.PRDLIB(FIN2016)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:      11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY'"
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//ZIP DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP16.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21.ANN.PRDLIB(FINANN21),
//  DISP=(SHR,PASS,KEEP)
```

Appendix A.150

JCL/SAS Code: FIN2016

```
// PEND
//*
//RUN EXEC STACK,YYYY=2016,YY=16
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM>
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM>
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM>
TO: <DPHELPS@MATHEMATICA-MPR.COM>
SUBJECT: FIN2016 COMPLETE
```

## Appendix A.151

JCL/SAS Code: FIN2017

```
// $4671F17 JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.$4671.DAF20.ANN.PRDLIB(FIN2017)
//* DESCRIPTION: FINALIZE ANNUALS
//*      SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:      11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//ZIP DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP17.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21.ANN.PRDLIB(FINANN21),
//  DISP=(SHR,PASS,KEEP)
```

Appendix A.151

JCL/SAS Code: FIN2017

```
// PEND
//*
//RUN EXEC STACK,YYYY=2017,YY=17
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM>
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM>
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM>
TO: <DPHELPS@MATHEMATICA-MPR.COM>
SUBJECT: FIN2017 COMPLETE
```

## Appendix A.152

JCL/SAS Code: FIN2018

```
//$4671F18 JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.$4671.DAF20.ANN.PRDLIB(FIN2018)
//* DESCRIPTION: FINALIZE ANNUALS
//*          SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//ZIP DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP18.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21.ANN.PRDLIB(FINANN21),
//  DISP=(SHR,PASS,KEEP)
```

Appendix A.152

JCL/SAS Code: FIN2018

```
// PEND
//*
//RUN EXEC STACK,YYYY=2018,YY=18
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM>
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM>
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM>
TO: <DPHELPS@MATHEMATICA-MPR.COM>
SUBJECT: FIN2018 COMPLETE
```

## Appendix A.153

JCL/SAS Code: FIN2019

```
// $4671F19 JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.$4671.DAF20.ANN.PRDLIB(FIN2019)
//* DESCRIPTION: FINALIZE ANNUALS
//*      SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//ZIP DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP19.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.E.SA.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.NE.SA.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.C10AN&YY.E.SA.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21.ANN.PRDLIB(FINANN21),
//      DISP=(SHR,PASS,KEEP)
```



Appendix A.153

JCL/SAS Code: FIN2019

```
// PEND
//*
//RUN EXEC STACK,YYYY=2019,YY=19
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM>
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM>
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM>
TO: <DPHELPS@MATHEMATICA-MPR.COM>
SUBJECT: FIN2019 COMPLETE
```

## Appendix A.154

JCL/SAS Code: FIN2020

```
//$4671F20 JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ANN.PRDLIB(FIN2020)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:     11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//      WORK='100000,50000',
//      PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//CER DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.COMBCERE.SA.V1,DISP=SHR
//ZIP DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP20.SA.V1,DISP=SHR
//FIND DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.MASTFIND.SA.V1,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.SSR&YYYY..SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.E.SA.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.NE.SA.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.C10AN&YY.E.SA.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
```

Appendix A.154

JCL/SAS Code: FIN2020

```
//    SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21.ANN.PRDLIB(FINANN16),
//    DISP=(SHR,PASS,KEEP)
//    PEND
//*
//RUN  EXEC STACK,YYYY=2020,YY=20
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2  DD SYSOUT=(B,SMTP)
//SYSUT1  DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM>
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM>
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM>
TO:  <DPHELPS@MATHEMATICA-MPR.COM>
SUBJECT: FIN2020 COMPLETE
```

## Appendix A.155

JCL/SAS Code: FIN2021

```
// $4671F21 JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.ANN.PRDLIB(FIN2021)
//* DESCRIPTION: FINALIZE ANNUALS
//*           SPLIT ANNUALS - ENROLLED AND NON-ENROLLED ANNUALS
//* DATE:      11/29/2021 ALEX BRYCE
//*****
//*
//STACK PROC
//JS001 EXEC SAS9,
//  WORK='100000,50000',
//  PARM='SYSPARM="'&YYYY"'
//*
//INANN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y&YYYY..SA.V1,DISP=SHR
//STWBFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW&YYYY..SA.V1,DISP=SHR
//COMM DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//ZIP DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP21.SA.V1,DISP=SHR
//OUTE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.E.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTNE DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y&YYYY.NE,
//  DISP=(MOD,CATLG,DELETE),
//  UNIT=TSILO
//OUTCNE DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.CAN&YY.NE.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* 10% DM FILES
//E10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.E,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//NE10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y&YYYY.NE,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUTCE10 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.C10AN&YY.E.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21.ANN.PRDLIB(FINANN21),
//  DISP=(SHR,PASS,KEEP)
```

Appendix A.155

JCL/SAS Code: FIN2021

```
// PEND
//*
//RUN EXEC STACK,YYYY=2021,YY=21
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM>
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM>
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM>
TO: <DPHELPS@MATHEMATICA-MPR.COM>
SUBJECT: FIN2021 COMPLETE
```

## Appendix A.156

JCL/SAS Code: FINANN

```

/*****/
/* PROJECT: DAF 50214 */
/* DIRECTOR: */
/* PROGRAM: OPDR.TG.PRD.ETTW.#4941.DAF16.ANN.PRDLIB(FINANN) */
/* PURPOSE: SPLIT ANNUALS FILES INTO ENROLLED/NON-ENROLLED */
/* ANY OBS MEETING ALL FOLLOWING CRITERIA WILL */
/* BE SENT TO NON-ENROLLED */
/* FRADATE<=1/1/&YYYY : FRA ALL YEAR */
/* */
/* LAFYMM = ' ' */
/* PSTAYMM = ' ' */
/* */
/* STWSSIYMM = . */
/* STWDIYMM = . */
/* STWCMYMM = . */
/* */
/* PAYDYMM <= 0 */
/* PAYSYMM <= 0 */
/* BFWSSIYMM <= 0 */
/* BFWDIYMM <= 0 */
/* BFWCMYMM <= 0 */
/* */
/*!! UPDATE YY TO CURRENT DAF YEAR FOR SSI AND SSDI RANGE BELOW !*/
/* */
/*****/
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY MPRINT OBS=MAX;

%LET YYYY=&SYSPARM;
%LET YR=%SUBSTR(&SYSPARM,3);

%PUT ***** FINALIZING ANNUAL &YYYY (&YR);

%MACRO SPLITANN;
DATA OUTE.Y&YYYY. ( DROP=V IN_ANN I)
OUTNE.Y&YYYY.NE( DROP=V IN_ANN I)
;
MERGE INANN.Y&YYYY.(DROP= DOBBEST DDBEST
SSI94- SSI99
SSI00- SSI21
SSDI94-SSDI99
SSDI00-SSDI21
SSDI_ONSET
SSI_MIN_ELG SSIELIG_ADULT
RENAME=(%IF &YR. = 21 %THEN %DO);

```

## Appendix A.156

JCL/SAS Code: FINANN

```
%DO M=1 %TO 12;
%LET MM=%SYSFUNC(PUTN(&M,Z2.));
MEDEX&YR.&MM = MEDX&YR.&MM
%END;
%END;

/*%IF &YR.=96 %THEN %DO;
%DO M=3 %TO 12;
%LET MM=%SYSFUNC(PUTN(&M,Z2.));
PST&YR.&MM = STATE&YR.&MM
EARN&YR.&MM = EARN_SSI&YR.&MM
EIND&YR.&MM = EIND_SSI&YR.&MM
%END;
%END; */

%IF &YR.=21 %THEN %DO;
%DO M=1 %TO 12;
%LET MM=%SYSFUNC(PUTN(&M,Z2.));
PST&YR.&MM = STATE&YR.&MM
EARN&YR.&MM = EARN_SSI&YR.&MM
EIND&YR.&MM = EIND_SSI&YR.&MM
%END;
%END;

/* ADDED IN DAF 21: RENAMING THE ZIP_
VARIABLES FROM CER AND DBAD */
%IF &YR.=96 %THEN %DO;
%DO M=6 %TO 12;
%LET MM=%SYSFUNC(PUTN(&M,Z2.));
ZIP_BEST&YR.&MM=ZIP_BEST_BASE&YR.&MM
ZIP_DI&YR.&MM = ZIP_DI_BASE&YR.&MM
%END; /* END MONTH BLOCK */
%END; /* END YEAR = 1996 BLOCK */
%IF &YR. = 97 %THEN %DO;
%DO M = 1 %TO 12;
%LET MM=%SYSFUNC(PUTN(&M,Z2.));
ZIP_BEST&YR.&MM=ZIP_BEST_BASE&YR.&MM
ZIP_DI&YR.&MM = ZIP_DI_BASE&YR.&MM
%IF &M >10 %THEN %DO;
ZIP_SSI&YR.&MM = ZIP_SSI_BASE&YR.&MM
%END;
%END;
%END;
%ELSE %IF NOT(&YR.=94)
AND NOT(&YR.=95) AND NOT(&YR.=96)
```

## Appendix A.156

JCL/SAS Code: FINANN

```

AND NOT (&YR.=97)
%THEN %DO;
%DO M=1 %TO 12;
  %LET MM=%SYSFUNC(PUTN(&M,Z2.));
  ZIP_BEST&YR.&MM=ZIP_BEST_BASE&YR.&MM
  ZIP_DI&YR.&MM = ZIP_DI_BASE&YR.&MM
  ZIP_SSI&YR.&MM = ZIP_SSI_BASE&YR.&MM
%END; /* END MONTH BLOCK */
%END;))/* END YEAR > 1996 BLOCK */

```

```

COMM.COMVARS(IN=KEEPCOM)

```

```

STWBFW.BFW (KEEP=SSN

```

```

  BMF&YR.: CFBMF&YR.: DENOM&YR.:
  FCI&YR.: NECALC&YR.: NONEARN&YR.:
  STWDI&YR.: STWSSI&YR.: STWCM&YR.:
  BFWDI&YR.: BFWSSI&YR.: BFWCM&YR.:
  BFWDI_PRIM&YR.:
  BFWDI_DEPEND&YR.:
  RENAME=(%DO M=1 %TO 12;
    %LET MM=%SYSFUNC(PUTN(&M,Z2.));
    BFWDI_PRIM&YR.&MM =BFWDI_PRI&YR.&MM
    BFWDI_DEPEND&YR.&MM=BFWDI_DEP&YR.&MM
  %END;))

```

```

;

```

```

BY SSN;

```

```

IF KEEPCOM;

```

```

/*MASK DAA VARIABLES*/

```

```

ARRAY PSTA (12) PSTA&YR.01 - PSTA&YR.12;

```

```

ARRAY CDR (12) CDR&YR.01 - CDR&YR.12;

```

```

ARRAY WIC (12) WIC&YR.01 - WIC&YR.12;

```

```

DO I = 1 TO 12;

```

```

  IF PSTA(I) IN('N10', 'N11', 'N12', 'N17', 'N18', 'N20', 'N36',
    'N37', 'N38', 'N39', 'N47', 'N48', 'N49', 'N50')

```

```

  THEN PSTA(I) = 'W01';

```

```

  IF CDR(I) IN('F', 'N', 'W') THEN CDR(I) = 'X';

```

```

  IF WIC(I) IN('A', 'B', 'C', 'N', 'P') THEN WIC(I) = 'S';

```

```

END;

```

```

/* USING ARRAY FOR VARIABLES BEING COMPARED TO '' */

```

```

ARRAY CHARMISS (*) LAF&YR.01 - LAF&YR.12

```

```

  PSTA&YR.01 - PSTA&YR.12;

```



## Appendix A.156

JCL/SAS Code: FINANN

```
/* USING ARRAY FOR VARIABLES BEING COMPARED TO . */
ARRAY NUMMISS (*) STWSSI&YR.01 - STWSSI&YR.12
      STWDI&YR.01 - STWDI&YR.12
      STWCM&YR.01 - STWCM&YR.12;

/* USING ARRAY FOR VARIABLES BEING COMPATED TO 0 (>=0) */
ARRAY LEQZERO (*) PAYD&YR.01 - PAYD&YR.12
      PAYS&YR.01 - PAYS&YR.12
      BFWSSI&YR.01 - BFWSSI&YR.12
      BFWDI&YR.01 - BFWDI&YR.12
      BFWCM&YR.01 - BFWCM&YR.12;

IN_ANN=0; *FLAG TO SPLIT DATASET BELOW;

/*SEARCHING THROUGH FIRST SET OF VARIABLES*/
DO V=1 TO DIM(CHARMISS) WHILE(IN_ANN=0);
  IF CHARMISS(V) NE ' ' THEN IN_ANN=1;
END;

/*SEARCHING THROUGH SECOND SET OF VARIABLES IF CRITERIA ABOVE NOT MET*/
IF IN_ANN=0 THEN
DO V=1 TO DIM(NUMMISS) WHILE(IN_ANN=0);
  IF NUMMISS(V) NE . THEN IN_ANN=1;
END;

/*SEARCHING THROUGH THIRD SET OF VARIABLES IF CRITERIA ABOVE NOT MET*/
IF IN_ANN=0 THEN
DO V=1 TO DIM(LEQZERO) WHILE(IN_ANN=0);
  IF LEQZERO(V) > 0 THEN IN_ANN=1;
END;

/*FRA ALL YEAR TRUMPS ALL PREVIOUS CRITERIA*/
IF FRADATE<="01JAN&YYYY."D AND FRADATE NE .
THEN IN_ANN=0;

%INCLUDE "OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(LABELS)";

/*DROP VARIABLES THAT SHOULD NOT BE ON FINAL DAF19 ANNUALS*/
%IF &YR=21 %THEN %DO;
  DROP MM SSDI_ONSET_LAST PST_MR PST_MR_DM FL_DEATH FL_UNBORN
  SGAB: SGANB: ;
%END;

* OUTPUT ENROLLED ANNUALS;
```

## Appendix A.156

JCL/SAS Code: FINANN

```
IF IN_ANN=1 THEN OUTPUT OUTE.Y&YYYY;

*OUTPUT NONENROLLED ANNUALS;
ELSE OUTPUT OUTNE.Y&YYYY.NE;

RUN;

%MEND SPLITANN;
%SPLITANN;

PROC FORMAT;
VALUE $WICDAA 'A' = 'A'
              'B' = 'B'
              'C' = 'C'
              'N' = 'N'
              'P' = 'P'
              'S' = 'S'
              OTHER = 'NOT DAA';
VALUE $CDRDAA 'F' = 'F'
              'N' = 'N'
              'W' = 'W'
              'X' = 'X'
              OTHER = 'NOT DAA';
VALUE $PSTADAA 'N10' = 'N10'
               'N11' = 'N11'
               'N12' = 'N12'
               'N17' = 'N17'
               'N18' = 'N18'
               'N20' = 'N20'
               'N36' = 'N36'
               'N37' = 'N37'
               'N38' = 'N38'
               'N39' = 'N39'
               'N47' = 'N47'
               'N48' = 'N48'
               'N49' = 'N49'
               'N50' = 'N50'
               'W01' = 'W01'
               OTHER = 'NOT DAA';
RUN;

TITLE "CHECK DAA MASKING IN ENROLLED FILE";
PROC FREQ DATA = OUTE.Y&YYYY.;
TABLES WIC: CDR: PSTA: / LIST MISSING;
FORMAT WIC: $WICDAA. CDR: $CDRDAA. PSTA: $PSTADAA.;
```

Appendix A.156

JCL/SAS Code: FINANN

RUN;

```
TITLE "CHECK DAA MASKING IN NON-ENROLLED FILE";
PROC FREQ DATA = OUTNE.Y&YYYY.NE;
TABLES WIC: CDR: PSTA: / LIST MISSING;
FORMAT WIC: $WICDAA. CDR: $CDRDAA. PSTA: $PSTADAA.;
RUN;
```

```
TITLE "FULL ENROLLED ANNUALS";
PROC CONTENTS DATA = OUTE.Y&YYYY.
    OUT = OUTCE.ANN&YR.CON; RUN;
TITLE "FULL NONENROLLED ANNUALS";
PROC CONTENTS DATA = OUTNE.Y&YYYY.NE
    OUT = OUTCNE.ANN&YR.CON; RUN;
```

```
/* CREATE 10%DM FOR ANNUAL FILES */
DATA E10.Y&YYYY.;
SET OUTE.Y&YYYY.;
IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9'))
    AND (SUBSTR(SSN,9,1) IN ('0', '5'));
RUN;
```

```
DATA NE10.Y&YYYY.NE;
SET OUTNE.Y&YYYY.NE;
IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9'))
    AND (SUBSTR(SSN,9,1) IN ('0', '5'));
RUN;
```

```
/*RUNNING AUTOFOF MACRO*/
PROC CONTENTS DATA=E10.Y&YYYY. OUT=OUTCE10.ANN&YR.CON; RUN;
```

Appendix A.157

JCL/SAS Code: FINANN16

```

/*****/
/* PROJECT: DAF 50214 */
/* DIRECTOR: */
/* PROGRAM: OPDR.TG.PRD.ETTW.#4941.DAF16.ANN.PRDLIB(FINANN) */
/* PURPOSE: SPLIT ANNUALS FILES INTO ENROLLED/NON-ENROLLED */
/* ANY OBS MEETING ALL FOLLOWING CRITERIA WILL */
/* BE SENT TO NON-ENROLLED */
/* FRADATE<=1/1/&YYYY : FRA ALL YEAR */
/* */
/* LAFYMM = ' ' */
/* PSTAYMM = ' ' */
/* */
/* STWSSIYMM = . */
/* STWDIYMM = . */
/* STWCMYMM = . */
/* */
/* PAYDYMM <= 0 */
/* PAYSYMM <= 0 */
/* BFWSSIYMM <= 0 */
/* BFWDIYMM <= 0 */
/* BFWCMYMM <= 0 */
/* */
/*!! UPDATE YY TO CURRENT DAF YEAR FOR SSI AND SSDI RANGE BELOW !*/
/* */
/*****/
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY MPRINT OBS=MAX;

%LET YYYY=&SYSPARM;
%LET YR=%SUBSTR(&SYSPARM,3);

%LET BEGYR=&SYSPARM;
%LET ENDYR=&SYSPARM; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

%PUT ***** FINALIZING ANNUAL &YYYY (&YR);

%MACRO SPLITANN;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

DATA OUTE.Y&YYYY. ( DROP=V IN_ANN I)
  OUTNE.Y&YYYY.NE( DROP=V IN_ANN I)
  ;
MERGE INANN.Y&YYYY.(DROP= DOBBEST DDBEST

```

## Appendix A.157

JCL/SAS Code: FINANN16

```
SSI94- SSI99
SSI00- SSI21
SSDI94-SSDI99
SSDI00-SSDI21
SSDI_ONSET
SSI_MIN_ELG SSIELIG_ADULT
/* ZIP-FIPS PATCH WILL BE REPLACING
   THE STATEYMM VARIABLES WHICH ON THE 2021
   INTERMEDIATE IS CALLED PSTYMM WHILE IT IS
   STATEYMM ON ALL OTHER YEARS */
%IF &YR.=21 %THEN PST2101-PST2112;
%ELSE STATE&YR.01-STATE&YR.12;
FIPS_BEST:
FIPS_DI:
FIPS_SSI:
ZIP_BEST:
ZIP_DI:
ZIP_SSI:

/* THE CER SOURCED VARIABLES ARE BEING
   REPLACED BY THE HISTORICAL CER/DBAD */
IEA_BLIND:
IEA_PASS:
IEA_LOSS:
IEA_NESE:
IEA_IRWE:
IEA_WAGE:
IEA_SEIE:
MEDC:
IUA_SSDI:
IUA_WC:
IUA_TANF:

/* THE VARS CREATED WITH CER VARS ARE BEING
   RECREATED USING THE HISTORICAL CER */
EARN_SSI&YR.01-EARN_SSI&YR.12
EIND_SSI&YR.01-EIND_SSI&YR.12
SSIERNPGA&YR.01-SSIERNPGA&YR.12
MEDE&YR.01-MEDE&YR.12

RENAME=(%IF &YR.=21 %THEN %DO;
        %DO M=1 %TO 12;
        %LET MM=%SYSFUNC(PUTN(&M,Z2.));
        MEDEX&YR.&MM = MEDX&YR.&MM
        %END;
```

## Appendix A.157

JCL/SAS Code: FINANN16

```
                                %END;))

COMM.COMVARS(IN=KEEPCOM)
STWBFW.BFW (KEEP=SSN
    BMF&YR.: CFBMF&YR.: DENOM&YR.:
    FCI&YR.: NECALC&YR.: NONEARN&YR.:
    STWDI&YR.: STWSSI&YR.: STWCM&YR.:
    BFWDI&YR.: BFWSSI&YR.: BFWCM&YR.:
    BFWDI_PRIM&YR.:
    BFWDI_DEPEND&YR.:
    RENAME=(%DO M=1 %TO 12;
        %LET MM=%SYSFUNC(PUTN(&M,Z2.));
        BFWDI_PRIM&YR.&MM =BFWDI_PRI&YR.&MM
        BFWDI_DEPEND&YR.&MM=BFWDI_DEP&YR.&MM
    %END;))
/* HISTORICAL CER */
CER.CER(IN=CER KEEP=IEA_BLIND&YR.:
    IEA_PASS&YR.:
    IEA_LOSS&YR.:
    IEA_NESE&YR.:
    IEA_IRWE&YR.:
    IEA_WAGE&YR.:
    IEA_SEIE&YR.:
    IET4&YR.:
    MEDC&YR.:
    IUA_SSDI&YR.:
    IUA_WC&YR.:
    IUA_TANF&YR.:
    SSN
)
/* ZIP-FIPS PATCH */
ZIP.ZIPFIP(IN=ZIP
    RENAME=(PST&YR.01-PST&YR.12=STATE&YR.01-STATE&YR.12))
/* MFT FOR THE SSIELIG FILE NEEDS TO BE KEPT TO RECREATE THE SSI
    SGA VARIABLE */
%IF &SYSPARM.=2016 %THEN
SSR.SSIELIG(KEEP=SSN MFT);
%ELSE SSR.SSR(KEEP=SSN MFT);
FIND.MASTFIND(IN=MASTFIND KEEP=SSN)
;
BY SSN;

* KEEP ALL BENES WHO HAVE A COMMVARS RECORD;
IF KEEPCOM;
```

## Appendix A.157

JCL/SAS Code: FINANN16

```
* FOR BENES WHO ARE ON THE COMBINED HISTORICAL CER BUT WERE NOT ON
THE ORIGINAL DAF FOR THAT DAF YEAR (IE - BENE WAS NOT IN THE DAF16
2016 ANNUAL BUT WAS ON THE COMBINED CER) CLEAR ALL THE POPULATED
CER VARIABLES. THIS WILL MIMIC WHAT WE WOULD HAVE DONE HAD WE
EXECUTED A 2ND CER PULL DURING THE CREATION OF THE DAF YEAR ANNUAL;
IF CER AND NOT MASTFIND THEN DO;
```

```
  ARRAY CERNVARS (*) IEA_BLIND&YR.:
```

```
    IEA_PASS&YR.:
```

```
    IEA_LOSS&YR.:
```

```
    IEA_NESE&YR.:
```

```
    IEA_IRWE&YR.:
```

```
    IEA_WAGE&YR.:
```

```
    IEA_SEIE&YR.:
```

```
    IUA_SSDI&YR.:
```

```
    IUA_WC&YR.:
```

```
    IUA_TANF&YR.:
```

```
  ;
```

```
  DO I=1 TO DIM(CERNVARS);
```

```
    CERNVARS(I)=.;
```

```
  END;
```

```
  ARRAY CERCVARS (*) MEDC&YR.: MFT
```

```
    IET4&YR.:
```

```
  ;
```

```
  DO I=1 TO DIM(CERCVARS);
```

```
    CERCVARS(I)="";
```

```
  END;
```

```
END;
```

```
/*MASK DAA VARIABLES*/
```

```
ARRAY PSTA (12) PSTA&YR.01 - PSTA&YR.12;
```

```
ARRAY CDR (12) CDR&YR.01 - CDR&YR.12;
```

```
ARRAY WIC (12) WIC&YR.01 - WIC&YR.12;
```

```
DO I = 1 TO 12;
```

```
  IF PSTA(I) IN('N10', 'N11', 'N12', 'N17', 'N18', 'N20', 'N36',  
               'N37', 'N38', 'N39', 'N47', 'N48', 'N49', 'N50')
```

```
  THEN PSTA(I) = 'W01';
```

```
  IF CDR(I) IN('F', 'N', 'W') THEN CDR(I) = 'X';
```

```
  IF WIC(I) IN('A', 'B', 'C', 'N', 'P') THEN WIC(I) = 'S';
```

```
END;
```

## Appendix A.157

JCL/SAS Code: FINANN16

```

*****
**** RECREATE THE VARIABLES THAT ARE BASED ON THE CER VARIABLES TO
    NOW USE THE HISTORICAL CERS - ORIGINAL CODE IN ANNCURR. FOR THIS
    PATCH THIS HAS TO BE EXECUTED FOR ALL YEARS;
ARRAY EARN  (*) %DO MM=1 %TO &TOTMOS; EARN_SSI&&YY&MM  %END;;
ARRAY IEA_NESE (*) %DO MM=1 %TO &TOTMOS; IEA_NESE&&YY&MM %END;;
ARRAY IEA_WAGE (*) %DO MM=1 %TO &TOTMOS; IEA_WAGE&&YY&MM %END;;
ARRAY EIND  (*) %DO MM=1 %TO &TOTMOS; EIND_SSI&&YY&MM  %END;;
ARRAY SGA   (*) %DO MM=1 %TO &TOTMOS; SSIERNSGA&&YY&MM %END;;

/*INCLUDE BLIND AND NONBLIND SGA VALUES FROM UTILITY LIBRARY*/
/*USE ENDYR TO REFERENCE TO SGA VALUES AS NEEDED*/
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(SGAB)";
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(SGANB)";

DO I = 1 TO DIM(EARN);
  EARN(I) = .;
  IF IEA_NESE(I) > .Z OR IEA_WAGE(I) > .Z THEN DO;
    EARN(I) = SUM(OF IEA_NESE(I),IEA_WAGE(I));
  END;

  IF EARN(I) > 0 THEN EIND(I) = 1;
  ELSE IF EARN(I) = 0 THEN EIND(I) = 0;
  ELSE EIND(I) = .;

  /* SGA IS DIFFERENT FOR BLIND PEOPLE */
  /* REVISED CODE UPDATED EVERY YEAR - THIS IS COMPUTED
    ONLY FOR THE NEW YEAR OF DATA COLLECTION. UPDATED
    EARNINGS CEILINGS ARE INCLUDED. EARLIER YEARS ALREADY
    HAVE THE VARIABLE AND IT IS MERGED ON FROM THE OLD FILE */
  IF SUBSTR(MFT,1,1) = 'B' THEN DO;/* BLIND PEOPLE */
    IF EARN(I) >= SGAB(&ENDYR.) THEN SGA(I) = 1;
    ELSE IF 0 <= EARN(I) < SGAB(&ENDYR.) THEN SGA(I) = 0;
    ELSE SGA(I) = .;
  END;
  ELSE DO;
    IF EARN(I) >= SGANB(&ENDYR.) THEN SGA(I) = 1;
    ELSE IF 0 <= EARN(I) < SGANB(&ENDYR.) THEN SGA(I) = 0;
    ELSE SGA(I) = .;
  END;
END;

/*CREATE AN INDICATOR FOR 1619(B) */
ARRAY MEDC ($) %DO MM=1 %TO &TOTMOS; MEDC&&YY&MM %END;;
ARRAY MEDE (*) %DO MM=1 %TO &TOTMOS; MEDE&&YY&MM %END;;

```



## Appendix A.157

JCL/SAS Code: FINANN16

```

DO I = 1 TO DIM(MEDC);
  IF MEDC(I) IN ('C','W','Y') THEN MEDE(I) = 1;
  ELSE IF MEDC(I) > '' THEN MEDE(I) = 0;
  ELSE MEDE(I) = .;
END;

***** SPLIT INTO ENROLLED & NON-ENROLLED *****;
/* USING ARRAY FOR VARIABLES BEING COMPARED TO '' */
ARRAY CHARMISS (*)    LAF&YR.01 -    LAF&YR.12
                    PSTA&YR.01 -    PSTA&YR.12;

/* USING ARRAY FOR VARIABLES BEING COMPARED TO . */
ARRAY NUMMISS (*)    STWSSI&YR.01 -    STWSSI&YR.12
                    STWDI&YR.01 -    STWDI&YR.12
                    STWCM&YR.01 -    STWCM&YR.12;

/* USING ARRAY FOR VARIABLES BEING COMPATED TO 0 (>=0) */
ARRAY LEQZERO (*)    PAYD&YR.01 -    PAYD&YR.12
                    PAYS&YR.01 -    PAYS&YR.12
                    BFWSSI&YR.01 -    BFWSSI&YR.12
                    BFWDI&YR.01 -    BFWDI&YR.12
                    BFWCM&YR.01 -    BFWCM&YR.12;

IN_ANN=0; *FLAG TO SPLIT DATASET BELOW;

/*SEARCHING THROUGH FIRST SET OF VARIABLES*/
DO V=1 TO DIM(CHARMISS) WHILE(IN_ANN=0);
  IF CHARMISS(V) NE '' THEN IN_ANN=1;
END;

/*SEARCHING THROUGH SECOND SET OF VARIABLES IF CRITERIA ABOVE NOT MET*/
IF IN_ANN=0 THEN
DO V=1 TO DIM(NUMMISS) WHILE(IN_ANN=0);
  IF NUMMISS(V) NE . THEN IN_ANN=1;
END;

/*SEARCHING THROUGH THIRD SET OF VARIABLES IF CRITERIA ABOVE NOT MET*/
IF IN_ANN=0 THEN
DO V=1 TO DIM(LEQZERO) WHILE(IN_ANN=0);
  IF LEQZERO(V) > 0 THEN IN_ANN=1;
END;

/*FRA ALL YEAR TRUMPS ALL PREVIOUS CRITERIA*/
IF FRADATE<="01JAN&YYYY."D AND FRADATE NE .

```

## Appendix A.157

JCL/SAS Code: FINANN16

```
THEN IN_ANN=0;

%INCLUDE "OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(LABELS)";

/*DROP VARIABLES THAT SHOULD NOT BE ON FINAL DAF ANNUALS*/
DROP /*MM SSDI_ONSET_LAST PST_MR PST_MR_DM FL_DEATH FL_UNBORN */
    SGAB: SGANB: IET4: MFT;

* OUTPUT ENROLLED ANNUALS;
IF IN_ANN=1 THEN OUTPUT OUTE.Y&YYYY;

*OUTPUT NONENROLLED ANNUALS;
ELSE OUTPUT OUTNE.Y&YYYY.NE;

RUN;

%MEND SPLITANN;
%SPLITANN;

PROC FORMAT;
VALUE $WICDAA 'A' = 'A'
              'B' = 'B'
              'C' = 'C'
              'N' = 'N'
              'P' = 'P'
              'S' = 'S'
              OTHER = 'NOT DAA';
VALUE $CDRDAA 'F' = 'F'
              'N' = 'N'
              'W' = 'W'
              'X' = 'X'
              OTHER = 'NOT DAA';
VALUE $PSTADAA 'N10' = 'N10'
               'N11' = 'N11'
               'N12' = 'N12'
               'N17' = 'N17'
               'N18' = 'N18'
               'N20' = 'N20'
               'N36' = 'N36'
               'N37' = 'N37'
               'N38' = 'N38'
               'N39' = 'N39'
               'N47' = 'N47'
               'N48' = 'N48'
               'N49' = 'N49'
```

## Appendix A.157

JCL/SAS Code: FINANN16

```
'N50' = 'N50'  
'W01' = 'W01'  
OTHER = 'NOT DAA';
```

```
RUN;
```

```
TITLE "CHECK DAA MASKING IN ENROLLED FILE";  
PROC FREQ DATA = OUTE.Y&YYYY.;  
TABLES WIC: CDR: PSTA: / LIST MISSING;  
FORMAT WIC: $WICDAA. CDR: $CDRDAA. PSTA: $PSTADAA.;  
RUN;
```

```
TITLE "CHECK DAA MASKING IN NON-ENROLLED FILE";  
PROC FREQ DATA = OUTNE.Y&YYYY.NE;  
TABLES WIC: CDR: PSTA: / LIST MISSING;  
FORMAT WIC: $WICDAA. CDR: $CDRDAA. PSTA: $PSTADAA.;  
RUN;
```

```
TITLE "FULL ENROLLED ANNUALS";  
PROC CONTENTS DATA = OUTE.Y&YYYY.  
    OUT = OUTCE.ANN&YR.CON; RUN;  
TITLE "FULL NONENROLLED ANNUALS";  
PROC CONTENTS DATA = OUTNE.Y&YYYY.NE  
    OUT = OUTCNE.ANN&YR.CON; RUN;
```

```
/* CREATE 10%DM FOR ANNUAL FILES */  
DATA E10.Y&YYYY.;  
    SET OUTE.Y&YYYY.;  
    IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9'))  
        AND (SUBSTR(SSN,9,1) IN ('0', '5'));  
RUN;
```

```
DATA NE10.Y&YYYY.NE;  
    SET OUTNE.Y&YYYY.NE;  
    IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9'))  
        AND (SUBSTR(SSN,9,1) IN ('0', '5'));  
RUN;
```

```
/*RUNNING AUTOFOF MACRO*/  
PROC CONTENTS DATA=E10.Y&YYYY. OUT=OUTCE10.ANN&YR.CON; RUN;
```

## Appendix A.158

JCL/SAS Code: FINANN21

```

/*****/
/* PROJECT: DAF 50214 */
/* DIRECTOR: */
/* PROGRAM: OPDR.TG.PRD.ETTW.#4941.DAF16.ANN.PRDLIB(FINANN) */
/* PURPOSE: SPLIT ANNUALS FILES INTO ENROLLED/NON-ENROLLED */
/* ANY OBS MEETING ALL FOLLOWING CRITERIA WILL */
/* BE SENT TO NON-ENROLLED */
/* FRADATE<=1/1/&YYYY : FRA ALL YEAR */
/* */
/* LAFYMM = ' ' */
/* PSTAYMM = ' ' */
/* */
/* STWSSIYMM = . */
/* STWDIYMM = . */
/* STWCMYMM = . */
/* */
/* PAYDYMM <= 0 */
/* PAYSYMM <= 0 */
/* BFWSSIYMM <= 0 */
/* BFWDIYMM <= 0 */
/* BFWCMYMM <= 0 */
/* */
/*!! UPDATE YY TO CURRENT DAF YEAR FOR SSI AND SSDI RANGE BELOW !*/
/* */
/*****/
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY MPRINT OBS=MAX;

%LET YYYY=&SYSPARM;
%LET YR=%SUBSTR(&SYSPARM,3);

%LET BEGYR=&SYSPARM;
%LET ENDYR=&SYSPARM; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

%PUT ***** FINALIZING ANNUAL &YYYY (&YR);

%MACRO SPLITANN;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

DATA OUTE.Y&YYYY. ( DROP=V IN_ANN I)
  OUTNE.Y&YYYY.NE( DROP=V IN_ANN I)
  ;
MERGE INANN.Y&YYYY.(DROP= DOBBEST DDBEST

```

## Appendix A.158

JCL/SAS Code: FINANN21

```
SSI94- SSI99
SSI00- SSI21
SSDI94-SSDI99
SSDI00-SSDI21
SSDI_ONSET
SSI_MIN_ELG SSIELIG_ADULT
/* ZIP-FIPS PATCH WILL BE REPLACING
   THE STATEYMM VARIABLES WHICH ON THE 2021
   INTERMEDIATE IS CALLED PSTYMM WHILE IT IS
   STATEYMM ON ALL OTHER YEARS */
%IF &YR.=21 %THEN PST2101-PST2112;
%ELSE STATE&YR.01-STATE&YR.12;
FIPS_BEST:
FIPS_DI:
FIPS_SSI:
ZIP_BEST:
ZIP_DI:
ZIP_SSI:

/* THE CER SOURCED VARIABLES ARE BEING
   REPLACED BY THE HISTORICAL CER/DBAD */
IEA_BLIND:
IEA_PASS:
IEA_LOSS:
IEA_NESE:
IEA_IRWE:
IEA_WAGE:
IEA_SEIE:
MEDC:
IUA_SSDI:
IUA_WC:
IUA_TANF:

/* THE VARS CREATED WITH CER VARS ARE BEING
   RECREATED USING THE HISTORICAL CER */
EARN_SSI&YR.01-EARN_SSI&YR.12
EIND_SSI&YR.01-EIND_SSI&YR.12
SSIERNPGA&YR.01-SSIERNPGA&YR.12
MEDE&YR.01-MEDE&YR.12

RENAME=(%IF &YR.=21 %THEN %DO;
        %DO M=1 %TO 12;
        %LET MM=%SYSFUNC(PUTN(&M,Z2.));
        MEDEX&YR.&MM = MEDX&YR.&MM
        %END;
```

## Appendix A.158

JCL/SAS Code: FINANN21

```
                                %END;))

COMM.COMVARS(IN=KEEPCOM)
STWBFW.BFW (KEEP=SSN
    BMF&YR.: CFBMF&YR.: DENOM&YR.:
    FCI&YR.: NECALC&YR.: NONEARN&YR.:
    STWDI&YR.: STWSSI&YR.: STWCM&YR.:
    BFWDI&YR.: BFWSSI&YR.: BFWCM&YR.:
    BFWDI_PRIM&YR.:
    BFWDI_DEPEND&YR.:
    RENAME=(%DO M=1 %TO 12;
        %LET MM=%SYSFUNC(PUTN(&M,Z2.));
        BFWDI_PRIM&YR.&MM =BFWDI_PRI&YR.&MM
        BFWDI_DEPEND&YR.&MM=BFWDI_DEP&YR.&MM
    %END;))
/* HISTORICAL CER */
CER.CER(IN=CER KEEP=IEA_BLIND&YR.:
    IEA_PASS&YR.:
    IEA_LOSS&YR.:
    IEA_NESE&YR.:
    IEA_IRWE&YR.:
    IEA_WAGE&YR.:
    IEA_SEIE&YR.:
    IET4&YR.:
    MEDC&YR.:
    IUA_SSDI&YR.:
    IUA_WC&YR.:
    IUA_TANF&YR.:
    SSN
)
/* ZIP-FIPS PATCH */
ZIP.ZIPFIP(IN=ZIP
    RENAME=(PST&YR.01-PST&YR.12=STATE&YR.01-STATE&YR.12))
/* MFT FOR THE SSIELIG FILE NEEDS TO BE KEPT TO RECREATE THE SSI
    SGA VARIABLE */
%IF &SYSPARM.=2016 %THEN
SSR.SSIELIG(KEEP=SSN MFT);
%ELSE SSR.SSR(KEEP=SSN MFT);
FIND.MASTFIND(IN=MASTFIND KEEP=SSN)
;
BY SSN;

* KEEP ALL BENES WHO HAVE A COMMVARS RECORD;
IF KEEPCOM;
```

## Appendix A.158

JCL/SAS Code: FINANN21

```
* FOR BENES WHO ARE ON THE COMBINED HISTORICAL CER BUT WERE NOT ON
THE ORIGINAL DAF FOR THAT DAF YEAR (IE - BENE WAS NOT IN THE DAF16
2016 ANNUAL BUT WAS ON THE COMBINED CER) CLEAR ALL THE POPULATED
CER VARIABLES. THIS WILL MIMIC WHAT WE WOULD HAVE DONE HAD WE
EXECUTED A 2ND CER PULL DURING THE CREATION OF THE DAF YEAR ANNUAL;
IF CER AND NOT MASTFIND THEN DO;
```

```
  ARRAY CERNVARS (*) IEA_BLIND&YR.:
```

```
    IEA_PASS&YR.:
```

```
    IEA_LOSS&YR.:
```

```
    IEA_NESE&YR.:
```

```
    IEA_IRWE&YR.:
```

```
    IEA_WAGE&YR.:
```

```
    IEA_SEIE&YR.:
```

```
    IUA_SSDI&YR.:
```

```
    IUA_WC&YR.:
```

```
    IUA_TANF&YR.:
```

```
  ;
```

```
  DO I=1 TO DIM(CERNVARS);
```

```
    CERNVARS(I)=.;
```

```
  END;
```

```
  ARRAY CERCVARS (*) MEDC&YR.: MFT
```

```
    IET4&YR.:
```

```
  ;
```

```
  DO I=1 TO DIM(CERCVARS);
```

```
    CERCVARS(I)="";
```

```
  END;
```

```
END;
```

```
/*MASK DAA VARIABLES*/
```

```
ARRAY PSTA (12) PSTA&YR.01 - PSTA&YR.12;
```

```
ARRAY CDR (12) CDR&YR.01 - CDR&YR.12;
```

```
ARRAY WIC (12) WIC&YR.01 - WIC&YR.12;
```

```
DO I = 1 TO 12;
```

```
  IF PSTA(I) IN('N10', 'N11', 'N12', 'N17', 'N18', 'N20', 'N36',  
               'N37', 'N38', 'N39', 'N47', 'N48', 'N49', 'N50')
```

```
  THEN PSTA(I) = 'W01';
```

```
  IF CDR(I) IN('F', 'N', 'W') THEN CDR(I) = 'X';
```

```
  IF WIC(I) IN('A', 'B', 'C', 'N', 'P') THEN WIC(I) = 'S';
```

```
END;
```

## Appendix A.158

JCL/SAS Code: FINANN21

```

*****
**** RECREATE THE VARIABLES THAT ARE BASED ON THE CER VARIABLES TO
    NOW USE THE HISTORICAL CERS - ORIGINAL CODE IN ANNCURR. FOR THIS
    PATCH THIS HAS TO BE EXECUTED FOR ALL YEARS;
ARRAY EARN  (*) %DO MM=1 %TO &TOTMOS; EARN_SSI&&YY&MM  %END;;
ARRAY IEA_NESE (*) %DO MM=1 %TO &TOTMOS; IEA_NESE&&YY&MM %END;;
ARRAY IEA_WAGE (*) %DO MM=1 %TO &TOTMOS; IEA_WAGE&&YY&MM %END;;
ARRAY EIND  (*) %DO MM=1 %TO &TOTMOS; EIND_SSI&&YY&MM  %END;;
ARRAY SGA   (*) %DO MM=1 %TO &TOTMOS; SSIERNSGA&&YY&MM %END;;

/*INCLUDE BLIND AND NONBLIND SGA VALUES FROM UTILITY LIBRARY*/
/*USE ENDYR TO REFERENCE TO SGA VALUES AS NEEDED*/
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(SGAB)";
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(SGANB)";

DO I = 1 TO DIM(EARN);
  EARN(I) = .;
  IF IEA_NESE(I) > .Z OR IEA_WAGE(I) > .Z THEN DO;
    EARN(I) = SUM(OF IEA_NESE(I),IEA_WAGE(I));
  END;

  IF EARN(I) > 0 THEN EIND(I) = 1;
  ELSE IF EARN(I) = 0 THEN EIND(I) = 0;
  ELSE EIND(I) = .;

  /* SGA IS DIFFERENT FOR BLIND PEOPLE */
  /* REVISED CODE UPDATED EVERY YEAR - THIS IS COMPUTED
    ONLY FOR THE NEW YEAR OF DATA COLLECTION. UPDATED
    EARNINGS CEILINGS ARE INCLUDED. EARLIER YEARS ALREADY
    HAVE THE VARIABLE AND IT IS MERGED ON FROM THE OLD FILE */
  IF SUBSTR(MFT,1,1) = 'B' THEN DO;/* BLIND PEOPLE */
    IF EARN(I) >= SGAB(&ENDYR.) THEN SGA(I) = 1;
    ELSE IF 0 <= EARN(I) < SGAB(&ENDYR.) THEN SGA(I) = 0;
    ELSE SGA(I) = .;
  END;
  ELSE DO;
    IF EARN(I) >= SGANB(&ENDYR.) THEN SGA(I) = 1;
    ELSE IF 0 <= EARN(I) < SGANB(&ENDYR.) THEN SGA(I) = 0;
    ELSE SGA(I) = .;
  END;
END;

/*CREATE AN INDICATOR FOR 1619(B) */
ARRAY MEDC (*) $ %DO MM=1 %TO &TOTMOS; MEDC&&YY&MM %END;;
ARRAY MEDE (*) %DO MM=1 %TO &TOTMOS; MEDE&&YY&MM %END;;

```



## Appendix A.158

JCL/SAS Code: FINANN21

```

DO I = 1 TO DIM(MEDC);
  IF MEDC(I) IN ('C','W','Y') THEN MEDE(I) = 1;
  ELSE IF MEDC(I) > '' THEN MEDE(I) = 0;
  ELSE MEDE(I) = .;
END;

***** SPLIT INTO ENROLLED & NON-ENROLLED *****;
/* USING ARRAY FOR VARIABLES BEING COMPARED TO '' */
ARRAY CHARMISS (*)    LAF&YR.01 -    LAF&YR.12
                    PSTA&YR.01 -    PSTA&YR.12;

/* USING ARRAY FOR VARIABLES BEING COMPARED TO . */
ARRAY NUMMISS (*)    STWSSI&YR.01 -  STWSSI&YR.12
                    STWDI&YR.01 -   STWDI&YR.12
                    STWCM&YR.01 -   STWCM&YR.12;

/* USING ARRAY FOR VARIABLES BEING COMPARED TO 0 (>=0) */
ARRAY LEQZERO (*)    PAYD&YR.01 -    PAYD&YR.12
                    PAYS&YR.01 -    PAYS&YR.12
                    BFWSSI&YR.01 -  BFWSSI&YR.12
                    BFWDI&YR.01 -  BFWDI&YR.12
                    BFWCM&YR.01 -  BFWCM&YR.12;

IN_ANN=0; *FLAG TO SPLIT DATASET BELOW;

/*SEARCHING THROUGH FIRST SET OF VARIABLES*/
DO V=1 TO DIM(CHARMISS) WHILE(IN_ANN=0);
  IF CHARMISS(V) NE '' THEN IN_ANN=1;
END;

/*SEARCHING THROUGH SECOND SET OF VARIABLES IF CRITERIA ABOVE NOT MET*/
IF IN_ANN=0 THEN
DO V=1 TO DIM(NUMMISS) WHILE(IN_ANN=0);
  IF NUMMISS(V) NE . THEN IN_ANN=1;
END;

/*SEARCHING THROUGH THIRD SET OF VARIABLES IF CRITERIA ABOVE NOT MET*/
IF IN_ANN=0 THEN
DO V=1 TO DIM(LEQZERO) WHILE(IN_ANN=0);
  IF LEQZERO(V) > 0 THEN IN_ANN=1;
END;

/*FRA ALL YEAR TRUMPS ALL PREVIOUS CRITERIA*/
IF FRADATE<="01JAN&YYYY."D AND FRADATE NE .

```

## Appendix A.158

JCL/SAS Code: FINANN21

```
THEN IN_ANN=0;

%INCLUDE "OPDR.TG.PRD.ETTW.#8047.DAF21.ANN.PRDLIB(LABELS)";

/*DROP VARIABLES THAT SHOULD NOT BE ON FINAL DAF ANNUALS*/
DROP /*MM SSDI_ONSET_LAST PST_MR PST_MR_DM FL_DEATH FL_UNBORN */
    SGAB: SGANB: IET4: MFT;

* OUTPUT ENROLLED ANNUALS;
IF IN_ANN=1 THEN OUTPUT OUTE.Y&YYYY;

*OUTPUT NONENROLLED ANNUALS;
ELSE OUTPUT OUTNE.Y&YYYY.NE;

RUN;

%MEND SPLITANN;
%SPLITANN;

PROC FORMAT;
VALUE $WICDAA 'A' = 'A'
              'B' = 'B'
              'C' = 'C'
              'N' = 'N'
              'P' = 'P'
              'S' = 'S'
              OTHER = 'NOT DAA';
VALUE $CDRDAA 'F' = 'F'
              'N' = 'N'
              'W' = 'W'
              'X' = 'X'
              OTHER = 'NOT DAA';
VALUE $PSTADAA 'N10' = 'N10'
               'N11' = 'N11'
               'N12' = 'N12'
               'N17' = 'N17'
               'N18' = 'N18'
               'N20' = 'N20'
               'N36' = 'N36'
               'N37' = 'N37'
               'N38' = 'N38'
               'N39' = 'N39'
               'N47' = 'N47'
               'N48' = 'N48'
               'N49' = 'N49'
```

## Appendix A.158

JCL/SAS Code: FINANN21

```
'N50' = 'N50'  
'W01' = 'W01'  
OTHER = 'NOT DAA';
```

```
RUN;
```

```
TITLE "CHECK DAA MASKING IN ENROLLED FILE";  
PROC FREQ DATA = OUTE.Y&YYYY.;  
TABLES WIC: CDR: PSTA: / LIST MISSING;  
FORMAT WIC: $WICDAA. CDR: $CDRDAA. PSTA: $PSTADAA.;  
RUN;
```

```
TITLE "CHECK DAA MASKING IN NON-ENROLLED FILE";  
PROC FREQ DATA = OUTNE.Y&YYYY.NE;  
TABLES WIC: CDR: PSTA: / LIST MISSING;  
FORMAT WIC: $WICDAA. CDR: $CDRDAA. PSTA: $PSTADAA.;  
RUN;
```

```
TITLE "FULL ENROLLED ANNUALS";  
PROC CONTENTS DATA = OUTE.Y&YYYY.  
    OUT = OUTCE.ANN&YR.CON; RUN;  
TITLE "FULL NONENROLLED ANNUALS";  
PROC CONTENTS DATA = OUTNE.Y&YYYY.NE  
    OUT = OUTCNE.ANN&YR.CON; RUN;
```

```
/* CREATE 10%DM FOR ANNUAL FILES */  
DATA E10.Y&YYYY.;  
    SET OUTE.Y&YYYY.;  
    IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9'))  
        AND (SUBSTR(SSN,9,1) IN ('0', '5'));  
RUN;
```

```
DATA NE10.Y&YYYY.NE;  
    SET OUTNE.Y&YYYY.NE;  
    IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9'))  
        AND (SUBSTR(SSN,9,1) IN ('0', '5'));  
RUN;
```

```
/*RUNNING AUTOFOF MACRO*/  
PROC CONTENTS DATA=E10.Y&YYYY. OUT=OUTCE10.ANN&YR.CON; RUN;
```

Appendix A.159

JCL/SAS Code: LABELS

label

ALLGAMT&YR.01="ALLEGED EARNINGS AMOUNT &YR.01"  
ALLGAMT&YR.02="ALLEGED EARNINGS AMOUNT &YR.02"  
ALLGAMT&YR.03="ALLEGED EARNINGS AMOUNT &YR.03"  
ALLGAMT&YR.04="ALLEGED EARNINGS AMOUNT &YR.04"  
ALLGAMT&YR.05="ALLEGED EARNINGS AMOUNT &YR.05"  
ALLGAMT&YR.06="ALLEGED EARNINGS AMOUNT &YR.06"  
ALLGAMT&YR.07="ALLEGED EARNINGS AMOUNT &YR.07"  
ALLGAMT&YR.08="ALLEGED EARNINGS AMOUNT &YR.08"  
ALLGAMT&YR.09="ALLEGED EARNINGS AMOUNT &YR.09"  
ALLGAMT&YR.10="ALLEGED EARNINGS AMOUNT &YR.10"  
ALLGAMT&YR.11="ALLEGED EARNINGS AMOUNT &YR.11"  
ALLGAMT&YR.12="ALLEGED EARNINGS AMOUNT &YR.12"  
ALX&YR.01="ADJUDICATIVE LEVEL OF DECISION &YR.01"  
ALX&YR.02="ADJUDICATIVE LEVEL OF DECISION &YR.02"  
ALX&YR.03="ADJUDICATIVE LEVEL OF DECISION &YR.03"  
ALX&YR.04="ADJUDICATIVE LEVEL OF DECISION &YR.04"  
ALX&YR.05="ADJUDICATIVE LEVEL OF DECISION &YR.05"  
ALX&YR.06="ADJUDICATIVE LEVEL OF DECISION &YR.06"  
ALX&YR.07="ADJUDICATIVE LEVEL OF DECISION &YR.07"  
ALX&YR.08="ADJUDICATIVE LEVEL OF DECISION &YR.08"  
ALX&YR.09="ADJUDICATIVE LEVEL OF DECISION &YR.09"  
ALX&YR.10="ADJUDICATIVE LEVEL OF DECISION &YR.10"  
ALX&YR.11="ADJUDICATIVE LEVEL OF DECISION &YR.11"  
ALX&YR.12="ADJUDICATIVE LEVEL OF DECISION &YR.12"  
BFWCM&YR.01=  
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.01"  
BFWCM&YR.02=  
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.02"  
BFWCM&YR.03=  
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.03"  
BFWCM&YR.04=  
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.04"  
BFWCM&YR.05=  
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.05"  
BFWCM&YR.06=  
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.06"  
BFWCM&YR.07=  
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.07"  
BFWCM&YR.08=  
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.08"  
BFWCM&YR.09=  
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.09"  
BFWCM&YR.10=

Appendix A.159

JCL/SAS Code: LABELS

"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.10"  
BFWCM&YR.11=  
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.11"  
BFWCM&YR.12=  
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK &YR.12"  
BFWDI&YR.01="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.01"  
BFWDI&YR.02="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.02"  
BFWDI&YR.03="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.03"  
BFWDI&YR.04="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.04"  
BFWDI&YR.05="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.05"  
BFWDI&YR.06="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.06"  
BFWDI&YR.07="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.07"  
BFWDI&YR.08="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.08"  
BFWDI&YR.09="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.09"  
BFWDI&YR.10="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.10"  
BFWDI&YR.11="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.11"  
BFWDI&YR.12="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.12"  
BFWDI\_DEP&YR.01=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.01, AUXILIARY BENEFICIARY"  
BFWDI\_DEP&YR.02=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.02, AUXILIARY BENEFICIARY"  
BFWDI\_DEP&YR.03=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.03, AUXILIARY BENEFICIARY"  
BFWDI\_DEP&YR.04=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.04, AUXILIARY BENEFICIARY"  
BFWDI\_DEP&YR.05=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.05, AUXILIARY BENEFICIARY"  
BFWDI\_DEP&YR.06=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.06, AUXILIARY BENEFICIARY"  
BFWDI\_DEP&YR.07=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.07, AUXILIARY BENEFICIARY"  
BFWDI\_DEP&YR.08=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.08, AUXILIARY BENEFICIARY"  
BFWDI\_DEP&YR.09=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.09, AUXILIARY BENEFICIARY"  
BFWDI\_DEP&YR.10=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.10, AUXILIARY BENEFICIARY"  
BFWDI\_DEP&YR.11=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.11, AUXILIARY BENEFICIARY"  
BFWDI\_DEP&YR.12=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.12, AUXILIARY BENEFICIARY"  
BFWDI\_PRI&YR.01=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.01, PRIMARY BENEFICIARY"  
BFWDI\_PRI&YR.02=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.02, PRIMARY BENEFICIARY"

Appendix A.159

JCL/SAS Code: LABELS

BFWDI\_PRI&YR.03=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.03, PRIMARY BENEFICIARY"  
BFWDI\_PRI&YR.04=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.04, PRIMARY BENEFICIARY"  
BFWDI\_PRI&YR.05=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.05, PRIMARY BENEFICIARY"  
BFWDI\_PRI&YR.06=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.06, PRIMARY BENEFICIARY"  
BFWDI\_PRI&YR.07=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.07, PRIMARY BENEFICIARY"  
BFWDI\_PRI&YR.08=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.08, PRIMARY BENEFICIARY"  
BFWDI\_PRI&YR.09=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.09, PRIMARY BENEFICIARY"  
BFWDI\_PRI&YR.10=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.10, PRIMARY BENEFICIARY"  
BFWDI\_PRI&YR.11=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.11, PRIMARY BENEFICIARY"  
BFWDI\_PRI&YR.12=  
"SSDI ESTIMATED BENEFITS FORGONE FOR WORK &YR.12, PRIMARY BENEFICIARY"  
BFWSSI&YR.01="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.01"  
BFWSSI&YR.02="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.02"  
BFWSSI&YR.03="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.03"  
BFWSSI&YR.04="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.04"  
BFWSSI&YR.05="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.05"  
BFWSSI&YR.06="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.06"  
BFWSSI&YR.07="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.07"  
BFWSSI&YR.08="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.08"  
BFWSSI&YR.09="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.09"  
BFWSSI&YR.10="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.10"  
BFWSSI&YR.11="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.11"  
BFWSSI&YR.12="SSI ESTIMATED BENEFITS FORGONE FOR WORK &YR.12"  
BPD&YR.01="SSDI BENEFIT PAYMENT DESIGNATION &YR.01"  
BPD&YR.02="SSDI BENEFIT PAYMENT DESIGNATION &YR.02"  
BPD&YR.03="SSDI BENEFIT PAYMENT DESIGNATION &YR.03"  
BPD&YR.04="SSDI BENEFIT PAYMENT DESIGNATION &YR.04"  
BPD&YR.05="SSDI BENEFIT PAYMENT DESIGNATION &YR.05"  
BPD&YR.06="SSDI BENEFIT PAYMENT DESIGNATION &YR.06"  
BPD&YR.07="SSDI BENEFIT PAYMENT DESIGNATION &YR.07"  
BPD&YR.08="SSDI BENEFIT PAYMENT DESIGNATION &YR.08"  
BPD&YR.09="SSDI BENEFIT PAYMENT DESIGNATION &YR.09"  
BPD&YR.10="SSDI BENEFIT PAYMENT DESIGNATION &YR.10"  
BPD&YR.11="SSDI BENEFIT PAYMENT DESIGNATION &YR.11"  
BPD&YR.12="SSDI BENEFIT PAYMENT DESIGNATION &YR.12"  
BMF&YR.01="BUDGET MONTH FACTOR &YR.01"

Appendix A.159

JCL/SAS Code: LABELS

BMF&YR.02="BUDGET MONTH FACTOR &YR.02"  
BMF&YR.03="BUDGET MONTH FACTOR &YR.03"  
BMF&YR.04="BUDGET MONTH FACTOR &YR.04"  
BMF&YR.05="BUDGET MONTH FACTOR &YR.05"  
BMF&YR.06="BUDGET MONTH FACTOR &YR.06"  
BMF&YR.07="BUDGET MONTH FACTOR &YR.07"  
BMF&YR.08="BUDGET MONTH FACTOR &YR.08"  
BMF&YR.09="BUDGET MONTH FACTOR &YR.09"  
BMF&YR.10="BUDGET MONTH FACTOR &YR.10"  
BMF&YR.11="BUDGET MONTH FACTOR &YR.11"  
BMF&YR.12="BUDGET MONTH FACTOR &YR.12"  
CDR&YR.01="SSDI CESSATION OF DISABILITY REASON &YR.01"  
CDR&YR.02="SSDI CESSATION OF DISABILITY REASON &YR.02"  
CDR&YR.03="SSDI CESSATION OF DISABILITY REASON &YR.03"  
CDR&YR.04="SSDI CESSATION OF DISABILITY REASON &YR.04"  
CDR&YR.05="SSDI CESSATION OF DISABILITY REASON &YR.05"  
CDR&YR.06="SSDI CESSATION OF DISABILITY REASON &YR.06"  
CDR&YR.07="SSDI CESSATION OF DISABILITY REASON &YR.07"  
CDR&YR.08="SSDI CESSATION OF DISABILITY REASON &YR.08"  
CDR&YR.09="SSDI CESSATION OF DISABILITY REASON &YR.09"  
CDR&YR.10="SSDI CESSATION OF DISABILITY REASON &YR.10"  
CDR&YR.11="SSDI CESSATION OF DISABILITY REASON &YR.11"  
CDR&YR.12="SSDI CESSATION OF DISABILITY REASON &YR.12"  
CFBMF&YR.01="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.01"  
CFBMF&YR.02="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.02"  
CFBMF&YR.03="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.03"  
CFBMF&YR.04="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.04"  
CFBMF&YR.05="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.05"  
CFBMF&YR.06="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.06"  
CFBMF&YR.07="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.07"  
CFBMF&YR.08="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.08"  
CFBMF&YR.09="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.09"  
CFBMF&YR.10="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.10"  
CFBMF&YR.11="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.11"  
CFBMF&YR.12="COUNTERFACTUAL BUDGET MONTH FACTOR &YR.12"  
CONC&YR.01="CONCURRENT BENEFICIARY STATUS &YR.01"  
CONC&YR.02="CONCURRENT BENEFICIARY STATUS &YR.02"  
CONC&YR.03="CONCURRENT BENEFICIARY STATUS &YR.03"  
CONC&YR.04="CONCURRENT BENEFICIARY STATUS &YR.04"  
CONC&YR.05="CONCURRENT BENEFICIARY STATUS &YR.05"  
CONC&YR.06="CONCURRENT BENEFICIARY STATUS &YR.06"  
CONC&YR.07="CONCURRENT BENEFICIARY STATUS &YR.07"  
CONC&YR.08="CONCURRENT BENEFICIARY STATUS &YR.08"  
CONC&YR.09="CONCURRENT BENEFICIARY STATUS &YR.09"  
CONC&YR.10="CONCURRENT BENEFICIARY STATUS &YR.10"

Appendix A.159

JCL/SAS Code: LABELS

CONC&YR.11="CONCURRENT BENEFICIARY STATUS &YR.11"  
CONC&YR.12="CONCURRENT BENEFICIARY STATUS &YR.12"  
CURCOMP&YR.01="CURRENT COMPOSITION CODE &YR.01"  
CURCOMP&YR.02="CURRENT COMPOSITION CODE &YR.02"  
CURCOMP&YR.03="CURRENT COMPOSITION CODE &YR.03"  
CURCOMP&YR.04="CURRENT COMPOSITION CODE &YR.04"  
CURCOMP&YR.05="CURRENT COMPOSITION CODE &YR.05"  
CURCOMP&YR.06="CURRENT COMPOSITION CODE &YR.06"  
CURCOMP&YR.07="CURRENT COMPOSITION CODE &YR.07"  
CURCOMP&YR.08="CURRENT COMPOSITION CODE &YR.08"  
CURCOMP&YR.09="CURRENT COMPOSITION CODE &YR.09"  
CURCOMP&YR.10="CURRENT COMPOSITION CODE &YR.10"  
CURCOMP&YR.11="CURRENT COMPOSITION CODE &YR.11"  
CURCOMP&YR.12="CURRENT COMPOSITION CODE &YR.12"  
DENOM&YR.01="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.01"  
DENOM&YR.02="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.02"  
DENOM&YR.03="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.03"  
DENOM&YR.04="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.04"  
DENOM&YR.05="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.05"  
DENOM&YR.06="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.06"  
DENOM&YR.07="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.07"  
DENOM&YR.08="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.08"  
DENOM&YR.09="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.09"  
DENOM&YR.10="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.10"  
DENOM&YR.11="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.11"  
DENOM&YR.12="DENOMINATOR FOR CALCULATING NONEARNED INCOME &YR.12"  
DIERNLVL&YR.01="SSDI EARNING LEVEL INDICATOR &YR.01"  
DIERNLVL&YR.02="SSDI EARNING LEVEL INDICATOR &YR.02"  
DIERNLVL&YR.03="SSDI EARNING LEVEL INDICATOR &YR.03"  
DIERNLVL&YR.04="SSDI EARNING LEVEL INDICATOR &YR.04"  
DIERNLVL&YR.05="SSDI EARNING LEVEL INDICATOR &YR.05"  
DIERNLVL&YR.06="SSDI EARNING LEVEL INDICATOR &YR.06"  
DIERNLVL&YR.07="SSDI EARNING LEVEL INDICATOR &YR.07"  
DIERNLVL&YR.08="SSDI EARNING LEVEL INDICATOR &YR.08"  
DIERNLVL&YR.09="SSDI EARNING LEVEL INDICATOR &YR.09"  
DIERNLVL&YR.10="SSDI EARNING LEVEL INDICATOR &YR.10"  
DIERNLVL&YR.11="SSDI EARNING LEVEL INDICATOR &YR.11"  
DIERNLVL&YR.12="SSDI EARNING LEVEL INDICATOR &YR.12"  
DIRPAY&YR.01="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.01"  
DIRPAY&YR.02="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.02"  
DIRPAY&YR.03="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.03"  
DIRPAY&YR.04="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.04"  
DIRPAY&YR.05="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.05"  
DIRPAY&YR.06="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.06"  
DIRPAY&YR.07="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.07"



Appendix A.159

JCL/SAS Code: LABELS

DIRPAY&YR.08="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.08"  
DIRPAY&YR.09="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.09"  
DIRPAY&YR.10="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.10"  
DIRPAY&YR.11="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.11"  
DIRPAY&YR.12="SSDI ACTUAL BENEFIT PAID AMOUNT &YR.12"  
DOBBEST="BEST DATE OF BIRTH"  
DODBEST="BEST DATE OF DEATH"  
DOEI="SSDI INITIAL ENTITLEMENT DATE"  
DPEN&YR.01="SSDI NUMBER OF DEPENDENTS &YR.01"  
DPEN&YR.02="SSDI NUMBER OF DEPENDENTS &YR.02"  
DPEN&YR.03="SSDI NUMBER OF DEPENDENTS &YR.03"  
DPEN&YR.04="SSDI NUMBER OF DEPENDENTS &YR.04"  
DPEN&YR.05="SSDI NUMBER OF DEPENDENTS &YR.05"  
DPEN&YR.06="SSDI NUMBER OF DEPENDENTS &YR.06"  
DPEN&YR.07="SSDI NUMBER OF DEPENDENTS &YR.07"  
DPEN&YR.08="SSDI NUMBER OF DEPENDENTS &YR.08"  
DPEN&YR.09="SSDI NUMBER OF DEPENDENTS &YR.09"  
DPEN&YR.10="SSDI NUMBER OF DEPENDENTS &YR.10"  
DPEN&YR.11="SSDI NUMBER OF DEPENDENTS &YR.11"  
DPEN&YR.12="SSDI NUMBER OF DEPENDENTS &YR.12"  
DUED&YR.01="SSDI BENEFIT DUE &YR.01"  
DUED&YR.02="SSDI BENEFIT DUE &YR.02"  
DUED&YR.03="SSDI BENEFIT DUE &YR.03"  
DUED&YR.04="SSDI BENEFIT DUE &YR.04"  
DUED&YR.05="SSDI BENEFIT DUE &YR.05"  
DUED&YR.06="SSDI BENEFIT DUE &YR.06"  
DUED&YR.07="SSDI BENEFIT DUE &YR.07"  
DUED&YR.08="SSDI BENEFIT DUE &YR.08"  
DUED&YR.09="SSDI BENEFIT DUE &YR.09"  
DUED&YR.10="SSDI BENEFIT DUE &YR.10"  
DUED&YR.11="SSDI BENEFIT DUE &YR.11"  
DUED&YR.12="SSDI BENEFIT DUE &YR.12"  
DUEO&YR.01="SSDI DEPENDENT BENEFIT DUE &YR.01"  
DUEO&YR.02="SSDI DEPENDENT BENEFIT DUE &YR.02"  
DUEO&YR.03="SSDI DEPENDENT BENEFIT DUE &YR.03"  
DUEO&YR.04="SSDI DEPENDENT BENEFIT DUE &YR.04"  
DUEO&YR.05="SSDI DEPENDENT BENEFIT DUE &YR.05"  
DUEO&YR.06="SSDI DEPENDENT BENEFIT DUE &YR.06"  
DUEO&YR.07="SSDI DEPENDENT BENEFIT DUE &YR.07"  
DUEO&YR.08="SSDI DEPENDENT BENEFIT DUE &YR.08"  
DUEO&YR.09="SSDI DEPENDENT BENEFIT DUE &YR.09"  
DUEO&YR.10="SSDI DEPENDENT BENEFIT DUE &YR.10"  
DUEO&YR.11="SSDI DEPENDENT BENEFIT DUE &YR.11"  
DUEO&YR.12="SSDI DEPENDENT BENEFIT DUE &YR.12"  
DUES&YR.01="SSI BENEFIT DUE &YR.01"

Appendix A.159

JCL/SAS Code: LABELS

```
DUES&YR.02="SSI BENEFIT DUE &YR.02"  
DUES&YR.03="SSI BENEFIT DUE &YR.03"  
DUES&YR.04="SSI BENEFIT DUE &YR.04"  
DUES&YR.05="SSI BENEFIT DUE &YR.05"  
DUES&YR.06="SSI BENEFIT DUE &YR.06"  
DUES&YR.07="SSI BENEFIT DUE &YR.07"  
DUES&YR.08="SSI BENEFIT DUE &YR.08"  
DUES&YR.09="SSI BENEFIT DUE &YR.09"  
DUES&YR.10="SSI BENEFIT DUE &YR.10"  
DUES&YR.11="SSI BENEFIT DUE &YR.11"  
DUES&YR.12="SSI BENEFIT DUE &YR.12"  
DXPRIBEST="PRIMARY DIAGNOSIS CODE MOST RECENT, BEST FROM ALL SOURCES"  
DXPRI_DI&YR.01=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.01"  
DXPRI_DI&YR.02=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.02"  
DXPRI_DI&YR.03=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.03"  
DXPRI_DI&YR.04=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.04"  
DXPRI_DI&YR.05=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.05"  
DXPRI_DI&YR.06=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.06"  
DXPRI_DI&YR.07=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.07"  
DXPRI_DI&YR.08=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.08"  
DXPRI_DI&YR.09=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.09"  
DXPRI_DI&YR.10=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.10"  
DXPRI_DI&YR.11=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.11"  
DXPRI_DI&YR.12=  
"SSDI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.12"  
DXPRI_SSI&YR.01=  
"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.01"  
DXPRI_SSI&YR.02=  
"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.02"  
DXPRI_SSI&YR.03=  
"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.03"  
DXPRI_SSI&YR.04=  
"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.04"  
DXPRI_SSI&YR.05=
```

Appendix A.159

JCL/SAS Code: LABELS

"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.05"  
DXPRI\_SSI&YR.06=  
"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.06"  
DXPRI\_SSI&YR.07=  
"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.07"  
DXPRI\_SSI&YR.08=  
"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.08"  
DXPRI\_SSI&YR.09=  
"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.09"  
DXPRI\_SSI&YR.10=  
"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.10"  
DXPRI\_SSI&YR.11=  
"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.11"  
DXPRI\_SSI&YR.12=  
"SSI PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.12"  
DXPRIBEST&YR.01="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.01"  
DXPRIBEST&YR.02="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.02"  
DXPRIBEST&YR.03="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.03"  
DXPRIBEST&YR.04="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.04"  
DXPRIBEST&YR.05="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.05"  
DXPRIBEST&YR.06="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.06"  
DXPRIBEST&YR.07="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.07"  
DXPRIBEST&YR.08="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.08"  
DXPRIBEST&YR.09="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.09"  
DXPRIBEST&YR.10="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.10"  
DXPRIBEST&YR.11="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.11"  
DXPRIBEST&YR.12="PRIMARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.12"  
DXSEC\_DI&YR.01=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.01"  
DXSEC\_DI&YR.02=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.02"  
DXSEC\_DI&YR.03=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.03"  
DXSEC\_DI&YR.04=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.04"  
DXSEC\_DI&YR.05=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.05"  
DXSEC\_DI&YR.06=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.06"  
DXSEC\_DI&YR.07=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.07"  
DXSEC\_DI&YR.08=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.08"  
DXSEC\_DI&YR.09=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.09"

Appendix A.159

JCL/SAS Code: LABELS

DXSEC\_DI&YR.10=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.10"  
DXSEC\_DI&YR.11=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.11"  
DXSEC\_DI&YR.12=  
"SSDI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.12"  
DXSEC\_SSI&YR.01=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.01"  
DXSEC\_SSI&YR.02=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.02"  
DXSEC\_SSI&YR.03=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.03"  
DXSEC\_SSI&YR.04=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.04"  
DXSEC\_SSI&YR.05=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.05"  
DXSEC\_SSI&YR.06=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.06"  
DXSEC\_SSI&YR.07=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.07"  
DXSEC\_SSI&YR.08=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.08"  
DXSEC\_SSI&YR.09=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.09"  
DXSEC\_SSI&YR.10=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.10"  
DXSEC\_SSI&YR.11=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.11"  
DXSEC\_SSI&YR.12=  
"SSI SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.12"  
DXSECBEST&YR.01=  
"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.01"  
DXSECBEST&YR.02=  
"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.02"  
DXSECBEST&YR.03=  
"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.03"  
DXSECBEST&YR.04=  
"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.04"  
DXSECBEST&YR.05=  
"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.05"  
DXSECBEST&YR.06=  
"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.06"  
DXSECBEST&YR.07=  
"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.07"  
DXSECBEST&YR.08=

Appendix A.159

JCL/SAS Code: LABELS

"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.08"  
DXSECBEST&YR.09=  
"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.09"  
DXSECBEST&YR.10=  
"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.10"  
DXSECBEST&YR.11=  
"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.11"  
DXSECBEST&YR.12=  
"SECONDARY DIAGNOSIS CODE, BEST FROM ALL SOURCES &YR.12"  
EARN\_SSI&YR.01="SSI EARNED INCOME &YR.01"  
EARN\_SSI&YR.02="SSI EARNED INCOME &YR.02"  
EARN\_SSI&YR.03="SSI EARNED INCOME &YR.03"  
EARN\_SSI&YR.04="SSI EARNED INCOME &YR.04"  
EARN\_SSI&YR.05="SSI EARNED INCOME &YR.05"  
EARN\_SSI&YR.06="SSI EARNED INCOME &YR.06"  
EARN\_SSI&YR.07="SSI EARNED INCOME &YR.07"  
EARN\_SSI&YR.08="SSI EARNED INCOME &YR.08"  
EARN\_SSI&YR.09="SSI EARNED INCOME &YR.09"  
EARN\_SSI&YR.10="SSI EARNED INCOME &YR.10"  
EARN\_SSI&YR.11="SSI EARNED INCOME &YR.11"  
EARN\_SSI&YR.12="SSI EARNED INCOME &YR.12"  
EDX&YR.01="EDUCATION LEVEL &YR.01"  
EDX&YR.02="EDUCATION LEVEL &YR.02"  
EDX&YR.03="EDUCATION LEVEL &YR.03"  
EDX&YR.04="EDUCATION LEVEL &YR.04"  
EDX&YR.05="EDUCATION LEVEL &YR.05"  
EDX&YR.06="EDUCATION LEVEL &YR.06"  
EDX&YR.07="EDUCATION LEVEL &YR.07"  
EDX&YR.08="EDUCATION LEVEL &YR.08"  
EDX&YR.09="EDUCATION LEVEL &YR.09"  
EDX&YR.10="EDUCATION LEVEL &YR.10"  
EDX&YR.11="EDUCATION LEVEL &YR.11"  
EDX&YR.12="EDUCATION LEVEL &YR.12"  
EICM&YR.01="SSI COUNTABLE EARNED INCOME &YR.01"  
EICM&YR.02="SSI COUNTABLE EARNED INCOME &YR.02"  
EICM&YR.03="SSI COUNTABLE EARNED INCOME &YR.03"  
EICM&YR.04="SSI COUNTABLE EARNED INCOME &YR.04"  
EICM&YR.05="SSI COUNTABLE EARNED INCOME &YR.05"  
EICM&YR.06="SSI COUNTABLE EARNED INCOME &YR.06"  
EICM&YR.07="SSI COUNTABLE EARNED INCOME &YR.07"  
EICM&YR.08="SSI COUNTABLE EARNED INCOME &YR.08"  
EICM&YR.09="SSI COUNTABLE EARNED INCOME &YR.09"  
EICM&YR.10="SSI COUNTABLE EARNED INCOME &YR.10"  
EICM&YR.11="SSI COUNTABLE EARNED INCOME &YR.11"  
EICM&YR.12="SSI COUNTABLE EARNED INCOME &YR.12"

Appendix A.159

JCL/SAS Code: LABELS

EIND\_SSI&YR.01="SSI EMPLOYMENT INDICATOR &YR.01"  
EIND\_SSI&YR.02="SSI EMPLOYMENT INDICATOR &YR.02"  
EIND\_SSI&YR.03="SSI EMPLOYMENT INDICATOR &YR.03"  
EIND\_SSI&YR.04="SSI EMPLOYMENT INDICATOR &YR.04"  
EIND\_SSI&YR.05="SSI EMPLOYMENT INDICATOR &YR.05"  
EIND\_SSI&YR.06="SSI EMPLOYMENT INDICATOR &YR.06"  
EIND\_SSI&YR.07="SSI EMPLOYMENT INDICATOR &YR.07"  
EIND\_SSI&YR.08="SSI EMPLOYMENT INDICATOR &YR.08"  
EIND\_SSI&YR.09="SSI EMPLOYMENT INDICATOR &YR.09"  
EIND\_SSI&YR.10="SSI EMPLOYMENT INDICATOR &YR.10"  
EIND\_SSI&YR.11="SSI EMPLOYMENT INDICATOR &YR.11"  
EIND\_SSI&YR.12="SSI EMPLOYMENT INDICATOR &YR.12"  
FAMT&YR.01="SSI FEDERAL BENEFIT DUE &YR.01"  
FAMT&YR.02="SSI FEDERAL BENEFIT DUE &YR.02"  
FAMT&YR.03="SSI FEDERAL BENEFIT DUE &YR.03"  
FAMT&YR.04="SSI FEDERAL BENEFIT DUE &YR.04"  
FAMT&YR.05="SSI FEDERAL BENEFIT DUE &YR.05"  
FAMT&YR.06="SSI FEDERAL BENEFIT DUE &YR.06"  
FAMT&YR.07="SSI FEDERAL BENEFIT DUE &YR.07"  
FAMT&YR.08="SSI FEDERAL BENEFIT DUE &YR.08"  
FAMT&YR.09="SSI FEDERAL BENEFIT DUE &YR.09"  
FAMT&YR.10="SSI FEDERAL BENEFIT DUE &YR.10"  
FAMT&YR.11="SSI FEDERAL BENEFIT DUE &YR.11"  
FAMT&YR.12="SSI FEDERAL BENEFIT DUE &YR.12"  
FCI&YR.01="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.01"  
FCI&YR.02="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.02"  
FCI&YR.03="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.03"  
FCI&YR.04="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.04"  
FCI&YR.05="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.05"  
FCI&YR.06="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.06"  
FCI&YR.07="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.07"  
FCI&YR.08="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.08"  
FCI&YR.09="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.09"  
FCI&YR.10="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.10"  
FCI&YR.11="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.11"  
FCI&YR.12="FEDERAL COUNTABLE INCOME FOR CALCULATING FAMT IN &YR.12"  
FIPS\_BEST&YR.01="FIPS CODE, BEST &YR.01"  
FIPS\_BEST&YR.02="FIPS CODE, BEST &YR.02"  
FIPS\_BEST&YR.03="FIPS CODE, BEST &YR.03"  
FIPS\_BEST&YR.04="FIPS CODE, BEST &YR.04"  
FIPS\_BEST&YR.05="FIPS CODE, BEST &YR.05"  
FIPS\_BEST&YR.06="FIPS CODE, BEST &YR.06"  
FIPS\_BEST&YR.07="FIPS CODE, BEST &YR.07"  
FIPS\_BEST&YR.08="FIPS CODE, BEST &YR.08"  
FIPS\_BEST&YR.09="FIPS CODE, BEST &YR.09"

Appendix A.159

JCL/SAS Code: LABELS

```
FIPS_BEST&YR.10="FIPS CODE, BEST &YR.10"  
FIPS_BEST&YR.11="FIPS CODE, BEST &YR.11"  
FIPS_BEST&YR.12="FIPS CODE, BEST &YR.12"  
FIPS_DI&YR.01="SSDI FIPS CODE &YR.01"  
FIPS_DI&YR.02="SSDI FIPS CODE &YR.02"  
FIPS_DI&YR.03="SSDI FIPS CODE &YR.03"  
FIPS_DI&YR.04="SSDI FIPS CODE &YR.04"  
FIPS_DI&YR.05="SSDI FIPS CODE &YR.05"  
FIPS_DI&YR.06="SSDI FIPS CODE &YR.06"  
FIPS_DI&YR.07="SSDI FIPS CODE &YR.07"  
FIPS_DI&YR.08="SSDI FIPS CODE &YR.08"  
FIPS_DI&YR.09="SSDI FIPS CODE &YR.09"  
FIPS_DI&YR.10="SSDI FIPS CODE &YR.10"  
FIPS_DI&YR.11="SSDI FIPS CODE &YR.11"  
FIPS_DI&YR.12="SSDI FIPS CODE &YR.12"  
FIPS_SSI&YR.01="SSI FIPS CODE &YR.01"  
FIPS_SSI&YR.02="SSI FIPS CODE &YR.02"  
FIPS_SSI&YR.03="SSI FIPS CODE &YR.03"  
FIPS_SSI&YR.04="SSI FIPS CODE &YR.04"  
FIPS_SSI&YR.05="SSI FIPS CODE &YR.05"  
FIPS_SSI&YR.06="SSI FIPS CODE &YR.06"  
FIPS_SSI&YR.07="SSI FIPS CODE &YR.07"  
FIPS_SSI&YR.08="SSI FIPS CODE &YR.08"  
FIPS_SSI&YR.09="SSI FIPS CODE &YR.09"  
FIPS_SSI&YR.10="SSI FIPS CODE &YR.10"  
FIPS_SSI&YR.11="SSI FIPS CODE &YR.11"  
FIPS_SSI&YR.12="SSI FIPS CODE &YR.12"  
FPMT&YR.01="SSI FEDERAL BENEFIT PAID AMOUNT &YR.01"  
FPMT&YR.02="SSI FEDERAL BENEFIT PAID AMOUNT &YR.02"  
FPMT&YR.03="SSI FEDERAL BENEFIT PAID AMOUNT &YR.03"  
FPMT&YR.04="SSI FEDERAL BENEFIT PAID AMOUNT &YR.04"  
FPMT&YR.05="SSI FEDERAL BENEFIT PAID AMOUNT &YR.05"  
FPMT&YR.06="SSI FEDERAL BENEFIT PAID AMOUNT &YR.06"  
FPMT&YR.07="SSI FEDERAL BENEFIT PAID AMOUNT &YR.07"  
FPMT&YR.08="SSI FEDERAL BENEFIT PAID AMOUNT &YR.08"  
FPMT&YR.09="SSI FEDERAL BENEFIT PAID AMOUNT &YR.09"  
FPMT&YR.10="SSI FEDERAL BENEFIT PAID AMOUNT &YR.10"  
FPMT&YR.11="SSI FEDERAL BENEFIT PAID AMOUNT &YR.11"  
FPMT&YR.12="SSI FEDERAL BENEFIT PAID AMOUNT &YR.12"  
FRADATE = "FULL RETIREMENT DATE"  
IEA_BLIND&YR.01=  
"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.01"  
IEA_BLIND&YR.02=  
"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.02"  
IEA_BLIND&YR.03=
```

Appendix A.159

JCL/SAS Code: LABELS

"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.03"  
IEA\_BLIND&YR.04=  
"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.04"  
IEA\_BLIND&YR.05=  
"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.05"  
IEA\_BLIND&YR.06=  
"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.06"  
IEA\_BLIND&YR.07=  
"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.07"  
IEA\_BLIND&YR.08=  
"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.08"  
IEA\_BLIND&YR.09=  
"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.09"  
IEA\_BLIND&YR.10=  
"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.10"  
IEA\_BLIND&YR.11=  
"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.11"  
IEA\_BLIND&YR.12=  
"SSI EARNED INCOME EXCLUSION AMOUNT - BLIND WORK EXPENSE &YR.12"  
IEA\_SEIE&YR.01=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.01"  
IEA\_SEIE&YR.02=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.02"  
IEA\_SEIE&YR.03=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.03"  
IEA\_SEIE&YR.04=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.04"  
IEA\_SEIE&YR.05=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.05"  
IEA\_SEIE&YR.06=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.06"  
IEA\_SEIE&YR.07=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.07"  
IEA\_SEIE&YR.08=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.08"  
IEA\_SEIE&YR.09=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.09"  
IEA\_SEIE&YR.10=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.10"  
IEA\_SEIE&YR.11=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.11"  
IEA\_SEIE&YR.12=  
"SSI EARNED INCOME EXCLUSION AMOUNT - STUDENT EARNED INCOME &YR.12"  
IEA\_IRWE&YR.01="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.01"  
IEA\_IRWE&YR.02="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.02"



Appendix A.159

JCL/SAS Code: LABELS

IEA\_IRWE&YR.03="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.03"  
IEA\_IRWE&YR.04="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.04"  
IEA\_IRWE&YR.05="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.05"  
IEA\_IRWE&YR.06="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.06"  
IEA\_IRWE&YR.07="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.07"  
IEA\_IRWE&YR.08="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.08"  
IEA\_IRWE&YR.09="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.09"  
IEA\_IRWE&YR.10="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.10"  
IEA\_IRWE&YR.11="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.11"  
IEA\_IRWE&YR.12="SSI EARNED INCOME EXCLUSION AMOUNT - IRWE &YR.12"  
IEA\_LOSS&YR.01=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.01"  
IEA\_LOSS&YR.02=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.02"  
IEA\_LOSS&YR.03=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.03"  
IEA\_LOSS&YR.04=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.04"  
IEA\_LOSS&YR.05=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.05"  
IEA\_LOSS&YR.06=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.06"  
IEA\_LOSS&YR.07=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.07"  
IEA\_LOSS&YR.08=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.08"  
IEA\_LOSS&YR.09=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.09"  
IEA\_LOSS&YR.10=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.10"  
IEA\_LOSS&YR.11=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.11"  
IEA\_LOSS&YR.12=  
"SSI EARNED INCOME EXCLUSION AMOUNT - NET LOSS FROM SELF-EMP &YR.12"  
IEA\_NESE&YR.01="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.01"  
IEA\_NESE&YR.02="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.02"  
IEA\_NESE&YR.03="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.03"  
IEA\_NESE&YR.04="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.04"  
IEA\_NESE&YR.05="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.05"  
IEA\_NESE&YR.06="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.06"  
IEA\_NESE&YR.07="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.07"  
IEA\_NESE&YR.08="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.08"  
IEA\_NESE&YR.09="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.09"  
IEA\_NESE&YR.10="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.10"  
IEA\_NESE&YR.11="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.11"

Appendix A.159

JCL/SAS Code: LABELS

IEA\_NESE&YR.12="SSI EARNED INCOME AMOUNT - FROM SELF-EMP &YR.12"  
IEA\_PASS&YR.01="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.01"  
IEA\_PASS&YR.02="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.02"  
IEA\_PASS&YR.03="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.03"  
IEA\_PASS&YR.04="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.04"  
IEA\_PASS&YR.05="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.05"  
IEA\_PASS&YR.06="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.06"  
IEA\_PASS&YR.07="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.07"  
IEA\_PASS&YR.08="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.08"  
IEA\_PASS&YR.09="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.09"  
IEA\_PASS&YR.10="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.10"  
IEA\_PASS&YR.11="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.11"  
IEA\_PASS&YR.12="SSI EARNED INCOME EXCLUSION AMOUNT - PASS &YR.12"  
IEA\_WAGE&YR.01="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.01"  
IEA\_WAGE&YR.02="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.02"  
IEA\_WAGE&YR.03="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.03"  
IEA\_WAGE&YR.04="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.04"  
IEA\_WAGE&YR.05="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.05"  
IEA\_WAGE&YR.06="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.06"  
IEA\_WAGE&YR.07="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.07"  
IEA\_WAGE&YR.08="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.08"  
IEA\_WAGE&YR.09="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.09"  
IEA\_WAGE&YR.10="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.10"  
IEA\_WAGE&YR.11="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.11"  
IEA\_WAGE&YR.12="SSI EARNED INCOME AMOUNT - FROM WAGES &YR.12"  
IUA\_SSDI&YR.01="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.01"  
IUA\_SSDI&YR.02="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.02"  
IUA\_SSDI&YR.03="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.03"  
IUA\_SSDI&YR.04="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.04"  
IUA\_SSDI&YR.05="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.05"  
IUA\_SSDI&YR.06="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.06"  
IUA\_SSDI&YR.07="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.07"  
IUA\_SSDI&YR.08="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.08"  
IUA\_SSDI&YR.09="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.09"  
IUA\_SSDI&YR.10="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.10"  
IUA\_SSDI&YR.11="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.11"  
IUA\_SSDI&YR.12="SSI UNEARNED INCOME - FROM SSDI BENEFIT &YR.12"  
IUA\_WC&YR.01="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.01"  
IUA\_WC&YR.02="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.02"  
IUA\_WC&YR.03="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.03"  
IUA\_WC&YR.04="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.04"  
IUA\_WC&YR.05="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.05"  
IUA\_WC&YR.06="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.06"  
IUA\_WC&YR.07="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.07"  
IUA\_WC&YR.08="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.08"

Appendix A.159

JCL/SAS Code: LABELS

IUA\_WC&YR.09="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.09"  
IUA\_WC&YR.10="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.10"  
IUA\_WC&YR.11="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.11"  
IUA\_WC&YR.12="SSI UNEARNED INCOME - FROM WORKERS' COMPENSATION &YR.12"  
IUA\_TANF&YR.01="SSI UNEARNED INCOME - FROM TANF &YR.01"  
IUA\_TANF&YR.02="SSI UNEARNED INCOME - FROM TANF &YR.02"  
IUA\_TANF&YR.03="SSI UNEARNED INCOME - FROM TANF &YR.03"  
IUA\_TANF&YR.04="SSI UNEARNED INCOME - FROM TANF &YR.04"  
IUA\_TANF&YR.05="SSI UNEARNED INCOME - FROM TANF &YR.05"  
IUA\_TANF&YR.06="SSI UNEARNED INCOME - FROM TANF &YR.06"  
IUA\_TANF&YR.07="SSI UNEARNED INCOME - FROM TANF &YR.07"  
IUA\_TANF&YR.08="SSI UNEARNED INCOME - FROM TANF &YR.08"  
IUA\_TANF&YR.09="SSI UNEARNED INCOME - FROM TANF &YR.09"  
IUA\_TANF&YR.10="SSI UNEARNED INCOME - FROM TANF &YR.10"  
IUA\_TANF&YR.11="SSI UNEARNED INCOME - FROM TANF &YR.11"  
IUA\_TANF&YR.12="SSI UNEARNED INCOME - FROM TANF &YR.12"  
LAF&YR.01="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.01"  
LAF&YR.02="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.02"  
LAF&YR.03="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.03"  
LAF&YR.04="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.04"  
LAF&YR.05="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.05"  
LAF&YR.06="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.06"  
LAF&YR.07="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.07"  
LAF&YR.08="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.08"  
LAF&YR.09="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.09"  
LAF&YR.10="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.10"  
LAF&YR.11="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.11"  
LAF&YR.12="SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS &YR.12"  
LIVF&YR.01="SSI LIVING ARRANGEMENT CODE &YR.01"  
LIVF&YR.02="SSI LIVING ARRANGEMENT CODE &YR.02"  
LIVF&YR.03="SSI LIVING ARRANGEMENT CODE &YR.03"  
LIVF&YR.04="SSI LIVING ARRANGEMENT CODE &YR.04"  
LIVF&YR.05="SSI LIVING ARRANGEMENT CODE &YR.05"  
LIVF&YR.06="SSI LIVING ARRANGEMENT CODE &YR.06"  
LIVF&YR.07="SSI LIVING ARRANGEMENT CODE &YR.07"  
LIVF&YR.08="SSI LIVING ARRANGEMENT CODE &YR.08"  
LIVF&YR.09="SSI LIVING ARRANGEMENT CODE &YR.09"  
LIVF&YR.10="SSI LIVING ARRANGEMENT CODE &YR.10"  
LIVF&YR.11="SSI LIVING ARRANGEMENT CODE &YR.11"  
LIVF&YR.12="SSI LIVING ARRANGEMENT CODE &YR.12"  
MBA&YR.01="SSDI BENEFIT DUE &YR.01"  
MBA&YR.02="SSDI BENEFIT DUE &YR.02"  
MBA&YR.03="SSDI BENEFIT DUE &YR.03"  
MBA&YR.04="SSDI BENEFIT DUE &YR.04"  
MBA&YR.05="SSDI BENEFIT DUE &YR.05"

Appendix A.159

JCL/SAS Code: LABELS

MBA&YR.06="SSDI BENEFIT DUE &YR.06"  
MBA&YR.07="SSDI BENEFIT DUE &YR.07"  
MBA&YR.08="SSDI BENEFIT DUE &YR.08"  
MBA&YR.09="SSDI BENEFIT DUE &YR.09"  
MBA&YR.10="SSDI BENEFIT DUE &YR.10"  
MBA&YR.11="SSDI BENEFIT DUE &YR.11"  
MBA&YR.12="SSDI BENEFIT DUE &YR.12"  
MBP&YR.01="SSDI BENEFIT PAID &YR.01"  
MBP&YR.02="SSDI BENEFIT PAID &YR.02"  
MBP&YR.03="SSDI BENEFIT PAID &YR.03"  
MBP&YR.04="SSDI BENEFIT PAID &YR.04"  
MBP&YR.05="SSDI BENEFIT PAID &YR.05"  
MBP&YR.06="SSDI BENEFIT PAID &YR.06"  
MBP&YR.07="SSDI BENEFIT PAID &YR.07"  
MBP&YR.08="SSDI BENEFIT PAID &YR.08"  
MBP&YR.09="SSDI BENEFIT PAID &YR.09"  
MBP&YR.10="SSDI BENEFIT PAID &YR.10"  
MBP&YR.11="SSDI BENEFIT PAID &YR.11"  
MBP&YR.12="SSDI BENEFIT PAID &YR.12"  
MEDC&YR.01="SSI MEDICAID ELIGIBILITY CODE &YR.01"  
MEDC&YR.02="SSI MEDICAID ELIGIBILITY CODE &YR.02"  
MEDC&YR.03="SSI MEDICAID ELIGIBILITY CODE &YR.03"  
MEDC&YR.04="SSI MEDICAID ELIGIBILITY CODE &YR.04"  
MEDC&YR.05="SSI MEDICAID ELIGIBILITY CODE &YR.05"  
MEDC&YR.06="SSI MEDICAID ELIGIBILITY CODE &YR.06"  
MEDC&YR.07="SSI MEDICAID ELIGIBILITY CODE &YR.07"  
MEDC&YR.08="SSI MEDICAID ELIGIBILITY CODE &YR.08"  
MEDC&YR.09="SSI MEDICAID ELIGIBILITY CODE &YR.09"  
MEDC&YR.10="SSI MEDICAID ELIGIBILITY CODE &YR.10"  
MEDC&YR.11="SSI MEDICAID ELIGIBILITY CODE &YR.11"  
MEDC&YR.12="SSI MEDICAID ELIGIBILITY CODE &YR.12"  
MEDE&YR.01="SSI MEDICAID ELIGIBILITY &YR.01"  
MEDE&YR.02="SSI MEDICAID ELIGIBILITY &YR.02"  
MEDE&YR.03="SSI MEDICAID ELIGIBILITY &YR.03"  
MEDE&YR.04="SSI MEDICAID ELIGIBILITY &YR.04"  
MEDE&YR.05="SSI MEDICAID ELIGIBILITY &YR.05"  
MEDE&YR.06="SSI MEDICAID ELIGIBILITY &YR.06"  
MEDE&YR.07="SSI MEDICAID ELIGIBILITY &YR.07"  
MEDE&YR.08="SSI MEDICAID ELIGIBILITY &YR.08"  
MEDE&YR.09="SSI MEDICAID ELIGIBILITY &YR.09"  
MEDE&YR.10="SSI MEDICAID ELIGIBILITY &YR.10"  
MEDE&YR.11="SSI MEDICAID ELIGIBILITY &YR.11"  
MEDE&YR.12="SSI MEDICAID ELIGIBILITY &YR.12"  
MEDX&YR.01="MEDICAL IMPROVEMENT CODE &YR.01"  
MEDX&YR.02="MEDICAL IMPROVEMENT CODE &YR.02"

Appendix A.159

JCL/SAS Code: LABELS

MEDX&YR.03="MEDICAL IMPROVEMENT CODE &YR.03"  
MEDX&YR.04="MEDICAL IMPROVEMENT CODE &YR.04"  
MEDX&YR.05="MEDICAL IMPROVEMENT CODE &YR.05"  
MEDX&YR.06="MEDICAL IMPROVEMENT CODE &YR.06"  
MEDX&YR.07="MEDICAL IMPROVEMENT CODE &YR.07"  
MEDX&YR.08="MEDICAL IMPROVEMENT CODE &YR.08"  
MEDX&YR.09="MEDICAL IMPROVEMENT CODE &YR.09"  
MEDX&YR.10="MEDICAL IMPROVEMENT CODE &YR.10"  
MEDX&YR.11="MEDICAL IMPROVEMENT CODE &YR.11"  
MEDX&YR.12="MEDICAL IMPROVEMENT CODE &YR.12"  
MEDPREM&YR.01="SSDI PHUS MEDICARE PREMIUM &YR.01"  
MEDPREM&YR.02="SSDI PHUS MEDICARE PREMIUM &YR.02"  
MEDPREM&YR.03="SSDI PHUS MEDICARE PREMIUM &YR.03"  
MEDPREM&YR.04="SSDI PHUS MEDICARE PREMIUM &YR.04"  
MEDPREM&YR.05="SSDI PHUS MEDICARE PREMIUM &YR.05"  
MEDPREM&YR.06="SSDI PHUS MEDICARE PREMIUM &YR.06"  
MEDPREM&YR.07="SSDI PHUS MEDICARE PREMIUM &YR.07"  
MEDPREM&YR.08="SSDI PHUS MEDICARE PREMIUM &YR.08"  
MEDPREM&YR.09="SSDI PHUS MEDICARE PREMIUM &YR.09"  
MEDPREM&YR.10="SSDI PHUS MEDICARE PREMIUM &YR.10"  
MEDPREM&YR.11="SSDI PHUS MEDICARE PREMIUM &YR.11"  
MEDPREM&YR.12="SSDI PHUS MEDICARE PREMIUM &YR.12"  
MEDR&YR.01="SSDI MEDICARE ELIGIBILITY &YR.01"  
MEDR&YR.02="SSDI MEDICARE ELIGIBILITY &YR.02"  
MEDR&YR.03="SSDI MEDICARE ELIGIBILITY &YR.03"  
MEDR&YR.04="SSDI MEDICARE ELIGIBILITY &YR.04"  
MEDR&YR.05="SSDI MEDICARE ELIGIBILITY &YR.05"  
MEDR&YR.06="SSDI MEDICARE ELIGIBILITY &YR.06"  
MEDR&YR.07="SSDI MEDICARE ELIGIBILITY &YR.07"  
MEDR&YR.08="SSDI MEDICARE ELIGIBILITY &YR.08"  
MEDR&YR.09="SSDI MEDICARE ELIGIBILITY &YR.09"  
MEDR&YR.10="SSDI MEDICARE ELIGIBILITY &YR.10"  
MEDR&YR.11="SSDI MEDICARE ELIGIBILITY &YR.11"  
MEDR&YR.12="SSDI MEDICARE ELIGIBILITY &YR.12"  
MTST&YR.01="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.01"  
MTST&YR.02="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.02"  
MTST&YR.03="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.03"  
MTST&YR.04="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.04"  
MTST&YR.05="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.05"  
MTST&YR.06="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.06"  
MTST&YR.07="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.07"  
MTST&YR.08="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.08"  
MTST&YR.09="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.09"  
MTST&YR.10="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.10"  
MTST&YR.11="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.11"

Appendix A.159

JCL/SAS Code: LABELS

MTST&YR.12="SSI MEDICAL AND SOCIAL SERVICE INCOME TEST &YR.12"  
NECALC&YR.01=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.01"  
NECALC&YR.02=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.02"  
NECALC&YR.03=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.03"  
NECALC&YR.04=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.04"  
NECALC&YR.05=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.05"  
NECALC&YR.06=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.06"  
NECALC&YR.07=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.07"  
NECALC&YR.08=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.08"  
NECALC&YR.09=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.09"  
NECALC&YR.10=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.10"  
NECALC&YR.11=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.12"  
NECALC&YR.12=  
"FLAG INDICATING DIRECT CALUCLATION OF NON-EARNED INCOME IN &YR.12"  
NONEARN&YR.01="DEEMED AND ISM INCOME (CALCULATED) IN &YR.01"  
NONEARN&YR.02="DEEMED AND ISM INCOME (CALCULATED) IN &YR.02"  
NONEARN&YR.03="DEEMED AND ISM INCOME (CALCULATED) IN &YR.03"  
NONEARN&YR.04="DEEMED AND ISM INCOME (CALCULATED) IN &YR.04"  
NONEARN&YR.05="DEEMED AND ISM INCOME (CALCULATED) IN &YR.05"  
NONEARN&YR.06="DEEMED AND ISM INCOME (CALCULATED) IN &YR.06"  
NONEARN&YR.07="DEEMED AND ISM INCOME (CALCULATED) IN &YR.07"  
NONEARN&YR.08="DEEMED AND ISM INCOME (CALCULATED) IN &YR.08"  
NONEARN&YR.09="DEEMED AND ISM INCOME (CALCULATED) IN &YR.09"  
NONEARN&YR.10="DEEMED AND ISM INCOME (CALCULATED) IN &YR.10"  
NONEARN&YR.11="DEEMED AND ISM INCOME (CALCULATED) IN &YR.11"  
NONEARN&YR.12="DEEMED AND ISM INCOME (CALCULATED) IN &YR.12"  
PAYD&YR.01="SSDI PHUS BENEFIT PAID &YR.01"  
PAYD&YR.02="SSDI PHUS BENEFIT PAID &YR.02"  
PAYD&YR.03="SSDI PHUS BENEFIT PAID &YR.03"  
PAYD&YR.04="SSDI PHUS BENEFIT PAID &YR.04"  
PAYD&YR.05="SSDI PHUS BENEFIT PAID &YR.05"  
PAYD&YR.06="SSDI PHUS BENEFIT PAID &YR.06"  
PAYD&YR.07="SSDI PHUS BENEFIT PAID &YR.07"  
PAYD&YR.08="SSDI PHUS BENEFIT PAID &YR.08"

Appendix A.159

JCL/SAS Code: LABELS

PAYD&YR.09="SSDI PHUS BENEFIT PAID &YR.09"  
PAYD&YR.10="SSDI PHUS BENEFIT PAID &YR.10"  
PAYD&YR.11="SSDI PHUS BENEFIT PAID &YR.11"  
PAYD&YR.12="SSDI PHUS BENEFIT PAID &YR.12"  
PAYO&YR.01="SSDI PHUS DEPENDENT BENEFIT PAID &YR.01"  
PAYO&YR.02="SSDI PHUS DEPENDENT BENEFIT PAID &YR.02"  
PAYO&YR.03="SSDI PHUS DEPENDENT BENEFIT PAID &YR.03"  
PAYO&YR.04="SSDI PHUS DEPENDENT BENEFIT PAID &YR.04"  
PAYO&YR.05="SSDI PHUS DEPENDENT BENEFIT PAID &YR.05"  
PAYO&YR.06="SSDI PHUS DEPENDENT BENEFIT PAID &YR.06"  
PAYO&YR.07="SSDI PHUS DEPENDENT BENEFIT PAID &YR.07"  
PAYO&YR.08="SSDI PHUS DEPENDENT BENEFIT PAID &YR.08"  
PAYO&YR.09="SSDI PHUS DEPENDENT BENEFIT PAID &YR.09"  
PAYO&YR.10="SSDI PHUS DEPENDENT BENEFIT PAID &YR.10"  
PAYO&YR.11="SSDI PHUS DEPENDENT BENEFIT PAID &YR.11"  
PAYO&YR.12="SSDI PHUS DEPENDENT BENEFIT PAID &YR.12"  
PAYS&YR.01="SSI BENEFIT PAID &YR.01"  
PAYS&YR.02="SSI BENEFIT PAID &YR.02"  
PAYS&YR.03="SSI BENEFIT PAID &YR.03"  
PAYS&YR.04="SSI BENEFIT PAID &YR.04"  
PAYS&YR.05="SSI BENEFIT PAID &YR.05"  
PAYS&YR.06="SSI BENEFIT PAID &YR.06"  
PAYS&YR.07="SSI BENEFIT PAID &YR.07"  
PAYS&YR.08="SSI BENEFIT PAID &YR.08"  
PAYS&YR.09="SSI BENEFIT PAID &YR.09"  
PAYS&YR.10="SSI BENEFIT PAID &YR.10"  
PAYS&YR.11="SSI BENEFIT PAID &YR.11"  
PAYS&YR.12="SSI BENEFIT PAID &YR.12"  
PROA&YR.01="SSI USE OF 1619A PROVISION INDICATOR &YR.01"  
PROA&YR.02="SSI USE OF 1619A PROVISION INDICATOR &YR.02"  
PROA&YR.03="SSI USE OF 1619A PROVISION INDICATOR &YR.03"  
PROA&YR.04="SSI USE OF 1619A PROVISION INDICATOR &YR.04"  
PROA&YR.05="SSI USE OF 1619A PROVISION INDICATOR &YR.05"  
PROA&YR.06="SSI USE OF 1619A PROVISION INDICATOR &YR.06"  
PROA&YR.07="SSI USE OF 1619A PROVISION INDICATOR &YR.07"  
PROA&YR.08="SSI USE OF 1619A PROVISION INDICATOR &YR.08"  
PROA&YR.09="SSI USE OF 1619A PROVISION INDICATOR &YR.09"  
PROA&YR.10="SSI USE OF 1619A PROVISION INDICATOR &YR.10"  
PROA&YR.11="SSI USE OF 1619A PROVISION INDICATOR &YR.11"  
PROA&YR.12="SSI USE OF 1619A PROVISION INDICATOR &YR.12"  
PROB&YR.01="SSI USE OF 1619B PROVISION INDICATOR &YR.01"  
PROB&YR.02="SSI USE OF 1619B PROVISION INDICATOR &YR.02"  
PROB&YR.03="SSI USE OF 1619B PROVISION INDICATOR &YR.03"  
PROB&YR.04="SSI USE OF 1619B PROVISION INDICATOR &YR.04"  
PROB&YR.05="SSI USE OF 1619B PROVISION INDICATOR &YR.05"

Appendix A.159

JCL/SAS Code: LABELS

PROB&YR.06="SSI USE OF 1619B PROVISION INDICATOR &YR.06"  
PROB&YR.07="SSI USE OF 1619B PROVISION INDICATOR &YR.07"  
PROB&YR.08="SSI USE OF 1619B PROVISION INDICATOR &YR.08"  
PROB&YR.09="SSI USE OF 1619B PROVISION INDICATOR &YR.09"  
PROB&YR.10="SSI USE OF 1619B PROVISION INDICATOR &YR.10"  
PROB&YR.11="SSI USE OF 1619B PROVISION INDICATOR &YR.11"  
PROB&YR.12="SSI USE OF 1619B PROVISION INDICATOR &YR.12"  
PSTA&YR.01="SSI PAYMENT STATUS (PSTAT) &YR.01"  
PSTA&YR.02="SSI PAYMENT STATUS (PSTAT) &YR.02"  
PSTA&YR.03="SSI PAYMENT STATUS (PSTAT) &YR.03"  
PSTA&YR.04="SSI PAYMENT STATUS (PSTAT) &YR.04"  
PSTA&YR.05="SSI PAYMENT STATUS (PSTAT) &YR.05"  
PSTA&YR.06="SSI PAYMENT STATUS (PSTAT) &YR.06"  
PSTA&YR.07="SSI PAYMENT STATUS (PSTAT) &YR.07"  
PSTA&YR.08="SSI PAYMENT STATUS (PSTAT) &YR.08"  
PSTA&YR.09="SSI PAYMENT STATUS (PSTAT) &YR.09"  
PSTA&YR.10="SSI PAYMENT STATUS (PSTAT) &YR.10"  
PSTA&YR.11="SSI PAYMENT STATUS (PSTAT) &YR.11"  
PSTA&YR.12="SSI PAYMENT STATUS (PSTAT) &YR.12"  
RFD&YR.01="SSDI REASON FOR DEDUCTION CODE &YR.01"  
RFD&YR.02="SSDI REASON FOR DEDUCTION CODE &YR.02"  
RFD&YR.03="SSDI REASON FOR DEDUCTION CODE &YR.03"  
RFD&YR.04="SSDI REASON FOR DEDUCTION CODE &YR.04"  
RFD&YR.05="SSDI REASON FOR DEDUCTION CODE &YR.05"  
RFD&YR.06="SSDI REASON FOR DEDUCTION CODE &YR.06"  
RFD&YR.07="SSDI REASON FOR DEDUCTION CODE &YR.07"  
RFD&YR.08="SSDI REASON FOR DEDUCTION CODE &YR.08"  
RFD&YR.09="SSDI REASON FOR DEDUCTION CODE &YR.09"  
RFD&YR.10="SSDI REASON FOR DEDUCTION CODE &YR.10"  
RFD&YR.11="SSDI REASON FOR DEDUCTION CODE &YR.11"  
RFD&YR.12="SSDI REASON FOR DEDUCTION CODE &YR.12"  
RFST&YR.01="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.01"  
RFST&YR.02="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.02"  
RFST&YR.03="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.03"  
RFST&YR.04="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.04"  
RFST&YR.05="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.05"  
RFST&YR.06="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.06"  
RFST&YR.07="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.07"  
RFST&YR.08="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.08"  
RFST&YR.09="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.09"  
RFST&YR.10="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.10"  
RFST&YR.11="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.11"  
RFST&YR.12="SSDI REASON FOR SUSPENSION OR TERMINATION CODE &YR.12"  
SAMT&YR.01="SSI STATE BENEFIT DUE &YR.01"  
SAMT&YR.02="SSI STATE BENEFIT DUE &YR.02"



Appendix A.159

JCL/SAS Code: LABELS

SAMT&YR.03="SSI STATE BENEFIT DUE &YR.03"  
SAMT&YR.04="SSI STATE BENEFIT DUE &YR.04"  
SAMT&YR.05="SSI STATE BENEFIT DUE &YR.05"  
SAMT&YR.06="SSI STATE BENEFIT DUE &YR.06"  
SAMT&YR.07="SSI STATE BENEFIT DUE &YR.07"  
SAMT&YR.08="SSI STATE BENEFIT DUE &YR.08"  
SAMT&YR.09="SSI STATE BENEFIT DUE &YR.09"  
SAMT&YR.10="SSI STATE BENEFIT DUE &YR.10"  
SAMT&YR.11="SSI STATE BENEFIT DUE &YR.11"  
SAMT&YR.12="SSI STATE BENEFIT DUE &YR.12"  
SCON&YR.01="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.01"  
SCON&YR.02="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.02"  
SCON&YR.03="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.03"  
SCON&YR.04="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.04"  
SCON&YR.05="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.05"  
SCON&YR.06="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.06"  
SCON&YR.07="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.07"  
SCON&YR.08="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.08"  
SCON&YR.09="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.09"  
SCON&YR.10="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.10"  
SCON&YR.11="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.11"  
SCON&YR.12="SSI STATE CONCURRENT ELIGIBILITY CODE &YR.12"  
SEX="SEX"  
SSIERNPGA&YR.01="SSI SGA FLAG &YR.01"  
SSIERNPGA&YR.02="SSI SGA FLAG &YR.02"  
SSIERNPGA&YR.03="SSI SGA FLAG &YR.03"  
SSIERNPGA&YR.04="SSI SGA FLAG &YR.04"  
SSIERNPGA&YR.05="SSI SGA FLAG &YR.05"  
SSIERNPGA&YR.06="SSI SGA FLAG &YR.06"  
SSIERNPGA&YR.07="SSI SGA FLAG &YR.07"  
SSIERNPGA&YR.08="SSI SGA FLAG &YR.08"  
SSIERNPGA&YR.09="SSI SGA FLAG &YR.09"  
SSIERNPGA&YR.10="SSI SGA FLAG &YR.10"  
SSIERNPGA&YR.11="SSI SGA FLAG &YR.11"  
SSIERNPGA&YR.12="SSI SGA FLAG &YR.12"  
SIND&YR.01="SSI SELF-EMP INDICATOR &YR.01"  
SIND&YR.02="SSI SELF-EMP INDICATOR &YR.02"  
SIND&YR.03="SSI SELF-EMP INDICATOR &YR.03"  
SIND&YR.04="SSI SELF-EMP INDICATOR &YR.04"  
SIND&YR.05="SSI SELF-EMP INDICATOR &YR.05"  
SIND&YR.06="SSI SELF-EMP INDICATOR &YR.06"  
SIND&YR.07="SSI SELF-EMP INDICATOR &YR.07"  
SIND&YR.08="SSI SELF-EMP INDICATOR &YR.08"  
SIND&YR.09="SSI SELF-EMP INDICATOR &YR.09"  
SIND&YR.10="SSI SELF-EMP INDICATOR &YR.10"

Appendix A.159

JCL/SAS Code: LABELS

SIND&YR.11="SSI SELF-EMP INDICATOR &YR.11"  
SIND&YR.12="SSI SELF-EMP INDICATOR &YR.12"  
SPMT&YR.01="SSI STATE BENEFIT PAID &YR.01"  
SPMT&YR.02="SSI STATE BENEFIT PAID &YR.02"  
SPMT&YR.03="SSI STATE BENEFIT PAID &YR.03"  
SPMT&YR.04="SSI STATE BENEFIT PAID &YR.04"  
SPMT&YR.05="SSI STATE BENEFIT PAID &YR.05"  
SPMT&YR.06="SSI STATE BENEFIT PAID &YR.06"  
SPMT&YR.07="SSI STATE BENEFIT PAID &YR.07"  
SPMT&YR.08="SSI STATE BENEFIT PAID &YR.08"  
SPMT&YR.09="SSI STATE BENEFIT PAID &YR.09"  
SPMT&YR.10="SSI STATE BENEFIT PAID &YR.10"  
SPMT&YR.11="SSI STATE BENEFIT PAID &YR.11"  
SPMT&YR.12="SSI STATE BENEFIT PAID &YR.12"  
SSDI\_ONSET="SSDI DISABILITY ONSET DATE EARLIEST PERIOD OF ELIGIBILITY"  
SSDI94="SSDI CURRENT PAY STATUS IN 1994"  
SSDI95="SSDI CURRENT PAY STATUS IN 1995"  
SSDI96="SSDI CURRENT PAY STATUS IN 1996"  
SSDI97="SSDI CURRENT PAY STATUS IN 1997"  
SSDI98="SSDI CURRENT PAY STATUS IN 1998"  
SSDI99="SSDI CURRENT PAY STATUS IN 1999"  
SSDI00="SSDI CURRENT PAY STATUS IN 2000"  
SSDI01="SSDI CURRENT PAY STATUS IN 2001"  
SSDI02="SSDI CURRENT PAY STATUS IN 2002"  
SSDI03="SSDI CURRENT PAY STATUS IN 2003"  
SSDI04="SSDI CURRENT PAY STATUS IN 2004"  
SSDI05="SSDI CURRENT PAY STATUS IN 2005"  
SSDI06="SSDI CURRENT PAY STATUS IN 2006"  
SSDI07="SSDI CURRENT PAY STATUS IN 2007"  
SSDI08="SSDI CURRENT PAY STATUS IN 2008"  
SSDI09="SSDI CURRENT PAY STATUS IN 2009"  
SSDI10="SSDI CURRENT PAY STATUS IN 2010"  
SSDI11="SSDI CURRENT PAY STATUS IN 2011"  
SSDI12="SSDI CURRENT PAY STATUS IN 2012"  
SSDI13="SSDI CURRENT PAY STATUS IN 2013"  
SSDI14="SSDI CURRENT PAY STATUS IN 2014"  
SSDI15="SSDI CURRENT PAY STATUS IN 2015"  
SSDI16="SSDI CURRENT PAY STATUS IN 2016"  
SSDI17="SSDI CURRENT PAY STATUS IN 2017"  
SSDI18="SSDI CURRENT PAY STATUS IN 2018"  
SSDI19="SSDI CURRENT PAY STATUS IN 2019"  
SSDI20="SSDI CURRENT PAY STATUS IN 2020"  
SSDI21="SSDI CURRENT PAY STATUS IN 2021"  
SSI94="SSI CURRENT PAY STATUS IN 1994"  
SSI95="SSI CURRENT PAY STATUS IN 1995"

Appendix A.159

JCL/SAS Code: LABELS

SSI96="SSI CURRENT PAY STATUS IN 1996"  
SSI97="SSI CURRENT PAY STATUS IN 1997"  
SSI98="SSI CURRENT PAY STATUS IN 1998"  
SSI99="SSI CURRENT PAY STATUS IN 1999"  
SSI00="SSI CURRENT PAY STATUS IN 2000"  
SSI01="SSI CURRENT PAY STATUS IN 2001"  
SSI02="SSI CURRENT PAY STATUS IN 2002"  
SSI03="SSI CURRENT PAY STATUS IN 2003"  
SSI04="SSI CURRENT PAY STATUS IN 2004"  
SSI05="SSI CURRENT PAY STATUS IN 2005"  
SSI06="SSI CURRENT PAY STATUS IN 2006"  
SSI07="SSI CURRENT PAY STATUS IN 2007"  
SSI08="SSI CURRENT PAY STATUS IN 2008"  
SSI09="SSI CURRENT PAY STATUS IN 2009"  
SSI10="SSI CURRENT PAY STATUS IN 2010"  
SSI11="SSI CURRENT PAY STATUS IN 2011"  
SSI12="SSI CURRENT PAY STATUS IN 2012"  
SSI13="SSI CURRENT PAY STATUS IN 2013"  
SSI14="SSI CURRENT PAY STATUS IN 2014"  
SSI15="SSI CURRENT PAY STATUS IN 2015"  
SSI16="SSI CURRENT PAY STATUS IN 2016"  
SSI17="SSI CURRENT PAY STATUS IN 2017"  
SSI18="SSI CURRENT PAY STATUS IN 2018"  
SSI19="SSI CURRENT PAY STATUS IN 2019"  
SSI20="SSI CURRENT PAY STATUS IN 2020"  
SSI21="SSI CURRENT PAY STATUS IN 2021"  
SSIELIG\_FIRST="SSI EARLIEST ELIGIBILITY DATE (CONSTRUCTED)"  
SSIERNLVL&YR.01="SSI EARNING LEVEL INDICATOR &YR.01"  
SSIERNLVL&YR.02="SSI EARNING LEVEL INDICATOR &YR.02"  
SSIERNLVL&YR.03="SSI EARNING LEVEL INDICATOR &YR.03"  
SSIERNLVL&YR.04="SSI EARNING LEVEL INDICATOR &YR.04"  
SSIERNLVL&YR.05="SSI EARNING LEVEL INDICATOR &YR.05"  
SSIERNLVL&YR.06="SSI EARNING LEVEL INDICATOR &YR.06"  
SSIERNLVL&YR.07="SSI EARNING LEVEL INDICATOR &YR.07"  
SSIERNLVL&YR.08="SSI EARNING LEVEL INDICATOR &YR.08"  
SSIERNLVL&YR.09="SSI EARNING LEVEL INDICATOR &YR.09"  
SSIERNLVL&YR.10="SSI EARNING LEVEL INDICATOR &YR.10"  
SSIERNLVL&YR.11="SSI EARNING LEVEL INDICATOR &YR.11"  
SSIERNLVL&YR.12="SSI EARNING LEVEL INDICATOR &YR.12"  
SSN="SOCIAL SECURITY NUMBER"  
STATE&YR.01="STATE POSTAL CODE (BEST) &YR.01"  
STATE&YR.02="STATE POSTAL CODE (BEST) &YR.02"  
STATE&YR.03="STATE POSTAL CODE (BEST) &YR.03"  
STATE&YR.04="STATE POSTAL CODE (BEST) &YR.04"  
STATE&YR.05="STATE POSTAL CODE (BEST) &YR.05"

Appendix A.159

JCL/SAS Code: LABELS

STATE&YR.06="STATE POSTAL CODE (BEST) &YR.06"  
STATE&YR.07="STATE POSTAL CODE (BEST) &YR.07"  
STATE&YR.08="STATE POSTAL CODE (BEST) &YR.08"  
STATE&YR.09="STATE POSTAL CODE (BEST) &YR.09"  
STATE&YR.10="STATE POSTAL CODE (BEST) &YR.10"  
STATE&YR.11="STATE POSTAL CODE (BEST) &YR.11"  
STATE&YR.12="STATE POSTAL CODE (BEST) &YR.12"  
STWCM&YR.01=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.01"  
STWCM&YR.02=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.02"  
STWCM&YR.03=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.03"  
STWCM&YR.04=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.04"  
STWCM&YR.05=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.05"  
STWCM&YR.06=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.06"  
STWCM&YR.07=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.07"  
STWCM&YR.08=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.08"  
STWCM&YR.09=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.09"  
STWCM&YR.10=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.10"  
STWCM&YR.11=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.11"  
STWCM&YR.12=  
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK &YR.12"  
STWDI&YR.01="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.01"  
STWDI&YR.02="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.02"  
STWDI&YR.03="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.03"  
STWDI&YR.04="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.04"  
STWDI&YR.05="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.05"  
STWDI&YR.06="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.06"  
STWDI&YR.07="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.07"  
STWDI&YR.08="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.08"  
STWDI&YR.09="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.09"  
STWDI&YR.10="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.10"  
STWDI&YR.11="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.11"  
STWDI&YR.12="SSDI SUSPENSE OR TERMINATION DUE TO WORK &YR.12"  
STWSSI&YR.01="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.01"  
STWSSI&YR.02="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.02"

Appendix A.159

JCL/SAS Code: LABELS

STWSSI&YR.03="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.03"  
STWSSI&YR.04="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.04"  
STWSSI&YR.05="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.05"  
STWSSI&YR.06="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.06"  
STWSSI&YR.07="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.07"  
STWSSI&YR.08="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.08"  
STWSSI&YR.09="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.09"  
STWSSI&YR.10="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.10"  
STWSSI&YR.11="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.11"  
STWSSI&YR.12="SSI SUSPENSE OR TERMINATION DUE TO WORK &YR.12"  
STUDM&YR.01="SSI Student Indicator (SSR) &YR.01"  
STUDM&YR.02="SSI Student Indicator (SSR) &YR.02"  
STUDM&YR.03="SSI Student Indicator (SSR) &YR.03"  
STUDM&YR.04="SSI Student Indicator (SSR) &YR.04"  
STUDM&YR.05="SSI Student Indicator (SSR) &YR.05"  
STUDM&YR.06="SSI Student Indicator (SSR) &YR.06"  
STUDM&YR.07="SSI Student Indicator (SSR) &YR.07"  
STUDM&YR.08="SSI Student Indicator (SSR) &YR.08"  
STUDM&YR.09="SSI Student Indicator (SSR) &YR.09"  
STUDM&YR.10="SSI Student Indicator (SSR) &YR.10"  
STUDM&YR.11="SSI Student Indicator (SSR) &YR.11"  
STUDM&YR.12="SSI Student Indicator (SSR) &YR.12"  
TKT02="TICKET ACTIVE IN 2002"  
TKT03="TICKET ACTIVE IN 2003"  
TKT04="TICKET ACTIVE IN 2004"  
TKT05="TICKET ACTIVE IN 2005"  
TKT06="TICKET ACTIVE IN 2006"  
TKT07="TICKET ACTIVE IN 2007"  
TKT08="TICKET ACTIVE IN 2008"  
TKT09="TICKET ACTIVE IN 2009"  
TKT10="TICKET ACTIVE IN 2010"  
TKT11="TICKET ACTIVE IN 2011"  
TKT12="TICKET ACTIVE IN 2012"  
TKT13="TICKET ACTIVE IN 2013"  
TKT14="TICKET ACTIVE IN 2014"  
TKT15="TICKET ACTIVE IN 2015"  
TKT16="TICKET ACTIVE IN 2016"  
TKT17="TICKET ACTIVE IN 2017"  
TKT18="TICKET ACTIVE IN 2018"  
TKT19="TICKET ACTIVE IN 2019"  
TKT20="TICKET ACTIVE IN 2020"  
TKT21="TICKET ACTIVE IN 2021"  
TWPDATA&YR.01="SSDI TRIAL WORK PERIOD CODE &YR.01"  
TWPDATA&YR.02="SSDI TRIAL WORK PERIOD CODE &YR.02"  
TWPDATA&YR.03="SSDI TRIAL WORK PERIOD CODE &YR.03"

Appendix A.159

JCL/SAS Code: LABELS

```
TWPDATA&YR.04="SSDI TRIAL WORK PERIOD CODE &YR.04"
TWPDATA&YR.05="SSDI TRIAL WORK PERIOD CODE &YR.05"
TWPDATA&YR.06="SSDI TRIAL WORK PERIOD CODE &YR.06"
TWPDATA&YR.07="SSDI TRIAL WORK PERIOD CODE &YR.07"
TWPDATA&YR.08="SSDI TRIAL WORK PERIOD CODE &YR.08"
TWPDATA&YR.09="SSDI TRIAL WORK PERIOD CODE &YR.09"
TWPDATA&YR.10="SSDI TRIAL WORK PERIOD CODE &YR.10"
TWPDATA&YR.11="SSDI TRIAL WORK PERIOD CODE &YR.11"
TWPDATA&YR.12="SSDI TRIAL WORK PERIOD CODE &YR.12"
T16BEXPAMT&YR.01=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.01"
T16BEXPAMT&YR.02=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.02"
T16BEXPAMT&YR.03=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.03"
T16BEXPAMT&YR.04=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.04"
T16BEXPAMT&YR.05=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.05"
T16BEXPAMT&YR.06=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.06"
T16BEXPAMT&YR.07=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.07"
T16BEXPAMT&YR.08=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.08"
T16BEXPAMT&YR.09=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.09"
T16BEXPAMT&YR.10=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.10"
T16BEXPAMT&YR.11=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.11"
T16BEXPAMT&YR.12=
"SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE) SSR &YR.12"
T16EXLAMT&YR.01="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.01"
T16EXLAMT&YR.02="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.02"
T16EXLAMT&YR.03="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.03"
T16EXLAMT&YR.04="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.04"
T16EXLAMT&YR.05="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.05"
T16EXLAMT&YR.06="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.06"
T16EXLAMT&YR.07="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.07"
T16EXLAMT&YR.08="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.08"
T16EXLAMT&YR.09="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.09"
T16EXLAMT&YR.10="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.10"
T16EXLAMT&YR.11="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.11"
T16EXLAMT&YR.12="SSI STUDENT EXCLUSION AMOUNT (SSR) &YR.12"
```

Appendix A.159

JCL/SAS Code: LABELS

T16EXPAMT&YR.01=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.01"  
T16EXPAMT&YR.02=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.02"  
T16EXPAMT&YR.03=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.03"  
T16EXPAMT&YR.04=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.04"  
T16EXPAMT&YR.05=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.05"  
T16EXPAMT&YR.06=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.06"  
T16EXPAMT&YR.07=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.07"  
T16EXPAMT&YR.08=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.08"  
T16EXPAMT&YR.09=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.09"  
T16EXPAMT&YR.10=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.10"  
T16EXPAMT&YR.11=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.11"  
T16EXPAMT&YR.12=  
"SSI IMPAIRMENT-RELATED WORK EXPENSE (IRWE) AMOUNT (SSR) &YR.12"  
T16GRSAMT&YR.01="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.01"  
T16GRSAMT&YR.02="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.02"  
T16GRSAMT&YR.03="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.03"  
T16GRSAMT&YR.04="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.04"  
T16GRSAMT&YR.05="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.05"  
T16GRSAMT&YR.06="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.06"  
T16GRSAMT&YR.07="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.07"  
T16GRSAMT&YR.08="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.08"  
T16GRSAMT&YR.09="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.09"  
T16GRSAMT&YR.10="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.10"  
T16GRSAMT&YR.11="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.11"  
T16GRSAMT&YR.12="SSI EARNINGS GROSS WAGE AMOUNT (SSR) &YR.12"  
T16NETAMT&YR.01="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.01"  
T16NETAMT&YR.02="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.02"  
T16NETAMT&YR.03="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.03"  
T16NETAMT&YR.04="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.04"  
T16NETAMT&YR.05="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.05"  
T16NETAMT&YR.06="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.06"  
T16NETAMT&YR.07="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.07"  
T16NETAMT&YR.08="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.08"  
T16NETAMT&YR.09="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.09"

Appendix A.159

JCL/SAS Code: LABELS

T16NETAMT&YR.10="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.10"  
T16NETAMT&YR.11="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.11"  
T16NETAMT&YR.12="SSI SELF-EMPLOYMENT NET LOSS AMOUNT (SSR) &YR.12"  
T16PASAMT&YR.01="SSI PASS AMOUNT (SSR) &YR.01"  
T16PASAMT&YR.02="SSI PASS AMOUNT (SSR) &YR.02"  
T16PASAMT&YR.03="SSI PASS AMOUNT (SSR) &YR.03"  
T16PASAMT&YR.04="SSI PASS AMOUNT (SSR) &YR.04"  
T16PASAMT&YR.05="SSI PASS AMOUNT (SSR) &YR.05"  
T16PASAMT&YR.06="SSI PASS AMOUNT (SSR) &YR.06"  
T16PASAMT&YR.07="SSI PASS AMOUNT (SSR) &YR.07"  
T16PASAMT&YR.08="SSI PASS AMOUNT (SSR) &YR.08"  
T16PASAMT&YR.09="SSI PASS AMOUNT (SSR) &YR.09"  
T16PASAMT&YR.10="SSI PASS AMOUNT (SSR) &YR.10"  
T16PASAMT&YR.11="SSI PASS AMOUNT (SSR) &YR.11"  
T16PASAMT&YR.12="SSI PASS AMOUNT (SSR) &YR.12"  
T16SEAMT&YR.01="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.01"  
T16SEAMT&YR.02="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.02"  
T16SEAMT&YR.03="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.03"  
T16SEAMT&YR.04="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.04"  
T16SEAMT&YR.05="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.05"  
T16SEAMT&YR.06="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.06"  
T16SEAMT&YR.07="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.07"  
T16SEAMT&YR.08="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.08"  
T16SEAMT&YR.09="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.09"  
T16SEAMT&YR.10="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.10"  
T16SEAMT&YR.11="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.11"  
T16SEAMT&YR.12="SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) &YR.12"  
T2CDNAMT&YR.01="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.01"  
T2CDNAMT&YR.02="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.02"  
T2CDNAMT&YR.03="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.03"  
T2CDNAMT&YR.04="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.04"  
T2CDNAMT&YR.05="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.05"  
T2CDNAMT&YR.06="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.06"  
T2CDNAMT&YR.07="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.07"  
T2CDNAMT&YR.08="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.08"  
T2CDNAMT&YR.09="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.09"  
T2CDNAMT&YR.10="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.10"  
T2CDNAMT&YR.11="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.11"  
T2CDNAMT&YR.12="SSDI SPECIAL CONDITION AMOUNT (DCF) &YR.12"  
T2EXPAMT&YR.01="SSDI WORK EXPENSE AMOUNT (DCF) &YR.01"  
T2EXPAMT&YR.02="SSDI WORK EXPENSE AMOUNT (DCF) &YR.02"  
T2EXPAMT&YR.03="SSDI WORK EXPENSE AMOUNT (DCF) &YR.03"  
T2EXPAMT&YR.04="SSDI WORK EXPENSE AMOUNT (DCF) &YR.04"  
T2EXPAMT&YR.05="SSDI WORK EXPENSE AMOUNT (DCF) &YR.05"  
T2EXPAMT&YR.06="SSDI WORK EXPENSE AMOUNT (DCF) &YR.06"



Appendix A.159

JCL/SAS Code: LABELS

T2EXPAMT&YR.07="SSDI WORK EXPENSE AMOUNT (DCF) &YR.07"  
T2EXPAMT&YR.08="SSDI WORK EXPENSE AMOUNT (DCF) &YR.08"  
T2EXPAMT&YR.09="SSDI WORK EXPENSE AMOUNT (DCF) &YR.09"  
T2EXPAMT&YR.10="SSDI WORK EXPENSE AMOUNT (DCF) &YR.10"  
T2EXPAMT&YR.11="SSDI WORK EXPENSE AMOUNT (DCF) &YR.11"  
T2EXPAMT&YR.12="SSDI WORK EXPENSE AMOUNT (DCF) &YR.12"  
T2FRAUDVER&YR.01="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.01"  
T2FRAUDVER&YR.02="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.02"  
T2FRAUDVER&YR.03="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.03"  
T2FRAUDVER&YR.04="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.04"  
T2FRAUDVER&YR.05="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.05"  
T2FRAUDVER&YR.06="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.06"  
T2FRAUDVER&YR.07="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.07"  
T2FRAUDVER&YR.08="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.08"  
T2FRAUDVER&YR.09="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.09"  
T2FRAUDVER&YR.10="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.10"  
T2FRAUDVER&YR.11="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.11"  
T2FRAUDVER&YR.12="SSDI FRAUD VERIFIED SWITCH (DCF) &YR.12"  
T2GRSAMT&YR.01="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.01"  
T2GRSAMT&YR.02="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.02"  
T2GRSAMT&YR.03="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.03"  
T2GRSAMT&YR.04="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.04"  
T2GRSAMT&YR.05="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.05"  
T2GRSAMT&YR.06="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.06"  
T2GRSAMT&YR.07="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.07"  
T2GRSAMT&YR.08="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.08"  
T2GRSAMT&YR.09="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.09"  
T2GRSAMT&YR.10="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.10"  
T2GRSAMT&YR.11="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.11"  
T2GRSAMT&YR.12="SSDI EARNINGS GROSS AMOUNT (DCF) &YR.12"  
T2NETAMT&YR.01="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.01"  
T2NETAMT&YR.02="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.02"  
T2NETAMT&YR.03="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.03"  
T2NETAMT&YR.04="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.04"  
T2NETAMT&YR.05="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.05"  
T2NETAMT&YR.06="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.06"  
T2NETAMT&YR.07="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.07"  
T2NETAMT&YR.08="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.08"  
T2NETAMT&YR.09="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.09"  
T2NETAMT&YR.10="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.10"  
T2NETAMT&YR.11="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.11"  
T2NETAMT&YR.12="SSDI SELF-EMP NET INCOME AMOUNT (DCF) &YR.12"  
T2SBDYAMT&YR.01="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.01"  
T2SBDYAMT&YR.02="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.02"  
T2SBDYAMT&YR.03="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.03"

Appendix A.159

JCL/SAS Code: LABELS

T2SBDYAMT&YR.04="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.04"  
T2SBDYAMT&YR.05="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.05"  
T2SBDYAMT&YR.06="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.06"  
T2SBDYAMT&YR.07="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.07"  
T2SBDYAMT&YR.08="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.08"  
T2SBDYAMT&YR.09="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.09"  
T2SBDYAMT&YR.10="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.10"  
T2SBDYAMT&YR.11="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.11"  
T2SBDYAMT&YR.12="SSDI EARNINGS SUBSIDY AMOUNT (DCF) &YR.12"  
T2SEHRS&YR.01="SSDI SELF-EMP HOURS (DCF) &YR.01"  
T2SEHRS&YR.02="SSDI SELF-EMP HOURS (DCF) &YR.02"  
T2SEHRS&YR.03="SSDI SELF-EMP HOURS (DCF) &YR.03"  
T2SEHRS&YR.04="SSDI SELF-EMP HOURS (DCF) &YR.04"  
T2SEHRS&YR.05="SSDI SELF-EMP HOURS (DCF) &YR.05"  
T2SEHRS&YR.06="SSDI SELF-EMP HOURS (DCF) &YR.06"  
T2SEHRS&YR.07="SSDI SELF-EMP HOURS (DCF) &YR.07"  
T2SEHRS&YR.08="SSDI SELF-EMP HOURS (DCF) &YR.08"  
T2SEHRS&YR.09="SSDI SELF-EMP HOURS (DCF) &YR.09"  
T2SEHRS&YR.10="SSDI SELF-EMP HOURS (DCF) &YR.10"  
T2SEHRS&YR.11="SSDI SELF-EMP HOURS (DCF) &YR.11"  
T2SEHRS&YR.12="SSDI SELF-EMP HOURS (DCF) &YR.12"  
T2SEVERIND&YR.01="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.01"  
T2SEVERIND&YR.02="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.02"  
T2SEVERIND&YR.03="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.03"  
T2SEVERIND&YR.04="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.04"  
T2SEVERIND&YR.05="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.05"  
T2SEVERIND&YR.06="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.06"  
T2SEVERIND&YR.07="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.07"  
T2SEVERIND&YR.08="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.08"  
T2SEVERIND&YR.09="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.09"  
T2SEVERIND&YR.10="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.10"  
T2SEVERIND&YR.11="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.11"  
T2SEVERIND&YR.12="SSDI SELF-EMP EARNINGS VERIFIED SWITCH (DCF) &YR.12"  
T2UBEAMT&YR.01="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.01"  
T2UBEAMT&YR.02="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.02"  
T2UBEAMT&YR.03="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.03"  
T2UBEAMT&YR.04="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.04"  
T2UBEAMT&YR.05="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.05"  
T2UBEAMT&YR.06="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.06"  
T2UBEAMT&YR.07="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.07"  
T2UBEAMT&YR.08="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.08"  
T2UBEAMT&YR.09="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.09"  
T2UBEAMT&YR.10="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.10"  
T2UBEAMT&YR.11="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.11"  
T2UBEAMT&YR.12="SSDI SELF-EMP UNPAID BUSINESS EXP AMT (DCF) &YR.12"

Appendix A.159

JCL/SAS Code: LABELS

T2VERIND&YR.01="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.01"  
T2VERIND&YR.02="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.02"  
T2VERIND&YR.03="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.03"  
T2VERIND&YR.04="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.04"  
T2VERIND&YR.05="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.05"  
T2VERIND&YR.06="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.06"  
T2VERIND&YR.07="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.07"  
T2VERIND&YR.08="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.08"  
T2VERIND&YR.09="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.09"  
T2VERIND&YR.10="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.10"  
T2VERIND&YR.11="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.11"  
T2VERIND&YR.12="SSDI GROSS EARNINGS AMOUNT VERIFIED SWITCH (DCF) &YR.12"  
UINC&YR.01="SSI COUNTABLE UNEARNED INCOME &YR.01"  
UINC&YR.02="SSI COUNTABLE UNEARNED INCOME &YR.02"  
UINC&YR.03="SSI COUNTABLE UNEARNED INCOME &YR.03"  
UINC&YR.04="SSI COUNTABLE UNEARNED INCOME &YR.04"  
UINC&YR.05="SSI COUNTABLE UNEARNED INCOME &YR.05"  
UINC&YR.06="SSI COUNTABLE UNEARNED INCOME &YR.06"  
UINC&YR.07="SSI COUNTABLE UNEARNED INCOME &YR.07"  
UINC&YR.08="SSI COUNTABLE UNEARNED INCOME &YR.08"  
UINC&YR.09="SSI COUNTABLE UNEARNED INCOME &YR.09"  
UINC&YR.10="SSI COUNTABLE UNEARNED INCOME &YR.10"  
UINC&YR.11="SSI COUNTABLE UNEARNED INCOME &YR.11"  
UINC&YR.12="SSI COUNTABLE UNEARNED INCOME &YR.12"  
WIC&YR.01="SSDI WORK INDICATION CODE &YR.01"  
WIC&YR.02="SSDI WORK INDICATION CODE &YR.02"  
WIC&YR.03="SSDI WORK INDICATION CODE &YR.03"  
WIC&YR.04="SSDI WORK INDICATION CODE &YR.04"  
WIC&YR.05="SSDI WORK INDICATION CODE &YR.05"  
WIC&YR.06="SSDI WORK INDICATION CODE &YR.06"  
WIC&YR.07="SSDI WORK INDICATION CODE &YR.07"  
WIC&YR.08="SSDI WORK INDICATION CODE &YR.08"  
WIC&YR.09="SSDI WORK INDICATION CODE &YR.09"  
WIC&YR.10="SSDI WORK INDICATION CODE &YR.10"  
WIC&YR.11="SSDI WORK INDICATION CODE &YR.11"  
WIC&YR.12="SSDI WORK INDICATION CODE &YR.12"  
ZIP\_BEST\_BASE&YR.01 =  
    "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.01"  
ZIP\_BEST\_BASE&YR.02 =  
    "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.02"  
ZIP\_BEST\_BASE&YR.03 =  
    "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.03"  
ZIP\_BEST\_BASE&YR.04 =  
    "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.04"  
ZIP\_BEST\_BASE&YR.05 =

Appendix A.159

JCL/SAS Code: LABELS

```

        "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.05"
ZIP_BEST_BASE&YR.06 =
        "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.06"
ZIP_BEST_BASE&YR.07 =
        "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.07"
ZIP_BEST_BASE&YR.08 =
        "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.08"
ZIP_BEST_BASE&YR.09 =
        "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.09"
ZIP_BEST_BASE&YR.10 =
        "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.10"
ZIP_BEST_BASE&YR.11 =
        "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.11"
ZIP_BEST_BASE&YR.12 =
        "ZIP CODE 5-DIGIT BASE, BEST FROM ALL SOURCES &YR.12"
ZIP_DI_BASE&YR.01="SSDI ZIP CODE 5-DIGIT BASE &YR.01"
ZIP_DI_BASE&YR.02="SSDI ZIP CODE 5-DIGIT BASE &YR.02"
ZIP_DI_BASE&YR.03="SSDI ZIP CODE 5-DIGIT BASE &YR.03"
ZIP_DI_BASE&YR.04="SSDI ZIP CODE 5-DIGIT BASE &YR.04"
ZIP_DI_BASE&YR.05="SSDI ZIP CODE 5-DIGIT BASE &YR.05"
ZIP_DI_BASE&YR.06="SSDI ZIP CODE 5-DIGIT BASE &YR.06"
ZIP_DI_BASE&YR.07="SSDI ZIP CODE 5-DIGIT BASE &YR.07"
ZIP_DI_BASE&YR.08="SSDI ZIP CODE 5-DIGIT BASE &YR.08"
ZIP_DI_BASE&YR.09="SSDI ZIP CODE 5-DIGIT BASE &YR.09"
ZIP_DI_BASE&YR.10="SSDI ZIP CODE 5-DIGIT BASE &YR.10"
ZIP_DI_BASE&YR.11="SSDI ZIP CODE 5-DIGIT BASE &YR.11"
ZIP_DI_BASE&YR.12="SSDI ZIP CODE 5-DIGIT BASE &YR.12"
ZIP_SSI_BASE&YR.01="SSI ZIP CODE 5-DIGIT BASE &YR.01"
ZIP_SSI_BASE&YR.02="SSI ZIP CODE 5-DIGIT BASE &YR.02"
ZIP_SSI_BASE&YR.03="SSI ZIP CODE 5-DIGIT BASE &YR.03"
ZIP_SSI_BASE&YR.04="SSI ZIP CODE 5-DIGIT BASE &YR.04"
ZIP_SSI_BASE&YR.05="SSI ZIP CODE 5-DIGIT BASE &YR.05"
ZIP_SSI_BASE&YR.06="SSI ZIP CODE 5-DIGIT BASE &YR.06"
ZIP_SSI_BASE&YR.07="SSI ZIP CODE 5-DIGIT BASE &YR.07"
ZIP_SSI_BASE&YR.08="SSI ZIP CODE 5-DIGIT BASE &YR.08"
ZIP_SSI_BASE&YR.09="SSI ZIP CODE 5-DIGIT BASE &YR.09"
ZIP_SSI_BASE&YR.10="SSI ZIP CODE 5-DIGIT BASE &YR.10"
ZIP_SSI_BASE&YR.11="SSI ZIP CODE 5-DIGIT BASE &YR.11"
ZIP_SSI_BASE&YR.12="SSI ZIP CODE 5-DIGIT BASE &YR.12"
ZIP_BEST_EXT&YR.01 =
        "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.01"
ZIP_BEST_EXT&YR.02 =
        "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.02"
ZIP_BEST_EXT&YR.03 =
        "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.03"

```

## Appendix A.159

JCL/SAS Code: LABELS

```
ZIP_BEST_EXT&YR.04 =  
    "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.04"  
ZIP_BEST_EXT&YR.05 =  
    "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.05"  
ZIP_BEST_EXT&YR.06 =  
    "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.06"  
ZIP_BEST_EXT&YR.07 =  
    "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.07"  
ZIP_BEST_EXT&YR.08 =  
    "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.08"  
ZIP_BEST_EXT&YR.09 =  
    "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.09"  
ZIP_BEST_EXT&YR.10 =  
    "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.10"  
ZIP_BEST_EXT&YR.11 =  
    "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.11"  
ZIP_BEST_EXT&YR.12 =  
    "ZIP CODE 4-DIGIT EXTENSION, BEST FROM ALL SOURCES &YR.12"  
ZIP_DI_EXT&YR.01 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.01"  
ZIP_DI_EXT&YR.02 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.02"  
ZIP_DI_EXT&YR.03 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.03"  
ZIP_DI_EXT&YR.04 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.04"  
ZIP_DI_EXT&YR.05 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.05"  
ZIP_DI_EXT&YR.06 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.06"  
ZIP_DI_EXT&YR.07 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.07"  
ZIP_DI_EXT&YR.08 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.08"  
ZIP_DI_EXT&YR.09 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.09"  
ZIP_DI_EXT&YR.10 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.10"  
ZIP_DI_EXT&YR.11 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.11"  
ZIP_DI_EXT&YR.12 ="SSDI ZIP CODE 4-DIGIT EXTENSION &YR.12"  
ZIP_SSI_EXT&YR.01 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.01"  
ZIP_SSI_EXT&YR.02 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.02"  
ZIP_SSI_EXT&YR.03 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.03"  
ZIP_SSI_EXT&YR.04 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.04"  
ZIP_SSI_EXT&YR.05 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.05"  
ZIP_SSI_EXT&YR.06 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.06"  
ZIP_SSI_EXT&YR.07 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.07"  
ZIP_SSI_EXT&YR.08 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.08"  
ZIP_SSI_EXT&YR.09 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.09"  
ZIP_SSI_EXT&YR.10 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.10"  
ZIP_SSI_EXT&YR.11 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.11"  
ZIP_SSI_EXT&YR.12 ="SSI ZIP CODE 4-DIGIT EXTENSION &YR.12"  
;
```

## Appendix A.160

JCL/SAS Code: STWDI

```

//#3590STD JOB (12510000,T715,,SAS,,ITC9FL),SVETLANA-BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21 CONSTRUCTION 50214.220
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.STW.PRDLIB(STWDI)
//*
//* PURPOSE: TO CREATE SSDI MONTHLY STW INDICATOR
//*
//* PROGRAMMER: SVETLANA BRONNIKOV
//* MPR PHONE 202 264-4221
//* E-MAIL SBRONNIKOV@MATHEMATICA-MPR.COM
//*****
//*
//STWDI EXEC SAS9,
//      WORK='100000,50000'
//*
//DEMO DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBRREST.SA.V1,DISP=SHR
//DCF DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.T2WKDETN.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//TEMP2 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//STWDI DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.STWDI.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//STWDIFQ DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.STWDIFQ.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//*
//SYSIN DD *

```

```

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY MACROGEN MPRINT
OBS=MAX;

```

```

* CHANGE ENDYR AND ENDMN FOR EACH RUN;
%LET BEGYR=1994;
%LET ENDYR=2021; * CHANGE AS NEEDED;
%LET ENDMN=12; * CHANGE AS NEEDED;

```

```

%MACRO START;

```

## Appendix A.160

JCL/SAS Code: STWDI

```

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

DATA TEMP1.STWDI(KEEP=SSN CAN BIC STW: LAF: CDR: RFST:)
  MATCHPRIM (KEEP=SSN CAN STW:
    SUSFLAG TERMFLAG: LAF: CDR: RFST:)
  MATCHSEC (KEEP=SSN CAN SUSFLAG TERMFLAG: STW: LAF: CDR: RFST:
    RENAME=(SUSFLAG=SSUSFLAG TERMFLAG=STERMFLAG))
;
MERGE DEMO.DMSTW(IN=DEMO
  KEEP=SSN CAN DOBBEST DODBEST TWPCMLMNT1
  FRADATE)
  MBR.MBR(IN=MBR KEEP=SSN BIC RFD: WIC: CDR: RFST: LAF:
  DROP=LAF)
  DCF.T2WKDETN(IN=E KEEP=COSSN TWPDATA: RENAME=(COSSN=SSN))
;
BY SSN;

IF DEMO AND MBR;

* INPUT ARRAYS;
ARRAY LAF  (*) %DO MM=1 %TO &TOTMOS.; LAF&&YY&MM. %END;;
ARRAY WIC  (*) %DO MM=1 %TO &TOTMOS.; WIC&&YY&MM. %END;;
ARRAY RFD  (*) %DO MM=1 %TO &TOTMOS.; RFD&&YY&MM. %END;;
ARRAY CDR  (*) %DO MM=1 %TO &TOTMOS.; CDR&&YY&MM. %END;;
ARRAY RFST (*) %DO MM=1 %TO &TOTMOS.; RFST&&YY&MM. %END;;
ARRAY TWPDATA (*) %DO MM=1 %TO &TOTMOS.; TWPDATA&&YY&MM. %END;;

* OUTPUT ARRAYS;
ARRAY STWDI (*) %DO MM=1 %TO &TOTMOS.; STWDI&&YY&MM %END;;
ARRAY PROCDATE(*) %DO MM=1 %TO &TOTMOS.; PROCDATE&&YY&MM %END;;

* FIND THE MONTH FLAGGED BY TWPCMLMNT1 TO BEGIN PROCESSING.
  FLAG THE DEATH OR RETIREMENT INDEX TO END PROCESSING;
LENGTH STWNAME $9;
STWNAME=' ';
DO I=1 TO DIM(STWDI);
  * CONSTRUCT DATE FOR THE ARRAY ELEMENT;
  STWNAME=VNAME(STWDI(I));
  PROCDATE(I)=MDY(SUBSTR(STWNAME,8,2),1,SUBSTR(STWNAME,6,2));
END;

* DETERMINE EARILEST TWP COMPLETION DATE;
DO I=1 TO DIM(TWPDATA) UNTIL(TWPDT NE .);
  IF TWPDATA(I)='9'

```

## Appendix A.160

JCL/SAS Code: STWDI

```

        THEN TWPDT=MDY(SUBSTR(VNAME(TWPDATA(I)),10,2)
            ,1
            ,SUBSTR(VNAME(TWPDATA(I)),8,2));
END;
TWPDT=MAX('01JAN1994'D,MIN(TWPDT,TWPCMPLMNT1));

FOUNDWORK=0;
* SET THE BEGINING MONTH FOR PROCESSING
IF TWP COMPLETION BEFORE JAN 1994 THEN SET FOUNDWORK TO 1
OTHERWISE LOOP THROUGH MONTHS TO FIND THE MONTH OF TWP COMPLETION
OR IF NO TWP COMPLETION THEN THE FIRST SUSPENSION TRIPLET;
IF TWPDT<=PROCDATE(1) AND TWPDT NE . THEN FOUNDWORK=1;
ELSE DO I=1 TO DIM(PROCDATE) UNTIL (FOUNDWORK>0);
  IF TWPDT=. THEN DO;
    IF RFD(I)='7' AND
      (WIC(I) IN ('2','7') OR RFST(I) IN ('EPESGA' 'DIBWRK'))
      THEN FOUNDWORK=I;
  END;
  ELSE IF TWPDT=PROCDATE(I) THEN FOUNDWORK=I;
END; * END SET BEGINNING MONTH FOR PROCESSING;

STOPIND=0;
* SET THE END MONTH FOR PROCESSING;
DO I=1 TO DIM(PROCDATE) UNTIL (STOPIND>0);
  * BUILD INDICATORS ONLY UP TO RETIREMENT DATE OR DEATH DATE
  MISSINGS BEGIN IN THE MONTH FOLLOWING THOSE EVENTS;
  IF (PROCDATE(I)>=INTNX('MONTHS',FRADATE,0,'BEG')
    AND FRADATE>.Z) OR
    (PROCDATE(I)>=INTNX('MONTHS',DODBEST,0,'BEG')
    AND DODBEST>.Z) THEN STOPIND=I;
END;

* WHERE NO WORK DEATH OR RETIREMENT LIMITS PROCESSING SET THE
  STOP INDIEX TO THE LAST MONTH IN THE EVALUATION;
IF STOPIND=0 THEN STOPIND=&TOTMOS.;

* INITIALIZE THE STW VARIABLES;
DO I=1 TO STOPIND;
  IF LAF(I)=' ' THEN STWDI(I)=.;
  ELSE IF SUBSTR(LAF(I),1,1) IN ('C' 'E') THEN STWDI(I)=0;
  ELSE IF SUBSTR(LAF(I),1,1)='T' THEN STWDI(I)=9;
  ELSE STWDI(I)=8;
END;

IF FOUNDWORK>0 THEN DO I=FOUNDWORK TO STOPIND;

```



## Appendix A.160

JCL/SAS Code: STWDI

```
* WORK SUSPENSIONS;
IF RFD(I)='7' AND
  (WIC(I) IN ('2','7') OR RFST(I) IN ('EPESGA' 'DIBWRK'))
THEN DO;
  STWDI(I)=1;
  SUSFLAG=1;
END;

* WORK TERMINATIONS;
ELSE IF RFD(I)='T' AND WIC(I) IN ('8','6')
  AND (CDR(I)='E' OR TWPDATA(I)='S')
THEN DO;
  STWDI(I)=2;
  TERMFLAG=1;
END;

END; * END FOUNDWORK LOOP;
DROP PROCDATE;;

* RECODE TERMINATIONS AFTER A SUSPENSION DUE TO WORK (STWDI=1) AS A
  TERMINATION DUE TO WORK (STWDI=2) PROVIDED THAT THE TERMINATION
  WAS NOT DUE TO A MEDICAL RECOVERY (CDRyymm ne 'M'). THIS MAINLY
  EFFECTS BENES WHO COMPLETED A TWP AND ENTERED AN EPE;
DO I=2 TO STOPIND;
  IF STWDI(I)=9 AND STWDI(I-1) IN (1,2)
    AND SUBSTR(LAF(I),1,1)='T'
    AND CDR(I) NE 'M'
  THEN DO;
    STWDI(I)=2;
  END;
END;

* ONCE TERMINATION FOR WORK ALWAYS TERMINATED FOR WORK UNTIL BACK
  IN CURRENT PAY, FRA, OR DEAD;
DO I=2 TO STOPIND;
  IF STWDI(I) NOT IN (0,.) AND STWDI(I-1)=2 THEN STWDI(I)=2;
END;

* OUTPUT 3 FILES
1. FLAGGED PRIMARIES
2. FLAGGED DACS
3. UNFLAGGED CASES
WE DO THIS BECAUSE WE NEED TO DETERMINE WHETHER WORK DATA ON
A DAC RECORD BELONGS TO THE PRIMARY AND NOT THE DAC;
```

## Appendix A.160

JCL/SAS Code: STWDI

```
IF SUSFLAG=1 OR TERMFLAG=1 THEN DO;
  IF SUBSTR(BIC,1,1)='C' THEN OUTPUT MATCHSEC; * DACS;
  ELSE OUTPUT MATCHPRIM; * PRIMARIES;
END;
ELSE OUTPUT TEMP1.STWDI; * NO WORK INDICATION ON THESE RECORDS;
RUN;

PROC SORT DATA=MATCHPRIM;
  BY CAN;
RUN;

PROC SORT DATA=MATCHSEC;
  BY CAN;
RUN;

DATA MATCHSEC (KEEP=SSN STW: RFST:);
  MERGE MATCHPRIM (IN=P KEEP=SSN CAN SUSFLAG TERMFLAG
    WHERE=(CAN NE ' '))
    MATCHSEC (IN=S)
  ;
  BY CAN;

  * KEEP ONLY DAC CASES;
  IF S;

  ARRAY LAF (*) %DO MM=1 %TO &TOTMOS.; LAF&&YY&MM %END;;
  ARRAY STWDI {*} %DO MM=1 %TO &TOTMOS.; STWDI&&YY&MM %END;;

  * IF DAC HAS A PRIMARY WHO HAS BEEN FLAGGED - RESET INDICATOR
  TO 9 FOR THE DAC BECAUSE THE WORK INDICATION MOST LIKELY
  BELONGS TO THE PRIMARY;
  IF P THEN DO;
    IF SUSFLAG=1 AND SSUSFLAG=1 THEN DO I=1 TO DIM(STWDI);
      IF STWDI(I)=1 THEN DO;
        IF SUBSTR(LAF(I),1,1)='T' THEN STWDI(I)=9;
        ELSE STWDI(I)=8;
      END;
    END;
  IF TERMFLAG=1 AND STERMFLAG=1 THEN DO I=1 TO DIM(STWDI);
    IF STWDI(I)=2 THEN DO;
      IF SUBSTR(LAF(I),1,1)='T' THEN STWDI(I)=9;
      ELSE STWDI(I)=8;
    END;
  END;
```

Appendix A.160

JCL/SAS Code: STWDI

```

    END; * END IF P LOOP;

RUN;

* PUT THE PIECES BACK TOGETHER;
PROC SORT DATA=MATCHPRIM;
  BY SSN;
RUN;

PROC SORT DATA=MATCHSEC;
  BY SSN;
RUN;

DATA TEMP2.STWDI (KEEP=SSN STW: RFST:);
SET TEMP1.STWDI
  MATCHPRIM
  MATCHSEC
  ;
BY SSN;

RUN;

* DEDUP THE FEW DUPLICATES BECAUSE OF BAD BOANS;
PROC SORT DATA=TEMP2.STWDI OUT=STWDI.STWDI NODUPKEY;
  BY SSN;
RUN;

PROC FREQ DATA=STWDI.STWDI;
  %DO MM=1 %TO &TOTMOS.;
    TABLES STWDI&&YY&MM / MISSING OUT=STWDIFQ.STWDIFREQ&&YY&MM;
  %END;
RUN;

%MEND;
%START;
/*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>

```

Appendix A.160

JCL/SAS Code: STWDI

```
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>  
DATA  
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>  
TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 STWDI COMPLETE
```

## Appendix A.161

JCL/SAS Code: STWSSI

```
//#3590STS JOB (12510000,T715,,SAS,,ITC9FL),SVETLANA-BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21 CONSTRUCTION 50214
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.STW.PRDLIB(STWSSI)
//*
//* PURPOSE: TO CREATE SSI MONTHLY STW INDICATOR
//*
//* PROGRAMMER:
//* SVETLANA BRONNIKOV
//* MPR PHONE 202 264-4221
//* E-MAIL SBRONNIKOV@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//DEMO  DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//SSR   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIELIG.SA.V1,DISP=SHR
//SCWF  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.STWSSI.SA.V1,
//      DISP=(NEW,CATLG,DELETE),UNIT=TSILO
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
```

```
%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */
```

```
%MACRO START;
```

```
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";
```

```
DATA SCWF.STWSSI /*(KEEP=SSN STWSSI:)*;/
MERGE DEMO.DMSTW(IN=DMG KEEP=SSN DOBBEST DODBEST FRADATE)
      SSR.SSIELIG(IN=SSR KEEP=SSN PSTA: EICM: CURCOMP: UINC: FCI: BMF:
      DROP=FCI BMF)
;
BY SSN;
IF SSR AND DMG;
```

```
* KEEP ONLY BENES WHO ARE 18 OR OLDER ON THE LAST DAY OF THE DAF YEAR
STWS ARE ONLY CONSTRUCTED FOR ADULT BENES;
```

## Appendix A.161

JCL/SAS Code: STWSSI

```

IF YRDIF(DOBBEST,"31DEC&ENDYR."D,'AGE')>=18;

* CREATE SSI STW INDICATORS;

* INPUT ARRAYS;
ARRAY PSTA(*) %DO MM=1 %TO &TOTMOS.; PSTA&&YY&MM %END;;
ARRAY EICM(*) %DO MM=1 %TO &TOTMOS.; EICM&&YY&MM %END;;
ARRAY UINC(*) %DO MM=1 %TO &TOTMOS.; UINC&&YY&MM %END;;
ARRAY CURCOMP(*) $ %DO MM=1 %TO &TOTMOS.; CURCOMP&&YY&MM %END;;
ARRAY FCI(*) %DO MM=1 %TO &TOTMOS.; FCI&&YY&MM %END;;
ARRAY BMF(*) %DO MM=1 %TO &TOTMOS.; BMF&&YY&MM %END;;

* FEDERAL BENEFIT RATES ARRAYS FOR INDIVIDUALS AND COUPLES
  UPDATE FOR COMING YEAR. FIND TABLE AT
  http://www.ssa.gov/oact/cola/SSlamts.html;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(FBRIN)";
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(FBRCP)";

* OUTPUT ARRAYS;
ARRAY FBR(*) %DO MM=1 %TO &TOTMOS.; FBR&&YY&MM %END;;

* THE STWSSI ARRAY HAS A LOWER BOUND OF 0 SO THAT WHEN LOOKING BACK
  AT THE PREVIOUS MONTH DURING THE STWSSI LOOP, SPECIAL CODING IS NOT
  NECESSARY TO HANDLE THE FIRST MONTH (JAN 1994). SO AN INDEX VALUE OF
  0 WOULD REFER TO THE DUMMYSTWSSI ARRAY ELEMENT AND AN INDEX VALUE OF
  1 WOULD REFER TO STWSSI9401;
ARRAY STWSSI(0:&TOTMOS.) DUMMYSTWSSI
  %DO MM=1 %TO &TOTMOS.; STWSSI&&YY&MM %END;;

ARRAY USED_UINC(*) %DO MM=1 %TO &TOTMOS.; USED_UINC&&YY&MM %END;;
ARRAY USED_TCNEI(*) %DO MM=1 %TO &TOTMOS.; USED_TCNEI&&YY&MM %END;;

* SET VARIABLES TO CONTROL THE LOOP SETTING THE STW VARIABLE:
  POPULATE STW FROM THE BEGINNING OF THE DAF DATA (JAN 1994) OR BENES
  18TH BIRTHDAY WHICHEVER COMES LATER UNTIL BENE DIES, ATTAINS FRA, OR
  THE END OF OUR DAF YEAR WHICHEVER COMES FIRST;
STARTLOOP=INTCK('MONTHS','01JAN1994'D,
  MAX('01JAN1994'D,
  INTNX('YEARS',DOBBEST,18,'SAME DAY')))+1;
STOPLOOP=INTCK('MONTHS','01JAN1994'D,
  MIN(DOBBEST,FRADATE,"31DEC&ENDYR."D))+1;

DO I=STARTLOOP TO STOPLOOP;

* SET YEAR BEING PROCESSED

```

## Appendix A.161

JCL/SAS Code: STWSSI

```
CONVERT RELATIVE MONTHS (REPRESENTED BY I) TO A YEAR VALUE BY
DIVIDING THE RELATIVE MONTHS BY 12 AND APPLY THE CEIL FUNCTION TO
RETURN THE NEXT LARGEST INTEGER OR EQUAL. SO AN INDEX VALUE OF 1
THROUGH 12 WOULD REFER TO YEAR 1994, 13 THROUGH 24 TO YEAR 1995 AND
SO ON;
```

```
LOOPYR=CEIL(I/12)+1993;
```

```
*FBR FOR COUPLE IS MAXIMUM SSI COUPLE PAYMENT DIVIDED BY 2.
```

```
FBR FOR INDIVIDUAL IS MAXIMUM SSI INDIVIDUAL PAYMENT;
```

```
IF CURCOMP(I)='6' THEN FBR(I)=FBRCP(LOOPYR)/2;
```

```
ELSE FBR(I)=FBRIN(LOOPYR);
```

```
* IN CURRENT PAY STW=0;
```

```
IF PSTA(I)='C01' THEN STWSSI(I)=0;
```

```
* ONCE TERMINATED FOR WORK ALWAYS TERMINATED FOR WORK UNTIL BACK IN
CURRENT PAY, FRA, OR DEAD. (THE FIRST TIME THROUGH THIS LOOP THERE
WILL BE NO VALUE IN THE PREVIOUS MONTH (I-1) SO THIS BLOCK OF CODE
WILL NOT EXECUTE);
```

```
ELSE IF STWSSI(I-1)=2 THEN DO;
```

```
  * WHEN THERE IS A STATUS CODE THEN STWSSI=2;
```

```
  IF PSTA(I) NE " THEN STWSSI(I)=2;
```

```
  * IF NO STATUS CODE THEN STWSSI=3;
```

```
  ELSE STWSSI(I)=3;
```

```
END;
```

```
ELSE IF STWSSI(I-1)=3 THEN STWSSI(I)=3;
```

```
* SUSPENDED FOR EXCESS INCOME (N01) THEN CHECK INCOME;
```

```
ELSE IF PSTA(I)='N01' THEN DO;
```

```
  * WHEN EARNED INCOME IS GREATER THAN ZERO OR THE PREVIOUS MONTH
  WAS AN STW SUSPENSE MONTH THEN CHECK NON-EARNED INCOME;
```

```
  IF EICM(I)>0 OR STWSSI(I-1)=1 THEN DO;
```

```
    * WHEN NON-EARNED INCOME IS BELOW FBR STW=1
```

```
    NON-EARNED INCOME IS ANY INCOME THAT IS NOT FROM WAGES
```

```
    (E.G. INTEREST, LOTTERY WINNINGS, VA BENEFITS, ETC.)
```

```
    TO DETERMINE NON-EARNED INCOME TAKE THE FEDERAL COUNTABLE
```

```
    INCOME FOR THIS MONTH AND SUBTRACT THE COUNTABLE EARNED
```

```
    INCOME FROM THIS MONTH. IF A BENE IS IN N01 STATUS THEN IT
```

```
    IS THE INCOME FROM THAT SAME MONTH THAT CAUSED THE NON PAY
```

```
    STATUS;
```

```
    IF FCI(I)=. THEN DO;
```

```
      IF UINC(I)<FBR(I) THEN STWSSI(I)=1;
```

## Appendix A.161

JCL/SAS Code: STWSSI

```
ELSE STWSSI(I)=4;
USED_UINC(I)=1;
END;
ELSE DO;
IF SUM(FCI(I),EICM(I)*-1)<FBR(I) THEN STWSSI(I)=1;
ELSE STWSSI(I)=4;
USED_TCNEI(I)=1;
END;
```

\* WHEN NON-EARNED INCOME IS AT OR ABOVE FBR STW=4. IF NON-EARNED INCOME WAS HIGH ENOUGH TO PRECLUDE A PAYMENT THEN THE EARNED INCOME WAS NOT RELEVANT TO THE SUSPENSE FOR EXCESS INCOME AND THEREFORE THE BENE IS NOT CONSIDERED SUSPENDED DUE TO WORK;

END;

\* WHEN SUSPENDED FOR EXCESS INCOME BUT HAVING NO EARNED INCOME AND NOT HAVING BEEN SUSPENDED FOR WORK IN THE PREVIOUS MONTH THE BENE IS SUSPENDED FOR REASONS OTHER THAN WORK;

ELSE STWSSI(I)=8;

END; \* END N01 LOOP;

\* WHEN PREVIOUS MONTH IS SUSPENDED FOR WORK AND CURRENT MONTH SUSPENDED FOR NON-EXCLUDEABLE RESOURCES OR FAILURE TO FILE FOR OTHER BENEFITS CHECK AND EARNED INCOME IS GREATER THAN 0 THEN CHECK UNEARNED INCOME;

ELSE IF PSTA(I) IN ('N04','N06') AND STWSSI(I-1)=1  
AND EICM(I)>0 THEN DO;

\* WHEN NON-EARNED INCOME IS BELOW FBR STW=1  
NON-EARNED INCOME IS ANY INCOME THAT IS NOT FROM WAGES  
(E.G. INTEREST, LOTTERY WINNINGS, VA BENEFITS, ETC.)  
TO DETERMINE NON-EARNED INCOME TAKE THE FEDERAL COUNTABLE  
INCOME FOR THIS MONTH AND SUBTRACT THE COUNTABLE EARNED  
INCOME FROM THIS MONTH. IF A BENE IS IN N01 STATUS THEN IT  
IS THE INCOME FROM THAT SAME MONTH THAT CAUSED THE NON PAY  
STATUS;

IF FCI(I)=. THEN DO;

IF UINC(I)<FBR(I) THEN STWSSI(I)=1;

ELSE STWSSI(I)=4;

USED\_UINC(I)=1;

END;

ELSE DO;

IF SUM(FCI(I),EICM(I)\*-1)<FBR(I) THEN STWSSI(I)=1;

ELSE STWSSI(I)=4;

USED\_TCNEI(I)=1;



## Appendix A.161

JCL/SAS Code: STWSSI

```
END;
```

```
* WHEN NON-EARNED INCOME IS AT OR ABOVE FBR STW=4. IF NON-
  EARNED INCOME WAS HIGH ENOUGH TO PRECLUDE A PAYMENT THEN
  THE EARNED INCOME WAS NOT RELEVANT TO THE SUSPENSE FOR
  EXCESS INCOME AND THEREFORE THE BENE IS NOT CONSIDERED
  SUSPENDED DUE TO WORK;
```

```
END; * END N04/N06 LOOP;
```

```
* WHEN TERMINATED FOR SOMETHING OTHER THAN DEATH AND PREVIOUS
  MONTH IS SUSPENDED OR TERMINATED FOR WORK THEN SUSPENDED FOR WORK;
  ELSE IF SUBSTR(PSTA(I),1,1)='T' AND PSTA(I) NE 'T01'
    AND STWSSI(I-1) IN (1,2) THEN STWSSI(I)=2;
```

```
* IF NONE OF THE ABOVE CONDITIONS ARE MET AND THERE IS A VALUE IN
  PSTA THEN SUSPENDED OR TERMINATION FOR SOMETHING OTHER THAN WORK;
  ELSE IF SUBSTR(PSTA(I),1,1)='T' THEN STWSSI(I)=9;
  ELSE IF PSTA(I) NE '' THEN STWSSI(I)=8;
```

```
* IF PSTA IS BLANK THEN IT WILL REMAIN MISSING;
```

```
END; * END MONTHLY LOOP;
```

```
LABEL
```

```
%DO MM=1 %TO &TOTMOS.;
```

```
STWSSI&&YY&MM = "&&YY&MM SSI LEFT DUE TO WORK INDICATOR"
```

```
%END;
```

```
;
```

```
RUN;
```

```
%MEND;
```

```
%START;
```

```
ENDSAS;
```

```
//*****
```

```
//* SENDING EMAIL FROM HOST USING IEBGENER
```

```
//*****
```

```
//SENDNOTE EXEC PGM=IEBGENER
```

```
//SYSIN DD DUMMY
```

```
//SYSPRINT DD SYSOUT=*
```

```
//SYSUT2 DD SYSOUT=(B,SMTP)
```

```
//SYSUT1 DD *
```

```
HELO MVSHOST
```

```
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
```

```
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
```

Appendix A.161

JCL/SAS Code: STWSSI

```
DATA  
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>  
TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 STWSSI IS COMPLETE
```

## Appendix A.162

JCL/SAS Code: STWCM

```
//#3590STW JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21 CONSTRUCTION 50214.220
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.STW.PRDLIB(STWCM)
//*
//* PURPOSE: TO CREATE MONTHLY COMBINED STW INDICATOR
//*
//* PROGRAMMER:
//* SVETLANA BRONNIKOV
//* MPR PHONE 202 264-4221
//* E-MAIL SBRONNIKOV@MATHEMATICA-MPR.COM
//*****
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//TEMP1 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE),VOL=(,,8)
//STWSSI DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.STWSSI.SA.V1,DISP=SHR
//STWDI  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.STWDI.SA.V1,DISP=SHR
//DEMO   DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//STWCM  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.STWCM.SA.V1,
//      DISP=(NEW,CATLG,DELETE),UNIT=TSILO
//STWCMFQ DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.STWCMFQ.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//SYSIN  DD *
OPTIONS NOCENTER COMPRESS=BINARY OBS=MAX MACROGEN MPRINT;

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

DATA STWCM.STWCM(KEEP=SSN STWSSI: STWDI: STWCM:);
  MERGE STWSSI.STWSSI(KEEP=SSN STWSSI:)
        STWDI.STWDI(KEEP=SSN STWDI: RFST:)
        DEMO.DMSTW(IN=DEMO KEEP=SSN FRADATE DODBEST)
  ;
```

Appendix A.162

JCL/SAS Code: STWCM

```

BY SSN;
IF DEMO;

* INPUT ARRAYS;
ARRAY STWDI (*) %DO MM=1 %TO &TOTMOS.; STWDI&&YY&MM %END;;
ARRAY STWSSI (*) %DO MM=1 %TO &TOTMOS.; STWSSI&&YY&MM %END;;
ARRAY RFST (*) %DO MM=1 %TO &TOTMOS.; RFST&&YY&MM %END;;

* OUTPUT ARRAYS;
ARRAY STWCM (*) %DO MM=1 %TO &TOTMOS.; STWCM&&YY&MM %END;;

DO I=1 TO &TOTMOS.;

    * SET INITIAL VALUE OF STWCM TO THE LOWER OF STWSSI AND STWSSI;
    STWCM(I)=MIN(STWDI(I),STWSSI(I));

    * RESET STWCM;
    SELECT(STWCM(I));

        * WHEN STWCM IS 4
        4=UNEARNED INCOME HIGHER THAN FBR.....TO 8
        ;
        WHEN(4) STWCM(I)=8;
        * OTHERWISE DO NOT RESET;
        OTHERWISE;
    END; * END SELECT;

    * IF THE STWSSI INDICATED SUSPENSE OF TERMINATION FOR WORK
    BUT SSDI DATA INDICATED THE BENE WAS IN PRISON THEN SET
    THE STWCM TO THAT OF THE STWDI;
    IF STWSSI(I) IN (1,2,3,4) AND RFST(I)='PRISON' THEN
    STWCM(I)=STWDI(I);

END; * END MONTHLY LOOPING;

* STRING STWCM 3S AND 9S IN THE STW INDICATORS;

* CONSTRUCT A DATE THAT IS THE EARLIEST OF FRA, DOD OR END OF DAF
YEAR THIS WILL ALLOW THE LOOP TO STOP STRINGING 3 WHEN A PERSON IS
DEAD OR FRA;
STOPLOOP=INTCK('MONTH','01JAN1994'D
,MIN(FRADATE,DODBEST,"31DEC&ENDYR."D))+1;

* STARTING IN FEB 1994 (2ND MONTH OF THE DAF) IF THE CURRENT MONTH
IS MISSING AND PREVIOUS MONTH IS 2 OR 3 (TERMINATED FOR WORK) THEN

```

## Appendix A.162

JCL/SAS Code: STWCM

```
SET CURRENT MONTH TO 3 ELSE IF PREVIOUS MONTH IS 9 THEN SET
CURRENT MONTH TO 9;
DO I=2 TO STOPLOOP;

  * STRING STWSSI;
  IF STWSSI(I)=. THEN DO;
    IF STWSSI(I-1) IN (2,3) THEN STWSSI(I)=3;
    ELSE IF STWSSI(I-1)=9 THEN STWSSI(I)=9;
  END;

  * STRING STWDI;
  IF STWDI(I)=. THEN DO;
    IF STWDI(I-1) IN (2,3) THEN STWDI(I)=3;
    ELSE IF STWDI(I-1)=9 THEN STWDI(I)=9;
  END;

  * STRING STWCOMB;
  IF STWCM(I)=. THEN DO;
    IF STWCM(I-1) IN (2,3) THEN STWCM(I)=3;
    ELSE IF STWCM(I-1)=9 THEN STWCM(I)=9;
  END;

END;
RUN;

PROC CONTENTS DATA=STWCM.STWCM;
TITLE "CONTENTS OF STWCM";
RUN;

PROC FREQ DATA=STWCM.STWCM;
%DO MM=1 %TO &TOTMOS.;
  TABLES STWCM&&YY&MM / MISSING OUT=STWCMFQ.STWCM&&YY&MM;
  TABLES STWDI&&YY&MM / MISSING OUT=STWCMFQ.STWDI&&YY&MM;
  TABLES STWSSI&&YY&MM / MISSING OUT=STWCMFQ.STWSSI&&YY&MM;
%END;

RUN;

%MEND;
%START;
ENDSAS;
/*****
/* SENDING EMAIL FROM HOST USING IEBGENER
/*****
//SENDNOTE EXEC PGM=IEBGENER
```

Appendix A.162

JCL/SAS Code: STWCM

```
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
DATA
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>
TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 STWCM COMPLETE
```

## Appendix A.163

JCL/SAS Code: BFWDI

```

//#3590BFD JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21 CONSTRUCTION 50214
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.STW.PRDLIB(BFWDI)
//*
//* PURPOSE: CREATE MONTHLY BENEFITS FORGONE FOR WORK FOR SSDI
//*
//* PROGRAMMER:
//* SVETLANA BRONNIKOV
//* MPR PHONE 202 264-4221
//* E-MAIL SBRONNIKOV@MATHEMATICA-MPR.COM
//*****
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//STWCM DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.STWCM.SA.V1,DISP=SHR
//DEMO DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//MBR DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.MBRREST.SA.V1,DISP=SHR
//BFWDI DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFWDI.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

/* STEP TO ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA */
%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */

%MACRO START;
DATA BFWDI.BFWDI(KEEP=SSN BFW:);
MERGE STWCM.STWCM(IN=INSTWCM KEEP=SSN STW:)
      DEMO.DMSTW(KEEP=SSN DOBBEST DODBEST FRADATE)
      MBR.MBR(KEEP=SSN LAF: DUED: DUEO: DROP=LAF)
;
BY SSN;
IF INSTWCM;

* SET COLA VALUES
  HTTPS://WWW.SSA.GOV/NEWS/COLA/;
%INCLUDE 'OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(COLA)';

* INPUT ARRAYS;

```

## Appendix A.163

JCL/SAS Code: BFWDI

```

ARRAY STWDI (&BEGYR.:&ENDYR.,1:12) %DO YEAR=&BEGYR %TO &ENDYR;
    %LET YR=%SUBSTR(&YEAR.,3,2);
    STWDI&YR.01-STWDI&YR.12
    %END;;
ARRAY LAF (&BEGYR.:&ENDYR.,1:12) %DO YEAR=&BEGYR %TO &ENDYR;
    %LET YR=%SUBSTR(&YEAR.,3,2);
    LAF&YR.01-LAF&YR.12
    %END;;
ARRAY DUED (&BEGYR.:&ENDYR.,1:12) %DO YEAR=&BEGYR %TO &ENDYR;
    %LET YR=%SUBSTR(&YEAR.,3,2);
    DUED&YR.01-DUED&YR.12
    %END;;
ARRAY DUEO (&BEGYR.:&ENDYR.,1:12) %DO YEAR=&BEGYR %TO &ENDYR;
    %LET YR=%SUBSTR(&YEAR.,3,2);
    DUEO&YR.01-DUEO&YR.12
    %END;;

* OUTPUT ARRAYS;
ARRAY BFWDI (&BEGYR.:&ENDYR.,1:12) %DO YEAR=&BEGYR %TO &ENDYR;
    %LET YR=%SUBSTR(&YEAR.,3,2);
    BFWDI&YR.01-BFWDI&YR.12
    %END;;
ARRAY BFWDI_PRIM (&BEGYR.:&ENDYR.,1:12) %DO YEAR=&BEGYR %TO &ENDYR;
    %LET YR=%SUBSTR(&YEAR.,3,2);
    BFWDI_PRIM&YR.01-BFWDI_PRIM&YR.12
    %END;;
ARRAY BFWDI_DEPEND (&BEGYR.:&ENDYR.,1:12)
    %DO YEAR=&BEGYR %TO &ENDYR;
    %LET YR=%SUBSTR(&YEAR.,3,2);
    BFWDI_DEPEND&YR.01-BFWDI_DEPEND&YR.12
    %END;;

* SET DATE TO STOP CALCULATING: END OF DAF YEAR, MONTH BENE TURNS FRA,
  OR MONTH BENE DIES;
STOPLOOP=MIN("31DEC&ENDYR."D,DODBEST,FRADATE);

* CALCULATE BFWDI
  BFWDI FOR BENES IN STW IS THE BENEFIT AMOUNT DUE FROM THE MOST
  RECENT PREVIOUS CURRENT PAY MONTH;
DO YR=1994 TO YEAR(STOPLOOP);

* WHEN STARTING A NEW YEAR ADJUST THE PREVIOUS DI PYAMENTS BY THE
  COLA FOR THIS YEAR;
PREVDUED=PREVDUED*(1+COLA(YR));

```



## Appendix A.163

JCL/SAS Code: BFWDI

```
PREVDUEO=PREVDUEO*(1+COLA(YR));

DO MO=1 TO 12 WHILE(STOPLOOP>MDY(MO,1,YR));

  * CALCULATE DI BENEFITS FORGONE FOR WORK (BFFW);
  IF STWDI(YR,MO) IN (1,2,3) THEN DO;
    BFWDI_PRIM(YR,MO)=PREVDUED;
    BFWDI_DEPEND(YR,MO)=PREVDUEO;
    BFWDI(YR,MO)=SUM(PREVDUED,PREVDUEO);
  END;

  * IF BENE IS IN CURRENT PAY THEN SET THE PREVIOUS DI BENEFITS PAID
  TO THE CURRENT DI BENEFIT PAID AMOUNTS;
  IF SUBSTR(LAF(YR,MO),1,1) IN ('C','E') THEN DO;
    PREVDUED=DUED(YR,MO);
    PREVDUEO=DUEO(YR,MO);
  END;
END; * END MONTH LOOP;
END; * END YEAR LOOP;

RUN;

%MEND;
%START;

PROC CONTENTS DATA=BFWDI.BFWDI;
TITLE "CONTENTS OF BFW";
RUN;

PROC MEANS DATA=BFWDI.BFWDI;
VAR BFW;
RUN;
/*****
/* SENDING EMAIL FROM HOST USING IEBGENER
/*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
DATA
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>
```

Appendix A.163

JCL/SAS Code: BFWDI

TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>  
SUBJECT: BFWDI COMPLETE

Appendix A.164

JCL/SAS Code: BFWDEEM

```
//#3590STD JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21 CONSTRUCTION 50214
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.STW.PRDLIB(BFWDEEM)
//*
//* PURPOSE: CALCULATE SCWF DEEMED (NONEARNED INCOME)
//*
//* PROGRAMMER:
//* SVETLANA BRONNIKOV
//* MPR PHONE 202 264-4221
//* E-MAIL SBRONNIKOV@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//STW  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.STWCM.SA.V1,DISP=SHR
//SSR  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIELIG.SA.V1,DISP=SHR
//DEMO DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//DEEM DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFWDEEM.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD *
```

OPTIONS NOCENTER COMPRESS=BINARY MPRINT OBS=MAX;

```
%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */
```

%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

%LET TOTMOS2=%EVAL(&TOTMOS.+12);

```
DATA DEEM.BFWDEEM;
MERGE STW.STWCM(IN=INSTWCM KEEP=SSN STW:)
      SSR.SSIELIG(KEEP=SSN UINC: FAMT: EICM: LIVF: CURCOMP:
                  FCI: BMF: DROP=BMF FCI)
      DEMO.DMSTW(KEEP=SSN DODBEST FRADATE)
;
```

## Appendix A.164

JCL/SAS Code: BFWDEEM

```

BY SSN;
IF INSTWCM;

* INPUT ARRAYS;
ARRAY STWSSI (-11:&TOTMOS2.) DUMMY1-DUMMY12
      %DO MM=1 %TO &TOTMOS.; STWSSI&&YY&MM %END;
      DUMMY13-DUMMY24
      ;
ARRAY UINC (*) %DO MM=1 %TO &TOTMOS.; UINC&&YY&MM %END;;
ARRAY EICM (*) %DO MM=1 %TO &TOTMOS.; EICM&&YY&MM %END;;
ARRAY FCI(*) %DO MM=1 %TO &TOTMOS.; FCI&&YY&MM %END;;
ARRAY BMF(*) %DO MM=1 %TO &TOTMOS.; BMF&&YY&MM %END;;

ARRAY NONEARN (-11:&TOTMOS2.) DUMMY1-DUMMY12
      %DO MM=1 %TO &TOTMOS.; NONEARN&&YY&MM %END;
      DUMMY13-DUMMY24
      ;
ARRAY NECALC (-11:&TOTMOS2.) DUMMY1-DUMMY12
      %DO MM=1 %TO &TOTMOS.; NECALC&&YY&MM %END;
      DUMMY13-DUMMY24
      ;
ARRAY DENOMA (*) %DO MM=1 %TO &TOTMOS.; DENOM&&YY&MM %END;;

* CACLUATE NON-EARNED INCOME: FEDERAL COUNTABLE EARNING IN THE CURRENT
  MONTH LESS THE COUNTABLE EARNED INCOME FROM THE BUDGET MONTH LESS THE
  COUNTABLE UNEARNED INCOME FROM THE BUDGET MONTH;
DO I=3 TO &TOTMOS.;
  IF BMF(I) NE . THEN DO;
    IF NECALC(I-BMF(I)) NE 1 AND STWSSI(I-BMF(I)) NE . THEN DO;
      NONEARN(I-BMF(I))=
        SUM(FLOOR(FCI(I)),SUM(EICM(I-BMF(I)),UINC(I-BMF(I)))*-1);
      IF .<NONEARN(I-BMF(I))<=1 THEN NONEARN(I-BMF(I))=0;
      NECALC(I-BMF(I))=1;
    END;
  END;
END;
DO I=1 TO &TOTMOS.;
  IF NECALC(I) NE 1 THEN DO;
    TEMPNE=0;
    DENOM=0;
    DO J=I-1 TO I-12 BY -1 UNTIL(DENOM=2);
      IF STWSSI(J)=0 AND NECALC(J)=1 THEN DO;
        TEMPNE=SUM(TEMPNE,NONEARN(J));
        DENOM=SUM(DENOM,1);
      END;
    END;
  END;

```

Appendix A.164

JCL/SAS Code: BFWDEEM

```
END;
DO J=I+1 TO I+12 UNTIL(DENOM=4);
  IF STWSSI(J)=0 AND NECALC(J)=1 THEN DO;
    TEMPNE=SUM(TEMPNE,NONEARN(J));
    DENOM=SUM(DENOM,1);
  END;
END;
NONEARN(I)=TEMPNE/DENOM;
DENOMA(I)=DENOM;
END;
END;

RUN;

PROC CONTENTS DATA=DEEM.BFWDEEM;
RUN;

%MEND;
%START;
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
DATA
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>
TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>
SUBJECT: SCWFDEEM COMPLETE
```

## Appendix A.165

JCL/SAS Code: BFWSSICM

```
//#3590STD JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21 CONSTRUCTION 50214.420
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.STW.PRDLIB(BFWSSICM)
//*
//* PURPOSE: CALCULATE SCWF BFWSSI
//*
//* PROGRAMMER:
//* SVETLANA BRONNIKOV
//* MPR PHONE 202 264-4221
//* E-MAIL SBRONNIKOV@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//*TEMP DD DSN=&&TEMPX,
//*      DISP=(NEW,DELETE,DELETE),
//*      UNIT=TSILO,VOL=(,,10)
//DEEM DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFWDEEM.SA.V1,DISP=SHR
//BFWDI DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFWDI.SA.V1,DISP=SHR
//BFW DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFWSSICM.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=BINARY MPRINT OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

%LET TOTMOS2=%EVAL(&TOTMOS.+12);

DATA BFW.BFWSSICM;
MERGE DEEM.BFWDEEM(IN=INSTWCM)
      BFWDI.BFWDI(KEEP=SSN BFWDI:)
;
BY SSN;
```

Appendix A.165

JCL/SAS Code: BFWSSICM

```

IF INSTWCM;

* FEDERAL BENEFIT RATES ARRAYS FOR INDIVIDUALS AND COUPLES
TABLE AT: HTTP://WWW.SOCIALSECURITY.GOV/OACT/COLA/SSIAMTS.HTML;

* SET SSI INDIVIDUAL AND COUPLE FBR MAX VALUES;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(FBRIN)";
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(FBRCP)";

* SET COLA VALUES;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(COLA)";

* INPUT ARRAYS;
ARRAY STWSSI (*) %DO MM=1 %TO &TOTMOS.; STWSSI&&YY&MM %END;;
ARRAY UINC (*) %DO MM=1 %TO &TOTMOS.; UINC&&YY&MM %END;;
ARRAY FAMT (*) %DO MM=1 %TO &TOTMOS.; FAMT&&YY&MM %END;;
ARRAY CURCOMP(*) $ %DO MM=1 %TO &TOTMOS.; CURCOMP&&YY&MM %END;;
ARRAY LIVF(*) $ %DO MM=1 %TO &TOTMOS.; LIVF&&YY&MM %END;;
ARRAY EICM (*) %DO MM=1 %TO &TOTMOS.; EICM&&YY&MM %END;;
ARRAY FCI(*) %DO MM=1 %TO &TOTMOS.; FCI&&YY&MM %END;;
ARRAY BMF(*) %DO MM=1 %TO &TOTMOS.; BMF&&YY&MM %END;;
ARRAY CFBMF(*) %DO MM=1 %TO &TOTMOS.; CFBMF&&YY&MM %END;;
ARRAY NONEARN (*) %DO MM=1 %TO &TOTMOS.; NONEARN&&YY&MM %END;;
ARRAY BFWDI (*) %DO MM=1 %TO &TOTMOS.; BFWDI&&YY&MM %END;;

* OUTPUT ARRAYS;
ARRAY BFWSSI (*) %DO MM=1 %TO &TOTMOS.;
      BFWSSI&&YY&MM
      %END;;
ARRAY BFWCM (*) %DO MM=1 %TO &TOTMOS.;
      BFWCM&&YY&MM
      %END;;
ARRAY FBR (*) %DO MM=1 %TO &TOTMOS.; FBR&&YY&MM %END;;

* CACULATE BFW;
* SET DATE TO STOP CALCULATING: END OF DAF YEAR, MONTH BENE TURNS FRA,
  OR MONTH BENE DIES;
STOPLOOP=INTCK('MONTHS','01JAN1994'D,
      MIN(DODBEST,FRADATE,"31DEC&ENDYR."D))+1;

* CALCULATE SSI BENEFITS FOREGONE FOR BENES RECEIVING BENEFITS;

* STRING INCOME AND ADJUST THE PREVIOUS YEARS INCOME;
DO I=2 TO DIM(STWSSI);
  * SET YEAR BEING PROCESSED. I REPRESENTS THE MONTH OF THE DAF

```

Appendix A.165

JCL/SAS Code: BFWSSICM

(MONTH 1 IS JAN 1994, MONTH 2 IS FEB 1994, MONTH 25 JAN 1996, ETC.)  
 TO DETERMINE THE YEAR THE CEIL FUNCTION IS APPLIED TO THE RESULTS  
 OF I DIVIDED BY 12 (CEIL RETURNS THE SMALLEST INTEGER THAT IS  
 GREATER THAN OR EQUAL TO THE ARGUMENT) THEN 1993 IS ADDED. SOMONTH  
 13 WOULD BE CEIL(13/12)=2+1993 OR 1995, MONTH 13 FALLS IN 1995;  
 LOOPYR=CEIL(I/12)+1993;

\* CALCULATE THE COUNTERFACTUAL BUDGET MONTH FACTOR (BMF);  
 CFBMF(I)=BMF(I);  
 IF STWSSI(I) IN (1,2,3) THEN DO;  
     IF CFBMF(I-1)=. THEN CFBMF(I)=0;  
     ELSE CFBMF(I)=MIN(2,CFBMF(I-1)+1);  
 END;

\* IF CURRENT MONTH STW IS 2 OR 3 THEN SET THE INCOME THIS MONTH TO  
 THAT OF THE PREVIOUS MONTH;  
 IF STWSSI(I) IN (2,3) THEN DO;

\* IF THIS MONTH IS JAN THEN ADJUST THE INCOME FROM THE PREVIOUS  
 MONTH BY COLA AND SET THIS MONTH TO THAT ADJUST AMOUNT;  
 IF (I-1)/12 = INT((I-1)/12) THEN DO;  
     UINC(I) =UINC(I-1)\*(1+COLA(LOOPYR));  
     EICM(I) =EICM(I-1)\*(1+COLA(LOOPYR));  
     NONEARN(I)=NONEARN(I-1)\*(1+COLA(LOOPYR));  
     FCI(I) =FCI(I-1)\*(1+COLA(LOOPYR));  
 END;

\* IF THIS MONTH IS NOT JAN THEN SET THE INCOME FOR THIS MONTH TO  
 THAT OF THE PREVIOUS MONTH;  
 ELSE DO;  
     UINC(I) =UINC(I-1);  
     EICM(I) =EICM(I-1);  
     FCI(I) =FCI(I-1);  
     NONEARN(I)=NONEARN(I-1);  
 END;

END; \* END STW IN 2 OR 3 LOOP;  
 END; \* END STRINGING INCOME LOOP;

\* CREATE BFW. THE LOOP STARTS IN MAR 1994 BECAUSE A 2 MONTH LOOK BACK  
 PERIOD IS REQUIRED. THIS RESULTS IN HAVING NO BFWSSI VALUES IN JAN &  
 FEB 1994.;  
 DO I=3 TO STOPLOOP;

\* SET YEAR BEING PROCESSED IN THIS EXECUTION OF THE LOOP;



## Appendix A.165

JCL/SAS Code: BFWSSICM

```
LOOPYR=CEIL(I/12)+1993;
```

```
* FBR FOR COUPLE IS MAXIMUM SSI COUPLE PAYMENT DIVIDED BY 2
  FBR FOR INDIVIDUAL IS MAXIMUM SSI INDIVIDUAL PAYMENT;
IF CURCOMP(I)='6' THEN FBR(I)=FBRCP(LOOPYR)/2;
ELSE FBR(I)=FBRIN(LOOPYR);
```

```
* BFW IS CALCULATED ONLY IN MONTHS WERE THE STW IS 0, 1, 2, OR 3.
  HOWEVER IF THERE IS NOT EARNED INCOME ONE MONTH PRIOR THEN EITHER
  THERE WAS NONRECURRING EARNED INCOME IN THE BM THAT DOES NOT AFFECT
  THE CURRENT MONTH, OR THERE WAS NO EARNED INCOME IN THE BM. IN
  EITHER CASE, THE BFWSSI VALUE IS 0. IF THERE WAS INCOME IN THE MONTH
  PRIOR THEN THE EICM FROM THE BM-1 IS THE BFWSSI VALUE;
IF STWSSI(I) IN (4,8,9,.) THEN BFWSSI(I)=0;
ELSE IF STWSSI(I)=0 AND UINC(I)>=FBR(I) THEN DO;
  BFWSSI(I)=0;
  STSUPONLY=1;
END;
```

```
* CREATE BFW FOR BENES IN CURRENT PAY;
ELSE IF STWSSI(I)=0 THEN DO;
  IF BMF(I)=. THEN BFWSSI(I)=.B;
  ELSE BFWSSI(I)=EICM(I-BMF(I));
END;
```

```
* CREATE BFW SSI FOR BENES WHO ARE SUSPENDED OR TERMINATED
  FOR WORK (STWSSI(I)=1/2/3) ;
ELSE BFWSSI(I)=SUM(FBR(I),UINC(I-CFBMF(I))*-1,NONEARN(I-CFBMF(I))*-1);
```

```
END; * END CREATE BFW LOOP;
```

```
* BFWSSI LOGICALLY CANNOT BE LESS THAN 0, GREATER THAN FBR, OR WHEN OFF
  THE ROLLS FOR WORK CANNOT BE 0. IF THE ALGORITHM HAS RESULTED IN
  THESE CASES THEN SET THE BFWSSI TO A SPECIAL MISSING;
DO I=3 TO DIM(BFWSSI);
IF BFWSSI(I)<0 OR BFWSSI(I)>FBR(I) OR
  (STWSSI(I) IN (1,2,3) AND BFWSSI(I)=0)
THEN BFWSSI(I)=.M;
```

```
* WHEN A BENE IS LIVING IN A MEDICAL FACILITY AND MEDICAID PAYS MORE
  THAN 50 PERCENT OF EXPENSES (LIVF=D) THEN THE MAX SSI BENEFIT IS $30
  SO THE BFWSSI CAN BE NO HIGHER THAN THAT;
IF LIVF(I)='D' THEN DO;
  IF BFWSSI(I) NE . THEN BFWSSI(I)=MIN(30,BFWSSI(I));
END;
```

Appendix A.165

JCL/SAS Code: BFWSSICM

```
END;

* CALCULATE COMBINED BFW;
DO I=1 TO &TOTMOS.;
  BFWCM(I)=SUM(BFWDI(I),BFWSSI(I));
END;

RUN;

PROC CONTENTS DATA=BFW.BFWSSICM;
RUN;

%MEND;
%START;
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
DATA
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>
TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 BFWSSICM COMPLETE
```

## Appendix A.166

JCL/SAS Code: SPBFW

```
//#3590SPB JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//*
//* *-----*
//* | THE MACRO SPLIT PROGRAM           |
//* *-----*
//*
//*
//* CREATED BY: SERGE LUKASHANETS, 5/26/2016
//* MODIFIED BY: SVETLANA BRONNIKOV 11/2/2022
//* E-MAIL:   SBRONNIKOV@MATHEMATICA-MPR.COM
//*          SVETLANA.BRONNIKOV@SSA.GOV
//*****
// SET REG='64M'
//*
//JS010 EXEC SAS9,
//      WORK='200000,100000', F10
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
//***DATA FOR SPLIT
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFWSSICM.SA.V1,DISP=SHR
//***OUT YEARLY FILES
//OUT94 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW1994.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT95 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW1995.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT96 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW1996.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT97 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW1997.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT98 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW1998.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT99 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW1999.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT00 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2000.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
```

## Appendix A.166

JCL/SAS Code: SPBFW

```
//      UNIT=TSILO,VOL=(,,10)
//OUT01 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2001.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT02 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2002.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT03 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2003.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT04 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2004.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT05 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2005.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT06 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2006.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT07 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2007.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT08 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2008.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT09 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2009.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2010.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT11 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2011.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2012.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT13 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2013.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT14 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2014.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT15 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2015.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
```

## Appendix A.166

JCL/SAS Code: SPBFW

```
//      UNIT=TSILO,VOL=(,,10)
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2016.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2017.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2018.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2019.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2020.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFW2021.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT OBS=MAX;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(MSPLIT)";

%OUTP(IN1.BFWSSICM,OUT,BFW,1994,2021,SSN);

PROC CONTENTS DATA=OUT94.BFW;RUN;
PROC CONTENTS DATA=OUT95.BFW;RUN;
PROC CONTENTS DATA=OUT96.BFW;RUN;
PROC CONTENTS DATA=OUT97.BFW;RUN;
PROC CONTENTS DATA=OUT98.BFW;RUN;
PROC CONTENTS DATA=OUT99.BFW;RUN;
PROC CONTENTS DATA=OUT00.BFW;RUN;
PROC CONTENTS DATA=OUT01.BFW;RUN;
PROC CONTENTS DATA=OUT02.BFW;RUN;
PROC CONTENTS DATA=OUT03.BFW;RUN;
PROC CONTENTS DATA=OUT04.BFW;RUN;
PROC CONTENTS DATA=OUT05.BFW;RUN;
PROC CONTENTS DATA=OUT06.BFW;RUN;
PROC CONTENTS DATA=OUT07.BFW;RUN;
PROC CONTENTS DATA=OUT08.BFW;RUN;
PROC CONTENTS DATA=OUT09.BFW;RUN;
PROC CONTENTS DATA=OUT10.BFW;RUN;
PROC CONTENTS DATA=OUT11.BFW;RUN;
PROC CONTENTS DATA=OUT12.BFW;RUN;
```

Appendix A.166

JCL/SAS Code: SPBFW

```
PROC CONTENTS DATA=OUT13.BFW;RUN;
PROC CONTENTS DATA=OUT14.BFW;RUN;
PROC CONTENTS DATA=OUT15.BFW;RUN;
PROC CONTENTS DATA=OUT16.BFW;RUN;
PROC CONTENTS DATA=OUT17.BFW;RUN;
PROC CONTENTS DATA=OUT18.BFW;RUN;
PROC CONTENTS DATA=OUT19.BFW;RUN;
PROC CONTENTS DATA=OUT20.BFW;RUN;
PROC CONTENTS DATA=OUT21.BFW;RUN;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
DATA
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>
TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>
SUBJECT: SPBFW COMPLETE
BFW SPLIT TO ANNUAL BFW COMPLETE.
```

## Appendix A.167

JCL/SAS Code: 1\_import\_new\_payments

```

/* ----- */
/* Name:      1_import_new_payments.sas          */
/* Date Created: May 26 2011 Miriam Loewenberg    */
/* Modified:   6/11/2013 Xiao Barry for DAF12     */
/*           5/19/2014 Patsy Lloyd for DAF13      */
/*           5/14/2015 Miaomaio Shen for DAF14   */
/*           7/05/2016 Serge Lukashanets for DAF15
/*           7/30/2018 Brittany Colip for DAF17   */
/*           8/06/2019 Serge Lukashanets for DAF18
/*           7/15/2020 Brittany Colip for DAF19   */
/*           9/2/2021 Serge Lukashanets for DAF20
/*           7/21/2022 Alex Bryce for DAF21
/* Purpose:   Produce a cleaned data file on EN level with:
/*           1) Compressed SSN and DUNS with no dashes
/*           2) Trim EN name
/*           3) Drop the total records with a WHERE clause
/*           4) Correct the triggering event (MONTH_S)
/* ----- */

```

```
options mlogic mprint symbolgen;
```

```
***** Set master macro variables *****;
```

```
%LET CurrDAF = 21;
```

```
%LET PrevDAF = 20;
```

```
*** Task Folder Paths ***;
```

```
%LET PrevDAF_Root = M:\DAF&PrevDAF.\TASK 15 Create Payments Component;
```

```
%LET CurrDAF_Root = M:\DAF&CurrDAF.\TASK 15 Create EN Payments Component;
```

```
**** Folder Path for Payment Input Files on the M: Drive ****;
```

```
%LET Payment_Input_Folder = M:\payment file with PII ;
```

```
**** Name of Input File for Current DAF ****;
```

```
%LET CurrDAF_Input_Name = EN Cumulative Payment Report July 1 2021 to present (7-19-22).xlsx ;
```

```
**** Name of the sheet from the input file to read-in ****;
```

```
%LET Sheet_Name = DAF_EN_Cumulative_Summary_PMTFL;
```

```
libname inlib "&CurrDAF_Root.\Data\PII\InputData"; /*Current lib to store read-in files*/
```

```
libname inter "&CurrDAF_Root.\Data\PII\IntermediateData"; /*Path to store Output from this program*/
```

```
libname intero "&PrevDAF_Root.\Data\PII\IntermediateData" access = readonly; /*Path where output from this program last year was stored*/
```

```
options mprint symbolgen;
```

## Appendix A.167

JCL/SAS Code: 1\_import\_new\_payments

```
*** Import the new Payments File ***;
PROC IMPORT OUT= inlib.ENPAY
  DATAFILE= "&Payment_Input_Folder.\&CurrDAF_Input_Name."
  DBMS=xlsx REPLACE;
      SHEET = "&Sheet_Name.";
  GETNAMES=YES;
  DATAROW=2;
RUN;

*** Print out the proc contents of the read-in and a few observations ***;
* You will want to review this and decide on which variables to keep/drop and rename;
* Compare this with the same file from the previous year to determine what needs to be changed;
ODS RTF FILE="&CurrDAF_Root.\Data\Contents\CONTENTS_ORIG.RTF";
TITLE "CONTENTS OF INPUT DATASET";
PROC CONTENTS DATA=inlib.ENPAY; RUN;
TITLE;
ODS RTF CLOSE;

title "Review this proc print to see if data is consistent with previous year";
PROC PRINT DATA=inlib.ENPAY (OBS=20);
RUN;
title;

PROC FORMAT;
value $ m3n 'Jan'=1
           'Feb'=2
           'Mar'=3
           'Apr'=4
           'May'=5
           'Jun'=6
           'Jul'=7
           'Aug'=8
           'Sep'=9
           'Oct'=10
           'Nov'=11
           'Dec'=12
;
run;

** Prep the new data so that it is in the same form as previous year;
** Using code from Serge from a different task to update for DAF19;
* After reviewing, set variables to keep and rename statements;
%LET KeepVars = COSSN BeneficiaryName DUNS ENET_NM TKT_PMT_TYP
                TKT_PAID_AMT TKT_PMT_DT CTL_DT MLSTON_NUM ACTL_PMT_NUM;
```



## Appendix A.167

JCL/SAS Code: 1\_import\_new\_payments

```

%LET RenameVars = %STR(ENET_NM = EN
                        BeneficiaryName = BENE_NAME
                        TKT_PMT_TYP = Pay_Type
                        TKT_PMT_DT = DATE_PAYMENT
                        TKT_PAID_AMT = Pay_Amt
                        );

DATA ENPAY (DROP = COSSN DUNS PAY_AMT
            RENAME = (COMPRESS_SSN = SSN
                     COMPRESS_DUNS = DUNS));
LENGTH COMPRESS_SSN COMPRESS_DUNS $9;
SET inlib.ENPAY (KEEP = &KeepVars.
                RENAME=(&RenameVars.)
                WHERE=(COSSN NE ' ')); *Drop the total records;

* Convert PAY_AMT to numeric ;
AMOUNT=PAY_AMT;
FORMAT AMOUNT DOLLAR21.2;

* Compress the dashes in the SSN and DUNS ;
COMPRESS_SSN = LEFT(TRIM(COMPRESS(COSSN,'-')));
COMPRESS_DUNS = LEFT(TRIM(COMPRESS((PUT(DUNS,9.)),'-')));

* Trim the EN name;
EN = LEFT(TRIM(EN));

* Variable MONTH_YR in DAF18 is provided as SAS date *;
* In DAF19 this variable is called CTL_DT;
TRIGGER_DATE = CTL_DT;

* In DAF19, the Pay_Num_Sort and Pay_Num variables were not provided ;
* Borrowing code from Serge from another task to create these vars;

if MLSTON_NUM>0 then Pay_Num_for_Sort="M";
if ACTL_PMT_NUM>0 then Pay_Num_for_Sort="O";

/*based on DAF rule - IF PAY_NUM = "OO" THEN PAY_NUM = "M0"*/

if MLSTON_NUM=0 & ACTL_PMT_NUM=0 then Pay_Num_for_Sort="M";
Pay_Num=Pay_Num_for_Sort;

RUN;

title "Check Derivation of Pay_Num and Pay_Num_for_Sort";
proc freq data=ENPAY;

```

## Appendix A.167

JCL/SAS Code: 1\_import\_new\_payments

```
tables Pay_Num*Pay_Num_for_Sort
      MLSTON_NUM*ACTL_PMT_NUM*Pay_Type*Pay_Num*Pay_Num_for_Sort/ list missing;
run;

***** Adding DAF19 *****;
* We need to recode the Pay_Type variable;
* If Pay_Type is B or M, code this to "MO". If it is O, code this as "OO";

title "Check Initial Coding of Pay_Type";
proc freq data = ENPAY;
  tables Pay_Type/missing;
run;

* Apply recode;
data ENPAY;
  set ENPAY;
  format Pay_Type_Recode $2.;
  if Pay_Type in ("B", "M") then Pay_Type_Recode = "MO";
  else if Pay_Type in ("O") then Pay_Type_Recode = "OO";
  drop Pay_Type;
  rename Pay_Type_Recode = Pay_Type;
run;

title "Check Pay_Type After Recoding applied";
proc freq data = ENPAY;
  tables Pay_Type/missing;
run;
title;

* Serge did some investigation in another project about duplicates that are occurring;
* I am borrowing the logic that de-dupes based on names ;
proc sort data=ENPAY nodupkeys out = inter.ENPAY;
by SSN DUNS EN Pay_Type AMOUNT trigger_date date_payment MLSTON_NUM ACTL_PMT_NUM;
run;

title "Check a few observations of cleaned ENPAY file (before combining with old file)";
PROC PRINT DATA=inter.ENPAY (OBS=25);
RUN;

*combine to Previous DAF;

data old;
```

## Appendix A.167

JCL/SAS Code: 1\_import\_new\_payments

```
set intero.ENPAY_FIN;
run;
```

```
* Serge also did some investigation into duplicates in the old file, but it appears de-duplicaton was not applied;
title "Check on Pay_Num_For_Sort in Previous DAF EN Payment File";
proc freq data=old;
tables Pay_Num_for_Sort / list missing;
run;
```

\* Serge also had some checks on a specific name, but it does not appear any further manipulation was performed after this check;

```
*** Combine New to Old data ***;
data new;
set inter.ENPAY;
run;
```

```
data both;
length EN $40. BENE_NAME $100.;
set old (in=a /*Rename removed -- this was specific to DAF17*/)
      new (in=b)
      ;
if a then flag=1;
if b then flag=2;
```

```
run;
```

```
title "Some Checks on Stacked Old and New EN Payments file";
proc means data=both min max;
var flag;
class pay_num;
run;
```

```
proc freq data=both;
tables Pay_Num_for_Sort*flag / list missing;
run;
```

```
*check overlap;
proc sql;
create table both2 as select *,n(distinct flag) as flags
from both group by SSN,pay_num,trigger_date,date_payment;
quit;
```

```
proc freq data=both2;
tables flags*flag
/ list missing;
```

## Appendix A.167

JCL/SAS Code: 1\_import\_new\_payments

```
run;

data both3;
set both2;
if flags=2 & flag=1 then delete;
run;

*OUTPUT;

data inter.ENPAY_FIN;
set both3;
drop flag flags ACTL_PMT_NUM CTL_DT MLSTON_NUM;
run;

ODS RTF FILE="&CurrDAF_Root.\Data\Contents\CONTENTS_VER.RTF";
TITLE "CONTENTS OF VERTICAL DATASET";
PROC CONTENTS DATA=inter.ENPAY_FIN OUT = inter.ENPAYVERCON ; RUN;
TITLE;
ODS RTF CLOSE;

*****
***** New check as of DAF19 *****
***** Compare Contents to previous DAF *****
*****

proc sort data = inter.ENPAYVERCON out = Con&CurrDAF.; by NAME; run;
proc sort data = intero.ENPAYVERCON out = Con&PrevDAF.; by NAME; run;

data ConComp;
  merge Con&CurrDAF. (keep= NAME Label rename=(Label = Label_&CurrDAF.) in= in&CurrDAF.)
    Con&PrevDAF. (keep = NAME Label rename=(Label = Label_&PrevDAF.) in = in&PrevDAF.)
  ;
  by NAME;
  if in&CurrDAF. then in_&CurrDAF. = 1;
  if in&PrevDAF. then in_&PrevDAF. = 1;
run;

title "Contents Comparison of Intermediate Vertical Payments File";
title2 "DAF&CurrDAF. vs. DAF&PrevDAF.";
proc print data = ConComp; run;
title; title2;
```

## Appendix A.168

JCL/SAS Code: 2\_payment\_person\_record

```

/* ----- */
/* NAME:      PAYMENT.SAS                */
/* DATE CREATED: MAY 28 2011            */
/* TASK       : PROCESS EN PAYMENT DATA 6FOR MINITRF          */
/* MODIFIED    JAN 18, 2012 JEREMY PAGE FOR TRF10              */
/*           August 22, 2012 XIAO BARRY FOR DAF11              */
/*           5/19/2014 PATSY LLOYD DAF13                       */
/*           5/14/2015 Miaomiao Shen for DAF14                 */
/*           7/5/2016 Serge Lukashanets for DAF15 */
/*           8/30/2017 Adele Rizzuto for DAF16                 */
/*           7/30/2018 Brittany Colip for DAF17 */
/*           8/06/2019 Serge Lukashanets for DAF18 */
/*           8/03/2020 Brittany Colip for DAF19 */
/*           9/02/2021 Serge Lukashanets for DAF20 */
/*           7/21/2022 Alex Bryce for DAF21 */
/* ----- */

***** Set master macro variables *****,
%LET CurrDAF = 21;
%LET PrevDAF = 20;

%LET PrevDAF_Root = M:\DAF&PrevDAF.\TASK 15 Create Payments Component;
%LET CurrDAF_Root = M:\DAF&CurrDAF.\TASK 15 Create EN Payments Component;

libname inter "&CurrDAF_Root.\Data\PII\IntermediateData";

libname out "&CurrDAF_Root.\Data\PII\OutputData";

libname outold "&PrevDaf_Root.\Data\PII\OutputData";

OPTIONS NOCENTER /*LS=max*/ MACROGEN MPRINT COMPRESS=BINARY
      PAGENO=1 LS=200 PS=80 FORMCHAR="|----|+|---+=|-\<>";
OPTIONS OBS=MAX;

%LET BEGYR=2002;
%LET ENDYR=2021; *Change the ENDYR for each DAF year;
%LET ENDMN=12;

%MACRO START;
/*INITIATE MONTH COUNTER, MOCNTER, TO 1*/
%LET MOCNTER=1;
/*LOOPING THROUGH EACH YEAR SPECIFIED ABOVE*/
%DO YEAR=&BEGYR. %TO &ENDYR.;

```

## Appendix A.168

JCL/SAS Code: 2\_payment\_person\_record

```
/*SETTING YR MACRO VAR TO LAST TWO DIGITS OF EACH YEAR*/
%LET YR=%SUBSTR(&YEAR.,3,2);

/*LOOP THROUGH MONTHS 1 TO 12 AND INCREMENT MONTH COUNTER*/
%DO M=1 %TO 12;
  %LET M=%SYSFUNC(PUTN(&M.,Z2.));
  %LET YY&MOCNTER=%EVAL(&YR)%EVAL(&M);
  %LET MOCNTER=%EVAL(&MOCNTER+1);
%END;
%END;
/*SET TOTAL MONTHS, TOTMOS, EQUAL TO MOCNTER - 1 TO GET
TOTAL MONTHS IN TIME SERIES*/
%LET TOTMOS=%EVAL(&MOCNTER-1);

/*USED TO PRING MAPPING TO LOG - CHECK DURING CODE REVIEW*/
%DO MM=1 % TO &TOTMOS;
  %PUT &MM. MAPS TO YYMM, &&YY&MM.;
%END;
%PUT TOTAL MONTHS = &TOTMOS.;

DATA PAYMENT O_PAYMENT M_PAYMENT ;
  SET INTER.ENPAY_FIN (DROP=BENE_NAME);

/* PAYMENTS WITH A PAY_NUM OF O0 (O,zero) ARE RECONCILIATION PAYMENTS FOR */
/* MISSED MILESTONE PAYMENTS. THEY ARE TRIGGERED BY THE PAYMENT OF THE FIRST */
/* OUTCOME PAYMENT SO THEREFORE ARE REFLECTED AS OUTCOME 0 PAYMENTS. WE */
/* RECODE THEM TO M0 SO THEY WILL FALL UNDER MILESTONE PAYMENTS INSTEAD OF */
/* OUTCOME. */

IF PAY_NUM = "O0" THEN PAY_NUM = "M0"; * by Xiao Barry on Sep.13th 2012;

LENGTH PAYMENT_TYPE $1;
PAY_TYPE_FOR_SORT = SUBSTR(PAY_NUM_FOR_SORT,1,1);

IF PAY_TYPE_FOR_SORT = 'B' THEN PAY_TYPE_FOR_SORT = 'M';

*correction for DAF18 - pay type = B;
IF PAY_TYPE = 'B' THEN PAY_TYPE = 'M';

IF PAY_TYPE in ('MO' 'M') AND PAY_TYPE_FOR_SORT = 'M' THEN PAYMENT_TYPE = 'M';
ELSE IF PAY_TYPE_FOR_SORT = 'O' THEN PAYMENT_TYPE = 'O';

/* A FEW RECORDS HAVE PAYMENT DATE EARLIER THAN THE TRIGGERING
MONTH. RECODE THE PAYMENT DATE TO THE MONTH DATE --
THERE IS NO RECORDS WITH PAYMENT DATE EARLIER THAN TRIGGERING MONTH IN DAF14 */
```

## Appendix A.168

JCL/SAS Code: 2\_payment\_person\_record

```
IF DATE_PAYMENT LT TRIGGER_DATE THEN DO; DATE_PAYMENT = TRIGGER_DATE; FLAG_DATE=1; END;
```

```
IF PAYMENT_TYPE = 'O' THEN OUTPUT O_PAYMENT;  
ELSE IF PAYMENT_TYPE = 'M' THEN OUTPUT M_PAYMENT;
```

```
OUTPUT PAYMENT;  
RUN;
```

```
PROC FREQ DATA=PAYMENT;  
  TABLES PAY_TYPE*PAY_TYPE_FOR_SORT*PAYMENT_TYPE/LIST MISSING;  
RUN;
```

```
/* SET ASIDE THE LIST OF EN'S FOR BENEFICIARIES TO MERGE ON LATER -  
  THERE IS MORE THAN 1 EN PER BENEFICIARY  
  WHERE SPLIT PAYMENTS ARE INDICATED THERE WILL BE GT 1 EN'S */
```

```
PROC SORT DATA=PAYMENT (KEEP=SSN EN DUNS) OUT=ENDATA NODUPKEY ;  
  BY SSN DUNS;  
RUN;
```

```
/* CRAETE MACRO VARIABLE INX */  
PROC SQL;  
  SELECT MAX(CT_EN) INTO :INX  
  FROM (SELECT SSN,COUNT(SSN) AS CT_EN  
        FROM ENDATA  
        GROUP BY SSN);  
QUIT;
```

```
/*PROCESS PAYMENTS DATA */  
DATA ENDATA (KEEP=SSN EN_NAME: DUNS_ID:);  
  SET ENDATA; BY SSN DUNS;
```

```
RETAIN  
%DO I = 1 %TO &INX;  
  EN_NAME&I  
  DUNS_ID&I  
%END;;
```

```
LENGTH  
%DO I = 1 %TO &INX;  
  EN_NAME&I  
%END;  
$80;
```

```
LENGTH
```

## Appendix A.168

JCL/SAS Code: 2\_payment\_person\_record

```
%DO I = 1 %TO &INX;
  DUNS_ID&I
%END;
$13;
```

```
ARRAY NAME (*) $ %DO I = 1 %TO &INX; EN_NAME&I %END;;
ARRAY NUMB (*) $ %DO I = 1 %TO &INX; DUNS_ID&I %END;;
```

```
IF FIRST.SSN THEN DO;
  %DO I = 1 %TO &INX;
    EN_NAME&I = ' ';
    DUNS_ID&I = ' ';
  %END;;
  REC = 1;
  NAME(REC) = EN;
  ***ADDING LEADING ZEROS TO DUNS_ID;
  *NUMB(REC) = PUT(DUNS,Z9.); /* IN 2014 DATA DUNS IS CHARACTER WITH LENGTH OF 9 */
END;
```

```
ELSE DO;
  REC + 1; /*IMPLICIT RETAIN */
  NAME(REC) = EN;
  ***ADDING LEADING ZEROS TO DUNS_ID;
  *NUMB(REC) = PUT(DUNS,Z9.); /* IN 2014 DATA DUNS IS CHARACTER WITH LENGTH OF 9 */
END;
```

```
IF LAST.SSN THEN OUTPUT;
RUN;
```

```
/*SORT DATA BY SSN AND TRIGGER_DATE */
PROC SORT DATA=O_PAYMENT; BY SSN TRIGGER_DATE; RUN;
PROC SORT DATA=M_PAYMENT; BY SSN TRIGGER_DATE; RUN;
```

```
/* THE FOLLOWING MACRO FIRST SUMS THE PAYMENT AMOUNTS FOR
PAYMENTS TRIGGERED ON THE SAME DATE. THEN IT CREATES
MONTHLY VARIABLES WHERE APPLICABLE AND OUTPUTS 1 RECORD
FOR EACH BENEFICIARY */
```

```
%MACRO SUMPAY (TYPE);
```

```
/* RESOLVE ALL DUPLICATE RECORDS */
DATA &TYPE._PAYMENT;
  SET &TYPE._PAYMENT; BY SSN TRIGGER_DATE;
```

```
RETAIN AMTPAID;
```



Appendix A.168

JCL/SAS Code: 2\_payment\_person\_record

```

IF FIRST.TRIGGER_DATE AND LAST.TRIGGER_DATE THEN DO;
  AMTPAID = AMOUNT;
END;

/* PROCESS REMAINING DUPLICATES */
ELSE DO;
  IF FIRST.TRIGGER_DATE THEN DO;
    AMTPAID = AMOUNT;
  END;
  /* SUM THE PAYMENTS ACROSS DUPLICATES (PAUL O'LEARY - AUG 2009) */
  ELSE DO;
    AMTPAID = SUM(AMTPAID,AMOUNT);
  END;
END;

IF LAST.TRIGGER_DATE THEN OUTPUT;

RUN;

PROC PRINT DATA=&TYPE._PAYMENT (OBS=25); VAR SSN
  TRIGGER_DATE PAY_NUM AMOUNT AMTPAID;;
  TITLE "AFTER FOR &TYPE DATA";
RUN;

DATA &TYPE._PAYMENT;
  SET &TYPE._PAYMENT;
  BY SSN;

/* PROCESS RECORDS TO RESOLVE PAYMENTS AND PRODUCE PERSON LEVEL DATA.
  - RANGE OF MONTHS 2002-2010 */

/* SET UP MONTHLY ARRAYS FOR PERSON LEVEL VARIABLES */

/*LENGTH %DO MM=1 %TO &TOTMOS; &TYPE._TRIGGERED&&YY&MM. %END; $10;*/
ARRAY PAYAMT{*} %DO MM=1 %TO &TOTMOS; EN_&TYPE.PAYAMT&&YY&MM. %END;;
ARRAY PAYDT{*} %DO MM=1 %TO &TOTMOS; EN_&TYPE.PAYDT&&YY&MM. %END;;
ARRAY PAYNUM{*} $ %DO MM=1 %TO &TOTMOS; EN_&TYPE.PAYNUM&&YY&MM. %END;;

/* SET UP ARRAYS FOR TEMPORARY VARIABLES */
ARRAY CYEAR (*) $ %DO MM=1 %TO &TOTMOS; C&&YY&MM. %END;;
ARRAY NYEAR (*) %DO MM=1 %TO &TOTMOS; N&&YY&MM. %END;;

RETAIN %DO MM=1 %TO &TOTMOS; EN_&TYPE.PAYAMT&&YY&MM.

```

## Appendix A.168

JCL/SAS Code: 2\_payment\_person\_record

```

                EN_&TYPE.PAYDT&&YY&MM.
                EN_&TYPE.PAYNUM&&YY&MM.
%END;;

/* INITIALIZE THE MONTHLY VARIABLES */
IF FIRST.SSN THEN DO;
  DO I = 1 TO &TOTMOS;
    PAYAMT(I) = . ;
    PAYDT(I) = . ;
    PAYNUM(I) = ' ';
  END;
END;

/* THE FOLLOWING ROUTINE CREATES MONTHLY REFERENCE DATES FROM
   JAN 2002 TO DEC 2017 */
J=1;
Y=&BEGYR;

DO I = 1 TO DIM(CYEAR); /* LOOP THROUGH ALL MONTH/YEAR REFERENCES */
  CYEAR(I) = PUT(Y,4.) || (PUT(J,Z2.));
  IF INT(I/12)=I/12 THEN DO; /*INCREMENT YEAR*/
    J=1;
    Y=Y+1;
  END;

  ELSE J=J+1; /*INCREMENT MONTH*/

  NYEAR(I)=INPUT(SUBSTR(CYEAR(I),1,4) || SUBSTR(CYEAR(I),5,2) || '01',YMMDD10.);

  IF TRIGGER_DATE = NYEAR(I) THEN DO;
    PAYAMT(I) = AMTPAID;
    PAYDT(I) = DATE_PAYMENT;
    PAYNUM(I) = PAY_NUM; *FOR BOTH TYPE, WE WANT TO RETAIN PHASE AND NUMBER OF THE MILESTONE
PAYMENTS;
  END;

END;

DROP PAY_TYPE PAY_NUM_FOR_SORT AMOUNT AMTPAID
  MONTH TRIGGER_DATE DATE_PAYMENT TRIGGERED_BY
  PAY_TYPE_FOR_SORT AMOUNT EN DUNS
  MONTH_YR YEAR PAY_NUM PAYMENT_TYPE I J Y;
DROP %DO MM = 1 %TO &TOTMOS; C&&YY&MM N&&YY&MM %END;;

IF LAST.SSN THEN OUTPUT;

```

Appendix A.168

JCL/SAS Code: 2\_payment\_person\_record

```
RUN;

%MEND SUMPAY;

%SUMPAY (M);
%SUMPAY (O);

/* MERGE THE DATASETS AND THE EN INFORMATION AND OUTPUT FINAL FILE */

DATA OUT.ENPAYMENTS ;

RETAIN SSN EN_NAME: DUNS_ID;;

MERGE M_PAYMENT (IN=INM)
      O_PAYMENT (IN=INO)
      ENDATA;
BY SSN;
HAS_M_DATA = INM;
HAS_O_DATA = INO;

RUN;

TITLE 'FINAL PERSON LEVEL EN PAYMENTS DATA';
TITLE2;
PROC FREQ; TABLES HAS_M_DATA*HAS_O_DATA/LIST MISSING; RUN;
PROC PRINT DATA=OUT.ENPAYMENTS (OBS=10) ; TITLE2 'FINAL OUTPUT RECORDS';
RUN;

ODS RTF FILE="M:\DAF19\TASK 15 Create Payments Component\Data\Contents\CONTENTS_HOR.RTF";
TITLE "CONTENTS OF HORIZONTAL DATASET";
PROC CONTENTS DATA=OUT.ENPAYMENTS OUT = OUT.ENPAYCON ; RUN;
TITLE;
ODS RTF CLOSE;

%MEND START;
%START;

/* NOTE: REMOVED OLD CODE PAYMENT_TYPE_OLD NOT EXIT ANYMORE */
PROC FREQ DATA= PAYMENT;
TITLE "CHECK FREQUENCIES";
TABLES FLAG_DATE /*PAYMENT_TYPE_OLD*PAYMENT_TYPE*//LIST MISSING;
RUN;

PROC PRINT DATA=PAYMENT;
TITLE "SAMPLE PRINT OF THOSE WITH FLAG_DATE=1";
```

## Appendix A.168

JCL/SAS Code: 2\_payment\_person\_record

```
WHERE FLAG_DATE=1;
RUN;
```

```
ODS RTF FILE="&CurrDAF_Root.\Data\Contents\CONTENTS_INTER.RTF";
TITLE1 "CONTENTS OF INTERMEDIATE DATASETS";
TITLE2 "CONTENTS OF PAYMENT DATASET";
PROC CONTENTS DATA=PAYMENT;
RUN;
```

```
TITLE2 "CONTENTS OF O_PAYMENT DATASET";
PROC CONTENTS DATA=O_PAYMENT;
RUN;
```

```
TITLE2 "CONTENTS OF M_PAYMENT DATASET";
PROC CONTENTS DATA=M_PAYMENT;
RUN;
TITLE;
ODS RTF CLOSE;
```

```
*****
***** New check as of DAF19 *****
***** Compare Contents to previous DAF *****
*****
```

```
proc sort data = out.ENPAYCON out = Con&CurrDAF.; by NAME; run;
proc sort data = outold.ENPAYCON out = Con&PrevDAF.; by NAME; run;
```

```
data ConComp;
  merge Con&CurrDAF. (keep= NAME Label rename=(Label = Label_&CurrDAF.) in= in&CurrDAF.)
    Con&PrevDAF. (keep = NAME Label rename=(Label = Label_&PrevDAF.) in = in&PrevDAF.)
  ;
  by NAME;
  if in&CurrDAF. then in_&CurrDAF. = 1;
  if in&PrevDAF. then in_&PrevDAF. = 1;
run;
```

```
title "Contents Comparison of Intermediate Horizontal Payments File";
title2 "DAF&CurrDAF. vs. DAF&PrevDAF.";
proc print data = ConComp;
where in_&CurrDAF. ~= 1 or in_&PrevDAF. ~= 1;
run;
title; title2;
```

## Appendix A.169

JCL/SAS Code: ENPAY\_UPLOAD

```
/******  
** Purpose: Upload the ENPAY horizontal and vertical file to the mainframe  
*   Upload their table contents to the mainframe  
*****/  
options comamid=tcp; %let SY56=SY56.sspf.ssa.gov; %let tcpsec=_prompt_; options remote=sy56.spawner; run;  
  
SIGNON ;  
LIBNAME LOCAL1 'M:\DAF21\TASK 15 Create EN Payments Component\Data\PII\OutputData';  
LIBNAME LOCAL2 'M:\DAF21\TASK 15 Create EN Payments Component\Data\PII\IntermediateData';  
RSUBMIT;  
  
LIBNAME ENPAY 'OPDR.TG.PRD.ETTW.#6502.DAF21I.ENPAY.SA.V1';  
LIBNAME ENPAYVER 'OPDR.TG.PRD.ETTW.#6502.DAF21I.ENPAYVER.SA.V1';  
OPTION COMPRESS=BINARY;  
  
*****;  
/* Use the firstobs and obs control only if needed. For example, if the first download */  
/* after the 499th record, then the code below will start downloading at the 500th */  
/* observation and go until it completes or crashes. */  
*****;  
PROC UPLOAD DATA=LOCAL1.ENPAYMENTS OUT=ENPAY.ENPAY (COMPRESS=BINARY);RUN;  
PROC UPLOAD DATA=LOCAL2.ENPAY_FIN OUT=ENPAYVER.ENPAYVER (COMPRESS=BINARY);RUN;  
  
ENDRSUBMIT;  
SIGNOFF;
```

Appendix A.170

JCL/SAS Code: FINPAYH

```
//#6502PHF JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT: BUILDING THE TICKET RESEARCH FILE (50214)
//*
//* PROJECT
//* DIRECTOR:  MATT URATO
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.ENPAY.PRDLIB(FINPAYH)
//*
//*
//* DATE:     10/05/2022
//* CONTACT:  ALEX.BRYCE@SSA.GOV
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//COM  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,
//  DISP=SHR
//IPAYH DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21I.ENPAY.SA.V1,
//  DISP=SHR
//OPAYH DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.ENPAY,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OPYHC DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.ENPAYCON.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10PAYH,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

%MACRO DOIT;

%LET DAFYY=21;

DATA OPAYH.ENPAY_FINAL;
  MERGE IPAYH.ENPAY (IN=ENPAY)
        COM.COMVARS
  ;
  BY SSN;
```

Appendix A.170

JCL/SAS Code: FINPAYH

```

IF ENPAY;

DROP DUNS_ID:
  EN_NAME:
  FLAG_DATE
  MONTH_YR_OLD
;
LABEL HAS_M_DATA="EN MILESTONE PAYMENTS GENERATED FLAG"
  HAS_O_DATA="EN OUTCOME PAYMENTS GENERATED FLAG"
  SSN="SOCIAL SECURITY NUMBER"
  FRADATE = "FULL RETIREMENT DATE"
  SSI&DAFY="SSI CURRENT PAY STATUS IN 20&DAFY."
  SSDI&DAFY="SSDI CURRENT PAY STATUS IN 20&DAFY."
  SEX = "GENDER (FROM DEMO FILE)"
  SSIELIG_FIRST="SSI EARLIEST ELIGIBILITY DATE (CONSTRUCTED)"
%DO YEAR=2002 %TO 20&DAFY;
%LET YY=%SUBSTR(&YEAR.,3,2);
%DO MO=101 %TO 112;
%LET MM=%SUBSTR(&MO,2,2);
  EN_MPAYDT&YY.&MM.="EN MILESTONE PAYMENT DATE &YY.&MM."
  EN_MPAYAMT&YY.&MM.="EN MILESTONE PAYMENT AMOUNT &YY.&MM."
  EN_MPAYNUM&YY.&MM.="EN MILESTONE PAYMENT NUMBER &YY.&MM."
  EN_OPAYDT&YY.&MM.="EN OUTCOME PAYMENT DATE &YY.&MM."
  EN_OPAYAMT&YY.&MM.="EN OUTCOME PAYMENT AMOUNT &YY.&MM."
  EN_OPAYNUM&YY.&MM.="EN OUTCOME PAYMENT NUMBER &YY.&MM."
%END;
%END;
;
RENAME
  HAS_M_DATA=EN_MPAYFLAG
  HAS_O_DATA=EN_OPAYFLAG
;
RUN;
%MEND DOIT;
%DOIT;

PROC CONTENTS DATA=OPAYH.ENPAY_FINAL OUT=OPYHC.ENPAYCON;
RUN;

DATA OUT10.P10PAYH;
SET OPAYH.ENPAY_FINAL;
IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9')) AND
  (SUBSTR(SSN,9,1) IN ('0', '5'));
RUN;

```

Appendix A.170

JCL/SAS Code: FINPAYH

```
PROC CONTENTS DATA=OUT10.P10PAYH; RUN;
```



Appendix A.171

JCL/SAS Code: FINPAYV

```

//#6502PVF JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT: BUILDING THE TICKET RESEARCH FILE (06979)
//*
//* PROJECT
//* DIRECTOR:  MATT URATO
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.ENPAY.PRDLIB(FINPAYV)
//*
//*
//* DATE:     10/05/2022
//* CONTACT:  ALEX.BRYCE@SSA.GOV
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//COM  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,
//  DISP=SHR
//IPAYV DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21I.ENPAYVER.SA.V1,
//  DISP=SHR
//OPAYV DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.ENPAYVER,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OPYVC DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.ENPYVCN.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10PAYV,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;
%MACRO DOIT;

%LET DAFYY=21;

PROC SORT DATA = IPAYV.ENPAYVER OUT = TEMP1.ENPAYVER;
BY SSN;
RUN;

```

## Appendix A.171

JCL/SAS Code: FINPAYV

```
DATA OPAYV.ENPAYVER_FINAL;
  MERGE TEMP1.ENPAYVER(IN=ENPAYVER
    RENAME=(PAY_TYPE=PYMT_TYPE
      BENE_NAME=BENE_NAME_PMNTS
      TRIGGER_DATE=MONTHDATE))
    COM.COMVARS
  ;
  BY SSN;
  IF ENPAYVER;

  DROP YEAR
    MONTH
    MONTH_YR MONTH_YR_OLD
    PAY_NUM_FOR_SORT
  ;
  LABEL SSN="SOCIAL SECURITY NUMBER"
    EN="EMPLOYMENT NETWORK NAME"
    DUNS="DUNS (DATA UNIVERSAL NUMBERING SYSTEM)"
    PYMT_TYPE="EN TICKET PAYMENT TYPE"
    PAY_NUM="EN TICKET PAYMENT NUMBER"
    BENE_NAME_PMNTS="BENEFICIARY'S NAME"
    MONTHDATE="EN TICKET PAYMENT TRIGGERING EVENT DATE"
    AMOUNT="EN TICKET PAYMENT AMOUNT"
    DATE_PAYMENT="EN TICKET PAYMENT MADE DATE"
    TRIGGERED_BY="EN TICKET PAYMENT TRIGGERED BY"
    SSIELIG_FIRST="SSI EARLIEST ELIGIBILITY DATE (CONSTRUCTED)"
    FRADATE = "FULL RETIREMENT DATE"
    SSI&DAFY="SSI CURRENT PAY STATUS IN 20&DAFY."
    SSDI&DAFY="SSDI CURRENT PAY STATUS IN 20&DAFY."
    SEX = "GENDER (FROM DEMO FILE)"
  ;
  RENAME EN=EN_NAME
    PYMT_TYPE=ENPAY_TYPE
    PAY_NUM=ENPAY_NUM
    MONTHDATE=ENPAY_TRIG_DATE
    AMOUNT=ENPAY_AMOUNT
    DATE_PAYMENT=ENPAY_DATE
  ;

  RUN;
  %MEND DOIT;
  %DOIT;

  PROC CONTENTS DATA=OPAYV.ENPAYVER_FINAL
```

Appendix A.171

JCL/SAS Code: FINPAYV

```
        OUT=OPYVC.ENPAYVER_CON;
RUN;

DATA OUT10.P10PAYV;
  SET OPAYV.ENPAYVER_FINAL;
  IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9')) AND
     (SUBSTR(SSN,9,1) IN ('0', '5'));
RUN;

PROC CONTENTS DATA=OUT10.P10PAYV; RUN;
```

Appendix A.172

JCL/SAS Code: 001\_LOAD\_VRRMS

```

*****
* PROJECT:   DAF2021 (50214)
* RESEARCHER:
* PROGRAM:   001_LOAD_VRRMS.sas
* PURPOSE:   READ IN THE LATEST NEW FORMAT VRRMS XLSX FILES
* PROGRAMMER: SERGE LUKASHANETS
* DESCRIPTION: THE PAYMENT AND CLAIM FILES ARE COMBINED WITH THE PREVIOUS DAF USING COSSN,
VR_CLM_SNUM, AND PMT_SEQ_NUM.
*****
OPTIONS MPRINT MACROGEN COMPRESS=BINARY NOCENTER /*NOLABEL*/ LINESIZE=MAX
        PAGESIZE=MAX REPLACE=YES OBS=MAX;

%let yr = 21;
%let prog_path = M:\DAF21\TASK 16 Create VRRMS Component\Version2\Programs;
%let prog_name=001_LOAD_VRRMS;

proc printto new
    log = "&prog_path.\&prog_name..log"
    print = "&prog_path.\&prog_name..lst";
run;

%LET OLD=M:\DAF20\TASK 16 Create VRRMS Component;
%LET DIR=M:\DAF21\TASK 16 Create VRRMS Component\Version2;

LIBNAME OUTLIB "&dir.\Data\SAS";
LIBNAME INOLD "&old.\Data\SAS" access = readonly;

/* Set the dataset names of the SAS claims and payments files that should be used from the previous DAF */
/* This is used when combining the new data and old data */
%LET Old_Claim_DSN = Vrhclaim_comb;
%LET Old_Pymnt_DSN = Vrpymnt_comb;

*****
* Load Excel file into SAS.
*****
%MACRO IMPORT_DT(FILENM=,SHEETNM=,DT=);
PROC IMPORT DATAFILE="&DIR.\Data\Excel\&FILENM..XLSX"
    OUT=OUTLIB.&DT. REPLACE
        DBMS=XLSX;
        SHEET="&SHEETNM.";
/* GETNAMES=YES;*/
RUN;

PROC CONTENTS DATA=OUTLIB.&DT.;
RUN;

```

Appendix A.172

JCL/SAS Code: 001\_LOAD\_VRRMS

```

* CHECK MISSING COSSN IN EACH DATASET *;
PROC SQL;
  CREATE TABLE &DT._CHECK_MISS1 AS
  SELECT SUM(CASE WHEN MISSING(COSSN) THEN 1 ELSE 0 END) AS CK_MISS_COSSN
  FROM OUTLIB.&DT.;
QUIT;

* CHECK MISSING VR_CLM_SNUM IN EACH DATASET *;
PROC SQL;
  CREATE TABLE &DT._CHECK_MISS2 AS
  SELECT SUM(CASE WHEN MISSING(VR_CLM_SNUM) THEN 1 ELSE 0 END) AS CK_MISS_SNUM
  FROM OUTLIB.&DT.;
QUIT;

* CHECK # OF OBS FOR EACH COSSN/VR_CLM_SNUM COMBINATION. *;
PROC SQL;
  CREATE TABLE &DT._CHECK AS
  SELECT CNT_COMBS
         ,COUNT(*) AS CNT_NUM_RECS
  FROM (SELECT COSSN
         ,VR_CLM_SNUM
         ,COUNT(COSSN) AS CNT_COMBS
  FROM OUTLIB.&DT.
  GROUP BY COSSN,VR_CLM_SNUM)
  GROUP BY CNT_COMBS
  ORDER BY CNT_COMBS;
QUIT;
%MEND IMPORT_DT;

%IMPORT_DT(FILENM = VRHCLAIM 10012020 to 12312021 - Copy,
           SHEETNM = DAF_VRHCLAIM,
           DT = VRHCLAIM);

%IMPORT_DT(FILENM = VRPAYMNT - 10012020 to 12312021,
           SHEETNM = DAF_VRPAYMNT,
           DT = VRPAYMNT);

```

```

/*****
****
*
*ADD TO PREVIOUS DAF VRRMS
*

```

\*\*\*\*\*  
\*\*\*\*\*/

## Appendix A.172

JCL/SAS Code: 001\_LOAD\_VRRMS

```

* DAF20 Claim file was missing leading zeros on SSN;
data PREV_CLAIM;
  set INOLD.&Old_Claim_DSN.(rename=(COSSN=COSSN_NUM));
  length COSSN $9.
  ;
  COSSN=put(COSSN_NUM,z9.);
  drop COSSN_NUM;
run;

proc compare base=OUTLIB.VRHCLAIM compare=PREV_CLAIM novalues;
run;
/* Listing of Common Variables with Conflicting Types

Variable      Dataset      Type Length Format  Informat Label
VR_CLNT_ENTRY_DT OUTLIB.VRHCLAIM Char 10 $10. $10. VR_CLNT_ENTRY_DT
                INOLD.VRHCLAIM_COMB Num 8 MMDDYY10. Vr Clnt Entry Dt
COSSN          OUTLIB.VRHCLAIM Char 9 $9. $9. COSSN
                INOLD.VRHCLAIM_COMB Num 8 BEST. Cossn
*/

proc compare base=OUTLIB.VRPAYMNT compare=INOLD.&Old_Pymnt_DSN. novalues;
run;

/***** Adding code to fix conflicting variable types *****/
* We will need to update this in the future.;
data VRPAYMNT;
  set OUTLIB.VRPAYMNT;
  if missing(cossn) then _miss_flag=1;
run;

proc freq data=VRPAYMNT;
  tables _miss_flag / list missing;
run;

proc sort data=VRPAYMNT nuniquekeys out=_a;by cossn VR_CLM_SNUM PMT_SEQ_NUM;run; /*no dups so far*/
proc sort data=VRPAYMNT nuniquekeys out=_a;by cossn VR_CLM_SNUM;run; /*normal - pairs due to
PMT_SEQ_NUM =1,2,3... */

/*combine */
data VRPAYMNT_both;
  set VRPAYMNT (in=a) INOLD.&Old_Pymnt_DSN. (in=b drop=WRKCS_UID_max);
  where strip(VR_PSTUS_CD)="P"; /*this restriction is used in VRRMS procession*/
  if a then _new=1;
  if b then _old=1;
  _PSTUS_TS = datepart(PSTUS_TS);

```

## Appendix A.172

JCL/SAS Code: 001\_LOAD\_VRRMS

```

run;

/*check duplicates*/
proc sort data=VRPAYMNT_both nuniquekeys out=_a;by cossn VR_CLM_SNUM;run;
proc sort data=VRPAYMNT_both nuniquekeys out=_a;by cossn VR_CLM_SNUM PMT_SEQ_NUM;run;
proc print data=_a (obs=40); run;

proc sql;
    create table VRPAYMNT_both_1 as select *,max(_new) as _new_old from VRPAYMNT_both group by
        cossn, VR_CLM_SNUM, PMT_SEQ_NUM;
quit;
proc sort data=VRPAYMNT_both_1 nuniquekeys out=_a;by cossn VR_CLM_SNUM PMT_SEQ_NUM;run;
proc print data=_a (obs=40); run;

PROC FREQ DATA=VRPAYMNT_both_1;
    TABLES _new*_old*_new_old*(VR_PSTUS_CD _PSTUS_TS)
        _old*_new*_new_old / LIST MISSING;
    FORMAT _PSTUS_TS YYMMN6.;
RUN;

proc freq data=OUTLIB.VRHCLAIM;
tables Entlt_Cd Vr_Pmt_Actn_Cd Vr_Sga_Cprd_Typ_Cd
/ list missing;
run;

data VRHCLAIM;
set OUTLIB.VRHCLAIM
    (rename=(Entlt_Cd=_Entlt_Cd
        /*Post_Empt_Prd_Endt=_Post_Empt_Prd_Endt
        Post_Empt_Prd_Stdt=_Post_Empt_Prd_Stdt*/
        Vr_Pmt_Actn_Cd=_Vr_Pmt_Actn_Cd
        Vr_Sga_Cprd_Typ_Cd=_Vr_Sga_Cprd_Typ_Cd
        Pmt_Ts=_Pmt_Ts
        ));
/*recode*/
if _Entlt_Cd ne "C" then Entlt_Cd=_Entlt_Cd;
if _Vr_Pmt_Actn_Cd ne "VVV" then Vr_Pmt_Actn_Cd=_Vr_Pmt_Actn_Cd;
if _Vr_Sga_Cprd_Typ_Cd ne "NN" then Vr_Sga_Cprd_Typ_Cd=_Vr_Sga_Cprd_Typ_Cd;

/*Post_Empt_Prd_Endt=put(left(put(_Post_Empt_Prd_Endt,MMDDYY10.)), $10.);
Post_Empt_Prd_Stdt=put(left(put(_Post_Empt_Prd_Stdt,MMDDYY10.)), $10.);*/

Pmt_Ts=datepart(_Pmt_Ts);
/*
if vtype(cossn) = 'C' then do;

```



## Appendix A.172

JCL/SAS Code: 001\_LOAD\_VRRMS

```

        cossn_num = input(cossn, best.);
        drop cossn;
        rename cossn_num = cossn;
    end;
/*
if vtype(pmt_prd_stdtd) = 'C' then do;
    pmt_prd_stdtd_num = input(pmt_prd_stdtd, 8.);
    drop pmt_prd_stdtd;
    rename pmt_prd_stdtd_num = pmt_prd_stdtd;
end;
*/
if vtype(vr_clnt_entry_dt) = 'C' then do;
    vr_clnt_entry_dt_num = input(vr_clnt_entry_dt, 8.);
    drop vr_clnt_entry_dt;
    rename vr_clnt_entry_dt_num = vr_clnt_entry_dt;
end;
run;

proc contents data = VRHCLAIM; run;

/*proc print data=VRHCLAIM (obs=10); var Post_Empt_Prd_Endtd _Post_Empt_Prd_Endtd; where
_Post_Empt_Prd_Endtd ne .; run;
proc print data=VRHCLAIM (obs=10); var Post_Empt_Prd_Stdtd _Post_Empt_Prd_Stdtd; where _Post_Empt_Prd_Stdtd
ne .; run;
*/
proc freq data=VRHCLAIM;
    tables          Entlt_Cd* _Entlt_Cd
                   Vr_Pmt_Actn_Cd* _Vr_Pmt_Actn_Cd
                   Vr_Sga_Cprd_Typ_Cd* _Vr_Sga_Cprd_Typ_Cd
    / list missing;
run;

/*test combine*/
data VRHCLAIM_both;
    /*length Post_Empt_Prd_Endtd Post_Empt_Prd_Stdtd $10.*/;
    set prev_claim (in=b) VRHCLAIM (in=a drop=_);
    if a then _new=1;
    if b then _old=1;
run;

/*same approach as for PAYMENT - the file should be unique by cossn VR_CLM_SNUM PMT_SEQ_NUM */

proc sort data=VRHCLAIM_both nuniquekeys out=_a; by cossn VR_CLM_SNUM PMT_SEQ_NUM; run; /*1.7K recs*/
proc print data=_a (obs=50); run;
/*will keep the newer - it's better populated*/

```

```
proc sql;
```

## Appendix A.172

JCL/SAS Code: 001\_LOAD\_VRRMS

```
        create table VRHCLAIM_both_1 as select *,max(_new) as _new_old from VRHCLAIM_both
        group by cossn, VR_CLM_SNUM, PMT_SEQ_NUM;
quit;
proc freq data=VRHCLAIM_both_1;
    tables _old*_new*_new_old / list missing;
run;

*****
*FINALIZE FOR PROCESSION                               *;
*****
data OUTLIB.VRHCLAIM_comb (drop=_:);
    set VRHCLAIM_both_1;
    if _new_old=1 & _new ne 1 then delete; /*drop the old rec from new-old comb*/
run;

data OUTLIB.VRPAYMNT_comb (drop=_:);
    set VRPAYMNT_both_1;
    if _new_old=1 & _new ne 1 then delete; /*drop the old rec from new-old comb*/
run;

proc printto;
run;
```

## Appendix A.173

JCL/SAS Code: 002\_PROCESS\_VRRMS

```

*****;
* PROJECT:   DAF21 (50214)
* RESEARCHER:
* PROGRAM:   002_PROCESS_VRRMS.sas
* PURPOSE:   PROCESS SAS VERSION OF THE VRRMS DATASETS
* PROGRAMMER: SERGE LUKASHANETS 08/19/2022
* MODIFICATION:
*****;
OPTIONS MPRINT MACROGEN COMPRESS=BINARY NOCENTER /*NOLABEL*/ LINESIZE=MAX
        PAGESIZE=MAX REPLACE=YES FORMCHAR='| -+++++++==Ý-/\<>*' OBS=MAX;

%let yr = 21;
%let prog_path = M:\DAF21\TASK 16 Create VRRMS Component\Version2\Programs;
%let prog_name = 002_PROCESS_VRRMS;

proc printto new
    log = "&prog_path.\&prog_name..log"
    print = "&prog_path.\&prog_name..lst";
run;

%LET DIR=M:\DAF21\TASK 16 Create VRRMS Component\Version2;

LIBNAME INLIB "&dir.\Data\SAS" access = readonly;
LIBNAME OUTLIB "&dir.\Data\Final";

/*****
*CREATE A FINDER AND SUBSET PAYMENT & CLAIM FILES
*****/
*STEP 1: READ PAYMENT FILE;

* CHECK VARIABLES IN VRPAYMNT.          *;
PROC FORMAT;
    VALUE MISSF
        . = "MISSING"
        OTHER = "NON-MISSING";
RUN;

proc contents data=inlib.vrpymnt_comb; run;
DATA VRPAYMNT (KEEP=COSSN In_COSSN PSTUS_TS TOT_DUE_AMT TOT_PAID_AMT
                VR_PSTUS_CD VOIDD_PMT_CD VR_CLM_SNUM PMT_SEQ_NUM);
SET INLIB.VRPAYMNT_COMB (RENAME=(PSTUS_TS=_PSTUS_TS));
/*if vtype(cossn) = 'C' then do;
    cossn_num = put(input(cossn, 9.), z9.);

```

## Appendix A.173

JCL/SAS Code: 002\_PROCESS\_VRRMS

```
drop cossn;
rename cossn_num = cossn;
end;*/

PSTUS_TS=DATEPART(_PSTUS_TS);
ln_COSSN = length(strip(COSSN));
if VOIDD_PMT_CD="Y" then delete;
if PSTUS_TS<"01MAR2017"D then delete;
if STRIP(VR_PSTUS_CD) ne "P" then delete;
RUN;

title "VRPAYMENT after subset";
PROC FREQ DATA=VRPAYMNT;
  TABLES ln_COSSN VR_PSTUS_CD PSTUS_TS VOIDD_PMT_CD VR_CLM_SNUM PMT_SEQ_NUM/ LIST MISSING;
  FORMAT PSTUS_TS YYMMN6. VR_CLM_SNUM MISSF.;
RUN;
title;

* IF THERE ARE MULTIPLE RECORDS BY COSSN AND VR_CLM_SNUM, THEN SUM  *;
* TOT_DUE_AMT AND TOT_PAID_AMT. *;

proc sort data=VRPAYMNT out=_a1 nouniquekeys; by COSSN VR_CLM_SNUM; run;
proc print data=_a1; run;

*STEP 2: READ CLAIM FILE;

data VRHCLAIM;
set INLIB.VRHCLAIM_COMB;
/*
format cossn_char $9.;
cossn_char = put(cossn, z9.);
drop cossn;
rename cossn_char = cossn;
*/

/*if vtype(cossn) = 'C' then do;
  cossn_num = put(input(cossn, 9.), z9.);
  drop cossn;
  rename cossn_num = cossn;
end;*/

/*if vtype(pmt_prd_stdtd) = 'C' then do;
  pmt_prd_stdtd_num = input(pmt_prd_stdtd, 8.);
  drop pmt_prd_stdtd;
  rename pmt_prd_stdtd_num = pmt_prd_stdtd;
```

## Appendix A.173

JCL/SAS Code: 002\_PROCESS\_VRRMS

```
end;*/

/*if vtype(vr_clnt_entry_dt) = 'C' then do;
  vr_clnt_entry_dt_num = input(vr_clnt_entry_dt, 8.);
  drop vr_clnt_entry_dt;
  rename vr_clnt_entry_dt_num = vr_clnt_entry_dt;
end;*/
RUN;

title "check COSSN";

proc sort data=VRHCLAIM out=_a1 nuniquekeys; by COSSN VR_CLM_SNUM; run;
PROC FREQ DATA=VRHCLAIM;
  TABLES VR_CLM_SNUM PMT_SEQ_NUM/ LIST MISSING;
  FORMAT VR_CLM_SNUM MISSF.;
RUN;

* CHECK IF THE VR_CLNT_ENTRY_DT IS THE SAME ACROSS THE SAME COSSN AND VR_CLM_SNUM;
proc sql;
create table _a1 as select n(distinct VR_CLNT_ENTRY_DT) as n_dt from VRHCLAIM
group by COSSN, VR_CLM_SNUM; quit;
title "N of distinct VR_CLNT_ENTRY_DT by COSSN AND VR_CLM_SNUM";
proc freq data=_a1; tables n_dt / list missing;run;
title;

*STEP 3: MAKE THE FINDER;

PROC SORT DATA=VRPAYMNT OUT=_VRPAYMNT NODUPKEYS; by COSSN VR_CLM_SNUM; RUN;
PROC SORT DATA=VRHCLAIM OUT=_VRHCLAIM NODUPKEYS; by COSSN VR_CLM_SNUM; RUN;
/*
proc freq data=_vrpaymnt; table cossn /list missing nocum; run;
proc freq data=_vrhclaim; table cossn /list missing nocum; run;
*/
DATA _FINDER (KEEP=COSSN VR_CLM_SNUM) not_finder (KEEP=COSSN VR_CLM_SNUM);
MERGE _VRPAYMNT (IN=A) _VRHCLAIM (IN=B);
BY COSSN VR_CLM_SNUM;
IF A AND B then output _FINDER;
else if a and not b then output not_finder;
RUN;

PROC SORT DATA=VRPAYMNT; by COSSN VR_CLM_SNUM; RUN;
PROC SORT DATA=VRHCLAIM; by COSSN VR_CLM_SNUM; RUN;

data _test; set VRPAYMNT; run;
```

## Appendix A.173

JCL/SAS Code: 002\_PROCESS\_VRRMS

```
DATA VRPAYMNT; MERGE VRPAYMNT_FINDER (IN=A); BY COSSN VR_CLM_SNUM; IF A; RUN;
DATA VRHCLAIM; MERGE VRHCLAIM_FINDER (IN=A); BY COSSN VR_CLM_SNUM; IF A; RUN;
/*now both files have only combinations we need*/
```

```
title "VRPAYMENT after finder is done";
PROC FREQ DATA=VRPAYMNT;
  TABLES In_COSSN VR_PSTUS_CD PSTUS_TS VOIDD_PMT_CD VR_CLM_SNUM PMT_SEQ_NUM/ LIST MISSING;
  FORMAT PSTUS_TS YMMN6. VR_CLM_SNUM MISSF.;
RUN;
title;
```

```
/*
```

```
PROC SORT DATA=_FINDER OUT=_FINDER1 NODUPKEYS; by COSSN; RUN;
DATA VRPAYMNT_1; MERGE _test_FINDER1 (IN=A); BY COSSN; IF A; RUN;
PROC FREQ DATA=VRPAYMNT_1;
  TABLES In_COSSN VR_PSTUS_CD PSTUS_TS VOIDD_PMT_CD VR_CLM_SNUM PMT_SEQ_NUM/ LIST MISSING;
  FORMAT PSTUS_TS YMMN6. VR_CLM_SNUM MISSF.;
RUN;
PROC SORT DATA=VRPAYMNT_1; by COSSN VR_CLM_SNUM; RUN;
```

```
data _test (keep=COSSN VR_CLM_SNUM);
merge VRPAYMNT_1 (in=a) VRPAYMNT (in=b);
by COSSN VR_CLM_SNUM;
if a and not b;
run;
```

```
*/
```

```
/******
*PROCESS PAYMENT FILE
******/
```

```
PROC SQL VERBOSE;
  CREATE TABLE VRRMS_TEMP1_PART1 AS
  SELECT COSSN
         ,VR_CLM_SNUM
         ,SUM(TOT_DUE_AMT) AS TOT_DUE_AMT
         ,SUM(TOT_PAID_AMT) AS TOT_PAID_AMT
         ,MAX(PSTUS_TS) AS PSTUS_TS
         ,COUNT(COSSN) AS NPAYSEQ
  FROM VRPAYMNT
```

Appendix A.173

JCL/SAS Code: 002\_PROCESS\_VRRMS

```
GROUP BY COSSN,VR_CLM_SNUM
ORDER BY COSSN,VR_CLM_SNUM;
QUIT;
```

```
/******
*PROCESS CLAIM FILE
*****/
```

```
* FOR THE VARIABLES LISTED BELOW, GET VALUES FROM THE LAST RECORDS BY *;
* COSSN, VR_CLM_SNUM AND PMT_SEQ_NUM. *;
PROC SORT DATA=VRHCLAIM;
  BY COSSN VR_CLM_SNUM PMT_SEQ_NUM;
RUN;
```

```
DATA VRRMS_TEMP1_PART2(KEEP=COSSN
  VR_CLM_SNUM
  VR_CLMBSS_CD
  VR_CLNT_ENTRY_DT
  SSA_RCPDT
  TOT_PAYBL_AMT
  CLNT_EMPT_STDT
  IPE_SGN_DT
  ST_VR_ABBR_CD
  VR_FNL_CLOSR_DT
  VR_PMT_ACTN_CD
  VR_PRD_ENDT
  VR_PRD_STDT);
```

```
SET VRHCLAIM;
BY COSSN VR_CLM_SNUM PMT_SEQ_NUM;
*IF LAST.PMT_SEQ_NUM THEN OUTPUT;
IF LAST.VR_CLM_SNUM THEN OUTPUT;
RUN;
```

```
/******
*MERGE CLAIM TO PAYMENT
*****/
```

```
DATA VRRMS_TEMP1;
MERGE VRRMS_TEMP1_PART1 (IN=A) VRRMS_TEMP1_PART2 (IN=B);
BY COSSN VR_CLM_SNUM;
IF A AND B;
RUN;
PROC FREQ DATA=VRRMS_TEMP1; TABLES VR_CLNT_ENTRY_DT / LIST MISSING;
  FORMAT VR_CLNT_ENTRY_DT MISSF.;
RUN;
```



## Appendix A.173

JCL/SAS Code: 002\_PROCESS\_VRRMS

```
PROC SORT DATA=VRRMS_TEMP1; BY COSSN VR_CLNT_ENTRY_DT VR_CLM_SNUM;RUN;
```

```
* ASSIGN SEQUENCE NUMBER WITHIN EACH GROUP OF COSSN AND VR_CLM_SNUM. *;
```

```
DATA VRRMS_TEMP1;  
  SET VRRMS_TEMP1;  
  BY COSSN VR_CLNT_ENTRY_DT VR_CLM_SNUM;  
  COUNT+1;  
  IF FIRST.COSSN THEN COUNT=1;
```

```
* REMOVE ALL LABELS *;
```

```
ATTRIB _ALL_ LABEL="";
```

```
RUN;
```

```
TITLE "CHECK NPAYSEQ AND COUNT VARIABLES IN TEMP1 DATASET";
```

```
PROC FREQ DATA=VRRMS_TEMP1;  
  TABLES NPAYSEQ COUNT / LIST MISSING;  
RUN;
```

```
TITLE "PRINT 20 OBS FOR TEMP1";
```

```
PROC PRINT DATA=VRRMS_TEMP1(OBS=20);  
RUN;  
TITLE;
```

```
***** PART2 *****;
```

```
* CREATE NCLAIM: NUMBER OF COMBINATION OF ENTRY_DT AND VR_CLM_SNUM. *;
```

```
* CREATE NSPELL: NUMBER OF UNIQUE ENTRY_DT WITHIN A SINGLR SSN. *;
```

```
PROC SQL VERBOSE NOPRINT;  
  CREATE TABLE VRRMS_TEMP2_PART1 AS  
  SELECT COSSN  
         ,COUNT(DISTINCT VR_CLM_SNUM) AS NCLAIM  
         ,COUNT(DISTINCT VR_CLNT_ENTRY_DT) AS NSPELL  
  FROM VRRMS_TEMP1  
  GROUP BY COSSN  
  ORDER BY COSSN;  
QUIT;
```

```
* RESHAPE THE DATASET FROM LONG TO WIDE. *;
```

```
%MACRO TRANS(VAR=);  
PROC TRANSPOSE DATA=VRRMS_TEMP1  
  OUT=&VAR. (DROP=_NAME_) PREFIX=&VAR.;  
  BY COSSN;  
  ID COUNT;  
  VAR &VAR.;  
RUN;
```

## Appendix A.173

JCL/SAS Code: 002\_PROCESS\_VRRMS

```
%MEND TRANS;

%TRANS(VAR=VR_CLM_SNUM);
%TRANS(VAR=NPAYSEQ);
%TRANS(VAR=VR_CLMBSS_CD);
%TRANS(VAR=VR_CLNT_ENTRY_DT);
%TRANS(VAR=SSA_RCPDT);
%TRANS(VAR=PSTUS_TS);
%TRANS(VAR=TOT_PAYBL_AMT);
%TRANS(VAR=TOT_DUE_AMT);
%TRANS(VAR=TOT_PAID_AMT);
%TRANS(VAR=CLNT_EMPT_STDT);
%TRANS(VAR=IPE_SGN_DT);
%TRANS(VAR=ST_VR_ABBR_CD);
%TRANS(VAR=VR_FNL_CLOSR_DT);
%TRANS(VAR=VR_PMT_ACTN_CD);
%TRANS(VAR=VR_PRD_ENDT);
%TRANS(VAR=VR_PRD_STDT);

* MERGE ALL TRANS DATASET TOGETHER.          *;
DATA VRRMS_TEMP2;
  MERGE VR_CLM_SNUM
        NPAYSEQ
        VR_CLMBSS_CD
        VR_CLNT_ENTRY_DT
        SSA_RCPDT
        PSTUS_TS
        TOT_PAYBL_AMT
        TOT_DUE_AMT
        TOT_PAID_AMT
        CLNT_EMPT_STDT
        IPE_SGN_DT
        ST_VR_ABBR_CD
        VR_FNL_CLOSR_DT
        VR_PMT_ACTN_CD
        VR_PRD_ENDT
        VR_PRD_STDT
        VRRMS_TEMP2_PART1;
  BY COSSN;
RUN;

TITLE "CHECK NCLAIM AND NSPELL IN TEMP2";
PROC FREQ DATA=VRRMS_TEMP2;
  TABLES NCLAIM NSPELL / LIST MISSING;
RUN;
```

## Appendix A.173

JCL/SAS Code: 002\_PROCESS\_VRRMS

```
TITLE;

***** PART3 *****;
* GET VALUE OF THE FOLLOWING VARAIBLES FROM THE LATEST ENTRY_DT WITHIN *;
* EACH COSSN. *;
proc sort data=VRHCLAIM;by COSSN VR_CLNT_ENTRY_DT;run;

data VRRMS_TEMP3;
set VRHCLAIM;
by COSSN;
if last.COSSN;
keep ACPL_CAAMT ACPL_CCAMT ADDL_ACPL_CSTS_AMT AVRCLSR_TRKG_CCAMT CLNT_FNM CLNT_LNM
CLNT_MNM CONTS_SGA_LAST_MDT
CONTS_SGA_MNTH1_DT CRNT_ENTLT_DT DIRCT_CAAMT DIRCT_CCAMT DOB ENTLT_CD
MED_SRVS_PRVDD_IND OTH_CAAMT
OTH_CCAMT PMT_PRD_CLM_BIC PMT_PRD_ENDT PMT_PRD_STDT PMT_SEQ_NUM PMT_TS
POST_EMPT_PRD_ENDT POST_EMPT_PRD_STDT
PRE_AWRD_AMT PRIOR_VRCLMPO_AMT SEX_CD SSA_TOT_SAVGS_AMT SVCALC_PIA SVCALC_T16_FEDBAMT
SVCALCWCMP_OFS_AMT
TFSRDUC_AMT TOT_ACPL_CSTS_AMT TOT_CCAMT TRKG_CAAMT VR_SGA_CPRD_TYP_CD VR_STYBLNDS_SW
VR_T16_SW VR_T2_BIC COSSN;
run;

***** PART4: FINAL STEP *****;
* COMBINE TEMP2 AND TEMP3 DATASETS TOGETHER. *;
DATA VRRMS(RENAME=(COSSN=SSN));
MERGE VRRMS_TEMP2 VRRMS_TEMP3;
BY COSSN;
RUN;

TITLE "CHECK CONTENTS OF FINAL VRRMS DATASET";
PROC CONTENTS DATA=VRRMS;
RUN;

TITLE "CHECK NCLAIM AND NSPELL IN VRRMS";
PROC FREQ DATA=VRRMS;
TABLES NCLAIM NSPELL / LIST MISSING;
RUN;

TITLE "PRINT 20 OBS OF FINAL DATASET FOR CHECKING";
PROC PRINT DATA=VRRMS(OBS=20);
RUN;
TITLE;

* OUTPUT PERMANENT DATASET. *;
```

```
DATA OUTLIB.VRRMS;  
  SET VRRMS;  
RUN;  
/*
```

Appendix A.173

JCL/SAS Code: 002\_PROCESS\_VRRMS

```
proc printto;  
run;  
*/
```

Appendix A.174

JCL/SAS Code: 003\_UPLOAD\_VRRMS

```
/******  
** Purpose: Upload the VRRMS file to the mainframe  
*****/  
options comamid=tcp; %let SY56=SY56.sspf.ssa.gov; %let tcpsec=_prompt_; options remote=sy56.spawner; run;  
  
SIGNON ;  
libname datain "M:\DAF21\Task 16 Create VRRMS Component\Version2\Data\Final";  
RSUBMIT;  
  
LIBNAME jcllib 'OPDR.TG.PRD.ETTW.$4671.DAF21.VRRMS.SA.V2';  
OPTION COMPRESS=BINARY REMOTE=session_id;  
  
PROC UPLOAD DATA=datain.vrrms OUT=jcllib.VRRMS (COMPRESS=BINARY);RUN;  
  
ENDRSUBMIT;  
SIGNOFF;
```

## Appendix A.175

JCL/SAS Code: FINVRRMS

```

//$4671FVR JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT: BUILDING THE TICKET RESEARCH FILE (50214)
//* DIRECTOR: MATT URATO
//* PROGRAM: OPDR.TG.PRD.ETTW.ID.DAF--.VRRMS.PRDLIB(VRRMSFIN)
//* DATE: 02/28/23 SERGE LUKASHANETS
//* UPDATED: DPHELPS@MATHEMATICA-MPR.COM
//*  DAWN.PHELPS@SSA.GOV
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//COM  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//VRM  DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21.VRRMS.SA.V2,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.VRRMS,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTCN DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.VRRMSCN.SA.V2,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10VRRMS,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(1000,100),RLSE)
//TEMP2 DD DSN=&&TEMPX,UNIT=SYSDA,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

%MACRO DOIT;

PROC SORT DATA=VRM.VRRMS OUT=TEMP1.VRRMS;
  BY SSN;
RUN;

DATA OUT.VRRMS;
  MERGE TEMP1.VRRMS(IN=VRRMS RENAME=(DOB=DOBVRRMS
    SEX_CD=SEXVRRMS))

```

Appendix A.175

JCL/SAS Code: FINVRRMS

```
        COM.COMVARS;
        BY SSN;
        IF VRRMS;

        %INCLUDE "OPDR.TG.PRD.ETTW.$4671.DAF21.VRRMS.PRDLIB(VRMLBLS)";
RUN;
%MEND DOIT;
%DOIT;

PROC CONTENTS DATA=OUT.VRRMS
        OUT=OUTCN.VRRMSCN;
RUN;

/* CREATE 10% DM FOR VRRMS */
DATA OUT10.P10VRRMS;
        SET OUT.VRRMS;
        IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9')) AND
            (SUBSTR(SSN,9,1) IN ('0', '5'));
RUN;

PROC CONTENTS DATA=OUT10.P10VRRMS
        OUT= OUTCN.P10VRRMS NOPRINT;
RUN;
```



Appendix A.176

JCL/SAS Code: VRMLBLS

```
%MACRO GETLB;
LABEL
ACPL_CAAMT="VR ACP COSTS ALLOWED"
ACPL_CCAMT="ACP COSTS CLAIMED DURING VR"
ADDL_ACPL_CSTS_AMT="ADDITIONAL ADMINISTRATIVE, COUNSELING,
AND PLACEMENT (ACP) COSTS AMOUNT"
AVRCLSR_TRKG_CCAMT="TRACKING COSTS CLAIMED AFTER VR"
SSN="SOCIAL SECURITY NUMBER"
CLNT_FNM="VR CLIENT FIRST NAME"
CLNT_LNM="VR CLIENT LAST NAME"
CLNT_MNM="VR CLIENT MIDDLE NAME"
CONTS_SGA_LAST_MDT="VR LAST MONTH OF CONTINUOUS SGA"
CONTS_SGA_MNTH1_DT="VR FIRST MONTH OF CONTINUOUS SGA"
CRNT_ENTLT_DT="VR CURRENT DATE OF ENTITLEMENT FOR THE ACCOUNT"
DIRCT_CAAMT="VR DIRECT COSTS ALLOWED"
DIRCT_CCAMT="VR DIRECT COSTS CLAIMED"
DOBVRRMS="DATE OF BIRTH-VRRMS"
ENTLT_CD="VR ENTITLEMENT CODE"
MED_SRVS_PRVDD_IND="VR MEDICAL SERVICES PROVIDED"
NCLAIM="VR NUMBER OF UNIQUE VR_CLM_SNUM WITHIN EACH SSN"
NSPELL="NUMBER OF UNIQUE VR_CLNT_ENTRY_DT WITHIN EACH SSN"
OTH_CAAMT="VR OTHER COSTS ALLOWED"
OTH_CCAMT="VR OTHER COSTS CLAIMED"
PMT_PRD_CLM_BIC="VR PAYMENT PERIOD CLAIM ACCOUNT BIC"
PMT_PRD_ENDT="VR PAYMENT PERIOD END DATE"
PMT_PRD_STDT="VR PAYMENT PERIOD BEGIN DATE"
PMT_SEQ_NUM="VR PAYMENT SEQUENCE NUMBER"
PMT_TS="VR PAYMENT TIMESTAMP"
POST_EMPT_PRD_ENDT="VR POST-EMPLOYMENT PERIOD END DATE"
POST_EMPT_PRD_STDT="VR POST-EMPLOYMENT PERIOD BEGIN DATE"
PRE_AWRD_AMT="PRE AWARD AMOUNT"
PRIOR_VRCLMPO_AMT="PRIOR PAYOUT"
SEXVRRMS="GENDER-VRRMS"
SSA_TOT_SAVGS_AMT="VR TOTAL SSA SAVINGS AMOUNT"
SVCALC_PIA="VR SAVINGS PIA USED AMOUNT"
SVCALC_T16_FEDBAMT="VR SSI FBA USED AMOUNT"
SVCALCWCMP_OFS_AMT="VR SAVINGS WORKERS COMP USED AMOUNT"
TFSRDUC_AMT="VR TRUST FUND SAVINGS REDUCTION"
TOT_ACPL_CSTS_AMT="TOTAL ADMINISTRATIVE, COUNSELING, AND PLACEMENT
(ACP) COSTS AMOUNT"
TOT_CCAMT="VR TOTAL COSTS CLAIMED"
TRKG_CAAMT="VR TRACKING COSTS ALLOWED"
VR_SGA_CPRD_TYP_CD="VR SGA PERIOD TYPE"
VR_STYBLNDS_SW="SVR STATUTORILY BLIND"
```

Appendix A.176

JCL/SAS Code: VRMLBLS

```
VR_T16_SW="SVR SSO INVOLVED"
VR_T2_BIC="SVR CLMBIC"
%DO I=1 %TO 7 %BY 1;
CLNT_EMPT_STDT&I.="VR DATE EMPLOYMENT BEGAN &I."
IPE_SGN_DT&I.="VR DATE SIGNED IPE &I."
NPAYSEQ&I.="VR NUMBER OF RECORDS FOR THE SAME VR_CLM_SNUM
WITHIN EACH SSN &I."
PSTUS_TS&I.="VR STATUS TIMESTAMP &I."
SSA_RCPDT&I.="VR SSA RECEIPT DATE &I."
ST_VR_ABBR_CD&I.="THE STATE VR AGENCY ABBRIVATION CODE &I."
TOT_DUE_AMT&I.="VR TOTAL AMOUNT DUE &I."
TOT_PAID_AMT&I.="VR TOTAL AMOUNT PAID &I."
TOT_PAYBL_AMT&I.="VR TOTAL PAYABLE &I."
VR_CLMBSS_CD&I.="VR CLAIM BASIS &I."
VR_CLM_SNUM&I.="VR CLAIM UID &I."
VR_CLNT_ENTRY_DT&I.="DATE CLIENT ENTERED VR &I."
VR_FNL_CLOSR_DT&I.="DATE OF FINAL VR CLOSURE &I."
VR_PMT_ACTN_CD&I.="VR PAYMENT ACTION CODE &I."
VR_PRD_ENDT&I.="VR PERIOD END DATE &I."
VR_PRD_STDT&I.="VR PERIOD BEGIN DATE &I."
%END;
;
%MEND GETLB;
%GETLB;
```

## Appendix A.177

JCL/SAS Code: FINVRRMO

```

//#6266VRM JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT: BUILDING THE DISABILITY ANALYSIS FILE (50214)
//* DIRECTOR: JODY SCHIMMEL-HYDE
//* PROGRAM: OPDR.TG.PRD.ETTW.#6266.DAF21.VRRMS.PRDLIB(FINVRRMO)
//* DATE: 10/18/22 SERGE LUKASHANETS
//*  SLUKASHANETS@MATHEMATICA-MPR.COM
//*  SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//COM  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.COMVARS.SA.V1,DISP=SHR
//VRMO DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF20P.VRRMS.HIST,DISP=SHR
//OUTO DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.VRRMS.HIST,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTCO DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.VRRMSCO.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTO10 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10VRRMS.HIST,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *

```

```

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

```

```

%MACRO DOIT;

```

```

DATA OUTO.VRRMS;
MERGE VRMO.VRRMS(IN=VRRMS)
      COM.COMVARS;
BY SSN;
IF VRRMS;

```

```

LABEL

```

```

C_PRIOR_PMT_THIS_PD="PRIOR PAYMENT FOR SAME PERIOD OF SVRA"

```

```

%MACRO LAB (N);

```

```

%DO I = 1 %TO &N.;

```

```

C_SVRA_CLOSED&I.="SVRA CLOSE DATE ON THE SPELL &I."

```

```

C_SVRA_ENTER&I. ="SVRA ENTER DATE ON THE SPELL &I."

```

```

%END;

```

Appendix A.177

JCL/SAS Code: FINVRRMO

```
%MEND;  
%LAB(9)  
;  
  
RUN;  
  
PROC CONTENTS DATA=OUTO.VRRMS OUT=OUTCO.VRRMSCN;  
RUN;  
  
* CREATE 10% DM FOR VRRMS;  
DATA OUTO10.P10VRRMS;  
  SET OUTO.VRRMS;  
  IF (SUBSTR(SSN,8,1) IN ('0', '2', '4', '7', '9')) AND  
     (SUBSTR(SSN,9,1) IN ('0', '5'));  
RUN;  
  
%MEND DOIT;  
%DOIT;
```

## Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

/\* -----

Code generated by a SAS task

Generated on Tuesday, July 13, 2021 at 10:12:28 AM

By task: Import Data Wizard

Source file: M:\2021 RSA 911 raw file from Dept of Education\SSACY2021.csv

Server: Local File System

Output data: WORK.SSACY2021

Server: Local

07/13/2021: Automated code developed by SAS Enterprise Guide

In order to read in the RSA data in the new format in a CSV file.

----- \*/

%let rootloc = \\ba.ad.ssa.gov\dcrdp\RemoteShares\MPR\DAF21\TASK 17 Create DAF-RSA Files\Programs;

proc printto

log = "&amp;rootloc.\Read\_RSA\_CSV.log"

new

print = "&amp;rootloc.\Read\_RSA\_CSV.lst" new;

run;

options obs=max mprint;

LIBNAME OUT '\\ba.ad.ssa.gov\dcrdp\RemoteShares\MPR\DAF21\RSA\upd';

%macro read;

DATA OUT.RSA21;

LENGTH

Year \$ 6

AgencyCode 8

IndividualID \$ 12

ApplicationDate 8

BirthDate 8

Sex \$ 1 /\*Formerly 8\*/

AmerIndian 8

Asian 8

Black 8

Hawaiian 8

White 8

Hispanic 8

Veteran 8

State \$ 2

CountyFIPSCode 8

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

ZIPCode 8  
 Referral 8  
 Student 8  
 EligibilityDate 8  
 EligibilityExt 8  
 OOSPlacementDate 8  
 OOSExitDate 8  
 Disability 8  
 PrimDisabilityCodeType 8  
 PrimDisabilityCodeSource 8  
 SecondaryDisabilityType 8  
 SecondaryDisabilitySource 8  
 DisabilitySigCode 8  
 TWESStartDate 8  
 TWEEndDate 8  
 IPESupportedEmpGoal 8  
 ipeEmpStatus 8  
 ipePrimaryOcc 8  
 ipeHourlyWage 8  
 ipeWeeklyHoursWorked 8  
 Adult 8  
 AdultEd 8  
 DislocatedWorker 8  
 JobCorps 8  
 VocRehab 8  
 WPEmpService 8  
 Youth 8  
 YouthBuild \$ 14  
 LongTermUnemp 8  
 ExhaustTANF 8  
 FosterCareYouth 8  
 HomelessOrRunaway 8  
 ExOffenderStatus 8  
 LowIncomeStatus 8  
 EnglishLearner 8  
 BasicSkillsDeficient 8  
 CulturalBarriers 8  
 SingleParent 8  
 DisHomemaker 8  
 MigrantFarmworker 8  
 StateDisStudentAgeRange 8  
 SchoolGradeCompleted 8  
 InSecondaryEd 8  
 SpecialEdCertCompDate 8  
 SecSchoolDiplomaDate 8

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```

GEDDate      8
EnrolledInPostsecEd  8
CredProgramEnrollDate  8
CompletedSomePostsecEd  8
AssociateDegreeDate  8
BachelorDegreeDate  8
MastersDegreeDate  8
DegreeAboveMastersDate  8
VTLicenseDate  8
VTCertificateDate  8
OtherLicOrCertDate  8
PETSSStartDate  8
jecVRAgencyStaff  8
jecVRAgencyPurchase  8
jecPurchaseProviderType  8
jecVRServicePurchaseExpenditure  8
wbleVRAgencyStaff  8
wbleVRAgencyPurchase  8
wblePurchaseProviderType  8
wbleVRServicePurchaseExpenditure  8
ceoVRAgencyStaff  8
ceoVRAgencyPurchase  8
ceoPurchaseProviderType  8
ceoVRServicePurchaseExpenditure  8
wrtVRAgencyStaff  8
wrtVRAgencyPurchase  8
wrtPurchaseProviderType  8
wrtVRServicePurchaseExpenditure  8
isaVRAgencyStaff  8
isaVRAgencyPurchase  8
isaPurchaseProviderType  8
isaVRServicePurchaseExpenditure  8
VRServiceStartDate  8
CareerServiceDate  8
gcutVRAgencyPurchase  8
gcutPurchaseProviderType  8
gcutVRTitleIFundsExpended  8
gcutCompServiceProvider  8
gcutCompServiceProviderType $ 8
fycutVRAgencyPurchase  8
fycutPurchaseProviderType  8
fycutVRTitleIFundsExpended  8
fycutCompServiceProvider  8
fycutCompServiceProviderType $ 8
jcctVRAgencyPurchase  8
    
```

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```
jcctPurchaseProviderType 8
jcctVRTitleIFundsExpended 8
jcctCompServiceProvider 8
jcctCompServiceProviderType $ 8
ovtVRAgencyStaff 8
ovtVRAgencyPurchase 8
ovtPurchaseProviderType 8
ovtVRTitleIFundsExpended 8
ovtCompServiceProvider 8
ovtCompServiceProviderType $ 8
ojtVRAgencyStaff 8
ojtVRAgencyPurchase 8
ojtPurchaseProviderType 8
ojtVRTitleIFundsExpended 8
ojtCompServiceProvider 8
ojtCompServiceProviderType $ 8
ratVRAgencyPurchase 8
ratPurchaseProviderType 8
ratVRTitleIFundsExpended 8
ratCompServiceProvider 8
ratCompServiceProviderType $ 8
barltVRAgencyStaff 8
barltVRAgencyPurchase 8
barltPurchaseProviderType 8
barltVRTitleIFundsExpended 8
barltCompServiceProvider 8
barltCompServiceProviderType $ 8
jrtVRAgencyStaff 8
jrtVRAgencyPurchase 8
jrtPurchaseProviderType 8
jrtVRTitleIFundsExpended 8
jrtCompServiceProvider 8
jrtCompServiceProviderType $ 8
drstVRAgencyStaff 8
drstVRAgencyPurchase 8
drstPurchaseProviderType 8
drstVRTitleIFundsExpended 8
drstCompServiceProvider 8
drstCompServiceProviderType $ 8
mtVRAgencyStaff 8
mtVRAgencyPurchase 8
mtPurchaseProviderType 8
mtVRTitleIFundsExpended 8
mtCompServiceProvider 8
mtCompServiceProviderType $ 8
```



Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```
rsetVRAgencyStaff 8
rsetVRAgencyPurchase 8
rsetPurchaseProviderType 8
rsetVRTitleIFundsExpended 8
rsetCompServiceProvider 8
rsetCompServiceProviderType $ 1
ctVRAgencyStaff 8
ctVRAgencyPurchase 8
ctPurchaseProviderType 8
ctVRTitleIFundsExpended 8
ctCompServiceProvider 8
ctCompServiceProviderType $ 1
assVRAgencyStaff 8
assVRAgencyPurchase 8
assPurchaseProviderType 8
assVRTitleIFundsExpended 8
assCompServiceProvider 8
assCompServiceProviderType $ 8
dtiVRAgencyStaff 8
dtiVRAgencyPurchase 8
dtiPurchaseProviderType 8
dtiVRTitleIFundsExpended 8
dtiCompServiceProvider 8
dtiCompServiceProviderType $ 8
vrcgVRAgencyStaff 8
vrcgVRAgencyPurchase 8
vrcgPurchaseProviderType 8
vrcgVRTitleIFundsExpended 8
vrcgCompServiceProvider 8
vrcgCompServiceProviderType $ 8
jsaVRAgencyStaff 8
jsaVRAgencyPurchase 8
jsaPurchaseProviderType 8
jsaVRTitleIFundsExpended 8
jsaCompServiceProvider 8
jsaCompServiceProviderType $ 8
jpaVRAgencyStaff 8
jpaVRAgencyPurchase 8
jpaPurchaseProviderType 8
jpaVRTitleIFundsExpended 8
jpaCompServiceProvider 8
jpaCompServiceProviderType $ 8
stjsVRAgencyStaff $ 1 /*Formerly 8*/
stjsVRAgencyPurchase 8
stjsPurchaseProviderType 8
```

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```
stjsVRTitleIFundsExpended 8
stjsCompServiceProvider 8
stjsCompServiceProviderType $ 8
sesVRAgencyStaff 8
sesVRAgencyPurchase 8
sesPurchaseProviderType 8
sesVRTitleIFundsExpended 8
sesSETitleVIFundsExpended 8
sesCompServiceProvider 8
sesCompServiceProviderType $ 8
irsVRAgencyStaff 8
irsVRAgencyPurchase 8
irsPurchaseProviderType 8
irsVRTitleIFundsExpended 8
irsCompServiceProvider 8
irsCompServiceProviderType $ 8
bcVRAgencyStaff 8
bcVRAgencyPurchase 8
bcPurchaseProviderType 8
bcVRTitleIFundsExpended 8
bcCompServiceProvider 8
bcCompServiceProviderType $ 8
cesVRAgencyStaff 8
cesVRAgencyPurchase 8
cesPurchaseProviderType 8
cesVRTitleIFundsExpended 8
cesSETitleVIFundsExpended 8
cesCompServiceProvider 8
cesCompServiceProviderType $ 8
esVRAgencyStaff 8
esVRAgencyPurchase 8
esPurchaseProviderType 8
esVRTitleIFundsExpended 8
esSETitleVIFundsExpended 8
tranVRAgencyStaff 8
tranVRAgencyPurchase 8
tranPurchaseProviderType 8
tranVRTitleIFundsExpended 8
tranCompServiceProvider 8
tranCompServiceProviderType $ 8
mntVRAgencyStaff 8
mntVRAgencyPurchase 8
mntPurchaseProviderType 8
mntVRTitleIFundsExpended 8
mntCompServiceProvider 8
```

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```

mntCompServiceProviderType $ 8
rtVRAgencyStaff 8
rtVRAgencyPurchase 8
rtPurchaseProviderType 8
rtVRTitleIFundsExpended 8
rtCompServiceProvider 8
rtCompServiceProviderType $ 8
pasVRAgencyStaff 8
pasVRAgencyPurchase 8
pasPurchaseProviderType 8
pasVRTitleIFundsExpended 8
pasCompServiceProvider 8
pasCompServiceProviderType $ 8 /*Formerly $ 7 */
tasVRAgencyStaff 8
tasVRAgencyPurchase 8
tasPurchaseProviderType 8
tasVRTitleIFundsExpended 8
tasCompServiceProvider 8
tasCompServiceProviderType $ 8 /*Formerly $ 5*/
rsVRAgencyStaff 8
rsVRAgencyPurchase 8
rsPurchaseProviderType 8
rsVRTitleIFundsExpended 8
rsCompServiceProvider 8
rsCompServiceProviderType $ 8 /*Formerly $ 5*/
isVRAgencyStaff 8
isVRAgencyPurchase 8
isPurchaseProviderType 8
isVRTitleIFundsExpended 8
isCompServiceProvider 8
isCompServiceProviderType $ 8
osVRAgencyStaff 8
osVRAgencyPurchase 8
osPurchaseProviderType 8
osVRTitleIFundsExpended 8
osCompServiceProvider 8
osCompServiceProviderType $ 8
EdFuncLevelDate 8
SecondaryDate 8
PostsecTransReportCardDate 8
TrainingMilestoneDate 8
SkillGainSkillsProgDate 8
eoPrimOccupation 8
exitDate 8
exitType 8

```

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```

exitReason      8
exitEmpOutcome  8
exitPrimOccupation 8
exitHourlyWage  8
exitWeeklyHoursWorked 8
peCredProgramEnrollDate 8
peCredAttainmentDate 8
peCredentialType 8
q1aeEmpStatus  8
q2aeEmpStatus  8
q2aeWages      8
q3aeEmpStatus  8
q4aeEmpStatus  8
q4aeRetentionQ2AndQ4 8
appMonthlyPubSup $ 7
appMedInsCov    $ 5
exitMonthlyPubSup $ 7
exitMedInsCov  $ 5
IPEInitialDate  8
IPEExtensionDate 8
EnrolledInSecEquiv 8
CompDisenrollMSG 8
wblevVRAgencyStaff 8
wblevVRAgencyPurchase 8
wblevPurchaseProviderType 8
wblevVRTitleIFundsExpended 8
wblevCompServiceProvider 8
wblevCompServiceProviderType $ 8
ImportAgency  $ 4
Report_Through  8
SSN            $ 9 ;

```

FORMAT

```

Year          $CHAR6.
AgencyCode   BEST3.
IndividualID  $CHAR12.
ApplicationDate YYMMDD10.
BirthDate     YYMMDD10.
Sex          $CHAR1. /*Formerly BEST1. */
AmerIndian   BEST1.
Asian        BEST1.
Black        BEST1.
Hawaiian     BEST1.
White        BEST1.
Hispanic     BEST1.
Veteran      BEST1.

```

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

State \$CHAR2.  
CountyFIPSCode BEST5.  
ZIPCode BEST5.  
Referral BEST2.  
Student BEST1.  
EligibilityDate YYYYMMDD10.  
EligibilityExt YYYYMMDD10.  
OOSPlacementDate YYYYMMDD10.  
OOSExitDate YYYYMMDD10.  
Disability BEST1.  
PrimDisabilityCodeType BEST2.  
PrimDisabilityCodeSource BEST2.  
SecondaryDisabilityType BEST2.  
SecondaryDisabilitySource BEST2.  
DisabilitySigCode BEST1.  
TWStartDate YYYYMMDD10.  
TWEendDate YYYYMMDD10.  
IPESupportedEmpGoal BEST1.  
ipeEmpStatus BEST2.  
ipePrimaryOcc BEST6.  
ipeHourlyWage BEST7.  
ipeWeeklyHoursWorked BEST5.  
Adult BEST1.  
AdultEd BEST1.  
DislocatedWorker BEST1.  
JobCorps BEST1.  
VocRehab BEST1.  
WPEmpService BEST1.  
Youth BEST1.  
YouthBuild \$CHAR14.  
LongTermUnemp BEST1.  
ExhaustTANF BEST1.  
FosterCareYouth BEST1.  
HomelessOrRunaway BEST1.  
ExOffenderStatus BEST1.  
LowIncomeStatus BEST1.  
EnglishLearner BEST1.  
BasicSkillsDeficient BEST1.  
CulturalBarriers BEST1.  
SingleParent BEST1.  
DisHomemaker BEST1.  
MigrantFarmworker BEST1.  
StateDisStudentAgeRange TIME8.  
SchoolGradeCompleted BEST2.  
InSecondaryEd BEST1.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

SpecialEdCertCompDate YYMMDD10.  
SecSchoolDiplomaDate YYMMDD10.  
GEDDate YYMMDD10.  
EnrolledInPostsecEd BEST1.  
CredProgramEnrollDate YYMMDD10.  
CompletedSomePostsecEd BEST1.  
AssociateDegreeDate YYMMDD10.  
BachelorDegreeDate YYMMDD10.  
MastersDegreeDate YYMMDD10.  
DegreeAboveMastersDate YYMMDD10.  
VTLicenseDate YYMMDD10.  
VTCertificateDate YYMMDD10.  
OtherLicOrCertDate YYMMDD10.  
PETSStartDate YYMMDD10.  
jecVRAgencyStaff BEST1.  
jecVRAgencyPurchase BEST1.  
jecPurchaseProviderType BEST1.  
jecVRServicePurchaseExpenditure BEST5.  
wbleVRAgencyStaff BEST1.  
wbleVRAgencyPurchase BEST1.  
wblePurchaseProviderType BEST1.  
wbleVRServicePurchaseExpenditure BEST5.  
ceoVRAgencyStaff BEST1.  
ceoVRAgencyPurchase BEST1.  
ceoPurchaseProviderType BEST1.  
ceoVRServicePurchaseExpenditure BEST5.  
wrtVRAgencyStaff BEST1.  
wrtVRAgencyPurchase BEST1.  
wrtPurchaseProviderType BEST1.  
wrtVRServicePurchaseExpenditure BEST5.  
isaVRAgencyStaff BEST1.  
isaVRAgencyPurchase BEST1.  
isaPurchaseProviderType BEST1.  
isaVRServicePurchaseExpenditure BEST5.  
VRServiceStartDate YYMMDD10.  
CareerServiceDate YYMMDD10.  
gcutVRAgencyPurchase BEST1.  
gcutPurchaseProviderType BEST1.  
gcutVRTitleIFundsExpended BEST6.  
gcutCompServiceProvider BEST1.  
gcutCompServiceProviderType \$CHAR8.  
fycutVRAgencyPurchase BEST1.  
fycutPurchaseProviderType BEST1.  
fycutVRTitleIFundsExpended BEST6.  
fycutCompServiceProvider BEST1.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

fycutCompServiceProviderType \$CHAR8.  
jcctVRAgencyPurchase BEST1.  
jcctPurchaseProviderType BEST1.  
jcctVRTitleIFundsExpended BEST5.  
jcctCompServiceProvider BEST1.  
jcctCompServiceProviderType \$CHAR8.  
ovtVRAgencyStaff BEST1.  
ovtVRAgencyPurchase BEST1.  
ovtPurchaseProviderType BEST1.  
ovtVRTitleIFundsExpended BEST5.  
ovtCompServiceProvider BEST1.  
ovtCompServiceProviderType \$CHAR8.  
ojtVRAgencyStaff BEST1.  
ojtVRAgencyPurchase BEST1.  
ojtPurchaseProviderType BEST1.  
ojtVRTitleIFundsExpended BEST5.  
ojtCompServiceProvider BEST1.  
ojtCompServiceProviderType \$CHAR8.  
ratVRAgencyPurchase BEST1.  
ratPurchaseProviderType BEST1.  
ratVRTitleIFundsExpended BEST4.  
ratCompServiceProvider BEST1.  
ratCompServiceProviderType \$CHAR8.  
barltVRAgencyStaff BEST1.  
barltVRAgencyPurchase BEST1.  
barltPurchaseProviderType BEST1.  
barltVRTitleIFundsExpended BEST5.  
barltCompServiceProvider BEST1.  
barltCompServiceProviderType \$CHAR8.  
jrtVRAgencyStaff BEST1.  
jrtVRAgencyPurchase BEST1.  
jrtPurchaseProviderType BEST1.  
jrtVRTitleIFundsExpended BEST5.  
jrtCompServiceProvider BEST1.  
jrtCompServiceProviderType \$CHAR8.  
drstVRAgencyStaff BEST1.  
drstVRAgencyPurchase BEST1.  
drstPurchaseProviderType BEST1.  
drstVRTitleIFundsExpended BEST7.  
drstCompServiceProvider BEST1.  
drstCompServiceProviderType \$CHAR8.  
mtVRAgencyStaff BEST1.  
mtVRAgencyPurchase BEST1.  
mtPurchaseProviderType BEST1.  
mtVRTitleIFundsExpended BEST6.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```
mtCompServiceProvider BEST1.  
mtCompServiceProviderType $CHAR8.  
rsetVRAgencyStaff BEST1.  
rsetVRAgencyPurchase BEST1.  
rsetPurchaseProviderType BEST1.  
rsetVRTitleIFundsExpended BEST5.  
rsetCompServiceProvider BEST1.  
rsetCompServiceProviderType $CHAR1.  
ctVRAgencyStaff BEST1.  
ctVRAgencyPurchase BEST1.  
ctPurchaseProviderType BEST1.  
ctVRTitleIFundsExpended BEST5.  
ctCompServiceProvider BEST1.  
ctCompServiceProviderType $CHAR1.  
assVRAgencyStaff BEST1.  
assVRAgencyPurchase BEST1.  
assPurchaseProviderType BEST1.  
assVRTitleIFundsExpended BEST6.  
assCompServiceProvider BEST1.  
assCompServiceProviderType $CHAR8.  
dtiVRAgencyStaff BEST1.  
dtiVRAgencyPurchase BEST1.  
dtiPurchaseProviderType BEST1.  
dtiVRTitleIFundsExpended BEST6.  
dtiCompServiceProvider BEST1.  
dtiCompServiceProviderType $CHAR8.  
vrcgVRAgencyStaff BEST1.  
vrcgVRAgencyPurchase BEST1.  
vrcgPurchaseProviderType BEST1.  
vrcgVRTitleIFundsExpended BEST5.  
vrcgCompServiceProvider BEST1.  
vrcgCompServiceProviderType $CHAR8.  
jsaVRAgencyStaff BEST1.  
jsaVRAgencyPurchase BEST1.  
jsaPurchaseProviderType BEST1.  
jsaVRTitleIFundsExpended BEST5.  
jsaCompServiceProvider BEST1.  
jsaCompServiceProviderType $CHAR8.  
jpaVRAgencyStaff BEST1.  
jpaVRAgencyPurchase BEST1.  
jpaPurchaseProviderType BEST1.  
jpaVRTitleIFundsExpended BEST5.  
jpaCompServiceProvider BEST1.  
jpaCompServiceProviderType $CHAR8.  
stjsVRAgencyStaff $CHAR1. /*Formerly BEST1.*/
```



Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

stjsVRAgencyPurchase BEST1.  
stjsPurchaseProviderType BEST1.  
stjsVRTitleIFundsExpended BEST5.  
stjsCompServiceProvider BEST1.  
stjsCompServiceProviderType \$CHAR8.  
sesVRAgencyStaff BEST1.  
sesVRAgencyPurchase BEST1.  
sesPurchaseProviderType BEST1.  
sesVRTitleIFundsExpended BEST5.  
sesSETitleVIFundsExpended BEST5.  
sesCompServiceProvider BEST1.  
sesCompServiceProviderType \$CHAR8.  
irsVRAgencyStaff BEST1.  
irsVRAgencyPurchase BEST1.  
irsPurchaseProviderType BEST1.  
irsVRTitleIFundsExpended BEST4.  
irsCompServiceProvider BEST1.  
irsCompServiceProviderType \$CHAR8.  
bcVRAgencyStaff BEST1.  
bcVRAgencyPurchase BEST1.  
bcPurchaseProviderType BEST1.  
bcVRTitleIFundsExpended BEST5.  
bcCompServiceProvider BEST1.  
bcCompServiceProviderType \$CHAR8.  
cesVRAgencyStaff BEST1.  
cesVRAgencyPurchase BEST1.  
cesPurchaseProviderType BEST1.  
cesVRTitleIFundsExpended BEST5.  
cesSETitleVIFundsExpended BEST5.  
cesCompServiceProvider BEST1.  
cesCompServiceProviderType \$CHAR8.  
esVRAgencyStaff BEST1.  
esVRAgencyPurchase BEST1.  
esPurchaseProviderType BEST1.  
esVRTitleIFundsExpended BEST5.  
esSETitleVIFundsExpended BEST4.  
tranVRAgencyStaff BEST1.  
tranVRAgencyPurchase BEST1.  
tranPurchaseProviderType BEST1.  
tranVRTitleIFundsExpended BEST6.  
tranCompServiceProvider BEST1.  
tranCompServiceProviderType \$CHAR8.  
mntVRAgencyStaff BEST1.  
mntVRAgencyPurchase BEST1.  
mntPurchaseProviderType BEST1.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

mntVRTitleIFundsExpended BEST5.  
mntCompServiceProvider BEST1.  
mntCompServiceProviderType \$CHAR8.  
rtVRAgencyStaff BEST1.  
rtVRAgencyPurchase BEST1.  
rtPurchaseProviderType BEST1.  
rtVRTitleIFundsExpended BEST6.  
rtCompServiceProvider BEST1.  
rtCompServiceProviderType \$CHAR8.  
pasVRAgencyStaff BEST1.  
pasVRAgencyPurchase BEST1.  
pasPurchaseProviderType BEST1.  
pasVRTitleIFundsExpended BEST6.  
pasCompServiceProvider BEST1.  
pasCompServiceProviderType \$CHAR8.  
tasVRAgencyStaff BEST1.  
tasVRAgencyPurchase BEST1.  
tasPurchaseProviderType BEST1.  
tasVRTitleIFundsExpended BEST6.  
tasCompServiceProvider BEST1.  
tasCompServiceProviderType \$CHAR8.  
rsVRAgencyStaff BEST1.  
rsVRAgencyPurchase BEST1.  
rsPurchaseProviderType BEST1.  
rsVRTitleIFundsExpended BEST5.  
rsCompServiceProvider BEST1.  
rsCompServiceProviderType \$CHAR8.  
isVRAgencyStaff BEST1.  
isVRAgencyPurchase BEST1.  
isPurchaseProviderType BEST1.  
isVRTitleIFundsExpended BEST6.  
isCompServiceProvider BEST1.  
isCompServiceProviderType \$CHAR8.  
osVRAgencyStaff BEST1.  
osVRAgencyPurchase BEST1.  
osPurchaseProviderType BEST1.  
osVRTitleIFundsExpended BEST6.  
osCompServiceProvider BEST1.  
osCompServiceProviderType \$CHAR8.  
EdFuncLevelDate YYMMDD10.  
SecondaryDate YYMMDD10.  
PostsecTransReportCardDate YYMMDD10.  
TrainingMilestoneDate YYMMDD10.  
SkillGainSkillsProgDate YYMMDD10.  
eoPrimOccupation BEST6.

## Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```
exitDate      YMMMDD10.
exitType      BEST1.
exitReason    BEST2.
exitEmpOutcome BEST1.
exitPrimOccupation BEST6.
exitHourlyWage BEST6.
exitWeeklyHoursWorked BEST2.
peCredProgramEnrollDate YMMMDD10.
peCredAttainmentDate YMMMDD10.
peCredentialType BEST1.
q1aeEmpStatus BEST1.
q2aeEmpStatus BEST1.
q2aeWages     BEST9.
q3aeEmpStatus BEST1.
q4aeEmpStatus BEST1.
q4aeRetentionQ2AndQ4 BEST1.
appMonthlyPubSup $CHAR7.
appMedInsCov    $CHAR5.
exitMonthlyPubSup $CHAR7.
exitMedInsCov  $CHAR5.
IPEInitialDate  YMMMDD10.
IPEExtensionDate YMMMDD10.
EnrolledInSecEquiv BEST1.
CompDisenrollMSG YMMMDD10.
wblevVRAgencyStaff BEST1.
wblevVRAgencyPurchase BEST1.
wblevPurchaseProviderType BEST1.
wblevVRTitleIFundsExpended BEST5.
wblevCompServiceProvider BEST1.
wblevCompServiceProviderType $CHAR8.
ImportAgency   $CHAR4.
Report_Through YMMMDD10.
SSN            $CHAR9. ;
INFORMAT
Year          $CHAR6.
AgencyCode   BEST3.
IndividualID  $CHAR12.
ApplicationDate YMMMDD10.
BirthDate    YMMMDD10.
Sex          $CHAR1. /*Formerly BEST1.*/
AmerIndian   BEST1.
Asian        BEST1.
Black        BEST1.
Hawaiian    BEST1.
White        BEST1.
```

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

Hispanic BEST1.  
Veteran BEST1.  
State \$CHAR2.  
CountyFIPSCode BEST5.  
ZIPCode BEST5.  
Referral BEST2.  
Student BEST1.  
EligibilityDate YYYYMMDD10.  
EligibilityExt YYYYMMDD10.  
OOSPlacementDate YYYYMMDD10.  
OOSExitDate YYYYMMDD10.  
Disability BEST1.  
PrimDisabilityCodeType BEST2.  
PrimDisabilityCodeSource BEST2.  
SecondaryDisabilityType BEST2.  
SecondaryDisabilitySource BEST2.  
DisabilitySigCode BEST1.  
TWEStartDate YYYYMMDD10.  
TWEEndDate YYYYMMDD10.  
IPESupportedEmpGoal BEST1.  
ipeEmpStatus BEST2.  
ipePrimaryOcc BEST6.  
ipeHourlyWage BEST7.  
ipeWeeklyHoursWorked BEST5.  
Adult BEST1.  
AdultEd BEST1.  
DislocatedWorker BEST1.  
JobCorps BEST1.  
VocRehab BEST1.  
WPEmpService BEST1.  
Youth BEST1.  
YouthBuild \$CHAR14.  
LongTermUnemp BEST1.  
ExhaustTANF BEST1.  
FosterCareYouth BEST1.  
HomelessOrRunaway BEST1.  
ExOffenderStatus BEST1.  
LowIncomeStatus BEST1.  
EnglishLearner BEST1.  
BasicSkillsDeficient BEST1.  
CulturalBarriers BEST1.  
SingleParent BEST1.  
DisHomemaker BEST1.  
MigrantFarmworker BEST1.  
StateDisStudentAgeRange TIME11.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

SchoolGradeCompleted BEST2.  
InSecondaryEd BEST1.  
SpecialEdCertCompDate YYMMDD10.  
SecSchoolDiplomaDate YYMMDD10.  
GEDDate YYMMDD10.  
EnrolledInPostsecEd BEST1.  
CredProgramEnrollDate YYMMDD10.  
CompletedSomePostsecEd BEST1.  
AssociateDegreeDate YYMMDD10.  
BachelorDegreeDate YYMMDD10.  
MastersDegreeDate YYMMDD10.  
DegreeAboveMastersDate YYMMDD10.  
VTLicenseDate YYMMDD10.  
VTCertificateDate YYMMDD10.  
OtherLicOrCertDate YYMMDD10.  
PETSSStartDate YYMMDD10.  
jecVRAgencyStaff BEST1.  
jecVRAgencyPurchase BEST1.  
jecPurchaseProviderType BEST1.  
jecVRServicePurchaseExpenditure BEST5.  
wbleVRAgencyStaff BEST1.  
wbleVRAgencyPurchase BEST1.  
wblePurchaseProviderType BEST1.  
wbleVRServicePurchaseExpenditure BEST5.  
ceoVRAgencyStaff BEST1.  
ceoVRAgencyPurchase BEST1.  
ceoPurchaseProviderType BEST1.  
ceoVRServicePurchaseExpenditure BEST5.  
wrtVRAgencyStaff BEST1.  
wrtVRAgencyPurchase BEST1.  
wrtPurchaseProviderType BEST1.  
wrtVRServicePurchaseExpenditure BEST5.  
isaVRAgencyStaff BEST1.  
isaVRAgencyPurchase BEST1.  
isaPurchaseProviderType BEST1.  
isaVRServicePurchaseExpenditure BEST5.  
VRServiceStartDate YYMMDD10.  
CareerServiceDate YYMMDD10.  
gcutVRAgencyPurchase BEST1.  
gcutPurchaseProviderType BEST1.  
gcutVRTitleIFundsExpended BEST5.  
gcutCompServiceProvider BEST1.  
gcutCompServiceProviderType \$CHAR8.  
fycutVRAgencyPurchase BEST1.  
fycutPurchaseProviderType BEST1.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

fycutVRTitleIFundsExpended BEST6.  
fycutCompServiceProvider BEST1.  
fycutCompServiceProviderType \$CHAR8.  
jcctVRAgencyPurchase BEST1.  
jcctPurchaseProviderType BEST1.  
jcctVRTitleIFundsExpended BEST5.  
jcctCompServiceProvider BEST1.  
jcctCompServiceProviderType \$CHAR8.  
ovtVRAgencyStaff BEST1.  
ovtVRAgencyPurchase BEST1.  
ovtPurchaseProviderType BEST1.  
ovtVRTitleIFundsExpended BEST5.  
ovtCompServiceProvider BEST1.  
ovtCompServiceProviderType \$CHAR8.  
ojtVRAgencyStaff BEST1.  
ojtVRAgencyPurchase BEST1.  
ojtPurchaseProviderType BEST1.  
ojtVRTitleIFundsExpended BEST5.  
ojtCompServiceProvider BEST1.  
ojtCompServiceProviderType \$CHAR8.  
ratVRAgencyPurchase BEST1.  
ratPurchaseProviderType BEST1.  
ratVRTitleIFundsExpended BEST4.  
ratCompServiceProvider BEST1.  
ratCompServiceProviderType \$CHAR8.  
barltVRAgencyStaff BEST1.  
barltVRAgencyPurchase BEST1.  
barltPurchaseProviderType BEST1.  
barltVRTitleIFundsExpended BEST5.  
barltCompServiceProvider BEST1.  
barltCompServiceProviderType \$CHAR8.  
jrtVRAgencyStaff BEST1.  
jrtVRAgencyPurchase BEST1.  
jrtPurchaseProviderType BEST1.  
jrtVRTitleIFundsExpended BEST5.  
jrtCompServiceProvider BEST1.  
jrtCompServiceProviderType \$CHAR8.  
drstVRAgencyStaff BEST1.  
drstVRAgencyPurchase BEST1.  
drstPurchaseProviderType BEST1.  
drstVRTitleIFundsExpended BEST7.  
drstCompServiceProvider BEST1.  
drstCompServiceProviderType \$CHAR8.  
mtVRAgencyStaff BEST1.  
mtVRAgencyPurchase BEST1.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

mtPurchaseProviderType BEST1.  
mtVRTitleIFundsExpended BEST6.  
mtCompServiceProvider BEST1.  
mtCompServiceProviderType \$CHAR8.  
rsetVRAgencyStaff BEST1.  
rsetVRAgencyPurchase BEST1.  
rsetPurchaseProviderType BEST1.  
rsetVRTitleIFundsExpended BEST5.  
rsetCompServiceProvider BEST1.  
rsetCompServiceProviderType \$CHAR1.  
ctVRAgencyStaff BEST1.  
ctVRAgencyPurchase BEST1.  
ctPurchaseProviderType BEST1.  
ctVRTitleIFundsExpended BEST5.  
ctCompServiceProvider BEST1.  
ctCompServiceProviderType \$CHAR1.  
assVRAgencyStaff BEST1.  
assVRAgencyPurchase BEST1.  
assPurchaseProviderType BEST1.  
assVRTitleIFundsExpended BEST6.  
assCompServiceProvider BEST1.  
assCompServiceProviderType \$CHAR8.  
dtiVRAgencyStaff BEST1.  
dtiVRAgencyPurchase BEST1.  
dtiPurchaseProviderType BEST1.  
dtiVRTitleIFundsExpended BEST6.  
dtiCompServiceProvider BEST1.  
dtiCompServiceProviderType \$CHAR8.  
vrcgVRAgencyStaff BEST1.  
vrcgVRAgencyPurchase BEST1.  
vrcgPurchaseProviderType BEST1.  
vrcgVRTitleIFundsExpended BEST5.  
vrcgCompServiceProvider BEST1.  
vrcgCompServiceProviderType \$CHAR8.  
jsaVRAgencyStaff BEST1.  
jsaVRAgencyPurchase BEST1.  
jsaPurchaseProviderType BEST1.  
jsaVRTitleIFundsExpended BEST5.  
jsaCompServiceProvider BEST1.  
jsaCompServiceProviderType \$CHAR8.  
jpaVRAgencyStaff BEST1.  
jpaVRAgencyPurchase BEST1.  
jpaPurchaseProviderType BEST1.  
jpaVRTitleIFundsExpended BEST5.  
jpaCompServiceProvider BEST1.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```
jpaCompServiceProviderType $CHAR8.  
stjsVRAgencyStaff $CHAR1. /*Formerly BEST1.*/  
stjsVRAgencyPurchase BEST1.  
stjsPurchaseProviderType BEST1.  
stjsVRTitleIFundsExpended BEST5.  
stjsCompServiceProvider BEST1.  
stjsCompServiceProviderType $CHAR8.  
sesVRAgencyStaff BEST1.  
sesVRAgencyPurchase BEST1.  
sesPurchaseProviderType BEST1.  
sesVRTitleIFundsExpended BEST5.  
sesSETitleVIFundsExpended BEST5.  
sesCompServiceProvider BEST1.  
sesCompServiceProviderType $CHAR8.  
irsVRAgencyStaff BEST1.  
irsVRAgencyPurchase BEST1.  
irsPurchaseProviderType BEST1.  
irsVRTitleIFundsExpended BEST4.  
irsCompServiceProvider BEST1.  
irsCompServiceProviderType $CHAR8.  
bcVRAgencyStaff BEST1.  
bcVRAgencyPurchase BEST1.  
bcPurchaseProviderType BEST1.  
bcVRTitleIFundsExpended BEST5.  
bcCompServiceProvider BEST1.  
bcCompServiceProviderType $CHAR8.  
cesVRAgencyStaff BEST1.  
cesVRAgencyPurchase BEST1.  
cesPurchaseProviderType BEST1.  
cesVRTitleIFundsExpended BEST5.  
cesSETitleVIFundsExpended BEST5.  
cesCompServiceProvider BEST1.  
cesCompServiceProviderType $CHAR8.  
esVRAgencyStaff BEST1.  
esVRAgencyPurchase BEST1.  
esPurchaseProviderType BEST1.  
esVRTitleIFundsExpended BEST5.  
esSETitleVIFundsExpended BEST4.  
tranVRAgencyStaff BEST1.  
tranVRAgencyPurchase BEST1.  
tranPurchaseProviderType BEST1.  
tranVRTitleIFundsExpended BEST6.  
tranCompServiceProvider BEST1.  
tranCompServiceProviderType $CHAR8.  
mntVRAgencyStaff BEST1.
```



Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

mntVRAgencyPurchase BEST1.  
mntPurchaseProviderType BEST1.  
mntVRTitleIFundsExpended BEST5.  
mntCompServiceProvider BEST1.  
mntCompServiceProviderType \$CHAR8.  
rtVRAgencyStaff BEST1.  
rtVRAgencyPurchase BEST1.  
rtPurchaseProviderType BEST1.  
rtVRTitleIFundsExpended BEST6.  
rtCompServiceProvider BEST1.  
rtCompServiceProviderType \$CHAR8.  
pasVRAgencyStaff BEST1.  
pasVRAgencyPurchase BEST1.  
pasPurchaseProviderType BEST1.  
pasVRTitleIFundsExpended BEST6.  
pasCompServiceProvider BEST1.  
pasCompServiceProviderType \$CHAR8.  
tasVRAgencyStaff BEST1.  
tasVRAgencyPurchase BEST1.  
tasPurchaseProviderType BEST1.  
tasVRTitleIFundsExpended BEST6.  
tasCompServiceProvider BEST1.  
tasCompServiceProviderType \$CHAR8.  
rsVRAgencyStaff BEST1.  
rsVRAgencyPurchase BEST1.  
rsPurchaseProviderType BEST1.  
rsVRTitleIFundsExpended BEST5.  
rsCompServiceProvider BEST1.  
rsCompServiceProviderType \$CHAR8.  
isVRAgencyStaff BEST1.  
isVRAgencyPurchase BEST1.  
isPurchaseProviderType BEST1.  
isVRTitleIFundsExpended BEST6.  
isCompServiceProvider BEST1.  
isCompServiceProviderType \$CHAR8.  
osVRAgencyStaff BEST1.  
osVRAgencyPurchase BEST1.  
osPurchaseProviderType BEST1.  
osVRTitleIFundsExpended BEST6.  
osCompServiceProvider BEST1.  
osCompServiceProviderType \$CHAR8.  
EdFuncLevelDate YYMMDD10.  
SecondaryDate YYMMDD10.  
PostsecTransReportCardDate YYMMDD10.  
TrainingMilestoneDate YYMMDD10.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```

SkillGainSkillsProgDate YYMMDD10.
eoPrimOccupation BEST6.
exitDate      YYMMDD10.
exitType      BEST1.
exitReason    BEST2.
exitEmpOutcome BEST1.
exitPrimOccupation BEST6.
exitHourlyWage BEST6.
exitWeeklyHoursWorked BEST2.
peCredProgramEnrollDate YYMMDD10.
peCredAttainmentDate YYMMDD10.
peCredentialType BEST1.
q1aeEmpStatus BEST1.
q2aeEmpStatus BEST1.
q2aeWages     BEST9.
q3aeEmpStatus BEST1.
q4aeEmpStatus BEST1.
q4aeRetentionQ2AndQ4 BEST1.
appMonthlyPubSup $CHAR7.
appMedInsCov     $CHAR5.
exitMonthlyPubSup $CHAR7.
exitMedInsCov   $CHAR5.
IPEInitialDate  YYMMDD10.
IPEExtensionDate YYMMDD10.
EnrolledInSecEquiv BEST1.
CompDisenrollMSG YYMMDD10.
wblevVRAgencyStaff BEST1.
wblevVRAgencyPurchase BEST1.
wblevPurchaseProviderType BEST1.
wblevVRTitleIFundsExpended BEST5.
wblevCompServiceProvider BEST1.
wblevCompServiceProviderType $CHAR8.
ImportAgency   $CHAR4.
Report_Through YYMMDD10.
SSN             $CHAR9. ;
INFILE '\\ba.ad.ssa.gov\dcrdp\RemoteShares\MPR\2021 RSA 911 raw file from Dept of Education\SSACY2021.csv'
LRECL=32767
FIRSTOBS=2
ENCODING="WLATIN1"
DLM='2c'x
MISSOVER
DSD ;
INPUT
Year      : $CHAR6.
AgencyCode : ?? BEST3.

```

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```

IndividualID : $CHAR12.
ApplicationDate : ?? YMMDD8.
BirthDate : ?? YMMDD8.
Sex : ?? $CHAR1. /*Formerly BEST1.*/
AmerIndian : ?? BEST1.
Asian : ?? BEST1.
Black : ?? BEST1.
Hawaiian : ?? BEST1.
White : ?? BEST1.
Hispanic : ?? BEST1.
Veteran : ?? BEST1.
State : $CHAR2.
CountyFIPSCode : ?? BEST5.
ZIPCode : ?? BEST5.
Referral : ?? BEST2.
Student : ?? BEST1.
EligibilityDate : ?? YMMDD8.
EligibilityExt : ?? YMMDD8.
OOSPlacementDate : ?? YMMDD8.
OOSExitDate : ?? YMMDD8.
Disability : ?? BEST1.
PrimDisabilityCodeType : ?? BEST2.
PrimDisabilityCodeSource : ?? BEST2.
SecondaryDisabilityType : ?? BEST2.
SecondaryDisabilitySource : ?? BEST2.
DisabilitySigCode : ?? BEST1.
TWEStartDate : ?? YMMDD8.
TWEEndDate : ?? YMMDD8.
IPESupportedEmpGoal : ?? BEST1.
ipeEmpStatus : ?? BEST2.
ipePrimaryOcc : ?? COMMA6. /*Formerly BEST6.*/
ipeHourlyWage : ?? COMMA7.
ipeWeeklyHoursWorked : ?? COMMA5.
Adult : ?? BEST1.
AdultEd : ?? BEST1.
DislocatedWorker : ?? BEST1.
JobCorps : ?? BEST1.
VocRehab : ?? BEST1.
WPEmpService : ?? BEST1.
Youth : ?? BEST1.
YouthBuild : $CHAR14.
LongTermUnemp : ?? BEST1.
ExhaustTANF : ?? BEST1.
FosterCareYouth : ?? BEST1.
HomelessOrRunaway : ?? BEST1.

```

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

ExOffenderStatus : ?? BEST1.  
LowIncomeStatus : ?? BEST1.  
EnglishLearner : ?? BEST1.  
BasicSkillsDeficient : ?? BEST1.  
CulturalBarriers : ?? BEST1.  
SingleParent : ?? BEST1.  
DisHomemaker : ?? BEST1.  
MigrantFarmworker : ?? BEST1.  
StateDisStudentAgeRange : ?? TIME5.  
SchoolGradeCompleted : ?? BEST2.  
InSecondaryEd : ?? BEST1.  
SpecialEdCertCompDate : ?? YYYYMMDD8.  
SecSchoolDiplomaDate : ?? YYYYMMDD8.  
GEDDate : ?? YYYYMMDD8.  
EnrolledInPostsecEd : ?? BEST1.  
CredProgramEnrollDate : ?? YYYYMMDD8.  
CompletedSomePostsecEd : ?? BEST1.  
AssociateDegreeDate : ?? YYYYMMDD8.  
BachelorDegreeDate : ?? YYYYMMDD8.  
MastersDegreeDate : ?? YYYYMMDD8.  
DegreeAboveMastersDate : ?? YYYYMMDD8.  
VTLicenseDate : ?? YYYYMMDD8.  
VTCertificateDate : ?? YYYYMMDD8.  
OtherLicOrCertDate : ?? YYYYMMDD8.  
PETSStartDate : ?? YYYYMMDD8.  
jecVRAgencyStaff : ?? BEST1.  
jecVRAgencyPurchase : ?? BEST1.  
jecPurchaseProviderType : ?? BEST1.  
jecVRServicePurchaseExpenditure : ?? BEST5.  
wbleVRAgencyStaff : ?? BEST1.  
wbleVRAgencyPurchase : ?? BEST1.  
wblePurchaseProviderType : ?? BEST1.  
wbleVRServicePurchaseExpenditure : ?? BEST5.  
ceoVRAgencyStaff : ?? BEST1.  
ceoVRAgencyPurchase : ?? BEST1.  
ceoPurchaseProviderType : ?? BEST1.  
ceoVRServicePurchaseExpenditure : ?? BEST5.  
wrtVRAgencyStaff : ?? BEST1.  
wrtVRAgencyPurchase : ?? BEST1.  
wrtPurchaseProviderType : ?? BEST1.  
wrtVRServicePurchaseExpenditure : ?? BEST5.  
isaVRAgencyStaff : ?? BEST1.  
isaVRAgencyPurchase : ?? BEST1.  
isaPurchaseProviderType : ?? BEST1.  
isaVRServicePurchaseExpenditure : ?? BEST5.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

VRServiceStartDate : ?? YYMMDD8.  
CareerServiceDate : ?? YYMMDD8.  
gcutVRAgencyPurchase : ?? BEST1.  
gcutPurchaseProviderType : ?? BEST1.  
gcutVRTitleIFundsExpended : ?? BEST5.  
gcutCompServiceProvider : ?? BEST1.  
gcutCompServiceProviderType : \$CHAR8.  
fycutVRAgencyPurchase : ?? BEST1.  
fycutPurchaseProviderType : ?? BEST1.  
fycutVRTitleIFundsExpended : ?? BEST6.  
fycutCompServiceProvider : ?? BEST1.  
fycutCompServiceProviderType : \$CHAR8.  
jcctVRAgencyPurchase : ?? BEST1.  
jcctPurchaseProviderType : ?? BEST1.  
jcctVRTitleIFundsExpended : ?? BEST5.  
jcctCompServiceProvider : ?? BEST1.  
jcctCompServiceProviderType : \$CHAR8.  
ovtVRAgencyStaff : ?? BEST1.  
ovtVRAgencyPurchase : ?? BEST1.  
ovtPurchaseProviderType : ?? BEST1.  
ovtVRTitleIFundsExpended : ?? BEST5.  
ovtCompServiceProvider : ?? BEST1.  
ovtCompServiceProviderType : \$CHAR8.  
ojtVRAgencyStaff : ?? BEST1.  
ojtVRAgencyPurchase : ?? BEST1.  
ojtPurchaseProviderType : ?? BEST1.  
ojtVRTitleIFundsExpended : ?? BEST5.  
ojtCompServiceProvider : ?? BEST1.  
ojtCompServiceProviderType : \$CHAR8.  
ratVRAgencyPurchase : ?? BEST1.  
ratPurchaseProviderType : ?? BEST1.  
ratVRTitleIFundsExpended : ?? BEST4.  
ratCompServiceProvider : ?? BEST1.  
ratCompServiceProviderType : \$CHAR8.  
barltVRAgencyStaff : ?? BEST1.  
barltVRAgencyPurchase : ?? BEST1.  
barltPurchaseProviderType : ?? BEST1.  
barltVRTitleIFundsExpended : ?? BEST5.  
barltCompServiceProvider : ?? BEST1.  
barltCompServiceProviderType : \$CHAR8.  
jrtVRAgencyStaff : ?? BEST1.  
jrtVRAgencyPurchase : ?? BEST1.  
jrtPurchaseProviderType : ?? BEST1.  
jrtVRTitleIFundsExpended : ?? BEST5.  
jrtCompServiceProvider : ?? BEST1.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

jrtCompServiceProviderType : \$CHAR8.  
drstVRAgencyStaff : ?? BEST1.  
drstVRAgencyPurchase : ?? BEST1.  
drstPurchaseProviderType : ?? BEST1.  
drstVRTitleIFundsExpended : ?? BEST7.  
drstCompServiceProvider : ?? BEST1.  
drstCompServiceProviderType : \$CHAR8.  
mtVRAgencyStaff : ?? BEST1.  
mtVRAgencyPurchase : ?? BEST1.  
mtPurchaseProviderType : ?? BEST1.  
mtVRTitleIFundsExpended : ?? BEST6.  
mtCompServiceProvider : ?? BEST1.  
mtCompServiceProviderType : \$CHAR8.  
rsetVRAgencyStaff : ?? BEST1.  
rsetVRAgencyPurchase : ?? BEST1.  
rsetPurchaseProviderType : ?? BEST1.  
rsetVRTitleIFundsExpended : ?? BEST5.  
rsetCompServiceProvider : ?? BEST1.  
rsetCompServiceProviderType : \$CHAR1.  
ctVRAgencyStaff : ?? BEST1.  
ctVRAgencyPurchase : ?? BEST1.  
ctPurchaseProviderType : ?? BEST1.  
ctVRTitleIFundsExpended : ?? BEST5.  
ctCompServiceProvider : ?? BEST1.  
ctCompServiceProviderType : \$CHAR1.  
assVRAgencyStaff : ?? BEST1.  
assVRAgencyPurchase : ?? BEST1.  
assPurchaseProviderType : ?? BEST1.  
assVRTitleIFundsExpended : ?? BEST6.  
assCompServiceProvider : ?? BEST1.  
assCompServiceProviderType : \$CHAR8.  
dtiVRAgencyStaff : ?? BEST1.  
dtiVRAgencyPurchase : ?? BEST1.  
dtiPurchaseProviderType : ?? BEST1.  
dtiVRTitleIFundsExpended : ?? BEST6.  
dtiCompServiceProvider : ?? BEST1.  
dtiCompServiceProviderType : \$CHAR8.  
vrcgVRAgencyStaff : ?? BEST1.  
vrcgVRAgencyPurchase : ?? BEST1.  
vrcgPurchaseProviderType : ?? BEST1.  
vrcgVRTitleIFundsExpended : ?? BEST5.  
vrcgCompServiceProvider : ?? BEST1.  
vrcgCompServiceProviderType : \$CHAR8.  
jsaVRAgencyStaff : ?? BEST1.  
jsaVRAgencyPurchase : ?? BEST1.

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```
jsaPurchaseProviderType : ?? BEST1.  
jsaVRTitleIFundsExpended : ?? BEST5.  
jsaCompServiceProvider : ?? BEST1.  
jsaCompServiceProviderType : $CHAR8.  
jpaVRAgencyStaff : ?? BEST1.  
jpaVRAgencyPurchase : ?? BEST1.  
jpaPurchaseProviderType : ?? BEST1.  
jpaVRTitleIFundsExpended : ?? BEST5.  
jpaCompServiceProvider : ?? BEST1.  
jpaCompServiceProviderType : $CHAR8.  
stjsVRAgencyStaff : ?? $CHAR1. /*Formerly BEST1.*/  
stjsVRAgencyPurchase : ?? BEST1.  
stjsPurchaseProviderType : ?? BEST1.  
stjsVRTitleIFundsExpended : ?? BEST5.  
stjsCompServiceProvider : ?? BEST1.  
stjsCompServiceProviderType : $CHAR8.  
sesVRAgencyStaff : ?? BEST1.  
sesVRAgencyPurchase : ?? BEST1.  
sesPurchaseProviderType : ?? BEST1.  
sesVRTitleIFundsExpended : ?? BEST5.  
sesSETitleVIFundsExpended : ?? BEST5.  
sesCompServiceProvider : ?? BEST1.  
sesCompServiceProviderType : $CHAR8.  
irsVRAgencyStaff : ?? BEST1.  
irsVRAgencyPurchase : ?? BEST1.  
irsPurchaseProviderType : ?? BEST1.  
irsVRTitleIFundsExpended : ?? BEST4.  
irsCompServiceProvider : ?? BEST1.  
irsCompServiceProviderType : $CHAR8.  
bcVRAgencyStaff : ?? BEST1.  
bcVRAgencyPurchase : ?? BEST1.  
bcPurchaseProviderType : ?? BEST1.  
bcVRTitleIFundsExpended : ?? BEST5.  
bcCompServiceProvider : ?? BEST1.  
bcCompServiceProviderType : $CHAR8.  
cesVRAgencyStaff : ?? BEST1.  
cesVRAgencyPurchase : ?? BEST1.  
cesPurchaseProviderType : ?? BEST1.  
cesVRTitleIFundsExpended : ?? BEST5.  
cesSETitleVIFundsExpended : ?? BEST5.  
cesCompServiceProvider : ?? BEST1.  
cesCompServiceProviderType : $CHAR8.  
esVRAgencyStaff : ?? BEST1.  
esVRAgencyPurchase : ?? BEST1.  
esPurchaseProviderType : ?? BEST1.
```

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

esVRTitleIFundsExpended : ?? BEST5.  
esSETitleVIFundsExpended : ?? BEST4.  
tranVRAgencyStaff : ?? BEST1.  
tranVRAgencyPurchase : ?? BEST1.  
tranPurchaseProviderType : ?? BEST1.  
tranVRTitleIFundsExpended : ?? BEST6.  
tranCompServiceProvider : ?? BEST1.  
tranCompServiceProviderType : \$CHAR8.  
mntVRAgencyStaff : ?? BEST1.  
mntVRAgencyPurchase : ?? BEST1.  
mntPurchaseProviderType : ?? BEST1.  
mntVRTitleIFundsExpended : ?? BEST5.  
mntCompServiceProvider : ?? BEST1.  
mntCompServiceProviderType : \$CHAR8.  
rtVRAgencyStaff : ?? BEST1.  
rtVRAgencyPurchase : ?? BEST1.  
rtPurchaseProviderType : ?? BEST1.  
rtVRTitleIFundsExpended : ?? BEST6.  
rtCompServiceProvider : ?? BEST1.  
rtCompServiceProviderType : \$CHAR8.  
pasVRAgencyStaff : ?? BEST1.  
pasVRAgencyPurchase : ?? BEST1.  
pasPurchaseProviderType : ?? BEST1.  
pasVRTitleIFundsExpended : ?? BEST6.  
pasCompServiceProvider : ?? BEST1.  
pasCompServiceProviderType : \$CHAR8.  
tasVRAgencyStaff : ?? BEST1.  
tasVRAgencyPurchase : ?? BEST1.  
tasPurchaseProviderType : ?? BEST1.  
tasVRTitleIFundsExpended : ?? BEST6.  
tasCompServiceProvider : ?? BEST1.  
tasCompServiceProviderType : \$CHAR8.  
rsVRAgencyStaff : ?? BEST1.  
rsVRAgencyPurchase : ?? BEST1.  
rsPurchaseProviderType : ?? BEST1.  
rsVRTitleIFundsExpended : ?? BEST5.  
rsCompServiceProvider : ?? BEST1.  
rsCompServiceProviderType : \$CHAR5.  
isVRAgencyStaff : ?? BEST1.  
isVRAgencyPurchase : ?? BEST1.  
isPurchaseProviderType : ?? BEST1.  
isVRTitleIFundsExpended : ?? BEST6.  
isCompServiceProvider : ?? BEST1.  
isCompServiceProviderType : \$CHAR8.  
osVRAgencyStaff : ?? BEST1.



## Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```
osVRAgencyPurchase : ?? BEST1.
osPurchaseProviderType : ?? BEST1.
osVRTitleIFundsExpended : ?? BEST6.
osCompServiceProvider : ?? BEST1.
osCompServiceProviderType : $CHAR8.
EdFuncLevelDate : ?? YYMMDD8.
SecondaryDate : ?? YYMMDD8.
PostsecTransReportCardDate : ?? YYMMDD8.
TrainingMilestoneDate : ?? YYMMDD8.
SkillGainSkillsProgDate : ?? YYMMDD8.
eoPrimOccupation : ?? BEST6.
exitDate : ?? YYMMDD8.
exitType : ?? BEST1.
exitReason : ?? BEST2.
exitEmpOutcome : ?? BEST1.
exitPrimOccupation : ?? BEST6.
exitHourlyWage : ?? COMMA6.
exitWeeklyHoursWorked : ?? BEST2.
peCredProgramEnrollDate : ?? YYMMDD8.
peCredAttainmentDate : ?? YYMMDD8.
peCredentialType : ?? BEST1.
q1aeEmpStatus : ?? BEST1.
q2aeEmpStatus : ?? BEST1.
q2aeWages : ?? COMMA9.
q3aeEmpStatus : ?? BEST1.
q4aeEmpStatus : ?? BEST1.
q4aeRetentionQ2AndQ4 : ?? BEST1.
appMonthlyPubSup : $CHAR7.
appMedInsCov : $CHAR5.
exitMonthlyPubSup : $CHAR7.
exitMedInsCov : $CHAR5.
IPEInitialDate : ?? YYMMDD8.
IPEExtensionDate : ?? YYMMDD8.
EnrolledInSecEquiv : ?? BEST1.
CompDisenrollMSG : ?? YYMMDD8.
wblevVRAgencyStaff : ?? BEST1.
wblevVRAgencyPurchase : ?? BEST1.
wblevPurchaseProviderType : ?? BEST1.
wblevVRTitleIFundsExpended : ?? BEST5.
wblevCompServiceProvider : ?? BEST1.
wblevCompServiceProviderType : $CHAR8.
ImportAgency : $CHAR4.
Report_Through : ?? YYMMDD8.
SSN : $CHAR9. ;
```

RUN;

Appendix A.178

JCL/SAS Code: Read\_RSA\_CSV

```
%mend read;
```

```
%read;
```

```
proc contents data = OUT.RSA21; run;
```

```
proc print data = OUT.RSA21 (obs = 10);
```

```
where ranuni(1100) < 0.01;
```

```
run;
```

```
/*PROC SORT DATA=OUT.RSA21; BY SSN; RUN;*/
```

## Appendix A.179

JCL/SAS Code: RSAUPLOAD

```

*****
* PROJECT: DAF21
* PROJECT
* DIRECTOR: MATTHEW URATO
*
* PROGRAM: M:\DAF21\TASK 17 Create DAF-RSA Files\Programs\RSAUPLOAD.sas
*
*
* DESCRIPTION: UPLOADS RSA DATA
*
* DATE: 07/16/2021 ALEX BRYCE
* Updated: 02/16/2023 SVETLANA BRONNIKOV
***** ;
/*%let rootloc = \\ba.ad.ssa.gov\dcrdp\RemoteShares\MPR\DAF21\TASK 17 Create DAF-RSA Files\Programs;

proc printto
  log = "&rootloc.\RSAUPLOAD.log"
  new
  print = "&rootloc.\RSAUPLOAD.lst" new;
run;*/

%include '\\ba.ad.ssa.gov\dcrdp\RemoteShares\MPR\Bronnikov\SBRONNIKOV_PW.sas';

%let UID = #3590;

options comamid=tcp; %let SY56=SY56.sspf.ssa.gov; %let tcpsec=_prompt_; options remote=sy56.spawner; run;
OPTIONS NOFMterr label nocenter PAGESIZE=MAX LINESIZE=MAX OBS=max
compress=binary formchar="|---|+|---+=|-\<>*";

signon sy56.spawner user=&TSO_UserID. password=&TSO_Password.;

/*set macro vars*/

%let libloc = \\ba.ad.ssa.gov\dcrdp\RemoteShares\MPR\DAF21\RSA\upd;

SIGNON ;
libname local "&libloc.";
rsubmit;
libname MISFDATA "OPDR.TG.PRD.ETTW.#3590.RSA21NEW.SA.V1";
option compress=binary;

***** ;
/* Use the firstobs and obs control only if needed. For example, if the first download */
/* after the 499th record, then the code below will start downloading at the 500th */

```

Appendix A.179

JCL/SAS Code: RSAUPLOAD

```
/* observation and go until it completes or crashes. */
*****,
```

```
proc upload data=local.RSA21 OUT=MISFDATA.RSA21 (compress=binary);
```

```
RUN;
endrsubmit;
signoff;
```

## Appendix A.180

JCL/SAS Code: RSACOMB

```

//#3590RSA JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: BUILD DAF21
//* PROJECT
//* DIRECTOR:  MATT URATO
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#3590.DAF21.RSA.PRDLIB(RSACOMB)
//*
//* DESCRIPTION: COMBINED RSA SAS LOADED FILES
//*
//* DATE:     02/16/23 SVETLANA BRONNIKOV
//*****
//*
//RSA   EXEC SAS9,
//      WORK='120000,60000'
//*
//* RSA FILE - OLD FORMAT ALL YEARS (DO NOT CHANGE)
//RSAPREV DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF17P.RSAALLOF.SA.V1,DISP=SHR
//* RSA FILE - NEW FORMAT ALL YEARS (PREVIOUS YEAR RSACOMB RSA OUTPUT)
//RSA20  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20.RSAALL.SA.V1,DISP=SHR
//* UPLOADED RSA FILE FROM CURRENT YEAR OF DAF PROCESSING - NEW FORMAT
//RSA21  DD DSN=OPDR.TG.PRD.ETTW.#3590.RSA21NEW.SA.V1,DISP=SHR
//*
//* RSA FILE IN NEW FORMAT
//RSA   DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSAALL.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(4000,2000),RLSE)
//* EVS FILE WITH SSN DOB GENDER
//EVS   DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAEVS.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(4000,2000),RLSE)
//*
//FIN   DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAFIN.FL.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//*
//SYSIN DD *

PROC FORMAT;
  VALUE $SEX '1'='M'
           '2'='F'

```

## Appendix A.180

JCL/SAS Code: RSACOMB

```
    ;
RUN;

* SORT RSA DATA;
PROC SORT DATA=RSA21.RSA21 OUT=TEMP1.RSA21;
BY SSN;
RUN;

* NEW FORMAT RSA DATA;
* NOTE: SET UPLOADED RSA DATA WITH NEW FORMAT RSAALL FROM PREV YEAR;
DATA RSA.RSAALL;
  SET RSA20.RSAALL
    TEMP1.RSA21 (RENAME=(BIRTHDATE=RSADOB SEX=RSASEX))
  ;
  BY SSN;
RUN;

* BUILDS EVS FILE WITH SSN DOB GENDER;
* NOTE: SET NEW FORMAT RSA (RSA.RSAALL) WITH OLD FORMAT RSA (RSAPREV);
DATA EVS.RSAEVS (RENAME = (RSADOB = DOB RSASEX = GENDER));
  SET RSAPREV.RSAALL (KEEP = SSN RSADOB RSASEX)
    RSA.RSAALL (KEEP = SSN RSADOB RSASEX);
  BY SSN;
  FORMAT RSADOB MMDDYY10.
    RSASEX $SEX.
  ;
  DOBYEAR=YEAR(RSADOB);

  IF LAST.SSN THEN DO;
    FILE FIN;
    PUT @001 SSN $CHAR9.
  ;
  END;
RUN;

PROC FREQ DATA=EVS.RSAEVS;
TABLES DOBYEAR*GENDER / LIST MISSING;WHERE SSN = "";
RUN;

PROC CONTENTS DATA=RSA.RSAALL ORDER=VARNUM;
RUN;

PROC CONTENTS DATA=EVS.RSAEVS;
RUN;
```

Appendix A.180

JCL/SAS Code: RSACOMB

```
TITLE1 'RSA: OLD FORMAT';  
PROC PRINT DATA=RSAPREV.RSAALL (OBS=50);  
RUN;
```

```
TITLE1 'RSA: NEW FORMAT';  
PROC PRINT DATA=RSA.RSAALL (OBS=50);  
RUN;
```

```
TITLE1 'RSA EVS: COMBINED OLD AND NEW FORMAT';  
PROC PRINT DATA = EVS.RSAEVS(OBS = 50);  
RUN;
```

## Appendix A.181

JCL/SAS Code: READNUM

```
//#3590NUM JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.RSA.PRDLIB(READNUM)
//* DESCRIPTION: LOAD RAW RSA NUMIDENT DATA INTO SAS
//* PROGRAMMER:
//* E-MAIL SBRONNIKOV@MATHEMATICA-MPR.COM
//* MATHEMATICA PHONE: 202-264-4221
//*****
// SET REG='64M'
//*
//JS010 EXEC SAS9,
//  WORK='200000,100000',
//  REGION=&REG,
//  PARM='MEMSIZE=&REG'
//*
//IN1  DD DSN=OPDR.TG.PRD.ETTW.NUMI.RSADAF21.FLAT.R2303,DISP=SHR
//TEMP DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO
//TEMP1 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSANUM.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  SPACE=(CYL,(1000,100),RLSE)
//*
OPTIONS NOCENTER COMPRESS=YES OBS=MAX;
/* READ IN NUMIDENT DATA */
DATA TEMP.RSANUM;
INFILE IN1 TRUNCOVER;
INPUT @027 REC $1.
@;
/* READ DATA FROM DIFFERENT RECORD TYPES */
IF REC IN ('G' 'J' 'K' 'O' '1' '2' '4' '5' '9'
          'P' 'S' 'W' 'V') THEN DO;
INPUT @13 SSN $9.
      @27 ENTCD $1.
      @29 CYCDTE $8.
      @37 FRSTNAM $15.
      @53 MIDNAM $15.
      @69 LASTNAM $20.
      @90 NAMESUF $4.
```



## Appendix A.181

JCL/SAS Code: READNUM

```
        @97 DOBR  $8.
        @105 SEXR $1.
        @106 RACER $1.
    ;
END;
ELSE IF REC IN ('D' 'L') THEN DO;
INPUT @13 SSN  $9.
    @27 ENTCD  $1.
    @29 CYCDTE $8.
    @37 FRSTNAM $15.
    @53 MIDNAM $15.
    @69 LASTNAM $20.
    @90 NAMESUF $4.
    @97 DOBR  $8.
    @105 SEXR $1.
    @298 DODR  $8.
;
END;
ELSE IF REC = 'T' THEN DO;
INPUT @13 SSN  $9.
    @27 ENTCD  $1.
    @29 CYCDTE $8.
    @37 FRSTNAM $15.
    @53 MIDNAM $15.
    @69 LASTNAM $20.
    @90 NAMESUF $4.
    @97 DOBR  $8.
    @105 SEXR $1.
    @145 DODR  $8.
;
END;
ELSE IF REC = 'E' THEN DO;
INPUT @13 SSN  $9.
    @27 ENTCD  $1.
    @29 CYCDTE $8.
    @37 FRSTNAM $15.
    @53 MIDNAM $15.
    @69 LASTNAM $20.
    @90 NAMESUF $4.
    @97 DOBR  $8.
;
END;

/* IF SSN NE ' ' */
RUN;
```

## Appendix A.181

JCL/SAS Code: READNUM

```
/* ORDER THE RECORDS CHRONOLOGICALLY */
PROC FREQ DATA=TEMP.RSANUM; TABLES REC / LIST MISSING; RUN;
PROC SORT DATA=TEMP.RSANUM OUT=TEMP1.RSANUM; BY SSN CYCDTE;
  WHERE SSN NE ' '; RUN;
PROC PRINT DATA=TEMP1.RSANUM (OBS=100); TITLE 'AFTER SORT';
RUN;
DATA OUT1.RSANUM(KEEP=SSN
  ENTLST
  CYCDTLST
  DOB
  SEX
  RACE
  FRSTNAME
  MIDNAME
  LASTNAME
  NAMESUFFIX
  DOD )
  ;
SET TEMP1.RSANUM; BY SSN CYCDTE;

/* DATA IS ORDERED BY CYCLE DATE FOR EACH SSN      */
/* READ THROUGH ALL RECORDS AND KEEP THE NON-BLANKS */

LENGTH
  ENTLST $1
  CYCDTLST $8
  DOB $8
  SEX $1
  RACE $1
  FRSTNAME $15
  MIDNAME $15
  LASTNAME $20
  NAMESUFFIX $4
  DOD $8
  ;
RETAIN
  ENTLST
  CYCDTLST
  DOB
  SEX
  RACE
  FRSTNAME
  MIDNAME
  LASTNAME
  NAMESUFFIX
```

## Appendix A.181

JCL/SAS Code: READNUM

```
      DOD
      ;
      ARRAY ALL(*) ENTLST
          CYCDTLST
          DOB
          SEX
          RACE
          FRSTNAME
          MIDNAME
          LASTNAME
          NAMESUF
          DOD;
      IF FIRST.SSN THEN DO;
      DO I = 1 TO DIM(ALL);
      ALL(I) = ' ';
      END;
      END;
      IF DOBR NE '' THEN DOB = DOBR;
      IF DODR NE '' THEN DOD = DODR;
      IF SEXR NE '' THEN SEX = SEXR;
      IF RACER GT '0' THEN RACE = RACER;
      ENTLST = ENTCD;
      CYCDTLST = CYCDTE;
      IF FRSTNAM NE '' THEN FRSTNAME = FRSTNAM;
      IF MIDNAM NE '' THEN MIDNAME = MIDNAM;
      IF LASTNAM NE '' THEN LASTNAME = LASTNAM;
      IF NAMESUF NE '' THEN NAMESUF = NAMESUF;
      IF LAST.SSN;
      RUN;
      PROC PRINT DATA=OUT1.RSANUM(OBS=50); TITLE 'AFTER LAST'; RUN;
      PROC FREQ DATA=OUT1.RSANUM; TABLES ENTLST SEX RACE; RUN;
```

Appendix A.182  
JCL/SAS Code: EVS

```
//#3590EVS JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,  
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590  
//*****  
//* PROJECT:   DAF21  
//* PROJECT  
//* DIRECTOR:  MATT URATO  
//* PROGRAM:   OPDR.TG.PRD.ETTW.#3590.DAF21.RSA.PRDLIB(EVS)  
//* DESCRIPTION: MPR-EVS SYSTEM -- VERIFIES SSNS BY COMPARING  
//*            SSN, DOB, AND/OR GENDER ON A SUPPLIED FILE  
//*            AGAINST THE RSA NUMIDENT FILE  
//*****  
//*  
//EVS EXEC SAS9,  
//  WORK='120000,60000'  
//*  
//* RSA NUMIDENT FILE  
//IN1 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSANUM.SA.V1,DISP=SHR  
//*  
//* FILE TO BE VERIFIED  
//IN2 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAEVS.SA.V1,DISP=SHR  
//*  
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAEVS.RP.V1,  
//  DISP=(OLD,CATLG,DELETE),  
//  SPACE=(CYL,(4000,2000),RLSE)  
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAEVS.SA.V1,  
//  DISP=(OLD,CATLG,DELETE),  
//  SPACE=(CYL,(4000,2000),RLSE)  
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),  
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)  
//SYSIN DD *
```

OPTIONS LS=132 COMPRESS=YES MPRINT OBS = MAX;

```
*****  
*  
* VERIFY CODE MAPPING  
*  
* blank = initial value  
* S = no match in TRF.Demo  
* N = fails name check  
* G = fails gender check  
* Y = passes EVS  
* B = fails date of birth check  
* D = duplicate SSN
```

## Appendix A.182

JCL/SAS Code: EVS

```

* D = duplicate SSN, DOB, GENDER, NAME or duplicate SSN, DOB, NAME (all
*   fields used for EVS are duplicates)
*
*
*****
%LET NAME_CHK=0;  /*** 0=Don't perform name check / 1=Do ***/

/*****/
/** Macro to store number of observations in a given dataset ***/
/*****/
%MACRO NUMOBS(ds,mvar,condition);
DATA TEMP.TEMP;
  SET &ds.&condition.;
RUN;

%global &mvar.;
%let dsid = %sysfunc(open(TEMP.TEMP));
%let &mvar. =%sysfunc(attrn(&dsid.,NOBS));
%let rc = %sysfunc(close(&dsid));
%MEND;

/*****/
/** Merge with submitted data      ***/
/*****/
DATA TEMP.SUBMITTED_DATA;
SET IN2.RSAEVS;
IF GENDER = "1" THEN GENDER = "M";
ELSE IF GENDER = "2" THEN GENDER = "F";
FORMAT _ALL_ ;
RUN;

PROC CONTENTS DATA=TEMP.SUBMITTED_DATA;
RUN;

DATA TEMP.SUBMITTED_DATA_MATCH TEMP.WD (KEEP=DOBRNUM);
MERGE TEMP.SUBMITTED_DATA(IN=IN1)
      IN1.RSANUM(IN=IN2
                KEEP=SSN DOB SEX
                RENAME=(DOB=DOBRNUM SEX=ORIG_SEX)
                );
BY SSN;
IF IN1;
IF IN1 AND IN2 THEN MRG_SSN=1;
ELSE DO;

```

## Appendix A.182

JCL/SAS Code: EVS

```
MRG_SSN=0;
VER_CODE="S";
END;
```

```
IF ORIG_SEX = '1' THEN SEX = 'M';
ELSE IF ORIG_SEX = '2' THEN SEX = 'F';
ELSE SEX = ' ';
```

```
***SCREEN OUT CASES WHICH HAVE THE WRONG DATA
FOR DOBRNUM IN NUMIDENT FILE***;
```

```
IF NOTDIGIT(TRIM(SUBSTR(DOBRNUM,5,4))) = 0
THEN NYR = SUBSTR(DOBRNUM,5,4);
NUMYR = INPUT(NYR,8.);
```

```
IF NOTDIGIT(TRIM(DOBRNUM)) = 0 THEN DO;
DOBNUM = INPUT(DOBRNUM,MMDDYY10.);
IF _ERROR_=1 THEN DO; OUTPUT TEMP.WD; END;
END;
FORMAT DOBNUM MMDDYY10. SEX;
OUTPUT TEMP.SUBMITTED_DATA_MATCH;
RUN;
```

```
TITLE "CHECK BAD DATES";
PROC FREQ DATA=TEMP.WD; TABLES DOBRNUM / LIST MISSING; RUN; TITLE;
```

```
DATA TEMP.SUBMITTED_DATA_MATCH;
SET TEMP.SUBMITTED_DATA_MATCH;
BY SSN;
```

```
NAME_CHK = &name_chk.;
```

```
DOB_SRCE=" ";
TRF_NAME=.;
```

```
/**/ Proceed only if match during merge /**/
IF MRG_SSN=1 /*AND VFLAG=0*/ THEN DO;
```

```
/**/ Verify Name /**/
IF &name_chk.=1 THEN DO;
IF COMPRESS(BGN)="" OR COMPRESS(BLN)="" THEN DO;
TRF_NAME=0;
VER_CODE="N";
END;
ELSE DO;
```

## Appendix A.182

JCL/SAS Code: EVS

```
IF UPCASE(SUBSTR(LNAME,1,7)) NE UPCASE(SUBSTR(BLN,1,7))
THEN DO;
  TRF_NAME=0;
  VER_CODE="N";
END;
ELSE DO;
  IF UPCASE(SUBSTR(FNAME,1,1)) NE UPCASE(SUBSTR(BGN,1,1))
  THEN DO;
    TRF_NAME=0;
    VER_CODE="N";
  END;
  ELSE DO;
    IF COMPRESS(SUBSTR(INIT,1,1)) NE ""
    AND COMPRESS(SUBSTR(BMI,1,1)) NE "" THEN DO;
      IF UPCASE(SUBSTR(INIT,1,1))=UPCASE(SUBSTR(BMI,1,1))
      THEN TRF_NAME=1;
    ELSE DO;
      TRF_NAME=0;
      VER_CODE="N";
    END;
  END;
  ELSE DO;
    TRF_NAME=1;
  END;
END;
END;
END; /*** end name check ***/

/*** proceed on only if name checking was sucessful (or not done)*/
IF (&name_chk.=1 AND TRF_NAME=1) OR &name_chk.=0 THEN DO;

  /*** Verify DOB ***/
  IF /*MONTH(DOB)=MONTH(DOBMBR) AND*/ YEAR(DOB)=NUMYR
  AND NUMYR NE . THEN DOB_SRCE="R";
  ELSE VER_CODE="B";

  * IF RECORD PASSES DOB CHECK;
  IF DOB_SRCE = "R" THEN DO;

  /*** Verify Gender ***/
  * IF RSA DOES NOT HAVE M,F GENDER THEN PASS;
  IF UPCASE(GENDER) NOTIN ("M","F") THEN DO;
    GENDER_VERIFY="Y";
    VER_CODE="Y";
```

## Appendix A.182

JCL/SAS Code: EVS

```
END;

* IF DAF HAS M,F GENDER AND RSA HAS M,F GENDER THEN PASS;
ELSE IF UPCASE(SEX) IN ("M","F") THEN DO;
  IF UPCASE(SEX)=UPCASE(GENDER) THEN DO;
    GENDER_VERIFY="Y";
    VER_CODE="Y";
  END;
  * IF INPUT SEX DOES NOT HAVE EQUAL RSA GENDER;
ELSE DO;
  GENDER_VERIFY="N";
  VER_CODE="G";
END;
END;

* RSA HAS M,F AND DAF DOES NOT HAVE M,F GENDER THEN PASS;
ELSE DO;
  GENDER_VERIFY="Y";
  VER_CODE="Y";
END;
END;
/*** If case makes it this far then it passes verification ***/
IF DOB_SRCE NE " " AND GENDER_VERIFY="Y" THEN DO;
  VER_CODE="Y";
  VFLAG=1;
END;
END;
END; /*** end of MRG_SSN=1 ***/
RUN;

PROC FREQ;
  TABLES MRG_SSN*VER_CODE*TRF_NAME*DOB_SRCE*GENDER_VERIFY
  /MISSING LIST;
  TABLES VER_CODE*GENDER*SEX
  /MISSING LIST;
RUN;

/*****
/***   Output File       ***
*****/

DATA OUT2.MPR_EVS4;
  SET TEMP.SUBMITTED_DATA_MATCH;
  KEEP /*STATUS*/ SSN FNAME INIT LNAME DOB GENDER
  VER_CODE NAME_CHK MRG_SSN TRF_NAME DOB_SRCE
```



## Appendix A.182

JCL/SAS Code: EVS

```

        DOBNUM SEX DOBYEAR
    ;
RUN;

/*****/
/**   Output Report       ***/
/*****/
%NUMOBS(TEMP.SUBMITTED_DATA,LINE1,);
%NUMOBS(OUT2.MPR_EVS4,LINE5,);
%NUMOBS(OUT2.MPR_EVS4,LINE6,%STR( (WHERE=(VER_CODE='Y')) ));
%NUMOBS(OUT2.MPR_EVS4,LINE7,%STR( (WHERE=(VER_CODE NE 'Y')) ));
%NUMOBS(OUT2.MPR_EVS4,LINE8,%STR( (WHERE=(VER_CODE='S')) ));
%NUMOBS(OUT2.MPR_EVS4,LINE10,%STR( (WHERE=(VER_CODE='N')) ));
%NUMOBS(OUT2.MPR_EVS4,LINE11,%STR( (WHERE=(VER_CODE='B')) ));
%NUMOBS(OUT2.MPR_EVS4,LINE12,%STR( (WHERE=(VER_CODE='G')) ));

DATA _NULL_;
FILE OUT1;

PUT "MPR-EVS: Summary Results &SYSDATE.";
put "for Submitted File: SUBMITTED_DATA";
IF &name_chk.=0 THEN PUT "Name checking was NOT performed";
ELSE PUT "Name checking WAS performed";
PUT;

PUT "&line1.=Total Input records";
PUT "&line5.=Output records";
PUT "&line6.=Input records that verified";
PUT "&line7.=Input records that did not verify";
PUT "&line8.=Input records that did not merge with TRF.DEMO";
PUT "&line10.=Input records that did not verify due to name";
PUT "&line11.=Input records that did not verify due to date of birth";
PUT "&line12.=Input records that did not verify due to gender";

RUN;

PROC FREQ;
    TABLES VER_CODE /MISSING LIST;
RUN;
PROC PRINT DATA = OUT2.MPR_EVS4(OBS = 100);
WHERE DOB_SRCE = "R";
VAR DOB DOBNUM SEX;
RUN;

*****

```

## Appendix A.182

JCL/SAS Code: EVS

```
PRINT OBSERVATIONS THAT DID NOT VERIFY FOR EACH TYPE
*****;
```

```
*OBS THAT DID NOT MATCH TO TRF.DEMO;
TITLE 'OBS THAT DID NOT MATCH TO TRF.DEMO';
PROC PRINT DATA = OUT2.MPR_EVS4 (OBS = 100);
WHERE RANUNI(1100) < 0.1 AND VER_CODE='S';
RUN;
```

```
PROC FORMAT;
  VALUE $SSNMISS ' ' = 'MISSING'
          OTHER='NOT MISSING'
;
RUN;
```

```
PROC FREQ DATA = OUT2.MPR_EVS4;
WHERE VER_CODE = 'S';
TABLES SSN / LIST MISSING;
FORMAT SSN $SSNMISS.;
RUN;
```

```
*OBS THAT DID NOT VERIFY DUE TO DOB;
TITLE 'OBS THAT DID NOT VERIFY DUE TO DOB';
PROC PRINT DATA = OUT2.MPR_EVS4 (OBS = 100);
WHERE RANUNI(1100) < 0.1 AND VER_CODE='B';
FORMAT DOB mmdyy10.;
RUN;
```

```
PROC FREQ DATA = OUT2.MPR_EVS4;
WHERE VER_CODE = 'B';
TABLES DOBYEAR / LIST MISSING;
RUN;
```

```
*OBS THAT DID NOT VERIFY DUE TO GENDER;
TITLE 'OBS THAT DID NOT VERIFY DUE TO GENDER';
PROC PRINT DATA = OUT2.MPR_EVS4 (OBS = 100);
WHERE RANUNI(1100) < 0.1 AND VER_CODE='G' ;
RUN;
```

```
PROC FREQ DATA = OUT2.MPR_EVS4;
WHERE VER_CODE = 'G';
TABLES DOBYEAR / LIST MISSING;
TABLES GENDER * SEX / LIST MISSING;
RUN;
```

Appendix A.182  
JCL/SAS Code: EVS

## Appendix A.183

JCL/SAS Code: RSAEVS

```
//#3590EVS JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF (50214) DAF21
//* PROJECT
//* DIRECTOR:  MATT URATO
//* PROGRAM:  OPDR.TG.PRD.ETTW.#3590.DAF20.RSA.PRDLIB(RSAEVS)
//* DESCRIPTION: MERGE EVS RESULTS TO ALL NEW FORMAT RSA DATA
//*****
//*
//EVS  EXEC SAS9,
//  WORK='120000,60000'
//* RSAEVS - EVS OUTPUT FROM CURRENT YEAR OF RSA PROCESSING
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAEVS.SA.V1,DISP=SHR
//* NEW FORMAT RSAALL INPUTS FROM 2018 ONWARD -- ADD NEW FILES HERE
//RSA18 DD DSN=OPDR.TG.PRD.ETTW.DAF18.RSANEW.SA.V1,DISP=SHR
//RSA19 DD DSN=OPDR.TG.PRD.ETTW.#6502.RSA19NEW.SA.V1,DISP=SHR
//RSA20 DD DSN=OPDR.TG.PRD.ETTW.#6502.RSA20NEW.SA.V1,DISP=SHR
//RSA21 DD DSN=OPDR.TG.PRD.ETTW.#3590.RSA21NEW.SA.V1,DISP=SHR
//* OUTPUT VERIFIED RSA FILES ONE FOR EACH YEAR OF PROCESSING
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER18.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER19.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER20.SA.V1,
//  DISP=(MOD,CATLG,DELETE),
//  UNIT=TSILO
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER21.SA.V1,
//  DISP=(MOD,CATLG,DELETE),
//  UNIT=TSILO
//TEMP2 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP  DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *
```

```
OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN OBS=MAX;
```

```
%LET BEGYR = 2018; /*FIRST YEAR OF NEW FORMAT RSA FILE. DO NOT CHANGE*/
```

```
%LET ENDYR = 2021; /*UPDATE FOR CURRENT YEAR OF RSA PROCESSING*/
```

## Appendix A.183

JCL/SAS Code: RSAEVS

```
%MACRO DOIT;

TITLE1 "RSA EVS - PROCESSING RSA EVS DATA";
* Read in RSA EVS DATA;
DATA TEMP.EVS(DROP=RSADOB);/* DROP RSADOB IN ORDER TO KEEP ORIGINAL */

    * RENAME VARIABLES FOR MERGING AND DROP UNNECESSARY ONES;
    SET IN1.MPR_EVS4(RENAME=(DOB=RSADOB
        GENDER=RSASEX )
        DROP=FNAME INIT LNAME)
        ;
    * CREATE A DOB YEAR VARIABLE;
    DOBYR=YEAR(RSADOB);
    * CONVERT RSASEX FROM CHARACTER FLAGS TO NUMERIC FLAGS;
    IF RSASEX='M' THEN RSASEX='1';
    ELSE IF RSASEX='F' THEN RSASEX='2';
RUN;

* SORT RSA EVS DATA BY VERIFICATION KEY AND DESCENDING VER_CODE
* SO THAT THE VER_CODE='Y' WILL BE FIRST;
PROC SORT DATA=TEMP.EVS;
    BY SSN DOBYR RSASEX DESCENDING VER_CODE;
RUN;

* SORT RSA EVS DATA BY VERIFICATION KEY KEEPING ONLY THE FIRST
* UNIQUE OCCURRENCE. IF MULTIPLE OBS WITH THE KEY ARE FOUND
* AND ONE HAS A VER_CODE='Y' THEN THAT ONE WILL BE KEPT;
PROC SORT DATA=TEMP.EVS OUT=TEMP2.EVS NODUPKEYS;
    BY SSN DOBYR RSASEX;
RUN;

TITLE1 'EVS';
PROC PRINT DATA=TEMP2.EVS(OBS=50);
RUN;

%DO YEAR=&BEGYR. %TO &ENDYR.;
TITLE1 "RSA EVS - PROCESSING &YEAR.";
/*SETTING YR MACRO VAR TO LAST TWO DIGITS OF EACH YEAR*/
%LET YR=%SUBSTR(&YEAR.,3,2);

* READ RSA ALL YEARS SAS FILES;
DATA TEMP.RSAALL;
    SET RSA&YR..%IF &YEAR.=2018 %THEN RSA17_NEWFORMAT;
        %ELSE %IF &YEAR.=2019 %THEN RSA19_SORTED;
```

## Appendix A.183

JCL/SAS Code: RSAEVS

```
        %ELSE %IF &YEAR.=2020 %THEN RSA20;
        %ELSE %IF &YEAR.=2021 %THEN RSA21;
        (RENAME=(BIRTHDATE=RSADOB SEX=RSASEX));
    * CREATE A DOB YEAR VARIABLE;
    DOBYR=YEAR(RSADOB);
RUN;

PROC SORT DATA=TEMP.RSAALL OUT=TEMP2.RSAALL;
    BY SSN DOBYR RSASEX;
RUN;

TITLE1 'RSA';
PROC PRINT DATA=TEMP2.RSAALL(OBS=50);
RUN;

* MERGE RSA EVS RESULTS WITH RSA ALL YEARS FILE;
DATA OUT&YR..RSAVER&YR.;
    MERGE TEMP2.RSAALL(IN=RSA)
          TEMP2.EVS(IN=EVS)
          ;
    BY SSN DOBYR RSASEX;

* KEEP ALL RECORDS IN THE RSA ALL YEARS FILE;
IF RSA;
IF EVS THEN BOTH=1;
ELSE BOTH=0;

RUN;

PROC FREQ DATA=OUT&YR..RSAVER&YR.;
    TABLE VER_CODE;
    TABLE BOTH;
RUN;

TITLE1 "RSA &YEAR. W/ EVS";
PROC PRINT DATA=OUT&YR..RSAVER&YR.(OBS=50);
RUN;

%END; /*END RSA YEAR LOOP PROCESSING */

%MEND;
%DOIT;
ENDSAS;
```

## Appendix A.184

JCL/SAS Code: RSAEVSOE

```
//#3590EVS JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF (50214) DAF21
//*
//* PROJECT
//* DIRECTOR:  MATT URATO
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#3590.DAF21.RSA.PRDLIB(RSAEVS)
//*
//* DESCRIPTION: MERGE EVS RESULTS TO ALL RSA DATA
//*****
//*
//EVS  EXEC SAS9,
//  WORK='120000,60000'
//* UPDATE WITH CURRENT DAF RSAEVS FILE
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAEVS.RSA.V1,DISP=SHR
//* NOTE - DO NOT CHANGE THIS RSAALL INPUT.
//RSA  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF17P.RSAALLOF.SA.V1,DISP=SHR
//*
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVEROF.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO
//TEMP2 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP  DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN OBS=MAX;

* Read in RSA EVS DATA;
DATA TEMP.EVS(DROP=RSADOB);/* DROP RSADOB IN ORDER TO KEEP ORIGINAL */

  * RENAME VARIABLES FOR MERGING AND DROP UNNECESSARY ONES;
  SET IN1.MPR_EVS4(RENAME=(DOB=RSADOB
    GENDER=RSASEX )
    DROP=FNAME INIT LNAME)
  ;
  * CREATE A DOB YEAR VARIABLE;
  DOBYR=YEAR(RSADOB);
  * CONVERT RSASEX FROM CHARACTER FLAGS TO NUMERIC FLAGS;
```

## Appendix A.184

JCL/SAS Code: RSAEVSOF

```
IF RSASEX='M' THEN RSASEX='1';
ELSE IF RSASEX='F' THEN RSASEX='2';
RUN;

* SORT RSA EVS DATA BY VERIFICATION KEY AND DESCENDING VER_CODE
* SO THAT THE VER_CODE='Y' WILL BE FIRST;
PROC SORT DATA=TEMP.EVS;
  BY SSN DOBYR RSASEX DESCENDING VER_CODE;
RUN;

* SORT RSA EVS DATA BY VERIFICATION KEY KEEPING ONLY THE FIRST
* UNIQUE OCCURRENCE. IF MULTIPLE OBS WITH THE KEY ARE FOUND
* AND ONE HAS A VER_CODE='Y' THEN THAT ONE WILL BE KEPT;
PROC SORT DATA=TEMP.EVS OUT=TEMP2.EVS NODUPKEYS;
  BY SSN DOBYR RSASEX;
RUN;

TITLE1 'EVS';
PROC PRINT DATA=TEMP2.EVS(OBS=50);
RUN;

* READ RSA ALL YEARS SAS FILES;
DATA TEMP.RSAALL;
  SET RSA.RSAALL
  ;
  * CREATE A DOB YEAR VARIABLE;
  DOBYR=YEAR(RSADOB);
RUN;

PROC SORT DATA=TEMP.RSAALL OUT=TEMP2.RSAALL;
  BY SSN DOBYR RSASEX;
RUN;

TITLE1 'RSA';
PROC PRINT DATA=TEMP2.RSAALL(OBS=50);
RUN;

* MERGE RSA EVS RESULTS WITH RSA ALL YEARS FILE;
DATA OUT.RSAVERCD;
  MERGE TEMP2.RSAALL(IN=RSA)
        TEMP2.EVS(IN=EVS)
  ;
  BY SSN DOBYR RSASEX;

  * KEEP ALL RECORDS IN THE RSA ALL YEARS FILE;
```



Appendix A.184

JCL/SAS Code: RSAEVSO

```
IF RSA;
IF EVS THEN BOTH=1;
ELSE BOTH=0;

RUN;

PROC FREQ DATA=OUT.RSAVERCD;
TABLE VER_CODE;
TABLE BOTH;
RUN;

TITLE1 'RSA W/ EVS';
PROC PRINT DATA=OUT.RSAVERCD(OBS=50);
RUN;

ENDSAS;
```

## Appendix A.185

JCL/SAS Code: RSAFLAT

```
//#3590RNF JOB (12510000,T715,,SAS,,ITC9FL),SVETLANA-BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21
//* PROJECT DIRECTOR: MATT URATO
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.RSA.PRDLIB(RSAFLAT)
//* DESCRIPTION: FLATTEN RSA RECORDS TO ONE PER SSN (NEW FORMAT)
//* PROGRAMMER: SVETLANA BRONNIKOV
//*      SBRONNIKOV@MATHEMATICA-MPR.COM
//*      202-264-4221
//*****
//*
//RSA EXEC SAS9,
//  WORK='120000,60000'
//*
//* INPUT
//* DAF18 VERIFIED RSA FILE
//IN18 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER18.SA.V1,DISP=SHR
//* DAF19 VERIFIED RSA FILE
//IN19 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER19.SA.V1,DISP=SHR
//* DAF20 VERIFIED RSA FILE
//IN20 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER20.SA.V1,DISP=SHR
//* DAF21 VERIFIED RSA FILE
//IN21 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER21.SA.V1,DISP=SHR
//* TEMP FILE
//TEMP DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//TPSRT18 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//TPSRT19 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//TPSRT20 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//TPSRT21 DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//TEMPCOMB DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//TEMPOUT DD DSN=&&TEMP,
```

## Appendix A.185

JCL/SAS Code: RSAFLAT

```

//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//TEMPCONT DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//* OUTPUTS - RSA VERIFIED NEW FORMAT FILE AND CONTENTS
//OUT  DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.RSAFLAT,
//*   DISP=(OLD,CATLG,DELETE),
//*   SPACE=(CYL,(3000,3000),RLSE)
//   DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//   UNIT=TSILO
//CONT  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAFLAT.CONT,
//   DISP=(NEW,CATLG,DELETE),
//   SPACE=(CYL,(400,200),RLSE)
//SYSIN DD *

OPTIONS OBS=MAX MLOGIC MPRINT MACROGEN;

%LET BEGYR = 2018; /*FIRST YEAR OF RSA NEW FORMAT. DO NOT CHANGE */
%LET ENDYR = 2021; /*UPDATE TO CURRENT YEAR OF PROCESSING */

%MACRO DOITCOMB;

*COMBINE PRIOR AND CURRENT DAF VERIFIED FILES;
%DO YEAR=&BEGYR. %TO &ENDYR.;
%LET YR=%SUBSTR(&YEAR.,3,2);

PROC SORT DATA=IN&YR..RSAVER&YR. (WHERE=(VER_CODE='Y'))
OUT = TPSRT&YR..RSAVER&YR.
  (KEEP = SSN INDIVIDUALID AGENCYCODE APPLICATIONDATE ELIGIBILITYDATE
  EXITTYPE EXITDATE
  %IF %EVAL(&YEAR.>=2020) %THEN IPEINITIALDATE;
  %ELSE IPEAMENDEDDATE; PETSSTARTDATE VER_CODE
  VRSERVICESTARTDATE);
BY SSN APPLICATIONDATE AGENCYCODE INDIVIDUALID;
RUN;
%END;

DATA TEMPComb.RSAVERCD (KEEP = IN_VER: FLAG_SSNMISS SOURCE
  SSN INDIVIDUALID AGENCYCODE APPLICATIONDATE ELIGIBILITYDATE:
  EXITTYPE: EXITDATE: IPEDATE: PETSSTARTDATE: VER_CODE:
  VRSERVICESTARTDATE:);
MERGE
  %DO YEAR = &BEGYR. %TO &ENDYR.;
  %LET YR=%SUBSTR(&YEAR.,3,2);

```

## Appendix A.185

JCL/SAS Code: RSAFLAT

```
TPSRT&YR..RSAVER&YR. (IN=INVER&YR.
RENAME = (ELIGIBILITYDATE = ELIGIBILITYDATE_RSA&YR.
  EXITDATE      = EXITDATE_RSA&YR.
  EXITTYPE      = EXITTYPE_RSA&YR.
  %IF %EVAL(&YEAR>=2020) %THEN
  IPEINITIALDATE = IPEINITIALDATE_RSA&YR.;
  %ELSE
  IPEAMENDEDDATE = IPEAMENDEDDATE_RSA&YR.;
  PETSSTARTDATE  = PETSSTARTDATE_RSA&YR.
  VER_CODE       = VER_CODE_RSA&YR.
  VRSERVICESTARTDATE = VRSERVICESTARTDATE_RSA&YR. ))
%END;;
;
BY SSN APPLICATIONDATE AGENCYCODE INDIVIDUALID;

IN_VER18 = INVER18;
IN_VER19 = INVER19;
IN_VER20 = INVER20;
IN_VER21 = INVER21;
IF MISSING(SSN) THEN FLAG_SSNMISS = 1; ELSE FLAG_SSNMISS = 0;

*TAKE MOST RECENT VALUES;
LENGTH SOURCE $5.;
IF IN_VER21 THEN DO;
  SOURCE = 'RSA21';
  ELIGIBILITYDATE = ELIGIBILITYDATE_RSA21;
  EXITDATE      = EXITDATE_RSA21;
  EXITTYPE      = EXITTYPE_RSA21;
  IPEDATE       = IPEINITIALDATE_RSA21;
  PETSSTARTDATE = PETSSTARTDATE_RSA21;
  VER_CODE      = VER_CODE_RSA21;
  VRSERVICESTARTDATE = VRSERVICESTARTDATE_RSA21;
END;
ELSE IF IN_VER20 THEN DO;
  SOURCE = 'RSA20';
  ELIGIBILITYDATE = ELIGIBILITYDATE_RSA20;
  EXITDATE      = EXITDATE_RSA20;
  EXITTYPE      = EXITTYPE_RSA20;
  IPEDATE       = IPEINITIALDATE_RSA20;
  PETSSTARTDATE = PETSSTARTDATE_RSA20;
  VER_CODE      = VER_CODE_RSA20;
  VRSERVICESTARTDATE = VRSERVICESTARTDATE_RSA20;
END;
ELSE IF IN_VER19 THEN DO;
  SOURCE = 'RSA19';
```

## Appendix A.185

JCL/SAS Code: RSAFLAT

```
    ELIGIBILITYDATE = ELIGIBILITYDATE_RSA19;
    EXITDATE        = EXITDATE_RSA19;
    EXITTYPE        = EXITTYPE_RSA19;
    IPEDATE         = IPEAMENDEDDATE_RSA19;
    PETSSTARTDATE   = PETSSTARTDATE_RSA19;
    VER_CODE        = VER_CODE_RSA19;
    VRSERVICESTARTDATE = VRSERVICESTARTDATE_RSA19;
END;
ELSE DO;
    SOURCE = 'RSA18';
    ELIGIBILITYDATE = ELIGIBILITYDATE_RSA18;
    EXITDATE        = EXITDATE_RSA18;
    EXITTYPE        = EXITTYPE_RSA18;
    IPEDATE         = IPEAMENDEDDATE_RSA18;
    PETSSTARTDATE   = PETSSTARTDATE_RSA18;
    VER_CODE        = VER_CODE_RSA18;
    VRSERVICESTARTDATE = VRSERVICESTARTDATE_RSA18;
END;
RUN;

PROC FREQ DATA=TEMPCOMB.RSAVERCD;
TABLES VER_CODE / LIST MISSING;
TABLES FLAG_SSNMISS / LIST MISSING;
TABLES SOURCE / LIST MISSING;
TABLES IN_VER18 * IN_VER19 * IN_VER20 * IN_VER21 / LIST MISSING;
TABLES VER_CODE * IN_VER18 * IN_VER19 * IN_VER20 * IN_VER21
/ LIST MISSING;
TABLES SOURCE * IN_VER18 * IN_VER19 * IN_VER20 * IN_VER21
/ LIST MISSING;
RUN;

PROC PRINT DATA = TEMPCOMB.RSAVERCD (OBS = 25);
WHERE IN_VER18 = 1 AND IN_VER19 = 1 AND IN_VER20 = 1 AND IN_VER21=1;
VAR SOURCE EXITTYPE EXITTYPE_RSA18 EXITTYPE_RSA19 EXITTYPE_RSA20
    EXITTYPE_RSA21
    EXITDATE EXITDATE_RSA18 EXITDATE_RSA19 EXITDATE_RSA20
    EXITDATE_RSA21
    VRSERVICESTARTDATE_RSA18
    VRSERVICESTARTDATE_RSA19
    VRSERVICESTARTDATE_RSA20
    VRSERVICESTARTDATE_RSA21;
RUN;
%MEND DOITCOMB;
%DOITCOMB;
```

## Appendix A.185

JCL/SAS Code: RSAFLAT

```
*CALCULATE THE MAX NUMBER OF RECS PER SSN;
PROC SQL NOPRINT; CREATE TABLE B AS
SELECT N(SSN) AS NSSN FROM TEMPComb.RSAVERCD WHERE VER_CODE='Y'
GROUP BY SSN;
SELECT MAX(NSSN) INTO :NR FROM B; QUIT;
%LET NR = %LEFT(&NR);
%PUT "&NR";
```

```
DATA TEMP.RSAFLAT(KEEP=SSN RSAAGY: RSADOA: RSADOE: RSAEXTTYP:
      RSADOEXT: RSAIPE: RSADOPREMP: RSASVSDT:);
RETAIN RSAAGY1-RSAAGY&NR.
      RSADOA1-RSADOA&NR.
      RSADOE1-RSADOE&NR.
      RSAEXTTYP1-RSAEXTTYP&NR.
      RSADOEXT1-RSADOEXT&NR.
      RSAIPE1-RSAIPE&NR.
      RSADOPREMP1-RSADOPREMP&NR.
      RSASVSDT1-RSASVSDT&NR.
      RECCNT
      ;
```

```
SET TEMPComb.RSAVERCD;
```

```
* KEEP ONLY EVS VERIFIED RECORDS;
WHERE VER_CODE='Y';
BY SSN;
```

```
LENGTH RSADOA1-RSADOA&NR.
      RSADOE1-RSADOE&NR.
      RSADOEXT1-RSADOEXT&NR.
      RSAIPE1-RSAIPE&NR.
      RSADOPREMP1-RSADOPREMP&NR.
      RSASVSDT1-RSASVSDT&NR.
      8.;
LENGTH RSAAGY1-RSAAGY&NR. $3.;
LENGTH RSAEXTTYP1-RSAEXTTYP&NR. $1.;
```

```
ARRAY RSAAGY RSAAGY1-RSAAGY&NR.;
ARRAY RSADOA RSADOA1-RSADOA&NR.;
ARRAY RSADOE RSADOE1-RSADOE&NR.;
ARRAY RSAEXTYP RSAEXTTYP1-RSAEXTTYP&NR.;
ARRAY RSADOEXT RSADOEXT1-RSADOEXT&NR.;
ARRAY RSAIPE RSAIPE1-RSAIPE&NR.;
ARRAY RSADOPREMP RSADOPREMP1-RSADOPREMP&NR.;
```

## Appendix A.185

JCL/SAS Code: RSAFLAT

```
ARRAY RSASVSDT RSASVSDT1-RSASVSDT&NR.;
```

```
IF FIRST.SSN THEN DO I=1 TO &NR.;
```

```
  RSAAGY(I)=' ';
```

```
  RSADOA(I)=.;
```

```
  RSADOEAI(I)=.;
```

```
  RSAEXTYP(I)=.;
```

```
  RSADOEXT(I)=.;
```

```
  RSAIPE(I)=' ';
```

```
  RSADOPREMP(I) = .;
```

```
  RSASVSDT(I) = .;
```

```
  RECCNT=1;
```

```
END;
```

```
RSAAGY(RECCNT)=AGENCYCODE;
```

```
RSADOA(RECCNT)=APPLICATIONDATE;
```

```
RSADOEAI(RECCNT)=ELIGIBILITYDATE;
```

```
RSAEXTYP(RECCNT)=EXITTYPE;
```

```
RSADOEXT(RECCNT)=EXITDATE;
```

```
RSAIPE(RECCNT)=IPEDATE;
```

```
RSADOPREMP(RECCNT)=PETSSTARTDATE;
```

```
RSASVSDT(RECCNT)=VRSERVICESTARTDATE;
```

```
RECCNT=RECCNT+1;
```

```
IF LAST.SSN THEN OUTPUT;
```

```
RUN;
```

```
%MACRO DOIT;
```

```
DATA OUT.RSAFLAT;
```

```
SET TEMP.RSAFLAT;
```

```
LABEL SSN = "SOCIAL SECURITY NUMBER"
```

```
  %DO I = 1 %TO &NR.;
```

```
  RSAAGY&I = "RSA AGENCY &I"
```

```
  RSADOA&I = "RSA DATE OF APPLICATION &I"
```

```
  RSADOEAI&I = "RSA DATE OF ELIGIBILITY DETERMINATION &I"
```

```
  RSAEXTTYP&I = "RSA EXIT TYPE &I"
```

```
  RSADOEXT&I = "RSA DATE OF EXIT &I"
```

```
  RSAIPE&I = "RSA DATE OF INDIVIDUALIZED PLAN FOR EMPLOYMENT &I"
```

```
  RSADOPREMP&I = "RSA DATE OF PRE-EMPLOYMENT TRANSITION SERVICES &I"
```

```
  RSASVSDT&I. = "RSA START DATE OF INITIAL VR SERVICE &I"
```

```
  %END;
```

```
;
```

Appendix A.185

JCL/SAS Code: RSAFLAT

```
RUN;  
%MEND DOIT;  
%DOIT;  
PROC PRINT DATA=TEMPCOMB.RSAVERCD(OBS=10);  
  WHERE VER_CODE='Y';  
RUN;
```

```
PROC MEANS DATA=OUT.RSAFLAT MIN MAX MEAN;  
VAR RSADOA1-RSADOA&NR.  
  RSADOE1-RSADOE&NR.  
  RSADOEXT1-RSADOEXT&NR.  
  RSADOPREMP1-RSADOPREMP&NR.  
  RSAIPE1-RSAIPE&NR.  
  RSASVSDT1-RSASVSDT&NR.  
;  
RUN;
```

```
PROC PRINT DATA=OUT.RSAFLAT(OBS=10);  
RUN;
```

```
PROC CONTENTS DATA=OUT.RSAFLAT OUT=CONT.RSAFLAT;  
RUN;
```

```
ENDSAS;
```



## Appendix A.186

JCL/SAS Code: RSAFLAOF

```

//#3590ROF JOB (12510000,T715,,SAS,,ITC9FL),SVETLANA-BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21
//*
//* PROJECT DIRECTOF: MATT URATO
//*
//* PROGRAM:  OPDR.TG.PRD.ETTW.#3590.DAF21.RSA.PRDLIB(RSAFLAOF)
//*
//* DESCRIPTION: FLATTEN RSA RECORDS TO ONE PER SSN (OLD FORMAT)
//*****
//*
//RSA  EXEC SAS9,
//  WORK='120000,60000'
//*
//* INPUT
//*
//IN  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVEROF.SA.V1,DISP=SHR
//*
//TEMP DD DSN=&&TEMP,
//  DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(1000,100),RLSE),VOL=(,,10)
//OUT  DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.RSAFLAOF,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//CONT DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAFLAOF.CONT,
//  DISP=(NEW,CATLG,DELETE),
//  SPACE=(CYL,(400,200),RLSE)
//SYSIN DD *

```

OPTIONS OBS=MAX;

```

*CALCULATE THE MAX NUMBER OF RECS PER SSN;
PROC SQL NOPRINT; CREATE TABLE B AS
SELECT N(SSN) AS NSSN FROM IN.RSAVERCD WHERE VER_CODE='Y'
GROUP BY SSN;
SELECT MAX(NSSN) INTO :NR FROM B; QUIT;
%LET NR = %LEFT(&NR);
%PUT "&NR";

```

```

DATA TEMP.RSAFLAT(KEEP=SSN RSAAGY: RSADOA: RSADOE: RSACLTY:
  RSADOC: RSASVS: RSAIPE:);
  RETAIN RSAAGY1-RSAAGY&NR.

```

Appendix A.186

JCL/SAS Code: RSAFLAOF

```

RSADOA1-RSADOA&NR.
RSADOE1-RSADOE&NR.
RSADOC1-RSADOC&NR.
RSASVS1-RSASVS&NR.
RSACLTY1-RSACLTY&NR.
RSAIPE1 - RSAIPE&NR.
RECCNT
;

```

```

SET IN.RSAVERCD;

```

```

* KEEP ONLY EVS VERIFIED RECORDS;
WHERE VER_CODE='Y';
BY SSN;

```

```

LENGTH RSADOA1-RSADOA&NR.
  RSADOE1-RSADOE&NR.
  RSADOC1-RSADOC&NR.
  RSASVS1-RSASVS&NR.
  RSAIPE1 - RSAIPE&NR.
  8.;

```

```

LENGTH RSAAGY1-RSAAGY&NR. $3.;
LENGTH RSACLTY1-RSACLTY&NR. $1.;

```

```

ARRAY RSAAGY RSAAGY1-RSAAGY&NR.;
ARRAY RSADOA RSADOA1-RSADOA&NR.;
ARRAY RSADOE RSADOE1-RSADOE&NR.;
ARRAY RSADOC RSADOC1-RSADOC&NR.;
ARRAY RSASVS RSASVS1-RSASVS&NR.;
ARRAY RSACLTY RSACLTY1-RSACLTY&NR.;
ARRAY RSAIPE RSAIPE1-RSAIPE&NR.;

```

```

IF FIRST.SSN THEN DO I=1 TO &NR.;
  RSAAGY(I)= ' ';
  RSADOA(I)=.;
  RSADOE(I)=.;
  RSADOC(I)=.;
  RSASVS(I)=.;
  RSACLTY(I)= ' ';
  RSAIPE(I) = .;
  RECCNT=1;
END;

```

```

RSAAGY(RECCNT)=AGENCY;
RSADOA(RECCNT)=APPDATE;

```

Appendix A.186

JCL/SAS Code: RSAFLAOF

```

RSADOE1(RECCNT)=RSADOE;
RSADOC1(RECCNT)=CLOSEDATE;
RSASVS1(RECCNT)=SVCS_PROV;
RSACL1(RECCNT)=CLTYPE;
RSAIPE1(RECCNT)=DOIPE;

RECCNT=RECCNT+1;

IF LAST.SSN THEN OUTPUT;

RUN;

%MACRO DOIT;
DATA OUT.RSAFLAT;
SET TEMP.RSAFLAT(DROP = RSADOE);

LABEL SSN = "SOCIAL SECURITY NUMBER"
      %DO I = 1 %TO &NR.;
      RSAAGY&I = "RSA AGENCY &I"
      RSADOA&I = "RSA DATE OF APPLICATION &I"
      RSADOE&I = "RSA DATE OF ELIGIBILITY DETERMINATION &I"
      RSACL1&I = "RSA CLOSURE TYPE &I"
      RSADOC&I = "RSA DATE OF CLOSURE &I"
      RSASVS&I = "RSA SERVICE USE FLAG &I"
      RSAIPE&I = "RSA DATE OF INDIVIDUALIZED PLAN FOR EMPLOYMENT &I"
      %END;
;
RUN;
%MEND DOIT;
%DOIT;
PROC PRINT DATA=IN.RSAVERCD(OBS=10);
  WHERE VER_CODE='Y';
RUN;

PROC MEANS DATA=OUT.RSAFLAT MIN MAX MEAN;
VAR RSADOA1-RSADOA&NR.
     RSADOE1-RSADOE&NR.
     RSADOC1-RSADOC&NR.
     RSASVS1-RSASVS&NR.
     RSAIPE1-RSAIPE&NR.
;
RUN;

```

Appendix A.186

JCL/SAS Code: RSAFLAOF

```
PROC PRINT DATA=OUT.RSAFLAT(OBS=10);  
RUN;
```

```
PROC CONTENTS DATA=OUT.RSAFLAT OUT=CONT.RSAFLAT;  
RUN;
```

```
ENDSAS;
```

## Appendix A.187

JCL/SAS Code: VERRSA

```
//#3590VER JOB (12510000,T715,,SAS,,ITC9FL),SVETLANA-BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: BUILD DAF20 (50214)
//* PROJECT DIRECTOR: MATT URATO
//* PROGRAM:  OPDR.TG.PRD.ETTW.#3590.DAF21.RSA.PRDLIB(VERRSA)
//* DESCRIPTION: CREATE LIST OF VERIFIED RSA SSNS
//*****
//*
//EVS  EXEC SAS9,
//  WORK='120000,60000'
//* NOTE -- ALL VERIFIED RSA FILES (RSAVER*) SHOULD BE LISTED HERE
//IN21 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER21.SA.V1,DISP=SHR
//IN20 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER20.SA.V1,DISP=SHR
//IN19 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER19.SA.V1,DISP=SHR
//IN18 DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVER18.SA.V1,DISP=SHR
//INOF DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAVEROF.SA.V1,DISP=SHR
//*
//OUT DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.VERRSA.SA.V1,
//  DISP=(MOD,CATLG,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE)
//SYSIN DD *
```

```
OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN /*OBS=0*/;
```

```
DATA OUT.VERRSA(KEEP=SSN RSADOB DOBYR SEX RSASEX);
  SET INOF.RSAVERCD (KEEP=SSN RSADOB DOBYR SEX RSASEX VER_CODE)
  IN18.RSAVER18 (KEEP=SSN RSADOB DOBYR SEX RSASEX VER_CODE)
  IN19.RSAVER19 (KEEP=SSN RSADOB DOBYR SEX RSASEX VER_CODE)
  IN20.RSAVER20 (KEEP=SSN RSADOB DOBYR SEX RSASEX VER_CODE)
  IN21.RSAVER21 (KEEP=SSN RSADOB DOBYR SEX RSASEX VER_CODE);
  WHERE VER_CODE='Y';
  DOBYR=YEAR(RSADOB);
RUN;
```

```
PROC FREQ DATA=OUT.VERRSA;
  TABLE SEX*RSASEX / LIST MISSING;
RUN;
```

```
PROC SORT DATA=OUT.VERRSA NODUPKEYS;
  BY SSN DOBYR RSASEX;
RUN;
```

```
ENDSAS;
```

Appendix A.187

JCL/SAS Code: VERRSA

## Appendix A.188

JCL/SAS Code: DMG

```

//#3590DEM JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: RSA-DAF MATCHED EXTRACT
//* PROJECT DIRECTOR: MATT URATO
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(DMG)
//* DESCRIPTION: CREATES THE RSA MINI-DAF DMG
//* PROGRAMMER: SVETLANA BRONNIKOV
//*      SBRONNIKOV@MATHEMATICA-MPR.COM
//*      202-264-4221
//*****
//DMG  EXEC SAS9,
//  WORK='120000,60000'
//*
//DMG  DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.DMG,DISP=SHR
//RSA  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.VERRSA.SA.V1,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.DMG,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#3590.RSADAF.SSN21.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

```

```

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

```

```

PROC SORT DATA=RSA.VERRSA OUT=VERRSA NODUPKEYS;
  BY SSN;
RUN;

```

```

* MERGE RSA VERIFIED FILE WITH DAF DMG FILE KEEP RSA VERIFIED OBS;
* DROP BENE NAME VARIABLES;

```

```

DATA OUT.DMG(DROP=CAN HUN
  BENE_FRSTNAME_MBR BENE_LASTNAME_MBR BMI
  BENE_FRSTNAME_NUMI BENE_LASTNAME_NUMI
  BENE_SUFNAME_NUMI);
MERGE DMG.DMG (IN=DMG)
  VERRSA(IN=RSA KEEP=SSN)
  ;
  BY SSN;
  IF RSA AND DMG;
RUN;

```

Appendix A.188

JCL/SAS Code: DMG

```
DATA OUT2.SSN;  
  SET OUT.DMG (KEEP = SSN);  
RUN;
```

```
PROC CONTENTS DATA=OUT.DMG;  
RUN;
```



## Appendix A.189

JCL/SAS Code: JCLANN

```
//#3590ANN JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF 50214
//* PROJECT DIRECTOR: MATT URATO
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(JCLANN)
//* DESCRIPTION: CREATES RSA MINI-DAF ANNUAL EXTRACTS
//*****
//*
//ANNPROC PROC
//ANN EXEC SAS9,
//  WORK='120000,60000',
//  PARM='SYSPARM="&YR"'
//*
//ANN DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y&YR.E,DISP=SHR
//LIBRARY DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.LAUS.FMTLIB,DISP=SHR
//LIBRAR2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.SAIBE.FMTLIB,DISP=SHR
//RSA DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.VERRSA.SA.V1,DISP=SHR
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.Y&YR.,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.L&YR.,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(ANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
//*
//RUN21 EXEC ANNPROC,YR=2021
//RUN20 EXEC ANNPROC,YR=2020
//RUN19 EXEC ANNPROC,YR=2019
//RUN18 EXEC ANNPROC,YR=2018
//RUN17 EXEC ANNPROC,YR=2017
//RUN16 EXEC ANNPROC,YR=2016
//RUN15 EXEC ANNPROC,YR=2015
//RUN14 EXEC ANNPROC,YR=2014
//RUN13 EXEC ANNPROC,YR=2013
//RUN12 EXEC ANNPROC,YR=2012
//RUN11 EXEC ANNPROC,YR=2011
//RUN10 EXEC ANNPROC,YR=2010
//RUN09 EXEC ANNPROC,YR=2009
//RUN08 EXEC ANNPROC,YR=2008
```

Appendix A.189

JCL/SAS Code: JCLANN

```
//RUN07 EXEC ANNPROC,YR=2007
//RUN06 EXEC ANNPROC,YR=2006
//RUN05 EXEC ANNPROC,YR=2005
//RUN04 EXEC ANNPROC,YR=2004
//RUN03 EXEC ANNPROC,YR=2003
//RUN02 EXEC ANNPROC,YR=2002
//RUN01 EXEC ANNPROC,YR=2001
//RUN00 EXEC ANNPROC,YR=2000
//RUN99 EXEC ANNPROC,YR=1999
//RUN98 EXEC ANNPROC,YR=1998
//RUN97 EXEC ANNPROC,YR=1997
//RUN96 EXEC ANNPROC,YR=1996
//RUN95 EXEC ANNPROC,YR=1995
//RUN94 EXEC ANNPROC,YR=1994
```

## Appendix A.190

JCL/SAS Code: ANN

```
* EXECUTED BY OPDR.TG.PRD.ETTW.#3590.DAF21.RSADEID.PRDLIB(JCLANN);
* THIS PROGRAM EXTRACTS THE DAF ANN DATA FOR VERIFIED RSA RECORDS;
```

```
OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX
FMTSEARCH = (LIBRARY LIBRAR2);
```

```
PROC SORT DATA=RSA.VERRSA OUT=VERRSA NODUPKEYS;
  BY SSN;
RUN;
```

```
%MACRO DOIT;
DATA OUT1.Y&SYSPARM.(DROP = LAUS: SAIPE:)
  OUT2.L&SYSPARM. (KEEP = SSN LAUS: SAIPE:)
  ;
MERGE ANN.Y&SYSPARM. (IN=ANN)
  VERRSA(IN=RSA KEEP=SSN)
  ;
BY SSN;
IF RSA AND ANN;
```

```
%LET YR=%SUBSTR(&SYSPARM.,3,2);
```

```
ARRAY FIPS_BEST (*) FIPS_BEST&YR.01-FIPS_BEST&YR.12;
ARRAY LAUS_UNEMP (*) LAUS_UNEMP&YR.01-LAUS_UNEMP&YR.12;
ARRAY SAIPE_POV (*) SAIPE_POV&YR.01 - SAIPE_POV&YR.12;
ARRAY SAIPE_INC (*) SAIPE_INC&YR.01 - SAIPE_INC&YR.12;
```

```
DO MO=1 TO 9;
IF LENGTH(FIPS_BEST(MO))=5 AND FIPS_BEST(MO) NE " THEN DO;
LAUS_UNEMP(MO)=PUT(CATS(FIPS_BEST(MO),"&YR.",'0',MO),$LAUS&YR.F.);
  %IF &SYSPARM. NE 1994 AND &SYSPARM. NE 1996 %THEN %DO;
SAIPE_POV(MO)=PUT(CATS(FIPS_BEST(MO),"&YR."),$SAIPEPOV&YR.F.);
SAIPE_INC(MO)=PUT(CATS(FIPS_BEST(MO),"&YR."),$SAIPEINC&YR.F.);
  %END;
END;
END;
```

```
DO MO=10 TO 12;
IF LENGTH(FIPS_BEST(MO))=5 AND FIPS_BEST(MO) NE " THEN DO;
LAUS_UNEMP(MO)=PUT(CATS(FIPS_BEST(MO),"&YR.",MO),$LAUS&YR.F.);
  %IF &SYSPARM. NE 1994 AND &SYSPARM. NE 1996 %THEN %DO;
SAIPE_POV(MO)=PUT(CATS(FIPS_BEST(MO),"&YR."),$SAIPEPOV&YR.F.);
SAIPE_INC(MO)=PUT(CATS(FIPS_BEST(MO),"&YR."),$SAIPEINC&YR.F.);
  %END;
```

Appendix A.190

JCL/SAS Code: ANN

```
END;  
END;  
RUN;
```

```
PROC CONTENTS DATA=OUT1.Y&SYSPARM.;  
RUN;  
PROC CONTENTS DATA=OUT2.L&SYSPARM.;  
RUN;
```

```
PROC PRINT DATA = OUT2.L&SYSPARM.(OBS = 25);  
RUN;  
%MEND DOIT;  
%DOIT;
```

## Appendix A.191

JCL/SAS Code: JCLTKT

```
//#3590TKT JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF 50214
//* PROJECT DIRECTOR: MATT URATO
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(JCLTKT)
//* DESCRIPTION: CREATES RSA MINI-DAF TICKET EXTRACTS
//* PROGRAMMER: SVETLANA BRONNIKOV
//*      SBRONNIKOV@MATHEMATICA-MPR.COM
//*      202-264-4221
//*****
//TKTBSE EXEC SAS9,
//  WORK='120000,60000'
//*
//TKTBS DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKETBS,DISP=SHR
//RSA DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.VERRSA.SA.V1,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.TKTBS,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(TKTBASE),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* DAF TKT ANNUALS EXTRACT
//*****
//TKTAPROC PROC
//TKTA EXEC SAS9,
//  WORK='120000,60000',
//  PARM='SYSPARM="20&YR"'
//*
//TKTANN DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET&YR.,DISP=SHR
//RSA DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.VERRSA.SA.V1,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.TKT&YR.,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(TKTANN),
//  DISP=(SHR,PASS,KEEP)
//  PEND
//*
```

Appendix A.191

JCL/SAS Code: JCLTKT

```
//RUN21 EXEC TKTAPROC,YR=21
//RUN20 EXEC TKTAPROC,YR=20
//RUN19 EXEC TKTAPROC,YR=19
//RUN18 EXEC TKTAPROC,YR=18
//RUN17 EXEC TKTAPROC,YR=17
//RUN16 EXEC TKTAPROC,YR=16
//RUN15 EXEC TKTAPROC,YR=15
//RUN14 EXEC TKTAPROC,YR=14
//RUN13 EXEC TKTAPROC,YR=13
//RUN12 EXEC TKTAPROC,YR=12
//RUN11 EXEC TKTAPROC,YR=11
//RUN10 EXEC TKTAPROC,YR=10
//RUN09 EXEC TKTAPROC,YR=09
//RUN08 EXEC TKTAPROC,YR=08
//RUN07 EXEC TKTAPROC,YR=07
//RUN06 EXEC TKTAPROC,YR=06
//RUN05 EXEC TKTAPROC,YR=05
//RUN04 EXEC TKTAPROC,YR=04
//RUN03 EXEC TKTAPROC,YR=03
//RUN02 EXEC TKTAPROC,YR=02
```

Appendix A.192

JCL/SAS Code: TKTBASE

```
* EXECUTED BY OPDR.TG.PRD.ETW.#3590.DAF21.RSADEID.PRDLIB(JCLTKT);  
* THIS PROGRAM EXTRACTS THE DAF TICKET BASE DATA FOR VERIFIED RSA  
* RECORDS;
```

```
OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;
```

```
PROC SORT DATA=RSA.VERRSA OUT=VERRSA NODUPKEYS;  
  BY SSN;  
RUN;
```

```
DATA OUT.TKTBASE(DROP=DUNS:);  
  MERGE TKTBS.TICKETBS(IN=TKT)  
        VERRSA(IN=RSA KEEP=SSN)  
        ;  
  BY SSN;  
  IF RSA AND TKT;  
RUN;
```

```
PROC CONTENTS DATA=OUT.TKTBASE;  
RUN;
```

Appendix A.193

JCL/SAS Code: TKTANN

```
* EXECUTED BY OPDR.TG.PRD.ETW.#3590.DAF21.RSADEID.PRDLIB(JCLTKT);  
* THIS PROGRAM EXTRACTS THE DAF TICKET ANNUAL DATA FOR VERIFIED RSA  
* RECORDS;
```

```
* SET MACRO VARIABLE TO THE SYSTEM PARAMETER;  
%LET YR=%SUBSTR(&SYSPARM.,3,2);
```

```
OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;
```

```
PROC SORT DATA=RSA.VERRSA OUT=VERRSA NODUPKEYS;  
  BY SSN;  
RUN;
```

```
DATA OUT.TKT&SYSPARM. ;  
  MERGE TKTANN.TICKET&YR.(IN=TKT)  
        VERRSA(IN=RSA KEEP=SSN)  
        ;  
  BY SSN;  
  IF RSA AND TKT;  
RUN;
```

```
PROC CONTENTS DATA=OUT.TKT&SYSPARM. ;  
RUN;
```



## Appendix A.194

JCL/SAS Code: ENPAY

```

//#3590ENP JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: RSA-DAF MATCHED EXTRACT FOR DOE
//* PROJECT DIRECTOR: MATT URATO
//* PROGRAM:  OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(ENPAY)
//* DESCRIPTION: CREATES ALL THE RSA-DAF MATCHED EXTRACTS
//*****
//*
//*****
//* EN EXTRACTS
//*****
//PAYH EXEC SAS9,
//  WORK='120000,60000'
//*
//PAYH DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.ENPAY,DISP=SHR
//PAYV DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.ENPAYVER,DISP=SHR
//RSA DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.VERRSA.SA.V1,DISP=SHR
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.ENPAY,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.ENPAV,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

PROC SORT DATA=RSA.VERRSA OUT=VERRSA NODUPKEYS;
  BY SSN;
RUN;

* MERGE RSA VERIFIED FILE WITH EN FILES KEEP RSA VERIFIED OBS;
DATA OUT1.ENPAY;
  MERGE PAYH.ENPAY_FINAL(IN=ENPAY)
    VERRSA(IN=RSA KEEP=SSN)
  ;
  BY SSN;
  IF RSA AND ENPAY;
RUN;

DATA OUT2.ENPAYVER(DROP=BENE_NAME_PMNTS);

```

Appendix A.194

JCL/SAS Code: ENPAY

```
MERGE PAYV.ENPAYVER_FINAL(IN=ENPAY)
      VERRSA(IN=RSA KEEP=SSN)
      ;
      BY SSN;
      IF RSA AND ENPAY;
RUN;
```

```
PROC CONTENTS DATA=OUT1.ENPAY;
RUN;
PROC CONTENTS DATA=OUT2.ENPAYVER;
RUN;
```

## Appendix A.195

JCL/SAS Code: VRRMS

```

//#3590VRR JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: RSA-DAF MATCHED EXTRACT FOR DOE
//* PROJECT DIRECTOR: MATT URATO
//* PROGRAM:  OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(VRRMS)
//* DESCRIPTION: CREATES ALL THE RSA-DAF MATCHED EXTRACTS
//*****
//*
//*****
//* DAF VRRMS EXTRACT
//*****
//VRMS  EXEC SAS9,
//  WORK='120000,60000'
//*
//VRMS  DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.VRRMS,DISP=SHR
//VRMSH DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.VRRMS.HIST,DISP=SHR
//RSA  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.VERRSA.SA.V1,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.VRRMS,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTH DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.VRRMS.HIST,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

```

```

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

```

```

PROC SORT DATA=RSA.VERRSA OUT=VERRSA NODUPKEYS;

```

```

  BY SSN;

```

```

RUN;

```

```

* MERGE RSA VERIFIED FILE WITH DAF VRRMS FILE KEEP RSA VERIFIED OBS;

```

```

DATA OUT.VRRMS;

```

```

  MERGE VRMS.VRRMS(IN=VRRMS DROP=CLNT_FNM CLNT_LNM CLNT_MNM)

```

```

    VERRSA(IN=RSA KEEP=SSN)

```

```

  ;

```

```

  BY SSN;

```

```

  IF RSA AND VRRMS;

```

```

RUN;

```

```

PROC CONTENTS DATA=OUT.VRRMS;

```

Appendix A.195

JCL/SAS Code: VRRMS

```
* MERGE RSA VERIFIED FILE WITH DAF HISTVRRMS FILE KEEP RSA VERIFIED OBS;  
DATA OUTH.VRRMS;  
  MERGE VRMSH.VRRMS(IN=VRRMS DROP=C_1ST_NAME C_LAST_NAME C_XREF_SSN)  
    VERRSA(IN=RSA KEEP=SSN)  
  ;  
  BY SSN;  
  IF RSA AND VRRMS;  
RUN;  
  
PROC CONTENTS DATA=OUTH.VRRMS;  
  
RUN;
```

## Appendix A.196

JCL/SAS Code: RSA

```
//#3590RSA JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: RSA-DAF MATCHED EXTRACT FOR DOE
//* PROJECT NUMBER: 50214
//* PROJECT DIRECTOR: MATT URATO
//* PROGRAM:  OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(RSA)
//* DESCRIPTION: CREATES ALL THE RSA-DAF MATCHED EXTRACTS
//*****
//*
//*****
//* RSA EXTRACTF
//*****
//RSA1998 EXEC SAS9,
//  WORK='120000,60000'
//*
//DEMO DD DSN=OPDR.TG.PRD.ETTW.#3590.RSADAF.SSN21.SA.V1,DISP=SHR
//RSA DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.VERRSA.SA.V1,DISP=SHR
//RSA98 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA98.SA.V1,DISP=SHR
//OUT98 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA98,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//RSA99 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA99.SA.V1,DISP=SHR
//OUT99 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA99,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//RSA00 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA00.SA.V1,DISP=SHR
//OUT00 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA00,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//RSA01 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA01.SA.V1,DISP=SHR
//OUT01 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA01,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//RSA02 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA02.SA.V1,DISP=SHR
//OUT02 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA02,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//RSA03 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA03.SA.V1,DISP=SHR
//OUT03 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA03,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//RSA04 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA04.SA.V1,DISP=SHR
//OUT04 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA04,
```

## Appendix A.196

JCL/SAS Code: RSA

```
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//RSA05 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA05.SA.V1,DISP=SHR
//OUT05 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA05,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//RSA06 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA06.SA.V1,DISP=SHR
//OUT06 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA06,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//RSA07 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA07.SA.V1,DISP=SHR
//OUT07 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA07,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//RSA08 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA08.SA.V1,DISP=SHR
//OUT08 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA08,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//RSA09 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA09.SA.V1,DISP=SHR
//OUT09 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA09,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//RSA10 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA10.SA.V1,DISP=SHR
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA10,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//RSA11 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA11.SA.V1,DISP=SHR
//OUT11 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA11,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//RSA12 DD DSN=OPDR.TG.PRD.ETTW.#3592.RSA12.SA.V1,DISP=SHR
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA12,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//RSA13 DD DSN=OPDR.TG.PRD.ETTW.#3590.RSA13.SA.V1,DISP=SHR
//OUT13 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA13,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//RSA14 DD DSN=OPDR.TG.PRD.ETTW.#6266.RSA14.SA.V1,DISP=SHR
//OUT14 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA14,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//RSA15 DD DSN=OPDR.TG.PRD.ETTW.#6266.RSA15.SA.V1,DISP=SHR
//OUT15 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA15,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
```

## Appendix A.196

JCL/SAS Code: RSA

```
//      SPACE=(CYL,(3000,3000),RLSE)
//RSA16 DD DSN=OPDR.TG.PRD.ETTW.#6266.RSA16.SA.V1,DISP=SHR
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA16,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* OLD FORMAT RSA
//RSA17 DD DSN=OPDR.TG.PRD.ETTW.#6502.RSA17OLD.SA.V1,DISP=SHR
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA17,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* NEW FORMAT RSA
//RSA18 DD DSN=OPDR.TG.PRD.ETTW.DAF18.RSANEW.SA.V1,DISP=SHR
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA18,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* DAF19
//RSA19 DD DSN=OPDR.TG.PRD.ETTW.#6502.RSA19NEW.SA.V1,DISP=SHR
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA19,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* DAF20
//RSA20 DD DSN=OPDR.TG.PRD.ETTW.#6502.RSA20NEW.SA.V1,DISP=SHR
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA20,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//* DAF21
//RSA21 DD DSN=OPDR.TG.PRD.ETTW.#3590.RSA21NEW.SA.V1,DISP=SHR
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA21,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;
```

```
%MACRO DOIT;
%MACRO RSA(YEAR=, N=, NAME=);
```

```
%LET YR=%SUBSTR(&YEAR.,3,2);
%IF &YEAR=2018 %THEN %LET YR2=17;
%ELSE %LET YR2=&YR;
```

## Appendix A.196

JCL/SAS Code: RSA

```
%IF &YR.>=13 & &YR.<90 %THEN %DO;
DATA TEMP1.RSA&YR.;
  SET RSA&YR.&N..RSA&YR2.&NAME.
  %IF &YR.>= 18 /*AND %LENGTH(&N.) NE 0*/ %THEN %DO;
  (RENAME=(SEX=RSASEX STATE = RSASTATE
    BIRTHDATE=RSADOB))
  %END;
  ;
  DOBYR=YEAR(RSADOB);
RUN;
%END;
%ELSE %DO;
DATA TEMP1.RSA&YR.;
  SET RSA&YR..RSA&YR.;
  DOBYR=YEAR(RSADOB);
  RENAME COSSN=SSN;
RUN;
%END;

PROC SORT DATA=TEMP1.RSA&YR.;
  BY SSN DOBYR RSASEX;
RUN;

DATA TEMP2.RSA&YR.;
  MERGE TEMP1.RSA&YR.(IN=RSALD)
    RSA.VERRSA(IN=RSA KEEP=SSN DOBYR RSASEX)
  ;
  BY SSN DOBYR RSASEX;
  IF RSA AND RSALD;

  *** CREATE GENDER VARIABLE ***;

  IF RSASEX='1' THEN GENDER='M';
  ELSE IF RSASEX='2' THEN GENDER='F';
RUN;

DATA OUT&YR.&N..RSA&YR.;
  MERGE TEMP2.RSA&YR. (IN=RSA)
    DEMO.SSN(IN=DEMO)
  ;
  BY SSN;
  IF RSA;

  *** CREATE FLAG IN_DAF ***;
  IF RSA AND DEMO THEN IN_DAF = 1;
```



## Appendix A.196

JCL/SAS Code: RSA

```
ELSE IF RSA AND NOT DEMO THEN IN_DAF = 0;

*** DAA MASKING FOR OLD RSA FORMAT ***;
%IF &YEAR < 2018 %THEN %DO;
  IF INSTTYPE IN ('07', '08', '14', '17')
  THEN INSTTYPE = '99';
%END;

*** IF RSA17 NEW FORMAT THEN APPLY LABELS ***;
%IF &YEAR>1997 & &YEAR<2002 %THEN %DO;
  LABEL RSADOB="RSA DATE OF BIRTH";
%END;
%ELSE %DO;
  LABEL INSTTYPE="TYPE OF INSTITUTION";
%END;

%IF &YEAR > 2013 & &YEAR<2018 %THEN %DO;
  LABEL
  CLED="EDUCATION AT CLOSURE"
  CLNJOBS="NUMBER OF JOBS AT CLOSURE"
  CLOCCUP="OCCUPATION AT CLOSURE"
  CLSHRSWK="HOURS WORKED AT RSA CLOSURE"
  CLSTDTPRIM="START DATE OF EMPLOYMENT IN PRIMARY OCCUPATION AT CLOSURE"
  CLSTUDDIS="STUDENT WITH DISABILITY IN SECONDARY EDUCATION AT CLOSURE"
  CLSWKEARN="WEEKLY EARNINGS AT RSA CLOSURE"
  CLSWRKSTAT="WORK STATUS AT RSA CLOSURE"
  ;
%END;

%IF &YEAR >=2018 %THEN %DO;
%INCLUDE "OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(RSALABEL)";
%END;

RUN;

PROC CONTENTS DATA=OUT&YR.&N..RSA&YR.;
  TITLE "CONTENTS RSA &YR.";
RUN;

PROC FREQ DATA=OUT&YR.&N..RSA&YR.;
  TABLES IN_DAF/MISSING;
  TITLE "CHECK THE FLAG VARIABLE IN_DAF &YR.";
RUN;
```

Appendix A.196

JCL/SAS Code: RSA

```
PROC FREQ DATA=OUT&YR.&N..RSA&YR.;
  TABLES RSASEX GENDER RSASEX*GENDER / LIST MISSING;
  TITLE "CHECK RSASEX AND GENDER VARIABLES &YR.";
RUN;
```

```
PROC FORMAT;
VALUE $INSTDAA '99' = '99'
              " " = "
              OTHER = 'NOT DAA';
RUN;
```

```
%IF &YEAR < 2018 %THEN %DO;
PROC FREQ DATA = OUT&YR.&N..RSA&YR.;
  TABLES INSTTYPE / LIST MISSING;
  FORMAT INSTTYPE $INSTDAA.;
  TITLE "CHECK DAA MASKING VARIABLES &YR.";
RUN;
%END;
```

```
PROC DATASETS LIBRARY=TEMP1;
  DELETE RSA&YR.;
RUN;
```

```
%MEND RSA;
```

```
%DO YEAR=1998 %TO 2017;
  %RSA(YEAR=&YEAR.);
%END;
  *FOR THE NEW FORMAT RSA;
  %RSA(YEAR=2018,NAME=_NEWFORMAT);
  %RSA(YEAR=2019,NAME=_SORTED);
  %RSA(YEAR=2020);
  %RSA(YEAR=2021);
%MEND DOIT;
%DOIT;
ENDSAS;
```

Appendix A.197

JCL/SAS Code: RSALABEL

label

SubmissionDate="RSA DATE REPORT SUBMITTED"

AgencyCode="RSA AGENCY CODE"

IndividualID="RSA UNIQUE IDENTIFIER"

ApplicationDate="RSA DATE OF VR APPLICATION"

RSADOB="RSA DATE OF BIRTH"

RSASex="RSA SEX"

AmerIndian="RSA RACE: AMERICAN INDIAN OR ALASKA NATIVE"

Asian="RSA RACE: ASIAN"

Black="RSA RACE: BLACK OR AFRICAN AMERICAN"

Hawaiian="RSA RACE: NATIVE HAWAIIAN OR OTHER PACIFIC ISLANDER"

White="RSA RACE: WHITE"

Hispanic="RSA ETHNICITY - HISPANIC OR LATINO"

Veteran="RSA VETERAN"

LivingArrangementCD="RSA LIVING ARRANGEMENT"

RSASate="RSA STATE POSTAL CODE OF RESIDENCE"

CountyFIPSCode="RSA COUNTY FIPS CODE"

Referral="RSA SOURCE OF REFERRAL"

Student="RSA STUDENT WITH A DISABILITY"

appSSDI="RSA SUPPORT: SOCIAL SECURITY DISABILITY INSURANCE (SSDI) AT APPLICATION"

appSSI="RSA SUPPORT: SUPPLEMENTAL SECURITY INCOME (SSI) FOR THE AGED, BLIND, OR DISABLED AT APPLICATION"

appTANF="RSA SUPPORT: TEMPORARY ASSISTANCE FOR NEEDY FAMILIES (TANF) AT APPLICATION"

appGenAssistance="RSA SUPPORT: GENERAL ASSISTANCE (STATE OR LOCAL GOVERNMENT) AT APPLICATION"

appVeteransDisability="RSA SUPPORT: VETERANS' DISABILITY BENEFITS AT APPLICATION"

appWorkersComp="RSA SUPPORT: WORKERS' COMPENSATION AT APPLICATION"

appUI="RSA SUPPORT: UNEMPLOYMENT COMPENSATION AT APPLICATION"

appPrimSupportCode="RSA SUPPORT: PRIMARY SOURCE OF SUPPORT AT APPLICATION"

appMedicaid="RSA MEDICAL INSURANCE COVERAGE: MEDICAID AT APPLICATION"

appMedicare="RSA MEDICAL INSURANCE COVERAGE: MEDICARE AT APPLICATION"

appACAExchange="RSA MEDICAL INSURANCE COVERAGE: STATE OR FEDERAL AFFORDABLE CARE ACT EXCHANGE AT APPLICATION"

appOtherPublicIns="RSA MEDICAL INSURANCE COVERAGE: PUBLIC INSURANCE FROM OTHER SOURCES AT APPLICATION"

appEmployerIns="RSA MEDICAL INSURANCE COVERAGE: PRIVATE INSURANCE THROUGH EMPLOYER AT APPLICATION"

appEmployerInsNYE="RSA MEDICAL INSURANCE COVERAGE: NOT YET ELIGIBLE FOR PRIVATE INSURANCE THROUGH EMPLOYER AT APPLICATION"

appOtherPrivateIns="RSA MEDICAL INSURANCE COVERAGE: PRIVATE INSURANCE

Appendix A.197

JCL/SAS Code: RSALABEL

THROUGH OTHER MEANS AT APPLICATION"  
EligibilityDate="RSA DATE OF VR ELIGIBILITY DETERMINATION"  
EligibilityExt="RSA VR ELIGIBILITY DETERMINATION EXTENSION"  
OOSPlacementDate="RSA DATE OF PLACEMENT ON OOS WAITING LIST"  
OOExitDate="RSA DATE OF EXIT FROM OOS WAITING LIST"  
Disability="RSA DISABILITY: INDIVIDUAL WITH A DISABILITY"  
PrimDisabilityCodeType="RSA DISABILITY: PRIMARY DISABILITY TYPE"  
PrimDisabilityCodeSource="RSA DISABILITY: PRIMARY DISABILITY SOURCE"  
SecondaryDisabilityType="RSA DISABILITY: SECONDARY DISABILITY TYPE"  
SecondaryDisabilitySource="RSA DISABILITY: SECONDARY DISABILITY SOURCE"  
DisabilitySigCode="RSA DISABILITY: SIGNIFICANCE OF DISABILITY"  
TWStartDate="RSA TRIAL WORK EXPERIENCE: START DATE OF TRIAL WORK EXPERIENCE"  
TWEEndDate="RSA TRIAL WORK EXPERIENCE: END DATE OF TRIAL WORK EXPERIENCE"  
IPEAmendedDate="RSA IPE: DATE OF MOST RECENT OR AMENDED IPE"  
IPESupportedEmpGoal="RSA IPE: SUPPORTED EMPLOYMENT GOAL ON CURRENT IPE"  
ipeEmpStatus="RSA IPE: EMPLOYMENT AT IPE"  
ipePrimaryOcc="RSA IPE: PRIMARY OCCUPATION AT IPE"  
ipeHourlyWage="RSA IPE: HOURLY WAGE AT IPE"  
ipeWeeklyHoursWorked="RSA IPE: HOURS WORKED IN A WEEK AT IPE"  
Adult="RSA WIOA PROGRAM: ADULT"  
AdultEd="RSA WIOA PROGRAM: ADULT EDUCATION"  
DislocatedWorker="RSA WIOA PROGRAM: DISLOCATED WORKER"  
JobCorps="RSA WIOA PROGRAM: JOB CORPS"  
VocRehab="RSA WIOA PROGRAM: VOCATIONAL REHABILITATION"  
WPEmpService="RSA WIOA PROGRAM: WAGNER-PEYSER EMPLOYMENT SERVICE"  
Youth="RSA WIOA PROGRAM: YOUTH"  
YouthBuild="RSA WIOA PROGRAM: YOUTHBUILD"  
LongTermUnemp="RSA BARRIERS TO EMPLOYMENT: LONG-TERM UNEMPLOYED"  
ExhaustTANF="RSA BARRIERS TO EMPLOYMENT: EXHAUSTING TANF WITHIN TWO YEARS"  
FosterCareYouth="RSA BARRIERS TO EMPLOYMENT: FOSTER CARE YOUTH"  
HomelessOrRunaway="RSA BARRIERS TO EMPLOYMENT: HOMELESS INDIVIDUAL, HOMELESS CHILDREN AND YOUTHS, OR RUNAWAY YOUTH"  
ExOffenderStatus="RSA BARRIERS TO EMPLOYMENT: EX-OFFENDER"  
LowIncomeStatus="RSA BARRIERS TO EMPLOYMENT: LOW INCOME"  
EnglishLearner="RSA BARRIERS TO EMPLOYMENT: ENGLISH LANGUAGE LEARNER"  
BasicSkillsDeficient="RSA BARRIERS TO EMPLOYMENT: BASIC SKILLS DEFICIENT/LOW LEVELS OF LITERACY"  
CulturalBarriers="RSA BARRIERS TO EMPLOYMENT: CULTURAL BARRIERS"  
SingleParent="RSA BARRIERS TO EMPLOYMENT: SINGLE PARENT"  
DisHomemaker="RSA BARRIERS TO EMPLOYMENT: DISPLACED HOMEMAKER"  
MigrantFarmworker="RSA BARRIERS TO EMPLOYMENT: MIGRANT AND SEASONAL FARMWORKER"

Appendix A.197

JCL/SAS Code: RSALABEL

StateDisStudentAgeRange="RSA EDUCATION: STATE DEFINITION FOR AGE OF STUDENTS WITH DISABILITIES"  
SchoolStatus="RSA EDUCATION: SCHOOL STATUS"  
EdLevelCompleted="RSA EDUCATION: HIGHEST EDUCATIONAL LEVEL COMPLETED"  
SchoolGradeCompleted="RSA EDUCATION: HIGHEST ELEMENTARY OR SECONDARY SCHOOL GRADE COMPLETED"  
InSecondaryEd="RSA EDUCATION: ENROLLED IN SECONDARY EDUCATION"  
SpecialEdCertCompDate="RSA EDUCATION: DATE RECEIVED SPECIAL EDUCATION CERTIFICATE OF COMPLETION"  
EnrolledInASE="RSA EDUCATION: ENROLLED IN A STATE ADULT SECONDARY SCHOOL AT THE HIGH ADULT SECONDARY EDUCATION (ASE) LEVEL"  
SecSchoolDiplomaDate="RSA EDUCATION: DATE ATTAINED SECONDARY SCHOOL DIPLOMA"  
GEDDate="RSA EDUCATION: DATE ATTAINED SECONDARY SCHOOL EQUIVALENCY (GED)"  
PostsecEdYear="RSA EDUCATION: ENROLLED IN POSTSECONDARY OR GRADUATE EDUCATION"  
EnrolledInPostsecEd="RSA EDUCATION: ENROLLED IN POSTSECONDARY EDUCATION"  
CredProgramEnrollDate="RSA EDUCATION: DATE ENROLLED DURING PROGRAM PARTICIPATION IN AN EDUCATION OR TRAINING PROGRAM LEADING TO A RECOGNIZED POSTSECONDARY CREDENTIAL OR EMPLOYMENT"  
CompletedSomePostsecEd="RSA EDUCATION: COMPLETED SOME POSTSECONDARY EDUCATION, NO DEGREE OR CERTIFICATE"  
AssociateDegreeDate="RSA EDUCATION: DATE ATTAINED ASSOCIATE S DEGREE"  
BachelorDegreeDate="RSA EDUCATION: DATE ATTAINED BACHELOR'S DEGREE"  
MastersDegreeDate="RSA EDUCATION: DATE ATTAINED MASTER S DEGREE"  
DegreeAboveMastersDate="RSA EDUCATION: DATE ATTAINED GRADUATE DEGREE (E.G., PH.D., ED.D., J.D., M.D.)"  
EnrolledInCTTPNoCred="RSA VOTECH: ENROLLED IN A CAREER OR TECHNICAL TRAINING PROGRAM, NOT LEADING TO A RECOGNIZED POSTSECONDARY CREDENTIAL"  
EnrolledInCTTPCred="RSA VOTECH: ENROLLED IN A CAREER OR TECHNICAL TRAINING PROGRAM, LEADING TO A RECOGNIZED POSTSECONDARY CREDENTIAL"  
VTLicenseDate="RSA VOTECH: DATE ATTAINED VOCATIONAL/TECHNICAL LICENSE"  
VTCertificateDate="RSA VOTECH: DATE ATTAINED VOCATIONAL/TECHNICAL CERTIFICATE"  
OtherLicOrCertDate="RSA OTHER EDUCATION OR TRAINING: DATE ATTAINED OTHER RECOGNIZED DIPLOMA, DEGREE, OR CERTIFICATE"  
PETSSStartDate="RSA PRE-EMP TRANSITION SERVICES: START DATE OF PRE-EMPLOYMENT TRANSITION SERVICES"  
jecVRAgencyStaff="RSA JOB EXPLORATION COUNSELING: SERVICE PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"  
jecVRAgencyPurchase="RSA JOB EXPLORATION COUNSELING: SERVICE PROVIDED THROUGH VR AGENCY PURCHASE"  
jecPurchaseProviderType="RSA JOB EXPLORATION COUNSELING: PURCHASED SERVICE PROVIDER TYPE"

Appendix A.197

JCL/SAS Code: RSALABEL

```
jecVRServicePurchaseExpenditure="RSA JOB EXPLORATION COUNSELING: VR  
PROGRAM EXPENDITURE FOR PURCHASED SERVICE"  
jecCompServiceProvider="RSA JOB EXPLORATION COUNSELING: SERVICE PROVIDED  
BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
jecCompServiceProviderType="RSA JOB EXPLORATION COUNSELING: COMPARABLE  
SERVICES AND BENEFITS PROVIDER TYPE"  
wbleVRAgencyStaff="RSA WORK BASED LEARNING EXPERIENCES: SERVICE PROVIDED  
BY VR AGENCY STAFF (IN-HOUSE)"  
wbleVRAgencyPurchase="RSA WORK BASED LEARNING EXPERIENCES: SERVICE  
PROVIDED THROUGH VR AGENCY PURCHASE"  
wblePurchaseProviderType="RSA WORK BASED LEARNINGS: PURCHASED SERVICE  
PROVIDER TYPE"  
wbleVRServicePurchaseExpenditure="RSA WORK BASED LEARNING: VR PROGRAM  
EXPENDITURE FOR PURCHASED SERVICE"  
wbleCompServiceProvider="RSA WORK BASED LEARNING: SERVICE PROVIDED BY  
COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
wbleCompServiceProviderType="RSA WORK BASED LEARNING: COMPARABLE  
SERVICES AND BENEFITS PROVIDER TYPE"  
ceoVRAgencyStaff="RSA COUNSELING ON ENROLLMENT OPPORTUNITIES: SERVICE  
PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"  
ceoVRAgencyPurchase="RSA COUNSELING ON ENROLLMENT OPPORTUNITIES: SERVICE  
PROVIDED THROUGH VR AGENCY PURCHASE"  
ceoPurchaseProviderType="RSA JOB EXPLORATION COUNSELING: PURCHASED  
SERVICE PROVIDER TYPE"  
ceoVRServicePurchaseExpenditure="RSA COUNSELING ON ENROLLMENT  
OPPORTUNITIES: VR PROGRAM EXPENDITURE FOR PURCHASED SERVICE"  
ceoCompServiceProvider="RSA COUNSELING ON ENROLLMENT OPPORTUNITIES:  
SERVICE PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
ceoCompServiceProviderType="RSA COUNSELING ON ENROLLMENT OPPORTUNITIES:  
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"  
wrtVRAgencyStaff="RSA WORKPLACE READINESS TRAINING: SERVICE PROVIDED BY  
VR AGENCY STAFF (IN-HOUSE)"  
wrtVRAgencyPurchase="RSA WORKPLACE READINESS TRAINING: SERVICE PROVIDED  
THROUGH VR AGENCY PURCHASE"  
wrtPurchaseProviderType="RSA WORKPLACE READINESS TRAINING: PURCHASED  
SERVICE PROVIDER TYPE"  
wrtVRServicePurchaseExpenditure="RSA WORKPLACE READINESS TRAINING: VR  
PROGRAM EXPENDITURE FOR PURCHASED SERVICE"  
wrtCompServiceProvider="RSA WORKPLACE READINESS TRAINING: SERVICE  
PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
wrtCompServiceProviderType="RSA WORKPLACE READINESS TRAINING: COMPARABLE  
SERVICES AND BENEFITS PROVIDER TYPE"  
isaVRAgencyStaff="RSA INSTRUCTION IN SELF ADVOCACY: SERVICE PROVIDED BY  
VR AGENCY STAFF (IN-HOUSE)"  
isaVRAgencyPurchase="RSA INSTRUCTION IN SELF ADVOCACY: SERVICE PROVIDED
```

Appendix A.197

JCL/SAS Code: RSALABEL

```
THROUGH VR AGENCY PURCHASE"
isaPurchaseProviderType="RSA INSTRUCTION IN SELF ADVOCACY: PURCHASED
SERVICE PROVIDER TYPE"
isaVRServicePurchaseExpenditure="RSA INSTRUCTION IN SELF ADVOCACY: VR
PROGRAM EXPENDITURE FOR PURCHASED SERVICE"
isaCompServiceProvider="RSA INSTRUCTION IN SELF ADVOCACY: SERVICE
PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"
isaCompServiceProviderType="RSA INSTRUCTION IN SELF ADVOCACY: COMPARABLE
SERVICES AND BENEFITS PROVIDER TYPE"
VRServiceStartDate="RSA START DATE OF INITIAL VR SERVICE ON OR AFTER
IPE"
CareerServiceDate="RSA DATE OF MOST RECENT CAREER SERVICE"
gcutVRAgencyStaff="RSA GRADUATE COLLEGE OR UNIVERSITY: SERVICE PROVIDED
BY VR AGENCY STAFF (IN-HOUSE)"
gcutVRAgencyPurchase="RSA GRADUATE COLLEGE OR UNIVERSITY: SERVICE
PROVIDED THROUGH VR AGENCY PURCHASE"
gcutPurchaseProviderType="RSA GRADUATE COLLEGE OR UNIVERSITY: PURCHASED
SERVICE PROVIDER TYPE"
gcutVRTitleIFundsExpended="RSA GRADUATE COLLEGE OR UNIVERSITY: AMOUNT OF
VR FUNDS EXPENDED FOR SERVICE (TITLE I)"
gcutSETitleVIFundsExpended="RSA GRADUATE COLLEGE OR UNIVERSITY: AMOUNT
OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"
gcutCompServiceProvider="RSA GRADUATE COLLEGE OR UNIVERSITY: SERVICE
PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"
gcutCompServiceProviderType="RSA GRADUATE COLLEGE OR UNIVERSITY:
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
fycutVRAgencyStaff="RSA FOUR-YEAR COLLEGE OR UNIVERSITY TRAINING:
SERVICE PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"
fycutVRAgencyPurchase="RSA FOUR-YEAR COLLEGE OR UNIVERSITY TRAINING:
SERVICE PROVIDED THROUGH VR AGENCY PURCHASE"
fycutPurchaseProviderType="RSA RSA FOUR-YEAR COLLEGE OR UNIVERSITY
TRAINING: PURCHASED SERVICE PROVIDER TYPE"
fycutVRTitleIFundsExpended="RSA RSA FOUR-YEAR COLLEGE OR UNIVERSITY
TRAINING: AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"
fycutSETitleVIFundsExpended="RSA RSA FOUR-YEAR COLLEGE OR UNIVERSITY
TRAINING: AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"
fycutCompServiceProvider="RSA RSA FOUR-YEAR COLLEGE OR UNIVERSITY
TRAINING: SERVICE PROVIDED BY COMPARABLE SERVICES AND BENEFITS
PROVIDERS"
fycutCompServiceProviderType="RSA RSA FOUR-YEAR COLLEGE OR UNIVERSITY
TRAINING: COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
jcctVRAgencyStaff="RSA JUNIOR OR COMMUNITY COLLEGE TRAINING: SERVICE
PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"
jcctVRAgencyPurchase="RSA JUNIOR OR COMMUNITY COLLEGE TRAINING: SERVICE
PROVIDED THROUGH VR AGENCY PURCHASE"
```

Appendix A.197

JCL/SAS Code: RSALABEL

```
jcctPurchaseProviderType="RSA JUNIOR OR COMMUNITY COLLEGE TRAINING:
PURCHASED SERVICE PROVIDER TYPE"
jcctVRTitleIFundsExpended="RSA JUNIOR OR COMMUNITY COLLEGE TRAINING:
AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"
jcctSETitleVIFundsExpended="RSA JUNIOR OR COMMUNITY COLLEGE TRAINING:
AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"
jcctCompServiceProvider="RSA JUNIOR OR COMMUNITY COLLEGE TRAINING:
SERVICE PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"
jcctCompServiceProviderType="RSA JUNIOR OR COMMUNITY COLLEGE TRAINING:
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
ovtVRAgencyStaff="RSA OCCUPATIONAL OR VOCATIONAL TRAINING: SERVICE
PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"
ovtVRAgencyPurchase="RSA OCCUPATIONAL OR VOCATIONAL TRAINING: SERVICE
PROVIDED THROUGH VR AGENCY PURCHASE"
ovtPurchaseProviderType="RSA OCCUPATIONAL OR VOCATIONAL TRAINING:
PURCHASED SERVICE PROVIDER TYPE"
ovtVRTitleIFundsExpended="RSA OCCUPATIONAL OR VOCATIONAL TRAINING:
AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"
ovtSETitleVIFundsExpended="RSA OCCUPATIONAL OR VOCATIONAL TRAINING:
AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"
ovtCompServiceProvider="RSA OCCUPATIONAL OR VOCATIONAL TRAINING: SERVICE
PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"
ovtCompServiceProviderType="RSA OCCUPATIONAL OR VOCATIONAL TRAINING:
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
ojtVRAgencyStaff="RSA ON THE JOB TRAINING: SERVICE PROVIDED BY VR AGENCY
STAFF (IN-HOUSE)"
ojtVRAgencyPurchase="RSA ON THE JOB TRAINING: SERVICE PROVIDED THROUGH
VR AGENCY PURCHASE"
ojtPurchaseProviderType="RSA ON THE JOB TRAINING: PURCHASED SERVICE
PROVIDER TYPE"
ojtVRTitleIFundsExpended="RSA ON THE JOB TRAINING: AMOUNT OF VR FUNDS
EXPENDED FOR SERVICE (TITLE I)"
ojtSETitleVIFundsExpended="RSA ON THE JOB TRAINING: AMOUNT OF SE FUNDS
EXPENDED FOR SERVICE (TITLE VI)"
ojtCompServiceProvider="RSA ON THE JOB TRAINING: SERVICE PROVIDED BY
COMPARABLE SERVICES AND BENEFITS PROVIDERS"
ojtCompServiceProviderType="RSA ON THE JOB TRAINING: COMPARABLE SERVICES
AND BENEFITS PROVIDER TYPE"
ratVRAgencyPurchase="RSA REGISTERED APPRENTICESHIP TRAINING: SERVICE
PROVIDED THROUGH VR AGENCY PURCHASE"
ratPurchaseProviderType="RSA REGISTERED APPRENTICESHIP TRAINING:
PURCHASED SERVICE PROVIDER TYPE"
ratVRTitleIFundsExpended="RSA REGISTERED APPRENTICESHIP TRAINING: AMOUNT
OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"
ratSETitleVIFundsExpended="RSA REGISTERED APPRENTICESHIP TRAINING:
```



Appendix A.197

JCL/SAS Code: RSALABEL

```
AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"
ratCompServiceProvider="RSA REGISTERED APPRENTICESHIP TRAINING: SERVICE
PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"
ratCompServiceProviderType="RSA REGISTERED APPRENTICESHIP TRAINING:
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
barltVRAgencyStaff="RSA BASIC ACADEMIC REMEDIAL OR LITERACY TRAINING:
SERVICE PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"
barltVRAgencyPurchase="RSA BASIC ACADEMIC REMEDIAL OR LITERACY TRAINING:
SERVICE PROVIDED THROUGH VR AGENCY PURCHASE"
barltPurchaseProviderType="RSA BASIC ACADEMIC REMEDIAL OR LITERACY
TRAINING: PURCHASED SERVICE PROVIDER TYPE"
barltVRTitleIFundsExpended="RSA BASIC ACADEMIC REMEDIAL OR LITERACY
TRAINING: AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"
barltSETitleVIFundsExpended="RSA BASIC ACADEMIC REMEDIAL OR LITERACY
TRAINING: AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"
barltCompServiceProvider="RSA BASIC ACADEMIC REMEDIAL OR LITERACY
TRAINING: SERVICE PROVIDED BY COMPARABLE SERVICES AND BENEFITS
PROVIDERS"
barltCompServiceProviderType="RSA BASIC ACADEMIC REMEDIAL OR LITERACY
TRAINING: COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
jrtVRAgencyStaff="RSA JOB READINESS TRAINING: SERVICE PROVIDED BY VR
AGENCY STAFF (IN-HOUSE)"
jrtVRAgencyPurchase="RSA JOB READINESS TRAINING: SERVICE PROVIDED
THROUGH VR AGENCY PURCHASE"
jrtPurchaseProviderType="RSA JOB READINESS TRAINING: PURCHASED SERVICE
PROVIDER TYPE"
jrtVRTitleIFundsExpended="RSA JOB READINESS TRAINING: AMOUNT OF VR FUNDS
EXPENDED FOR SERVICE (TITLE I)"
jrtSETitleVIFundsExpended="RSA JOB READINESS TRAINING: AMOUNT OF SE
FUNDS EXPENDED FOR SERVICE (TITLE VI)"
jrtCompServiceProvider="RSA JOB READINESS TRAINING: SERVICE PROVIDED BY
COMPARABLE SERVICES AND BENEFITS PROVIDERS"
jrtCompServiceProviderType="RSA JOB READINESS TRAINING: COMPARABLE
SERVICES AND BENEFITS PROVIDER TYPE"
drstVRAgencyStaff="RSA DISABILITY RELATED SKILLS TRAINING: SERVICE
PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"
drstVRAgencyPurchase="RSA DISABILITY RELATED SKILLS TRAINING: SERVICE
PROVIDED THROUGH VR AGENCY PURCHASE"
drstPurchaseProviderType="RSA DISABILITY RELATED SKILLS TRAINING:
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
drstVRTitleIFundsExpended="RSA DISABILITY RELATED SKILLS TRAINING:
AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"
drstSETitleVIFundsExpended="RSA DISABILITY RELATED SKILLS TRAINING:
AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"
drstCompServiceProvider="RSA DISABILITY RELATED SKILLS TRAINING: SERVICE
```

Appendix A.197

JCL/SAS Code: RSALABEL

```
PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"
drstCompServiceProviderType="RSA DISABILITY RELATED SKILLS TRAINING:
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
mtVRAgencyStaff="RSA MISCELLANEOUS TRAINING: SERVICE PROVIDED BY VR
AGENCY STAFF (IN-HOUSE)"
mtVRAgencyPurchase="RSA MISCELLANEOUS TRAINING: SERVICE PROVIDED THROUGH
VR AGENCY PURCHASE"
mtPurchaseProviderType="RSA MISCELLANEOUS TRAINING: PURCHASED SERVICE
PROVIDER TYPE"
mtVRTitleIFundsExpended="RSA MISCELLANEOUS TRAINING: AMOUNT OF VR FUNDS
EXPENDED FOR SERVICE (TITLE I)"
mtSETitleVIFundsExpended="RSA MISCELLANEOUS TRAINING: AMOUNT OF SE FUNDS
EXPENDED FOR SERVICE (TITLE VI)"
mtCompServiceProvider="RSA MISCELLANEOUS TRAINING: SERVICE PROVIDED BY
COMPARABLE SERVICES AND BENEFITS PROVIDERS"
mtCompServiceProviderType="RSA MISCELLANEOUS TRAINING: COMPARABLE
SERVICES AND BENEFITS PROVIDER TYPE"
rsetVRAgencyStaff="RSA RANDOLPH-SHEPPARD ENTREPRENEURIAL TRAINING:
SERVICE PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"
rsetVRAgencyPurchase="RSA RANDOLPH-SHEPPARD ENTREPRENEURIAL TRAINING:
SERVICE PROVIDED THROUGH VR AGENCY PURCHASE"
rsetPurchaseProviderType="RSA RANDOLPH-SHEPPARD ENTREPRENEURIAL
TRAINING: PURCHASED SERVICE PROVIDER TYPE"
rsetVRTitleIFundsExpended="RSA RANDOLPH-SHEPPARD ENTREPRENEURIAL
TRAINING: AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"
rsetSETitleVIFundsExpended="RSA RANDOLPH-SHEPPARD ENTREPRENEURIAL
TRAINING: AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"
rsetCompServiceProvider="RSA RANDOLPH-SHEPPARD ENTREPRENEURIAL TRAINING:
SERVICE PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"
rsetCompServiceProviderType="RSA RANDOLPH-SHEPPARD ENTREPRENEURIAL
TRAINING: COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
ctVRAgencyStaff="RSA CUSTOMIZED TRAINING: SERVICE PROVIDED BY VR AGENCY
STAFF (IN-HOUSE)"
ctVRAgencyPurchase="RSA CUSTOMIZED TRAINING: SERVICE PROVIDED THROUGH VR
AGENCY PURCHASE"
ctPurchaseProviderType="RSA CUSTOMIZED TRAINING: PURCHASED SERVICE
PROVIDER TYPE"
ctVRTitleIFundsExpended="RSA CUSTOMIZED TRAINING: AMOUNT OF VR FUNDS
EXPENDED FOR SERVICE (TITLE I)"
ctSETitleVIFundsExpended="RSA CUSTOMIZED TRAINING: AMOUNT OF SE FUNDS
EXPENDED FOR SERVICE (TITLE VI)"
ctCompServiceProvider="RSA CUSTOMIZED TRAINING: SERVICE PROVIDED BY
COMPARABLE SERVICES AND BENEFITS PROVIDERS"
ctCompServiceProviderType="RSA CUSTOMIZED TRAINING: COMPARABLE SERVICES
AND BENEFITS PROVIDER TYPE"
```

Appendix A.197

JCL/SAS Code: RSALABEL

```
assVRAgencyStaff="RSA ASSESSMENT: SERVICE PROVIDED BY VR AGENCY STAFF  
(IN-HOUSE)"  
assVRAgencyPurchase="RSA ASSESSMENT: SERVICE PROVIDED THROUGH VR AGENCY  
PURCHASE"  
assPurchaseProviderType="RSA ASSESSMENT: PURCHASED SERVICE PROVIDER  
TYPE"  
assVRTitleIFundsExpended="RSA ASSESSMENT: AMOUNT OF VR FUNDS EXPENDED  
FOR SERVICE (TITLE I)"  
assSETitleVIFundsExpended="RSA ASSESSMENT: AMOUNT OF SE FUNDS EXPENDED  
FOR SERVICE (TITLE VI)"  
assCompServiceProvider="RSA ASSESSMENT: SERVICE PROVIDED BY COMPARABLE  
SERVICES AND BENEFITS PROVIDERS"  
assCompServiceProviderType="RSA ASSESSMENT: COMPARABLE SERVICES AND  
BENEFITS PROVIDER TYPE"  
dtiVRAgencyStaff="RSA DIAGNOSIS AND TREATMENT OF IMPAIRMENTS: SERVICE  
PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"  
dtiVRAgencyPurchase="RSA DIAGNOSIS AND TREATMENT OF IMPAIRMENTS: SERVICE  
PROVIDED THROUGH VR AGENCY PURCHASE"  
dtiPurchaseProviderType="RSA DIAGNOSIS AND TREATMENT OF IMPAIRMENTS:  
PURCHASED SERVICE PROVIDER TYPE"  
dtiVRTitleIFundsExpended="RSA DIAGNOSIS AND TREATMENT OF IMPAIRMENTS:  
AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"  
dtiSETitleVIFundsExpended="RSA DIAGNOSIS AND TREATMENT OF IMPAIRMENTS:  
AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"  
dtiCompServiceProvider="RSA DIAGNOSIS AND TREATMENT OF IMPAIRMENTS:  
SERVICE PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
dtiCompServiceProviderType="RSA DIAGNOSIS AND TREATMENT OF IMPAIRMENTS:  
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"  
vrcgVRAgencyStaff="RSA VOCATIONAL REHABILITATION COUNSELING AND  
GUIDANCE: SERVICE PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"  
vrcgVRAgencyPurchase="RSA VOCATIONAL REHABILITATION COUNSELING AND  
GUIDANCE: SERVICE PROVIDED THROUGH VR AGENCY PURCHASE"  
vrcgPurchaseProviderType="RSA VOCATIONAL REHABILITATION COUNSELING AND  
GUIDANCE: PURCHASED SERVICE PROVIDER TYPE"  
vrcgVRTitleIFundsExpended="RSA VOCATIONAL REHABILITATION COUNSELING AND  
GUIDANCE: AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"  
vrcgSETitleVIFundsExpended="RSA VOCATIONAL REHABILITATION COUNSELING AND  
GUIDANCE: AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"  
vrcgCompServiceProvider="RSA VOCATIONAL REHABILITATION COUNSELING AND  
GUIDANCE: SERVICE PROVIDED BY COMPARABLE SERVICES AND BENEFITS  
PROVIDERS"  
vrcgCompServiceProviderType="RSA VOCATIONAL REHABILITATION COUNSELING  
AND GUIDANCE: COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"  
jsaVRAgencyStaff="RSA JOB SEARCH ASSISTANCE: SERVICE PROVIDED BY VR  
AGENCY STAFF (IN-HOUSE)"
```

Appendix A.197

JCL/SAS Code: RSALABEL

```
jsaVRAgencyPurchase="RSA JOB SEARCH ASSISTANCE: SERVICE PROVIDED THROUGH  
VR AGENCY PURCHASE"  
jsaPurchaseProviderType="RSA JOB SEARCH ASSISTANCE: PURCHASED SERVICE  
PROVIDER TYPE"  
jsaVRTitleIFundsExpended="RSA JOB SEARCH ASSISTANCE: AMOUNT OF VR FUNDS  
EXPENDED FOR SERVICE (TITLE I)"  
jsaSETitleVIFundsExpended="RSA JOB SEARCH ASSISTANCE: AMOUNT OF SE FUNDS  
EXPENDED FOR SERVICE (TITLE VI)"  
jsaCompServiceProvider="RSA JOB SEARCH ASSISTANCE: SERVICE PROVIDED BY  
COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
jsaCompServiceProviderType="RSA JOB SEARCH ASSISTANCE: COMPARABLE  
SERVICES AND BENEFITS PROVIDER TYPE"  
jpaVRAgencyStaff="RSA JOB PLACEMENT ASSISTANCE: SERVICE PROVIDED BY VR  
AGENCY STAFF (IN-HOUSE)"  
jpaVRAgencyPurchase="RSA JOB PLACEMENT ASSISTANCE: SERVICE PROVIDED  
THROUGH VR AGENCY PURCHASE"  
jpaPurchaseProviderType="RSA JOB PLACEMENT ASSISTANCE: PURCHASED SERVICE  
PROVIDER TYPE"  
jpaVRTitleIFundsExpended="RSA JOB PLACEMENT ASSISTANCE: AMOUNT OF VR  
FUNDS EXPENDED FOR SERVICE (TITLE I)"  
jpaSETitleVIFundsExpended="RSA JOB PLACEMENT ASSISTANCE: AMOUNT OF SE  
FUNDS EXPENDED FOR SERVICE (TITLE VI)"  
jpaCompServiceProvider="RSA JOB PLACEMENT ASSISTANCE: SERVICE PROVIDED  
BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
stjsVRAgencyStaff="RSA SHORT TERM JOB SUPPORTS: SERVICE PROVIDED BY VR  
AGENCY STAFF (IN-HOUSE)"  
stjsVRAgencyPurchase="RSA SHORT TERM JOB SUPPORTS: SERVICE PROVIDED  
THROUGH VR AGENCY PURCHASE"  
stjsPurchaseProviderType="RSA SHORT TERM JOB SUPPORTS: PURCHASED SERVICE  
PROVIDER TYPE"  
stjsVRTitleIFundsExpended="RSA SHORT TERM JOB SUPPORTS: AMOUNT OF VR  
FUNDS EXPENDED FOR SERVICE (TITLE I)"  
stjsSETitleVIFundsExpended="RSA SHORT TERM JOB SUPPORTS: AMOUNT OF SE  
FUNDS EXPENDED FOR SERVICE (TITLE VI)"  
stjsCompServiceProvider="RSA SHORT TERM JOB SUPPORTS: SERVICE PROVIDED  
BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
stjsCompServiceProviderType="RSA SHORT TERM JOB SUPPORTS: COMPARABLE  
SERVICES AND BENEFITS PROVIDER TYPE"  
sesVRAgencyStaff="RSA SUPPORTED EMPLOYMENT SERVICES: SERVICE PROVIDED BY  
VR AGENCY STAFF (IN-HOUSE)"  
sesVRAgencyPurchase="RSA SUPPORTED EMPLOYMENT SERVICES: SERVICE PROVIDED  
THROUGH VR AGENCY PURCHASE"  
sesPurchaseProviderType="RSA SUPPORTED EMPLOYMENT SERVICES: PURCHASED  
SERVICE PROVIDER TYPE"  
sesVRTitleIFundsExpended="RSA SUPPORTED EMPLOYMENT SERVICES: AMOUNT OF
```

Appendix A.197

JCL/SAS Code: RSALABEL

```
VR FUNDS EXPENDED FOR SERVICE (TITLE I)"
sesSETitleVIFundsExpended="RSA SUPPORTED EMPLOYMENT SERVICES: AMOUNT OF
SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"
sesCompServiceProvider="RSA SUPPORTED EMPLOYMENT SERVICES: SERVICE
PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"
sesCompServiceProviderType="RSA SUPPORTED EMPLOYMENT SERVICES:
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
irsVRAgencyStaff="RSA INFORMATION AND REFERRAL SERVICES: SERVICE
PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"
irsVRAgencyPurchase="RSA INFORMATION AND REFERRAL SERVICES: SERVICE
PROVIDED THROUGH VR AGENCY PURCHASE"
irsPurchaseProviderType="RSA INFORMATION AND REFERRAL SERVICES:
PURCHASED SERVICE PROVIDER TYPE"
irsVRTitleIFundsExpended="RSA INFORMATION AND REFERRAL SERVICES: AMOUNT
OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"
irsSETitleVIFundsExpended="RSA INFORMATION AND REFERRAL SERVICES: AMOUNT
OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"
irsCompServiceProvider="RSA INFORMATION AND REFERRAL SERVICES: SERVICE
PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"
irsCompServiceProviderType="RSA INFORMATION AND REFERRAL SERVICES:
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
bcVRAgencyStaff="RSA BENEFITS COUNSELING: SERVICE PROVIDED BY VR AGENCY
STAFF (IN-HOUSE)"
bcVRAgencyPurchase="RSA BENEFITS COUNSELING: SERVICE PROVIDED THROUGH VR
AGENCY PURCHASE"
bcPurchaseProviderType="RSA BENEFITS COUNSELING: PURCHASED SERVICE
PROVIDER TYPE"
bcVRTitleIFundsExpended="RSA BENEFITS COUNSELING: AMOUNT OF VR FUNDS
EXPENDED FOR SERVICE (TITLE I)"
bcSETitleVIFundsExpended="RSA BENEFITS COUNSELING: AMOUNT OF SE FUNDS
EXPENDED FOR SERVICE (TITLE VI)"
bcCompServiceProvider="RSA BENEFITS COUNSELING: SERVICE PROVIDED BY
COMPARABLE SERVICES AND BENEFITS PROVIDERS"
bcCompServiceProviderType="RSA BENEFITS COUNSELING: COMPARABLE SERVICES
AND BENEFITS PROVIDER TYPE"
cesVRAgencyStaff="RSA CUSTOMIZED EMPLOYMENT SERVICES: SERVICE PROVIDED
BY VR AGENCY STAFF (IN-HOUSE)"
cesVRAgencyPurchase="RSA CUSTOMIZED EMPLOYMENT SERVICES: SERVICE
PROVIDED THROUGH VR AGENCY PURCHASE"
cesPurchaseProviderType="RSA CUSTOMIZED EMPLOYMENT SERVICES: PURCHASED
SERVICE PROVIDER TYPE"
cesVRTitleIFundsExpended="RSA CUSTOMIZED EMPLOYMENT SERVICES: AMOUNT OF
VR FUNDS EXPENDED FOR SERVICE (TITLE I)"
cesSETitleVIFundsExpended="RSA CUSTOMIZED EMPLOYMENT SERVICES: AMOUNT OF
SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"
```

Appendix A.197

JCL/SAS Code: RSALABEL

```
cesCompServiceProvider="RSA CUSTOMIZED EMPLOYMENT SERVICES: SERVICE
PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"
cesCompServiceProviderType="RSA CUSTOMIZED EMPLOYMENT SERVICES:
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"
esVRAgencyStaff="RSA EXTENDED SERVICES: SERVICE PROVIDED BY VR AGENCY
STAFF (IN-HOUSE)"
esVRAgencyPurchase="RSA EXTENDED SERVICES: SERVICE PROVIDED THROUGH VR
AGENCY PURCHASE"
esPurchaseProviderType="RSA EXTENDED SERVICES: PURCHASED SERVICE
PROVIDER TYPE"
esVRTitleIFundsExpended="RSA EXTENDED SERVICES: AMOUNT OF VR FUNDS
EXPENDED FOR SERVICE (TITLE I)"
esSETitleVIFundsExpended="RSA EXTENDED SERVICES: AMOUNT OF SE FUNDS
EXPENDED FOR SERVICE (TITLE VI)"
tranVRAgencyStaff="RSA TRANSPORTATION: SERVICE PROVIDED BY VR AGENCY
STAFF (IN-HOUSE)"
tranVRAgencyPurchase="RSA TRANSPORTATION: SERVICE PROVIDED THROUGH VR
AGENCY PURCHASE"
tranPurchaseProviderType="RSA TRANSPORTATION: PURCHASED SERVICE PROVIDER
TYPE"
tranVRTitleIFundsExpended="RSA TRANSPORTATION: AMOUNT OF VR FUNDS
EXPENDED FOR SERVICE (TITLE I)"
tranSETitleVIFundsExpended="RSA TRANSPORTATION: AMOUNT OF SE FUNDS
EXPENDED FOR SERVICE (TITLE VI)"
tranCompServiceProvider="RSA TRANSPORTATION: SERVICE PROVIDED BY
COMPARABLE SERVICES AND BENEFITS PROVIDERS"
tranCompServiceProviderType="RSA TRANSPORTATION: COMPARABLE SERVICES AND
BENEFITS PROVIDER TYPE"
mntVRAgencyStaff="RSA MAINTENANCE: SERVICE PROVIDED BY VR AGENCY STAFF
(IN-HOUSE)"
mntVRAgencyPurchase="RSA MAINTENANCE: SERVICE PROVIDED THROUGH VR AGENCY
PURCHASE"
mntPurchaseProviderType="RSA MAINTENANCE: PURCHASED SERVICE PROVIDER
TYPE"
mntVRTitleIFundsExpended="RSA MAINTENANCE: AMOUNT OF VR FUNDS EXPENDED
FOR SERVICE (TITLE I)"
mntSETitleVIFundsExpended="RSA MAINTENANCE: AMOUNT OF SE FUNDS EXPENDED
FOR SERVICE (TITLE VI)"
mntCompServiceProvider="RSA MAINTENANCE: SERVICE PROVIDED BY COMPARABLE
SERVICES AND BENEFITS PROVIDERS"
mntCompServiceProviderType="RSA MAINTENANCE: COMPARABLE SERVICES AND
BENEFITS PROVIDER TYPE"
rtVRAgencyStaff="RSA REHABILITATION TECHNOLOGY: SERVICE PROVIDED BY VR
AGENCY STAFF (IN-HOUSE)"
rtVRAgencyPurchase="RSA REHABILITATION TECHNOLOGY: SERVICE PROVIDED
```

Appendix A.197

JCL/SAS Code: RSALABEL

THROUGH VR AGENCY PURCHASE"  
rtPurchaseProviderType="RSA REHABILITATION TECHNOLOGY: PURCHASED SERVICE PROVIDER TYPE"  
rtVRTitleIFundsExpended="RSA REHABILITATION TECHNOLOGY: AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"  
rtSETitleVIFundsExpended="RSA REHABILITATION TECHNOLOGY: AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"  
rtCompServiceProvider="RSA REHABILITATION TECHNOLOGY: SERVICE PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
rtCompServiceProviderType="RSA REHABILITATION TECHNOLOGY: COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"  
pasVRAgencyStaff="RSA PERSONAL ASSISTANCE SERVICES: SERVICE PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"  
pasVRAgencyPurchase="RSA PERSONAL ASSISTANCE SERVICES: SERVICE PROVIDED THROUGH VR AGENCY PURCHASE"  
pasPurchaseProviderType="RSA PERSONAL ASSISTANCE SERVICES: PURCHASED SERVICE PROVIDER TYPE"  
pasVRTitleIFundsExpended="RSA PERSONAL ASSISTANCE SERVICES: AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"  
pasSETitleVIFundsExpended="RSA PERSONAL ASSISTANCE SERVICES: AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"  
pasCompServiceProvider="RSA PERSONAL ASSISTANCE SERVICES: SERVICE PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
pasCompServiceProviderType="RSA PERSONAL ASSISTANCE SERVICES: COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"  
tasVRAgencyStaff="RSA TECHNICAL ASSISTANCE SERVICES INCLUDING SELF-EMPLOYMENT: SERVICE PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"  
tasVRAgencyPurchase="RSA TECHNICAL ASSISTANCE SERVICES INCLUDING SELF-EMPLOYMENT: SERVICE PROVIDED THROUGH VR AGENCY PURCHASE"  
tasPurchaseProviderType="RSA TECHNICAL ASSISTANCE SERVICES INCLUDING SELF-EMPLOYMENT: PURCHASED SERVICE PROVIDER TYPE"  
tasVRTitleIFundsExpended="RSA TECHNICAL ASSISTANCE SERVICES INCLUDING SELF-EMPLOYMENT: AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"  
tasSETitleVIFundsExpended="RSA TECHNICAL ASSISTANCE SERVICES INCLUDING SELF-EMPLOYMENT: AMOUNT OF SE FUNDS EXPENDED FOR SERVICE (TITLE VI)"  
tasCompServiceProvider="RSA TECHNICAL ASSISTANCE SERVICES INCLUDING SELF-EMPLOYMENT: SERVICE PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
tasCompServiceProviderType="RSA TECHNICAL ASSISTANCE SERVICES INCLUDING SELF-EMPLOYMENT: COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"  
rsVRAgencyStaff="RSA READER SERVICES: SERVICE PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"  
rsVRAgencyPurchase="RSA READER SERVICES: SERVICE PROVIDED THROUGH VR AGENCY PURCHASE"  
rsPurchaseProviderType="RSA READER SERVICES: PURCHASED SERVICE PROVIDER

Appendix A.197

JCL/SAS Code: RSALABEL

```
TYPE"
rsVRTitleIFundsExpended="RSA READER SERVICES: AMOUNT OF VR FUNDS
EXPENDED FOR SERVICE (TITLE I)"
rsSETitleVIFundsExpended="RSA READER SERVICES: AMOUNT OF SE FUNDS
EXPENDED FOR SERVICE (TITLE VI)"
rsCompServiceProvider="RSA READER SERVICES: SERVICE PROVIDED BY
COMPARABLE SERVICES AND BENEFITS PROVIDERS"
rsCompServiceProviderType="RSA READER SERVICES: COMPARABLE SERVICES AND
BENEFITS PROVIDER TYPE"
isVRAgencyStaff="RSA INTERPRETER SERVICES: SERVICE PROVIDED BY VR AGENCY
STAFF (IN-HOUSE)"
isVRAgencyPurchase="RSA INTERPRETER SERVICES: SERVICE PROVIDED THROUGH
VR AGENCY PURCHASE"
isPurchaseProviderType="RSA INTERPRETER SERVICES: PURCHASED SERVICE
PROVIDER TYPE"
isVRTitleIFundsExpended="RSA INTERPRETER SERVICES: AMOUNT OF VR FUNDS
EXPENDED FOR SERVICE (TITLE I)"
isSETitleVIFundsExpended="RSA INTERPRETER SERVICES: AMOUNT OF SE FUNDS
EXPENDED FOR SERVICE (TITLE VI)"
isCompServiceProvider="RSA INTERPRETER SERVICES: SERVICE PROVIDED BY
COMPARABLE SERVICES AND BENEFITS PROVIDERS"
isCompServiceProviderType="RSA INTERPRETER SERVICES: COMPARABLE SERVICES
AND BENEFITS PROVIDER TYPE"
osVRAgencyStaff="RSA OTHER SERVICES: SERVICE PROVIDED BY VR AGENCY STAFF
(IN-HOUSE)"
osVRAgencyPurchase="RSA OTHER SERVICES: SERVICE PROVIDED THROUGH VR
AGENCY PURCHASE"
osPurchaseProviderType="RSA OTHER SERVICES: PURCHASED SERVICE PROVIDER
TYPE"
osVRTitleIFundsExpended="RSA OTHER SERVICES: AMOUNT OF VR FUNDS EXPENDED
FOR SERVICE (TITLE I)"
osSETitleVIFundsExpended="RSA OTHER SERVICES: AMOUNT OF SE FUNDS
EXPENDED FOR SERVICE (TITLE VI)"
osCompServiceProvider="RSA OTHER SERVICES: SERVICE PROVIDED BY
COMPARABLE SERVICES AND BENEFITS PROVIDERS"
osCompServiceProviderType="RSA OTHER SERVICES: COMPARABLE SERVICES AND
BENEFITS PROVIDER TYPE"
EdFuncLevelDate="RSA DATE OF MOST RECENT MEASURABLE SKILL GAIN:
EDUCATIONAL FUNCTIONING LEVEL (EFL)"
SecondaryDate="RSA DATE OF MOST RECENT MEASURABLE SKILL GAIN: SECONDARY"
PostsecTransReportCardDate="RSA DATE OF MOST RECENT MEASURABLE SKILL
GAIN: POSTSECONDARY TRANSCRIPT/REPORT CARD"
TrainingMilestoneDate="RSA DATE OF MOST RECENT MEASURABLE SKILL GAIN:
TRAINING MILESTONE"
SkillGainSkillsProgDate="RSA DATE OF MOST RECENT MEASURABLE SKILL GAIN:
```



Appendix A.197

JCL/SAS Code: RSALABEL

SKILLS PROGRESSION"

EmpOutcome="RSA EMPLOYMENT OUTCOME"

eoPrimOccupation="RSA PRIMARY OCCUPATION AT EMPLOYMENT OUTCOME"

eoPrimOccupationStartDate="RSA START DATE OF EMPLOYMENT IN PRIMARY OCCUPATION"

eoHourlyWage="RSA HOURLY WAGE AT EMPLOYMENT"

eoHoursWorkedPerWeek="RSA HOURS WORKED IN A WEEK AT EMPLOYMENT OUTCOME"

exitDate="RSA DATE OF VR EXIT"

exitType="RSA TYPE OF VR EXIT"

exitReason="RSA REASON FOR VR EXIT"

exitEmpOutcome="RSA EMPLOYMENT OUTCOME AT VR EXIT"

exitPrimOccupation="RSA PRIMARY OCCUPATION AT VR EXIT"

exitPrimOccStartDate="RSA START DATE OF EMPLOYMENT IN PRIMARY OCCUPATION AT VR EXIT"

exitHourlyWage="RSA HOURLY WAGE AT VR EXIT"

exitWeeklyHoursWorked="RSA HOURS WORKED IN A WEEK AT EXIT"

exitSSDI="RSA SOCIAL SECURITY DISABILITY INSURANCE (SSDI) AT VR EXIT"

exitSSI="RSA SUPPLEMENTAL SECURITY INCOME (SSI) FOR THE AGED, BLIND, OR DISABLED AT VR EXIT"

exitTANF="RSA TEMPORARY ASSISTANCE FOR NEEDY FAMILIES (TANF) AT VR EXIT"

exitGenAssistance="RSA GENERAL ASSISTANCE (STATE OR LOCAL GOVERNMENT) AT VR EXIT"

exitVeteransDisability="RSA VETERANS' DISABILITY BENEFITS AT VR EXIT"

exitWorkersComp="RSA WORKERS' COMPENSATION AT VR EXIT"

exitOtherSupport="RSA OTHER PUBLIC SUPPORT AT VR EXIT"

exitPrimSupportCode="RSA PRIMARY SOURCE OF SUPPORT AT VR EXIT"

exitMedicaid="RSA MEDICAID AT VR EXIT"

exitMedicare="RSA MEDICARE AT VR EXIT"

exitACAExchange="RSA STATE OR FEDERAL AFFORDABLE CARE ACT EXCHANGE AT VR EXIT"

exitOtherPublicIns="RSA PUBLIC INSURANCE FROM OTHER SOURCES AT VR EXIT"

exitEmployerIns="RSA PRIVATE INSURANCE THROUGH EMPLOYER AT VR EXIT"

exitEmployerInsNYE="RSA NOT YET ELIGIBLE FOR PRIVATE INSURANCE THROUGH EMPLOYER AT VR EXIT"

exitOtherPrivateIns="RSA PRIVATE INSURANCE THROUGH OTHER MEANS AT EXIT"

peCredProgramEnrollDate="RSA DATE ENROLLED IN POST-EXIT EDUCATION OR TRAINING PROGRAM LEADING TO A RECOGNIZED A.POSTSECONDARY CREDENTIAL"

peCredAttainmentDate="RSA DATE OF ATTAINMENT OF POST-EXIT RECOGNIZED CREDENTIAL"

peCredentialType="RSA TYPE OF RECOGNIZED CREDENTIAL ATTAINED POST-EXIT"

q1aeEmpStatus="RSA EMPLOYMENT - FIRST QUARTER AFTER EXIT QUARTER"

q1aeEmpMatchType="RSA TYPE OF EMPLOYMENT MATCH -- FIRST QUARTER AFTER EXIT QUARTER"

q1aeWages="RSA WAGES -- FIRST QUARTER AFTER EXIT QUARTER"

q2aeEmpRelatedToTraining="RSA EMPLOYMENT RELATED TO TRAINING -- SECOND

Appendix A.197

JCL/SAS Code: RSALABEL

QUARTER AFTER EXIT QUARTER"  
q2aeEmpStatus="RSA EMPLOYMENT - SECOND QUARTER AFTER EXIT QUARTER"  
q2aeEmpMatchType="RSA TYPE OF EMPLOYMENT MATCH — SECOND QUARTER AFTER  
EXIT QUARTER"  
q2aeWages="RSA WAGES — SECOND QUARTER AFTER EXIT QUARTER"  
q3aeEmpStatus="RSA EMPLOYMENT - THIRD QUARTER AFTER EXIT QUARTER"  
q3aeEmpMatchType="RSA TYPE OF EMPLOYMENT MATCH — THIRD QUARTER AFTER  
EXIT QUARTER"  
q3aeWages="RSA WAGES — THIRD QUARTER AFTER EXIT QUARTER"  
q4aeEmpStatus="RSA EMPLOYMENT - FOURTH QUARTER AFTER EXIT QUARTER"  
q4aeEmpMatchType="RSA TYPE OF EMPLOYMENT MATCH — FOURTH QUARTER AFTER  
EXIT QUARTER"  
q4aeWages="RSA WAGES — FOURTH QUARTER AFTER EXIT QUARTER"  
q4aeRetentionQ2AndQ4="RSA RETENTION WITH THE SAME EMPLOYER IN THE SECOND  
QUARTER AND THE FOURTH QUARTER — FOURTH QUARTER AFTER EXIT QUARTER"  
appOtherSupport="RSA SUPPORT: OTHER PUBLIC SUPPORT AT APPLICATION"  
ImportAgency="RSA IMPORT AGENCY"  
Report\_Through="RSA REPORT THROUGH"  
SSN="SOCIAL SECURITY NUMBER"

CompDisenrollMSG="RSA DATE COMPLETED DURING PROGRAM PARTICIPATION IN AN  
EDUCATION OR TRAINING PROGRAM LEADING TO A RECOGNIZED POSTSECONDARY  
CREDENTIAL OR EMPLOYMENT"  
EnrolledInSecEquiv="RSA ENROLLED IN A RECOGNIZED SECONDARY SCHOOL  
EQUIVALENCY PROGRAM"  
IPEExtensionDate="RSA IPE DEVELOPMENT EXTENSION"  
IPEInitialDate="RSA DATE OF INITIAL IPE"  
ZIPCode="RSA ZIP CODE"  
appMedInsCov="RSA MEDICAL INSURANCE COVERAGE AT APPLICATION"  
appMonthlyPubSup="RSA MONTHLY PUBLIC SUPPORT AT APPLICATION"  
exitMedInsCov="RSA MEDICAL INSURANCE COVERAGE AT EXIT"  
exitMonthlyPubSup="RSA MONTHLY PUBLIC SUPPORT AT EXIT"  
jpaCompServiceProviderType="RSA RSA JOB PLACEMENT ASSISTANCE: COMPARABLE  
SERVICES AND BENEFITS PROVIDER TYPE"  
wblevCompServiceProvider="RSA WORK BASED LEARNING EXPERIENCE, SERVICE  
PROVIDED BY COMPARABLE SERVICES AND BENEFITS PROVIDERS"  
wblevCompServiceProviderType="RSA WORK BASED LEARNING EXPERIENCE,  
COMPARABLE SERVICES AND BENEFITS PROVIDER TYPE"  
wblevPurchaseProviderType="RSA WORK BASED LEARNING EXPERIENCE,  
PURCHASED SERVICE PROVIDER TYPE"  
wblevVRAgencyPurchase="RSA WORK BASED LEARNING EXPERIENCE, SERVICE  
PROVIDED THROUGH VR AGENCY PURCHASE"  
wblevVRAgencyStaff="RSA WORK BASED LEARNING EXPERIENCE, SERVICE  
PROVIDED BY VR AGENCY STAFF (IN-HOUSE)"  
wblevVRTitleIFundsExpended="RSA WORK BASED LEARNING EXPERIENCE,

Appendix A.197

JCL/SAS Code: RSALABEL

AMOUNT OF VR FUNDS EXPENDED FOR SERVICE (TITLE I)"

;

## Appendix A.198

JCL/SAS Code: RSASSNLT

```

//#3590SSN JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: RSA-DAF MATCHED EXTRACT FOR DOE
//* PROJECT DIRECTOR:  MATT URATO
//* PROGRAM:  OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(RSASSNLT)
//* DESCRIPTION: CREATES LIST OF ALL DAF21 SSNS TO BE PINNED
//*****
//*
//*****
//* DAF DEMO EXTRACT
//*****
//DEMO EXEC SAS9,
//  WORK='120000,60000'
//*
//DEMO DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.DMG,DISP=SHR
//RSA DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAEVS.SA.V1,DISP=SHR
//TKTBS DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKETBS,DISP=SHR
//VRRMS DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.VRRMS,DISP=SHR
//VRRMSH DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.VRRMS.HIST,DISP=SHR
//ENPAY DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.ENPAY,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SSNLIST.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP3 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP4 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP5 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP6 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

DATA TEMP1.DMG;
  SET DEMO.DMG (KEEP=SSN);
  BY SSN;
RUN;

```

Appendix A.198

JCL/SAS Code: RSASSNLT

```
PROC SORT DATA=RSA.RSAEVS (KEEP=SSN) NODUPKEY
  OUT=TEMP2.RSA ;
  BY SSN;
RUN;

PROC SORT DATA=TKTBS.TICKETBS (KEEP=SSN) NODUPKEY
  OUT=TEMP3.TKTBS;
  BY SSN;
RUN;

PROC SORT DATA=VRRMS.VRRMS (KEEP=SSN) NODUPKEY
  OUT=TEMP4.VRRMS ;
  BY SSN;
RUN;

PROC SORT DATA=VRRMSH.VRRMS (KEEP=SSN) NODUPKEY
  OUT=TEMP5.VRRMSH ;
  BY SSN;
RUN;

PROC SORT DATA=ENPAY.ENPAY_FINAL (KEEP=SSN) NODUPKEY
  OUT=TEMP6.ENPAY ;
  BY SSN;
RUN;

* MERGE FILES BY SSN KEEPING ANY SSN THAT APPEARS IN ANY FILES;
DATA OUT.SSNLIST;
  MERGE TEMP1.DMG
        TEMP2.RSA
        TEMP3.TKTBS
        TEMP4.VRRMS
        TEMP5.VRRMSH
        TEMP6.ENPAY
  ;
  BY SSN;
RUN;

PROC CONTENTS DATA=OUT.SSNLIST; RUN;

PROC SORT DATA=OUT.SSNLIST NODUPKEY;
BY SSN;
RUN;
```

## Appendix A.199

JCL/SAS Code: JCLPIN

```
//$7400PIN JOB (12510000,T715,,SAS,,ITC9FL),PAUL-OLEARY,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$7400
//*****
//* PROJECT: RSA-DAF MATCHED EXTRACT FOR DOE
//* PROJECT NUMBER: 50214
//* PROJECT DIRECTOR: MATT URATO
//* PROGRAM:  OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(JCLPIN)
//* DESCRIPTION: CREATES ALL THE RSA-DAF MATCHED EXTRACTS WITH PIN
//* DATE:    03/30/23 SVETLANA BRONNIKOV
//*****
//*
//*****
//*NEW PIN CREATION
//*****
//PIN  EXEC SAS9,
//  WORK='120000,60000',
//  PARM='SYSPARM="2021"'
//*
//PREVPIN DD DSN=OPDR.TG.PRD.RTWR.DAF20P.ALLPINS.FORTRF20,DISP=SHR
//TRFSSNS DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SSNLIST.SA.V1,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,
//  DISP=(MOD,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP5 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//TEMP6 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(CRPIN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* DAF DEMO EXTRACT
//*****
//DEMO  EXEC SAS9,
//  WORK='120000,60000',
//  PARM='SYSPARM="2021DMG"'
```

## Appendix A.199

JCL/SAS Code: JCLPIN

```
//*
//PIN DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.DMG,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21.DMG.PIN,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
// DISP=(SHR,PASS,KEEP)
//*
//*****
//*DAF ANNUALS EXTRACT
//*****
//ANNPROC PROC
//ANN EXEC SAS9,
// WORK='120000,60000',
// PARM='SYSPARM="2021Y&YR"'
//*
//PIN DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.Y&YR.,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21.Y&YR..PIN,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
// DISP=(SHR,PASS,KEEP)
// PEND
//*
//ANNRUN21 EXEC ANNPROC,YR=2021
//ANNRUN20 EXEC ANNPROC,YR=2020
//ANNRUN19 EXEC ANNPROC,YR=2019
//ANNRUN18 EXEC ANNPROC,YR=2018
//ANNRUN17 EXEC ANNPROC,YR=2017
//ANNRUN16 EXEC ANNPROC,YR=2016
//ANNRUN15 EXEC ANNPROC,YR=2015
//ANNRUN14 EXEC ANNPROC,YR=2014
//ANNRUN13 EXEC ANNPROC,YR=2013
//ANNRUN12 EXEC ANNPROC,YR=2012
//ANNRUN11 EXEC ANNPROC,YR=2011
//ANNRUN10 EXEC ANNPROC,YR=2010
//ANNRUN09 EXEC ANNPROC,YR=2009
//ANNRUN08 EXEC ANNPROC,YR=2008
//ANNRUN07 EXEC ANNPROC,YR=2007
```

## Appendix A.199

JCL/SAS Code: JCLPIN

```
//ANNRUN06 EXEC ANNPROC,YR=2006
//ANNRUN05 EXEC ANNPROC,YR=2005
//ANNRUN04 EXEC ANNPROC,YR=2004
//ANNRUN03 EXEC ANNPROC,YR=2003
//ANNRUN02 EXEC ANNPROC,YR=2002
//ANNRUN01 EXEC ANNPROC,YR=2001
//ANNRUN00 EXEC ANNPROC,YR=2000
//ANNRUN99 EXEC ANNPROC,YR=1999
//ANNRUN98 EXEC ANNPROC,YR=1998
//ANNRUN97 EXEC ANNPROC,YR=1997
//ANNRUN96 EXEC ANNPROC,YR=1996
//ANNRUN95 EXEC ANNPROC,YR=1995
//ANNRUN94 EXEC ANNPROC,YR=1994
//*
//*****
//* DAF ANNUAL LAUS/SAIPE EXTRACT
//*****
//LAUPROC PROC
//LAU EXEC SAS9,
// WORK='120000,60000',
// PARM='SYSPARM="2021L&YR"'
//*
//PIN DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.L&YR.,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21.L&YR..PIN,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
// DISP=(SHR,PASS,KEEP)
// PEND
//*
//LAURUN21 EXEC LAUPROC,YR=2021
//LAURUN20 EXEC LAUPROC,YR=2020
//LAURUN19 EXEC LAUPROC,YR=2019
//LAURUN18 EXEC LAUPROC,YR=2018
//LAURUN17 EXEC LAUPROC,YR=2017
//LAURUN16 EXEC LAUPROC,YR=2016
//LAURUN15 EXEC LAUPROC,YR=2015
//LAURUN14 EXEC LAUPROC,YR=2014
//LAURUN13 EXEC LAUPROC,YR=2013
//LAURUN12 EXEC LAUPROC,YR=2012
//LAURUN11 EXEC LAUPROC,YR=2011
//LAURUN10 EXEC LAUPROC,YR=2010
```



## Appendix A.199

JCL/SAS Code: JCLPIN

```
//LAURUN09 EXEC LAUPROC,YR=2009
//LAURUN08 EXEC LAUPROC,YR=2008
//LAURUN07 EXEC LAUPROC,YR=2007
//LAURUN06 EXEC LAUPROC,YR=2006
//LAURUN05 EXEC LAUPROC,YR=2005
//LAURUN04 EXEC LAUPROC,YR=2004
//LAURUN03 EXEC LAUPROC,YR=2003
//LAURUN02 EXEC LAUPROC,YR=2002
//LAURUN01 EXEC LAUPROC,YR=2001
//LAURUN00 EXEC LAUPROC,YR=2000
//LAURUN99 EXEC LAUPROC,YR=1999
//LAURUN98 EXEC LAUPROC,YR=1998
//LAURUN97 EXEC LAUPROC,YR=1997
//LAURUN96 EXEC LAUPROC,YR=1996
//LAURUN95 EXEC LAUPROC,YR=1995
//LAURUN94 EXEC LAUPROC,YR=1994
//*
//*****
//* DAF TKT BASE EXTRACT
//*****
//TKTBSE EXEC SAS9,
//   WORK='120000,60000',
//   PARM='SYSPARM="2021TKTBASE"'
//*
//PIN DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.TKTBS,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21.TKTBASE.PIN,
//   DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//   SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//   SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
//   DISP=(SHR,PASS,KEEP)
//*
//*****
//* DAF TKT ANNUALS EXTRACT
//*****
//TKTPROC PROC
//TKTANN EXEC SAS9,
//   WORK='120000,60000',
//   PARM='SYSPARM="2021TKT20&YR"'
//*
//PIN DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.TKT&YR.,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21.TKT&YR..PIN,
```

## Appendix A.199

JCL/SAS Code: JCLPIN

```
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
// DISP=(SHR,PASS,KEEP)
// PEND
//*
//TKTRUN21 EXEC TKTPROC,YR=21
//TKTRUN20 EXEC TKTPROC,YR=20
//TKTRUN19 EXEC TKTPROC,YR=19
//TKTRUN18 EXEC TKTPROC,YR=18
//TKTRUN17 EXEC TKTPROC,YR=17
//TKTRUN16 EXEC TKTPROC,YR=16
//TKTRUN15 EXEC TKTPROC,YR=15
//TKTRUN14 EXEC TKTPROC,YR=14
//TKTRUN13 EXEC TKTPROC,YR=13
//TKTRUN12 EXEC TKTPROC,YR=12
//TKTRUN11 EXEC TKTPROC,YR=11
//TKTRUN10 EXEC TKTPROC,YR=10
//TKTRUN09 EXEC TKTPROC,YR=09
//TKTRUN08 EXEC TKTPROC,YR=08
//TKTRUN07 EXEC TKTPROC,YR=07
//TKTRUN06 EXEC TKTPROC,YR=06
//TKTRUN05 EXEC TKTPROC,YR=05
//TKTRUN04 EXEC TKTPROC,YR=04
//TKTRUN03 EXEC TKTPROC,YR=03
//TKTRUN02 EXEC TKTPROC,YR=02
//*
//*****
//* RSA EXTRACT FILES
//*****
//RSAPROC1 PROC
//RSA1 EXEC SAS9,
// WORK='120000,60000',
// PARM='SYSPARM="2021RSA&YR"'
//*
//PIN DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.RSA&YR.,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21.RSA&YR..PIN,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
```

## Appendix A.199

JCL/SAS Code: JCLPIN

```
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//RSARUN98 EXEC RSAPROC1,YR=98
//RSARUN99 EXEC RSAPROC1,YR=99
//RSARUN00 EXEC RSAPROC1,YR=00
//RSARUN01 EXEC RSAPROC1,YR=01
//RSARUN02 EXEC RSAPROC1,YR=02
//RSARUN03 EXEC RSAPROC1,YR=03
//RSARUN04 EXEC RSAPROC1,YR=04
//RSARUN05 EXEC RSAPROC1,YR=05
//RSARUN06 EXEC RSAPROC1,YR=06
//RSARUN07 EXEC RSAPROC1,YR=07
//RSARUN08 EXEC RSAPROC1,YR=08
//RSARUN09 EXEC RSAPROC1,YR=09
//RSARUN10 EXEC RSAPROC1,YR=10
//RSARUN11 EXEC RSAPROC1,YR=11
//RSARUN12 EXEC RSAPROC1,YR=12
//RSARUN13 EXEC RSAPROC1,YR=13
//RSARUN14 EXEC RSAPROC1,YR=14
//RSARUN15 EXEC RSAPROC1,YR=15
//RSARUN16 EXEC RSAPROC1,YR=16
//RSARUN17 EXEC RSAPROC1,YR=17
//RSARUN18 EXEC RSAPROC1,YR=18
//RSARUN19 EXEC RSAPROC1,YR=19
//RSARUN20 EXEC RSAPROC1,YR=20
//RSARUN21 EXEC RSAPROC1,YR=21
//*
//*****
//* RSA LINKABLE FILES
//*****
//RSALINK EXEC SAS9,
//      WORK='120000,60000',
//      PARM='SYSPARM="2021RSAFLAT"'
//*
//PIN  DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN   DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.RSAFLAT,DISP=SHR
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//OUT  DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21P.RSAFLAT.PIN,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
//      DISP=(SHR,PASS,KEEP)
//*
```

## Appendix A.199

JCL/SAS Code: JCLPIN

```

//RSAOLD EXEC SAS9,
//  WORK='120000,60000',
//  PARM='SYSPARM="2021RSAFLAT"'
//*
//PIN  DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN   DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.RSAFLAOF,DISP=SHR
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//OUT  DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21P.RSAFLAOF.PIN,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* VRRMS EXTRACT
//*****
//VRMS EXEC SAS9,
//  WORK='120000,60000',
//  PARM='SYSPARM="2021VRRMS"'
//*
//PIN  DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN   DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.VRRMS,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21.VRRMS.PIN,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* VRRMS EXTRACT HISTORICAL
//*****
//VRMS EXEC SAS9,
//  WORK='120000,60000',
//  PARM='SYSPARM="2021VRRMS"'
//*
//PIN  DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN   DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.VRRMS.HIST,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21.VRRMS.HIST.PIN,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)

```

## Appendix A.199

JCL/SAS Code: JCLPIN

```
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
// DISP=(SHR,PASS,KEEP)
//*
//*****
//* ENPAY HORIZONTAL EXTRACT
//*****
//ENPA EXEC SAS9,
// WORK='120000,60000',
// PARM='SYSPARM="2021ENPAY"'
//*
//PIN DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.ENPAY,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21.ENPAY.PIN,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
// DISP=(SHR,PASS,KEEP)
//*
//*****
//* ENPAY VERTICAL EXTRACT
//*****
//ENPV EXEC SAS9,
// WORK='120000,60000',
// PARM='SYSPARM="2021ENPAYVER"'
//*
//PIN DD DSN=OPDR.TG.PRD.RTWR.DAF21P.ALLPINS.FORTRF21,DISP=SHR
//IN DD DSN=OPDR.TG.PRD.ETTW.FINAL.RSA.DAF21.ENPAV,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.RSAVER.DAF21.ENPAV.PIN,
// DISP=(MOD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(APPIN),
// DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
```

Appendix A.199

JCL/SAS Code: JCLPIN

```
HELO MVSHOST
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
DATA
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>
TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 RSA JCLPIN IS COMPLETE
```

## Appendix A.200

JCL/SAS Code: CRPIN

```
OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN OBS=MAX;

%MACRO DOIT;

* SYSPARM MACRO VARIABLE COMES IN FROM THE JCL;

* SET MACRO VARIABLE TO 2 DIGIT YEAR;
%LET YR=%SUBSTR(&SYSPARM.,3,2);
%PUT YR=&YR.;

* SET MACRO VARIABLE TO PRIOR YEAR;
%LET PYEAR=%EVAL(&SYSPARM.-1);
%PUT PYEAR=&PYEAR.;

* SET MACRO VARIABLE TO 2 DIGIT PRIOR YEAR;
%LET PYR=%SUBSTR(&PYEAR.,3,2);
%PUT PYR=&PYR.;

***** GET THE HIGHEST PIN NUMBER FOR PREVIOUS PIN FILE *****;
PROC SORT DATA=PREVPIN.ALL_TRF&PYR._DEMO_PINS OUT=TEMP1.PREVPINS;
  BY DESCENDING PIN;
RUN;

DATA _NULL_;
  SET TEMP1.PREVPINS(FIRSTOBS=1 OBS=1);
  CALL SYMPUTX('HIGHPIN',PIN);
RUN;
%PUT HIGHPIN=&HIGHPIN.;

***** DETERMINE NEW SSNS THAT NEED PINS *****;
PROC SORT DATA=PREVPIN.ALL_TRF&PYR._DEMO_PINS OUT=TEMP2.PREVPINS;
  BY SSN;
RUN;

DATA TEMP3.NEWSSENS;
  MERGE TEMP2.PREVPINS(IN=PREV)
        TRFSSNS.SSNLIST(IN=TRF KEEP=SSN)
  ;
  BY SSN;
  IF TRF AND NOT PREV;
  IF SSN="" THEN DELETE;
  IF SSN='000000000' THEN DELETE;
  IF NOTDIGIT(SSN)>0 THEN DELETE;
```

## Appendix A.200

JCL/SAS Code: CRPIN

```
* GENERATE RANDOM NUMBER FOR SORTING;
RAND_NUM=RANUNI(0);

IF MISSING(SSN) THEN MISSSSN=1;
ELSE MISSSSN=0;
IF MISSING(PIN) THEN MISSPIN=1;
ELSE MISSPIN=0;
RUN;

TITLE1 'FREQ OF MISSING PINS & SSNS FROM NEW LIST OF SSNS TO PIN';
TITLE2 'NO SSNS SHOULD BE MISSING';
TITLE3 'ALL PINS SHOULD BE MISSING';
PROC FREQ DATA=TEMP3.NEWSSNS;
    TABLE MISSSSN / MISSING;
    TABLE MISSPIN / MISSING;
RUN;

***** ASSIGN NEW PINS *****;
* SORT BY RANDOM NUMBER;
PROC SORT DATA=TEMP3.NEWSSNS OUT=TEMP4.NEWSSNS;
    BY RAND_NUM;
RUN;

* ADD THE OBSERVATION NUMBER TO THE PREVIOUS HIGHEST PIN TO CREATE PIN
FOR NEW SSN;
DATA TEMP5.NEWPINS;
    SET TEMP4.NEWSSNS;
    NEWPIN=_n_+&HIGHPIN.;
    PIN=PUT(NEWPIN, 8.);

IF MISSING(SSN) THEN MISSSSN=1;
ELSE MISSSSN=0;
IF MISSING(PIN) THEN MISSPIN=1;
ELSE MISSPIN=0;

DROP NEWPIN
    RAND_NUM
    ;

RUN;

TITLE1 'FREQ OF MISSING PINS & SSNS AFTER ASSIGNING PINS TO NEW SSNS';
TITLE2 'NO SSNS SHOULD BE MISSING';
TITLE3 'NO PINS SHOULD BE MISSING';
```



Appendix A.200

JCL/SAS Code: CRPIN

```
PROC FREQ DATA=TEMP5.NEWPINS;
  TABLE MISSSSN / MISSING;
  TABLE MISSPIN / MISSING;
RUN;

***** ADD NEW PINS TO PREVIOUS PINS *****;
PROC SORT DATA=TEMP5.NEWPINS OUT=TEMP6.NEWPINS;
  BY SSN;
RUN;

DATA OUT.ALL_TRF&YR._DEMO_PINS;
  SET TEMP2.PREVPINS(IN=A)
    TEMP6.NEWPINS(IN=B)
  ;
  BY SSN;

  IF MISSING(SSN) THEN MISSSSN=1;
  ELSE MISSSSN=0;
  IF MISSING(PIN) THEN MISSPIN=1;
  ELSE MISSPIN=0;
RUN;

TITLE1 'MISSING PINS & SSNS AFTER COMBINING OLD PINS WITH NEW PINS';
TITLE2 'NO SSNS SHOULD BE MISSING';
TITLE3 'NO PINS SHOULD BE MISSING';
PROC FREQ DATA=OUT.ALL_TRF&YR._DEMO_PINS;
  TABLE MISSSSN / MISSING;
  TABLE MISSPIN / MISSING;
RUN;

%MEND DOIT;
%DOIT;
```

Appendix A.201

JCL/SAS Code: APPIN

```
* EXECUTED BY OPDR.TG.PRD.ETW.#3590.DAF21.RSADAF.PRDLIB(JCLPIN);
* THIS PROGRAM ADDS THE PINS TO DAF DATA FOR VERIFIED RSA RECORDS;
```

```
OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;
```

```
%LET YEAR=%SUBSTR(&SYSPARM.,1,4);
%LET YR=%SUBSTR(&SYSPARM.,3,2);
%LET MEM=%SUBSTR(&SYSPARM.,5);
```

```
%MACRO DOIT;
DATA TEMP.&MEM.(DROP=SSN);
  MERGE IN.&MEM.(IN=A )
        PIN.ALL_TRF&YR._DEMO_PINS(IN=P)
        ;
  BY SSN;
  IF A;
  IF P THEN INPIN=1;
  ELSE INPIN=0;
RUN;
```

```
PROC SORT DATA = TEMP.&MEM. OUT = OUT.&MEM.;
BY PIN;
RUN;
```

```
TITLE1 'AFTER PINNING DATA';
TITLE2 'ALL OBSERVATION SHOULD BE IN THE PIN FILE';
PROC FREQ DATA=OUT.&MEM.;
  TABLE INPIN / MISSING;
RUN;
```

```
TITLE1;
TITLE2;
PROC CONTENTS DATA=OUT.&MEM.;
RUN;
%MEND DOIT;
%DOIT;
ENDSAS;
```

## Appendix A.202

JCL/SAS Code: MEFFNDR

```

//#3590MEF JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: BUILD DAF21
//* PROJECT
//* DIRECTOR:  MATT URATO
//*
//* PROGRAM:   OPDR.TG.PRD.ETTW.#3590.DAF21.RSADAF.PRDLIB(MEFFNDR)
//*
//* DESCRIPTION: MERGES VERIFIED RSA RECORDS WITH DAF MASTER FINDER
//*              TO BUILD THE MEF FINDER FILE
//* DATE:       3/30/23 SVETLANA BRONNIKOV
//*****
//*
//RSA   EXEC SAS9,
//      WORK='120000,60000'
//*
//* INPUT FILE 1
//* EVS-VERIFIED RSA FILE
//IN1   DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.RSAEVS.RSA.V1,DISP=SHR
//* INPUT FILE 2
//* DAF21 MASTER FINDER FILE
//IN2   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.MASTFIND.SA.V1,DISP=SHR
//*
//* OUTPUT FILE - MEF FINDER FILE WITH ONLY SSN
//FIN   DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.MEFFIND.FL.V1,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//*
//SYSIN DD *

*KEEP ONLY EVS-VERIFIED SSNS;
DATA TEMP1.RSAEVS_VERIFIED;
  SET IN1.MPR_EVS4 (KEEP = SSN VER_CODE
  WHERE=(VER_CODE = 'Y'));
  BY SSN;
RUN;

PROC FREQ DATA = TEMP1.RSAEVS_VERIFIED;
TABLES VER_CODE / LIST MISSING;

```

Appendix A.202

JCL/SAS Code: MEFFNDR

RUN;

\*CREATING SAS FINDER FILE FOR MEF;

DATA TEMP2.MEF\_FINDER;

MERGE TEMP1.RSAEVS\_VERIFIED (KEEP = SSN IN=INA)

IN2.MASTFIND (KEEP = SSN IN=INB);

;

BY SSN;

\*KEEP ANY SSN THAT SHOWS UP IN EITHER INPUT FILES;

IF INA OR INB;

\*WE ONLY NEED ONE RECORD FOR EACH DISTINCT SSN;

IF LAST.SSN;

FILE FIN;

PUT @001 SSN \$CHAR9.;

RUN;

PROC CONTENTS DATA=TEMP2.MEF\_FINDER; RUN;

## Appendix A.203

JCL/SAS Code: LAUS\_FORMAT

```
/******  
PROGRAM: LAUS FORMATS  
AUTHOR: PATSY LLOYD  
PURPOSE: TO CREATE LAUS FORMATS  
DATE: 10/25/2013  
UPDATED: 12/12/2022 SL -- FOR DAF21  
*****/  
%let todaydt = %sysfunc(today());  
%let todaydt = %sysfunc(putn(&todaydt,yymmdd10.));  
%put &todaydt;  
  
options obs=MAX;  
  
%let root=M:\DAF21\TASK 18 Create LAUS-SAIPE Formats;  
  
proc printto  
  log = "&root.\Programs\LAUS_FORMAT_&todaydt..log"  
  new  
  print = "&root.\Programs\LAUS_FORMAT_&todaydt..lst";  
run;  
  
%LET RAWDIR=&root.\Raw Data;  
  
LIBNAME OUTPUT "&root.\Output";  
  
%LET BEGYR=1994;  
%LET ENDYR=2021;  
  
* FROM DP: Read the LAUS Area code variable from the text file  
M:\DAF16\Task 18 - LAUS-SAIPE Processing\Raw Data\laucntycur14.txt into SAS;  
  
PROC IMPORT OUT= OUTPUT.LAUCNTYCUR14  
  DATAFILE= "&RAWDIR.\laucntycur14.txt"  
  DBMS=DLM REPLACE;  
  DELIMITER='7C'x;  
  GETNAMES=YES;  
  DATAROW=7; /* Brittany Colip -- changed to 7 instead of 8 in DAF17 */  
  /**/  
  GUESSINGROWS=45000;  
RUN;  
  
DATA OUTPUT.LAUCNTYCUR14;  
SET OUTPUT.LAUCNTYCUR14;
```

## Appendix A.203

JCL/SAS Code: LAUS\_FORMAT

```
RENAME Labor_force_data_by_county__not=LAC;
LABEL Labor_force_data_by_county__not="LAUS AREA CODE";

*DELETE IF STATE FIPS CODE IS MISSING;
*IF VAR2=. THEN DELETE;*UPDATED TO WHERE;
WHERE VAR2^=.;

FIPS=TRIM(PUT(VAR2,Z2.)||PUT(VAR3,Z3.));

*ONLY KEEP COUNTY LEVEL RECORDS;

*IF SUBSTR(Labor_force_data_by_county__not,1,2) in ("CN","PA");

*ONLY NEED TO KEEP THE LAC VARIABLE;

KEEP Labor_force_data_by_county__not FIPS;
RUN;

*FROM DP: Sort the resulting SAS file from Step1 by the LAUS Area code;
PROC SORT DATA=OUTPUT.LAUCNTYCUR14 NODUPKEY; BY LAC;RUN;

*FROM DP: SAS load the monthly county employment rate files
M:\DAF13\Task 20 - LAUS - SAIPE\LAUS\Raw Data\la.data.0.CurrentUyy-yy
We only need unemployment rates from Jan 1994 through Dec 2013. ;

*CHANGED TO KEYWORD PARAMETERS IN DAF13;
%MACRO IMPORT(INPUT=,OUTPUT=);

PROC IMPORT OUT= OUTPUT.&OUTPUT.
    DATAFILE= "&INPUT"
    DBMS=TAB REPLACE;
    GETNAMES=YES;
    GUESSINGROWS=MAX;
RUN;

DATA OUTPUT.&OUTPUT.;
    SET OUTPUT.&OUTPUT.;
    LENGTH LAC $ 15;
    LAC=SUBSTR(SERIES_ID,4);
    WHERE SUBSTR(SERIES_ID,LENGTH(SERIES_ID)-1,2) = "03" AND /*GETTING IDS ENDING WITH 03*/
        PERIOD NE "M13" AND /*M13 SPECIFIES THE AVERAGE, WHICH WE DO
            NOT NEED*/
        1994<=YEAR<=&ENDYR; /*WE ONLY NEED UNEMPLOYMENT RATES BETWEEN
            JAN 1994 - DEC of the current DAF year*/
RUN;
```

## Appendix A.203

JCL/SAS Code: LAUS\_FORMAT

```
PROC SORT DATA=OUTPUT.&OUTPUT.; BY LAC;RUN;
%MEND;
%IMPORT(INPUT=&RAWDIR.\la.data.0.CurrentU90-94.txt, OUTPUT=MER_90_94);
%IMPORT(INPUT=&RAWDIR.\la.data.0.CurrentU95-99.txt, OUTPUT=MER_95_99);
%IMPORT(INPUT=&RAWDIR.\la.data.0.CurrentU00-04.txt, OUTPUT=MER_00_04);
%IMPORT(INPUT=&RAWDIR.\la.data.0.CurrentU05-09.txt, OUTPUT=MER_05_09);
%IMPORT(INPUT=&RAWDIR.\la.data.0.CurrentU10-14.txt, OUTPUT=MER_10_14);
%IMPORT(INPUT=&RAWDIR.\la.data.0.CurrentU15-19.txt, OUTPUT=MER_15_19);
%IMPORT(INPUT=&RAWDIR.\la.data.0.CurrentU20-24.txt, OUTPUT=MER_20_24);
```

```
/*THE VARIABLE "VALUE" WAS IMPORTED AS A CHARACTER FIELD, SO
 I AM CONVERTING TO NUMERIC SINCE IT THIS IS THE FORMAT OF THIS
 VARIABLE IN THE OTHER DATASETS*/
```

```
DATA OUTPUT.MER_05_09;
  SET OUTPUT.MER_05_09;
  IF _____value="-" THEN _____value=.;
  _____value1=_____value+0;
  DROP _____value;
  RENAME _____value1=_____value;
  LABEL _____value1=_____value;
RUN;
```

```
DATA OUTPUT.MER_20_24;
  SET OUTPUT.MER_20_24;
  IF _____value="-" THEN _____value=.;
  _____value1=_____value+0;
  DROP _____value;
  RENAME _____value1=_____value;
  LABEL _____value1=_____value;
RUN;
```

```
*MERGE COUNTY LEVEL LACs FILE WITH MONTHLY EMPLOYMENT RATES;
```

```
DATA STEP1;
  SET
    OUTPUT.MER_00_04
    OUTPUT.MER_05_09
    OUTPUT.MER_10_14
    OUTPUT.MER_90_94
    OUTPUT.MER_95_99
      OUTPUT.MER_15_19
      OUTPUT.MER_20_24
  ;
  BY LAC;
```

## Appendix A.203

JCL/SAS Code: LAUS\_FORMAT

```
RUN;
```

```
PROC MEANS DATA=STEP1;  
CLASS YEAR;  
VAR _____value;  
RUN;
```

```
DATA OUTPUT.LAUS;  
MERGE OUTPUT.LAUCNTYCUR14 (IN=A)  
STEP1 (rename=(_____value=VALUE));
```

```
BY LAC;  
IF A;
```

```
* CREATE A FIPS/YR/MO VARIABLE;
```

```
FIPSYRMO=TRIM(FIPS| |SUBSTR(PUT(YEAR,$4.),3,2)| |SUBSTR(PERIOD,2,2));
```

```
* IN ORDER TO CREATE THE FORMAT THE VARIABLE CONTAINING THE VALUE  
TO BE FORMATTED MUST BE NAMED "START" AND THE VARIABLE CONTAINING  
THE FORMATTED VALUE MUST BE NAME "LABEL";
```

```
START=FIPSYRMO;  
LABEL=VALUE;
```

```
* A VARIABLE NAMED FMTNAME MUST BE CREATED TO STORE THE NAME OF THE  
FORMAT --- THE NAME USED TO REFERENCE THE FORMAT;  
FMTNAME='$LAUS';
```

```
OUTPUT;
```

```
RUN;
```

```
* CREATE THE FORMAT FROM THE FILE CREATED IN PREVIOUS DATASTEP;  
PROC FORMAT CNTLIN=OUTPUT.LAUS LIBRARY=OUTPUT ; RUN;
```

```
* CHECK THAT THE FORMAT IS WORKING CORRECTLY BY PRINTING THE FORMATED  
VALUE OF "START" FROM THE CNTLIN FILE AND THE VALUE OF "LABEL". IF  
THE FORMAT CREATION WAS SUCCESSFULL THESE VALUES WILL BE THE SAME;
```

```
OPTIONS FMTSEARCH = (OUTPUT);  
PROC PRINT DATA=OUTPUT.LAUS (OBS=100);  
VAR FIPSYRMO START LABEL;  
FORMAT START $LAUS.;
```



Appendix A.203

JCL/SAS Code: LAUS\_FORMAT

```
RUN;
```

Appendix A.204

JCL/SAS Code: Upload LAUS Data

```
*****
* PROJECT: Evaluation of the Ticket to Work Program-Part A
*   Evaluation Implementation (6979) TRF12
* PROJECT
* DIRECTOR:  LESLEY HILDEBRAND
*
* PROGRAM:   M:\DAF12\TKT\Upload LAUS SAIPE Data.sas
*
*
* DESCRIPTION: UPLOADS LAUSE SAIPE DATA TO CREATE FORMATS
*
* DATE:      01/13/2016 JOE MASTRIANNI
* UPDATED:   12/15/2022 ALEX BRYCE FOR DAF18
***** ;
options comamid=tcp; %let SY56=SY56.sspf.ssa.gov; %let tcpsec=_prompt_; options remote=sy56.spawner; run;
libname local 'M:\DAF21\TASK 18 Create LAUS-SAIPE Formats\Output';

SIGNON ;

rsubmit;

%LET ENDYR=2021; *UPDATE ACCORDINGLY;
TITLE1 "DAF21 LAUS FORMATS"; *UPDATE ACCORDINGLY;

libname LAUS 'OPDR.TG.PRD.ETTW.#6266.DAF21P.FRMTLAUS.SA.V1';

option compress=binary;

proc upload data=local.LAUS OUT=LAUS.FRMTLAUS (compress=binary);RUN;

endrsubmit;
signoff;
```

## Appendix A.205

JCL/SAS Code: LAUS

```

//#6266FOR JOB (12510000,T715,,SAS,,ITC9TH),LUKASHANETS,CLASS=T,
// MSGCLASS=1,MSGLEVEL=(1,1),REGION=4096K,NOTIFY=#6266
/*SSAMAIN SASBASIC
//*****
/* PROJECT: DAF
/*
/* PROJECT
/* DIRECTOR:  MATT URATO
/*
/* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.LAUS.PRDLIB(LAUS)
/*
/* DESCRIPTION: CREATE FORMAT LIBRARY FOR LAUS AND SAIPE
/* UPDATED 12/15/2022 FOR DAF21
/*
/* EMAIL:   ABRYCE@MATHEMATICA-MPR.COM
//*****
/*
//JS010 EXEC SAS9,
//  WORK='120000,60000'
/*
//LAUS DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.FRMTLAUS.SA.V1,DISP=SHR
//LIBLASA DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.LAUS.FMTLIB,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(4000,4000),RLSE)
//SYSIN DD *
OPTIONS NOCENTER FMTSEARCH = (LIBLASA) MPRINT SYMBOLGEN;

%LET ENDYR=2021; *UPDATE ACCORDINGLY;
TITLE1 "DAF21 LAUS/SAIPE FORMATS"; *UPDATE ACCORDINGLY;
%MACRO READYRS();
%DO YEAR=1994 %TO &ENDYR. ;

    /*KEEP ONLY LAST TWO DIGITS OF THE YEAR*/
    %LET YRS=%SUBSTR(&YEAR,3,2);

DATA LAUS&YRS;
  SET LAUS.FRMTLAUS (WHERE=(YEAR=&YEAR.) DROP=FMTNAME);
  LENGTH FMTNAME $20;
  FMTNAME="$LAUS&YRS.F";
RUN;

  /*USE CNTLIN TO ADD $LAUS&YRS.F FORMAT.
  TO THE FORMAT LIBRARY */
PROC FORMAT CNTLIN=LAUS&YRS LIBRARY=LIBLASA ; RUN;

```

Appendix A.205

JCL/SAS Code: LAUS

```
PROC PRINT DATA=LAUS&YRS. (OBS=100);  
TITLE2 "CHECK THAT FORMATS WERE CREATED CORRECTLY, YEAR: &YEAR.";  
TITLE3 "START SHOULD EQUAL LABEL";  
  VAR FIPSYRMO START LABEL;  
  FORMAT START $LAUS&YRS.F.;  
RUN;
```

```
%END;  
%MEND;  
%READYRS();
```

## Appendix A.206

JCL/SAS Code: SAIPE\_FORMAT

```

/*****
PROGRAM: SAIPE FORMATS
PURPOSE: TO CREATE SAIPE FORMATS

SPECS:

1.  Read the SAIPE files estyyall.txtLAUS in
N:\Shared-DC1\06979_TRF\Common\DAF12\Data Processing\SAIPED into SAS;
use the accompanying READMEyy files to determine positions of the FIPSCODE
and "Estimated percent of people of all ages in poverty" and "Estimate of
median household income". 2012 data is not yet available but was suppose
to be this month (October) --- we will have to add this data when it
becomes available. NOTE: there are no 1994 values and there is no 1996
file with county data; we will address the 1996 State values in the documentation.

2.  Create a 2 formats from the dataset created in Step 1 call them $SAIPEPOV
and $SAIPEINC. The value should be the 5 digit FIPS concatenated with the yymm
(like you did with the LAUS format) and the formatted value should be the poverty
rate or the median household income.

*****/
%let todaydt = %sysfunc(today());
%let todaydt = %sysfunc(putn(&todaydt,yymmdd10.));
%put &todaydt;

*%let root=M:\DAF20\TASK 18 Create LAUS-SAIPE Formats;
%let root=\\ba.ad.ssa.gov\dcrdp\RemoteShares\SAS Data\MPR\DAF21\TASK 18 Create LAUS-SAIPE Formats;

proc printto
  log = "&root.\Programs\SAIPE_FORMAT_&todaydt..log"
  new
  print = "&root.\Programs\SAIPE_FORMAT_&todaydt..lst";
run;

OPTIONS OBS=MAX; *FOR TESTING;

%LET ENDYR=2021; *UPDATE ACCORDINGLY;
%LET YY=%SUBSTR(&ENDYR,3);
%PUT DAF&ENDYR -> DAF&YY;

TITLE1 "DAF&YY SAIPE FORMATS"; *UPDATE ACCORDINGLY;

LIBNAME OUTPUT "&ROOT.\OUTPUT";

```

## Appendix A.206

JCL/SAS Code: SAIPE\_FORMAT

```
%MACRO READYRS;

%DO YEAR=1995 %TO &ENDYR. ;
/*NOTE FROM DP: there are no 1994 values and there is no 1996
file with county data;

USE %GOTO TO END THE LOOP*/

%IF &YEAR=1996 %THEN %GOTO OUT;
%LET YRS=%SUBSTR(&YEAR,3,2); *KEEP ONLY LAST TWO DIGITS OF THE YEAR;

/*READ IN DATA*/
DATA OUTPUT.EST&YRS.ALL;
  LENGTH FIPS $7.;
  KEEP FIPS_ST FIPS_CTY PCT_POV MED_INC FIPS_FMTNAME START LABEL;
  INFILE "&ROOT.\Raw Data\est&YRS.all.txt"
  FIRSTOBS=1 IRECL=3400;
  INPUT @1          FIPS_ST    2.0
         @4  FIPS_CTY    3.0
         @35 PCT_POV     4.2
         @134 MED_INC     6.0;

  FIPS=TRIM(PUT(FIPS_ST,Z2.)||PUT(FIPS_CTY,Z3.)||LEFT(PUT(&YRS.,Z2.)));
  START=FIPS;
  LABEL=PCT_POV;

/*A VARIABLE NAMED FMTNAME MUST BE CREATED TO STORE
THE NAME OF THE FORMAT --- THE NAME USED TO
REFERENCE THE FORMAT*/

  FMTNAME="$SAIPEPOV&YRS.F";
  OUTPUT;
RUN;

/*USE CNTLIN TO ADD $SAIPEPOV FORMAT. TO THE FORMAT
LIBRARY */

PROC FORMAT CNTLIN=OUTPUT.EST&YRS.ALL LIBRARY=OUTPUT ; RUN;

/*RENAME START AND LABEL TO MED_INC TO ASSOCIATE
THE FORMAT WITH THIS VARIABLE*/

DATA OUTPUT.EST&YRS.ALL;
  SET OUTPUT.EST&YRS.ALL;
  LABEL=MED_INC;
```

Appendix A.206

JCL/SAS Code: SAIPE\_FORMAT

```
FMTNAME="$SAIPEINC&YRS.F";  
OUTPUT;  
RUN;
```

```
/*USE CNTLIN TO ADD $SAIPEINC FORMAT.  
TO THE FORMAT LIBRARY */
```

```
PROC FORMAT CNTLIN=OUTPUT.EST&YRS.ALL LIBRARY=OUTPUT ; RUN;
```

```
OPTIONS FMTSEARCH = (OUTPUT);
```

```
/*PRODUCE A PRINT TO CHECK THAT THE FORMAT  
WAS CREATED CORRECTLY*/
```

```
PROC PRINT DATA=OUTPUT.EST&YRS.ALL (OBS=100);  
  TITLE2 "CHECK THAT FORMATS WERE CREATED CORRECTLY, YEAR: &YEAR."  
  TITLE3 "FIPS SHOULD EQUAL PCT_POV, AND START SHOULD EQUAL MED_INC";  
  VAR FIPS PCT_POV START MED_INC;  
  FORMAT START $SAIPEINC&YRS.F. FIPS $SAIPEPOV&YRS.F.;
```

```
RUN;
```

```
%OUT:
```

```
%END;
```

```
%MEND;
```

```
%READYRS;
```

Appendix A.207

JCL/SAS Code: Upload SAIPE Data

```

*****
* PROJECT:   DAF
* PROJECT
* DIRECTOR:  MATT URATO
* PROGRAM:   Upload SAIPE Data.sas
* DESCRIPTION: UPLOADED LAUSE SAIPE DATA TO CREATE FORMATS
***** ;
options comamid=tcp; %let SY56=SY56.sspf.ssa.gov; %let tcpsec=_prompt_; options remote=sy56.spawner; run;

signon sy56.spawner user="#5413" password="cd275123";

*LIBNAME OUT 'M:\DAF20\TASK 18 Create LAUS-SAIPE Formats\Output';
LIBNAME OUT '\\ba.ad.ssa.gov\dcrdp\RemoteShares\SAS Data\MPR\DAF21\TASK 18 Create LAUS-SAIPE
Formats\output';
SIGNON ;
*libname local 'M:\DAF20\TASK 18 Create LAUS-SAIPE Formats\Output';
libname local '\\ba.ad.ssa.gov\dcrdp\RemoteShares\SAS Data\MPR\DAF21\TASK 18 Create LAUS-SAIPE
Formats\output';
rsubmit;

%LET ENDYR=2021; *UPDATE ACCORDINGLY;
%LET YY=%SUBSTR(&ENDYR,3);
%PUT DAF&ENDYR -> DAF&YY;
TITLE1 "DAF&YY. SAIPE FORMATS"; *UPDATE ACCORDINGLY;

*Not using &yy in libname because you'll probably copy and paste the entire dsn;
libname SAIPEDAT 'OPDR.TG.PRD.ETTW.#5413.DAF21P.FRMTSAIP.SA.V1' ;

%MACRO LIBS();
option compress=binary;

%DO YEAR=1995 %TO &ENDYR. ;
/*NOTE FROM DP: there are no 1994 values and there is no 1996
file with county data;

USE %GOTO TO END THE LOOP*/

%IF &YEAR=1996 %THEN %GOTO OUT;
%LET YRS=%SUBSTR(&YEAR,3,2); *KEEP ONLY LAST TWO DIGITS OF THE YEAR;

proc upload data=local.est&YRS.all OUT=SAIPEDAT.est&YRS.all (compress=binary);RUN;
%OUT;
%END;
%MEND;
%LIBS;

```



Appendix A.207

JCL/SAS Code: Upload SAIPE Data

```
endsubmit;  
signoff;
```

## Appendix A.208

JCL/SAS Code: SAIPE

```

//#5413FRS JOB (12510000,T715,,SAS,,ITC9TH),CARRILLO-PAZ,CLASS=T,
// MSGCLASS=1,MSGLEVEL=(1,1),REGION=4096K,NOTIFY=#5413
/*SSAMAIN SASBASIC
//*****
/* PROJECT: DAF
/*
/* PROJECT
/* DIRECTOR:  MATT URATO
/*
/* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.LAUS.PRDLIB(SAIPE)
/*
/* DESCRIPTION: CREATE FORMAT LIBRARY FOR SAIPE
/* UPDATED:   12/22/2021 FOR DAF21
/* EMAIL:    CCARRILLO@MATHEMATICA-MPR.COM
//*****
/*
//JS010 EXEC SAS9,
//  WORK='120000,60000'
/*
//SAIPE DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.FRMTSAIP.SA.V1,DISP=SHR
//LIBLASA DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.SAIPE.FMTLIB,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(4000,4000),RLSE)
//SYSIN DD *
OPTIONS NOCENTER FMTSEARCH = (LIBLASA) MPRINT SYMBOLGEN;

%LET ENDYR=2021; *UPDATE ACCORDINGLY;
%LET YY=%SUBSTR(&ENDYR,3);
%PUT DAF&ENDYR -> DAF&YY;
TITLE1 "DAF&YY. SAIPE FORMATS";
%MACRO READYRS;
%DO YEAR=1995 %TO &ENDYR. ;

/*NOTE FROM DP: THERE ARE NO 1994 VALUES AND THERE IS NO 1996
FILE WITH COUNTY DATA USE %GOTO TO END THE LOOP */

%IF &YEAR=1996 %THEN %GOTO OUT;
/*KEEP ONLY LAST TWO DIGITS OF THE YEAR*/
%LET YRS=%SUBSTR(&YEAR,3,2);

/****      CREATING MED_INC FORMATS      ****/
/*RENAME START AND LABEL TO MED_INC TO ASSOCIATE
THE FORMAT WITH THIS VARIABLE*/
DATA EST&YRS.ALL;

```

## Appendix A.208

JCL/SAS Code: SAIPE

```
SET SAIPE.EST&YRS.ALL (DROP=FMTNAME);
LENGTH FMTNAME $ 20;
LABEL=MED_INC;
FMTNAME="$SAIPEINC&YRS.F";
RUN;
```

```
/*USE CNTLIN TO ADD $SAIPEINC FORMAT TO THE FORMAT LIBRARY*/
PROC FORMAT CNTLIN=EST&YRS.ALL LIBRARY=LIBLASA ; RUN;
```

```
/**      CREATING POVERTY FORMATS      ***/
DATA EST&YRS.ALL2;
SET SAIPE.EST&YRS.ALL (DROP=FMTNAME);
LENGTH FMTNAME $20;
LABEL=PCT_POV;
FMTNAME="$SAIPEPOV&YRS.F";
OUTPUT;
RUN;
```

```
/*USE CNTLIN TO ADD $SAIPEPOV FORMAT TO THE FORMAT LIBRARY*/
```

```
PROC FORMAT CNTLIN=EST&YRS.ALL2 LIBRARY=LIBLASA ; RUN;
```

```
/*PRODUCE A PRINT TO CHECK THAT THE FORMAT
WAS CREATED CORRECTLY*/
```

```
PROC PRINT DATA=SAIPE.EST&YRS.ALL(OBS=100);
TITLE2 "CHECK THAT FORMATS WERE CREATED CORRECTLY, YEAR: &YEAR.";
TITLE3 "FIPS SHOULD EQUAL PCT_POV, AND START SHOULD EQUAL MED_INC";
VAR FIPS PCT_POV START MED_INC;
FORMAT START $SAIPEINC&YRS.F. FIPS $SAIPEPOV&YRS.F.;
RUN;
```

```
%OUT:
%END;
%MEND;
%READYRS;
```

Appendix A.209

JCL/SAS Code: SSIBEN

```
//$4671SIB JOB (12510000,T715,,SAS,,ITC9FL),CARRILLO-PAZ,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT: 50214 DAF
//* PROJECT
//* DIRECTOR: MATTHEW URATO
//* PROGRAM: OPDR.TG.PRD.ETTW.$4671.DAF21.VAL.PRDLIB(SSIBEN)
//* DESCRIPTION: CREATE BENCHMARKING
//*
//* INPUT: ANNUAL FILES
//* OPDR.TG.PRD.ETTW.<ID>.DAF<YY>I.Y20<YY>.SA.V<V>
//*
//* DEMOGRAPHIC FILE
//* OPDR.TG.PRD.ETTW.<ID>.DAF<YY>I.DEMO.SA.V<V>
//*
//* OUTPUT: OPDR.TG.PRD.ETTW.<ID>.VAL<YY>P.SSIBENES.SA.V<V>
//*
//* PURPOSE: CREATE TABLES FOR BENCHMARKING
//*****
//SAS EXEC SAS9,
// WORK='120000,60000'
//*
//DEM DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.DMG,DISP=SHR
//DEM2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.DMG.SA.V1,DISP=SHR
//ANN DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2021E,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.$4671.VAL21P.SSIBENES.SA.V1,
// DISP=(NEW,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

TITLE1 "BENCHMARKING";
TITLE2 "SSI BENEFICIARIES";

%LET YR=21;
%LET DATE = '31DEC2021'D;

%MACRO GETDATA;
DATA OUT.Y20&YR.;
MERGE ANN.Y20&YR.(IN=ANNX KEEP=SSN PSTA&YR.12 STATE&YR.12
```

## Appendix A.209

JCL/SAS Code: SSIBEN

```
                PAYS&YR.12 DUES&YR.12)
DEM.DMG(IN=DEMX KEEP=SSN SEX DOBBEST BIC TOC1 DOEI DOEC ELG_RD
        MINFRSTP DOBSSR SSIELIG_FIRST
        SSIELIG_ADULT
        DXPRI_SSI DXPRIBEST)
/* STATECD_ENDYR DROPPED IN THE DMG FINALIZATION PROGRAM.
   GET THIS VARIABLE FROM THE DMG INTERMEDIATE FILE */
DEM2.DMG(KEEP=SSN STATECD_ENDYR);
BY SSN;

IF DEMX AND ANNX;

AGE&YR.=INTCK('MONTHS',DOBSSR,&DATE)/12;
AGE&YR.MPR=INTCK('MONTHS',DOBBEST,&DATE)/12;

*** FLAG SSI BENES IN CURRENT PAY STATUS AS OF DEC;
SSIBENE=0;
IF PSTA&YR.12 IN ('C01','M01','M02') THEN SSIBENE=1;
RUN;

%MEND GETDATA;
%GETDATA;

PROC FREQ DATA=OUT.Y20&YR.;
  TABLES SSIBENE / LIST MISSING;
RUN;

PROC PRINT DATA=OUT.Y20&YR.(OBS=500);
  WHERE AGE&YR. NE AGE&YR.MPR;
  VAR AGE&YR. AGE&YR.MPR DOBSSR DOBBEST;
RUN;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM>
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM>
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM>
```

Appendix A.209

JCL/SAS Code: SSIBEN

TO: <DPHELPS@MATHEMATICA-MPR.COM>  
SUBJECT: SSIBEN IS COMPLETE.

Appendix A.210

JCL/SAS Code: SSIVALT

```
// $4671SIB JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//   MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT:   50214 DAF
//* PROJECT DIRECTOR:  MATTTHEW URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.$4671.DAF21.VAL.PRDLIB(SSIVALT)
//*
//* DESCRIPTION: CREATE BENCHMARKING SSI TABLE 5, 10, AND 39
//*
//* PURPOSE: CREATE TABLES FOR BENCHMARKING
//*
//* PROGRAMMER: CHRISTIAN CARRILLO-PAZ
//*   DPHELPS@MATHEMATICA-MPR.COM
//*****
//*
//SAS   EXEC SAS9,
//   WORK='120000,60000'
//*
//IN DD DSN=OPDR.TG.PRD.ETTW.$4671.VAL21P.SSIBENES.SA.V1,
//   DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.$4671.VAL21P.SSIVALT.SA.V1,
//   DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//   SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//   SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *
```

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

```
TITLE1 "BENCHMARKING";
TITLE2 "SSI BENEFICIARIES";
TITLE3 "SSI TABLE 5, 10, & 36";
```

```
PROC FORMAT;
  VALUE AGEGRP 18-<65 = '1' .='.' OTHER='0'
  ;
```

RUN;

```
%MACRO YEAR(YR=,TYR=);
  DATA Y&YR.;
    SET IN.Y&YR.(KEEP=SEX SSIBENE AGE&TYR. STATE&TYR.12
      STATECD_ENDYR DXPRIBEST PAYS&TYR.12);
    AGEGRP=PUT(AGE&TYR.,AGEGRP.);*ALL TABLES;
```

## Appendix A.210

JCL/SAS Code: SSIVALT

```
*IF PAYS&TYR.12>0 THEN ALLPAY=1;*TABLE 5 & 36;
RUN;

PROC FREQ DATA=Y&YR;
  TABLES AGE&TYR.*AGEGRP / LIST MISSING;
RUN;

PROC SORT DATA=Y&YR.(WHERE=(SSIBENE=1))
  OUT=SSIPOP;
  BY SEX;
RUN;

*CREATING DATASETS FOR TABLE 5;
PROC FREQ DATA=SSIPOP;
  WHERE SEX IN ('M','F') AND AGEGRP='1';
  *TABLES ALLPAY / OUT=OUT.ALLP&YR.;
  TABLE SSIBENE / OUT=OUT.ALLP&YR.;
RUN;

PROC FREQ DATA=SSIPOP;
  WHERE SEX IN ('M','F') AND AGEGRP='1';
  BY SEX;
  *TABLE ALLPAY / OUT=OUT.GENP&YR.;
  TABLE SSIBENE / OUT=OUT.GENP&YR.;
RUN;

PROC MEANS DATA=SSIPOP SUM;
  WHERE SSIBENE=1 AND SEX IN ('M','F') AND AGEGRP='1';
  VAR PAYS&TYR.12;
  OUTPUT OUT=OUT.TOTPAY&TYR. SUM=SUMTOT&TYR.;
RUN;

PROC MEANS DATA=SSIPOP SUM;
  WHERE SSIBENE=1 AND SEX IN ('M','F') AND AGEGRP='1';
  VAR PAYS&TYR.12;
  BY SEX;
  OUTPUT OUT=OUT.PAYGEN&TYR. SUM=SUMGEN&TYR.;
RUN;
*END TABLE 5 DATASETS;

*CREATING DATASETS FOR TABLE 10;
PROC FREQ DATA=SSIPOP ORDER=FORMATTED;
  BY SEX;
  TABLE STATE&TYR.12*AGEGRP / OUT=OUT.ST20&TYR.;
RUN;
```



## Appendix A.210

JCL/SAS Code: SSIVALT

```
PROC FREQ DATA=SSIPOP ORDER=FORMATTED;
  BY SEX;
  TABLE STATECD_ENDYR *AGEGRP / OUT=OUT.PAYS20&TYR.;
RUN;
*END TABLE 10 DATASETS;
```

```
*CREATE DATASETS FOR TABLE 36;
PROC FREQ DATA=SSIPOP;
  WHERE /*ALLPAY=1 AND*/ SEX IN ('M','F');
  TABLES DXPRIBEST / OUT=OUT.DIAG&YR.;
RUN;
```

```
PROC FREQ DATA=SSIPOP;
  WHERE /*ALLPAY=1 AND*/ SEX IN ('M','F');
  TABLES DXPRIBEST*AGEGRP / OUT=OUT.AGED&YR.;
RUN;
```

```
PROC FREQ DATA=SSIPOP ORDER=FORMATTED;
  *WHERE ALLPAY=1;
  BY SEX;
  TABLE DXPRIBEST*AGEGRP / OUT=OUT.DGEN20&TYR.;
RUN;
*END TABLE 36 DATASETS;
```

```
%MEND YEAR;
```

```
%YEAR(YR=2021, TYR=21);
```

```
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM>
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM>
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM>
TO: <DPHELPS@MATHEMATICA-MPR.COM>
SUBJECT: SSIVALT IS COMPLETE
```

## Appendix A.211

JCL/SAS Code: CPORT2

```
//#3590CPO JOB (12510000,T715,,SAS,,ITC919DC),BRONNIKOV,
// MSGCLASS=Q,MSGLEVEL=(1,1),REGION=7168K,NOTIFY=#3590
//*
//STEP1 EXEC SAS9
//SYSPRINT DD SYSOUT=*
//IN1 DD DSN=OPDR.TG.PRD.ETTW.$4671.VAL21P.SSIVALT.SA.V1,DISP=SHR
//OUT1 DD DSN=#3590.XPORT.SSI.ALLP2021,
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)
//OUT2 DD DSN=#3590.XPORT.SSI.GENP2021,
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)
//OUT3 DD DSN=#3590.XPORT.SSI.TOTPAY21,
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)
//OUT4 DD DSN=#3590.XPORT.SSI.PAYGEN21,
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)
//OUT5 DD DSN=#3590.XPORT.SSI.ST2021,
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)
//OUT6 DD DSN=#3590.XPORT.SSI.PAYS2021,
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)
//OUT7 DD DSN=#3590.XPORT.SSI.DIAG2021,
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)
//OUT8 DD DSN=#3590.XPORT.SSI.AGED2021,
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)
//OUT9 DD DSN=#3590.XPORT.SSI.DGEN2021,
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)
OPTIONS NOCENTER;

PROC CPORT DATA=IN1.ALLP2021 FILE=OUT1;
RUN;
PROC CPORT DATA=IN1.GENP2021 FILE=OUT2;
RUN;
PROC CPORT DATA=IN1.TOTPAY21 FILE=OUT3;
RUN;
PROC CPORT DATA=IN1.PAYGEN21 FILE=OUT4;
RUN;
PROC CPORT DATA=IN1.ST2021 FILE=OUT5;
```

Appendix A.211

JCL/SAS Code: CPORT2

```
RUN;  
PROC CPORT DATA=IN1.PAYS2021 FILE=OUT6;  
RUN;  
PROC CPORT DATA=IN1.DIAG2021 FILE=OUT7;  
RUN;  
PROC CPORT DATA=IN1.AGED2021 FILE=OUT8;  
RUN;  
PROC CPORT DATA=IN1.DGEN2021 FILE=OUT9;  
RUN;
```

## Appendix A.212

JCL/SAS Code: Download\_Tables

```
*LIBNAME IN "M:\DAF20\Validation\SSI\Data";
%LET basepath = \\ba.ad.ssa.gov\dcrdp\RemoteShares\MPR\DAF21\Validation;

%let PrintPath = &basepath.\SSI\Programs;
%let ProgName = Download_Tables;

proc printto new
  log= "&PrintPath./&ProgName..log"
  print= "&PrintPath./&ProgName..lst"
;
run;

%put basepath = &basepath.;

LIBNAME IN "&basepath.\SSI\Data";

/*TABLE 36*/
FILENAME DIAG2021 "&basepath.\SSI\Data\DIAG2021.XPT";
PROC CIMPORT LIBRARY=IN INFILE=DIAG2021;
RUN;
FILENAME AGED2021 "&basepath.\SSI\Data\AGED2021.XPT";
PROC CIMPORT LIBRARY=IN INFILE=AGED2021;
RUN;
FILENAME DGEN2021 "&basepath.\SSI\Data\DGEN2021.XPT";
PROC CIMPORT LIBRARY=IN INFILE=DGEN2021;
RUN;

/*TABLE 5*/
FILENAME ALLP2021 "&basepath.\SSI\Data\ALLP2021.XPT";
PROC CIMPORT LIBRARY=IN INFILE=ALLP2021;
RUN;
FILENAME GENP2021 "&basepath.\SSI\Data\GENP2021.XPT";
PROC CIMPORT LIBRARY=IN INFILE=GENP2021;
RUN;
FILENAME TOTPAY21 "&basepath.\SSI\Data\TOTPAY21.XPT";
PROC CIMPORT LIBRARY=IN INFILE=TOTPAY21;
RUN;
FILENAME PAYGEN21 "&basepath.\SSI\Data\PAYGEN21.XPT";
PROC CIMPORT LIBRARY=IN INFILE=PAYGEN21;
RUN;

/*TABLE 10*/
FILENAME ST2021 "&basepath.\SSI\Data\ST2021.XPT";
```

Appendix A.212

JCL/SAS Code: Download\_Tables

```
PROC CIMPORT LIBRARY=IN INFILE=ST2021;  
RUN;  
FILENAME PAYS2021 "&basepath.\SSI\Data\PAYS2021.XPT";  
PROC CIMPORT LIBRARY=IN INFILE=PAYS2021;  
RUN;
```

```
proc printto; run;
```

Appendix A.213

JCL/SAS Code: Table5\_SSI

```

/*****/
/* PROGRAM: TABLE5SSI.SAS          */
/* PURPOSE: TO READ DATA FOR 2021 INTO AN EXCEL SHELL FOR          */
/*   COMPARISON TO TABLE 5 OF THE SSA SSI TABLES          */
/*****/

%let year=2021;
%let yr=21;

LIBNAME IN 'N:\Project\51390_DAF21\Restricted\DC1\DAF21\Validation\SSI\Data';

DATA ALLP&year;
  SET IN.ALLP&year;
  IF SSIBENE=1;
RUN;

PROC PRINT DATA=ALLP&year; RUN;

FILENAME SSIVAL DDE "EXCEL|Table5!R6C2:R6C2";

DATA _NULL_;
  SET ALLP&year;
  FILE SSIVAL NOTAB LRECL=200;
  PUT COUNT;
RUN;

DATA TEMP_M;
  SET IN.GENP&year;
  IF SEX='M' AND SSIBENE=1;
RUN;

PROC PRINT DATA=TEMP_M; RUN;

FILENAME SSIVAL DDE "EXCEL|Table5!R7C2:R7C2";

DATA _NULL_;
  SET TEMP_M;
  FILE SSIVAL NOTAB LRECL=200;
  PUT COUNT;
RUN;

DATA TEMP_F;
  SET IN.GENP&year;

```

Appendix A.213

JCL/SAS Code: Table5\_SSI

```
IF SEX='F' AND SSIBENE=1;
RUN;

PROC PRINT DATA=TEMP_F; RUN;

FILENAME SSIVAL DDE "EXCEL|Table5!R8C2:R8C2";

DATA _NULL_;
  SET TEMP_F;
  FILE SSIVAL NOTAB LRECL=200;
  PUT COUNT;
RUN;

DATA TOTPAY&yr;
  SET IN.TOTPAY&yr;
RUN;

PROC PRINT DATA=TOTPAY&yr; RUN;

FILENAME SSIVAL DDE "EXCEL|Table5!R12C3:R12C3";

DATA _NULL_;
  SET TOTPAY&yr;
  FILE SSIVAL NOTAB LRECL=200;
  PUT SUMTOT&yr;
RUN;

DATA PAYGEN&yr._M;
  SET IN.PAYGEN&yr;
  IF SEX='M';
RUN;

PROC PRINT DATA=PAYGEN&yr._M; RUN;

FILENAME SSIVAL DDE "EXCEL|Table5!R13C3:R13C3";

DATA _NULL_;
  SET PAYGEN&yr._M;
  FILE SSIVAL NOTAB LRECL=200;
  PUT SUMGEN&yr;
RUN;

DATA PAYGEN&yr._F;
  SET IN.PAYGEN&yr;
  IF SEX='F';
```

Appendix A.213

JCL/SAS Code: Table5\_SSI

```
RUN;
```

```
PROC PRINT DATA=PAYGEN&yr._F; RUN;
```

```
FILENAME SSIVAL DDE "EXCEL|Table5!R14C3:R14C3";
```

```
DATA _NULL_;
```

```
  SET PAYGEN&yr._F;
```

```
  FILE SSIVAL NOTAB LRECL=200;
```

```
  PUT SUMGEN&yr;
```

```
RUN;
```



Appendix A.214

JCL/SAS Code: Table10\_PST

```

/*****
/* PROGRAM: TABLE10_PST.SAS                */
/* PURPOSE: TO READ DATA FOR 2021 INTO AN EXCEL SHELL FOR          */
/*   COMPARISON TO TABLE 10 OF THE SSA SSI TABLES                */
*****/

```

```
LIBNAME IN 'N:\Project\51390_DAF21\Restricted\DC1\DAF21\Validation\SSI\Data';
```

```
%let yr=21;
```

```

DATA TEMP;
  SET IN.ST20&yr(rename=(state&yr.12 = PST&yr.12));
  WHERE ^MISSING(SEX) AND AGEGRP='1' and PST&yr.12 ne '--';

```

```

IF PST&yr.12='AL' THEN ROW=1;
ELSE IF PST&yr.12='AK' THEN ROW=2;
ELSE IF PST&yr.12='AZ' THEN ROW=3;
ELSE IF PST&yr.12='AR' THEN ROW=4;
ELSE IF PST&yr.12='CA' THEN ROW=5;
ELSE IF PST&yr.12='CO' THEN ROW=6;
ELSE IF PST&yr.12='CT' THEN ROW=7;
ELSE IF PST&yr.12='DE' THEN ROW=8;
ELSE IF PST&yr.12='DC' THEN ROW=9;
ELSE IF PST&yr.12='FL' THEN ROW=10;
ELSE IF PST&yr.12='GA' THEN ROW=11;
ELSE IF PST&yr.12='HI' THEN ROW=12;
ELSE IF PST&yr.12='ID' THEN ROW=13;
ELSE IF PST&yr.12='IL' THEN ROW=14;
ELSE IF PST&yr.12='IN' THEN ROW=15;
ELSE IF PST&yr.12='IA' THEN ROW=16;
ELSE IF PST&yr.12='KS' THEN ROW=17;
ELSE IF PST&yr.12='KY' THEN ROW=18;
ELSE IF PST&yr.12='LA' THEN ROW=19;
ELSE IF PST&yr.12='ME' THEN ROW=20;
ELSE IF PST&yr.12='MD' THEN ROW=21;
ELSE IF PST&yr.12='MA' THEN ROW=22;
ELSE IF PST&yr.12='MI' THEN ROW=23;
ELSE IF PST&yr.12='MN' THEN ROW=24;
ELSE IF PST&yr.12='MS' THEN ROW=25;
ELSE IF PST&yr.12='MO' THEN ROW=26;
ELSE IF PST&yr.12='MT' THEN ROW=27;
ELSE IF PST&yr.12='NE' THEN ROW=28;

```

## Appendix A.214

JCL/SAS Code: Table10\_PST

```
ELSE IF PST&yr.12='NV' THEN ROW=29;
ELSE IF PST&yr.12='NH' THEN ROW=30;
ELSE IF PST&yr.12='NJ' THEN ROW=31;
ELSE IF PST&yr.12='NM' THEN ROW=32;
ELSE IF PST&yr.12='NY' THEN ROW=33;
ELSE IF PST&yr.12='NC' THEN ROW=34;
ELSE IF PST&yr.12='ND' THEN ROW=35;
ELSE IF PST&yr.12='OH' THEN ROW=36;
ELSE IF PST&yr.12='OK' THEN ROW=37;
ELSE IF PST&yr.12='OR' THEN ROW=38;
ELSE IF PST&yr.12='PA' THEN ROW=39;
ELSE IF PST&yr.12='RI' THEN ROW=40;
ELSE IF PST&yr.12='SC' THEN ROW=41;
ELSE IF PST&yr.12='SD' THEN ROW=42;
ELSE IF PST&yr.12='TN' THEN ROW=43;
ELSE IF PST&yr.12='TX' THEN ROW=44;
ELSE IF PST&yr.12='UT' THEN ROW=45;
ELSE IF PST&yr.12='VT' THEN ROW=46;
ELSE IF PST&yr.12='VA' THEN ROW=47;
ELSE IF PST&yr.12='WA' THEN ROW=48;
ELSE IF PST&yr.12='WV' THEN ROW=49;
ELSE IF PST&yr.12='WI' THEN ROW=50;
ELSE IF PST&yr.12='WY' THEN ROW=51;
ELSE IF PST&yr.12='GU' THEN ROW=54;
ELSE IF PST&yr.12='MP' THEN ROW=55;
ELSE IF PST&yr.12='PR' THEN ROW=56;
ELSE IF PST&yr.12='VI' THEN ROW=57;
ELSE IF PST&yr.12='MH' THEN ROW=58; *NEW FOR DAF12 - MARSHALL ISLANDS;
ELSE IF PST&yr.12=' ' THEN ROW=59;
```

```
RUN;
```

```
PROC SORT DATA=TEMP;
  BY SEX ROW;
RUN;
```

```
/*NEW FOR DAF12 - CODE ENSURES ROW VALUES 1-59 ARE PRESENT IN DATASET. THIS IS FOR A CLEAN LINK TO EXCEL VIA DDE*/
```

```
DATA TEMP;
SET TEMP;
BY SEX ROW;
RETAIN X1;
IF FIRST.SEX THEN X1=1;
IF LAST.SEX THEN END=59;
ELSE END=ROW;
```

```
DO X1=X1 TO END;
```

Appendix A.214

JCL/SAS Code: Table10\_PST

```
    IF X1 NE ROW THEN X2=.;
    ELSE X2=ROW;
    OUTPUT;
END;

RUN;

DATA TEMP;
SET TEMP;
IF ROW^=X1 THEN DO;
    COUNT=0;
    PERCENT=.;
    ROW=X1;
END;

KEEP SEX PST&YR.12 AGEGRP COUNT PERCENT ROW;

RUN;

PROC SORT DATA=TEMP(WHERE=(SEX="F")) OUT=TEMP_F;
    BY ROW;
RUN;

PROC SUMMARY DATA=TEMP_F SUM;
    BY ROW;
    VAR COUNT;
    OUTPUT OUT=OUT_F SUM=SUM;
RUN;

PROC PRINT DATA=OUT_F; RUN;

FILENAME SSIVAL DDE "EXCEL|Table10 PST!R126C2:R176C2";

DATA _NULL_;
    SET OUT_F;
    FILE SSIVAL NOTAB LRECL=200;
    IF 1 <= ROW <= 51;
    PUT SUM;
RUN;

FILENAME SSIVAL DDE "EXCEL|Table10 PST!R178C2:R183C2";

DATA _NULL_;
    SET OUT_F;
    FILE SSIVAL NOTAB LRECL=200;
```

Appendix A.214

JCL/SAS Code: Table10\_PST

```
WHERE 54 <= ROW <= 59;
PUT SUM;
RUN;
```

```
PROC SORT DATA=TEMP(WHERE=(SEX="M")) OUT=TEMP_M;
BY ROW;
RUN;
```

```
PROC SUMMARY DATA=TEMP_M SUM;
BY ROW;
VAR COUNT;
OUTPUT OUT=OUT_M SUM=SUM;
RUN;
```

```
PROC PRINT DATA=OUT_M; RUN;
```

```
FILENAME SSIVAL DDE "EXCEL|Table10 PST!R66C2:R116C2";
```

```
DATA _NULL_;
SET OUT_M;
FILE SSIVAL NOTAB LRECL=200;
IF 1 <= ROW <= 51;
PUT SUM;
RUN;
```

```
FILENAME SSIVAL DDE "EXCEL|Table10 PST!R118C2:R123C2";
```

```
DATA _NULL_;
SET OUT_M;
FILE SSIVAL NOTAB LRECL=200;
IF 54 <= ROW <= 59;
PUT SUM;
RUN;
```

Appendix A.215

JCL/SAS Code: Table36\_SSI (DXPRIBEST)

```

/*****
/* PROGRAM: TABLE36.SAS
/* PURPOSE: TO READ DATA FOR 2021 INTO AN EXCEL SHELL FOR
/* COMPARISON TO TABLE 36 OF THE SSA SSI TABLES
*****/

```

```
LIBNAME IN 'N:\Project\51390_DAF21\Restricted\DC1\DAF21\Validation\SSI\Data';
```

```

***This program formats the diagnostic groups and regroups them into the ***;
***categories used in the benchmark table ***;

```

```

PROC FORMAT;
VALUE DIAG
  1=CONGEN_ANOM
  2=ENDOCRINE
  3=INFECTIOUS
  4=INJURIES
  5=AUTISTIC
  6=DEVELOPMENT
  7=CHILDHOOD
  8=INT_DIS
  9=MOOD
  10=ORGANIC
  11=SCHIZOPHREN
  12=MENTAL_OTHER
  13=NEOPLASMS
  14=BLOOD
  15=CIRCULATORY
  16=DIGESTIVE
  17=GENITO
  18=MUSCULO
  19=NERVOUS
  20=RESPIRATORY
  21=SKIN
  22=OTHER
  23=MISSING;
RUN;

```

```
%MACRO DIAGYR(YEAR = ,YRDIG = ,ROWYEAR = ,RESTRICT = ,FILE = );
```

```
DATA TEMP&YRDIG;
SET IN.&FILE.&YEAR;
```

```
IF &RESTRICT;
```

## Appendix A.215

JCL/SAS Code: Table36\_SSI (DXPRIBEST)

```
*THE CODE FROM SSA;
```

```
*** INFECTIOUS/PARASITIC DISEASES ***;
```

```
IF DXPRIBEST>='0020' AND DXPRIBEST<='0189' THEN GROUP=3;
  ELSE IF DXPRIBEST>='0200' AND DXPRIBEST<='0279' THEN GROUP=3;
  ELSE IF DXPRIBEST>='0300' AND DXPRIBEST<='0419' THEN GROUP=3;
  ELSE IF DXPRIBEST>='0430' AND DXPRIBEST<='0579' THEN GROUP=3;
  ELSE IF DXPRIBEST>='0600' AND DXPRIBEST<='0669' THEN GROUP=3;
  ELSE IF DXPRIBEST>='0700' AND DXPRIBEST<='0889' THEN GROUP=3;
  ELSE IF DXPRIBEST>='0900' AND DXPRIBEST<='1049' THEN GROUP=3;
  ELSE IF DXPRIBEST>='1100' AND DXPRIBEST<='1189' THEN GROUP=3;
  ELSE IF DXPRIBEST>='1200' AND DXPRIBEST<='1359' THEN GROUP=3;
  ELSE IF DXPRIBEST>='1370' AND DXPRIBEST<='1399' THEN GROUP=3;
ELSE IF DXPRIBEST>='7710' AND DXPRIBEST<='7719' THEN GROUP=3;
```

```
*** NEOPLASMS ***;
```

```
ELSE IF DXPRIBEST>='0420' AND DXPRIBEST<='0429' THEN GROUP=13;
ELSE IF DXPRIBEST>='1400' AND DXPRIBEST<='1659' THEN GROUP=13;
  ELSE IF DXPRIBEST>='1700' AND DXPRIBEST<='1769' THEN GROUP=13;
  ELSE IF DXPRIBEST>='1780' AND DXPRIBEST<='2089' THEN GROUP=13;
  ELSE IF DXPRIBEST>='2100' AND DXPRIBEST<='2399' THEN GROUP=13;
```

```
*** ENDOCRINE ***;
```

```
ELSE IF DXPRIBEST>='2400' AND DXPRIBEST<='2469' THEN GROUP=2;
ELSE IF DXPRIBEST>='2500' AND DXPRIBEST<='2539' THEN GROUP=2;
ELSE IF DXPRIBEST>='2550' AND DXPRIBEST<='2559' THEN GROUP=2;
ELSE IF DXPRIBEST>='2600' AND DXPRIBEST<='2799' THEN GROUP=2;
```

```
*** DISEASES OF THE BLOOD ***;
```

```
ELSE IF DXPRIBEST>='2800' AND DXPRIBEST<='2899' THEN GROUP=14;
ELSE IF DXPRIBEST>='7720' AND DXPRIBEST<='7739' THEN GROUP=14;
ELSE IF DXPRIBEST>='7760' AND DXPRIBEST<='7769' THEN GROUP=14;
```

```
*** AUTISM ***;
```

```
ELSE IF DXPRIBEST>='2990' AND DXPRIBEST<='2999' THEN GROUP=5;
```

```
*** DEVELOPMENTAL DISORDERS ***;
```

```
ELSE IF DXPRIBEST>='3150' AND DXPRIBEST<='3159' THEN GROUP=6;
```

```
/** CHILDHOOD/ADOLESCENT DISORDERS **/
```

```
ELSE IF DXPRIBEST>='3120' AND DXPRIBEST<='3139' THEN GROUP=7;
```

```
/** INTELLECTUAL DISABILITY **/
```

```
ELSE IF DXPRIBEST>='3170' AND DXPRIBEST<='3194' THEN GROUP=8;
```

## Appendix A.215

JCL/SAS Code: Table36\_SSI (DXPRIBEST)

```
ELSE IF DXPRIBEST>='3196' AND DXPRIBEST<='3199' THEN GROUP=8;

/*** DEPRESSIVE, BIPOLAR DISORDERS ***/
ELSE IF DXPRIBEST>='2960' AND DXPRIBEST<='2969' THEN GROUP=9;
ELSE IF DXPRIBEST>='3110' AND DXPRIBEST<='3119' THEN GROUP=9;

/*** NEUROCOGNITIVE MENTAL DISORDERS ***/
ELSE IF DXPRIBEST>='2900' AND DXPRIBEST<='2909' THEN GROUP=10;
ELSE IF DXPRIBEST>='2940' AND DXPRIBEST<='2949' THEN GROUP=10;
ELSE IF DXPRIBEST>='3100' AND DXPRIBEST<='3109' THEN GROUP=10;

/*** SCHIZOPHRENIC ***/
ELSE IF DXPRIBEST>='2950' AND DXPRIBEST<='2959' THEN GROUP=11;
ELSE IF DXPRIBEST>='2970' AND DXPRIBEST<='2989' THEN GROUP=11;

/*** OTHER MENTAL DISORDERS ***/
ELSE IF DXPRIBEST>='2910' AND DXPRIBEST<='2939' THEN GROUP=12;
ELSE IF DXPRIBEST>='3000' AND DXPRIBEST<='3099' THEN GROUP=12;
ELSE IF DXPRIBEST>='3140' AND DXPRIBEST<='3149' THEN GROUP=12;
ELSE IF DXPRIBEST>='3160' AND DXPRIBEST<='3169' THEN GROUP=12;
ELSE IF DXPRIBEST='3195' THEN GROUP=12;

*** NERVOUS SYSTEM ***;
ELSE IF DXPRIBEST>='3200' AND DXPRIBEST<='3269' THEN GROUP=19;
ELSE IF DXPRIBEST>='3290' AND DXPRIBEST<='3379' THEN GROUP=19;
ELSE IF DXPRIBEST>='3400' AND DXPRIBEST<='3749' THEN GROUP=19;
ELSE IF DXPRIBEST>='3760' AND DXPRIBEST<='3899' THEN GROUP=19;

*** CIRCULATORY ***;
ELSE IF DXPRIBEST>='3750' AND DXPRIBEST<='3759' THEN GROUP=15;
ELSE IF DXPRIBEST>='3900' AND DXPRIBEST<='3989' THEN GROUP=15;
ELSE IF DXPRIBEST>='4010' AND DXPRIBEST<='4059' THEN GROUP=15;
ELSE IF DXPRIBEST>='4100' AND DXPRIBEST<='4179' THEN GROUP=15;
ELSE IF DXPRIBEST>='4200' AND DXPRIBEST<='4389' THEN GROUP=15;
ELSE IF DXPRIBEST>='4400' AND DXPRIBEST<='4449' THEN GROUP=15;
ELSE IF DXPRIBEST>='4460' AND DXPRIBEST<='4489' THEN GROUP=15;
ELSE IF DXPRIBEST>='4510' AND DXPRIBEST<='4599' THEN GROUP=15;

*** RESPIRATORY ***;
ELSE IF DXPRIBEST>='4600' AND DXPRIBEST<='4669' THEN GROUP=20;
ELSE IF DXPRIBEST>='4700' AND DXPRIBEST<='4789' THEN GROUP=20;
ELSE IF DXPRIBEST>='4800' AND DXPRIBEST<='4879' THEN GROUP=20;
ELSE IF DXPRIBEST>='4900' AND DXPRIBEST<='4969' THEN GROUP=20;
ELSE IF DXPRIBEST>='5000' AND DXPRIBEST<='5089' THEN GROUP=20;
ELSE IF DXPRIBEST>='5100' AND DXPRIBEST<='5199' THEN GROUP=20;
```

## Appendix A.215

JCL/SAS Code: Table36\_SSI (DXPRIBEST)

```
ELSE IF DXPRIBEST>='7680' AND DXPRIBEST<='7709' THEN GROUP=20;

*** DIGESTIVE ***;
ELSE IF DXPRIBEST>='5200' AND DXPRIBEST<='5379' THEN GROUP=16;
ELSE IF DXPRIBEST>='5400' AND DXPRIBEST<='5439' THEN GROUP=16;
ELSE IF DXPRIBEST>='5500' AND DXPRIBEST<='5539' THEN GROUP=16;
ELSE IF DXPRIBEST>='5550' AND DXPRIBEST<='5589' THEN GROUP=16;
ELSE IF DXPRIBEST>='5600' AND DXPRIBEST<='5609' THEN GROUP=16;
ELSE IF DXPRIBEST>='5620' AND DXPRIBEST<='5629' THEN GROUP=16;
ELSE IF DXPRIBEST>='5640' AND DXPRIBEST<='5799' THEN GROUP=16;
ELSE IF DXPRIBEST>='7770' AND DXPRIBEST<='7779' THEN GROUP=16;

*** GENITOURINARY ***;
ELSE IF DXPRIBEST>='5800' AND DXPRIBEST<='6089' THEN GROUP=17;
ELSE IF DXPRIBEST>='6100' AND DXPRIBEST<='6119' THEN GROUP=17;
ELSE IF DXPRIBEST>='6140' AND DXPRIBEST<='6299' THEN GROUP=17;

*** SKIN ***;
ELSE IF DXPRIBEST>='6800' AND DXPRIBEST<='6869' THEN GROUP=21;
ELSE IF DXPRIBEST>='6900' AND DXPRIBEST<='6989' THEN GROUP=21;
ELSE IF DXPRIBEST>='7000' AND DXPRIBEST<='7099' THEN GROUP=21;
ELSE IF DXPRIBEST>='7780' AND DXPRIBEST<='7789' THEN GROUP=21;

*** MUSCULOSKELETAL ***;
ELSE IF DXPRIBEST>='7100' AND DXPRIBEST<='7399' THEN GROUP=18;

*** CONGENITAL ***;
ELSE IF DXPRIBEST>='7400' AND DXPRIBEST<='7599' THEN GROUP=1;

*** INJURIES ***;
ELSE IF DXPRIBEST>='8000' AND DXPRIBEST<='8489' THEN GROUP=4;
ELSE IF DXPRIBEST>='8500' AND DXPRIBEST<='8549' THEN GROUP=4;
ELSE IF DXPRIBEST>='8600' AND DXPRIBEST<='8879' THEN GROUP=4;
ELSE IF DXPRIBEST>='8900' AND DXPRIBEST<='8979' THEN GROUP=4;
ELSE IF DXPRIBEST>='9000' AND DXPRIBEST<='9059' THEN GROUP=4;
ELSE IF DXPRIBEST>='9070' AND DXPRIBEST<='9099' THEN GROUP=4;
ELSE IF DXPRIBEST>='9200' AND DXPRIBEST<='9299' THEN GROUP=4;
ELSE IF DXPRIBEST>='9400' AND DXPRIBEST<='9599' THEN GROUP=4;

*** OTHER ***;
ELSE IF DXPRIBEST>='7600' AND DXPRIBEST<='7609' THEN GROUP=22;
ELSE IF DXPRIBEST>='7640' AND DXPRIBEST<='7669' THEN GROUP=22;
ELSE IF DXPRIBEST>='7800' AND DXPRIBEST<='7809' THEN GROUP=22;
ELSE IF DXPRIBEST>='7830' AND DXPRIBEST<='7849' THEN GROUP=22;
ELSE IF DXPRIBEST='9330' THEN GROUP=22;
```



## Appendix A.215

JCL/SAS Code: Table36\_SSI (DXPRIBEST)

```
*** UNKNOWN ***;
ELSE IF DXPRIBEST='' THEN GROUP=23;
ELSE IF DXPRIBEST>='0000' AND DXPRIBEST<='0019' THEN GROUP=23;
ELSE IF DXPRIBEST>='0190' AND DXPRIBEST<='0199' THEN GROUP=23;
ELSE IF DXPRIBEST>='0280' AND DXPRIBEST<='0299' THEN GROUP=23;
ELSE IF DXPRIBEST>='0580' AND DXPRIBEST<='0599' THEN GROUP=23;
ELSE IF DXPRIBEST>='0670' AND DXPRIBEST<='0699' THEN GROUP=23;
ELSE IF DXPRIBEST>='0890' AND DXPRIBEST<='0899' THEN GROUP=23;
ELSE IF DXPRIBEST>='1050' AND DXPRIBEST<='1099' THEN GROUP=23;
ELSE IF DXPRIBEST>='1190' AND DXPRIBEST<='1199' THEN GROUP=23;
ELSE IF DXPRIBEST>='1360' AND DXPRIBEST<='1369' THEN GROUP=23;
ELSE IF DXPRIBEST>='1660' AND DXPRIBEST<='1699' THEN GROUP=23;
ELSE IF DXPRIBEST>='1770' AND DXPRIBEST<='1779' THEN GROUP=23;
ELSE IF DXPRIBEST>='2090' AND DXPRIBEST<='2099' THEN GROUP=23;
ELSE IF DXPRIBEST>='2470' AND DXPRIBEST<='2499' THEN GROUP=23;
ELSE IF DXPRIBEST>='2540' AND DXPRIBEST<='2549' THEN GROUP=23;
ELSE IF DXPRIBEST>='2560' AND DXPRIBEST<='2599' THEN GROUP=23;
ELSE IF DXPRIBEST>='3270' AND DXPRIBEST<='3289' THEN GROUP=23;
ELSE IF DXPRIBEST>='3380' AND DXPRIBEST<='3399' THEN GROUP=23;
ELSE IF DXPRIBEST>='3990' AND DXPRIBEST<='4009' THEN GROUP=23;
ELSE IF DXPRIBEST>='4060' AND DXPRIBEST<='4099' THEN GROUP=23;
ELSE IF DXPRIBEST>='4180' AND DXPRIBEST<='4199' THEN GROUP=23;
ELSE IF DXPRIBEST>='4390' AND DXPRIBEST<='4399' THEN GROUP=23;
ELSE IF DXPRIBEST>='4450' AND DXPRIBEST<='4459' THEN GROUP=23;
ELSE IF DXPRIBEST>='4490' AND DXPRIBEST<='4509' THEN GROUP=23;
ELSE IF DXPRIBEST>='4670' AND DXPRIBEST<='4699' THEN GROUP=23;
ELSE IF DXPRIBEST>='4790' AND DXPRIBEST<='4799' THEN GROUP=23;
ELSE IF DXPRIBEST>='4880' AND DXPRIBEST<='4899' THEN GROUP=23;
ELSE IF DXPRIBEST>='4970' AND DXPRIBEST<='4999' THEN GROUP=23;
ELSE IF DXPRIBEST>='5090' AND DXPRIBEST<='5099' THEN GROUP=23;
ELSE IF DXPRIBEST>='5380' AND DXPRIBEST<='5399' THEN GROUP=23;
ELSE IF DXPRIBEST>='5440' AND DXPRIBEST<='5499' THEN GROUP=23;
ELSE IF DXPRIBEST>='5540' AND DXPRIBEST<='5549' THEN GROUP=23;
ELSE IF DXPRIBEST>='5590' AND DXPRIBEST<='5599' THEN GROUP=23;
ELSE IF DXPRIBEST>='5610' AND DXPRIBEST<='5619' THEN GROUP=23;
ELSE IF DXPRIBEST>='5630' AND DXPRIBEST<='5639' THEN GROUP=23;
ELSE IF DXPRIBEST>='6090' AND DXPRIBEST<='6099' THEN GROUP=23;
ELSE IF DXPRIBEST>='6120' AND DXPRIBEST<='6139' THEN GROUP=23;
ELSE IF DXPRIBEST>='6300' AND DXPRIBEST<='6799' THEN GROUP=23;
ELSE IF DXPRIBEST>='6870' AND DXPRIBEST<='6899' THEN GROUP=23;
ELSE IF DXPRIBEST>='6990' AND DXPRIBEST<='6999' THEN GROUP=23;
ELSE IF DXPRIBEST>='7610' AND DXPRIBEST<='7639' THEN GROUP=23;
ELSE IF DXPRIBEST>='7670' AND DXPRIBEST<='7679' THEN GROUP=23;
ELSE IF DXPRIBEST>='7790' AND DXPRIBEST<='7799' THEN GROUP=23;
```

## Appendix A.215

JCL/SAS Code: Table36\_SSI (DXPRIBEST)

```
ELSE IF DXPRIBEST>='7810' AND DXPRIBEST<='7829' THEN GROUP=23;
ELSE IF DXPRIBEST>='7850' AND DXPRIBEST<='7999' THEN GROUP=23;
ELSE IF DXPRIBEST>='8490' AND DXPRIBEST<='8499' THEN GROUP=23;
ELSE IF DXPRIBEST>='8550' AND DXPRIBEST<='8599' THEN GROUP=23;
ELSE IF DXPRIBEST>='8880' AND DXPRIBEST<='8899' THEN GROUP=23;
ELSE IF DXPRIBEST>='8980' AND DXPRIBEST<='8999' THEN GROUP=23;
ELSE IF DXPRIBEST>='9060' AND DXPRIBEST<='9069' THEN GROUP=23;
ELSE IF DXPRIBEST>='9100' AND DXPRIBEST<='9199' THEN GROUP=23;
ELSE IF DXPRIBEST>='9300' AND DXPRIBEST<='9329' THEN GROUP=23;
ELSE IF DXPRIBEST>='9331' AND DXPRIBEST<='9399' THEN GROUP=23;
ELSE IF DXPRIBEST>='9600' AND DXPRIBEST<='9999' THEN GROUP=23;
ELSE GROUP=23;
FORMAT GROUP DIAG.;
RUN;

TITLE "FREQUENCY OF DIG CODES WITHIN UNKNOWN/MISSING GROUP";
TITLE2 "&FILE FILE, &RESTRICT RESTRICTION";
PROC FREQ DATA= TEMP&YRDIG;
  TABLES DXPRIBEST*GROUP*COUNT/LIST MISSING;
  WHERE GROUP=23 or GROUP=. and &RESTRICT;
RUN;

PROC SORT DATA=TEMP&YRDIG;
  BY GROUP;
RUN;

PROC FREQ DATA=TEMP&YRDIG;
  TABLES GROUP / LIST MISSING;
  TITLE "DIAGNOSTIC GROUPS &YEAR";
RUN;

PROC SUMMARY DATA=TEMP&YRDIG SUM;
  BY GROUP;
  VAR COUNT;
  OUTPUT OUT=OUT&YRDIG SUM=SUM;
RUN;

PROC PRINT DATA=OUT&YRDIG;
RUN;

PROC TRANSPOSE DATA=OUT&YRDIG OUT=OUT&ROWYEAR;
  VAR SUM;
  ID GROUP;
RUN;
```

Appendix A.215

JCL/SAS Code: Table36\_SSI (DXPRIBEST)

```
PROC PRINT DATA=OUT&YRDIG;  
RUN;
```

```
FILENAME SSIVAL DDE "EXCEL|Table36!R&ROWYEAR.C3:R&ROWYEAR.C25";
```

```
DATA _NULL_;  
  SET OUT&ROWYEAR;  
  FILE SSIVAL NOTAB LRECL=200;  
  PUT CONGEN_ANOM '09'X ENDOCRINE '09'X INFECTIOUS '09'X INJURIES '09'X AUTISTIC '09'X DEVELOPMENT '09'X  
    CHILDHOOD '09'X INT_DIS '09'X MOOD '09'X ORGANIC '09'X SCHIZOPHREN '09'X MENTAL_OTHER '09'X  
    NEOPLASMS '09'X BLOOD '09'X CIRCULATORY '09'X DIGESTIVE '09'X GENITO '09'X MUSCULO '09'X NERVOUS '09'X  
    RESPIRATORY '09'X SKIN '09'X OTHER '09'X MISSING;  
RUN;  
%MEND;
```

```
/*really don't need yrdig... could drop it and leave the dataset name non-year specific*/  
%DIAGYR(YEAR = 2021,YRDIG =21,ROWYEAR = 6,RESTRICT = AGEGRP = '1', FILE = AGED)  
%DIAGYR(YEAR = 2021,YRDIG =21,ROWYEAR = 9,RESTRICT = AGEGRP = '1' AND SEX='F', FILE = DGEN)  
%DIAGYR(YEAR = 2021,YRDIG =21,ROWYEAR = 8,RESTRICT = AGEGRP = '1' AND SEX='M', FILE = DGEN)
```

```
endsas;
```

Appendix A.216

JCL/SAS Code: STEP1DI

```

//$4671DI1 JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT: 50214 DAF
//* PROJECT
//* DIRECTOR: MATTHEW URATO
//* PROGRAM: OPDR.TG.PRD.ETTW.$4671.DAF21.VAL.PRDLIB(STEP1DI)
//*
//* DESCRIPTION: CREATE BENCHMARKING FOR DI TABLES 19, 21, 27
//*
//* INPUT: CURRENT DAF YEAR ANNUAL FILE
//* OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2021
//*
//* DEMOGRAPHIC FILE
//* OPDR.TG.PRD.ETTW.FINAL.DAF21P.DMG
//*
//* OUTPUT: OPDR.TG.PRD.ETTW.$4671.VAL21P.STEP1DI.SA.V1
//* OPDR.TG.PRD.ETTW.$4671.VAL21P.DIVAL17.SA.V1
//*
//* UPDATED: 12/16/2022 SVETLANA BRONNIKOV
//*
//* PURPOSE: CREATE TABLES FOR BENCHMARKING
//*****
//SAS EXEC SAS9,
// WORK='120000,60000'
//*
//DEM DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.DMG,DISP=SHR
//Y21 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2021E,DISP=SHR
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
// SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//OUT DD DSN=OPDR.TG.PRD.ETTW.$4671.VAL21P.STEP1DI.SA.V1,
// DISP=(NEW,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.$4671.VAL21P.DIVAL21.SA.V1,
// DISP=(NEW,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

TITLE1 "BENCHMARKING";
TITLE2 "DI BENEFICIARIES";
TITLE3;

```

## Appendix A.216

JCL/SAS Code: STEP1DI

```
*** CHANGE FOR TRF09 ***;
*** TOC CHANGED TO TOC1 TO REFLECT ADDITION ***;
*** OF MULTIPLE TOC VARIABLES ***;
*** CHANGE FOR TRF10 ***;
*** TOC1 CHANGED TO FIND MOST RECENT TOC AND ***
*** USE THAT VALUE ***;

%LET YR=21;
%LET NEXTYR=22;
%LET DATE = '31DEC2021'D;

DATA OUT.STEP1DI;
  MERGE Y&YR..Y20&YR.(IN=ANN KEEP=SSN LAF: DXPRIBEST: DXSECBEST:
    STATE: FPMT: PROA: PROB:)
    DEM.DMG(IN=DEMO KEEP=SSN SEX DOBBEST BIC TOC: DOEI DOEC
      /* CHANGE TO REFLECT SSA RESTRICTIONS */
      WHERE=(BIC='A'))
  ;
  BY SSN;

  IF DEMO AND ANN;

  *** THIS CODE WAS USED TO CHECK THAT THE CALCULATIONS BELOW ***;
  *** WERE WORKING ***;
  IF TOC_START2 NE . AND (TOC_START1>TOC_START2) THEN DO;
    FLAG_START=1;
    FLAG_TOC1=TOC1;
    FLAG_TOC2=TOC2;
    FLAG_TOCSTART1=TOC_START1;
    FLAG_TOCSTART2=TOC_START2;
  END;

  IF TOC_NUM=. THEN TOC=' ';
  ELSE IF TOC_NUM=1 THEN TOC=TOC1;

  ARRAY TOCSTART(20) TOC_START1-TOC_START20;
  ARRAY TOCS(20) TOC1-TOC20;

  IF TOC_NUM > 1 THEN DO;
    DO I=1 TO (TOC_NUM-1);
      IF TOCSTART(I) >= TOCSTART(I+1) THEN DO;
        TOCS(I+1)=TOCS(I);
        TOCSTART(I+1)=TOCSTART(I);
        TOC=TOCS(I);
      END;
    END;
  END;
```

Appendix A.216

JCL/SAS Code: STEP1DI

```
    END;
    ELSE TOC=TOCS(I+1);
  END;
END;

IF TOC IN ('5','6');

*** CHECK THAT DOEC IS BEFORE 12/31 OF CURRENT YEAR ***;
IF DOEC < &DATE. THEN DOECFLAG=1;
*** ADDED IN 2009 ***;
IF DOEC NE .;

* CALCULATE FULL RETIREMENT AGE (FRA) BASED ON SPECS IN TRF06
  USERS MANUAL;

* ADJUST RETIREMENT DATE FOR BENEFICIARIES BORN ON JAN 1;
IF MONTH(DOBBEST)=1 AND DAY(DOBBEST)=1 THEN
  DOBRETIRE=INTNX('DAY',DOBBEST,-1);
ELSE DOBRETIRE=DOBBEST;

* SET MONTHS TO NUMBER OF MONTHS UNTIL FRA;
IF YEAR(DOBRETIRE)<=1937 THEN MONTHS=780;
IF YEAR(DOBRETIRE)=1938 THEN MONTHS=782;
IF YEAR(DOBRETIRE)=1939 THEN MONTHS=784;
IF YEAR(DOBRETIRE)=1940 THEN MONTHS=786;
IF YEAR(DOBRETIRE)=1941 THEN MONTHS=788;
IF YEAR(DOBRETIRE)=1942 THEN MONTHS=790;
IF 1943<=YEAR(DOBRETIRE)<=1954 THEN MONTHS=792;
IF YEAR(DOBRETIRE)=1955 THEN MONTHS=794;
IF YEAR(DOBRETIRE)=1956 THEN MONTHS=796;
IF YEAR(DOBRETIRE)=1957 THEN MONTHS=798;
IF YEAR(DOBRETIRE)=1958 THEN MONTHS=800;
IF YEAR(DOBRETIRE)=1959 THEN MONTHS=802;
IF YEAR(DOBRETIRE)>=1960 THEN MONTHS=804;

RETIREDATE=INTNX('MONTH',DOBRETIRE,MONTHS,'SAMEDAY');

FRA=YRDIF(DOBRETIRE,RETIREDATE,'SAMEDAY');

*** ADD AGE CALCULATION TO MATCH SSA ***;
SELDATEMM=1;
SELDATEYY=20&NEXTYR.;
SELDATEDD=1;
SELDATE2=MDY(SELDATEMM, SELDATEDD, SELDATEYY);
MONTHS=INTCK('MONTH',DOBBEST,SELDATE2);
```

## Appendix A.216

JCL/SAS Code: STEP1DI

```
DOBDD=DAY(DOBBEST);
IF DOBDD>1 THEN MONTHS=MONTHS-1;
AGE&YR.=MONTHS/12;

*** FLAG BENES WHOSE AGE IS GREATER THAN FRA ***;
IF AGE&YR.>=FRA THEN DELETEDFRA=1;

*** FLAG DI BENES IN CURRENT PAY STATUS AS OF DEC;
DIBENE=0;
IF SUBSTR(LEFT(LAF&YR.12),1,1) IN ('C','E') THEN DIBENE=1;

RUN;

*** THIS CODE WAS FOR TESTING ONLY - CAN BE UNCOMMENTED FOR ***;
*** RETESTING ***;
*PROC FREQ DATA=OUT.STEP1DI;
* TABLES FLAG_START;
*RUN;

*PROC PRINT DATA=OUT.STEP1DI;
* WHERE FLAG_START=1;
* VAR FLAG: TOC;
*RUN;

PROC FREQ DATA=OUT.STEP1DI;
TABLES TOC_NUM TOC1 TOC2 TOC3 TOC4 TOC5 TOC6 TOC
/ LIST MISSING;
RUN;

PROC FREQ DATA=OUT.STEP1DI;
TABLES DIBENE DELETEDFRA DIBENE*DELETEDFRA DOECFLAG/ LIST MISSING;
RUN;

PROC FREQ DATA=OUT.STEP1DI;
TABLES AGE&YR.
DELETEDFRA*AGE&YR. / LIST MISSING;
RUN;

PROC FORMAT;
VALUE AGEGRP LOW-<30= '1'
30-<35 = '2'
35-<40 = '3'
40-<45 = '4'
45-<50 = '5'
50-<55 = '6'
```

## Appendix A.216

JCL/SAS Code: STEP1DI

```
        55-<60 = '7'
        60-HIGH= '8';
VALUE AGEGR LOW-<35='1'
        35-<40 = '2'
        40-<45 = '3'
        45-<50 = '4'
        50-<55 = '5'
        55-<60 = '6'
        60-65 = '7';
RUN;

DATA OUT&YR..Y20&YR.;
  SET OUT.STEP1DI(KEEP=SEX DIBENE AGE&YR. DXPRIBEST&YR.12 DOECFLAG
    STATE&YR.12 DELETFRA);
  AGEGRP=PUT(AGE&YR.,AGEGRP.);
  AGEGR=PUT(AGE&YR.,AGEGR.);
RUN;

PROC CONTENTS DATA=OUT&YR..Y20&YR;
RUN;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM>
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM>
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM>
TO: <DPHELPS@MATHEMATICA-MPR.COM>
SUBJECT: STEP1DI IS COMPLETE
```



## Appendix A.217

JCL/SAS Code: STEP2DI

```

//$4671DI2 JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT: 50214 DAF
//*
//* PROJECT
//* DIRECTOR:  MATTHEW URATO
//*
//* PROGRAM:  OPDR.TG.PRD.ETTW.$4671.VAL21.PRDLIB(STEP2DI)
//*
//* DESCRIPTION: CREATE BENCHMARKING
//*
//* INPUT:  OPDR.TG.PRD.ETTW.$4671.VAL21P.STEP1DI.SA.V1
//*
//* OUTPUT:  OPDR.TG.PRD.ETTW.$4671.VAL21P.STEP2DI.SA.V1
//*
//* PURPOSE: CREATE TABLES FOR BENCHMARKING
//*****
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//IN  DD DSN=OPDR.TG.PRD.ETTW.$4671.VAL21P.DIVAL21.SA.V1,
//  DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.$4671.VAL21P.STEP2DI.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

```

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

```

TITLE1 "BENCHMARKING";
TITLE2 "DI BENEFICIARIES";
TITLE3;

```

```
%LET YR=21;
```

```

*** RUN FREQUENCY OF GENDER BY AGE FOR TABLE 19 ***;
PROC FREQ DATA=IN.Y20&YR. ORDER=FORMATTED;
  WHERE DIBENE=1 AND DOECFLAG=1 AND DELETEDFRA NE 1;
  TABLE SEX*AGEGRP / OUT=OUT.YF20&YR.;
RUN;

```

## Appendix A.217

JCL/SAS Code: STEP2DI

```
*** RUN MEAN AGE FOR TABLE 19 ***;
PROC MEANS DATA=IN.Y20&YR.;
  WHERE DIBENE=1 AND DOECFLAG=1 AND DELETEDFRA NE 1;
  VAR AGE&YR.;
  OUTPUT OUT=OUT.MA20&YR. MEAN=MEANAGE;
RUN;

PROC SORT DATA=IN.Y20&YR. OUT=OUTY20&YR.;
  BY SEX;
RUN;

*** RUN MEAN AGE BY GENDER FOR TABLE 19 ***;
PROC MEANS DATA=OUTY20&YR.;
  WHERE DIBENE=1 AND DOECFLAG=1 AND DELETEDFRA NE 1;
  VAR AGE&YR.;
  BY SEX;
  OUTPUT OUT=OUT.YM20&YR. MEAN=MEANAGE;
RUN;

*** RUN FREQUENCIES FOR TABLE 21 ***;
PROC FREQ DATA=IN.Y20&YR.;
  WHERE DIBENE=1 AND DOECFLAG=1 AND DELETEDFRA NE 1 AND SEX IN
    ('M','F');
  TABLES DXPRIBEST&YR.12 / OUT=OUT.DIAG20&YR.;
RUN;

PROC FORMAT;
VALUE AGEGR
  LOW-<35='1'
  35-<40='2'
  40-<45='3'
  45-<50='4'
  50-<55='5'
  55-<60='6'
  60-65='7';
RUN;

*** RUN FREQUENCIES FOR TABLE 27 ***;
DATA Y20&YR.;
  SET IN.Y20&YR.;
  AGEGR=PUT(AGE&YR.,AGEGR.);
RUN;

PROC SORT DATA=Y20&YR.;
```

Appendix A.217

JCL/SAS Code: STEP2DI

```
BY SEX;  
RUN;
```

```
PROC FREQ DATA=Y20&YR. ORDER=FORMATTED;  
  WHERE DIBENE=1 AND DOECFLAG=1 AND DELETFRA NE 1;  
  BY SEX;  
  TABLE STATE&YR.12*AGEGR / OUT=OUT.ST20&YR.;  
RUN;
```

## Appendix A.218

JCL/SAS Code: CPORT1

```
//#3590CPO JOB (12510000,T715,,SAS,,ITC919DC),SBRONNIKOV,  
// MSGCLASS=Q,MSGLEVEL=(1,1),REGION=7168K,NOTIFY=#3590  
//*  
//STEP1 EXEC SAS9  
//SYSPRINT DD SYSOUT=*  
//IN1 DD DSN=OPDR.TG.PRD.ETTW.$4671.VAL21P.STEP2DI.SA.V1,DISP=SHR  
//OUT1 DD DSN=#3590.XPORT.DIAG2021,  
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),  
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)  
//OUT2 DD DSN=#3590.XPORT.MA2021,  
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),  
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)  
//OUT3 DD DSN=#3590.XPORT.ST2021,  
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),  
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)  
//OUT4 DD DSN=#3590.XPORT.YF2021,  
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),  
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)  
//OUT5 DD DSN=#3590.XPORT.YM2021,  
//  DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(100,50),RLSE),  
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000,DSORG=PS)  
OPTIONS NOCENTER;  
  
PROC CPORT DATA=IN1.DIAG2021 FILE=OUT1;  
RUN;  
PROC CPORT DATA=IN1.MA2021 FILE=OUT2;  
RUN;  
PROC CPORT DATA=IN1.ST2021 FILE=OUT3;  
RUN;  
PROC CPORT DATA=IN1.YF2021 FILE=OUT4;  
RUN;  
PROC CPORT DATA=IN1.YM2021 FILE=OUT5;  
RUN;
```

Appendix A.219

JCL/SAS Code: Step3DI

```
%LET basepath = \\ba.ad.ssa.gov\dcrdp\RemoteShares\MPR\DAF21\Validation;
```

```
%let PrintPath = &basepath.\SSDI\Programs;
```

```
%let ProgName = Step3DI;
```

```
proc printto new
```

```
  log= "&PrintPath./&ProgName..log"
```

```
  print= "&PrintPath./&ProgName..lst"
```

```
;
```

```
run;
```

```
%put basepath = &basepath.;
```

```
LIBNAME IN "&basepath.\SSDI\Data";
```

```
FILENAME YF2021 "&basepath.\SSDI\Data\yf2021.xpt";
```

```
PROC CIMPORT LIBRARY=IN INFILE=YF2021;
```

```
RUN;
```

```
FILENAME MA2021 "&basepath.\SSDI\Data\ma2021.xpt";
```

```
PROC CIMPORT LIBRARY=IN INFILE=MA2021;
```

```
RUN;
```

```
FILENAME YM2021 "&basepath.\SSDI\Data\ym2021.xpt";
```

```
PROC CIMPORT LIBRARY=IN INFILE=YM2021;
```

```
RUN;
```

```
FILENAME DIAG2021 "&basepath.\SSDI\Data\diag2021.xpt";
```

```
PROC CIMPORT LIBRARY=IN INFILE=DIAG2021;
```

```
RUN;
```

```
FILENAME ST2021 "&basepath.\SSDI\Data\st2021.xpt";
```

```
PROC CIMPORT LIBRARY=IN INFILE=ST2021;
```

```
RUN;
```

```
proc printto; run;
```

## Appendix A.220

JCL/SAS Code: Table19up

```
/******  
/* PROGRAM: TABLE19.SAS */  
/* PURPOSE: TO READ DATA FOR 2021 INTO AN EXCEL SHELL FOR */  
/* COMPARISON TO TABLE 19 OF THE SSA DI TABLES */  
/******  
  
options mprint macrogen;  
  
LIBNAME IN 'N:\Project\51390_DAF21\Restricted\DC1\DAF21\Validation\SSDI\Data';  
  
*** READ YEARLY DATA BY AGE INTO EXCEL ***;  
*** SET MISSING AGEGRUP SO THAT THE NUMBER WILL RUN INTO EXCEL ***;  
%MACRO RUNYEAR(YEAR=,ROWM=,ROWF=,ROWMMEAN=,ROWFMEAN=,ROWYEAR=);  
DATA YF&YEAR;  
  SET IN.YF&YEAR;  
  IF AGEGRP NE "*" THEN DO;  
    AGEGRUP=INPUT(AGEGRP,2.);  
  END;  
  ELSE IF AGEGRP="*" THEN AGEGRUP=9;  
RUN;  
  
PROC PRINT DATA=YF&YEAR;  
RUN;  
  
PROC SORT DATA=YF&YEAR;  
  BY SEX AGEGRUP;  
RUN;  
  
PROC TRANSPOSE DATA=YF&YEAR OUT=OUT_YF&YEAR PREFIX=COL;  
  BY SEX;  
  ID AGEGRUP;  
  VAR COUNT;  
RUN;  
  
*** RUN YEARLY DATA INTO EXCEL ***;  
FILENAME SSDIVAL DDE "EXCEL|Table19!R&ROWM.C4:R&ROWM.C12";  
  
DATA _NULL_;  
  SET OUT_YF&YEAR;  
  FILE SSDIVAL NOTAB LRECL=2000;  
  WHERE SEX='M';  
  PUT COL1 '09'X COL2 '09'X COL3 '09'X COL4 '09'X COL5 '09'X COL6 '09'X COL7 '09'X COL8;  
RUN;
```

Appendix A.220

JCL/SAS Code: Table19up

```
FILENAME SSDIVAL DDE "EXCEL|Table19!R&ROWF.C4:R&ROWF.C12";
```

```
DATA _NULL_;  
  SET OUT_YF&YEAR;  
  FILE SSDIVAL NOTAB LRECL=2000;  
  WHERE SEX='F';  
  PUT COL1 '09'X COL2 '09'X COL3 '09'X COL4 '09'X COL5 '09'X COL6 '09'X COL7 '09'X COL8;  
RUN;
```

```
*** RUN MEAN AGE DATA BY GENDER INTO EXCEL ***;  
FILENAME SSDIVAL DDE "EXCEL|Table19!R&ROWMMEAN.C12:R&ROWMMEAN.C12";
```

```
DATA _NULL_;  
  SET IN.YM&YEAR;  
  FILE SSDIVAL NOTAB LRECL=200;  
  WHERE SEX='M';  
  PUT MEANAGE;  
RUN;
```

```
FILENAME SSDIVAL DDE "EXCEL|Table19!R&ROWFMEAN.C12:R&ROWFMEAN.C12";
```

```
DATA _NULL_;  
  SET IN.YM&YEAR;  
  FILE SSDIVAL NOTAB LRECL=200;  
  WHERE SEX='F';  
  PUT MEANAGE;  
RUN;
```

```
*** RUN MEAN AGE DATA OVERALL INTO EXCEL ***;  
FILENAME SSDIVAL DDE "EXCEL|Table19!R&ROWYEAR.C12:R&ROWYEAR.C12";
```

```
DATA _NULL_;  
  SET IN.MA&YEAR;  
  FILE SSDIVAL NOTAB LRECL=200;  
  PUT MEANAGE;  
RUN;  
%MEND RUNYEAR;
```

```
%RUNYEAR(YEAR=2021,ROWM=17,ROWF=19,ROWMMEAN=8,ROWFMEAN=10,ROWYEAR=6);
```

## Appendix A.221

JCL/SAS Code: Table21up

```
/******  
/* PROGRAM: TABLE21.SAS */  
/* PURPOSE: TO READ DATA FOR 2021 INTO AN EXCEL SHELL FOR */  
/* COMPARISON TO TABLE 21 OF THE SSA DI TABLES */  
/******
```

```
LIBNAME IN 'N:\Project\51390_DAF21\Restricted\DC1\DAF21\Validation\SSDI\Data';
```

```
***This program formats the diagnostic groups and regroups them into the ***;  
***categories used in the benchmark table ***;
```

```
PROC FORMAT;  
VALUE DIAG  
1=CONGEN_ANOM  
2=ENDOCRINE  
3=INFECTIOUS  
4=INJURIES  
5=MENTAL_RET  
6=MENTAL_OTHER  
7=NEOPLASMS  
8=BLOOD  
9=CIRCULATORY  
10=DIGESTIVE  
11=GENITO  
12=MUSCULO  
13=NERVOUS  
14=RESPIRATORY  
15=SKIN  
16=OTHER  
17=MISSING;
```

```
RUN;
```

```
%MACRO DIAGYR(YEAR,YRDIG,ROWYEAR);
```

```
DATA TEMP&YRDIG;
```

```
SET IN.DIAG&YEAR;
```

```
IF DXPRIBEST&YRDIG.12>='0000' AND DXPRIBEST&YRDIG.12<='0009' THEN GROUP=17;  
ELSE IF DXPRIBEST&YRDIG.12>='0010' AND DXPRIBEST&YRDIG.12<='0019' THEN GROUP=17;  
ELSE IF DXPRIBEST&YRDIG.12>='0190' AND DXPRIBEST&YRDIG.12<='0199' THEN GROUP=17;  
ELSE IF DXPRIBEST&YRDIG.12>='0280' AND DXPRIBEST&YRDIG.12<='0289' THEN GROUP=17;  
ELSE IF DXPRIBEST&YRDIG.12>='0290' AND DXPRIBEST&YRDIG.12<='0299' THEN GROUP=17;  
ELSE IF DXPRIBEST&YRDIG.12>='0580' AND DXPRIBEST&YRDIG.12<='0589' THEN GROUP=17;  
ELSE IF DXPRIBEST&YRDIG.12>='0590' AND DXPRIBEST&YRDIG.12<='0599' THEN GROUP=17;  
ELSE IF DXPRIBEST&YRDIG.12>='0670' AND DXPRIBEST&YRDIG.12<='0679' THEN GROUP=17;
```













## Appendix A.221

JCL/SAS Code: Table21up

```
ELSE IF DXPRIBEST&YRDIG.12 IN ('3138','3152','3153','3195') THEN GROUP=6;
ELSE IF DXPRIBEST&YRDIG.12>='2900' AND DXPRIBEST&YRDIG.12<='3169' THEN GROUP=6;

ELSE IF DXPRIBEST&YRDIG.12>='3170' AND DXPRIBEST&YRDIG.12<='3199' THEN GROUP=5;

ELSE IF DXPRIBEST&YRDIG.12>='3200' AND DXPRIBEST&YRDIG.12<='3749' THEN GROUP=13;
ELSE IF DXPRIBEST&YRDIG.12>='3760' AND DXPRIBEST&YRDIG.12<='3899' THEN GROUP=13;

ELSE IF DXPRIBEST&YRDIG.12>='3750' AND DXPRIBEST&YRDIG.12<='3759' THEN GROUP=9;
ELSE IF DXPRIBEST&YRDIG.12>='3900' AND DXPRIBEST&YRDIG.12<='4599' THEN GROUP=9;

ELSE IF DXPRIBEST&YRDIG.12>='4600' AND DXPRIBEST&YRDIG.12<='5199' THEN GROUP=14;
ELSE IF DXPRIBEST&YRDIG.12>='7680' AND DXPRIBEST&YRDIG.12<='7709' THEN GROUP=14;

ELSE IF DXPRIBEST&YRDIG.12>='5200' AND DXPRIBEST&YRDIG.12<='5799' THEN GROUP=10;
ELSE IF DXPRIBEST&YRDIG.12>='7770' AND DXPRIBEST&YRDIG.12<='7779' THEN GROUP=10;

ELSE IF DXPRIBEST&YRDIG.12>='5800' AND DXPRIBEST&YRDIG.12<='6299' THEN GROUP=11;

ELSE IF DXPRIBEST&YRDIG.12>='6800' AND DXPRIBEST&YRDIG.12<='7099' THEN GROUP=15;
ELSE IF DXPRIBEST&YRDIG.12>='7780' AND DXPRIBEST&YRDIG.12<='7789' THEN GROUP=15;

ELSE IF DXPRIBEST&YRDIG.12>='7100' AND DXPRIBEST&YRDIG.12<='7399' THEN GROUP=12;

ELSE IF DXPRIBEST&YRDIG.12>='7400' AND DXPRIBEST&YRDIG.12<='7599' THEN GROUP=1;

ELSE IF DXPRIBEST&YRDIG.12>='7600' AND DXPRIBEST&YRDIG.12<='7849' THEN GROUP=16;
ELSE IF DXPRIBEST&YRDIG.12='9330' THEN GROUP=16;

ELSE IF DXPRIBEST&YRDIG.12>='8000' AND DXPRIBEST&YRDIG.12<='9989' THEN GROUP=4;

ELSE GROUP=17;
FORMAT GROUP DIAG.;
RUN;

PROC SORT DATA=TEMP&YRDIG;
  BY GROUP;
RUN;

PROC FREQ DATA=TEMP&YRDIG;
  TABLES GROUP / LIST MISSING;
  TITLE "DIAGNOSTIC GROUPS & YEAR";
```

Appendix A.221

JCL/SAS Code: Table21up

```
RUN;

PROC SUMMARY DATA=TEMP&YRDIG SUM;
  BY GROUP;
  VAR COUNT;
  OUTPUT OUT=OUT&YRDIG SUM=SUM;
RUN;

PROC PRINT DATA=OUT&YRDIG;
RUN;

PROC TRANSPOSE DATA=OUT&YRDIG OUT=OUT&YEAR;
  VAR SUM;
  ID GROUP;
RUN;

FILENAME ssdival DDE "EXCEL|Table21!R&ROWYEAR.C3:R&ROWYEAR.C20";

DATA _NULL_;
  SET OUT&YEAR;
  FILE ssdival NOTAB LRECL=200;
  PUT CONGEN_ANOM '09'X ENDOCRINE '09'X INFECTIOUS '09'X INJURIES '09'X '09'X MENTAL_RET '09'X
MENTAL_OTHER '09'X
  NEOPLASMS '09'X BLOOD '09'X CIRCULATORY '09'X DIGESTIVE '09'X GENITO '09'X MUSCULO '09'X NERVOUS '09'X
  RESPIRATORY '09'X SKIN '09'X OTHER '09'X MISSING;
RUN;
%MEND;

%DIAGYR(2021,21,6);
```

## Appendix A.222

JCL/SAS Code: Table27up

```
/******  
/* PROGRAM: TABLE27.SAS */  
/* PURPOSE: TO READ DATA FOR 2021 INTO AN EXCEL SHELL FOR */  
/* COMPARISON TO TABLE 27 OF THE SSA DI TABLES */  
/******
```

```
LIBNAME IN 'N:\Project\51390_DAF21\Restricted\DC1\DAF21\Validation\SSDI\Data';
```

```
%let year=2021;
```

```
%let yr=21;
```

```
DATA TEMP;
```

```
SET IN.ST&year(rename = (state&yr.12 = PST&yr.12));
```

```
IF PST&yr.12='AL' THEN ROW=1;
```

```
ELSE IF PST&yr.12='AK' THEN ROW=2;
```

```
ELSE IF PST&yr.12='AZ' THEN ROW=3;
```

```
ELSE IF PST&yr.12='AR' THEN ROW=4;
```

```
ELSE IF PST&yr.12='CA' THEN ROW=5;
```

```
ELSE IF PST&yr.12='CO' THEN ROW=6;
```

```
ELSE IF PST&yr.12='CT' THEN ROW=7;
```

```
ELSE IF PST&yr.12='DE' THEN ROW=8;
```

```
ELSE IF PST&yr.12='DC' THEN ROW=9;
```

```
ELSE IF PST&yr.12='FL' THEN ROW=10;
```

```
ELSE IF PST&yr.12='GA' THEN ROW=11;
```

```
ELSE IF PST&yr.12='HI' THEN ROW=12;
```

```
ELSE IF PST&yr.12='ID' THEN ROW=13;
```

```
ELSE IF PST&yr.12='IL' THEN ROW=14;
```

```
ELSE IF PST&yr.12='IN' THEN ROW=15;
```

```
ELSE IF PST&yr.12='IA' THEN ROW=16;
```

```
ELSE IF PST&yr.12='KS' THEN ROW=17;
```

```
ELSE IF PST&yr.12='KY' THEN ROW=18;
```

```
ELSE IF PST&yr.12='LA' THEN ROW=19;
```

```
ELSE IF PST&yr.12='ME' THEN ROW=20;
```

```
ELSE IF PST&yr.12='MD' THEN ROW=21;
```

```
ELSE IF PST&yr.12='MA' THEN ROW=22;
```

```
ELSE IF PST&yr.12='MI' THEN ROW=23;
```

```
ELSE IF PST&yr.12='MN' THEN ROW=24;
```

```
ELSE IF PST&yr.12='MS' THEN ROW=25;
```

```
ELSE IF PST&yr.12='MO' THEN ROW=26;
```

```
ELSE IF PST&yr.12='MT' THEN ROW=27;
```

```
ELSE IF PST&yr.12='NE' THEN ROW=28;
```

```
ELSE IF PST&yr.12='NV' THEN ROW=29;
```

```
ELSE IF PST&yr.12='NH' THEN ROW=30;
```

```
ELSE IF PST&yr.12='NJ' THEN ROW=31;
```



## Appendix A.222

JCL/SAS Code: Table27up

```
ELSE IF PST&yr.12='NM' THEN ROW=32;
ELSE IF PST&yr.12='NY' THEN ROW=33;
ELSE IF PST&yr.12='NC' THEN ROW=34;
ELSE IF PST&yr.12='ND' THEN ROW=35;
ELSE IF PST&yr.12='OH' THEN ROW=36;
ELSE IF PST&yr.12='OK' THEN ROW=37;
ELSE IF PST&yr.12='OR' THEN ROW=38;
ELSE IF PST&yr.12='PA' THEN ROW=39;
ELSE IF PST&yr.12='RI' THEN ROW=40;
ELSE IF PST&yr.12='SC' THEN ROW=41;
ELSE IF PST&yr.12='SD' THEN ROW=42;
ELSE IF PST&yr.12='TN' THEN ROW=43;
ELSE IF PST&yr.12='TX' THEN ROW=44;
ELSE IF PST&yr.12='UT' THEN ROW=45;
ELSE IF PST&yr.12='VT' THEN ROW=46;
ELSE IF PST&yr.12='VA' THEN ROW=47;
ELSE IF PST&yr.12='WA' THEN ROW=48;
ELSE IF PST&yr.12='WV' THEN ROW=49;
ELSE IF PST&yr.12='WI' THEN ROW=50;
ELSE IF PST&yr.12='WY' THEN ROW=51;
ELSE IF PST&yr.12='GU' THEN ROW=54;
ELSE IF PST&yr.12='MP' THEN ROW=55;
ELSE IF PST&yr.12='PR' THEN ROW=56;
ELSE IF PST&yr.12='VI' THEN ROW=57;
ELSE IF PST&yr.12=' ' THEN ROW=58;
```

```
RUN;
```

```
DATA TEMP_F;
  SET TEMP;
  IF SEX='F';
  IF AGEGR IN ('1','2','3','4','5','6','7','*');
RUN;
```

```
PROC SORT DATA=TEMP_F;
  BY ROW;
RUN;
```

```
PROC SUMMARY DATA=TEMP_F SUM;
  BY ROW;
  VAR COUNT;
  OUTPUT OUT=OUT_F SUM=SUM;
RUN;
```

```
FILENAME ssdival DDE "EXCEL|Table27!R126C2:R176C2";
```

Appendix A.222

JCL/SAS Code: Table27up

```
DATA _NULL_;
  SET OUT_F;
  FILE ssdival NOTAB LRECL=200;
  IF 1 <= ROW <= 51;
  PUT SUM;
RUN;

FILENAME ssdival DDE "EXCEL|Table27!R179C2:R183C2";
```

```
DATA _NULL_;
  SET OUT_F;
  FILE ssdival NOTAB LRECL=200;
  IF 54 <= ROW <= 58;
  PUT SUM;
RUN;
```

```
DATA TEMP_M;
  SET TEMP;
  IF SEX='M';
  IF AGEGR IN ('1','2','3','4','5','6','7','*');
RUN;
```

```
PROC SORT DATA=TEMP_M;
  BY ROW;
RUN;
```

```
PROC SUMMARY DATA=TEMP_M SUM;
  BY ROW;
  VAR COUNT;
  OUTPUT OUT=OUT_M SUM=SUM;
RUN;
```

```
FILENAME ssdival DDE "EXCEL|Table27!R66C2:R116C2";
```

```
DATA _NULL_;
  SET OUT_M;
  FILE ssdival NOTAB LRECL=200;
  IF 1 <= ROW <= 51;
  PUT SUM;
RUN;
```

```
FILENAME ssdival DDE "EXCEL|Table27!R119C2:R123C2";
```

```
DATA _NULL_;
```

Appendix A.222

JCL/SAS Code: Table27up

```
SET OUT_M;  
FILE ssdival NOTAB LRECL=200;  
IF 54 <= ROW <= 58;  
PUT SUM;  
RUN;
```

## Appendix A.223

JCL/SAS Code: TKTVAL

```

//#6502TKV JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#6502
//*****
//* PROJECT: DAF21
//* LEADER: MATT URATO
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.#6502.DAF21.TKTVAL.PRDLIB(TKTVAL) |
//* *-----*
//*
//* PURPOSE: COMPARE DAFYY TICKET BASE FILE AGAINST MAXIMUS
//* REPORT FOR VALIDATION PURPOSES
//*
//* DATE: 11/05/2018 SERGE LUKASHANETS
//*
//* INPUT:  TKT BASE: OPDR.TG.PRD.ETTW.DRAFT.DAF<YY>P.TICKETBS
//* ANN: OPDR.TG.PRD.ETTW.<ID>.DAF<YY>P.Y20<YY>.SA.V1
//*
//* OUTPUT: OPDR.TG.PRD.ETTW.<ID>.DAF<YY>E.TKTVAL.SA.V1(PST1712)
//*
//* AUTHOR: SERGE LUKASHANETS
//* CONTACT: SLUKASHANETS@MATHEMATICA-MPR.COM
//* PHONE #:
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//TBS   DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKETBS,DISP=SHR
//ANN   DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2021.SA.V1,DISP=SHR
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21E.TKTVAL.SA.V1,
//       DISP=(MOD,CATLG,DELETE),
//       SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//SYSIN DD *
*GET TICKET COUNTS FROM TICKETBASE AND ANNUALS DATA TO COMPARE TO
MAXIMUS REPORT;
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
%LET ANNYR=2021;*CURRENT YEAR FOR ANNUALS FILE;
%LET PSTYR=21;*CURRENT YEAR FOR PST VARIABLE;
%LET MAXRPTDT='30JUN2022';*DATE MAXIMUS ISSUED REPORT;

%MACRO TKTVAL;
DATA TBSTEST;
      SET TBS.TICKETBS(KEEP=SSN TKTASGNDDT: TKTUNASGDT: TKTERMDDT:

```

## Appendix A.223

JCL/SAS Code: TKTVL

```
            TKT_NOE PROVTYPE: PMTTYPER);
ARRAY ATKTASGNDDT(*) TKTASGNDDT;
ARRAY ATKTUNASGDT(*) TKTUNASGDT;
ARRAY ATKTTERMDDT(*) TKTTERMDDT;
ARRAY APROVTYPE(*) PROVTYPE;
ARRAY APMTTYPER(*) PMTTYPER;
IF .<ATKTASGNDDT(TKT_NOE)<=&MAXRPTDT.D THEN DO;
    TKTASSGND=ATKTASGNDDT(TKT_NOE);
    TKTUASSGND=ATKTUNASGDT(TKT_NOE);
    TKTTERMD=ATKTTERMDDT(TKT_NOE);
    PROVIDER=APROVTYPE(TKT_NOE);
    PAYMENT=APMTTYPER(TKT_NOE);
END;
ELSE DO I=1 TO DIM(ATKTASGNDDT)
    WHILE(ATKTASGNDDT(I)<=&MAXRPTDT.D);
        TKTASSGND =ATKTASGNDDT(I);
        TKTUASSGND=ATKTUNASGDT(I);
        TKTTERMD =ATKTTERMDDT(I);
        PROVIDER =APROVTYPE(I);
        PAYMENT =APMTTYPER(I);
    END;
KEEP SSN TKTASSGND TKTUASSGND TKTTERMD PROVIDER PAYMENT;
IF (TKTASSGND <= &MAXRPTDT.D AND TKTASSGND NE .) AND
    (TKTUASSGND =. OR TKTUASSGND > &MAXRPTDT.D) AND
    (TKTTERMD =. OR TKTTERMD > &MAXRPTDT.D);
RUN;

DATA ANNTEST;
SET ANN.Y&ANNYR(KEEP=SSN PST&PSTYR.12);
RENAME PST&PSTYR.12=STATE&PSTYR.12;
RUN;

DATA TBSANNTEST;
MERGE TBSTEST(IN=TBS) ANNTEST;
BY SSN;
IF TBS;
ENM=(PROVIDER='E' AND PAYMENT='M');
ENO=(PROVIDER='E' AND PAYMENT='O');
VRV=(PROVIDER='V' AND PAYMENT='V');
VRV2=(PROVIDER='E' AND PAYMENT='V');
VRO=(PROVIDER='V' AND PAYMENT='O');
VRM=(PROVIDER='V' AND PAYMENT='M');
KEEP STATE&PSTYR.12 ENM ENO VRV VRV2 VRO VRM PROVIDER PAYMENT;
RUN;
```

Appendix A.223

JCL/SAS Code: TKTVAL

```
PROC FREQ DATA=TBSANNTTEST;  
TABLES PROVIDER*PAYMENT*ENM*ENO*VRV*VRV2*VRO*VRM / LIST MISSING;  
RUN;
```

```
PROC SORT DATA=TBSANNTTEST;  
  BY STATE&PSTYR.12;  
RUN;
```

```
PROC MEANS DATA=TBSANNTTEST SUM NOPRINT;  
  VAR ENM ENO VRV VRV2 VRO VRM;  
  BY STATE&PSTYR.12;  
OUTPUT OUT=OUT.TKTVAL(DROP=_TYPE_) SUM= /AUTONAME;  
RUN;
```

```
%MEND TKTVAL;
```

```
%TKTVAL;
```

Appendix A.224

JCL/SAS Code: Download\_TKTVAL

```

*****;
/* Use the firstobs and obs control only if needed. For example, if the first download */
/* after the 499th record, then the code below will start downloading at the 500th */
/* observation and go until it completes or crashes. */
*****;
libname local 'M:\DAF21\Validation\Ticket\Output';
options comamid=tcp; %let SY56=SY56.sspf.ssa.gov; %let tcpsec=_prompt_; options remote=sy56.spawner; run;

SIGNON ;
*****NOTE THAT DSN COLUMN IS FORMATED WITH " SO ONLY ADD NAMES;
*          V V V V V V V V V V V V V V V V V V V V V ;
rsubmit;
;

libname MISF4 'OPDR.TG.PRD.ETTW.#6502.DAF21E.TKTVAL.SA.V1';

proc download data=MISF4.TKTVAL out=local.TKTVAL2112_v3(compress=binary);
run;

```

Appendix A.225

JCL/SAS Code: DOWNLOAD\_DATA

```
options comamid=tcp; %let SY56=SY56.sspf.ssa.gov; %let tcpsec=_prompt_; options remote=sy56.spawner; run;
```

```
LIBNAME LOCAL "M:\DAF21\Validation\VRRMS";
```

```
SIGNON;
```

```
RSUBMIT;
```

```
LIBNAME LOC 'OPDR.TG.PRD.ETTW.DRAFT.DAF21P.VRRMS';
```

```
PROC DOWNLOAD DATA=LOC.VRRMS
```

```
    OUT=LOCAL.VRRMS(COMPRESS=BINARY);
```

```
RUN;
```

```
ENDRSUBMIT;
```

```
SIGNOFF;
```



## Appendix A.226

JCL/SAS Code: VRRMS\_Validation

```

/*****
*** VALIDATE DAF15 VRRMS Data
*** PROGRAMMER : MIAOMIAO SHEN
*** DATE      : 10/13/2016
*****/
libname in "M:\DAF21\Task 16 Create VRRMS Component\Version2\Data\Final";
libname out "M:\DAF21\Task 16 Create VRRMS Component\Version2\QA";

OPTIONS NOCENTER OBS=MAX FORMDLIM=' ' PAGENO=1 LS=200 PS=80
        FORMCHAR="|----|+|---+=|-\<>*" ;

data vrrms_wide;
set in.vrrms;

*Exclude the claim with no Total Amount of Dollars Allowed;
if TOT_PAID_AMT1 ~= .;

*Construct the Federal Fiscal Year based on the dispatch date;
*For example, Fiscal Year 2012 is between 10/1/2011 and 9/30/2012;

array disp_dte(7) PSTUS_TS1-PSTUS_TS7;    *Input array;
array fc_yr(7) fiscal_yr1-fiscal_yr7;     *Output array;

do i = 1 to 7;
if month(disp_dte(i)) in (10, 11, 12) then fc_yr(i) = year(disp_dte(i))+1;
else fc_yr(i) = year(disp_dte(i));
end;
run;

* Transpose data from wide to long;
proc transpose data = vrrms_wide out = long_tlamt prefix = tl_amt_allowed;
by SSN;
var TOT_PAID_AMT1-TOT_PAID_AMT7;
run;

data long_tlamt;
set long_tlamt (rename=(tl_amt_allowed1=tl_amt_allowed));
count = input(substr(_name_,13),5.);
drop _name_;
run;

proc transpose data = vrrms_wide out = long_fcyr prefix = fiscal_yr;
by SSN;
var fiscal_yr1-fiscal_yr7;

```

## Appendix A.226

JCL/SAS Code: VRRMS\_Validation

```
run;

data long_fcy;
  set long_fcy (rename=(fiscal_yr1=fiscal_yr));
  count = input(substr(_name_,10),5.);
  drop _name_;
run;

* Merge Total Amount of Dollars Allowed and Fiscal Year two variables onto one data set;
data vrrms_long;
  merge long_tlamt
        long_fcy
  ;
  by SSN count;
run;

* Check if Total Amount of Dollars Allowed and Fiscal Year two variables have the same missingness;
proc format;
  value miss low-high ="nonmissing" other="missing";
run;

proc freq data = vrrms_long;
  tables tl_amt_allowed*fiscal_yr/missing list;
  format tl_amt_allowed fiscal_yr miss.;
run;

* Remove the records with missing Total Amount of Dollars Allowed and Fiscal Year;
data vrrms_long;
  set vrrms_long (where = (tl_amt_allowed~= . and fiscal_yr~= .));
run;

proc sort data = vrrms_long;
  by fiscal_yr;
run;

proc means data = vrrms_long n sum mean /*noprnt*/;
  var tl_amt_allowed;
  by fiscal_yr;
  output out=vrrms_validation_table1
         (drop=_TYPE__FREQ_
          rename=(tl_amt_allowed=number_of_claims_allowed))
         n=;
run;

proc means data = vrrms_long n sum mean /*noprnt*/;
```

## Appendix A.226

JCL/SAS Code: VRRMS\_Validation

```
var tl_amt_allowed;
by fiscal_yr;
output out=vrms_validation_table2
  (drop=_TYPE__FREQ_
   rename=(tl_amt_allowed=amount_of_dollars_allowed))
  sum=;
run;

proc means data = vrms_long n sum mean /*noprint*/;
var tl_amt_allowed;
by fiscal_yr;
output out=vrms_validation_table3
  (drop=_TYPE__FREQ_
   rename=(tl_amt_allowed=average_cost_per_claim))
  mean=;
run;

data out.vrms_validation_table;
  merge vrms_validation_table1 vrms_validation_table2 vrms_validation_table3;
  amount_of_dollars_allowed=round(amount_of_dollars_allowed,0.01);
  average_cost_per_claim=round(average_cost_per_claim,0.01);
  by fiscal_yr;
run;

title "DAF21 VRRMS Validation Table";
proc export data=out.vrms_validation_table
  outfile="M:\DAF21\Task 16 Create VRRMS
Component\Version2\QA\vrms_validation_table_DAF21_updated.xlsx"
  dbms=xlsx replace;
  sheet="vrms_validation_table";
run;
title;
```

## Appendix A.227

JCL/SAS Code: MAKEDATA

```

//$4671VAL JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT:  50214 DAF
//* DIRECTOR:  MATT URATO
//* PROGRAM:  MAKEDATA
//*
//* DESCRIPTION: MAKE ANALYTIC DATASET FOR VALIDATION
//*
//* INPUT:  ANNUAL FILE FOR YEAR OF INTEREST
//*
//* OUTPUT:  OPDR.TG.PRD.ETTW.$4671.VAL<YY>P.VALDATA.SA.V1
//*
//* CONTACT: BRITTANY COLIP
//* UPDATED: 01/14/2022
//*
//* PURPOSE: CREATE DATA THAT CAN BE USED TO RUN VALIDATION
//*  TABLES
//*****
//SAS EXEC SAS9,
//  WORK='120000,60000'
//*
//Y21  DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2021E,DISP=SHR
//DEM  DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.DMG,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.$4671.VAL21P.VALDATA.SA.V1,
//  DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,
//  DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//SYSIN DD *

```

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

```

* SET MASTER MACRO VARIABLES BELOW;
%LET YR = 21; /* 2-DIGIT YEAR OF INTEREST */
%LET DATE = '31DEC2021'D; /* LAST DAY OF THE YEAR OF INTEREST */
%LET NEXTYR = %EVAL(&YR. + 1); /*SETS TO THE NEXT YEAR*/
* VARIABLES TO KEEP FROM THE ANNUALS FILE;
%LET ANN_KEEPPARS = LAF&YR.12 PSTA&YR.12 PAYS&YR.12 STATE&YR.12
  DUES&YR.12 DUED&YR.12 PAYD&YR.12
  DXPRIBEST
  PROA&YR.12 PROB&YR.12 SSIERNLVL&YR.12

```

## Appendix A.227

JCL/SAS Code: MAKEDATA

```
T16GRSAMT&YR.12 T16SEAMT&YR.12
T16NETAMT&YR.12 T16PASAMT&YR.12
T16EXLAMT&YR.12 T16EXPAMT&YR.12
T16BEXPAMT&YR.12
/*INCLUDE SEPTEMBER VARS FOR SSI-TS13*/
PROA&YR.09 PROB&YR.09 SSIERNLVL&YR.09
PSTA&YR.09 T16GRSAMT&YR.09 T16SEAMT&YR.09
;
* VARIABLES TO KEEP FROM THE DMG FILE;
%LET DMG_KEEPPARS = SEX BIC DOBBEST DOEC MFT REPPYSSI REPPYSSD
COUNTRY_OF_BIRTH FOREIGN_BORN
PIA1-PIA50 PIED1-PIED50
CITIZEN_NUMI CITIZEN_CER
BLINDDT PAYEE_TYP_SSI;

* THIS IS THE FRA CALCULATION FROM THE SSDI VALIDATION PROGRAM;
%MACRO CALCFRA;

* ADJUST RETIREMENT DATE FOR BENEFICIARIES BORN ON JAN 1;
IF MONTH(DOBBEST)=1 AND DAY(DOBBEST)=1 THEN
  DOBRETIRE=INTNX('DAY',DOBBEST,-1);
ELSE DOBRETIRE=DOBBEST;

* SET MONTHS TO NUMBER OF MONTHS UNTIL FRA;
IF YEAR(DOBRETIRE)<=1937 THEN MONTHS=780;
IF YEAR(DOBRETIRE)=1938 THEN MONTHS=782;
IF YEAR(DOBRETIRE)=1939 THEN MONTHS=784;
IF YEAR(DOBRETIRE)=1940 THEN MONTHS=786;
IF YEAR(DOBRETIRE)=1941 THEN MONTHS=788;
IF YEAR(DOBRETIRE)=1942 THEN MONTHS=790;
IF 1943<=YEAR(DOBRETIRE)<=1954 THEN MONTHS=792;
IF YEAR(DOBRETIRE)=1955 THEN MONTHS=794;
IF YEAR(DOBRETIRE)=1956 THEN MONTHS=796;
IF YEAR(DOBRETIRE)=1957 THEN MONTHS=798;
IF YEAR(DOBRETIRE)=1958 THEN MONTHS=800;
IF YEAR(DOBRETIRE)=1959 THEN MONTHS=802;
IF YEAR(DOBRETIRE)>=1960 THEN MONTHS=804;

RETIREDATE=INTNX('MONTH',DOBRETIRE,MONTHS,'SAME DAY');

FRA=YRDIF(DOBRETIRE,RETIREDATE,'SAME DAY');

%MEND CALCFRA;
```

## Appendix A.227

JCL/SAS Code: MAKEDATA

```
* MACRO TO CALCULATE AGE, BASED ON SSDI VALIDATION PROGRAMS;  
%MACRO CALCAGE;
```

```
*** ADD AGE CALCULATION TO MATCH SSA ***;  
SELDATEMM=1;  
SELDATEYY=20&NEXTYR.;  
SELDATEDD=1;  
SELDATE2=MDY(SELDATEMM, SELDATEDD, SELDATEYY);  
MONTHS=INTCK('MONTH', DOBBEST, SELDATE2);  
DOBDD=DAY(DOBBEST);  
IF DOBDD>1 THEN MONTHS=MONTHS-1;  
AGE20&YR.=MONTHS/12;
```

```
%MEND CALCAGE;
```

```
* MACRO TO CREATE ANALYTIC DATASET;
```

```
%MACRO MAKEDATA;  
DATA OUT.VALIDATE;  
MERGE  
  Y&YR..Y20&YR.(IN=ANN KEEP=SSN  
    &ANN_KEEPVARS.  
  )  
  DEM.DMG(IN=DEMO KEEP=SSN  
    &DMG_KEEPVARS.  
  )  
;  
BY SSN;  
IF ANN AND DEMO;
```

```
/** SOME FLAGS USED IN SSDI DEFINITIONS **/  
/* THIS WILL LIKELY NOT BE USED, BUT KEEPING HERE IN CASE */  
/* FLAG FOR DOEC */  
IF . < DOEC < &DATE. THEN DOECFLAG = 1;  
ELSE DOECFLAG = 0;
```

```
/* AGE */  
%CALCAGE;
```

```
/* FULL RETIREMENT AGE */ /* FROM STEP1DI.TXT */  
%CALCFRA;  
FRA_EXCLUDE = (AGE20&YR.>=FRA);
```

```
/** FLAG DI BENES IN CURRENT PAY STATUS IN DECEMBER ***/
```

## Appendix A.227

JCL/SAS Code: MAKEDATA

```

IF SUBSTR(LEFT(LAF&YR.12),1,1) IN ('C','E')
  AND FRA_EXCLUDE ^= 1 /*AND DOECFLAG = 1*/
  THEN SSDI_BENE = 1;
ELSE SSDI_BENE = 0;

/**** FLAG SSDI DISABLED WORKERS *****/
IF BIC='A' AND SSDI_BENE = 1 THEN SSDI_DISABLED = 1;
ELSE SSDI_DISABLED = 0;

/**** FLAG SSDI WIDOW(ER)S *****/
IF SUBSTR(BIC,1,1) ='W'
  AND SSDI_BENE = 1 THEN SSDI_WIDOW = 1;
ELSE SSDI_WIDOW = 0;

/**** FLAG SSDI ADULT CHILDREN *****/
IF SUBSTR(BIC,1,1) ='C'
  AND SSDI_BENE = 1 THEN SSDI_ADULTCHILD = 1;
ELSE SSDI_ADULTCHILD = 0;

/**** FLAG OTHER SSDI CATEGORIES *****/
IF SSDI_DISABLED = 0 AND SSDI_WIDOW = 0 AND SSDI_ADULTCHILD=0
  AND SSDI_BENE = 1 THEN SSDI_OTHER = 1;
ELSE SSDI_OTHER = 0;

/***** FLAG SSI BENES *****/
IF PSTA&YR.12 IN ('C01','M01','M02') THEN SSI_BENE = 1;
ELSE SSI_BENE = 0;

/***** FLAG SSI ALL BLIND/DISABLED *****/
/**** THIS IS USED FOR THE UNIVERSE IN SSR-RELATED TABLES *****/
IF SSI_BENE OR PROB&YR.12=1 THEN SSR_BLIND_DISABLED =1;
ELSE SSR_BLIND_DISABLED=0;

/***** EARNINGS FOR FLAGGING SSI WORKING CATEGORIES *****/
/* EARNINGS CALCULATIONS BORROWED FROM DAWN */
ARRAY T16VARS (*) T16GRSAMT&YR.12 T16NETAMT&YR.12
  T16PASAMT&YR.12 T16EXLAMT&YR.12 T16EXPAMT&YR.12
  T16BEXPAMT&YR.12 T16SEAMT&YR.12;
DO I=1 TO DIM(T16VARS);
  IF T16VARS(I)=0 THEN T16VARS(I)=.;
END;

* DO NOT COUNT EARNINGS FOR BENES WHO ARE NOT IN CURRENT AND NOT
  PROB;
  IF PSTA&YR.12 NE 'C01' AND PROB&YR.12 NE 1 THEN DO;

```

## Appendix A.227

JCL/SAS Code: MAKEDATA

```
        DO I=1 TO DIM(T16VARS);
          IF T16VARS(I)=0 THEN T16VARS(I)=.;
        END;
      END;

* COMBINE WAGES AND SELF-EMP EARNINGS;
TOTERN=SUM(T16GRSAMT&YR.12,SUM(T16SEAMT&YR.12,T16NETAMT&YR.12*-1));

HAS_SSR=0;
IF TOTERN>0 THEN HAS_SSR=1;/* FLAG IF BENE HAS EARNINGS */

* CREATE PASS FOR WORKERS ONLY;
IF TOTERN=. OR PSTA&YR.12 NE 'C01' THEN DO;
  T16PASAMT&YR.12=.;
  T16EXPAMT&YR.12=.;
  T16BEXPAMT&YR.12=.;
END;

IF T16PASAMT&YR.12>0 THEN PASGRP=1;
IF T16EXPAMT&YR.12>0 THEN EXPGRP=1;
IF T16BEXPAMT&YR.12>0 THEN BEXPGRP=1;

/** SSI_WORKERS **/
/** PROA ****/
IF PROA&YR.12=1 AND PSTA&YR.12 IN ("C01","E01","N01")
  AND HAS_SSR=1 THEN WORKING_1619A=1;
ELSE WORKING_1619A=0;
/** PROB ****/
IF PROB&YR.12=1 AND PSTA&YR.12 IN ("C01","E01","N01")
  AND HAS_SSR=1 THEN WORKING_1619B=1;
ELSE WORKING_1619B=0;

/**** OTHER SSI WORKING ****/
IF PROA&YR.12=0 AND PROB&YR.12=0 AND
  (PSTA&YR.12 IN ("C01","E01","N01") OR SSI_BENE =1)
  AND HAS_SSR = 1
  THEN OTHER_SSI_WORKING = 1;
ELSE OTHER_SSI_WORKING=0;

/**** FLAG FOR ANY SSI WORKER *****/
IF WORKING_1619A=1 OR WORKING_1619B=1 OR OTHER_SSI_WORKING=1
  THEN SSI_WORKING = 1;
ELSE SSI_WORKING = 0;

/* STATE OF RESIDENCE */
```



## Appendix A.227

JCL/SAS Code: MAKEDATA

```
STATE = STATE&YR.12;

/* CITIZEN */

/* PRIMARY DIAGNOSIS GROUP */
%INCLUDE "OPDR.TG.PRD.ETTW.$4671.DAF&YR..VAL.PRDLIB(DXGROUPS)";

/* PRIMARY INSURANCE AMOUNT */
ARRAY PIA(*) PIA1-PIA50;
ARRAY PIED(*) PIED1-PIED50;
PIED_N = . ; /* INITIALIZE TO MISSING */
DO I = 1 TO 50;
  IF PIED(I) <= &DATE. AND NOT MISSING(PIED(I))
  THEN PIED_N = I;
END;
PRI_INS_AMT = .;
IF NOT MISSING(PIED_N) THEN PRI_INS_AMT=PIA(PIED_N);

RUN;
%MEND;

%MAKEDATA;

TITLE "PROC CONTENTS OF VALIDATION DATASET";
PROC CONTENTS DATA = OUT.VALIDATE; RUN;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM>
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM>
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM>
TO: <DPHELPS@MATHEMATICA-MPR.COM>
SUBJECT: VALIDATION DATA FOR DAF21 IS READY
```

## Appendix A.228

JCL/SAS Code: DXGROUPS

```
/** INFECTIONOUS/PARASITIC DISEASES */
IF DXPRIBEST>='0020' AND DXPRIBEST<='0189' THEN GROUP=3;
ELSE IF DXPRIBEST>='0200' AND DXPRIBEST<='0279' THEN GROUP=3;
ELSE IF DXPRIBEST>='0300' AND DXPRIBEST<='0419' THEN GROUP=3;
ELSE IF DXPRIBEST>='0430' AND DXPRIBEST<='0579' THEN GROUP=3;
ELSE IF DXPRIBEST>='0600' AND DXPRIBEST<='0669' THEN GROUP=3;
ELSE IF DXPRIBEST>='0700' AND DXPRIBEST<='0889' THEN GROUP=3;
ELSE IF DXPRIBEST>='0900' AND DXPRIBEST<='1049' THEN GROUP=3;
ELSE IF DXPRIBEST>='1100' AND DXPRIBEST<='1189' THEN GROUP=3;
ELSE IF DXPRIBEST>='1200' AND DXPRIBEST<='1359' THEN GROUP=3;
ELSE IF DXPRIBEST>='1370' AND DXPRIBEST<='1399' THEN GROUP=3;
ELSE IF DXPRIBEST>='7710' AND DXPRIBEST<='7719' THEN GROUP=3;

/** NEOPLASMS */
ELSE IF DXPRIBEST>='0420' AND DXPRIBEST<='0429' THEN GROUP=13;
ELSE IF DXPRIBEST>='1400' AND DXPRIBEST<='1659' THEN GROUP=13;
ELSE IF DXPRIBEST>='1700' AND DXPRIBEST<='1769' THEN GROUP=13;
ELSE IF DXPRIBEST>='1780' AND DXPRIBEST<='2089' THEN GROUP=13;
ELSE IF DXPRIBEST>='2100' AND DXPRIBEST<='2399' THEN GROUP=13;

/** ENDOCRINE */
ELSE IF DXPRIBEST>='2400' AND DXPRIBEST<='2469' THEN GROUP=2;
ELSE IF DXPRIBEST>='2500' AND DXPRIBEST<='2539' THEN GROUP=2;
ELSE IF DXPRIBEST>='2550' AND DXPRIBEST<='2559' THEN GROUP=2;
ELSE IF DXPRIBEST>='2600' AND DXPRIBEST<='2799' THEN GROUP=2;

/** DISEASES OF THE BLOOD */
ELSE IF DXPRIBEST>='2800' AND DXPRIBEST<='2899' THEN GROUP=14;
ELSE IF DXPRIBEST>='7720' AND DXPRIBEST<='7739' THEN GROUP=14;
ELSE IF DXPRIBEST>='7760' AND DXPRIBEST<='7769' THEN GROUP=14;

/** AUTISM */
ELSE IF DXPRIBEST>='2990' AND DXPRIBEST<='2999' THEN GROUP=5;

/** DEVELOPMENTAL DISORDERS */
ELSE IF DXPRIBEST>='3150' AND DXPRIBEST<='3159' THEN GROUP=6;

/** CHILDHOOD/ADOLESCENT DISORDERS */
ELSE IF DXPRIBEST>='3120' AND DXPRIBEST<='3139' THEN GROUP=7;

/** INTELLECTUAL DISABILITY */
ELSE IF DXPRIBEST>='3170' AND DXPRIBEST<='3194' THEN GROUP=8;
ELSE IF DXPRIBEST>='3196' AND DXPRIBEST<='3199' THEN GROUP=8;
```

## Appendix A.228

JCL/SAS Code: DXGROUPS

```
/** DEPRESSIVE, BIPOLAR DISORDERS */
ELSE IF DXPRIBEST>='2960' AND DXPRIBEST<='2969' THEN GROUP=9;
ELSE IF DXPRIBEST>='3110' AND DXPRIBEST<='3119' THEN GROUP=9;

/** NEUROCOGNITIVE MENTAL DISORDERS */
ELSE IF DXPRIBEST>='2900' AND DXPRIBEST<='2909' THEN GROUP=10;
ELSE IF DXPRIBEST>='2940' AND DXPRIBEST<='2949' THEN GROUP=10;
ELSE IF DXPRIBEST>='3100' AND DXPRIBEST<='3109' THEN GROUP=10;

/** SCHIZOPHRENIC */
ELSE IF DXPRIBEST>='2950' AND DXPRIBEST<='2959' THEN GROUP=11;
ELSE IF DXPRIBEST>='2970' AND DXPRIBEST<='2989' THEN GROUP=11;

/** OTHER MENTAL DISORDERS */
ELSE IF DXPRIBEST>='2910' AND DXPRIBEST<='2939' THEN GROUP=12;
ELSE IF DXPRIBEST>='3000' AND DXPRIBEST<='3099' THEN GROUP=12;
ELSE IF DXPRIBEST>='3140' AND DXPRIBEST<='3149' THEN GROUP=12;
ELSE IF DXPRIBEST>='3160' AND DXPRIBEST<='3169' THEN GROUP=12;
ELSE IF DXPRIBEST='3195' THEN GROUP=12;

/** NERVOUS SYSTEM */
ELSE IF DXPRIBEST>='3200' AND DXPRIBEST<='3269' THEN GROUP=19;
ELSE IF DXPRIBEST>='3290' AND DXPRIBEST<='3379' THEN GROUP=19;
ELSE IF DXPRIBEST>='3400' AND DXPRIBEST<='3749' THEN GROUP=19;
ELSE IF DXPRIBEST>='3760' AND DXPRIBEST<='3899' THEN GROUP=19;

/** CIRCULATORY */
ELSE IF DXPRIBEST>='3750' AND DXPRIBEST<='3759' THEN GROUP=15;
ELSE IF DXPRIBEST>='3900' AND DXPRIBEST<='3989' THEN GROUP=15;
ELSE IF DXPRIBEST>='4010' AND DXPRIBEST<='4059' THEN GROUP=15;
ELSE IF DXPRIBEST>='4100' AND DXPRIBEST<='4179' THEN GROUP=15;
ELSE IF DXPRIBEST>='4200' AND DXPRIBEST<='4389' THEN GROUP=15;
ELSE IF DXPRIBEST>='4400' AND DXPRIBEST<='4449' THEN GROUP=15;
ELSE IF DXPRIBEST>='4460' AND DXPRIBEST<='4489' THEN GROUP=15;
ELSE IF DXPRIBEST>='4510' AND DXPRIBEST<='4599' THEN GROUP=15;

/** RESPIRATORY */
ELSE IF DXPRIBEST>='4600' AND DXPRIBEST<='4669' THEN GROUP=20;
ELSE IF DXPRIBEST>='4700' AND DXPRIBEST<='4789' THEN GROUP=20;
ELSE IF DXPRIBEST>='4800' AND DXPRIBEST<='4879' THEN GROUP=20;
ELSE IF DXPRIBEST>='4900' AND DXPRIBEST<='4969' THEN GROUP=20;
ELSE IF DXPRIBEST>='5000' AND DXPRIBEST<='5089' THEN GROUP=20;
ELSE IF DXPRIBEST>='5100' AND DXPRIBEST<='5199' THEN GROUP=20;
ELSE IF DXPRIBEST>='7680' AND DXPRIBEST<='7709' THEN GROUP=20;
```

## Appendix A.228

JCL/SAS Code: DXGROUPS

```
/** DIGESTIVE **/  
ELSE IF DXPRIBEST>='5200' AND DXPRIBEST<='5379' THEN GROUP=16;  
ELSE IF DXPRIBEST>='5400' AND DXPRIBEST<='5439' THEN GROUP=16;  
ELSE IF DXPRIBEST>='5500' AND DXPRIBEST<='5539' THEN GROUP=16;  
ELSE IF DXPRIBEST>='5550' AND DXPRIBEST<='5589' THEN GROUP=16;  
ELSE IF DXPRIBEST>='5600' AND DXPRIBEST<='5609' THEN GROUP=16;  
ELSE IF DXPRIBEST>='5620' AND DXPRIBEST<='5629' THEN GROUP=16;  
ELSE IF DXPRIBEST>='5640' AND DXPRIBEST<='5799' THEN GROUP=16;  
ELSE IF DXPRIBEST>='7770' AND DXPRIBEST<='7779' THEN GROUP=16;  
  
/** GENITOURINARY **/  
ELSE IF DXPRIBEST>='5800' AND DXPRIBEST<='6089' THEN GROUP=17;  
ELSE IF DXPRIBEST>='6100' AND DXPRIBEST<='6119' THEN GROUP=17;  
ELSE IF DXPRIBEST>='6140' AND DXPRIBEST<='6299' THEN GROUP=17;  
  
/** SKIN **/  
ELSE IF DXPRIBEST>='6800' AND DXPRIBEST<='6869' THEN GROUP=21;  
ELSE IF DXPRIBEST>='6900' AND DXPRIBEST<='6989' THEN GROUP=21;  
ELSE IF DXPRIBEST>='7000' AND DXPRIBEST<='7099' THEN GROUP=21;  
ELSE IF DXPRIBEST>='7780' AND DXPRIBEST<='7789' THEN GROUP=21;  
  
/** MUSCULOSKELETAL **/  
ELSE IF DXPRIBEST>='7100' AND DXPRIBEST<='7399' THEN GROUP=18;  
  
/** CONGENITAL **/  
ELSE IF DXPRIBEST>='7400' AND DXPRIBEST<='7599' THEN GROUP=1;  
  
/** INJURIES **/  
ELSE IF DXPRIBEST>='8000' AND DXPRIBEST<='8489' THEN GROUP=4;  
ELSE IF DXPRIBEST>='8500' AND DXPRIBEST<='8549' THEN GROUP=4;  
ELSE IF DXPRIBEST>='8600' AND DXPRIBEST<='8879' THEN GROUP=4;  
ELSE IF DXPRIBEST>='8900' AND DXPRIBEST<='8979' THEN GROUP=4;  
ELSE IF DXPRIBEST>='9000' AND DXPRIBEST<='9059' THEN GROUP=4;  
ELSE IF DXPRIBEST>='9070' AND DXPRIBEST<='9099' THEN GROUP=4;  
ELSE IF DXPRIBEST>='9200' AND DXPRIBEST<='9299' THEN GROUP=4;  
ELSE IF DXPRIBEST>='9400' AND DXPRIBEST<='9599' THEN GROUP=4;  
  
/** OTHER **/  
ELSE IF DXPRIBEST>='7600' AND DXPRIBEST<='7609' THEN GROUP=22;  
ELSE IF DXPRIBEST>='7640' AND DXPRIBEST<='7669' THEN GROUP=22;  
ELSE IF DXPRIBEST>='7800' AND DXPRIBEST<='7809' THEN GROUP=22;  
ELSE IF DXPRIBEST>='7830' AND DXPRIBEST<='7849' THEN GROUP=22;  
ELSE IF DXPRIBEST='9330' THEN GROUP=22;  
  
/** UNKNOWN **/
```

## Appendix A.228

JCL/SAS Code: DXGROUPS

```
ELSE IF DXPRIBEST='' THEN GROUP=23;
ELSE IF DXPRIBEST>='0000' AND DXPRIBEST<='0019' THEN GROUP=23;
ELSE IF DXPRIBEST>='0190' AND DXPRIBEST<='0199' THEN GROUP=23;
ELSE IF DXPRIBEST>='0280' AND DXPRIBEST<='0299' THEN GROUP=23;
ELSE IF DXPRIBEST>='0580' AND DXPRIBEST<='0599' THEN GROUP=23;
ELSE IF DXPRIBEST>='0670' AND DXPRIBEST<='0699' THEN GROUP=23;
ELSE IF DXPRIBEST>='0890' AND DXPRIBEST<='0899' THEN GROUP=23;
ELSE IF DXPRIBEST>='1050' AND DXPRIBEST<='1099' THEN GROUP=23;
ELSE IF DXPRIBEST>='1190' AND DXPRIBEST<='1199' THEN GROUP=23;
ELSE IF DXPRIBEST>='1360' AND DXPRIBEST<='1369' THEN GROUP=23;
ELSE IF DXPRIBEST>='1660' AND DXPRIBEST<='1699' THEN GROUP=23;
ELSE IF DXPRIBEST>='1770' AND DXPRIBEST<='1779' THEN GROUP=23;
ELSE IF DXPRIBEST>='2090' AND DXPRIBEST<='2099' THEN GROUP=23;
ELSE IF DXPRIBEST>='2470' AND DXPRIBEST<='2499' THEN GROUP=23;
ELSE IF DXPRIBEST>='2540' AND DXPRIBEST<='2549' THEN GROUP=23;
ELSE IF DXPRIBEST>='2560' AND DXPRIBEST<='2599' THEN GROUP=23;
ELSE IF DXPRIBEST>='3270' AND DXPRIBEST<='3289' THEN GROUP=23;
ELSE IF DXPRIBEST>='3380' AND DXPRIBEST<='3399' THEN GROUP=23;
ELSE IF DXPRIBEST>='3990' AND DXPRIBEST<='4009' THEN GROUP=23;
ELSE IF DXPRIBEST>='4060' AND DXPRIBEST<='4099' THEN GROUP=23;
ELSE IF DXPRIBEST>='4180' AND DXPRIBEST<='4199' THEN GROUP=23;
ELSE IF DXPRIBEST>='4390' AND DXPRIBEST<='4399' THEN GROUP=23;
ELSE IF DXPRIBEST>='4450' AND DXPRIBEST<='4459' THEN GROUP=23;
ELSE IF DXPRIBEST>='4490' AND DXPRIBEST<='4509' THEN GROUP=23;
ELSE IF DXPRIBEST>='4670' AND DXPRIBEST<='4699' THEN GROUP=23;
ELSE IF DXPRIBEST>='4790' AND DXPRIBEST<='4799' THEN GROUP=23;
ELSE IF DXPRIBEST>='4880' AND DXPRIBEST<='4899' THEN GROUP=23;
ELSE IF DXPRIBEST>='4970' AND DXPRIBEST<='4999' THEN GROUP=23;
ELSE IF DXPRIBEST>='5090' AND DXPRIBEST<='5099' THEN GROUP=23;
ELSE IF DXPRIBEST>='5380' AND DXPRIBEST<='5399' THEN GROUP=23;
ELSE IF DXPRIBEST>='5440' AND DXPRIBEST<='5499' THEN GROUP=23;
ELSE IF DXPRIBEST>='5540' AND DXPRIBEST<='5549' THEN GROUP=23;
ELSE IF DXPRIBEST>='5590' AND DXPRIBEST<='5599' THEN GROUP=23;
ELSE IF DXPRIBEST>='5610' AND DXPRIBEST<='5619' THEN GROUP=23;
ELSE IF DXPRIBEST>='5630' AND DXPRIBEST<='5639' THEN GROUP=23;
ELSE IF DXPRIBEST>='6090' AND DXPRIBEST<='6099' THEN GROUP=23;
ELSE IF DXPRIBEST>='6120' AND DXPRIBEST<='6139' THEN GROUP=23;
ELSE IF DXPRIBEST>='6300' AND DXPRIBEST<='6799' THEN GROUP=23;
ELSE IF DXPRIBEST>='6870' AND DXPRIBEST<='6899' THEN GROUP=23;
ELSE IF DXPRIBEST>='6990' AND DXPRIBEST<='6999' THEN GROUP=23;
ELSE IF DXPRIBEST>='7610' AND DXPRIBEST<='7639' THEN GROUP=23;
ELSE IF DXPRIBEST>='7670' AND DXPRIBEST<='7679' THEN GROUP=23;
ELSE IF DXPRIBEST>='7790' AND DXPRIBEST<='7799' THEN GROUP=23;
ELSE IF DXPRIBEST>='7810' AND DXPRIBEST<='7829' THEN GROUP=23;
ELSE IF DXPRIBEST>='7850' AND DXPRIBEST<='7999' THEN GROUP=23;
```

Appendix A.228

JCL/SAS Code: DXGROUPS

```
ELSE IF DXPRIBEST>='8490' AND DXPRIBEST<='8499' THEN GROUP=23;
ELSE IF DXPRIBEST>='8550' AND DXPRIBEST<='8599' THEN GROUP=23;
ELSE IF DXPRIBEST>='8880' AND DXPRIBEST<='8899' THEN GROUP=23;
ELSE IF DXPRIBEST>='8980' AND DXPRIBEST<='8999' THEN GROUP=23;
ELSE IF DXPRIBEST>='9060' AND DXPRIBEST<='9069' THEN GROUP=23;
ELSE IF DXPRIBEST>='9100' AND DXPRIBEST<='9199' THEN GROUP=23;
ELSE IF DXPRIBEST>='9300' AND DXPRIBEST<='9329' THEN GROUP=23;
ELSE IF DXPRIBEST>='9331' AND DXPRIBEST<='9399' THEN GROUP=23;
ELSE IF DXPRIBEST>='9600' AND DXPRIBEST<='9999' THEN GROUP=23;
ELSE GROUP=23;
```

## Appendix A.229

JCL/SAS Code: MAKETBLS

```
//$4671TBL JOB (12510000,T715,,SAS,,ITC9FL),CARRILLO-PAZ,
//   MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT:   50214 DAF
//* DIRECTOR:  MATT URATO
//* PROGRAM:   MAKETBLS
//*
//* INPUT:  OPDR.TG.PRD.ETTW.$4671.VAL21P.VALDATA.SA.V1
//*
//* OUTPUT:  TABLE OUTPUT PRINTED TO LOG/LIST FILE
//*
//* CONTACT: BRITTANY COLIP
//* UPDATED: 12/20/2022
//*
//* PURPOSE: PRODUCE TABLES FOR VALIDATION
//*****
//*
//SAS EXEC SAS9,
//  WORK='120000,60000'
//*
//VAL DD DSN=OPDR.TG.PRD.ETTW.$4671.VAL21P.VALDATA.SA.V1,
//  DISP=SHR
//TEMP DD DSN=&&TEMPX,
//  DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,10)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN LS=MAX OBS=MAX;

* SET MASTER MACRO VARIABLES BELOW;
%LET YR = 21; /* 2-DIGIT YEAR OF INTEREST */
%LET DATE = '31DEC2021'D; /* LAST DAY OF THE YEAR OF INTEREST */
%LET DSN = VAL.VALIDATE ; /* NAME OF ANALYTIC DATASET */
/* LOAD FORMATS */
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(PROCFMT)";

/* LOAD STATE CODE LIST */
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(STATECD)";

/* RUN TABLE MACROS */
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(I1020304)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(I10506)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(I107)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(I10809)";
```

Appendix A.229

JCL/SAS Code: MAKETBLS

```
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II1011)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II12)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II13)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II14)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II16)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II17)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II18)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II19)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II20)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II21)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II22)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(II23)";
%INCLUDE "OPDR.TG.PRD.ETTW.#5413.DAF&YR..VAL.PRDLIB(SSITS5)";
```

```
/* *****
/* * SENDING EMAIL FROM HOST USING IEBGENER
/* *****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM>
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM>
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM>
TO: <DPHELPS@MATHEMATICA-MPR.COM>
SUBJECT: VALIDATION TABLES ARE READY
```



## Appendix A.230

JCL/SAS Code: PROC FMT

```
PROC FORMAT;
  VALUE AGE_GRP
    0-<25= "UNDER 25"
    25-<30= "25-29"
    30-<35= "30-34"
    35-<40= "35-39"
    40-<45= "40-44"
    45-<50= "45-49"
    50-<55= "50-54"
    55-<60= "55-59"
    60-<65 = "60-64"
    65-HIGH= "65-FRA"
  ;
  VALUE DITS4_AGE
    LOW-<35 = "UNDER 35"
    35-<45 = "35-44"
    45-<55 = "45-54"
    55-HIGH = "55-FRA"
  ;
  VALUE SSI_AGE
    0-<18 = "UNDER 18"
    18-<65 = "18-64"
    65-HIGH = "65 OR OLDER"
  ;
  VALUE SSI_WORKER_AGE
    0-<18 = "UNDER 18"
    18-<22 = "18-21"
    22-<26 = "22-25"
    26-<30 = "26-29"
    30-<40 = "30-39"
    40-<50 = "40-49"
    50-<60 = "50-59"
    60-<65 = "60-64"
    65-HIGH = "65 OR OLDER"
  ;
  VALUE $ BENE_FMT
    "SSDI_BENE" = "DISABLED BENEFICIARIES"
    "SSDI_DISABLED" = "DISABLED WORKERS"
    "SSI_BENE" = "SSI BENEFICIARY"
  ;
  VALUE $ STATE_FMT
    "AK" = "ALASKA"
    "AL" = "ALABAMA"
    "AR" = "ARKANSAS"
```

Appendix A.230

JCL/SAS Code: PROC FMT

"AZ" = "ARIZONA"  
"CA" = "CALIFORNIA"  
"CO" = "COLORADO"  
"CT" = "CONNECTICUT"  
"DC" = "DISTRICT OF COLUMBIA"  
"DE" = "DELAWARE"  
"FL" = "FLORIDA"  
"GA" = "GEORGIA"  
"GU" = "OUTLYING AREAS"  
"HI" = "HAWAII"  
"IA" = "IOWA"  
"ID" = "IDAHO"  
"IL" = "ILLINOIS"  
"IN" = "INDIANA"  
"KS" = "KANSAS"  
"KY" = "KENTUCKY"  
"LA" = "LOUISIANA"  
"MA" = "MASSACHUSETTS"  
"MD" = "MARYLAND"  
"ME" = "MAINE"  
"MI" = "MICHIGAN"  
"MN" = "MINNESOTA"  
"MO" = "MISSOURI"  
"MP" = "OUTLYING AREAS"  
"MS" = "MISSISSIPPI"  
"MT" = "MONTANA"  
"NC" = "NORTH CAROLINA"  
"ND" = "NORTH DAKOTA"  
"NE" = "NEBRASKA"  
"NH" = "NEW HAMPSHIRE"  
"NJ" = "NEW JERSEY"  
"NM" = "NEW MEXICO"  
"NV" = "NEVADA"  
"NY" = "NEW YORK"  
"OH" = "OHIO"  
"OK" = "OKLAHOMA"  
"OR" = "OREGON"  
"PA" = "PENNSYLVANIA"  
"PR" = "OUTLYING AREAS"  
"RI" = "RHODE ISLAND"  
"SC" = "SOUTH CAROLINA"  
"SD" = "SOUTH DAKOTA"  
"TN" = "TENNESSEE"  
"TX" = "TEXAS"  
"UT" = "UTAH"

## Appendix A.230

JCL/SAS Code: PROC FMT

```
"VA" = "VIRGINIA"  
"VI" = "OUTLYING AREAS"  
"VT" = "VERMONT"  
"WA" = "WASHINGTON"  
"WI" = "WISCONSIN"  
"WV" = "WEST VIRGINIA"  
"WY" = "WYOMING"  
;  
VALUE $ REGION_ST_FMT  
"CT" = "01: BOSTON - CONNECTICUT"  
"ME" = "01: BOSTON - MAINE"  
"MA" = "01: BOSTON - MASSACHUSETTS"  
"NH" = "01: BOSTON - NEW HAMPSHIRE"  
"RI" = "01: BOSTON - RHODE ISLAND"  
"VT" = "01: BOSTON - VERMONT"  
"NY" = "02: NEW YORK - NEW YORK"  
"NJ" = "02: NEW YORK - NEW JERSEY"  
"DE" = "03: PHILADELPHIA - DELAWARE"  
"DC" = "03: PHILADELPHIA - DISTRICT OF COLUMBIA"  
"MD" = "03: PHILADELPHIA - MARYLAND"  
"PA" = "03: PHILADELPHIA - PENNSYLVANIA"  
"VA" = "03: PHILADELPHIA - VIRGINIA"  
"WV" = "03: PHILADELPHIA - WEST VIRGINIA"  
"AL" = "04: ATLANTA - ALABAMA"  
"FL" = "04: ATLANTA - FLORIDA"  
"GA" = "04: ATLANTA - GEORGIA"  
"KY" = "04: ATLANTA - KENTUCKY"  
"MS" = "04: ATLANTA - MISSISSIPPI"  
"NC" = "04: ATLANTA - NORTH CAROLINA"  
"SC" = "04: ATLANTA - SOUTH CAROLINA"  
"TN" = "04: ATLANTA - TENNESSEE"  
"IL" = "05: CHICAGO - ILLINOIS"  
"IN" = "05: CHICAGO - INDIANA"  
"MI" = "05: CHICAGO - MICHIGAN"  
"MN" = "05: CHICAGO - MINNESOTA"  
"OH" = "05: CHICAGO - OHIO"  
"WI" = "05: CHICAGO - WISCONSIN"  
"AR" = "06: DALLAS - ARKANSAS"  
"LA" = "06: DALLAS - LOUISIANA"  
"NM" = "06: DALLAS - NEW MEXICO"  
"OK" = "06: DALLAS - OKLAHOMA"  
"TX" = "06: DALLAS - TEXAS"  
"IA" = "07: KANSAS CITY - IOWA"  
"KS" = "07: KANSAS CITY - KANSAS"  
"MO" = "07: KANSAS CITY - MISSOURI"
```

Appendix A.230

JCL/SAS Code: PROC FMT

```
"NE" = "07: KANSAS CITY - NEBRASKA"  
"CO" = "08: DENVER - COLORADO"  
"MT" = "08: DENVER - MONTANA"  
"ND" = "08: DENVER - NORTH DAKOTA"  
"SD" = "08: DENVER - SOUTH DAKOTA"  
"UT" = "08: DENVER - UTAH"  
"WY" = "08: DENVER - WYOMING"  
"AZ" = "09: SAN FRANCISCO - ARIZONA"  
"CA" = "09: SAN FRANCISCO - CALIFORNIA"  
"HI" = "09: SAN FRANCISCO - HAWAII"  
"NV" = "09: SAN FRANCISCO - NEVADA"  
"MP" = "09: SAN FRANCISCO - NORTHERN MARIANA ISLANDS"  
"AK" = "10: SEATTLE - ALASKA"  
"ID" = "10: SEATTLE - IDAHO"  
"OR" = "10: SEATTLE - OREGON"  
"WA" = "10: SEATTLE - WASHINGTON"  
"GU" = "OUTLYING AREAS"  
"PR" = "OUTLYING AREAS"  
"VI" = "OUTLYING AREAS"
```

;

VALUE DIAG

```
1="01: CONGENITAL"  
2="02: ENDOCRINE"  
3="03: INFECTIOUS"  
4="04: INJURIES"  
5="05: MENTAL--AUTISTIC"  
6="06: MENTAL--DEVELOPMENTAL"  
7="07: MENTAL--CHILDHOOD"  
8="08: MENTAL--INTELLECTUAL"  
9="09: MENTAL--DEPRESSIVE, BIPOLAR"  
10="10: MENTAL--NEUROCOGNITIVE"  
11="11: MENTAL--SCHIZOPHRENIC"  
12="12: MENTAL--OTHER"  
13="13: NEOPLASMS"  
14="14: BLOOD DISEASE"  
15="15: CIRCULATORY DISEASE"  
16="16: DIGESTIVE DISEASE"  
17="17: GENITURINARY DISEASE"  
18="18: MUSCULOSKELETAL DISEASE"  
19="19: NERVOUS SYSTEM DISEASE"  
20="20: RESPIRATORY DISEASE"  
21="21: SKIN DISEASE"  
22="22: OTHER"  
23="23: UNKNOWN"
```

;

## Appendix A.230

JCL/SAS Code: PROC FMT

## VALUE \$ COB

```
"CV" = "3: AFRICA - CAPE VERDE"  
"CH" = "4: ASIA - CHINA"  
"CU" = "2: LATIN AMERICA - CUBA"  
"DR" = "2: LATIN AMERICA - DOMINICAN REPUBLIC"  
"EG" = "5: MIDDLE EAST - EGYPT"  
"ES" = "2: LATIN AMERICA - EL SALVADOR"  
"ET" = "3: AFRICA - ETHIOPIA"  
"GM" = "7: EUROPE - GERMANY"  
"BZ" = "7: EUROPE - GERMANY"  
"GC" = "7: EUROPE - GERMANY"  
"GE" = "7: EUROPE - GERMANY"  
"HA" = "2: LATIN AMERICA - HAITI"  
"IN" = "4: ASIA - INDIA"  
"IR" = "5: MIDDLE EAST - IRAN"  
"IZ" = "5: MIDDLE EAST - IRAQ"  
"KS" = "4: ASIA - SOUTH KOREA"  
"LE" = "5: MIDDLE EAST - LEBANON"  
"LI" = "3: AFRICA - LIBERIA"  
"MX" = "2: LATIN AMERICA - MEXICO"  
"NI" = "3: AFRICA - NIGERIA"  
"RP" = "4: ASIA - PHILIPPINES"  
"PL" = "7: EUROPE - POLAND"  
"PO" = "7: EUROPE - PORTUGAL"  
"RQ" = "1: NORTH AMERICA - PUERTO RICO"  
"SO" = "3: AFRICA - SOMALIA"  
"SY" = "5: MIDDLE EAST - SYRIA"  
"UK" = "7: EUROPE - UNITED KINGDOM"  
"VM" = "4: ASIA - VIETNAM"  
"VN" = "4: ASIA - VIETNAM"  
"VS" = "4: ASIA - VIETNAM"  
"YO" = "7: EUROPE - FORMER YUGOSLAVIA"  
"BK" = "7: EUROPE - FORMER YUGOSLAVIA"  
"HR" = "7: EUROPE - FORMER YUGOSLAVIA"  
"MK" = "7: EUROPE - FORMER YUGOSLAVIA"  
"MJ" = "7: EUROPE - FORMER YUGOSLAVIA"  
"SI" = "7: EUROPE - FORMER YUGOSLAVIA"  
"RB" = "7: EUROPE - FORMER YUGOSLAVIA"  
"MW" = "7: EUROPE - FORMER YUGOSLAVIA"  
"SR" = "7: EUROPE - FORMER YUGOSLAVIA"  
/* OTHER BY REGION PER SSA */  
"CA" = "1: NORTH AMERICA - OTHER"  
"GL" = "1: NORTH AMERICA - OTHER"  
"SB" = "1: NORTH AMERICA - OTHER"  
"AA" = "2: LATIN AMERICA - OTHER"
```

Appendix A.230

JCL/SAS Code: PROC FMT

"AC" = "2: LATIN AMERICA - OTHER"  
"AR" = "2: LATIN AMERICA - OTHER"  
"AV" = "2: LATIN AMERICA - OTHER"  
"BB" = "2: LATIN AMERICA - OTHER"  
"BD" = "2: LATIN AMERICA - OTHER"  
"BF" = "2: LATIN AMERICA - OTHER"  
"BH" = "2: LATIN AMERICA - OTHER"  
"BL" = "2: LATIN AMERICA - OTHER"  
"BQ" = "2: LATIN AMERICA - OTHER"  
"BR" = "2: LATIN AMERICA - OTHER"  
"CI" = "2: LATIN AMERICA - OTHER"  
"CJ" = "2: LATIN AMERICA - OTHER"  
"CO" = "2: LATIN AMERICA - OTHER"  
"CS" = "2: LATIN AMERICA - OTHER"  
"DO" = "2: LATIN AMERICA - OTHER"  
"EC" = "2: LATIN AMERICA - OTHER"  
"FG" = "2: LATIN AMERICA - OTHER"  
"FK" = "2: LATIN AMERICA - OTHER"  
"GJ" = "2: LATIN AMERICA - OTHER"  
"GP" = "2: LATIN AMERICA - OTHER"  
"GT" = "2: LATIN AMERICA - OTHER"  
"GY" = "2: LATIN AMERICA - OTHER"  
"HO" = "2: LATIN AMERICA - OTHER"  
"IP" = "2: LATIN AMERICA - OTHER"  
"JM" = "2: LATIN AMERICA - OTHER"  
"MB" = "2: LATIN AMERICA - OTHER"  
"MH" = "2: LATIN AMERICA - OTHER"  
"NS" = "2: LATIN AMERICA - OTHER"  
"NT" = "2: LATIN AMERICA - OTHER"  
"NU" = "2: LATIN AMERICA - OTHER"  
"PA" = "2: LATIN AMERICA - OTHER"  
"PE" = "2: LATIN AMERICA - OTHER"  
"PM" = "2: LATIN AMERICA - OTHER"  
"SC" = "2: LATIN AMERICA - OTHER"  
"ST" = "2: LATIN AMERICA - OTHER"  
"TD" = "2: LATIN AMERICA - OTHER"  
"TK" = "2: LATIN AMERICA - OTHER"  
"UY" = "2: LATIN AMERICA - OTHER"  
"VC" = "2: LATIN AMERICA - OTHER"  
"VE" = "2: LATIN AMERICA - OTHER"  
"VI" = "2: LATIN AMERICA - OTHER"  
"AG" = "3: AFRICA - OTHER"  
"AO" = "3: AFRICA - OTHER"  
"BC" = "3: AFRICA - OTHER"  
"BN" = "3: AFRICA - OTHER"

Appendix A.230

JCL/SAS Code: PROC FMT

"BS" = "3: AFRICA - OTHER"  
"BY" = "3: AFRICA - OTHER"  
"CD" = "3: AFRICA - OTHER"  
"CF" = "3: AFRICA - OTHER"  
"CG" = "3: AFRICA - OTHER"  
"CM" = "3: AFRICA - OTHER"  
"CN" = "3: AFRICA - OTHER"  
"CT" = "3: AFRICA - OTHER"  
"DJ" = "3: AFRICA - OTHER"  
"EK" = "3: AFRICA - OTHER"  
"ER" = "3: AFRICA - OTHER"  
"EU" = "3: AFRICA - OTHER"  
"GA" = "3: AFRICA - OTHER"  
"GB" = "3: AFRICA - OTHER"  
"GH" = "3: AFRICA - OTHER"  
"GO" = "3: AFRICA - OTHER"  
"GV" = "3: AFRICA - OTHER"  
"HM" = "3: AFRICA - OTHER"  
"IV" = "3: AFRICA - OTHER"  
"JU" = "3: AFRICA - OTHER"  
"KE" = "3: AFRICA - OTHER"  
"LT" = "3: AFRICA - OTHER"  
"LY" = "3: AFRICA - OTHER"  
"MA" = "3: AFRICA - OTHER"  
"MF" = "3: AFRICA - OTHER"  
"MI" = "3: AFRICA - OTHER"  
"ML" = "3: AFRICA - OTHER"  
"MO" = "3: AFRICA - OTHER"  
"MP" = "3: AFRICA - OTHER"  
"MZ" = "3: AFRICA - OTHER"  
"NG" = "3: AFRICA - OTHER"  
"RE" = "3: AFRICA - OTHER"  
"ZI" = "3: AFRICA - OTHER"  
"RW" = "3: AFRICA - OTHER"  
"SE" = "3: AFRICA - OTHER"  
"SF" = "3: AFRICA - OTHER"  
"SG" = "3: AFRICA - OTHER"  
"SH" = "3: AFRICA - OTHER"  
"SL" = "3: AFRICA - OTHER"  
"SU" = "3: AFRICA - OTHER"  
"TO" = "3: AFRICA - OTHER"  
"TP" = "3: AFRICA - OTHER"  
"TS" = "3: AFRICA - OTHER"  
"TZ" = "3: AFRICA - OTHER"  
"UG" = "3: AFRICA - OTHER"

Appendix A.230

JCL/SAS Code: PROC FMT

"UV" = "3: AFRICA - OTHER"  
"WA" = "3: AFRICA - OTHER"  
"WI" = "3: AFRICA - OTHER"  
"WZ" = "3: AFRICA - OTHER"  
"ZA" = "3: AFRICA - OTHER"  
"AF" = "4: ASIA - OTHER"  
"BG" = "4: ASIA - OTHER"  
"BT" = "4: ASIA - OTHER"  
"BX" = "4: ASIA - OTHER"  
"CB" = "4: ASIA - OTHER"  
"CE" = "4: ASIA - OTHER"  
"HK" = "4: ASIA - OTHER"  
"ID" = "4: ASIA - OTHER"  
"IO" = "4: ASIA - OTHER"  
"JA" = "4: ASIA - OTHER"  
"KN" = "4: ASIA - OTHER"  
"LA" = "4: ASIA - OTHER"  
"MC" = "4: ASIA - OTHER"  
"MG" = "4: ASIA - OTHER"  
"BM" = "4: ASIA - OTHER"  
"MQ" = "4: ASIA - OTHER"  
"MV" = "4: ASIA - OTHER"  
"MY" = "4: ASIA - OTHER"  
"NP" = "4: ASIA - OTHER"  
"PF" = "4: ASIA - OTHER"  
"PK" = "4: ASIA - OTHER"  
"SN" = "4: ASIA - OTHER"  
"TE" = "4: ASIA - OTHER"  
"TH" = "4: ASIA - OTHER"  
"TW" = "4: ASIA - OTHER"  
"BA" = "5: MIDDLE EAST - OTHER"  
"CY" = "5: MIDDLE EAST - OTHER"  
"GZ" = "5: MIDDLE EAST - OTHER"  
"IS" = "5: MIDDLE EAST - OTHER"  
"IY" = "5: MIDDLE EAST - OTHER"  
"JO" = "5: MIDDLE EAST - OTHER"  
"KU" = "5: MIDDLE EAST - OTHER"  
"MU" = "5: MIDDLE EAST - OTHER"  
"QA" = "5: MIDDLE EAST - OTHER"  
"SA" = "5: MIDDLE EAST - OTHER"  
"TC" = "5: MIDDLE EAST - OTHER"  
"TU" = "5: MIDDLE EAST - OTHER"  
"WE" = "5: MIDDLE EAST - OTHER"  
"YE" = "5: MIDDLE EAST - OTHER"  
"AJ" = "6: FORMER SOVIET REPUBLICS"



Appendix A.230

JCL/SAS Code: PROC FMT

"AM" = "6: FORMER SOVIET REPUBLICS"  
"BO" = "6: FORMER SOVIET REPUBLICS"  
"EN" = "6: FORMER SOVIET REPUBLICS"  
"GG" = "6: FORMER SOVIET REPUBLICS"  
"KG" = "6: FORMER SOVIET REPUBLICS"  
"KZ" = "6: FORMER SOVIET REPUBLICS"  
"LG" = "6: FORMER SOVIET REPUBLICS"  
"LH" = "6: FORMER SOVIET REPUBLICS"  
"MD" = "6: FORMER SOVIET REPUBLICS"  
"RS" = "6: FORMER SOVIET REPUBLICS"  
"TI" = "6: FORMER SOVIET REPUBLICS"  
"TX" = "6: FORMER SOVIET REPUBLICS"  
"UP" = "6: FORMER SOVIET REPUBLICS"  
"UR" = "6: FORMER SOVIET REPUBLICS"  
"UZ" = "6: FORMER SOVIET REPUBLICS"  
"AL" = "7: EUROPE - OTHER"  
"AN" = "7: EUROPE - OTHER"  
"AU" = "7: EUROPE - OTHER"  
"BE" = "7: EUROPE - OTHER"  
"BU" = "7: EUROPE - OTHER"  
"DA" = "7: EUROPE - OTHER"  
"EZ" = "7: EUROPE - OTHER"  
"FI" = "7: EUROPE - OTHER"  
"FO" = "7: EUROPE - OTHER"  
"FR" = "7: EUROPE - OTHER"  
"GI" = "7: EUROPE - OTHER"  
"GK" = "7: EUROPE - OTHER"  
"GR" = "7: EUROPE - OTHER"  
"EI" = "7: EUROPE - OTHER"  
"HU" = "7: EUROPE - OTHER"  
"IC" = "7: EUROPE - OTHER"  
"IM" = "7: EUROPE - OTHER"  
"IT" = "7: EUROPE - OTHER"  
"JE" = "7: EUROPE - OTHER"  
"JN" = "7: EUROPE - OTHER"  
"LO" = "7: EUROPE - OTHER"  
"LS" = "7: EUROPE - OTHER"  
"LU" = "7: EUROPE - OTHER"  
"MN" = "7: EUROPE - OTHER"  
"MT" = "7: EUROPE - OTHER"  
"NL" = "7: EUROPE - OTHER"  
"NO" = "7: EUROPE - OTHER"  
"RO" = "7: EUROPE - OTHER"  
"SM" = "7: EUROPE - OTHER"  
"SV" = "7: EUROPE - OTHER"

## Appendix A.230

JCL/SAS Code: PROC FMT

```
"SW" = "7: EUROPE - OTHER"
"SZ" = "7: EUROPE - OTHER"
OTHER = "99: OTHER"
;
VALUE $ SEX
  "F" = "FEMALE"
  "M" = "MALE"
;
VALUE SSI_WRKR_EARN
  LOW-<66 = "01: $65 OR LESS"
  66-<100 = "02: $66-$99"
  100-<200 = "03: $100-$199"
  200-<300 = "04: $200-$299"
  300-<400 = "05: $300-$399"
  400-<500 = "06: $400-$499"
  500-<600 = "07: $500-$599"
  600-<700 = "08: $600-$699"
  700-<800 = "09: $700-$799"
  800-<900 = "10: $800-$899"
  900-<1000 = "11: $900-$999"
  1000-<1100 = "12: $1000-$1099"
  1100-<1200 = "13: $1100-$1199"
  1200-<1300 = "14: $1200-$1299"
  1300-<1400 = "15: $1300-$1399"
  1400-<1500 = "16: $1400-$1499"
  1500-<1600 = "17: $1500-$1599"
  1600-<1700 = "18: $1600-$1699"
  1700-<1800 = "19: $1700-$1799"
  1800-<1900 = "20: $1800-$1899"
  1900-HIGH = "21: $1900 OR MORE"
;
VALUE SSI_SECTAB_EARN
  LOW-<400 = "01: $400 OR LESS"
  400-<500 = "02: $400-$499"
  500-<600 = "03: $500-$599"
  600-<700 = "04: $600-$699"
  700-<800 = "05: $700-$799"
  800-<900 = "06: $800-$899"
  900-<1000 = "07: $900-$999"
  1000-<1100 = "08: $1000-$1099"
  1100-<1200 = "09: $1100-$1199"
  1200-<1300 = "10: $1200-$1299"
  1300-<1400 = "11: $1300-$1399"
  1400-<1500 = "12: $1400-$1499"
```

Appendix A.230

JCL/SAS Code: PROC FMT

```
1500-<1600= "13: $1500-$1599"  
1600-<1700= "14: $1600-$1699"  
1700-<1800= "15: $1700-$1799"  
1800-<1900= "16: $1800-$1899"  
1900-HIGH = "17: $1900 OR MORE"
```

```
;
```

```
RUN;
```

Appendix A.231

JCL/SAS Code: STATECD

```
%LET STATE_CODES = "AL" "AK" "AZ" "AR" "CA" "CO" "CT" "DE" "DC" "FL"  
"GA" "HI" "ID" "IL" "IN" "IA" "KS" "KY" "LA" "ME"  
"MD" "MA" "MI" "MN" "MS" "MO" "MT" "NE" "NV" "NH"  
"NJ" "NM" "NY" "NC" "ND" "OH" "OK" "OR" "PA" "RI"  
"SC" "SD" "TN" "TX" "UT" "VT" "VA" "WA" "WV" "WI"  
"WY"  
;
```

## Appendix A.232

JCL/SAS Code: II020304

```
%MACRO MAKEGLOBALVARS(  
    SSDI /* 0 OR 1 */  
    , SSI /* 0 OR 1 */  
    , BENE_TYPE /* EITHER DISABLED, WIDOW, ADULTCHILD, OR OTHER*/  
    , OUTVAR /* NAME OF THE OUTPUT MACRO VARIABLE */);  
  
%GLOBAL &OUTVAR.;  
SELECT  
SUM((SSDI_BENE=&SSDI.)*(SSDI_&BENE_TYPE.=1)*(SSI_BENE=&SSI.))  
    INTO :&OUTVAR. FROM &DSN  
    WHERE 18 <= AGE20&YR. < 65;  
  
%MEND MAKEGLOBALVARS;  
  
%MACRO MAKE_N_VARS(  
    SSDI /* 0 OR 1 */  
    , SSI /* 0 OR 1 */  
    , BENE_TYPE /* EITHER DISABLED, WIDOW, ADULTCHILD*/  
    , OUTVAR /* NAME OF THE OUTPUT DATASET VARIABLE */);  
  
    ,SUM((SSDI_BENE=&SSDI.)*(SSDI_&BENE_TYPE.=1)*(SSI_BENE=&SSI.))  
        AS &OUTVAR.  
  
%MEND MAKE_N_VARS;  
  
%MACRO MAKE_PERC_VARS(  
    SSDI /* 0 OR 1 */  
    , SSI /* 0 OR 1 */  
    , BENE_TYPE /* EITHER DISABLED, WIDOW, ADULTCHILD*/  
    , OUTVAR /* NAME OF THE OUTPUT DATASET VARIABLE */);  
  
%IF &SSDI = 1 AND &SSI = 0 %THEN %DO;  
    %LET GLOBALVAR = TOT_SSDI_&BENE_TYPE.;  
%END;  
%ELSE %IF &SSDI = 1 AND &SSI = 1 %THEN %DO;  
    %LET GLOBALVAR = TOT_BOTH_&BENE_TYPE.;  
%END;  
  
    ,SUM((SSDI_BENE=&SSDI.)*(SSDI_&BENE_TYPE.=1)*(SSI_BENE=&SSI.))  
        /&&&GLOBALVAR.*100  
        AS &OUTVAR. FORMAT = 8.1  
  
%MEND MAKE_PERC_VARS;
```

## Appendix A.232

JCL/SAS Code: II020304

```
%MACRO II_02_03_04;

PROC SQL NOPRINT;
  /* GET SUMS TO USE AS DENOMINATORS FOR PERCENTS */
  SELECT SUM((SSDI_BENE=1)*(SSI_BENE=0)) INTO :TOT_SSDI
    FROM &DSN
    WHERE 18 <= AGE20&YR. < 65;
  /* CREATE OTHER CATEGORIES FOR TOTALS */
  /* SSDI ONLY OTHER CATEGORIES */
  %MAKEGLOBALVARS(1, 0, DISABLED, TOT_SSDI_DISABLED);
  %MAKEGLOBALVARS(1, 0, WIDOW, TOT_SSDI_WIDOW);
  %MAKEGLOBALVARS(1, 0, ADULTCHILD, TOT_SSDI_ADULTCHILD);
  %MAKEGLOBALVARS(1, 0, OTHER, TOT_SSDI_OTHER);
  /* SSI ONLY TOTALS - NO SEPARATE CATEGORIES FOR THIS ONE */
  SELECT SUM((SSDI_BENE=0)*(SSI_BENE=1)) INTO :TOT_SSI
    FROM &DSN
    WHERE 18 <= AGE20&YR. < 65;
  /* BOTH SSI AND SSDI TOTALS */
  SELECT SUM((SSDI_BENE=1)*(SSI_BENE=1)) INTO :TOT_BOTH
    FROM &DSN
    WHERE 18 <= AGE20&YR. < 65;
  /* SEPARATE CATEGORIES FOR BOTH SSI AND SSDI */
  %MAKEGLOBALVARS(1, 1, DISABLED, TOT_BOTH_DISABLED);
  %MAKEGLOBALVARS(1, 1, WIDOW, TOT_BOTH_WIDOW);
  %MAKEGLOBALVARS(1, 1, ADULTCHILD, TOT_BOTH_ADULTCHILD);
  %MAKEGLOBALVARS(1, 1, OTHER, TOT_BOTH_OTHER);
QUIT;

PROC SQL;
  /**** OVERALL ROW FOR FREQ *****/
  SELECT
    "II.2, II.3, II.4" AS TABLE_SHELL
    ,"TOTAL" AS DX_GROUP
    ,SUM((SSDI_BENE=1)*(SSI_BENE=0)) AS NUM_SSDI_ONLY
    %MAKE_N_VARS(1,0, DISABLED, NUM_SSDI_ONLY_WORKERS)
    %MAKE_N_VARS(1,0, WIDOW, NUM_SSDI_ONLY_WIDOW)
    %MAKE_N_VARS(1,0, ADULTCHILD, NUM_SSDI_ONLY_ADULTCHILD)
    %MAKE_N_VARS(1,0, OTHER, NUM_SSDI_ONLY_OTHER)
    ,SUM((SSDI_BENE=0)*(SSI_BENE=1)) AS NUM_SSI_ONLY
    ,SUM((SSDI_BENE=1)*(SSI_BENE=1)) AS NUM_BOTH
    %MAKE_N_VARS(1,1, DISABLED, NUM_BOTH_WORKERS)
    %MAKE_N_VARS(1,1, WIDOW, NUM_BOTH_WIDOW)
    %MAKE_N_VARS(1,1, ADULTCHILD, NUM_BOTH_ADULTCHILD)
    %MAKE_N_VARS(1,1, OTHER, NUM_BOTH_OTHER)
  FROM &DSN
```

## Appendix A.232

JCL/SAS Code: II020304

```

WHERE 18 <= AGE20&YR. < 65

OUTER UNION CORR
/**** BY DX GROUP FOR FREQ *****/
SELECT
  "II.2, II.3, II.4" AS TABLE_SHELL
  ,PUT(GROUP,DIAG.) AS DX_GROUP
  ,SUM((SSDI_BENE=1)*(SSI_BENE=0)) AS NUM_SSDI_ONLY
  %MAKE_N_VARS(1,0, DISABLED,NUM_SSDI_ONLY_WORKERS)
  %MAKE_N_VARS(1,0, WIDOW,NUM_SSDI_ONLY_WIDOW)
  %MAKE_N_VARS(1,0, ADULTCHILD,NUM_SSDI_ONLY_ADULTCHILD)
  %MAKE_N_VARS(1,0, OTHER,NUM_SSDI_ONLY_OTHER)
  ,SUM((SSDI_BENE=0)*(SSI_BENE=1)) AS NUM_SSI_ONLY
  ,SUM((SSDI_BENE=1)*(SSI_BENE=1)) AS NUM_BOTH
  %MAKE_N_VARS(1,1, DISABLED,NUM_BOTH_WORKERS)
  %MAKE_N_VARS(1,1, WIDOW,NUM_BOTH_WIDOW)
  %MAKE_N_VARS(1,1, ADULTCHILD,NUM_BOTH_ADULTCHILD)
  %MAKE_N_VARS(1,1, OTHER,NUM_BOTH_OTHER)
FROM &DSN
WHERE 18 <= AGE20&YR. < 65
GROUP BY DX_GROUP
;
/**** OVERALL ROW FOR PERCENT ***/
SELECT
  "II.2, II.3, II.4" AS TABLE_SHELL
  ,"TOTAL" AS DX_GROUP
  ,SUM((SSDI_BENE=1)*(SSI_BENE=0))/&TOT_SSDI.*100
  AS PERC_SSDI_ONLY FORMAT = 8.1
  %MAKE_PERC_VARS(1,0,DISABLED,PERC_SSDI_ONLY_WORKERS)
  %MAKE_PERC_VARS(1,0, WIDOW,PERC_SSDI_ONLY_WIDOW)
  %MAKE_PERC_VARS(1,0, ADULTCHILD,PERC_SSDI_ONLY_ADULTCHILD)
  %MAKE_PERC_VARS(1,0, OTHER,PERC_SSDI_ONLY_OTHER)
  ,SUM((SSDI_BENE=0)*(SSI_BENE=1))/&TOT_SSI.*100
  AS PERC_SSI_ONLY FORMAT = 8.1
  ,SUM((SSDI_BENE=1)*(SSI_BENE=1))/&TOT_BOTH.*100
  AS PERC_BOTH FORMAT = 8.1
  %MAKE_PERC_VARS(1,1, DISABLED,PERC_BOTH_DISABLED)
  %MAKE_PERC_VARS(1,1, WIDOW,PERC_BOTH_WIDOW)
  %MAKE_PERC_VARS(1,1, ADULTCHILD,PERC_BOTH_ADULTCHILD)
  %MAKE_PERC_VARS(1,1, OTHER,PERC_BOTH_OTHER)
FROM &DSN
WHERE 18 <= AGE20&YR. < 65

OUTER UNION CORR
/**** BY DX GROUP FOR FREQ *****/

```

## Appendix A.232

JCL/SAS Code: II020304

```
SELECT
  "II.2, II.3, II.4" AS TABLE_SHELL
  ,PUT(GROUP,DIAG.) AS DX_GROUP
  ,SUM(((SSDI_BENE=1)*(SSI_BENE=0))/&TOT_SSDI.*100
    AS PERC_SSDI_ONLY FORMAT = 8.1
  %MAKE_PERC_VARS(1,0, DISABLED,PERC_SSDI_ONLY_WORKERS)
  %MAKE_PERC_VARS(1,0, WIDOW,PERC_SSDI_ONLY_WIDOW)
  %MAKE_PERC_VARS(1,0, ADULTCHILD,PERC_SSDI_ONLY_ADULTCHILD)
  %MAKE_PERC_VARS(1,0, OTHER,PERC_SSDI_ONLY_OTHER)
  ,SUM(((SSDI_BENE=0)*(SSI_BENE=1))/&TOT_SSI.*100
    AS PERC_SSI_ONLY FORMAT = 8.1
  ,SUM(((SSDI_BENE=1)*(SSI_BENE=1))/&TOT_BOTH.*100
    AS PERC_BOTH FORMAT = 8.1
  %MAKE_PERC_VARS(1,1, DISABLED,PERC_BOTH_DISABLED)
  %MAKE_PERC_VARS(1,1, WIDOW,PERC_BOTH_WIDOW)
  %MAKE_PERC_VARS(1,1, ADULTCHILD,PERC_BOTH_ADULTCHILD)
  %MAKE_PERC_VARS(1,1, OTHER,PERC_BOTH_OTHER)
FROM &DSN
WHERE 18 <= AGE20&YR. < 65
GROUP BY DX_GROUP
;
QUIT;

%MEND;

%II_02_03_04;
```



## Appendix A.233

JCL/SAS Code: II0506

```
%MACRO II_05_06(POP = /*NAME OF POPULATION VARIABLE TO SUBSET ON*/  
  ,SEX = TOTAL /*SEX CATEGORY, SET TO TOTAL BY DEFAULT*/  
  );
```

```
PROC SQL;  
  SELECT  
    "II.5, II.6" AS TABLE_SHELL  
    ,PUT("&POP.", $BENE_FMT.) AS POPULATION  
    ,PUT(AGE20&YR., AGE_GRP.) AS AGE  
    , "&SEX." AS GENDER  
    ,COUNT(*) AS NUMBER  
    ,AVG(PRI_INS_AMT) AS AVG_PIA  
    ,AVG(DUED&YR.12) AS AVG_MONTHLY_BENEFIT  
  FROM &DSN  
  WHERE &POP. = 1  
    %IF &SEX. ~= TOTAL %THEN %DO;  
    AND SEX = "&SEX."  
  %END;  
  GROUP BY AGE
```

```
/** ADD A ROW FOR TOTAL OF ALL AGE GROUPS ***/  
OUTER UNION CORR
```

```
SELECT  
  "II.5, II.6" AS TABLE_SHELL  
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION  
  , "ALL AGES" AS AGE  
  , "&SEX." AS GENDER  
  ,COUNT(*) AS NUMBER  
  ,AVG(PRI_INS_AMT) AS AVG_PIA  
  ,AVG(DUED&YR.12) AS AVG_MONTHLY_BENEFIT  
FROM &DSN  
WHERE &POP. = 1  
  %IF &SEX. ~= TOTAL %THEN %DO;  
  AND SEX = "&SEX."  
  %END;  
;  
QUIT;  
TITLE;
```

```
%MEND;
```

```
%II_05_06(POP = SSDI_BENE);  
%II_05_06(POP = SSDI_DISABLED);
```

Appendix A.233

JCL/SAS Code: II0506

```
%II_05_06(POP = SSDI_DISABLED, SEX = M);  
%II_05_06(POP = SSDI_DISABLED, SEX = F);
```

## Appendix A.234

JCL/SAS Code: II07

```
%MACRO II_07(POP = /*POPULATION TO RUN DATA ON*/);

PROC SQL;
  /**** OVERALL ROW ****/

  SELECT
    "II.7" AS TABLE_SHELL
    ,PUT("&POP.", $BENE_FMT.) AS POPULATION
    ,"ALL BENEFICIARIES" AS AGEGRP
    ,COUNT(*) AS NUM_BENES
    ,SUM(REPPYSSD) AS NUM_WITH_REPPY
    ,CALCULATED NUM_WITH_REPPY/CALCULATED NUM_BENES*100
      AS PERC_WITH_REPPY FORMAT =8.1
  FROM &DSN
  WHERE &POP. = 1

  OUTER UNION CORR
  /*** AGE GROUP ROWS *****/

  SELECT
    "II.7" AS TABLE_SHELL
    ,PUT("&POP.", $BENE_FMT.) AS POPULATION
    ,PUT(AGE20&YR.,DITS4_AGE. ) AS AGEGRP
    ,COUNT(*) AS NUM_BENES
    ,SUM(REPPYSSD) AS NUM_WITH_REPPY
    ,CALCULATED NUM_WITH_REPPY/CALCULATED NUM_BENES*100
      AS PERC_WITH_REPPY FORMAT =8.1
  FROM &DSN
  WHERE &POP. = 1
  GROUP BY AGEGRP
  ;
QUIT;

%MEND;

%II_07(POP = SSDI_BENE);
%II_07(POP = SSDI_DISABLED);
```

## Appendix A.235

JCL/SAS Code: II0809

```
%MACRO II_08_09(POP = /*POPULATION TO RUN DATA ON*/);
```

```
PROC SQL;
```

```
  /**** OVERALL ROW ****/
```

```
SELECT
```

```
  "II.8, II.9" AS TABLE_SHELL  
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION  
  , "ALL AREAS" AS STATE_AREA  
  ,COUNT(*) AS TOTAL  
  ,SUM(SS1_BENE) AS NUM_WITH_SSI  
  ,AVG(CASE  
    WHEN SSI_BENE THEN DUES&YR.12  
    ELSE . END) AS AVG_SSI_BENEFIT
```

```
FROM &DSN
```

```
WHERE &POP. = 1 AND 18 <= AGE20&YR. < 65  
AND NOT MISSING(STATE)
```

```
OUTER UNION CORR
```

```
 /*** STATES ROWS *****/
```

```
SELECT
```

```
  "II.8, II.9" AS TABLE_SHELL  
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION  
  ,PUT(STATE, $STATE_FMT.) AS STATE_AREA  
  ,COUNT(*) AS TOTAL  
  ,SUM(SS1_BENE) AS NUM_WITH_SSI  
  ,AVG(CASE  
    WHEN SSI_BENE THEN DUES&YR.12  
    ELSE . END) AS AVG_SSI_BENEFIT
```

```
FROM &DSN
```

```
WHERE &POP. = 1 AND 18 <= AGE20&YR. < 65  
AND NOT MISSING(STATE)  
AND STATE NOT IN ("GU", "MP", "PR", "VI")
```

```
GROUP BY STATE_AREA
```

```
OUTER UNION CORR
```

```
 /**** OTHER TERRITORIES ROW *****/
```

```
SELECT
```

```
  "II.8, II.9" AS TABLE_SHELL  
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION  
  ,PUT(STATE, $STATE_FMT.) AS STATE_AREA  
  ,COUNT(*) AS TOTAL  
  ,SUM(SS1_BENE) AS NUM_WITH_SSI
```

Appendix A.235

JCL/SAS Code: II0809

```
      ,AVG(CASE
        WHEN SSI_BENE THEN DUES&YR.12
        ELSE . END) AS AVG_SSI_BENEFIT
FROM &DSN
WHERE &POP. = 1 AND 18 <= AGE20&YR. < 65
      AND NOT MISSING(STATE)
      AND STATE IN ("GU","MP","PR","VI")
GROUP BY STATE_AREA
;
QUIT;

%MEND;

%II_08_09(POP = SSDI_BENE);
```

## Appendix A.236

JCL/SAS Code: II1011

```
%MACRO II_10_11(POP = /*POPULATION TO RUN DATA ON*/);
```

```
PROC SQL;
```

```
  /**** OVERALL ROW ****/
```

```
SELECT
```

```
  "II.10, II.11" AS TABLE_SHELL  
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION  
  , "ALL AREAS" AS STATE_AREA  
  ,AVG(CASE  
    WHEN SSI_BENE THEN DUES&YR.12  
    ELSE . END) AS AVG_PAY_TOTAL  
  ,AVG(CASE  
    WHEN SSI_BENE  
    AND AGE20&YR. < 18 THEN DUES&YR.12  
    ELSE . END) AS AVG_PAY_UNDER18  
  ,AVG(CASE  
    WHEN SSI_BENE  
    AND 18<=AGE20&YR.<65 THEN DUES&YR.12  
    ELSE . END) AS AVG_PAY_18TO64
```

```
FROM &DSN
```

```
WHERE &POP. = 1
```

```
  AND NOT MISSING(STATE)
```

```
OUTER UNION CORR
```

```
/** STATES ROWS *****/
```

```
SELECT
```

```
  "II.10, II.11" AS TABLE_SHELL  
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION  
  ,PUT(STATE, $STATE_FMT.) AS STATE_AREA  
  ,AVG(CASE  
    WHEN SSI_BENE THEN DUES&YR.12  
    ELSE . END) AS AVG_PAY_TOTAL  
  ,AVG(CASE  
    WHEN SSI_BENE  
    AND AGE20&YR. < 18 THEN DUES&YR.12  
    ELSE . END) AS AVG_PAY_UNDER18  
  ,AVG(CASE  
    WHEN SSI_BENE  
    AND 18<=AGE20&YR.<65 THEN DUES&YR.12  
    ELSE . END) AS AVG_PAY_18TO64
```

```
FROM &DSN
```

```
WHERE &POP. = 1
```

## Appendix A.236

JCL/SAS Code: II1011

```
AND NOT MISSING(STATE)
AND STATE NOT IN ("GU","MP","PR","VI")
GROUP BY STATE_AREA

OUTER UNION CORR
/**** OTHER TERRITORIES ROW *****/
SELECT
  "II.10, II.11" AS TABLE_SHELL
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION
  ,"OUTLYING AREA - NORTHERN MARIANA ISLANDS"
    AS STATE_AREA
  ,AVG(CASE
    WHEN SSI_BENE THEN DUES&YR.12
    ELSE . END) AS AVG_PAY_TOTAL
  ,AVG(CASE
    WHEN SSI_BENE
    AND AGE20&YR. < 18 THEN DUES&YR.12
    ELSE . END) AS AVG_PAY_UNDER18
  ,AVG(CASE
    WHEN SSI_BENE
    AND 18<=AGE20&YR.<65 THEN DUES&YR.12
    ELSE . END) AS AVG_PAY_18TO64
FROM &DSN
WHERE &POP. = 1
AND NOT MISSING(STATE)
AND STATE IN ("MP")
GROUP BY STATE_AREA
;
QUIT;

%MEND;

%II_10_11(POP = SSI_BENE);
```

## Appendix A.237

JCL/SAS Code: II12

```
%MACRO II_12(POP = /*POPULATION TO RUN DATA ON*/);

PROC SQL;
  /**** OVERALL ROW ****/

  SELECT
    "II.12" AS TABLE_SHELL
    ,PUT("&POP.", $BENE_FMT.) AS POPULATION
    ,"ALL BENEFICIARIES" AS AGEGRP
    ,COUNT(*) AS NUM_BENES
    ,SUM(REPPYSSI=1) AS NUM_WITH_REPPY
    ,SUM(REPPYSSI=0) AS NUM_WOUT_REPPY
    ,SUM(CALCULATED NUM_WITH_REPPY,CALCULATED NUM_WOUT_REPPY)
      /CALCULATED NUM_BENES*100
      AS PERC_TOTAL FORMAT =8.1
    ,CALCULATED NUM_WITH_REPPY/CALCULATED NUM_BENES*100
      AS PERC_WITH_REPPY FORMAT =8.1
    ,CALCULATED NUM_WOUT_REPPY/CALCULATED NUM_BENES*100
      AS PERC_WOUT_REPPY FORMAT =8.1
  FROM &DSN
  WHERE &POP. = 1 AND AGE20&YR. < 65

  OUTER UNION CORR
  /*** AGE GROUP ROWS *****/

  SELECT
    "II.12" AS TABLE_SHELL
    ,PUT("&POP.", $BENE_FMT.) AS POPULATION
    ,PUT(AGE20&YR., SSI_AGE.) AS AGEGRP
    ,COUNT(*) AS NUM_BENES
    ,SUM(REPPYSSI=1) AS NUM_WITH_REPPY
    ,SUM(REPPYSSI=0) AS NUM_WOUT_REPPY
    ,SUM(CALCULATED NUM_WITH_REPPY,CALCULATED NUM_WOUT_REPPY)
      /CALCULATED NUM_BENES*100
      AS PERC_TOTAL FORMAT =8.1
    ,CALCULATED NUM_WITH_REPPY/CALCULATED NUM_BENES*100
      AS PERC_WITH_REPPY FORMAT =8.1
    ,CALCULATED NUM_WOUT_REPPY/CALCULATED NUM_BENES*100
      AS PERC_WOUT_REPPY FORMAT =8.1
  FROM &DSN
  WHERE &POP. = 1 AND AGE20&YR. < 65
  GROUP BY AGEGRP
  ;
QUIT;
```



Appendix A.237

JCL/SAS Code: II12

```
%MEND;
```

```
%II_12(POP = SSI_BENE);
```

Appendix A.238

JCL/SAS Code: II13

```
%MACRO II_13(POP = /*POPULATION TO RUN DATA ON*/);
```

```
PROC SQL;
```

```
  /**** OVERALL ROW ****/
```

```
SELECT
```

```
  "II.13" AS TABLE_SHELL  
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION  
  , "ALL AREAS" AS REGION_STATE  
  ,SUM(SS1_BENE) AS NUM_SSI_CHILD  
FROM &DSN  
WHERE &POP. = 1  
  AND AGE20&YR. < 18  
  AND STATE IN (&STATE_CODES. "MP")
```

```
OUTER UNION CORR
```

```
 /*** STATES ROWS *****/
```

```
SELECT
```

```
  "II.13" AS TABLE_SHELL  
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION  
  ,PUT(STATE, $REGION_ST_FMT.) AS REGION_STATE  
  ,SUM(SS1_BENE) AS NUM_SSI_CHILD  
FROM &DSN  
WHERE &POP. = 1  
  AND AGE20&YR. < 18  
  AND STATE IN (&STATE_CODES. "MP")
```

```
GROUP BY REGION_STATE
```

```
;
```

```
QUIT;
```

```
%MEND;
```

```
%II_13(POP = SSI_BENE);
```

## Appendix A.239

JCL/SAS Code: II14

```
%MACRO II_14;

PROC SQL NOPRINT;
  /* GET SUMS TO USE AS DENOMINATORS FOR PERCENTS */
  SELECT SUM(SSI_BENE=1) INTO :TOT_UNDER18
    FROM &DSN
    WHERE AGE20&YR. < 18;
  SELECT SUM(SSI_BENE=1) INTO :TOT_UNDER3
    FROM &DSN
    WHERE AGE20&YR. < 3;
  SELECT SUM(SSI_BENE=1) INTO :TOT_3TO5
    FROM &DSN
    WHERE 3 <= AGE20&YR. < 6;
  SELECT SUM(SSI_BENE=1) INTO :TOT_6TO12
    FROM &DSN
    WHERE 6 <= AGE20&YR. < 13;
  SELECT SUM(SSI_BENE=1) INTO :TOT_13TO17
    FROM &DSN
    WHERE 13 <= AGE20&YR. < 18;
QUIT;

PROC SQL;
  /*** OVERALL ROW FOR FREQ *****/
  SELECT
    "II.14" AS TABLE_SHELL
    ,"TOTAL" AS DX_GROUP
    ,SUM(SSI_BENE=1) AS NUM_ALL_AGES
    ,SUM((SSI_BENE=1)*(AGE20&YR.<3))
      AS NUM_UNDER3
    ,SUM((SSI_BENE=1)*(3<=AGE20&YR.<6))
      AS NUM_3TO5
    ,SUM((SSI_BENE=1)*(6<=AGE20&YR.<13))
      AS NUM_6TO12
    ,SUM((SSI_BENE=1)*(13<=AGE20&YR.<18))
      AS NUM_13TO17
  FROM &DSN
  WHERE AGE20&YR. < 18

  OUTER UNION CORR
  /*** BY DX GROUP FOR FREQ *****/
  SELECT
    "II.14" AS TABLE_SHELL
    ,PUT(GROUP,DIAG.) AS DX_GROUP
    ,SUM(SSI_BENE=1) AS NUM_ALL_AGES
```

## Appendix A.239

JCL/SAS Code: II14

```

,SUM((SSI_BENE=1)*(AGE20&YR.<3))
  AS NUM_UNDER3
,SUM((SSI_BENE=1)*(3<=AGE20&YR.<6))
  AS NUM_3TO5
,SUM((SSI_BENE=1)*(6<=AGE20&YR.<13))
  AS NUM_6TO12
,SUM((SSI_BENE=1)*(13<=AGE20&YR.<18))
  AS NUM_13TO17
FROM &DSN
WHERE AGE20&YR. < 18
GROUP BY DX_GROUP
;
/**** OVERALL ROW FOR PERCENT ****/
SELECT
  "II.14" AS TABLE_SHELL
  ,"TOTAL" AS DX_GROUP
  ,SUM(SSI_BENE=1)/&TOT_UNDER18.*100
    AS PERC_ALL_AGES FORMAT = 8.1
  ,SUM((SSI_BENE=1)*(AGE20&YR.<3))/&TOT_UNDER3.*100
    AS PERC_UNDER3 FORMAT = 8.1
  ,SUM((SSI_BENE=1)*(3<=AGE20&YR.<6))/&TOT_3TO5.*100
    AS PERC_3TO5  FORMAT = 8.1
  ,SUM((SSI_BENE=1)*(6<=AGE20&YR.<13))/&TOT_6TO12.*100
    AS PERC_6TO12 FORMAT = 8.1
  ,SUM((SSI_BENE=1)*(13<=AGE20&YR.<18))/&TOT_13TO17.*100
    AS PERC_13TO17 FORMAT = 8.1
FROM &DSN
WHERE AGE20&YR. < 18

OUTER UNION CORR
/**** BY DX GROUP FOR FREQ *****/
SELECT
  "II.14" AS TABLE_SHELL
  ,PUT(GROUP,DIAG.) AS DX_GROUP
  ,SUM(SSI_BENE=1)/&TOT_UNDER18.*100
    AS PERC_ALL_AGES FORMAT = 8.1
  ,SUM((SSI_BENE=1)*(AGE20&YR.<3))/&TOT_UNDER3.*100
    AS PERC_UNDER3 FORMAT = 8.1
  ,SUM((SSI_BENE=1)*(3<=AGE20&YR.<6))/&TOT_3TO5.*100
    AS PERC_3TO5  FORMAT = 8.1
  ,SUM((SSI_BENE=1)*(6<=AGE20&YR.<13))/&TOT_6TO12.*100
    AS PERC_6TO12 FORMAT = 8.1
  ,SUM((SSI_BENE=1)*(13<=AGE20&YR.<18))/&TOT_13TO17.*100
    AS PERC_13TO17 FORMAT = 8.1
FROM &DSN

```

Appendix A.239

JCL/SAS Code: II14

```
WHERE AGE20&YR. < 18
GROUP BY DX_GROUP
;
QUIT;

%MEND;

%II_14;
```

## Appendix A.240

JCL/SAS Code: II16

```
%MACRO II_16(POP = /*POPULATION TO RUN DATA ON*/);
```

```
PROC SQL;
```

```
  /**** OVERALL ROW ****/
```

```
SELECT
```

```
  "II.16" AS TABLE_SHELL
```

```
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION
```

```
  , "ALL AREAS" AS COUNTRY_OF_ORIGIN
```

```
  , COUNT(*) AS TOTAL
```

```
  , SUM(AGE20&YR. < 18) AS UNDER_18
```

```
  , SUM(18<=AGE20&YR.<65) AS AGE_18TO64
```

```
FROM &DSN
```

```
WHERE &POP. = 1
```

```
  AND AGE20&YR. < 65
```

```
  AND ((FOREIGN_BORN=""
```

```
    AND COUNTRY_OF_BIRTH NOT IN (&STATE_CODES.))
```

```
    OR FOREIGN_BORN = "**")
```

```
OUTER UNION CORR
```

```
  /*** COUNTRY OF BIRTH ROWS*****/
```

```
SELECT
```

```
  "II.16" AS TABLE_SHELL
```

```
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION
```

```
  ,PUT(COUNTRY_OF_BIRTH, $COB.) AS COUNTRY_OF_ORIGIN
```

```
  , COUNT(*) AS TOTAL
```

```
  , SUM(AGE20&YR. < 18) AS UNDER_18
```

```
  , SUM(18<=AGE20&YR.<65) AS AGE_18TO64
```

```
FROM &DSN
```

```
WHERE &POP. = 1
```

```
  AND AGE20&YR. < 65
```

```
  AND ((FOREIGN_BORN=""
```

```
    AND COUNTRY_OF_BIRTH NOT IN (&STATE_CODES.))
```

```
    OR FOREIGN_BORN = "**")
```

```
GROUP BY COUNTRY_OF_ORIGIN
```

```
;
```

```
QUIT;
```

```
%MEND;
```

```
%II_16(POP = SSI_BENE);
```

Appendix A.241

JCL/SAS Code: II17

```
%MACRO II_17();

%LET WORKING=(OTHER_SSI_WORKING=1 OR PROA&YR.12=1 OR PROB&YR.12=1);

PROC SQL;

SELECT
  "II.17" AS TABLE_SHELL
  ,"DECEMBER 20&YR." AS MONTH
  ,SUM(SSR_BLIND_DISABLED=1) AS ALL_BLIND_DISABLED
  ,SUM(&WORKING.) AS TOTAL_SSI_WORKING
  ,SUM(WORKING_1619A=1 OR OTHER_SSI_WORKING=1)
    AS RECEIVING_SSI
  ,SUM(WORKING_1619B=1) AS SECTION_1619B
  ,CALCULATED TOTAL_SSI_WORKING/
    CALCULATED ALL_BLIND_DISABLED*100
    AS PERC_ALL FORMAT=8.1
FROM &DSN
QUIT;

%MEND;

%II_17();
```

## Appendix A.242

JCL/SAS Code: II18

```
%MACRO TS12_VARS(ERNVAR=,WRKR=1,SCTAB=1);

  /*DAF19 UPDATE -- SPLITTING UP EARNINGS CATEGORIES FOR SSI WORKERS
  AND 1619A/1619B. MACRO IS SET TO 1 FOR BOTH BY DEFAULT AND
  CAN BE CHANGED IN THE MACRO CALLS FOR THE TABLES; */
  %IF &WRKR. =1 %THEN %DO;
  /* SSI WORKERS */
  ,SUM(&WORKING.) AS TOTAL_SSI_WORKING
  ,CALCULATED TOTAL_SSI_WORKING/
    &TOTAL_SSI_WORKING*100
    AS PERC_WORKING FORMAT=8.1
  ,AVG(CASE WHEN &WORKING.
    THEN &ERNVAR.
    ELSE . END) AS SSI_WORKING_AVG_EARN
  %END;

  %IF &SCTAB. = 1 %THEN %DO;
  /* SECTION 1619 A */
  ,SUM(&S1619A.) AS SECTION_1619A
  ,CALCULATED SECTION_1619A/
    &SECTION_1619A*100
    AS PERC_1619A FORMAT=8.1
  ,AVG(CASE WHEN &S1619A.
    THEN &ERNVAR.
    ELSE . END) AS SECTION_1619A_AVG_EARN

  /* SECTION 1619 B */
  ,SUM(WORKING_1619B=1) AS SECTION_1619B
  ,CALCULATED SECTION_1619B/
    &SECTION_1619B*100
    AS PERC_1619B FORMAT=8.1
  ,AVG(CASE WHEN WORKING_1619B = 1
    THEN &ERNVAR.
    ELSE . END) AS SECTION_1619B_AVG_EARN
  %END;

%MEND TS12_VARS;

%MACRO II_18();

  %LET WORKING=(OTHER_SSI_WORKING=1 OR PROA&YR.12=1 OR PROB&YR.12=1);
  %LET S1619A = (SSI_BENE=1 AND WORKING_1619A=1);

  PROC SQL NOPRINT;
```



## Appendix A.242

JCL/SAS Code: II18

```
SELECT SUM(&WORKING.) INTO :TOTAL_SSI_WORKING
  FROM &DSN. ;
SELECT SUM(&S1619A.) INTO :SECTION_1619A
  FROM &DSN. ;
SELECT SUM(WORKING_1619B=1) INTO :SECTION_1619B
  FROM &DSN. ;
```

```
QUIT;
```

```
PROC SQL;
```

```
  /*** OVERALL ROW ***/
```

```
  SELECT
```

```
    "II.18" AS TABLE_SHELL
```

```
    ,"TOTAL" AS CHARACTERISTIC
```

```
    %TS12_VARS(ERNVAR=TOTERN)
```

```
  FROM &DSN.
```

```
  ;
```

```
QUIT;
```

```
PROC SQL;
```

```
  /**** AGE GROUP ROWS ***/
```

```
  SELECT
```

```
    "II.18" AS TABLE_SHELL
```

```
    ,"AGE: " || PUT(AGE20&YR., SSI_WORKER_AGE.)
```

```
    AS CHARACTERISTIC
```

```
    %TS12_VARS(ERNVAR=TOTERN)
```

```
  FROM &DSN
```

```
  GROUP BY CHARACTERISTIC
```

```
  ;
```

```
QUIT;
```

```
PROC SQL;
```

```
  /**** GENDER ROWS ***/
```

```
  SELECT
```

```
    "II.18" AS TABLE_SHELL
```

```
    ,"SEX: " || PUT(SEX,$SEX.)
```

```
    AS CHARACTERISTIC
```

```
    %TS12_VARS(ERNVAR=TOTERN)
```

```
  FROM &DSN
```

```
  GROUP BY CHARACTERISTIC
```

```
  ;
```

```
QUIT;
```

```
PROC SQL;
```

```
  /**** WAGES ROWS ***/
```

```
  SELECT
```

```
    "II.18" AS TABLE_SHELL
```

## Appendix A.242

JCL/SAS Code: II18

```
      ,"EARNED INCOME: WAGES"  
      AS CHARACTERISTIC  
      %TS12_VARS(ERNVAR=T16GRSAMT&YR.12)  
FROM &DSN  
WHERE T16GRSAMT&YR.12 > 0  
;  
QUIT;  
PROC SQL;  
  /**** SELF-EMPLOYMENT ROWS ****/  
  SELECT  
    "II.18" AS TABLE_SHELL  
    ,"EARNED INCOME: SELF-EMPLOYMENT"  
    AS CHARACTERISTIC  
    %TS12_VARS(ERNVAR=T16SEAMT&YR.12)  
  FROM &DSN  
  WHERE T16SEAMT&YR.12 > 0  
  ;  
QUIT;  
  
PROC SQL;  
  /**** EARNINGS CATEGORY ROWS ****/  
  /**** FOR SSI WORKERS ONLY ****/  
  SELECT  
    "II.18" AS TABLE_SHELL  
    ,"EARNINGS: "||  
    PUT(SUM(T16GRSAMT&YR.12,T16SEAMT&YR.12),SSI_WRKR_EARN.)  
    AS CHARACTERISTIC  
    %TS12_VARS(ERNVAR=TOTERN,SCTAB=0)  
  FROM &DSN  
  GROUP BY CHARACTERISTIC  
  ;  
QUIT;  
PROC SQL;  
  /**** EARNINGS CATEGORY ROWS ****/  
  /**** FOR SECT1619 A AND B ONLY****/  
  SELECT  
    "II.18" AS TABLE_SHELL  
    ,"EARNINGS: "||  
    PUT(SUM(T16GRSAMT&YR.12,T16SEAMT&YR.12),SSI_SECTAB_EARN.)  
    AS CHARACTERISTIC  
    %TS12_VARS(ERNVAR=TOTERN,WRKR=0)  
  FROM &DSN  
  GROUP BY CHARACTERISTIC  
  ;  
QUIT;
```

## Appendix A.242

JCL/SAS Code: II18

```
PROC SQL;
  /**** WORK INCENTIVES ROWS ****/
  SELECT
    "II.18" AS TABLE_SHELL
    ,"WORK INCENTIVES: SECTION 1619A"
    AS CHARACTERISTIC
    %TS12_VARS(ERNVAR=TOTERN)
  FROM &DSN
  WHERE &S1619A.
  ;
QUIT;
PROC SQL;
  SELECT
    "II.18" AS TABLE_SHELL
    ,"WORK INCENTIVES: SECTION 1619B"
    AS CHARACTERISTIC
    %TS12_VARS(ERNVAR=TOTERN)
  FROM &DSN
  WHERE WORKING_1619B=1
  ;
QUIT;
PROC SQL;
  SELECT
    "II.18" AS TABLE_SHELL
    ,"WORK INCENTIVES: PLAN TO ACHIEVE SELF-SUPPORT (PASS)"
    AS CHARACTERISTIC
    %TS12_VARS(ERNVAR=TOTERN)
  FROM &DSN
  WHERE PASGRP=1
  ;
QUIT;
PROC SQL;
  SELECT
    "II.18" AS TABLE_SHELL
    ,"WORK INCENTIVES: IMPAIRMENT RELATED WORK EXPENSE"
    AS CHARACTERISTIC
    %TS12_VARS(ERNVAR=TOTERN)
  FROM &DSN
  WHERE EXPGRP=1
  ;
QUIT;
PROC SQL;
  SELECT
    "II.18" AS TABLE_SHELL
    ,"WORK INCENTIVES: BLIND WORK EXPENSE"
```

Appendix A.242

JCL/SAS Code: II18

```
        AS CHARACTERISTIC
        %TS12_VARS(ERNVAR=TOTERN)
        FROM &DSN
        WHERE BEXPGRP=1
        ;
QUIT;

%MEND;

%II_18();
```

Appendix A.243

JCL/SAS Code: II19

```
%MACRO II_19();
```

```
%LET WORKING=(OTHER_SSI_WORKING=1 OR PROA&YR.12=1 OR PROB&YR.12=1);
```

```
PROC SQL;
```

```
  /**** OVERALL ROW ****/
```

```
  SELECT
```

```
    "II.19" AS TABLE_SHELL
  , "ALL AREAS" AS STATE_AREA
  , SUM(SSR_BLIND_DISABLED=1) AS ALL_BLIND_DISABLED
  , SUM(&WORKING.) AS TOTAL_SSI_WORKING
  , CALCULATED TOTAL_SSI_WORKING/
    CALCULATED ALL_BLIND_DISABLED*100
    AS PERC_OF_ALL FORMAT=8.1
  , SUM(WORKING_1619A=1) AS SECTION_1619A
  , CALCULATED SECTION_1619A/
    CALCULATED TOTAL_SSI_WORKING*100
    AS PERC_WORKING_1619A FORMAT=8.1
  , SUM(WORKING_1619B=1) AS SECTION_1619B
  , CALCULATED SECTION_1619B/
    CALCULATED TOTAL_SSI_WORKING*100
    AS PERC_WORKING_1619B FORMAT=8.1
  , SUM(OTHER_SSI_WORKING=1) AS OTHER_SSI_WORKING
  , CALCULATED OTHER_SSI_WORKING/
    CALCULATED TOTAL_SSI_WORKING*100
    AS PERC_WORKING_OTHER FORMAT=8.1
```

```
  FROM &DSN
```

```
  OUTER UNION CORR
```

```
  /**** ADD STATE ROWS ****/
```

```
  SELECT
```

```
    "II.19" AS TABLE_SHELL
  , PUT(STATE, $STATE_FMT.) AS STATE_AREA
  , SUM(SSR_BLIND_DISABLED=1) AS ALL_BLIND_DISABLED
  , SUM(&WORKING.) AS TOTAL_SSI_WORKING
  , CALCULATED TOTAL_SSI_WORKING/
    CALCULATED ALL_BLIND_DISABLED*100
    AS PERC_OF_ALL FORMAT=8.1
  , SUM(WORKING_1619A=1) AS SECTION_1619A
  , CALCULATED SECTION_1619A/
    CALCULATED TOTAL_SSI_WORKING*100
```

Appendix A.243

JCL/SAS Code: II19

```
    AS PERC_WORKING_1619A FORMAT=8.1
  ,SUM(WORKING_1619B=1) AS SECTION_1619B
  ,CALCULATED SECTION_1619B/
    CALCULATED TOTAL_SSI_WORKING*100
  AS PERC_WORKING_1619B FORMAT=8.1
  ,SUM(OTHER_SSI_WORKING=1) AS OTHER_SSI_WORKING
  ,CALCULATED OTHER_SSI_WORKING/
    CALCULATED TOTAL_SSI_WORKING*100
  AS PERC_WORKING_OTHER FORMAT=8.1
FROM &DSN
GROUP BY STATE_AREA
;

QUIT;

%MEND;

%II_19();
```

## Appendix A.244

JCL/SAS Code: II20

```
%MACRO II_20();

%LET WORKING=(OTHER_SSI_WORKING=1 OR PROA&YR.12=1 OR PROB&YR.12=1);

PROC SQL NOPRINT;
  SELECT SUM(SSI_BENE=1) INTO :ALL_BLIND_DISABLED
    FROM &DSN. WHERE 18 <= AGE20&YR. < 65;
  SELECT SUM(&WORKING.) INTO :TOTAL_SSI_WORKING
    FROM &DSN. WHERE 18 <= AGE20&YR. < 65;
  SELECT SUM(WORKING_1619A=1) INTO :SECTION_1619A
    FROM &DSN. WHERE 18 <= AGE20&YR. < 65;
  SELECT SUM(WORKING_1619B=1) INTO :SECTION_1619B
    FROM &DSN. WHERE 18 <= AGE20&YR. < 65;
QUIT;

PROC SQL;
  /**** OVERALL NUMBER ROW ****/

  SELECT
    "II.20" AS TABLE_SHELL
    ,"TOTAL NUMBER" AS DX_GROUP
    ,SUM(SSI_BENE=1) AS ALL_BLIND_DISABLED
    ,SUM(&WORKING.) AS TOTAL_SSI_WORKING
    ,SUM(WORKING_1619A=1) AS SECTION_1619A
    ,SUM(WORKING_1619B=1) AS SECTION_1619B
  FROM &DSN.
  WHERE 18 <= AGE20&YR. < 65

  OUTER UNION CORR
  /*** OVERALL PERCENT ROW ***/
  SELECT
    "II.20" AS TABLE_SHELL
    ,"TOTAL PERCENT" AS DX_GROUP
    ,SUM(SSI_BENE=1)/&ALL_BLIND_DISABLED.*100
      AS ALL_BLIND_DISABLED FORMAT=8.1
    ,SUM(&WORKING.)/&TOTAL_SSI_WORKING.*100
      AS TOTAL_SSI_WORKING FORMAT=8.1
    ,SUM(WORKING_1619A=1)/&SECTION_1619A.*100
      AS SECTION_1619A
    ,SUM(WORKING_1619B=1)/&SECTION_1619B.*100
      AS SECTION_1619B
  FROM &DSN.
  WHERE 18 <= AGE20&YR. < 65
```

Appendix A.244

JCL/SAS Code: II20

```
OUTER UNION CORR
/*** DX GROUPS ROWS ***/
SELECT
  "II.20" AS TABLE_SHELL
  ,PUT(GROUP,DIAG.) AS DX_GROUP
  ,SUM(SS1_BENE=1)/&ALL_BLIND_DISABLED.*100
    AS ALL_BLIND_DISABLED FORMAT=8.1
  ,SUM(&WORKING.)/&TOTAL_SSI_WORKING.*100
    AS TOTAL_SSI_WORKING FORMAT=8.1
  ,SUM(WORKING_1619A=1)/&SECTION_1619A.*100
    AS SECTION_1619A
  ,SUM(WORKING_1619B=1)/&SECTION_1619B.*100
    AS SECTION_1619B
FROM &DSN.
WHERE 18 <= AGE20&YR. < 65
GROUP BY DX_GROUP
;

QUIT;

%MEND;

%II_20();
```



## Appendix A.245

JCL/SAS Code: II21

```
%MACRO II_21();

%LET ALL=(SSI_BENE=1 OR WORKING_1619B=1);
%LET WORKING=(OTHER_SSI_WORKING=1 OR PROA&YR.12=1 OR PROB&YR.12=1);

PROC SQL NOPRINT;

    SELECT SUM(&ALL.) INTO :ALL_BLIND_DISABLED
    FROM &DSN. WHERE 18 <= AGE20&YR. < 65;
QUIT;

PROC SQL;
  /*** OVERALL ROW ***/
  SELECT
    "II.21" AS TABLE_SHELL
    ,"TOTAL" AS AGEGRP
    ,SUM(&ALL.) AS ALL_BLIND_DISABLED
    ,CALCULATED ALL_BLIND_DISABLED/&ALL_BLIND_DISABLED*100
      AS PERC_ALL FORMAT=8.1
    ,SUM(&WORKING.) AS TOTAL_SSI_WORKING
    ,CALCULATED TOTAL_SSI_WORKING/
      CALCULATED ALL_BLIND_DISABLED*100
      AS PERC_WORKING FORMAT=8.1
    ,SUM(WORKING_1619B=1) AS SECTION_1619B
    ,CALCULATED SECTION_1619B/
      CALCULATED ALL_BLIND_DISABLED*100
      AS PERC_1619B FORMAT=8.1
  FROM &DSN
  WHERE 18 <= AGE20&YR. < 65

  OUTER UNION CORR

  /**** AGE GROUP ROWS ***/
  SELECT
    "II.21" AS TABLE_SHELL
    ,PUT(AGE20&YR., SSI_WORKER_AGE.) AS AGEGRP
    ,SUM(&ALL.) AS ALL_BLIND_DISABLED
    ,CALCULATED ALL_BLIND_DISABLED/&ALL_BLIND_DISABLED*100
      AS PERC_ALL FORMAT=8.1
    ,SUM(&WORKING.) AS TOTAL_SSI_WORKING
    ,CALCULATED TOTAL_SSI_WORKING/
      CALCULATED ALL_BLIND_DISABLED*100
      AS PERC_WORKING FORMAT=8.1
    ,SUM(WORKING_1619B=1) AS SECTION_1619B
```

Appendix A.245

JCL/SAS Code: II21

```
      ,CALCULATED SECTION_1619B/  
        CALCULATED ALL_BLIND_DISABLED*100  
        AS PERC_1619B FORMAT=8.1  
FROM &DSN  
WHERE 18 <= AGE20&YR. < 65  
GROUP BY AGEGRP  
;  
  
QUIT;  
  
%MEND;  
  
%II_21();
```

## Appendix A.246

JCL/SAS Code: II22

```
%MACRO II_22();

%LET S1619A = (SSI_BENE=1 AND WORKING_1619A=1);

PROC SQL;
  /**** OVERALL ROW ****/

  SELECT
    "II.22" AS TABLE_SHELL
    ,"ALL AREAS" AS STATE_AREA
    ,SUM(&S1619A.) AS SECT_1619A
    ,AVG(CASE WHEN &S1619A.
      THEN SUM(T16GRSAMT&YR.12,T16SEAMT&YR.12)
      ELSE . END) AS SECT_1619A_AVG_EARN
    ,SUM(WORKING_1619B=1) AS SECT_1619B
    ,AVG(CASE WHEN WORKING_1619B = 1
      THEN SUM(T16GRSAMT&YR.12,T16SEAMT&YR.12)
      ELSE . END) AS SECT_1619B_AVG_EARN
  FROM &DSN

  OUTER UNION CORR

  /**** ADD STATE ROWS ****/
  SELECT
    "II.22" AS TABLE_SHELL
    ,PUT(STATE, $STATE_FMT.) AS STATE_AREA
    ,SUM(&S1619A.) AS SECT_1619A
    ,AVG(CASE WHEN &S1619A.
      THEN SUM(T16GRSAMT&YR.12,T16SEAMT&YR.12)
      ELSE . END) AS SECT_1619A_AVG_EARN
    ,SUM(WORKING_1619B=1) AS SECT_1619B
    ,AVG(CASE WHEN WORKING_1619B = 1
      THEN SUM(T16GRSAMT&YR.12,T16SEAMT&YR.12)
      ELSE . END) AS SECT_1619B_AVG_EARN
  FROM &DSN
  GROUP BY STATE_AREA
  ;

QUIT;

%MEND;

%II_22();
```

Appendix A.247

JCL/SAS Code: II23

```
%MACRO II_23();

PROC SQL;

SELECT
  "II.23" AS TABLE_SHELL
  ,"ALL AREAS" AS STATE_AREA
  ,SUM(PASGRP=1) AS NUM_PASS
  ,SUM(EXPGRP=1) AS NUM_IRWE
  ,SUM(BEXPGRP=1) AS NUM_BWE
FROM &DSN

OUTER UNION CORR
/**** ADD STATE ROWS ****/
SELECT
  "II.23" AS TABLE_SHELL
  ,PUT(STATE, $STATE_FMT.) AS STATE_AREA
  ,SUM(PASGRP=1) AS NUM_PASS
  ,SUM(EXPGRP=1) AS NUM_IRWE
  ,SUM(BEXPGRP=1) AS NUM_BWE
FROM &DSN
GROUP BY STATE_AREA
;
QUIT;

%MEND;

%II_23();
```

## Appendix A.248

JCL/SAS Code: SSITS5

```
%MACRO SSITS5(POP = /*POPULATION TO RUN DATA ON*/);
```

```
PROC SQL;
```

```
  /**** OVERALL ROW ****/
```

```
SELECT
```

```
  "SSI-TS5" AS TABLE_SHELL  
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION  
  , "ALL CITIZEN" AS STATE  
  ,COUNT(*) AS TOTAL  
  ,SUM(AGE20&YR. < 18) AS UNDER_18  
  ,SUM(18<=AGE20&YR.<65) AS AGE_18TO64
```

```
FROM &DSN
```

```
WHERE &POP. = 1
```

```
  AND AGE20&YR. < 65
```

```
  AND CITIZEN_NUMI NE "A"
```

```
  AND CITIZEN_NUMI NE ""
```

```
OUTER UNION CORR
```

```
/** CITIZEN NUM*****/
```

```
SELECT
```

```
  "SSI-TS5" AS TABLE_SHELL  
  ,PUT("&POP.", $BENE_FMT.) AS POPULATION  
  ,STATE  
  ,COUNT(*) AS TOTAL  
  ,SUM(AGE20&YR. < 18) AS UNDER_18  
  ,SUM(18<=AGE20&YR.<65) AS AGE_18TO64
```

```
FROM &DSN
```

```
WHERE &POP. = 1
```

```
  AND AGE20&YR. < 65
```

```
  AND CITIZEN_NUMI NE "A"
```

```
  AND CITIZEN_NUMI NE ""
```

```
GROUP BY STATE
```

```
;
```

```
QUIT;
```

```
%MEND;
```

```
%SSITS5(POP = SSI_BENE);
```

## Appendix A.249

JCL/SAS Code: STWSSI

```

//#3590STW JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21 CONSTRUCTION 50214.220
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.SSRF.PRDLIB(STWSSI)
//*
//* PURPOSE: TO CREATE SSI MONTHLY STW INDICATOR
//*
//* PROGRAMMER: SVETLANA BRONNIKOV
//* E-MAIL SBRONNIKOV@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//DEMO  DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//SSR   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIELIG.SA.V1,DISP=SHR
//STWSSI DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SCWfstw.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//FREQS DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SCWfstwQ.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//SYSIN DD *

```

```

OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;

```

```

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

```

```

%MACRO START;

```

```

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

```

```

DATA STWSSI.STWSSI(KEEP=SSN STWSSI:);
MERGE DEMO.DMSTW(IN=DEMO KEEP=SSN DOBBEST DODBEST FRADATE)
      SSR.SSIELIG(IN=SSR KEEP=SSN PSTA: EICM: UINC: CURCOMP:);
;
BY SSN;
IF SSR AND DEMO;

```

```

* KEEP ONLY BENES WHO ARE 18 OR OLDER ON THE LAST DAY OF THE DAF YEAR

```

Appendix A.249

JCL/SAS Code: STWSSI

```
STWS ARE ONLY CONSTRUCTED FOR ADULT BENES;
IF YRDIF(DOBBEST,"31DEC&ENDYR."D,'AGE')>=18;
```

```
* CREATE SSI STW INDICATORS;
```

```
* INPUT ARRAYS;
```

```
ARRAY PSTA(*) %DO MM=1 %TO &TOTMOS.; PSTA&&YY&MM %END;;
ARRAY EICM(*) %DO MM=1 %TO &TOTMOS.; EICM&&YY&MM %END;;
ARRAY UINC(*) %DO MM=1 %TO &TOTMOS.; UINC&&YY&MM %END;;
ARRAY CURCOMP(*) $ %DO MM=1 %TO &TOTMOS.; CURCOMP&&YY&MM %END;;
```

```
* FEDERAL BENEFIT RATES ARRAYS FOR INDIVIDUALS AND COUPLES
```

```
UPDATE FOR COMING YEAR. FIND TABLE AT
http://www.ssa.gov/oact/cola/SSlamts.html;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(FBRIN)";
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(FBRCP)";
```

```
* OUTPUT ARRAYS;
```

```
ARRAY FBR(*) %DO MM=1 %TO &TOTMOS.; FBR&&YY&MM %END;;
```

```
* THE STWSSI ARRAY HAS A LOWER BOUND OF 0 SO THAT WHEN LOOKING BACK
AT THE PREVIOUS MONTH DURING THE STWSSI LOOP, SPECIAL CODING IS NOT
NECESSARY TO HANDLE THE FIRST MONTH (JAN 1994). SO AN INDEX VALUE OF
0 WOULD REFER TO THE DUMMYSTWSSI ARRAY ELEMENT AND AN INDEX VALUE OF
1 WOULD REFER TO STWSSI9401;
```

```
ARRAY STWSSI(0:&TOTMOS.) DUMMYSTWSSI
      %DO MM=1 %TO &TOTMOS.; STWSSI&&YY&MM %END;;
```

```
* SET VARIABLES TO CONTROL THE LOOP SETTING THE STW VARIABLE:
```

```
POPULATE STW FROM THE BEGINNING OF THE DAF DATA (JAN 1994) OR BENES
18TH BIRTHDAY WHICHEVER COMES LATER UNTIL BENE DIES, ATTAINS FRA, OR
THE END OF OUR DAF YEAR WHICHEVER COMES FIRST;
STARTLOOP=INTCK('MONTHS','01JAN1994'D,
      MAX('01JAN1994'D,
      INTNX('YEARS',DOBBEST,18,'SAME DAY')))+1;
STOPLOOP=INTCK('MONTHS','01JAN1994'D,
      MIN(DOBBEST,FRADATE,"31DEC&ENDYR."D))+1;
```

```
DO I=STARTLOOP TO STOPLOOP;
```

```
* SET YEAR BEING PROCESSED
```

```
CONVERT RELATIVE MONTHS (REPRESENTED BY I) TO A YEAR VALUE BY
DIVIDING THE RELATIVE MONTHS BY 12 AND APPLY THE CEIL FUNCTION TO
RETURN THE NEXT LARGEST INTEGER OR EQUAL. SO AN INDEX VALUE OF 1
THROUGH 12 WOULD REFER TO YEAR 1994, 13 THROUGH 24 TO YEAR 1995 AND
```

Appendix A.249

JCL/SAS Code: STWSSI

```

SO ON;
LOOPYR=CEIL(I/12)+1993;

*FBR FOR COUPLE IS MAXIMUM SSI COUPLE PAYMENT DEVIDED BY 2.
FBR FOR INDIVIDUAL IS MAXIMUM SSI INDIVIDUAL PAYMENT;
IF CURCOMP(I)='6' THEN FBR(I)=FBRCP(LOOPYR)/2;
ELSE FBR(I)=FBRIN(LOOPYR);

* IN CURRENT PAY STW=0;
IF PSTA(I)='C01' THEN STWSSI(I)=0;

* ONCE TERMINATED FOR WORK ALWAYS TERMINATED FOR WORK UNTIL BACK IN
CURRENT PAY, FRA, OR DEAD. (THE FIRST TIME THROUGH THIS LOOP THERE
WILL BE NO VALUE IN THE PREVIOUS MONTH (I-1) SO THIS BLOCK OF CODE
WILL NOT EXECUTE);
ELSE IF STWSSI(I-1)=2 THEN DO;
  * WHEN THERE IS A STATUS CODE THEN STWSSI=2;
  IF PSTA(I) NE '' THEN STWSSI(I)=2;
END;

* SUSPENDED FOR EXCESS INCOME (N01) THEN CHECK INCOME;
ELSE IF PSTA(I)='N01' THEN DO;
  * WHEN EARNED INCOME IS GREATER THAN ZERO OR THE PREVIOUS MONTH
  WAS AN STW MONTH THE CHECK UNEARNED INCOME;
  IF EICM(I)>0 OR STWSSI(I-1)=1 THEN DO;
    * WHEN UNEARNED INCOME IS BELOW FBR STW=1;
    IF UINC(I)<FBR(I) THEN STWSSI(I)=1;
    * WHEN UNEARNED INCOME IS AT OR ABOVE FBR STW=4;
    ELSE STWSSI(I)=4;
  END;
  ELSE STWSSI(I)=8;
END; * END N01 LOOP;

* WHEN PREVIOUS MONTH IS SUSPENDED FOR WORK AND CURRENT MONTH
SUSPENDED FOR NON-EXCLUDEABLE RESOURCES OR FAILURE TO FILE FOR
OTHER BENEFITS CHECK AND EARNED INCOME IS GREATER THAN 0 THEN
CHECK UNEARNED INCOME;
ELSE IF PSTA(I) IN ('N04','N06') AND STWSSI(I-1)=1
AND EICM(I)>0 THEN DO;

  * UNEARNED INCOME LESS THAN FBR STW=1;
  IF UINC(I)<FBR(I) THEN STWSSI(I)=1;
  * UNEARNED INCOME IS AT OR ABOVE FBR STW=4;
  ELSE STWSSI(I)=4;

```



## Appendix A.249

JCL/SAS Code: STWSSI

```
END; * END N04/N06 LOOP;

* WHEN TERMINATED FOR SOMETHING OTHER THAN DEATH AND PREVIOUS
  MONTH IS SUSPENDED OR TERMINATED FOR WORK THEN SUSPENDED FOR WORK;
ELSE IF SUBSTR(PSTA(I),1,1)='T' AND PSTA(I) NE 'T01'
  AND STWSSI(I-1) IN (1,2) THEN STWSSI(I)=2;

* IF NONE OF THE ABOVE CONDITIONS ARE MET AND THERE IS A VALUE IN
  PSTA THEN SUSPENDED OR TERMINATION FOR SOMETHING OTHER THAN WORK;
ELSE IF SUBSTR(PSTA(I),1,1)='T' THEN STWSSI(I)=9;
ELSE IF PSTA(I) NE '' THEN STWSSI(I)=8;

* IF PSTA IS BLANK THEN IT WILL REMAIN MISSING;

END; * END MONTHLY LOOP;

LABEL
%DO MM=1 %TO &TOTMOS.;
STWSSI&&YY&MM = "&&YY&MM SSI LEFT DUE TO WORK INDICATOR"
%END;
;
RUN;

PROC FREQ DATA=STWSSI.STWSSI;
  %DO MM=1 %TO &TOTMOS.;
    TABLES STWSSI&&YY&MM / MISSING OUT=FREQS.FREQ&&YY&MM;
  %END;
RUN;

%MEND;
%START;
ENDSAS;
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
DATA
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>
```

Appendix A.249

JCL/SAS Code: STWSSI

TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>

SUBJECT: DAF21 SCWFSTW IS COMPLETE

\*\*\*\*\* Bottom of Data \*\*\*\*\*

## Appendix A.250

JCL/SAS Code: STWCM

```
//#3590STW JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21 CONSTRUCTION 50214.220
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.SCWF.PRDLIB(STWCM)
//*
//* PURPOSE: TO CREATE MONTHLY COMBINED STW INDICATOR
//*
//* PROGRAMMER:
//* SVETLANA BRONNIKOV
//* E-MAIL SBRONNIKOV@MATHEMATICA-MPR.COM
//*****
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//TEMP1 DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      SPACE=(CYL,(1000,100),RLSE),VOL=(,,8)
//STWSSI DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SCWFSTW.SA.V1,DISP=SHR
//STWDI  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.STWDI.SA.V1,DISP=SHR
//DEMO   DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//STWCM  DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SCWFSTWC.SA.V1,
//      DISP=(NEW,CATLG,DELETE),UNIT=TSILO
//STWCMFQ DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SCWFQ.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=BINARY OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";

DATA STWCM.STWCM(KEEP=SSN STWSSI: STWDI: STWCM:);
  MERGE STWSSI.STWSSI(KEEP=SSN STWSSI:)
        STWDI.STWDI(KEEP=SSN STWDI: RFST:)
        DEMO.DMSTW(IN=DEMO KEEP=SSN FRADATE DODBEST)
  ;
  BY SSN;
```

## Appendix A.250

JCL/SAS Code: STWCM

```
IF DEMO;

* INPUT ARRAYS;
ARRAY STWDI (*) %DO MM=1 %TO &TOTMOS.; STWDI&&YY&MM %END;;
ARRAY STWSSI (*) %DO MM=1 %TO &TOTMOS.; STWSSI&&YY&MM %END;;
ARRAY RFST (*) %DO MM=1 %TO &TOTMOS.; RFST&&YY&MM %END;;

* OUTPUT ARRAYS;
ARRAY STWCM (*) %DO MM=1 %TO &TOTMOS.; STWCM&&YY&MM %END;;

DO I=1 TO &TOTMOS.;

    * SET INITIAL VALUE OF STWCM TO THE LOWER OF STWSSI AND STWSSI;
    STWCM(I)=MIN(STWDI(I),STWSSI(I));

    * RESET STWCM;
    SELECT(STWCM(I));

        * WHEN STWCM IS 4
        4=UNEARNED INCOME HIGHER THAN FBR.....TO 8
        ;
        WHEN(4) STWCM(I)=8;
        * OTHERWISE DO NOT RESET;
        OTHERWISE;
    END; * END SELECT;

    * IF THE STWSSI INDICATED SUSPENSE OF TERMINATION FOR WORK
    BUT SSDI DATA INDICATED THE BENE WAS IN PRISON THEN SET
    THE STWCM TO THAT OF THE STWDI;
    IF STWSSI(I) IN (1,2,3,4) AND RFST(I)='PRISON' THEN
    STWCM(I)=STWDI(I);

END; * END MONTHLY LOOPING;

* STRING STWCM 3'S AND 9'S IN THE STW INDICATORS;

* CONSTRUCT A DATE THAT IS THE EARLIEST OF FRA, DOD OR END OF DAF
YEAR THIS WILL ALLOW THE LOOP TO STOP STRINGING 3 WHEN A PERSON IS
DEAD OR FRA;
STOPLOOP=INTCK('MONTH','01JAN1994'D
,MIN(FRADATE,DODBEST,"31DEC&ENDYR."D))+1;

* STARTING IN FEB 1994 (2ND MONTH OF THE DAF) IF THE CURRENT MONTH
IS MISSING AND PREVIOUS MONTH IS 2 OR 3 (TERMINATED FOR WORK) THEN
SET CURRENT MONTH TO 3 ELSE IF PREVIOUS MONTH IS 9 THEN SET
```

## Appendix A.250

JCL/SAS Code: STWCM

```
    CURRENT MONTH TO 9;
    DO I=2 TO STOPLOOP;

        * STRING STWSSI;
        IF STWSSI(I)=. THEN DO;
            IF STWSSI(I-1) IN (2,3) THEN STWSSI(I)=3;
            ELSE IF STWSSI(I-1)=9 THEN STWSSI(I)=9;
        END;

        * STRING STWDI;
        IF STWDI(I)=. THEN DO;
            IF STWDI(I-1) IN (2,3) THEN STWDI(I)=3;
            ELSE IF STWDI(I-1)=9 THEN STWDI(I)=9;
        END;

        * STRING STWCOMB;
        IF STWCM(I)=. THEN DO;
            IF STWCM(I-1) IN (2,3) THEN STWCM(I)=3;
            ELSE IF STWCM(I-1)=9 THEN STWCM(I)=9;
        END;

    END;
RUN;

PROC CONTENTS DATA=STWCM.STWCM;
TITLE "CONTENTS OF STWCM";
RUN;

PROC FREQ DATA=STWCM.STWCM;
%DO MM=1 %TO &TOTMOS.;
    TABLES STWCM&&YY&MM / MISSING OUT=STWCMFQ.STWCM&&YY&MM;
    TABLES STWDI&&YY&MM / MISSING OUT=STWCMFQ.STWDI&&YY&MM;
    TABLES STWSSI&&YY&MM / MISSING OUT=STWCMFQ.STWSSI&&YY&MM;
%END;

RUN;

%MEND;
%START;
ENDSAS;
/*****
//* SENDING EMAIL FROM HOST USING IEBGENER
/*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.250

JCL/SAS Code: STWCM

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
DATA
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>
TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 SCWF STWCM COMPLETE
```

## Appendix A.251

JCL/SAS Code: BFWSSI

```
//#3590BFW JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21 CONSTRUCTION 50214.220
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.SCWF.PRDLIB(BFWSSI)
//*
//* PURPOSE: CREATE MONTHLY BENEFITS FORGONE FOR WORK FOR SSI
//*
//* PROGRAMMER:
//* SVETLANA BRONNIKOV
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//TEMP DD DSN=&&TEMPX,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//STWCM DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SCWFSTWC.SA.V1,DISP=SHR
//DEMO DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//SSR DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSIELIG.SA.V1,DISP=SHR
//BFWSSI DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SCWFBFW.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=BINARY MPRINT OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

%MACRO START;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

DATA BFWSSI.BFWSSI;
MERGE STWCM.STWCM(IN=INSTWCM KEEP=SSN STW:)
      DEMO.DMSTW(KEEP=SSN DODBEST FRADATE)
      SSR.SSIELIG(KEEP=SSN UINC: FAMT: EICM: LIVF: CURCOMP:)
;
BY SSN;
IF INSTWCM;
```

Appendix A.251

JCL/SAS Code: BFWSSI

```

* FEDERAL BENEFIT RATES ARRAYS FOR INDIVIDUALS AND COUPLES
  TABLE AT: HTTPS://WWW.SSA.GOV/OACT/COLA/SSIAMTS.HTML;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(FBRIN)";
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(FBRCP)";

* SET COLA VALUES
  HTTPS://WWW.SSA.GOV/NEWS/COLA/;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(COLA)";

* INPUT ARRAYS;
ARRAY STWSSI (*) %DO MM=1 %TO &TOTMOS.; STWSSI&&YY&MM %END;;
ARRAY UINC (*) %DO MM=1 %TO &TOTMOS.; UINC&&YY&MM %END;;
ARRAY FAMT (*) %DO MM=1 %TO &TOTMOS.; FAMT&&YY&MM %END;;
ARRAY CURCOMP(*) $ %DO MM=1 %TO &TOTMOS.; CURCOMP&&YY&MM %END;;
ARRAY LIVF(*) $ %DO MM=1 %TO &TOTMOS.; LIVF&&YY&MM %END;;
ARRAY EICM (*) %DO MM=1 %TO &TOTMOS.; EICM&&YY&MM %END;;

* OUTPUT ARRAYS;
ARRAY BFWSSI_DRAFT (*) %DO MM=1 %TO &TOTMOS.;
      BFWSSI_DRAFT&&YY&MM
      %END;;
ARRAY BUDGET_MO (*) %DO MM=1 %TO &TOTMOS.; BUDGET_MO&&YY&MM %END;;
ARRAY FBR (*) %DO MM=1 %TO &TOTMOS.; FBR&&YY&MM %END;;

* SET DATE TO STOP CALCULATING: END OF DAF YEAR, MONTH BENE TURNS FRA,
  OR MONTH BENE DIES;
STOPLOOP=INTCK('MONTHS','01JAN1994'D,
      MIN(DODBEST,FRADATE,"31DEC&ENDYR."D))+1;

* CALCULATE SSI BENEFITS FOREGONE FOR BENES RECEIVING BENEFITS;

* STRING INCOME AND ADJUST THE PREVIOUS YEARS INCOME;
DO I=2 TO DIM(STWSSI);
  * SET YEAR BEING PROCESSED. I REPRESENTS THE MONTH OF THE DAF
  (MONTH 1 IS JAN 1994, MONTH 2 IS FEB 1994, MONTH 25 JAN 1996, ETC.)
  TO DETERMINE THE YEAR THE CEIL FUNCTION IS APPLIED TO THE RESULTS
  OF I DIVIDED BY 12 (CEIL RETURNS THE SMALLEST INTEGER THAT IS
  GREATER THAN OR EQUAL TO THE ARGUMENT) THEN 1993 IS ADDED. SOMONTH
  13 WOULD BE CEIL(13/12)=2+1993 OR 1995, MONTH 13 FALLS IN 1995;
  LOOPYR=CEIL(I/12)+1993;

  * IF CURRENT MONTH STW IS 2 OR 3 THEN SET THE INCOME THIS MONTH TO
  THAT OF THE PREVIOUS MONTH;
  IF STWSSI(I) IN (2,3) THEN DO;

```



Appendix A.251

JCL/SAS Code: BFWSSI

```

* IF THIS MONTH IS JAN THEN ADJUST THE INCOME FROM THE PREVIOUS
MONTH BY COLA AND SET THIS MONTH TO THAT ADJUST AMOUNT;
IF (I-1)/12 = INT((I-1)/12) THEN DO;
  UINC(I) = UINC(I-1)*(1+COLA(LOOPYR));
  EICM(I) = EICM(I-1)*(1+COLA(LOOPYR));
END;

* IF THIS MONTH IS NOT JAN THEN SET THE INCOME FOR THIS MONTH TO
THAT OF THE PREVIOUS MONTH;
ELSE DO;
  UINC(I) = UINC(I-1);
  EICM(I) = EICM(I-1);
END;

END; * END STW IN 2 OR 3 LOOP;
END; * END STRINGING INCOME LOOP;

* DUE TO A CHANGE IN SSAS SSI PAYMENT CALCULATION ALGORITHM
BEGINNING IN APR 2005 THERE ARE TWO DIFFERENT BFW ALGORITHMS: ONE
FOR MOTHS PRIOR TO APR 2005 AND ONE FOR MONTHS FROM APR 2005 ONWARD.
THIS CALCULATED VARIABLE WILL BE USING IN THE CREATE BFW LOOP BELOW;
MAR2005=INTCK('MONTH','1JAN1994'D,'31MAR2005'D)+1;

* CREATE BFW. THE LOOP STARTS IN MAR 1994 BECAUSE A 2 MONTH LOOK BACK
PERIOD IS REQUIRED. THIS RESULTS IN HAVING NO BFWSSI VALUES IN JAN &
FEB 1994.;
DO I=3 TO STOPLOOP;

* SET YEAR BEING PROCESSED IN THIS EXECUTION OF THE LOOP;
LOOPYR=CEIL(I/12)+1993;

* FBR FOR COUPLE IS MAXIMUM SSI COUPLE PAYMENT DIVIDED BY 2
FBR FOR INDIVIDUAL IS MAXIMUM SSI INDIVIDUAL PAYMENT;
IF CURCOMP(I)='6' THEN FBR(I)=FBRCP(LOOPYR)/2;
ELSE FBR(I)=FBRIN(LOOPYR);

* CREATE A BUDGET MONTH FOR MONTHS THAT REQUIRE A BFW CALCULATION
(STW 0, 1, 2, OR 3);
IF STWSSI(I) IN (0,1,2,3) THEN DO;
  * FIRST WE CHECK THE CASE WHERE A BENEFICIARY WAS DUE A FEDERAL
  SSI PAYMENT BUT WAS NOT DUE A FEDERAL PAYMENT IN THE PRIOR MONTH
  AS A RESULT OF UNEARNED INCOME.BECAUSE THE REMOVAL OF ALL EARNED
  INCOME IN THE RELEVANT COMPUTATION MONTHS DOES NOT CHANGE THE BM,
  THE BM REMAINS AT 0;
  IF UINC(I-1) >= FBR(I-1) THEN BUDGET_MO(I) = 0;

```

Appendix A.251

JCL/SAS Code: BFWSSI

```

* NEXT WE CHECK IN WHICH UNEARNED INCOME ALONE IN TWO MONTH LOOK BACK
  ELIMINATES THE POSSIBILITY OF RECEIVING A FEDERAL SSI PAYMENT. AS A
  RESULT, THE BM BECOMES -1 IN THIS CASE. IF BENE RECEIVED FEDERAL
  SSI PAYMENT IN ALL THREE MONTHS THEN THE BM BECOMES -2;
ELSE IF STWSSI(I-1) IN (0,1,2,3) THEN DO;
  IF UINC(I-2)>=FBR(I-2) THEN BUDGET_MO(I)=-1;
  ELSE IF STWSSI(I-2) IN (0,1,2,3) THEN BUDGET_MO(I)=-2;

* IF I-2 IS AN INELIGIBILITY MONTH THEN BM=-1;
  ELSE BUDGET_MO(I)= -1;
END;

* IF THE PRIOR MONTH WAS AN INELIGIBILITY MONTH WITHOUT REGARD TO
  EARNED INCOME, THE BM WILL REMAIN 0;
  ELSE BUDGET_MO(I)= 0;

END; * END BUDGET MONTH CALCULATION LOOP;

* BFW IS CACLUATED ONLY IN MONTHS WERE THE STW IS 0, 1, 2, OR 3.
  HOWEVER IF THERE IS NOT EARNED INCOME ONE MONTH PRIOR THEN EITHER
  THERE WAS NONRECURRING EARNED INCOME IN THE BM THAT DOES NOT AFFECT
  THE CURRENT MONTH, OR THERE WAS NO EARNED INCOME IN THE BM. IN
  EITHER CASE, THE BFWSSI VALUE IS 0. IF THERE WAS INCOME IN THE MONTH
  PRIOR THEN THE EICM FROM THE BM-1 IS THE BFWSSI VALUE;
  IF STWSSI(I) IN (4,8,9,.) OR (STWSSI(I)=0 AND UINC(I)>=FBR(I))
  THEN BFWSSI_DRAFT(I)=0;

* CREATE BFWS FOR BENES IN CURRENT PAY;
  ELSE IF STWSSI(I)=0 THEN DO;

* MONTHS THRU TO MARCH 2005;
  IF I<=MAR2005 THEN DO;
  IF FAMT(I) NE 0 AND FAMT(I-1) NE 0 AND FAMT(I-2) NE 0
  THEN BFWSSI_DRAFT(I)=EICM(I-2);
  ELSE IF FAMT(I) NE 0 AND FAMT(I-1) NE 0 AND FAMT(I-2)=0
  THEN BFWSSI_DRAFT(I)=EICM(I-1);
  ELSE IF FAMT(I) NE 0 AND FAMT(I-1)=0
  THEN BFWSSI_DRAFT(I)=EICM(I);
  END; * END CALCULATIONS THRU MAR 2005;

* MONTHS AFTER MARCH 2005;
  ELSE DO;
  IF FAMT(I) NE 0 AND FAMT(I-1) NE 0 AND FAMT(I-2) NE 0

```

## Appendix A.251

JCL/SAS Code: BFWSSI

```

AND FAMT(I-3) NE 0 THEN BFWSSI_DRAFT(I)=EICM(I-2);
ELSE IF FAMT(I) NE 0 AND FAMT(I-1) NE 0 AND FAMT(I-2) NE 0
  AND FAMT(I-3)=0 AND EICM(I-1)=0 THEN BFWSSI_DRAFT(I)=0;
ELSE IF FAMT(I) NE 0 AND FAMT(I-1) NE 0 AND FAMT(I-2) NE 0
  AND FAMT(I-3)=0 AND EICM(I-1) NE 0
  THEN BFWSSI_DRAFT(I)=EICM(I-2);
ELSE IF FAMT(I) NE 0 AND FAMT(I-1) NE 0 AND FAMT(I-2)=0
  AND EICM(I)=0 THEN BFWSSI_DRAFT(I)=0;
ELSE IF FAMT(I) NE 0 AND FAMT(I-1) NE 0 AND FAMT(I-2)=0
  AND EICM(I) NE 0 THEN BFWSSI_DRAFT(I)=EICM(I-1);
ELSE IF FAMT(I) NE 0 AND FAMT(I-1)=0 THEN BFWSSI_DRAFT(I)=EICM(I);
END; * END CACLUATIONS AFTER MAR 2005;
END; * END WHEN STW=0 LOOP;

* CREATE BFW SSI FOR BENES WHO ARE SUSPENDED OR TERMINATED
  FOR WORK (STWSSI(I)=1/2/3) ;
ELSE BFWSSI_DRAFT(I)=FBR(I)-UINC(I+BUDGET_MO(I))-FAMT(I);

END; * END CREATE BFW LOOP;

* BFWSSI LOGICALLY CANNOT BE LESS THAN 0, GREATER THAN FBR, OR WHEN OFF
  THE ROLLS FOR WORK CANNOT BE 0. IF THE ALGORITHM HAS RESULTED IN
  THESE CASES THEN SET THE BFWSSI TO A SPECIAL MISSING;
DO I=3 TO DIM(BFWSSI_DRAFT);
IF BFWSSI_DRAFT(I)<0 OR BFWSSI_DRAFT(I)>FBR(I) OR
  (STWSSI(I) IN (1,2,3) AND BFWSSI_DRAFT(I)=0)
THEN BFWSSI_DRAFT(I)=.M;

* WHEN A BENE IS LIVING IN A MEDICAL FACILITY AND MEDICAID PAYS MORE
  THAN 50 PERCENT OF EXPENSES (LIVF=D) THEN THE MAX SSI BENEFIT IS $30
  SO THE BFWSSI CAN BE NO HIGHER THAN THAT;
IF LIVF(I)='D' AND BFWSSI_DRAFT(I) NE .
THEN BFWSSI_DRAFT(I)=MIN(30,BFWSSI_DRAFT(I));
END;

RUN;

* MEANS TO BE EXECUTED IN SEPERATE PROGRAM;
*PROC MEANS DATA = BFWSSI.BFWSSI;
*VAR BFWSSI_DRAFT;;
*RUN;

%MEND;
%START;
/*****

```

Appendix A.251

JCL/SAS Code: BFWSSI

```
//* SENDING EMAIL FROM HOST USING IEBGENER
/*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
DATA
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>
TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 SCWF BFWSSI COMPLETE
```

## Appendix A.252

JCL/SAS Code: BFWCM

```
//#3590BFW JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21 CONSTRUCTION 50214.220
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.SCWF.PRDLIB(BFWCM)
//*
//* PURPOSE: CREATE MONTHLY BENEFITS FORGONE FOR SSDI & SSI COMBINED
//*
//* PROGRAMMER:
//* SVETLANA BRONNIKOV
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//TEMP DD DSN=&&TEMPX,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,VOL=(,,10)
//BFWSSI DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SCWFBFW.SA.V1,DISP=SHR
//BFWDI DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.BFWDI.SA.V1,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SCWFBFWC.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;
```

```
* STEP TO ASSIGN MACRO VARIABLES TO HANDLE TIME SERIES DATA;
```

```
%LET BEGYR=1994;
```

```
%LET ENDYR=2021; /* CHANGE AS NEEDED */
```

```
%LET ENDMN=12; /* CHANGE AS NEEDED */
```

```
%MACRO START;
```

```
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";
```

```
DATA OUT.BFW(KEEP=SSN BFW: STW:);
```

```
MERGE BFWSSI.BFWSSI
```

```
      BFWDI.BFWDI
```

```
;
```

```
BY SSN;
```

```
* INPUT ARRAYS;
```

## Appendix A.252

JCL/SAS Code: BFWCM

```
ARRAY BFWSSI_DRAFT (*) %DO MM=1 %TO &TOTMOS.;
      BFWSSI_DRAFT&&YY&MM
      %END;;
ARRAY STWSSI(*) %DO MM=1 %TO &TOTMOS.; STWSSI&&YY&MM %END;;
ARRAY BFWDI (*) %DO MM=1 %TO &TOTMOS.; BFWDI&&YY&MM %END;;
ARRAY STWDI (*) %DO MM=1 %TO &TOTMOS.; STWDI&&YY&MM %END;;
ARRAY STWCM (*) %DO MM=1 %TO &TOTMOS.; STWCM&&YY&MM %END;;

* OUTPUT ARRAYS;
ARRAY BFWCM_DRAFT (*) %DO MM=1 %TO &TOTMOS.; BFWCM_DRAFT&&YY&MM %END;;

* RECODE BFWSSI_DRAFT AND BFWDI VARIABLES *;
DO J=1 TO DIM(BFWSSI_DRAFT);
  IF BFWSSI_DRAFT(J)<0 AND NOT MISSING(STWSSI(J))
    THEN BFWSSI_DRAFT(J)=0;
  ELSE IF MISSING(STWSSI(J)) THEN BFWSSI_DRAFT(J)=.;

  IF BFWDI(J)<0 AND NOT MISSING(STWDI(J))
    THEN BFWDI(J)=0;
  ELSE IF MISSING(STWDI(J)) THEN BFWDI(J)=.;
END;

DO I=2 TO DIM(BFWSSI_DRAFT);
  BFWCM_DRAFT(I)=SUM(OF BFWDI(I),BFWSSI_DRAFT(I));

  IF BFWCM_DRAFT(I)<0 AND NOT MISSING(STWCM(I))
    THEN BFWCM_DRAFT(I)=0;
  ELSE IF MISSING(STWCM(I)) THEN BFWCM_DRAFT(I)=.;
END;

LABEL %DO MM=1 %TO &TOTMOS.;
BFWDI&&YY&MM="SSDI ESTIMATED BENEFITS FORGONE FOR WORK &&YY&MM"
BFWSSI_DRAFT&&YY&MM=
"SSI ESTIMATED BENEFITS FORGONE FOR WORK (DRAFT) &&YY&MM"
BFWCM_DRAFT&&YY&MM=
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK (DRAFT) &&YY&MM"
%END;
RUN;

PROC CONTENTS DATA = OUT.BFW;
RUN;

OPTIONS NOLABEL;
PROC MEANS DATA = OUT.BFW;
  VAR BFW: ;
```

Appendix A.252

JCL/SAS Code: BFWCM

```
RUN;

%MEND;
%START;
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
DATA
FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>
TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 SCWF BFWCM COMPLETE
```

## Appendix A.253

JCL/SAS Code: FINSCWF

```

//#3590STW JOB (12510000,T715,,SAS,,ITC9FL),BRONNIKOV,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#3590
//*****
//* PROJECT: DAF21 CONSTRUCTION 50214
//*
//* PROGRAM: OPDR.TG.PRD.ETTW.#3590.DAF21.SCWF.PRDLIB(FINSCWF)
//*
//* PURPOSE: FINALIZE SCWF -- NOTE THAT THIS DOES NOT INCLUDE THE
//*      ADDITION OF COMMON VARIABLES
//*
//* PROGRAMMER:
//* SVETLANA BRONNIKOV
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//SCWF DD DSN=OPDR.TG.PRD.ETTW.#3590.DAF21P.SCWFBFWC.SA.V1,DISP=SHR
//FIN DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.SCWF,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD *

```

```
OPTIONS NOCENTER COMPRESS=BINARY MPRINT OBS=MAX;
```

```

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */
%LET ENDMN=12; /* CHANGE AS NEEDED */

```

```
%MACRO START;
```

```
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMMACR)";
```

```

DATA FIN.SCWF;
SET SCWF.BFW;
DROP STWDI;
  BFWDI;
  BFWDI_DEPEND;
  BFWDI_PRIM;
;
RENAME %DO MM=1 %TO &TOTMOS.;
  STWSSI&&YY&MM=STWSSI_OLD&&YY&MM
  STWCM&&YY&MM=STWCM_OLD&&YY&MM
  BFWSSI_DRAFT&&YY&MM=BFWSSI_OLD&&YY&MM
  BFWCM_DRAFT&&YY&MM=BFWCM_OLD&&YY&MM

```



Appendix A.253

JCL/SAS Code: FINSCWF

```

        %END;;
    LABEL
    %DO MM=1 %TO &TOTMOS.;
    STWSSI&&YY&MM=
    "SCWF SSI SUSPENSE OR TERMINATION DUE TO WORK (OLD) &&YY&MM"
    STWCM&&YY&MM=
    "SCWF COMBINED (SSDI/SSI) SUSPENSE OR TERM DUE TO WORK (OLD) &&YY&MM"
    BFWSSI_DRAFT&&YY&MM=
    "SCWF SSI ESTIMATED BENEFITS FORGONE FOR WORK (OLD) &&YY&MM"
    BFWCM_DRAFT&&YY&MM=
    "SCWF COMBINED (SSDI/SSI) EST. BENEFITS FORGONE FOR WORK (OLD) &&YY&MM"
    %END;;

    RUN;

    PROC CONTENTS DATA=FIN.SCWF;
    RUN;

    %MEND;
    %START;
    //*****
    // * SENDING EMAIL FROM HOST USING IEBGENER
    //*****
    //SENDNOTE EXEC PGM=IEBGENER
    //SYSIN DD DUMMY
    //SYSPRINT DD SYSOUT=*
    //SYSUT2 DD SYSOUT=(B,SMTP)
    //SYSUT1 DD *
    HELO MVSHOST
    MAIL FROM:<SBRONNIKOV@MATHEMATICA-MPR.COM>
    RCPT TO:<SBRONNIKOV@MATHEMATICA-MPR.COM>
    DATA
    FROM: <SBRONNIKOV@MATHEMATICA-MPR.COM>
    TO: <SBRONNIKOV@MATHEMATICA-MPR.COM>
    SUBJECT: DAF21 FINSCWF COMPLETE

```

## Appendix A.254

JCL/SAS Code: RDWATERF

```

//#5413RDW JOB (12510000,T715,,SAS,,ITC9FL),CARRILLO,
// MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#5413
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.#5413.DAF21.WATRF.PRDLIB(RDWATRF) |
//* | |
//* *-----*
//* READ IN THE WATERFALL AND TRACK ACTIVE FILES
//* CONTACT CHRISTIAN CARRILLO
//* E-MAIL CCARRILLO@MATHEMATICA.COM
//*****
//*
//JS010 EXEC SAS9,
// WORK='120000,60000'
//*
//WTRF DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21.WATERFAL.FY21,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.WATERFAL.SA.V1,
// DISP=(NEW,CATLG,DELETE),
// SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES OBS=MAX MPRINT MACROGEN;

***** READ IN THE WATERFALL AND TRACK ACTIVE FILES*****;
* THE FLAT FILE WE RECEIVED FROM SSA CONTAINS TWO FILES IN ONE.
THE WATERFALL FILE IS LOCATED IN LRECL 0-100.
THE ATIVE TRACKING FILE IS LOCATED IN LRECL 101-875.
FOR READING IN THE ACTIVE TRACKING LAYOUT,
100 MUST BE ADDED TO THE START POSITION NOTED IN THE LAYOUT.;

DATA OUT.WATERFALL;
  INFILE WTRF TRUNCOVER;

INPUT
@102 CAN $9. /*SSN FROM TRACK ACTIVE*/
@034 EXSSN $9. /*SSN FROM WATERFALL*/
@001 COHORT $4. /*BEGIN OTHER VARS FROM WATERFALL*/
@005 FYINIACT $4.
@009 FYACTION $4.
@013 ACTDTE_DATE ?? YMMN6. /*READ DATE AS YYYYMM*/
@019 FASUB $1.
@020 CESS $1.
@021 FLIP $1.
@022 PREVOACT $1.

```

## Appendix A.254

JCL/SAS Code: RDWATERF

```
@023 PREVFLIP $1.
@024 EXFYTAG $10.
@043 EXBIC $2.
@045 EXFILE $2.
@047 EXDOM_DATE ?? YMMDD8. /*READ DATE AS YYYYMMDD*/
@055 EXTOC $2.
@057 FILLER $44.
@101 DEC $1. /*BEGIN TRACK ACTIVE, ADD 100 TO LAYOUT*/
@111 BIC $2.
@113 TOC $2.
@115 SSN $9.
@124 CLMTYPE $1.
@125 PSC $1.
@126 INTEG $1.
@127 STYPE $1.
@128 SCORE 4.
@132 SELECT_IN $1.
@133 PROFCD $2.
@135 STATE $2.
@137 REGION $2.
@139 DOC $3.
@142 PMF_TSITE $3.
@145 PMF_TCOM $5.
@150 PMF_LOCF $4.
@154 PMF_TLOCS $4.
@158 NIF_ST_SENT $2.
@160 PROFILE_SAMPLE_FY $2.
@162 DOM_DATE ?? YMMDD8. /*READ-IN DOM DATE YYYYMMDD*/
@170 ROM_DATE ?? YMMDD8. /*READ-IN ROM DATE YYYYMMDD*/
@178 CDRTYPE $1.
@179 DECREC $1.
@180 FLIPCODE $1.
@181 IDRI $1.
@182 IDRIDTE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@190 IDR $1.
@191 IDRDTTE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@199 IDR_OLD $1.
@200 IDRDTTE_OLD ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@208 IDR_IDRI_OLD_IND $1.
@209 FILLER_2 $8.
@217 MBRIDR $1.
@218 ICR454_IND $1.
@219 CDIO_DATE ?? YMMN6. /* READ-IN DATE AS YYYYMM*/
@225 CDIE_DATE ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@231 MDR_ORIG $1.
```

## Appendix A.254

JCL/SAS Code: RDWATERF

```
@232 MRED_DATE ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@238 MDR_MBR $1.
@239 MRED_MBR_DATE ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@245 LAF $2.
@247 PRY $2.
@249 TWPSD_DATE ?? YMMN6. /* READ-IN DATE AS YYYYMM*/
@255 TWPESM $1.
@256 TWPDD_DATE ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@262 CWSD_DATE ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@268 DM_EC $3.
@271 DM_ED_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@279 DM_DEST $3.
@282 SCORE_CMS 4.
@286 RELTYPE $1.
@287 PROFILE_SAMPLE $1.
@288 DOB ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@296 RACE $1.
@297 SEX $1.
@298 IMP $4.
@302 BS $2.
@304 MDR_VAL $1.
@305 MRED_VAL_DATE ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@311 MDRTYPE $4.
@315 ENTLEVEL $1.
@316 DOO_DATE ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@322 DOE_DATE ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@328 WRC $1.
@329 DCF_REV_TYPE $1.
@330 DCF_REV_NUM $4.
@334 PS_LOW_MAILER $1.
@335 SNO $1.
@336 FILLER_4 $11.
@347 DEFERRAL_FY 2.
@349 FY_MAIL 4.
@353 TITLE $2.
@355 TOU $6.
@361 TOA $1.
@362 DATE_TOA ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@370 DES_CASE $5.
@375 FRZ_FLD_IND $1.
@376 T6CH2_STATE $2.
@378 PEND_IND $1.
@379 GOLD_K_IND $1.
@380 GOLD_K_DT ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@386 DR_1619 $2.
```

## Appendix A.254

JCL/SAS Code: RDWATERF

```
@388 DR_1619_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@396 OLD_CURSTATE $3.
@399 CUR_CHG_DT ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@405 CURSTAT $3.
@408 T6HUN $9.
@417 T6_MDR $1.
@418 T6_MRED_DATE ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@424 PL_CODE $1.
@425 PL_CDE_DT ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@433 T6_DPM $1.
@434 KATRINA $1.
@435 ERROR_PHONE $2.
@437 R1EGS 5.
@442 R2EGS 5.
@447 R3EGS 5.
@452 TOP $1.
@453 CC $1.
@454 PCDR $2.
@456 P1_DODEC_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@464 LAST_ACT_OACT $1.
@465 LAST_REV_OACT $1.
@466 LT_ACT_DT_OACT_DATE ?? YMMN6. /*READ-IN DATE AS YYYYMM*/
@472 STATUS_LOC $1.
@473 STATUS_OC_SENT $1.
@474 D1_DODEC_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@482 D1_RB $2.
@484 D1_PDX $4.
@488 D1_SDX $4.
@492 D1_MLN $7.
@499 D1_VRN $5.
@504 DPM_DCF $1.
@505 STRDT_DPM_DCF_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@513 PENDING_FASTRACK $1.
@514 D3_DODEC_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@522 D3_AL $1.
@523 D3_RFC $2.
@525 D3_BS $2.
@527 D3_PDX $4.
@531 D3_SDX $4.
@435 LFL $3.
@538 OTL $3.
@541 JSITE $1.
@542 CSITE $3.
@545 PLOC $5.
@550 PDIL_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
```

## Appendix A.254

JCL/SAS Code: RDWATERF

```
@558 PFUNC $4.  
@562 PSTA $4.  
@566 T6_CFL $4.  
@570 T6_DIL_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/  
@578 T6_CFLIND $1.  
@579 T6_FRC $2.  
@581 DS_EC $3.  
@584 DS_ED_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/  
@592 DS_DEST $3.  
@595 CURSTAT_TERM $3.  
@598 CURSTAT_TRM_DT ?? YMMN6. /*READ-IN DATE AS YYYYMM*/  
@504 DS_PDI $1.  
@505 FT_DEC $1.  
@606 STRDT_833_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/  
@614 STRDT_ND_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/  
@622 STRDT_RCN_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/  
@630 RCPT_ALJ_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/  
@638 RCPT_AC_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/  
@646 LAST_RCPT_AL $1.  
@647 NIF_PROJ_IND $1.  
@648 NIF_PROJ_DT ?? YMMN6. /*READ-IN DATE AS YYYYMM*/  
@654 DODEC_833_DATE ?? YMMDD8. /*READ-IN DATE AS YMMDD8*/  
@662 AL_833 $1.  
@663 RDT_833 $1.  
@664 RID_833 $1.  
@665 DDS_833 $3.  
@668 RFC_833 $2.  
@670 BFD833 $1.  
@671 WRM833 $2.  
@673 BS833 $2.  
@675 RPD833 $4.  
@679 RSD833 $4.  
@683 CDT833 $2.  
@685 EOR833 $1.  
@686 CER833 $1.  
@687 FS833 $1.  
@688 DR833 $1.  
@689 DD833_DATE ?? YMMN6. /*READ-IN DATE AS YYYYMM*/  
@695 FILLER_12 $7.  
@702 DODECND_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/  
@710 ALND $1.  
@711 HGT_COURT $2.  
@713 DISP_COURT $1.  
@714 ACT_COURT $1.  
@715 DODEC_COURT_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
```

Appendix A.254

JCL/SAS Code: RDWATERF

```
@723 AL_COURT $1.
@724 RDT_COURT $1.
@725 RID_COURT $1.
@726 ISSUE_COURT $1.
@727 RPDY_COURT $4.
@731 DODEC_RCN_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@739 AL_RCN $1.
@740 RDT_RCN $1.
@741 RID_RCN $1.
@742 DDS_RCN $3.
@745 RFC_RCN $2.
@747 BFD_RCN $1.
@748 WRM_RCN $2.
@750 BS_RCN $2.
@752 RPDY_RCN $4.
@756 RSDX_RCN $4.
@760 CDT_RCN $2.
@762 EOR_RCN $1.
@763 CER_RCN $1.
@764 FS_RCN $1.
@765 RCPT_COURT_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@773 HGT_ALJ $2.
@775 PFI_ALJ $1.
@776 ORI_ALJ $1.
@777 DISP_ALJ $1.
@778 ACT_ALJ $1.
@779 DODEC_ALJ_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@787 AL_ALJ $1.
@788 RDT_ALJ $1.
@789 RID_ALJ $1.
@790 ISSUE_ALJ $1.
@791 FILLER_J $1.
@792 RPDY_ALJ $4.
@796 OFC_ALJ $4.
@800 CLT_ALJ $4.
@804 STUDY_CA $1.
@805 FILLER_15 $6.
@811 HGT_AC $2.
@813 DISP_AC $1.
@814 ACT_AC $1.
@815 DODEC_AC_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@823 AL_AC $1.
@824 RDT_AC $1.
@825 RID_AC $1.
@826 ISSUE_AC $1.
```

Appendix A.254

JCL/SAS Code: RDWATERF

```
@827 FILLER_AC $1.
@828 RPDY_AC $4.
@832 OFC_AC $4.
@836 CLT_AC $4.
@840 ELECTRONIC_SW $1.
@841 FY_TAG_TYPE $1.
@842 FY_TAG_WRKAUX $3.
@845 FY_TAG_YEAR $2.
@847 FY_TAG_RESULT $4.
@851 LAST_UPDATE $6.
@857 LAST_ACT $1.
@858 DT_LAST_ACTION_DATE ?? YMMDD8. /*READ-IN DATE AS YYYYMMDD*/
@866 FY_COUNT $4.
@870 ACT_COUNT $1.
@871 DES_COUNT $5.
@;
RUN;
```

```
TITLE "PROC CONTENTS OF WATERFALL FILE";
PROC CONTENTS DATA = OUT.WATERFALL VARNUM; RUN;
TITLE;
```

```
/*
/*****
/* SENDING EMAIL FROM HOST USING IEBGENER
/*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<CCARRILLO@MATHEMATICA-MPR.COM>
RCPT TO:<CCARRILLO@MATHEMATICA-MPR.COM>
DATA
FROM: <CCARRILLO@MATHEMATICA-MPR.COM>
TO: <CCARRILLO@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 READ WATERFALL FILE COMPLETE
/*
```



## Appendix A.255

JCL/SAS Code: PROCWTRF

```

//#5413PWF JOB (12510000,T715,,SAS,,ITC9FL),CARRILLO,
// MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#5413
//*****
//* PROJECT: DAF
//* PROGRAM: OPDR.TG.PRD.ETTW.#5413.DAF21.SCDR.PRDLIB(PROCWTRF)
//* PUPROSE: PROCESS WATERFALL FILE FOR DAF
//* CONTACT: CHRISTIAN CARRILLO
//*      CCARRILLO@MATHEMATICA-MPR.COM
//*****
//*
//JS010 EXEC SAS9,
//  WORK='120000,60000'
//*
//WTRF  DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.WATERFAL.SA.V1,
//  DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.WTRFCDRS.SA.V1,
//  DISP=(OLD,CATLG,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,10)
//TEMPRT DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMPCNT DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMPMAX DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP  DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMPOUT DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT SYMBOLGEN OBS=MAX;

%MACRO DOIT;

* FORMAT APPLIED TO SSN VARS FOR MASKING;
PROC FORMAT;
  VALUE $SSN_F
    MISSING = "MISSING"
    '      ' = "MISSING"
    OTHER = "POPULATED"
  ;
RUN;

/* SORT BY SSN AND STRDT_833_DATE TO PUT THE CDRS IN ORDER */
PROC SORT DATA=WTRF.WATERFALL

```

## Appendix A.255

JCL/SAS Code: PROCWTRF

```
OUT = TEMPSRT.WATERFALL;
BY SSN STRDT_833_DATE ;
RUN;
```

```
PROC FREQ DATA=TEMPSRT.WATERFALL;
TITLE "CHECK IF MISSING SSNS CAN BE BACKFILLED WITH CAN IF BIC = A";
WHERE MISSING(SSN);
TABLES SSN * CAN * BIC / LIST MISSING;
FORMAT SSN CAN $SSN_F.;
RUN; TITLE;
```

```
/* FLAG SSNS WITH MULTIPLE RECORDS TO TRACK PROCESSING */
DATA TEMPSRT.WATERFALL;
SET TEMPSRT.WATERFALL;
BY SSN STRDT_833_DATE ;
IF NOT (FIRST.SSN AND LAST.SSN) THEN FLAG_MULTRECS = 1;
ELSE FLAG_MULTRECS = 0;
RUN;
```

```
/*FIND HIGHEST REV_SEQ_NUM*/
/*WE NEED THIS VALUE TO DYNAMICALLY DEVELOP BENE-LEVEL ARRAYS*/
%MACRO MAXVAL;
```

```
%GLOBAL MAX_NUM;
```

```
PROC SQL;
CREATE TABLE TEMPCNT.CDRCNT AS
SELECT SSN
      ,COUNT(*) AS CDR_WFAL_NUM
FROM TEMPSRT.WATERFALL (KEEP=SSN)
GROUP BY SSN
ORDER BY SSN;
QUIT;
```

```
PROC MEANS DATA=TEMPCNT.CDRCNT MAX;
WHERE NOT MISSING(SSN) ;
VAR CDR_WFAL_NUM;
OUTPUT OUT=TEMPMAX.CDRMAX MAX=;
RUN;
```

```
DATA TEMPMAX.CDRMAX;
SET TEMPMAX.CDRMAX;
CALL SYMPUTX("MAX_NUM",PUT(CDR_WFAL_NUM,BEST32.));
RUN;
```

## Appendix A.255

JCL/SAS Code: PROCWTRF

```
%PUT MAXMIUM VALUE FOR CDR_WFAL_NUM IS &&MAX_NUM.;
```

```
PROC FREQ DATA=TEMPCNT.CDRCNT;
TITLE "QA CHECK: ONLY MISSING SSN WILL HAVE CDR COUNT > &&MAX_NUM.";
WHERE CDR_WFAL_NUM > &&MAX_NUM.;
TABLES SSN / LIST MISSING;
FORMAT SSN $SSN_F.;
RUN;
```

```
%MEND;
%MAXVAL;
```

```
/* BUILD SSN-LEVEL FILE */
DATA TEMP.WATERFALL (KEEP = SSN CDR_WFAL_ : FLAG_MULTRECS);
SET TEMPSRT.WATERFALL
  (KEEP= SSN ROM_DATE CDRTYPE WRM833 STRDT_833_DATE
   DODEC_833_DATE AL_833 RDT_833 STRDT_RCN_DATE
   DODEC_RCN_DATE AL_RCN RDT_RCN RCPT_ALJ_DATE
   DODEC_ALJ_DATE AL_ALJ RDT_ALJ DODEC_AC_DATE
   AL_AC RDT_AC DODEC_COURT_DATE STRDT_ND_DATE
   AL_COURT RDT_COURT FASUB ACTDTE_DATE FLAG_MULTRECS
   WHERE=(NOT MISSING(SSN)));
BY SSN STRDT_833_DATE;
```

```
/*ASSIGN CDR ORDER VALUES FOR ARRAY*/
RETAIN CDR_ORDER_NUM;
IF FIRST.SSN THEN CDR_ORDER_NUM = 1;
ELSE CDR_ORDER_NUM = CDR_ORDER_NUM + 1;
```

```
/* INITIALIZE N OUTPUT VARIABLES FOR PROCESSING */
ARRAY WROM (*) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ROM&MM. %END;;
ARRAY WCDRT ($) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_TYPE&MM. %END;;
ARRAY WWRM ($) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_WRM&MM. %END;;
ARRAY WINST (*) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_INIT_STRDT&MM. %END;;
ARRAY WIDEC (*) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_INIT_DODEC&MM. %END;;
ARRAY WINAL ($) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_INIT_AL&MM. %END;;
ARRAY WIRDT ($) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_INIT_RDT&MM. %END;;
ARRAY WRCST (*) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_RCN_STRDT&MM. %END;;
ARRAY WRDEC (*) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_RCN_DODEC&MM. %END;;
ARRAY WRAL ($) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_RCN_AL&MM. %END;;
ARRAY WRRDT ($) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_RCN_RDT&MM. %END;;
ARRAY WALJR (*) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ALJ_RCPTDT&MM. %END;;
ARRAY WALJD (*) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ALJ_DODEC&MM. %END;;
ARRAY WALJA ($) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ALJ_AL&MM. %END;;
ARRAY WALRD ($) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ALJ_RDT&MM. %END;;
```

## Appendix A.255

JCL/SAS Code: PROCWTRF

```

ARRAY WACDO (*) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_AC_DODEC&MM. %END;;
ARRAY WACAL (*) $ %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_AC_AL&MM. %END;;
ARRAY WACR (*) $ %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_AC_RDT&MM. %END;;
ARRAY WDCDO (*) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_DC_DODEC&MM. %END;;
ARRAY WNDST (*) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ND_STRDT&MM. %END;;
ARRAY WDCAL (*) $ %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_DC_AL&MM. %END;;
ARRAY WDCRD (*) $ %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_DC_RDT&MM. %END;;
ARRAY WFIAC (*) $ %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_FIN_ACT&MM. %END;;
ARRAY WFIDT (*) %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_FIN_ACT_DT&MM. %END;;
ARRAY WADJL (*) $ %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_AL&MM. %END;;

```

```

FORMAT CDR_WFAL_ROM: CDR_WFAL_INIT_STRDT: CDR_WFAL_INIT_DODEC:
      CDR_WFAL_RCN_STRDT: CDR_WFAL_RCN_DODEC: CDR_WFAL_ALJ_RCPTDT:
      CDR_WFAL_ALJ_DODEC: CDR_WFAL_AC_DODEC: CDR_WFAL_DC_DODEC:
      CDR_WFAL_ND_STRDT: YYMMDD10.
      CDR_WFAL_FIN_ACT_DT: YYMMD7.;

```

```

RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ROM&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_TYPE&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_WRM&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_INIT_STRDT&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_INIT_DODEC&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_INIT_AL&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_INIT_RDT&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_RCN_STRDT&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_RCN_DODEC&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_RCN_AL&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_RCN_RDT&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ALJ_RCPTDT&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ALJ_DODEC&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ALJ_AL&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ALJ_RDT&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_AC_DODEC&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_AC_AL&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_AC_RDT&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_DC_DODEC&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_ND_STRDT&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_DC_AL&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_DC_RDT&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_FIN_ACT&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_FIN_ACT_DT&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM.; CDR_WFAL_AL&MM. %END;;

```

```

LABEL
%DO MM=1 %TO &&MAX_NUM.;

```

## Appendix A.255

JCL/SAS Code: PROCWTRF

```
CDR_WFAL_ROM&MM.="MEDICAL CDR DATE OF MAILER RECEIPT &MM."  
CDR_WFAL_TYPE&MM.="MEDICAL CDR TYPE &MM."  
CDR_WFAL_WRM&MM.="CDR WHY REVIEW MADE &MM."  
CDR_WFAL_INIT_STRDT&MM.=  
  "MEDICAL CDR STATE DDS RECEIPT DATE FOR INITIAL LEVEL &MM."  
CDR_WFAL_INIT_DODEC&MM.=  
  "MEDICAL CDR DATE OF DECISION AT INITIAL LEVEL &MM."  
CDR_WFAL_INIT_AL&MM.=  
  "MEDICAL CDR ADJUDICATIVE LEVEL OF INITIAL DECISION &MM."  
CDR_WFAL_INIT_RDT&MM.=  
  "MEDICAL CDR RESULT OF DETERMINATION FOR INITIAL DECISION &MM."  
CDR_WFAL_RCN_STRDT&MM.=  
  "MEDICAL CDR STATE DDS RECEIPT DATE FOR RECONSIDERATION LEVEL &MM."  
CDR_WFAL_RCN_DODEC&MM.=  
  "MEDICAL CDR DATE OF DECISION AT RECONSIDERATION LEVEL &MM."  
CDR_WFAL_RCN_AL&MM.=  
  "MEDICAL CDR ADJUDICATIVE LEVEL OF RECONSIDERATION DECISION &MM."  
CDR_WFAL_RCN_RDT&MM.=  
  "MEDICAL CDR RESULT OF DETERMINATION FOR RECONSIDERATION DECISION &MM."  
CDR_WFAL_ALJ_RCPTDT&MM.=  
  "MEDICAL CDR RECEIPT DATE FOR ADMINISTRATIVE LAW JUDGE LEVEL &MM."  
CDR_WFAL_ALJ_DODEC&MM.=  
  "MEDICAL CDR DATE OF DECISION AT ADMINISTRATIVE LAW JUDGE LEVEL &MM."  
CDR_WFAL_ALJ_AL&MM.=  
  "MEDICAL CDR ADJUDICATIVE LEVEL OF ADMINISTRATIVE LAW JUDGE DECISION &MM."  
CDR_WFAL_ALJ_RDT&MM.=  
  "MEDICAL CDR RESULT OF DETERMINATION FOR ADMINISTRATIVE LAW JUDGE DECISION &MM."  
  
CDR_WFAL_AC_DODEC&MM.=  
  "MEDICAL CDR DATE OF DECISION AT APPEALS COUNCIL LEVEL &MM."  
CDR_WFAL_AC_AL&MM.=  
  "MEDICAL CDR ADJUDICATIVE LEVEL OF APPEALS COUNCIL DECISION &MM."  
CDR_WFAL_AC_RDT&MM.=  
  "MEDICAL CDR RESULT OF DETERMINATION FOR APPEALS COUNCIL DECISION &MM."  
CDR_WFAL_DC_DODEC&MM.=  
  "MEDICAL CDR DATE OF DECISION AT DISTRICT COURT LEVEL &MM."  
CDR_WFAL_ND_STRDT&MM.=  
  "MEDICAL CDR STATE DDS RECEIPT DATE FOR NO DECISION &MM."  
CDR_WFAL_DC_AL&MM.=  
  "MEDICAL CDR ADJUDICATIVE LEVEL OF DISTRICT COURT DECISION &MM."  
CDR_WFAL_DC_RDT&MM.=  
  "MEDICAL CDR RESULT OF DETERMINATION FOR DISTRICT COURT DECISION &MM."  
CDR_WFAL_FIN_ACT&MM.= "MEDICAL CDR FINAL ACTION &MM."  
CDR_WFAL_FIN_ACT_DT&MM.= "MEDICAL CDR DATE OF FINAL ACTION &MM."  
CDR_WFAL_AL&MM.= "MEDICAL CDR HIGHEST ADJUDICATIVE LEVEL &MM."
```

## Appendix A.255

JCL/SAS Code: PROCWTRF

```
%END;;
```

```
IF FIRST.SSN THEN DO;
```

```
  DO MM=1 TO &&MAX_NUM.;
```

```
    WROM(MM) = .;
```

```
    WCDRT(MM) = ";
```

```
    WWRM(MM) = ";
```

```
    WINST(MM) = .;
```

```
    WIDEC(MM) = .;
```

```
    WINAL(MM) = ";
```

```
    WIRDT(MM) = ";
```

```
    WRCST(MM) = .;
```

```
    WRDEC(MM) = .;
```

```
    WRAL(MM) = ";
```

```
    WRRDT(MM) = ";
```

```
    WALJR(MM) = .;
```

```
    WALJD(MM) = .;
```

```
    WALJA(MM) = ";
```

```
    WALRD(MM) = ";
```

```
    WACDO(MM) = .;
```

```
    WACAL(MM) = ";
```

```
    WACR(MM) = ";
```

```
    WDCDO(MM) = .;
```

```
    WNDST(MM) = .;
```

```
    WDCAL(MM) = ";
```

```
    WDCRD(MM) = ";
```

```
    WFIAC(MM) = ";
```

```
    WFIDT(MM) = .;
```

```
    WADJL(MM) = "";
```

```
  END;
```

```
END;
```

```
/* FILL IN ARRAYS BASED ON CDR ORDER NUMBER*/
```

```
WROM(CDR_ORDER_NUM) = ROM_DATE;
```

```
WCDRT(CDR_ORDER_NUM) = CDRTYPE;
```

```
WWRM(CDR_ORDER_NUM) = WRM833;
```

```
WINST(CDR_ORDER_NUM) = STRDT_833_DATE;
```

```
WIDEC(CDR_ORDER_NUM) = DODEC_833_DATE;
```

```
WINAL(CDR_ORDER_NUM) = AL_833;
```

```
WIRDT(CDR_ORDER_NUM) = RDT_833;
```

```
WRCST(CDR_ORDER_NUM) = STRDT_RCN_DATE;
```

```
WRDEC(CDR_ORDER_NUM) = DODEC_RCN_DATE;
```

```
WRAL(CDR_ORDER_NUM) = AL_RCN;
```

```
WRRDT(CDR_ORDER_NUM) = RDT_RCN;
```

```
WALJR(CDR_ORDER_NUM) = RCPT_ALJ_DATE;
```

## Appendix A.255

JCL/SAS Code: PROCWTRF

```
WALJD(CDR_ORDER_NUM) = DODEC_ALJ_DATE;
WALJA(CDR_ORDER_NUM) = AL_ALJ;
WALRD(CDR_ORDER_NUM) = RDT_ALJ;
WACDO(CDR_ORDER_NUM) = DODEC_AC_DATE;
WACAL(CDR_ORDER_NUM) = AL_AC;
WACR(CDR_ORDER_NUM) = RDT_AC;
WDCDO(CDR_ORDER_NUM) = DODEC_COURT_DATE;
WNDST(CDR_ORDER_NUM) = STRDT_ND_DATE;
WDCAL(CDR_ORDER_NUM) = AL_COURT;
WDCRD(CDR_ORDER_NUM) = RDT_COURT;
WFIAC(CDR_ORDER_NUM) = FASUB;
WFIDT(CDR_ORDER_NUM) = ACTDTE_DATE;

/* CONSTRUCT CDR_WFAL_AL */
DO MM=1 TO &&MAX_NUM.;
  IF WDCAL(MM) = "F" THEN WADJL(MM) = "F";
  ELSE IF WACAL(MM) = "E" THEN WADJL(MM) = "E";
  ELSE IF WACAL(MM) = "A" THEN WADJL(MM) = "J";
  ELSE IF WALJA(MM) = "D" THEN WADJL(MM) = "D";
  ELSE IF WALJA(MM) = "H" THEN WADJL(MM) = "H";
  ELSE IF WRAL(MM) = "B" THEN WADJL(MM) = "B";
  ELSE IF WRAL(MM) = "C" THEN WADJL(MM) = "C";
  ELSE IF WRAL(MM) = "G" THEN WADJL(MM) = "G";
  ELSE IF WINAL(MM) = "A" THEN WADJL(MM) = "A";
END;

/* IF LAST SSN THEN OUTPUT*/
IF LAST.SSN THEN OUTPUT;
DROP MM;
RUN;

/* CHECK CONSTRUCTION OF ALL CDR_WFAL_ALN VARIABLES */
%DO MM=1 %TO &&MAX_NUM.;
  TITLE "CHECK CROSTAB FOR CDR_WFAL_AL&MM.";
  PROC FREQ DATA= TEMP.WATERFALL;
    TABLE CDR_WFAL_AL&MM.
      CDR_WFAL_AL&MM.*CDR_WFAL_INIT_AL&MM.*CDR_WFAL_RCN_AL&MM.*
      CDR_WFAL_ALJ_AL&MM.*CDR_WFAL_AC_AL&MM.*CDR_WFAL_DC_AL&MM.
    / LIST MISSING;
  RUN;
%END;

DATA OUT.WATERFALL;
```

## Appendix A.255

JCL/SAS Code: PROCWTRF

```

MERGE TEMPCNT.CDRCNT (WHERE=(NOT MISSING(SSN)))
  TEMP.WATERFALL (IN=INFALL DROP=FLAG_MULTRECS
    %DO I=1 %TO &&MAX_NUM.;
    CDR_WFAL_INIT_AL&I.
    CDR_WFAL_RCN_AL&I.
    CDR_WFAL_ALJ_AL&I.
    CDR_WFAL_AC_AL&I.
    CDR_WFAL_DC_AL&I.
    %END;
  );
BY SSN;
IF INFALL;
LABEL CDR_WFAL_NUM = "NUMBER OF CDRS, WATERFALL";
RUN;

PROC CONTENTS DATA=OUT.WATERFALL; RUN;

/*CHECK VARIABLE CONSTRUCTION HERE.*/
TITLE "CHECK FIRST 5 SSNS PRE PROCESSING";
PROC PRINT DATA = TEMPSRT.WATERFALL (OBS = 20);
WHERE FLAG_MULTRECS = 1 AND NOT MISSING(SSN);
VAR SSN ROM_DATE CDRTYPE WRM833 STRDT_833_DATE
  DODEC_833_DATE AL_833 RDT_833 STRDT_RCN_DATE
  DODEC_RCN_DATE AL_RCN RDT_RCN RCPT_ALJ_DATE
  DODEC_ALJ_DATE AL_ALJ RDT_ALJ DODEC_AC_DATE
  AL_AC RDT_AC DODEC_COURT_DATE STRDT_ND_DATE
  AL_COURT RDT_COURT FASUB ACTDTE_DATE;
FORMAT ROM_DATE STRDT_833_DATE DODEC_833_DATE STRDT_RCN_DATE
  DODEC_RCN_DATE RCPT_ALJ_DATE DODEC_ALJ_DATE DODEC_AC_DATE
  DODEC_COURT_DATE STRDT_ND_DATE YYMMDDD10.
  ACTDTE_DATE YYMMD7.;
RUN; TITLE;

TITLE "CHECK FIRST FEW SSNS AFTER PROCESSING";
PROC PRINT DATA = TEMP.WATERFALL (OBS = 8);
WHERE FLAG_MULTRECS= 1;
RUN; TITLE;

%MEND DOIT;
%DOIT;

/*
//*****
/* SENDING EMAIL FROM HOST USING IEBGENER

```



Appendix A.255

JCL/SAS Code: PROCWTRF

```
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<CCARRILLO@MATHEMATICA-MPR.COM>  
RCPT TO:<CCARRILLO@MATHEMATICA-MPR.COM>  
DATA  
FROM: <CCARRILLO@MATHEMATICA-MPR.COM>  
TO: <CCARRILLO@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 PROC WATERFALL FILE COMPLETE  
//*
```

## Appendix A.256

JCL/SAS Code: DCFPULL

```
//#5413DCF JOB (12510000,T715,,SAS,,ITC9TH),CARRILLO,CLASS=T,
//  MSGCLASS=1,MSGLEVEL=(1,1),REGION=4096K,NOTIFY=#5413
/*SSAMAIN SASBASIC
//*****
/* PROJECT:   BUILD DAF20 (50214)
/* DIRECTOR:  MATT URATO
/* PROGRAM:   OPDR.TG.PRD.ETTW.#5413.DAF21.SCDR.PRDLIB(DCFPULL)
/* DESCRIPTION: CREATES EXTRACT OF DCF EVENT AND DCF REVIEW FILES
/*           TO DEVELOP DCF CDR VARIABLES
/* CONTACT:   CHRISTIAN CARRILLO
/* UPDATED:   ALEX BRYCE 09/23/2021 FOR DAF20
/*           CHRISTIAN CARRILLO FOR DAF21
/* EMAIL:     CCARRILLO@MATHEMATICA-MPR.COM
/*
//*****
//S1 EXEC SAS9,LOAD='DBP8.DB2.SDSNLOAD',SORT='15000',
//  WORK='15000,15000',COND=EVEN
//RVW DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.DCFRVW.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=TSILO,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//EVT DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.DCFEVT.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=TSILO,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HRVW DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.DCFHRVW.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=TSILO,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//HEVT DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.DCFHEVT.SA.V1,
//  DISP=(,CATLG,DELETE),
//  UNIT=TSILO,
//  VOL=(,,255),
//  DCB=(BLKSIZE=6144,DSORG=PS,LRECL=6144,RECFM=FS)
//SYSIN DD *
```

```
OPTIONS NOCENTER OBS=MAX;
```

```
LIBNAME RVW TAPE;
```

```
LIBNAME MDCF DB2 SSID=DBP8 SCHEMA=MDCF ;
```

Appendix A.256

JCL/SAS Code: DCFPULL

```
* CREATE SAS EXTRACT OF DCF REVIEW DATA;
PROC SQL;
CREATE TABLE RVW.RVW AS
SELECT * FROM MDCF.REV;
QUIT;
RUN;
```

```
PROC CONTENTS DATA=RVW.RVW; RUN;
```

```
LIBNAME EVT TAPE;
LIBNAME MDCF DB2 SSID=DBP8 SCHEMA=MDCF ;
```

```
* CREATE SAS EXTRACT OF DCF EVENT DATA;
PROC SQL;
CREATE TABLE EVT.EVT AS
SELECT * FROM MDCF.EVENT;
QUIT;
RUN;
```

```
PROC CONTENTS DATA=EVT.EVT; RUN;
```

```
LIBNAME HRVW TAPE;
LIBNAME MDCF DB2 SSID=DBP8 SCHEMA=MDCF;
```

```
* CREATE SAS EXTRACT OF DCF HISTORICAL REVIEW DATA;
PROC SQL;
CREATE TABLE HRVW.HRVW AS
SELECT * FROM MDCF.HREV;
QUIT;
RUN;
```

```
PROC CONTENTS DATA=HRVW.HRVW; RUN;
```

```
LIBNAME HEVT TAPE;
LIBNAME MDCF DB2 SSID=DBP8 SCHEMA=MDCF ;
```

```
* CREATE SAS EXTRACT OF DCF HISTORICAL EVENT DATA;
PROC SQL;
CREATE TABLE HEVT.HEVT AS
SELECT * FROM MDCF.HEVENT;
QUIT;
RUN;
```

```
PROC CONTENTS DATA= HEVT.HEVT; RUN;
```

Appendix A.256

JCL/SAS Code: DCFPULL

```
//*****  
//* SENDING EMAIL FROM HOST USING IEBGENER  
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<CCARRILLO@MATHEMATICA-MPR.COM>  
RCPT TO:<CCARRILLO@MATHEMATICA-MPR.COM>  
DATA  
FROM: <CCARRILLO@MATHEMATICA-MPR.COM>  
TO: <CCARRILLO@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 DCFPULL IS COMPLETE
```

## Appendix A.257

JCL/SAS Code: PROCDCF

```
//#5413CDR JOB (12510000,T715,,SAS,,ITC9FL),CARRILLO,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#5413
//*****
//* PROJECT:   DAF21 CONSTRUCTION
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#5413.DAF21.OVRPAY.PRDLIB(DCFCOMB1)
//* DESCRIPTION: COMBINE THE DCF REVIEW AND DCF EVENT TABLES
//*           CREATE DATASET AT CDR LEVEL AND AT BENE LEVEL
//*           INVESTIGATE FOR INCORPORATION INTO DAF
//* DATE:     11/9/2022 FOR DAF21
//* CONTACT:  CHRISTIAN CARRILLO
//* MPR EMAIL: CCARRILLO@MATHEMATICA-MPR.COM
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//* INPUT
//EVT  DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.DCFEVT.SA.V1,DISP=SHR
//RVW  DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.DCFRVW.SA.V1,DISP=SHR
//* OUTPUT
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.DCFCDRS.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* TEMP FILES
//TEMPRWMX DD DSN=&&TEMPX,UNIT=TSILO,DISP=(NEW,DELETE,DELETE)
//TEMPEVT DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMPRVW DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMPCOMB DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMPMAX DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMPBENE DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMPSSN DD DSN=&&TEMPX,UNIT=TSILO,DISP=(NEW,DELETE,DELETE)
//TEMPFIN DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN NOCENTER OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2020; /* CHANGE AS NEEDED */
```

## Appendix A.257

JCL/SAS Code: PROCDCF

```
%MACRO DOIT;

/* ROLL UP REV_SEQ_NUM AND WORK CDRS FOR ALL DCF BENES */
PROC SQL;
CREATE TABLE TEMPRWMX.SSNLIST AS
SELECT COSSN
      ,MAX(REV_SEQ_NUM) AS CDR_DCF_NUM
      ,SUM(CASE WHEN REV_TYP = 'W' THEN 1 ELSE 0 END) AS CDR_DCF_WRK_NUM
FROM RVW.RVW
WHERE NOT MISSING(ENDT) AND CID = '00'
GROUP BY COSSN;
QUIT;

PROC MEANS DATA=TEMPRWMX.SSNLIST
N MEAN MIN P10 P20 P30 P40 P50 P60 P70 P80 P90 P95 P99 MAX;
TITLE 'DISTRIBUTION OF MAX DCF REVIEW REV_SEQ_NUM FOR DCF REVIEW';
VAR CDR_DCF_NUM CDR_DCF_WRK_NUM;
RUN;
TITLE;

PROC MEANS DATA=TEMPRWMX.SSNLIST
N MEAN MIN P10 P20 P30 P40 P50 P60 P70 P80 P90 P95 P99 MAX;
TITLE 'DISTRIBUTION OF NONZERO WORK_CDR_COUNT';
WHERE CDR_DCF_WRK_NUM > 0;
VAR CDR_DCF_NUM CDR_DCF_WRK_NUM;
RUN;
TITLE;

/*PROCESS DCF EVT EXTRACT */
TITLE 'DCF EVENT CONTENTS';
PROC CONTENTS DATA = EVT.EVT; RUN;
TITLE ;

/*CREATE TEMPORARY SUBSET OF EVT DATA*/
DATA TEMPEVT.EVT;
  SET EVT.EVT (KEEP=COSSN REV_SEQ_NUM EVNT_SEQ_NUM EVNT_CD CID);
RUN;

/*SORT BY SSN REV_SEQ_NUM AND EVNT_SEQ_NUM */
PROC SORT DATA=TEMPEVT.EVT;
  BY COSSN REV_SEQ_NUM EVNT_SEQ_NUM;
RUN;

/*KEEP EVNT_CD FROM LAST EVNT_SEQ_NUM FOR EACH SSN-REV_SEQ_NUM PAIR*/
```

## Appendix A.257

JCL/SAS Code: PROCDCF

```
DATA TEMPEVT.EVT;
  SET TEMPEVT.EVT (KEEP=COSSN REV_SEQ_NUM EVNT_SEQ_NUM EVNT_CD);
  BY COSSN REV_SEQ_NUM EVNT_SEQ_NUM;
  IF LAST.REV_SEQ_NUM;
RUN;
```

```
/* PROCESS DCF REVIEW EXTRACT */
TITLE 'DCF REVIEW CONTENTS';
PROC CONTENTS DATA = RVW.RVW ; RUN;
TITLE;
```

```
DATA TEMPRVW.RVW;
  SET RVW.RVW (KEEP=COSSN REV_SEQ_NUM REV_TYP CTGY_CD ORIG_CD
              ENDT STDT CID);
RUN;
```

```
PROC SORT DATA=TEMPRVW.RVW;
  BY COSSN REV_SEQ_NUM;
RUN;
```

```
/*TACK ON CDR_DCF_NUM ONTO RVW FILE*/
DATA TEMPRVW.RVW;
  MERGE TEMPRVW.RVW (IN=INRVW)
        TEMPRWMX.SSNLIST (IN=INLST);
  BY COSSN;
  IF INRVW;
  IN_RVW = INRVW;
  IN_LST = INLST;
RUN;
```

```
PROC FREQ DATA=TEMPRVW.RVW;
TITLE 'CHECK DCF RVW MERGE';
TABLES IN_RVW * IN_LST / LIST MISSING;
RUN; TITLE;
```

```
/* MERGE WITH DCF EVENT EXTRACT TO CREATE CDR-LEVEL DATASET*/
DATA TEMPComb.COMBINED_RVWEVT;
  MERGE TEMPRVW.RVW (IN=INRVW)
        TEMPEVT.EVT (IN=INEVT);
  BY COSSN REV_SEQ_NUM;

  IN_RVW = INRVW;
  IN_EVT = INEVT;
```

## Appendix A.257

JCL/SAS Code: PROCDCF

```
*IDENTIFY IN-PROGRESS REVIEWS (CDRS WITH MISSING END DATE) TO DROP;  
IF MISSING(ENDT) THEN DROP_ENDT = 1;  
ELSE DROP_ENDT = 0;
```

```
*IDENTIFY NON '00' CIDS TO DROP - BENES OWN SSN IS NOT AVAILBLE;  
IF CID = '00' THEN DROP_CID = 0;  
ELSE DROP_CID = 1;
```

```
*PRODUCE START AND END YEAR TO COUNT CDRS;  
STDT_YEAR = YEAR(STDT);  
ENDT_YEAR = YEAR(ENDT);
```

```
RUN;
```

```
PROC FREQ DATA=TEMPCOMB.COMBINED_RVWEVT;  
TITLE "CHECK RVW EVT MERGE";  
TABLES IN_RVW * IN_EVT / LIST MISSING;  
TABLES DROP_ENDT * DROP_CID / LIST MISSING;  
RUN; TITLE "" ;
```

```
PROC FREQ DATA=TEMPCOMB.COMBINED_RVWEVT;  
TITLE "FREQ OF CDR BY START YEAR AND END YEAR - EXCLUSIONS APPLIED";  
WHERE DROP_ENDT = 0 AND DROP_CID = 0 ;  
TABLES STDT_YEAR / LIST MISSING;  
TABLES ENDT_YEAR / LIST MISSING;  
RUN; TITLE "";
```

```
PROC FREQ DATA=TEMPCOMB.COMBINED_RVWEVT;  
TITLE "FREQ OF REV_TYP EVNT_CD AND CTGY_CD - EXCLUSIONS APPLIED";  
WHERE DROP_ENDT = 0 AND DROP_CID = 0 ;  
TABLES REV_TYP / LIST MISSING;  
TABLES EVNT_CD / LIST MISSING;  
TABLES REV_TYP * CTGY_CD / LIST MISSING;  
RUN; TITLE "";
```

```
PROC FREQ DATA=TEMPCOMB.COMBINED_RVWEVT;  
TITLE "FREQ OF OTHER VARIABLES BEFORE ROLL UP - EXCLUSIONS APPLIED";  
WHERE DROP_ENDT = 0 AND DROP_CID = 0 ;  
TABLES REV_SEQ_NUM / LIST MISSING;  
TABLES CDR_DCF_NUM / LIST MISSING;  
TABLES CDR_DCF_WRK_NUM / LIST MISSING;  
TABLES ORIG_CD / LIST MISSING;  
TABLES STDT / LIST MISSING;  
TABLES ENDT / LIST MISSING;  
FORMAT STDT ENDT YYMMDD.;
```



## Appendix A.257

JCL/SAS Code: PROCDCF

```
RUN; TITLE """;

/*FIND MAX AND AND 99TH PERCENTILE REV_SEQ_NUM*/
/*WE NEED THESE VALUES TO DEVELOP BENE-LEVEL ARRAYS*/
%MACRO MAXVAL;

%GLOBAL MAX_NUM;
%GLOBAL P99_NUM;

PROC MEANS DATA=TEMPRWMX.SSNLIST MAX P99;
  VAR CDR_DCF_NUM;
  OUTPUT OUT=TEMPMAX.REVMAX MAX=MAX_NUM P99=P99_NUM;
RUN;

/*
PROC MEANS DATA=TEMPCOMB.COMBINED_RVWEVT MAX P99;
  WHERE DROP_ENDT = 0 AND DROP_CID = 0 ;
  VAR REV_SEQ_NUM;
  OUTPUT OUT=TEMPMAX.REVMAX MAX=MAX_NUM P99=P99_NUM;
RUN;
*/

PROC PRINT DATA=TEMPMAX.REVMAX;
TITLE "VALUES OF P99 AND MAX REV_SEQ_NUM";
VAR P99_NUM MAX_NUM;
RUN;

DATA TEMPMAX.REVMAX;
  SET TEMPMAX.REVMAX;
  CALL SYMPUTX("MAX_NUM",PUT(MAX_NUM,BEST32.));
  CALL SYMPUTX("P99_NUM",PUT(P99_NUM,BEST32.));
RUN;

%PUT MAXIMUM VALUE FOR REV_SEQ_NUM IS &&MAX_NUM.;
%PUT 99TH PERCENTILE FOR REV_SEQ_NUM IS &&P99_NUM.;

%MEND;
%MAXVAL;

/*APPLY EXCLUSIONS BEFORE COLLAPSING*/
PROC SORT DATA=TEMPCOMB.COMBINED_RVWEVT
  (WHERE=(DROP_ENDT=0 AND DROP_CID=0 ));
BY COSSN REV_SEQ_NUM;
RUN;
```

## Appendix A.257

JCL/SAS Code: PROCDCF

```
/* COLLAPSE CDR LEVEL DATA INTO BENE LEVEL DATA */
DATA TEMPBENE.RVWEVT_BENE
  (KEEP= COSSN CDR.);
SET TEMPCOMB.COMBINED_RVWEVT;
  BY COSSN REV_SEQ_NUM;

/* CREATE ARRAYS FOR N VARIABLES FOR PROCESSING */
ARRAY ATYPE (*) $ %DO MM=1 %TO &&MAX_NUM; CDR_TYPE_&MM. %END;;
ARRAY ACTGY (*) $ %DO MM=1 %TO &&MAX_NUM; CDR_TYPEDET_&MM. %END;;
ARRAY AORIG (*) $ %DO MM=1 %TO &&MAX_NUM; CDR_ORIG_CD_&MM. %END;;
ARRAY ASTDT (*) %DO MM=1 %TO &&MAX_NUM; CDR_START_DATE_&MM. %END;;
ARRAY AENDT (*) %DO MM=1 %TO &&MAX_NUM; CDR_END_DATE_&MM. %END;;
ARRAY AEVNT (*) $ %DO MM=1 %TO &&MAX_NUM; CDR_FINAL_EVENT_&MM. %END;;

FORMAT CDR_START_DATE_ : CDR_END_DATE_ : YMMMD.;

RETAIN %DO MM=1 %TO &&MAX_NUM; CDR_TYPE_&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM; CDR_TYPEDET_&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM; CDR_ORIG_CD_&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM; CDR_START_DATE_&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM; CDR_END_DATE_&MM. %END;;
RETAIN %DO MM=1 %TO &&MAX_NUM; CDR_FINAL_EVENT_&MM. %END;;

LABEL CDR_DCF_NUM = "NUMBER OF CDRS, DCF"
  CDR_DCF_WRK_NUM = "NUMBER OF WORK CDRS, DCF"
  %DO MM=1 %TO &&MAX_NUM;
  CDR_TYPE_&MM. = "DCF CDR REVIEW TYPE &MM."
  CDR_TYPEDET_&MM. = "DCF CDR REVIEW TYPE DETAIL &MM."
  CDR_ORIG_CD_&MM. = "DCF CDR ORIGINATION &MM."
  CDR_START_DATE_&MM. = "DATE THE CDR REVIEW STARTED &MM."
  CDR_END_DATE_&MM. = "DATE THE CDR REVIEW ENDED &MM."
  CDR_FINAL_EVENT_&MM. = "CDF FINAL EVENT CODE &MM."
  %END;;

/* INITIALIZE VARIABLES */
IF FIRST.COSSN THEN DO;
  DO MM=1 TO DIM(ATYPE);
    ATYPE(MM) = "";
    ACTGY(MM) = "";
    AORIG(MM) = "";
    ASTDT(MM) = .;
    AENDT(MM) = .;
    AEVNT(MM) = "";
  END;
END;
```

## Appendix A.257

JCL/SAS Code: PROCDCF

```
/* FILL IN ARRAYS BASED ON REV_SEQ_NUM*/
ATYPE(REV_SEQ_NUM) = REV_TYP;
ACTGY(REV_SEQ_NUM) = CTGY_CD;
AORIG(REV_SEQ_NUM) = ORIG_CD;
ASTDT(REV_SEQ_NUM) = STDT;
AENDT(REV_SEQ_NUM) = ENDT;
AEVNT(REV_SEQ_NUM) = EVNT_CD;

/* IF LAST SSN THEN OUTPUT*/
IF LAST.COSSN THEN OUTPUT;

RUN;

/*PROCESS SO THAT WE ONLY KEEP 99 PERCENT OF CDRS*/
DATA TEMPFIN.RVWEVT_FINAL;
SET TEMPBENE.RVWEVT_BENE(RENAME=(COSSN=SSN));
BY SSN;

/* CREATE ARRAYS FOR N VARIABLES FOR PROCESSING */
ARRAY ATYPE (*) $ %DO MM=1 %TO &&MAX_NUM; CDR_TYPE_&MM. %END;;
ARRAY ACTGY (*) $ %DO MM=1 %TO &&MAX_NUM; CDR_TYPERDET_&MM. %END;;
ARRAY AORIG (*) $ %DO MM=1 %TO &&MAX_NUM; CDR_ORIG_CD_&MM. %END;;
ARRAY ASTDT (*) %DO MM=1 %TO &&MAX_NUM; CDR_START_DATE_&MM. %END;;
ARRAY AENDT (*) %DO MM=1 %TO &&MAX_NUM; CDR_END_DATE_&MM. %END;;
ARRAY AEVNT (*) $ %DO MM=1 %TO &&MAX_NUM; CDR_FINAL_EVENT_&MM. %END;;

/* CREATE ARRAYS FOR FINAL VARIABLES FOR OUTPUT */
/* CUTTING OFF AT 99TH PERCENTILE OF REV_SEQ_NUM */
/* IN CONSULTATION WITH MICHAEL T. ANDERSON */
ARRAY FTYPE (*) $ %DO MM=1 %TO &&P99_NUM.; CDR_DCF_REV_TYP&MM. %END;;
ARRAY FCTGY (*) $ %DO MM=1 %TO &&P99_NUM.; CDR_DCF_REV_DET&MM. %END;;
ARRAY FORIG (*) $ %DO MM=1 %TO &&P99_NUM.; CDR_DCF_ORIG_CD&MM. %END;;
ARRAY FSTDT (*) %DO MM=1 %TO &&P99_NUM.; CDR_DCF_STDT&MM. %END;;
ARRAY FENDT (*) %DO MM=1 %TO &&P99_NUM.; CDR_DCF_ENDT&MM. %END;;
ARRAY FEVNT (*) $ %DO MM=1 %TO &&P99_NUM.; CDR_DCF_FIN_EVT&MM. %END;;

FORMAT CDR_DCF_STDT: CDR_DCF_ENDT: YYMMD.;

LABEL %DO MM=1 %TO &&P99_NUM.;
    CDR_DCF_REV_TYP&MM. = "DCF CDR REVIEW TYPE &MM."
    CDR_DCF_REV_DET&MM. = "DCF CDR REVIEW TYPE DETAIL &MM."
    CDR_DCF_ORIG_CD&MM. = "DCF CDR ORIGINATION &MM."
    CDR_DCF_STDT&MM. = "DCF CDR START DATE &MM."
    CDR_DCF_ENDT&MM. = "DCF CDR END DATE &MM."
```

## Appendix A.257

JCL/SAS Code: PROCDCF

```
        CDR_DCF_FIN_EVT&MM. = "DCF CDR FINAL EVENT CODE &MM."
        %END; ;

/* INITIALIZE OUTPUT VARIABLES */
IF FIRST.SSN THEN DO;
  DO MM=1 TO DIM(FTYPE);
    FTYPE(MM) = "";
    FCTGY(MM) = "";
    FORIG(MM) = "";
    FSTDT(MM) = .;
    FENDT(MM) = .;
    FEVNT(MM) = "";
  END;
END;

/* IF CDR_DCF_NUM <= 99TH PERCENTILE THEN RETAIN ALL CDRS */
/* IF CDR_DCF_NUM > 99TH PERCENTILE THEN RETAIN MOST RECENT CDRS */
/* EXAMPLE IF CDR_DCF_NUM = 20 AND 99TH PERCENTILE IS 12    */
/* THEN START AT 20-12+1 = 9TH CDR.                        */
_START = CDR_DCF_NUM - &&P99_NUM. + 1;
IF _START < 1 THEN _START = 1;
_END = CDR_DCF_NUM;
DO I = _START TO _END BY 1;
  /* SET INDICATOR VARIABLE FOR THE FINAL ARRAY */
  IF I = _START THEN J = 1;
  ELSE J = J + 1;

  FTYPE(J) = ATYPE(I);
  FCTGY(J) = ACTGY(I);
  FORIG(J) = AORIG(I);
  FSTDT(J) = ASTDT(I);
  FENDT(J) = AENDT(I);
  FEVNT(J) = AEVNT(I);
END;
RUN;

PROC MEANS DATA=TEMPFIN.RVWEVT_FINAL
  N MEAN MIN P10 P20 P30 P40 P50 P60 P70 P80 P90 P95 P99 MAX;
  TITLE 'DISTRIBUTION OF MAX DCF REVIEW REV_SEQ_NUM FOR FINAL FILE';
  VAR CDR_DCF_NUM CDR_DCF_WRK_NUM;
RUN;
TITLE;

PROC PRINT DATA = TEMPFIN.RVWEVT_FINAL (OBS = 30);
  TITLE "CHECK INDEX OF CDR_DCF_NUM TO MAKE SURE LOOP IS CORRECT";
```

## Appendix A.257

JCL/SAS Code: PROCDCF

```
WHERE RANUNI(50214) <= 0.1 AND CDR_DCF_NUM > 10;
VAR CDR_DCF_NUM _START _END;
RUN; TITLE "";

PROC PRINT DATA = TEMPFIN.RVWEVT_FINAL (OBS = 5) HEADING=VERTICAL;
TITLE "CHECK VARIABLE CONSTRUCTION WHERE CDR_DCF_NUM <= 99 PCT";
WHERE RANUNI(50214) <= 0.1 AND CDR_DCF_NUM <= &&P99_NUM.;
VAR _START _END CDR_DCF_STDT:
  CDR_START_DATE_1 - CDR_START_DATE_&&P99_NUM.;
RUN; TITLE "";

PROC PRINT DATA = TEMPFIN.RVWEVT_FINAL (OBS = 5) HEADING=VERTICAL;
TITLE "CHECK VARIABLE CONSTRUCTION WHERE CDR_DCF_NUM >= 97";
WHERE CDR_DCF_NUM >= 97;
VAR _START _END CDR_DCF_STDT: CDR_START_DATE_86-CDR_START_DATE_98;
RUN; TITLE "";

PROC FREQ DATA=TEMPFIN.RVWEVT_FINAL (KEEP = CDR_DCF_);
TITLE 'FREQ OF ALL DCF_CDR VARIABLES IN FINAL FILE';
TABLES _ALL_ / LIST MISSING;
FORMAT CDR_DCF_STDT: CDR_DCF_ENDT: YYMMD.;
RUN;
TITLE;

/*GENERATE THE FINAL FILE*/
DATA OUT.DCFCDRS;
SET TEMPFIN.RVWEVT_FINAL (KEEP = SSN CDR_DCF_);
RUN;

TITLE "FINAL DCF CONTENTS AND PRINT OUT";
PROC PRINT DATA=OUT.DCFCDRS (OBS=10); RUN;
PROC CONTENTS DATA = OUT.DCFCDRS; RUN;

%MEND;
%DOIT;

//*****
/* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
```

Appendix A.257

JCL/SAS Code: PROCDCF

```
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<CCARRILLO@MATHEMATICA-MPR.COM>  
RCPT TO:<CCARRILLO@MATHEMATICA-MPR.COM>  
DATA  
FROM: <CCARRILLO@MATHEMATICA-MPR.COM>  
TO: <CCARRILLO@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 PROCDCF IS COMPLETE
```

Appendix A.258

JCL/SAS Code: COMBSCDR

```
//#5413CDR JOB (12510000,T715,,SAS,,ITC9FL),CARRILLO-PAZ,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#5413
//*****
//* PROJECT:   DAF20 CONSTRUCTION
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#5413.DAF21.SCDR.PRDLIB(COMBSCDR)
//* DESCRIPTION: COMBINE DCF CDR AND WATERFALL CDR THEN
//*           RESTRICT COMBINED CDRS TO DAF BENES
//* DATE:     12/21/2021 FOR DAF21
//* CONTACT:  CHRISTIAN CARRILLO-PAZ
//* MPR EMAIL: CCARRILLO@MATHEMATICA-MPR.COM
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//* INPUT
//DCF  DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.DCFCDRS.SA.V1,DISP=SHR
//WTRF DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.WTRFCDRS.SA.V1,DISP=SHR
//DMG  DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.DMG,DISP=SHR
//MED  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDRMED.SA.V1,DISP=SHR
//* OUTPUT
//OUT  DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.SCDR,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,99),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUTCE DD DSN=OPDR.TG.PRD.ETTW.#5413.DAF21P.SCDRCON.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//* TEMP FILES
//TEMPCDR DD DSN=&&TEMPX,UNIT=TSILO,DISP=(NEW,DELETE,DELETE)
//TEMPVAL DD DSN=&&TEMPX,UNIT=TSILO,DISP=(NEW,DELETE,DELETE)
//TEMPOUT DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,99)
//SYSIN DD *
```

OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN NOCENTER OBS=MAX;

%MACRO DOIT;

/\*MERGE WATERFALL AND DCF CDR WITH DMG FILE \*/

DATA TEMPCDR.CDR;

MERGE WTRF.WATERFALL (IN=INWTRF)

DCF.DCFCDRS (IN=INDCF)

DMG.DMG (IN=INDMG KEEP=SSN)

MED.MEDCDR (IN=INMEDCDR);

## Appendix A.258

JCL/SAS Code: COMBSCDR

```
BY SSN;
```

```
/*WE KEEP ALL RECORDS IN DMG REGARDLESS IF THEY HAVE A CDR*/  
IF INDMG;
```

```
/*QA MERGE*/  
IN_WTRF = INWTRF;  
IN_DCF = INDCF;  
IN_DMG = INDMG;  
IN_MED = INMEDCDR;
```

```
/* SET MISSING CDRS TO 0 */  
IF NOT INWTRF THEN CDR_WFAL_NUM = 0;  
IF NOT INDCF THEN CDR_DCF_NUM = 0;  
IF NOT INDCF THEN CDR_DCF_WRK_NUM = 0;
```

```
RUN;
```

```
PROC FREQ DATA=TEMPCDR.CDR;  
TITLE "CHECK CDR MERGE";  
TABLES IN_WTRF * IN_DCF * IN_DMG * IN_MED / LIST MISSING;  
RUN; TITLE;
```

```
PROC FREQ DATA=TEMPCDR.CDR;  
TITLE "IF NO WATERFALL CDR THEN CHECK THAT CDR_WFAL_NUM = 0";  
WHERE IN_WTRF = 0;  
TABLES IN_WTRF * IN_DCF * IN_DMG CDR_WFAL_NUM / LIST MISSING;  
RUN;
```

```
PROC FREQ DATA=TEMPCDR.CDR;  
TITLE "IF NO DCF CDR THEN CHECK THAT CDR_DCF_NUM = 0";  
WHERE IN_DCF = 0;  
TABLES IN_DCF * IN_WTRF * IN_DMG * CDR_DCF_NUM / LIST MISSING;  
TABLES IN_DCF * IN_WTRF * IN_DMG * CDR_DCF_WRK_NUM / LIST MISSING;  
RUN;
```

```
PROC MEANS DATA=TEMPCDR.CDR MAXDEC=1  
N MEAN MIN P10 P20 P30 P40 P50 P60 P70 P80 P90 P95 P99 MAX;  
TITLE 'DISTRIBUTION OF CDR_*_NUM FOR ALL DAF BENES';  
WHERE IN_DMG = 1;  
VAR CDR_DCF_NUM CDR_DCF_WRK_NUM CDR_WFAL_NUM;  
RUN; TITLE;
```

```
PROC MEANS DATA=TEMPCDR.CDR MAXDEC=1
```



## Appendix A.258

JCL/SAS Code: COMBSCDR

```
N MEAN MIN P10 P20 P30 P40 P50 P60 P70 P80 P90 P95 P99 MAX;
TITLE 'DISTRIBUTION OF CDR_*_NUM FOR DAF DCF BENES';
WHERE IN_DMG = 1 AND IN_DCF = 1;
VAR CDR_DCF_NUM CDR_DCF_WRK_NUM;
RUN; TITLE;
```

```
PROC MEANS DATA=TEMPCDR.CDR MAXDEC=1
N MEAN MIN P10 P20 P30 P40 P50 P60 P70 P80 P90 P95 P99 MAX;
TITLE 'DISTRIBUTION OF CDR_*_NUM FOR DAF WATERFALL BENES';
WHERE IN_DMG = 1 AND IN_WTRF = 1;
VAR CDR_WFAL_NUM;
RUN; TITLE;
```

```
/*GENERATE THE FINAL FILE*/
DATA OUT.SCDR (DROP= IN_WTRF IN_DCF IN_DMG
                CDR_DCF_WRK_NUM IN_MED );
SET TEMPCDR.CDR;
```

```
/*REMOVING LABEL FROM SSN*/
LABEL SSN = ' ';
RUN;
```

```
/* COMMENT PRINT IN OR OUT AS NEEDED */
PROC PRINT DATA=OUT.SCDR (DROP=SSN OBS=10);
TITLE 'PRINT OF 10 RANDOM RECORDS. SSN EXCLUDED FROM PRINT';
WHERE RANUNI(50214) < 0.01 AND (CDR_DCF_NUM > 1 OR CDR_WFAL_NUM > 1);
RUN; TITLE;
/**/
```

```
PROC CONTENTS DATA = OUT.SCDR OUT=OUTCE.SCDR; RUN;
```

```
PROC MEANS DATA=OUT.SCDR
N MEAN MIN P10 P20 P30 P40 P50 P60 P70 P80 P90 P95 P99 MAX;
TITLE 'DISTRIBUTION OF CDR_*_NUM FOR ALL DAF BENES';
VAR CDR_DCF_NUM CDR_WFAL_NUM;
RUN; TITLE;
```

```
PROC FREQ DATA=OUT.SCDR (DROP = SSN CDR_DCF_NUM CDR_WFAL_NUM);
TABLES _ALL_ / LIST MISSING;
FORMAT CDR_WFAL_ROM: CDR_WFAL_INIT_STRDT: CDR_WFAL_INIT_DODEC:
        CDR_WFAL_RCN_STRDT: CDR_WFAL_RCN_DODEC: CDR_WFAL_ALJ_RCPTDT:
        CDR_WFAL_ALJ_DODEC: CDR_WFAL_AC_DODEC: CDR_WFAL_DC_DODEC:
        CDR_WFAL_ND_STRDT: /*YYMMDD10.*/
        CDR_WFAL_FIN_ACT_DT: /*YYMMD7.*/
        CDR_DCF_STDT: CDR_DCF_ENDT:
```

Appendix A.258

JCL/SAS Code: COMBSCDR

```
    CDR_DCF_MED_STDT
    CDR_DCF_MED_MRE_DRY_MDT
    CDR_DCF_MED_NDDSS_DT
    CDR_DCF_MED_INSRT_TS
    CDR_DCF_MED_LU_TS
    YMMMD7.;
RUN;

%MEND;
%DOIT;

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2  DD SYSOUT=(B,SMTP)
//SYSUT1  DD *
HELO MVSHOST
MAIL FROM:<CCARRILLO@MATHEMATICA-MPR.COM>
RCPT TO:<CCARRILLO@MATHEMATICA-MPR.COM>
DATA
FROM: <CCARRILLO@MATHEMATICA-MPR.COM>
TO: <CCARRILLO@MATHEMATICA-MPR.COM>
SUBJECT: CDRCOMB IS COMPLETE
```

## Appendix A.259

JCL/SAS Code: PROCMED3

```

//#6502MED JOB (12510000,T715,,SAS,,ITC9FL),ALEX-BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   DAF21 CONSTRUCTION
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DEMO.PRDLIB(DCFMED)
//* DESCRIPTION: CONVERT THE DCF MEDICAL TABLE INTO AN SSN-LEVEL
//*            FILE WITH OTHER VARIABLES CONVERTED INTO N VARIABLES
//*            FOR MERGE WITH SCDR FILE - UTILIZES MEDEX YMMM LOGIC
//* DATE:      04/22/2022 FOR DAF21
//* CONTACT:   ALEX BRYCE
//* MPR EMAIL: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//* INPUT
//MED  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.DCFMED.SA.V1,DISP=SHR
//FIND  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.MASTFIND.SA.V1,DISP=SHR
//* OUTPUT
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.CDRMED.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP2 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP3 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//TEMP4 DD DSN=&&TEMPX,UNIT=SYSDA,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(4000,2000),RLSE),VOL=(,,10)
//SYSIN DD *

OPTIONS LS=132 COMPRESS=YES MPRINT MACROGEN NOCENTER OBS=MAX;

%LET BEGYR=1994;
%LET ENDYR=2021; /* CHANGE AS NEEDED */

%MACRO START;
* MERGE DCF MED FILE WITH DEMO SSN LINK FILE TO ELIMINATE SSNS
  THAT ARE NOT IN THE DAF *;

*SELECT SUBSET OF DCFMED WE NEED TO REDUCE STRAIN ON SQL PROCESSING;
DATA TEMP1.MED_SUBSET

```

## Appendix A.259

JCL/SAS Code: PROCMED3

```
(KEEP=COSSN CTL_DT DETN_SEQ_NUM MED_STDT LU_TS MDR DRY_PERM_CD
MED_TYP MRE_DRY_MDT LISTG_CD VORULE RBSS NDDSS_DT INSRT_TS
);
SET MED.MED
(WHERE=(LU_PGM_NM NE 'CDCNVCLM' AND
CID = '00' AND
COSSN NE '000000000' AND
COSSN = CLMSSN AND
MED_STDT NE .))
;
RUN;

PROC SORT DATA=TEMP1.MED_SUBSET;
BY COSSN;
RUN;

* MERGE MED WITH MASTFIND BY SSN *;
PROC SQL;
CREATE TABLE TEMP2.DCF AS
SELECT A.SSN,
      B.CTL_DT,
      B.DETN_SEQ_NUM,
      B.MED_STDT,
      B.LU_TS,
      B.MDR,
      B.DRY_PERM_CD,
      B.MED_TYP,
      B.MRE_DRY_MDT,
      B.LISTG_CD,
      B.VORULE,
      B.RBSS,
      B.NDDSS_DT,
      B.INSRT_TS
FROM FIND.MASTFIND AS A
INNER JOIN
(SELECT COSSN AS SSN,
      CTL_DT,
      DETN_SEQ_NUM,
      MED_STDT,
      LU_TS,
      MDR,
      DRY_PERM_CD,
      MED_TYP,
      MRE_DRY_MDT,
      LISTG_CD,
```

## Appendix A.259

JCL/SAS Code: PROCMED3

```
VORULE,
RBSS,
NDDSS_DT,
INSRT_TS
FROM TEMP1.MED_SUBSET) AS B
ON B.SSN=A.SSN
ORDER BY A.SSN, B.CTL_DT, B.DETN_SEQ_NUM, B.MED_STDT, B.LU_TS
;
QUIT;
```

```
* FIND THE MOST RECENT LU_TS *;
* FOR EACH SSN-CTL_DT-DETN_SEQ_NUM-MED_STDT GROUP;
PROC SQL;
CREATE TABLE TEMP3.DCF AS
SELECT *,
MAX(LU_TS) AS LU_TS_MAX
FROM TEMP2.DCF
GROUP BY SSN,CTL_DT,DETN_SEQ_NUM,MED_STDT;
QUIT;
```

```
* OUTPUT DATASET WITH MOST LU_TS *;
DATA TEMP3.DCF(DROP=LU_TS_MAX);
SET TEMP3.DCF;

IF LU_TS=LU_TS_MAX THEN OUTPUT;
RUN;
```

```
* FIND THE MOST RECENT MED_STDT FOR EACH SSN;
PROC SORT DATA=TEMP3.DCF;
BY SSN MED_STDT;
RUN;
```

```
* CREATE FINAL FILE WITH DATA FROM LATEST MED_STDT RECORD;
DATA OUT.MEDCDR (KEEP= SSN CDR_DCF_MED_.);
SET TEMP3.DCF (RENAME = (
MED_STDT = CDR_DCF_MED_STDT
MED_TYP = CDR_DCF_MED_TYP
MDR = CDR_DCF_MED_MDR
DRY_PERM_CD = CDR_DCF_MED_DRY_PERM_CD
MRE_DRY_MDT = CDR_DCF_MED_MRE_DRY_MDT
LISTG_CD = CDR_DCF_MED_LISTG_CD
VORULE = CDR_DCF_MED_VORULE
RBSS = CDR_DCF_MED_RBSS
NDDSS_DT = CDR_DCF_MED_NDDSS_DT
```

Appendix A.259

JCL/SAS Code: PROCMED3

```
INSRT_TS = CDR_DCF_MED_INSRT_TS
LU_TS = CDR_DCF_MED_LU_TS
));
```

```
BY SSN CDR_DCF_MED_STDT;
IF LAST.SSN;
```

LABEL

```
CDR_DCF_MED_STDT = "DCF CDR MEDICAL START DATE"
CDR_DCF_MED_TYP = "DCF CDR MEDICAL TYPE"
CDR_DCF_MED_MDR = "DCF CDR MEDICAL DIARY REASON"
CDR_DCF_MED_DRY_PERM_CD = "DCF CDR DIARY PERMANENT CODE"
CDR_DCF_MED_MRE_DRY_MDT =
  "DCF CDR MEDICAL REEXAMINATION DIARY MONTH DATE"
CDR_DCF_MED_LISTG_CD = "DCF CDR LISTING CODE"
CDR_DCF_MED_VORULE = "DCF CDR VOCATIONAL RULE"
CDR_DCF_MED_RBSS = "DCF CDR REGULATION BASIS CODE"
CDR_DCF_MED_NDDSS_DT =
  "DCF CDR NATIONAL DISABILITY DETERMINATION SERVICES SYSTEM DATE"
CDR_DCF_MED_INSRT_TS = "DCF CDR INSERTED TIME STAMP"
CDR_DCF_MED_LU_TS = "DCF CDR LAST UPDATED TIME STAMP"
```

```
RUN;
```

```
TITLE "CONTENTS FOR OUT MEDCDR";
PROC CONTENTS DATA=OUT.MEDCDR;
RUN; TITLE;
```

```
TITLE "PRINT OF 10 RECORDS (NO SSN)";
PROC PRINT DATA=OUT.MEDCDR (DROP=SSN OBS=10);
RUN; TITLE;
```

```
%MEND;
%START;
```

```
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
```

Appendix A.259

JCL/SAS Code: PROCMED3

```
DATA  
FROM: <ABRYCE@MATHEMATICA-MPR.COM>  
TO: <ABRYCE@MATHEMATICA-MPR.COM>  
SUBJECT: DCFMED IS COMPLETE
```

## Appendix A.260

JCL/SAS Code: P10SAMP

```
//#8047P10 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//*
//* *-----*
//*  ! OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(P10SAMP) !
//* *-----*
//*
//* CREATE PUF 10% SAMPLE FILE FROM DAF DMG
//* CONTACT INFO: ALEX BRYCE, ABRYCE@MATHEMATICA-MPR.COM,
//*
//* UPDATED 2/15/22 ALEX BRYCE FOR PUF20
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000'          * SPACE IN BLKS *
//*
//DMG DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.DMG,DISP=SHR
//* OLD15 DD DSN=OPDR.TG.PRD.ETTW.#6293.DAF15T.PUF.P10.SA.V1,DISP=SHR
//OLD17 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF17T.PUF.P10.SA.V1,DISP=SHR
//OLD18 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF18T.PUF.P10.SA.V1,DISP=SHR
//OLD19 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF19T.PUF.P10.SA.V1,DISP=SHR
//OLD20 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20T.PUF.P10.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//* TEMP15 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//*  UNIT=TSILO
//TEMP17 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TEMP18 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TEMP19 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TEMP20 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.PUF.P10.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *
```



## Appendix A.260

JCL/SAS Code: P10SAMP

```
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS = MAX MPRINT
MACROGEN;
```

```
%LET YR = 21; /*SET TO CURRENT DAF YEAR*/
%LET PREVYR = 20; /*SET TO PREVIOUS DAF YEAR*/
```

```
%MACRO DOIT;
/* SELECT RANDOM 10% OF DMG BENEFICIARIES */
DATA TEMP1.TEMP;
  SET DMG.DMG (KEEP=SSN);
  RAND_NUM=RANUNI(&YR.);
RUN;
```

```
DATA TEMP2.P10SAMP ;
  SET TEMP1.TEMP;
  IF RAND_NUM<=.1;
RUN;
```

```
PROC SORT DATA=TEMP2.P10SAMP OUT=TEMP3.P10SAMP;
  BY RAND_NUM;
RUN;
```

```
/* BUILD PUF PINS */
DATA OUT1.P10SAMP (KEEP = PUF PIN SSN);
  SET TEMP3.P10SAMP;
  NEWPIN = _N_ + 1000;
  PIN_T = PUT(NEWPIN, 8.);
```

```
LENGTH PUF PIN $14.;
PUF PIN = CATS("DAF&YR._" , PIN_T);
RUN;
```

```
PROC SORT DATA=OUT1.P10SAMP;
  BY SSN;
RUN;
```

```
PROC PRINT DATA=OUT1.P10SAMP (OBS=50);
  TITLE 'CHECK CONSTRUCTION OF PUF PINS';
  VAR SSN PUF PIN;
RUN;
```

```
/* CHECK TO MAKE SURE PUF SAMPLE IS DIFFERENT FROM PREVIOUS PUF S */
/* PROC SORT DATA=OLD15.P10SAMP OUT=TEMP15.P10SAMP_DAF15;
  BY SSN;
RUN; */
```

## Appendix A.260

JCL/SAS Code: P10SAMP

```
%DO YY = 17 %TO &PREVYR;
PROC SORT DATA=OLD&YY..P10SAMP OUT=TEMP&YY..P10SAMP_DAF&YY.;
  BY SSN;
RUN;
%END;

DATA TEMP1.SAMPLEMERGE;
MERGE OUT1.P10SAMP (KEEP = SSN IN=DAF&YR.)
  /*TEMP15.P10SAMP_DAF15 (KEEP = SSN IN=DAF15)*/
  %DO YY = 17 %TO &PREVYR.;
  TEMP&YY..P10SAMP_DAF&YY. (KEEP = SSN IN=DAF&YY.)
  %END;;

BY SSN;

*CREATE FLAGS FOR CROSSTABS;
/* IN_DAF15 = DAF15; */
%DO YY = 17 %TO &YR.;
  IN_DAF&YY. = DAF&YY.;
%END;

*ALTERNATE FREQ - TO USE WHEN WE HAVE TOO MANY PUF'S FOR CROSSTABS;
IN_CURR = DAF&YR.;
IN_PREV = /* DAF15 */ DAF17 %DO YY=18 %TO &PREVYR.;
  OR DAF&YY.
  %END;;
SSNCOUNT = SUM(OF IN_DAF:);
RUN;

PROC FREQ DATA=TEMP1.SAMPLEMERGE;
TITLE 'CHECK FOR OVERLAP BETWEEN PUF SAMPLES';
TABLES IN_CURR * IN_PREV / LIST MISSING;
TABLES SSNCOUNT / LIST MISSING;
RUN;

%MEND;
%DOIT;

//*****
/* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
```

Appendix A.260

JCL/SAS Code: P10SAMP

```
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF 10% SAMPLE
DAF21 PUF SAMPLE HAS FINISHED RUNNING
```

## Appendix A.261

JCL/SAS Code: JCLYRSP

```
//#8047MSP JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//*
//* *-----*
//* | THE MACRO SPLIT PROGRAM           |
//* *-----*
//*
//*
/** CREATED BY: SERGE LUKASHANETS, 5/26/2016
/** MODIFIED BY: ALEX BRYCE, 02/15/2022
/** E-MAIL ABRYCE@MATHEMATICA-MPR.COM
//*****
// SET REG='64M'
/*
//JS010 EXEC SAS9,
//      WORK='200000,100000', F10
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
/*
//***DATA FOR SPLIT
//IN1  DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.PUF.P10.SA.V1,DISP=SHR
//***OUT YEARLY FILES
//OUT94 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY94.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUT95 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY95.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUT96 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY96.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUT97 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY97.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUT98 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY98.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUT99 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY99.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//OUT00 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY00.SA.V1,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
```

## Appendix A.261

JCL/SAS Code: JCLYRSP

```
//OUT01 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY01.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT02 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY02.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT03 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY03.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT04 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY04.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT05 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY05.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT06 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY06.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT07 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY07.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT08 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY08.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT09 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY09.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY10.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT11 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY11.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY12.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT13 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY13.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT14 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY14.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT15 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY15.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
```

## Appendix A.261

JCL/SAS Code: JCLYRSP

```
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY16.SA.V1,
// DISP=(NEW,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY17.SA.V1,
// DISP=(NEW,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY18.SA.V1,
// DISP=(NEW,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY19.SA.V1,
// DISP=(NEW,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY20.SA.V1,
// DISP=(NEW,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY21.SA.V1,
// DISP=(NEW,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT OBS=MAX;

%MACRO DOIT;
%DO YEAR = 1994 %TO 2021;

%LET YR = %SUBSTR(&YEAR.,3,2);

DATA OUT&YR..P10SAMP;
SET IN1.P10SAMP;
RUN;
PROC CONTENTS DATA=OUT&YR..P10SAMP; RUN;

%END;
%MEND;
%DOIT;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
```

Appendix A.261

JCL/SAS Code: JCLYRSP

TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: DAF21 JCLYRSP ARE COMPLETE  
PUF SPLIT TO ANNUAL SAMP ARE COMPLETE.

## Appendix A.262

JCL/SAS Code: PUFDMG

```
//#8047DMG JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//*
//*  *-----*
//*  ! OPDR.TG.PRD.ETTW.#6502.DAF19.PUF.PRDLIB(DMGPUF) !
//*  *-----*
//*
//* CREATE PUF VARIABLES FROM DAF15 10% DM
//* CONTACT INFO: ALEX BRYCE, ABRYCE@MATHEMATICA-MPR.COM
//*
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000'          * SPACE IN BLKS *
//*
//DEMO DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.DMG,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.PUF.P10.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.DMG,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS = MAX MPRINT
MACROGEN;

* SET DAF YEAR MACRO - UPDATE FOR EACH YMAR;
%LET DAFYR = 2021;
%LET YR = 21;

TITLE1 "DAF&YR. PUF DMG PROCESSING";

* SORT BY SSN;
PROC SORT DATA=SAMP.P10SAMP OUT=TEMP1.SAMP;
  BY SSN;
RUN;
```



## Appendix A.262

JCL/SAS Code: PUFDMG

```
* RECODE ALL VARIABLES ACCORDING TO DOCUMENTATION;

DATA TEMP1.P10DMGPUF;
MERGE TEMP1.SAMP (IN=A)
      DEMO.DMG (IN=B);
BY SSN;
IF A;

/* CODE UNKNOWN SEX AS MISSING */
LENGTH SEX_PUF $2;
SEX_PUF = SEX;
IF SEX = 'U' OR MISSING(SEX) THEN SEX_PUF = '.M';

/* MASK DATE VARIABLES- ONES NOT RELIANT ON DOEI OR SSIELIG_FIRST */
%MACRO MASKDT(DT=,CUT=);

&DT._PUF = INTNX('MONTH',&DT.,0,'BEG') + 14;
IF &DT._PUF < &CUT. AND NOT MISSING(&DT._PUF) THEN &DT._PUF = &CUT.;
ELSE IF MISSING(&DT._PUF) THEN &DT._PUF = .M;
FORMAT &DT._PUF MMDDYY10.;

%MEND MASKDT;
%MASKDT(DT=DOBBEST,CUT='15JAN1930'D);
%MASKDT(DT=DODBEST,CUT='15JAN1996'D);
%MASKDT(DT=FRADATE,CUT='15FEB1996'D);

/*TOP CODE DOBBEST AND FRADATE*/
IF YEAR(DOBBEST_PUF) > &DAFYR. THEN DOBBEST_PUF = .M;
IF YEAR(DOBBEST_PUF) > &DAFYR. THEN FRADATE_PUF = .M;

/* MASK DATE MACRO FOR VARIABLES RELIANT ON DOEI */
%MACRO MASKDT2(DT=,YRDIFF=);

%LET CUTYR = %EVAL(&DAFYR. - &YRDIFF.);
%PUT &CUTYR.;
%LET CUT = "15JAN&CUTYR."D;
%PUT &CUT.;

IF MISSING(DOEI) AND MISSING(&DT) THEN &DT._PUF = .N;
ELSE IF MISSING(&DT) THEN &DT._PUF = .M;
ELSE IF NOT MISSING(&DT) THEN
  &DT._PUF = INTNX('MONTH',&DT.,0,'BEG') + 14;

IF &DT._PUF < &CUT. AND NOT MISSING(&DT._PUF)
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
    THEN &DT._PUF = &CUT.;
    FORMAT &DT._PUF MMDDYY10.;

%MEND MASKDT2;
%MASKDT2(DT=SSDI_ONSET,YRDIFF=50);
%MASKDT2(DT=BDOF1,YRDIFF=50);
%MASKDT2(DT=BDOF2,YRDIFF=50);
%MASKDT2(DT=DOEI,YRDIFF=50);
%MASKDT2(DT=DOEC,YRDIFF=50);
%MASKDT2(DT=BDOE_START1,YRDIFF=45);
%MASKDT2(DT=BDOE_START2,YRDIFF=45);
%MASKDT2(DT=TWPCMPLMNT1,YRDIFF=40);

*MASK DATES RELIANT ON DOEI WITH DIFFERENT MASK DATES;
%MACRO MASKDT2B(DT=,CUT=);

    IF MISSING(DOEI) AND MISSING(&DT) THEN &DT._PUF = .N;
    ELSE IF MISSING(&DT) THEN &DT._PUF = .M;
    ELSE IF NOT MISSING(&DT) THEN
        &DT._PUF = INTNX('MONTH',&DT.,0,'BEG') + 14;

    IF &DT._PUF < &CUT. AND NOT MISSING(&DT._PUF)
        THEN &DT._PUF = &CUT.;
    FORMAT &DT._PUF MMDDYY10.;

%MEND MASKDT2B;
%MASKDT2B(DT=LAF_MR_DATE,CUT='15MAR1996'D);

/* DO BDOE_TERM SEPARATELY AND CONDITION ON BDOE_START */
%MACRO TERM;
%DO I = 1 %TO 2;

%LET CUTYR = %EVAL(&DAFYR. - 35);
%LET CUT = "15JAN&CUTYR."D;

    IF MISSING(BDOE_TERM&I.) AND MISSING(DOEI)
        AND MISSING(BDOE_START&I.) THEN BDOE_TERM&I._PUF = .N;
    ELSE IF MISSING(BDOE_TERM&I.) AND MISSING(DOEI)
        AND NOT MISSING(BDOE_START&I.) THEN BDOE_TERM&I._PUF = .M;
    ELSE IF MISSING(BDOE_TERM&I.) AND NOT MISSING(DOEI)
        AND MISSING(BDOE_START&I.) THEN BDOE_TERM&I._PUF = .M;
    ELSE IF MISSING(BDOE_TERM&I.) AND NOT MISSING(DOEI)
        AND NOT MISSING(BDOE_START&I.) THEN BDOE_TERM&I._PUF = .M;
    ELSE IF NOT MISSING(BDOE_TERM&I.) THEN
        BDOE_TERM&I._PUF = INTNX('MONTH',BDOE_TERM&I.,0,'BEG') + 14;
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
ELSE IF NOT MISSING(BDOE_TERM&I) THEN
  BDOE_TERM&I._PUF = INTNX('MONTH',BDOE_TERM&I.,0,'BEG') + 14;

IF BDOE_TERM&I._PUF < &CUT. AND NOT MISSING(BDOE_TERM&I._PUF)
  THEN BDOE_TERM&I._PUF = &CUT.;
FORMAT BDOE_TERM&I._PUF MMDDYY10.;

%END;
%MEND TERM;
%TERM;

/* MASK DATE MACRO FOR VARIABLES RELIANT ON SSIELIG_FIRST */
/* SPLIT INTO A AND B MACROS FOR DIFF BOTTOM-CODING */
%MACRO MASKDT3A(DT=,CUT=);

IF MISSING(SSIELIG_FIRST) AND MISSING(&DT) THEN &DT._PUF = .N;
ELSE IF MISSING(&DT) THEN &DT._PUF = .M;
ELSE IF NOT MISSING(&DT) THEN
  &DT._PUF = INTNX('MONTH',&DT.,0,'BEG') + 14;

IF &DT._PUF < &CUT. AND NOT MISSING(&DT._PUF) THEN &DT._PUF = &CUT.;
FORMAT &DT._PUF MMDDYY10.;

%MEND MASKDT3A;

%MASKDT3A(DT=SSIELIG_FIRST,CUT='15JAN1974'D);
%MASKDT3A(DT=ELG_RD,CUT='15JAN1974'D);

%MACRO MASKDT3B(DT=,YRDIFF=);

%LET CUTYR = %EVAL(&DAFYR. - &YRDIFF.);
%LET CUT = "15JAN&CUTYR."D;

IF MISSING(SSIELIG_FIRST) AND MISSING(&DT) THEN &DT._PUF = .N;
ELSE IF MISSING(&DT) THEN &DT._PUF = .M;
ELSE IF NOT MISSING(&DT) THEN
  &DT._PUF = INTNX('MONTH',&DT.,0,'BEG') + 14;

IF &DT._PUF < &CUT. AND NOT MISSING(&DT._PUF) THEN &DT._PUF = &CUT.;
FORMAT &DT._PUF MMDDYY10.;

%MEND MASKDT3B;
%MASKDT3B(DT=CLM_FIL,YRDIFF=30);
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
/* MASK DATE MACRO FOR VARIABLES RELIANT ON BOTH */
%MACRO MASKDT4(DT=,CUT=);

IF MISSING(SSIELIG_FIRST) AND MISSING(DOEI) AND MISSING(&DT)
  THEN &DT._PUF = .N;
ELSE IF MISSING(&DT) THEN &DT._PUF = .M;
ELSE IF NOT MISSING(&DT) THEN
  &DT._PUF = INTNX('MONTH',&DT.,0,'BEG') + 14;

IF &DT._PUF < &CUT. AND NOT MISSING(&DT._PUF) THEN &DT._PUF = &CUT.;
FORMAT &DT._PUF MMDDYY10.;

%MEND MASKDT4;

%MASKDT4(DT=FIRSTDD,CUT='15JAN1988'D);

%MACRO DDLOOP;
%DO I=1 %TO 5;
  %MASKDT4(DT=DD&I.,CUT='15JAN1988'D);
  %MASKDT4(DT=DODEC&I.,CUT='15JAN1988'D);

  LABEL DD&I._PUF = "MEDICAL CONTINUING DISABILITY REVIEW DATE &I."
    DODEC&I._PUF = "DATE OF MEDICAL DECISION (SSA OR STATE DDS) &I";
%END;
%MEND DDLOOP;
%DDLOOP;

* TOC_START DATES;
%MACRO TOCSTART;
%DO I= 1 %TO 2;

IF MISSING(TOC&I) AND MISSING(TOC_START&I)
  THEN TOC_START&I._PUF = .N;
ELSE IF MISSING(TOC_START&I) THEN TOC_START&I._PUF = .M;
ELSE IF NOT MISSING(TOC_START&I) THEN
  TOC_START&I._PUF = INTNX('MONTH',TOC_START&I.,0,'BEG') + 14;

IF TOC_START&I._PUF < '15DEC2007'D AND NOT MISSING(TOC_START&I._PUF)
  THEN TOC_START&I._PUF = '15DEC2007'D;
FORMAT TOC_START&I._PUF MMDDYY10.;

%END;
%MEND TOCSTART;
%TOCSTART;
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```

* BIC_HIST_START DATES AND TOC_HIST_START DATES;
%MACRO HISTSTART(DT=,I=,CUT=);

IF MISSING(DOEI) THEN &DT._START&I._PUF = .N;
ELSE IF MISSING(&DT._START&I) THEN &DT._START&I._PUF = .M;
ELSE IF NOT MISSING(&DT._START&I) THEN
  &DT._START&I._PUF = INTNX('MONTH',&DT._START&I.,0,'BEG') + 14;

IF &DT._START&I._PUF < &CUT. AND NOT MISSING(&DT._START&I._PUF)
  THEN &DT._START&I._PUF = &CUT.;
FORMAT &DT._START&I._PUF MMDDYY10.;
%MEND HISTSTART;

%HISTSTART(DT=BIC_HIST,I=1,CUT='15JAN1965'D);
%HISTSTART(DT=BIC_HIST,I=2,CUT='15JAN1984'D);
%HISTSTART(DT=BIC_HIST,I=3,CUT='15JUN1996'D);
%HISTSTART(DT=TOC_HIST,I=1,CUT='15DEC2007'D);
%HISTSTART(DT=TOC_HIST,I=2,CUT='15DEC2007'D);
%HISTSTART(DT=TOC_HIST,I=3,CUT='15DEC2007'D);

/* TOP CODE DOEI FOR EACH PUF */

IF DOEI_PUF > "31DEC&DAFYR."D AND NOT MISSING(DOEI_PUF)
  THEN DOEI_PUF = .N;

/* COMBINE CATEGORIES FOR DXPRIBEST AND DXSECBEST */

%MACRO DGROUP (VAR=,GROUP=);

*** INFECTIOUS/PARASITIC DISEASES ***;
IF &VAR>='0020' AND &VAR<='0189' THEN &GROUP=6;
ELSE IF &VAR>='0200' AND &VAR<='0279' THEN &GROUP=6;
ELSE IF &VAR>='0300' AND &VAR<='0419' THEN &GROUP=6;
ELSE IF &VAR>='0430' AND &VAR<='0579' THEN &GROUP=6;
ELSE IF &VAR>='0600' AND &VAR<='0669' THEN &GROUP=6;
ELSE IF &VAR>='0700' AND &VAR<='0889' THEN &GROUP=6;
ELSE IF &VAR>='0900' AND &VAR<='1049' THEN &GROUP=6;
ELSE IF &VAR>='1100' AND &VAR<='1189' THEN &GROUP=6;
ELSE IF &VAR>='1200' AND &VAR<='1359' THEN &GROUP=6;
ELSE IF &VAR>='1370' AND &VAR<='1399' THEN &GROUP=6;
ELSE IF &VAR>='7710' AND &VAR<='7719' THEN &GROUP=6;

*** NEOPLASMS ***;
ELSE IF &VAR>='0420' AND &VAR<='0429' THEN &GROUP=6;

```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
ELSE IF &VAR>='1400' AND &VAR<='1659' THEN &GROUP=6;
ELSE IF &VAR>='1700' AND &VAR<='1769' THEN &GROUP=6;
ELSE IF &VAR>='1780' AND &VAR<='2089' THEN &GROUP=6;
ELSE IF &VAR>='2100' AND &VAR<='2399' THEN &GROUP=6;
```

```
*** ENDOCRINE ***;
```

```
ELSE IF &VAR>='2400' AND &VAR<='2469' THEN &GROUP=4;
ELSE IF &VAR>='2500' AND &VAR<='2539' THEN &GROUP=4;
ELSE IF &VAR>='2550' AND &VAR<='2559' THEN &GROUP=4;
ELSE IF &VAR>='2600' AND &VAR<='2799' THEN &GROUP=4;
```

```
*** DISEASES OF THE BLOOD ***;
```

```
ELSE IF &VAR>='2800' AND &VAR<='2899' THEN &GROUP=7;
ELSE IF &VAR>='7720' AND &VAR<='7739' THEN &GROUP=7;
ELSE IF &VAR>='7760' AND &VAR<='7769' THEN &GROUP=7;
```

```
*** AUTISM ***;
```

```
ELSE IF &VAR>='2990' AND &VAR<='2999' THEN &GROUP=1;
```

```
*** DEVELOPMENTAL DISORDERS ***;
```

```
ELSE IF &VAR>='3150' AND &VAR<='3159' THEN &GROUP=1;
```

```
*** CHILDHOOD/ADOLESCENT DISORDERS ***;
```

```
ELSE IF &VAR>='3120' AND &VAR<='3139' THEN &GROUP=1;
```

```
*** INTELLECTUAL DISABILITY ***;
```

```
ELSE IF &VAR>='3170' AND &VAR<='3194' THEN &GROUP=2;
ELSE IF &VAR>='3196' AND &VAR<='3199' THEN &GROUP=2;
```

```
*** MOOD DISORDERS ***;
```

```
ELSE IF &VAR>='2960' AND &VAR<='2969' THEN &GROUP=3;
ELSE IF &VAR>='3110' AND &VAR<='3119' THEN &GROUP=3;
ELSE IF &VAR>='3140' AND &VAR<='3149' THEN &GROUP=3;
```

```
*** ORGANIC MENTAL DISORDERS ***;
```

```
ELSE IF &VAR>='2900' AND &VAR<='2909' THEN &GROUP=3;
ELSE IF &VAR>='2940' AND &VAR<='2949' THEN &GROUP=3;
ELSE IF &VAR>='3100' AND &VAR<='3109' THEN &GROUP=3;
```

```
*** SCHIZOPHRENIC ***;
```

```
ELSE IF &VAR>='2950' AND &VAR<='2959' THEN &GROUP=3;
ELSE IF &VAR>='2970' AND &VAR<='2989' THEN &GROUP=3;
```

```
*** OTHER MENTAL DISORDERS ***;
```

```
ELSE IF &VAR>='2910' AND &VAR<='2939' THEN &GROUP=3;
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
ELSE IF &VAR>='3000' AND &VAR<='3099' THEN &GROUP=3;
ELSE IF &VAR>='3160' AND &VAR<='3169' THEN &GROUP=3;
ELSE IF &VAR='3195' THEN &GROUP=3;
```

```
*** NERVOUS SYSTEM ***;
```

```
ELSE IF &VAR>='3200' AND &VAR<='3269' THEN &GROUP=4;
ELSE IF &VAR>='3290' AND &VAR<='3379' THEN &GROUP=4;
ELSE IF &VAR>='3400' AND &VAR<='3749' THEN &GROUP=4;
ELSE IF &VAR>='3760' AND &VAR<='3899' THEN &GROUP=4;
```

```
*** CIRCULATORY ***;
```

```
ELSE IF &VAR>='3750' AND &VAR<='3759' THEN &GROUP=4;
ELSE IF &VAR>='3900' AND &VAR<='3989' THEN &GROUP=4;
ELSE IF &VAR>='4010' AND &VAR<='4059' THEN &GROUP=4;
ELSE IF &VAR>='4100' AND &VAR<='4179' THEN &GROUP=4;
ELSE IF &VAR>='4200' AND &VAR<='4389' THEN &GROUP=4;
ELSE IF &VAR>='4400' AND &VAR<='4449' THEN &GROUP=4;
ELSE IF &VAR>='4460' AND &VAR<='4489' THEN &GROUP=4;
ELSE IF &VAR>='4510' AND &VAR<='4599' THEN &GROUP=4;
```

```
*** RESPIRATORY ***;
```

```
ELSE IF &VAR>='4600' AND &VAR<='4669' THEN &GROUP=4;
ELSE IF &VAR>='4700' AND &VAR<='4789' THEN &GROUP=4;
ELSE IF &VAR>='4800' AND &VAR<='4879' THEN &GROUP=4;
ELSE IF &VAR>='4900' AND &VAR<='4969' THEN &GROUP=4;
ELSE IF &VAR>='5000' AND &VAR<='5089' THEN &GROUP=4;
ELSE IF &VAR>='5100' AND &VAR<='5199' THEN &GROUP=4;
ELSE IF &VAR>='7680' AND &VAR<='7709' THEN &GROUP=4;
```

```
*** DIGESTIVE ***;
```

```
ELSE IF &VAR>='5200' AND &VAR<='5379' THEN &GROUP=4;
ELSE IF &VAR>='5400' AND &VAR<='5439' THEN &GROUP=4;
ELSE IF &VAR>='5500' AND &VAR<='5539' THEN &GROUP=4;
ELSE IF &VAR>='5550' AND &VAR<='5589' THEN &GROUP=4;
ELSE IF &VAR>='5600' AND &VAR<='5609' THEN &GROUP=4;
ELSE IF &VAR>='5620' AND &VAR<='5629' THEN &GROUP=4;
ELSE IF &VAR>='5640' AND &VAR<='5799' THEN &GROUP=4;
ELSE IF &VAR>='7770' AND &VAR<='7779' THEN &GROUP=4;
```

```
*** GENITOURINARY ***;
```

```
ELSE IF &VAR>='5800' AND &VAR<='6089' THEN &GROUP=4;
ELSE IF &VAR>='6100' AND &VAR<='6119' THEN &GROUP=4;
ELSE IF &VAR>='6140' AND &VAR<='6299' THEN &GROUP=4;
```

```
*** SKIN ***;
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
ELSE IF &VAR>='6800' AND &VAR<='6869' THEN &GROUP=7;
ELSE IF &VAR>='6900' AND &VAR<='6989' THEN &GROUP=7;
ELSE IF &VAR>='7000' AND &VAR<='7099' THEN &GROUP=7;
ELSE IF &VAR>='7780' AND &VAR<='7789' THEN &GROUP=7;

*** MUSCULOSKELETAL ***;
ELSE IF &VAR>='7100' AND &VAR<='7399' THEN &GROUP=5;

*** CONGENITAL ***;
ELSE IF &VAR>='7400' AND &VAR<='7599' THEN &GROUP=7;

*** INJURIES ***;
ELSE IF &VAR>='8000' AND &VAR<='8489' THEN &GROUP=6;
ELSE IF &VAR>='8500' AND &VAR<='8549' THEN &GROUP=6;
ELSE IF &VAR>='8600' AND &VAR<='8879' THEN &GROUP=6;
ELSE IF &VAR>='8900' AND &VAR<='8979' THEN &GROUP=6;
ELSE IF &VAR>='9000' AND &VAR<='9059' THEN &GROUP=6;
ELSE IF &VAR>='9070' AND &VAR<='9099' THEN &GROUP=6;
ELSE IF &VAR>='9200' AND &VAR<='9299' THEN &GROUP=6;
ELSE IF &VAR>='9400' AND &VAR<='9599' THEN &GROUP=6;

*** OTHER ***;
ELSE IF &VAR>='7600' AND &VAR<='7609' THEN &GROUP=7;
ELSE IF &VAR>='7640' AND &VAR<='7669' THEN &GROUP=7;
ELSE IF &VAR>='7800' AND &VAR<='7809' THEN &GROUP=7;
ELSE IF &VAR>='7830' AND &VAR<='7849' THEN &GROUP=7;
ELSE IF &VAR='9330' THEN &GROUP=7;

*** UNKNOWN/MISSING ***;
ELSE IF NOT MISSING(&VAR) THEN &GROUP=8;
ELSE IF MISSING(&VAR)
    AND MISSING(SSIELIG_FIRST) AND MISSING(DOEI)
    THEN &GROUP=.N;
ELSE IF MISSING(&VAR) THEN &GROUP=.M;

%MEND DGROUP;
%DGROUP(VAR=DXPRIBEST,GROUP=DXPRIBEST_PUF);
%DGROUP(VAR=DXSECBEST,GROUP=DXSECBEST_PUF);

/* COMBINE CATEGORIES FOR TOC1 AND TOC2 AND TOC_HISTS */
%MACRO TOC_V1(VAR=);
IF MISSING(DOEI) AND MISSING(&VAR.) THEN &VAR._PUF = .N;
ELSE IF &VAR. = '1' THEN &VAR._PUF=1;
ELSE IF &VAR. IN ('2','0','9','4') THEN &VAR._PUF=2;
```



## Appendix A.262

JCL/SAS Code: PUFDMG

```
ELSE IF &VAR. IN ('3','8') THEN &VAR._PUF=3;
ELSE IF &VAR. IN ('5','6') THEN &VAR._PUF=5;
ELSE IF &VAR. IN ('7','R') THEN &VAR._PUF=7;
ELSE IF MISSING(&VAR.) THEN &VAR._PUF = .M;
%MEND TOC_V1;
%TOC_V1(VAR=TOC1);
%TOC_V1(VAR=TOC_HIST1);

%MACRO TOC_V2(VAR=,MVAR=);
IF MISSING(DOEI) AND MISSING(&VAR.) THEN &VAR._PUF = .N;
ELSE IF MISSING(&MVAR.) AND MISSING(&VAR.) THEN &VAR._PUF = .N;
ELSE IF &VAR. = '1' THEN &VAR._PUF=1;
ELSE IF &VAR. IN ('2','3','4','0','8','9') THEN &VAR._PUF=2;
ELSE IF &VAR. IN ('5','6','7','R') THEN &VAR._PUF=3;
ELSE IF MISSING(&VAR.) THEN &VAR._PUF = .M;
%MEND TOC_V2;

%TOC_V2(VAR=TOC2,MVAR=TOC1);
%TOC_V2(VAR=TOC_HIST2,MVAR=TOC_HIST1);
%TOC_V2(VAR=TOC_HIST3,MVAR=TOC_HIST2);

/* TOPCODE 3 OR MORE FOR TOC_NUM TO 3 */
IF MISSING(DOEI) AND MISSING(TOC_NUM) THEN TOC_NUM_PUF = .N;
ELSE IF TOC_NUM >= 3 THEN TOC_NUM_PUF=3;
ELSE IF MISSING(TOC_NUM) THEN TOC_NUM_PUF = .M;
ELSE IF TOC_NUM = 0 THEN TOC_NUM_PUF = .M;
ELSE TOC_NUM_PUF = TOC_NUM;

/* COMBINE CATEGORIES FOR BIC AND BIC_HIST */
%MACRO BIC(VAR=);
IF MISSING(DOEI) AND MISSING(&VAR.) THEN &VAR._PUF = '.N';
ELSE IF &VAR. = 'A' THEN &VAR._PUF='A';
ELSE IF SUBSTR(&VAR.,1,1)='C' THEN &VAR._PUF='C';
ELSE IF NOT MISSING(&VAR.) THEN &VAR._PUF='O';
ELSE IF MISSING(&VAR.) THEN &VAR._PUF = '.M';
%MEND BIC;
%BIC(VAR=BIC);
%BIC(VAR=BIC_HIST1);
%BIC(VAR=BIC_HIST2);
%BIC(VAR=BIC_HIST3);

/* COMBINE CATEGORIES FOR MFT */
IF MISSING(SSIELIG_FIRST) AND MISSING(MFT) THEN MFT_PUF = .N;
ELSE IF MFT IN ('AI','BI','DI') THEN MFT_PUF=1;
ELSE IF MFT IN ('AS','BS','DS') THEN MFT_PUF=2;
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
ELSE IF MFT IN ('DC','BC') THEN MFT_PUF=3;
ELSE IF MFT IN ('XF','XM','XP','XS','S') THEN MFT_PUF=.M;
ELSE IF MISSING(MFT) THEN MFT_PUF = .M;

/* TKT 2002-2005 INDICATOR */
IF TKT02=1 OR TKT03=1 OR TKT04=1 OR TKT05=1 THEN TKT0205_PUF = 1;
ELSE IF TKT02=0 AND TKT03=0 AND TKT04=0 AND TKT05=0 THEN
  TKT0205_PUF = 0;
ELSE IF MISSING(TKT02) AND MISSING(TKT03) AND MISSING(TKT04)
  AND MISSING(TKT05) THEN TKT0205_PUF = .M;

IF MISSING(SSIELIG_FIRST) AND MISSING(DOEI)
  AND MISSING(TKT02) AND MISSING(TKT03) AND MISSING(TKT04)
  AND MISSING(TKT05) THEN TKT0205_PUF = .N;

%MACRO TKTN(VAR=);

  %DO Y= 6 %TO &YR.;
  %LET I = %SYSFUNC(PUTN(&Y.,Z2));

  TKT&I._PUF = TKT&I;
  IF MISSING(TKT&I) THEN TKT&I._PUF = .M;
  IF MISSING(DOEI) AND MISSING(SSIELIG_FIRST) AND MISSING(TKT&I)
    THEN TKT&I._PUF=.N;

  LABEL TKT&I._PUF = "TICKET ACTIVE STATUS IN YEAR (YY = YEAR)";

  %END;
%MEND TKTN;
%TKTN;

/*RECODE SSI AND SSDI AS PUF*/
%MACRO SSI_SSDI;
%DO I = 94 %TO 99;
  IF NOT MISSING(DOEI) AND MISSING(SSDI&I.) THEN SSDI&I._PUF = .M;
  ELSE IF MISSING (SSDI&I) THEN SSDI&I._PUF = .N;
  ELSE SSDI&I._PUF = SSDI&I.;

  IF NOT MISSING(SSIELIG_FIRST) AND MISSING(SSI&I.)
    THEN SSI&I._PUF = .M;
  ELSE IF MISSING (SSI&I) THEN SSI&I._PUF = .N;
  ELSE SSI&I._PUF = SSI&I.;
%END;
%DO Y = 0 %TO &YR.;
  %LET I = %SYSFUNC(PUTN(&Y.,Z2));
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
IF NOT MISSING(DOEI) AND MISSING(SSDI&I.) THEN SSDI&I._PUF = .M;  
ELSE IF MISSING (SSDI&I) THEN SSDI&I._PUF = .N;  
ELSE SSDI&I._PUF = SSDI&I.;
```

```
IF NOT MISSING(SSIELIG_FIRST) AND MISSING(SSI&I.)  
THEN SSI&I._PUF = .M;  
ELSE IF MISSING (SSI&I) THEN SSI&I._PUF = .N;  
ELSE SSI&I._PUF = SSI&I.;
```

```
%END;
```

```
%MEND SSI_SSDI;
```

```
%SSI_SSDI;
```

```
/* RECODE FIRSTDR AND DR(N) INTO CATEGORIES */
```

```
%MACRO DR(VAR=);
```

```
LENGTH &VAR._PUF $4;
```

```
IF MISSING(DOEI) AND MISSING(SSIELIG_FIRST) AND MISSING(&VAR)
```

```
THEN &VAR._PUF = '.N';
```

```
ELSE IF &VAR. IN ('7','5') THEN &VAR._PUF = 'MINE';
```

```
ELSE IF &VAR. IN ('3') THEN &VAR._PUF = 'MIP';
```

```
ELSE IF ANYALPHA(&VAR.) > 0 THEN &VAR._PUF = 'MIE';
```

```
ELSE &VAR._PUF = '.M';
```

```
%MEND DR;
```

```
%DR(VAR=DR1);
```

```
%DR(VAR=DR2);
```

```
%DR(VAR=DR3);
```

```
%DR(VAR=DR4);
```

```
%DR(VAR=DR5);
```

```
%DR(VAR=FIRSTDR);
```

```
/* RECODE FIRSTDT AND DT(N) INTO CATEGORIES */
```

```
%MACRO DTN(VAR=);
```

```
LENGTH &VAR._PUF $4;
```

```
IF MISSING(DOEI) AND MISSING(SSIELIG_FIRST) AND MISSING(&VAR)
```

```
THEN &VAR._PUF = '.N';
```

```
ELSE IF &VAR. = '3' THEN &VAR._PUF = 'MIEP';
```

```
ELSE IF &VAR. = '4' THEN &VAR._PUF = 'MINE';
```

```
ELSE IF &VAR NOT IN ('3','4') THEN &VAR._PUF = '.M';
```

```
%MEND DTN;
```

```
%DTN(VAR=DT1);
```

```
%DTN(VAR=DT2);
```

```
%DTN(VAR=DT3);
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
%DTN(VAR=DT4);
%DTN(VAR=DT5);
%DTN(VAR=FIRSTDT);

%MACRO LBL;
%DO I = 1 %TO 5;

  LABEL DR&I._PUF = "MEDICAL CONTINUING DISABILITY REVIEW REASON &I."
  DT&I._PUF = "MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE &I.";
%END;
%MEND LBL;
%LBL;

/* FIND MAX AND MIN PIA WITH CORRESPONDING PIED VALUES */
ARRAY PIA(*) PIA1 - PIA50;
ARRAY PIED(*) PIED1 - PIED50;

IF NOT MISSING(PIA1) THEN DO;

  PIA_MAX_PUF = MAX(OF PIA1-PIA50);
  PIA_MIN_PUF = MIN(OF PIA1-PIA50);

  * CREATE RANDCENTS TO HELP BREAK TIES DURING TOPCODING;
  RANDCENTS = RAND("UNIFORM");
  IF PIA_MAX_PUF > 0
  THEN PIA_MAX_CENTS = SUM(PIA_MAX_PUF,RANDCENTS);
  IF PIA_MIN_PUF > 0
  THEN PIA_MIN_CENTS = SUM(PIA_MIN_PUF,RANDCENTS);

  DO I= 1 TO DIM(PIA) UNTIL(PIED_MAX NE .);
    IF PIA(I) = PIA_MAX_PUF THEN PIED_MAX = PIED(I);
  END;

  DO I = DIM(PIA)TO 1 BY -1 UNTIL(PIED_MIN NE .);
    IF PIA(I) = PIA_MIN_PUF AND NOT MISSING (PIED(I))
    THEN PIED_MIN = PIED(I);
  END;
END;

/* MASK PIED MIN AND MAX VARIABLES */
/* MASK TO 55 YEARS BEFORE DAF YEAR*/
%MACRO MASKP(DT=,PIA=);

%LET CUTYR = %EVAL(&DAFYR. - 55);
%LET CUT = "15JAN&CUTYR."D;
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
IF MISSING(&PIA) AND MISSING(&DT) THEN &DT._PUF = .N;
ELSE IF NOT MISSING(&DT.) THEN
  &DT._PUF = INTNX('MONTH',&DT.,0,'BEG') + 14;
ELSE IF MISSING(&DT.) THEN &DT._PUF = .M;

IF &DT._PUF < &CUT. AND NOT MISSING(&DT._PUF) THEN &DT._PUF = &CUT.;
FORMAT &DT._PUF MMDDYY10.;

%MEND MASKP;
%MASKP(DT=PIED_MIN,PIA=PIA_MIN_PUF);
%MASKP(DT=PIED_MAX,PIA=PIA_MAX_PUF);

/* CREATE OTHER VARIABLES */
IF NOT MISSING(FRA) THEN FRA_PUF = FRA;
ELSE IF MISSING(FRA) THEN FRA_PUF = .M;

IF MISSING(DOEI) AND REPPYSSD = 0 THEN REPPYSSD_PUF = .N;
ELSE IF MISSING(DOEI) AND MISSING(REPPYSSD) THEN REPPYSSD_PUF = .N;
ELSE IF MISSING(REPPYSSD) THEN REPPYSSD_PUF = .M;
ELSE REPPYSSD_PUF = REPPYSSD;

IF MISSING(PAYEE_TYP_SSDI) AND MISSING(DOEI)
  THEN PAYEE_TYP_SSDI_PUF = .N;
ELSE IF MISSING(PAYEE_TYP_SSDI) AND NOT MISSING(DOEI)
  THEN PAYEE_TYP_SSDI_PUF = .M;
ELSE IF PAYEE_TYP_SSDI IN ('R','S')
  THEN PAYEE_TYP_SSDI_PUF = 1;
ELSE IF PAYEE_TYP_SSDI IN ('B','C','D','E','F','G','H','I')
  THEN PAYEE_TYP_SSDI_PUF = 2;
ELSE IF PAYEE_TYP_SSDI IN ('Q','O','M','K','P','N','L','J')
  THEN PAYEE_TYP_SSDI_PUF = 3;
ELSE IF PAYEE_TYP_SSDI IN ('U','T','A','V','W','Y')
  THEN PAYEE_TYP_SSDI_PUF = 4;

IF MISSING(SSIELIG_FIRST) AND REPPYSSI = 0 THEN REPPYSSI_PUF = .N;
ELSE IF MISSING(SSIELIG_FIRST) AND MISSING(REPPYSSI)
  THEN REPPYSSI_PUF = .N;
ELSE IF MISSING(REPPYSSI) THEN REPPYSSI_PUF = .M;
ELSE REPPYSSI_PUF = REPPYSSI;

IF MISSING(SSIELIG_FIRST) AND MISSING(PAYEE_TYP_SSI)
  THEN PAYEE_TYP_SSI_PUF = .N;
ELSE IF MISSING(PAYEE_TYP_SSI) THEN PAYEE_TYP_SSI_PUF = .M;
ELSE IF PAYEE_TYP_SSI IN ('AGY','FIN') THEN PAYEE_TYP_SSI_PUF=1;
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```

ELSE IF PAYEE_TYP_SSI
  IN ('FTH','MTH','SFT','SMT','GPR','CHD','REL','SPO')
  THEN PAYEE_TYP_SSI_PUF=2;
ELSE IF PAYEE_TYP_SSI
  IN ('NPM','PRM','SLM','FDM','NPO','PRO','SLO','FDO')
  THEN PAYEE_TYP_SSI_PUF=3;
ELSE IF PAYEE_TYP_SSI IN('OTH','OFF','PYE','SEL','RPD','INP','ESP','')
  THEN PAYEE_TYP_SSI_PUF=4;

```

```

* LAF_MR;
LENGTH LAF_MR_PUF $2;
IF MISSING(DOEI) THEN LAF_MR_PUF = '.N';
ELSE IF MISSING(LAF_MR) THEN LAF_MR_PUF = '.M';
ELSE IF SUBSTR(LAF_MR,1,1) = 'C' THEN LAF_MR_PUF = 'C';
ELSE IF SUBSTR(LAF_MR,1,1) = 'S' THEN LAF_MR_PUF = 'S';
ELSE IF SUBSTR(LAF_MR,1,1) = 'T' THEN LAF_MR_PUF = 'T';
ELSE IF NOT MISSING(LAF_MR) THEN LAF_MR_PUF = 'O';

```

```
/* ADD ALL VARIABLE LABELS */
```

```

LABEL BDOE_START1_PUF = "SSDI DATE OF ENTITLEMENT START 1"
  BDOE_START2_PUF = "SSDI DATE OF ENTITLEMENT START 2"
  BDOE_TERM1_PUF = "SSDI DATE OF ENTITLEMENT TERMINIATION 1"
  BDOE_TERM2_PUF = "SSDI DATE OF ENTITLEMENT TERMINIATION 2"
  BDOF1_PUF = "SSDI DATE OF FILING ENTRY 1"
  BDOF2_PUF = "SSDI DATE OF FILING ENTRY 2"
  BIC_PUF = "SSDI BENEFICIARY IDENTIFICATION CODE (BIC)"
  BIC_HIST1_PUF =
    "SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 1"
  BIC_HIST2_PUF =
    "SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 2"
  BIC_HIST3_PUF =
    "SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) 3"
  BIC_HIST_START1_PUF =
    "SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 1"
  BIC_HIST_START2_PUF =
    "SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 2"
  BIC_HIST_START3_PUF =
    "SSDI HISTORICAL BENEFICIARY IDENTIFICATION CODE (BIC) START DATE 3"
  CLM_FIL_PUF = "SSI CLAIM FILED DATE"
  DOBBEST_PUF = "BEST DATE OF BIRTH"
  DODBEST_PUF = "BEST DATE OF DEATH"
  DOEC_PUF = "SSDI MOST RECENT ENTITLEMENT DATE"
  DOEI_PUF = "SSDI INITIAL ENTITLEMENT DATE"
  DXPRIBEST_PUF =

```

Appendix A.262

JCL/SAS Code: PUFDMG

"PRIMARY DIAGNOSIS GROUP, MOST RECENT, BEST FROM ALL SOURCES"  
DXSECBEST\_PUF =  
"SECONDARY DIAGNOSIS GROUP, MOST RECENT, BEST FROM ALL SOURCES"  
ELG\_RD\_PUF = "SSI DATE OF MOST RECENT ELIGIBILITY"  
FIRSTDD\_PUF =  
"MEDICAL CONTINUING DISABILITY REVIEW DATE, EARLIEST"  
FIRSTDR\_PUF =  
"MEDICAL CONTINUING DISABILITY REVIEW REASON, EARLIEST"  
FIRSTDT\_PUF =  
"MEDICAL CONTINUING DISABILITY REVIEW REASON TYPE, EARLIEST"  
FRA\_PUF = "BENEFICIARY'S FULL RETIREMENT AGE"  
FRADATE\_PUF = "DATE WHEN FULL RETIREMENT AGE IS REACHED"  
LAF\_MR\_PUF = "LEDGER ACCOUNT FILE STATUS, MOST RECENT"  
LAF\_MR\_DATE\_PUF = "DATE OF MOST RECENT LEDGER ACCOUNT FILE STATUS"  
MFT\_PUF = "SSI MASTER FILE TYPE"  
REPPYSSD\_PUF = "SSDI REPRESENTATIVE PAYEE INDICATOR"  
PAYEE\_TYP\_SSDI\_PUF = "SSDI TYPE OF PAYEE"  
REPPYSSI\_PUF = "SSI REPRESENTATIVE PAYEE INDICATOR"  
PAYEE\_TYP\_SSI\_PUF = "SSI TYPE OF PAYEE CODE"  
SEX\_PUF = "SEX"  
SSDI\_ONSET\_PUF =  
"SSDI DISABILITY ONSET DATE EARLIEST PERIOD OF ELIGIBILITY"  
SSIELIG\_FIRST\_PUF = "SSI EARLIEST ELIGIBILITY DATE"  
SSDI94\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI95\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI96\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI97\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI98\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI99\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI00\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI01\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI02\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI03\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI04\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI05\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI06\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI07\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI08\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI09\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI10\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI11\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI12\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI13\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI14\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI15\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"

Appendix A.262

JCL/SAS Code: PUFDMG

SSDI16\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI17\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI18\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI19\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI20\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSDI21\_PUF = "SSDI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSI94\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSI95\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSI96\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSI97\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSI98\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSI99\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO0\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO1\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO2\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO3\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO4\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO5\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO6\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO7\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO8\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO9\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO10\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO11\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO12\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO13\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO14\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO15\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO16\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO17\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO18\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO19\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO20\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
SSIO21\_PUF = "SSI CURRENT PAY STATUS IN YEAR (YY = YEAR)"  
TKT0205\_PUF = "TICKET ACTIVE STATUS IN YEARS (2002-2005)"  
TOC\_NUM\_PUF = "SSDI NUMBER OF TYPE OF CLAIM (TOC) OCCURRENCES"  
TOC1\_PUF = "SSDI TYPE OF CLAIM 1"  
TOC2\_PUF = "SSDI TYPE OF CLAIM 2"  
TOC\_START1\_PUF = "SSDI TYPE OF CLAIM (TOC) START DATE 1"  
TOC\_START2\_PUF = "SSDI TYPE OF CLAIM (TOC) START DATE 2"  
TOC\_HIST1\_PUF = "SSDI HISTORICAL TYPE OF CLAIM (TOC) 1"  
TOC\_HIST2\_PUF = "SSDI HISTORICAL TYPE OF CLAIM (TOC) 2"  
TOC\_HIST3\_PUF = "SSDI HISTORICAL TYPE OF CLAIM (TOC) 3"  
TOC\_HIST\_START1\_PUF =  
"SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 1"



## Appendix A.262

JCL/SAS Code: PUFDMG

```
TOC_HIST_START2_PUF =  
    "SSDI HISTORICAL TYPE OF CLAIM (TOC) START DATE 2"  
TOC_HIST_START3_PUF =  
    "SSDI HISTORICAL TYPE OF CLAIM (TOC) STARTDATE 3"  
TWPCMPLMNTNTH1_PUF =  
    "FIRST SSDI TRIAL WORK PERIOD (TWP) COMPLETION MONTH"  
PIA_MAX_PUF = "SSDI PRIMARY INS. AMOUNT, MAXIMUM"  
PIA_MIN_PUF = "SSDI PRIMARY INS. AMOUNT, MINIMUM"  
PIED_MAX_PUF = "SSDI MAX PIA EFFECTIVE DATE"  
PIED_MIN_PUF = "SSDI MIN PIA EFFECTIVE DATE"  
;
```

```
RUN;
```

```
* TOP CODE TOP 0.5 PERCENT OF PIA_MIN AND PIA_MAX;  
%MACRO TC(VAR=);
```

```
%GLOBAL AVG_&VAR. PCT_&VAR. NUM_&VAR.;
```

```
PROC SQL;  
CREATE TABLE TEMP4.&VAR. AS  
SELECT SUM(CASE WHEN NOT MISSING(&VAR._PUF)  
    AND &VAR._PUF NE 0 THEN 1 ELSE 0 END)  
    AS &VAR._NUM  
FROM TEMP1.P10DMGPIUF;  
QUIT;
```

```
DATA TEMP4.&VAR.;;  
SET TEMP4.&VAR.;;  
CALL SYMPUTX("NUM_&VAR.", PUT(&VAR._NUM, BEST32.));  
CALL SYMPUTX("PCT_&VAR.",.);  
CALL SYMPUTX("AVG_&VAR.",.);  
RUN;
```

```
%PUT NUMBER OF NONZERO CASES FOR &VAR. IS &&NUM_&VAR.;
```

```
/*IF AT LEAST 6000 NONZERO CASES THEN FIND 99.5TH PERCENTILE*/  
%IF &&NUM_&VAR. GE 6000 %THEN %DO;
```

```
PROC UNIVARIATE DATA=TEMP1.P10DMGPIUF;  
WHERE NOT MISSING(&VAR._PUF) AND &VAR._PUF NE 0;  
VAR &VAR._PUF;  
OUTPUT OUT=TEMP2.&VAR. PCTLPTS=99.5 PCTLPRE=&VAR.;;  
RUN;
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
DATA TEMP2.&VAR.;
  SET TEMP2.&VAR.;
  CALL SYMPUTX("PCT_&VAR.",PUT(&VAR.99_5,BEST32.));
RUN;

%PUT TOPCODING NUMBER FOR &VAR. IS &&PCT_&VAR.;

PROC MEANS DATA=TEMP1.P10DMGPUF MEAN;
  VAR &VAR._PUF;
  WHERE &VAR._PUF >= &&PCT_&VAR.;
  OUTPUT OUT=TEMP3.&VAR. MEAN=;
RUN;

DATA TEMP3.&VAR.;
  SET TEMP3.&VAR.;
  CALL SYMPUTX("AVG_&VAR.",PUT(&VAR._PUF,BEST32.));
RUN;

%PUT TOPCODING AVERAGE FOR &VAR. IS &&AVG_&VAR.;
%END;

/*IF 30 - 5999 NONZERO CASES THEN TOP CODE BASED ON TOP 30 CASES*/
%ELSE %IF 30 LE &&NUM_&VAR. AND &&NUM_&VAR. < 6000
%THEN %DO;

PROC SORT DATA = TEMP1.P10DMGPUF
  (KEEP = &VAR._PUF &VAR._CENTS)
  OUT = TEMP2.&VAR.;
WHERE &VAR._PUF > 0;
BY DESCENDING &VAR._CENTS;
RUN;

DATA TEMP2.&VAR.;
  SET TEMP2.&VAR.;
  BY DESCENDING &VAR._CENTS;
  IF _N_ <= 30;
  IF _N_ = 30
  THEN CALL SYMPUTX("PCT_&VAR.",PUT(&VAR._CENTS, BEST32.));
RUN;

%PUT TOPCODING NUM FOR &VAR. IS &&PCT_&VAR.;

PROC MEANS DATA=TEMP2.&VAR. MEAN NOPRINT;
  VAR &VAR._PUF;
  WHERE &VAR._PUF NE 0;
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
OUTPUT OUT=TEMP3.&VAR. MEAN=;
RUN;

DATA TEMP3.&VAR.;
SET TEMP3.&VAR.;
CALL SYMPUTX("AVG_&VAR.",PUT(&VAR._PUF,BEST32.));
RUN;

%PUT TOPCODING AVERAGE FOR &VAR. IS &&AVG_&VAR.;
%END;

/*IF LESS THAN 30 NONZERO CASES THEN SET ALL TO MISSING*/
%ELSE %IF &&NUM_&VAR. < 30 %THEN %DO;
  %PUT TOPCODING NUM FOR &VAR. IS &&PCT_&VAR.;
  %PUT TOPCODING AVERAGE FOR &VAR. IS &&AVG_&VAR.;
%END;

%MEND TC;
%TC(VAR=PIA_MAX);
%TC(VAR=PIA_MIN);

PROC CONTENTS DATA=TEMP1.P10DMGPUF
  OUT=TEMP3.CONTENTS(KEEP=NAME) NOPRINT ;
RUN;

PROC SQL NOPRINT;
SELECT NAME INTO :KEEPVARS SEPARATED BY ' '
FROM TEMP3.CONTENTS
WHERE UPCASE(NAME) LIKE '%^_PUF' ESCAPE '^' ;
QUIT;

DATA OUT1.P10DMGPUF (KEEP= PUFPIN &KEEPVARS);
SET TEMP1.P10DMGPUF;

%MACRO SETTC(VAR=);

  &VAR._T = &&AVG_&VAR.;

  /*IF LESS THAN 30 NONZERO CASES THEN SET ALL TO MISSING;
  *IF VARIABLE CODED AS .P (NOT AVAILABLE) THEN LEAVE AS .P;*/
  %IF &&NUM_&VAR. < 30 %THEN %DO;
    IF &VAR._PUF NE .P THEN &VAR._PUF = .X;
  %END;

  /*IF 30 TO 5999 NONZERO CASES THEN TOP CODE USING CENTSYYMM;*/
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
%ELSE %IF 30 <= &&NUM_&VAR. AND &&NUM_&VAR. < 6000 %THEN %DO;
  IF NOT MISSING(&VAR._PUF) AND &VAR._PUF NE 0 THEN DO;
    IF PUT(&VAR._CENTS,BEST32.) >= &&PCT_&VAR.
      THEN &VAR._PUF = &VAR._T;
  END;
%END;

/* IF 6000 OR MORE NONZERO CASES THEN TOP CODE NORMALLY;*/
%ELSE %IF &&NUM_&VAR. GE 6000 %THEN %DO;
  IF NOT MISSING(&VAR._PUF) AND &VAR._PUF NE 0 THEN DO;
    IF PUT(&VAR._PUF,BEST32.) >= &&PCT_&VAR.
      THEN &VAR._PUF = &VAR._T;
  END;
%END;
DROP &VAR._T;

%MEND SETTC;
%SETTC(VAR=PIA_MAX);
%SETTC(VAR=PIA_MIN);

/* NOW ROUND DOLLAR AMOUNTS FOR PIA MAX AND MIN */
%MACRO DOLLAMT(VAR=);

  IF MISSING(DOEI_PUF) AND MISSING(&VAR.) THEN &VAR. = .N;
  ELSE IF &VAR. > 0 AND &VAR. < 8 THEN &VAR. = 4;
  ELSE IF &VAR. >= 8 AND &VAR. < 1000 THEN &VAR. = ROUND(&VAR.,10);
  ELSE IF &VAR. >= 1000 AND &VAR. < 50000 THEN &VAR. = ROUND(&VAR.,100);
  ELSE IF &VAR. >= 50000 THEN &VAR. = ROUND(&VAR.,1000);
  ELSE IF &VAR. = 0 THEN &VAR. = 0;
  ELSE IF MISSING(&VAR.) THEN &VAR. = .M;

%MEND DOLLAMT;
%DOLLAMT(VAR=PIA_MAX_PUF);
%DOLLAMT(VAR=PIA_MIN_PUF);

RUN;
/*
TITLE2 'CONTENTS OF PROCESSED DMG';
PROC CONTENTS DATA=OUT1.P10DMGPUF; RUN;
*/
* FREQS OF DOEI AND SSIELIG_FIRST AS MISSING OR NOT;
TITLE2 'FREQS OF DOEI AND SSIELIG_FIRST';
PROC FORMAT;
  VALUE PRESENT .= 'MISSING' OTHER= 'NONMISSING';
RUN;
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
PROC FREQ DATA=TEMP1.P10DMGPUF;
  TABLES DOEI SSIELIG_FIRST / LIST MISSING;
  FORMAT DOEI SSIELIG_FIRST PRESENT.;
RUN;
```

```
PROC FREQ DATA=TEMP1.P10DMGPUF;
  TABLES DOEI * SSIELIG_FIRST / LIST MISSING;
  FORMAT DOEI SSIELIG_FIRST PRESENT.;
RUN;
```

```
/*NEW IN DAF17 - RAW FREQS OF NEW VARIABLES ADDED TO PUF*/
```

```
TITLE2 'FREQS OF RAW VARIABLES TO BE ADDED TO PUF';
PROC FREQ DATA=TEMP1.P10DMGPUF;
  TABLES BIC_HIST1 - BIC_HIST3 / LIST MISSING;
RUN;
PROC FREQ DATA=TEMP1.P10DMGPUF;
  TABLES BIC_HIST_START1 - BIC_HIST_START3 / LIST MISSING;
  FORMAT BIC_HIST_START: MMDDYY10.;
RUN;
PROC FREQ DATA=TEMP1.P10DMGPUF;
  TABLES ELG_RD / LIST MISSING;
RUN;
PROC FREQ DATA=TEMP1.P10DMGPUF;
  TABLES LAF_MR / LIST MISSING;
RUN;
PROC FREQ DATA=TEMP1.P10DMGPUF;
  TABLES LAF_MR_DATE / LIST MISSING;
  FORMAT LAF_MR_DATE MMDDYY10.;
RUN;
```

```
/* NOT AVAILABLE IN DAF17. WILL BE ADDED IN DAF18
```

```
PROC FREQ DATA=TEMP1.P10DMGPUF;
  TABLES PSTA_MR / LIST MISSING;
RUN;
PROC FREQ DATA=TEMP1.P10DMGPUF;
  TABLES PSTA_MR_DATE / LIST MISSING;
  FORMAT PSTA_MR_DATE MMDDYY10.;
RUN;
*/
PROC FREQ DATA=TEMP1.P10DMGPUF;
  TABLES SSI94 - SSI99 SSI00-SSI&YR./LIST MISSING;
  TABLES SSDI94 - SSDI99 SSDI00-SSDI&YR./LIST MISSING;
RUN;
```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
PROC FREQ DATA=TEMP1.P10DMGPIUF;
  TABLES DOEI*(SSDI94 - SSDI99 SSDI00-SSDI&YR.)/LIST MISSING;
RUN;
PROC FREQ DATA=TEMP1.P10DMGPIUF;
  TABLES SSIELIG_FIRST*(SSI94 - SSI99 SSI00-SSI&YR.)/LIST MISSING;
RUN;
PROC FREQ DATA=TEMP1.P10DMGPIUF;
  TABLES TOC_HIST1 - TOC_HIST3 / LIST MISSING;
RUN;
PROC FREQ DATA=TEMP1.P10DMGPIUF;
  TABLES TOC_HIST_START1 - TOC_HIST_START3 / LIST MISSING;
  FORMAT TOC_HIST_START: MMDDYY10.;
RUN;

/*NEW IN DAF17: RENAME VARIABLES FROM N_PUF TO _PUFN*/
DATA OUT1.P10DMGPIUF
  (DROP = BIC_HIST_PUF: BIC_HIST_START_PUF: ELG_RD_PUF
    LAF_MR_PUF LAF_MR_DATE_PUF SSDI_PUF: SSI_PUF:
    TOC_HIST_PUF: TOC_HIST_START_PUF:
    /*REGION_MR_PUF PSTA_MR_PUF PSTA_MR_DATE_PUF */);
  SET OUT1.P10DMGPIUF;

RENAME BDOE_START1_PUF = BDOE_START_PUF1
  BDOE_START2_PUF = BDOE_START_PUF2
  BDOE_TERM1_PUF = BDOE_TERM_PUF1
  BDOE_TERM2_PUF = BDOE_TERM_PUF2
  BDOF1_PUF = BDOF_PUF1
  BDOF2_PUF = BDOF_PUF2
  BIC_HIST1_PUF = BIC_HIST_PUF1
  BIC_HIST2_PUF = BIC_HIST_PUF2
  BIC_HIST3_PUF = BIC_HIST_PUF3
  BIC_HIST_START1_PUF = BIC_HIST_START_PUF1
  BIC_HIST_START2_PUF = BIC_HIST_START_PUF2
  BIC_HIST_START3_PUF = BIC_HIST_START_PUF3
  DD1_PUF = DD_PUF1
  DD2_PUF = DD_PUF2
  DD3_PUF = DD_PUF3
  DD4_PUF = DD_PUF4
  DD5_PUF = DD_PUF5
  DODEC1_PUF = DODEC_PUF1
  DODEC2_PUF = DODEC_PUF2
  DODEC3_PUF = DODEC_PUF3
  DODEC4_PUF = DODEC_PUF4
  DODEC5_PUF = DODEC_PUF5
  DR1_PUF = DR_PUF1
```

Appendix A.262

JCL/SAS Code: PUFDMG

```
DR2_PUF = DR_PUF2
DR3_PUF = DR_PUF3
DR4_PUF = DR_PUF4
DR5_PUF = DR_PUF5
DT1_PUF = DT_PUF1
DT2_PUF = DT_PUF2
DT3_PUF = DT_PUF3
DT4_PUF = DT_PUF4
DT5_PUF = DT_PUF5
SSDI94_PUF = SSDI_PUF94
SSDI95_PUF = SSDI_PUF95
SSDI96_PUF = SSDI_PUF96
SSDI97_PUF = SSDI_PUF97
SSDI98_PUF = SSDI_PUF98
SSDI99_PUF = SSDI_PUF99
SSDI00_PUF = SSDI_PUF00
SSDI01_PUF = SSDI_PUF01
SSDI02_PUF = SSDI_PUF02
SSDI03_PUF = SSDI_PUF03
SSDI04_PUF = SSDI_PUF04
SSDI05_PUF = SSDI_PUF05
SSDI06_PUF = SSDI_PUF06
SSDI07_PUF = SSDI_PUF07
SSDI08_PUF = SSDI_PUF08
SSDI09_PUF = SSDI_PUF09
SSDI10_PUF = SSDI_PUF10
SSDI11_PUF = SSDI_PUF11
SSDI12_PUF = SSDI_PUF12
SSDI13_PUF = SSDI_PUF13
SSDI14_PUF = SSDI_PUF14
SSDI15_PUF = SSDI_PUF15
SSDI16_PUF = SSDI_PUF16
SSDI17_PUF = SSDI_PUF17
SSDI18_PUF = SSDI_PUF18
SSDI19_PUF = SSDI_PUF19
SSDI20_PUF = SSDI_PUF20
SSDI21_PUF = SSDI_PUF21
SSI94_PUF = SSI_PUF94
SSI95_PUF = SSI_PUF95
SSI96_PUF = SSI_PUF96
SSI97_PUF = SSI_PUF97
SSI98_PUF = SSI_PUF98
SSI99_PUF = SSI_PUF99
SSI00_PUF = SSI_PUF00
SSI01_PUF = SSI_PUF01
```

Appendix A.262

JCL/SAS Code: PUFDMG

```
SSI02_PUF = SSI_PUF02
SSI03_PUF = SSI_PUF03
SSI04_PUF = SSI_PUF04
SSI05_PUF = SSI_PUF05
SSI06_PUF = SSI_PUF06
SSI07_PUF = SSI_PUF07
SSI08_PUF = SSI_PUF08
SSI09_PUF = SSI_PUF09
SSI10_PUF = SSI_PUF10
SSI11_PUF = SSI_PUF11
SSI12_PUF = SSI_PUF12
SSI13_PUF = SSI_PUF13
SSI14_PUF = SSI_PUF14
SSI15_PUF = SSI_PUF15
SSI16_PUF = SSI_PUF16
SSI17_PUF = SSI_PUF17
SSI18_PUF = SSI_PUF18
SSI19_PUF = SSI_PUF19
SSI20_PUF = SSI_PUF20
SSI21_PUF = SSI_PUF21
TOC_HIST1_PUF = TOC_HIST_PUF1
TOC_HIST2_PUF = TOC_HIST_PUF2
TOC_HIST3_PUF = TOC_HIST_PUF3
TOC_HIST_START1_PUF = TOC_HIST_START_PUF1
TOC_HIST_START2_PUF = TOC_HIST_START_PUF2
TOC_HIST_START3_PUF = TOC_HIST_START_PUF3
TKT0205_PUF = TKT_PUF0205
TKT06_PUF = TKT_PUF06
TKT07_PUF = TKT_PUF07
TKT08_PUF = TKT_PUF08
TKT09_PUF = TKT_PUF09
TKT10_PUF = TKT_PUF10
TKT11_PUF = TKT_PUF11
TKT12_PUF = TKT_PUF12
TKT13_PUF = TKT_PUF13
TKT14_PUF = TKT_PUF14
TKT15_PUF = TKT_PUF15
TKT16_PUF = TKT_PUF16
TKT17_PUF = TKT_PUF17
TKT18_PUF = TKT_PUF18
TKT19_PUF = TKT_PUF19
TKT20_PUF = TKT_PUF20
TKT21_PUF = TKT_PUF21
TOC1_PUF = TOC_PUF1
TOC2_PUF = TOC_PUF2
```



Appendix A.262

JCL/SAS Code: PUFDMG

```

    TOC_START1_PUF = TOC_START_PUF1
    TOC_START2_PUF = TOC_START_PUF2
    TOC_HIST1_PUF = TOC_HIST_PUF1
    TOC_HIST2_PUF = TOC_HIST_PUF2
    TOC_HIST3_PUF = TOC_HIST_PUF3
    TOC_HIST_START1_PUF = TOC_HIST_START_PUF1
    TOC_HIST_START2_PUF = TOC_HIST_START_PUF2
    TOC_HIST_START3_PUF = TOC_HIST_START_PUF3
    TWPCMPLMNTNTH1_PUF = TWPCMPLMNTNTH_PUF1
;
RUN;

TITLE2 'CONTENTS OF FINAL PUF DMG FILE - AFTER RENAMING';
PROC CONTENTS DATA=OUT1.P10DMGPUF; RUN;

PROC FORMAT;
  VALUE $MASKED '='='MISSING' OTHER='NONMISSING';
RUN;

%MACRO RUNFINALFREQ(VAR=);

PROC FREQ DATA=OUT1.P10DMGPUF;
  TABLES &VAR. / MISSING;
RUN;

%MEND RUNFINALFREQ;

TITLE2 'FREQ OF FINAL PUF DMG FILE - AFTER RENAMING';
%RUNFINALFREQ(VAR=BDOE_START_PUF1);
%RUNFINALFREQ(VAR=BDOE_START_PUF2);
%RUNFINALFREQ(VAR=BDOE_TERM_PUF1);
%RUNFINALFREQ(VAR=BDOE_TERM_PUF2);
%RUNFINALFREQ(VAR=BDOF_PUF1);
%RUNFINALFREQ(VAR=BDOF_PUF2);
%RUNFINALFREQ(VAR=BIC_PUF);
  %RUNFINALFREQ(VAR=BIC_HIST_PUF1);
  %RUNFINALFREQ(VAR=BIC_HIST_PUF2);
  %RUNFINALFREQ(VAR=BIC_HIST_PUF3);
  %RUNFINALFREQ(VAR=BIC_HIST_START_PUF1);
  %RUNFINALFREQ(VAR=BIC_HIST_START_PUF2);
  %RUNFINALFREQ(VAR=BIC_HIST_START_PUF3);
%RUNFINALFREQ(VAR=CLM_FIL_PUF);
%RUNFINALFREQ(VAR=DD_PUF1);
%RUNFINALFREQ(VAR=DD_PUF2);
%RUNFINALFREQ(VAR=DD_PUF3);

```

## Appendix A.262

JCL/SAS Code: PUFDMG

```
%RUNFINALFREQ(VAR=DD_PUF4);
%RUNFINALFREQ(VAR=DD_PUF5);
%RUNFINALFREQ(VAR=DOBBEST_PUF);
%RUNFINALFREQ(VAR=DODBEST_PUF);
%RUNFINALFREQ(VAR=DODEC_PUF1);
%RUNFINALFREQ(VAR=DODEC_PUF2);
%RUNFINALFREQ(VAR=DODEC_PUF3);
%RUNFINALFREQ(VAR=DODEC_PUF4);
%RUNFINALFREQ(VAR=DODEC_PUF5);
%RUNFINALFREQ(VAR=DOEC_PUF);
%RUNFINALFREQ(VAR=DOEI_PUF);
%RUNFINALFREQ(VAR=DR_PUF1);
%RUNFINALFREQ(VAR=DR_PUF2);
%RUNFINALFREQ(VAR=DR_PUF3);
%RUNFINALFREQ(VAR=DR_PUF4);
%RUNFINALFREQ(VAR=DR_PUF5);
%RUNFINALFREQ(VAR=DT_PUF1);
%RUNFINALFREQ(VAR=DT_PUF2);
%RUNFINALFREQ(VAR=DT_PUF3);
%RUNFINALFREQ(VAR=DT_PUF4);
%RUNFINALFREQ(VAR=DT_PUF5);
%RUNFINALFREQ(VAR=DXPRIBEST_PUF);
%RUNFINALFREQ(VAR=DXSECBEST_PUF);
  *%RUNFINALFREQ(VAR=ELG_RD_PUF);
%RUNFINALFREQ(VAR=FIRSTDD_PUF);
%RUNFINALFREQ(VAR=FIRSTDR_PUF);
%RUNFINALFREQ(VAR=FIRSTDT_PUF);
%RUNFINALFREQ(VAR=FRA_PUF);
%RUNFINALFREQ(VAR=FRADATE_PUF);
  *%RUNFINALFREQ(VAR=LAF_MR_PUF);
  *%RUNFINALFREQ(VAR=LAF_MR_DATE_PUF);
%RUNFINALFREQ(VAR=MFT_PUF);
%RUNFINALFREQ(VAR=PIA_MAX_PUF);
%RUNFINALFREQ(VAR=PIA_MIN_PUF);
%RUNFINALFREQ(VAR=PIED_MAX_PUF);
%RUNFINALFREQ(VAR=PIED_MIN_PUF);
%RUNFINALFREQ(VAR=REPPYSSD_PUF);
%RUNFINALFREQ(VAR=REPPYSSI_PUF);
%RUNFINALFREQ(VAR=PAYEE_TYP_SSI_PUF);
%RUNFINALFREQ(VAR=SEX_PUF);
%RUNFINALFREQ(VAR=SSDI_ONSET_PUF);
  *%RUNFINALFREQ(VAR=SSDI_PUF94);
  *%RUNFINALFREQ(VAR=SSDI_PUF95);
  *%RUNFINALFREQ(VAR=SSDI_PUF96);
  *%RUNFINALFREQ(VAR=SSDI_PUF97);
```

Appendix A.262

JCL/SAS Code: PUFDMG

```
*%RUNFINALFREQ(VAR=SSDI_PUF98);
*%RUNFINALFREQ(VAR=SSDI_PUF99);
*%RUNFINALFREQ(VAR=SSDI_PUF00);
*%RUNFINALFREQ(VAR=SSDI_PUF01);
*%RUNFINALFREQ(VAR=SSDI_PUF02);
*%RUNFINALFREQ(VAR=SSDI_PUF03);
*%RUNFINALFREQ(VAR=SSDI_PUF04);
*%RUNFINALFREQ(VAR=SSDI_PUF05);
*%RUNFINALFREQ(VAR=SSDI_PUF06);
*%RUNFINALFREQ(VAR=SSDI_PUF07);
*%RUNFINALFREQ(VAR=SSDI_PUF08);
*%RUNFINALFREQ(VAR=SSDI_PUF09);
*%RUNFINALFREQ(VAR=SSDI_PUF10);
*%RUNFINALFREQ(VAR=SSDI_PUF11);
*%RUNFINALFREQ(VAR=SSDI_PUF12);
*%RUNFINALFREQ(VAR=SSDI_PUF13);
*%RUNFINALFREQ(VAR=SSDI_PUF14);
*%RUNFINALFREQ(VAR=SSDI_PUF15);
*%RUNFINALFREQ(VAR=SSDI_PUF16);
*%RUNFINALFREQ(VAR=SSDI_PUF17);
*%RUNFINALFREQ(VAR=SSDI_PUF18);
*%RUNFINALFREQ(VAR=SSDI_PUF19);
*%RUNFINALFREQ(VAR=SSDI_PUF20);
*%RUNFINALFREQ(VAR=SSDI_PUF21);
*%RUNFINALFREQ(VAR=SSI_PUF94);
*%RUNFINALFREQ(VAR=SSI_PUF95);
*%RUNFINALFREQ(VAR=SSI_PUF96);
*%RUNFINALFREQ(VAR=SSI_PUF97);
*%RUNFINALFREQ(VAR=SSI_PUF98);
*%RUNFINALFREQ(VAR=SSI_PUF99);
*%RUNFINALFREQ(VAR=SSI_PUF00);
*%RUNFINALFREQ(VAR=SSI_PUF01);
*%RUNFINALFREQ(VAR=SSI_PUF02);
*%RUNFINALFREQ(VAR=SSI_PUF03);
*%RUNFINALFREQ(VAR=SSI_PUF04);
*%RUNFINALFREQ(VAR=SSI_PUF05);
*%RUNFINALFREQ(VAR=SSI_PUF06);
*%RUNFINALFREQ(VAR=SSI_PUF07);
*%RUNFINALFREQ(VAR=SSI_PUF08);
*%RUNFINALFREQ(VAR=SSI_PUF09);
*%RUNFINALFREQ(VAR=SSI_PUF10);
*%RUNFINALFREQ(VAR=SSI_PUF11);
*%RUNFINALFREQ(VAR=SSI_PUF12);
*%RUNFINALFREQ(VAR=SSI_PUF13);
*%RUNFINALFREQ(VAR=SSI_PUF14);
```

Appendix A.262

JCL/SAS Code: PUFDMG

```

*%RUNFINALFREQ(VAR=SSI_PUF15);
*%RUNFINALFREQ(VAR=SSI_PUF16);
*%RUNFINALFREQ(VAR=SSI_PUF17);
*%RUNFINALFREQ(VAR=SSI_PUF18);
*%RUNFINALFREQ(VAR=SSI_PUF19);
*%RUNFINALFREQ(VAR=SSI_PUF20);
*%RUNFINALFREQ(VAR=SSI_PUF21);
%RUNFINALFREQ(VAR=SSIELIG_FIRST_PUF);
%RUNFINALFREQ(VAR=TKT_PUF0205);
%RUNFINALFREQ(VAR=TKT_PUF06);
%RUNFINALFREQ(VAR=TKT_PUF07);
%RUNFINALFREQ(VAR=TKT_PUF08);
%RUNFINALFREQ(VAR=TKT_PUF09);
%RUNFINALFREQ(VAR=TKT_PUF10);
%RUNFINALFREQ(VAR=TKT_PUF11);
%RUNFINALFREQ(VAR=TKT_PUF12);
%RUNFINALFREQ(VAR=TKT_PUF13);
%RUNFINALFREQ(VAR=TKT_PUF14);
%RUNFINALFREQ(VAR=TKT_PUF15);
%RUNFINALFREQ(VAR=TKT_PUF16);
%RUNFINALFREQ(VAR=TKT_PUF17);
%RUNFINALFREQ(VAR=TKT_PUF18);
%RUNFINALFREQ(VAR=TKT_PUF19);
%RUNFINALFREQ(VAR=TKT_PUF20);
%RUNFINALFREQ(VAR=TKT_PUF21);
%RUNFINALFREQ(VAR=TOC_NUM_PUF);
%RUNFINALFREQ(VAR=TOC_PUF1);
%RUNFINALFREQ(VAR=TOC_PUF2);
%RUNFINALFREQ(VAR=TOC_START_PUF1);
%RUNFINALFREQ(VAR=TOC_START_PUF2);
*%RUNFINALFREQ(VAR=TOC_HIST_PUF1);
*%RUNFINALFREQ(VAR=TOC_HIST_PUF2);
*%RUNFINALFREQ(VAR=TOC_HIST_PUF3);
*%RUNFINALFREQ(VAR=TOC_HIST_START_PUF1);
*%RUNFINALFREQ(VAR=TOC_HIST_START_PUF2);
*%RUNFINALFREQ(VAR=TOC_HIST_START_PUF3);
%RUNFINALFREQ(VAR=PAYEE_TYP_SSDI_PUF);
%RUNFINALFREQ(VAR=TWPCMPLMNTH_PUF1);

//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY

```

Appendix A.262

JCL/SAS Code: PUFDMG

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 PUF DMG COMPLETED
DAF21 DMG PUF HAS FINISHED RUNNING
```

## Appendix A.263

JCL/SAS Code: Y94JCL

```
//#8047A94 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="1994"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1994E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY94.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW94.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW94.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN94.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.263

JCL/SAS Code: Y94JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y94 IS COMPLETE.
```

## Appendix A.264

JCL/SAS Code: Y95JCL

```
//#8047A94 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="1995"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1995E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY95.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW95.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW95.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN95.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```



Appendix A.264

JCL/SAS Code: Y95JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y95 IS COMPLETE.
```

## Appendix A.265

JCL/SAS Code: Y96JCL

```
//#8047A94 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="1996"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1996E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY96.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW96.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW96.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN96.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.265

JCL/SAS Code: Y96JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y96 IS COMPLETE.
```

## Appendix A.266

JCL/SAS Code: Y97JCL

```
//#8047A94 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="1997"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1997E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY97.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW97.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW97.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN97.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.266

JCL/SAS Code: Y97JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y97 IS COMPLETE.
```

## Appendix A.267

JCL/SAS Code: Y98JCL

```

//#8047A98 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="1998"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1998E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY98.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW98.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW98.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN98.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY

```

Appendix A.267

JCL/SAS Code: Y98JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y98 IS COMPLETE.
```

## Appendix A.268

JCL/SAS Code: Y99JCL

```
//#8047A98 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="1999"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1999E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY99.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW99.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW99.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN99.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```



Appendix A.268

JCL/SAS Code: Y99JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y99 IS COMPLETE.
```

## Appendix A.269

JCL/SAS Code: Y00JCL

```

//#8047A98 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2000"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2000E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY00.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW00.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW00.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN00.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY

```

Appendix A.269

JCL/SAS Code: Y00JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y00 IS COMPLETE.
```

## Appendix A.270

JCL/SAS Code: Y01JCL

```
//#8047A98 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2001"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2001E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY01.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW01.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW01.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN01.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.270

JCL/SAS Code: Y01JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y01 IS COMPLETE.
```

## Appendix A.271

JCL/SAS Code: Y02JCL

```
//#8047A02 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2002"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2002E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY02.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW02.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW02.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN02.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.271

JCL/SAS Code: Y02JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y02 IS COMPLETE.
```

## Appendix A.272

JCL/SAS Code: Y03JCL

```

//#8047A02 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2003"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2003E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY03.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW03.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW03.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN03.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY

```



Appendix A.272

JCL/SAS Code: Y03JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y03 IS COMPLETE.
```

## Appendix A.273

JCL/SAS Code: Y04JCL

```
//#8047A02 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2004"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2004E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY04.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW04.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW04.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN04.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.273

JCL/SAS Code: Y04JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y04 IS COMPLETE.
```

## Appendix A.274

JCL/SAS Code: Y05JCL

```
//#8047A02 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2005"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2005E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY05.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW05.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW05.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN05.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.274

JCL/SAS Code: Y05JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y05 IS COMPLETE.
```

## Appendix A.275

JCL/SAS Code: Y06JCL

```
//#8047A06 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2006"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2006E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY06.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW06.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW06.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN06.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.275

JCL/SAS Code: Y06JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y06 IS COMPLETE.
```

## Appendix A.276

JCL/SAS Code: Y07JCL

```

//#8047A06 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2007"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2007E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY07.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW07.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW07.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN07.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY

```



Appendix A.276

JCL/SAS Code: Y07JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y07 IS COMPLETE.
```

## Appendix A.277

JCL/SAS Code: Y08JCL

```

//#8047A06 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2008"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2008E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY08.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW08.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW08.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN08.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY

```

Appendix A.277

JCL/SAS Code: Y08JCL

```
//SYSRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: PUF Y08 IS COMPLETE.
```

## Appendix A.278

JCL/SAS Code: Y09JCL

```
//#8047A06 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2009"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2009E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY09.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW09.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW09.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN09.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.278

JCL/SAS Code: Y09JCL

```
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: PUF Y09 IS COMPLETE.
```

## Appendix A.279

JCL/SAS Code: Y10JCL

```
//#8047A10 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2010"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2010E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY10.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW10.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW10.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN10.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.279

JCL/SAS Code: Y10JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y10 IS COMPLETE.
```

## Appendix A.280

JCL/SAS Code: Y11JCL

```

//#8047A10 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2011"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2011E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY11.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW11.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW11.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN11.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY

```



Appendix A.280

JCL/SAS Code: Y11JCL

```
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: PUF Y11 IS COMPLETE.
```

## Appendix A.281

JCL/SAS Code: Y12JCL

```
//#8047A10 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2012"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2012E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY12.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW12.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW12.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN12.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.281

JCL/SAS Code: Y12JCL

```
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>  
DATA  
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>  
SUBJECT: PUF Y12 IS COMPLETE.
```

## Appendix A.282

JCL/SAS Code: Y13JCL

```
//#8047A10 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2013"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2013E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY13.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW13.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW13.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN13.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.282

JCL/SAS Code: Y13JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y13 IS COMPLETE.
```

## Appendix A.283

JCL/SAS Code: Y14JCL

```
//#8047A14 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2014"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2014E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY14.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW14.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW14.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN14.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.283

JCL/SAS Code: Y14JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y14 IS COMPLETE.
```

## Appendix A.284

JCL/SAS Code: Y15JCL

```
//#8047A14 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2015"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2015E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY15.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW15.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW15.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN15.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```



Appendix A.284

JCL/SAS Code: Y15JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y15 IS COMPLETE.
```

## Appendix A.285

JCL/SAS Code: Y16JCL

```
//#8047A14 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2016"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2016E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY16.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW16.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW16.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN16.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.285

JCL/SAS Code: Y16JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y16 IS COMPLETE.
```

## Appendix A.286

JCL/SAS Code: Y17JCL

```
//#8047A14 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2017"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2017E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY17.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW17.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW17.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN17.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.286

JCL/SAS Code: Y17JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y17 IS COMPLETE.
```

## Appendix A.287

JCL/SAS Code: Y18JCL

```
//#8047A18 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2018"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2018E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY18.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW18.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW18.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN18.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.287

JCL/SAS Code: Y18JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y17 IS COMPLETE.
```

## Appendix A.288

JCL/SAS Code: Y19JCL

```
//#8047A18 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2019"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2019E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY19.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW19.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW19.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN19.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```



Appendix A.288

JCL/SAS Code: Y19JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y19 IS COMPLETE.
```

## Appendix A.289

JCL/SAS Code: Y20JCL

```

//#8047A18 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2020"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2020E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY20.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW20.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW20.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN20.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY

```

Appendix A.289

JCL/SAS Code: Y20JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y20 IS COMPLETE.
```

## Appendix A.290

JCL/SAS Code: Y21JCL

```
//#8047A18 JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* DAF50214
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(PUFANN)  |
//*  *-----*
//* PURPOSE:
//* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS
//*
//* TO CHANGE THE ANNUALS YEAR, UPDATE DSN'S AND SYSPARM MACRO
//* PROGRAMMER: ALEX BRYCE
//* CONTACT: ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000',          * SPACE IN BLKS *
//  PARM='SYSPARM="2021"'
//*
//ANNU DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2021E,DISP=SHR
//SAMP DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21T.P10PY21.SA.V1,DISP=SHR
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//TEMP4 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  SPACE=(CYL,(3000,3000),RLSE),VOL=(,,20)
//*RAW1 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF21P.PUF.RW20.SA.V1,DISP=SHR
//RAW1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.RW21.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN21.SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.PUF.PRDLIB(PUFANN),
//  DISP=(SHR,PASS,KEEP)
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.290

JCL/SAS Code: Y21JCL

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: PUF Y21 IS COMPLETE.
```

## Appendix A.291

JCL/SAS Code: PUFANN

```

/*****/
/*                               */
/* *-----* */
/* | OPDR.TG.PRD.ETTW.#6502.DAF15.PUF.PRDLIB(PUFANNTX) | */
/* *-----* */
/*                               */
/* CREATE ANNUALS PUF FILE FROM RANDOM 10% OF ANNUALS */
/*                               */
/* TO CHANGE THE ANNUALS YEAR, UPDATE YXXJCL PROGRAM */
/*****/

```

```

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS = MAX MPRINT
MACROGEN;

```

```

%LET YR = %SUBSTR(&SYSPARM,3,2); /* SET IN YYYJCL PROGRAM */
%LET YEAR = &SYSPARM;

```

```

%MACRO DOIT;
/*
* SORT BY SSN;
PROC SORT DATA= SAMP.P10SAMP OUT= TEMP1.SAMP;
  BY SSN;
RUN;
*/
/*SAVE ANNUALS PARED DOWN TO PUF BENES FOR FASTER ADHOC CHECKS */
DATA RAW1.ANN&YR.PUFRAW;
  MERGE SAMP.P10SAMP (IN=A)
        ANNU.Y&YEAR. (IN=B);
  BY SSN;
  IF A AND B;
RUN;

/*PROC CONTENTS DATA = RAW1.ANN&YR.PUFRAW; RUN; */

```

```

DATA TEMP1.ANN&YR.PUF;
  SET RAW1.ANN&YR.PUFRAW;
  BY SSN;

```

```

%MACRO MLOOP(VAR=);

```

```

%DO Y= 1 %TO 12;
%LET I = %SYSFUNC(PUTN(&Y.,Z2));
  IF &VAR.&YR.&I = 1 THEN &VAR._&I = 1;

```

## Appendix A.291

JCL/SAS Code: PUFANN

```

ELSE IF &VAR.&YR.&I = 0 OR
(MISSING(&VAR.&YR.&I) AND STWSSI&YR.&I IN (1,2,3,4))
THEN &VAR._&I = 0;

ELSE IF &VAR.&YR.&I NOT IN (0,1) AND
(NOT MISSING(PSTA&YR.&I) OR STWSSI&YR.&I IN (8,9))
THEN &VAR._&I = .M;

ELSE IF MISSING(&VAR.&YR.&I) THEN &VAR._&I = .N;
%END;

IF &VAR._01=1 OR &VAR._02=1 OR &VAR._03=1 OR &VAR._04=1 OR &VAR._05= 1
OR &VAR._06=1 OR &VAR._07=1 OR &VAR._08=1 OR &VAR._09=1
OR &VAR._10=1 OR &VAR._11=1 OR &VAR._12=1 THEN &VAR._PUF&YR. = 1;

ELSE IF &VAR._01=0 OR &VAR._02=0 OR &VAR._03=0 OR &VAR._04=0
OR &VAR._05=0 OR &VAR._06=0 OR &VAR._07=0 OR &VAR._08=0
OR &VAR._09=0 OR &VAR._10=0 OR &VAR._11=0 OR &VAR._12=0
THEN &VAR._PUF&YR. = 0;

ELSE IF &VAR._01=.M OR &VAR._02=.M OR &VAR._03=.M OR &VAR._04=.M
OR &VAR._05=.M OR &VAR._06=.M OR &VAR._07=.M OR &VAR._08=.M
OR &VAR._09=.M OR &VAR._10=.M OR &VAR._11=.M OR &VAR._12=.M
THEN &VAR._PUF&YR. = .M;

ELSE &VAR._PUF&YR. = .N;

%MEND MLOOP;
%MLOOP(VAR=PROA);
%MLOOP(VAR=PROB);

LABEL PROA_PUF&YR. = "SSI USE OF 1619A PROVISION INDICATOR (YY=YEAR)"
PROB_PUF&YR. = "SSI USE OF 1619B PROVISION INDICATOR (YY=YEAR)";

%DO Y= 1 %TO 12; /* BEGIN MONTH LOOP */
%LET I = %SYSFUNC(PUTN(&Y.,Z2));

* MEDR;
IF MEDR&YR.&I = 0 THEN MEDR_PUF&YR.&I = 0;
ELSE IF MEDR&YR.&I=1 THEN MEDR_PUF&YR.&I. = 1;
ELSE IF MISSING(MEDR&YR.&I) AND NOT MISSING(LAF&YR.&I)
THEN MEDR_PUF&YR.&I. = .M;
ELSE IF MISSING(MEDR&YR.&I) AND MISSING(LAF&YR.&I)
THEN MEDR_PUF&YR.&I. = .N;

```

## Appendix A.291

JCL/SAS Code: PUFANN

```
* SSICOUPL;
IF CURCOMP&YR.&I = 6 THEN SSICOUPL_PUF&YR.&I = 1;
ELSE IF CURCOMP&YR.&I >= 0 THEN SSICOUPL_PUF&YR.&I = 0;
ELSE IF NOT MISSING(PSTA&YR.&I) AND MISSING(CURCOMP&YR.&I)
  THEN SSICOUPL_PUF&YR.&I = .M;
ELSE IF MISSING(CURCOMP&YR.&I) AND MISSING(PSTA&YR.&I)
  THEN SSICOUPL_PUF&YR.&I = .N;

* LIVE;
IF LIVF&YR.&I NOT IN ('A','B','C','D') AND NOT MISSING(PSTA&YR.&I)
  THEN LIVF_PUF&YR.&I = '.M';
ELSE IF LIVF&YR.&I NOT IN ('A','B','C','D') AND MISSING(PSTA&YR.&I)
  THEN LIVF_PUF&YR.&I = '.N';
ELSE IF LIVF&YR.&I IN ('A','B','C','D') THEN
  LIVF_PUF&YR.&I = LIVF&YR.&I;

* CDR;
* INCLUDED X IN GROUP 3 FOR DAA MASKING;
IF CDR&YR.&I IN ('E','S') THEN CDR_PUF&YR.&I = 1;
ELSE IF CDR&YR.&I='M' THEN CDR_PUF&YR.&I = 2;
ELSE IF CDR&YR.&I IN ('F','N','V','W','X') THEN CDR_PUF&YR.&I = 3;
ELSE IF MISSING(CDR&YR.&I) AND NOT MISSING(LAF&YR.&I)
  THEN CDR_PUF&YR.&I = .M;
ELSE IF MISSING(CDR&YR.&I) AND MISSING(LAF&YR.&I)
  THEN CDR_PUF&YR.&I = .N;

* LAF;
LENGTH LAF_PUF&YR.&I. $2;
IF SUBSTR(LAF&YR.&I,1,1) = 'C' THEN LAF_PUF&YR.&I = 'C';
ELSE IF SUBSTR(LAF&YR.&I,1,1) = 'S' THEN LAF_PUF&YR.&I = 'S';
ELSE IF SUBSTR(LAF&YR.&I,1,1) = 'T' THEN LAF_PUF&YR.&I = 'T';
ELSE IF NOT MISSING(LAF&YR.&I) THEN LAF_PUF&YR.&I = 'O';
ELSE IF MISSING(LAF&YR.&I) THEN LAF_PUF&YR.&I = '.N';

*PSTA;
IF SUBSTR(PSTA&YR.&I,1,1) IN ('C','M') THEN PSTA_PUF&YR.&I = 1;
ELSE IF PSTA&YR.&I IN ('N31','N32','N33','N42','N43')
  THEN PSTA_PUF&YR.&I = 2;
ELSE IF SUBSTR(PSTA&YR.&I,1,1) IN ('E','N')
  OR PSTA&YR.&I IN ('W01')
  THEN PSTA_PUF&YR.&I.=3;
ELSE IF SUBSTR(PSTA&YR.&I,1,1) = 'T' THEN PSTA_PUF&YR.&I = 4;
ELSE IF NOT MISSING(PSTA&YR.&I) THEN PSTA_PUF&YR.&I = 5;
ELSE IF MISSING(PSTA&YR.&I) THEN PSTA_PUF&YR.&I = .N;
```



## Appendix A.291

JCL/SAS Code: PUFANN

```
* STWDI;
IF STWDI&YR.&I = 0 THEN STWDI_PUF&YR.&I. = 0;
ELSE IF STWDI&YR.&I = 1 THEN STWDI_PUF&YR.&I. = 1;
ELSE IF STWDI&YR.&I IN (2,3) THEN STWDI_PUF&YR.&I. = 2;
ELSE IF STWDI&YR.&I IN (8,9) THEN STWDI_PUF&YR.&I. = 8;
ELSE IF MISSING (STWDI&YR.&I) AND NOT MISSING (LAF&YR.&I)
  THEN STWDI_PUF&YR.&I. = .M;
ELSE IF MISSING (STWDI&YR.&I) AND MISSING (LAF&YR.&I)
  THEN STWDI_PUF&YR.&I. = .N;

* STWSSI;
IF STWSSI&YR.&I = 0 THEN STWSSI_PUF&YR.&I. = 0;
ELSE IF STWSSI&YR.&I = 1 THEN STWSSI_PUF&YR.&I.=1;
ELSE IF STWSSI&YR.&I IN (2,3) THEN STWSSI_PUF&YR.&I.=2;
ELSE IF STWSSI&YR.&I IN (8,9,4) THEN STWSSI_PUF&YR.&I. = 8;
ELSE IF MISSING(STWSSI&YR.&I) AND NOT MISSING(PSTA&YR.&I)
  THEN STWSSI_PUF&YR.&I. = .M;
ELSE IF MISSING(STWSSI&YR.&I) AND MISSING(PSTA&YR.&I)
  THEN STWSSI_PUF&YR.&I. = .N;

* STWCM;
IF STWCM&YR.&I = 0 THEN STWCM_PUF&YR.&I. = 0;
ELSE IF STWCM&YR.&I = 1 THEN STWCM_PUF&YR.&I. = 1;
ELSE IF STWCM&YR.&I IN (2,3) THEN STWCM_PUF&YR.&I. = 2;
ELSE IF STWCM&YR.&I IN (8,9) THEN STWCM_PUF&YR.&I. = 8;
ELSE IF MISSING(STWCM&YR.&I) AND (NOT MISSING(LAF&YR.&I)
  OR NOT MISSING(PSTA&YR.&I)) THEN STWCM_PUF&YR.&I. = .M;
ELSE IF MISSING(STWCM&YR.&I) AND (MISSING(LAF&YR.&I)
  AND MISSING(PSTA&YR.&I)) THEN STWCM_PUF&YR.&I. = .N;

* TWPDATA;
LENGTH TWPDATA_PUF&YR.&I. $2;
IF MISSING(STWDI&YR.&I) AND MISSING(TWPDATA&YR.&I)
  THEN TWPDATA_PUF&YR.&I. = '.N';
ELSE IF STWDI&YR.&I IN (8,9) AND MISSING(TWPDATA&YR.&I)
  THEN TWPDATA_PUF&YR.&I. = '.M';
ELSE IF TWPDATA&YR.&I IN ('1','2','3','4','5','6','7','8','9') AND
  STWDI&YR.&I IN (0,1,2,3) THEN TWPDATA_PUF&YR.&I. = 'T';
ELSE IF TWPDATA&YR.&I = 'N' AND STWDI&YR.&I IN (0,1,2,3)
  THEN TWPDATA_PUF&YR.&I. = 'N';
ELSE IF TWPDATA&YR.&I IN ('E','S','U') AND
  STWDI&YR.&I IN (0,1,2,3) THEN TWPDATA_PUF&YR.&I. = 'S';
ELSE IF NOT MISSING (TWPDATA&YR.&I) AND STWDI&YR.&I IN (0,1,2,3)
  THEN TWPDATA_PUF&YR.&I. = 'O';
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
ELSE IF MISSING (TWPDATA&YR.&I) AND STWDI&YR.&I IN (0,1,2,3)
  THEN TWPDATA_PUF&YR.&I. = '.A';
ELSE TWPDATA_PUF&YR.&I. = '.M';

* DIERNLVL;
IF MISSING(LAF&YR.&I) AND MISSING(DIERNLVL&YR.&I)
  THEN DIERNLVL_PUF&YR.&I. = '.N';
ELSE IF NOT MISSING(LAF&YR.&I) AND DIERNLVL&YR.&I NOT IN
  ('O','L','T','G','S') THEN DIERNLVL_PUF&YR.&I. = '.M';
ELSE IF DIERNLVL&YR.&I IN ('O','L','T','G','S') THEN
  DIERNLVL_PUF&YR.&I. = DIERNLVL&YR.&I;
* AGGREGATE S AND G TOGETHER;
IF DIERNLVL&YR.&I = 'S' THEN DIERNLVL_PUF&YR.&I. = 'G';

* MEDEX;
IF (MISSING(PSTA&YR.&I) AND MISSING(LAF&YR.&I))
  AND MISSING(MEDX&YR.&I) THEN MEDX_PUF&YR.&I. = '.N';
ELSE IF (NOT MISSING(PSTA&YR.&I) OR NOT MISSING(LAF&YR.&I))
  AND MISSING(MEDX&YR.&I) THEN MEDX_PUF&YR.&I. = '.M';
ELSE MEDX_PUF&YR.&I. = MEDX&YR.&I;

* MEDE;
* NOT AVAILABLE IN JAN 94 - FEB 96;
%IF &YR.=94 OR &YR.=95 %THEN %DO;
  MEDE_PUF&YR.&I. = .P;
%END;
%ELSE %IF &YR.=96 AND (&Y.=1 OR &Y.=2) %THEN %DO;
  MEDE_PUF&YR.&I. = .P;
%END;
%ELSE %DO;
IF MISSING(PSTA&YR.&I) AND MISSING(MEDE&YR.&I)
  THEN MEDE_PUF&YR.&I. = .N;
ELSE IF NOT MISSING(PSTA&YR.&I) AND MISSING(MEDE&YR.&I)
  THEN MEDE_PUF&YR.&I. = .M;
ELSE IF MEDE&YR.&I IN (0,1) THEN MEDE_PUF&YR.&I. = MEDE&YR.&I;
%END;

* EDIT DOLLAR AMOUNT VARIABLES;

* CREATE RANDCENTS TO HELP BREAK TIES DURING TOPCODING;
RANDCENTS = RAND("UNIFORM");

* DUED;
IF MISSING(LAF&YR.&I) AND MISSING(DUED&YR.&I)
  THEN DUED_PUF&YR.&I. = .N;
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
ELSE IF NOT MISSING(LAF&YR.&I) AND LAF&YR.&I ~= 'C'
  AND MISSING(DUED&YR.&I) THEN DUED_PUF&YR.&I. = .M;
ELSE IF MISSING(DUED&YR.&I) AND LAF&YR.&I = 'C' THEN DUED_PUF&YR.&I. =0;
ELSE DUED_PUF&YR.&I. = DUED&YR.&I;
IF DUED_PUF&YR.&I. > 0 THEN
  DUED_CENTS&YR.&I = SUM(DUED_PUF&YR.&I.,RANDCENTS);

*PAYD;
IF MISSING(LAF&YR.&I) AND MISSING(PAYD&YR.&I)
  THEN PAYD_PUF&YR.&I. = .N;
ELSE IF NOT MISSING(LAF&YR.&I) AND LAF&YR.&I ~= 'C'
  AND MISSING(PAYD&YR.&I) THEN PAYD_PUF&YR.&I.=.M;
ELSE IF MISSING(PAYD&YR.&I) AND LAF&YR.&I = 'C' THEN PAYD_PUF&YR.&I.=0;
ELSE PAYD_PUF&YR.&I. = PAYD&YR.&I;
IF NOT MISSING(PAYD_PUF&YR.&I.) AND PAYD_PUF&YR.&I. < 0
  THEN PAYD_PUF&YR.&I. = 0;
IF PAYD_PUF&YR.&I. > 0 THEN
  PAYD_CENTS&YR.&I = SUM(PAYD_PUF&YR.&I.,RANDCENTS);

* PAYO;
IF MISSING(LAF&YR.&I) AND MISSING(PAYO&YR.&I)
  THEN PAYO_PUF&YR.&I. = .N;
ELSE IF NOT MISSING(LAF&YR.&I) AND LAF&YR.&I ~= 'C'
  AND MISSING(PAYO&YR.&I) THEN PAYO_PUF&YR.&I. = .M;
ELSE IF LAF&YR.&I='C' AND MISSING(PAYO&YR.&I) THEN PAYO_PUF&YR.&I.=0;
ELSE PAYO_PUF&YR.&I. = PAYO&YR.&I;
IF NOT MISSING(PAYO_PUF&YR.&I.) AND PAYO_PUF&YR.&I. < 0
  THEN PAYO_PUF&YR.&I. = 0;
IF PAYO_PUF&YR.&I. > 0 THEN
  PAYO_CENTS&YR.&I = SUM(PAYO_PUF&YR.&I.,RANDCENTS);

* DIRPAY;
IF MISSING(LAF&YR.&I) AND MISSING(DIRPAY&YR.&I)
  THEN DIRPAY_PUF&YR.&I. = .N;
ELSE IF NOT MISSING(LAF&YR.&I) AND LAF&YR.&I ~= 'C'
  AND MISSING(DIRPAY&YR.&I) THEN DIRPAY_PUF&YR.&I.=.M;
ELSE IF MISSING(DIRPAY&YR.&I) AND LAF&YR.&I = 'C'
  THEN DIRPAY_PUF&YR.&I.=0;
ELSE DIRPAY_PUF&YR.&I. = DIRPAY&YR.&I;
IF NOT MISSING(DIRPAY_PUF&YR.&I.) AND DIRPAY_PUF&YR.&I. < 0
  THEN DIRPAY_PUF&YR.&I. = .M;
IF DIRPAY_PUF&YR.&I. > 0 THEN
  DIRPAY_CENTS&YR.&I = SUM(DIRPAY_PUF&YR.&I.,RANDCENTS);

* MEDPREM;
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
IF MISSING(LAF&YR.&I) AND MISSING(MEDPREM&YR.&I)
  THEN MEDPREM_PUF&YR.&I = .N;
ELSE IF LAF&YR.&I ~= 'C' AND NOT MISSING(LAF&YR.&I)
  AND MISSING(MEDPREM&YR.&I) THEN MEDPREM_PUF&YR.&I = .M;
ELSE IF MISSING(MEDPREM&YR.&I) AND LAF&YR.&I = 'C'
  THEN MEDPREM_PUF&YR.&I = .U;
ELSE MEDPREM_PUF&YR.&I = MEDPREM&YR.&I;
IF NOT MISSING(MEDPREM_PUF&YR.&I.) AND MEDPREM_PUF&YR.&I. < 0
  THEN MEDPREM_PUF&YR.&I. = .M;
IF MEDPREM_PUF&YR.&I. > 0 THEN
  MEDPREM_CENTS&YR.&I = SUM(MEDPREM_PUF&YR.&I.,RANDCENTS);

* DUES;
IF MISSING(PSTA&YR.&I) AND MISSING(DUES&YR.&I)
  THEN DUES_PUF&YR.&I = .N;
ELSE IF NOT MISSING(PSTA&YR.&I) AND PSTA&YR.&I ~= 'C01'
  AND MISSING(DUES&YR.&I) THEN DUES_PUF&YR.&I = .M;
ELSE IF MISSING(DUES&YR.&I) AND PSTA&YR.&I = 'C01'
  THEN DUES_PUF&YR.&I = 0;
ELSE DUES_PUF&YR.&I = DUES&YR.&I;
IF DUES_PUF&YR.&I. > 0 THEN
  DUES_CENTS&YR.&I = SUM(DUES_PUF&YR.&I.,RANDCENTS);

* PAYS;
IF MISSING(PSTA&YR.&I) THEN PAYS_PUF&YR.&I = .N;
ELSE IF NOT MISSING(PSTA&YR.&I) AND PSTA&YR.&I ~= 'C01'
  THEN PAYS_PUF&YR.&I = .M;
ELSE IF MISSING(PAYS&YR.&I) AND PSTA&YR.&I = 'C01'
  THEN PAYS_PUF&YR.&I = 0;
ELSE PAYS_PUF&YR.&I = PAYS&YR.&I.;
IF NOT MISSING(PAYS_PUF&YR.&I.) AND PAYS_PUF&YR.&I. < 0
  THEN PAYS_PUF&YR.&I. = 0;
IF PAYS_PUF&YR.&I. > 0 THEN
  PAYS_CENTS&YR.&I = SUM(PAYS_PUF&YR.&I.,RANDCENTS);

* T2GRSAMT;
IF MISSING(LAF&YR.&I) AND MISSING(T2GRSAMT&YR.&I)
  THEN T2GRSAMT_PUF&YR.&I = .N;
ELSE IF NOT MISSING(LAF&YR.&I) AND MISSING(T2GRSAMT&YR.&I)
  THEN T2GRSAMT_PUF&YR.&I = .M;
ELSE T2GRSAMT_PUF&YR.&I = T2GRSAMT&YR.&I;
IF NOT MISSING(T2GRSAMT_PUF&YR.&I.) AND T2GRSAMT_PUF&YR.&I. < 0
  THEN T2GRSAMT_PUF&YR.&I. = 0;
IF T2GRSAMT_PUF&YR.&I. > 0 THEN
  T2GRSAMT_CENTS&YR.&I = SUM(T2GRSAMT_PUF&YR.&I.,RANDCENTS);
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
* T2NETAMT;
IF MISSING(LAF&YR.&I) AND MISSING(T2NETAMT&YR.&I)
  THEN T2NETAMT_PUF&YR.&I = .N;
ELSE IF MISSING(T2NETAMT&YR.&I) AND NOT MISSING(LAF&YR.&I)
  THEN T2NETAMT_PUF&YR.&I = 0;
ELSE T2NETAMT_PUF&YR.&I = T2NETAMT&YR.&I;
IF NOT MISSING(T2NETAMT_PUF&YR.&I.) AND T2NETAMT_PUF&YR.&I. < 0
  THEN T2NETAMT_PUF&YR.&I = 0;
IF T2NETAMT_PUF&YR.&I. > 0 THEN
  T2NETAMT_CENTS&YR.&I = SUM(T2NETAMT_PUF&YR.&I.,RANDCENTS);

* BFWDI;
IF MISSING(STWDI&YR.&I) THEN BFWDI_PUF&YR.&I = .N;
ELSE IF NOT MISSING(STWDI&YR.&I) AND MISSING(BFWDI&YR.&I)
  THEN BFWDI_PUF&YR.&I = 0;
ELSE IF STWDI&YR.&I IN (0,8,9) THEN BFWDI_PUF&YR.&I = 0;
ELSE IF STWDI&YR.&I IN (1,2,3) THEN BFWDI_PUF&YR.&I = BFWDI&YR.&I;
IF BFWDI_PUF&YR.&I. > 0 THEN
  BFWDI_CENTS&YR.&I = SUM(BFWDI_PUF&YR.&I.,RANDCENTS);

* T16GRSAMT;
* NOTE: DCF DATA NOT AVAILABLE IN 1994-MARCH 2001;
* NOTE: SSR DATA AVAILABLE FROM 1994 - SWITCHING TO SSR SOURCE;
/*
%IF &YR.=94 OR &YR.=95 OR &YR.=96 OR &YR.=97 OR
  &YR.=98 OR &YR.=99 OR &YR.=00 %THEN %DO;
  T16GRSAMT_PUF&YR.&I = .P;
  T16GRSAMT_CENTS&YR.&I = .P;
%END;
%ELSE %IF &YR.=01 AND (&Y.=1 OR &Y.=2 OR &Y.=3) %THEN %DO;
  T16GRSAMT_PUF&YR.&I = .P;
  T16GRSAMT_CENTS&YR.&I = .P;
%END;
%ELSE %DO; */
IF MISSING(PSTA&YR.&I) AND MISSING(T16GRSAMT&YR.&I)
  THEN T16GRSAMT_PUF&YR.&I = .N;
ELSE IF MISSING(T16GRSAMT&YR.&I) AND NOT MISSING(PSTA&YR.&I)
  THEN T16GRSAMT_PUF&YR.&I = .M;
ELSE T16GRSAMT_PUF&YR.&I = T16GRSAMT&YR.&I;
IF NOT MISSING(T16GRSAMT_PUF&YR.&I.) AND T16GRSAMT_PUF&YR.&I. < 0
  THEN T16GRSAMT_PUF&YR.&I = 0;
IF T16GRSAMT_PUF&YR.&I. > 0 THEN
  T16GRSAMT_CENTS&YR.&I = SUM(T16GRSAMT_PUF&YR.&I.,RANDCENTS);
/*%END;*/
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
* T16NETAMT;
* NOTE: DCF DATA NOT AVAILABLE IN 1994-MARCH 2001;
* NOTE: SSR DATA AVAILABLE FROM 1994 - SWITCHING TO SSR SOURCE;
/*
%IF &YR.=94 OR &YR.=95 OR &YR.=96 OR &YR.=97 OR
  &YR.=98 OR &YR.=99 OR &YR.=00 %THEN %DO;
  T16NETAMT_PUF&YR.&I. = .P;
  T16NETAMT_CENTS&YR.&I. = .P;
%END;
%ELSE %IF &YR.=01 AND (&Y.=1 OR &Y.=2 OR &Y.=3) %THEN %DO;
  T16NETAMT_PUF&YR.&I. = .P;
  T16NETAMT_CENTS&YR.&I. = .P;
%END;
%ELSE %DO; */
IF MISSING(PSTA&YR.&I) AND MISSING(T16NETAMT&YR.&I)
  THEN T16NETAMT_PUF&YR.&I. = .N;
ELSE IF NOT MISSING(PSTA&YR.&I) AND MISSING(T16NETAMT&YR.&I)
  THEN T16NETAMT_PUF&YR.&I. = .M;
ELSE T16NETAMT_PUF&YR.&I.=T16NETAMT&YR.&I;
IF T16NETAMT_PUF&YR.&I. < 0 AND NOT MISSING(T16NETAMT_PUF&YR.&I.)
  THEN T16NETAMT_PUF&YR.&I. = 0;
IF T16NETAMT_PUF&YR.&I. > 0 THEN
  T16NETAMT_CENTS&YR.&I. = SUM(T16NETAMT_PUF&YR.&I.,RANDCENTS);
/*%END;*/

* T16EXPAMT;
* NOTE: DCF DATA NOT AVAILABLE IN 1994-MARCH 2001;
* NOTE: SSR DATA AVAILABLE FROM 1994 - SWITCHING TO SSR SOURCE;
* NOTE: NOT INCLUDED IN PUF - NOT DRB REVIEWED YET;
/*
%IF &YR.=94 OR &YR.=95 OR &YR.=96 OR &YR.=97 OR
  &YR.=98 OR &YR.=99 OR &YR.=00 %THEN %DO;
  T16EXPAMT_PUF&YR.&I. = .P;
  T16EXPAMT_CENTS&YR.&I. = .P;
%END;
%ELSE %IF &YR.=01 AND (&Y.=1 OR &Y.=2 OR &Y.=3) %THEN %DO;
  T16EXPAMT_PUF&YR.&I. = .P;
  T16EXPAMT_CENTS&YR.&I. = .P;
%END;
%ELSE %DO;
IF MISSING(T16GRSAMT&YR.&I) AND MISSING(T16EXPAMT&YR.&I)
  THEN T16EXPAMT_PUF&YR.&I. = .N;
ELSE IF MISSING(T16EXPAMT&YR.&I) THEN T16EXPAMT_PUF&YR.&I. = .M;
ELSE T16EXPAMT_PUF&YR.&I. = T16EXPAMT&YR.&I;
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
IF T16EXPAMT_PUF&YR.&I. > 0 THEN
  T16EXPAMT_CENTS&YR.&I = SUM(T16EXPAMT_PUF&YR.&I.,RANDCENTS);
%END; */

* T16PASAMT;
* NOTE: NOT AVAILABLE IN 1994-MARCH 2001;
* NOTE: SSR DATA AVAILABLE FROM 1994 - SWITCHING TO SSR SOURCE;
* NOTE: NOT INCLUDED IN PUF20 - NOT DRB REVIEWED YET;
/*
%IF &YR.=94 OR &YR.=95 OR &YR.=96 OR &YR.=97 OR
  &YR.=98 OR &YR.=99 OR &YR.=00 %THEN %DO;
  T16PASAMT_PUF&YR.&I. = .P;
  T16PASAMT_CENTS&YR.&I = .P;
%END;
%ELSE %IF &YR.=01 AND (&Y.=1 OR &Y.=2 OR &Y.=3) %THEN %DO;
  T16PASAMT_PUF&YR.&I. = .P;
  T16PASAMT_CENTS&YR.&I = .P;
%END;
%ELSE %DO;
IF MISSING(T16GRSAMT&YR.&I) AND MISSING(T16PASAMT&YR.&I)
  THEN T16PASAMT_PUF&YR.&I. = .N;
ELSE IF MISSING(T16PASAMT&YR.&I) THEN T16PASAMT_PUF&YR.&I. = .M;
ELSE T16PASAMT_PUF&YR.&I. = T16PASAMT&YR.&I;
IF T16PASAMT_PUF&YR.&I. > 0 THEN
  T16PASAMT_CENTS&YR.&I = SUM(T16PASAMT_PUF&YR.&I.,RANDCENTS);
%END; */

* T16SEAMT;
* NOTE: NOT AVAILABLE ON DCF, ONLY SSR. SWITCHED TO SSR DATA;
* NOTE: REMOVED IN PUF20 SINCE NOT DRB-REVIEWED YET;
/*
%IF &YR.=94 OR &YR.=95 OR &YR.=96 OR &YR.=97 OR
  &YR.=98 OR &YR.=99 OR &YR.=00 %THEN %DO;
  T16SEAMT_PUF&YR.&I. = .P;
  T16SEAMT_CENTS&YR.&I = .P;
%END;
%ELSE %IF &YR.=01 AND (&Y.=1 OR &Y.=2 OR &Y.=3) %THEN %DO;
  T16SEAMT_PUF&YR.&I. = .P;
  T16SEAMT_CENTS&YR.&I = .P;
%END;
%ELSE %DO;
IF MISSING(T16GRSAMT&YR.&I) AND MISSING(T16SEAMT&YR.&I)
  THEN T16SEAMT_PUF&YR.&I. = .N;
ELSE IF MISSING(T16SEAMT&YR.&I) THEN T16SEAMT_PUF&YR.&I. = .M;
ELSE T16SEAMT_PUF&YR.&I. = T16SEAMT&YR.&I;
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
IF T16SEAMT_PUF&YR.&I. > 0 THEN
  T16SEAMT_CENTS&YR.&I = SUM(T16SEAMT_PUF&YR.&I.,RANDCENTS);
%END;
*/

* T16EXLAMT;
* NOTE: DCF DATA NOT AVAILABLE IN 1994-MARCH 2001;
* NOTE: SWITCHED TO SSR DATA - AVAILABLE FROM 1994;
* NOTE: REMOVED IN PUF20 - NOT DRB REVIEWED YET;
/*
%IF &YR.=94 OR &YR.=95 OR &YR.=96 OR &YR.=97 OR
  &YR.=98 OR &YR.=99 OR &YR.=00 %THEN %DO;
  T16EXLAMT_PUF&YR.&I. = .P;
  T16EXLAMT_CENTS&YR.&I = .P;
%END;
%ELSE %IF &YR.=01 AND (&Y.=1 OR &Y.=2 OR &Y.=3) %THEN %DO;
  T16EXLAMT_PUF&YR.&I. = .P;
  T16EXLAMT_CENTS&YR.&I = .P;
%END;
%ELSE %DO;
IF MISSING(T16GRSAMT&YR.&I) AND MISSING(T16EXLAMT&YR.&I)
  THEN T16EXLAMT_PUF&YR.&I. = .N;
ELSE IF MISSING(T16EXLAMT&YR.&I) THEN T16EXLAMT_PUF&YR.&I. = .M;
ELSE T16EXLAMT_PUF&YR.&I. = T16EXLAMT&YR.&I;
IF T16EXLAMT_PUF&YR.&I. > 0 THEN
  T16EXLAMT_CENTS&YR.&I = SUM(T16EXLAMT_PUF&YR.&I.,RANDCENTS);
%END; */

* T16BEXPAMT;
* NOTE: NOT AVAILABLE ON DCF, ONLY SSR. SWITCHED TO SSR DATA;
* NOTE: REMOVED IN PUF20 SINCE NOT DRB-REVIEWED YET;
/*
%IF &YR.=94 OR &YR.=95 OR &YR.=96 OR &YR.=97 OR
  &YR.=98 OR &YR.=99 OR &YR.=00 %THEN %DO;
  T16BEXPAMT_PUF&YR.&I. = .P;
  T16BEXPAMT_CENTS&YR.&I = .P;
%END;
%ELSE %IF &YR.=01 AND (&Y.=1 OR &Y.=2 OR &Y.=3) %THEN %DO;
  T16BEXPAMT_PUF&YR.&I. = .P;
  T16BEXPAMT_CENTS&YR.&I = .P;
%END;
%ELSE %DO;
IF MISSING(T16GRSAMT&YR.&I) AND MISSING(T16BEXPAMT&YR.&I)
  THEN T16BEXPAMT_PUF&YR.&I. = .N;
ELSE IF MISSING(T16BEXPAMT&YR.&I) THEN T16BEXPAMT_PUF&YR.&I. = .M;
```



## Appendix A.291

JCL/SAS Code: PUFANN

```
ELSE T16BEXPAMT_PUF&YR.&I. = T16BEXPAMT&YR.&I.;
IF T16BEXPAMT_PUF&YR.&I. > 0 THEN
  T16BEXPAMT_CENTS&YR.&I. = SUM(T16BEXPAMT_PUF&YR.&I.,RANDCENTS);
%END;
*/

* UINC;
IF MISSING(PSTA&YR.&I) AND MISSING(UINC&YR.&I)
  THEN UINC_PUF&YR.&I. = .N;
ELSE IF NOT MISSING(PSTA&YR.&I) AND MISSING(UINC&YR.&I)
  THEN UINC_PUF&YR.&I. = .M;
ELSE UINC_PUF&YR.&I.=UINC&YR.&I.;
IF UINC_PUF&YR.&I. > 0 THEN
  UINC_CENTS&YR.&I. = SUM(UINC_PUF&YR.&I.,RANDCENTS);

* EICM;
IF MISSING(PSTA&YR.&I) AND MISSING(EICM&YR.&I)
  THEN EICM_PUF&YR.&I. = .N;
ELSE IF NOT MISSING(PSTA&YR.&I) AND MISSING(EICM&YR.&I)
  THEN EICM_PUF&YR.&I. = .M;
ELSE EICM_PUF&YR.&I.=EICM&YR.&I.;
IF EICM_PUF&YR.&I. > 0 THEN
  EICM_CENTS&YR.&I. = SUM(EICM_PUF&YR.&I.,RANDCENTS);

* BFWSSI;
* NOT AVAILABLE IN JAN-FEB 1994;
%IF &YR. = 94 AND (&Y.=1 OR &Y.=2) %THEN %DO;
  BFWSSI_PUF&YR.&I. = .P;
  BFWSSI_CENTS&YR.&I. = .P;
%END;
%ELSE %DO;
IF MISSING(STWSSI&YR.&I) AND MISSING(BFWSSI&YR.&I)
  THEN BFWSSI_PUF&YR.&I. = .N;
ELSE IF MISSING(STWSSI&YR.&I) AND BFWSSI&YR.&I = 0
  THEN BFWSSI_PUF&YR.&I. = .N;
ELSE IF MISSING(STWSSI&YR.&I) THEN BFWSSI_PUF&YR.&I. = .N;
ELSE IF NOT MISSING(STWSSI&YR.&I) AND MISSING(BFWSSI&YR.&I)
  THEN BFWSSI_PUF&YR.&I. = 0;
ELSE IF STWSSI&YR.&I IN (4,8,9) THEN BFWSSI_PUF&YR.&I. = 0;
ELSE IF STWSSI&YR.&I IN (0,1,2,3)
  THEN BFWSSI_PUF&YR.&I. = BFWSSI&YR.&I.;
IF BFWSSI_PUF&YR.&I. > 0 THEN
  BFWSSI_CENTS&YR.&I. = SUM(BFWSSI_PUF&YR.&I.,RANDCENTS);
%END;
```

## Appendix A.291

JCL/SAS Code: PUFANN

```

* BFWCM;
* NOT AVAILABLE IN JAN-FEB 1994;
%IF &YR. = 94 AND (&Y.=1 OR &Y.=2) %THEN %DO;
  BFWCM_PUF&YR.&I. = .P;
  BFWCM_CENTS&YR.&I. = .P;
%END;
%ELSE %DO;
IF MISSING(STWCM&YR.&I) AND MISSING(BFWCM&YR.&I)
  THEN BFWCM_PUF&YR.&I. = .N;
ELSE IF MISSING(STWCM&YR.&I) AND BFWCM&YR.&I = 0
  THEN BFWCM_PUF&YR.&I. = .N;
ELSE IF MISSING(STWCM&YR.&I) THEN BFWCM_PUF&YR.&I. = .N;
ELSE IF NOT MISSING(STWCM&YR.&I) AND MISSING(BFWCM&YR.&I)
  THEN BFWCM_PUF&YR.&I. = 0;
ELSE IF STWCM&YR.&I IN (4,8,9) THEN BFWCM_PUF&YR.&I. = 0;
ELSE IF STWCM&YR.&I IN (0,1,2,3)
  THEN BFWCM_PUF&YR.&I. = BFWCM&YR.&I;
IF BFWCM_PUF&YR.&I. > 0 THEN
  BFWCM_CENTS&YR.&I. = SUM(BFWCM_PUF&YR.&I.,RANDCENTS);
%END;

* ADD LABELS TO ALL MONTHLY VARIABLES;
LABEL
  BFWCM_PUF&YR.&I. =
"COMBINED (SSDI/SSI) ESTIMATED BENEFITS FORGONE FOR WORK (YYMM = YEAR/MONTH)"
  BFWDI_PUF&YR.&I. =
  "SSDI ESTIMATED BENEFITS FORGONE FOR WORK (YYMM = YEAR/MONTH)"
  BFWSSI_PUF&YR.&I. =
  "SSI ESTIMATED BENEFITS FORGONE FOR WORK (YYMM = YEAR/MONTH)"
  CDR_PUF&YR.&I. =
  "SSDI CESSATION OF DISABILITY REASON (YYMM = YEAR/MONTH)"
  DIERNLVL_PUF&YR.&I. =
  "SSDI EARNING LEVEL INDICATOR (YYMM = YEAR/MONTH)"
  T2GRSAMT_PUF&YR.&I. =
  "SSDI EARNINGS GROSS AMOUNT (SSR) (YYMM = YEAR/MONTH)"
  DIRPAY_PUF&YR.&I. =
  "SSDI ACTUAL BENEFIT PAID AMOUNT (YYMM = YEAR/MONTH)"
  T2NETAMT_PUF&YR.&I. =
  "SSDI SELF-EMP NET INCOME AMOUNT (SSR) (YYMM = YEAR/MONTH)"
  DUED_PUF&YR.&I. = "SSDI BENEFIT DUE (YYMM = YEAR/MONTH)"
  DUES_PUF&YR.&I. = "SSI BENEFIT DUE (YYMM = YEAR/MONTH)"
  EICM_PUF&YR.&I. = "SSI COUNTABLE EARNED INCOME (YYMM = YEAR/MONTH)"
  LAF_PUF&YR.&I. =
"SSDI LEDGER ACCOUNT FILE (LAF)- PARTICIPATION STATUS (YYMM = YEAR/MONTH)"
  LIVF_PUF&YR.&I. = "SSI LIVING ARRANGEMENT CODE (YYMM = YEAR/MONTH)"

```

## Appendix A.291

JCL/SAS Code: PUFANN

```

MEDX_PUF&YR.&I. = "MEDICAL IMPROVEMENT CODE (YYMM = YEAR/MONTH)"
MEDE_PUF&YR.&I. = "SSI MEDICAID ELIGIBILITY (YYMM = YEAR/MONTH)"
MEDPREM_PUF&YR.&I. = "SSDI PHUS MEDICARE PREMIUM (YYMM = YEAR/MONTH)"
MEDR_PUF&YR.&I. = "SSDI MEDICARE ELIGIBILITY (YYMM = YEAR/MONTH)"
PAYD_PUF&YR.&I. = "SSDI PHUS BENEFIT PAID (YYMM = YEAR/MONTH)"
PAYO_PUF&YR.&I. =
    "SSDI PHUS DEPENDENT BENEFIT PAID (YYMM = YEAR/MONTH)"
PAYS_PUF&YR.&I. = "SSI BENEFIT PAID (YYMM = YEAR/MONTH)"
PSTA_PUF&YR.&I. = "SSI PAYMENT STATUS (PSTAT) (YYMM = YEAR/MONTH)"
SSICOUPL_PUF&YR.&I. = "SSI COUPLE INDICATOR (YYMM = YEAR/MONTH)"
/*
T16BEXPAMT_PUF&YR.&I.= "SSI BLIND IMPAIRMENT-RELATED WORK EXPENSE (BLIND IRWE
) (SSR) (YYMM = YEAR/MONTH)"
*/
T16GRSAMT_PUF&YR.&I. =
    "SSI EARNINGS GROSS AMOUNT (SSR) (YYMM = YEAR/MONTH)"
/*
T16EXLAMT_PUF&YR.&I.=
    "SSI STUDENT EXCLUSION AMOUNT (SSR) (YYMM = YEAR/MONTH)"
T16EXPAMT_PUF&YR.&I. =
    "SSI WORK EXPENSE AMOUNT (SSR) (YYMM = YEAR/MONTH)"
T16PASAMT_PUF&YR.&I. = "SSI PASS AMOUNT (SSR) (YYMM = YEAR/MONTH)"
*/
T16NETAMT_PUF&YR.&I. =
    "SSI SELF-EMP NET INCOME AMOUNT (SSR) (YYMM = YEAR/MONTH)"
/*
T16SEAMT_PUF&YR.&I. =
    "SSI SELF-EMPLOYMENT INCOME AMOUNT (SSR) (YYMM = YEAR/MONTH)"
*/
STWCM_PUF&YR.&I. =
"COMBINED (SSDI/SSI) SUSPENSE OR TERMINATION DUE TO WORK (YYMM = YEAR/MONTH)"
STWDI_PUF&YR.&I. =
    "SSDI SUSPENSE OR TERMINATION DUE TO WORK (YYMM = YEAR/MONTH)"
STWSSI_PUF&YR.&I. =
    "SSI SUSPENSE OR TERMINATION DUE TO WORK (YYMM = YEAR/MONTH)"
TWPDATA_PUF&YR.&I.= "SSDI TRIAL WORK PERIOD CODE (YYMM = YEAR/MONTH)"
UINC_PUF&YR.&I. = "SSI COUNTABLE UNEARNED INCOME (YYMM = YEAR/MONTH)"
;

%END; /* END MONTH LOOP */

* EDX NOT MONTHLY;
IF EDX&YR.12 NOT IN('','ZZ','99','1:','1L') THEN EDXINT = EDX&YR.12;
ELSE IF EDX&YR.11 NOT IN('','ZZ','99','1:','1L')
    THEN EDXINT = EDX&YR.11;

```

Appendix A.291

JCL/SAS Code: PUFANN

```

ELSE IF EDX&YR.10 NOT IN('','ZZ','99','1:','1L')
    THEN EDXINT = EDX&YR.10;
ELSE IF EDX&YR.09 NOT IN('','ZZ','99','1:','1L')
    THEN EDXINT = EDX&YR.09;
ELSE IF EDX&YR.08 NOT IN('','ZZ','99','1:','1L')
    THEN EDXINT = EDX&YR.08;
ELSE IF EDX&YR.07 NOT IN('','ZZ','99','1:','1L')
    THEN EDXINT = EDX&YR.07;
ELSE IF EDX&YR.06 NOT IN('','ZZ','99','1:','1L')
    THEN EDXINT = EDX&YR.06;
ELSE IF EDX&YR.05 NOT IN('','ZZ','99','1:','1L')
    THEN EDXINT = EDX&YR.05;
ELSE IF EDX&YR.04 NOT IN('','ZZ','99','1:','1L')
    THEN EDXINT = EDX&YR.04;
ELSE IF EDX&YR.03 NOT IN('','ZZ','99','1:','1L')
    THEN EDXINT = EDX&YR.03;
ELSE IF EDX&YR.02 NOT IN('','ZZ','99','1:','1L')
    THEN EDXINT = EDX&YR.02;
ELSE IF EDX&YR.01 NOT IN('','ZZ','99','1:','1L')
    THEN EDXINT = EDX&YR.01;
ELSE EDXINT = "";

IF EDXINT = "" THEN EDXINT = "";
EDXINTN = INPUT(EDXINT, 2.);

IF EDXINTN IN (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11)
    THEN EDX_PUF&YR. = 1;
ELSE IF EDXINTN = 12 THEN EDX_PUF&YR. = 2;
ELSE IF EDXINTN IN (13, 14, 15) THEN EDX_PUF&YR. = 3;
ELSE IF EDXINTN IN (16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26)
    THEN EDX_PUF&YR. = 4;
ELSE IF MISSING(EDXINTN) THEN EDX_PUF&YR. = .M;
ELSE IF NOT MISSING(EDXINTN) THEN EDX_PUF&YR. = .M;

LABEL EDX_PUF&YR. =
"EDUCATION LEVEL IN THE LAST MONTH OBSERVED DURING THE CALENDAR YEAR (YY=YEAR)"
;

* REGION NOT MONTHLY;
*NOTE: NOT AVAILABLE IN 1994-1995 AND JAN-FEB 1996;
%IF &YR.=94 OR &YR.=95 %THEN %DO;
    REGION_PUF&YR. = .P;
%END;
%ELSE %DO;
IF NOT MISSING(STATE&YR.12) AND STATE&YR.12 ~= '---'

```

## Appendix A.291

JCL/SAS Code: PUFANN

```

    THEN REG_T = STATE&YR.12;
  ELSE IF NOT MISSING(STATE&YR.11) AND STATE&YR.11 ^= '--'
    THEN REG_T = STATE&YR.11;
  ELSE IF NOT MISSING(STATE&YR.10) AND STATE&YR.10 ^= '--'
    THEN REG_T = STATE&YR.10;
  ELSE IF NOT MISSING(STATE&YR.09) AND STATE&YR.09 ^= '--'
    THEN REG_T = STATE&YR.09;
  ELSE IF NOT MISSING(STATE&YR.08) AND STATE&YR.08 ^= '--'
    THEN REG_T = STATE&YR.08;
  ELSE IF NOT MISSING(STATE&YR.07) AND STATE&YR.07 ^= '--'
    THEN REG_T = STATE&YR.07;
  ELSE IF NOT MISSING(STATE&YR.06) AND STATE&YR.06 ^= '--'
    THEN REG_T = STATE&YR.06;
  ELSE IF NOT MISSING(STATE&YR.05) AND STATE&YR.05 ^= '--'
    THEN REG_T = STATE&YR.05;
  ELSE IF NOT MISSING(STATE&YR.04) AND STATE&YR.04 ^= '--'
    THEN REG_T = STATE&YR.04;
  ELSE IF NOT MISSING(STATE&YR.03) AND STATE&YR.03 ^= '--'
    THEN REG_T = STATE&YR.03;
  %IF &YR. NE 96 %THEN %DO;
  ELSE IF NOT MISSING(STATE&YR.02) AND STATE&YR.02 ^= '--'
    THEN REG_T = STATE&YR.02;
  ELSE IF NOT MISSING(STATE&YR.01) AND STATE&YR.01 ^= '--'
    THEN REG_T = STATE&YR.01;
  %END;
  ELSE REG_T = "";

  * RECODE INTO REGIONS;
  IF REG_T IN ('CT','ME','MA','NH','RI','VT') THEN REGION_PUF&YR. = 1;
  ELSE IF REG_T IN ('NJ','NY','PR','VI') THEN REGION_PUF&YR. = 2;
  ELSE IF REG_T IN ('DC','DE','MD','PA','VA','WV') THEN REGION_PUF&YR. = 3;
  ELSE IF REG_T IN ('AL','FL','GA','KY','MS','NC','SC','TN')
    THEN REGION_PUF&YR. = 4;
  ELSE IF REG_T IN ('IL','IN','MI','MN','OH','WI') THEN REGION_PUF&YR. = 5;
  ELSE IF REG_T IN ('AR','LA','NM','OK','TX') THEN REGION_PUF&YR. = 6;
  ELSE IF REG_T IN ('IA','KS','MO','NE') THEN REGION_PUF&YR. = 7;
  ELSE IF REG_T IN ('CO','MT','ND','SD','UT','WY') THEN REGION_PUF&YR. = 8;
  ELSE IF REG_T IN ('AS','AZ','CA','GU','HI','NV','MP')
    THEN REGION_PUF&YR. = 9;
  ELSE IF REG_T IN ('AK','ID','OR','WA') THEN REGION_PUF&YR. = 10;
  ELSE REGION_PUF&YR. = .M;
  %END;

  LABEL REGION_PUF&YR. =
  "SSA REGION NUMBER IN THE LAST MONTH OBSERVED DURING THE CALENDAR YEAR

```

## Appendix A.291

JCL/SAS Code: PUFANN

```
(YY=YEAR)";
```

```
RUN;
```

```
* TOP CODE TOP .5 PERCENT OF DOLLAR VARIABLES ;
```

```
%DO Y= 1 %TO 12;
```

```
%LET I = %SYSFUNC(PUTN(&Y.,Z2));
```

```
%MACRO TC(VAR=);
```

```
%GLOBAL AVG&VAR.&YR.&I PCT&VAR.&YR.&I NUM&VAR.&YR.&I;
```

```
PROC SQL;
```

```
CREATE TABLE TEMP4.&VAR.&YR.&I AS
```

```
SELECT SUM(CASE WHEN NOT MISSING(&VAR._PUF&YR.&I.)
```

```
AND &VAR._PUF&YR.&I. NE 0 THEN 1 ELSE 0 END)
```

```
AS &VAR.&YR.&I._NUM
```

```
FROM TEMP1.ANN&YR.PUF;
```

```
QUIT;
```

```
DATA TEMP4.&VAR.&YR.&I;
```

```
SET TEMP4.&VAR.&YR.&I;
```

```
CALL SYMPUTX("NUM&VAR.&YR.&I.",PUT(&VAR.&YR.&I._NUM,BEST32.));
```

```
CALL SYMPUTX("PCT&VAR.&YR.&I.",.);
```

```
CALL SYMPUTX("AVG&VAR.&YR.&I.",.);
```

```
RUN;
```

```
%PUT TOPCODING NONZERO COUNT FOR &VAR.&YR.&I IS &&NUM&VAR.&YR.&I;
```

```
/*IF AT LEAST 6000 NONZERO CASES THEN FIND 99.5TH PERCENTILE*/
```

```
%IF &&NUM&VAR.&YR.&I GE 6000 %THEN %DO;
```

```
PROC UNIVARIATE DATA=TEMP1.ANN&YR.PUF NOPRINT;
```

```
WHERE NOT MISSING(&VAR._PUF&YR.&I) AND &VAR._PUF&YR.&I. NE 0;
```

```
VAR &VAR._PUF&YR.&I.;
```

```
OUTPUT OUT=TEMP2.&VAR.&YR.&I. PCTLPTS=99.5 PCTLPRE=&VAR.&YR.&I.;
```

```
RUN;
```

```
DATA TEMP2.&VAR.&YR.&I.;
```

```
SET TEMP2.&VAR.&YR.&I.;
```

```
CALL SYMPUTX("PCT&VAR.&YR.&I.",PUT(&VAR.&YR.&I.99_5,BEST32.));
```

```
RUN;
```

```
%PUT TOPCODING NUM FOR &VAR.&YR.&I. IS &&PCT&VAR.&YR.&I.;
```

```
PROC MEANS DATA=TEMP1.ANN&YR.PUF MEAN NOPRINT;
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
VAR &VAR._PUF&YR.&I.;
WHERE &VAR._PUF&YR.&I. >= &&PCT&VAR.&YR.&I
AND &VAR._PUF&YR.&I. NE 0;
OUTPUT OUT=TEMP3.&VAR.&YR.&I. MEAN=;
RUN;

DATA TEMP3.&VAR.&YR.&I;
SET TEMP3.&VAR.&YR.&I;
CALL SYMPUTX("AVG&VAR.&YR.&I.",PUT(&VAR._PUF&YR.&I.,BEST32.));
RUN;

%PUT TOPCODING AVERAGE FOR &VAR.&YR.&I IS &&AVG&VAR.&YR.&I;
%END;

/*IF 30 - 5999 NONZERO CASES THEN TOP CODE BASED ON TOP 30 CASES*/
%ELSE %IF 30 LE &&NUM&VAR.&YR.&I AND &&NUM&VAR.&YR.&I < 6000
%THEN %DO;

PROC SORT DATA = TEMP1.ANN&YR.PUF
      (KEEP = &VAR._PUF&YR.&I. &VAR._CENTS&YR.&I.)
      OUT = TEMP2.&VAR.&YR.&I.;
WHERE &VAR._PUF&YR.&I. > 0;
BY DESCENDING &VAR._CENTS&YR.&I.;
RUN;

DATA TEMP2.&VAR.&YR.&I.;
SET TEMP2.&VAR.&YR.&I.;
BY DESCENDING &VAR._CENTS&YR.&I.;
IF _N_ <= 30;
IF _N_ = 30
THEN CALL SYMPUTX("PCT&VAR.&YR.&I.",PUT(&VAR._CENTS&YR.&I.,BEST32.));
RUN;

%PUT TOPCODING NUM FOR &VAR.&YR.&I. IS &&PCT&VAR.&YR.&I;
/*
PROC PRINT DATA = TEMP2.&VAR.&YR.&I.;
FORMAT &VAR._CENTS&YR.&I. BEST32.;
RUN;
*/
PROC MEANS DATA=TEMP2.&VAR.&YR.&I. MEAN NOPRINT;
VAR &VAR._PUF&YR.&I.;
WHERE &VAR._PUF&YR.&I. NE 0;
OUTPUT OUT=TEMP3.&VAR.&YR.&I. MEAN=;
RUN;
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
DATA TEMP3.&VAR.&YR.&I;
  SET TEMP3.&VAR.&YR.&I;
  CALL SYMPUTX("AVG&VAR.&YR.&I.",PUT(&VAR._PUF&YR.&I.,BEST32.));
RUN;
```

```
%PUT TOPCODING AVERAGE FOR &VAR.&YR.&I IS &&AVG&VAR.&YR.&I;
%END;
```

```
/*IF LESS THAN 30 NONZERO CASES THEN SET ALL TO MISSING*/
%ELSE %IF &&NUM&VAR.&YR.&I < 30 %THEN %DO;
  %PUT TOPCODING NUM FOR &VAR.&YR.&I IS &&PCT&VAR.&YR.&I;
  %PUT TOPCODING AVERAGE FOR &VAR.&YR.&I IS &&AVG&VAR.&YR.&I;
%END;
```

```
%MEND TC;
%TC(VAR=DUED);
%TC(VAR=PAYD);
%TC(VAR=DIRPAY);
%TC(VAR=MEDPREM);
%TC(VAR=DUES);
%TC(VAR=PAYS);
%TC(VAR=T2GRSAMT);
%TC(VAR=T2NETAMT);
%TC(VAR=BFWDI);
%TC(VAR=T16GRSAMT);
%TC(VAR=T16NETAMT);
%TC(VAR=UINC);
%TC(VAR=EICM);
%TC(VAR=BFWSSI);
%TC(VAR=BFWCM);
*%TC(VAR=T16EXPAMT);
*%TC(VAR=T16PASAMT);
*%TC(VAR=T16SEAMT);
*%TC(VAR=T16EXLAMT);
*%TC(VAR=T16BEXPAMT);
%TC(VAR=PAYO);
```

```
%END;
```

```
PROC CONTENTS DATA=TEMP1.ANN&YR.PUF
  OUT=TEMP3.CONTENTS(KEEP=NAME) NOPRINT ;
RUN;
```

```
PROC SQL NOPRINT;
```



## Appendix A.291

JCL/SAS Code: PUFANN

```
SELECT NAME INTO :KEEPVARS SEPARATED BY ' '
FROM TEMP3.CONTENTES
WHERE UPCASE(NAME) LIKE '%^_PUF%' ESCAPE '^' ;
QUIT;

DATA OUT1.ANN&YR.PUF (KEEP= PUFPIN &KEEPVARS);
SET TEMP1.ANN&YR.PUF;

%DO Y= 1 %TO 12;
%LET I = %SYSFUNC(PUTN(&Y.,Z2));

%MACRO SETTC(VAR=);

&VAR.&YR.&I._T = &&AVG&VAR.&YR.&I;

/*IF LESS THAN 30 NONZERO CASES THEN SET ALL TO MISSING;
*IF VARIABLE CODED AS .P (NOT AVAILABLE) THEN LEAVE AS .P;*/
%IF &&NUM&VAR.&YR.&I < 30 %THEN %DO;
  IF &VAR._PUF&YR.&I. NE .P THEN &VAR._PUF&YR.&I. = .X;
%END;

/*IF 30 TO 5999 NONZERO CASES THEN TOP CODE USING CENTSYYMM;*/
%ELSE %IF 30 <= &&NUM&VAR.&YR.&I
AND &&NUM&VAR.&YR.&I < 6000 %THEN %DO;
  IF NOT MISSING(&VAR._PUF&YR.&I.) AND
  &VAR._PUF&YR.&I. NE 0 THEN DO;
    IF PUT(&VAR._CENTS&YR.&I.,BEST32.) >= &&PCT&VAR.&YR.&I
    THEN &VAR._PUF&YR.&I. = &VAR.&YR.&I._T;
  END;
%END;

/* IF 6000 OR MORE NONZERO CASES THEN TOP CODE NORMALLY;*/
%ELSE %IF &&NUM&VAR.&YR.&I GE 6000 %THEN %DO;
  IF NOT MISSING(&VAR._PUF&YR.&I.) AND
  &VAR._PUF&YR.&I. NE 0 THEN DO;
    IF PUT(&VAR._PUF&YR.&I.,BEST32.) >= &&PCT&VAR.&YR.&I
    THEN &VAR._PUF&YR.&I. = &VAR.&YR.&I._T;
  END;
%END;

DROP &VAR.&YR.&I._T;

%MEND SETTC;
%SETTC(VAR=DUED);
%SETTC(VAR=PAYD);
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
%SETTC(VAR=DIRPAY);
%SETTC(VAR=MEDPREM);
%SETTC(VAR=DUES);
%SETTC(VAR=PAYS);
%SETTC(VAR=T2GRSAMT);
%SETTC(VAR=T2NETAMT);
%SETTC(VAR=BFWDI);
%SETTC(VAR=T16GRSAMT);
%SETTC(VAR=T16NETAMT);
%SETTC(VAR=UINC);
%SETTC(VAR=EICM);
%SETTC(VAR=BFWSSI);
%SETTC(VAR=BFWCM);
*%SETTC(VAR=T16EXPAMT);
*%SETTC(VAR=T16PASAMT);
*%SETTC(VAR=T16SEAMT);
*%SETTC(VAR=T16EXLAMT);
*%SETTC(VAR=T16BEXPAMT);
%SETTC(VAR=PAYO);

* SET UP DOLLAR MACRO TO ROUND ACCORDING TO CENSUS STANDARDS;
%MACRO DOLLAMT (VAR=);
  IF &VAR._PUF&YR.&I. > 0 AND &VAR._PUF&YR.&I. <8 THEN
    &VAR._PUF&YR.&I. = 4;
  ELSE IF &VAR._PUF&YR.&I.>=8 AND &VAR._PUF&YR.&I. <1000 THEN
    &VAR._PUF&YR.&I. = ROUND(&VAR._PUF&YR.&I.,10);
  ELSE IF &VAR._PUF&YR.&I.>=1000 AND &VAR._PUF&YR.&I.<50000 THEN
    &VAR._PUF&YR.&I. = ROUND(&VAR._PUF&YR.&I.,100);
  ELSE IF &VAR._PUF&YR.&I.>=50000 THEN
    &VAR._PUF&YR.&I. = ROUND(&VAR._PUF&YR.&I.,1000);
  ELSE IF &VAR._PUF&YR.&I. = 0 THEN &VAR._PUF&YR.&I. = 0;

  ELSE IF &VAR._PUF&YR.&I. < 0 AND &VAR._PUF&YR.&I. >-8 THEN
    &VAR._PUF&YR.&I. = -4;
  ELSE IF &VAR._PUF&YR.&I.<=-8 AND &VAR._PUF&YR.&I.>-1000 THEN
    &VAR._PUF&YR.&I. = ROUND(&VAR._PUF&YR.&I.,10);
  ELSE IF &VAR._PUF&YR.&I.<=-1000 AND &VAR._PUF&YR.&I.>-50000 THEN
    &VAR._PUF&YR.&I. = ROUND(&VAR._PUF&YR.&I.,100);
  ELSE IF &VAR._PUF&YR.&I.<=-50000 AND NOT MISSING(&VAR._PUF&YR.&I.)
    THEN &VAR._PUF&YR.&I. = ROUND(&VAR._PUF&YR.&I.,1000);
%MEND DOLLAMT;

%DOLLAMT(VAR=DUED);
%DOLLAMT(VAR=PAYD);
%DOLLAMT(VAR=DIRPAY);
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
%DOLLAMT(VAR=MEDPREM);
%DOLLAMT(VAR=DUES);
%DOLLAMT(VAR=PAYS);
%DOLLAMT(VAR=T2GRSAMT)
%DOLLAMT(VAR=T2NETAMT);
%DOLLAMT(VAR=BFWDI);
%DOLLAMT(VAR=T16GRSAMT);
%DOLLAMT(VAR=T16NETAMT);
%DOLLAMT(VAR=UINC);
%DOLLAMT(VAR=EICM);
%DOLLAMT(VAR=BFWSSI);
%DOLLAMT(VAR=BFWCM);
*%DOLLAMT(VAR=T16EXPAMT);
*%DOLLAMT(VAR=T16PASAMT);
*%DOLLAMT(VAR=T16SEAMT);
*%DOLLAMT(VAR=T16EXLAMT);
*%DOLLAMT(VAR=T16BEXPAMT);
%DOLLAMT(VAR=PAYO);
%END;
RUN;
```

```
* AJB 12.29.17 COMMENT OUT COUNTS OF ORIGINAL VARIABLES;
* COMMENT OUT SO THAT WE CAN INCLUDE THEM IN FOR OTHER YRS AS NEEDED;
* COUNTS OF ORIGINAL VARIABLES;
```

```
PROC FORMAT;
VALUE $PRESENT ' '= 'MISSING' OTHER='NONMISSING';
VALUE PRESENTN . = 'MISSING' OTHER='NONMISSING';
VALUE $LAFC 'C'='C' OTHER='OTHER';
VALUE $PSTAC 'C01'='C01' OTHER='OTHER';
VALUE STWSSIC 0-4='0-4' OTHER='OTHER';
VALUE STWDIC 0-3='0-3' OTHER='OTHER';
VALUE STWCMC 0-3='0-3' OTHER='OTHER';
VALUE T16AMT . = 'MISSING' 0='0' OTHER='NONZERO';
RUN;
```

```
/* RAW FREQS TO DETERMINE PROCESSING. COMMENT OUT WHEN NOT NEEDED*/
```

```
/*
```

```
%DO Y= 1 %TO 12;
%LET I = %SYSFUNC(PUTN(&Y.,Z2));
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
  TABLES CDR&YR.&I / LIST MISSING;
RUN;
```

```
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
```

## Appendix A.291

JCL/SAS Code: PUFANN

```
TABLES LAF&YR.&I PSTA&YR.&I STWSSI&YR.&I STWDI&YR.&I STWCM&YR.&I /MISSING;
FORMAT LAF&YR.&I PSTA&YR.&I $PRESENT.
    STWSSI&YR.&I STWDI&YR.&I STWCM&YR.&I PRESENTN.;
RUN;
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
    TABLES LAF&YR.&I / MISSING;
    FORMAT LAF&YR.&I $LAFC.;
RUN;
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
    TABLES PSTA&YR.&I / MISSING;
    FORMAT PSTA&YR.&I $PSTAC.;
RUN;
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
    TABLES STWSSI&YR.&I / MISSING;
    FORMAT STWSSI&YR.&I STWSSIC.;
RUN;
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
    TABLES STWDI&YR.&I / MISSING;
    FORMAT STWDI&YR.&I STWDIC.;
RUN;
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
    TABLES STWCM&YR.&I / MISSING;
    FORMAT STWCM&YR.&I STWCMC.;
RUN;
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
    TABLES T16EXPAMT&YR.&I / MISSING LIST;
    FORMAT T16EXPAMT&YR.&I T16AMT.;
RUN;
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
    TABLES T16PASAMT&YR.&I / MISSING LIST;
    FORMAT T16PASAMT&YR.&I T16AMT.;
RUN;
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
    TABLES T16SEAMT&YR.&I / MISSING LIST;
    FORMAT T16SEAMT&YR.&I T16AMT.;
RUN;
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
    TABLES T16EXLAMT&YR.&I / MISSING LIST;
    FORMAT T16EXLAMT&YR.&I T16AMT.;
RUN;
PROC FREQ DATA=TEMP1.ANN&YR.PUF;
    TABLES T16BEXPAMT&YR.&I / MISSING LIST;
    FORMAT T16BEXPAMT&YR.&I T16AMT.;
RUN;
%END;
```

Appendix A.291

JCL/SAS Code: PUFANN

```
*/

* FREQUENCIES IN ALPHABETICAL ORDER;
%MACRO MFREQ(VAR=);

%DO Y= 1 %TO 12;
%LET I = %SYSFUNC(PUTN(&Y.,Z2));

PROC FREQ DATA=OUT1.ANN&YR.PUF;
  TABLES &VAR._PUF&YR.&I. / MISSING LIST;
RUN;

%END;
%MEND MFREQ;

%MFREQ(VAR=BFWCM);
%MFREQ(VAR=BFWDI);
%MFREQ(VAR=BFWSSI);
%MFREQ(VAR=CDR);
%MFREQ(VAR=DIERNLVL);
%MFREQ(VAR=DIRPAY);
%MFREQ(VAR=DUED);
%MFREQ(VAR=DUES);

PROC FREQ DATA=OUT1.ANN&YR.PUF;
  TABLE EDX_PUF&YR. / MISSING LIST;
RUN;

%MFREQ(VAR=EICM);
%MFREQ(VAR=LAF);
%MFREQ(VAR=LIVF);
%MFREQ(VAR=MEDX);
%MFREQ(VAR=MEDE);
%MFREQ(VAR=MEDPREM);
%MFREQ(VAR=MEDR);
%MFREQ(VAR=PAYD);
%MFREQ(VAR=PAYO);
%MFREQ(VAR=PAYS);

PROC FREQ DATA=OUT1.ANN&YR.PUF;
  TABLES PROA_PUF&YR. PROB_PUF&YR. / MISSING LIST;
RUN;

%MFREQ(VAR=PSTA);
```

Appendix A.291

JCL/SAS Code: PUFANN

```
PROC FREQ DATA=OUT1.ANN&YR.PUF;
  TABLES REGION_PUF&YR. / MISSING LIST;
RUN;

%MFREQ(VAR=SSICOUPL);
%MFREQ(VAR=STWCM);
%MFREQ(VAR=STWDI);
%MFREQ(VAR=STWSSI);
%MFREQ(VAR=TWPDATA);
  *%MFREQ(VAR=T16BEXPAMT); *7.07.20 NOT DRB REVIEWED YET;
  *%MFREQ(VAR=T16EXLAMT); *7.07.20 NOT DRB REVIEWED YET;
  *%MFREQ(VAR=T16EXPAMT); *7.07.20 NOT ENOUGH NONZERO VALUES FOR DRB;
%MFREQ(VAR=T16GRSAMT);
%MFREQ(VAR=T16NETAMT);
  *%MFREQ(VAR=T16PASAMT); *07.07.20 NOT ENOUGH NONZERO VALUES FOR DRB;
  *%MFREQ(VAR=T16SEAMT); *07.07.20 NOT DRB REVIEWED YET;
%MFREQ(VAR=T2GRSAMT);
%MFREQ(VAR=T2NETAMT);
%MFREQ(VAR=UINC);

%MEND DOIT;
%DOIT;
```

## Appendix A.292

JCL/SAS Code: REGNMASK

```
//#8947RGN JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.#6502.DAF20.PUF.PRDLIB(REGIONQA)
//*  *-----*
//*
//* CREATE PROC CONTENTS OF PUF FILES
//* CONTACT INFO: ALEX BRYCE, ABRYCE@MATHEMATICA-MPR.COM,
//*
//*****
//*
//JS030 EXEC SAS9,
//  WORK='120000,60000'          * SPACE IN BLKS *
//*
//* INPUT FILES
//AN94 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN94.SA.V1,DISP=SHR
//AN95 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN95.SA.V1,DISP=SHR
//AN96 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN96.SA.V1,DISP=SHR
//AN97 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN97.SA.V1,DISP=SHR
//AN98 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN98.SA.V1,DISP=SHR
//AN99 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN99.SA.V1,DISP=SHR
//AN00 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN00.SA.V1,DISP=SHR
//AN01 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN01.SA.V1,DISP=SHR
//AN02 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN02.SA.V1,DISP=SHR
//AN03 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN03.SA.V1,DISP=SHR
//AN04 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN04.SA.V1,DISP=SHR
//AN05 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN05.SA.V1,DISP=SHR
//AN06 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN06.SA.V1,DISP=SHR
//AN07 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN07.SA.V1,DISP=SHR
//AN08 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN08.SA.V1,DISP=SHR
//AN09 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN09.SA.V1,DISP=SHR
//AN10 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN10.SA.V1,DISP=SHR
//AN11 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN11.SA.V1,DISP=SHR
//AN12 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN12.SA.V1,DISP=SHR
//AN13 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN13.SA.V1,DISP=SHR
//AN14 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN14.SA.V1,DISP=SHR
//AN15 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN15.SA.V1,DISP=SHR
//AN16 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN16.SA.V1,DISP=SHR
//AN17 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN17.SA.V1,DISP=SHR
//AN18 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN18.SA.V1,DISP=SHR
//AN19 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN19.SA.V1,DISP=SHR
//AN20 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN20.SA.V1,DISP=SHR
```

## Appendix A.292

JCL/SAS Code: REGNMASK

```
//AN21 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.PUF.AN21.SA.V1,DISP=SHR
//* TEMP FILES
//TP94 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP95 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP96 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP97 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP98 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP99 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP00 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP01 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP02 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP03 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP04 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP05 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP06 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP07 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP08 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP09 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP10 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP11 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP12 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP13 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP14 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
//   VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP15 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
```



## Appendix A.292

JCL/SAS Code: REGNMASK

```
// VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP16 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
// VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP17 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
// VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP18 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
// VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP19 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
// VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP20 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
// VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TP21 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
// VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),
// VOL=(,,10),SPACE=(CYL,(3000,3000),RLSE)
//* OUTPUT FILES
//AN94F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN94,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN95F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN95,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN96F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN96,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN97F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN97,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN98F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN98,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN99F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN99,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN00F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN00,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN01F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN01,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN02F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN02,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN03F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN03,
// DISP=(OLD,CATLG,DELETE),VOL=(,,10),
```

## Appendix A.292

JCL/SAS Code: REGNMASK

```
// SPACE=(CYL,(3000,3000),RLSE)
//AN04F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN04,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN05F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN05,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN06F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN06,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN07F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN07,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN08F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN08,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN09F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN09,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN10F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN10,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN11F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN11,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN12F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN12,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN13F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN13,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN14F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN14,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN15F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN15,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN16F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN16,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN17F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN17,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN18F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN18,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
```

## Appendix A.292

JCL/SAS Code: REGNMASK

```
// SPACE=(CYL,(3000,3000),RLSE)
//AN19F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN19,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN20F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN20,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//AN21F DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.PUF.AN21,
// DISP=(OLD,CATLG,DELETE),VOL=(,,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=BINARY OBS = MAX MPRINT
MACROGEN;

TITLE "QA ON REGIONYY_PUF";

%MACRO DOIT;

%LET BEGYR = 1994;
%LET ENDYR = 2021;

%DO YEAR=&BEGYR. %TO &ENDYR.;
%LET YR = %SUBSTR(&YEAR.,3,2);
PROC SORT DATA=AN&YR..ANN&YR.PUF
  OUT = TP&YR..ANN&YR.PUF;
  BY PUFFPIN;
  RUN;
%END;

DATA TEMP.REGIONQA;
MERGE
%DO YEAR=&BEGYR. %TO &ENDYR.;
%LET YR = %SUBSTR(&YEAR.,3,2);
TP&YR..ANN&YR.PUF (IN=INAN&YR. KEEP = PUFFPIN REGION_PUF&YR.)
%END;;
BY PUFFPIN;

ARRAY REGION(&BEGYR.:&ENDYR.)
%DO YEAR=&BEGYR. %TO &ENDYR.;
%LET YR = %SUBSTR(&YEAR.,3,2);
REGION_PUF&YR.
%END;;
```

## Appendix A.292

JCL/SAS Code: REGNMASK

```
ARRAY MOVE(1995:&ENDYR.) $
  MOVE9495 MOVE9596 MOVE9697 MOVE9798 MOVE9899 MOVE9900 MOVE0001
  MOVE0102 MOVE0203 MOVE0304 MOVE0405 MOVE0506 MOVE0607 MOVE0708
  MOVE0809 MOVE0910 MOVE1011 MOVE1112 MOVE1213 MOVE1314 MOVE1415
  MOVE1516 MOVE1617 MOVE1718 MOVE1819 MOVE1920 MOVE2021;

CNTMOVE = 0;

*CONVERT .M TO . FOR EASIER TRACKING;
DO I=&BEGYR. TO &ENDYR.;
IF MISSING(REGION(I)) THEN REGION(I) = .;
END;

*INITIALIZE ALLYRS. ALWAYS MISSING IN 1994;
LENGTH ALLYRS $50;
ALLYRS = '.';

DO I = 1995 TO &ENDYR.;
  J = I - 1;
  IF NOT MISSING(REGION(I)) AND NOT MISSING(REGION(J))
  AND REGION(I) NE REGION(J) THEN DO;
    CNTMOVE = CNTMOVE + 1;
    MOVE(I) = CATS(REGION(J),REGION(I));
  END;

  IF MISSING(REGION(I)) THEN ALLYRS=CATS(ALLYRS,'. ');
  ELSE ALLYRS=CATS(ALLYRS,REGION(I));
END;
ALLYRS=TRIM(ALLYRS);
RUN;

PROC FREQ DATA=TEMP.REGIONQA;
TITLE2 'FREQ OF MOVES';
TABLES CNTMOVE / LIST MISSING;
RUN;
/*
%DO YEAR=1995 %TO &ENDYR.;
  %LET YR = %SUBSTR(&YEAR.,3,2);
  %LET YEARM1 = %EVAL(&YEAR - 1);
  %LET YRM1 = %SUBSTR(&YEARM1.,3,2);

  PROC FREQ DATA=TEMP.REGIONQA;
  TITLE2 "TRACKING MOVERS IN &YRM1.&YR.";
  WHERE CNTMOVE GE 2 AND NOT MISSING(MOVE&YRM1.&YR.);
  TABLES CNTMOVE * MOVE&YRM1.&YR. / LIST MISSING;
```

## Appendix A.292

JCL/SAS Code: REGNMASK

```
RUN;  
%END;
```

```
PROC FREQ DATA=TEMP.REGIONQA;  
TITLE2 'TRACKING ALL MOVES FOR BENES WITH 2+ CNTMOVE';  
WHERE CNTMOVE GE 2;  
TABLES CNTMOVE * ALLYRS / LIST MISSING;  
RUN;  
*/
```

```
*GET PUFFINS WHO MOVED 3+ TIMES;  
PROC SORT DATA = TEMP.REGIONQA  
OUT = TEMP.REGIONQA_LIST (KEEP = PUFFIN CNTMOVE);  
WHERE CNTMOVE > 2;  
BY PUFFIN;  
RUN;
```

```
PROC FREQ DATA = TEMP.REGIONQA_LIST;  
TITLE2 "FREQ OF MOVERS TO RECODE";  
TABLES CNTMOVE / LIST MISSING;  
RUN;
```

```
%DO YEAR=&BEGYR. %TO &ENDYR.;  
%LET YR = %SUBSTR(&YEAR.,3,2);
```

```
DATA TP&YR..ANN&YR.PUF;  
MERGE TP&YR..ANN&YR.PUF (IN=INANN)  
TEMP.REGIONQA_LIST (IN=INMOVERS);  
BY PUFFIN;  
IF INANN;  
IN_MOVERS = INMOVERS;  
RUN;
```

```
PROC FREQ DATA = TP&YR..ANN&YR.PUF;  
TITLE2 "BEFORE FREQ ON REGION FOR CNTMOVE < 3 FOR ANNUALS &YR.";  
WHERE CNTMOVE < 3;  
TABLES REGION_PUF&YR. / LIST MISSING;  
RUN;
```

```
PROC FREQ DATA = TP&YR..ANN&YR.PUF;  
TITLE2 "BEFORE FREQ ON REGION FOR CNTMOVE 3+ FOR ANNUALS &YR.";  
WHERE CNTMOVE > 2;  
TABLES REGION_PUF&YR. / LIST MISSING;  
RUN;
```

## Appendix A.292

JCL/SAS Code: REGNMASK

```
DATA TP&YR..ANN&YR.PUF;
SET TP&YR..ANN&YR.PUF;

RANDNUM = RANUNI(1100);
IF IN_MOVERS AND NOT MISSING(REGION_PUF&YR.) THEN DO;
  IF REGION_PUF&YR. IN (1,2,3) THEN DO;
    IF 0<=RANDNUM<.33 THEN REGION_PUF&YR.=1;
    ELSE IF .33<=RANDNUM<.66 THEN REGION_PUF&YR.=2;
    ELSE IF .66<=RANDNUM THEN REGION_PUF&YR.=3;
  END;

  IF REGION_PUF&YR. IN (4,6) THEN DO;
    IF 0<=RANDNUM<.50 THEN REGION_PUF&YR.=4;
    ELSE IF .50<=RANDNUM THEN REGION_PUF&YR.=6;
  END;

  IF REGION_PUF&YR. IN (5,7) THEN DO;
    IF 0<=RANDNUM<.50 THEN REGION_PUF&YR.=5;
    ELSE IF .50<=RANDNUM THEN REGION_PUF&YR.=7;
  END;

  IF REGION_PUF&YR. IN (8,9,10) THEN DO;
    IF 0<=RANDNUM<.33 THEN REGION_PUF&YR.=8;
    ELSE IF .33<=RANDNUM<.66 THEN REGION_PUF&YR.=9;
    ELSE IF .66<=RANDNUM THEN REGION_PUF&YR.=10;
  END;
END;

LABEL REGION_PUF&YR. = "SSA REGION NUMBER IN THE LAST MONTH OBSERVED DURING THE
CALENDAR YEAR (YY = YEAR)"
;

RUN;

PROC FREQ DATA = TP&YR..ANN&YR.PUF;
TITLE2 "AFTER FREQ ON REGION FOR CNTMOVE < 3 FOR ANNUALS &YR.";
WHERE CNTMOVE < 3;
TABLES REGION_PUF&YR. / LIST MISSING;
RUN;

PROC FREQ DATA = TP&YR..ANN&YR.PUF;
TITLE2 "AFTER FREQ ON REGION FOR CNTMOVE 3+ FOR ANNUALS &YR.";
WHERE CNTMOVE > 2;
TABLES REGION_PUF&YR. / LIST MISSING;
RUN;
```

## Appendix A.292

JCL/SAS Code: REGNMASK

```
/*OUTPUT FINAL FILE AND DROP VARIABLES AS NEEDED*/
DATA AN&YR.F.ANN&YR.PUF
  (DROP = CNTMOVE RANDNUM IN_MOVERS
    /*T16PASAMT: T16EXPAMT: T16EXLAMT: T16SEAMT: T16BEXPAMT:*/);
SET TP&YR..ANN&YR.PUF;
BY PUFPIN;
RUN;

PROC CONTENTS DATA = AN&YR.F.ANN&YR.PUF;
TITLE2 "PROC CONTENTS FOR ANNUALS &YR.";
RUN;

PROC FREQ DATA = AN&YR..ANN&YR.PUF;
TITLE2 "FREQ ON REGION BEFORE QA FOR ANNUALS &YR.";
TABLES REGION_PUF&YR. / LIST MISSING;
RUN;

PROC FREQ DATA = AN&YR.F.ANN&YR.PUF;
TITLE2 "FREQ ON REGION AFTER QA FOR ANNUALS &YR.";
TABLES REGION_PUF&YR. / LIST MISSING;
RUN;

%END;
%MEND DOIT;
%DOIT;

//*****
/* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: DAF21 PUF REGIONQA
DAF21 PUF REGIONQA HAS FINISHED RUNNING
```

## Appendix A.293

JCL/SAS Code: DBAEXTJ

```
//#6266DBA JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.FINDER.PRDLIB(DBADJCL)
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   2/23/2021 SERGE LUKASHANETS FOR DAF20
//* PHONE:
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//FINDPROC PROC
//FIND   EXEC SAS9,
//      WORK='120000,60000',PARM='SYSPARM="'&Y.&M"'
//*
//IN    DD DSN=MTOSI.T2.DBADM BR.D&Y.&M,DISP=SHR
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,(VOL=,,,5)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.HDBD&Y.&M..SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20.HISTDBD.PRDLIB(DBAEXTJ),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//FINYY  PROC
//FRUN01 EXEC FINDPROC,M=01
//FRUN02 EXEC FINDPROC,M=02
//FRUN03 EXEC FINDPROC,M=03
//FRUN04 EXEC FINDPROC,M=04
//FRUN05 EXEC FINDPROC,M=05
//FRUN06 EXEC FINDPROC,M=06
//FRUN07 EXEC FINDPROC,M=07
//FRUN08 EXEC FINDPROC,M=08
//FRUN09 EXEC FINDPROC,M=09
//FRUN10 EXEC FINDPROC,M=10
//FRUN11 EXEC FINDPROC,M=11
//FRUN12 EXEC FINDPROC,M=12
//      PEND
//*
//ARUN11 EXEC FINYY,Y=11
```



Appendix A.293

JCL/SAS Code: DBAEXTJ

```
//ARUN12 EXEC FINYY,Y=12
//ARUN13 EXEC FINYY,Y=13
//ARUN14 EXEC FINYY,Y=14
//ARUN15 EXEC FINYY,Y=15
//ARUN16 EXEC FINYY,Y=16
//ARUN17 EXEC FINYY,Y=17
//ARUN18 EXEC FINYY,Y=18
//ARUN19 EXEC FINYY,Y=19
//*
```

## Appendix A.294

JCL/SAS Code: DBDEXTJ3

```
//#6266DBA JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.FINDER.PRDLIB(DBADJCL)
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   2/23/2021 SERGE LUKASHANETS FOR DAF20
//* PHONE:
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//FINDPROC PROC
//FIND   EXEC SAS9,
//      WORK='120000,60000',PARM='SYSPARM="'&Y.&M"'
//*
//IN     DD DSN=MTOSI.T2.DBADM BR.D&Y.&M,DISP=SHR
//TEMP   DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,(VOL=,,,5)
//OUT    DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.HDBD&Y.&M..SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20.HISTDBD.PRDLIB(DBADEXT),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//FINYY  PROC
//FRUN01 EXEC FINDPROC,M=01
//FRUN02 EXEC FINDPROC,M=02
//FRUN03 EXEC FINDPROC,M=03
//FRUN04 EXEC FINDPROC,M=04
//FRUN05 EXEC FINDPROC,M=05
//FRUN06 EXEC FINDPROC,M=06
//FRUN07 EXEC FINDPROC,M=07
//FRUN08 EXEC FINDPROC,M=08
//FRUN09 EXEC FINDPROC,M=09
//FRUN10 EXEC FINDPROC,M=10
//FRUN11 EXEC FINDPROC,M=11
//FRUN12 EXEC FINDPROC,M=12
//      PEND
//*
//ARUN09 EXEC FINYY,Y=09
```

Appendix A.294

JCL/SAS Code: DBDEXTJ3

```
//ARUN10 EXEC FINYY,Y=10  
//*
```

## Appendix A.295

JCL/SAS Code: DBADEXTO

```
//#6266DBA JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.FINDER.PRDLIB(DBADJCL)
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   2/23/2021 SERGE LUKASHANETS FOR DAF20
//* PHONE:
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//FINDPROC PROC
//FIND   EXEC SAS9,
//      WORK='120000,60000',PARM='SYSPARM="'&Y.&M"'
//*
//IN    DD DSN=MTOSI.T2.DBADMBR.D&Y.&M.B,DISP=SHR
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,(VOL=,,,5)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.HDB&Y.&M.B.SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20.HISTDBD.PRDLIB(DBADEXT),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//FINYY  PROC
//FRUN01 EXEC FINDPROC,M=01
//      PEND
//*
//ARUN09 EXEC FINYY,Y=10
//*
```

## Appendix A.296

JCL/SAS Code: DBADEXT

```
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES OBS=MAX;
%LET YYMM=%SUBSTR(&SYSPARM,1,4);

%MACRO _CAN;
%IF &YYMM<1206 %THEN %DO;
%GLOBAL A; %LET A = 09;
%END;
%ELSE %DO;
%GLOBAL A; %LET A = 10;
%END;
%MEND;
%_CAN

%PUT "CAN STARTS FROM &A.";

DATA TEMP.DBAD;
  INFILE IN TRUNCOVER;
  INPUT

  @00&A. CAN    $9.    /* CLAIM ACCOUNT NUMBER (SSN) */
  @0376 BIC    $2.    /* BENE IDENTIFICATION CODE */
  @1113 BOAN    $9.    /* BENEFICIARY OWN ACCOUNT NUMBER */

  @3321 ZIP_DI  $5.    /* ZIP CODE */
  @3326 ZIP_DIADD $4.  /* ZIP ADD ON CODE */
  ;

IF SUBSTR(BIC,1,1) = 'A' THEN SSN = CAN;
ELSE SSN = BOAN;

IF SSN="" OR SSN="000000000" THEN DELETE;

RUN;

PROC SORT DATA=TEMP.DBAD OUT=OUT.DBAD NODUPKEY; BY SSN BIC; RUN;

PROC CONTENTS DATA=OUT.DBAD;
RUN;

/* CHECK DATA */
```

Appendix A.296

JCL/SAS Code: DBADEXT

```
TITLE "CHECK OUTPUT DATASET (OBS=20)";  
PROC PRINT DATA=OUT.DBAD(OBS=20);  
RUN;  
TITLE;
```

## Appendix A.297

JCL/SAS Code: DBDCMBJ

```
//#6266DBA JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.FINDER.PRDLIB(DBADJCL)
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   2/23/2021 SERGE LUKASHANETS FOR DAF20
//* PHONE:
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//FINDPROC PROC
//FIND   EXEC SAS9,
//      WORK='120000,60000',PARM='SYSPARM="'&Y.'"
//*
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,(VOL=,,,5)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.HDBDCM&Y..SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20.HISTDBD.PRDLIB(DBADEXTC),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//FRUN01 EXEC FINDPROC,Y=09
//FRUN02 EXEC FINDPROC,Y=10
//FRUN03 EXEC FINDPROC,Y=11
//FRUN04 EXEC FINDPROC,Y=12
//FRUN05 EXEC FINDPROC,Y=13
//FRUN06 EXEC FINDPROC,Y=14
//FRUN07 EXEC FINDPROC,Y=15
//FRUN08 EXEC FINDPROC,Y=16
//FRUN09 EXEC FINDPROC,Y=17
//FRUN10 EXEC FINDPROC,Y=18
//FRUN11 EXEC FINDPROC,Y=19
//*
```

Appendix A.298

JCL/SAS Code: DBADEXTC

```
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES MPRINT MACROGEN;
%LET YR=%SUBSTR(&SYSPARM,1,2);

%LET BEGYR=20&YR.;
%LET ENDYR=20&YR.;

%MACRO START;

/* INCLUDE YYMM MACRO PROGRAM */

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF20.UTILITY.PRDLIB(YYMMMCR)";

/* INPUT FILES */
%DO MM = 1 %TO &TOTMOS.;

LIBNAME IN&MM. "OPDR.TG.PRD.ETTW.#6266.DAF20P.HDBD&&YY&MM...SA.V1"
        DISP=SHR;
%END;

/* COMBINE 12 DBAD FILES INTO ONE DBAD FILE */
DATA OUT.DBAD;
MERGE
%DO MM = 1 %TO &TOTMOS.;
IN&MM..DBAD(RENAME=(ZIP_DI=ZIP_DI&&YY&MM
        ZIP_DIADD=ZIP_DIADD&&YY&MM))
%END;
;
BY SSN BIC;

RUN;

PROC CONTENTS DATA = OUT.DBAD;
RUN;

%MEND START;

%START;
```



## Appendix A.299

JCL/SAS Code: DBADEXTF

```

//#6266DBA JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=&SYSUID
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#7429.DAF19.FINDER.PRDLIB(DBADCOMB)
//* PURPOSE:   COMBINE ALL THE DBAD FILES FOR 2019
//*           CASES IN DBAD FROM JAN 2019 - DEC 2019
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   BRITTANY COLIP 01/31/2020 FOR DAF19
//* E-MAIL:    BCOLIP@MATHEMATICA-MPR.COM
//*           BRITTANY.COLIP@SSA.GOV
//*****
//*
// EXEC SAS9,
//   WORK='120000,60000'
//*
//**SEE INPUT FILES IN LIBNAME STATEMENTS BELOW
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.CMBDBDH.SA.V1,
//   DISP=(MOD,CATLG,DELETE),
//   UNIT=TSILO
//SYSIN DD *

```

```

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES MPRINT MACROGEN;

```

```

%MACRO START;

```

```

/* INCLUDE YMM MACRO PROGRAM */

```

```

/* INPUT FILES */

```

```

%DO SEG = 9 %TO 19;

```

```

  %LET SG = %SYSFUNC(PUTN(&SEG.,Z2.));

```

```

LIBNAME IN&SG. "OPDR.TG.PRD.ETTW.#6266.DAF20P.HDBDCM&SG..SA.V1"
      DISP=SHR;

```

```

%END;

```

```

/* COMBINE 11 DBAD FILES INTO ONE DBAD FILE */

```

```

DATA OUT.DBAD;

```

```

  MERGE

```

```

  %DO SEG= 9 %TO 19;

```

```

    %LET SG = %SYSFUNC(PUTN(&SEG.,Z2.));

```

```

    IN&SG..DBAD

```

```

  %END;

```

```

;

```

Appendix A.299

JCL/SAS Code: DBADEXTF

BY SSN BIC;

RUN;

PROC CONTENTS DATA = OUT.DBAD;

RUN;

%MEND START;

%START;

## Appendix A.300

JCL/SAS Code: CEREXTJ1

```
//#6266CER JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.FINDER.PRDLIB(DBADJCL)
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   2/23/2021 SERGE LUKASHANETS FOR DAF20
//* PHONE:
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//CERPROC PROC
//FIND EXEC SAS9,
//      WORK='120000,60000',PARM='SYSPARM="'&Y.&M"'
//*
//IN DD DSN=MTOSI.CER100.FIELD.D&Y.&M,DISP=SHR
//TEMP DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,(VOL=,,,5)
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.HCER&Y.&M..SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20.HISTDBD.PRDLIB(CEREXT),
//      DISP=(SHR,PASS,KEEP)
// PEND
//*
//CERY PROC
//FRUN01 EXEC CERPROC,M=01
//FRUN02 EXEC CERPROC,M=02
//FRUN03 EXEC CERPROC,M=03
//FRUN04 EXEC CERPROC,M=04
//FRUN05 EXEC CERPROC,M=05
//FRUN06 EXEC CERPROC,M=06
//FRUN07 EXEC CERPROC,M=07
//FRUN08 EXEC CERPROC,M=08
//FRUN09 EXEC CERPROC,M=09
//FRUN10 EXEC CERPROC,M=10
//FRUN11 EXEC CERPROC,M=11
//FRUN12 EXEC CERPROC,M=12
// PEND
//*
//ARUN10 EXEC CERY,Y=10
```

Appendix A.300

JCL/SAS Code: CEREXTJ1

```
//ARUN11 EXEC CERY,Y=11  
//ARUN12 EXEC CERY,Y=12  
//ARUN13 EXEC CERY,Y=13  
//ARUN14 EXEC CERY,Y=14  
//ARUN15 EXEC CERY,Y=15  
//ARUN16 EXEC CERY,Y=16  
//ARUN17 EXEC CERY,Y=17  
//ARUN18 EXEC CERY,Y=18  
//ARUN19 EXEC CERY,Y=19  
//*
```

## Appendix A.301

JCL/SAS Code: CEREXTJ2

```
//#6266CER JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.FINDER.PRDLIB(DBADJCL)
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   2/23/2021 SERGE LUKASHANETS FOR DAF20
//* PHONE:
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//CERPROC PROC
//FIND   EXEC SAS9,
//      WORK='120000,60000',PARM='SYSPARM="'&YM"'
//*
//IN    DD DSN=MTOSI.CER100.FIELD.D&YM.,DISP=SHR
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,(VOL=,,,5)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.HCER&YM..SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20.HISTDBD.PRDLIB(CEREXT),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//FRUN02 EXEC CERPROC,YM=0902
//FRUN03 EXEC CERPROC,YM=0903
//FRUN04 EXEC CERPROC,YM=0904
//FRUN05 EXEC CERPROC,YM=0905
//FRUN06 EXEC CERPROC,YM=0906
//FRUN07 EXEC CERPROC,YM=0907
//FRUN08 EXEC CERPROC,YM=0908
//FRUN09 EXEC CERPROC,YM=0909
//FRUN10 EXEC CERPROC,YM=0910
//FRUN11 EXEC CERPROC,YM=0911
//FRUN12 EXEC CERPROC,YM=0912
//*
```

## Appendix A.302

JCL/SAS Code: CEREXTJ3

```
//#6266CER JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.FINDER.PRDLIB(DBADJCL)
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   2/23/2021 SERGE LUKASHANETS FOR DAF20
//* PHONE:
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//CERPROC PROC
//FIND   EXEC SAS9,
//      WORK='120000,60000',PARM='SYSPARM="'&YM"'
//*
//IN    DD DSN=MTOSI.CER100.FIELD.D&YM.,DISP=SHR
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,(VOL=,,,5)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.HCR&YM..SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20.HISTDBD.PRDLIB(CEREXT),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//FRUN02 EXEC CERPROC,YM=12120
//FRUN03 EXEC CERPROC,YM=13010
//FRUN04 EXEC CERPROC,YM=1301N
//*
```

Appendix A.303

JCL/SAS Code: CEREXT

OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES OBS=MAX;

%MACRO READIN;  
 DATA CER;  
 INFILE IN TRUNCOVER;  
 INPUT

@0015 SSN \$9. /\* SSN SOCIAL SECURITY NUMBER \*/  
 @;  
 IF SSN GT '000000000'; /\*DAF13 MOVED TO INPUT PART OF CODE\*/

INPUT  
 @0108 RCD\_EST ?? YMMDD8. /\* RCD-EST-JD RECORD ESTABLISHMENT DATE\*/

@0223 ERP\_IND \$1. /\* ECONOMIC RECOVERY PAYMENT  
 INDICATOR \*/  
 @0326 RDX \$4. /\* PRIMARY DISABILITY DIAG. CODE\*/  
 /\*THE VARIABLE PDSCC THAT BEGINS AT POSITION 343 IS LISTED AS 6 CHARS  
 LONG AND CONTAINS 2 DIGIT STATE, 3 DIGIT COUNTY, AND 1 DIGIT DO.  
 WE READ ONLY THE FIRST 5 POSITIONS AS TWO VARIABLES PDS2 AND PDS3.\*/  
 @0343 PDS2 \$2. /\* PDSCC RESIDENCE STATE CODE \*/  
 @0345 PDS3 \$3. /\* PDSCC RESIDENCE COUNTY CODE \*/  
 @0349 PDZIP1\_5 \$5. /\* PDZIP1-5 PAYEE'S ZIP CODE \*/  
 @0354 PDZIP6\_9 \$4. /\* PDZIP6-9 PAYEE'S ZIP CODE SUFFIX \*/  
 @0361 IET4 \$1. /\* EARNED INC. TYPE (SELF EMPLOYMENT)\*/  
 @0368 IEA\_NESE 6.2 /\* EARNED INC. AMT. SELF EMPLOYMENT \*/  
 @0382 IEA\_WAGE 6.2 /\* EARNED INC. AMT. (WAGES) \*/  
 @0396 IEA\_BLIND 6.2 /\* EARNED INC. AMT. (BLIND W E)\*/  
 @0410 IEA\_PASS 6.2 /\* EARNED INC. AMT. (PASS)\*/  
 @0424 IEA\_IRWE 6.2 /\* EARNED INC. AMT. (IRWE)\*/  
 @0438 IEA\_LOSS 6.2 /\* EARNED INC. AMT. (NET LOSS)\*/

@;  
 POS = 445;  
 ARRAY UT (6) \$ T1-T6; /\* UNEARNED INCOME TYPE CODE \*/  
 ARRAY UA (6) UN1-UN6; /\* UNEARNED INCOME AMOUNT \*/  
 DO I = 1 TO 6;

INPUT  
 @POS UT(I) \$2.  
 @POS+8 UA(I) 6.2  
 @;  
 POS = POS + 27;  
 END;

INPUT

## Appendix A.303

JCL/SAS Code: CEREXT

```
@0691 AAZIP1_5 $5. /* AAZIP1-5 APPLICANT'S ADDRESS ZIP*/
@0696 AAZIP6_9 $4. /* AAZIP6-9 APPLICANT'S ZIP CODE SUF*/

@0794 CPST $3. /* PSTAT-CURR-PAYMENT STATUS CODE*/
@0811 MEDC $1. /* MEDIC MEDICAID ELIGIBILITY CODE*/

@1003 CIT $1. /* ALIEN ALIEN REFUGE INDICATOR */
@1050 IEA_SEIE 6.2 /* STUDENT EARNED INCOME EXCLUSION
AMOUNT IETYP=B (IEAMT) */
;

DO I = 1 TO 6;
  IF UT(I) = 'A' THEN IUA_SSDI = UA(I); /* TYPE II */
  ELSE IF UT(I) = 'F' THEN IUA_TANF = UA(I); /* TANF */
  ELSE IF UT(I) = 'Q' THEN IUA_WC = UA(I); /* WORKERS COMP */
END;

/*CHECK AAZIP1_5 IF IT IS A VALID ZIP CODES-00000 OR 0000A ARE NOT VALID
NOTDIGIT FUNCTION RETURNS THE POSITION OF THE FIRST NON-DIGIT VALUE
IN THE CHAR STRING*/
IF NOTDIGIT(AAZIP1_5)=0 AND AAZIP1_5 NE '00000' THEN DO;
  RESZIP = AAZIP1_5;
  RZIPS = AAZIP6_9;
END;
ELSE DO;
  RESZIP = PDZIP1_5;
  RZIPS = PDZIP6_9;
END;

FORMAT
  IEA_BLIND IEA_PASS IEA_LOSS IEA_NESE IEA_IRWE
  IEA_WAGE IEA_SEIE IUA_SSDI IUA_TANF IUA_WC 11.2
  RCD_EST YMMDD10.;
;

DROP I POS UN1-UN6 T1-T6;

RUN;

PROC SORT DATA=CER OUT=TEMP.CER; BY SSN RCD_EST; RUN;

DATA OUT.CER;
  SET TEMP.CER;
```



Appendix A.303

JCL/SAS Code: CEREXT

```
BY SSN;  
IF LAST.SSN;  
RUN;
```

```
PROC CONTENTS DATA=OUT.CER; RUN;
```

```
/* CHECK OUTPUT */
```

```
TITLE "CHECK OUTPUT DATASET (OBS=10)";  
PROC PRINT DATA=OUT.CER(OBS=10);  
RUN;  
TITLE;
```

```
%MEND;  
%READIN;
```

## Appendix A.304

JCL/SAS Code: CERCMBJ

```
//#6266CRA JOB (12510000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* PROGRAM:   OPDR.TG.PRD.ETTW.<ID>.DAFY.FINDER.PRDLIB(DBADJCL)
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAMMER: MIRIAM LOEWENBERG
//* UPDATED:   2/23/2021 SERGE LUKASHANETS FOR DAF20
//* PHONE:
//* E-MAIL:    SLUKASHANETS@MATHEMATICA-MPR.COM
//*           SIARHEI.LUKASHANETS@SSA.GOV
//*****
//*
//FINDPROC PROC
//FIND   EXEC SAS9,
//      WORK='120000,60000',PARM='SYSPARM="'&Y.'"
//*
//TEMP  DD DSN=&&TEMP,
//      DISP=(NEW,DELETE,DELETE),
//      UNIT=TSILO,(VOL=,,,5)
//OUT   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.HCERCMB&Y..SA.V1,
//      DISP=(MOD,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20.HISTDBD.PRDLIB(CEREXTC),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//FRUN01 EXEC FINDPROC,Y=09
//FRUN02 EXEC FINDPROC,Y=10
//FRUN03 EXEC FINDPROC,Y=11
//FRUN04 EXEC FINDPROC,Y=12
//FRUN05 EXEC FINDPROC,Y=13
//FRUN06 EXEC FINDPROC,Y=14
//FRUN07 EXEC FINDPROC,Y=15
//FRUN08 EXEC FINDPROC,Y=16
//FRUN09 EXEC FINDPROC,Y=17
//FRUN10 EXEC FINDPROC,Y=18
//FRUN11 EXEC FINDPROC,Y=19
//*
```

## Appendix A.305

JCL/SAS Code: CEREXTC

```
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES MPRINT MACROGEN OBS=MAX;
```

```
%LET YR=%SUBSTR(&SYSPARM,1,2);
```

```
  %LET BEGYR=20&YR.;
```

```
  %LET ENDYR=20&YR.;
```

```
%MACRO START;
```

```
  /* INCLUDE YMMM MACRO PROGRAM */
```

```
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF20.UTILITY.PRDLIB(YMMMCR)";
```

```
%IF &BEGYR = 2009 %THEN %DO;
```

```
  %LET START=2;
```

```
%END;
```

```
%IF &BEGYR > 2009 %THEN %DO;
```

```
  %LET START=1;
```

```
%END;
```

```
%PUT "&BEGYR. STARTS FROM &START.";
```

```
  /* INPUT FILES */
```

```
  %DO MM = &START. %TO &TOTMOS.;
```

```
LIBNAME IN&MM. "OPDR.TG.PRD.ETTW.#6266.DAF20P.HCER&&YY&MM...SA.V1"  
  DISP=SHR;
```

```
  %END;
```

```
  /* COMBINE 12 CER FILES INTO ONE CER FILE */
```

```
DATA OUT.CER;
```

```
  MERGE
```

```
  %DO MM=&START. %TO &TOTMOS.;
```

```
    IN&MM..CER(RENAME=(
```

```
      CIT=    CIT&&YY&MM
```

```
      RDX=    RDX&&YY&MM
```

```
      IET4=   IET4&&YY&MM
```

```
      IEA_BLIND= IEA_BLIND&&YY&MM
```

```
      IEA_PASS= IEA_PASS&&YY&MM
```

```
      IEA_LOSS= IEA_LOSS&&YY&MM
```

```
      IEA_NESE= IEA_NESE&&YY&MM
```

```
      IEA_IRWE= IEA_IRWE&&YY&MM
```

```
      IEA_WAGE= IEA_WAGE&&YY&MM
```

```
      IUA_SSDI= IUA_SSDI&&YY&MM
```

```
      IUA_TANF= IUA_TANF&&YY&MM
```

Appendix A.305

JCL/SAS Code: CEREXTC

```
IUA_WC= IUA_WC&&YY&MM
MEDC= MEDC&&YY&MM
PDS2= PDS2&&YY&MM
PDS3= PDS3&&YY&MM
CPST= CPST&&YY&MM
RESZIP = RZIP&&YY&MM
RZIPS = RZPS&&YY&MM
ERP_IND= ERP_IND&&YY&MM
IEA_SEIE= IEA_SEIE&&YY&MM
PDZIP1_5 = PAYEEZIP&&YY&MM
PDZIP6_9 = PAYEEZIP_EXT&&YY&MM
AAZIP1_5 = APPZIP&&YY&MM
AAZIP6_9 = APPZIP_EXT&&YY&MM
))
%END;
;
BY SSN;
```

```
RUN;
```

```
PROC CONTENTS DATA=OUT.CER; RUN;
```

```
%MEND START;
```

```
%START;
```

## Appendix A.306

JCL/SAS Code: CEREXTF

```
//#6266CRM JOB (11710000,T715,,SAS,,ITC9FL),LUKASHANETS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6266
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6266.DAF20.FINDER.PRDLIB(CERCOMB)
//* PURPOSE:   COMBINE THE MONTHLY CER EXTRACTS AND OUTPUT ONE
//*            RECORD PER BENE
//* UPDATED:   SERGE LUKASHANETS 02/17/2021 FOR DAF20
//* EMAIL:     SLUKASHANETS@MATHEMATICA-MPR.COM
//*
//*****
//*
// SET REG='64M'
//JS010 EXEC SAS9,
//      WORK='200000,100000',          * SPACE IN BLKS *
//      REGION=&REG,
//      PARM='MEMSIZE=&REG'
//*
// *SEE INPUT FILES IN LIBNAME STATEMENTS BELOW
//ADD DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.HCR1212O.SA.V1,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.CMBCERH.SA.V1,
//     DISP=(MOD,CATLG,DELETE),
//     UNIT=TSILO
//SYSIN DD *
OPTIONS NOCENTER LS=132 PS=60 COMPRESS=YES MPRINT MACROGEN OBS=MAX;

%MACRO START;

    %DO SEG = 9 %TO 19;
        %LET SG = %SYSFUNC(PUTN(&SEG.,Z2.));
LIBNAME IN&SG. "OPDR.TG.PRD.ETTW.#6266.DAF20P.HCERCM&SG..SA.V1"
        DISP=SHR;
    %END;

/* COMBINE 12 CER FILES INTO ONE CER FILE */
DATA OUT.CER;
MERGE
    %DO SEG = 9 %TO 19;
        %LET SG = %SYSFUNC(PUTN(&SEG.,Z2.));
        IN&SG..CER
    %END;
```

Appendix A.306

JCL/SAS Code: CEREXTF

```
ADD.CER
  (RENAME=(
CIT=  CIT1212
RDX=  RDX1212
IET4=  IET41212
IEA_BLIND= IEA_BLIND1212
IEA_PASS= IEA_PASS1212
IEA_LOSS= IEA_LOSS1212
IEA_NESE= IEA_NESE1212
IEA_IRWE= IEA_IRWE1212
IEA_WAGE= IEA_WAGE1212
IUA_SSDI= IUA_SSDI1212
IUA_TANF= IUA_TANF1212
IUA_WC=  IUA_WC1212
MEDC=  MEDC1212
PDS2=  PDS21212
PDS3=  PDS31212
CPST=  CPST1212
RESZIP = RZIP1212
RZIPS = RZPS1212
ERP_IND= ERP_IND1212
IEA_SEIE= IEA_SEIE1212
PDZIP1_5 = PAYEEZIP1212
PDZIP6_9 = PAYEEZIP_EXT1212
AAZIP1_5 = APPZIP1212
AAZIP6_9 = APPZIP_EXT1212
  ))

;
BY SSN;

RUN;

PROC CONTENTS DATA=OUT.CER; RUN;

%MEND START;

%START;
```

## Appendix A.307

JCL/SAS Code: 01\_READIN\_ZIPXWALK

```

/*****
* PROJECT:   DAF 50214
* PROGRAM:   01_Readin_ZipXwalk
* DESCRIPTION: READ IN AND QA CROSSWALKS
* DATE:     3/16/2022 MIRIAM CHAPPELKA
* UPDATE:   3/16/2022 MIRIAM CHAPPELKA
* EMAIL:    MCHAPPELKA@MATHEMATICA-MPR.COM
*           MIRIAM.CHAPPELKA@SSA.GOV
*****/

%let inputpath = M:\DAF20\ZIPS TO FIPS\data\input;
LIBNAME OUT 'M:\DAF20\ZIPS TO FIPS\data\output';

%LET STARTYEAR = 12;
%LET ENDYEAR= 21;

/*LIBNAME LOC&YEAR. 'OPRD.TG.PRD.ETTW.#8047.DAF20.ZIPWALK'*/

PROC FORMAT;
  VALUE $MISSING_FMT
    ' ' = 'MISSING'
    " " = 'MISSING'
    '!' = 'MISSING'
    OTHER = 'PRESENT';
RUN;

%MACRO READIN();
%DO YY=&STARTYEAR. %TO &ENDYEAR.;
data xwalk&YY.;
  %let _EFIERR_ = 0;
  infile "&inputpath.\mcmx-zips-full-20&YY..csv" delimiter=',' MISSOVER DSD lrecl=32767 firstobs=2;
    informat ZipCode $9.;
    informat Beg_4 $4. ;
    informat End_4 $4. ;
    informat State $2. ;
    informat CountyFIPS $3.;
    informat CountyFIPS $3.;
    informat PrimaryCity $30.;
  input
    ZipCode $
    Beg_4 $
    End_4 $
    State $
    CountyFIPS

```

Appendix A.307

JCL/SAS Code: 01\_READIN\_ZIPXWALK

Primarycity \$;

```
if _ERROR_ then call symputx('_EFIERR_', 1);
run;
```

```
/* qa: print 10 records to compare with crosswalk files */
```

```
PROC SQL OUTOBS = 10;
title "Check print of 10 obs in 20&YY. crosswalk file";
SELECT *
FROM xwalk&YY.;
QUIT;
```

```
/* qa: confirm that there are no missing zip codes in the crosswalk files*/
```

```
PROC FREQ DATA=XWALK&YY.;
title "Confirm that there are no missing zip codes in the 20&YY. crosswalk files";
format zipcode $MISSING_FMT.;
TABLE ZipCode/LIST MISSING NOCUM;
RUN;
```

```
data out.xwalk&YY.;
set xwalk&YY.;
run;
```

```
%END;
%MEND;
```

```
%READIN();
```



Appendix A.308

JCL/SAS Code: 02\_UPLOAD\_XWALK

```

/*****
* PROJECT:   DAF 50214
* PROGRAM:   02_Upload_Xwalk.sas
* DESCRIPTION: Upload crosswalks to the mainframe
* DATE:      3/16/2022 MIRIAM CHAPPELKA
* UPDATE:    3/16/2022 MIRIAM CHAPPELKA
* EMAIL:     MCHAPPELKA@MATHEMATICA-MPR.COM
*           MIRIAM.CHAPPELKA@SSA.GOV
*****/

```

```
options comamid=tcp; %let SY56=SY56.sspf.ssa.gov; %let tcpsec=_prompt_; options remote=sy56.spawner; run;
```

```
LIBNAME IN 'M:\DAF20\ZIPS TO FIPS\data\output';
```

```
SIGNON;
RSUBMIT;
```

```

libname ZPWK12 "OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY12.SA.V1";
libname ZPWK13 "OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY13.SA.V1";
libname ZPWK14 "OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY14.SA.V1";
libname ZPWK15 "OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY15.SA.V1";
libname ZPWK16 "OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY16.SA.V1";
libname ZPWK17 "OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY17.SA.V1";
libname ZPWK18 "OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY18.SA.V1";
libname ZPWK19 "OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY19.SA.V1";
libname ZPWK20 "OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY20.SA.V1";
libname ZPWK21 "OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY21.SA.V1";

```

```
%let STARTYEAR = 2012;
%let ENDYEAR = 2021;
```

```
%MACRO upload();
  %DO YEAR=&STARTYEAR. %TO &ENDYEAR.;
    %LET YY = %SUBSTR(&YEAR.,3,2);
```

```

    /* upload to mainframe */
    PROC UPLOAD DATA=IN.xwalk&YY.
      OUT=ZPWK&yy..xwalk&yy.(COMPRESS=BINARY);
    RUN;
```

```
%END;
%MEND;
```

```
%upload();
```

Appendix A.308

JCL/SAS Code: 02\_UPLOAD\_XWALK

```
ENDRSUBMIT;  
SIGNOFF;
```

## Appendix A.309

JCL/SAS Code: CVTFMT12

```
//#8047CFT JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#8047
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.EXT.PRDLIB(CVTFMT12) |
//* *-----*
//*
//* PURPOSE: CONVERT CROSSWALKS FROM DATASETS TO FORMATS
//* PROGRAMMER: MIRIAM CHAPPELKA (MCHAPPELKA@MATHEMATICA-MPR.COM)
//* DATE: 3/21/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//ZP12 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY12.SA.V1,DISP=SHR
//OUT12 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSP9.Y12.FMT,
//      DISP=(OLD,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
```

```
%LET YY = 12;
```

```
DATA TEST ;
INPUT ZIP9 $9. ;
DATALINES;
006700001
006700150
006700300
006709999
006700000
;
RUN;
PROC PRINT DATA=TEST;TITLE "SHOW TEST"; RUN;
DATA FMT&YY.;
  SET ZP&YY..XWALK&YY. END = LAST;
  FORMAT START $9. END $9. LABEL $5.;
  STATEFIPS = STFIPS(STATE);
  START = STRIP(ZIPCODE)||STRIP(BEG_4);
  END = STRIP(ZIPCODE)||STRIP(END_4);
  FMTNAME = "$FIPS&YY.ZIP";
```

Appendix A.309

JCL/SAS Code: CVTFMT12

```
LABEL = PUT(STATEFIPS, Z2.) || COUNTYFIPS;
OUTPUT;
IF LAST THEN DO;
  START='OTHER';
  END='OTHER';
  LABEL = ' ';
  OUTPUT;
END;
RUN;
/*PROC SORT DATA=FMT&YY. OUT=SORTEDFMT&YY.;
  BY START END;
RUN; */

/* OUTPUT A FORMAT LIBRARY*/
PROC FORMAT CNTLIN=FMT&YY.
  LIBRARY=OUT&YY. FMTLIB;
RUN;

/*POINT TO THE FORMATS*/
OPTIONS FMTSEARCH=(OUT&YY.);

DATA TEST_CONVERTED;
  SET TEST;
  FORMAT FIPS_ZIP9 $5.;
  FIPS_ZIP9 = PUT(ZIP9, $FIPS&YY.ZIP.);
RUN;

PROC PRINT DATA=TEST_CONVERTED;
  TITLE "TEST FORMAT FOR YEAR 20&YY.";
RUN;
```

## Appendix A.310

JCL/SAS Code: CVTFMT13

```
//#8047CFT JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#8047
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.EXT.PRDLIB(CVTFMT13)|
//*  *-----*
//*
//* PURPOSE: CONVERT CROSSWALKS FROM DATASETS TO FORMATS
//* PROGRAMMER: MIRIAM CHAPPELKA (MCHAPPELKA@MATHEMATICA-MPR.COM)
//* DATE: 3/23/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//ZP13 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY13.SA.V1,DISP=SHR
//OUT13 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSZP9.Y13.FMT,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
```

```
%LET YY = 13;
```

```
DATA TEST ;
INPUT ZIP9 $9. ;
DATALINES;
006700001
006700150
006700300
006709999
006700000
;
RUN;
PROC PRINT DATA=TEST;TITLE "SHOW TEST"; RUN;
DATA FMT&YY.;
  SET ZP&YY..XWALK&YY. END = LAST;
  FORMAT START $9. END $9. LABEL $5.;
  STATEFIPS = STFIPS(STATE);
  START = STRIP(ZIPCODE)||STRIP(BEG_4);
  END = STRIP(ZIPCODE)||STRIP(END_4);
  FMTNAME = "$FIPS&YY.ZIP";
```

Appendix A.310

JCL/SAS Code: CVTFMT13

```
LABEL = PUT(STATEFIPS, Z2.) || COUNTYFIPS;
OUTPUT;
IF LAST THEN DO;
  START='OTHER';
  END='OTHER';
  LABEL = ' ';
  OUTPUT;
END;
RUN;
/*PROC SORT DATA=FMT&YY. OUT=SORTEDFMT&YY.;
  BY START END;
RUN; */

/* OUTPUT A FORMAT LIBRARY*/
PROC FORMAT CNTLIN=FMT&YY.
  LIBRARY=OUT&YY. FMTLIB;
RUN;

/*POINT TO THE FORMATS*/
OPTIONS FMTSEARCH=(OUT&YY.);

DATA TEST_CONVERTED;
  SET TEST;
  FORMAT FIPS_ZIP9 $5.;
  FIPS_ZIP9 = PUT(ZIP9, $FIPS&YY.ZIP.);
RUN;

PROC PRINT DATA=TEST_CONVERTED;
  TITLE "TEST FORMAT FOR YEAR 20&YY.";
RUN;
```

## Appendix A.311

JCL/SAS Code: CVTFMT14

```
//#8047CFT JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#8047
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.EXT.PRDLIB(CVTFMT14)|
//* *-----*
//*
//* PURPOSE: CONVERT CROSSWALKS FROM DATASETS TO FORMATS
//* PROGRAMMER: MIRIAM CHAPPELKA (MCHAPPELKA@MATHEMATICA-MPR.COM)
//* DATE: 3/23/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//ZP14 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY14.SA.V1,DISP=SHR
//OUT14 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSP9.Y14.FMT,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
```

```
%LET YY = 14;
```

```
DATA TEST ;
INPUT ZIP9 $9. ;
DATALINES;
006700001
006700150
006700300
006709999
006700000
;
RUN;
PROC PRINT DATA=TEST;TITLE "SHOW TEST"; RUN;
DATA FMT&YY.;
  SET ZP&YY..XWALK&YY. END = LAST;
  FORMAT START $9. END $9. LABEL $5.;
  STATEFIPS = STFIPS(STATE);
  START = STRIP(ZIPCODE)||STRIP(BEG_4);
  END = STRIP(ZIPCODE)||STRIP(END_4);
  FMTNAME = "$FIPS&YY.ZIP";
```

Appendix A.311

JCL/SAS Code: CVTFMT14

```
LABEL = PUT(STATEFIPS, Z2.) || COUNTYFIPS;
OUTPUT;
IF LAST THEN DO;
  START='OTHER';
  END='OTHER';
  LABEL = ' ';
  OUTPUT;
END;
RUN;
/*PROC SORT DATA=FMT&YY. OUT=SORTEDFMT&YY.;
  BY START END;
RUN; */

/* OUTPUT A FORMAT LIBRARY*/
PROC FORMAT CNTLIN=FMT&YY.
  LIBRARY=OUT&YY. FMTLIB;
RUN;

/*POINT TO THE FORMATS*/
OPTIONS FMTSEARCH=(OUT&YY.);

DATA TEST_CONVERTED;
  SET TEST;
  FORMAT FIPS_ZIP9 $5.;
  FIPS_ZIP9 = PUT(ZIP9, $FIPS&YY.ZIP.);
RUN;

PROC PRINT DATA=TEST_CONVERTED;
  TITLE "TEST FORMAT FOR YEAR 20&YY.";
RUN;
```



## Appendix A.312

JCL/SAS Code: CVTFMT15

```
//#8047CFT JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#8047
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.EXT.PRDLIB(CVTFMT15)|
//*  *-----*
//*
//* PURPOSE: CONVERT CROSSWALKS FROM DATASETS TO FORMATS
//* PROGRAMMER: MIRIAM CHAPPELKA (MCHAPPELKA@MATHEMATICA-MPR.COM)
//* DATE: 3/23/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//ZP15 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY15.SA.V1,DISP=SHR
//OUT15 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSP9.Y15.FMT,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
```

```
%LET YY = 15;
```

```
DATA TEST ;
INPUT ZIP9 $9. ;
DATALINES;
006700001
006700150
006700300
006709999
006700000
;
RUN;
PROC PRINT DATA=TEST;TITLE "SHOW TEST"; RUN;
DATA FMT&YY.;
  SET ZP&YY..XWALK&YY. END = LAST;
  FORMAT START $9. END $9. LABEL $5.;
  STATEFIPS = STFIPS(STATE);
  START = STRIP(ZIPCODE)||STRIP(BEG_4);
  END = STRIP(ZIPCODE)||STRIP(END_4);
  FMTNAME = "$FIPS&YY.ZIP";
```

Appendix A.312

JCL/SAS Code: CVTFMT15

```
LABEL = PUT(STATEFIPS, Z2.) || COUNTYFIPS;
OUTPUT;
IF LAST THEN DO;
  START='OTHER';
  END='OTHER';
  LABEL = ' ';
  OUTPUT;
END;
RUN;
/*PROC SORT DATA=FMT&YY. OUT=SORTEDFMT&YY.;
  BY START END;
RUN; */

/* OUTPUT A FORMAT LIBRARY*/
PROC FORMAT CNTLIN=FMT&YY.
  LIBRARY=OUT&YY. FMTLIB;
RUN;

/*POINT TO THE FORMATS*/
OPTIONS FMTSEARCH=(OUT&YY.);

DATA TEST_CONVERTED;
  SET TEST;
  FORMAT FIPS_ZIP9 $5.;
  FIPS_ZIP9 = PUT(ZIP9, $FIPS&YY.ZIP.);
RUN;

PROC PRINT DATA=TEST_CONVERTED;
  TITLE "TEST FORMAT FOR YEAR 20&YY.";
RUN;
```

## Appendix A.313

JCL/SAS Code: CVTFMT16

```

//#8047CFT JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#8047
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.EXT.PRDLIB(CVTFMT16)|
//*  *-----*
//*
//* PURPOSE: CONVERT CROSSWALKS FROM DATASETS TO FORMATS
//* PROGRAMMER: MIRIAM CHAPPELKA (MCHAPPELKA@MATHEMATICA-MPR.COM)
//* DATE: 3/23/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//ZP16 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY16.SA.V1,DISP=SHR
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSP9.Y16.FMT,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *

```

```

OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;

```

```

%LET YY = 16;

```

```

DATA TEST ;
INPUT ZIP9 $9. ;
DATALINES;
006700001
006700150
006700300
006709999
006700000
;
RUN;
PROC PRINT DATA=TEST;TITLE "SHOW TEST"; RUN;
DATA FMT&YY.;
  SET ZP&YY..XWALK&YY. END = LAST;
  FORMAT START $9. END $9. LABEL $5.;
  STATEFIPS = STFIPS(STATE);
  START = STRIP(ZIPCODE)||STRIP(BEG_4);
  END = STRIP(ZIPCODE)||STRIP(END_4);
  FMTNAME = "$FIPS&YY.ZIP";

```

Appendix A.313

JCL/SAS Code: CVTFMT16

```
LABEL = PUT(STATEFIPS, Z2.) || COUNTYFIPS;
OUTPUT;
IF LAST THEN DO;
  START='OTHER';
  END='OTHER';
  LABEL = ' ';
  OUTPUT;
END;
RUN;
/*PROC SORT DATA=FMT&YY. OUT=SORTEDFMT&YY.;
  BY START END;
RUN; */

/* OUTPUT A FORMAT LIBRARY*/
PROC FORMAT CNTLIN=FMT&YY.
  LIBRARY=OUT&YY. FMTLIB;
RUN;

/*POINT TO THE FORMATS*/
OPTIONS FMTSEARCH=(OUT&YY.);

DATA TEST_CONVERTED;
  SET TEST;
  FORMAT FIPS_ZIP9 $5.;
  FIPS_ZIP9 = PUT(ZIP9, $FIPS&YY.ZIP.);
RUN;

PROC PRINT DATA=TEST_CONVERTED;
  TITLE "TEST FORMAT FOR YEAR 20&YY.";
RUN;
```

## Appendix A.314

JCL/SAS Code: CVTFMT17

```
//#8047CFT JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#8047
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.EXT.PRDLIB(CVTFMT17)|
//* *-----*
//*
//* PURPOSE: CONVERT CROSSWALKS FROM DATASETS TO FORMATS
//* PROGRAMMER: MIRIAM CHAPPELKA (MCHAPPELKA@MATHEMATICA-MPR.COM)
//* DATE: 3/23/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//ZP17 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY17.SA.V1,DISP=SHR
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSZP9.Y17.FMT,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
```

```
%LET YY = 17;
```

```
DATA TEST ;
INPUT ZIP9 $9. ;
DATALINES;
006700001
006700150
006700300
006709999
006700000
;
RUN;
PROC PRINT DATA=TEST;TITLE "SHOW TEST"; RUN;
DATA FMT&YY.;
  SET ZP&YY..XWALK&YY. END = LAST;
  FORMAT START $9. END $9. LABEL $5.;
  STATEFIPS = STFIPS(STATE);
  START = STRIP(ZIPCODE)||STRIP(BEG_4);
  END = STRIP(ZIPCODE)||STRIP(END_4);
  FMTNAME = "$FIPS&YY.ZIP";
```

Appendix A.314

JCL/SAS Code: CVTFMT17

```
LABEL = PUT(STATEFIPS, Z2.) || COUNTYFIPS;
OUTPUT;
IF LAST THEN DO;
  START='OTHER';
  END='OTHER';
  LABEL = ' ';
  OUTPUT;
END;
RUN;
/*PROC SORT DATA=FMT&YY. OUT=SORTEDFMT&YY.;
  BY START END;
RUN; */

/* OUTPUT A FORMAT LIBRARY*/
PROC FORMAT CNTLIN=FMT&YY.
  LIBRARY=OUT&YY. FMTLIB;
RUN;

/*POINT TO THE FORMATS*/
OPTIONS FMTSEARCH=(OUT&YY.);

DATA TEST_CONVERTED;
  SET TEST;
  FORMAT FIPS_ZIP9 $5.;
  FIPS_ZIP9 = PUT(ZIP9, $FIPS&YY.ZIP.);
RUN;

PROC PRINT DATA=TEST_CONVERTED;
  TITLE "TEST FORMAT FOR YEAR 20&YY.";
RUN;
```

## Appendix A.315

JCL/SAS Code: CVTFMT18

```
//#8047CFT JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#8047
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.EXT.PRDLIB(CVTFMT18)|
//* *-----*
//*
//* PURPOSE: CONVERT CROSSWALKS FROM DATASETS TO FORMATS
//* PROGRAMMER: MIRIAM CHAPPELKA (MCHAPPELKA@MATHEMATICA-MPR.COM)
//* DATE: 3/23/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//ZP18 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY18.SA.V1,DISP=SHR
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSP9.Y18.FMT,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
```

```
%LET YY = 18;
```

```
DATA TEST ;
INPUT ZIP9 $9. ;
DATALINES;
006700001
006700150
006700300
006709999
006700000
;
RUN;
PROC PRINT DATA=TEST;TITLE "SHOW TEST"; RUN;
DATA FMT&YY.;
  SET ZP&YY..XWALK&YY. END = LAST;
  FORMAT START $9. END $9. LABEL $5.;
  STATEFIPS = STFIPS(STATE);
  START = STRIP(ZIPCODE)||STRIP(BEG_4);
  END = STRIP(ZIPCODE)||STRIP(END_4);
  FMTNAME = "$FIPS&YY.ZIP";
```

Appendix A.315

JCL/SAS Code: CVTFMT18

```
LABEL = PUT(STATEFIPS, Z2.) || COUNTYFIPS;
OUTPUT;
IF LAST THEN DO;
  START='OTHER';
  END='OTHER';
  LABEL = ' ';
  OUTPUT;
END;
RUN;
/*PROC SORT DATA=FMT&YY. OUT=SORTEDFMT&YY.;
  BY START END;
RUN; */

/* OUTPUT A FORMAT LIBRARY*/
PROC FORMAT CNTLIN=FMT&YY.
  LIBRARY=OUT&YY. FMTLIB;
RUN;

/*POINT TO THE FORMATS*/
OPTIONS FMTSEARCH=(OUT&YY.);

DATA TEST_CONVERTED;
  SET TEST;
  FORMAT FIPS_ZIP9 $5.;
  FIPS_ZIP9 = PUT(ZIP9, $FIPS&YY.ZIP.);
RUN;

PROC PRINT DATA=TEST_CONVERTED;
  TITLE "TEST FORMAT FOR YEAR 20&YY.";
RUN;
```



## Appendix A.316

JCL/SAS Code: CVTFMT19

```
//#8047CFT JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#8047
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.EXT.PRDLIB(CVTFMT19)|
//* *-----*
//*
//* PURPOSE: CONVERT CROSSWALKS FROM DATASETS TO FORMATS
//* PROGRAMMER: MIRIAM CHAPPELKA (MCHAPPELKA@MATHEMATICA-MPR.COM)
//* DATE: 3/23/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//ZP19 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY19.SA.V1,DISP=SHR
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSP9.Y19.FMT,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
```

```
%LET YY = 19;
```

```
DATA TEST ;
INPUT ZIP9 $9. ;
DATALINES;
006700001
006700150
006700300
006709999
006700000
;
RUN;
PROC PRINT DATA=TEST;TITLE "SHOW TEST"; RUN;
DATA FMT&YY.;
  SET ZP&YY..XWALK&YY. END = LAST;
  FORMAT START $9. END $9. LABEL $5.;
  STATEFIPS = STFIPS(STATE);
  START = STRIP(ZIPCODE)||STRIP(BEG_4);
  END = STRIP(ZIPCODE)||STRIP(END_4);
  FMTNAME = "$FIPS&YY.ZIP";
```

Appendix A.316

JCL/SAS Code: CVTFMT19

```
LABEL = PUT(STATEFIPS, Z2.) || COUNTYFIPS;
OUTPUT;
IF LAST THEN DO;
  START='OTHER';
  END='OTHER';
  LABEL = ' ';
  OUTPUT;
END;
RUN;
/*PROC SORT DATA=FMT&YY. OUT=SORTEDFMT&YY.;
  BY START END;
RUN; */

/* OUTPUT A FORMAT LIBRARY*/
PROC FORMAT CNTLIN=FMT&YY.
  LIBRARY=OUT&YY. FMTLIB;
RUN;

/*POINT TO THE FORMATS*/
OPTIONS FMTSEARCH=(OUT&YY.);

DATA TEST_CONVERTED;
  SET TEST;
  FORMAT FIPS_ZIP9 $5.;
  FIPS_ZIP9 = PUT(ZIP9, $FIPS&YY.ZIP.);
RUN;

PROC PRINT DATA=TEST_CONVERTED;
  TITLE "TEST FORMAT FOR YEAR 20&YY.";
RUN;
```

## Appendix A.317

JCL/SAS Code: CVTFMT20

```
//#8047CFT JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#8047
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.EXT.PRDLIB(CVTFMT20)|
//* *-----*
//*
//* PURPOSE: CONVERT CROSSWALKS FROM DATASETS TO FORMATS
//* PROGRAMMER: MIRIAM CHAPPELKA (MCHAPPELKA@MATHEMATICA-MPR.COM)
//* DATE: 3/23/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//ZP20 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY20.SA.V1,DISP=SHR
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSP9.Y20.FMT,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
```

```
%LET YY = 20;
```

```
DATA TEST ;
INPUT ZIP9 $9. ;
DATALINES;
006700001
006700150
006700300
006709999
006700000
;
RUN;
PROC PRINT DATA=TEST;TITLE "SHOW TEST"; RUN;
DATA FMT&YY.;
  SET ZP&YY..XWALK&YY. END = LAST;
  FORMAT START $9. END $9. LABEL $5.;
  STATEFIPS = STFIPS(STATE);
  START = STRIP(ZIPCODE)||STRIP(BEG_4);
  END = STRIP(ZIPCODE)||STRIP(END_4);
  FMTNAME = "$FIPS&YY.ZIP";
```

Appendix A.317

JCL/SAS Code: CVTFMT20

```
LABEL = PUT(STATEFIPS, Z2.) || COUNTYFIPS;
OUTPUT;
IF LAST THEN DO;
  START='OTHER';
  END='OTHER';
  LABEL = ' ';
  OUTPUT;
END;
RUN;
/*PROC SORT DATA=FMT&YY. OUT=SORTEDFMT&YY.;
  BY START END;
RUN; */

/* OUTPUT A FORMAT LIBRARY*/
PROC FORMAT CNTLIN=FMT&YY.
  LIBRARY=OUT&YY. FMTLIB;
RUN;

/*POINT TO THE FORMATS*/
OPTIONS FMTSEARCH=(OUT&YY.);

DATA TEST_CONVERTED;
  SET TEST;
  FORMAT FIPS_ZIP9 $5.;
  FIPS_ZIP9 = PUT(ZIP9, $FIPS&YY.ZIP.);
RUN;

PROC PRINT DATA=TEST_CONVERTED;
  TITLE "TEST FORMAT FOR YEAR 20&YY.";
RUN;
```

## Appendix A.318

JCL/SAS Code: CVTFMT21

```
//#8047CFT JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=#8047
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.<ID>.DAF<YY>.EXT.PRDLIB(CVTFMT21)|
//*  *-----*
//*
//* PURPOSE: CONVERT CROSSWALKS FROM DATASETS TO FORMATS
//* PROGRAMMER: MIRIAM CHAPPELKA (MCHAPPELKA@MATHEMATICA-MPR.COM)
//* DATE: 3/24/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000'
//*
//ZP21 DD DSN=OPDR.TG.PRD.ETTW.#6502.DAF20P.ZPXWKY21.SA.V1,DISP=SHR
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSP9.Y21.FMT,
//      DISP=(NEW,CATLG,DELETE),
//      SPACE=(CYL,(1000,100),RLSE)
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
```

```
%LET YY = 21;
```

```
DATA TEST ;
INPUT ZIP9 $9. ;
DATALINES;
006700001
006700150
006700300
006709999
006700000
;
RUN;
PROC PRINT DATA=TEST;TITLE "SHOW TEST"; RUN;
DATA FMT&YY.;
  SET ZP&YY..XWALK&YY. END = LAST;
  FORMAT START $9. END $9. LABEL $5.;
  STATEFIPS = STFIPS(STATE);
  START = STRIP(ZIPCODE)||STRIP(BEG_4);
  END = STRIP(ZIPCODE)||STRIP(END_4);
  FMTNAME = "$FIPS&YY.ZIP";
```

Appendix A.318

JCL/SAS Code: CVTFMT21

```
LABEL = PUT(STATEFIPS, Z2.) || COUNTYFIPS;
OUTPUT;
IF LAST THEN DO;
  START='OTHER';
  END='OTHER';
  LABEL = ' ';
  OUTPUT;
END;
RUN;
/*PROC SORT DATA=FMT&YY. OUT=SORTEDFMT&YY.;
  BY START END;
RUN; */

/* OUTPUT A FORMAT LIBRARY*/
PROC FORMAT CNTLIN=FMT&YY.
  LIBRARY=OUT&YY. FMTLIB;
RUN;

/*POINT TO THE FORMATS*/
OPTIONS FMTSEARCH=(OUT&YY.);

DATA TEST_CONVERTED;
  SET TEST;
  FORMAT FIPS_ZIP9 $5.;
  FIPS_ZIP9 = PUT(ZIP9, $FIPS&YY.ZIP.);
RUN;

PROC PRINT DATA=TEST_CONVERTED;
  TITLE "TEST FORMAT FOR YEAR 20&YY.";
RUN;
```

## Appendix A.319

JCL/SAS Code: ZJCL1619

```
// $4671ZIP JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=$4671
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.$4671.DAF21.ZIPFIP.PRDLIB(ZIPMATCH)|
//* *-----*
//*
//* PURPOSE: ADDITIONAL ANALYSIS TO SEE PERCENTAGE OF ZIP9 MATCHING
//*      RATES FOR SSI AND SSDI BENES OVER TIME
//* PROGRAMMER: BRITTANY COLIP (BCOLIP@MATHEMATICA-MPR.COM)
//* DATE: 8/10/2022
//*****
//*
//FIPS  PROC
//JS010  EXEC SAS9,
//      WORK='120000,60000',
//      PARM='SYSPARM="'&YYYY"'
//*
//* 2009-2019 DBAD
//DBAD  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.CMBDBDH.SA.V1,DISP=SHR
//* 2009-2019 CER
//CER   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF20P.CMBCERH.SA.V1,DISP=SHR
//* FIPS FORMAT FILES
//FMT   DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSZP9.Y&YY..FMT,DISP=SHR
//FMTH  DD DSN=OPDR.TG.PRD.ETTW.&FID..DAF&YY..FIPS.FMTLIB,DISP=SHR
//* TEMP OUTPUT
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//* OUTPUT FILE
//OUT   DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.ZIP&YY..SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21.ZIPFIP.PRDLIB(ZIP2FIP),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//RUN   EXEC FIPS,YYYY=2016,YY=16,FID=#6502
//RUN   EXEC FIPS,YYYY=2017,YY=17,FID=#6502
//RUN   EXEC FIPS,YYYY=2018,YY=18,FID=#4941
//RUN   EXEC FIPS,YYYY=2019,YY=19,FID=#6502
//*
```

## Appendix A.320

JCL/SAS Code: zip2fip

```

*****
*   *-----*
*   | OPDR.TG.PRD.ETTW.$4671.DAF21.ZIPFIP.PRDLIB(ZIPTOFIP)|
*   *-----*
*
* EXECUTE BY OPDR.TG.PRD.ETTW.$4671.DAF21.ZIPFIP.PRDLIB(ZJCL???)
*
* PURPOSE: ADDITIONAL ANALYSIS TO SEE PERCENTAGE OF ZIP9 MATCHING
*          RATES FOR SSI AND SSDI BENES OVER TIME
* PROGRAMMER: DAWN PHELPS (DPHELPS@MATHEMATICA-MPR.COM)
* DATE: 11/10/2022
*****
*
;

OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;

%MACRO DOIT;

* POINT TO THE FORMATS;
OPTIONS FMTSEARCH=(FMT FMTH);

* SET MACRO VARS;
%LET YYYY=&SYSPARM.; * SYSPARM IS PASSED FORM THE ZIPJCL PGM;
%LET YR=%SUBSTR(&SYSPARM,3);

%LET BEGYR = &YYYY.;
%LET ENDYR = &YYYY.;
%LET ENDMN = 12;

* INCLUDING THE YMMM MACRO FROM THE UTILITY LIBRARY;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMCR)";

%MACRO PREP_DI_SSI(
  INDSN= /* LIBREF & NAME OF INPUT DATASET (EITHER CER OR DBAD) */,
  OUTDSN= /* LIBREF & NAME OF OUTPUT DATASET */,
  POP= /* EITHER DI OR SSI */,
  ZIP5_VAR= /* NAME OF THE 5-DIGIT ZIP VARIABLE TO USE */,
  ZIPADD_VAR= /* NAME OF THE ZIP EXTENSION VARIABLE TO USE */
);

* CREATE A TEMPORARY PROGRAM FILE;
DATA &OUTDSN.;
  SET &INDSN. (KEEP=SSN &ZIP5_VAR.&YR.: &ZIPADD_VAR.&YR.);

```



## Appendix A.320

JCL/SAS Code: zip2fip

```
*** CREATE NEW VARIABLES USING ARRAYS ***;

* INPUT ARRAYS;
ARRAY ZIPMAIN (*) &ZIP5_VAR.&YR.: ;
ARRAY ZIPADD (*) &ZIPADD_VAR.&YR.;;

* OUTPUT ARRAYS;
ARRAY FIPS (*) $5 %DO MM=1 %TO &TOTMOS; FIPS_&POP.&&YY&MM %END;;

* PERFORM CALCS;
DO I=1 TO DIM(ZIPMAIN);

    * APPLY NEW ZIP/FIPS FORMAT TO 9 DIGIT ZIP;
    FIPS(I)=PUT(STRIP(ZIPMAIN(I)) | STRIP(ZIPADD(I)),$FIPS&YR.ZIP.);

    * APPLY OLD ZIP/FIPS FORMAT TO 5 DIGIT ZIP;
    IF FIPS(I)=" THEN FIPS(I)=PUT(ZIPMAIN(I),$FIPS.);

END;
RUN;

%MEND PREP_DI_SSI;

%PREP_DI_SSI(INDSN=DBAD.DBAD
,OUTDSN=TEMP1.DBAD
,POP=DI
,ZIP5_VAR=ZIP_DI
,ZIPADD_VAR=ZIP_DIADD
);
%PREP_DI_SSI(INDSN=CER.CER
,OUTDSN=TEMP2.CER
,POP=SSI
,ZIP5_VAR=PAYEEZIP
,ZIPADD_VAR=PAYEEZIP_EXT
);

* DBAD NEEDS TO BE REDUCED TO 1 REC PER SSN - THIS METHOD OF REDUCTION
IS CONSISTENT WITH HOW WE USE THE DBAD FILE IN CREATION OF ANNUALS;
PROC SORT DATA=TEMP1.DBAD OUT=TEMP3.DBAD NODUPKEYS;
    BY SSN;
RUN;

DATA OUT.ZIP&YR.;
    MERGE TEMP3.DBAD(RENAME=(%DO MM=1 %TO &TOTMOS;
        ZIP_DI&&YY&MM=ZIP_DI_BASE&&YY&MM
```

## Appendix A.320

JCL/SAS Code: zip2fip

```

        ZIP_DIADD&&YY&MM=ZIP_DI_EXT&&YY&MM
    %END;))
TEMP2.CER(RENAME=(%DO MM=1 %TO &TOTMOS;
    PAYEEZIP&&YY&MM=ZIP_SSI_BASE&&YY&MM
    PAYEEZIP_EXT&&YY&MM=ZIP_SSI_EXT&&YY&MM
    %END;))
;
BY SSN;

* FIPS ARRAYS;
ARRAY FIPS_SSI (*) $5. %DO MM=1 %TO &TOTMOS; FIPS_SSI&&YY&MM
    %END;;
ARRAY FIPS_DI (*) $5. %DO MM=1 %TO &TOTMOS; FIPS_DI&&YY&MM
    %END;;
ARRAY FIPS_BEST (*) $5. %DO MM=1 %TO &TOTMOS; FIPS_BEST&&YY&MM
    %END;;

/* CREATE BEST FIPS */
DO I = 1 TO DIM(FIPS_SSI);
    IF FIPS_SSI(I) NE ' ' THEN FIPS_BEST(I)=FIPS_SSI(I);
    ELSE FIPS_BEST(I)=FIPS_DI(I);
END;

ARRAY ZIP_SSI (*) $5. %DO MM=1 %TO &TOTMOS; ZIP_SSI_BASE&&YY&MM
    %END;;
ARRAY ZIP_SSI_EXT (*) $4. %DO MM=1 %TO &TOTMOS; ZIP_SSI_EXT&&YY&MM
    %END;;
ARRAY ZIP_DI (*) $5. %DO MM=1 %TO &TOTMOS; ZIP_DI_BASE&&YY&MM
    %END;;
ARRAY ZIP_DI_EXT (*) $4. %DO MM=1 %TO &TOTMOS; ZIP_DI_EXT&&YY&MM
    %END;;
ARRAY ZIP_BEST (*) $5. %DO MM=1 %TO &TOTMOS; ZIP_BEST_BASE&&YY&MM
    %END;;
ARRAY ZIP_BEST_EXT (*) $4. %DO MM=1 %TO &TOTMOS; ZIP_BEST_EXT&&YY&MM
    %END;;

/* CREATE BEST ZIPS */
DO I = 1 TO DIM(ZIP_SSI);
    IF ZIP_SSI(I) NE ' ' AND ZIP_SSI(I) NE '00000'
    THEN DO;
        ZIP_BEST(I)=ZIP_SSI(I);
        ZIP_BEST_EXT(I)=ZIP_SSI_EXT(I);
    END;
    ELSE DO;
        ZIP_BEST(I)=ZIP_DI(I);

```

Appendix A.320

JCL/SAS Code: zip2fip

```

        ZIP_BEST_EXT(I)=ZIP_DI_EXT(I);
    END;
END;

/* CREATE MONTHLY STATE INDICATORS FOR PHASE CONSTRUCTION */
ARRAY PST  (*) $2 %DO MM=1 %TO &TOTMOS; PST&&YY&MM %END;;

DO I=1 TO DIM(ZIP_SSI);
    PST(I)=ZIPSTATE(ZIP_BEST(I));
END;

RUN;

%MACRO SKIP;
PROC FREQ DATA=OUT.ZIP&YR.;
    %DO MM=1 %TO &TOTMOS;
    TITLE1 "FOR YYMM=&&YY&MM";
    TABLE ZIP_DI_BASE&&YY&MM*ZIP_DI_EXT&&YY&MM
        *ZIP_SSI_BASE&&YY&MM*ZIP_SSI_EXT&&YY&MM
        *ZIP_BEST_BASE&&YY&MM*ZIP_BEST_EXT&&YY&MM
        *FIPS_DI&&YY&MM
        *FIPS_SSI&&YY&MM
        *FIPS_BEST&&YY&MM
        *PST&&YY&MM
        / MISSING LIST;
    %END;
%MEND SKIP;

PROC CONTENTS DATA=OUT.ZIP&YR.;
RUN;

%MEND DOIT;
%DOIT;

```

## Appendix A.321

JCL/SAS Code: ZJCL2021

```
//$4671Z12 JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=$4671
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.$4671.DAF21.ZIPFIP.PRDLIB(ZIPMATCH)|
//* *-----*
//*
//* PURPOSE: ADDITIONAL ANALYSIS TO SEE PERCENTAGE OF ZIP9 MATCHING
//*      RATES FOR SSI AND SSDI BENES OVER TIME
//* PROGRAMMER: BRITTANY COLIP (BCOLIP@MATHEMATICA-MPR.COM)
//* DATE: 8/10/2022
//*****
//*
//FIPS  PROC
//JS010  EXEC SAS9,
//      WORK='120000,60000',
//      PARM='SYSPARM="'&YYYY"'
//*
//* 2020-2021 DBAD
//DBAD  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF&YY.P.CMBDBDE.SA.V1,DISP=SHR
//* 2020-2021 CER
//CER   DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF&YY.P.COMBCERE.SA.V1,DISP=SHR
//* FIPS FORMAT FILES
//FMT   DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF20.FIPSZP9.Y&YY..FMT,DISP=SHR
//FMTH  DD DSN=OPDR.TG.PRD.ETTW.&FID..DAF&YY..FIPS.FMTLIB,DISP=SHR
//* TEMP OUTPUT
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//TEMP3 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),UNIT=TSILO,VOL=(,,10)
//* OUTPUT FILE
//OUT   DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.ZIP&YY..SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21.ZIPFIP.PRDLIB(ZIP2FIP),
//      DISP=(SHR,PASS,KEEP)
//      PEND
//*
//RUN   EXEC FIPS,YYYY=2020,YY=20,FID=#8047
//RUN   EXEC FIPS,YYYY=2021,YY=21,FID=#6266
//*
```

## Appendix A.322

JCL/SAS Code: prop

```
//$4671POP JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=$4671
//*****
//*
//*  *-----*
//*  | OPDR.TG.PRD.ETTW.$4671.DAF21.ZIPFIP.PRDLIB(ZIPMATCH)|
//*  *-----*
//*
//* PURPOSE: ADDITIONAL ANALYSIS TO SEE PERCENTAGE OF ZIP9 MATCHING
//*      RATES FOR SSI AND SSDI BENES OVER TIME
//* PROGRAMMER: BRITTANY COLIP (BCOLIP@MATHEMATICA-MPR.COM)
//* DATE: 8/10/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000',
//      PARM='SYSPARM="'&YYYY'"
//*
//Y15 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21I.Y2015.SA.V1,DISP=SHR
//Y16 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.ZIP16.SA.V1,DISP=SHR
//Y17 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.ZIP17.SA.V1,DISP=SHR
//Y18 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.ZIP18.SA.V1,DISP=SHR
//Y19 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.ZIP19.SA.V1,DISP=SHR
//Y20 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.ZIP20.SA.V1,DISP=SHR
//Y21 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.ZIP21.SA.V1,DISP=SHR
//* OUTPUT FILE
//OUT DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.ZIPALL.SA.V1,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD *
```

```
OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;
```

```
%MACRO DOIT;
* SET MACRO VARS;
```

```
%LET BEGYR = 2016;
%LET ENDYR = 2021;
%LET ENDMN = 12;
```

```
* INCLUDING THE YMM MACRO FROM THE UTILITY LIBRARY;
%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";
```

```
DATA OUT.ZIPALL;
```

## Appendix A.322

JCL/SAS Code: prop

```
MERGE Y15.Y2015(IN=Y15 KEEP=SSN ZIP_BEST1512 FIPS_BEST1512)
  Y16.ZIP16(IN=Z16)
  Y17.ZIP17(IN=Z17)
  Y18.ZIP18(IN=Z18)
  Y19.ZIP19(IN=Z19)
  Y20.ZIP20(IN=Z20)
  Y21.ZIP21(IN=Z21)
  ;
BY SSN;

ARRAY ZIPB (*) $ ZIP_BEST1512
      %DO MM=1 %TO &TOTMOS.; ZIP_BEST_BASE&&YY&MM %END;;
ARRAY ZIPE (*) $ ZIP_DUMMY
      %DO MM=1 %TO &TOTMOS.; ZIP_BEST_EXT&&YY&MM %END;;
ARRAY FIP (*) $ FIPS_BEST1512
      %DO MM=1 %TO &TOTMOS.; FIPS_BEST&&YY&MM %END;;
ARRAY PST (*) $ STATE1512
      %DO MM=1 %TO &TOTMOS.; PST&&YY&MM %END;;

/* PROPAGNATE ZIP_BEST */
DO I=2 TO DIM(ZIPB);
  IF MISSING(ZIPB(I)) THEN DO;
    ZIPB(I)=ZIPB(I-1);
    ZIPE(I)=ZIPE(I-1);
  END;
  IF MISSING(FIP(I)) THEN FIP(I)=FIP(I-1);
  IF MISSING(PST(I)) THEN PST(I)=PST(I-1);
END;
RUN;

PROC CONTENTS DATA=OUT.ZIPALL;
RUN;

%MEND DOIT;
%DOIT;
```

## Appendix A.323

JCL/SAS Code: jclyrsp

```
//$4671MSP JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//*
//* *-----*
//* | THE MACRO SPLIT PROGRAM           |
//* *-----*
//*
//* CREATED BY: SERGE LUKASHANETS, 5/26/2016
//* E-MAIL DPHELPS@MATHEMATICA-MPR.COM
//*****
// SET REG='64M'
//*
//JS010 EXEC SAS9,
//  WORK='200000,100000',
//  REGION=&REG,
//  PARM='MEMSIZE=&REG'
//*
//***DATA FOR SPLIT
//IN1  DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.ZIPALL.SA.V1,DISP=SHR
//***OUT YEARLY FILES
//OUT21 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP21.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT20 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP20.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT19 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP19.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT18 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP18.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT17 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP17.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//OUT16 DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21P.ZIPFIP16.SA.V1,
//  DISP=(NEW,CATLG,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
OPTIONS NOCENTER COMPRESS=YES MPRINT OBS=MAX;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(MSPLIT)";
```

Appendix A.323

JCL/SAS Code: jclyrsp

```
%OUTP(IN1.ZIPALL(DROP=I STATE1512 ZIP_BEST1512 ZIP_DUMMY)
,OUT,ZIPFIP,2016,2021,SSN);
```

```
PROC CONTENTS DATA=OUT21.ZIPFIP21;RUN;
PROC CONTENTS DATA=OUT20.ZIPFIP20;RUN;
PROC CONTENTS DATA=OUT19.ZIPFIP19;RUN;
PROC CONTENTS DATA=OUT18.ZIPFIP18;RUN;
PROC CONTENTS DATA=OUT17.ZIPFIP17;RUN;
PROC CONTENTS DATA=OUT16.ZIPFIP16;RUN;
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<DPHELPS@MATHEMATICA-MPR.COM
RCPT TO:<DPHELPS@MATHEMATICA-MPR.COM
DATA
FROM: <DPHELPS@MATHEMATICA-MPR.COM
TO: <DPHELPS@MATHEMATICA-MPR.COM
SUBJECT: DAF21 JCLYRSP IS COMPLETE
```



## Appendix A.324

JCL/SAS Code: STATEMR

```

//$4671STM JOB (12510000,T715,,SAS,,ITC9FL),PHELPS,
//      MSGCLASS=1,MSGLEVEL=(1,1),REGION=2047M,NOTIFY=$4671
//*****
//*
//* *-----*
//* | OPDR.TG.PRD.ETTW.$4671.DAF21.ZIPFIP.PRDLIB(ZIPMATCH)|
//* *-----*
//*
//* PURPOSE: ADDITIONAL ANALYSIS TO SEE PERCENTAGE OF ZIP9 MATCHING
//*      RATES FOR SSI AND SSDI BENES OVER TIME
//* PROGRAMMER: BRITTANY COLIP (BCOLIP@MATHEMATICA-MPR.COM)
//* DATE: 8/10/2022
//*****
//*
//JS010 EXEC SAS9,
//      WORK='120000,60000',
//      PARM='SYSPARM="'&YYYY"'
//*
//* INPUT FILE
//IN   DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.ZIPALL.SA.V1,DISP=SHR
//FIN  DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.DMSTW.SA.V1,DISP=SHR
//* OUTPUT FILE
//OUT  DD DSN=OPDR.TG.PRD.ETTW.$4671.DAF21I.PSTM.R.SA.V2,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MACROGEN MPRINT OBS=MAX;

%MACRO DOIT;
  * SET MACRO VARS;

  %LET BEGYR = 2016;
  %LET ENDYR = 2021;
  %LET ENDMN = 12;

  * INCLUDING THE YMM MACRO FROM THE UTILITY LIBRARY;
  %INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF21.UTILITY.PRDLIB(YMMMACR)";

DATA OUT.PSTM.R;
  MERGE IN.ZIPALL(IN=A KEEP=SSN STATE1512 PST:)
        FIN.DMSTW(IN=B KEEP=SSN STATE_MR
                  RENAME=(STATE_MR=STATE_MR_DM))
  ;

```

Appendix A.324

JCL/SAS Code: STATEMR

```
BY SSN;

ARRAY PST (*) $ STATE1512
      %DO MM=1 %TO &TOTMOS.; PST&&YY&MM %END;;

/* CREATE SINGLE STATE VARIABLE PST_MR - NEW FOR DAF16 */
LENGTH STATE_MR $2.;
DO I=DIM(PST) TO 1 BY -1 UNTIL (NOT MISSING(STATE_MR));
  IF NOT MISSING(PST(I)) AND PST(I) NE "--"
    THEN STATE_MR=PST(I);
END;
IF STATE_MR=" " THEN STATE_MR=STATE_MR_DM;

RUN;

PROC CONTENTS DATA=OUT.PSTMTR;
RUN;

%MEND DOIT;
%DOIT;
```

## Appendix A.325

JCL/SAS Code: COPYTKT

```
//#6502CTA JOB (12510000,T715,,SAS,,ITC9FL),BRYCE-ALEX,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DLVR.PRDLIB(COPYTKT)
//* DESCRIPTION: COPY BASE AND ANNUAL TICKET FILES FOR DELIVERY
//* DATE:      12/21/2022
//* PROGRAMMER: ALEX BRYCE
//*           ABRYCE@MATHEMATICA-MPR.COM
//*
//*****
//* COPY TICKET21
//Y2021 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET21,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET21,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2021C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET21,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET21,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2021D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT21,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT21,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET20
//Y2020 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET20,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET20,
//      DISP=(NEW,CATLG,DELETE),
```

## Appendix A.325

JCL/SAS Code: COPYTKT

```
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2020C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET20,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET20,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2020D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT20,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT20,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET19
//Y2019 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET19,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET19,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2019C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET19,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET19,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2019D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT19,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT19,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET18
//Y2018 EXEC PGM=IEBGENER
```

## Appendix A.325

JCL/SAS Code: COPYTKT

```
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET18,DISP=SHR,
//    UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET18,
//    DISP=(NEW,CATLG,DELETE),
//    UNIT=TSILO
//SYSIN DD DUMMY
//Y2018C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET18,DISP=SHR,
//    UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET18,
//    DISP=(NEW,CATLG,DELETE),
//    UNIT=TSILO
//SYSIN DD DUMMY
//Y2018D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT18,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT18,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//    SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET17
//Y2017 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET17,DISP=SHR,
//    UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET17,
//    DISP=(NEW,CATLG,DELETE),
//    UNIT=TSILO
//SYSIN DD DUMMY
//Y2017C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET17,DISP=SHR,
//    UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET17,
//    DISP=(NEW,CATLG,DELETE),
//    UNIT=TSILO
//SYSIN DD DUMMY
//Y2017D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT17,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT17,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,10),
```

## Appendix A.325

JCL/SAS Code: COPYTKT

```
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET16
//Y2016 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET16,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET16,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2016C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET16,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET16,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2016D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT16,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT16,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET15
//Y2015 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET15,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET15,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2015C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET15,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET15,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
```

## Appendix A.325

JCL/SAS Code: COPYTKT

```
//Y2015D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT15,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT15,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//    SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET14
//Y2014 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET14,DISP=SHR,
//    UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET14,
//    DISP=(NEW,CATLG,DELETE),
//    UNIT=TSILO
//SYSIN DD DUMMY
//Y2014C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET14,DISP=SHR,
//    UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET14,
//    DISP=(NEW,CATLG,DELETE),
//    UNIT=TSILO
//SYSIN DD DUMMY
//Y2014D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT14,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT14,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//    SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET13
//Y2013 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET13,DISP=SHR,
//    UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET13,
//    DISP=(NEW,CATLG,DELETE),
//    UNIT=TSILO
//SYSIN DD DUMMY
//Y2013C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET13,DISP=SHR,
```

## Appendix A.325

JCL/SAS Code: COPYTKT

```
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET13,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2013D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT13,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT13,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET12
//Y2012 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET12,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET12,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2012C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET12,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET12,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2012D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT12,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT12,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET11
//Y2011 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET11,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET11,
//      DISP=(NEW,CATLG,DELETE),
```



## Appendix A.325

JCL/SAS Code: COPYTKT

```
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2011C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET11,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET11,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2011D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT11,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT11,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET10
//Y2010 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET10,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET10,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2010C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET10,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET10,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2010D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT10,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT10,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET09
//Y2009 EXEC PGM=IEBGENER
```

## Appendix A.325

JCL/SAS Code: COPYTKT

```
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET09,DISP=SHR,
// UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET09,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO
//SYSIN DD DUMMY
//Y2009C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET09,DISP=SHR,
// UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET09,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO
//SYSIN DD DUMMY
//Y2009D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT09,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT09,
// DISP=(NEW,CATLG,DELETE),VOL=(,,10),
// SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET08
//Y2008 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET08,DISP=SHR,
// UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET08,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO
//SYSIN DD DUMMY
//Y2008C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET08,DISP=SHR,
// UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET08,
// DISP=(NEW,CATLG,DELETE),
// UNIT=TSILO
//SYSIN DD DUMMY
//Y2008D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT08,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT08,
// DISP=(NEW,CATLG,DELETE),VOL=(,,10),
```

## Appendix A.325

JCL/SAS Code: COPYTKT

```
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET07
//Y2007 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET07,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET07,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2007C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET07,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET07,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2007D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT07,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT07,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET06
//Y2006 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET06,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET06,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2006C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET06,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET06,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
```

## Appendix A.325

JCL/SAS Code: COPYTKT

```
//Y2006D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT06,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT06,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//    SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET05
//Y2005 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET05,DISP=SHR,
//    UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET05,
//    DISP=(NEW,CATLG,DELETE),
//    UNIT=TSILO
//SYSIN DD DUMMY
//Y2005C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET05,DISP=SHR,
//    UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET05,
//    DISP=(NEW,CATLG,DELETE),
//    UNIT=TSILO
//SYSIN DD DUMMY
//Y2005D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT05,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT05,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//    SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET04
//Y2004 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET04,DISP=SHR,
//    UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET04,
//    DISP=(NEW,CATLG,DELETE),
//    UNIT=TSILO
//SYSIN DD DUMMY
//Y2004C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET04,DISP=SHR,
```

## Appendix A.325

JCL/SAS Code: COPYTKT

```
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET04,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2004D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT04,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT04,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET03
//Y2003 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET03,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET03,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2003C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET03,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET03,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2003D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT03,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT03,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY TICKET02
//Y2002 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET02,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKET02,
//      DISP=(NEW,CATLG,DELETE),
```

## Appendix A.325

JCL/SAS Code: COPYTKT

```
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2002C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKET02,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKET02,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//Y2002D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKT02,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKT02,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//* COPY BASE TICKET
//TKTBS EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKETBS,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.TICKETBS,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//TKTBSC EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.TICKETBS,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.TICKETBS,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//TKTBSD EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10TKTBS,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10TKTBS,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
```

Appendix A.325

JCL/SAS Code: COPYTKT

```
//*****  
//SENDNOTE EXEC PGM=IEBGENER  
//SYSIN DD DUMMY  
//SYSPRINT DD SYSOUT=*  
//SYSUT2 DD SYSOUT=(B,SMTP)  
//SYSUT1 DD *  
HELO MVSHOST  
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>  
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>  
DATA  
FROM: ABRYCE@MATHEMATICA-MPR.COM  
TO: ABRYCE@MATHEMATICA-MPR.COM  
SUBJECT: COPYTKT
```

COPYTKT HAS FINISHED RUNNING

## Appendix A.326

JCL/SAS Code: CPYENPAY

```
//#6502CPD JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.DLVR.PRDLIB(COPYDMG)
//* DESCRIPTION: COPY EN PAY FILES FOR DELIVERY
//* DATE:      12/07/2021
//* PROGRAMMER: ALEX BRYCE
//*           ABRYCE@MATHEMATICA-MPR.COM
//*
//*****
//* FULL ENPAY HORTIZONAL TO FINAL
//ENPAY EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.ENPAY,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.ENPAY,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//* FULL ENPAY HORTIZONAL TO COPY
//ENPAYC EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.ENPAY,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.ENPAY,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//* 10% ENPAY TO FINAL
//ENPAYD EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10PAYH,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10PAYH,
//      DISP=(MOD,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//* FULL ENPAY VERTICAL TO FINAL
//ENPAV EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.ENPAYVER,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.ENPAYVER,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
```



Appendix A.326

JCL/SAS Code: CPYENPAY

```
//* FULL ENPAY VERTICAL TO COPY
//ENPAVC EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.ENPAYVER,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.ENPAYVER,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//    SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//* 10% ENPAY VERTICAL TO FINAL
//ENPAYD EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10PAYV,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10PAYV,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//    SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: ABRYCE@MATHEMATICA-MPR.COM
TO: ABRYCE@MATHEMATICA-MPR.COM
SUBJECT: CPYENPAY

CPYENPAY HAS FINISHED RUNNING
```

## Appendix A.327

JCL/SAS Code: COPYDMG

```
//#6502CPD JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DLVR.PRDLIB(COPYDMG)
//* DESCRIPTION: COPY DMG FILES FOR DELIVERY
//* DATE:      12/21/2022
//* PROGRAMMER: ALEX BRYCE
//*           ABRYCE@MATHEMATICA-MPR.COM
//*
//*****
//* FULL DMG FILE DRAFT TO FINAL
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.DMG,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.DMG,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//* FULL DMG FILE DRAFT TO COPY
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.DMG,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.DMG,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//* 10% DMG FILE
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10DMG,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10DMG,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
```

Appendix A.327

JCL/SAS Code: COPYDMG

```
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: ABRYCE@MATHEMATICA-MPR.COM
TO: ABRYCE@MATHEMATICA-MPR.COM
SUBJECT: COPYDMG
```

COPYDMG HAS FINISHED RUNNING

## Appendix A.328

JCL/SAS Code: COPYANN

```
//#6502CYA JOB (12510000,T715,,SAS,,ITC9FL),BRYCE-ALEX,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DLVR.PRDLIB(COPYANN)
//* DESCRIPTION: COPY ANNUAL FILES FOR DELIVERY
//* DATE:     12/21/2022
//* PROGRAMMER: ALEX BRYCE
//*          ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//* COPY 1994
//*STEP1994 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1994E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1994E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP94C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1994E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y1994E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP94D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y1994E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y1994E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*
//* COPY 1995
//*STEP1995 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1995E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1995E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP95C EXEC PGM=IEBGENER
```

## Appendix A.328

JCL/SAS Code: COPYANN

```
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1995E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y1995E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP95D EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y1995E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y1995E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*
/* COPY 1996
/*STEP1996 EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1996E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1996E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP96C EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1996E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y1996E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP96D EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y1996E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y1996E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*
/* COPY 1997
/*STEP1997 EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1997E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1997E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
```

## Appendix A.328

JCL/SAS Code: COPYANN

```
/*STEP97C EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1997E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y1997E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP97D EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y1997E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y1997E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*
/* COPY 1998
/*STEP1998 EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1998E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1998E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP98C EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1998E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y1998E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP98D EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y1998E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y1998E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*
/* COPY 1999
/*STEP1999 EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1999E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y1999E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
```

## Appendix A.328

JCL/SAS Code: COPYANN

```
//*SYSIN DD DUMMY
//*STEP99C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y1999E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y1999E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/* UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP99D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y1999E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y1999E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/* UNIT=TSILO
//*SYSIN DD DUMMY
/*
/* COPY 2000
//*STEP2000 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2000E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2000E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/* UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP00C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2000E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2000E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/* UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP00D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2000E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2000E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/* UNIT=TSILO
//*SYSIN DD DUMMY
/*
/* COPY 2001
//*STEP2001 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2001E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2001E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
```

## Appendix A.328

JCL/SAS Code: COPYANN

```
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP01C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2001E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2001E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP01D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2001E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2001E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*
//* COPY 2002
//*STEP2002 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2002E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2002E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP02C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2002E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2002E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP02D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2002E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2002E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*
//* COPY 2003
//*STEP2003 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2003E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2003E,
```



## Appendix A.328

JCL/SAS Code: COPYANN

```
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP03C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2003E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2003E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP03D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2003E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2003E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*
//* COPY 2004
//*STEP2004 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2004E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2004E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP04C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2004E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2004E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP04D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2004E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2004E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*
//* COPY 2005
//*STEP2005 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2005E,DISP=SHR
```

## Appendix A.328

JCL/SAS Code: COPYANN

```
/**SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2005E,
/**      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/**      UNIT=TSILO
/**SYSIN DD DUMMY
/**STEP05C EXEC PGM=IEBGENER
/**SYSPRINT DD SYSOUT=*
/**SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2005E,DISP=SHR
/**SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2005E,
/**      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/**      UNIT=TSILO
/**SYSIN DD DUMMY
/**STEP05D EXEC PGM=IEBGENER
/**SYSPRINT DD SYSOUT=*
/**SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2005E,DISP=SHR
/**SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2005E,
/**      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/**      UNIT=TSILO
/**SYSIN DD DUMMY
/**
/** COPY 2006
/**STEP2006 EXEC PGM=IEBGENER
/**SYSPRINT DD SYSOUT=*
/**SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2006E,DISP=SHR
/**SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2006E,
/**      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/**      UNIT=TSILO
/**SYSIN DD DUMMY
/**STEP06C EXEC PGM=IEBGENER
/**SYSPRINT DD SYSOUT=*
/**SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2006E,DISP=SHR
/**SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2006E,
/**      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/**      UNIT=TSILO
/**SYSIN DD DUMMY
/**STEP06D EXEC PGM=IEBGENER
/**SYSPRINT DD SYSOUT=*
/**SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2006E,DISP=SHR
/**SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2006E,
/**      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/**      UNIT=TSILO
/**SYSIN DD DUMMY
/**
/** COPY 2007
/**STEP2007 EXEC PGM=IEBGENER
/**SYSPRINT DD SYSOUT=*
```

## Appendix A.328

JCL/SAS Code: COPYANN

```
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2007E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2007E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP07C EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2007E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2007E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP07D EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2007E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2007E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*
/* COPY 2008
/*STEP2008 EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2008E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2008E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP08C EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2008E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2008E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP08D EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2008E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2008E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/*      UNIT=TSILO
/*SYSIN DD DUMMY
/*
/* COPY 2009
/*STEP2009 EXEC PGM=IEBGENER
```

## Appendix A.328

JCL/SAS Code: COPYANN

```
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2009E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2009E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP09C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2009E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2009E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP09D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2009E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2009E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
/*
/* COPY 2010
/*STEP2010 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2010E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2010E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
/*STEP10C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2010E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2010E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
/*STEP10D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2010E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2010E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
/*
/* COPY 2011
```

## Appendix A.328

JCL/SAS Code: COPYANN

```
/*STEP2011 EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2011E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2011E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/* UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP11C EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2011E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2011E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/* UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP11D EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2011E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2011E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/* UNIT=TSILO
/*SYSIN DD DUMMY
/*
/* COPY 2012
/*STEP2012 EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2012E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2012E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/* UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP12C EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2012E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2012E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/* UNIT=TSILO
/*SYSIN DD DUMMY
/*STEP12D EXEC PGM=IEBGENER
/*SYSPRINT DD SYSOUT=*
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2012E,DISP=SHR
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2012E,
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/* UNIT=TSILO
/*SYSIN DD DUMMY
/*
```

## Appendix A.328

JCL/SAS Code: COPYANN

```
//* COPY 2013
//*STEP2013 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2013E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2013E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP13C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2013E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2013E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP13D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2013E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2013E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
/*
/* COPY 2014
//*STEP2014 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2014E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2014E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP14C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2014E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2014E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP14D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2014E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2014E,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
/*      UNIT=TSILO
//*SYSIN DD DUMMY
```

## Appendix A.328

JCL/SAS Code: COPYANN

```
//*  
/* COPY 2015  
/*STEP2015 EXEC PGM=IEBGENER  
/*SYSPRINT DD SYSOUT=*  
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2015E,DISP=SHR  
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2015E,  
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),  
/* UNIT=TSILO  
/*SYSIN DD DUMMY  
/*STEP15C EXEC PGM=IEBGENER  
/*SYSPRINT DD SYSOUT=*  
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2015E,DISP=SHR  
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2015E,  
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),  
/* UNIT=TSILO  
/*SYSIN DD DUMMY  
/*STEP15D EXEC PGM=IEBGENER  
/*SYSPRINT DD SYSOUT=*  
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2015E,DISP=SHR  
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2015E,  
/* DISP=(NEW,CATLG,DELETE),VOL=(,,,10),  
/* UNIT=TSILO  
/*SYSIN DD DUMMY  
/*  
/* COPY 2016  
/*STEP2016 EXEC PGM=IEBGENER  
/*SYSPRINT DD SYSOUT=*  
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2016E,DISP=SHR  
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2016E,  
// DISP=(NEW,CATLG,DELETE),VOL=(,,,10),  
// UNIT=TSILO  
/*SYSIN DD DUMMY  
/*STEP16C EXEC PGM=IEBGENER  
/*SYSPRINT DD SYSOUT=*  
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2016E,DISP=SHR  
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2016E,  
// DISP=(NEW,CATLG,DELETE),VOL=(,,,10),  
// UNIT=TSILO  
/*SYSIN DD DUMMY  
/*STEP16D EXEC PGM=IEBGENER  
/*SYSPRINT DD SYSOUT=*  
/*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2016E,DISP=SHR  
/*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2016E,  
// DISP=(NEW,CATLG,DELETE),VOL=(,,,10),  
// UNIT=TSILO
```

## Appendix A.328

JCL/SAS Code: COPYANN

```
//SYSIN DD DUMMY
//*
//* COPY 2017
//STEP2017 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2017E,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2017E,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
//    UNIT=TSILO
//SYSIN DD DUMMY
//STEP17C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2017E,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2017E,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
//    UNIT=TSILO
//SYSIN DD DUMMY
//STEP17D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2017E,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2017E,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
//    UNIT=TSILO
//SYSIN DD DUMMY
//*
//* COPY 2018
//STEP2018 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2018E,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2018E,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
//    UNIT=TSILO
//SYSIN DD DUMMY
//STEP18C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2018E,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2018E,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
//    UNIT=TSILO
//SYSIN DD DUMMY
//STEP18D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2018E,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2018E,
//    DISP=(NEW,CATLG,DELETE),VOL=(,,,10),
```



## Appendix A.328

JCL/SAS Code: COPYANN

```
//      UNIT=TSILO
//SYSIN DD DUMMY
//*
//* COPY 2019
//STEP2019 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2019E,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2019E,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//SYSIN DD DUMMY
//STEP19C EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2019E,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2019E,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//SYSIN DD DUMMY
//STEP19D EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2019E,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2019E,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//SYSIN DD DUMMY
//*
//* COPY 2020
//*STEP2020 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2020E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2020E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP20C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2020E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2020E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP20D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2020E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2020E,
```

Appendix A.328

JCL/SAS Code: COPYANN

```
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*
//* COPY 2021
//*STEP2021 EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2021E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y2021E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP21C EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.Y2021E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.Y2021E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*STEP20D EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.Y2021E,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.Y2021E,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      UNIT=TSILO
//*SYSIN DD DUMMY
//*
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: ABRYCE@MATHEMATICA-MPR.COM
TO: ABRYCE@MATHEMATICA-MPR.COM
SUBJECT: COPYANN
```

COPYANN HAS FINISHED RUNNING

## Appendix A.329

JCL/SAS Code: CPYVRRMS

```
//#6502CPD JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DLVR.PRDLIB(CPYVRRMS)
//* DESCRIPTION: COPY VRRMS FILES FOR DELIVERY
//* DATE:      01/04/2023
//* PROGRAMMER: ALEX BRYCE
//*           ABRYCE@MATHEMATICA-MPR.COM
//*
//*****
//* FULL VRRMS FILE FROM DRAFT TO FINAL
//VRRMS EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.VRRMS,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.VRRMS,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//* FULL VRRMS FILE FROM DRAFT TO COPY
//VRRMSC EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.VRRMS,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.VRRMS,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//* 10% VRRMS FILE
//STEPS EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10VRRMS,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10VRRMS,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//* HISTORICAL VRRMS FROM DRAFT TO FINAL
//VRRMSH EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.VRRMS.HIST,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.VRRMS.HIST,
//*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//*      SPACE=(CYL,(3000,3000),RLSE)
//*SYSIN DD DUMMY
```

Appendix A.329

JCL/SAS Code: CPYVRRMS

```
//* HISTORICAL VRRMS FROM DRAFT TO COPY
//*VRRMSHC EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.VRRMS.HIST,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21C.VRRMS.HIST,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/*      SPACE=(CYL,(3000,3000),RLSE)
//*SYSIN DD DUMMY
/* 10% HISTORICAL VRRMS
//*VRRMSHD EXEC PGM=IEBGENER
//*SYSPRINT DD SYSOUT=*
//*SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21D.P10VRRMS.HIST,DISP=SHR
//*SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.P10VRRMS.HIST,
/*      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
/*      SPACE=(CYL,(3000,3000),RLSE)
//*SYSIN DD DUMMY
/*
/******
/* SENDING EMAIL FROM HOST USING IEBGENER
/******
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: ABRYCE@MATHEMATICA-MPR.COM
TO: ABRYCE@MATHEMATICA-MPR.COM
SUBJECT: CPYVRRMS

CPYVRRMS HAS FINISHED RUNNING
```

## Appendix A.330

JCL/SAS Code: CPYSCFLS

```
//#6502CPD JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.DLVR.PRDLIB(CPYSCFLS)
//* DESCRIPTION: COPY SCWF AND SCDR FILES FOR DELIVERY
//* DATE:      01/04/2023
//* PROGRAMMER: ALEX BRYCE
//*           ABRYCE@MATHEMATICA-MPR.COM
//*
//*****
//* FULL SCWF FILE TO FINAL
//*SCWF EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.SCWF,DISP=SHR,
//*      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.SCWF,
//*      DISP=(NEW,CATLG,DELETE),
//*      UNIT=TSILO
//SYSIN DD DUMMY
//* FULL SCDR FILE DRAFT TO FINAL
//SCDR EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.SCDR,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.SCDR,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,99),
//      SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DUMMY
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: ABRYCE@MATHEMATICA-MPR.COM
TO: ABRYCE@MATHEMATICA-MPR.COM
```

Appendix A.330

JCL/SAS Code: CPYSCFLS

SUBJECT: CPYSCFLS

CPYSCFLS HAS FINISHED RUNNING

Appendix A.331

JCL/SAS Code: COPYADM

```
//#6502CPY JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF21.FOF.PRDLIB(COPYADM)
//* DESCRIPTION: COPY ADM FILE FOR DELIVERY
//* DATE:      02/28/2022
//* PROGRAMMER: ALEX BRYCE
//*           ABRYCE@MATHEMATICA-MPR.COM
//*
//*****
//* FULL ADM FILE TO FINAL
//SCWF EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.AWARDEE,DISP=SHR,
//      UNIT=TSILO
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21D.AWARDEE,
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=TSILO
//SYSIN DD DUMMY
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
DATA
FROM: ABRYCE@MATHEMATICA-MPR.COM
TO: ABRYCE@MATHEMATICA-MPR.COM
SUBJECT: COPYADM
```

COPYADM HAS FINISHED RUNNING

## **Appendix B:**

### **ADM Code**



Appendix B.1

JCL/SAS Code: ADMBENES

```
//#8047ADM JOB (12510000,T715,,SAS,,ITC9FL),BRYCE,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#6502
//*****
//* PROJECT:   DAF 50214(06979)
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ADM.PRDLIB(ADMBENES)
//* DESCRIPTION: AWARDEE DATA MART BENEFICIARY SELECTION
//* DATE:     11/14/13 SVETLANA BRONNIKOV
//* UPDATED:  1/7/15 SVETLANA BRONNIKOV
//* UPDATED:  11/17/2015 JOE MASTRIANNI
//* UPDATED:  12/15/2016 MIAOMIAO SHEN
//* UPDATED:  3/2/2017 MIAOMIAO SHEN USE V2 DEMO
//* UPDATED:  1/04/2021 SERGE LUKASHANETS
//* UPDATED:  8/12/2021 ALEX BRYCE (TESTING ADDING CHILD SSI BENES)
//* UPDATED:  12/8/2021 ALEX BRYCE UPDATING FOR DAF20
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//DEM  DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.DMG,DISP=SHR
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.ADMBENES.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
//SYSIN DD *
```

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

%LET ENDYR=2021; /\* CHANGE AS NEEDED \*/

\* DEFINE THE INITIAL POPULATION BEFORE FLAGGING SSI AND DI BENES;

%MACRO START;

DATA TEMP.ADMBENES(KEEP=SSN DOBBEST DODBEST FRA DOEI DOEC KEEP:

    BIC SSIELIG\_ADULT MINFRSTP RID: RDT:

    AGESTART AGEEND AGESTART94

    DODEC: FIRSTDODEC AGE18\_CDR SEX

    SSIELIG\_FIRST);

SET DEM.DMG;

\* IF DOEI NOT MISSING OR SSIELIG\_FIRST NOT MISSING \*;

IF DOEI NE . OR SSIELIG\_FIRST NE .;

## Appendix B.1

JCL/SAS Code: ADMBENES

```
* CREATE FLAGS BASED ON AGE ;
***CALCULATE AGE***;
AGESTART = INTCK('MONTH',DOBBEST,'01JAN1996'd,'c')/12;
AGESTART94 = INTCK('MONTH',DOBBEST,'01JAN1994'd,'c')/12;
AGEEND=INTCK('MONTH',DOBBEST,"31DEC&ENDYR."d,'c')/12;

IF /*AGEEND>=18 AND */ AGESTART<FRA THEN KEEPAGE = 1;
ELSE KEEPAGE = 0;

*** KEEP IF NOT DEAD AT START OF YEAR 1996 ***;
IF YEAR(DODBEST) = . OR DODBEST > '01JAN1996'D THEN KEEPYRDOD=1;
ELSE KEEPYRDOD = 0;

*** IDENTIFY ELIGIBLE BENEFICIARIES;
IF KEEPAGE = 1 AND KEEPYRDOD = 1 THEN KEEP=1;
ELSE KEEP = 0;

RUN;
%MEND START;
%START;
TITLE "CHECK ON KEEP VARIABLES";
PROC FREQ DATA=TEMP.ADMBENES;
  TABLES KEEP * KEEPAGE * KEEPYRDOD / LIST MISSING;
RUN;

DATA OUT.ADMBENES(DROP = AGESTART AGEEND AGESTART94)
;
SET TEMP.ADMBENES;
IF KEEP = 1;
DROP KEEP;;
* CREATE FLAGS FOR SSDI AND SSI BENES;
* INITIALIZE THESE FLAGS TO 0;
ADM_SSDI = 0;
ADM_SSI = 0;

***FLAG SSI BENES WHERE MIN_SSI_ELG IS
NOT MISSING AND MINFRSTP IS ON OR AFTER
JANUARY 1,1996***;

IF (SSIELIG_FIRST NE .) AND (MINFRSTP >='01JAN1996'D)
THEN ADM_SSI = 1;

***FLAG DI BENES WHERE DOEI IS NOT MISSING
EXCLUDE WHO HAVE DOEI BEFORE 1994 AND
18 BIRTHDAY BEFORE 1994***;
```

## Appendix B.1

JCL/SAS Code: ADMBENES

```
IF DOEI NE . AND (DOEI>='01JAN1994'D OR AGESTART94<=18)
THEN DO;
  /*ADDITIONAL CRITERIA FOR FLAGGING DI BENES*/
  REFDATE = '01JAN1996'D;
  DOECFLAG = 1;
  FRAFLAG = 1;
  /* FLAG THE RESTRICTIONS FOR THE BENEFICIARIES */
  IF DOEC LT REFDATE AND DOEC GT DOEI THEN DOECFLAG = 0;
  /* ESTABLISH AGE AT DOEI DATE FROM THE END OF THE DOEI
  MONTH AND COMPARE TO AGE AT RETIRMENT */
  NEWDOEI = INTNX('MONTH',DOEI,0,'END');
  AGEDOEI = INTCK('MONTHS',DOBBEST,NEWDOEI,'C')/12;
  IF AGEDOEI GE FRA THEN FRAFLAG = 0;

  /* FLAG THE BENE IF DI CRITERIA IS MET */
  IF DOECFLAG AND FRAFLAG THEN ADM_SSDI = 1;
END;

/*KEEP ONLY THE OBS THAT HAVE BEEN FLAGGED*/
IF ADM_SSI = 1 OR ADM_SSDI = 1;
RUN;

TITLE "CONTENTS OF ADMBENES DATA";
PROC CONTENTS DATA = OUT.ADMBENES;
RUN;

TITLE "GET FREQS OF SSDI AND SSI BENES";
PROC FREQ DATA = OUT.ADMBENES;
  TABLES ADM_SSDI*ADM_SSI/MISSING LIST;
RUN;

TITLE "ADMBENES DATA (SSDI POPULATION)";
PROC FREQ DATA=OUT.ADMBENES (WHERE = (ADM_SSDI = 1));
  TABLES DOEI / MISSING;
  FORMAT DOEI YYMMN6.;
RUN;

TITLE 'EXAMINE DISTRIBUTION OF THE FLAGS (SSDI POPULATION)';
PROC PRINT DATA=OUT.ADMBENES (WHERE = (ADM_SSDI = 1) OBS = 50);
  FORMAT DOEI NEWDOEI DOBBEST DODBEST DOEC YYMMDD10.;
RUN;

PROC FREQ DATA=OUT.ADMBENES (WHERE = (ADM_SSDI = 1));
  TABLES DOECFLAG * FRAFLAG / LIST MISSING;
RUN;
```

Appendix B.1

JCL/SAS Code: ADMBENES

```
TITLE "ADMBENES DATA (SSI POPULATION)";
PROC FREQ DATA=OUT.ADMBENES (WHERE = (ADM_SSI = 1));
  TABLES SSIELIG_FIRST MINFRSTP / MISSING;
  FORMAT SSIELIG_FIRST MINFRSTP YMMMN6.;
RUN;
```

```
TITLE;
```

```
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: ADMBENES IS COMPLETE
DAF21 ADMBENES IS COMPLETE.
```

Appendix B.2

JCL/SAS Code: JCLADMAN

```

//#8047ADD JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214(06979)
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#8047.DAF21.ADM.PRDLIB(JCLADMAN)
//* DESCRIPTION:
//* DATE:      12/07/15
//* PROGRAMMER: JOE MASTRIANNI
//* UPDATE:    01/05/2017 MIAOMIAO SHEN
//* UPDATE:    04/18/2018 SERGE LUKASHANETS
//* UPDATE:    05/09/2018 DAWN PHELPS
//* UPDATE:    01/10/2019 BRITTANY COLIP
//*           (CONSOLIDATING OLDER PROGRAMS)
//* UPDATE:    08/26/2021 ALEX BRYCE (ADDING SSI KIDS)
//* UPDATE:    12/08/2021 ALEX BRYCE (UPDATING FOR DAF20)
//* EMAIL:     ALEX.BRYCE@SSA.GOV
//*           ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//MRGPROC PROC
//STEP1 EXEC SAS9,
//  WORK='120000,60000',
//  PARM='SYSPARM="'&YYYY'"
//*
//IN1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.ADMBENES.SA.V1,DISP=SHR
//ANN DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.Y&YYYY.E,DISP=SHR
//OUT DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.ADMY&YYYY..SA.V1,
//  DISP=(OLD,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//SYSIN DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21.ADM.PRDLIB(ADMANN),
//  DISP=(SHR,PASS,KEEP)
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA

```

Appendix B.2

JCL/SAS Code: JCLADMAN

```
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: JCLADMAN COMPLETE &YYYY.
// PEND
//*
//RUN28 EXEC MRGPROC,YYYY=2021
//RUN27 EXEC MRGPROC,YYYY=2020
//RUN26 EXEC MRGPROC,YYYY=2019
//RUN25 EXEC MRGPROC,YYYY=2018
//RUN24 EXEC MRGPROC,YYYY=2017
//RUN23 EXEC MRGPROC,YYYY=2016
//RUN22 EXEC MRGPROC,YYYY=2015
//RUN21 EXEC MRGPROC,YYYY=2014
//RUN20 EXEC MRGPROC,YYYY=2013
//RUN19 EXEC MRGPROC,YYYY=2012
//RUN18 EXEC MRGPROC,YYYY=2011
//RUN17 EXEC MRGPROC,YYYY=2010
//RUN16 EXEC MRGPROC,YYYY=2009
//RUN15 EXEC MRGPROC,YYYY=2008
//RUN14 EXEC MRGPROC,YYYY=2007
//RUN13 EXEC MRGPROC,YYYY=2006
//RUN12 EXEC MRGPROC,YYYY=2005
//RUN11 EXEC MRGPROC,YYYY=2004
//RUN10 EXEC MRGPROC,YYYY=2003
//RUN9 EXEC MRGPROC,YYYY=2002
//RUN8 EXEC MRGPROC,YYYY=2001
//RUN7 EXEC MRGPROC,YYYY=2000
//RUN6 EXEC MRGPROC,YYYY=1999
//RUN5 EXEC MRGPROC,YYYY=1998
//RUN4 EXEC MRGPROC,YYYY=1997
//RUN3 EXEC MRGPROC,YYYY=1996
//RUN2 EXEC MRGPROC,YYYY=1995
//RUN1 EXEC MRGPROC,YYYY=1994
//*
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<ABRYCE@MATHEMATICA-MPR.COM>
RCPT TO:<ABRYCE@MATHEMATICA-MPR.COM>
```

Appendix B.2

JCL/SAS Code: JCLADMAN

DATA

FROM: <ABRYCE@MATHEMATICA-MPR.COM>

TO: <ABRYCE@MATHEMATICA-MPR.COM>

SUBJECT: JCLADMAN COMPLETE 1994-2021

Appendix B.3

JCL/SAS Code: ADMANN

```
*****
* PROJECT: DAF 50214(06979)
* DIRECTOR: MATT URATO
* PROGRAM: OPDR.TG.PRD.ETTW.#7429.DAF17.ADM.PRDLIB(ADMANN)
* DESCRIPTION: EXTRACT ANNUAL VARIABLES FOR ADM BENES
* THE FORMER SSIANN AND DIANN INCLUDE FILES ARE
* BEING CONSOLIDATED INTO THIS PROGRAM
* DATE: 01/12/2022
* PROGRAMMER: ALEX BRYCE
* EMAIL: ABRYCE@MATHEMATICA-MPR.COM
*****;
```

```
OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX DKRICOND=WARN;
```

```
%MACRO DOIT;
```

```
DATA OUT.ADMY&SYSPARM.;
MERGE IN1.ADMBENES(IN=ADM)
ANN.Y&SYSPARM.(KEEP=SSN /*BOTH SSI AND SSDI*/
PSTA: /*SSI ONLY*/
PAYS: /*SSI ONLY*/
DUES: /*SSI ONLY*/
LAF: /*BOTH SSI AND SSDI*/
PAYD: /*BOTH SSI AND SSDI*/
DUED: /*BOTH SSI AND SSDI*/
%IF &SYSPARM. > 1995 %THEN %DO;
STATE: /*BOTH SSI AND SSDI FROM 1996 ON*/
%END;
);
```

```
BY SSN;
IF ADM;
```

```
RUN;
```

```
* IF THE BENE IS SSDI ONLY, SET THE SSI VARS TO MISSING;
* PAYS AND DUES ARE NUMERIC ACCORDING TO
* THE ANNUALS PROC CONTENTS ROLLUP;
* PSTA IS CHARACTER ACCORDING TO PROC CONTENTS ROLLUP;
```

```
DATA OUT.ADMY&SYSPARM.;
SET OUT.ADMY&SYSPARM.;
ARRAY PAYS (*) PAYS; ;
ARRAY DUES (*) DUES; ;
ARRAY PSTA (*) PSTA; ;
DO I = 1 TO DIM(PAYS);
IF ADM_SSDI = 1 AND ADM_SSI = 0 THEN PAYS(I) = .;
```



Appendix B.3

JCL/SAS Code: ADMANN

```
END;
DO I = 1 TO DIM(DUES);
  IF ADM_SSDI = 1 AND ADM_SSI = 0 THEN DUES(I) = .;
END;
DO I = 1 TO DIM(PSTA);
  IF ADM_SSDI = 1 AND ADM_SSI = 0 THEN PSTA(I) = "";
END;
DROP I;
RUN;

TITLE "CHECK MERGED &SYSPARM FILE (OBS=10)";
PROC PRINT DATA=OUT.ADMY&SYSPARM.(OBS=10);
RUN;

%MEND DOIT;
%DOIT;
```

## Appendix B.4

JCL/SAS Code: COMBANN

```

//#8047CMA JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214(06979)
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ADM.PRDLIB(COMBANN)
//* DESCRIPTION: COMBINE ALL SSI DATA FOR ANALYSIS
//* DATE:      05/29/12
//* PROGRAMMER: DAWN PHELPS
//* UPDATE:    12/09/15 JOE MASTRIANNI
//* UPDATE:    12/15/2016 MIAOMIAO SHEN
//* UPDATE:    01/05/2021 SERGE LUKASHANETS
//* UPDATE:    08/27/2021 ALEX BRYCE (ADDING SSI KIDS)
//* UPDATE:    12/08/2021 ALEX BRYCE DAF20
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//BENE DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.ADMBENES.SA.V1,DISP=SHR
//SSI  DD DSN=OPDR.TG.PRD.ETTW.#6266.DAF21P.SSR2.SA.V1,DISP=SHR
//DEM  DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.DMG,DISP=SHR
//* SEE THE INPUT STATEMENT IN THE LIBNAME BELOW
//OUT  DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.COMBANN.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TEMP1 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//TEMP2 DD DSN=&&TEMPX,DISP=(NEW,DELETE,DELETE),VOL=(,,10),
//  UNIT=TSILO
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

%MACRO DOIT;

%DO YEAR=1994 %TO 2021;
LIBNAME IN&YEAR. "OPDR.TG.PRD.ETTW.#8047.DAF21P.ADMY&YEAR..SA.V1"
      DISP=SHR;
%END;

* THE VARIABLES TO KEEP FROM THE ANNUALS MERGE IN THE
  PREVIOUS PROGRAM. WE WILL CONDITION ON YEAR FOR STATE: VARIABLES;
%LET KEEPVARS = SSN PSTA: PAYS: DUES: LAF: PAYD: DUED: ;

```

## Appendix B.4

JCL/SAS Code: COMBANN

```

DATA TEMP1.COMBANN;
  MERGE BENE.ADMBENES(IN=BENE)
    SSI.LONG
    DEM.DMG(KEEP=SSN SSDI_ONSET)
    %DO YEAR=1994 %TO 2021;
      IN&YEAR..ADMY&YEAR.(KEEP= &KEEPVARS.
        %IF &YEAR. > 1995 %THEN %DO;
          STATE:
        %END;
      )
    %END;
  ;
  BY SSN;
  IF BENE;
RUN;

* IF THE BENE IS DI ONLY (ADM_SSDI = 1 AND ADM_SSI = 0), THEN
  WE NEED TO MAKE SURE THE SSI LONG VARS ARE SET TO . ;
* FIRST, GET THE NAMES OF VARS FROM SSI.LONG DATASET;
* WE NEED TO SEPARATE THE LIST INTO CHAR AND NUM VARS;
PROC SQL;
  * THIS GETS NUMERIC VARIABLES;
  SELECT NAME INTO: SSI_NUM_VARS SEPARATED BY " "
  FROM SASHELP.VCOLUMN
  WHERE (LIBNAME = "SSI"
    AND MEMNAME = "LONG"
    AND NAME ~= "SSN"
    AND UPCASE(TYPE) = "NUM");

  * THIS GETS CHARACTER VARIABLES;
  SELECT NAME INTO: SSI_CHAR_VARS SEPARATED BY " "
  FROM SASHELP.VCOLUMN
  WHERE (LIBNAME = "SSI"
    AND MEMNAME = "LONG"
    AND NAME ~= "SSN"
    AND UPCASE(TYPE) = "CHAR");
QUIT;

%PUT NUMERIC VARIABLES IN THE SSI.LONG DATASET: ;
%PUT &SSI_NUM_VARS;

%PUT CHARACTER VARIABLES IN THE SSI.LONG DATASET: ;
%PUT &SSI_CHAR_VARS;

```

## Appendix B.4

JCL/SAS Code: COMBANN

```
DATA OUT.COMBANN;
  SET TEMP1.COMBANN;
  ARRAY SSI_NUM_VARS (*) &SSI_NUM_VARS;
  ARRAY SSI_CHAR_VARS (*) &SSI_CHAR_VARS;

  * LOOP THROUGH THE NUM VARS;
  DO I = 1 TO DIM(SSI_NUM_VARS);
    IF ADM_SSDI = 1 AND ADM_SSI = 0
      THEN SSI_NUM_VARS(I) = . ;
  END;
  * LOOP THROUGH THE CHAR VARS;
  DO J = 1 TO DIM(SSI_CHAR_VARS);
    IF ADM_SSDI = 1 AND ADM_SSI = 0
      THEN SSI_CHAR_VARS(J) = "" ;
  END;

  * ALSO, IF BENE IS SSI ONLY, WE NEED TO
  SET SSDI_ONSET FROM THE DMG TO MISSING;
  IF ADM_SSDI = 0 AND ADM_SSI = 1
    THEN SSDI_ONSET = .;
  DROP I J;
RUN;

%MEND DOIT;
%DOIT;

ENDSAS;

//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: ADM COMBANN IS COMPLETE
```

## Appendix B.5

JCL/SAS Code: FINADM

```
//#8047ADM JOB (12510000,T715,,SAS,,ITC9FL),CHAPPELKA,
//  MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=#8047
//*****
//* PROJECT:   DAF 50214(06979)
//* DIRECTOR:  MATT URATO
//* PROGRAM:   OPDR.TG.PRD.ETTW.#6502.DAF20.ADM.PRDLIB(FINADM)
//* DESCRIPTION: CREATE THE FINAL ADM DATASET
//* DATE:      02/04/13 XIAO BARRY
//* UPDATED:   01/07/16 JOE MASTRIANNI
//* UPDATED:   03/19/2018 SERGE LUKASHANETS
//* UPDATED:   09/08/2021 ALEX BRYCE FOR SSIKIDS
//* UPDATED:   01/28/2022 ALEX BRYCE FOR DAF20
//* EMAIL:     ALEX.BRYCE@SSA.GOV
//*           ABRYCE@MATHEMATICA-MPR.COM
//*****
//*
//SAS  EXEC SAS9,
//  WORK='120000,60000'
//*
//DEMO DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF21P.DMG,DISP=SHR
//* ADM DATASET
//ADM  DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.COMBANN.SA.V1,DISP=SHR
//* SSDI DATASET
//OUT1 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.ADMDIAWD.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//* SSI ADULTS DATASET
//OUT2 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.ADMSIAWD.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//* SSI KIDS DATASET
//OUT3 DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.ADMSKAWD.SA.V1,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//* FINAL ADM OUTPUT DATASET
//OUT  DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF21P.AWARDEE,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  UNIT=TSILO
//* PROC CONTENTS FOR DOCUMENTATION
//CNT  DD DSN=OPDR.TG.PRD.ETTW.#8047.DAF21P.ADM.ADMCNTS,
//  DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//  SPACE=(CYL,(3000,3000),RLSE)
//TEMP1 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//  UNIT=TSILO,VOL=(,,10)
```

## Appendix B.5

JCL/SAS Code: FINADM

```
//TEMP2 DD DSN=&&TEMP,DISP=(NEW,DELETE,DELETE),
//    UNIT=TSILO,VOL=(,,10)
//SYSIN DD *

OPTIONS NOCENTER COMPRESS=BINARY MPRINT MACROGEN OBS=MAX;

%LET ENDYR=2021; /*CURRENT DAF YEAR*/

* FLAG THE DI BENES;
%INCLUDE "OPDR.TG.PRD.ETTW.#8047.DAF21.ADM.PRDLIB(DIFLAGS)";

* FLAG THE SSI ADULT BENES;
%INCLUDE "OPDR.TG.PRD.ETTW.#8047.DAF21.ADM.PRDLIB(SSIFLAGS)";

* FLAG THE SSI CHILD BENES;
%INCLUDE "OPDR.TG.PRD.ETTW.#8047.DAF21.ADM.PRDLIB(SSIKIDFG)";

%MACRO DOIT;

DATA OUT.AWARDEE
  (DROP=FPMT: PAIDPRE94 PAYS: SPMT: START SO5_BEG
   STATE0: STATE9: STATE1: STATE2: STATE_MR_DM
   PSTA9: PSTA0: PSTA1: PSTA2: PSTA7: PSTA8:);
  * IN THE PREVIOUS PROGRAMS, BENES WHOSE DI AWARD DATE IS GT
  FRA OR WHOSE SSI AWARD DATE IS GT FRA WERE DELETED;
  * NOW, INSTEAD, WE WILL EXCLUDE THEM DURING THE MERGE;
  MERGE OUT1.ADMDIAWD (IN=DI
    WHERE = (DIAWARDAGE<FRA))
    OUT2.ADMSSIAWD(IN=SSI
    WHERE = (SSIADLTAWARDAGE<FRA))
    OUT3.ADMSSIKAWD (IN=SSIK
    WHERE = (SSICHLDAWARDAGE<FRA))
    DEMO.DMG
  ;
  BY SSN;
  IF DI OR SSI OR SSIK;

  * FLAG POSSIBLE SERIAL BENEFICIARIES;
  IF DI AND SSI AND 1<=INTCK('MONTHS',SSIADLTAWARDDT,DIAWARDDT,'C')<=5
  THEN POSSERIAL=1;

  * CREATE OVERALL ADULT VARIABLES;
  ADLTAWARDDT=MIN(DIAWARDDT,SSIADLTAWARDDT);
  ADLTELIGDT=MIN(DIELIGDT,SSIADLTELIGDT);
```

Appendix B.5

JCL/SAS Code: FINADM

```
IF DIAWARDST=ADLTAWARDST THEN DO;
  ADLTAWARDST=DIAWARDST;
  ADLTAWARDAGE=DIAWARDAGE;
END;
ELSE IF SSIADLTAWARDST=ADLTAWARDST THEN DO;
  ADLTAWARDST=SSIADLTAWARDST;
  ADLTAWARDAGE=SSIADLTAWARDAGE;
END;

* CREATE OVERALL VARIABLES;
OVERAWARDDT=MIN(DIAWARDST,SSIADLTAWARDST,SSICHLDAWARDST);
OVERELIGDT=MIN(DIELIGDT,SSIADLTELIGDT,SSICHLDELIGDT);

IF DIAWARDST=OVERAWARDDT THEN DO;
  OVERAWARDDT=DIAWARDST;
  OVERAWARDAGE=DIAWARDAGE;
END;
ELSE IF SSIADLTAWARDST=OVERAWARDDT THEN DO;
  OVERAWARDDT=SSIADLTAWARDST;
  OVERAWARDAGE=SSIADLTAWARDAGE;
END;
ELSE IF SSICHLDAWARDST=OVERAWARDDT THEN DO;
  OVERAWARDDT=SSICHLDAWARDST;
  OVERAWARDAGE=SSICHLDAWARDAGE;
END;

* CREATE ADLTELIGAGE AND OVERELIGAGE;
IF SSIADLTELIGDT=ADLTELIGDT THEN ADLTELIGAGE=SSIADLTELIGAGE;
ELSE IF DIELIGDT=ADLTELIGDT THEN ADLTELIGAGE=DIELIGAGE;

IF SSIADLTELIGDT=OVERELIGDT THEN OVERELIGAGE=SSIADLTELIGAGE;
ELSE IF SSICHLDELIGDT=OVERELIGDT THEN OVERELIGAGE=SSICHLDELIGAGE;
ELSE IF DIELIGDT=OVERELIGDT THEN OVERELIGAGE=DIELIGAGE;

LABEL ADLTAWARDAGE ='AGE AT EARLIEST ADULT SSI OR SSDI AWARD'
ADLTAWARDDT ='DATE OF EARLIEST ADULT SSI OR SSDI AWARD'
ADLTAWARDST ='STATE OF RESIDENCE AT EARLIEST ADULT AWARD DT'
OVERAWARDAGE ='AGE AT EARLIEST OVERALL SSI OR SSDI AWARD'
OVERAWARDDT ='DATE OF EARLIEST OVERALL SSI OR SSDI AWARD'
OVERAWARDST ='STATE OF RESIDENCE AT EARLIEST OVERALL AWARD DATE'
DIAWARDAGE ='AGE AT FIRST ADULT SSDI AWARD'
DIAWARDDAC ='DOEI BEFORE 18TH BIRTHDAY'
DIAWARDST ='DATE OF FIRST ADULT SSDI AWARD'
DIAWARDST ='STATE OF RESIDENCE AT FIRST ADULT SSDI AWARD DATE'
```





Appendix B.5

JCL/SAS Code: FINADM

```
SSICHLDAWARDAGE
ADLTAWARDAGE
OVERAWARDAGE
```

```
DIELIGAGE
SSIADLTELIGAGE
SSICHLDELIGAGE
ADLTELIGAGE
OVERELIGAGE ;
```

```
RUN;
```

```
PROC CONTENTS DATA=OUT.AWARDEE
    OUT=CNT.ADM&ENDYR.;
```

```
RUN;
```

```
%MEND DOIT;
%DOIT;
```

```
//*****
//* SENDING EMAIL FROM HOST USING IEBGENER
//*****
//SENDNOTE EXEC PGM=IEBGENER
//SYSIN DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=(B,SMTP)
//SYSUT1 DD *
HELO MVSHOST
MAIL FROM:<MCHAPPELKA@MATHEMATICA-MPR.COM>
RCPT TO:<MCHAPPELKA@MATHEMATICA-MPR.COM>
DATA
FROM: <MCHAPPELKA@MATHEMATICA-MPR.COM>
TO: <MCHAPPELKA@MATHEMATICA-MPR.COM>
SUBJECT: SUBMISSION DAF20 ADM FINADM IS COMPLETE
DAF21 ADM FINADM IS COMPLETE.
```

## Appendix B.6

JCL/SAS Code: DIFLAGS

```

*****
***** DIFLAGS INCLUDE PROGRAM *****
*****
%LET BEGYR=1994;
%LET ENDYR=&ENDYR; /* UPDATED IN MASTER PROGRAM */
%LET ENDMN=12;

%MACRO FLAG_DI_BENES;

%INCLUDE "OPDR.TG.PRD.ETTW.#6266.DAF20.UTILITY.PRDLIB(YMMMACR)";

DATA TEMP1.ADMDIARD;
  MERGE ADM.COMBANN(IN=ADM
    WHERE = (ADM_SSDI = 1)) /*KEEP DI BENES ONLY*/
    DEMO.DMG(KEEP=SSN_SSDI_ONSET)
  ;
  BY SSN;
  IF ADM;

  /* SET UP ARRAYS FOR PROCESSING */
  ARRAY PAYD{*} %DO MM=1 %TO &TOTMOS; PAYD&&YY&MM %END;;
  ARRAY LAFCODE{*} %DO MM=1 %TO &TOTMOS; LAF&&YY&MM %END;;

  /* PSTYMM ONLY POPULATED FROM 1996 ON */
  ARRAY PST{*} %DO MM=1 %TO &TOTMOS; STATE&&YY&MM %END;;

  /* CREATE A FLAG FOR CASES WHOSE DOEI IS BEFORE JAN, 1994 */
  SUSPECT = 0;
  IF DOEI LT '01JAN1994'D THEN SUSPECT = 1;

  /* COMPUTE AGE AT DATE OF INITIAL ENTITLEMENT */
  DOEIAGE = INTCK('MONTH',DOBBEST,DOEI,'C')/12;

  /* IF DOEI IS BEFORE 18TH BIRTHDAY THEN START LOOKING
  AT 18TH BIRTHDAT. OTHERWISE START LOOKING AT DOEI.
  CREATE A DATE THAT EXPRESSES THIS CHOICE */
  IF DOEIAGE LT 18 THEN DO;
    STDT = INTNX('YEAR',DOBBEST,18,'SAMEDAY');
    STARTDT = INTNX('MONTH',STDT,0,'BEGINNING');
  END;
  ELSE STARTDT = DOEI;

  /* CREATE A DATE VARIABLE FROM THE ARRAY OF VARIABLE NAMES

```

## Appendix B.6

JCL/SAS Code: DIFLAGS

```
USE IT TO COMPARE TO STARTDT TO GET A START POINT TO LOOK
FOR AWARD DATE */
ARRAY ARRDATE{*} %DO MM=1 %TO &TOTMOS; ARRDATE&&YY&MM %END;;
LENGTH ARRNAME $8;
ARRNAME = ' ';

DO I=1 TO DIM(PAYD);
  /* CONSTRUCT DATE FOR THE ARRAY ELEMENT */
  ARRNAME = VNAME(PAYD(I));
  /*PRIOR LOGIC NO LONGER WORKING, UPDATING HERE*/
  YR = INPUT(SUBSTR(ARRNAME,5,2),8.);
  MO = INPUT(SUBSTR(ARRNAME,7,2),8.);
  DAY = 1;
  ARRDATE(I) = MDY(MO,DAY,YR);
END;

/* ESTABLISH THE ARRAY POSITION FOR STARTING */
STARTP = 0;

DO I = 1 TO DIM(ARRDATE) UNTIL (STARTP GT 0);
  IF STARTDT = ARRDATE(I) THEN STARTP = I;
END;

/* FIND THE EARLIEST DATE FROM STARTDT WHERE BENE IN CURRENT
PAY AND RECEIVED A PAYMENT-THIS IS HIS AWARD DATE.
ELIMINATE ANY BENE WITH A PAYMENT IN 1994 OR 1995.
SET THE AWARD STATE ACCORDING TO PST IN THAT MONTH.
FLAG THE ARRAY POSITION FOR THE MONTH */

/* INITIALIZE AWARD DATE AND AWARD STATE TO MISSING
AND FOUNDAWARD TO 0 */
LENGTH DIAWARDST FSTATE BSTATE $2;
DIAWARDDT = .;
DIAWARDST = ' ';
FOUNDAWARD = 0;

/* DROP BENES WITH PAYMENTS IN 1994 OR 1995 */
DROP_BENE = 0;
DO I = 1 TO 24;
  IF SUBSTR(LAFCODE(I),1,1) IN ('C' 'E') AND PAYD(I) GT 0
    THEN DROP_BENE = 1;
END;

/* SET AN AWARD DATE ONLY IF THE PAYMENT IS AFTER 1996
EXCLUDING BENE WITH A PAYMENT IN 1994 OR 1995 */
```

## Appendix B.6

JCL/SAS Code: DIFLAGS

```
IF STARTP GT 0 AND DROP_BENE = 0 THEN DO;
  DO I = STARTP TO DIM(ARRDATE) UNTIL (DIAWARDDT NE .);
    IF SUBSTR(LAFCODE(I),1,1) IN ('C' 'E') AND PAYD(I) GT 0 THEN DO;
      DIAWARDDT = ARRRDATE(I);
      DIAWARDST = PST(I);
      FOUNDAWARD = I;
    END;
  END;
END;

/* IF NO VALUE FOR STATE IS FOUND IN THE AWARD MONTH-SEARCH
FOR THE CLOSEST NON-MISSING STATE VALUE. LOOK FORWARD AND
BACKWARD FROM THE AWARD MONTH. TAKE THE STATE FOUND AFTER
THE FEWEST MONTHS */
IF DIAWARDST = '' AND FOUNDAWARD GT 0 THEN DO;
  /* GOING FORWARD */
  COUNTF = 0;

  DO I = FOUNDAWARD TO DIM(PST) UNTIL (FSTATE NE ' ');
    IF PST(I) NE '' THEN DO;
      FSTATE = PST(I);
      COUNTF = I - FOUNDAWARD;
    END;
  END;

  /* GOING BACKWARD */
  COUNTB = 0;

  DO I = FOUNDAWARD TO 1 BY -1 UNTIL (BSTATE NE ' ');
    IF PST(I) NE '' THEN DO;
      BSTATE = PST(I);
      COUNTB = (FOUNDAWARD - I);
    END;
  END;

  /* SELECT THE CLOSEST STATE */

  /* IF STATE IS FOUND IN 1 DIRECTION AND NOT THE OTHER */
  IF COUNTF GT 0 AND COUNTB EQ 0 THEN DIAWARDST = FSTATE;
  ELSE IF COUNTB GT 0 AND COUNTF EQ 0 THEN DIAWARDST = BSTATE;

  /* IF STATE IS FOUND IN BOTH DIRECTIONS */
  ELSE IF COUNTF GT 0 AND COUNTB GT 0 THEN DO;
    IF COUNTF LE COUNTB THEN DIAWARDST = FSTATE;
    ELSE DIAWARDST = BSTATE;
```

## Appendix B.6

JCL/SAS Code: DIFLAGS

```
END;

END; /* LOOP TO LOOK FOR MISSING STATE */

/* COMPUTE AGE AT DIAWARDDT */
DIAWARDAGE = INTCK('MONTH',DOBBEST,DIAWARDDT,'C')/12;

/* FLAG BENES WHOSE DOEI IS BEFORE HIS 18TH BIRTHDAT */
IF DOEIAGE LT 18 THEN DIAWARDDAC = 1;
ELSE DIAWARDDAC = 0;

YEAR_AWARD = YEAR(DIAWARDDT);

/* DELETE RECORDS WHERE NO PAYMENT WAS MADE AFTER 1996.
THESE CASES ARE MISSING AWARDDATE */
IF DIAWARDDT = . THEN DELETE;

* SET SSDI ELIGIBILITY DATE;
* FIND THE FIRST MONTH OF ELIGIBILITY BASED ON MONTH OF ONSET;
* FIND FIVE FULL MONTHS AFTER DISABILITY ONSET;
IF SSDI_ONSET NE . THEN DO;
IF DAY(SSDI_ONSET)=1
THEN SO5_BEG=INTNX('MONTH',SSDI_ONSET,5,'BEG');
ELSE SO5_BEG=INTNX('MONTH',SSDI_ONSET,6,'BEG');
END;
RUN;

* USING ONE DATA STEP FROM OLD CODE NO LONGER WORKING;
* GETTING ERROR MESSAGE ABOUT LAF VARIABLE ALREADY DEFINED;
* SEPARATING INTO TWO SEPARATE DATA STEPS;
DATA TEMP2.ADMEDIAWD;
SET TEMP1.ADMEDIAWD;

* FIND THE FIRST LAF IN C,E AND THE FIRST DUED NE ZERO;
ARRAY LAFCODE (1994:&ENDYR.,1:12) %DO YEAR=1994 %TO &ENDYR.;
%LET YR=%SUBSTR(&YEAR.,3,2);
LAF&YR.01-LAF&YR.12
%END;;

ARRAY DUED (1994:&ENDYR.,1:12) %DO YEAR=1994 %TO &ENDYR.;
%LET YR=%SUBSTR(&YEAR.,3,2);
DUED&YR.01-DUED&YR.12
%END;;

DIELIGDT=.;
```

## Appendix B.6

JCL/SAS Code: DIFLAGS

```
START=MAX(DOBBEST, '01JAN1994'D, SO5_BEG);

DO YR2=YEAR(START) TO &ENDYR. UNTIL (DIELIGDT NE.);
  IF YR2=YEAR(START) THEN STARTMO=MONTH(START);
  ELSE STARTMO=1;

  DO MO=STARTMO TO 12 UNTIL (DIELIGDT NE.);
    IF SUBSTR(LAFCODE(YR2,MO),1,1) IN ('C' 'E')
      AND DUED(YR2,MO) NOTIN (0,.)
      THEN DIELIGDT=MDY(MO,1,YR2);
  END; * END MONTH LOOP;
END; * END YEAR LOOP

* SET AGE AT ELIGIBILITY;
DIELIGAGE=INTCK('MONTHS',DOBBEST,DIELIGDT,'C')/12;

LABEL DIAWARDST = 'DATE OF FIRST ADULT SSDI AWARD'
      DIAWARDAGE = 'AGE AT FIRST ADULT SSDI AWARD'
      DIAWARDST = 'STATE OF RESIDENCE AT FIRST ADULT SSDI AWARD DATE'
      DIAWARDDAC = 'DOEI BEFORE 18TH BIRTHDAY'
      SUSPECT = 'DOEI EARLIER THAN JAN 1994'
      DIELIGDT = 'DATE OF FIRST ADULT SSDI ELIGIBILITY'
      DIELIGAGE = 'AGE AT FIRST ADULT SSDI ELIGIBILITY';
RUN;

TITLE "CHECKS BEFORE OUTPUTTING SSDI FINAL DATA";
PROC FREQ DATA=TEMP2.ADMDIWD; TABLES YEAR_AWARD; RUN;
PROC PRINT DATA=TEMP2.ADMDIWD (OBS=25);
  FORMAT DOEI LDWDAT: DIAWARDST DOBBEST STARTDT YYMMDD.;
  VAR DOEI DOEIAGE STARTDT DOBBEST FOUNDAWARD DIAWARDST
      DIAWARDAGE DIAWARDDAC COUNTF COUNTB STATE:
      LAF: PAYD;;
  WHERE DOEIAGE LT 18 ;
RUN;

/* OUTPUT DATASET */
DATA OUT1.ADMDIWD (KEEP=SSN DIAWARDST DIAWARDAGE
      DIAWARDDAC SUSPECT DIELIGDT DIELIGAGE
      START SO5_BEG SSDI_ONSET
      FRA DOBBEST);
  SET TEMP2.ADMDIWD;
RUN;

TITLE "CHECK VARIABLES IN FINAL SSDI DATASET";
```

Appendix B.6

JCL/SAS Code: DIFLAGS

```
PROC FREQ DATA=OUT1.ADMEDIAWD;  
  TABLES DIAWARDST DIAWARDAGE DIAWARDDAC SUSPECT / MISSING;  
  TABLES DIAWARDDT / LIST MISSING;  
  FORMAT DIAWARDDT YEAR4.;  
RUN;  
TITLE;  
%MEND FLAG_DI_BENES;  
%FLAG_DI_BENES;
```

## Appendix B.7

JCL/SAS Code: SSIFLAGS

```

*****
***** SSIFLAGS INCLUDE PROGRAM *****
*****
%LET DAFYR=&ENDYR; /* UPDATED IN THE MASTER PROGRAM*/

%MACRO FLAG_SSI_BENES;

DATA OUT2.ADMSSIAWD(KEEP=SSIADLTAWARDDT SSIADLTAWARDST SSICHILDAT18
      SSN SSIADLTAWARDAGE DOBBEST DODBEST STATE: PSTA:
      PAYS: FPMT: SPMT: PAIDPRE94 FRA
      SSIADLTELIGDT SSIADLTELIGAGE);
SET ADM.COMBANN(IN=ADM
      WHERE = (ADM_SSI = 1)) /*KEEP SSI BENES ONLY*/
      ;

* KEEP BENES WHO WILL BECOME 18+ BEFORE END OF DAF YEAR;
IF YEAR(INTNX('MONTHS',DOBBEST,18*12,'SAMEDAY')) <= &DAFYR.;

ARRAY PSTA (1974:&DAFYR.,1:12) %DO YEAR=1974 %TO &DAFYR.;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      PSTA&YR.01-PSTA&YR.12
      %END;;
ARRAY PAYS (1974:&DAFYR.,1:12) %DO YEAR=1974 %TO &DAFYR.;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      PAYS&YR.01-PAYS&YR.12
      %END;;
ARRAY FPMT (1974:1993,1:12) %DO YEAR=1974 %TO 1993;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      FPMT&YR.01-FPMT&YR.12
      %END;;
ARRAY SPMT (1974:1993,1:12) %DO YEAR=1974 %TO 1993;
      %LET YR=%SUBSTR(&YEAR.,3,2);
      SPMT&YR.01-SPMT&YR.12
      %END;;

* COUNT NUMBER OF PAYMENTS BEFORE 1994;
DO YR=1974 TO 1993;
  DO MO=1 TO 12;
    PAYS(YR,MO)=SUM(FPMT(YR,MO),SPMT(YR,MO));
    IF INTCK('MONTH',DOBBEST,MDY(MO,1,YR),'C')/12>18
      AND PSTA(YR,MO) IN ('C01','M01','M02')
      AND PAYS(YR,MO)>0 THEN PAIDPRE94=SUM(PAIDPRE94,1);
  END;

```



## Appendix B.7

JCL/SAS Code: SSIFLAGS

END;

\* SET SSI AWARD DATE, CHILD, AND PAYMENTS AT 18TH BIRTHDAY FLAGS;

\* FIND FIRST MONTH BENE IS FULLY 18 YR OLD;

\* DROP IF THIS YMM IS AFTER DAF YEAR;

\_18BDAY=INTNX('MONTHS',DOBBEST,18\*12,'SAME DAY');

IF DAY(\_18BDAY) = 1 THEN \_18BDMTH = \_18BDAY;

ELSE \_18BDMTH = INTNX('MONTHS',\_18BDAY,1,'SAME DAY');

IF YEAR(\_18BDMTH) &lt;= &amp;DAFYR.;

\* SET STARTLOOP AS LATEST OF JAN 1 1974 OR 1ST MONTH AS 18 YR OLD;

STARTLOOP = MAX("01JAN1974"D, \_18BDMTH);

\* SET STOPLOOP AS EARLIEST OF END OF DAF, DOD OR FRA;

STOPLOOP = MIN("31DEC&amp;DAFYR."D,DODBEST,

INTNX('MONTHS',DOBBEST,FRA\*12,'SAME DAY'));

\* START LOOPING FROM STARTLOOP TO STOPLOOP;

DO YR=YEAR(STARTLOOP) TO YEAR(STOPLOOP) UNTIL(SSIADLTAWARDDT NE.);

IF YR=YEAR(STOPLOOP) THEN ENDMO=MONTH(STOPLOOP);

ELSE ENDMO=12;

IF YR=YEAR(STARTLOOP) THEN STARTMO=MONTH(STARTLOOP);

ELSE STARTMO=1;

DO MO=STARTMO TO ENDMO UNTIL(SSIADLTAWARDDT NE.);

\* CREATE PAYS FOR 1974 THRU 1993;

IF YR&lt;1994 THEN PAYS(YR,MO)=SUM(FPMT(YR,MO),SPMT(YR,MO));

\* IF BENE IN CURRENT PAY AND RECEIVING A PAYMENT THEN

SET FLAGS;

IF PSTA(YR,MO) IN ('C01','M01','M02')

AND PAYS(YR,MO)&gt;0 THEN DO;

\* SET FLAG FOR BENES WHO RECEIVED SSI AS A CHILD;

IF INTCK('MONTH',DOBBEST,MDY(MO,1,YR),'C')/12&lt;18

THEN SSICHILD=1;

ELSE DO;

\* SET SSI AWARD DATE FOR BENES WHO RECEIVE A

PAYMENT AS AN ADULT (18 OR OVER);

SSIADLTAWARDDT=MDY(MO,1,YR);

## Appendix B.7

JCL/SAS Code: SSIFLAGS

```
* SET FLAG FOR CHILD BENES WHO WERE RECEIVING SSI
  AT TIME OF 18TH BIRTHDAY;
  IF INTCK('MONTH',DOBBEST,MDY(MO,1,YR),'C')/12=18
  AND SSICHILD=1 THEN SSICHILDAT18=1;
```

```
  END;
```

```
  END;
```

```
  END;
```

```
END;
```

```
* DELETE BENES WHO RECEIVE THEIR FIRST PAYMENT AS AN ADULT BEFORE
  1996 OR WHO HAVE NO SSI AWARD;
  IF SSIADLTAWARDDT<'01JAN1996'D THEN DELETE;
```

```
* SET AGE AT AWARD;
  SSIADLTAWARDAGE=INTCK('MONTHS',DOBBEST,SSIADLTAWARDDT,'C')/12;
```

```
* SET STATE AT AWARD;
  ARRAY PST (1994:&DAFYR.,1:12) %DO YEAR=1994 %TO &DAFYR.;
    %LET YR=%SUBSTR(&YEAR.,3,2);
    STATE&YR.01-STATE&YR.12
  %END;;
  SSIADLTAWARDST=PST(YEAR(SSIADLTAWARDDT),MONTH(SSIADLTAWARDDT));
```

```
* IF AWARD STATE IS MISSING THEN SEARCH FORWARD FOR A STATE;
  IF SSIADLTAWARDST="" THEN DO YR=YEAR(SSIADLTAWARDDT) TO &DAFYR.
  UNTIL(SSIADLTAWARDST NE "");
  IF YR=YEAR(SSIADLTAWARDDT) THEN STARTMO=MONTH(SSIADLTAWARDDT);
  ELSE STARTMO=1;
```

```
  DO MO=STARTMO TO 12 UNTIL(SSIADLTAWARDST NE "");
```

```
    SSIADLTAWARDST=PST(YR,MO);
```

```
  END;
```

```
END;
```

```
* IF AWARD STATE IS MISSING THEN SEARCH BACKWARD FOR A STATE;
  IF SSIADLTAWARDST="" THEN DO YR=YEAR(SSIADLTAWARDDT) TO 1994 BY -1
  UNTIL(SSIADLTAWARDST NE "");
  IF YR=YEAR(SSIADLTAWARDDT) THEN STARTMO=MONTH(SSIADLTAWARDDT);
  ELSE STARTMO=12;
```

```
  DO MO=STARTMO TO 1 BY -1 UNTIL(SSIADLTAWARDST NE "");
```

```
    SSIADLTAWARDST=PST(YR,MO);
```

```
  END;
```

```
END;
```

## Appendix B.7

JCL/SAS Code: SSIFLAGS

```
* SET SSI ELIGIBILITY DATE;
DO YR=YEAR(STARTLOOP) TO YEAR(STOPLOOP) UNTIL(SSIADLTELIGDT NE.);
  IF YR=YEAR(STOPLOOP) THEN ENDMO=MONTH(STOPLOOP);
  ELSE ENDMO=12;

  IF YR=YEAR(STARTLOOP) THEN STARTMO = MONTH(STARTLOOP);
  ELSE STARTMO = 1;

  DO MO=STARTMO TO ENDMO UNTIL(SSIADLTELIGDT NE.);
    * IF BENE IN CURRENT PAY AND RECEIVE A PAYMENT AS AN ADULT
      (18 OR OVER);
    IF PSTA(YR,MO) IN ('C01','M01','M02') AND
    INTCK('MONTH',DOBBEST,MDY(MO,1,YR),'C')/12>=18
    THEN SSIADLTELIGDT=MDY(MO,1,YR);
  END;
END;

* SET AGE AT ELIGIBILITY;
SSIADLTELIGAGE=INTCK('MONTHS',DOBBEST,SSIADLTELIGDT,'C')/12;

LABEL SSIADLTAWARDAGE='AGE AT ADULT SSI AWARD'
      SSIADLTAWARDDT='DATE OF ADULT SSI AWARD'
      SSIADLTAWARDST='STATE AT ADULT SSI AWARD'
      SSICCHILD='RECEIVED SSI CHILD PMTS'
      SSICCHILDAT18='RECEIVING SSI CHILD PMTS AT 18TH BIRTHDAY'
      ;
RUN;

PROC FREQ DATA=OUT2.ADMSSIAWD;
  FORMAT SSIADLTAWARDDT DOBBEST YEAR4.;
  TABLE PAIDPRE94 / MISSING;
  TABLE SSIADLTAWARDDT / MISSING;
  TABLE SSIADLTAWARDDT*DOBBEST / LIST MISSING;
  TABLE SSICCHILDAT18 / LIST MISSING;
RUN;

PROC CONTENTS DATA=OUT2.ADMSSIAWD;
RUN;

%MEND FLAG_SSI_BENES;
%FLAG_SSI_BENES;
```



## Appendix B.8

JCL/SAS Code: SSIKIDFG

```
STOPLOOP = MIN("31DEC&DAFYR."D,DODBEST,
               INTNX('MONTHS',DOBBEST,18*12,'SAMEDAY'));

* START LOOPING FROM STARTLOOP TO STOPLOOP;
DO YR=YEAR(STARTLOOP) TO YEAR(STOPLOOP) UNTIL(SSICHLDAWARDDT NE .);
  IF YR=YEAR(STOPLOOP) THEN ENDMO=MONTH(STOPLOOP);
  ELSE ENDMO=12;

  IF YR=YEAR(STARTLOOP) THEN STARTMO=MONTH(STARTLOOP);
  ELSE STARTMO=1;

  DO MO=STARTMO TO ENDMO UNTIL(SSICHLDAWARDDT NE .);

    * CREATE PAYS FOR 1974 THRU 1993;
    IF YR<1994 THEN PAYS(YR,MO)=SUM(FPMT(YR,MO),SPMT(YR,MO));

    * IF BENE IN CURRENT PAY AND RECEIVING A PAYEMNT THEN
      SET FLAGS;
      IF PSTA(YR,MO) IN ('C01','M01','M02')
        AND PAYS(YR,MO)>0 THEN DO;

        * SET FLAG FOR BENES WHO RECEIVED SSI AS A CHILD;
        IF INTCK('MONTH',DOBBEST,MDY(MO,1,YR),'C')/12<18
          THEN DO;
          SSICHLCHILD = 1;
          SSICHLDAWARDDT = MDY(MO,1,YR);
        END;

        ELSE DO;
          * SET FLAG FOR CHILD BENES WHO WERE RECEIVING SSI
            AT TIME OF 18TH BIRTHDAY;
          IF INTCK('MONTH',DOBBEST,MDY(MO,1,YR),'C')/12=18
            AND SSICHLCHILD=1 THEN SSICHLDAT18=1;

        END;
      END;
    END;
  END;

* DELETE BENES WHO RECEIVE THEIR FIRST PAYMENT AS AN CHILD BEFORE
  1996 OR WHO HAVE NO SSI AWARD;
IF SSICHLDAWARDDT<'01JAN1996'D THEN DELETE;

* SET AGE AT AWARD;
SSICHLDAWARDAGE=INTCK('MONTHS',DOBBEST,SSICHLDAWARDDT,'C')/12;
```

## Appendix B.8

JCL/SAS Code: SSIKIDFG

```
* SET STATE AT AWARD;
ARRAY PST (1994:&DAFYR.,1:12) %DO YEAR=1994 %TO &DAFYR.;
    %LET YR=%SUBSTR(&YEAR.,3,2);
    STATE&YR.01-STATE&YR.12
    %END;;
SSICHLDAWARDST=PST(YEAR(SSICHLDAWARDST),MONTH(SSICHLDAWARDST));

* IF AWARD STATE IS MISSING THEN SEARCH FORWARD FOR A STATE;
IF SSICHLDAWARDST="" THEN DO YR=YEAR(SSICHLDAWARDST) TO &DAFYR.
UNTIL(SSICHLDAWARDST NE "");
    IF YR=YEAR(SSICHLDAWARDST) THEN STARTMO=MONTH(SSICHLDAWARDST);
    ELSE STARTMO=1;

    DO MO=STARTMO TO 12 UNTIL(SSICHLDAWARDST NE "");
        SSICHLDAWARDST=PST(YR,MO);
    END;
END;

* IF AWARD STATE IS MISSING THEN SEARCH BACKWARD FOR A STATE;
IF SSICHLDAWARDST="" THEN DO YR=YEAR(SSICHLDAWARDST) TO 1994 BY -1
UNTIL(SSICHLDAWARDST NE "");
    IF YR=YEAR(SSICHLDAWARDST) THEN STARTMO=MONTH(SSICHLDAWARDST);
    ELSE STARTMO=12;

    DO MO=STARTMO TO 1 BY -1 UNTIL(SSICHLDAWARDST NE "");
        SSICHLDAWARDST=PST(YR,MO);
    END;
END;

* SET SSI ELIGIBILITY DATE;
DO YR=YEAR(STARTLOOP) TO YEAR(STOPLOOP) UNTIL(SSICHLDELIGDT NE .);
    IF YR=YEAR(STOPLOOP) THEN ENDMO=MONTH(STOPLOOP);
    ELSE ENDMO=12;

    IF YR=YEAR(STARTLOOP) THEN STARTMO=MONTH(STARTLOOP);
    ELSE STARTMO=1;

    DO MO=STARTMO TO ENDMO UNTIL(SSICHLDELIGDT NE .);
        * IF BENE IN CURRENT PAY AND RECEIVE A PAYMENT AS AN ADULT
        (18 OR OVER);
        IF PSTA(YR,MO) IN ('C01','M01','M02') AND
        INTCK('MONTH',DOBBEST,MDY(MO,1,YR),'C')/12 <18
        THEN SSICHLDELIGDT=MDY(MO,1,YR);
    END;
```

Appendix B.8

JCL/SAS Code: SSIKIDFG

```
END;

* SET AGE AT ELIGIBILITY;
SSICHLDELIGAGE=INTCK('MONTHS',DOBBEST,SSICHLDELIGDT,'C')/12;

LABEL SSICHLDAWARDAGE='AGE AT CHILD SSI AWARD'
      SSICHLDAWARDDT='DATE OF CHILD SSI AWARD'
      SSICHLDAWARDST='STATE AT CHILD SSI AWARD'
      SSICHLCHILD='RECEIVED SSI CHILD PMTS'
      SSICHLCHILDAT18='RECEIVING SSI CHILD PMTS AT 18TH BIRTHDAY'
      ;
RUN;

PROC FREQ DATA=OUT3.ADMSSIKAWD;
  FORMAT SSICHLDAWARDDT DOBBEST YEAR4.;
  TABLE SSICHLDAWARDDT / MISSING;
  TABLE SSICHLDAWARDDT*DOBBEST / LIST MISSING;
RUN;

PROC CONTENTS DATA=OUT3.ADMSSIKAWD;
RUN;

%MEND FLAG_SSI_KIDS;
%FLAG_SSI_KIDS;
```

Appendix B.9

JCL/SAS Code: COPYADM

```
//$4671CP1 JOB (12510000,T715,,SAS,,ITC9FL),PHELPS-DAWN,
//      MSGCLASS=1,MSGLEVEL=(1,1),NOTIFY=$4671
//*****
//* PROJECT:   DAF 50214
//* DIRECTOR:  JODY SCHIMMEL HYDE
//* PROGRAM:   OPDR.TG.PRD.ETTW.$4671.DAF19.ADM.PRDLIB(COPYADM)
//* DESCRIPTION: CREATE ARCHIVE COPY OF EXISTING DAF FILES
//* DATE:      02/19/2021
//* PROGRAMMER: DAWN PHELPS
//*           DPHELPS@MATHEMATICA-MPR.COM
//*           202-554-7556
//*****
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=OPDR.TG.PRD.ETTW.DRAFT.DAF19D.AWARDEE,DISP=SHR
//SYSUT2 DD DSN=OPDR.TG.PRD.ETTW.FINAL.DAF19D.AWARDEE,
//      DISP=(NEW,CATLG,DELETE),VOL=(,,10),
//      UNIT=TSILO
//SYSIN DD DUMMY
```



---

**Mathematica Inc.**

Princeton, NJ • Ann Arbor, MI • Cambridge, MA  
Chicago, IL • Oakland, CA • Seattle, WA  
Woodlawn, MD • Washington, DC

**EDI Global, a Mathematica Company**

Operating in Tanzania, Uganda, Kenya, Mozambique, and the United Kingdom

Mathematica, Progress Together, and the “spotlight M” logo are registered trademarks of Mathematica Inc.



mathematica.org [website](#)