

IDENTIFICATION DIVISION.

PROGRAM-ID. PSPCOBRA.

ENVIRONMENT DIVISION.

* *

* *

* *

* *

* This software and related documentation are provided under a *

* license agreement containing restrictions on use and *

* disclosure and are protected by intellectual property *

* laws. Except as expressly permitted in your license agreement *

* or allowed by law, you may not use, copy, reproduce, *

* translate, broadcast, modify, license, transmit, distribute, *

* exhibit, perform, publish or display any part, in any form or *

* by any means. Reverse engineering, disassembly, or *

* decompilation of this software, unless required by law for *

* interoperability, is prohibited. *

* The information contained herein is subject to change without *

* notice and is not warranted to be error-free. If you find any *

* errors, please report them to us in writing. *

* *

* *

* Copyright (C) 1988, 2019, Oracle and/or its affiliates. *

* All Rights Reserved. *

* *

* \$Release: HR92 *

* \$Bug: 25861189 *

* *

* *

* PROGRAM DESCRIPTION: *

* *

* COBRA ADMINISTRATION *

* *

* MODIFIED FOR FEDERAL *

* FED0005 ADDITIONAL JOB FIELDS FOR ELIGIBILITY CHECKING *FED0005

* FED0006 CHOOSE THE SC DATE USED FOR MONTHS OF SERVICE ELIG *FED0006

* FEDMERG FEDERAL MERGE *FEDMERG

* MODIFIED FOR EDUCATION & GOVERNMENT *

DATA DIVISION.

WORKING-STORAGE SECTION.

01 PROGRAM-IDENTITY PIC X(8) VALUE 'PSPCOBRA'.

01 W-WK.

02 TIME-OUT PIC 99B99B99/99.

02 TIME-COLLECT PIC 9(8).

02 FILLER REDEFINES TIME-COLLECT.

03 TIME-COLLECT-HR PIC 99.

03 TIME-COLLECT-MM PIC 99.

03 TIME-COLLECT-SS PIC 99.

03 TIME-COLLECT-HH PIC 99.

02 TIME-CLOCK PIC S9(5) COMP-3.

02 TIME-START PIC S9(5) COMP-3.

02 TIME-ELAPSED PIC S9(5) COMP-3.

02 DISPLAY-TIME.

03 DISPLAY-MM PIC 99.

03 FILLER PIC X VALUE ':'.
.

03 DISPLAY-SS PIC 99.

02 DISPLAY-COUNT-1 PIC Z(6)9.

02 DISPLAY-COUNT-2 PIC Z(6)9.

02 COVERED-PERS-TYP-CNT PIC S9(5).

02 PROGRESS-TOTAL PIC S9(7) COMP-3.

02 PROGRESS-COUNT PIC S9(7) COMP-3.

02 PROGRESS-REPORT PIC S9(7) COMP-3.

88 PROGRESS-REPORT-OVERAGE VALUE 20.

88 PROGRESS-REPORT-ACTIVITY VALUE 20.

88 PROGRESS-REPORT-QUALIFY VALUE 20.

88 PROGRESS-REPORT-PARTIC VALUE 20.

02 WORK-COUNT PIC 9999 VALUE 0 COMP.

02 DEPRUL-PTR PIC 9999 COMP.

01 W-CVG.

02 COVRG-CD PIC X(2).

02 CBR-COVRG-SET PIC XX.

02 CC-CTL-TOTAL-FLG PIC X.

02 MIN-NUM-TOTAL PIC 99 COMP.

02 MAX-NUM-TOTAL PIC 99 COMP.

02 EFFDT PIC X(10).

02 COVERED-PERSON-TYP PIC XX.

02 SUM-MIN-NUM-CVRD PIC 99 COMP.

02 SUM-MAX-NUM-CVRD PIC 999 COMP.

02 MIN-NUM-CVRD-CH PIC 99 COMP.

02 MAX-NUM-CVRD-CH PIC 999 COMP.

02	MIN-NUM-CVRD-SP	PIC 99	COMP.
02	MAX-NUM-CVRD-SP	PIC 99	COMP.
02	WCVGCD-MAX-EFFDT	PIC X(10).	

01 W-SW.

02	RESELECT-SW	PIC X.	
88	RESELECT-NO	VALUE 'N'.	
88	RESELECT-YES	VALUE 'Y'.	
02	COBRA-PLAN-SW	PIC X.	
88	COBRA-PLAN-YES	VALUE 'Y'.	
88	COBRA-PLAN-NO	VALUE 'N'.	
02	OVERAGE-DEP-SW	PIC X.	
88	OVERAGE-DEP-YES	VALUE 'Y'.	
88	OVERAGE-DEP-NO	VALUE 'N'.	
02	OVERAGE-S-SW	PIC X.	
88	OVERAGE-S-YES	VALUE 'Y'.	
88	OVERAGE-S-NO	VALUE 'N'.	
02	OVERAGE-NS-SW	PIC X.	
88	OVERAGE-NS-YES	VALUE 'Y'.	
88	OVERAGE-NS-NO	VALUE 'N'.	

02 ERROR-SW PIC X.

88 ERROR-NO VALUE 'N'.

88 ERROR-YES VALUE 'Y'.

02 VALID-COVRG-CD-SW PIC X.

88 VALID-COVRG-CD-NO VALUE 'N'.

88 VALID-COVRG-CD-YES VALUE 'Y'.

02 VALID-OPTION-SW PIC X.

88 VALID-OPTION-NO VALUE 'N'.

88 VALID-OPTION-YES VALUE 'Y'.

02 VALID-OPTION-CD-SW PIC X.

88 VALID-OPTION-CD-NO VALUE 'N'.

88 VALID-OPTION-CD-YES VALUE 'Y'.

02 PREP-OPTIONS-FOUND-SW PIC X.

88 OPTIONS-FOUND-NO VALUE 'N'.

88 OPTIONS-FOUND-YES VALUE 'Y'.

02 VALID-ELECTION-SW PIC X.

88 VALID-ELECTION-NO VALUE 'N'.

88 VALID-ELECTION-YES VALUE 'Y'.

02 VALID-OPT-PREP-SW PIC X.

88 VALID-OPT-PREP-NO VALUE 'N'.

88 VALID-OPT-PREP-YES VALUE 'Y'.

02 VALID-DEPENDENT-SW PIC X.

88 VALID-DEPENDENT-NO VALUE 'N'.

88 VALID-DEPENDENT-YES VALUE 'Y'.

02 LOAD-EMPL-DEP-SW PIC X.

88 LOAD-EMPL-DEP-NO VALUE 'N'.

88 LOAD-EMPL-DEP-YES VALUE 'Y'.

02 CBR-REPROCESS-IND PIC X.

88 CBR-REPROCESS-NONE VALUE 'N'.

88 CBR-REPROCESS-QUALIFY VALUE 'Q'.

88 CBR-REPROCESS-PREPARE VALUE 'P'.

88 CBR-REPROCESS-ELECT VALUE 'E'.

88 CBR-REPROCESS-VOID VALUE 'V'.

02 CBR-QUALIFY-SW PIC X.

88 CBR-QUALIFY-NO VALUE 'N'.

88 CBR-QUALIFY-YES VALUE 'Y'.

02 CBR-QUALIFY-ERROR-SW PIC X.

88 CBR-QUALIFY-ERROR-NO VALUE 'N'.

88 CBR-QUALIFY-ERROR-YES VALUE 'Y'.

02 CBR-QUALIFY-PENDING-SW PIC X.

88 CBR-QUALIFY-PENDING-NO	VALUE 'N'.
88 CBR-QUALIFY-PENDING-YES	VALUE 'Y'.

02 RESET-EVENT-STATUS-SW PIC X.

88 RESET-EVENT-STATUS-NO	VALUE 'N'.
88 RESET-EVENT-STATUS-YES	VALUE 'Y'.

02 REPROCESS-PARTIC-SW PIC X.

88 REPROCESS-PARTIC-NO	VALUE 'N'.
88 REPROCESS-PARTIC-YES	VALUE 'Y'.

02 REPROCESS-PLAN-SW PIC X.

88 REPROCESS-PLAN-NO	VALUE 'N'.
88 REPROCESS-PLAN-YES	VALUE 'Y'.

02 DUPLICATE-EVENT-SW PIC X.

88 DUPLICATE-EVENT-NO	VALUE 'N'.
88 DUPLICATE-EVENT-YES	VALUE 'Y'.

02 REG-REGION-IND PIC X.

88 REG-REGION-NOT-USA	VALUE 'N'.
88 REG-REGION-USA	VALUE 'Y'.

02 OPTN-FOUND-SW PIC X.

88 OPTN-FOUND-NO	VALUE 'N'.
88 OPTN-FOUND-YES	VALUE 'Y'.

02 BASE-OPTN-MATCH-SW PIC X.

88 BASE-OPTN-MATCH-NO VALUE 'N'.

88 BASE-OPTN-MATCH-YES VALUE 'Y'.

02 BASE-CVGCD-MATCH-SW PIC X.

88 BASE-CVGCD-MATCH-NO VALUE 'N'.

88 BASE-CVGCD-MATCH-YES VALUE 'Y'.

02 BENADMIN-ELIG-OPTN-SW PIC X.

88 BENADMIN-ELIG-OPTN-NO VALUE 'N'.

88 BENADMIN-ELIG-OPTN-YES VALUE 'Y'.

02 BENADMIN-ELIG-CVGCD-SW PIC X.

88 BENADMIN-ELIG-CVGCD-NO VALUE 'N'.

88 BENADMIN-ELIG-CVGCD-YES VALUE 'Y'.

02 STORE-HR-SW PIC X.

88 STORE-HR-NO VALUE 'N'.

88 STORE-HR-YES VALUE 'Y'.

02 STORE-BP-PGM-SW PIC X.

88 STORE-BP-PGM-NO VALUE 'N'.

88 STORE-BP-PGM-YES VALUE 'Y'.

02 EE-FOUND-SW PIC X.

88 EE-FOUND-NO VALUE 'N'.

88 EE-FOUND-YES VALUE 'Y'.

02 OTHR-DEP-FOUND-SW PIC X.

88 OTHR-DEP-FOUND-NO VALUE 'N'.

88 OTHR-DEP-FOUND-YES VALUE 'Y'.

02 BENPRG-FOUND-SW PIC X.

88 BENPRG-FOUND-NO VALUE 'N'.

88 BENPRG-FOUND-YES VALUE 'Y'.

01 W-COUNT.

02 INIT-PLAN-COUNT PIC 9999 VALUE ZERO COMP.

02 INIT-ERROR-COUNT PIC 9999 VALUE ZERO COMP.

02 INIT-ENROLL-COUNT PIC 9999 VALUE ZERO COMP.

02 INIT-NOT-ENROLL-COUNT PIC 9999 VALUE ZERO COMP.

02 INIT-ENTERED-COUNT PIC 9999 VALUE ZERO COMP.

02 SCND-PLAN-COUNT PIC 9999 VALUE ZERO COMP.

02 SCND-PENDING-COUNT PIC 9999 VALUE ZERO COMP.

02 NOT-DETERM-PLAN-COUNT PIC 9999 VALUE ZERO COMP.

01 W-CNTL.

02 CHKPT-INTERVAL PIC 999V99 COMP-3.

02 COBRA-PHASE PIC X.

88 COBRA-PHASE-READY VALUE 'R'.

88 COBRA-PHASE-OVERAGE VALUE 'O'.

88 COBRA-PHASE-ACTIVITY VALUE 'A'.

88 COBRA-PHASE-QUALIFY VALUE 'Q'.

88 COBRA-PHASE-PARTIC VALUE 'P'.

88 COBRA-PHASE-COMPLETE VALUE 'C'.

02 OVERAGE-PROCESS PIC X.

88 OVERAGE-PROCESS-YES VALUE 'Y'.

88 OVERAGE-PROCESS-NO VALUE 'N'.

02 BENEFIT-ADMINISTRN PIC X.

88 BENADMIN-YES VALUE 'Y'.

88 BENADMIN-NO VALUE 'N'.

02 BEN-BILLING PIC X.

88 BILLING-YES VALUE 'Y'.

88 BILLING-NO VALUE 'N'.

02 MULTIJOB PIC X.

88 MULTIJOB-NO VALUE 'N'.

88 MULTIJOB-YES VALUE 'Y'.

02 COBRA-EMPLID.

NOCLN 03 COBRA-PREFIX PIC X VALUE 'C'.

03 COBRA-EMPLID-LAST PIC 9(10).

02 RESTART-KEY.

03 BENEFIT-PROGRAM PIC X(10).

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-DT PIC X(10).

03 COBRA-EVENT-ID PIC 999 COMP.

03 DEPENDENT-BENEF PIC XX.

03 COBRA-ACTION PIC XXX.

02 PROCESS-DT PIC X(10).

02 AGE-AS-OF-DT PIC X(10).

02 COBRA-EVENT-CLASS PIC XXX.

88 COBRA-EVENT-OVERAGE VALUE 'OVG'.

02 CBR-ACTION-SOURCE PIC XXX.

88 CBR-SOURCE-OVG-PROCESS VALUE 'O'.

02 SQL-CONTROLS.

03 REPROCESS-CURSORS.

04 S-REPRT.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-REPRPAR.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-REPRPLN.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
03 OVERAGE-CURSORS.
04 S-BENPGM.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-CBR-EVT.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-EE-PGM.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 I-CBRACTY.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
03 OBTAIN-COBRA-DATA-CURSORS.
04 S-CBRACTY.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-EVT.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-PAR1.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-PAR2.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-PRCSPAR.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-PARPLN1.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-PARPLN2.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-PARPLN3.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-PAROPTN.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-PARDPND.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

03 STORE-COBRA-DATA-CURSORS.

04 I-EVT.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 I-PAR.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 I-PARPLN.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 I-PAROPTN.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

03 UPDATE-COBRA-DATA-CURSORS.

04 U-EVT.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 U-RESEVT.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 U-PAR.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 U-PARPLN.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 U-PARDPND.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

03 DELETE-COBRA-DATA-CURSORS.

04 D-CBRCTY.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 D-PAR.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 D-PARPLN.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 D-PAROPTN.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 D-PARDPD1.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 D-PARDPND.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

03 OBTAIN-ELECT-CURSORS.

04 S-BP-PGM.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-HTH.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-HTH-DEP.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-DEPEND.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-DEP-NID.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-STDNT.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-FSA.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-HTHTERM.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-FSATERM.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-DEPTERM.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-HTHWAIV.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 S-FSAWAIV.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
03 STORE-ELECT-CURSORS.
04 I-BP-PGM.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 I-BP-PLN.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 I-HTH.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 I-HTH-DEP.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.
04 I-FSA.
05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

03 UPDATE-ELECT-CURSORS.

04 U-HTHTERM.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 U-FSATERM.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 U-DEPEND.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 U-XHTH.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 U-VDPBEN.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

03 DELETE-ELECT-CURSORS.

04 D-BP-PGM.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 D-BP-PLN.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 D-HTH.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 D-HTH-DEP.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 D-FSA.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 D-XDEP.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

03 OBTAIN-HR-DATA-CURSORS.

04 S-PERDATA.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-SMOKER.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-ADDRESS.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-CURJOB.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-EMPL-HR.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-ELGDATA.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 S-DISABLE.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

03 STORE-HR-DATA-CURSORS.

* 04 I-PERDATA.

* 05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

* 04 I-PEREFDT.

* 05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

* 04 I-PER-NID.

* 05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

* 04 I-EMPLMNT.

* 05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

* 04 I-JOB.

* 05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

* 04 I-JOBJR.

* 05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

* 04 I-PRIJOB.

* 05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

* 04 I-PERADDR.

* 05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

* 04 I-PERNAMES.

* 05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

* 04 I-PERPHONE.

* 05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

04 I-NONEE.

05 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

01 W-SAVE.

02 WCBREVT-INIT PIC 9999 VALUE ZERO COMP.

02 WCBREVT-SCND PIC 9999 VALUE ZERO COMP.

02 RATE-PERCENT PIC 999 VALUE ZERO COMP.

02 PLAN-TYPE PIC XX VALUE SPACES.

02 OVERAGE-DT PIC X(10).

01 W-EVENT.

02 EMPLID PIC X(20).

02 EMPL-RCD-NO PIC 999 COMP.

02 BENEFIT-RCD-NO PIC 999 COMP.

02 PER-STATUS PIC X.

88 EMPLOYEE VALUE 'E'.

88 NON-EMPLOYEE VALUE 'N'.

02 PER-TYPE PIC X.
 88 NONE VALUE ''.
 88 COBRA-PARTICIPANT VALUE 'C'.
 88 PENSION-PAYEE VALUE 'P'.
02 DISABLED PIC XX.
 88 DISABLED-YES VALUE 'Y'.
 88 DISABLED-NO VALUE 'N'.
02 BIRTHDATE PIC X(10).
02 MEDICARE-ENTLD-DT PIC X(10).
02 SEX PIC X.
02 SMOKER PIC X.
02 HIGHLY-COMP-EMPL-C PIC X.
02 ADDRESS-DATA.
 03 COUNTRY PIC XXX.
 03 ADDRESS1 PIC X(55).
 03 ADDRESS2 PIC X(55).
 03 ADDRESS3 PIC X(55).
 03 ADDRESS4 PIC X(55).
 03 CITY PIC X(30).
 03 NUM1 PIC X(6).
 03 NUM2 PIC XXXX.
 03 HOUSE-TYPE PIC XX.
 03 ADDR-FIELD1 PIC XX.
 03 ADDR-FIELD2 PIC XXXX.
 03 ADDR-FIELD3 PIC XXXX.
 03 COUNTY PIC X(30).

03 STATE PIC X(6).

03 POSTAL PIC X(20).

03 GEO-CODE PIC X(11).

03 IN-CITY-LIMIT PIC X.

02 DEP-MIN-COVRG-END-DT PIC X(10).

02 EMPL-HR-DATA.

03 SERVICE-DT PIC X(10).

03 TERMINATION-DT PIC X(10).

03 UNION-CD PIC X(10).

03 DEPTID PIC X(10).

03 JOBCODE PIC X(20).

03 EMPL-STATUS PIC X.

03 ACTION PIC XXX.

03 ACTION-DT PIC X(10).

03 ACTION-REASON PIC XXX.

03 SETID-LOCATION PIC X(5).

03 LOCATION PIC X(20).

03 TAX-LOCATION-CD PIC X(10).

03 REG-TEMP PIC X.

03 FULL-PART-TIME PIC X.

03 OFFICER-CD PIC X.

03 COMPANY PIC X(10).

03 PAYGROUP PIC X(10).

03 BAS-GROUP-ID PIC X(10).

03 ELIG-CONFIG1 PIC X(10).

03 ELIG-CONFIG2 PIC X(10).

03	ELIG-CONFIG3	PIC X(10).	
03	ELIG-CONFIG4	PIC X(10).	
03	ELIG-CONFIG5	PIC X(10).	
03	ELIG-CONFIG6	PIC X(10).	
03	ELIG-CONFIG7	PIC X(10).	
03	ELIG-CONFIG8	PIC X(10).	
03	ELIG-CONFIG9	PIC X(10).	
03	BEN-STATUS	PIC XXXX.	
03	EMPL-TYPE	PIC X.	
03	STD-HOURS	PIC S9999V99	COMP-3.
03	STD-HRS-FREQUENCY	PIC X(5).	
03	STDHRS-FREQ-FACTOR	PIC 9(4)V9(7)	COMP-3.
03	FLSA-STATUS	PIC X.	
03	SETID-SALARY	PIC X(5).	
03	SAL-ADMIN-PLAN	PIC XXXX.	FEDMERG
03	GRADE	PIC XXX.	
03	REG-REGION	PIC X(10).	
03	BUSINESS-UNIT	PIC X(5).	
03	EMPL-CLASS	PIC XXX.	
03	SETID-DEPT	PIC X(5).	
03	SETID-JOB CODE	PIC X(5).	
03	PAY-SYSTEM-FLG	PIC XX.	
03	BENEFIT-SYSTEM	PIC XX.	
03	GVT-HIRE-DATE	PIC X(10).	FED0006
03	GVT-ELIG-FEHB	PIC XXX.	FED0005
03	GVT-RETIRE-PLAN	PIC XX.	FED0005

03 JOB-EMPL-RCD PIC 999 COMP.
03 JOB-EFFDT PIC X(10).
03 JOB-EFFSEQ PIC 999 COMP.

02 WEVENT-DATA.

03 COBRA-EVENT-ID PIC 999 COMP.
03 COBRA-EVENT-DT PIC X(10).
03 COBRA-EVENT-CLASS PIC XXX.
88 COBRA-EVENT-OVERAGE VALUE 'OVG'.
88 COBRA-EVENT-DEPENDENT VALUE 'DEP'.
88 COBRA-EVENT-MEDICARE VALUE 'MED'.
03 SCHED-ID PIC X(10).
03 EVENT-ID PIC 9(6) COMP.
03 CBR-ACTION-SOURCE PIC X.
03 CBR-COV-OVERRIDE PIC X.
88 CBR-COV-OVERRIDE-NO VALUE 'N'.
88 CBR-COV-OVERRIDE-YES VALUE 'Y'.
03 CBR-ALT-COV-TRM-DT PIC X(10).
03 CBR-COV-AS-OF-DT PIC X(10).
03 CBR-QUALIFY-STATUS PIC XX.
88 CBR-QUALIFY-UNPROCESSED VALUE 'UN'.
88 CBR-NOT-QUALIFIED VALUE 'NQ'.
88 CBR-QUALIFY-ERROR VALUE 'QE'.
88 CBR-QUALIFIED VALUE 'QL'.
88 CBR-QUALIFY-PENDING VALUE 'QP'.
03 CBR-PROCESS-STATUS PIC X.

88 CBR-PROCESS-CLOSED VALUE 'C'.

88 CBR-PROCESS-OPEN VALUE 'O'.

88 CBR-PROCESS-VOID VALUE 'V'.

03 CBR-EVT-REPRCS-IND PIC X.

88 CBR-REPROCESS-NONE VALUE 'N'.

88 CBR-REPROCESS-QUALIFY VALUE 'Q'.

88 CBR-REPROCESS-PREPARE VALUE 'P'.

88 CBR-REPROCESS-VOID VALUE 'V'.

03 CBR-EVENT-CONFLICT PIC X.

88 CBR-EVT-CONFLICT-NO VALUE 'N'.

88 CBR-EVT-CONFLICT-YES VALUE 'Y'.

03 BAS-DATA-CHG PIC X.

88 BAS-DATA-CHG-NO VALUE 'N'.

88 BAS-DATA-CHG-YES VALUE 'Y'.

03 UPDATE-SW PIC X.

88 UPDATE-NO VALUE 'N'.

88 UPDATE-YES VALUE 'Y'.

03 INSERT-SW PIC X.

88 INSERT-NO VALUE 'N'.

88 INSERT-YES VALUE 'Y'.

02 WPARTIC-COUNT PIC 9999 VALUE ZERO COMP.

88 WPARTIC-COUNT-MAX VALUE 50.

02 WPARTIC-DATA OCCURS 50

INDEXED BY

WPARTIC-IDX.

03 DEPENDENT-BENEF PIC XX.

03 CBR-QUALIFY-STATUS PIC XX.

88 CBR-QUALIFY-UNPROCESSED VALUE 'UN'.

88 CBR-NOT-QUALIFIED VALUE 'NQ'.

88 CBR-QUALIFY-ERROR VALUE 'QE'.

88 CBR-QUALIFIED VALUE 'QL'.

88 CBR-QUALIFY-PENDING VALUE 'QP'.

03 CBR-PROCESS-STATUS PIC X.

88 CBR-PROCESS-CLOSED VALUE 'C'.

88 CBR-PROCESS-OPEN VALUE 'O'.

88 CBR-PROCESS-VOID VALUE 'V'.

03 CBR-INIT-EVENT-STS PIC XX.

88 CBR-INIT-ENROLL-COMPLETE VALUE 'EC'.

88 CBR-INIT-ELECT-ERROR VALUE 'EE'.

88 CBR-INIT-ELECT-ENROLLED VALUE 'ER'.

88 CBR-INIT-ELECT-ENTERED VALUE 'ET'.

88 CBR-INIT-NOT-QUALIFIED VALUE 'NQ'.

88 CBR-INIT-NOTIFIED VALUE 'NT'.

88 CBR-INIT-OPTION-PREPARED VALUE 'OP'.

88 CBR-INIT-OPT-PREP-ERROR VALUE 'OE'.

88 CBR-INIT-QUALIFIED VALUE 'QL'.

88 CBR-INIT-QUALIFY-PENDING VALUE 'QP'.

03 CBR-SCND-EVENT-STS PIC XX.

88 CBR-SCND-COVRG-EXTENDED VALUE 'CE'.

88 CBR-SCND-NOT-QUALIFIED VALUE 'NQ'.

88 CBR-SCND-NOTIFIED VALUE 'NT'.
88 CBR-SCND-QUALIFIED VALUE 'QL'.
88 CBR-SCND-QUALIFY-PENDING VALUE 'QP'.
03 CBR-INIT-NOTIFY-DT PIC X(10).
03 CBR-SCND-NOTIFY-DT PIC X(10).
03 COBRA-ELECT PIC X.
88 COBRA-ELECT-ELECT VALUE 'E'.
88 COBRA-ELECT-WAIVE VALUE 'W'.
88 COBRA-ELECT-NONE VALUE '''.
03 COBRA-ELECT-DT PIC X(10).
03 COBRA-WAIVE-DT PIC X(10).
03 COBRA-REVOKE-DT PIC X(10).
03 COBRA-EMPLID PIC X(11).
03 BENEFIT-PROGRAM PIC X(10).
03 BAS-ASSIGN-STATUS PIC X(2).
88 BAS-ASSIGN-ERROR VALUE 'AE'.
88 BAS-ASSIGN-NONE VALUE 'AN'.
88 BAS-ASSIGNED VALUE 'AS'.
03 CBR-TERM-NOTIFY-DT PIC X(10).
03 CBR-PAR-REPRCS-IND PIC X.
88 CBR-REPROCESS-NONE VALUE 'N'.
88 CBR-REPROCESS-QUALIFY VALUE 'Q'.
88 CBR-REPROCESS-PREPARE VALUE 'P'.
88 CBR-REPROCESS-ELECT VALUE 'E'.
88 CBR-REPROCESS-VOID VALUE 'V'.
03 UPDATE-SW PIC X.

88	UPDATE-NO	VALUE 'N'.
88	UPDATE-YES	VALUE 'Y'.
03	INSERT-SW	PIC X.
88	INSERT-NO	VALUE 'N'.
88	INSERT-YES	VALUE 'Y'.
02	WPLAN-COUNT	PIC 9999 VALUE ZERO COMP.
88	WPLAN-COUNT-MAX	VALUE 125.
02	WPLAN-DATA	OCCURS 125
		INDEXED BY
		WPLAN-IDX.
03	DEPENDENT-BENEF	PIC XX.
03	PLAN-TYPE	PIC XX.
88	PLAN-TYPE-HEALTH	VALUE '10' THRU '19'
		'1A' THRU '1Z'.
88	PLAN-TYPE-FSA	VALUE '60'.
03	COBRA-EVENT-TYPE	PIC X.
88	COBRA-EVENT-INITIAL	VALUE 'I'.
88	COBRA-EVENT-SECONDARY	VALUE 'S'.
88	COBRA-EVENT-NOT-DETERMINED	VALUE 'N'.
88	COBRA-EVENT-UNQUALIFIED	VALUE ''.
03	CBR-INIT-EVENT-ID	PIC 999 COMP.
03	CBR-SCND-EVENT-ID	PIC 999 COMP.
03	CBR-PROCESS-STATUS	PIC X.
88	CBR-PROCESS-CLOSED	VALUE 'C'.

88 CBR-PROCESS-OPEN VALUE 'O'.
 88 CBR-PROCESS-VOID VALUE 'V'.
 03 CBR-ENROLL-STATUS PIC X.
 88 CBR-ENROLL-NO VALUE 'N'.
 88 CBR-ENROLLED VALUE 'E'.
 88 COBRA-COVERAGE-EXTENDED VALUE 'C'.
 88 COBRA-COVERAGE-INSERTED VALUE 'I'.
 03 CBR-PLN-REPRCS-IND PIC X.
 88 CBR-REPROCESS-NONE VALUE 'N'.
 88 CBR-REPROCESS-QUALIFY VALUE 'Q'.
 88 CBR-REPROCESS-PREPARE VALUE 'P'.
 88 CBR-REPROCESS-ELECT VALUE 'E'.
 88 CBR-REPROCESS-VOID VALUE 'V'.
 03 COVERAGE-BEGIN-DT PIC X(10).
 03 DISABL-CVG-BEG-DT PIC X(10).
 03 COVERAGE-END-DT PIC X(10).
 03 COBRA-PD-BEGINS-DT PIC X(10).
 03 COBRA-ELECT-END-DT PIC X(10).
 03 COBRA-ELECT PIC X.
 88 COBRA-ELECT-ELECT VALUE 'E'.
 88 COBRA-ELECT-WAIVE VALUE 'W'.
 88 COBRA-ELECT-NONE VALUE '''.
 03 COBRA-ELECT-DT PIC X(10).
 03 COBRA-WAIVE-DT PIC X(10).
 03 COBRA-REVOKE-DT PIC X(10).
 03 COBRA-TERM-DT PIC X(10).

03 COBRA-TERM-REASON PIC X.

88 COBRA-NOT-TERMINATED VALUE 'N'.

03 OPTION-CD PIC XXX.

03 BENEFIT-PLAN PIC X(10).

03 COVRG-CD PIC X(2).

03 CBR-COVRG-SET PIC XX.

03 MIN-NUM-OF-DEPS PIC 99 COMP.

03 MAX-NUM-OF-DEPS PIC 99 COMP.

03 SPOUSE-COVERAGE PIC X.

88 SPOUSE-COVERAGE-ALLOWED VALUE 'A'.

88 SPOUSE-COVERAGE-NOT-ALLOWED VALUE 'N'.

88 SPOUSE-COVERAGE-REQUIRED VALUE 'R'.

88 SPOUSE-COVERAGE-SPOUSE-ONLY VALUE 'S'.

88 SPOUSE-COVERAGE-DEPEND-ONLY VALUE 'D'.

88 SPOUSE-COVERAGE-SPSDEP-ONLY VALUE 'O'.

03 DEPENDENT-COUNT PIC 99 VALUE ZERO COMP.

03 DEP-SPOUSE-IND PIC X.

88 DEP-SPOUSE-YES VALUE 'Y'.

88 DEP-SPOUSE-NO VALUE 'N'.

03 EMPL-CONTRBUTN-AMT PIC S9999V99 COMP-3.

03 ANN-EX-CREDIT-FSA PIC S9(5)V99 COMP-3.

03 ANNUAL-PLEDGE PIC S9(5)V99 COMP-3.

03 FSA-SUB-AMT-YTD PIC S9(6)V99 COMP-3.

03 FSA-APR-AMT-YTD PIC S9(6)V99 COMP-3.

03 FSA-PD-AMT-YTD PIC S9(6)V99 COMP-3.

03 COBRA-ERROR PIC X.

88 COBRA-ERROR-NO VALUE 'N'.
88 COBRA-ERROR-YES VALUE 'Y'.
03 UPDATE-SW PIC X.
88 UPDATE-NO VALUE 'N'.
88 UPDATE-YES VALUE 'Y'.
03 INSERT-SW PIC X.
88 INSERT-NO VALUE 'N'.
88 INSERT-YES VALUE 'Y'.
03 CLEAR-SCND-EVT-IND PIC X.
88 CLEAR-SCND-EVT-NO VALUE 'N'.
88 CLEAR-SCND-EVT-YES VALUE 'Y'.
03 CUR-COVRG-IND PIC X.
88 CUR-COVRG-NONE VALUE 'N'.
88 CUR-COVRG-ACTIVE VALUE 'A'.
88 CUR-COVRG-TERM VALUE 'T'.
03 COBRA-PD-BEGINS-CD PIC X.
88 COBRA-PD-BEGINS-EVENT-DT VALUE 'E'.
88 COBRA-PD-BEGINS-ON-AFT-MM VALUE 'M'.
88 COBRA-PD-BEGINS-NEXT-MM VALUE 'A'.
88 COBRA-PD-BEGINS-NEXT-PAYPD VALUE 'P'.
03 INIT-BENEFIT-PGM PIC X(10).
03 INIT-PLAN-ENROLL-SW PIC X.
88 INIT-PLAN-ENROLL-NO VALUE 'N'.
88 INIT-PLAN-ENROLLED VALUE 'E'.
03 INIT-PLAN-ELECT-SW PIC X.
88 INIT-PLAN-ELECT-ELECT VALUE 'E'.

88 INIT-PLAN-ELECT-WAIVE VALUE 'W'.
 88 INIT-PLAN-ELECT-NONE VALUE ''.

03 INIT-DISABL-CVG-DT PIC X(10).
 03 INIT-COVRG-END-DT PIC X(10).
 03 INIT-COVRG-BEGIN-DT PIC X(10).
 03 INIT-COBRA-EVENT-ID PIC 999 COMP.
 03 SET-SCND-EVT-IND PIC X.
 88 SET-SCND-EVT-NO VALUE 'N'.
 88 SET-SCND-EVT-YES VALUE 'Y'.
 03 HLTH-PROVIDER-ID PIC X(30).
 03 PREVIOUSLY-SEEN PIC X.
 03 OTH-INSURANCE-IND PIC X.
 03 OTH-INSURANCE-NAME PIC X(30).

02 WOPTN-COUNT PIC 9999 VALUE ZERO COMP.
 88 WOPTN-COUNT-MAX VALUE 1000.

02 WOPTN-DATA OCCURS 1000
 INDEXED BY
 WOPTN-IDX.

03 DEPENDENT-BENEF PIC XX.
 03 PLAN-TYPE PIC XX.
 88 PLAN-TYPE-HEALTH VALUE '10' THRU '19'
 '1A' THRU '1Z'.
 88 PLAN-TYPE-FSA VALUE '60'.

03	OPTION-ID	PIC 9(4)	COMP.
03	BENEFIT-PLAN	PIC X(10).	
03	COVRG-CD	PIC X(2).	
03	OPTION-CD	PIC XXX.	
03	CBR-MM-COST	PIC S9(8)V99	COMP-3.
03	CBR-DISABL-MM-COST	PIC S9(8)V99	COMP-3.
03	INSERT-SW	PIC X.	
88	INSERT-NO	VALUE 'N'.	
88	INSERT-YES	VALUE 'Y'.	
02	WDPND-COUNT	PIC 9999	VALUE ZERO COMP.
88	WDPND-COUNT-MAX	VALUE 75.	
02	WDPND-DATA	OCCURS 75	
		INDEXED BY	
		WDPND-IDX.	
03	DEPENDENT-BENEF	PIC XX.	
03	PLAN-TYPE	PIC XX.	
88	PLAN-TYPE-HEALTH	VALUE '10' THRU '19'	
		'1A' THRU '1Z'.	
88	PLAN-TYPE-FSA	VALUE '60'.	
03	COBRA-DEP-BENEF	PIC XX.	
03	COBRA-ERROR	PIC X.	
88	COBRA-ERROR-NO	VALUE 'N'.	
88	COBRA-ERROR-YES	VALUE 'Y'.	
03	UPDATE-SW	PIC X.	

88 UPDATE-NO VALUE 'N'.

88 UPDATE-YES VALUE 'Y'.

03 HLTH-PROVIDER-ID PIC X(30).

03 PREVIOUSLY-SEEN PIC X.

03 OTH-INSURANCE-IND PIC X.

03 OTH-INSURANCE-NAME PIC X(30).

02 CBR-NONEE-CREATE PIC X.

88 CBR-NONEE-CREATE-PERSON VALUE 'P'.

88 CBR-NONEE-CREATE-JOB VALUE 'J'.

88 CBR-NONEE-CREATE-ALL VALUE 'A'.

01 W-MIN-DEPRULE.

02 DEP-AGE-LIMIT PIC 99 COMP.

02 STUDENT-AGE-LIMIT PIC 99 COMP.

02 EXCL-DISABLED-AGE PIC X VALUE SPACES.

88 EXCL-DISABLED-YES VALUE 'Y'.

88 EXCL-DISABLED-NO VALUE 'N'.

02 DEPENDENT-MARRIAGE PIC X VALUE SPACES.

88 MARRIED-DEP-NOT-ALLOWED VALUE 'Y'.

88 MARRIED-DEP-ALLOWED VALUE 'N'.

02 OVERAGE-PD-BEG-CD PIC X.

88 OVERAGE-PD-BEG-EVENT-DT VALUE 'E'.

88 OVERAGE-PD-BEG-ON-AFT-MM VALUE 'M'.

01 W-DEPRULE.

02 DEP-AGE-LIMIT PIC 99 COMP.

02 STUDENT-AGE-LIMIT PIC 99 COMP.

02 EXCL-DISABLED-AGE PIC X VALUE SPACES.

88 EXCL-DISABLED-YES VALUE 'Y'.

88 EXCL-DISABLED-NO VALUE 'N'.

02 DEPENDENT-MARRIAGE PIC X VALUE SPACES.

88 MARRIED-DEP-NOT-ALLOWED VALUE 'Y'.

88 MARRIED-DEP-ALLOWED VALUE 'N'.

02 OVERAGE-PD-BEG-CD PIC X.

88 OVERAGE-PD-BEG-EVENT-DT VALUE 'E'.

88 OVERAGE-PD-BEG-ON-AFT-MM VALUE 'M'.

01 W-CBRDEFN.

02 BENEFIT-PROGRAM PIC X(10) VALUE SPACES.

02 EFFDT PIC X(10) VALUE SPACES.

02 COBRA-SURCHARGE PIC 99 COMP.

02 COBRA-DISABL-SURCG PIC 99 COMP.

02 WCBRPLN-COUNT PIC 9999 COMP.

02 WCBRPLN-DATA OCCURS 50

INDEXED BY

WCBRPLN-IDX.

03 PLAN-TYPE PIC XX.

88 PLAN-TYPE-HEALTH VALUE '10' THRU '19'

'1A' THRU '1Z'.

88 PLAN-TYPE-FSA VALUE '60'.

03 EVENT-RULES-ID PIC X(10).

03 CHECK-EMPL-TERM-SW PIC X.

88 CHECK-EMPL-TERM-NO VALUE 'N'.

88 CHECK-EMPL-TERM-YES VALUE 'Y'.

03 EMPL-COVRG-TERM-IND PIC X.

88 EMPL-COVRG-TERM-NO VALUE 'N'.

88 EMPL-COVRG-TERM-YES VALUE 'Y'.

03 EMPL-COVRG-TERM-DT PIC X(10).

03 SECONDARY-ONLY-IND PIC X.

88 SECONDARY-ONLY-NO VALUE 'N'.

88 SECONDARY-ONLY-YES VALUE 'Y'.

03 DEP-RULE-ID PIC X(5) VALUE SPACES.

03 DEP-AGE-LIMIT PIC 99 COMP.

03 STUDENT-AGE-LIMIT PIC 99 COMP.

03 EXCL-DISABLED-AGE PIC X VALUE SPACES.

88 EXCL-DISABLED-YES VALUE 'Y'.

88 EXCL-DISABLED-NO VALUE 'N'.

03 DEPENDENT-MARRIAGE PIC X VALUE SPACES.

88 MARRIED-DEP-NOT-ALLOWED VALUE 'Y'.

88 MARRIED-DEP-ALLOWED VALUE 'N'.

03 OVERAGE-PD-BEG-CD PIC X.

88 OVERAGE-PD-BEG-EVENT-DT VALUE 'E'.

88 OVERAGE-PD-BEG-ON-AFT-MM VALUE 'M'.

01 W-CBR-EVT.

02 WCBR-EVT-MAX-EFFDT PIC X(10).

02 WCBR-EVT-COUNT PIC 9999 COMP.

88 WCBR-EVNT-COUNT-MAX VALUE 50.

02 WCBR-EVT-DATA OCCURS 50

INDEXED BY

WCBR-EVT-IDX.

03 COBRA-EVENT-CLASS PIC XXX.

03 EFFDT PIC X(10).

03 EFF-STATUS PIC X.

88 EFF-STATUS-ACTIVE VALUE 'A'.

88 EFF-STATUS-INACTIVE VALUE 'I'.

03 MM-COVERAGE PIC 99 COMP.

03 ADDL-MM-DISABLED PIC 99 COMP.

03 USE-BAS-EVENT-RULE PIC X.

88 USE-BAS-EVENT-RULES-YES VALUE 'Y'.

88 USE-BAS-EVENT-RULES-NO VALUE 'N'.

03 COBRA-PD-BEGINS-CD PIC X.

88 COBRA-PD-BEGINS-EVENT-DT VALUE 'E'.

88 COBRA-PD-BEGINS-ON-AFT-MM VALUE 'M'.

88 COBRA-PD-BEGINS-NEXT-MM VALUE 'A'.

88 COBRA-PD-BEGINS-NEXT-PAYPD VALUE 'P'.

03 INCL-ALT-COVRG PIC X.

88 INCL-ALT-COVRG-YES VALUE 'Y'.

88 INCL-ALT-COVRG-NO VALUE 'N'.

03 EVENT-NOTIFY-DAYS PIC 99 COMP.

03 RESPONSE-DAYS PIC 99 COMP.

03 PAYMENT-GRACE-DAYS PIC 99 COMP.

03 WAIVE-COBRA-SURCHG PIC X.

88 WAIVE-COBRA-SURCHG-NO VALUE 'N'.

88 WAIVE-COBRA-SURCHG-YES VALUE 'Y'.

03 SECOND-EVENT-ROLE PIC X.

88 PRECEEDING VALUE 'P'.

88 SUCCEEDING VALUE 'S'.

03 SECOND-EVENT-MONTH PIC 99 COMP.

03 SECOND-EVENT-MODE PIC X.

88 SCND-EVT-MODE-ADD VALUE 'A'.

88 SCND-EVT-MODE-EXTEND VALUE 'E'.

03 BENEFIT-LOST-LEVEL PIC X.

88 BENEFIT-LOST-EMPL VALUE 'E'.

88 BENEFIT-LOST-DEPEND VALUE 'D'.

02 WCBR-BEN-COUNT PIC 9999 COMP.

88 WCBR-BEN-COUNT-MAX VALUE 100.

02 WCBR-BEN-DATA OCCURS 100

INDEXED BY

WCBR-BEN-IDX.

03 COBRA-EVENT-CLASS PIC XXX.

03 COVERED-PERSON-TYP PIC XX.

88 EMPLOYEE VALUE 'EE'.

88 SPOUSE VALUE 'SP'.

88 SAME-SEX-SPOUSE VALUE 'SS'.

88 EX-SPOUSE VALUE 'XS'.

88 CHILD VALUE 'CH'.

01 W-CVGCD.

02 WCVGCD-MAX-EFFDT PIC X(10).

02 WCVGCD-COUNT PIC 9999 VALUE ZERO COMP.

88 WCVGCD-COUNT-MAX VALUE 50.

02 WCVGCD-DATA OCCURS 50

INDEXED BY

WCVGCD-IDX.

03 COVRG-CD PIC X(2).

03 CBR-COVRG-SET PIC XX.

03 CC-CTL-TOTAL-FLG PIC X.

88 CC-CTL-TOTAL-YES VALUE 'Y'.

88 CC-CTL-TOTAL-NO VALUE 'N'.

03 MIN-NUM-TOTAL PIC 99 COMP.

03 MAX-NUM-TOTAL PIC 99 COMP.

03 EFFDT PIC X(10).

03 MIN-NUM-OF-DEPS PIC 99 COMP.

03 MAX-NUM-OF-DEPS PIC 99 COMP.

03 MIN-NUM-CHILD PIC 99 COMP.

03 MAX-NUM-CHILD PIC 999 COMP.

03 SPOUSE-COVERAGE PIC X.

88 SPOUSE-COVERAGE-ALLOWED VALUE 'A'.

88 SPOUSE-COVERAGE-NOT-ALLOWED VALUE 'N'.

88 SPOUSE-COVERAGE-REQUIRED VALUE 'R'.

88 SPOUSE-COVERAGE-SPOUSE-ONLY VALUE 'S'.

88 SPOUSE-COVERAGE-DEPEND-ONLY VALUE 'D'.

88 SPOUSE-COVERAGE-SPSDEP-ONLY VALUE 'O'.

03 OTHR-DEP-IN-CVGCD PIC X.

88 OTHR-DEP-IN-CVGCD-YES VALUE 'Y'.

88 OTHR-DEP-IN-CVGCD-NO VALUE 'N'.

01 W-DEPEND.

02 WDEPEND-ASOFDT PIC X(10).

02 WDEPEND-BREAK PIC XX.

02 WDEPEND-COUNT PIC 9999 VALUE ZERO COMP.

88 WDEPEND-COUNT-MAX VALUE 50.

02 WDEPEND-DATA OCCURS 50

INDEXED BY

WDEPEND-IDX

WDEPINS-IDX.

03 DEPENDENT-BENEF PIC XX.

03 SAME-ADDRESS-EMPL PIC X.

88 SAME-ADDRESS-EMPL-NO VALUE 'N'.

03 ADDRESS-TYPE PIC X(4).

03 DEP-ADDRESS-SBR.

04 COUNTRY PIC XXX.

04 ADDRESS1 PIC X(55).
04 ADDRESS2 PIC X(55).
04 ADDRESS3 PIC X(55).
04 ADDRESS4 PIC X(55).
04 CITY PIC X(30).
04 NUM1 PIC X(6).
04 NUM2 PIC XXXX.
04 HOUSE-TYPE PIC XX.
04 ADDR-FIELD1 PIC XX.
04 ADDR-FIELD2 PIC XXXX.
04 ADDR-FIELD3 PIC XXXX.
04 COUNTY PIC X(30).
04 STATE PIC X(6).
04 POSTAL PIC X(20).
04 GEO-CODE PIC X(11).
04 IN-CITY-LIMIT PIC X.
03 BIRTHDATE PIC X(10).
03 DT-OF-DEATH PIC X(10).
03 MEDICARE-ENTLD-DT PIC X(10).
03 COBRA-ACTION PIC XXX.
88 COBRA-ACTION-OVERAGE VALUE 'OVG'.
88 COBRA-ACTION-MARRIED VALUE 'DEP'.
88 COBRA-ACTION-NONE VALUE ''.
03 COBRA-EVENT-DT PIC X(10).
03 COBRA-EMPLID PIC X(11).
03 DEPENDENT-AGE PIC 9999 VALUE ZERO COMP.

03 QUALIFIED-IND PIC X.

88 QUALIFIED-YES VALUE 'Y'.

88 QUALIFIED-NO VALUE 'N'.

03 OVERAGE-IND PIC X.

88 OVERAGE-YES VALUE 'Y'.

88 OVERAGE-NO VALUE 'N'.

03 OVERAGE-DT PIC X(10).

03 MAX-COVRG-END-DT PIC X(10).

03 UPDATE-SW PIC X.

88 UPDATE-YES VALUE 'Y'.

88 UPDATE-NO VALUE 'N'.

03 WDEPEFF-START PIC 9999 VALUE ZERO COMP.

02 SEARCH-EFFDT PIC X(10).

02 WDEPEFF-COUNT PIC 9999 VALUE ZERO COMP.

88 WDEPEFF-COUNT-MAX VALUE 50.

02 WDEPEFF-DATA OCCURS 50

INDEXED BY

WDEPEFF-IDX.

03 DEPENDENT-BEN PIC XX.

03 EFFDT PIC X(10).

03 COVERED-PERSON-TYP PIC XX.

88 EMPLOYEE VALUE 'EE'.

88 SPOUSE VALUE 'SP'.

88 SAME-SEX-SPOUSE VALUE 'SS'.

88 EX-SPOUSE VALUE 'XS'.

88 CHILD VALUE 'CH'.

03 DEP-BENEF-TYPE PIC X.

03 MAR-STATUS PIC X.

88 MARRIED VALUE 'M'.

03 SEX PIC X.

03 STUDENT-STATUS PIC X.

88 STUDENT VALUE 'Y'.

88 NON-STUDENT VALUE 'N'.

03 STUDENT-STATUS-DT PIC X(10).

03 DISABLED PIC XX.

88 DISABLED-YES VALUE 'Y'.

88 DISABLED-NO VALUE 'N'.

03 SMOKER PIC X.

88 SMOKER-YES VALUE 'Y'.

88 SMOKER-NO VALUE 'N'.

03 AGE-LIMIT-FLG PIC X.

88 AGE-LIMIT-YES VALUE 'Y'.

88 AGE-LIMIT-NO VALUE 'N'.

01 W-DEPNID.

02 WDEPNID-COUNT PIC 9999 VALUE ZERO COMP.

88 WDEPNID-COUNT-MAX VALUE 250.

02 WDEPNID-DATA OCCURS 250

INDEXED BY

WDEPNID-IDX.

03 DEPENDENT-BENEF PIC XX.

03 NID-COUNTRY PIC XXX.

03 NATIONAL-ID-TYPE PIC XXXX.

03 NATIONAL-ID PIC X(20).

03 PRIMARY-NID PIC X.

01 W-CURELT.

02 COBRA-EVENT-ID PIC 999 COMP.

02 BENEFIT-PROGRAM PIC X(10).

02 WCUR-COUNT PIC 9999 VALUE ZERO COMP.

88 WCUR-COUNT-MAX VALUE 25.

02 WCUR-DATA OCCURS 25

INDEXED BY

WCUR-IDX.

03 PLAN-TYPE PIC XX.

03 EFFDT PIC X(10).

03 BENEFIT-PLAN PIC X(10).

03 COVRG-CD PIC X(2).

03 COVERAGE-BEGIN-DT PIC X(10).

03 DEDUCTION-END-DT PIC X(10).

03 COVERAGE-END-DT PIC X(10).

03 COVERAGE-ELECT PIC X.

88 COVERAGE-ELECT-ELECT VALUE 'E'.

88 COVERAGE-ELECT-WAIVE VALUE 'W'.

88 COVERAGE-ELECT-TERM VALUE 'T'.

88 COVERAGE-ELECT-NONE VALUE ''.

03 EMPL-CONTRIBUTN-AMT PIC S9999V99 COMP-3.

03 ANN-EX-CREDIT-FSA PIC S9(5)V99 COMP-3.

03 ANNUAL-PLEDGE PIC S9(5)V99 COMP-3.

03 FSA-SUB-AMT-YTD PIC S9(6)V99 COMP-3.

03 FSA-APR-AMT-YTD PIC S9(6)V99 COMP-3.

03 FSA-PD-AMT-YTD PIC S9(6)V99 COMP-3.

03 HLTH-PROVIDER-ID PIC X(30).

03 PREVIOUSLY-SEEN PIC X.

03 OTH-INSURANCE-IND PIC X.

03 OTH-INSURANCE-NAME PIC X(30).

02 WCURDB-COUNT PIC 9999 VALUE ZERO COMP.

88 WCURDB-COUNT-MAX VALUE 75.

02 WCURDB-DATA OCCURS 75

INDEXED BY

WCURDB-IDX.

03 PLAN-TYPE PIC XX.

03 BENEFIT-PLAN PIC X(10).

03 COVRG-CD PIC X(2).

03 DEPENDENT-BENEF PIC XX.

03 OVERAGE-IND PIC X.
88 OVERAGE-YES VALUE 'Y'.
88 OVERAGE-NO VALUE 'N'.
03 OVERAGE-DT PIC X(10).
03 HLTH-PROVIDER-ID PIC X(30).
03 PREVIOUSLY-SEEN PIC X.
03 OTH-INSURANCE-IND PIC X.
03 OTH-INSURANCE-NAME PIC X(30).

01 W-ADJCVG.

02 WADJ-COUNT PIC 9999 VALUE ZERO COMP.
88 WADJ-COUNT-MAX VALUE 25.

02 WADJ-DATA OCCURS 25
INDEXED BY
WADJ-IDX.

03 PLAN-TYPE PIC XX.
03 EFFDT PIC X(10).
03 COVERAGE-ELECT PIC X.
03 BENEFIT-PLAN PIC X(10).
03 COVRG-CD PIC X(2).
03 CBR-COVRG-SET PIC XX.
03 DEPENDENT-COUNT PIC 99 VALUE ZERO COMP.
03 CHILD-COUNT PIC 999 VALUE ZERO COMP.
03 DEP-SPOUSE-IND PIC X.

88 DEP-SPOUSE-YES VALUE 'Y'.
 88 DEP-SPOUSE-NO VALUE 'N'.
 03 WADJ-EXIST-ENR PIC X VALUE 'N'.
 88 WADJ-EXIST-ENR-YES VALUE 'Y'.
 88 WADJ-EXIST-ENR-NO VALUE 'N'.
 03 BENEFIT-PROGRAM PIC X(10).
 03 HLTH-PROVIDER-ID PIC X(30).
 03 PREVIOUSLY-SEEN PIC X.
 03 OTH-INSURANCE-IND PIC X.
 03 OTH-INSURANCE-NAME PIC X(30).

 02 WADJDB-COUNT PIC 9999 VALUE ZERO COMP.
 88 WADJDB-COUNT-MAX VALUE 75.

 02 WADJDB-DATA OCCURS 75
 INDEXED BY
 WADJDB-IDX.

 03 PLAN-TYPE PIC XX.
 03 EFFDT PIC X(10).
 03 DEPENDENT-BENEF PIC XX.
 03 HLTH-PROVIDER-ID PIC X(30).
 03 PREVIOUSLY-SEEN PIC X.
 03 OTH-INSURANCE-IND PIC X.
 03 OTH-INSURANCE-NAME PIC X(30).

01 W-OVGACT.

02 WOVGACT-COUNT PIC 9999 VALUE ZERO COMP.

88 WOVGACT-COUNT-MAX VALUE 25.

02 WOVGACT-DATA OCCURS 25

INDEXED BY

WOVGACT-IDX.

03 COBRA-EVENT-DT PIC X(10).

/*****

* INSTALLATION SQL BUFFER AND STMT *

01 S-INSTALL.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_INSTALL'.

02 BIND-SETUP.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC XXX VALUE ALL 'C'.
03 FILLER PIC X(6) VALUE '0PPPPP'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 BENEFIT-ADMINISTRN PIC X.
03 BEN-BILLING PIC X.
03 COMPANY PIC XXX.
03 COBRA-EMPLID-LAST PIC 9(10) COMP-3.
03 FILLER PIC X VALUE 'Z'.

/*****

* OPTIONS SQL BUFFER AND STMT *

01 S-OPTIONS.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_OPTIONS'.

02 BIND-SETUP.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 MULTIJOB PIC X.

03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA_RUNCTL SQL BUFFER AND STMT *

*****/

01 S-RUNCTL.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_RUNCTL'.

02 BIND-SETUP.

03 FILLER PIC X(30) VALUE ALL 'C'.

03 FILLER PIC X(30) VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 OPRID PIC X(30).

03 BATCH-RUN-ID PIC X(30).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER	PIC XXX	VALUE ALL '2PP'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X(10)	VALUE ALL 'C'.
03 FILLER	PIC X(20)	VALUE ALL 'H'.
03 FILLER	PIC XX	VALUE ALL 'S'.
03 FILLER	PIC XX	VALUE ALL 'M'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC XX	VALUE ALL 'S'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC XXX	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE 'Z'.

02 SELECT-DATA.

03 PROCESS-DT	PIC X(10).	
03 CHKPT-INTERVAL	PIC 999V99	COMP-3.
03 COBRA-PHASE	PIC X.	
03 OVERAGE-PROCESS	PIC X.	
03 BENEFIT-PROGRAM	PIC X(10).	
03 EMPLID	PIC X(20).	
03 EMPL-RCD-NO	PIC 999	COMP.
03 BENEFIT-RCD-NO	PIC 999	COMP.
03 COBRA-EVENT-DT	PIC X(10).	
03 COBRA-EVENT-ID	PIC 999	COMP.
03 DEPENDENT-BENEF	PIC XX.	
03 COBRA-ACTION	PIC XXX.	


```
03 INDUSTRY          PIC X(4).          MERGE
03 INDUSTRY-SECTOR   PIC X(4).          MERGE
03 FILLER            PIC X    VALUE 'Z'.  MERGE
```

/*****

* COBRA_RUNCTL SQL DELETE STMT *

01 D-RUNCTL.

```
02 SQL-STMT          PIC X(18) VALUE
                      'PSPCOBRA_D_RUNCTL'.
```

02 BIND-SETUP.

```
03 FILLER            PIC X(30) VALUE ALL 'C'.
03 FILLER            PIC X(30) VALUE ALL 'H'.
03 FILLER            PIC X    VALUE 'Z'.
```

02 BIND-DATA.

```
03 OPRID             PIC X(30).
03 BATCH-RUN-ID      PIC X(30).
03 FILLER            PIC X    VALUE 'Z'.
```

/*****

* COBRA_MESSAGES SQL DELETE STMT *

01 D-MSG.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_D_MSG'.

02 BIND-SETUP.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA_RUNCTL SQL UPDATE STMT *

01 U-RESTART.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_U_RESTART'.

02 BIND-SETUP.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X(20) VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC XXX VALUE ALL 'H'.
03 FILLER PIC X(30) VALUE ALL 'C'.
03 FILLER PIC X(30) VALUE ALL 'H'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 COBRA-PHASE PIC X VALUE SPACES.
03 OVERAGE-PROCESS PIC X VALUE SPACES.
03 BENEFIT-PROGRAM PIC X(10) VALUE SPACES.
03 EMPLID PIC X(20) VALUE SPACES.
03 EMPL-RCD-NO PIC 999 COMP.
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-DT PIC X(10) VALUE SPACES.
03 COBRA-EVENT-ID PIC 999 COMP.
03 DEPENDENT-BENEF PIC XX VALUE SPACES.
03 COBRA-ACTION PIC XXX VALUE SPACES.
03 OPRID PIC X(30) VALUE SPACES.
03 BATCH-RUN-ID PIC X(30) VALUE SPACES.
03 FILLER PIC X VALUE 'Z'.

/*****

* INSTALLATION SQL BUFFER AND STMT *

01 U-INSTALL.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_U_INSTALL'.

02 BIND-SETUP.

03 FILLER PIC X(6) VALUE '0PPPPP'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 COBRA-EMPLID-LAST PIC 9(10) COMP-3.

03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA_EVENT REPROCESS SQL BUFFER AND STMT *

01 S-REPREVT.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_REPREVT'.

02 BIND-SETUP.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
 03 FILLER PIC XX VALUE ALL 'S'.
 03 FILLER PIC XX VALUE ALL 'M'.
 03 FILLER PIC X(10) VALUE ALL 'D'.
 03 FILLER PIC XXX VALUE ALL 'C'.
 03 FILLER PIC XX VALUE ALL 'H'.
 03 FILLER PIC X VALUE ALL 'C'.
 03 FILLER PIC X VALUE ALL 'H'.
 03 FILLER PIC X VALUE ALL 'C'.
 03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID PIC X(20).
 03 BENEFIT-RCD-NO PIC 999 COMP.
 03 COBRA-EVENT-ID PIC 999 COMP.
 03 COBRA-EVENT-DT PIC X(10).
 03 COBRA-EVENT-CLASS PIC XXX.
 03 CBR-QUALIFY-STATUS PIC XX.
 03 CBR-PROCESS-STATUS PIC X.
 03 CBR-EVT-REPRCS-IND PIC X.
 03 CBR-EVENT-CONFLICT PIC X.
 03 FILLER PIC X VALUE 'Z'.

/*****

01 S-REPRPAR.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_REPRPAR'.

02 BIND-SETUP.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X(11) VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID PIC X(20).

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.
03 DEPENDENT-BENEF PIC XX.
03 COBRA-EMPLID PIC X(11).
03 CBR-QUALIFY-STATUS PIC XX.
03 CBR-INIT-EVENT-STS PIC XX.
03 CBR-SCND-EVENT-STS PIC XX.
03 CBR-PAR-REPRCS-IND PIC X.
03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA_PARTIC_PLAN REPROCESS SQL BUFFER AND STMT *

*****/

01 S-REPRPLN.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_REPRPLN'.

02 BIND-SETUP.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER	PIC XX	VALUE ALL 'M'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC X(11)	VALUE ALL 'H'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC XX	VALUE ALL 'S'.
03 FILLER	PIC XX	VALUE ALL 'M'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID	PIC X(20).	
03 BENEFIT-RCD-NO	PIC 999	COMP.
03 COBRA-EVENT-ID	PIC 999	COMP.
03 DEPENDENT-BENEF	PIC XX.	
03 COBRA-EMPLID	PIC X(11).	
03 PLAN-TYPE	PIC XX.	
03 COBRA-EVENT-TYPE	PIC X.	
03 CBR-INIT-EVENT-ID	PIC 999	COMP.
03 CBR-SCND-EVENT-ID	PIC 999	COMP.
03 CBR-PROCESS-STATUS	PIC X.	
03 CBR-ENROLL-STATUS	PIC X.	
03 CBR-PLN-REPRCS-IND	PIC X.	
03 FILLER	PIC X	VALUE 'Z'.

/*****

* BEN_PROG_PARTIC SQL DELETE STMT *

01 D-BP-PGM.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_D_BP_PGM'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 FILLER PIC X VALUE 'Z'.

/*****

* BENEFIT_PARTIC - SQL DELETE STMT *

01 D-BP-PLN.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_D_BP_PLN'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EMPL-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 PLAN-TYPE PIC XX.
03 FILLER PIC X VALUE 'Z'.

/*****

* HEALTH_BENEFIT SQL DELETE STMT *

01 D-HTH.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_D_HTH'.

/*****

* HEALTH_DEPENDNT SQL DELETE STMT *

01 D-HTH-DEP.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_D_HTH_DEP'.

/*****

* FSA_BENEFIT SQL DELETE STMT *

01 D-FSA.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_D_FSA'.

/*****

* INIT COBRA-PARTIC-PLAN END-DT SQL BUFFER AND STMT *

01 S-INITEND.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_INITEND'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'H'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 DEPENDENT-BENEF PIC XX.
03 PLAN-TYPE PIC XX.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 COVERAGE-END-DT PIC X(10).
03 COBRA-TERM-DT PIC X(10).
03 CBR-ENROLL-STATUS PIC X.
88 CBR-ENROLLED VALUE 'E'.
03 FILLER PIC X VALUE 'Z'.

/*****

* BI_BEN_ENR_EFDT SQL SELECT STMT *

*****/

01 S-BI-EFDT.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_BI_EFDT'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.

03 PLAN-TYPE PIC XX.

03 COBRA-EVENT-ID PIC 999 COMP.

03 EFFDT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 RATE-PERCENT PIC 999 COMP.

03 FILLER PIC X VALUE 'Z'.

/*****

* HEALTH_BENEFIT SQL UPDATE TERM-DT STMT *

01 U-HTHTERM.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_U-HTHTERM'.

02 BIND-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EFFDT PIC X(10) VALUE SPACES.

03 COVERAGE-BEGIN-DT PIC X(10) VALUE SPACES.

03 EMPLID PIC X(20) VALUE SPACES.

03 EMPL-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 PLAN-TYPE PIC XX VALUE SPACES.

03 EFFDT-2 PIC X(10) VALUE SPACES.

03 FILLER PIC X VALUE 'Z'.

/*****

* FSA_BENEFIT SQL UPDATE TERM-DT STMT *

01 U-FSATERM.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_U_FSATERM'.

/*****

* COBRA-EVENT SQL DELETE STMT *

01 D-CBRACTY.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_D_CBRACTY'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC XXX VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EMPL-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-DT PIC X(10).
03 COBRA-ACTION PIC XXX.
03 FILLER PIC X VALUE 'Z'.

/*****

* CBR_PARTIC SQL DELETE STMT *

01 D-PAR.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_D_PAR'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 FILLER PIC X VALUE 'Z'.

/*****

* CBR_PARTIC_PLAN SQL DELETE STMT *

01 D-PARPLN.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_D_PARPLAN'.

/*****

* CBR_PARTIC_OPTN SQL DELETE STMT *

01 D-PAROPTN.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_D_PAROPTN'.

/*****

* CBR_PARTIC_DPND SQL DELETE STMT *

01 D-PARDPD1.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_D_PARDPD1'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 FILLER PIC X VALUE 'Z'.

/*****

* CBR_PARTIC_DPND SQL DELETE STMT *

01 D-PARDPND.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_D_PARDPD2'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'H'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 DEPENDENT-BENEF PIC XX.
03 PLAN-TYPE PIC XX.
03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA_EVENT_RULES SQL BUFFER AND STMT *

01 S-CBR-EVT.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_CBR_EVT'.

02 BIND-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EFFDT PIC X(10).

03 EFFDT-2 PIC X(10).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XXX VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 COBRA-EVENT-CLASS PIC XXX.

03 EFFDT PIC X(10).

03 MM-COVERAGE PIC 99 COMP.

03 ADDL-MM-DISABLED PIC 99 COMP.

03 USE-BAS-EVENT-RULE PIC X.

03 COBRA-PD-BEGINS-CD PIC X.

03 INCL-ALT-COVRG PIC X.

03 EVENT-NOTIFY-DAYS PIC 99 COMP.

03 RESPONSE-DAYS PIC 99 COMP.

03 PAYMENT-GRACE-DAYS PIC 99 COMP.

03 WAIVE-COBRA-SURCHG PIC X.

03 SECOND-EVENT-ROLE PIC X.

03 SECOND-EVENT-MONTH PIC 99 COMP.

03 SECOND-EVENT-MODE PIC X.

03 BENEFIT-LOST-LEVEL PIC X.

03 EFF-STATUS PIC X.

03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA_EVENT_BENEF SQL BUFFER AND STMT *

01 S-CBR-BEN.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_CBR_BEN'.

02 BIND-SETUP.

03 FILLER PIC XXX VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 COBRA-EVENT-CLASS PIC XXX.

03 EFFDT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XXX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 COBRA-EVENT-CLASS PIC XXX.

03 COVERED-PERSON-TYP PIC XX.

03 FILLER PIC X VALUE 'Z'.

/*****

* ACTIVE BEN_DEFN_PGM SQL BUFFER AND STMT *

01 S-BENPGM.

```

02 SQL-STMT          PIC X(18).

      88 SQL-STMT-OVGPGM          VALUE

                                     'PSPCOBRA_S_OVGPGM'.

      88 SQL-STMT-BENPGM          VALUE

                                     'PSPCOBRA_S_BENPGM'.

```

02 BIND-SETUP-OVGPGM.

```

      03 FILLER          PIC X(10)  VALUE ALL 'C'.

      03 FILLER          PIC X(10)  VALUE ALL 'H'.

      03 FILLER          PIC X(10)  VALUE ALL 'C'.

      03 FILLER          PIC X(10)  VALUE ALL 'D'.

      03 FILLER          PIC X(10)  VALUE ALL 'A'.

      03 FILLER          PIC X      VALUE 'Z'.

```

02 BIND-DATA-OVGPGM.

```

      03 BENEFIT-PROGRAM  PIC X(10).

      03 BENEFIT-PROGRAM-2 PIC X(10).

      03 BENEFIT-PROGRAM-3 PIC X(10).

      03 EFFDT           PIC X(10).

      03 EFFDT-2         PIC X(10).

      03 FILLER          PIC X      VALUE 'Z'.

```

02 BIND-SETUP.

```

      03 FILLER          PIC X(10)  VALUE ALL 'C'.

      03 FILLER          PIC X(10)  VALUE ALL 'D'.

      03 FILLER          PIC X(10)  VALUE ALL 'A'.

      03 FILLER          PIC X      VALUE 'Z'.

```

02 BIND-DATA.

03 BENEFIT-PROGRAM PIC X(10).
03 EFFDT PIC X(10).
03 EFFDT-2 PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 BENEFIT-PROGRAM PIC X(10) VALUE SPACES.
03 EFFDT PIC X(10) VALUE SPACES.
03 COBRA-SURCHARGE PIC 99 COMP.
03 COBRA-DISABL-SURCG PIC 99 COMP.
03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA BEN_DEFN_PLN SQL BUFFER AND STMT *

01 S-CBRPLN.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_CBRPLN'.

02 BIND-SETUP.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 BENEFIT-PROGRAM PIC X(10).

03 EFFDT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'H'.

03 FILLER PIC X(5) VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 PLAN-TYPE PIC XX.

03 EVENT-RULES-ID PIC X(10).

03 DEP-RULE-ID PIC X(5) VALUE SPACES.

03 FILLER PIC X VALUE 'Z'.

/*****

* DEP_RULE_TBL SQL BUFFER AND STMT *

*****/

01 S-DEPRUL.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_DEPRUL'.

02 BIND-SETUP.

03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 BENEFIT-PROGRAM PIC X(10).
03 EFFDT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 DEP-AGE-LIMIT PIC 99 COMP.
 03 STUDENT-AGE-LIMIT PIC 99 COMP.
 03 EXCL-DISABLED-AGE PIC X.
 03 DEPENDENT-MARRIAGE PIC X.
 03 OVERAGE-PD-BEG-CD PIC X.
 03 FILLER PIC X VALUE 'Z'.

/*****

* PRIOR CBR_PARTIC_PLAN SQL BUFFER AND STMT *

01 S-PRREVT.

02 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_PRREVT'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-DT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE 'Z'.

02 SELECT-DATA.

03 PLAN-TYPE PIC XX.

03 FILLER PIC X VALUE 'Z'.

/*****

* COVRG_CD_TBL SQL BUFFER AND STMT *

*****/

01 S-CVGCD.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_CVGCD'.

02 BIND-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EFFDT PIC X(10).

03 EFFDT-2 PIC X(10).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(2) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 COVRG-CD PIC X(2).

03 CBR-COVRG-SET PIC XX.

03 CC-CTL-TOTAL-FLG PIC X.

03 MIN-NUM-TOTAL PIC 99 COMP.

03 MAX-NUM-TOTAL PIC 99 COMP.

03 EFFDT PIC X(10).

03 COVERED-PERSON-TYP PIC XX.

03 MIN-NUM-COVERED PIC 99 COMP.

03 MAX-NUM-COVERED PIC 99 COMP.

03 FILLER PIC X VALUE 'Z'.

/*****

* DEPENDENT_BENEF SQL BUFFER AND STMT *

*****/

01 S-DEPEND.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_DEPEND'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EFFDT-1 PIC X(10).
03 EFFDT-2 PIC X(10).
03 EFFDT-3 PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X(4) VALUE ALL 'C'.

03 FILLER.

04 FILLER	PIC XXX	VALUE ALL 'H'.
04 FILLER	PIC X(55)	VALUE ALL 'C'.
04 FILLER	PIC X(55)	VALUE ALL 'H'.
04 FILLER	PIC X(55)	VALUE ALL 'C'.
04 FILLER	PIC X(55)	VALUE ALL 'H'.
04 FILLER	PIC X(30)	VALUE ALL 'C'.
04 FILLER	PIC X(6)	VALUE ALL 'H'.
04 FILLER	PIC XXXX	VALUE ALL 'C'.
04 FILLER	PIC XX	VALUE ALL 'H'.
04 FILLER	PIC XX	VALUE ALL 'C'.
04 FILLER	PIC XXXX	VALUE ALL 'H'.
04 FILLER	PIC XXXX	VALUE ALL 'C'.
04 FILLER	PIC X(30)	VALUE ALL 'H'.
04 FILLER	PIC X(6)	VALUE ALL 'C'.
04 FILLER	PIC X(20)	VALUE ALL 'H'.
04 FILLER	PIC X(11)	VALUE ALL 'C'.
04 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC XX	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X	VALUE ALL 'C'.

03 FILLER	PIC XX	VALUE ALL 'H'.	FEDMERG
03 FILLER	PIC X(10)	VALUE ALL 'D'.	
03 FILLER	PIC XXX	VALUE ALL 'H'.	
03 FILLER	PIC X(10)	VALUE ALL 'D'.	
03 FILLER	PIC X(11)	VALUE ALL 'C'.	
03 FILLER	PIC X	VALUE ALL 'H'.	
03 FILLER	PIC X	VALUE ALL 'C'.	
03 FILLER	PIC X	VALUE ALL 'H'.	
03 FILLER	PIC X	VALUE 'Z'.	

02 SELECT-DATA.

03 DEPENDENT-BENEF PIC XX.

03 SAME-ADDRESS-EMPL PIC X.

88 SAME-ADDRESS-EMPL-NO VALUE 'N'.

03 ADDRESS-TYPE PIC X(4).

03 DEP-ADDRESS-SBR.

04 COUNTRY PIC XXX.

04 ADDRESS1 PIC X(55).

04 ADDRESS2 PIC X(55).

04 ADDRESS3 PIC X(55).

04 ADDRESS4 PIC X(55).

04 CITY PIC X(30).

04 NUM1 PIC X(6).

04 NUM2 PIC XXXX.

04 HOUSE-TYPE PIC XX.

04 ADDR-FIELD1 PIC XX.

04	ADDR-FIELD2	PIC XXXX.	
04	ADDR-FIELD3	PIC XXXX.	
04	COUNTY	PIC X(30).	
04	STATE	PIC X(6).	
04	POSTAL	PIC X(20).	
04	GEO-CODE	PIC X(11).	
04	IN-CITY-LIMIT	PIC X.	
03	EFFDT	PIC X(10).	
03	COVERED-PERSON-TYP	PIC XX.	
03	DEP-BENEF-TYPE	PIC X.	
03	MAR-STATUS	PIC X.	
03	SEX	PIC X.	
03	BIRTHDATE	PIC X(10).	
03	DT-OF-DEATH	PIC X(10).	
03	STUDENT	PIC X.	
03	DISABLED	PIC XX.	FEDMERG
03	MEDICARE-ENTLD-DT	PIC X(10).	
03	COBRA-ACTION	PIC XXX.	
03	COBRA-EVENT-DT	PIC X(10).	
03	COBRA-EMPLID	PIC X(11).	
03	SMOKER	PIC X.	
03	AGE-LIMIT-FLG	PIC X.	
03	EFF-STATUS	PIC X.	
88	EFF-STATUS-ACTIVE	VALUE 'A'.	
88	EFF-STATUS-INACTIVE	VALUE 'I'.	
03	FILLER	PIC X	VALUE 'Z'.

/*****

* STUDENT TRANSITION DATE SQL BUFFER AND STMT *

*****/

01 S-STDNT.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_STDNT'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'H'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 DEPENDENT-BENEF PIC XX.
03 EFFDT-1 PIC X(10).
03 EFFDT-2 PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 STUDENT-STATUS-DT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* DEP_BENEF_NID SQL BUFFER AND STMT *

*****/

01 S-DEP-NID.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_DEP_NID'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 DEPENDENT-BENEF PIC XX.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XXX VALUE ALL 'C'.

03 FILLER PIC XXXX VALUE ALL 'H'.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 NID-COUNTRY PIC XXX.

03 NATIONAL-ID-TYPE PIC XXXX.

03 NATIONAL-ID PIC X(20).

03 PRIMARY-NID PIC X.

03 FILLER PIC X VALUE 'Z'.

/*****

* BEN_PROG_PARTIC SQL BUFFER AND STMT *

01 S-EE-PGM.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_EE_PGM'.

02 BIND-SETUP.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X(20) VALUE ALL 'H'.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X(20) VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 BENEFIT-PROGRAM PIC X(10).
03 EFFDT PIC X(10).
03 EFFDT-2 PIC X(10).
03 EMPLID PIC X(20).
03 EMPLID-2 PIC X(20).
03 EMPLID-3 PIC X(20).
03 EMPLID-4 PIC X(20).
03 EMPL-RCD-NO PIC 999 COMP.
03 EMPL-RCD-NO-2 PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
* 03 FILLER PIC X VALUE ALL 'C'.
* 03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 BENEFIT-PROGRAM PIC X(10).

03 BENEFIT-RCD-NO PIC 999 COMP.

* 03 PER-STATUS PIC X.

* 03 PER-TYPE PIC X.

03 FILLER PIC X VALUE 'Z'.

/*****

* JOB SQL BUFFER AND STMT *

01 S-CURJOB.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_CURJOB'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.

03 EFFDT PIC X(10).

03 EFFDT-2 PIC X(10).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 COMPANY PIC X(10).

03 PAYGROUP PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* PERSONAL-DATA SQL BUFFER AND STMT *

01 S-PERDATA.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_PERDATA'.

02 BIND-SETUP.

03 FILLER	PIC X(20) VALUE ALL 'C'.
03 FILLER	PIC X(10) VALUE ALL 'D'.
03 FILLER	PIC X(10) VALUE ALL 'A'.
03 FILLER	PIC X(10) VALUE ALL 'D'.
03 FILLER	PIC X(10) VALUE ALL 'A'.
03 FILLER	PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID	PIC X(20).
03 EFFDT	PIC X(10).
03 EFFDT-2	PIC X(10).
03 EFFDT-3	PIC X(10).
03 EFFDT-4	PIC X(10).
03 FILLER	PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER	PIC X(10) VALUE ALL 'D'.
03 FILLER	PIC X(10) VALUE ALL 'A'.
03 FILLER	PIC X VALUE ALL 'C'.
03 FILLER	PIC X VALUE ALL 'H'.
03 FILLER.	
04 FILLER	PIC XXX VALUE ALL 'C'.
04 FILLER	PIC X(55) VALUE ALL 'H'.
04 FILLER	PIC X(55) VALUE ALL 'C'.
04 FILLER	PIC X(55) VALUE ALL 'H'.
04 FILLER	PIC X(55) VALUE ALL 'C'.

04 FILLER	PIC X(30)	VALUE ALL 'H'.
04 FILLER	PIC X(6)	VALUE ALL 'C'.
04 FILLER	PIC XXXX	VALUE ALL 'H'.
04 FILLER	PIC XX	VALUE ALL 'C'.
04 FILLER	PIC XX	VALUE ALL 'H'.
04 FILLER	PIC XXXX	VALUE ALL 'C'.
04 FILLER	PIC XXXX	VALUE ALL 'H'.
04 FILLER	PIC X(30)	VALUE ALL 'C'.
04 FILLER	PIC X(6)	VALUE ALL 'H'.
04 FILLER	PIC X(20)	VALUE ALL 'C'.
04 FILLER	PIC X(11)	VALUE ALL 'H'.
04 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE 'Z'.

02 SELECT-DATA.

03 BIRTHDATE	PIC X(10).
03 MEDICARE-ENTLD-DT	PIC X(10).
03 SEX	PIC X.
03 HIGHLY-COMP-EMPL-C	PIC X.
03 EMPL-ADDRESS-SBR.	
04 COUNTRY	PIC XXX.
04 ADDRESS1	PIC X(55).
04 ADDRESS2	PIC X(55).
04 ADDRESS3	PIC X(55).
04 ADDRESS4	PIC X(55).
04 CITY	PIC X(30).

04	NUM1	PIC X(6).
04	NUM2	PIC XXXX.
04	HOUSE-TYPE	PIC XX.
04	ADDR-FIELD1	PIC XX.
04	ADDR-FIELD2	PIC XXXX.
04	ADDR-FIELD3	PIC XXXX.
04	COUNTY	PIC X(30).
04	STATE	PIC X(6).
04	POSTAL	PIC X(20).
04	GEO-CODE	PIC X(11).
04	IN-CITY-LIMIT	PIC X.
03	FILLER	PIC X VALUE 'Z'.

/*****

* SMOKER SQL BUFFER AND STMT *

01 S-SMOKER.

02	SQL-STMT	PIC X(18) VALUE
		'PSPCOBRA_S_SMOKER'.

02 BIND-SETUP.

03	FILLER	PIC X(20) VALUE ALL 'C'.
03	FILLER	PIC X(10) VALUE ALL 'D'.
03	FILLER	PIC X(10) VALUE ALL 'A'.
03	FILLER	PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 SMOKER-DT PIC X(10).
03 SMOKER-DT-1 PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 SMOKER PIC X.
03 FILLER PIC X VALUE 'Z'.

/*****

* ADDRESSES SQL BUFFER AND STMT *

01 S-ADDRESS.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_ADDRESS'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC X(4) VALUE ALL 'H'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 ADDRESS-TYPE PIC X(4).
03 EFFDT-1 PIC X(10).
03 EFFDT-2 PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER.
04 FILLER PIC XXX VALUE ALL 'C'.
04 FILLER PIC X(55) VALUE ALL 'H'.
04 FILLER PIC X(55) VALUE ALL 'C'.
04 FILLER PIC X(55) VALUE ALL 'H'.
04 FILLER PIC X(55) VALUE ALL 'C'.
04 FILLER PIC X(30) VALUE ALL 'H'.
04 FILLER PIC X(6) VALUE ALL 'C'.
04 FILLER PIC XXXX VALUE ALL 'H'.
04 FILLER PIC XX VALUE ALL 'C'.
04 FILLER PIC XX VALUE ALL 'H'.
04 FILLER PIC XXXX VALUE ALL 'C'.
04 FILLER PIC XXXX VALUE ALL 'H'.
04 FILLER PIC X(30) VALUE ALL 'C'.
04 FILLER PIC X(6) VALUE ALL 'H'.
04 FILLER PIC X(20) VALUE ALL 'C'.
04 FILLER PIC X(11) VALUE ALL 'H'.

04 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EMPL-ADDRESS-SBR.

04 COUNTRY PIC XXX.
04 ADDRESS1 PIC X(55).
04 ADDRESS2 PIC X(55).
04 ADDRESS3 PIC X(55).
04 ADDRESS4 PIC X(55).
04 CITY PIC X(30).
04 NUM1 PIC X(6).
04 NUM2 PIC XXXX.
04 HOUSE-TYPE PIC XX.
04 ADDR-FIELD1 PIC XX.
04 ADDR-FIELD2 PIC XXXX.
04 ADDR-FIELD3 PIC XXXX.
04 COUNTY PIC X(30).
04 STATE PIC X(6).
04 POSTAL PIC X(20).
04 GEO-CODE PIC X(11).
04 IN-CITY-LIMIT PIC X.
03 FILLER PIC X VALUE 'Z'.

/*****

* DISABLE SQL BUFFER AND STMT *

01 S-DISABLE.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_DISABLE'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 FILLER PIC X(20).

03 DISABLED PIC X.

88 DISABLED-YES VALUE 'Y'.

88 DISABLED-NO VALUE 'N'.

03 DISABLED-VET PIC X.

88 DISABLED-VET-YES VALUE 'Y'.

88 DISABLED-VET-NO VALUE 'N'.

03 FILLER PIC X VALUE 'Z'.

/*****

* BEN_PROG_PARTIC SQL BUFFER AND STMT *

01 S-BP-PGM.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_BP_PGM'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.

03 EFFDT PIC X(10).

03 EFFDT-2 PIC X(10).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 BENEFIT-PROGRAM PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* HEALTH_BENEFIT SQL BUFFER AND STMT *

*****/

01 S-HTH.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_HTH'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 COVERAGE-END-DT PIC X(10).
03 COVERAGE-BEGIN-DT PIC X(10).
03 COVERAGE-BEGIN-DT-2 PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC X(2) VALUE ALL 'H'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X(30) VALUE ALL 'H'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X(30) VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 PLAN-TYPE PIC XX.
03 BENEFIT-NO PIC 999 COMP.

03 EFFDT PIC X(10).

03 DEDUCTION-END-DT PIC X(10).

03 COVERAGE-BEGIN-DT PIC X(10).

03 COVERAGE-END-DT PIC X(10).

03 BENEFIT-PLAN PIC X(10).

03 COVRG-CD PIC X(2).

03 COVERAGE-ELECT PIC X.

03 HLTH-PROVIDER-ID PIC X(30).

03 PREVIOUSLY-SEEN PIC X.

03 OTH-INSURANCE-IND PIC X.

03 OTH-INSURANCE-NAME PIC X(30).

03 FILLER PIC X VALUE 'Z'.

/*****

* HEALTH_DEPENDENT SQL BUFFER AND STMT *

01 S-HTH-DEP.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_HTH_DEP'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EMPL-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 PLAN-TYPE PIC XX.
03 BENEFIT-NO PIC 999 COMP.
03 EFFDT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X(30) VALUE ALL 'H'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X(30) VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 DEPENDENT-BENEF PIC XX.
03 HLTH-PROVIDER-ID PIC X(30).
03 PREVIOUSLY-SEEN PIC X.
03 OTH-INSURANCE-IND PIC X.

03 OTH-INSURANCE-NAME PIC X(30).

03 FILLER PIC X VALUE 'Z'.

/*****

* FSA_BENEFIT SQL BUFFER AND STMT *

01 S-FSA.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_FSA'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC XXXX VALUE '2PPP'.

03 FILLER PIC XXXX VALUE '2PPP'.

03 FILLER PIC XXXX VALUE '2PPP'.

03 FILLER PIC X(5) VALUE '2PPPP'.

03 FILLER PIC X(5) VALUE '2PPPP'.

03 FILLER PIC X(5) VALUE '2PPPP'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 PLAN-TYPE PIC XX.

03 BENEFIT-NO PIC 999 COMP.

03 EFFDT PIC X(10).

03 DEDUCTION-END-DT PIC X(10).

03 COVERAGE-BEGIN-DT PIC X(10).

03 COVERAGE-END-DT PIC X(10).

03 BENEFIT-PLAN PIC X(10).

03 COVERAGE-ELECT PIC X.

03 EMPL-CONTRIBUTN-AMT PIC S9999V99 COMP-3.

03 ANN-EX-CREDIT-FSA PIC S9(5)V99 COMP-3.

03 ANNUAL-PLEDGE PIC S9(5)V99 COMP-3.

03 FSA-SUB-AMT-YTD PIC S9(6)V99 COMP-3.

03 FSA-APR-AMT-YTD PIC S9(6)V99 COMP-3.

03 FSA-PD-AMT-YTD PIC S9(6)V99 COMP-3.

03 FILLER PIC X VALUE 'Z'.

/*****

* TERMINATED HEALTH_BENEFIT SQL BUFFER AND STMT *

01 S-HTHTERM.

02 SQL-STMT PIC X(18) VALUE

'PSPCOBRA_S_HTHTERM'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EMPL-RCD-NO PIC 999 COMP.
03 PLAN-TYPE PIC XX.
03 COVERAGE-BEGIN-DT PIC X(10).
03 COVERAGE-BEGIN-DT-2 PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EFFDT PIC X(10).
03 COVERAGE-BEGIN-DT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

/*****

* TERMINATED FSA_BENEFIT SQL BUFFER AND STMT *

01 S-FSATERM.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_FSATERM'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EFFDT PIC X(10).

03 COVERAGE-BEGIN-DT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* TERMINATED HEALTH_DEPENDENT SQL BUFFER AND STMT *

01 S-DEPTERM.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_DEPTERM'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EMPL-RCD-NO PIC 999 COMP.
03 PLAN-TYPE PIC XX.
03 COVERAGE-BEGIN-DT PIC X(10).
03 COVERAGE-BEGIN-DT-2 PIC X(10).
03 DEPENDENT-BENEF PIC XX.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EFFDT PIC X(10).

03 COVERAGE-BEGIN-DT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* MEDICARE WAIVED HEALTH_BENEFIT SQL BUFFER AND STMT *

01 S-HTHWAIV.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_HTHWAIV'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.

03 PLAN-TYPE PIC XX.

03 COVERAGE-BEGIN-DT PIC X(10).

03 COVERAGE-BEGIN-DT-2 PIC X(10).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EFFDT PIC X(10).

03 COVERAGE-BEGIN-DT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* MEDICARE WAIVED FSA_BENEFIT SQL BUFFER AND STMT *

01 S-FSAWAIV.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_FSAWAIV'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EFFDT PIC X(10).

03 COVERAGE-BEGIN-DT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* CURRENT COBRA HEALTH_BENEFIT SQL BUFFER AND STMT *

01 S-CBRHTH.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_CBRHTH'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 PLAN-TYPE PIC XX.

03 EFFDT PIC X(10).

03 BENEFIT-PROGRAM PIC X(10).

03 EFFDT-2 PIC X(10).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X(2) VALUE ALL 'H'.

03 FILLER PIC XXX VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 BENEFIT-PLAN PIC X(10).

03 COVRG-CD PIC X(2).

03 OPTION-CD PIC XXX.

03 FILLER PIC X VALUE 'Z'.

/*****

* BEN-DEFN-OPTN SQL BUFFER AND STMT *

01 S-BENOPTN.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_BENOPTN'.

02 BIND-SETUP.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.
03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC X(2) VALUE ALL 'H'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 BENEFIT-PROGRAM PIC X(10).
03 PLAN-TYPE PIC XX.
03 BENEFIT-PLAN PIC X(10).
03 COVRG-CD PIC X(2).
03 EFFDT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XXX VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 OPTION-CD PIC XXX.
03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA-ACTIVITY SQL INSERT STMT *

01 I-CBRACTY.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_I_CBRACTY'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC XXX VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC XXXX VALUE ALL 'I'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20) VALUE SPACES.

03 EMPL-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-DT PIC X(10) VALUE SPACES.

03 COBRA-ACTION PIC XXX VALUE SPACES.

03 CBR-ACTION-SOURCE PIC X VALUE SPACES.

03 BENEFIT-RCD-NO PIC 999 COMP.

03 SCHED-ID PIC X(10) VALUE SPACES.

03 EVENT-ID PIC 9(6) COMP.

03 CBR-COV-OVERRIDE PIC X VALUE SPACES.

88 CBR-COV-OVERRIDE-NO VALUE 'N'.

03 FILLER PIC X VALUE 'Z'.

/*****

* DEP-BEN SQL UPDATE STMT *

01 U-DEPEND.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_U_DEPEND'.

02 BIND-SETUP.

03 FILLER PIC X(11) VALUE ALL 'C'.

03 FILLER PIC XXX VALUE ALL 'H'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 COBRA-EMPLID PIC X(11) VALUE SPACES.

03 COBRA-ACTION PIC XXX VALUE SPACES.

03 COBRA-EVENT-DT PIC X(10) VALUE SPACES.

03 EMPLID PIC X(20) VALUE SPACES.

03 DEPENDENT-BENEF PIC XX VALUE SPACES.

03 FILLER PIC X VALUE 'Z'.

/*****

* RELATED EMPLOYEE OF COBRA PARTIC SQL BUFFER AND STMT *

*****/

01 S-RELEMP.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_RELEMP'.

02 BIND-SETUP.

03 FILLER PIC X(11) VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 COBRA-EMPLID PIC X(11).
03 EFFDT PIC X(10).
03 EFFDT-2 PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'H'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'H'. FEDMERG

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID PIC X(20).

03 DEPENDENT-BENEF PIC XX.

03 COVERED-PERSON-TYP PIC XX.

03 DISABLED PIC XX. FEDMERG

03 BIRTHDATE PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* INIT COBRA-PARTIC-PLAN EXISTS? SQL BUFFER AND STMT *

01 S-INITPLN.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_INITPLN'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 DEPENDENT-BENEF PIC XX.
03 PLAN-TYPE PIC XX.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 CBR-ENROLL-STATUS PIC X.
03 COBRA-TERM-DT PIC X(10).
03 COVERAGE-END-DT PIC X(10).
03 COVERAGE-BEGIN-DT PIC X(10).
03 DISABL-CVG-BEG-DT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA-ACTIVITY SQL SELECT STMT *

*****/

01 S-CBRCTY.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_CBRCTY'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X(20) VALUE ALL 'H'.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X(20) VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC XXX VALUE ALL 'C'.

03 FILLER PIC XXX VALUE ALL 'H'.

03 FILLER PIC XXX VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EMPLID-2 PIC X(20).

03 EMPLID-3 PIC X(20).

03 EMPLID-4 PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.
03 EMPL-RCD-NO-2 PIC 999 COMP.
03 COBRA-EVENT-DT PIC X(10).
03 COBRA-EVENT-DT-2 PIC X(10).
03 COBRA-ACTION PIC XXX.
03 COBRA-ACTION-2 PIC XXX.
03 COBRA-ACTION-3 PIC XXX.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC XXX VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC XXXX VALUE ALL 'I'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID PIC X(20).
03 EMPL-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-DT PIC X(10).
 03 COBRA-ACTION PIC XXX.
 03 CBR-ACTION-SOURCE PIC X.
 88 CBR-SOURCE-BENADMIN VALUE 'B'.
 88 CBR-SOURCE-DEP-CHG VALUE 'D'.
 88 CBR-SOURCE-JOB-CHG VALUE 'J'.
 88 CBR-SOURCE-OVG-PROCESS VALUE 'O'.
 88 CBR-SOURCE-PERS-DATA-CHG VALUE 'P'.
 88 CBR-SOURCE-MANUAL-EVENT VALUE 'M'.
 03 BENEFIT-RCD-NO PIC 999 COMP.
 03 SCHED-ID PIC X(10).
 03 EVENT-ID PIC 9(6) COMP.
 03 CBR-COV-OVERRIDE PIC X.
 03 CBR-ALT-COV-TRM-DT PIC X(10).
 03 CBR-COV-AS-OF-DT PIC X(10).
 03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA_EVENT SQL BUFFER AND STMT *

01 S-EVENTID.

02 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_EVENTID'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 COBRA-EVENT-ID PIC 999 COMP.

03 FILLER PIC X VALUE 'Z'.

/*****

* MULTI-JOB SQL BUFFER AND STMT *

01 S-MJOB.

02 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_MJOB'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EFFDT PIC X(10).
03 EFFDT-2 PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EMPL-RCD-NO PIC 999 COMP.
03 BENEFIT-RCD-NO PIC 999 COMP.
03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA-EVENT SQL INSERT STMT *

01 I-EVT.

02 SQL-STMT PIC X(18) VALUE

'PSPCOBRA_I_EVT'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC XXX VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'H'.

03 FILLER PIC XXXX VALUE ALL 'I'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20) VALUE SPACES.

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 COBRA-EVENT-DT PIC X(10) VALUE SPACES.

03 COBRA-EVENT-CLASS PIC XXX VALUE SPACES.
 03 SCHED-ID PIC X(10) VALUE SPACES.
 03 EVENT-ID PIC 9(6) COMP.
 03 EMPL-RCD-NO PIC 999 COMP.
 03 CBR-ACTION-SOURCE PIC X VALUE SPACES.
 03 CBR-QUALIFY-STATUS PIC XX VALUE SPACES.
 03 CBR-PROCESS-STATUS PIC X VALUE SPACES.
 03 CBR-EVT-REPRCS-IND PIC X VALUE SPACES.
 03 CBR-EVENT-CONFLICT PIC X VALUE SPACES.
 03 BAS-DATA-CHG PIC X VALUE SPACES.
 03 CBR-COV-OVERRIDE PIC X VALUE SPACES.
 03 CBR-ALT-COV-TRM-DT PIC X(10) VALUE SPACES.
 03 CBR-COV-AS-OF-DT PIC X(10) VALUE SPACES.
 03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA EVENT SQL BUFFER AND STMT *

01 S-EVT.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_EVT'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
 03 FILLER PIC X(20) VALUE ALL 'H'.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC X(20) VALUE ALL 'H'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EMPLID-2 PIC X(20).
03 EMPLID-3 PIC X(20).
03 EMPLID-4 PIC X(20).
03 BENEFIT-RCD-NO PIC 999 COMP.
03 BENEFIT-RCD-NO-2 PIC 999 COMP.
03 COBRA-EVENT-DT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC XXX VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'H'.
03 FILLER PIC XXXX VALUE ALL 'I'.
03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X	VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID	PIC X(20).	
03 BENEFIT-RCD-NO	PIC 999	COMP.
03 COBRA-EVENT-ID	PIC 999	COMP.
03 COBRA-EVENT-DT	PIC X(10).	
03 COBRA-EVENT-CLASS	PIC XXX.	
03 SCHED-ID	PIC X(10).	
03 EVENT-ID	PIC 9(6)	COMP.
03 EMPL-RCD-NO	PIC 999	COMP.
03 CBR-QUALIFY-STATUS	PIC XX.	
03 CBR-PROCESS-STATUS	PIC X.	
03 CBR-EVT-REPRCS-IND	PIC X.	
03 CBR-EVENT-CONFLICT	PIC X.	
03 BAS-DATA-CHG	PIC X.	
03 CBR-COV-OVERRIDE	PIC X.	
03 CBR-ALT-COV-TRM-DT	PIC X(10).	

03 CBR-COV-AS-OF-DT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA PARTIC SQL BUFFER AND STMT *

01 S-PAR1.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_PAR1'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER	PIC XX	VALUE ALL 'M'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'H'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(11)	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'H'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID	PIC X(20).	
03 BENEFIT-RCD-NO	PIC 999	COMP.
03 COBRA-EVENT-ID	PIC 999	COMP.
03 DEPENDENT-BENEF	PIC XX.	
03 CBR-QUALIFY-STATUS	PIC XX.	
03 CBR-PROCESS-STATUS	PIC X.	

03 CBR-INIT-EVENT-STS PIC XX.
 03 CBR-SCND-EVENT-STS PIC XX.
 03 CBR-INIT-NOTIFY-DT PIC X(10).
 03 CBR-SCND-NOTIFY-DT PIC X(10).
 03 COBRA-ELECT PIC X.
 03 COBRA-ELECT-DT PIC X(10).
 03 COBRA-WAIVE-DT PIC X(10).
 03 COBRA-REVOKE-DT PIC X(10).
 03 COBRA-EMPLID PIC X(11).
 03 BENEFIT-PROGRAM PIC X(10).
 03 BAS-ASSIGN-STATUS PIC XX.
 03 CBR-TERM-NOTIFY-DT PIC X(10).
 03 CBR-PAR-REPRCS-IND PIC X.
 03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA PARTIC SQL BUFFER AND STMT *

01 S-PAR2.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_PAR2'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
 03 FILLER PIC XX VALUE ALL 'S'.
 03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 DEPENDENT-BENEF PIC XX.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(11) VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'H'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID PIC X(20).
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 DEPENDENT-BENEF PIC XX.
03 CBR-QUALIFY-STATUS PIC XX.
03 CBR-PROCESS-STATUS PIC X.
03 CBR-INIT-EVENT-STS PIC XX.
03 CBR-SCND-EVENT-STS PIC XX.
03 CBR-INIT-NOTIFY-DT PIC X(10).
03 CBR-SCND-NOTIFY-DT PIC X(10).
03 COBRA-ELECT PIC X.
03 COBRA-ELECT-DT PIC X(10).
03 COBRA-WAIVE-DT PIC X(10).
03 COBRA-REVOKE-DT PIC X(10).
03 COBRA-EMPLID PIC X(11).
03 BENEFIT-PROGRAM PIC X(10).
03 BAS-ASSIGN-STATUS PIC XX.
03 CBR-TERM-NOTIFY-DT PIC X(10).
03 CBR-PAR-REPRCS-IND PIC X.

03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA PARTIC SQL BUFFER AND STMT *

01 S-PRCSPAR.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_PRCSPAR'.

02 BIND-SETUP.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'H'.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'H'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X(20) VALUE ALL 'H'.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X(20) VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 BENEFIT-PROGRAM PIC X(10).

03 BENEFIT-PROGRAM-2 PIC X(10).

03 BENEFIT-PROGRAM-3 PIC X(10).

03 BENEFIT-PROGRAM-4 PIC X(10).

03 COBRA-EVENT-DT PIC X(10).

03 COBRA-EVENT-DT-2 PIC X(10).

03 EMPLID PIC X(20).

03 EMPLID-2 PIC X(20).

03 EMPLID-3 PIC X(20).

03 EMPLID-4 PIC X(20).

03 BENEFIT-RCD-NO PIC 999 COMP.

03 BENEFIT-RCD-NO-2 PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 COBRA-EVENT-ID-2 PIC 999 COMP.

03 DEPENDENT-BENEF PIC XX.

03 DEPENDENT-BENEF-2 PIC XX.

03 DEPENDENT-BENEF-3 PIC XX.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER	PIC XX	VALUE ALL 'M'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC XX	VALUE ALL 'S'.
03 FILLER	PIC X(10)	VALUE ALL 'C'.
03 FILLER	PIC XXXX	VALUE ALL 'I'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC XXX	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'H'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(11)	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'H'.
03 FILLER	PIC XX	VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID PIC X(20).
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 COBRA-EVENT-DT PIC X(10).
03 EMPL-RCD-NO PIC 999 COMP.
03 SCHED-ID PIC X(10).
03 EVENT-ID PIC 9(6) COMP.
03 EVT-PROCESS-STATUS PIC X.
03 EVT-QUALIFY-STATUS PIC XX.
03 CBR-EVT-REPRCS-IND PIC X.
03 COBRA-EVENT-CLASS PIC XXX.
03 CBR-COV-OVERRIDE PIC X.
03 CBR-ALT-COV-TRM-DT PIC X(10).
03 CBR-COV-AS-OF-DT PIC X(10).
03 DEPENDENT-BENEF PIC XX.
03 CBR-QUALIFY-STATUS PIC XX.
03 CBR-PROCESS-STATUS PIC X.
03 CBR-INIT-EVENT-STS PIC XX.
03 CBR-SCND-EVENT-STS PIC XX.
03 CBR-INIT-NOTIFY-DT PIC X(10).
03 CBR-SCND-NOTIFY-DT PIC X(10).

03 COBRA-ELECT PIC X.
 03 COBRA-ELECT-DT PIC X(10).
 03 COBRA-WAIVE-DT PIC X(10).
 03 COBRA-REVOKE-DT PIC X(10).
 03 COBRA-EMPLID PIC X(11).
 03 BENEFIT-PROGRAM PIC X(10).
 03 BAS-ASSIGN-STATUS PIC XX.
 03 CBR-TERM-NOTIFY-DT PIC X(10).
 03 CBR-PAR-REPRCS-IND PIC X.
 03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA PARTIC PLAN SQL BUFFER AND STMT *

01 S-PARPLN1.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_PARPLN1'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
 03 FILLER PIC XX VALUE ALL 'S'.
 03 FILLER PIC XX VALUE ALL 'M'.
 03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'H'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC XXX	VALUE ALL 'H'.
03 FILLER	PIC X(10)	VALUE ALL 'C'.
03 FILLER	PIC X(2)	VALUE ALL 'H'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(30)	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X(30)	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID	PIC X(20).	
03 BENEFIT-RCD-NO	PIC 999	COMP.
03 COBRA-EVENT-ID	PIC 999	COMP.
03 DEPENDENT-BENEF	PIC XX.	
03 PLAN-TYPE	PIC XX.	

03 COBRA-EVENT-TYPE	PIC X.	
03 CBR-INIT-EVENT-ID	PIC 999	COMP.
03 CBR-SCND-EVENT-ID	PIC 999	COMP.
03 INIT-BENEFIT-PGM	PIC X(10).	
03 CBR-PROCESS-STATUS	PIC X.	
03 CBR-ENROLL-STATUS	PIC X.	
03 CBR-PLN-REPRCS-IND	PIC X.	
03 COVERAGE-BEGIN-DT	PIC X(10).	
03 DISABL-CVG-BEG-DT	PIC X(10).	
03 COVERAGE-END-DT	PIC X(10).	
03 COBRA-ELECT-END-DT	PIC X(10).	
03 COBRA-ELECT	PIC X.	
03 COBRA-ELECT-DT	PIC X(10).	
03 COBRA-WAIVE-DT	PIC X(10).	
03 COBRA-REVOKE-DT	PIC X(10).	
03 COBRA-TERM-DT	PIC X(10).	
03 COBRA-TERM-REASON	PIC X.	
03 OPTION-CD	PIC XXX.	
03 BENEFIT-PLAN	PIC X(10).	
03 COVRG-CD	PIC X(2).	
03 EMPL-CONTRBUTN-AMT	PIC S9999V99	COMP-3.
03 ANN-EX-CREDIT-FSA	PIC S9(5)V99	COMP-3.
03 ANNUAL-PLEDGE	PIC S9(5)V99	COMP-3.
03 FSA-SUB-AMT-YTD	PIC S9(6)V99	COMP-3.
03 FSA-APR-AMT-YTD	PIC S9(6)V99	COMP-3.
03 FSA-PD-AMT-YTD	PIC S9(6)V99	COMP-3.

03 COBRA-ERROR PIC X.

03 HLTH-PROVIDER-ID PIC X(30).

03 PREVIOUSLY-SEEN PIC X.

03 OTH-INSURANCE-IND PIC X.

03 OTH-INSURANCE-NAME PIC X(30).

03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA PARTIC PLAN SQL BUFFER AND STMT *

01 S-PARPLN2.

02 SQL-STMT PIC X(18) VALUE

'PSPCOBRA_S_PARPLN2'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 DEPENDENT-BENEF PIC XX.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC XXX	VALUE ALL 'H'.
03 FILLER	PIC X(10)	VALUE ALL 'C'.
03 FILLER	PIC X(2)	VALUE ALL 'H'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(30)	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X(30)	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID	PIC X(20).
03 BENEFIT-RCD-NO	PIC 999 COMP.
03 COBRA-EVENT-ID	PIC 999 COMP.
03 DEPENDENT-BENEF	PIC XX.
03 PLAN-TYPE	PIC XX.
03 COBRA-EVENT-TYPE	PIC X.
03 CBR-INIT-EVENT-ID	PIC 999 COMP.

03 CBR-SCND-EVENT-ID	PIC 999	COMP.
03 INIT-BENEFIT-PGM	PIC X(10).	
03 CBR-PROCESS-STATUS	PIC X.	
03 CBR-ENROLL-STATUS	PIC X.	
03 CBR-PLN-REPRCS-IND	PIC X.	
03 COVERAGE-BEGIN-DT	PIC X(10).	
03 DISABL-CVG-BEG-DT	PIC X(10).	
03 COVERAGE-END-DT	PIC X(10).	
03 COBRA-ELECT-END-DT	PIC X(10).	
03 COBRA-ELECT	PIC X.	
03 COBRA-ELECT-DT	PIC X(10).	
03 COBRA-WAIVE-DT	PIC X(10).	
03 COBRA-REVOKE-DT	PIC X(10).	
03 COBRA-TERM-DT	PIC X(10).	
03 COBRA-TERM-REASON	PIC X.	
03 OPTION-CD	PIC XXX.	
03 BENEFIT-PLAN	PIC X(10).	
03 COVRG-CD	PIC X(2).	
03 EMPL-CONTRBUTN-AMT	PIC S9999V99	COMP-3.
03 ANN-EX-CREDIT-FSA	PIC S9(5)V99	COMP-3.
03 ANNUAL-PLEDGE	PIC S9(5)V99	COMP-3.
03 FSA-SUB-AMT-YTD	PIC S9(6)V99	COMP-3.
03 FSA-APR-AMT-YTD	PIC S9(6)V99	COMP-3.
03 FSA-PD-AMT-YTD	PIC S9(6)V99	COMP-3.
03 COBRA-ERROR	PIC X.	
03 HLTH-PROVIDER-ID	PIC X(30).	

03 PREVIOUSLY-SEEN PIC X.

03 OTH-INSURANCE-IND PIC X.

03 OTH-INSURANCE-NAME PIC X(30).

03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA PARTIC PLAN SQL BUFFER AND STMT *

01 S-PARPLN3.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_PARPLN3'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER	PIC X(10) VALUE ALL 'D'.
03 FILLER	PIC X(10) VALUE ALL 'A'.
03 FILLER	PIC X VALUE ALL 'C'.
03 FILLER	PIC X(10) VALUE ALL 'D'.
03 FILLER	PIC X(10) VALUE ALL 'A'.
03 FILLER	PIC X(10) VALUE ALL 'D'.
03 FILLER	PIC X(10) VALUE ALL 'A'.
03 FILLER	PIC X VALUE ALL 'C'.
03 FILLER	PIC XXX VALUE ALL 'H'.
03 FILLER	PIC X(10) VALUE ALL 'C'.
03 FILLER	PIC X(2) VALUE ALL 'H'.
03 FILLER	PIC XXXX VALUE '2PPP'.
03 FILLER	PIC XXXX VALUE '2PPP'.
03 FILLER	PIC XXXX VALUE '2PPP'.
03 FILLER	PIC X(5) VALUE '2PPPP'.
03 FILLER	PIC X(5) VALUE '2PPPP'.
03 FILLER	PIC X(5) VALUE '2PPPP'.
03 FILLER	PIC X VALUE ALL 'C'.
03 FILLER	PIC X(30) VALUE ALL 'H'.
03 FILLER	PIC X VALUE ALL 'C'.
03 FILLER	PIC X VALUE ALL 'H'.
03 FILLER	PIC X(30) VALUE ALL 'C'.
03 FILLER	PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID	PIC X(20).
-----------	------------

03	BENEFIT-RCD-NO	PIC 999	COMP.
03	COBRA-EVENT-ID	PIC 999	COMP.
03	DEPENDENT-BENEF	PIC XX.	
03	PLAN-TYPE	PIC XX.	
03	COBRA-EVENT-TYPE	PIC X.	
03	CBR-INIT-EVENT-ID	PIC 999	COMP.
03	CBR-SCND-EVENT-ID	PIC 999	COMP.
03	INIT-BENEFIT-PGM	PIC X(10).	
03	CBR-PROCESS-STATUS	PIC X.	
03	CBR-ENROLL-STATUS	PIC X.	
03	CBR-PLN-REPRCS-IND	PIC X.	
03	COVERAGE-BEGIN-DT	PIC X(10).	
03	DISABL-CVG-BEG-DT	PIC X(10).	
03	COVERAGE-END-DT	PIC X(10).	
03	COBRA-ELECT-END-DT	PIC X(10).	
03	COBRA-ELECT	PIC X.	
03	COBRA-ELECT-DT	PIC X(10).	
03	COBRA-WAIVE-DT	PIC X(10).	
03	COBRA-REVOKE-DT	PIC X(10).	
03	COBRA-TERM-DT	PIC X(10).	
03	COBRA-TERM-REASON	PIC X.	
03	OPTION-CD	PIC XXX.	
03	BENEFIT-PLAN	PIC X(10).	
03	COVRG-CD	PIC X(2).	
03	EMPL-CONTRBUTN-AMT	PIC S9999V99	COMP-3.
03	ANN-EX-CREDIT-FSA	PIC S9(5)V99	COMP-3.

03 ANNUAL-PLEDGE PIC S9(5)V99 COMP-3.
 03 FSA-SUB-AMT-YTD PIC S9(6)V99 COMP-3.
 03 FSA-APR-AMT-YTD PIC S9(6)V99 COMP-3.
 03 FSA-PD-AMT-YTD PIC S9(6)V99 COMP-3.
 03 COBRA-ERROR PIC X.
 03 HLTH-PROVIDER-ID PIC X(30).
 03 PREVIOUSLY-SEEN PIC X.
 03 OTH-INSURANCE-IND PIC X.
 03 OTH-INSURANCE-NAME PIC X(30).
 03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA PARTIC OPTN SQL BUFFER AND STMT *

01 S-PAROPTN.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_PAROPTN'.

02 SELECT-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
 03 FILLER PIC XX VALUE ALL 'S'.
 03 FILLER PIC XX VALUE ALL 'M'.
 03 FILLER PIC XX VALUE ALL 'C'.
 03 FILLER PIC XX VALUE ALL 'H'.
 03 FILLER PIC XX VALUE ALL 'S'.
 03 FILLER PIC X(10) VALUE ALL 'C'.
 03 FILLER PIC X(2) VALUE ALL 'H'.

03 FILLER PIC XXX VALUE ALL 'C'.
 03 FILLER PIC X(6) VALUE '2PPPPP'.
 03 FILLER PIC X(6) VALUE '2PPPPP'.
 03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID PIC X(20).
 03 BENEFIT-RCD-NO PIC 999 COMP.
 03 COBRA-EVENT-ID PIC 999 COMP.
 03 DEPENDENT-BENEF PIC XX.
 03 PLAN-TYPE PIC XX.
 03 OPTION-ID PIC 9(4) COMP.
 03 BENEFIT-PLAN PIC X(10).
 03 COVRG-CD PIC X(2).
 03 OPTION-CD PIC XXX.
 03 CBR-MM-COST PIC S9(8)V99 COMP-3.
 03 CBR-DISABL-MM-COST PIC S9(8)V99 COMP-3.
 03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA PARTIC DPND SQL BUFFER AND STMT *

01 S-PARDPND.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_PARDPND'.

02 SELECT-SETUP.

03 FILLER	PIC X(20)	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'S'.
03 FILLER	PIC XX	VALUE ALL 'M'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'H'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X(30)	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(30)	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE 'Z'.

02 SELECT-DATA.

03 EMPLID	PIC X(20).	
03 BENEFIT-RCD-NO	PIC 999	COMP.
03 COBRA-EVENT-ID	PIC 999	COMP.
03 DEPENDENT-BENEF	PIC XX.	
03 PLAN-TYPE	PIC XX.	
03 COBRA-DEP-BENEF	PIC XX.	
03 COBRA-ERROR	PIC X.	
03 HLTH-PROVIDER-ID	PIC X(30).	
03 PREVIOUSLY-SEEN	PIC X.	
03 OTH-INSURANCE-IND	PIC X.	

03 OTH-INSURANCE-NAME PIC X(30).

03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA-EVENT SQL UPDATE STMT *

01 U-EVT.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_U_EVT'.

02 BIND-SETUP.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 CBR-QUALIFY-STATUS PIC XX VALUE SPACES.

03 CBR-PROCESS-STATUS PIC X VALUE SPACES.

03 CBR-EVT-REPRCS-IND PIC X VALUE SPACES.

03 CBR-EVENT-CONFLICT PIC X VALUE SPACES.

03 EMPLID PIC X(20) VALUE SPACES.
 03 BENEFIT-RCD-NO PIC 999 COMP.
 03 COBRA-EVENT-ID PIC 999 COMP.
 03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA-EVENT RESET SQL UPDATE STMT *

01 U-RESEVT.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_U_RESEVT'.

02 BIND-SETUP.

03 FILLER PIC XX VALUE ALL 'C'.
 03 FILLER PIC X VALUE ALL 'H'.
 03 FILLER PIC X VALUE ALL 'C'.
 03 FILLER PIC X(20) VALUE ALL 'H'.
 03 FILLER PIC XX VALUE ALL 'S'.
 03 FILLER PIC XX VALUE ALL 'M'.
 03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 CBR-QUALIFY-STATUS PIC XX VALUE SPACES.
 03 CBR-PROCESS-STATUS PIC X VALUE SPACES.
 03 CBR-EVT-REPRCS-IND PIC X VALUE SPACES.

03 EMPLID PIC X(20) VALUE SPACES.
 03 BENEFIT-RCD-NO PIC 999 COMP.
 03 COBRA-EVENT-ID PIC 999 COMP.
 03 FILLER PIC X VALUE 'Z'.

/*****

* CBR_PARTIC SQL UPDATE STMT *

01 U-PAR.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_U_PAR'.

02 BIND-SETUP.

03 FILLER PIC XX VALUE ALL 'C'.
 03 FILLER PIC X VALUE ALL 'H'.
 03 FILLER PIC X VALUE ALL 'C'.
 03 FILLER PIC XX VALUE ALL 'H'.
 03 FILLER PIC XX VALUE ALL 'C'.
 03 FILLER PIC X(11) VALUE ALL 'H'.
 03 FILLER PIC X(20) VALUE ALL 'C'.
 03 FILLER PIC XX VALUE ALL 'S'.
 03 FILLER PIC XX VALUE ALL 'M'.
 03 FILLER PIC XX VALUE ALL 'C'.
 03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 CBR-QUALIFY-STATUS PIC XX VALUE SPACES.
03 CBR-PROCESS-STATUS PIC X VALUE SPACES.
03 CBR-PAR-REPRCS-IND PIC X VALUE SPACES.
03 CBR-INIT-EVENT-STS PIC XX VALUE SPACES.
03 CBR-SCND-EVENT-STS PIC XX VALUE SPACES.
03 COBRA-EMPLID PIC X(11) VALUE SPACES.
03 EMPLID PIC X(20) VALUE SPACES.
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 DEPENDENT-BENEF PIC XX VALUE SPACES.
03 FILLER PIC X VALUE 'Z'.

/*****

* CBR_PARTIC_PLAN SQL UPDATE STMT *

01 U-PARPLN.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_U_PARPLN'.

02 BIND-SETUP.

03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.

03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC XXX	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'H'.
03 FILLER	PIC X(2)	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X(20)	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'S'.
03 FILLER	PIC XX	VALUE ALL 'M'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE 'Z'.

02 BIND-DATA.

03 CBR-PROCESS-STATUS	PIC X	VALUE SPACES.
03 CBR-PLN-REPRCS-IND	PIC X	VALUE SPACES.
03 CBR-ENROLL-STATUS	PIC X	VALUE SPACES.
03 COBRA-ELECT	PIC X	VALUE SPACES.
03 COBRA-ELECT-DT	PIC X(10)	VALUE SPACES.
03 COBRA-TERM-REASON	PIC X	VALUE SPACES.
03 COBRA-TERM-DT	PIC X(10)	VALUE SPACES.
03 OPTION-CD	PIC XXX	VALUE SPACES.
03 BENEFIT-PLAN	PIC X(10)	VALUE SPACES.
03 COVRG-CD	PIC X(2)	VALUE SPACES.
03 COBRA-ERROR	PIC X	VALUE SPACES.

03 EMPLID PIC X(20) VALUE SPACES.

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 DEPENDENT-BENEF PIC XX VALUE SPACES.

03 PLAN-TYPE PIC XX VALUE SPACES.

03 FILLER PIC X VALUE 'Z'.

/*****

* CBR_PARTIC_DPND SQL UPDATE STMT *

01 U-PARDPND.

02 SQL-STMT PIC X(18) VALUE

 'PSPCOBRA_U_PARDPND'.

02 BIND-SETUP.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X(20) VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 COBRA-ERROR PIC X VALUE SPACES.
 03 EMPLID PIC X(20) VALUE SPACES.
 03 BENEFIT-RCD-NO PIC 999 COMP.
 03 COBRA-EVENT-ID PIC 999 COMP.
 03 DEPENDENT-BENEF PIC XX VALUE SPACES.
 03 PLAN-TYPE PIC XX VALUE SPACES.
 03 COBRA-DEP-BENEF PIC XX VALUE SPACES.
 03 FILLER PIC X VALUE 'Z'.

/*****

* CLEAR SCND-EVENT-ID CBR_PARTIC_PLAN SQL UPDATE STMT *

*****/

01 U-CLRSCND.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_U_CLRSCND'.

02 BIND-SETUP.

03 FILLER PIC XX VALUE ALL 'S'.
 03 FILLER PIC X(20) VALUE ALL 'C'.
 03 FILLER PIC XX VALUE ALL 'S'.
 03 FILLER PIC XX VALUE ALL 'M'.
 03 FILLER PIC XX VALUE ALL 'C'.
 03 FILLER PIC XX VALUE ALL 'H'.
 03 FILLER PIC XX VALUE ALL 'S'.
 03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 CBR-SCND-EVENT-ID PIC 999 COMP.
03 EMPLID PIC X(20) VALUE SPACES.
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 DEPENDENT-BENEF PIC XX VALUE SPACES.
03 PLAN-TYPE PIC XX VALUE SPACES.
03 CBR-SCND-EVT-ID-2 PIC 999 COMP.
03 FILLER PIC X VALUE 'Z'.

/*****

* SET SCND-EVENT-ID CBR_PARTIC_PLAN SQL UPDATE STMT *

01 U-SETSCND.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_U_SETSCND'.

02 BIND-SETUP.

03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 CBR-SCND-EVENT-ID PIC 999 COMP.

03 EMPLID PIC X(20) VALUE SPACES.

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 DEPENDENT-BENEF PIC XX VALUE SPACES.

03 PLAN-TYPE PIC XX VALUE SPACES.

03 FILLER PIC X VALUE 'Z'.

/*****

* DUPLICATE COBRA EVENT? SQL BUFFER AND STMT *

01 S-EVTDUPL.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_EVTDUPL'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC XXX VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20) VALUE SPACES.

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-DT PIC X(10) VALUE SPACES.

03 COBRA-EVENT-CLASS PIC XXX VALUE SPACES.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 FILLER-X PIC X.

03 FILLER PIC X VALUE 'Z'.

/*****

* REG-REGION-TBL SQL BUFFER AND STMT *

01 S-REGION.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_REGION'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EFFDT PIC X(10).

03 EFFDT-2 PIC X(10).

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 FILLER-X PIC X.

03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA-EVENT IN CONFLICT? SQL BUFFER AND STMT *

01 S-EVTCONF.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_EVTCONF'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-DT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 COBRA-EVENT-ID PIC 999 COMP.
03 COBRA-EVENT-DT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

/*****

* UPDATE COBRA_EVENT IN CONFLICT UPDATE STMT *

01 U-EVTCONF.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_U_EVTCONF'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20) VALUE SPACES.
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA PARTIC SQL INSERT STMT *

01 I-PAR.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_I_PAR'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER	PIC XX	VALUE ALL 'M'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'H'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'H'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(11)	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'H'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE 'Z'.

02 BIND-DATA.

03 EMPLID	PIC X(20)	VALUE SPACES.
03 BENEFIT-RCD-NO	PIC 999	COMP.
03 COBRA-EVENT-ID	PIC 999	COMP.
03 DEPENDENT-BENEF	PIC XX	VALUE SPACES.
03 BENEFIT-PROGRAM	PIC X(10)	VALUE SPACES.

03 COBRA-EVENT-DT PIC X(10) VALUE SPACES.
 03 CBR-QUALIFY-STATUS PIC XX VALUE SPACES.
 03 CBR-PROCESS-STATUS PIC X VALUE SPACES.
 03 CBR-INIT-EVENT-STS PIC XX VALUE SPACES.
 03 CBR-SCND-EVENT-STS PIC XX VALUE SPACES.
 03 CBR-INIT-NOTIFY-DT PIC X(10) VALUE SPACES.
 03 CBR-SCND-NOTIFY-DT PIC X(10) VALUE SPACES.
 03 COBRA-ELECT PIC X VALUE SPACES.
 03 COBRA-ELECT-DT PIC X(10) VALUE SPACES.
 03 COBRA-WAIVE-DT PIC X(10) VALUE SPACES.
 03 COBRA-REVOKE-DT PIC X(10) VALUE SPACES.
 03 COBRA-EMPLID PIC X(11) VALUE SPACES.
 03 BAS-ASSIGN-STATUS PIC XX VALUE SPACES.
 03 CBR-TERM-NOTIFY-DT PIC X(10) VALUE SPACES.
 03 CBR-PAR-REPRCS-IND PIC X VALUE SPACES.
 03 FILLER PIC X VALUE 'Z'.

/*****

* COBRA PARTIC PLAN SQL INSERT STMT *

01 I-PARPLN.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_I_PARPLN'.

02 BIND-SETUP.

03 FILLER	PIC X(20) VALUE ALL 'C'.
03 FILLER	PIC XX VALUE ALL 'S'.
03 FILLER	PIC XX VALUE ALL 'M'.
03 FILLER	PIC XX VALUE ALL 'C'.
03 FILLER	PIC XX VALUE ALL 'H'.
03 FILLER	PIC X(10) VALUE ALL 'C'.
03 FILLER	PIC X(10) VALUE ALL 'D'.
03 FILLER	PIC X VALUE ALL 'C'.
03 FILLER	PIC XX VALUE ALL 'S'.
03 FILLER	PIC XX VALUE ALL 'M'.
03 FILLER	PIC X(10) VALUE ALL 'C'.
03 FILLER	PIC X VALUE ALL 'H'.
03 FILLER	PIC X VALUE ALL 'C'.
03 FILLER	PIC X VALUE ALL 'H'.
03 FILLER	PIC X(10) VALUE ALL 'D'.
03 FILLER	PIC X(10) VALUE ALL 'A'.
03 FILLER	PIC X(10) VALUE ALL 'D'.
03 FILLER	PIC X(10) VALUE ALL 'A'.
03 FILLER	PIC X VALUE ALL 'C'.
03 FILLER	PIC X(10) VALUE ALL 'D'.
03 FILLER	PIC X(10) VALUE ALL 'A'.
03 FILLER	PIC X(10) VALUE ALL 'D'.
03 FILLER	PIC X(10) VALUE ALL 'A'.
03 FILLER	PIC X VALUE ALL 'C'.
03 FILLER	PIC XXX VALUE ALL 'H'.
03 FILLER	PIC X(10) VALUE ALL 'C'.

03 FILLER	PIC X(2)	VALUE ALL 'H'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(30)	VALUE ALL 'H'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE ALL 'H'.
03 FILLER	PIC X(30)	VALUE ALL 'C'.
03 FILLER	PIC X	VALUE 'Z'.

02 BIND-DATA.

03 EMPLID	PIC X(20)	VALUE SPACES.
03 BENEFIT-RCD-NO	PIC 999	COMP.
03 COBRA-EVENT-ID	PIC 999	COMP.
03 DEPENDENT-BENEF	PIC XX	VALUE SPACES.
03 PLAN-TYPE	PIC XX	VALUE SPACES.
03 BENEFIT-PROGRAM	PIC X(10)	VALUE SPACES.
03 COBRA-EVENT-DT	PIC X(10)	VALUE SPACES.
03 COBRA-EVENT-TYPE	PIC X	VALUE SPACES.
03 CBR-INIT-EVENT-ID	PIC 999	COMP.
03 CBR-SCND-EVENT-ID	PIC 999	COMP.
03 INIT-BENEFIT-PGM	PIC X(10)	VALUE SPACES.

03 CBR-PROCESS-STATUS PIC X VALUE SPACES.
 03 CBR-ENROLL-STATUS PIC X VALUE SPACES.
 03 CBR-PLN-REPRCS-IND PIC X VALUE SPACES.
 03 COVERAGE-BEGIN-DT PIC X(10) VALUE SPACES.
 03 DISABL-CVG-BEG-DT PIC X(10) VALUE SPACES.
 03 COVERAGE-END-DT PIC X(10) VALUE SPACES.
 03 COBRA-ELECT-END-DT PIC X(10) VALUE SPACES.
 03 COBRA-ELECT PIC X VALUE SPACES.
 03 COBRA-ELECT-DT PIC X(10) VALUE SPACES.
 03 COBRA-WAIVE-DT PIC X(10) VALUE SPACES.
 03 COBRA-REVOKE-DT PIC X(10) VALUE SPACES.
 03 COBRA-TERM-DT PIC X(10) VALUE SPACES.
 03 COBRA-TERM-REASON PIC X VALUE SPACES.
 88 COBRA-NOT-TERMINATED VALUE 'N'.
 03 OPTION-CD PIC XXX VALUE SPACES.
 03 BENEFIT-PLAN PIC X(10) VALUE SPACES.
 03 COVRG-CD PIC X(2) VALUE SPACES.
 03 EMPL-CONTRBUTN-AMT PIC S9999V99 COMP-3.
 03 ANN-EX-CREDIT-FSA PIC S9(5)V99 COMP-3.
 03 ANNUAL-PLEDGE PIC S9(5)V99 COMP-3.
 03 FSA-SUB-AMT-YTD PIC S9(6)V99 COMP-3.
 03 FSA-APR-AMT-YTD PIC S9(6)V99 COMP-3.
 03 FSA-PD-AMT-YTD PIC S9(6)V99 COMP-3.
 03 COBRA-ERROR PIC X VALUE SPACES.
 03 HLTH-PROVIDER-ID PIC X(30) VALUE SPACES.
 03 PREVIOUSLY-SEEN PIC X VALUE SPACES.

```

      88 PREVIOUSLY-SEEN-NO      VALUE 'N'.

03 OTH-INSURANCE-IND  PIC X      VALUE SPACES.

      88 OTH-INSURANCE-NO        VALUE 'N'.

03 OTH-INSURANCE-NAME PIC X(30)  VALUE SPACES.

03 FILLER             PIC X      VALUE 'Z'.

```

```

/*****

```

```

*          COBRA PARTIC OPTN SQL INSERT STMT          *

```

```

*****

```

```

01 I-PAROPTN.

```

```

      02 SQL-STMT             PIC X(18)  VALUE

                                   'PSPCOBRA_I_PAROPTN'.

```

```

02 BIND-SETUP.

```

```

      03 FILLER             PIC X(20)  VALUE ALL 'C'.

      03 FILLER             PIC XX      VALUE ALL 'S'.

      03 FILLER             PIC XX      VALUE ALL 'M'.

      03 FILLER             PIC XX      VALUE ALL 'C'.

      03 FILLER             PIC XX      VALUE ALL 'H'.

      03 FILLER             PIC XX      VALUE ALL 'S'.

      03 FILLER             PIC X(10)  VALUE ALL 'C'.

      03 FILLER             PIC X(10)  VALUE ALL 'D'.

      03 FILLER             PIC X(10)  VALUE ALL 'C'.

      03 FILLER             PIC X(2)   VALUE ALL 'H'.

      03 FILLER             PIC XXX     VALUE ALL 'C'.

```


03 FILLER PIC X(6) VALUE '2PPPPP'.
03 FILLER PIC X(6) VALUE '2PPPPP'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20) VALUE SPACES.
03 BENEFIT-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 DEPENDENT-BENEF PIC XX VALUE SPACES.
03 PLAN-TYPE PIC XX VALUE SPACES.
03 OPTION-ID PIC 9(4) COMP.
03 BENEFIT-PROGRAM PIC X(10) VALUE SPACES.
03 COBRA-EVENT-DT PIC X(10) VALUE SPACES.
03 BENEFIT-PLAN PIC X(10) VALUE SPACES.
03 COVRG-CD PIC X(2) VALUE SPACES.
03 OPTION-CD PIC XXX VALUE SPACES.
03 CBR-MM-COST PIC S9(8)V99 COMP-3.
03 CBR-DISABL-MM-COST PIC S9(8)V99 COMP-3.
03 FILLER PIC X VALUE 'Z'.

/*****

* PRIOR CBR_PARTIC_PLAN SQL BUFFER AND STMT *

01 S-PRRPLN.

02 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_PRRPLN'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'H'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 BENEFIT-RCD-NO PIC 999 COMP.
03 DEPENDENT-BENEF PIC XX.
03 PLAN-TYPE PIC XX.
03 COBRA-EVENT-DT PIC X(10).
03 COBRA-EVENT-DT-2 PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XXX VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 BENEFIT-PROGRAM PIC X(10).
03 COBRA-EVENT-ID PIC 999 COMP.
03 COBRA-EVENT-CLASS PIC XXX.
03 COBRA-ELECT PIC X.
 88 COBRA-ELECT-WAIVE VALUE 'W'.
03 CBR-ENROLL-STATUS PIC X.
03 COVERAGE-BEGIN-DT PIC X(10).
03 COVERAGE-END-DT PIC X(10).
03 COBRA-TERM-DT PIC X(10).
03 BENEFIT-PLAN PIC X(10).
03 CBR-PROCESS-STATUS PIC X.
 88 CBR-PROCESS-OPEN VALUE 'O'.
 88 CBR-PROCESS-CLOSED VALUE 'C'.
03 FILLER PIC X VALUE 'Z'.

/*****

* PRIOR CBR_PARTIC_DPND SQL BUFFER AND STMT *

01 S-PRRDPND.

02 SQL-CURSOR PIC 9999 VALUE ZERO COMP.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_PRRDPND'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 BENEFIT-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 PLAN-TYPE PIC XX.

03 COBRA-DEP-BENEF PIC XX.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'A'.

03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 COVERAGE-END-DT PIC X(10).
03 COBRA-TERM-DT PIC X(10).
03 BENEFIT-PLAN PIC X(10).
03 CBR-PROCESS-STATUS PIC X.
 88 CBR-PROCESS-OPEN VALUE 'O'.
 88 CBR-PROCESS-CLOSED VALUE 'C'.
03 FILLER PIC X VALUE 'Z'.

/*****

* BAS_EVENT_RULE SQL BUFFER AND STMT *

01 S-BAS-EVT.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_BAS_EVT'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC XXXX VALUE ALL 'I'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 BENEFIT-RCD-NO PIC 999 COMP.

03 SCHED-ID PIC X(10).

03 EVENT-ID PIC 9(6) COMP.

03 PLAN-TYPE PIC XX.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 COVERAGE-BEGINS-CD PIC X.

03 COVERAGE-ENDS-CD PIC X.

03 FILLER PIC X VALUE 'Z'.

/*****

* PAY_CALENDAR SQL BUFFER AND STMT *

01 S-CAL.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_CAL'.

02 BIND-SETUP.

03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'H'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 COMPANY PIC X(10).
03 PAYGROUP PIC X(10).
03 PAY-BEGIN-DT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 PAY-BEGIN-DT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

/*****

* EMPLOYEE'S PRIOR HR DATA SQL BUFFER AND STMT

*

01 S-EMPL-HR.

02 SQL-STMT PIC X(18).

88 SQL-STMT-BENADMIN VALUE
'PSPCOBRA_S_EMPLHR1'.

88 SQL-STMT-BASEBEN VALUE
'PSPCOBRA_S_EMPLHR2'.

02 BIND-SETUP PIC X(47).

02 BIND-SETUP-BENADMIN REDEFINES BIND-SETUP.

03 FILLER PIC X(20).

88 EMPLID VALUE ALL 'C'.

NOCLN 03 FILLER PIC XX.

88 BENEFIT-RCD-NO VALUE ALL 'S'.

03 FILLER PIC X(10).

88 SCHED-ID VALUE ALL 'C'.

NOCLN 03 FILLER PIC XXXX.

88 EVENT-ID VALUE ALL 'I'.

NOCLN 03 FILLER PIC X(10).

88 STDHRS-FREQ-EFFDT VALUE ALL 'D'.

NOCLN 03 FILLER PIC X.

88 DELIMCHAR VALUE 'Z'.

02 BIND-SETUP-BASEBEN REDEFINES BIND-SETUP.

03 FILLER PIC X(20).

88 EMPLID VALUE ALL 'C'.

NOCLN 03 FILLER PIC XX.

	88	EMPL-RCD-NO	VALUE ALL 'S'.
NOCLN	03	FILLER	PIC X(10).
	88	EFFDT	VALUE ALL 'D'.
NOCLN	03	FILLER	PIC X(10).
	88	STDHRS-FREQ-EFFDT	VALUE ALL 'A'.
NOCLN	03	FILLER	PIC X.
	88	DELIMCHAR	VALUE 'Z'.

02 BIND-DATA PIC X(47).

02 BIND-DATA-BENADMIN REDEFINES BIND-DATA.

03	EMPLID	PIC X(20).	
03	BENEFIT-RCD-NO	PIC 999	COMP.
03	SCHED-ID	PIC X(10).	
03	EVENT-ID	PIC 9(6)	COMP.
03	STDHRS-FREQ-EFFDT	PIC X(10).	
03	FILLER	PIC X.	
	88	DELIMCHAR	VALUE 'Z'.

02 BIND-DATA-BASEBEN REDEFINES BIND-DATA.

03	EMPLID	PIC X(20).	
03	EMPL-RCD-NO	PIC 999	COMP.
03	EFFDT	PIC X(10).	
03	STDHRS-FREQ-EFFDT	PIC X(10).	
03	FILLER	PIC X.	
	88	DELIMCHAR	VALUE 'Z'.

03 FILLER	PIC X	VALUE ALL 'H'.	
03 FILLER	PIC XXXX	VALUE '2PPP'.	
03 FILLER	PIC X	VALUE ALL 'H'.	
03 FILLER	PIC X(5)	VALUE ALL 'C'.	
03 FILLER	PIC XXXX	VALUE ALL 'H'.	FEDMERG
03 FILLER	PIC XXX	VALUE ALL 'C'.	
03 FILLER	PIC X(10)	VALUE ALL 'H'.	
03 FILLER	PIC X(5)	VALUE ALL 'C'.	
03 FILLER	PIC XXX	VALUE ALL 'H'.	
03 FILLER	PIC X(5)	VALUE ALL 'C'.	
03 FILLER	PIC X(5)	VALUE ALL 'H'.	
03 FILLER	PIC XX	VALUE ALL 'C'.	
03 FILLER	PIC XX	VALUE ALL 'H'.	
03 FILLER	PIC X(10)	VALUE ALL 'D'.	
03 FILLER	PIC X(10)	VALUE ALL 'A'.	
03 FILLER	PIC X(10)	VALUE ALL 'C'.	
03 FILLER	PIC X(10)	VALUE ALL 'D'.	FED0006
* 03 FILLER	PIC XXX	VALUE ALL 'H'.	FED0005
* 03 FILLER	PIC X	VALUE ALL 'C'.	FED0005
03 FILLER	PIC X(5)	VALUE ALL 'H'.	
03 FILLER	PIC X(6)	VALUE '7PPPPP'.	
03 FILLER	PIC XX	VALUE ALL 'S'.	
03 FILLER	PIC X(10)	VALUE ALL 'D'.	
03 FILLER	PIC XX	VALUE ALL 'S'.	
03 FILLER	PIC X	VALUE 'Z'.	

02 SELECT-DATA.

03 DEPTID	PIC X(10).
03 JOBCODE	PIC X(20).
03 EMPL-STATUS	PIC X.
03 ACTION	PIC XXX.
03 ACTION-DT	PIC X(10).
03 ACTION-REASON	PIC XXX.
03 SETID-LOCATION	PIC X(5).
03 LOCATION	PIC X(20).
03 TAX-LOCATION-CD	PIC X(10).
03 REG-TEMP	PIC X.
03 FULL-PART-TIME	PIC X.
03 OFFICER-CD	PIC X.
03 COMPANY	PIC X(10).
03 PAYGROUP	PIC X(10).
03 BAS-GROUP-ID	PIC X(10).
03 ELIG-CONFIG1	PIC X(10).
03 ELIG-CONFIG2	PIC X(10).
03 ELIG-CONFIG3	PIC X(10).
03 ELIG-CONFIG4	PIC X(10).
03 ELIG-CONFIG5	PIC X(10).
03 ELIG-CONFIG6	PIC X(10).
03 ELIG-CONFIG7	PIC X(10).
03 ELIG-CONFIG8	PIC X(10).
03 ELIG-CONFIG9	PIC X(10).
03 BEN-STATUS	PIC XXXX.

03	EMPL-TYPE	PIC X.		
03	STD-HOURS	PIC S9999V99	COMP-3.	
03	FLSA-STATUS	PIC X.		
03	SETID-SALARY	PIC X(5).		
03	SAL-ADMIN-PLAN	PIC XXXX.	FEDMERG	
03	GRADE	PIC XXX.		
03	REG-REGION	PIC X(10).		
03	BUSINESS-UNIT	PIC X(5).		
03	EMPL-CLASS	PIC XXX.		
03	SETID-DEPT	PIC X(5).		
03	SETID-JOBCODE	PIC X(5).		
03	PAY-SYSTEM-FLG	PIC XX.		
03	BENEFIT-SYSTEM	PIC XX.		
03	SERVICE-DT	PIC X(10).		
03	TERMINATION-DT	PIC X(10).		
03	UNION-CD	PIC X(10).		
03	GVT-HIRE-DATE	PIC X(10).	FED0006	
*	03	GVT-ELIG-FEHB	PIC XXX.	FED0005
*	03	GVT-RETIRE-PLAN	PIC XX.	FED0005
03	STD-HRS-FREQUENCY	PIC X(5).		
03	STDHRS-FREQ-FACTOR	PIC 9(4)V9(7)	COMP-3.	
03	JOB-EMPL-RCD	PIC 999	COMP.	
03	JOB-EFFDT	PIC X(10).		
03	JOB-EFFSEQ	PIC 999	COMP.	
03	FILLER	PIC X	VALUE 'Z'.	

/*****

* ELIGIBILITY DATA SQL BUFFER AND STMT *

*****/

01 S-ELGDATA.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_ELGDATA'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EFFDT PIC X(10).
03 EFFDT-2 PIC X(10).
03 EFFDT-3 PIC X(10).
03 EFFDT-4 PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER	PIC XX	VALUE ALL 'S'.	
03 FILLER	PIC XX	VALUE ALL 'M'.	
03 FILLER	PIC X(10)	VALUE ALL 'D'.	
03 FILLER	PIC XX	VALUE ALL 'S'.	
03 FILLER	PIC X	VALUE ALL 'C'.	
03 FILLER	PIC XXXX	VALUE ALL 'H'.	
03 FILLER	PIC X	VALUE ALL 'C'.	
03 FILLER	PIC XXX	VALUE ALL 'H'.	
03 FILLER	PIC X	VALUE ALL 'C'.	
03 FILLER	PIC X	VALUE ALL 'H'.	
03 FILLER	PIC X	VALUE ALL 'C'.	
03 FILLER	PIC X(10)	VALUE ALL 'H'.	
03 FILLER	PIC X(10)	VALUE ALL 'C'.	
03 FILLER	PIC X(10)	VALUE ALL 'H'.	
03 FILLER	PIC X(5)	VALUE ALL 'C'.	
03 FILLER	PIC X(20)	VALUE ALL 'H'.	
03 FILLER	PIC XXXX	VALUE '2PPP'.	
03 FILLER	PIC X	VALUE ALL 'C'.	
03 FILLER	PIC X(5)	VALUE ALL 'H'.	
03 FILLER	PIC XXXX	VALUE ALL 'C'.	FEDMERG
03 FILLER	PIC XXX	VALUE ALL 'H'.	
03 FILLER	PIC X(10)	VALUE ALL 'C'.	
03 FILLER	PIC X(5)	VALUE ALL 'H'.	
03 FILLER	PIC X(5)	VALUE ALL 'C'.	
03 FILLER	PIC X(10)	VALUE '6PPPPPPPPP'.	
03 FILLER	PIC X(10)	VALUE '3PPPPPPPPP'.	

03 FILLER	PIC X(10)	VALUE '6PPPPPPPPP'.	
03 FILLER	PIC X(10)	VALUE '3PPPPPPPPP'.	
03 FILLER	PIC XXX	VALUE ALL 'H'.	
03 FILLER	PIC X(10)	VALUE ALL 'C'.	
03 FILLER	PIC X(10)	VALUE ALL 'H'.	
03 FILLER	PIC X(10)	VALUE ALL 'C'.	
03 FILLER	PIC X(10)	VALUE ALL 'H'.	
03 FILLER	PIC X(10)	VALUE ALL 'C'.	
03 FILLER	PIC X(10)	VALUE ALL 'H'.	
03 FILLER	PIC X(10)	VALUE ALL 'C'.	
03 FILLER	PIC X(10)	VALUE ALL 'H'.	
03 FILLER	PIC X(10)	VALUE ALL 'C'.	
03 FILLER	PIC X(4)	VALUE '6PPP'.	
03 FILLER	PIC XXX	VALUE ALL 'H'.	
* 03 FILLER	PIC XXX	VALUE ALL 'C'.	FED0005
* 03 FILLER	PIC X	VALUE ALL 'H'.	FED0005
* 03 FILLER	PIC X(2)	VALUE ALL 'C'.	FED0021
* 03 FILLER	PIC X(4)	VALUE '6PPP'.	
* 03 FILLER	PIC X(4)	VALUE '6PPP'.	
* 03 FILLER	PIC X(4)	VALUE '6PPP'.	
03 FILLER	PIC X(10)	VALUE ALL 'D'.	
03 FILLER	PIC X(10)	VALUE ALL 'A'.	
03 FILLER	PIC X(10)	VALUE ALL 'D'.	FED0006
03 FILLER	PIC XX	VALUE ALL 'C'.	
03 FILLER	PIC X	VALUE ALL 'H'.	
03 FILLER	PIC X	VALUE ALL 'C'.	

03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X(6) VALUE '7PPPPP'.
03 FILLER PIC X(5) VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X(6) VALUE '7PPPPP'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 BENEFIT-RCD-NO PIC 999 COMP.
03 EMPL-RCD-NO PIC 999 COMP.
03 EFFDT PIC X(10).
03 EFFSEQ PIC 999 COMP.
03 EMPL-STATUS PIC X.
 88 EMPL-STATUS-TERMINATED VALUE 'T'.
03 BEN-STATUS PIC XXXX.
03 EMPL-TYPE PIC X.
03 EMPL-CLASS PIC XXX.
03 REG-TEMP PIC X.
03 FULL-PART-TIME PIC X.
03 OFFICER-CD PIC X.
03 COMPANY PIC X(10).
03 BAS-GROUP-ID PIC X(10).
03 PAYGROUP PIC X(10).
03 SETID-LOCATION PIC X(5).
03 LOCATION PIC X(20).

03	STD-HOURS	PIC S9999V99	COMP-3.
03	FLSA-STATUS	PIC X.	
03	SETID-SALARY	PIC X(5).	
03	SAL-ADMIN-PLAN	PIC XXXX.	FEDMERG
03	GRADE	PIC XXX.	
03	REG-REGION	PIC X(10).	
03	BUSINESS-UNIT	PIC X(5).	
03	COMP-FREQUENCY	PIC X(5).	
03	COMPRATE	PIC S9(12)V9(6)	COMP-3.
03	ANNUAL-RT	PIC S9(15)V9(3)	COMP-3.
03	HOURLY-RT	PIC S9(12)V9(6)	COMP-3.
03	ANNL-BENEF-BASE-RT	PIC S9(15)V9(3)	COMP-3.
03	UNION-CD	PIC XXX.	
03	ELIG-CONFIG1	PIC X(10).	
03	ELIG-CONFIG2	PIC X(10).	
03	ELIG-CONFIG3	PIC X(10).	
03	ELIG-CONFIG4	PIC X(10).	
03	ELIG-CONFIG5	PIC X(10).	
03	ELIG-CONFIG6	PIC X(10).	
03	ELIG-CONFIG7	PIC X(10).	
03	ELIG-CONFIG8	PIC X(10).	
03	ELIG-CONFIG9	PIC X(10).	
03	FTE	PIC 9V9(6)	COMP-3.
03	COBRA-ACTION	PIC XXX.	
*	03 GVT-ELIG-FEHB	PIC XXX.	FED0005
*	03 GVT-RETIRE-PLAN	PIC XX.	FED0005

* 03 GVT-PAY-PLAN PIC X(2). FED0021
 * 03 GVT-FEGLI-OPT-PCT PIC S9V9(6) COMP-3.
 * 03 GVT-FEGLI-BASC-PCT PIC S9V9(6) COMP-3.
 * 03 GVT-FEHB-PCT PIC S9V9(6) COMP-3.
 03 SERVICE-DT PIC X(10).
 03 TERMINATION-DT PIC X(10).
 03 GVT-HIRE-DATE PIC X(10). FED0006
 03 BENEFIT-SYSTEM PIC XX.
 88 BENEFIT-SYSTEM-BENADMIN VALUE 'BA'.
 88 BENEFIT-SYSTEM-BASE VALUE 'BN'.
 03 PRIMARY-JOB-IND PIC X.
 88 PRIMARY-JOB-YES VALUE 'Y'.
 88 PRIMARY-JOB-NO VALUE 'N'.
 03 PRIMARY-FLAG1 PIC X.
 88 INCLUDE-FOR-ELIG-YES VALUE 'Y'.
 88 INCLUDE-FOR-ELIG-NO VALUE 'N'.
 03 PRIMARY-FLAG2 PIC X.
 88 INCLUDE-FOR-DED-YES VALUE 'Y'.
 88 INCLUDE-FOR-DED-NO VALUE 'N'.
 03 COMP-FREQ-TYPE PIC X.
 03 COMP-FREQ-FACTOR PIC 9(4)V9(7) COMP-3.
 03 STD-HRS-FREQUENCY PIC X(5).
 03 STD-HRS-FREQ-TYPE PIC X.
 03 STDHRS-FREQ-FACTOR PIC 9(4)V9(7) COMP-3.
 03 FILLER PIC X VALUE 'Z'.

/*****

* BEN_PROG_PARTIC SQL INSERT STMT *

01 I-BP-PGM.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_I_BP_PGM'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X(10) VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20) VALUE SPACES.

03 EMPL-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 EFFDT PIC X(10) VALUE SPACES.

03 BENEFIT-PROGRAM PIC X(10) VALUE SPACES.

03 FILLER PIC X VALUE 'Z'.

/*****

* BENEFIT_PARTIC SQL INSERT STMT *

01 I-BP-PLN.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_I_BP_PLN'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20) VALUE SPACES.

03 EMPL-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 PLAN-TYPE PIC XX VALUE SPACES.

03 BENEFIT-NO PIC 999 COMP.

03 FILLER PIC X VALUE 'Z'.

/*****

* HEALTH_BENEFIT SQL INSERT STMT *

01 I-HTH.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_I_HTH'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'A'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC X(2) VALUE ALL 'H'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC XXXX VALUE '2PPP'.
03 FILLER PIC X(30) VALUE ALL 'C'.
03 FILLER PIC X VALUE ALL 'H'.
03 FILLER PIC X VALUE ALL 'C'.
03 FILLER PIC X(30) VALUE ALL 'H'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20) VALUE SPACES.

03 EMPL-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 PLAN-TYPE PIC XX VALUE SPACES.

03 BENEFIT-NO PIC 999 COMP.

03 EFFDT PIC X(10) VALUE SPACES.

03 DEDUCTION-END-DT PIC X(10) VALUE SPACES.

03 COVERAGE-BEGIN-DT PIC X(10) VALUE SPACES.

03 COVERAGE-END-DT PIC X(10) VALUE SPACES.

03 COVERAGE-ELECT PIC X VALUE SPACES.

88 COVERAGE-ELECT-ELECT VALUE 'E'.

88 COVERAGE-ELECT-TERM VALUE 'T'.

03 COVERAGE-ELECT-DT PIC X(10) VALUE SPACES.

03 BENEFIT-PLAN PIC X(10) VALUE SPACES.

03 COVRG-CD PIC X(2) VALUE SPACES.

03 HIPAA-REPORT-DT PIC X(10) VALUE SPACES.

03 PRETAX-CD PIC X VALUE SPACES.

88 PRETAX-CD-YES VALUE 'Y'.

03 MED-PREM-PRETAX PIC S9999V99 COMP-3.

03 HLTH-PROVIDER-ID PIC X(30) VALUE SPACES.

03 PREVIOUSLY-SEEN PIC X VALUE SPACES.

88 PREVIOUSLY-SEEN-NO VALUE 'N'.

03 OTH-INSURANCE-IND PIC X VALUE SPACES.

88 OTH-INSURANCE-NO VALUE 'N'.

03 OTH-INSURANCE-NAME PIC X(30) VALUE SPACES.

03 FILLER PIC X VALUE 'Z'.

/*****

* HEALTH_DEPENDNT SQL INSERT STMT *

01 I-HTH-DEP.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_I-HTH-DEP'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X(30) VALUE ALL 'H'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE ALL 'H'.

03 FILLER PIC X(30) VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20) VALUE SPACES.
 03 EMPL-RCD-NO PIC 999 COMP.
 03 COBRA-EVENT-ID PIC 999 COMP.
 03 PLAN-TYPE PIC XX VALUE SPACES.
 03 BENEFIT-NO PIC 999 COMP.
 03 EFFDT PIC X(10) VALUE SPACES.
 03 DEPENDENT-BENEF PIC XX VALUE SPACES.
 03 HLTH-PROVIDER-ID PIC X(30) VALUE SPACES.
 03 PREVIOUSLY-SEEN PIC X VALUE SPACES.
 88 PREVIOUSLY-SEEN-NO VALUE 'N'.
 03 OTH-INSURANCE-IND PIC X VALUE SPACES.
 88 OTH-INSURANCE-NO VALUE 'N'.
 03 OTH-INSURANCE-NAME PIC X(30) VALUE SPACES.
 03 FILLER PIC X VALUE 'Z'.

/*****

* FSA_BENEFIT SQL INSERT STMT *

01 I-FSA.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_I_FSA'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
 03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER	PIC XX	VALUE ALL 'M'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC XX	VALUE ALL 'S'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'A'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(10)	VALUE ALL 'D'.
03 FILLER	PIC X(10)	VALUE ALL 'C'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC XXXX	VALUE '2PPP'.
03 FILLER	PIC X	VALUE ALL 'C'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC XX	VALUE ALL 'C'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X(5)	VALUE '2PPPP'.
03 FILLER	PIC X	VALUE 'Z'.

02 BIND-DATA.

03 EMPLID	PIC X(20)	VALUE SPACES.
-----------	-----------	---------------

03 EMPL-RCD-NO PIC 999 COMP.
 03 COBRA-EVENT-ID PIC 999 COMP.
 03 PLAN-TYPE PIC XX VALUE SPACES.
 03 BENEFIT-NO PIC 999 COMP.
 03 EFFDT PIC X(10) VALUE SPACES.
 03 DEDUCTION-END-DT PIC X(10) VALUE SPACES.
 03 COVERAGE-BEGIN-DT PIC X(10) VALUE SPACES.
 03 COVERAGE-END-DT PIC X(10) VALUE SPACES.
 03 COVERAGE-ELECT PIC X VALUE SPACES.
 88 COVERAGE-ELECT-ELECT VALUE 'E'.
 88 COVERAGE-ELECT-TERM VALUE 'T'.
 03 COVERAGE-ELECT-DT PIC X(10) VALUE SPACES.
 03 BENEFIT-PLAN PIC X(10) VALUE SPACES.
 03 EMPL-CONTRIBUTN-AMT PIC S9999V99 COMP-3.
 03 ANN-EX-CREDIT-FSA PIC S9(5)V99 COMP-3.
 03 ANNUAL-PLEDGE PIC S9(5)V99 COMP-3.
 03 FSA-ACCT-STATUS PIC X VALUE SPACES.
 88 FSA-ACCT-STATUS-ACTIVE VALUE 'A'.
 88 FSA-ACCT-STATUS-TERMINATED VALUE 'T'.
 03 FSA-SUB-AMT-YTD PIC S9(6)V99 COMP-3.
 03 FSA-APR-AMT-YTD PIC S9(6)V99 COMP-3.
 03 FSA-PD-AMT-YTD PIC S9(6)V99 COMP-3.
 03 CARRYFORWARD-EE PIC XX VALUE SPACES.
 88 CARRYFORWARD-CLAIMS VALUE 'CL'.
 03 CARRYFWD-AMT-CLM PIC S9(6)V99 COMP-3.
 03 CARRYFWD-AMT-CRD PIC S9(6)V99 COMP-3.

03 CARRYFWD-CLM-SPENT PIC S9(6)V99 COMP-3.
03 CARRYFWD-CRD-SPENT PIC S9(6)V99 COMP-3.
03 FILLER PIC X VALUE 'Z'.

/*****

* EXISTING HEALTH_BENEFIT ENROLLMENT SQL BUFFER AND STMT *

01 S-XHTH.

02 SQL-STMT PIC X(18) VALUE
'PSPCOBRA_S_XHTH'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EMPL-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 PLAN-TYPE PIC XX.
03 EFFDT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC X(2) VALUE ALL 'H'.
03 FILLER PIC X(10) VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 BENEFIT-NO PIC 999 COMP.
03 BENEFIT-PLAN PIC X(10).
03 COVRG-CD PIC X(2).
03 BENEFIT-PROGRAM PIC X(10).
03 FILLER PIC X VALUE 'Z'.

/*****

* EXISTING HEALTH_DEPENDENT SQL BUFFER AND STMT *

01 S-XHTH-DEP.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_S_XH_DEP'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EMPL-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 PLAN-TYPE PIC XX.
03 BENEFIT-NO PIC 999 VALUE ZERO COMP.
03 EFFDT PIC X(10).
03 FILLER PIC X VALUE 'Z'.

02 SELECT-SETUP.

03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X VALUE 'Z'.

02 SELECT-DATA.

03 DEPENDENT-BENEF PIC XX.
03 FILLER PIC X VALUE 'Z'.

/*****

* UPDATE EXISTING HEALTH_BENEFIT ENROLLMENT SQL BUFFER AND STMT *

*****/

01 U-XHTH.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_U_XHTH'.

02 BIND-SETUP.

03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X(20) VALUE ALL 'H'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'M'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 COVRG-CD PIC XX VALUE SPACES.
03 EMPLID PIC X(20) VALUE SPACES.
03 EMPL-RCD-NO PIC 999 COMP.
03 COBRA-EVENT-ID PIC 999 COMP.
03 PLAN-TYPE PIC XX VALUE SPACES.
03 EFFDT PIC X(10) VALUE SPACES.
03 FILLER PIC X VALUE 'Z'.

/*****

* DELETE EXISTING HEALTH_BENEFIT ENROLLMENT SQL BUFFER AND STMT *

01 D-XDEP.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_D_XDEP'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'M'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.

03 COBRA-EVENT-ID PIC 999 COMP.

03 PLAN-TYPE PIC XX.

03 EFFDT PIC X(10).

03 DEPENDENT-BENEF PIC XX.

03 FILLER PIC X VALUE 'Z'.

/*****

01 I-NONEE.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_I_NONEE'.

02 BIND-SETUP.

03 FILLER PIC X(30) VALUE ALL 'C'.

03 FILLER PIC X(30) VALUE ALL 'H'.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X VALUE ALL 'C'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 OPRID PIC X(30) VALUE SPACES.

03 BATCH-RUN-ID PIC X(30) VALUE SPACES.

03 EMPLID PIC X(20) VALUE SPACES.

03 DEPENDENT-BENEF PIC XX VALUE SPACES.

03 JOB-EMPL-RCD PIC 999 COMP.

03 JOB-EFFDT PIC X(10) VALUE SPACES.

03 JOB-EFFSEQ PIC 999 COMP.
03 COBRA-EMPLID PIC X(20) VALUE SPACES.
03 COBRA-EVENT-DT PIC X(10) VALUE SPACES.
03 CBR-NONEE-CREATE PIC X VALUE SPACES.
03 FILLER PIC X VALUE 'Z'.

/*****

* HEALTH_BENEFIT SQL DELETE STMT *

01 D-VHTH.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_D_VHTH'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.
03 FILLER PIC XX VALUE ALL 'S'.
03 FILLER PIC XX VALUE ALL 'C'.
03 FILLER PIC X(10) VALUE ALL 'D'.
03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).
03 EMPL-RCD-NO PIC 999 COMP.
03 PLAN-TYPE PIC XX.
03 EFFDT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* HEALTH_DEPENDNT SQL DELETE STMT *

01 D-VHTH-DP.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_D_VHTH_DP'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.

03 PLAN-TYPE PIC XX.

03 EFFDT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* FSA_BENEFIT SQL DELETE STMT *

01 D-VFSA.

02 SQL-STMT PIC X(18) VALUE
 'PSPCOBRA_D_VFSA'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'S'.

03 FILLER PIC XX VALUE ALL 'C'.

03 FILLER PIC X(10) VALUE ALL 'D'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20).

03 EMPL-RCD-NO PIC 999 COMP.

03 PLAN-TYPE PIC XX.

03 EFFDT PIC X(10).

03 FILLER PIC X VALUE 'Z'.

/*****

* DEPENDENT_BENEF SQL UPDATE STMT *

01 U-VDPBEN.

02 SQL-STMT PIC X(18) VALUE

'PSPCOBRA_U_VDPBEN'.

02 BIND-SETUP.

03 FILLER PIC X(20) VALUE ALL 'C'.

03 FILLER PIC XX VALUE ALL 'H'.

03 FILLER PIC X VALUE 'Z'.

02 BIND-DATA.

03 EMPLID PIC X(20) VALUE SPACES.

03 DEPENDENT-BENEF PIC XX VALUE SPACES.

03 FILLER PIC X VALUE 'Z'.

/*****

* ADDITIONAL LIBRARY FUNCTION PARAMETERS *

01 LIBWK. COPY PTCLIBWK.

/*****

* SQL COMMUNICATION *

01 SQLRT. COPY PTCSQLRT.

/*****

* PROCESS SCHEDULER REQUEST STATUS INTERFACE *

01 USTAT. COPY PTCUSTAT.

/*****

* DATE WORK LINKAGE *

01 DTWRK. COPY PTCDTWRK.

/*****

* PAY MESSAGE INTERFACE *

01 PYMSG. COPY PSCPYMSG.

/*****

* BENADMIN INTERFACE PROCESSING DATA *

01 ADMIN. COPY PSCADMIN.

/*****

* BENADMIN TABLE ACCESS INTERFACE *

01 BATBL. COPY PSCBATBL.

/*****

* BENADMIN BENEFIT PROGRAM DEFINITION *

01 PDEFN. COPY PSCPDEFN.

/*****

* BENADMIN OPTION DEFINITION ARRAY *

01 ODEFN. COPY PSCODEFN.

/*****

* BENADMIN DEDUCTION CALCULATION INTERFACE *

01 BADED. COPY PSCBADED.

/*****

* BENADMIN INTERFACE PARTICIPANT DATA *

01 PARTC. COPY PSCPARTC.

/*****

* BENADMIN PAY CALENDAR ARRAY *

01 CALTB. COPY PSCCALTB.

/*****

* BILLING CALENDARS DATA *

01 BCLTB. COPY PSCBCLTB.

/*****

* BENADMIN INTERFACE PARTICIPANT DATA - OPTIONS *

01 POPTN. COPY PSCPOPTN.

/*****

* BENADMIN CURRENT ELECTION DATA *

01 CRELT. COPY PSCCRELT.

/*****

* BENADMIN ELIGIBILITY HISTORY DATA *

01 EHIST. COPY PSCEHIST.

/*****

* BILLING INTERFACE *

01 BIINT. COPY PSCBIINT.

/*****

* GEOGRAPHIC ELIGIBILITY *

01 GEOTB. COPY PSCGEOTB.

/*****

* *

PROCEDURE DIVISION.

* *

* *

AA000-MAIN SECTION.

AA000.

* *

COPY PTCLIBFX.

COPY PSCBNVER.

PERFORM BA000-INIT

IF COBRA-PHASE-READY OF W-CNTL

PERFORM WV000-CLEAR-MESSAGES

PERFORM DA000-RESET-FOR-REPROCESS

END-IF

IF COBRA-PHASE-READY OF W-CNTL

OR COBRA-PHASE-OVERAGE OF W-CNTL

PERFORM FA000-OVERAGE-DEPENDENT

END-IF

IF COBRA-PHASE-ACTIVITY OF W-CNTL

PERFORM GA000-PROCESS-ACTIVITY

END-IF

IF COBRA-PHASE-QUALIFY OF W-CNTL

PERFORM HA000-QUALIFY-EVENT

END-IF

IF COBRA-PHASE-PARTIC OF W-CNTL

PERFORM LA000-PROCESS-PARTICIPANT

END-IF

PERFORM CA000-TERM

COPY PSCRTNCD.

.

MAIN-EXIT.

STOP RUN.

/*****

* *

BA000-INIT SECTION.

BA000.

* *

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'COBRA Administration Started at '

TIME-OUT OF W-WK '.'

PERFORM BB000-CONNECT-TO-SQLRT

IF PROCESS-INSTANCE OF SQLRT NOT = ZERO

PERFORM BD000-SET-RUN-STAT-PROCESSING

END-IF

PERFORM BF000-GET-INSTALLATION-DATA

PERFORM BG000-GET-OPTIONS-DATA

PERFORM BH000-GET-RUNCTL-DATA

PERFORM BH100-GET-INDUSTRY-DATA MERGE

PERFORM BJ000-DISPLAY-RUN-CONTROL

PERFORM WZ000-COMMIT

* PERFORM WX000-DISCONNECT-COMMON-CURSOR

.

INIT-EXIT.

/*****

* *

BB000-CONNECT-TO-SQLRT SECTION.

BB000.

* *

CALL 'PTPSQLRT' USING ACTION-CONNECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

MOVE 'CONNECT-TO-SQLRT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

CONNECT-TO-SQLRT-EXIT.

/*****

* *

BD000-SET-RUN-STAT-PROCESSING SECTION.

BD000.

* *

INITIALIZE USTAT

MOVE PROCESS-INSTANCE OF SQLRT TO PROCESS-INSTANCE OF USTAT

SET RUN-STATUS-PROCESSING OF USTAT TO TRUE

CALL 'PTPUSTAT' USING SQLRT

USTAT

IF RTNCD-ERROR OF SQLRT

MOVE 'SET-RUN-STAT-PROCESSING(PTPUSTAT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SET-RUN-STAT-PROCESSING-EXIT.

/*****

*

*

BF000-GET-INSTALLATION-DATA SECTION.

BF000.

*

*

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-INSTALL

BIND-SETUP OF S-INSTALL

BIND-DATA OF S-INSTALL

SELECT-SETUP OF S-INSTALL

SELECT-DATA OF S-INSTALL

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-INSTALLATION-DATA(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-INSTALL

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-INSTALLATION-DATA(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

MOVE BENEFIT-ADMINISTRN OF S-INSTALL

TO BENEFIT-ADMINISTRN OF W-CNTL

MOVE BEN-BILLING OF S-INSTALL

TO BEN-BILLING OF W-CNTL

MOVE COBRA-EMPLID-LAST OF S-INSTALL

TO COBRA-EMPLID-LAST OF W-CNTL
SET BAS-TYPE-COBRA OF ADMIN TO TRUE
SET REPROCESS-NONE OF ADMIN TO TRUE
SET DEBUG-ELIG-NO OF ADMIN TO TRUE

.

GET-INSTALLATION-DATA-EXIT.

/*****

* *

BG000-GET-OPTIONS-DATA SECTION.

BG000.

* *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT
SQLRT
SQL-CURSOR-COMMON OF SQLRT
SQL-STMT OF S-OPTIONS
BIND-SETUP OF S-OPTIONS
BIND-DATA OF S-OPTIONS
SELECT-SETUP OF S-OPTIONS
SELECT-DATA OF S-OPTIONS
IF RTNCD-ERROR OF SQLRT

MOVE 'GET-OPTIONS-DATA(SELECT)' TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

INITIALIZE SELECT-DATA OF S-OPTIONS

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT
SQLRT
SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-OPTIONS-DATA(FETCH)' TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

MOVE MULTIJOB OF S-OPTIONS TO MULTIJOB OF W-CNTL

.

GET-OPTIONS-DATA-EXIT.

/*****

* *

BH000-GET-RUNCTL-DATA SECTION.

BH000.

* *

MOVE OPRID OF SQLRT TO OPRID OF S-RUNCTL

MOVE BATCH-RUN-ID OF SQLRT TO BATCH-RUN-ID OF S-RUNCTL

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-RUNCTL

BIND-SETUP OF S-RUNCTL

BIND-DATA OF S-RUNCTL

SELECT-SETUP OF S-RUNCTL

SELECT-DATA OF S-RUNCTL

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-RUNCTL-DATA(SELECT)' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-RUNCTL

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

DISPLAY 'COBRA Run Control Missing.'

DISPLAY ' for Operator ID ' OPRID OF S-RUNCTL

DISPLAY ' and Batch Run ID '

BATCH-RUN-ID OF S-RUNCTL

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

ELSE

MOVE 'GET-RUNCTL-DATA(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

MOVE PROCESS-DT OF S-RUNCTL TO PROCESS-DT OF W-CNTL

MOVE CHKPT-INTERVAL OF S-RUNCTL TO CHKPT-INTERVAL OF W-CNTL

MOVE COBRA-PHASE OF S-RUNCTL TO COBRA-PHASE OF W-CNTL

MOVE OVERAGE-PROCESS OF S-RUNCTL

TO OVERAGE-PROCESS OF W-CNTL

MOVE BENEFIT-PROGRAM OF S-RUNCTL

TO BENEFIT-PROGRAM OF W-CNTL

MOVE EMPLID OF S-RUNCTL TO EMPLID OF W-CNTL

MOVE EMPL-RCD-NO OF S-RUNCTL TO EMPL-RCD-NO OF W-CNTL

MOVE BENEFIT-RCD-NO OF S-RUNCTL TO BENEFIT-RCD-NO OF W-CNTL

MOVE COBRA-EVENT-DT OF S-RUNCTL TO COBRA-EVENT-DT OF W-CNTL

MOVE COBRA-EVENT-ID OF S-RUNCTL TO COBRA-EVENT-ID OF W-CNTL

MOVE DEPENDENT-BENEF OF S-RUNCTL

TO DEPENDENT-BENEF OF W-CNTL

MOVE COBRA-ACTION OF S-RUNCTL TO COBRA-ACTION OF W-CNTL

.

GET-RUNCTL-DATA-EXIT.

/*****

* *

BH100-GET-INDUSTRY-DATA SECTION.

BH000.

* *

IF COMPANY OF S-INSTALL NOT = SPACE MERGE

MERGE

MOVE COMPANY OF S-INSTALL TO COMPANY OF S-INDSTRY MERGE

MOVE PROCESS-DT OF S-RUNCTL MERGE

TO EFFDT OF BIND-DATA OF S-INDSTRY MERGE

MERGE

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT MERGE

SQLRT MERGE

SQL-CURSOR-COMMON OF SQLRT MERGE

SQL-STMT OF S-INDSTRY MERGE

BIND-SETUP OF S-INDSTRY MERGE

BIND-DATA OF S-INDSTRY	MERGE
SELECT-SETUP OF S-INDSTRY	MERGE
SELECT-DATA OF S-INDSTRY	MERGE
MERGE	
IF RTNCD-ERROR OF SQLRT	MERGE
MERGE	
MOVE 'SELECT-INDUSTRY-CODE(SELECT)'	MERGE
TO ERR-SECTION OF SQLRT	MERGE
PERFORM ZZ000-SQL-ERROR	MERGE
END-IF	MERGE
MERGE	
INITIALIZE SELECT-DATA OF S-INDSTRY	MERGE
MERGE	
CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT	MERGE
SQLRT	MERGE
SQL-CURSOR-COMMON OF SQLRT	MERGE
IF RTNCD-ERROR OF SQLRT	MERGE
MERGE	
IF RTNCD-END OF SQLRT	MERGE
MERGE	
SET CORE-INDUSTRY OF INDUSTRY OF ADMIN TO TRUE	MERGE
SET CORE-SECTOR OF INDUSTRY-SECTOR OF ADMIN	MERGE
TO TRUE	MERGE
ELSE	MERGE
MERGE	
MOVE 'SELECT-INDUSTRY-CODE(FETCH)'	MERGE

```

        TO ERR-SECTION OF SQLRT          MERGE
        PERFORM ZZ000-SQL-ERROR          MERGE
        END-IF                          MERGE
    ELSE                                MERGE
        MERGE
        MOVE INDUSTRY OF S-INDSTRY TO INDUSTRY OF ADMIN    MERGE
        MOVE INDUSTRY-SECTOR OF S-INDSTRY                MERGE
        TO INDUSTRY-SECTOR OF ADMIN                      MERGE
    END-IF                          MERGE
    ELSE                                MERGE
        MERGE
        SET CORE-INDUSTRY OF INDUSTRY OF ADMIN TO TRUE    MERGE
        SET CORE-SECTOR OF INDUSTRY-SECTOR OF ADMIN TO TRUE  MERGE
    END-IF                          MERGE
        MERGE
        .
        MERGE
        MERGE
GET-INDUSTRY-DATA-EXIT.
```

/*****

*

*

BJ000-DISPLAY-RUN-CONTROL SECTION.

BJ000.

*

*

DISPLAY ' for Operator ID : ' OPRID OF SQLRT

DISPLAY ' and Batch Run ID : ' BATCH-RUN-ID OF SQLRT

DISPLAY ' and Process Date : ' PROCESS-DT OF W-CNTL

DISPLAY ' '

DISPLAY ' CheckPoint/Restart Data'

DISPLAY ' CheckPoint Interval:'

CHKPT-INTERVAL OF W-CNTL

DISPLAY ' and COBRA Phase : ' COBRA-PHASE OF W-CNTL

DISPLAY ' and Overage Process : '

OVERAGE-PROCESS OF W-CNTL

DISPLAY ' and Benefit Program : ' BENEFIT-PROGRAM OF W-CNTL

DISPLAY ' and Employee ID : ' EMPLID OF W-CNTL

DISPLAY ' and Employee Record No : ' EMPL-RCD-NO OF W-CNTL

DISPLAY ' and Benefit Record No : ' BENEFIT-RCD-NO OF W-CNTL

DISPLAY ' and COBRA Event Date : ' COBRA-EVENT-DT OF W-CNTL

DISPLAY ' and COBRA Event ID : ' COBRA-EVENT-ID OF W-CNTL

DISPLAY ' and Dependent Benef : ' DEPENDENT-BENEF OF W-CNTL

.

DISPLAY-RUN-CONTROL-EXIT.

/*****

*

*

CA000-TERM SECTION.

CA000.

*

*

PERFORM CD000-CLEAR-RUNCTL

IF PROCESS-INSTANCE OF SQLRT NOT = ZERO

PERFORM CG000-SET-RUN-STAT-SUCCESSFUL

END-IF

PERFORM WZ000-COMMIT

PERFORM WX000-DISCONNECT-COMMON-CURSOR

PERFORM CJ000-DISCONNECT-FROM-SQLRT

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'COBRA Administration Ended at '

TIME-OUT OF W-WK '.'

.

TERM-EXIT.

/*****

*

*

CD000-CLEAR-RUNCTL SECTION.

CD000.

*

*

MOVE OPRID OF SQLRT TO OPRID OF D-RUNCTL

MOVE BATCH-RUN-ID OF SQLRT TO BATCH-RUN-ID OF D-RUNCTL

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF D-RUNCTL

BIND-SETUP OF D-RUNCTL

BIND-DATA OF D-RUNCTL

IF RTNCD-ERROR OF SQLRT

MOVE 'CLEAR-RUNCTL(RUNCTL)' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

CLEAR-RUNCTL-EXIT.

/*****

*

*

CG000-SET-RUN-STAT-SUCCESSFUL SECTION.

CG000.

*

*

SET RUN-STATUS-SUCCESSFUL OF USTAT TO TRUE

CALL 'PTPUSTAT' USING SQLRT

USTAT

IF RTNCD-ERROR OF SQLRT

MOVE 'SET-RUN-STAT-SUCCESSFUL(PTPUSTAT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SET-RUN-STAT-SUCCESSFUL-EXIT.

/*****

* *

CJ000-DISCONNECT-FROM-SQLRT SECTION.

CJ000.

* *

CALL 'PTPSQLRT' USING ACTION-DISCONNECT-ALL OF SQLRT

SQLRT

IF RTNCD=ERROR OF SQLRT

MOVE 'DISCONNECT-FROM-SQLRT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DISCONNECT-FROM-SQLRT-EXIT.

/*****

* *

DA000-RESET-FOR-REPROCESS SECTION.

DA000.

* *

PERFORM DD000-INIT-REPROCESS

PERFORM DG000-REPROCESS-EVENT

PERFORM DJ000-REPROCESS-PARTIC

PERFORM DM000-REPROCESS-PLAN

PERFORM DV000-DISC-REPROCESS

PERFORM DY000-TERM-REPROCESS

.

RESET-FOR-REPROCESS-EXIT.

/*****

* *

DD000-INIT-REPROCESS SECTION.

DD000.

* *

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'COBRA Reset for Reprocessing Started at '

TIME-OUT OF W-WK '.'

.

INIT-REPROCESS-EXIT.

/*****

* *

DG000-REPROCESS-EVENT SECTION.

DG000.

* *

MOVE ZERO TO WPARTIC-COUNT OF W-EVENT

MOVE ZERO TO WPLAN-COUNT OF W-EVENT

MOVE ZERO TO WOPTN-COUNT OF W-EVENT
MOVE ZERO TO WDPND-COUNT OF W-EVENT
SET REPROCESS-PARTIC-NO OF W-SW TO TRUE
SET REPROCESS-PLAN-NO OF W-SW TO TRUE

PERFORM DG100-SELECT-REPROCESS-EVENT
PERFORM DG200-FETCH-REPROCESS-EVENT

PERFORM UNTIL RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE
PERFORM DG300-SAVE-REPROCESS-EVENT

MOVE EMPLID OF W-EVENT
TO EMPLID OF BIND-DATA OF S-PAR1
MOVE BENEFIT-RCD-NO OF W-EVENT
TO BENEFIT-RCD-NO OF BIND-DATA OF S-PAR1
MOVE COBRA-EVENT-ID OF W-EVENT
TO COBRA-EVENT-ID OF BIND-DATA OF S-PAR1

PERFORM XE000-OBTAIN-PARTIC1

MOVE EMPLID OF W-EVENT
TO EMPLID OF BIND-DATA OF S-PARPLN1
MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF BIND-DATA OF S-PARPLN1
MOVE COBRA-EVENT-ID OF W-EVENT
TO COBRA-EVENT-ID OF BIND-DATA OF S-PARPLN1

PERFORM XV000-OBTAIN-PARTIC-PLN1

PERFORM XY000-DELETE-COBRA-BENEFIT

MOVE EMPLID OF W-EVENT TO EMPLID OF D-PAR

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF D-PAR

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF D-PAR

IF CBR-REPROCESS-QUALIFY OF WEVENT-DATA OF W-EVENT

PERFORM DG400-DELETE-PARTIC

PERFORM DG500-DELETE-PARTIC-PLAN

END-IF

IF CBR-REPROCESS-QUALIFY OF WEVENT-DATA OF W-EVENT

OR CBR-REPROCESS-PREPARE OF WEVENT-DATA

OF W-EVENT

PERFORM DG600-DELETE-PARTIC-OPTN

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF D-PARDPD1

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO

OF BIND-DATA OF D-PARDPD1

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID

OF BIND-DATA OF D-PARDPD1

PERFORM DG710-DELETE-PARTIC-DPD

END-IF

IF CBR-REPROCESS-PREPARE OF WEVENT-DATA OF W-EVENT

MOVE ZERO TO COBRA-EVENT-ID OF W-CURELT

PERFORM XM200-OBTAIN-HTH-DATA

PERFORM XM300-OBTAIN-FSA-DATA

END-IF

PERFORM DG800-UPDATE-REPROCESS-EVENT

PERFORM XW000-STORE-COBRA-DATA

PERFORM DG200-FETCH-REPROCESS-EVENT

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

REPROCESS-EVENT-EXIT.

/*****

*

*

DG100-SELECT-REPROCESS-EVENT SECTION.

DG100.

*

*

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-REPREVT OF W-CNTL

SQL-STMT OF S-REPREVT

BIND-SETUP OF S-REPREVT

BIND-DATA OF S-REPREVT

SELECT-SETUP OF S-REPREVT

SELECT-DATA OF S-REPREVT

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-REPROCESS-EVENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

SELECT-REPROCESS-EVENT-EXIT.

/*****

* *

DG200-FETCH-REPROCESS-EVENT SECTION.

DG200.

* *

INITIALIZE SELECT-DATA OF S-REPREVT

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-REPREVT OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-REPROCESS-EVENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-REPROCESS-EVENT-EXIT.

/*****

* *

DG300-SAVE-REPROCESS-EVENT SECTION.

DG300.

* *

INITIALIZE W-EVENT

MOVE EMPLID OF S-REPREVT TO EMPLID OF W-EVENT

MOVE BENEFIT-RCD-NO OF S-REPREVT

TO BENEFIT-RCD-NO OF W-EVENT

MOVE COBRA-EVENT-ID OF S-REPREVT

TO COBRA-EVENT-ID OF W-EVENT

MOVE COBRA-EVENT-DT OF S-REPREVT

TO COBRA-EVENT-DT OF W-EVENT

MOVE COBRA-EVENT-CLASS OF S-REPREVT

TO COBRA-EVENT-CLASS OF W-EVENT

MOVE CBR-QUALIFY-STATUS OF S-REPREVT

TO CBR-QUALIFY-STATUS OF WEVENT-DATA OF W-EVENT

MOVE CBR-PROCESS-STATUS OF S-REPREVT

TO CBR-PROCESS-STATUS OF WEVENT-DATA OF W-EVENT

MOVE CBR-EVT-REPRCS-IND OF S-REPREVT

TO CBR-EVT-REPRCS-IND OF W-EVENT

MOVE CBR-EVT-REPRCS-IND OF S-REPREVT

TO CBR-REPROCESS-IND OF W-SW

MOVE CBR-EVENT-CONFLICT OF S-REPREVT

TO CBR-EVENT-CONFLICT OF W-EVENT

.

SAVE-REPROCESS-EVENT-EXIT.

/*****

*

*

DG400-DELETE-PARTIC SECTION.

DG400.

*

*

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF D-PAR

BIND-SETUP OF D-PAR

BIND-DATA OF D-PAR

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid/BenRcd#/CBREvtID: '

EMPLID OF W-EVENT '/'

BENEFIT-RCD-NO OF W-EVENT '/'

COBRA-EVENT-ID OF W-EVENT

MOVE 'DELETE-PARTIC' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-PARTIC-EXIT.

/*****

* *

DG500-DELETE-PARTIC-PLAN SECTION.

DG500.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF D-PARPLN

BIND-SETUP OF D-PAR

BIND-DATA OF D-PAR

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid/BenRcd#/CBREvtID: '

EMPLID OF W-EVENT '/'

BENEFIT-RCD-NO OF W-EVENT '/'

COBRA-EVENT-ID OF W-EVENT

MOVE 'DELETE-PARTIC-PLAN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-PARTIC-PLAN-EXIT.

/*****

* *

DG600-DELETE-PARTIC-OPTN SECTION.

DG600.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF D-PAROPTN

BIND-SETUP OF D-PAR

BIND-DATA OF D-PAR

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid/BenRcd#/CBREvtID: '

EMPLID OF W-EVENT '/'

BENEFIT-RCD-NO OF W-EVENT '/'

COBRA-EVENT-ID OF W-EVENT

MOVE 'DELETE-PARTIC-OPTN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-PARTIC-OPTN-EXIT.

/*****

* *

DG700-DELETE-PARTIC-DPND SECTION.

DG700.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF D-PARDPND

BIND-SETUP OF D-PARDPND

BIND-DATA OF D-PARDPND

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid/BenRcd#/CBREvtID: '

EMPLID OF W-EVENT '/'

BENEFIT-RCD-NO OF W-EVENT '/'

COBRA-EVENT-ID OF W-EVENT

MOVE 'DELETE-PARTIC-DPND' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-PARTIC-DPND-EXIT.

/*****

* *

DG710-DELETE-PARTIC-DPD SECTION.

DG710.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF D-PARDPD1

BIND-SETUP OF D-PARDPD1

BIND-DATA OF D-PARDPD1

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid/BenRcd#/CBREvtID: '

EMPLID OF W-EVENT '/'

BENEFIT-RCD-NO OF W-EVENT '/'

COBRA-EVENT-ID OF W-EVENT

MOVE 'DELETE-PARTIC-DPD' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-PARTIC-DPD-EXIT.

/*****

* *

DG800-UPDATE-REPROCESS-EVENT SECTION.

DG800.

* *

SET UPDATE-YES OF WEVENT-DATA OF W-EVENT TO TRUE

IF CBR-REPROCESS-PREPARE OF WEVENT-DATA OF W-EVENT

PERFORM VARYING WPARTIC-IDX FROM 1 BY 1

UNTIL WPARTIC-IDX > WPARTIC-COUNT OF W-EVENT

IF NOT CBR-INIT-NOT-QUALIFIED OF W-EVENT(WPARTIC-IDX)

AND NOT CBR-REPROCESS-VOID OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

SET CBR-REPROCESS-NONE OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

TO TRUE

SET UPDATE-YES OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

TO TRUE

SET CBR-INIT-QUALIFIED OF W-EVENT(WPARTIC-IDX)
TO TRUE

SET CBR-PROCESS-OPEN OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

TO TRUE

END-IF

END-PERFORM

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

SET CBR-REPROCESS-PREPARE OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

TO TRUE

PERFORM DM400-UPDATE-REPROCESS-PLAN

END-PERFORM

END-IF

IF CBR-REPROCESS-VOID OF WEVENT-DATA OF W-EVENT

SET CBR-PROCESS-VOID OF WEVENT-DATA OF W-EVENT TO TRUE

PERFORM VARYING WPARTIC-IDX FROM 1 BY 1

UNTIL WPARTIC-IDX > WPARTIC-COUNT OF W-EVENT

SET CBR-REPROCESS-VOID OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

TO TRUE

PERFORM DJ400-UPDATE-REPROCESS-PARTIC

END-PERFORM

END-IF

IF CBR-REPROCESS-QUALIFY OF WEVENT-DATA OF W-EVENT

SET CBR-QUALIFY-UNPROCESSED OF WEVENT-DATA OF W-EVENT

TO TRUE

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

IF COBRA-EVENT-SECONDARY OF W-EVENT(WPLAN-IDX)

SET CLEAR-SCND-EVT-YES OF W-EVENT(WPLAN-IDX)

TO TRUE

END-IF

END-PERFORM

END-IF

SET CBR-REPROCESS-NONE OF WEVENT-DATA OF W-EVENT TO TRUE

.

UPDATE-REPROCESS-EVENT-EXIT.

/*****

* *

DJ000-REPROCESS-PARTIC SECTION.

DJ000.

* *

MOVE ZERO TO WPARTIC-COUNT OF W-EVENT

MOVE ZERO TO WPLAN-COUNT OF W-EVENT

MOVE ZERO TO WOPTN-COUNT OF W-EVENT

MOVE ZERO TO WDPND-COUNT OF W-EVENT

SET REPROCESS-PARTIC-YES OF W-SW TO TRUE

SET REPROCESS-PLAN-NO OF W-SW TO TRUE

PERFORM DJ100-SELECT-REPROCESS-PARTIC

PERFORM DJ200-FETCH-REPROCESS-PARTIC

PERFORM UNTIL RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ADD 1 TO WPARTIC-COUNT OF W-EVENT

PERFORM DJ300-SAVE-REPROCESS-PARTIC

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF S-PARPLN2

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO

OF BIND-DATA OF S-PARPLN2

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID

OF BIND-DATA OF S-PARPLN2

MOVE DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

TO DEPENDENT-BENEF

OF BIND-DATA OF S-PARPLN2

PERFORM XB000-OBTAIN-PARTIC-PLN2

PERFORM XY000-DELETE-COBRA-BENEFIT

PERFORM DJ400-UPDATE-REPROCESS-PARTIC

PERFORM XW000-STORE-COBRA-DATA

PERFORM DJ200-FETCH-REPROCESS-PARTIC

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

REPROCESS-PARTIC-EXIT.

/*****

* *

DJ100-SELECT-REPROCESS-PARTIC SECTION.

DJ100.

* *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-REPRPAR OF W-CNTL

SQL-STMT OF S-REPRPAR

BIND-SETUP OF S-REPRPAR

BIND-DATA OF S-REPRPAR

SELECT-SETUP OF S-REPRPAR

SELECT-DATA OF S-REPRPAR

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-REPROCESS-PARTIC' TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

.

SELECT-REPROCESS-PARTIC-EXIT.

/*****

* *

DJ200-FETCH-REPROCESS-PARTIC SECTION.

DJ200.

* *

INITIALIZE SELECT-DATA OF S-REPRPAR

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-REPRPAR OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-REPROCESS-PARTIC' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

FETCH-REPROCESS-PARTIC-EXIT.

/*****

*

*

DJ300-SAVE-REPROCESS-PARTIC SECTION.

DJ300.

*

*

SET WPARTIC-IDX TO WPARTIC-COUNT OF W-EVENT

INITIALIZE WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

MOVE EMPLID OF S-REPRPAR TO EMPLID OF W-EVENT

MOVE BENEFIT-RCD-NO OF S-REPRPAR

TO BENEFIT-RCD-NO OF W-EVENT

MOVE COBRA-EVENT-ID OF S-REPRPAR

TO COBRA-EVENT-ID OF W-EVENT

MOVE COBRA-EMPLID OF S-REPRPAR

TO COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

MOVE DEPENDENT-BENEF OF S-REPRPAR

TO DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

MOVE CBR-QUALIFY-STATUS OF S-REPRPAR

TO CBR-QUALIFY-STATUS OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

MOVE CBR-INIT-EVENT-STS OF S-REPRPAR

TO CBR-INIT-EVENT-STS OF W-EVENT(WPARTIC-IDX)

MOVE CBR-SCND-EVENT-STS OF S-REPRPAR

TO CBR-SCND-EVENT-STS OF W-EVENT(WPARTIC-IDX)

MOVE CBR-PAR-REPRCS-IND OF S-REPRPAR

TO CBR-PAR-REPRCS-IND OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

MOVE CBR-PAR-REPRCS-IND OF S-REPRPAR

TO CBR-REPROCESS-IND OF W-SW

.

SAVE-REPROCESS-PARTIC-EXIT.

/*****

* *

DJ400-UPDATE-REPROCESS-PARTIC SECTION.

DJ400.

* *

IF CBR-REPROCESS-VOID OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

SET CBR-PROCESS-VOID OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)


```
        TO TRUE
SET CBR-REPROCESS-NONE OF WPARTIC-DATA
    OF W-EVENT(WPARTIC-IDX)
        TO TRUE
SET UPDATE-YES OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
    TO TRUE

IF CBR-INIT-ENROLL-COMPLETE OF W-EVENT(WPARTIC-IDX)
    OR CBR-INIT-ELECT-ERROR OF W-EVENT(WPARTIC-IDX)
    OR CBR-INIT-ELECT-ENROLLED
        OF W-EVENT(WPARTIC-IDX)
    OR CBR-INIT-ELECT-ENTERED OF W-EVENT(WPARTIC-IDX)
    OR CBR-INIT-NOTIFIED OF W-EVENT(WPARTIC-IDX)

    IF CBR-INIT-NOTIFY-DT OF W-EVENT(WPARTIC-IDX)
        NOT = SPACE

        SET CBR-INIT-NOTIFIED OF W-EVENT(WPARTIC-IDX)
            TO TRUE
    ELSE
        SET CBR-INIT-OPTION-PREPARED
            OF W-EVENT(WPARTIC-IDX)
            TO TRUE
    END-IF
END-IF
```

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

SET CBR-REPROCESS-VOID OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO TRUE

PERFORM DM400-UPDATE-REPROCESS-PLAN

END-PERFORM

END-IF

.

UPDATE-REPROCESS-PARTIC-EXIT.

/*****

* *

DM000-REPROCESS-PLAN SECTION.

DM000.

* *

MOVE ZERO TO WPARTIC-COUNT OF W-EVENT

MOVE ZERO TO WPLAN-COUNT OF W-EVENT

MOVE ZERO TO WOPTN-COUNT OF W-EVENT

MOVE ZERO TO WDPND-COUNT OF W-EVENT

SET REPROCESS-PARTIC-NO OF W-SW TO TRUE

SET REPROCESS-PLAN-YES OF W-SW TO TRUE

PERFORM DM100-SELECT-REPROCESS-PLAN

PERFORM DM200-FETCH-REPROCESS-PLAN

PERFORM UNTIL RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ADD 1 TO WPLAN-COUNT OF W-EVENT

PERFORM DM300-SAVE-REPROCESS-PLAN

PERFORM XY000-DELETE-COBRA-BENEFIT

SET WPLAN-IDX TO WPLAN-COUNT OF W-EVENT

IF (CBR-REPROCESS-VOID OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

OR CBR-REPROCESS-ELECT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX))

AND (REPROCESS-PARTIC-YES OF W-SW

OR REPROCESS-PLAN-YES OF W-SW)

AND PLAN-TYPE-HEALTH OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF D-PARDPND

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO

```
        OF BIND-DATA OF D-PARDPND
MOVE COBRA-EVENT-ID OF W-EVENT
    TO COBRA-EVENT-ID
        OF BIND-DATA OF D-PARDPND
MOVE DEPENDENT-BENEF OF WPLAN-DATA
    OF W-EVENT(WPLAN-IDX)
        TO DEPENDENT-BENEF
            OF BIND-DATA
                OF D-PARDPND
MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
    TO PLAN-TYPE
        OF BIND-DATA OF D-PARDPND

PERFORM DG700-DELETE-PARTIC-DPND
END-IF

PERFORM VARYING WPLAN-IDX FROM 1 BY 1
    UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

PERFORM DM400-UPDATE-REPROCESS-PLAN
END-PERFORM

PERFORM XW000-STORE-COBRA-DATA
PERFORM DM200-FETCH-REPROCESS-PLAN

END-PERFORM
```

SET RTNCD-OK OF SQLRT TO TRUE

.

REPROCESS-PLAN-EXIT.

/*****

* *

DM100-SELECT-REPROCESS-PLAN SECTION.

DM100.

* *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-REPRPLN OF W-CNTL

SQL-STMT OF S-REPRPLN

BIND-SETUP OF S-REPRPLN

BIND-DATA OF S-REPRPLN

SELECT-SETUP OF S-REPRPLN

SELECT-DATA OF S-REPRPLN

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-REPROCESS-PLAN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-REPROCESS-PLAN-EXIT.

/*****

* *

DM200-FETCH-REPROCESS-PLAN SECTION.

DM200.

* *

INITIALIZE SELECT-DATA OF S-REPRPLN

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-REPRPLN OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-REPROCESS-PLAN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-REPROCESS-PLAN-EXIT.

/*****

* *

DM300-SAVE-REPROCESS-PLAN SECTION.

DM300.

* *

ADD 1 TO WPARTIC-COUNT OF W-EVENT

SET WPARTIC-IDX TO WPARTIC-COUNT OF W-EVENT

SET WPLAN-IDX TO WPLAN-COUNT OF W-EVENT

INITIALIZE WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE EMPLID OF S-REPRPLN TO EMPLID OF W-EVENT

MOVE BENEFIT-RCD-NO OF S-REPRPLN

TO BENEFIT-RCD-NO OF W-EVENT

MOVE COBRA-EVENT-ID OF S-REPRPLN

TO COBRA-EVENT-ID OF W-EVENT

MOVE DEPENDENT-BENEF OF S-REPRPLN

TO DEPENDENT-BENEF OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EMPLID OF S-REPRPLN

TO COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

MOVE DEPENDENT-BENEF OF S-REPRPLN

TO DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

MOVE PLAN-TYPE OF S-REPRPLN

TO PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-TYPE OF S-REPRPLN

TO COBRA-EVENT-TYPE OF W-EVENT(WPLAN-IDX)

MOVE CBR-INIT-EVENT-ID OF S-REPRPLN

TO CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

MOVE CBR-SCND-EVENT-ID OF S-REPRPLN

TO CBR-SCND-EVENT-ID OF W-EVENT(WPLAN-IDX)

MOVE CBR-PROCESS-STATUS OF S-REPRPLN

TO CBR-PROCESS-STATUS OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE CBR-ENROLL-STATUS OF S-REPRPLN

TO CBR-ENROLL-STATUS OF W-EVENT(WPLAN-IDX)

MOVE CBR-PLN-REPRCS-IND OF S-REPRPLN

TO CBR-PLN-REPRCS-IND

OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE CBR-PLN-REPRCS-IND OF S-REPRPLN

TO CBR-REPROCESS-IND OF W-SW

.

SAVE-REPROCESS-PLAN-EXIT.

/*****

*

*

DM400-UPDATE-REPROCESS-PLAN SECTION.

DM400.

*

*

IF CBR-REPROCESS-VOID OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

OR CBR-REPROCESS-PREPARE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

OR CBR-REPROCESS-ELECT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

SET CBR-ENROLL-NO OF W-EVENT(WPLAN-IDX) TO TRUE

SET COBRA-ERROR-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO TRUE

MOVE SPACE

TO COBRA-ELECT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE SPACE

TO COBRA-ELECT-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE SPACE

TO COBRA-WAIVE-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE SPACE

TO COBRA-REVOKE-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE SPACE TO OPTION-CD OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE SPACE TO BENEFIT-PLAN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE SPACE TO COVRG-CD OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE SPACE

TO COBRA-TERM-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

SET COBRA-NOT-TERMINATED OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO TRUE

SET UPDATE-YES OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO TRUE

END-IF

IF CBR-REPROCESS-PREPARE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

PERFORM DM410-SET-EMPL-CUR-COVRG

END-IF

END-IF

IF CBR-REPROCESS-VOID OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

SET CBR-PROCESS-VOID OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO TRUE

IF COBRA-EVENT-SECONDARY OF W-EVENT(WPLAN-IDX)

SET CLEAR-SCND-EVT-YES OF W-EVENT(WPLAN-IDX)
TO TRUE

END-IF

ELSE

SET CBR-PROCESS-OPEN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO TRUE

END-IF

SET CBR-REPROCESS-NONE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO TRUE

.

SAVE-REPROCESS-PLAN-EXIT.

/*****

* *

DM410-SET-EMPL-CUR-COVRG SECTION.

DM410.

* *

SET WCUR-IDX TO 1

SEARCH WCUR-DATA

AT END

CONTINUE

WHEN WCUR-IDX > WCUR-COUNT OF W-CURELT

CONTINUE

WHEN PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
= PLAN-TYPE OF WCUR-DATA OF W-CURELT(WCUR-IDX)

MOVE BENEFIT-PLAN OF WCUR-DATA OF W-CURELT(WCUR-IDX)
TO BENEFIT-PLAN OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

MOVE COVRG-CD OF WCUR-DATA OF W-CURELT(WCUR-IDX)
TO COVRG-CD OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

MOVE EMPL-CONTRBUTN-AMT OF W-CURELT(WCUR-IDX)
TO EMPL-CONTRBUTN-AMT OF W-EVENT(WPLAN-IDX)

MOVE ANN-EX-CREDIT-FSA OF W-CURELT(WCUR-IDX)
TO ANN-EX-CREDIT-FSA OF W-EVENT(WPLAN-IDX)

MOVE ANNUAL-PLEDGE OF W-CURELT(WCUR-IDX)
TO ANNUAL-PLEDGE OF W-EVENT(WPLAN-IDX)
MOVE FSA-SUB-AMT-YTD OF W-CURELT(WCUR-IDX)
TO FSA-SUB-AMT-YTD OF W-EVENT(WPLAN-IDX)
MOVE FSA-APR-AMT-YTD OF W-CURELT(WCUR-IDX)
TO FSA-APR-AMT-YTD OF W-EVENT(WPLAN-IDX)
MOVE FSA-PD-AMT-YTD OF W-CURELT(WCUR-IDX)
TO FSA-PD-AMT-YTD OF W-EVENT(WPLAN-IDX)
MOVE HLTH-PROVIDER-ID OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
TO HLTH-PROVIDER-ID OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE PREVIOUSLY-SEEN OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
TO PREVIOUSLY-SEEN OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE OTH-INSURANCE-IND OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
TO OTH-INSURANCE-IND OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE OTH-INSURANCE-NAME OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
TO OTH-INSURANCE-NAME OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

END-SEARCH

SET-EMPL-CUR-COVRG-EXIT.

/*****

* *

DV000-DISC-REPROCESS SECTION.

DV000.

* *

PERFORM QA000-DISC-CURSORS

IF SQL-CURSOR OF S-REPREVT OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-REPREVT OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(S-REPREVT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-REPRPAR OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-REPRPAR OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(S-REPRPAR)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-REPRPLN OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-REPRPLN OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(S-REPRPLN)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF D-BP-PGM OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF D-BP-PGM OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(D-BP-PGM)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF D-BP-PLN OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF D-BP-PLN OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(D-BP-PLN)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF D-HTH OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF D-HTH OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(D-HTH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF D-HTH-DEP OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF D-HTH-DEP OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(D-HTH-DEP)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF D-FSA OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF D-FSA OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(D-FSA)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF D-PAR OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF D-PAR OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(D-PAR)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF D-PARPLN OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF D-PARPLN OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(D-PARPLN)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF D-PAROPTN OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF D-PAROPTN OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(D-PAROPTN)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF D-PARDPD1 OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF D-PARDPD1 OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(D-PARDPD1)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF D-PARDPND OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF D-PARDPND OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-REPROCESS(D-PARDPND)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

DISC-REPROCESS-EXIT.

/*****

* *

DY000-TERM-REPROCESS SECTION.

DY000.

* *

PERFORM WA000-CLEAR-RESTART-KEYS

PERFORM WD000-SAVE-RESTART

PERFORM WZ000-COMMIT

* PERFORM WX000-DISCONNECT-COMMON-CURSOR

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'COBRA Reset for Reprocessing Ended at '
TIME-OUT OF W-WK '.'

IF OVERAGE-PROCESS-NO OF W-CNTL

SET COBRA-PHASE-ACTIVITY OF W-CNTL TO TRUE

END-IF

.

TERM-REPROCESS-EXIT.

/*****

* *

FA000-OVERAGE-DEPENDENT SECTION.

FA000.

* *

PERFORM FD000-INIT-OVERAGE-DEP

PERFORM FG000-SELECT-ACTIVE-PGM

PERFORM FJ000-FETCH-ACTIVE-PGM

PERFORM UNTIL RTNCD-END OF SQLRT

MOVE BENEFIT-PROGRAM OF SELECT-DATA OF S-BENPGM

TO BENEFIT-PROGRAM OF W-CBRDEFN

MOVE BENEFIT-PROGRAM OF SELECT-DATA OF S-BENPGM

TO BENEFIT-PROGRAM OF W-CNTL

MOVE EFFDT OF SELECT-DATA OF S-BENPGM

TO EFFDT OF W-CBRDEFN

MOVE COBRA-SURCHARGE OF S-BENPGM

TO COBRA-SURCHARGE OF W-CBRDEFN

MOVE COBRA-DISABL-SURCG OF S-BENPGM

TO COBRA-DISABL-SURCG OF W-CBRDEFN

PERFORM XO100-GET-MIN-DEP-RULE

PERFORM FL000-LOAD-COBRA-PLAN-DATA

IF WCBRPLN-COUNT OF W-CBRDEFN > ZERO

PERFORM FM000-PROCESS-EMPL-IN-PGM

END-IF

PERFORM FP000-FINISH-CURRENT-PGM

PERFORM FJ000-FETCH-ACTIVE-PGM

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

PERFORM FV000-DISC-OVERAGE-DEP

PERFORM FY000-TERM-OVERAGE-DEP

.

OVERAGE-DEPENDENT-EXIT.

/*****

* * *

FD000-INIT-OVERAGE-DEP SECTION.

FD000.

* * *

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'Overage Dependent Processing Started at '

TIME-OUT OF W-WK '.'

SET COBRA-PHASE-OVERAGE OF W-CNTL TO TRUE

SET PROGRESS-REPORT-OVERAGE OF W-WK TO TRUE

PERFORM TA000-START-RUN-UNIT

PERFORM FD100-GET-OVERAGE-EVENT

.

INIT-OVERAGE-DEP-EXIT.

/*****

* *

FD100-GET-OVERAGE-EVENT SECTION.

FD100.

* *

MOVE PROCESS-DT OF W-CNTL

TO COBRA-EVENT-DT OF W-EVENT

PERFORM XF000-LOAD-CBR-EVT-RULES

SET COBRA-EVENT-OVERAGE OF W-CNTL TO TRUE

SET CBR-SOURCE-OVG-PROCESS OF W-CNTL TO TRUE

PERFORM XH000-GET-CBR-EVT-RULE

.

GET-OVERAGE-EVENT-EXIT.

/*****

* *

FG000-SELECT-ACTIVE-PGM SECTION.

FG000.

* *

SET SQL-STMT-OVGPGM OF S-BENPGM TO TRUE

MOVE BENEFIT-PROGRAM OF W-CNTL

TO BENEFIT-PROGRAM OF BIND-DATA-OVGPGM OF S-BENPGM

BENEFIT-PROGRAM-2 OF BIND-DATA-OVGPGM OF S-BENPGM

BENEFIT-PROGRAM-3 OF BIND-DATA-OVGPGM OF S-BENPGM

MOVE PROCESS-DT OF W-CNTL

TO EFFDT OF BIND-DATA-OVGPGM OF S-BENPGM

MOVE PROCESS-DT OF W-CNTL

TO EFFDT-2 OF BIND-DATA-OVGPGM OF S-BENPGM

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-BENPGM OF W-CNTL

SQL-STMT OF S-BENPGM

BIND-SETUP-OVGPGM OF S-BENPGM

BIND-DATA-OVGPGM OF S-BENPGM

SELECT-SETUP OF S-BENPGM

SELECT-DATA OF S-BENPGM

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-ACTIVE-PGM' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-ACTIVE-PGM-EXIT.

/*****

* *

FJ000-FETCH-ACTIVE-PGM SECTION.

FJ000.

* *

INITIALIZE SELECT-DATA OF S-BENPGM

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-BENPGM OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

DISPLAY 'BENEFIT PROGRAM: '

BENEFIT-PROGRAM OF SELECT-DATA OF S-BENPGM

'PROCESS DATE: '

EFFDT OF SELECT-DATA OF S-BENPGM

MOVE 'FETCH-ACTIVE-PGM' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-ACTIVE-PGM-EXIT.

/*****

*

*

FL000-LOAD-COBRA-PLAN-DATA SECTION.

FL000.

*

*

MOVE ZERO TO WCBRPLN-COUNT OF W-CBRDEFN

PERFORM FL100-SELECT-CBRPLN

PERFORM FL200-FETCH-CBRPLN

PERFORM UNTIL RTNCD-END OF SQLRT

ADD 1 TO WCBRPLN-COUNT OF W-CBRDEFN

PERFORM FL300-SAVE-CBRPLN

PERFORM FL200-FETCH-CBRPLN

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

LOAD-COBRA-PLAN-EXIT.

/*****

* *

FL100-SELECT-CBRPLN SECTION.

FL100.

* *

MOVE BENEFIT-PROGRAM OF W-CBRDEFN

TO BENEFIT-PROGRAM OF S-CBRPLN

MOVE EFFDT OF W-CBRDEFN TO EFFDT OF S-CBRPLN

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-CBRPLN

BIND-SETUP OF S-CBRPLN

BIND-DATA OF S-CBRPLN

SELECT-SETUP OF S-CBRPLN

SELECT-DATA OF S-CBRPLN

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-CBRPLN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-CBRPLN-EXIT.

/*****

* *

FL200-FETCH-CBRPLN SECTION.

FL200.

* *

INITIALIZE SELECT-DATA OF S-CBRPLN

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

DISPLAY 'BENEFIT PROGRAM: ' BENEFIT-PROGRAM OF S-CBRPLN

'EFFECTIVE DATE: ' EFFDT OF S-CBRPLN

'PLAN TYPE: ' PLAN-TYPE OF S-CBRPLN

'EVENT RULES ID: ' EVENT-RULES-ID OF S-CBRPLN

MOVE 'FETCH-CBRPLN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-CBRPLN-EXIT.

/*****

* *

FL300-SAVE-CBRPLN SECTION.

FL300.

* *

SET WCBRPLN-IDX TO WCBRPLN-COUNT OF W-CBRDEFN

INITIALIZE WCBRPLN-DATA OF W-CBRDEFN(WCBRPLN-IDX)

MOVE PLAN-TYPE OF SELECT-DATA OF S-CBRPLN

TO PLAN-TYPE OF W-CBRDEFN(WCBRPLN-IDX)

MOVE DEP-RULE-ID OF SELECT-DATA OF S-CBRPLN

TO DEP-RULE-ID OF W-CBRDEFN(WCBRPLN-IDX)

.

SAVE-CBRPLN-EXIT.

/*****

* *

FM000-PROCESS-EMPL-IN-PGM SECTION.

FM000.

* *

PERFORM FM100-SELECT-EMPL-IN-PGM

PERFORM FM200-FETCH-EMPL-IN-PGM

PERFORM UNTIL RTNCD-END OF SQLRT

PERFORM FM300-INIT-EMPLOYEE

MOVE PROCESS-DT OF W-CNTL

TO BEGIN-DT OF DTWRK

MOVE 1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO COBRA-EVENT-DT OF W-EVENT

PERFORM XM200-OBTAIN-HTH-DATA

IF WCURDB-COUNT OF W-CURELT > 0

PERFORM FM400-PROCESS-DEPENDENT

IF OVERAGE-DEP-YES OF W-SW

PERFORM FM500-ADJUST-COVERAGE

IF ERROR-NO OF W-SW

PERFORM FM600-STORE-OVERAGE-ACTIVITY

END-IF

END-IF

END-IF

IF COBRA-EVENT-ID OF W-CURELT NOT = ZERO

AND NON-EMPLOYEE OF W-EVENT

AND COBRA-PARTICIPANT OF W-EVENT

AND WCUR-COUNT OF W-CURELT > 0

PERFORM FM700-COBRA-PARTIC-OVERAGE

END-IF

MOVE EMPLID OF SELECT-DATA OF S-EE-PGM

TO EMPLID OF W-CNTL

MOVE EMPL-RCD-NO OF SELECT-DATA OF S-EE-PGM

TO EMPL-RCD-NO OF W-CNTL

MOVE COBRA-EVENT-ID OF SELECT-DATA OF S-EE-PGM

TO COBRA-EVENT-ID OF W-CNTL

PERFORM TD000-CHECKPOINT-RUN-UNIT

IF RESELECT-YES OF W-SW

PERFORM FM100-SELECT-EMPL-IN-PGM

PERFORM FG000-SELECT-ACTIVE-PGM

END-IF

PERFORM FM200-FETCH-EMPL-IN-PGM

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

PROCESS-EMPL-IN-PGM-EXIT.

/*****

* *

FM100-SELECT-EMPL-IN-PGM SECTION.

FM100.

* *

MOVE BENEFIT-PROGRAM OF W-CBRDEFN

TO BENEFIT-PROGRAM OF BIND-DATA OF S-EE-PGM

MOVE PROCESS-DT OF W-CNTL

TO EFFDT OF BIND-DATA OF S-EE-PGM

MOVE PROCESS-DT OF W-CNTL

TO EFFDT-2 OF BIND-DATA OF S-EE-PGM

MOVE EMPLID OF W-CNTL TO EMPLID OF BIND-DATA OF S-EE-PGM

MOVE EMPLID OF W-CNTL TO EMPLID-2 OF BIND-DATA OF S-EE-PGM

EMPLID-3 OF BIND-DATA OF S-EE-PGM

EMPLID-4 OF BIND-DATA OF S-EE-PGM

MOVE EMPL-RCD-NO OF W-CNTL

TO EMPL-RCD-NO OF BIND-DATA OF S-EE-PGM

MOVE EMPL-RCD-NO OF W-CNTL

TO EMPL-RCD-NO-2 OF BIND-DATA OF S-EE-PGM

MOVE COBRA-EVENT-ID OF W-CNTL

TO COBRA-EVENT-ID OF BIND-DATA OF S-EE-PGM

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-EE-PGM OF W-CNTL

SQL-STMT OF S-EE-PGM

BIND-SETUP OF S-EE-PGM

BIND-DATA OF S-EE-PGM

SELECT-SETUP OF S-EE-PGM

SELECT-DATA OF S-EE-PGM

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-EMPL-IN-PGM' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-EMPL-IN-PGM-EXIT.

/*****

*

*

FM200-FETCH-EMPL-IN-PGM SECTION.

FM200.

*

*

INITIALIZE SELECT-DATA OF S-EE-PGM

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-EE-PGM OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

DISPLAY 'BENEFIT PROGRAM/PROCESS DT: '

BENEFIT-PROGRAM OF BIND-DATA OF S-EE-PGM ' / '

EFFDT OF BIND-DATA OF S-EE-PGM

DISPLAY 'EMPLID/EMPL-RCD-NO/COBRA-EVENT-ID: '

EMPLID OF SELECT-DATA OF S-EE-PGM ' / '

EMPL-RCD-NO OF SELECT-DATA OF S-EE-PGM '/'

COBRA-EVENT-ID OF SELECT-DATA OF S-EE-PGM

MOVE 'FETCH-EMPL-IN-PGM' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-EMPL-IN-PGM-EXIT.

/*****

* *

FM300-INIT-EMPLOYEE SECTION.

FM300.

* *

MOVE ZERO TO WDEPEND-COUNT OF W-DEPEND

MOVE ZERO TO WDEPEFF-COUNT OF W-DEPEND

MOVE ZERO TO WDEPNID-COUNT OF W-DEPNID

MOVE ZERO TO WCUR-COUNT OF W-CURELT

MOVE ZERO TO WCURDB-COUNT OF W-CURELT

MOVE ZERO TO WOVGACT-COUNT OF W-OVGACT

MOVE ZERO TO WADJ-COUNT OF W-ADJCVG

MOVE ZERO TO WADJDB-COUNT OF W-ADJCVG

SET OVERAGE-DEP-NO OF W-SW TO TRUE

SET ERROR-NO OF W-SW TO TRUE

MOVE EMPLID OF SELECT-DATA OF S-EE-PGM TO EMPLID OF W-EVENT

MOVE EMPL-RCD-NO OF SELECT-DATA OF S-EE-PGM

TO EMPL-RCD-NO OF W-EVENT

MOVE BENEFIT-RCD-NO OF S-EE-PGM

TO BENEFIT-RCD-NO OF W-EVENT

MOVE BENEFIT-PROGRAM OF SELECT-DATA OF S-EE-PGM

TO BENEFIT-PROGRAM OF W-CURELT

MOVE COBRA-EVENT-ID OF SELECT-DATA OF S-EE-PGM

TO COBRA-EVENT-ID OF W-CURELT

* MOVE PER-STATUS OF S-EE-PGM TO PER-STATUS OF W-EVENT
* MOVE PER-TYPE OF S-EE-PGM TO PER-TYPE OF W-EVENT
MOVE ZERO TO COBRA-EVENT-ID OF W-EVENT

.

INIT-EMPLOYEE-EXIT.

/*****

* *

FM400-PROCESS-DEPENDENT SECTION.

FM400.

* *

SET LOAD-EMPL-DEP-NO OF W-SW TO TRUE

PERFORM XN000-LOAD-DEPENDENT

MOVE BENEFIT-PROGRAM OF W-CBRDEFN

TO BENEFIT-PROGRAM OF BATBL

MOVE PROCESS-DT OF W-CNTL TO EVENT-DT OF BATBL

SET ACTION-LOAD-PGM-DEFN OF BATBL TO TRUE

PERFORM RA000-TABLE-ACCESS

PERFORM VARYING WDEPEND-IDX FROM 1 BY 1

UNTIL WDEPEND-IDX > WDEPEND-COUNT OF W-DEPEND

MOVE COBRA-EVENT-DT OF W-EVENT
TO SEARCH-EFFDT OF W-DEPEND
PERFORM XN700-SRCH-DEP-EFF-DATA

SET OVERAGE-NO OF W-DEPEND(WDEPEND-IDX) TO TRUE
PERFORM XP000-QUALIFY-DEPENDENT

IF QUALIFIED-YES OF W-DEPEND(WDEPEND-IDX)

IF AGE-LIMIT-YES OF W-DEPEND(WDEPEFF-IDX)

IF (DISABLED-YES OF W-DEPEND(WDEPEFF-IDX)
AND EXCL-DISABLED-NO OF W-DEPRULE)
OR DISABLED-NO
OF W-DEPEND(WDEPEFF-IDX)

PERFORM FM410-CHECK-HEALTH-DEPENDENT

END-IF

END-IF

END-IF

END-PERFORM

.

PROCESS-DEPENDENT-EXIT.

/*****

*

*

FM410-CHECK-HEALTH-DEPENDENT SECTION.

FM410.

*

*

PERFORM VARYING WCURDB-IDX FROM 1 BY 1

UNTIL WCURDB-IDX > WCURDB-COUNT OF W-CURELT

IF DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

= DEPENDENT-BENEF OF W-CURELT(WCURDB-IDX)

PERFORM FM411-CHECK-COBRA-PLAN

IF COBRA-PLAN-YES OF W-SW

MOVE PLAN-TYPE OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX) TO PLAN-TYPE OF W-SAVE

PERFORM XO950-SEARCH-DEP-PLNDFN

MOVE PROCESS-DT OF W-CNTL TO

AGE-AS-OF-DT OF W-CNTL

PERFORM XQ000-DEPENDENT-AGE-CHECK

MOVE OVERAGE-DT OF W-SAVE TO

OVERAGE-DT OF W-DEPEND(WDEPEND-IDX)

IF OVERAGE-S-YES OF W-SW

OR OVERAGE-NS-YES OF W-SW

* Check Ben Prog previously selected As of Process Date is still

* in effect as of Overage Date which is actual COBRA Event Date

MOVE OVERAGE-DT OF W-SAVE TO

COBRA-EVENT-DT OF W-EVENT

PERFORM XM110-SELECT-BP-PGM

PERFORM XM120-FETCH-BP-PGM

IF BENPRG-FOUND-YES OF W-SW

SET OVERAGE-YES OF W-DEPEND(WDEPEND-IDX)

TO TRUE

IF (COBRA-ACTION OF W-DEPEND(WDEPEND-IDX)

NOT = SPACE

AND NOT COBRA-ACTION-OVERAGE

OF W-DEPEND(WDEPEND-IDX))

PERFORM FM415-DPND-COBRA-CONFLICT

ELSE

SET OVERAGE-YES

OF W-CURELT(WCURDB-IDX) TO TRUE
MOVE OVERAGE-DT OF W-SAVE TO
OVERAGE-DT OF W-CURELT(WCURDB-IDX)

PERFORM FM412-LOAD-ADJUST-COVERAGE

IF COBRA-EVENT-ID OF W-CURELT
NOT = ZERO

IF NON-EMPLOYEE OF W-EVENT
AND COBRA-PARTICIPANT OF W-EVENT

PERFORM FM413-GET-RELATED-EMPLID
END-IF

PERFORM FM414-GET-INITIAL-PLAN
ELSE
MOVE COBRA-EVENT-CLASS OF W-CNTL
TO COBRA-ACTION
OF W-DEPEND(WDEPEND-IDX)

END-IF

IF COBRA-ACTION-OVERAGE
OF W-DEPEND(WDEPEND-IDX)

PERFORM
XS000-LOAD-OVERAGE-ACTIVITY

SET OVERAGE-DEP-YES OF W-SW
TO TRUE

END-IF

END-IF

END-IF

END-IF

END-IF

END-IF

END-PERFORM

.

CHECK-HEALTH-DEPENDENT-EXIT.

/*****

* *

FM411-CHECK-COBRA-PLAN SECTION.

FM411.

* *

SET COBRA-PLAN-NO OF W-SW TO TRUE

SET WCBRPLN-IDX TO 1

SEARCH WCBRPLN-DATA

WHEN PLAN-TYPE OF WCURDB-DATA OF W-CURELT(WCURDB-IDX)
= PLAN-TYPE OF W-CBRDEFN(WCBRPLN-IDX)

SET COBRA-PLAN-YES OF W-SW TO TRUE

END-SEARCH

.

CHECK-COBRA-PLAN-EXIT.

/*****

* *

FM412-LOAD-ADJUST-COVERAGE SECTION.

FM412.

* *

SET WADJ-IDX TO 1

SEARCH WADJ-DATA

AT END

DISPLAY 'Max # of Adjusted Coverage Occurs Exceeded'

DISPLAY 'Occurs count: ' WADJ-COUNT OF W-ADJCVG

'Emplid: ' EMPLID OF W-EVENT
MOVE 'LOAD-ADJUST-COVERAGE'
TO ERR-SECTION OF SQLRT
SET RTNCD-USER-ERROR OF SQLRT TO TRUE
PERFORM ZZ000-SQL-ERROR

WHEN WADJ-IDX > WADJ-COUNT OF W-ADJCVG

ADD 1 TO WADJ-COUNT OF W-ADJCVG
SET WADJ-IDX TO WADJ-COUNT
SET WADJ-EXIST-ENR-NO OF W-ADJCVG(WADJ-IDX) TO TRUE
MOVE OVERAGE-DT OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)
TO EFFDT OF WADJ-DATA
OF W-ADJCVG(WADJ-IDX)
MOVE PLAN-TYPE OF WCURDB-DATA OF W-CURELT(WCURDB-IDX)
TO PLAN-TYPE OF WADJ-DATA
OF W-ADJCVG(WADJ-IDX)

* *** SEE IF THERE IS AN ALREADY EXISTING ENROLLMENT FOR THE
* *** OVERAGE DATE. IF SO, SET THE FLAG AND LOAD THE
* *** BENEFIT PROGRAM, BENEFIT-PLAN AND COVRG-CD FROM THE
* *** EXISTING ENROLLMENT

IF COBRA-PHASE-OVERAGE OF W-CNTL
PERFORM FM420-SELECT-EXISTING-ENR
PERFORM FM421-FETCH-EXISTING-ENR

IF NOT RTNCD-END OF SQLRT

SET WADJ-EXIST-ENR-YES OF W-ADJCVG(WADJ-IDX)

TO TRUE

MOVE BENEFIT-PROGRAM OF SELECT-DATA OF S-XHTH

TO BENEFIT-PROGRAM OF W-ADJCVG(WADJ-IDX)

MOVE BENEFIT-PLAN OF SELECT-DATA OF S-XHTH

TO BENEFIT-PLAN OF W-ADJCVG(WADJ-IDX)

MOVE COVRG-CD OF SELECT-DATA OF S-XHTH

TO COVRG-CD OF W-ADJCVG(WADJ-IDX)

END-IF

SET RTNCD-OK OF SQLRT TO TRUE

END-IF

MOVE ZERO TO DEPENDENT-COUNT OF W-ADJCVG(WADJ-IDX)

MOVE ZERO TO CHILD-COUNT OF W-ADJCVG(WADJ-IDX)

SET DEP-SPOUSE-NO OF W-ADJCVG(WADJ-IDX) TO TRUE

WHEN OVERAGE-DT OF WCURDB-DATA OF W-CURELT(WCURDB-IDX)

= EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

AND PLAN-TYPE OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

= PLAN-TYPE OF WADJ-DATA

OF W-ADJCVG(WADJ-IDX)

CONTINUE

END-SEARCH

LOAD-ADJUST-COVERAGE-EXIT.

/*****

* *

FM413-GET-RELATED-EMPLID SECTION.

FM413.

* *

MOVE EMPLID OF W-EVENT TO COBRA-EMPLID OF S-RELEMP

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT OF S-RELEMP

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT-2 OF S-RELEMP

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-RELEMP

BIND-SETUP OF S-RELEMP

BIND-DATA OF S-RELEMP

SELECT-SETUP OF S-RELEMP

SELECT-DATA OF S-RELEMP

IF RTNCD-ERROR OF SQLRT

```
MOVE 'GET-RELATED-EMPLID(SELECT)'
    TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

INITIALIZE SELECT-DATA OF S-RELEMP

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT
    SQLRT
    SQL-CURSOR-COMMON OF SQLRT
IF RTNCD-ERROR OF SQLRT

    IF RTNCD-END OF SQLRT

        SET RTNCD-OK OF SQLRT TO TRUE
    ELSE

        DISPLAY 'EMPLID: ' COBRA-EMPLID OF S-RELEMP
        DISPLAY 'EFFDT: ' EFFDT OF S-RELEMP
        MOVE 'GET-RELATED-EMPL-DATA(FETCH)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

MOVE EMPLID OF S-RELEMP TO EMPLID OF W-EVENT
```


GET-RELATED-EMPLID-EXIT.

/*****

* *

FM414-GET-INITIAL-PLAN SECTION.

FM414.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF S-INITPLN

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF S-INITPLN

MOVE COBRA-EVENT-ID OF W-CURELT

TO COBRA-EVENT-ID OF S-INITPLN

MOVE DEPENDENT-BENEF OF W-CURELT(WCURDB-IDX)

TO DEPENDENT-BENEF OF S-INITPLN

MOVE PLAN-TYPE OF WCURDB-DATA OF W-CURELT(WCURDB-IDX)

TO PLAN-TYPE OF S-INITPLN

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-INITPLN

BIND-SETUP OF S-INITPLN

BIND-DATA OF S-INITPLN

SELECT-SETUP OF S-INITPLN

SELECT-DATA OF S-INITPLN

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-INITIAL-PLAN(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-INITPLN

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

DISPLAY 'Emplid/BenRcd#/CBREvtID/DepBen/PlnTyp: '

EMPLID OF S-INITPLN '/'

BENEFIT-RCD-NO OF S-INITPLN '/'

COBRA-EVENT-ID OF S-INITPLN '/'

DEPENDENT-BENEF OF S-INITPLN

PLAN-TYPE OF S-INITPLN

MOVE 'GET-INITIAL-PLAN(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

MOVE COBRA-EVENT-CLASS OF W-CNTL

TO COBRA-ACTION OF W-DEPEND(WDEPEND-IDX)

END-IF

.

GET-INITIAL-PLAN-EXIT.

/*****

* *

FM415-DPND-COBRA-CONFLICT SECTION.

FM415.

* *

SET MSGID-DPND-COBRA-CONFLICT OF PYMSG TO TRUE

MOVE COBRA-EVENT-ID OF W-EVENT TO COBRA-EVENT-ID OF PYMSG

MOVE DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

TO DEPENDENT-BENEF OF PYMSG

MOVE COBRA-ACTION OF W-DEPEND(WDEPEND-IDX)

TO MSGDATA1 OF PYMSG

MOVE COBRA-EVENT-DT OF W-DEPEND(WDEPEND-IDX)

TO MSGDATA2 OF PYMSG

PERFORM ZM000-MESSAGE

.

DPND-COBRA-CONFLICT-EXIT.

/*****

*

*

FM420-SELECT-EXISTING-ENR SECTION.

FM420.

*

*

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF S-XHTH

MOVE EMPL-RCD-NO OF W-EVENT

TO EMPL-RCD-NO OF BIND-DATA OF S-XHTH

MOVE COBRA-EVENT-ID OF W-CURELT

TO COBRA-EVENT-ID OF BIND-DATA OF S-XHTH

MOVE PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO PLAN-TYPE OF BIND-DATA OF S-XHTH

MOVE EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO EFFDT OF BIND-DATA OF S-XHTH

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-XHTH

BIND-SETUP OF S-XHTH

BIND-DATA OF S-XHTH

SELECT-SETUP OF S-XHTH

SELECT-DATA OF S-XHTH

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-EXISTING-ENR' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-EXISTING-ENR-EXIT.

/*****

* *

FM421-FETCH-EXISTING-ENR SECTION.

FM421.

* *

INITIALIZE SELECT-DATA OF S-XHTH

SET RTNCD-OK OF SQLRT TO TRUE

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

DISPLAY 'Emplid/EmplRcd#/CBREvtID: '

EMPLID OF BIND-DATA OF S-XHTH '/'

EMPL-RCD-NO OF BIND-DATA OF S-XHTH '/'

COBRA-EVENT-ID OF BIND-DATA OF S-XHTH

DISPLAY 'PlnTyp/EffDt: '

PLAN-TYPE OF BIND-DATA OF S-XHTH '/'

EFFDT OF BIND-DATA OF S-XHTH

MOVE 'FETCH-EXISTING-ENR' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-EXISTING-ENR-EXIT.

/*****

*

*

FM500-ADJUST-COVERAGE SECTION.

FM500.

*

*

PERFORM VARYING WADJ-IDX FROM 1 BY 1

UNTIL WADJ-IDX > WADJ-COUNT OF W-ADJCVG

PERFORM FM510-LOAD-ADJUST-DEPENDENT

PERFORM FM520-SET-CURRENT-COVERAGE

PERFORM FM530-LOAD-TABLE-DATA

PERFORM FM540-VALIDATE-CUR-COVRG-CD

IF VALID-COVRG-CD-NO OF W-SW

PERFORM FM550-DETERMINE-NEW-COVRG-CD

IF VALID-COVRG-CD-NO OF W-SW

SET ERROR-YES OF W-SW TO TRUE

SET MSGID-OVG-COVRG-CD-INVALID OF PYMSG

TO TRUE

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF PYMSG

MOVE DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

TO DEPENDENT-BENEF OF PYMSG
MOVE PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)
TO MSGDATA1 OF PYMSG
MOVE BENEFIT-PLAN OF W-ADJCVG(WADJ-IDX)
TO MSGDATA2 OF PYMSG
MOVE COVRG-CD OF W-ADJCVG(WADJ-IDX)
TO MSGDATA3 OF PYMSG
PERFORM ZM000-MESSAGE
END-IF

END-IF

END-PERFORM

IF ERROR-NO OF W-SW

PERFORM VARYING WADJ-IDX FROM 1 BY 1
UNTIL WADJ-IDX > WADJ-COUNT OF W-ADJCVG

IF WADJ-EXIST-ENR-NO OF W-ADJCVG(WADJ-IDX)

PERFORM FM560-ADJUST-HEALTH-BENEFIT

ELSE

PERFORM FM562-UPDATE-HEALTH-BENEFIT

PERFORM FM564-DELETE-HEALTH-DEPENDNT

END-IF

END-PERFORM

PERFORM VARYING WADJDB-IDX FROM 1 BY 1

UNTIL WADJDB-IDX > WADJDB-COUNT OF W-ADJCVG

SET WADJ-IDX TO 1

SEARCH WADJ-DATA

AT END

CONTINUE

WHEN WADJ-IDX > WADJ-COUNT

CONTINUE

WHEN PLAN-TYPE OF WADJ-DATA

OF W-ADJCVG(WADJ-IDX)

= PLAN-TYPE OF WADJDB-DATA

OF W-ADJCVG(WADJDB-IDX)

IF WADJ-EXIST-ENR-NO

OF W-ADJCVG(WADJ-IDX)

PERFORM

FM570-ADJUST-HEALTH-DEPENDENT

END-IF

END-SEARCH

END-PERFORM

END-IF

ADJUST-COVERAGE-EXIT.

/*****

*

*

FM510-LOAD-ADJUST-DEPENDENT SECTION.

FM510.

*

*

IF WADJ-EXIST-ENR-YES OF W-ADJCVG(WADJ-IDX)

PERFORM FM512-RELOAD-CUR-DEPENDENT

END-IF

PERFORM VARYING WCURDB-IDX FROM 1 BY 1

UNTIL WCURDB-IDX > WCURDB-COUNT OF W-CURELT

IF PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

= PLAN-TYPE OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

AND ((EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

< OVERAGE-DT OF W-CURELT(WCURDB-IDX))

OR OVERAGE-NO OF W-CURELT(WCURDB-IDX))

ADD 1 TO WADJDB-COUNT OF W-ADJCVG
ADD 1 TO DEPENDENT-COUNT OF W-ADJCVG(WADJ-IDX)
SET WADJDB-IDX TO WADJDB-COUNT
MOVE EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)
TO EFFDT OF WADJDB-DATA
OF W-ADJCVG(WADJDB-IDX)
MOVE PLAN-TYPE OF WCURDB-DATA OF W-CURELT(WCURDB-IDX)
TO PLAN-TYPE OF WADJDB-DATA
OF W-ADJCVG(WADJDB-IDX)
MOVE DEPENDENT-BENEF OF W-CURELT(WCURDB-IDX)
TO DEPENDENT-BENEF OF WADJDB-DATA
OF W-ADJCVG(WADJDB-IDX)
MOVE HLTH-PROVIDER-ID OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)
TO HLTH-PROVIDER-ID OF WADJDB-DATA
OF W-ADJCVG(WADJDB-IDX)
MOVE PREVIOUSLY-SEEN OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)
TO PREVIOUSLY-SEEN OF WADJDB-DATA
OF W-ADJCVG(WADJDB-IDX)
MOVE OTH-INSURANCE-IND OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)
TO OTH-INSURANCE-IND OF WADJDB-DATA
OF W-ADJCVG(WADJDB-IDX)
MOVE OTH-INSURANCE-NAME OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)

TO OTH-INSURANCE-NAME OF WADJDB-DATA
OF W-ADJCVG(WADJDB-IDX)
PERFORM FM511-GET-DEP-RELATIONSHIP
SET WDEPEFF-IDX TO WDEPEND-IDX

IF SPOUSE OF W-DEPEND(WDEPEFF-IDX) OR
SAME-SEX-SPOUSE OF W-DEPEND(WDEPEFF-IDX)
SET DEP-SPOUSE-YES OF W-ADJCVG(WADJ-IDX)
TO TRUE
END-IF

IF NOT SPOUSE OF W-DEPEND(WDEPEFF-IDX)
AND NOT SAME-SEX-SPOUSE OF W-DEPEND(WDEPEFF-IDX)
AND NOT EX-SPOUSE OF W-DEPEND(WDEPEFF-IDX)
AND NOT EMPLOYEE OF W-DEPEND(WDEPEFF-IDX)

ADD 1 TO CHILD-COUNT OF W-ADJCVG(WADJ-IDX)

END-IF

END-IF

END-PERFORM

.

LOAD-ADJUST-DEPENDENT-EXIT.

/*****

* *

FM511-GET-DEP-RELATIONSHIP SECTION.

FM511.

* *

SET WDEPEND-IDX TO 1

SEARCH WDEPEND-DATA

WHEN DEPENDENT-BENEF OF W-ADJCVG(WADJDB-IDX)

= DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

CONTINUE

END-SEARCH

.

GET-DEP-RELATIONSHIP-EXIT.

/*****

* *

FM512-RELOAD-CUR-DEPENDENT SECTION.

FM512.

* *

```
PERFORM VARYING WCURDB-IDX FROM 1 BY 1
    UNTIL WCURDB-IDX > WCURDB-COUNT OF W-CURELT
    IF PLAN-TYPE OF WCURDB-DATA OF W-CURELT(WCURDB-IDX)
        = PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)
        INITIALIZE WCURDB-DATA OF W-CURELT(WCURDB-IDX)
    END-IF
END-PERFORM
```

```
PERFORM FM514-SELECT-EXISTING-DEP
PERFORM FM515-FETCH-EXISTING-DEP
PERFORM UNTIL RTNCD-END OF SQLRT
    PERFORM FM516-SAVE-RELOAD-DEP
    PERFORM FM515-FETCH-EXISTING-DEP
END-PERFORM
SET RTNCD-OK OF SQLRT TO TRUE
```

.

RELOAD-CUR-DEPENDENT-EXIT.

/*****

* * *

FM514-SELECT-EXISTING-DEP SECTION.

FM514.

*

*

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF S-XHTH-DEP

MOVE EMPL-RCD-NO OF W-EVENT

TO EMPL-RCD-NO OF BIND-DATA OF S-XHTH-DEP

MOVE COBRA-EVENT-ID OF W-CURELT

TO COBRA-EVENT-ID OF BIND-DATA OF S-XHTH-DEP

MOVE PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO PLAN-TYPE OF BIND-DATA OF S-XHTH-DEP

MOVE EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO EFFDT OF BIND-DATA OF S-XHTH-DEP

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-XHTH-DEP

BIND-SETUP OF S-XHTH-DEP

BIND-DATA OF S-XHTH-DEP

SELECT-SETUP OF S-XHTH-DEP

SELECT-DATA OF S-XHTH-DEP

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-EXISTING-DEP' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-EXISTING-DEP-EXIT.

/*****

* *

FM515-FETCH-EXISTING-DEP SECTION.

FM515.

* *

INITIALIZE SELECT-DATA OF S-XHTH-DEP

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

DISPLAY 'Emplid/EmplRcd#/CBREvtID: '

EMPLID OF BIND-DATA OF S-XHTH-DEP '/'

EMPL-RCD-NO OF BIND-DATA OF S-XHTH-DEP '/'

COBRA-EVENT-ID OF BIND-DATA OF S-XHTH-DEP '/'

DISPLAY 'PlnTyp/EffDt: '

PLAN-TYPE OF BIND-DATA OF S-XHTH-DEP '/'

EFFDT OF BIND-DATA OF S-XHTH-DEP '/'

MOVE 'FETCH-EXISTING-DEP' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-EXISTING-DEP-EXIT.

/*****

* *

FM516-SAVE-RELOAD-DEP SECTION.

FM516.

* *

* FOR THIS PLAN-TYPE, WE MAY HAVE SOME AVAILABLE SLOTS IN THE

* MIDDLE OF THE STACK THAT NEED TO BE RE-USED.

*

SET WCURDB-IDX TO 1

MOVE 1 TO WORK-COUNT OF W-WK

SEARCH WCURDB-DATA

AT END

DISPLAY 'MAXIMUM # OF CURRENT DEPENDENT ENROLLMENTS'

DISPLAY 'Occurs count: ' WCURDB-COUNT OF W-CURELT

'Emplid: ' EMPLID OF W-EVENT

MOVE 'SAVE-RELOAD-DEP'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

WHEN PLAN-TYPE OF WCURDB-DATA OF W-CURELT(WCURDB-IDX)

= SPACES

OR WCURDB-IDX > WCURDB-COUNT OF W-CURELT

MOVE PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO PLAN-TYPE OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

MOVE BENEFIT-PLAN OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO BENEFIT-PLAN OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

MOVE DEPENDENT-BENEF OF S-XHTH-DEP

TO DEPENDENT-BENEF OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

MOVE COVRG-CD OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO COVRG-CD OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

SET WDEPEND-IDX TO 1

SEARCH WDEPEND-DATA

AT END

CONTINUE

WHEN DEPENDENT-BENEF OF WDEPEND-DATA

OF W-DEPEND(WDEPEND-IDX)

= DEPENDENT-BENEF OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

MOVE OVERAGE-IND OF WDEPEND-DATA

OF W-DEPEND(WDEPEND-IDX)

TO OVERAGE-IND OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

MOVE OVERAGE-DT OF WDEPEND-DATA

OF W-DEPEND(WDEPEND-IDX)

TO OVERAGE-DT OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

END-SEARCH

ADD 1 TO WORK-COUNT OF W-WK

END-SEARCH

IF WCURDB-IDX > WCURDB-COUNT OF W-CURELT

MOVE WORK-COUNT OF W-WK TO WCURDB-COUNT OF W-CURELT

END-IF

.

SAVE-RELOAD-DEP-EXIT.

/*****

* *

FM520-SET-CURRENT-COVERAGE SECTION.

FM520.

* *

SET WCUR-IDX TO 1

SEARCH WCUR-DATA

WHEN PLAN-TYPE OF WCUR-DATA OF W-CURELT(WCUR-IDX)
= PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

IF WADJ-EXIST-ENR-NO OF W-ADJCVG(WADJ-IDX)

MOVE BENEFIT-PLAN OF WCUR-DATA

OF W-CURELT(WCUR-IDX)

TO BENEFIT-PLAN OF W-ADJCVG(WADJ-IDX)

MOVE COVRG-CD OF WCUR-DATA OF W-CURELT(WCUR-IDX)

TO COVRG-CD OF W-ADJCVG(WADJ-IDX)

MOVE HLTH-PROVIDER-ID OF WCUR-DATA

OF W-CURELT(WCUR-IDX)

TO HLTH-PROVIDER-ID OF WADJ-DATA

OF W-ADJCVG(WADJ-IDX)

MOVE PREVIOUSLY-SEEN OF WCUR-DATA

OF W-CURELT(WCUR-IDX)

TO PREVIOUSLY-SEEN OF WADJ-DATA
OF W-ADJCVG(WADJ-IDX)
MOVE OTH-INSURANCE-IND OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
TO OTH-INSURANCE-IND OF WADJ-DATA
OF W-ADJCVG(WADJ-IDX)
MOVE OTH-INSURANCE-NAME OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
TO OTH-INSURANCE-NAME OF WADJ-DATA
OF W-ADJCVG(WADJ-IDX)
END-IF
END-SEARCH

.

SET-CURRENT-COVERAGE-EXIT.

/*****

* *

FM530-LOAD-TABLE-DATA SECTION.

FM530.

* *

IF COBRA-PHASE-OVERAGE OF W-CNTL
AND WADJ-EXIST-ENR-YES OF W-ADJCVG(WADJ-IDX)

MOVE BENEFIT-PROGRAM OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)
TO BENEFIT-PROGRAM OF BATBL

ELSE

MOVE BENEFIT-PROGRAM OF W-CBRDEFN
TO BENEFIT-PROGRAM OF BATBL

END-IF

MOVE EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)
TO EVENT-DT OF BATBL
SET ACTION-LOAD-PGM-DEFN OF BATBL TO TRUE
PERFORM RA000-TABLE-ACCESS

MOVE EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)
TO EFFDT OF BIND-DATA OF S-CVGCD
MOVE EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)
TO EFFDT-2 OF BIND-DATA OF S-CVGCD
PERFORM XJ000-LOAD-COVRG-CD-TBL

.

LOAD-TABLE-DATA-EXIT.

/*****

*

*

FM540-VALIDATE-CUR-COVRG-CD SECTION.

FM540.

*

*

SET WCVGCD-IDX TO 1

SEARCH WCVGCD-DATA

AT END

SET VALID-COVRG-CD-NO OF W-SW TO TRUE

WHEN WCVGCD-IDX > WCVGCD-COUNT OF W-CVGCD

SET VALID-COVRG-CD-NO OF W-SW TO TRUE

WHEN COVRG-CD OF W-ADJCVG(WADJ-IDX)

= COVRG-CD OF W-CVGCD(WCVGCD-IDX)

MOVE CBR-COVRG-SET OF W-CVGCD(WCVGCD-IDX)

TO CBR-COVRG-SET OF W-ADJCVG(WADJ-IDX)

IF (DEP-SPOUSE-NO OF W-ADJCVG(WADJ-IDX)

AND SPOUSE-COVERAGE-REQUIRED

OF W-CVGCD(WCVGCD-IDX))

OR (SPOUSE-COVERAGE-SPOUSE-ONLY

```

        OF W-CVGCD(WCVGCD-IDX)
    OR  SPOUSE-COVERAGE-DEPEND-ONLY
        OF W-CVGCD(WCVGCD-IDX)
    OR  SPOUSE-COVERAGE-SPSDEP-ONLY
        OF W-CVGCD(WCVGCD-IDX))
    OR (DEPENDENT-COUNT OF W-ADJCVG(WADJ-IDX)
        >  MAX-NUM-OF-DEPS
        OF W-CVGCD(WCVGCD-IDX))
    OR (DEPENDENT-COUNT OF W-ADJCVG(WADJ-IDX)
        <  MIN-NUM-OF-DEPS
        OF W-CVGCD(WCVGCD-IDX))
    OR (CC-CTL-TOTAL-NO OF W-CVGCD(WCVGCD-IDX)
        AND CHILD-COUNT OF W-ADJCVG(WADJ-IDX)
        >  MAX-NUM-CHILD
        OF W-CVGCD(WCVGCD-IDX))
    OR (CC-CTL-TOTAL-NO OF W-CVGCD(WCVGCD-IDX)
        AND CHILD-COUNT OF W-ADJCVG(WADJ-IDX)
        <  MIN-NUM-CHILD
        OF W-CVGCD(WCVGCD-IDX))

    SET VALID-COVRG-CD-NO OF W-SW  TO  TRUE
ELSE
    SET VALID-COVRG-CD-YES OF W-SW  TO  TRUE
    PERFORM FM551-CHECK-OPTN-MATCH
END-IF
END-SEARCH

```


VALIDATE-CUR-COVRG-CD-EXIT.

/*****

*

*

FM550-DETERMINE-NEW-COVRG-CD SECTION.

FM550.

*

*

PERFORM VARYING WCVGCD-IDX FROM 1 BY 1

UNTIL WCVGCD-IDX > WCVGCD-COUNT OF W-CVGCD

OR VALID-COVRG-CD-YES OF W-SW

IF (DEP-SPOUSE-NO OF W-ADJCVG(WADJ-IDX)

AND SPOUSE-COVERAGE-REQUIRED

OF W-CVGCD(WCVGCD-IDX))

OR (DEP-SPOUSE-YES OF W-ADJCVG(WADJ-IDX)

AND SPOUSE-COVERAGE-NOT-ALLOWED

OF W-CVGCD(WCVGCD-IDX))

OR (SPOUSE-COVERAGE-SPOUSE-ONLY

OF W-CVGCD(WCVGCD-IDX)

OR SPOUSE-COVERAGE-DEPEND-ONLY

OF W-CVGCD(WCVGCD-IDX)

```

        OR SPOUSE-COVERAGE-SPSDEP-ONLY
          OF W-CVGCD(WCVGCD-IDX))
    OR (DEPENDENT-COUNT OF W-ADJCVG(WADJ-IDX)
        > MAX-NUM-OF-DEPS
          OF W-CVGCD(WCVGCD-IDX))
    OR (DEPENDENT-COUNT OF W-ADJCVG(WADJ-IDX)
        < MIN-NUM-OF-DEPS
          OF W-CVGCD(WCVGCD-IDX))
    OR (CC-CTL-TOTAL-NO OF W-CVGCD(WCVGCD-IDX)
        AND CHILD-COUNT OF W-ADJCVG(WADJ-IDX)
        > MAX-NUM-CHILD
          OF W-CVGCD(WCVGCD-IDX))
    OR (CC-CTL-TOTAL-NO OF W-CVGCD(WCVGCD-IDX)
        AND CHILD-COUNT OF W-ADJCVG(WADJ-IDX)
        < MIN-NUM-CHILD
          OF W-CVGCD(WCVGCD-IDX))

    SET VALID-COVRG-CD-NO OF W-SW TO TRUE
ELSE
    IF CBR-COVRG-SET OF W-CVGCD(WCVGCD-IDX)
        = CBR-COVRG-SET OF W-ADJCVG(WADJ-IDX)
        MOVE COVRG-CD OF W-CVGCD(WCVGCD-IDX)
            TO COVRG-CD OF W-ADJCVG(WADJ-IDX)
        PERFORM FM551-CHECK-OPTN-MATCH
    END-IF
END-IF

```

END-PERFORM

.

DETERMINE-NEW-COVRG-CD-EXIT.

/*****

* *

FM551-CHECK-OPTN-MATCH SECTION.

FM551.

* *

SET PLANDF-IDX TO 1

SEARCH PLANDF-DATA

AT END

DISPLAY

'Cobra Plan Not Defined in Pgm/PlanType/Effdt: '

BENEFIT-PROGRAM OF PGMDF-DATA

/'

PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

/'

EFFDT OF PGMDF-DATA

DISPLAY

'Emplid: ' EMPLID OF W-EVENT

MOVE 'CHECK-OPTN-MATCH'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

WHEN PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

= PLAN-TYPE OF PLANDF-DATA OF PDEFN(PLANDF-IDX)

CONTINUE

END-SEARCH

SET OPTNDF-IDX TO OPTNDF-START OF PDEFN(PLANDF-IDX)

SEARCH OPTNDF-DATA

AT END

SET VALID-COVRG-CD-NO OF W-SW TO TRUE

WHEN OPTNDF-IDX > OPTNDF-END OF PDEFN(PLANDF-IDX)

SET VALID-COVRG-CD-NO OF W-SW TO TRUE

WHEN BENEFIT-PLAN OF W-ADJCVG(WADJ-IDX)

= BENEFIT-PLAN OF ODEFN(OPTNDF-IDX)
AND COVRG-CD OF W-ADJCVG(WADJ-IDX)
= COVRG-CD OF ODEFN(OPTNDF-IDX)

SET VALID-COVRG-CD-YES OF W-SW TO TRUE
END-SEARCH

CHECK-OPTN-MATCH-EXIT.

/*****

* *

FM560-ADJUST-HEALTH-BENEFIT SECTION.

FM560.

* *

INITIALIZE BIND-DATA OF I-HTH
MOVE EMPLID OF W-EVENT TO EMPLID OF I-HTH
MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF I-HTH
MOVE COBRA-EVENT-ID OF W-CURELT TO COBRA-EVENT-ID OF I-HTH
MOVE PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)
TO PLAN-TYPE OF I-HTH
MOVE EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)
TO EFFDT OF I-HTH

MOVE SPACE TO DEDUCTION-END-DT OF I-HTH

MOVE EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO COVERAGE-BEGIN-DT OF I-HTH

MOVE SPACE TO COVERAGE-END-DT OF I-HTH

SET COVERAGE-ELECT-ELECT OF I-HTH TO TRUE

MOVE PROCESS-DT OF W-CNTL TO COVERAGE-ELECT-DT OF I-HTH

MOVE BENEFIT-PLAN OF W-ADJCVG(WADJ-IDX)

TO BENEFIT-PLAN OF I-HTH

MOVE COVRG-CD OF W-ADJCVG(WADJ-IDX) TO COVRG-CD OF I-HTH

MOVE SPACE TO HIPAA-REPORT-DT OF I-HTH

SET PRETAX-CD-YES OF I-HTH TO TRUE

MOVE HLTH-PROVIDER-ID OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO HLTH-PROVIDER-ID OF I-HTH

MOVE PREVIOUSLY-SEEN OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO PREVIOUSLY-SEEN OF I-HTH

MOVE OTH-INSURANCE-IND OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO OTH-INSURANCE-IND OF I-HTH

MOVE OTH-INSURANCE-NAME OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO OTH-INSURANCE-NAME OF I-HTH

PERFORM XX200-INSERT-HEALTH-BENEFIT

.

ADJUST-HEALTH-BENEFIT-EXIT.

/*****

*

*

FM562-UPDATE-HEALTH-BENEFIT SECTION.

FM562.

*

*

INITIALIZE BIND-DATA OF U-XHTH

MOVE EMPLID OF W-EVENT TO EMPLID OF U-XHTH

MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF U-XHTH

MOVE COBRA-EVENT-ID OF W-CURELT TO COBRA-EVENT-ID OF U-XHTH

MOVE PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO PLAN-TYPE OF U-XHTH

MOVE EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO EFFDT OF U-XHTH

MOVE COVRG-CD OF W-ADJCVG(WADJ-IDX) TO COVRG-CD OF U-XHTH

PERFORM XX600-UPDATE-HEALTH-BENEFIT

.

UPDATE-HEALTH-BENEFIT-EXIT.

/*****

*

*

FM564-DELETE-HEALTH-DEPENDNT SECTION.

FM564.

*

*

SET WDEPEND-IDX TO 1

SEARCH WDEPEND-DATA

AT END

CONTINUE

WHEN WDEPEND-IDX > WDEPEND-COUNT

CONTINUE

WHEN OVERAGE-YES OF WDEPEND-DATA

OF W-DEPEND(WDEPEND-IDX)

INITIALIZE BIND-DATA OF D-XDEP

MOVE EMPLID OF W-EVENT TO EMPLID OF D-XDEP

MOVE BENEFIT-RCD-NO OF W-EVENT

TO EMPL-RCD-NO OF D-XDEP

MOVE COBRA-EVENT-ID OF W-CURELT

TO COBRA-EVENT-ID OF D-XDEP

MOVE PLAN-TYPE OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO PLAN-TYPE OF D-XDEP

MOVE EFFDT OF WADJ-DATA OF W-ADJCVG(WADJ-IDX)

TO EFFDT OF D-XDEP

MOVE DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

TO DEPENDENT-BENEF OF D-XDEP

PERFORM XX610-DELETE-HEALTH-DEPENDENT

SET RTNCD-OK OF SQLRT TO TRUE

END-SEARCH

DELETE-HEALTH-DEPENDNT-EXIT.

/*****

* *

FM570-ADJUST-HEALTH-DEPENDENT SECTION.

FM570.

* *

INITIALIZE BIND-DATA OF I-HTH-DEP

MOVE EMPLID OF W-EVENT TO EMPLID OF I-HTH-DEP

MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF I-HTH-DEP

MOVE COBRA-EVENT-ID OF W-CURELT

TO COBRA-EVENT-ID OF I-HTH-DEP

MOVE PLAN-TYPE OF WADJDB-DATA OF W-ADJCVG(WADJDB-IDX)

TO PLAN-TYPE OF I-HTH-DEP

MOVE EFFDT OF WADJDB-DATA OF W-ADJCVG(WADJDB-IDX)

TO EFFDT OF I-HTH-DEP

MOVE DEPENDENT-BENEF OF W-ADJCVG(WADJDB-IDX)

TO DEPENDENT-BENEF OF I-HTH-DEP

MOVE HLTH-PROVIDER-ID OF WADJDB-DATA OF W-ADJCVG(WADJDB-IDX)

TO HLTH-PROVIDER-ID OF I-HTH-DEP

MOVE PREVIOUSLY-SEEN OF WADJDB-DATA OF W-ADJCVG(WADJDB-IDX)

TO PREVIOUSLY-SEEN OF I-HTH-DEP
MOVE OTH-INSURANCE-IND OF WADJDB-DATA
OF W-ADJCVG(WADJDB-IDX)
TO OTH-INSURANCE-IND OF I-HTH-DEP
MOVE OTH-INSURANCE-NAME OF WADJDB-DATA
OF W-ADJCVG(WADJDB-IDX)
TO OTH-INSURANCE-NAME OF I-HTH-DEP

PERFORM XX300-INSERT-HEALTH-DEPENDENT

.

ADJUST-HEALTH-DEPENDENT-EXIT.

/*****

* *

FM600-STORE-OVERAGE-ACTIVITY SECTION.

FM600.

* *

PERFORM VARYING WDEPEND-IDX FROM 1 BY 1
UNTIL WDEPEND-IDX > WDEPEND-COUNT OF W-DEPEND

IF OVERAGE-YES OF W-DEPEND(WDEPEND-IDX)

MOVE OVERAGE-DT OF W-DEPEND(WDEPEND-IDX)

TO COBRA-EVENT-DT OF W-DEPEND(WDEPEND-IDX)

PERFORM XT000-UPDATE-DEP-BEN

END-IF

END-PERFORM

PERFORM VARYING WOVGACT-IDX FROM 1 BY 1

UNTIL WOVGACT-IDX > WOVGACT-COUNT OF W-OVGACT

PERFORM FM610-INSERT-COBRA-ACTIVITY

END-PERFORM

.

STORE-OVERAGE-EVENT-DATA-EXIT.

/*****

*

*

FM610-INSERT-COBRA-ACTIVITY SECTION.

FM610.

*

*

MOVE EMPLID OF W-EVENT TO EMPLID OF I-CBRACTY

MOVE EMPL-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF I-CBRACTY

MOVE COBRA-EVENT-DT OF W-OVGACT(WOVGACT-IDX)

```
        TO COBRA-EVENT-DT OF I-CBRACTY
MOVE COBRA-EVENT-CLASS OF W-CNTL
        TO COBRA-ACTION OF I-CBRACTY
MOVE CBR-ACTION-SOURCE OF W-CNTL
        TO CBR-ACTION-SOURCE OF I-CBRACTY
MOVE BENEFIT-RCD-NO OF W-EVENT
        TO BENEFIT-RCD-NO OF I-CBRACTY
MOVE SPACES TO SCHED-ID OF I-CBRACTY
MOVE ZERO TO EVENT-ID OF I-CBRACTY
SET CBR-COV-OVERRIDE-NO OF I-CBRACTY TO TRUE

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT
        SQLRT
        SQL-CURSOR OF I-CBRACTY OF W-CNTL
        SQL-STMT OF I-CBRACTY
        BIND-SETUP OF I-CBRACTY
        BIND-DATA OF I-CBRACTY
IF RTNCD-ERROR OF SQLRT

        IF RTNCD-DUPL OF SQLRT

                SET RTNCD-OK OF SQLRT TO TRUE
ELSE
        DISPLAY 'Emplid/EmplRcd#: '
                EMPLID OF I-CBRACTY '/'
                EMPL-RCD-NO OF I-CBRACTY
```

DISPLAY 'CBREvtDt/CBREvtCls: '

COBRA-EVENT-DT OF I-CBRACTY '/'

COBRA-ACTION OF I-CBRACTY

MOVE 'INSERT-COBRA-ACTIVITY'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

INSERT-COBRA-ACTIVITY-EXIT.

/*****

* *

FM700-COBRA-PARTIC-OVERAGE SECTION.

FM700.

* *

PERFORM FM413-GET-RELATED-EMPLID

PERFORM FM710-SETUP-RELATED-EMPL-DATA

PERFORM XP000-QUALIFY-DEPENDENT

IF QUALIFIED-YES OF W-DEPEND(WDEPEND-IDX)

IF (DISABLED-YES OF W-DEPEND(WDEPEFF-IDX)

AND EXCL-DISABLED-NO OF W-DEPRULE)

OR DISABLED-NO OF W-DEPEND(WDEPEFF-IDX)

MOVE CORR W-MIN-DEPRULE TO W-DEPRULE

MOVE PROCESS-DT OF W-CNTL TO AGE-AS-OF-DT OF W-CNTL

PERFORM XQ000-DEPENDENT-AGE-CHECK

MOVE OVERAGE-DT OF W-SAVE

TO OVERAGE-DT OF W-DEPEND(WDEPEND-IDX)

IF OVERAGE-S-YES OF W-SW

OR OVERAGE-NS-YES OF W-SW

SET OVERAGE-YES OF W-DEPEND(WDEPEND-IDX)

TO TRUE

MOVE OVERAGE-DT OF W-SAVE

TO OVERAGE-DT OF W-DEPEND(WDEPEND-IDX)

MOVE OVERAGE-DT OF W-SAVE

TO COBRA-EVENT-DT OF W-DEPEND(WDEPEND-IDX)

END-IF

IF OVERAGE-YES OF W-DEPEND(WDEPEND-IDX)

SET COBRA-ACTION-OVERAGE OF W-DEPEND(WDEPEND-IDX)

TO TRUE

PERFORM XS000-LOAD-OVERAGE-ACTIVITY

PERFORM XT000-UPDATE-DEP-BEN

PERFORM FM610-INSERT-COBRA-ACTIVITY

END-IF

END-IF

END-IF

.

COBRA-PARTIC-OVERAGE-EXIT.

/*****

* *

FM710-SETUP-RELATED-EMPL-DATA SECTION.

FM710.

* *

ADD 1 TO WDEPEND-COUNT OF W-DEPEND

SET WDEPEND-IDX TO WDEPEND-COUNT

INITIALIZE WDEPEND-DATA OF W-DEPEND(WDEPEND-IDX)

MOVE DEPENDENT-BENEF OF S-RELEMP

TO DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

MOVE BIRTHDATE OF S-RELEMP

TO BIRTHDATE OF W-DEPEND(WDEPEND-IDX)

MOVE COBRA-EMPLID OF S-RELEMP

TO COBRA-EMPLID OF W-DEPEND(WDEPEND-IDX)
SET COBRA-ACTION-NONE OF W-DEPEND(WDEPEND-IDX) TO TRUE

MOVE WDEPEFF-COUNT OF W-DEPEND
TO WDEPEFF-START OF W-DEPEND(WDEPEND-IDX)

ADD 1 TO WDEPEFF-START OF W-DEPEND(WDEPEND-IDX)

ADD 1 TO WDEPEFF-COUNT OF W-DEPEND
SET WDEPEFF-IDX TO WDEPEFF-COUNT
INITIALIZE WDEPEFF-DATA OF W-DEPEND(WDEPEFF-IDX)

MOVE DEPENDENT-BENEF OF S-RELEMP
TO DEPENDENT-BEN OF W-DEPEND(WDEPEFF-IDX)
MOVE COBRA-EVENT-DT OF W-EVENT
TO EFFDT OF W-DEPEND(WDEPEFF-IDX)
MOVE COVERED-PERSON-TYP OF S-RELEMP
TO COVERED-PERSON-TYP OF W-DEPEND(WDEPEFF-IDX)
MOVE DISABLED OF S-RELEMP
TO DISABLED OF W-DEPEND(WDEPEFF-IDX)

.

GET-ELATED-EMPL-DATA-EXIT.

/*****

*

*

FP000-FINISH-CURRENT-PGM SECTION.

FP000.

*

*

MOVE SPACE TO EMPLID OF W-CNTL

MOVE SPACE TO COBRA-EVENT-DT OF W-CNTL

MOVE SPACE TO DEPENDENT-BENEF OF W-CNTL

MOVE SPACE TO COBRA-ACTION OF W-CNTL

MOVE ZERO TO BENEFIT-RCD-NO OF W-CNTL

MOVE ZERO TO EMPL-RCD-NO OF W-CNTL

MOVE ZERO TO COBRA-EVENT-ID OF W-CNTL

PERFORM XD000-GET-ELAPSED-TIME

PERFORM UA000-REPORT-PROGRESS

PERFORM UD000-TAKE-CHECKPOINT

IF CURSOR-NORMAL OF SQLRT

PERFORM FG000-SELECT-ACTIVE-PGM

END-IF

MOVE ZERO TO PROGRESS-COUNT OF W-WK

.

FINISH-CURRENT-PGM-EXIT.

/*****

* *

FV000-DISC-OVERAGE-DEP SECTION.

FV000.

* *

PERFORM QA000-DISC-CURSORS

IF SQL-CURSOR OF S-BENPGM OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-BENPGM OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-OVERAGE-DEP(S-BENPGM)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-EE-PGM OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
SQLRT
SQL-CURSOR OF S-EE-PGM OF W-CNTL
IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-OVERAGE-DEP(S-EE-PGM)'
TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF
END-IF

IF SQL-CURSOR OF I-CBRCTY OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
SQLRT
SQL-CURSOR OF I-CBRCTY OF W-CNTL
IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-OVERAGE-DEP(I-CBRCTY)'
TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF
END-IF

.

DISC-OVERAGE-DEP-EXIT.

/*****

* *

FY000-TERM-OVERAGE-DEP SECTION.

FY000.

* *

SET COBRA-PHASE-ACTIVITY OF W-CNTL TO TRUE

PERFORM TG000-FINISH-RUN-UNIT

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'Overage Dependent Processing Ended at '
TIME-OUT OF W-WK '.'

TERM-OVERAGE-DEP-EXIT.

/*****

* *

GA000-PROCESS-ACTIVITY SECTION.

GA000.

* *

PERFORM GD000-INIT-PROCESS-ACTIVITY

PERFORM GG000-SELECT-COBRA-ACTIVITY

PERFORM GJ000-FETCH-COBRA-ACTIVITY

MOVE HIGH-VALUE TO WCBR-EVT-MAX-EFFDT OF W-CBR-EVT

PERFORM UNTIL RTNCD-END OF SQLRT

MOVE EMPLID OF SELECT-DATA OF S-CBRACY

TO EMPLID OF W-EVENT

MOVE EMPL-RCD-NO OF SELECT-DATA OF S-CBRACY

TO EMPL-RCD-NO OF W-EVENT

MOVE BENEFIT-RCD-NO OF SELECT-DATA OF S-CBRACY

TO BENEFIT-RCD-NO OF W-EVENT

IF COBRA-EVENT-DT OF W-EVENT

< WCBR-EVT-MAX-EFFDT OF W-CBR-EVT

PERFORM GA100-LOAD-CBR-EVT-RULE

END-IF

PERFORM GM000-PROCESS-COBRA-ACTIVITY

MOVE EMPLID OF SELECT-DATA OF S-CBRACY

TO EMPLID OF W-CNTL

MOVE EMPL-RCD-NO OF SELECT-DATA OF S-CBRCTY

TO EMPL-RCD-NO OF W-CNTL

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-CBRCTY

TO COBRA-EVENT-DT OF W-CNTL

MOVE COBRA-ACTION OF SELECT-DATA OF S-CBRCTY

TO COBRA-ACTION OF W-CNTL

PERFORM TD000-CHECKPOINT-RUN-UNIT

IF RESELECT-YES OF W-SW

PERFORM GG000-SELECT-COBRA-ACTIVITY

END-IF

PERFORM GJ000-FETCH-COBRA-ACTIVITY

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

PERFORM GV000-DISC-PROCESS-ACTIVITY

PERFORM GY000-TERM-PROCESS-ACTIVITY

.

PROCESS-COBRA-ACTIVITY-EXIT.

/*****

* *

GA100-LOAD-CBR-EVT-RULE SECTION.

GA100.

* *

MOVE ZERO TO WCBR-EVT-COUNT OF W-CBR-EVT

MOVE ZERO TO WCBR-BEN-COUNT OF W-CBR-EVT

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-CBR-ACTY
TO COBRA-EVENT-DT OF W-EVENT

PERFORM XF100-SELECT-CBR-EVT-RULES

PERFORM XF200-FETCH-CBR-EVT-RULES

PERFORM UNTIL RTNCD-END OF SQLRT

IF WCBR-EVNT-COUNT-MAX OF W-CBR-EVT

DISPLAY 'Max # of COBRA Event Rules Exceeded'

MOVE 'LOAD-CBR-EVT-RULE' TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO WCBR-EVT-COUNT OF W-CBR-EVT

PERFORM XF300-SAVE-CBR-EVT-RULES

PERFORM XF200-FETCH-CBR-EVT-RULES

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

LOAD-CBR-EVT-RULE-EXIT.

/*****

* *

GD000-INIT-PROCESS-ACTIVITY SECTION.

GD000.

* *

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'COBRA Activity Processing Started at '

TIME-OUT OF W-WK '.

SET COBRA-PHASE-ACTIVITY OF W-CNTL TO TRUE

SET PROGRESS-REPORT-ACTIVITY OF W-WK TO TRUE

PERFORM TA000-START-RUN-UNIT

.

INIT-PROCESS-ACTIVITY-EXIT.

/*****

*

*

GG000-SELECT-COBRA-ACTIVITY SECTION.

GG000.

*

*

MOVE EMPLID OF W-CNTL TO EMPLID OF BIND-DATA OF S-CBRACTY

MOVE EMPLID OF W-CNTL TO EMPLID-2 OF S-CBRACTY

EMPLID-3 OF S-CBRACTY

EMPLID-4 OF S-CBRACTY

MOVE EMPL-RCD-NO OF W-CNTL

TO EMPL-RCD-NO OF BIND-DATA OF S-CBRACTY

MOVE EMPL-RCD-NO OF W-CNTL

TO EMPL-RCD-NO-2 OF S-CBRACTY

MOVE COBRA-EVENT-DT OF W-CNTL

TO COBRA-EVENT-DT OF BIND-DATA OF S-CBRACTY

MOVE COBRA-EVENT-DT OF W-CNTL

TO COBRA-EVENT-DT-2 OF S-CBRACTY

MOVE COBRA-ACTION OF W-CNTL

TO COBRA-ACTION OF BIND-DATA OF S-CBRACTY

COBRA-ACTION-2 OF BIND-DATA OF S-CBRACTY

COBRA-ACTION-3 OF BIND-DATA OF S-CBRACTY

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-CBRACTY OF W-CNTL

SQL-STMT OF S-CBRACTY

BIND-SETUP OF S-CBRACTY

BIND-DATA OF S-CBRACTY

SELECT-SETUP OF S-CBRACTY

SELECT-DATA OF S-CBRACTY

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-COBRA-ACTIVITY' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-COBRA-ACTIVITY-EXIT.

/*****

* *

GJ000-FETCH-COBRA-ACTIVITY SECTION.

GJ000.

* *

INITIALIZE SELECT-DATA OF S-CBRACTY

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-CBRACTY OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-COBRA-ACTIVITY' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-COBRA-ACTIVITY-EXIT.

/*****

* *

GM000-PROCESS-COBRA-ACTIVITY SECTION.

GM000.

* *

MOVE COBRA-ACTION OF SELECT-DATA OF S-CBRACTY

TO COBRA-EVENT-CLASS OF W-CNTL

PERFORM XH000-GET-CBR-EVT-RULE

IF EFF-STATUS-ACTIVE OF W-CBR-EVT(WCBR-EVT-IDX)

IF EMPLID OF SELECT-DATA OF S-CBRACTY

> EMPLID OF W-CNTL

PERFORM GM100-GET-LAST-COBRA-EVENTID

END-IF

IF (CBR-SOURCE-PERS-DATA-CHG OF S-CBRACTY

OR CBR-SOURCE-DEP-CHG OF S-CBRACTY

OR CBR-SOURCE-MANUAL-EVENT OF S-CBRACTY)

AND MULTIJOB-YES OF W-CNTL

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF S-MJOB

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-CBRACTY

TO EFFDT OF BIND-DATA OF S-MJOB

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-CBRACTY

TO EFFDT-2 OF BIND-DATA OF S-MJOB

PERFORM GM200-SELECT-MULTI-JOB

PERFORM GM300-FETCH-MULTI-JOB

PERFORM UNTIL RTNCD-END OF SQLRT

MOVE EMPL-RCD-NO OF S-MJOB

```
        TO EMPL-RCD-NO OF W-EVENT
MOVE BENEFIT-RCD-NO OF S-MJOB
        TO BENEFIT-RCD-NO OF W-EVENT
SET DUPLICATE-EVENT-NO OF W-SW TO TRUE
PERFORM GM400-CHECK-DUPLICATE-EVENT

IF DUPLICATE-EVENT-NO OF W-SW

        PERFORM GM600-INIT-EVENT
        PERFORM XW000-STORE-COBRA-DATA
END-IF

PERFORM GM300-FETCH-MULTI-JOB
END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE
ELSE

SET DUPLICATE-EVENT-NO OF W-SW TO TRUE
PERFORM GM400-CHECK-DUPLICATE-EVENT

SET REG-REGION-USA OF W-SW TO TRUE
IF (CBR-SOURCE-PERS-DATA-CHG OF S-CBRACTY
    OR CBR-SOURCE-DEP-CHG OF S-CBRACTY
    OR CBR-SOURCE-MANUAL-EVENT OF S-CBRACTY)
        PERFORM GM500-CHECK-REG-REGION
END-IF
```

IF DUPLICATE-EVENT-NO OF W-SW AND

REG-REGION-USA OF W-SW

PERFORM GM600-INIT-EVENT

PERFORM XW000-STORE-COBRA-DATA

END-IF

END-IF

END-IF

PERFORM GM700-DELETE-COBRA-ACTIVITY

.

PROCESS-COBRA-ACTIVITY-EXIT.

/*****

* *

GM100-GET-LAST-COBRA-EVENTID SECTION.

GM100.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF S-EVENTID

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-EVENTID

SQL-STMT OF S-EVENTID

BIND-SETUP OF S-EVENTID

BIND-DATA OF S-EVENTID

SELECT-SETUP OF S-EVENTID

SELECT-DATA OF S-EVENTID

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-LAST-EVENT-ID(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-EVENTID

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-EVENTID

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

MOVE ZERO TO COBRA-EVENT-ID OF S-EVENTID

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

DISPLAY 'Emplid: ' EMPLID OF S-EVENTID

MOVE 'GET-LAST-EVENT-ID(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

GET-LAST-COBRA-EVENTID-EXIT.

/*****

* *

GM200-SELECT-MULTI-JOB SECTION.

GM200.

* *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-MJOB

SQL-STMT OF S-MJOB

BIND-SETUP OF S-MJOB

BIND-DATA OF S-MJOB

SELECT-SETUP OF S-MJOB

SELECT-DATA OF S-MJOB

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-MULTI-JOB' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-JOB-EXIT.

/*****

* *

GM300-FETCH-MULTI-JOB SECTION.

GM300.

* *

INITIALIZE SELECT-DATA OF S-MJOB

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-MJOB

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

DISPLAY 'Emplid/EvtDt: '

EMPLID OF BIND-DATA OF S-MJOB '/'

EFFDT OF BIND-DATA OF S-MJOB

MOVE 'FETCH-JOB' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-JOB-EXIT.

/*****

* *

GM400-CHECK-DUPLICATE-EVENT SECTION.

GM400.

* *

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF S-EVTDUPL

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF BIND-DATA OF S-EVTDUPL

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-CBRACY

TO COBRA-EVENT-DT OF S-EVTDUPL

MOVE COBRA-ACTION OF SELECT-DATA OF S-CBRACY

TO COBRA-EVENT-CLASS OF S-EVTDUPL

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-EVTDUPL

BIND-SETUP OF S-EVTDUPL

BIND-DATA OF S-EVTDUPL

SELECT-SETUP OF S-EVTDUPL

SELECT-DATA OF S-EVTDUPL

IF RTNCD-ERROR OF SQLRT

MOVE 'CHECK-DUPLICATE-EVENT(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-EVTDUPL

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

DISPLAY 'Emplid: ' EMPLID OF BIND-DATA OF S-EVTDUPL

MOVE 'CHECK-DUPLICATE-EVENT(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

SET DUPLICATE-EVENT-YES OF W-SW TO TRUE

SET MSGID-DUPL-COBRA-EVENT OF PYMSG TO TRUE

MOVE ZERO TO COBRA-EVENT-ID OF PYMSG

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-CBRACY

TO MSGDATA1 OF PYMSG

MOVE COBRA-ACTION OF SELECT-DATA OF S-CBRACY

TO MSGDATA2 OF PYMSG

MOVE EMPL-RCD-NO OF SELECT-DATA OF S-CBRACY

TO MSGDATA3 OF PYMSG

PERFORM ZM000-MESSAGE

END-IF

.

CHECK-DUPLICATE-EVENT-EXIT.

/*****

* * *

GM500-CHECK-REG-REGION SECTION.

GM500.

* * *

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF S-REGION

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-CBRACY

TO EFFDT OF S-REGION

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-CBRACY

TO EFFDT-2 OF S-REGION

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-REGION

BIND-SETUP OF S-REGION

BIND-DATA OF S-REGION

SELECT-SETUP OF S-REGION

SELECT-DATA OF S-REGION

IF RTNCD-ERROR OF SQLRT

MOVE 'CHECK-REG-REGION(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-REGION

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

SET REG-REGION-NOT-USA OF W-SW TO TRUE

ELSE

MOVE 'CHECK-REG-REGION(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

CHECK-REG-REGION-EXIT.

/*****

* *

GM600-INIT-EVENT SECTION.

GM600.

* *

ADD 1 TO COBRA-EVENT-ID OF S-EVENTID
MOVE COBRA-EVENT-ID OF S-EVENTID
TO COBRA-EVENT-ID OF W-EVENT
MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-CBRACY
TO COBRA-EVENT-DT OF W-EVENT
MOVE COBRA-ACTION OF SELECT-DATA OF S-CBRACY
TO COBRA-EVENT-CLASS OF W-EVENT
MOVE SCHED-ID OF S-CBRACY
TO SCHED-ID OF W-EVENT
MOVE EVENT-ID OF S-CBRACY
TO EVENT-ID OF W-EVENT
MOVE CBR-ACTION-SOURCE OF S-CBRACY
TO CBR-ACTION-SOURCE OF W-EVENT
MOVE CBR-COV-OVERRIDE OF S-CBRACY
TO CBR-COV-OVERRIDE OF W-EVENT
MOVE CBR-ALT-COV-TRM-DT OF S-CBRACY
TO CBR-ALT-COV-TRM-DT OF W-EVENT
MOVE CBR-COV-AS-OF-DT OF S-CBRACY
TO CBR-COV-AS-OF-DT OF W-EVENT
SET CBR-QUALIFY-UNPROCESSED OF WEVENT-DATA OF W-EVENT
TO TRUE
SET CBR-PROCESS-OPEN OF WEVENT-DATA OF W-EVENT TO TRUE
SET CBR-REPROCESS-NONE OF WEVENT-DATA OF W-EVENT TO TRUE
SET CBR-REPROCESS-NONE OF W-SW TO TRUE
SET CBR-EVT-CONFLICT-NO OF W-EVENT TO TRUE

SET BAS-DATA-CHG-NO OF W-EVENT TO TRUE
SET INSERT-YES OF WEVENT-DATA OF W-EVENT TO TRUE
SET UPDATE-NO OF WEVENT-DATA OF W-EVENT TO TRUE
MOVE ZERO TO WPARTIC-COUNT OF W-EVENT
MOVE ZERO TO WPLAN-COUNT OF W-EVENT
MOVE ZERO TO WOPTN-COUNT OF W-EVENT
MOVE ZERO TO WDPND-COUNT OF W-EVENT

.
INIT-EVENT-EXIT.

/*****

* *

GM700-DELETE-COBRA-ACTIVITY SECTION.

GM700.

* *

MOVE EMPLID OF SELECT-DATA OF S-CBRACY
TO EMPLID OF D-CBRACY
MOVE EMPL-RCD-NO OF SELECT-DATA OF S-CBRACY
TO EMPL-RCD-NO OF D-CBRACY
MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-CBRACY
TO COBRA-EVENT-DT OF D-CBRACY
MOVE COBRA-ACTION OF SELECT-DATA OF S-CBRACY

TO COBRA-ACTION OF D-CBRACTY

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF D-CBRACTY

BIND-SETUP OF D-CBRACTY

BIND-DATA OF D-CBRACTY

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid/EmplRcd#: '

EMPLID OF D-CBRACTY '/'

EMPL-RCD-NO OF D-CBRACTY

DISPLAY 'CBREvtDt/CBRAct: '

COBRA-EVENT-DT OF D-CBRACTY '/'

COBRA-ACTION OF D-CBRACTY

MOVE 'DELETE-COBRA-ACTIVITY' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-COBRA-ACTIVITY-EXIT.

/*****

*

*

GV000-DISC-PROCESS-ACTIVITY SECTION.

GV000.

*

*

PERFORM QA000-DISC-CURSORS

IF SQL-CURSOR OF S-CBRACTY OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-CBRACTY OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-PROCESS-ACTIVITY(S-CBRACTY)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-EVENTID NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-EVENTID

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-PROCESS-ACTIVITY(S-EVENTID)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-MJOB NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-MJOB

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-PROCESS-ACTIVITY(S-MJOB)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF I-EVT OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF I-EVT OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-PROCESS-ACTIVITY(I-EVT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF D-CBRCTY OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF D-CBRCTY OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-PROCESS-ACTIVITY(D-CBRCTY)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

DISC-PROCESS-ACTIVITY-EXIT.

/*****

*

*

GY000-TERM-PROCESS-ACTIVITY SECTION.

GY000.

*

*

SET COBRA-PHASE-QUALIFY OF W-CNTL TO TRUE

PERFORM TG000-FINISH-RUN-UNIT

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'COBRA Activity Processing Ended at '

TIME-OUT OF W-WK '.'

.

TERM-PROCESS-ACTIVITY-EXIT.

/*****

*

*

HA000-QUALIFY-EVENT SECTION.

HA000.

*

*

PERFORM HD000-INIT-QUALIFY-EVENT

PERFORM HG000-SELECT-EVENT

PERFORM HJ000-FETCH-EVENT

PERFORM UNTIL RTNCD-END OF SQLRT

PERFORM HM000-SAVE-EVENT

IF EMPLID OF W-EVENT NOT = EMPLID OF W-CNTL

OR COBRA-EVENT-DT OF W-EVENT

NOT = WDEPEND-ASOFDT OF W-DEPEND

SET CBR-QUALIFY-ERROR-NO OF W-SW TO TRUE

MOVE ZERO TO WDEPEND-COUNT OF W-DEPEND

MOVE ZERO TO WDEPEFF-COUNT OF W-DEPEND

MOVE ZERO TO WDEPNID-COUNT OF W-DEPNID

SET LOAD-EMPL-DEP-YES OF W-SW TO TRUE

PERFORM XN000-LOAD-DEPENDENT

END-IF

PERFORM HP000-QUALIFY-COBRA-EVENT

PERFORM XW000-STORE-COBRA-DATA

MOVE EMPLID OF SELECT-DATA OF S-EVT TO EMPLID OF W-CNTL

MOVE BENEFIT-RCD-NO OF SELECT-DATA OF S-EVT

TO BENEFIT-RCD-NO OF W-CNTL

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-EVT

TO COBRA-EVENT-DT OF W-CNTL

PERFORM TD000-CHECKPOINT-RUN-UNIT

IF RESELECT-YES OF W-SW

PERFORM HG000-SELECT-EVENT

END-IF

PERFORM HJ000-FETCH-EVENT

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

PERFORM HV000-DISC-QUALIFY-EVENT

PERFORM HY000-TERM-QUALIFY-EVENT

.

QUALIFY-EVENT-EXIT.

/*****

* *

HD000-INIT-QUALIFY-EVENT SECTION.

HD000.

* *

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'Qualify Event Processing Started at '

TIME-OUT OF W-WK '.'

SET COBRA-PHASE-QUALIFY OF W-CNTL TO TRUE

SET PROGRESS-REPORT-QUALIFY OF W-WK TO TRUE

PERFORM TA000-START-RUN-UNIT

.

INIT-QUALIFY-EVENT-EXIT.

/*****

* *

HG000-SELECT-EVENT SECTION.

HG000.

* *

MOVE EMPLID OF W-CNTL TO EMPLID OF BIND-DATA OF S-EVT

MOVE EMPLID OF W-CNTL TO EMPLID-2 OF S-EVT

EMPLID-3 OF BIND-DATA OF S-EVT

EMPLID-4 OF BIND-DATA OF S-EVT

MOVE BENEFIT-RCD-NO OF W-CNTL

TO BENEFIT-RCD-NO OF BIND-DATA OF S-EVT

MOVE BENEFIT-RCD-NO OF W-CNTL TO BENEFIT-RCD-NO-2 OF S-EVT

MOVE COBRA-EVENT-DT OF W-CNTL

TO COBRA-EVENT-DT OF BIND-DATA OF S-EVT

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-EVT OF W-CNTL

SQL-STMT OF S-EVT

BIND-SETUP OF S-EVT

BIND-DATA OF S-EVT

SELECT-SETUP OF S-EVT

SELECT-DATA OF S-EVT

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-EVENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-EVENT-EXIT.

/*****

* *

HJ000-FETCH-EVENT SECTION.

HJ000.

* *

INITIALIZE SELECT-DATA OF S-EVT

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-EVT OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-EVENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-EVENT-EXIT.

/*****

* *

HM000-SAVE-EVENT SECTION.

HM000.

* *

INITIALIZE W-EVENT

SET INSERT-NO OF WEVENT-DATA OF W-EVENT TO TRUE

SET UPDATE-NO OF WEVENT-DATA OF W-EVENT TO TRUE

MOVE EMPLID OF SELECT-DATA OF S-EVT TO EMPLID OF W-EVENT

MOVE EMPL-RCD-NO OF SELECT-DATA OF S-EVT

TO EMPL-RCD-NO OF W-EVENT

MOVE BENEFIT-RCD-NO OF SELECT-DATA OF S-EVT

TO BENEFIT-RCD-NO OF W-EVENT

MOVE COBRA-EVENT-ID OF SELECT-DATA OF S-EVT

TO COBRA-EVENT-ID OF W-EVENT

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-EVT

TO COBRA-EVENT-DT OF W-EVENT

MOVE COBRA-EVENT-CLASS OF S-EVT

TO COBRA-EVENT-CLASS OF W-EVENT

MOVE COBRA-EVENT-CLASS OF S-EVT

TO COBRA-EVENT-CLASS OF W-CNTL

MOVE SCHED-ID OF S-EVT TO SCHED-ID OF W-EVENT

MOVE EVENT-ID OF S-EVT TO EVENT-ID OF W-EVENT

MOVE CBR-COV-OVERRIDE OF S-EVT

TO CBR-COV-OVERRIDE OF W-EVENT

MOVE CBR-ALT-COV-TRM-DT OF S-EVT

TO CBR-ALT-COV-TRM-DT OF W-EVENT

MOVE CBR-COV-AS-OF-DT OF S-EVT

TO CBR-COV-AS-OF-DT OF W-EVENT

MOVE CBR-QUALIFY-STATUS OF S-EVT

TO CBR-QUALIFY-STATUS OF WEVENT-DATA OF W-EVENT

MOVE CBR-PROCESS-STATUS OF S-EVT

TO CBR-PROCESS-STATUS OF WEVENT-DATA OF W-EVENT

MOVE CBR-EVT-REPRCS-IND OF S-EVT

TO CBR-EVT-REPRCS-IND OF W-EVENT
MOVE CBR-EVENT-CONFLICT OF S-EVT
TO CBR-EVENT-CONFLICT OF W-EVENT
MOVE BAS-DATA-CHG OF S-EVT
TO BAS-DATA-CHG OF W-EVENT

MOVE ZERO TO WPARTIC-COUNT OF W-EVENT
MOVE ZERO TO WPLAN-COUNT OF W-EVENT
MOVE ZERO TO WOPTN-COUNT OF W-EVENT
MOVE ZERO TO WDPND-COUNT OF W-EVENT
MOVE ZERO TO WCUR-COUNT OF W-CURELT
MOVE ZERO TO WCURDB-COUNT OF W-CURELT
SET CBR-QUALIFY-PENDING-NO OF W-SW TO TRUE
SET CBR-QUALIFY-NO OF W-SW TO TRUE

.

SAVE-EVENT-EXIT.

/*****

* *

HP000-QUALIFY-COBRA-EVENT SECTION.

HP000.

* *

PERFORM HP100-CHECK-EVENT-CONFLICT

PERFORM XF000-LOAD-CBR-EVT-RULES

PERFORM XH000-GET-CBR-EVT-RULE

PERFORM HP200-GET-CURRENT-JOB

PERFORM XM000-OBTAIN-CUR-ELECT-DATA

PERFORM HP300-LOAD-COBRA-PLAN-DATA

PERFORM VARYING WDEPEND-IDX FROM 1 BY 1

UNTIL WDEPEND-IDX > WDEPEND-COUNT OF W-DEPEND

MOVE COBRA-EVENT-DT OF W-EVENT

TO SEARCH-EFFDT OF W-DEPEND

PERFORM XN700-SRCH-DEP-EFF-DATA

PERFORM XP000-QUALIFY-DEPENDENT

IF QUALIFIED-YES OF W-DEPEND(WDEPEND-IDX)

PERFORM HP400-INIT-PARTIC

PERFORM VARYING WCBRPLN-IDX FROM 1 BY 1

UNTIL WCBRPLN-IDX

> WCBRPLN-COUNT OF W-CBRDEFN

IF (PLAN-TYPE-FSA OF W-CBRDEFN(WCBRPLN-IDX)

AND EMPLOYEE OF W-DEPEND(WDEPEFF-IDX))

OR PLAN-TYPE-HEALTH

OF W-CBRDEFN(WCBRPLN-IDX)

PERFORM HP500-INIT-PARTIC-PLAN

PERFORM HP600-QUALIFY-PARTIC-PLAN

END-IF

END-PERFORM

PERFORM HP700-QUALIFY-PARTIC

IF INIT-PLAN-COUNT OF W-COUNT > ZERO

AND BENADMIN-YES OF W-CNTL

PERFORM HP800-ASSIGN-BENEFIT-PROGRAM

END-IF

SET CBR-PROCESS-CLOSED OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX) TO TRUE

SET WPLAN-IDX TO 1

SEARCH WPLAN-DATA

WHEN DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

= DEPENDENT-BENEF OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

CONTINUE

END-SEARCH

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

IF CBR-PROCESS-OPEN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

SET CBR-PROCESS-OPEN OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX) TO TRUE

END-IF

END-PERFORM

END-IF

END-PERFORM

IF CBR-QUALIFY-ERROR-YES OF W-SW

SET CBR-QUALIFY-ERROR OF WEVENT-DATA OF W-EVENT TO TRUE

ELSE

IF CBR-QUALIFY-PENDING-YES OF W-SW

SET CBR-QUALIFY-PENDING OF WEVENT-DATA OF W-EVENT

TO TRUE
ELSE

IF CBR-QUALIFY-YES OF W-SW

SET CBR-QUALIFIED OF WEVENT-DATA OF W-EVENT
TO TRUE

ELSE

SET CBR-NOT-QUALIFIED OF WEVENT-DATA OF W-EVENT
TO TRUE

SET CBR-PROCESS-CLOSED OF WEVENT-DATA OF W-EVENT
TO TRUE

END-IF

END-IF

END-IF

SET UPDATE-YES OF WEVENT-DATA OF W-EVENT TO TRUE

.

QUALIFY-COBRA-EVENT-EXIT.

/*****

* *

HP100-CHECK-EVENT-CONFLICT SECTION.

HP100.

*

*

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF S-EVTCONF

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF BIND-DATA OF S-EVTCONF

MOVE COBRA-EVENT-DT OF W-EVENT

TO COBRA-EVENT-DT OF BIND-DATA OF S-EVTCONF

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-EVTCONF

BIND-SETUP OF S-EVTCONF

BIND-DATA OF S-EVTCONF

SELECT-SETUP OF S-EVTCONF

SELECT-DATA OF S-EVTCONF

IF RTNCD-ERROR OF SQLRT

MOVE 'CHECK-EVENT-CONFLICT(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-EVTCONF

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

DISPLAY 'Emplid: ' EMPLID OF BIND-DATA OF S-EVTCONF

MOVE 'CHECK-EVENT-CONFLICT(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

SET MSGID-EVENT-COBRA-CONFLICT OF PYMSG TO TRUE

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF PYMSG

MOVE COBRA-EVENT-DT OF WEVENT-DATA OF W-EVENT

TO MSGDATA1 OF PYMSG

MOVE COBRA-EVENT-ID OF S-EVTCONF

TO MSGDATA2 OF PYMSG

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-EVTCONF

TO MSGDATA3 OF PYMSG

PERFORM ZM000-MESSAGE

SET CBR-EVT-CONFLICT-YES OF W-EVENT TO TRUE
PERFORM HP110-UPDATE-EVENT-CONFLICT
END-IF

CHECK-EVENT-CONFLICT-EXIT.

/******

* *

HP110-UPDATE-EVENT-CONFLICT SECTION.

HP110.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF U-EVTCONF

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF U-EVTCONF

MOVE COBRA-EVENT-ID OF S-EVTCONF

TO COBRA-EVENT-ID OF U-EVTCONF

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF U-EVTCONF

BIND-SETUP OF U-EVTCONF

BIND-DATA OF U-EVTCONF

IF RTNCD=ERROR OF SQLRT

DISPLAY 'Emplid: ' EMPLID OF U-EVTCONF

MOVE 'UPDATE-EVENT-CONFLICT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

UPDATE-EVENT-CONFLICT-EXIT.

/*****

* *

HP200-GET-CURRENT-JOB SECTION.

HP200.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF S-CURJOB

MOVE EMPL-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF S-CURJOB

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT OF BIND-DATA OF S-CURJOB

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT-2 OF BIND-DATA OF S-CURJOB

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-CURJOB OF W-CNTL

SQL-STMT OF S-CURJOB

BIND-SETUP OF S-CURJOB

BIND-DATA OF S-CURJOB

SELECT-SETUP OF S-CURJOB

SELECT-DATA OF S-CURJOB

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-CURRENT-JOB(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-CURJOB

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-CURJOB OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

DISPLAY 'Emplid/EmplRcd#/CBREvtDt: '

EMPLID OF S-CURJOB '/'

EMPL-RCD-NO OF S-CURJOB '/'

EFFDT OF BIND-DATA OF S-CURJOB

MOVE 'GET-CURRENT-JOB(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

MOVE COMPANY OF S-CURJOB TO COMPANY OF W-EVENT

MOVE PAYGROUP OF S-CURJOB TO PAYGROUP OF W-EVENT

.

GET-CURRENT-JOB-EXIT.

/*****

*

*

HP300-LOAD-COBRA-PLAN-DATA SECTION.

HP300.

*

*

PERFORM HP310-LOAD-PROGRAM-DATA

MOVE ZERO TO WCBRPLN-COUNT OF W-CBRDEFN

PERFORM HP320-SELECT-CBRPLN

PERFORM HP330-FETCH-CBRPLN

PERFORM UNTIL RTNCD-END OF SQLRT

ADD 1 TO WCBRPLN-COUNT OF W-CBRDEFN

PERFORM HP340-SAVE-CBRPLN

PERFORM HP330-FETCH-CBRPLN

END-PERFORM

PERFORM HP350-SELECT-PRREVT

PERFORM HP360-FETCH-PRREVT

PERFORM UNTIL RTNCD-END OF SQLRT

SET WCBRPLN-IDX TO 1

SEARCH WCBRPLN-DATA

WHEN WCBRPLN-IDX > WCBRPLN-COUNT OF W-CBRDEFN

ADD 1 TO WCBRPLN-COUNT OF W-CBRDEFN

PERFORM HP370-SAVE-SECONDARY-CBRPLN

WHEN PLAN-TYPE OF S-PRREVT

= PLAN-TYPE OF W-CBRDEFN(WCBRPLN-IDX)

CONTINUE

END-SEARCH

PERFORM HP360-FETCH-PRREVT

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

LOAD-COBRA-PLAN-EXIT.

/*****

* *

HP310-LOAD-PROGRAM-DATA SECTION.

HP310.

* *

SET SQL-STMT-BENPGM OF S-BENPGM TO TRUE

MOVE BENEFIT-PROGRAM OF W-CURELT

TO BENEFIT-PROGRAM OF BIND-DATA OF S-BENPGM

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT OF BIND-DATA OF S-BENPGM

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT-2 OF BIND-DATA OF S-BENPGM

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-BENPGM

BIND-SETUP OF S-BENPGM

BIND-DATA OF S-BENPGM

SELECT-SETUP OF S-BENPGM

SELECT-DATA OF S-BENPGM

IF RTNCD-ERROR OF SQLRT

MOVE 'LOAD-PROGRAM-DATA(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-BENPGM

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'LOAD-PROGRAM-DATA(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

MOVE BENEFIT-PROGRAM OF SELECT-DATA OF S-BENPGM

TO BENEFIT-PROGRAM OF W-CBRDEFN

MOVE EFFDT OF SELECT-DATA OF S-BENPGM TO EFFDT OF W-CBRDEFN

MOVE COBRA-SURCHARGE OF S-BENPGM

TO COBRA-SURCHARGE OF W-CBRDEFN

MOVE COBRA-DISABL-SURCG OF SELECT-DATA OF S-BENPGM

TO COBRA-DISABL-SURCG OF W-CBRDEFN

/* LOAD PLAN DEFN TO BE USED EXTRACT DEP RULE DATA PER PLAN TYPE

/* IN OVERAGE PROCESS

IF COBRA-EVENT-OVERAGE OF W-EVENT

MOVE BENEFIT-PROGRAM OF W-CURELT

TO BENEFIT-PROGRAM OF BATBL

MOVE PROCESS-DT OF W-CNTL TO EVENT-DT OF BATBL

SET ACTION-LOAD-PGM-DEFN OF BATBL TO TRUE

PERFORM RA000-TABLE-ACCESS

END-IF

LOAD-PROGRAM-DATA-EXIT.

/*****

* *

HP320-SELECT-CBRPLN SECTION.

HP320.

* *

MOVE BENEFIT-PROGRAM OF W-CBRDEFN

TO BENEFIT-PROGRAM OF S-CBRPLN

MOVE EFFDT OF W-CBRDEFN TO EFFDT OF S-CBRPLN

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-CBRPLN

BIND-SETUP OF S-CBRPLN

BIND-DATA OF S-CBRPLN

SELECT-SETUP OF S-CBRPLN

SELECT-DATA OF S-CBRPLN

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-CBRPLN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-CBRPLN-EXIT.

/*****

* *

HP330-FETCH-CBRPLN SECTION.

HP330.

* *

*****/

INITIALIZE SELECT-DATA OF S-CBRPLN

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-CBRPLN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.
FETCH-CBRPLN-EXIT.

/*****

* *

HP340-SAVE-CBRPLN SECTION.

HP340.

* *

SET WCBRPLN-IDX TO WCBRPLN-COUNT OF W-CBRDEFN

INITIALIZE WCBRPLN-DATA OF W-CBRDEFN(WCBRPLN-IDX)

MOVE PLAN-TYPE OF SELECT-DATA OF S-CBRPLN

TO PLAN-TYPE OF W-CBRDEFN(WCBRPLN-IDX)

MOVE EVENT-RULES-ID OF SELECT-DATA OF S-CBRPLN

TO EVENT-RULES-ID OF W-CBRDEFN(WCBRPLN-IDX)

MOVE DEP-RULE-ID OF SELECT-DATA OF S-CBRPLN

TO DEP-RULE-ID OF W-CBRDEFN(WCBRPLN-IDX)

SET CHECK-EMPL-TERM-YES OF W-CBRDEFN(WCBRPLN-IDX) TO TRUE

SET EMPL-COVRG-TERM-NO OF W-CBRDEFN(WCBRPLN-IDX) TO TRUE

SET SECONDARY-ONLY-NO OF W-CBRDEFN(WCBRPLN-IDX) TO TRUE

.
SAVE-CBRPLN-EXIT.

/*****

*

*

HP350-SELECT-PRREVT SECTION.

HP350.

*

*

MOVE EMPLID OF W-EVENT TO EMPLID OF S-PRREVT

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF S-PRREVT

MOVE COBRA-EVENT-DT OF W-EVENT

TO COBRA-EVENT-DT OF S-PRREVT

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-PRREVT

BIND-SETUP OF S-PRREVT

BIND-DATA OF S-PRREVT

SELECT-SETUP OF S-PRREVT

SELECT-DATA OF S-PRREVT

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-PRREVT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-PRREVT-EXIT.

/*****

* *

HP360-FETCH-PRREVT SECTION.

HP360.

* *

INITIALIZE SELECT-DATA OF S-PRREVT

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

DISPLAY 'Emplid/BenRcd#/CBREvtDt: '

EMPLID OF S-PRREVT '/'

BENEFIT-RCD-NO OF S-PRREVT '/'

COBRA-EVENT-DT OF S-PRREVT

MOVE 'FETCH-PRREVT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-PRREVT-EXIT.

/*****

*

*

HP370-SAVE-SECONDARY-CBRPLN SECTION.

HP370.

*

*

SET WCBRPLN-IDX TO WCBRPLN-COUNT OF W-CBRDEFN

INITIALIZE WCBRPLN-DATA OF W-CBRDEFN(WCBRPLN-IDX)

MOVE PLAN-TYPE OF S-PRREVT

TO PLAN-TYPE OF W-CBRDEFN(WCBRPLN-IDX)

SET SECONDARY-ONLY-YES OF W-CBRDEFN(WCBRPLN-IDX) TO TRUE

SET CHECK-EMPL-TERM-YES OF W-CBRDEFN(WCBRPLN-IDX) TO TRUE

SET EMPL-COVRG-TERM-NO OF W-CBRDEFN(WCBRPLN-IDX) TO TRUE

.

SAVE-SECONDARY-CBRPLN-EXIT.

/*****

* *

HP400-INIT-PARTIC SECTION.

HP400.

* *

IF WPARTIC-COUNT-MAX OF W-EVENT

DISPLAY 'Maximum CBR-Partic Entries Exceeded'

DISPLAY 'CBR-Partic occurs: ' WPARTIC-COUNT OF W-EVENT

'Emplid: ' EMPLID OF W-EVENT

MOVE 'INIT-PARTIC' TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO WPARTIC-COUNT OF W-EVENT

SET WPARTIC-IDX TO WPARTIC-COUNT OF W-EVENT

INITIALIZE WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

MOVE DEPENDENT-BENEF OF WDEPEND-DATA(WDEPEND-IDX)

TO DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

SET CBR-NOT-QUALIFIED OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

TO TRUE

SET CBR-PROCESS-OPEN OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

TO TRUE

SET CBR-INIT-NOT-QUALIFIED OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX) TO TRUE

SET CBR-SCND-NOT-QUALIFIED OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX) TO TRUE

MOVE COBRA-EMPLID OF WDEPEND-DATA(WDEPEND-IDX)

TO COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

MOVE BENEFIT-PROGRAM OF W-CURELT

TO BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)

SET CBR-REPROCESS-NONE OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX) TO TRUE

SET INSERT-NO OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

TO TRUE

SET UPDATE-NO OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

TO TRUE

MOVE ZERO TO INIT-PLAN-COUNT OF W-COUNT

MOVE ZERO TO SCND-PLAN-COUNT OF W-COUNT

MOVE ZERO TO SCND-PENDING-COUNT OF W-COUNT

MOVE ZERO TO NOT-DETERM-PLAN-COUNT OF W-COUNT

.

INIT-PARTIC-EXIT.

/*****

*

*

HP500-INIT-PARTIC-PLAN SECTION.

HP500.

*

*

IF WPLAN-COUNT-MAX OF W-EVENT

DISPLAY 'Maximum CBR-Partic-Plan Entries Exceeded'

DISPLAY 'CBR-Partic-Pln occurs: ' WPLAN-COUNT OF W-EVENT

'Emplid: ' EMPLID OF W-EVENT

MOVE 'INIT-PARTIC-PLAN' TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO WPLAN-COUNT OF W-EVENT

SET WPLAN-IDX TO WPLAN-COUNT OF W-EVENT

INITIALIZE WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE DEPENDENT-BENEF OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

TO DEPENDENT-BENEF OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE PLAN-TYPE OF W-CBRDEFN(WCBRPLN-IDX)

TO PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

SET COBRA-EVENT-UNQUALIFIED OF W-EVENT(WPLAN-IDX) TO TRUE

SET CBR-PROCESS-OPEN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO TRUE

SET CBR-ENROLL-NO OF W-EVENT(WPLAN-IDX) TO TRUE
SET COBRA-ERROR-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO TRUE
SET CBR-REPROCESS-NONE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO TRUE
SET INSERT-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX) TO TRUE
SET UPDATE-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX) TO TRUE
SET CLEAR-SCND-EVT-NO OF W-EVENT(WPLAN-IDX) TO TRUE
SET CUR-COVRG-NONE OF W-EVENT(WPLAN-IDX) TO TRUE

.

INIT-PARTIC-PLAN-EXIT.

/*****

* *

HP600-QUALIFY-PARTIC-PLAN SECTION.

HP600.

* *

IF EMPLOYEE OF W-DEPEND(WDEPEFF-IDX)

PERFORM HP610-CHECK-EMPL-CUR-COVRG

ELSE

PERFORM HP620-CHECK-DEPEND-CUR-COVRG

END-IF

MOVE 0 TO DEPRUL-PTR

IF CUR-COVRG-ACTIVE OF W-EVENT(WPLAN-IDX)

IF BENEFIT-LOST-EMPL OF W-CBR-EVT(WCBR-EVT-IDX)

PERFORM HP630-CHECK-EMPL-TERM-COVRG

ELSE

PERFORM HP640-CHECK-DEPEND-TERM-COVRG

END-IF

END-IF

IF CUR-COVRG-NONE OF W-EVENT(WPLAN-IDX)

OR CUR-COVRG-TERM OF W-EVENT(WPLAN-IDX)

PERFORM HP650-DETERMINE-INIT-SCND

END-IF

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

OR COBRA-EVENT-SECONDARY OF W-EVENT(WPLAN-IDX)

PERFORM HP660-CALC-COBRA-COVRG-DATES

END-IF

EVALUATE TRUE

WHEN COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

ADD 1 TO INIT-PLAN-COUNT OF W-COUNT

SET INSERT-YES OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO TRUE

WHEN COBRA-EVENT-SECONDARY OF W-EVENT(WPLAN-IDX)

ADD 1 TO SCND-PLAN-COUNT OF W-COUNT

SET INSERT-YES OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO TRUE

WHEN COBRA-EVENT-NOT-DETERMINED OF W-EVENT(WPLAN-IDX)

ADD 1 TO NOT-DETERM-PLAN-COUNT OF W-COUNT

SET INSERT-YES OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO TRUE

WHEN COBRA-EVENT-UNQUALIFIED OF W-EVENT(WPLAN-IDX)

SET CBR-PROCESS-CLOSED OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO TRUE

END-EVALUATE

.

QUALIFY-PARTIC-PLAN-EXIT.

/*****

* *

HP610-CHECK-EMPL-CUR-COVRG SECTION.

HP610.

* *

SET WCUR-IDX TO 1

SEARCH WCUR-DATA

AT END

CONTINUE

WHEN WCUR-IDX > WCUR-COUNT OF W-CURELT

CONTINUE

WHEN PLAN-TYPE OF WCBRPLN-DATA(WCBRPLN-IDX)

= PLAN-TYPE OF WCUR-DATA OF W-CURELT(WCUR-IDX)

MOVE BENEFIT-PLAN OF WCUR-DATA OF W-CURELT(WCUR-IDX)

TO BENEFIT-PLAN OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE COVRG-CD OF WCUR-DATA OF W-CURELT(WCUR-IDX)
TO COVRG-CD OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE EMPL-CONTRIBUTN-AMT OF W-CURELT(WCUR-IDX)
TO EMPL-CONTRIBUTN-AMT OF W-EVENT(WPLAN-IDX)
MOVE ANN-EX-CREDIT-FSA OF W-CURELT(WCUR-IDX)
TO ANN-EX-CREDIT-FSA OF W-EVENT(WPLAN-IDX)
MOVE ANNUAL-PLEDGE OF W-CURELT(WCUR-IDX)
TO ANNUAL-PLEDGE OF W-EVENT(WPLAN-IDX)
MOVE FSA-SUB-AMT-YTD OF W-CURELT(WCUR-IDX)
TO FSA-SUB-AMT-YTD OF W-EVENT(WPLAN-IDX)
MOVE FSA-APR-AMT-YTD OF W-CURELT(WCUR-IDX)
TO FSA-APR-AMT-YTD OF W-EVENT(WPLAN-IDX)
MOVE FSA-PD-AMT-YTD OF W-CURELT(WCUR-IDX)
TO FSA-PD-AMT-YTD OF W-EVENT(WPLAN-IDX)
MOVE HLTH-PROVIDER-ID OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
TO HLTH-PROVIDER-ID OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE PREVIOUSLY-SEEN OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
TO PREVIOUSLY-SEEN OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE OTH-INSURANCE-IND OF WCUR-DATA

OF W-CURELT(WCUR-IDX)
TO OTH-INSURANCE-IND OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE OTH-INSURANCE-NAME OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
TO OTH-INSURANCE-NAME OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

SET CUR-COVRG-ACTIVE OF W-EVENT(WPLAN-IDX) TO TRUE

END-SEARCH

.
CHECK-EMPL-CUR-COVRG-EXIT.

/*****

* *

HP620-CHECK-DEPEND-CUR-COVRG SECTION.

HP620.

* *

SET WCURDB-IDX TO 1

SEARCH WCURDB-DATA

AT END

CONTINUE

WHEN WCURDB-IDX > WCURDB-COUNT OF W-CURELT

CONTINUE

WHEN DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

= DEPENDENT-BENEF OF W-CURELT(WCURDB-IDX)

AND PLAN-TYPE OF WCBRPLN-DATA(WCBRPLN-IDX)

= PLAN-TYPE OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

MOVE BENEFIT-PLAN OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

TO BENEFIT-PLAN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COVRG-CD OF WCURDB-DATA OF W-CURELT(WCURDB-IDX)

TO COVRG-CD OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE HLTH-PROVIDER-ID OF WCURDB-DATA

OF W-CURELT(WCURDB-IDX)

TO HLTH-PROVIDER-ID OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE PREVIOUSLY-SEEN OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)
TO PREVIOUSLY-SEEN OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE OTH-INSURANCE-IND OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)
TO OTH-INSURANCE-IND OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE OTH-INSURANCE-NAME OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)
TO OTH-INSURANCE-NAME OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
SET CUR-COVRG-ACTIVE OF W-EVENT(WPLAN-IDX) TO TRUE

END-SEARCH

.

CHECK-DEPEND-CUR-COVRG-EXIT.

/*****

* *

HP630-CHECK-EMPL-TERM-COVRG SECTION.

HP630.

* *

IF CHECK-EMPL-TERM-YES OF W-CBRDEFN(WCBRPLN-IDX)

MOVE EMPLID OF W-EVENT TO EMPLID OF S-HTHTERM

MOVE BENEFIT-RCD-NO OF W-EVENT

TO EMPL-RCD-NO OF S-HTHTERM

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF S-HTHTERM

IF CBR-COV-OVERRIDE-YES

MOVE CBR-ALT-COV-TRM-DT OF W-EVENT

TO COVERAGE-BEGIN-DT OF BIND-DATA OF S-HTHTERM

MOVE CBR-ALT-COV-TRM-DT OF W-EVENT

TO COVERAGE-BEGIN-DT-2 OF BIND-DATA OF S-HTHTERM

ELSE

MOVE COBRA-EVENT-DT OF W-EVENT

TO COVERAGE-BEGIN-DT OF BIND-DATA OF S-HTHTERM

MOVE COBRA-EVENT-DT OF W-EVENT

TO COVERAGE-BEGIN-DT-2 OF BIND-DATA OF S-HTHTERM

END-IF

MOVE EMPLID OF W-EVENT TO EMPLID OF S-HTHWAIV

MOVE BENEFIT-RCD-NO OF W-EVENT

TO EMPL-RCD-NO OF S-HTHWAIV

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF S-HTHWAIV

MOVE COBRA-EVENT-DT OF W-EVENT

TO COVERAGE-BEGIN-DT OF BIND-DATA OF S-HTHWAIV

MOVE COBRA-EVENT-DT OF W-EVENT

TO COVERAGE-BEGIN-DT-2 OF BIND-DATA OF S-HTHWAIV

IF PLAN-TYPE-HEALTH OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM HP631-GET-EMPL-HTH-TERM

IF EMPL-COVRG-TERM-NO OF W-CBRDEFN(WCBRPLN-IDX)

AND COBRA-EVENT-MEDICARE OF W-EVENT

PERFORM HP633-GET-EMPL-HTH-WAIVE

END-IF

ELSE

IF PLAN-TYPE-FSA OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM HP632-GET-EMPL-FSA-TERM

IF EMPL-COVRG-TERM-NO OF W-CBRDEFN(WCBRPLN-IDX)

AND COBRA-EVENT-MEDICARE OF W-EVENT

PERFORM HP634-GET-EMPL-FSA-WAIVE

END-IF

END-IF

SET CHECK-EMPL-TERM-NO OF W-CBRDEFN(WCBRPLN-IDX)

TO TRUE

END-IF

END-IF

IF EMPL-COVRG-TERM-YES OF W-CBRDEFN(WCBRPLN-IDX)

SET CUR-COVRG-TERM OF W-EVENT(WPLAN-IDX) TO TRUE

IF CBR-COV-AS-OF-DT OF W-EVENT >=

EMPL-COVRG-TERM-DT OF W-CBRDEFN(WCBRPLN-IDX)

MOVE CBR-COV-AS-OF-DT OF W-EVENT

TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

ELSE

MOVE EMPL-COVRG-TERM-DT OF W-CBRDEFN(WCBRPLN-IDX)

TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

END-IF

ELSE

MOVE EMPL-COVRG-TERM-DT OF W-CBRDEFN(WCBRPLN-IDX)

TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

END-IF

.

CHECK-EMPL-TERM-COVRG-EXIT.

/*****

* *

HP631-GET-EMPL-HTH-TERM SECTION.

HP631.

* *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-HTHTERM OF W-CNTL

SQL-STMT OF S-HTHTERM

BIND-SETUP OF S-HTHTERM

BIND-DATA OF S-HTHTERM

SELECT-SETUP OF S-HTHTERM

SELECT-DATA OF S-HTHTERM

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-EMPL-HTH-TERM(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-HTHTERM

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-HTHTERM OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET WCUR-IDX TO 1

SEARCH WCUR-DATA

WHEN PLAN-TYPE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

= PLAN-TYPE OF WCUR-DATA

OF W-CURELT(WCUR-IDX)

CONTINUE

END-SEARCH

IF COVERAGE-END-DT OF W-CURELT(WCUR-IDX)

NOT = SPACE

MOVE COVERAGE-END-DT OF W-CURELT(WCUR-IDX)

TO BEGIN-DT OF DTWRK

MOVE 1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

IF END-DT OF DTWRK >= COBRA-EVENT-DT OF W-EVENT

OR (CBR-COV-OVERRIDE-YES

AND END-DT OF DTWRK >= CBR-ALT-COV-TRM-DT

OF W-EVENT)

SET EMPL-COVRG-TERM-YES

OF W-CBRDEFN(WCBRPLN-IDX)

TO TRUE

MOVE END-DT OF DTWRK

TO EMPL-COVRG-TERM-DT

OF W-CBRDEFN(WCBRPLN-IDX)

ELSE

SET EMPL-COVRG-TERM-NO

OF W-CBRDEFN(WCBRPLN-IDX)

TO TRUE

ELSE

SET EMPL-COVRG-TERM-NO OF W-CBRDEFN(WCBRPLN-IDX)

TO TRUE

END-IF

ELSE

MOVE 'GET-EMPL-HTH-TERM(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

SET EMPL-COVRG-TERM-YES OF W-CBRDEFN(WCBRPLN-IDX)

TO TRUE

MOVE COVERAGE-BEGIN-DT OF SELECT-DATA OF S-HTHTERM

TO EMPL-COVRG-TERM-DT OF W-CBRDEFN(WCBRPLN-IDX)

END-IF

.

GET-EMPL-HTH-TERM-EXIT.

/*****

*

*

HP632-GET-EMPL-FSA-TERM SECTION.

HP632.

*

*

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-FSATERM OF W-CNTL

SQL-STMT OF S-FSATERM

BIND-SETUP OF S-HTHTERM

BIND-DATA OF S-HTHTERM

SELECT-SETUP OF S-FSATERM

SELECT-DATA OF S-FSATERM

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-EMPL-FSA-TERM(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-FSATERM

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-FSATERM OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET WCUR-IDX TO 1

SEARCH WCUR-DATA

WHEN PLAN-TYPE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

= PLAN-TYPE OF WCUR-DATA

OF W-CURELT(WCUR-IDX)

CONTINUE

END-SEARCH

IF COVERAGE-END-DT OF W-CURELT(WCUR-IDX)

NOT = SPACE

MOVE COVERAGE-END-DT OF W-CURELT(WCUR-IDX)

TO BEGIN-DT OF DTWRK

MOVE 1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

IF END-DT OF DTWRK >= COBRA-EVENT-DT OF W-EVENT

OR (CBR-COV-OVERRIDE-YES

AND END-DT OF DTWRK >= CBR-ALT-COV-TRM-DT

OF W-EVENT)

```
        SET EMPL-COVRG-TERM-YES
          OF W-CBRDEFN(WCBRPLN-IDX)
            TO TRUE
        MOVE END-DT OF DTWRK
          TO EMPL-COVRG-TERM-DT
            OF W-CBRDEFN(WCBRPLN-IDX)
    ELSE
        SET EMPL-COVRG-TERM-NO
          OF W-CBRDEFN(WCBRPLN-IDX)
            TO TRUE
    ELSE
        SET EMPL-COVRG-TERM-NO OF W-CBRDEFN(WCBRPLN-IDX)
          TO TRUE
    END-IF
ELSE
    MOVE 'GET-EMPL-FSA-TERM(FETCH)'
      TO ERR-SECTION OF SQLRT
    PERFORM ZZ000-SQL-ERROR
    END-IF
ELSE
    SET EMPL-COVRG-TERM-YES OF W-CBRDEFN(WCBRPLN-IDX)
      TO TRUE
    MOVE COVERAGE-BEGIN-DT OF SELECT-DATA OF S-FSATERM
      TO EMPL-COVRG-TERM-DT OF W-CBRDEFN(WCBRPLN-IDX)
    END-IF
```

GET-EMPL-FSA-TERM-EXIT.

/*****

*

*

HP633-GET-EMPL-HTH-WAIVE SECTION.

HP633.

*

*

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-HTHWAIV OF W-CNTL

SQL-STMT OF S-HTHWAIV

BIND-SETUP OF S-HTHWAIV

BIND-DATA OF S-HTHWAIV

SELECT-SETUP OF S-HTHWAIV

SELECT-DATA OF S-HTHWAIV

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-EMPL-HTH-WAIVE(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-HTHWAIV

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-HTHWAIV OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF NOT RTNCD-END OF SQLRT

MOVE 'GET-EMPL-HTH-WAIVE(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

SET EMPL-COVRG-TERM-YES OF W-CBRDEFN(WCBRPLN-IDX)

TO TRUE

MOVE COVERAGE-BEGIN-DT OF SELECT-DATA OF S-HTHWAIV

TO EMPL-COVRG-TERM-DT OF W-CBRDEFN(WCBRPLN-IDX)

END-IF

.

GET-EMPL-HTH-WAIVE-EXIT.

/*****

* *

HP634-GET-EMPL-FSA-WAIVE SECTION.

HP634.

* *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-FSAWAIV OF W-CNTL

SQL-STMT OF S-FSAWAIV

BIND-SETUP OF S-HTHWAIV

BIND-DATA OF S-HTHWAIV

SELECT-SETUP OF S-FSAWAIV

SELECT-DATA OF S-FSAWAIV

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-EMPL-FSA-WAIVE(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-FSAWAIV

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-FSAWAIV OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF NOT RTNCD-END OF SQLRT

MOVE 'GET-EMPL-FSA-WAIVE(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

SET EMPL-COVRG-TERM-YES OF W-CBRDEFN(WCBRPLN-IDX)

TO TRUE

MOVE COVERAGE-BEGIN-DT OF SELECT-DATA OF S-FSAWAIV

TO EMPL-COVRG-TERM-DT OF W-CBRDEFN(WCBRPLN-IDX)

END-IF

.

GET-EMPL-FSA-WAIVE-EXIT.

/*****

*

*

HP640-CHECK-DEPEND-TERM-COVRG SECTION.

HP640.

*

*

MOVE EMPLID OF W-EVENT TO EMPLID OF S-DEPTERM

MOVE BENEFIT-RCD-NO OF W-EVENT

TO EMPL-RCD-NO OF S-DEPTERM

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF S-DEPTERM

IF CBR-COV-OVERRIDE-YES

MOVE CBR-ALT-COV-TRM-DT OF W-EVENT

TO COVERAGE-BEGIN-DT OF BIND-DATA OF S-DEPTERM

MOVE CBR-ALT-COV-TRM-DT OF W-EVENT

TO COVERAGE-BEGIN-DT-2 OF BIND-DATA OF S-DEPTERM

ELSE

MOVE COBRA-EVENT-DT OF W-EVENT

TO COVERAGE-BEGIN-DT OF BIND-DATA OF S-DEPTERM

MOVE COBRA-EVENT-DT OF W-EVENT

TO COVERAGE-BEGIN-DT-2 OF BIND-DATA OF S-DEPTERM

END-IF

MOVE DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO DEPENDENT-BENEF OF BIND-DATA OF S-DEPTERM

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-DEPTERM OF W-CNTL

SQL-STMT OF S-DEPTERM

BIND-SETUP OF S-DEPTERM

BIND-DATA OF S-DEPTERM

SELECT-SETUP OF S-DEPTERM

SELECT-DATA OF S-DEPTERM

IF RTNCD-ERROR OF SQLRT

MOVE 'CHECK-DEPEND-TERM-COVRG(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-DEPTERM

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-DEPTERM OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF NOT RTNCD-END OF SQLRT

MOVE 'CHECK-DEPEND-TERM-COVRG(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

SET CUR-COVRG-TERM OF W-EVENT(WPLAN-IDX) TO TRUE

IF CBR-COV-AS-OF-DT OF W-EVENT >=

COVERAGE-BEGIN-DT OF SELECT-DATA OF S-DEPTERM

MOVE CBR-COV-AS-OF-DT OF W-EVENT

TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

ELSE

MOVE COVERAGE-BEGIN-DT OF SELECT-DATA OF S-DEPTERM

TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

END-IF

* If Overage use Overage Date for Coverage Begin Dt per

* Plan Type's Dependent Rule

IF COBRA-EVENT-OVERAGE OF W-EVENT

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF W-SAVE

PERFORM XO950-SEARCH-DEP-PLNDFN

MOVE PROCESS-DT OF W-CNTL TO AGE-AS-OF-DT OF W-CNTL

PERFORM XQ000-DEPENDENT-AGE-CHECK

MOVE OVERAGE-DT OF W-SAVE

TO OVERAGE-DT OF W-DEPEND(WDEPEND-IDX)

IF OVERAGE-DT OF W-DEPEND(WDEPEND-IDX) >=

COVERAGE-BEGIN-DT OF SELECT-DATA OF S-DEPTERM

MOVE OVERAGE-DT OF W-DEPEND(WDEPEND-IDX)

TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

ELSE

MOVE COVERAGE-BEGIN-DT OF SELECT-DATA OF S-DEPTERM

TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

END-IF

END-IF

END-IF

.

CHECK-DEPEND-TERM-COVRG-EXIT.

/*****

*

*

HP650-DETERMINE-INIT-SCND SECTION.

HP650.

*

*

MOVE EMPLID OF W-EVENT TO EMPLID OF S-PRRPLN

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF S-PRRPLN

MOVE DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO DEPENDENT-BENEF OF BIND-DATA OF S-PRRPLN

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF S-PRRPLN

MOVE COBRA-EVENT-DT OF W-EVENT

TO COBRA-EVENT-DT OF BIND-DATA OF S-PRRPLN

COBRA-EVENT-DT-2 OF BIND-DATA OF S-PRRPLN

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-PRRPLN

BIND-SETUP OF S-PRRPLN

BIND-DATA OF S-PRRPLN

SELECT-SETUP OF S-PRRPLN

SELECT-DATA OF S-PRRPLN

IF RTNCD-ERROR OF SQLRT

MOVE 'DETERMINE-INIT-SCND(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-PRRPLN

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

IF CUR-COVRG-TERM OF W-EVENT(WPLAN-IDX)

AND SECONDARY-ONLY-NO

OF W-CBRDEFN(WCBRPLN-IDX)

SET COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

TO TRUE

ELSE

SET COBRA-EVENT-UNQUALIFIED OF W-EVENT(WPLAN-IDX)

TO TRUE

END-IF

ELSE

DISPLAY 'Emplid: ' EMPLID OF S-PRRPLN

MOVE 'DETERMINE-INIT-SCND(FETCH)'

```
        TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
ELSE

    IF COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX) NOT = SPACE
        AND COVERAGE-BEGIN-DT OF S-PRRPLN
            > COVERAGE-BEGIN-DT
                OF W-EVENT(WPLAN-IDX)

        SET COBRA-EVENT-NOT-DETERMINED OF W-EVENT(WPLAN-IDX)
            TO TRUE
    ELSE

        IF (COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)
            > COVERAGE-END-DT OF S-PRRPLN)
            AND CUR-COVRG-TERM OF W-EVENT(WPLAN-IDX)
            AND SECONDARY-ONLY-NO
                OF W-CBRDEFN(WCBRPLN-IDX)

            SET COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)
                TO TRUE
        ELSE

            IF COBRA-TERM-DT OF S-PRRPLN NOT = SPACE
                AND (COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)
```


> COBRA-TERM-DT OF S-PRRPLN)

AND CUR-COVRG-TERM

OF W-EVENT(WPLAN-IDX)

AND SECONDARY-ONLY-NO

OF W-CBRDEFN(WCBRPLN-IDX)

SET COBRA-EVENT-INITIAL

OF W-EVENT(WPLAN-IDX)

TO TRUE

ELSE

IF COBRA-ELECT-WAIVE OF S-PRRPLN

AND CUR-COVRG-TERM OF W-EVENT(WPLAN-IDX)

AND SECONDARY-ONLY-NO

OF W-CBRDEFN(WCBRPLN-IDX)

SET COBRA-EVENT-INITIAL

OF W-EVENT(WPLAN-IDX)

TO TRUE

ELSE

IF SUCCEEDING OF W-CBR-EVT(WCBR-EVT-IDX)

PERFORM HP651-GET-INIT-CBR-EVT-RULE

IF PRECEEDING

OF W-CBR-EVT(WCBREVT-INIT)

IF CBR-PROCESS-OPEN

OF S-PRRPLN

SET COBRA-EVENT-SECONDARY

OF W-EVENT(WPLAN-IDX)

TO TRUE

ADD 1 TO

SCND-PENDING-COUNT

OF W-COUNT

ELSE

PERFORM HP652-QUALIFY-SCND-EVENT

END-IF

END-IF

END-IF

END-IF

END-IF

END-IF

END-IF

END-IF

.

DETERMINE-INIT-SCND-EXIT.

/*****

* *

HP651-GET-INIT-CBR-EVT-RULE SECTION.

HP651.

* *

SET WCBREVT-SCND TO WCBR-EVT-IDX

MOVE COBRA-EVENT-CLASS OF S-PRRPLN

TO COBRA-EVENT-CLASS OF W-CNTL

PERFORM XH000-GET-CBR-EVT-RULE

SET WCBREVT-INIT TO WCBR-EVT-IDX

SET WCBR-EVT-IDX TO WCBREVT-SCND

MOVE COBRA-EVENT-CLASS OF W-EVENT

TO COBRA-EVENT-CLASS OF W-CNTL

.

GET-INIT-CBR-EVT-RULE-EXIT.

/*****

* *

HP652-QUALIFY-SCND-EVENT SECTION.

HP652.

*

*

MOVE COBRA-EVENT-ID OF S-PRRPLN

TO CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

MOVE BENEFIT-PROGRAM OF S-PRRPLN

TO INIT-BENEFIT-PGM OF W-EVENT(WPLAN-IDX)

MOVE CBR-ENROLL-STATUS OF S-PRRPLN

TO INIT-PLAN-ENROLL-SW OF W-EVENT(WPLAN-IDX)

MOVE COBRA-ELECT OF S-PRRPLN

TO INIT-PLAN-ELECT-SW OF W-EVENT(WPLAN-IDX)

IF COBRA-TERM-DT OF S-PRRPLN = SPACE

MOVE COVERAGE-END-DT OF S-PRRPLN

TO INIT-COVRG-END-DT OF W-EVENT(WPLAN-IDX)

ELSE

MOVE COBRA-TERM-DT OF S-PRRPLN TO BEGIN-DT OF DTWRK

MOVE -1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO INIT-COVRG-END-DT OF W-EVENT(WPLAN-IDX)

END-IF

IF INIT-PLAN-ELECT-WAIVE OF W-EVENT(WPLAN-IDX)

MOVE EMPLID OF W-EVENT TO EMPLID OF S-PRRDPND
MOVE BENEFIT-RCD-NO OF W-EVENT
TO BENEFIT-RCD-NO OF S-PRRDPND
MOVE CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)
TO COBRA-EVENT-ID OF S-PRRDPND
MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO PLAN-TYPE OF S-PRRDPND
MOVE DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO COBRA-DEP-BENEF OF BIND-DATA OF S-PRRDPND

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT
SQLRT
SQL-CURSOR-COMMON OF SQLRT
SQL-STMT OF S-PRRDPND
BIND-SETUP OF S-PRRDPND
BIND-DATA OF S-PRRDPND
SELECT-SETUP OF S-PRRDPND
SELECT-DATA OF S-PRRDPND

IF RTNCD-ERROR OF SQLRT

MOVE 'QUALIFY-SCND-EVENT(SELECT)'
TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

INITIALIZE SELECT-DATA OF S-PRRDPND

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

IF NOT RTNCD-END OF SQLRT

DISPLAY 'Emplid:' EMPLID OF S-PRRDPND

MOVE 'QUALIFY-SCND-EVENT(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

IF COBRA-TERM-DT OF S-PRRDPND = SPACE

MOVE COVERAGE-END-DT OF S-PRRDPND

TO INIT-COVRG-END-DT OF W-EVENT(WPLAN-IDX)

ELSE

MOVE COBRA-TERM-DT OF S-PRRDPND

TO BEGIN-DT OF DTWRK

MOVE -1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO INIT-COVRG-END-DT OF W-EVENT(WPLAN-IDX)

END-IF

IF INIT-COVRG-END-DT OF W-EVENT(WPLAN-IDX)

>= COBRA-EVENT-DT OF W-EVENT

SET COBRA-EVENT-SECONDARY OF W-EVENT(WPLAN-IDX)

TO TRUE

MOVE BENEFIT-PLAN OF S-PRRDPND

TO BENEFIT-PLAN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-DT OF W-EVENT

TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

SET COBRA-ELECT-ELECT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO TRUE

MOVE PROCESS-DT OF W-CNTL

TO COBRA-ELECT-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

END-IF

ELSE

IF INIT-COVRG-END-DT OF W-EVENT(WPLAN-IDX)

>= COBRA-EVENT-DT OF W-EVENT

SET COBRA-EVENT-SECONDARY OF W-EVENT(WPLAN-IDX)

TO TRUE

MOVE BENEFIT-PLAN OF S-PRRPLN

TO BENEFIT-PLAN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-DT OF W-EVENT

TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

SET COBRA-ELECT-ELECT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO TRUE

MOVE PROCESS-DT OF W-CNTL

TO COBRA-ELECT-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

END-IF

.

QUALIFY-SCND-EVENT-EXIT.

/*****

*

*

HP660-CALC-COBRA-COVRG-DATES SECTION.

HP660.

*

*

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

IF INCL-ALT-COVRG-NO OF W-CBR-EVT(WCBR-EVT-IDX)

OR (COBRA-EVENT-OVERAGE OF W-EVENT

AND DEPRUL-PTR NOT= 0)

- * No Alternate Grace Period Coverage or Overage which
- * previously calculated COBRA Period Begin based on
- * Dependent Rule, then COBRA Begin same as Covrg Begin Date

MOVE COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

TO COBRA-PD-BEGINS-DT OF W-EVENT(WPLAN-IDX)

ELSE

IF EVENT-ID OF W-EVENT NOT = ZERO

AND USE-BAS-EVENT-RULES-YES

OF W-CBR-EVT(WCBR-EVT-IDX)

PERFORM HP661-GET-BAS-COVRG-ENDS-CD

IF BENEFIT-LOST-EMPL OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE COVERAGE-ENDS-CD OF S-BAS-EVT

TO COBRA-PD-BEGINS-CD

```
                OF W-EVENT(WPLAN-IDX)

ELSE

    MOVE COVERAGE-BEGINS-CD OF S-BAS-EVT
      TO COBRA-PD-BEGINS-CD
        OF W-EVENT(WPLAN-IDX)

    END-IF

ELSE

    MOVE COBRA-PD-BEGINS-CD
      OF W-CBR-EVT(WCBR-EVT-IDX)
        TO COBRA-PD-BEGINS-CD
          OF W-EVENT(WPLAN-IDX)

    END-IF

    PERFORM HP662-CALC-COBRA-PD-BEGINS-DT

    END-IF

    MOVE COBRA-PD-BEGINS-DT OF W-EVENT(WPLAN-IDX)
      TO BEGIN-DT OF DTWRK

    MOVE MM-COVERAGE OF W-CBR-EVT(WCBR-EVT-IDX)
      TO MONTHS OF DTWRK

    SET OPTION-ADD-MONTHS OF DTWRK TO TRUE

    CALL 'PTPDTWRK' USING  DTWRK

    MOVE END-DT OF DTWRK TO BEGIN-DT OF DTWRK
```

MOVE -1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

IF DISABLED-YES OF W-DEPEND(WDEPEFF-IDX)

AND ADDL-MM-DISABLED OF W-CBR-EVT(WCBR-EVT-IDX)

> ZERO

MOVE COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

TO BEGIN-DT OF DTWRK

MOVE 1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO DISABL-CVG-BEG-DT OF W-EVENT(WPLAN-IDX)

MOVE COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

TO BEGIN-DT OF DTWRK

MOVE ADDL-MM-DISABLED OF W-CBR-EVT(WCBR-EVT-IDX)

TO MONTHS OF DTWRK

SET OPTION-ADD-MONTHS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

END-IF

MOVE COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

TO BEGIN-DT OF DTWRK

MOVE RESPONSE-DAYS OF W-CBR-EVT(WCBR-EVT-IDX)

TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO COBRA-ELECT-END-DT OF W-EVENT(WPLAN-IDX)

ELSE

IF SCND-EVT-MODE-EXTEND OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE COVERAGE-BEGIN-DT OF S-PRRPLN

TO BEGIN-DT OF DTWRK

ELSE

MOVE COBRA-EVENT-DT OF W-EVENT TO BEGIN-DT OF DTWRK
END-IF

MOVE SECOND-EVENT-MONTH OF W-CBR-EVT(WCBR-EVT-IDX)
TO MONTHS OF DTWRK
SET OPTION-ADD-MONTHS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK TO BEGIN-DT OF DTWRK
MOVE -1 TO DAYS OF DTWRK
SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK
TO COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)
END-IF

IF NOT EMPLOYEE OF W-DEPEND(WDEPEFF-IDX)

IF COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)
< DEP-MIN-COVRG-END-DT OF W-EVENT

MOVE DEP-MIN-COVRG-END-DT OF W-EVENT
TO COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

END-IF

END-IF

IF EMPLOYEE OF W-DEPEND(WDEPEFF-IDX)

OR SPOUSE OF W-DEPEND(WDEPEFF-IDX)

OR SAME-SEX-SPOUSE OF W-DEPEND(WDEPEFF-IDX)

OR EX-SPOUSE OF W-DEPEND(WDEPEFF-IDX)

IF COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

> MAX-COVRG-END-DT OF W-DEPEND(WDEPEND-IDX)

SET MSGID-CBR-REDUCED-END-DT TO TRUE

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF PYMSG

MOVE PLAN-TYPE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO MSGDATA1 OF PYMSG

MOVE COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

TO MSGDATA2 OF PYMSG

MOVE MAX-COVRG-END-DT OF W-DEPEND(WDEPEND-IDX)

TO MSGDATA3 OF PYMSG

PERFORM ZM000-MESSAGE

MOVE MAX-COVRG-END-DT OF W-DEPEND(WDEPEND-IDX)

TO COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

IF COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

< COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

SET MSGID-MEDICARE-PRIOR-INIT OF PYMSG

TO TRUE

ELSE

SET MSGID-MEDICARE-PRIOR-SCND OF PYMSG

TO TRUE

END-IF

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF PYMSG

MOVE PLAN-TYPE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO MSGDATA1 OF PYMSG

MOVE COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

TO MSGDATA2 OF PYMSG

MOVE COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

TO MSGDATA3 OF PYMSG

PERFORM ZM000-MESSAGE

SET COBRA-EVENT-UNQUALIFIED OF W-EVENT(WPLAN-IDX)

TO TRUE

END-IF

END-IF

END-IF

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

AND COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

< COBRA-PD-BEGINS-DT OF W-EVENT(WPLAN-IDX)

SET MSGID-BEGIN-PRIOR-CBR-PERIOD OF PYMSG TO TRUE

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF PYMSG

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO MSGDATA1 OF PYMSG

MOVE COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

TO MSGDATA2 OF PYMSG

MOVE COBRA-PD-BEGINS-DT OF W-EVENT(WPLAN-IDX)

TO MSGDATA3 OF PYMSG

PERFORM ZM000-MESSAGE

SET COBRA-EVENT-UNQUALIFIED OF W-EVENT(WPLAN-IDX)

TO TRUE

SET CBR-QUALIFY-ERROR-YES OF W-SW TO TRUE

END-IF

IF COBRA-EVENT-SECONDARY OF W-EVENT(WPLAN-IDX)

AND COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

< INIT-COVRG-END-DT OF W-EVENT(WPLAN-IDX)

SET MSGID-SCND-END-PRIOR-INIT-END OF PYMSG TO TRUE

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO MSGDATA1 OF PYMSG

MOVE COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

TO MSGDATA2 OF PYMSG

MOVE COVERAGE-END-DT OF S-PRRPLN TO MSGDATA3 OF PYMSG

PERFORM ZM000-MESSAGE

SET COBRA-EVENT-UNQUALIFIED OF W-EVENT(WPLAN-IDX)

TO TRUE

END-IF

.

CALC-COBRA-COVRG-DATES-EXIT.

/*****

* *

HP661-GET-BAS-COVRG-ENDS-CD SECTION.

HP661.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF S-BAS-EVT

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF S-BAS-EVT

MOVE SCHED-ID OF W-EVENT

TO SCHED-ID OF S-BAS-EVT

MOVE EVENT-ID OF W-EVENT

TO EVENT-ID OF S-BAS-EVT

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF S-BAS-EVT

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-BAS-EVT

BIND-SETUP OF S-BAS-EVT

BIND-DATA OF S-BAS-EVT

SELECT-SETUP OF S-BAS-EVT

SELECT-DATA OF S-BAS-EVT

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-BAS-COVRG-ENDS-CD(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-BAS-EVT

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

DISPLAY 'BAS Event Rules Not Found For '

'Emplid/Rcd#/SchedID/EvtID/PlanType: '

DISPLAY EMPLID OF S-BAS-EVT ' / '

BENEFIT-RCD-NO OF S-BAS-EVT ' / '

SCHED-ID OF S-BAS-EVT ' / '

EVENT-ID OF S-BAS-EVT ' / '

PLAN-TYPE OF S-BAS-EVT ' . '

MOVE 'GET-BAS-COVRG-ENDS-CD(FETCH)'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

ELSE

MOVE 'GET-BAS-COVRG-ENDS-CD(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

GET-BAS-COVRG-ENDS-CD-EXIT.

/*****

*

*

HP662-CALC-COBRA-PD-BEGINS-DT SECTION.

HP662.

*

*

EVALUATE TRUE

WHEN COBRA-PD-BEGINS-EVENT-DT OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-DT OF W-EVENT

TO COBRA-PD-BEGINS-DT OF W-EVENT(WPLAN-IDX)

WHEN COBRA-PD-BEGINS-ON-AFT-MM OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-DT OF W-EVENT TO BEGIN-DT OF DTWRK

IF MONTH-DAY OF BEGIN-DT OF DTWRK NOT = 1

MOVE 1 TO MONTHS OF DTWRK

SET OPTION-ADD-MONTHS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE 1 TO MONTH-DAY OF END-DT OF DTWRK

MOVE END-DT OF DTWRK

```
        TO COBRA-PD-BEGINS-DT OF W-EVENT(WPLAN-IDX)
ELSE
    MOVE COBRA-EVENT-DT OF W-EVENT
        TO COBRA-PD-BEGINS-DT OF W-EVENT(WPLAN-IDX)
END-IF
```

```
WHEN COBRA-PD-BEGINS-NEXT-MM OF W-EVENT(WPLAN-IDX)
```

```
    MOVE COBRA-EVENT-DT OF W-EVENT TO BEGIN-DT OF DTWRK
```

```
    MOVE 1 TO MONTHS OF DTWRK
```

```
    SET OPTION-ADD-MONTHS OF DTWRK TO TRUE
```

```
    CALL 'PTPDTWRK' USING DTWRK
```

```
    MOVE 1 TO MONTH-DAY OF END-DT OF DTWRK
```

```
    MOVE END-DT OF DTWRK
```

```
        TO COBRA-PD-BEGINS-DT OF W-EVENT(WPLAN-IDX)
```

```
WHEN COBRA-PD-BEGINS-NEXT-PAYPD OF W-EVENT(WPLAN-IDX)
```

```
    PERFORM HP663-GET-NEXT-PAYPERIOD
```

```
    MOVE PAY-BEGIN-DT OF SELECT-DATA OF S-CAL
```

```
        TO COBRA-PD-BEGINS-DT OF W-EVENT(WPLAN-IDX)
```

```
END-EVALUATE
```

CALC-COBRA-PD-BEGINS-DT-EXIT.

/*****

* *

HP663-GET-NEXT-PAYPERIOD SECTION.

HP663.

* *

MOVE COMPANY OF W-EVENT TO COMPANY OF BIND-DATA OF S-CAL

MOVE PAYGROUP OF W-EVENT TO PAYGROUP OF BIND-DATA OF S-CAL

MOVE COBRA-EVENT-DT OF W-EVENT

TO PAY-BEGIN-DT OF BIND-DATA OF S-CAL

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-CAL

BIND-SETUP OF S-CAL

BIND-DATA OF S-CAL

SELECT-SETUP OF S-CAL

SELECT-DATA OF S-CAL

IF RTNCD-ERROR OF SQLRT

```
MOVE 'GET-NEXT-PAYPERIOD(SELECT)'
    TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

INITIALIZE SELECT-DATA OF S-CAL

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT
    SQLRT
    SQL-CURSOR-COMMON OF SQLRT
IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

    DISPLAY 'Next Pay Period Not Found: '
        PAY-BEGIN-DT OF BIND-DATA OF S-CAL '.'
    DISPLAY 'Emplid: ' EMPLID OF W-EVENT '.'
    MOVE 'GET-NEXT-PAYPERIOD(FETCH)'
        TO ERR-SECTION OF SQLRT
    SET RTNCD-USER-ERROR OF SQLRT TO TRUE
    PERFORM ZZ000-SQL-ERROR
ELSE

    MOVE 'GET-NEXT-PAYPERIOD(FETCH)'
        TO ERR-SECTION OF SQLRT
```

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

GET-NEXT-PAYPERIOD-EXIT.

/*****

* *

HP700-QUALIFY-PARTIC SECTION.

HP700.

* *

IF INIT-PLAN-COUNT OF W-COUNT = ZERO

SET CBR-INIT-NOT-QUALIFIED OF W-EVENT(WPARTIC-IDX)

TO TRUE

ELSE

IF NOT-DETERM-PLAN-COUNT OF W-COUNT > ZERO

SET CBR-INIT-QUALIFY-PENDING

OF W-EVENT(WPARTIC-IDX) TO TRUE

ELSE


```
        SET CBR-INIT-QUALIFIED OF W-EVENT(WPARTIC-IDX)
            TO TRUE
    END-IF
END-IF

IF SCND-PLAN-COUNT OF W-COUNT = ZERO

    SET CBR-SCND-NOT-QUALIFIED OF W-EVENT(WPARTIC-IDX)
        TO TRUE
ELSE

    IF SCND-PENDING-COUNT OF W-COUNT > ZERO

        SET CBR-SCND-QUALIFY-PENDING OF W-EVENT(WPARTIC-IDX)
            TO TRUE
    ELSE
        SET CBR-SCND-QUALIFIED OF W-EVENT(WPARTIC-IDX)
            TO TRUE
    END-IF
END-IF

IF CBR-INIT-NOT-QUALIFIED OF W-EVENT(WPARTIC-IDX)
    AND CBR-SCND-NOT-QUALIFIED OF W-EVENT(WPARTIC-IDX)

    SET CBR-NOT-QUALIFIED OF WPARTIC-DATA
        OF W-EVENT(WPARTIC-IDX) TO TRUE
```

ELSE

SET INSERT-YES OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO TRUE

IF CBR-INIT-QUALIFY-PENDING OF W-EVENT(WPARTIC-IDX)
OR CBR-SCND-QUALIFY-PENDING
OF W-EVENT(WPARTIC-IDX)

SET CBR-QUALIFY-PENDING OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX) TO TRUE

SET CBR-QUALIFY-PENDING-YES OF W-SW TO TRUE

ELSE

SET CBR-QUALIFIED OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX) TO TRUE

SET CBR-QUALIFY-YES OF W-SW TO TRUE

END-IF

END-IF

.

QUALIFY-PARTIC-EXIT.

/*****

* *

HP800-ASSIGN-BENEFIT-PROGRAM SECTION.

HP800.

* *

PERFORM XL000-GET-ELIG-PARAMS

MOVE COBRA-EVENT-DT OF W-EVENT TO EVENT-DT OF BATBL

SET ACTION-LOAD-ASSIGN-TBLS OF BATBL TO TRUE

PERFORM RA000-TABLE-ACCESS

SET PROCESS-OPTION-ASSIGN OF ADMIN TO TRUE

SET BAS-TYPE-COBRA OF ADMIN TO TRUE

CALL 'PSPBAOPT' USING SQLRT

ADMIN

PARTC

POPTN

CRELT

EHIST

BATBL

BADED

PDEFN

ODEFN

CALTB

BCLTB

GEOTB

IF RTNCD-ERROR OF SQLRT

```
DISPLAY 'Emplid: ' EMPLID OF W-EVENT
MOVE 'ASSIGN-BENEFIT-PROGRAM(PSPBAOPT)'
    TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

IF PROCESS-STATUS-ASSIGN-NONE OF PARTC

    SET BAS-ASSIGN-NONE OF W-EVENT(WPARTIC-IDX) TO TRUE
    SET MSGID-BAS-ASSIGN-NONE OF PYMSG TO TRUE
    MOVE COBRA-EVENT-ID OF W-EVENT
        TO COBRA-EVENT-ID OF PYMSG
    MOVE DEPENDENT-BENEF OF WPARTIC-DATA
        OF W-EVENT(WPARTIC-IDX)
        TO DEPENDENT-BENEF OF PYMSG
    MOVE BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)
        TO MSGDATA1 OF PYMSG
    PERFORM ZM000-MESSAGE
END-IF
```

```
IF PROCESS-STATUS-ASSIGN-ERR OF PARTC

    SET BAS-ASSIGN-ERROR OF W-EVENT(WPARTIC-IDX) TO TRUE
    SET MSGID-BAS-ASSIGN-ERROR OF PYMSG TO TRUE
    MOVE COBRA-EVENT-ID OF W-EVENT
        TO COBRA-EVENT-ID OF PYMSG
```

MOVE DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

TO DEPENDENT-BENEF OF PYMSG

MOVE BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)

TO MSGDATA1 OF PYMSG

PERFORM ZM000-MESSAGE

END-IF

IF PROCESS-STATUS-ASSIGNED OF PARTC

SET BAS-ASSIGNED OF W-EVENT(WPARTIC-IDX) TO TRUE

MOVE BENEFIT-PROGRAM OF PARTC

TO BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)

END-IF

.

ASSIGN-BENEFIT-PROGRAM-EXIT.

/*****

*

*

HV000-DISC-QUALIFY-EVENT SECTION.

HV000.

*

*

PERFORM QA000-DISC-CURSORS

IF SQL-CURSOR OF S-EVT OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-EVT OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-QUALIFY-EVENT(S-EVT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-CURJOB OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-CURJOB OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-QUALIFY-EVENT(S-CURJOB)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF I-PAR OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF I-PAR OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-QUALIFY-EVENT(I-PAR)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF I-PARPLN OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF I-PARPLN OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-QUALIFY-EVENT(I-PARPLN)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-PRRPLN NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PRRPLN

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-QUALIFY-EVENT(S-PRRPLN)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-PRRDPND NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PRRDPND

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-QUALIFY-EVENT(S-PRRDPND)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-PRRDPND NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PRRDPND

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-QUALIFY-EVENT(S-PRRDPND)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

DISC-QUALIFY-EVENT-EXIT.

/*****

* *

HY000-TERM-QUALIFY-EVENT SECTION.

HY000.

* *

SET COBRA-PHASE-PARTIC OF W-CNTL TO TRUE

PERFORM TG000-FINISH-RUN-UNIT

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'Qualify Event Processing Ended at '

TIME-OUT OF W-WK '.'

.

TERM-QUALIFY-EVENT-EXIT.

/*****

* *

LA000-PROCESS-PARTICIPANT SECTION.

LA000.

* *

PERFORM LD000-INIT-PROCESS-PARTIC

PERFORM LG000-SELECT-PARTIC-DATA

PERFORM LJ000-FETCH-PROCESS-PARTIC

PERFORM UNTIL RTNCD-END OF SQLRT

PERFORM LM000-SAVE-PROCESS-PARTIC

IF EMPLID OF W-EVENT NOT = EMPLID OF W-CNTL

OR COBRA-EVENT-DT OF W-EVENT

NOT = WDEPEND-ASOFDT OF W-DEPEND

MOVE ZERO TO WDEPEND-COUNT OF W-DEPEND

MOVE ZERO TO WDEPNID-COUNT OF W-DEPNID

MOVE ZERO TO WDEPEFF-COUNT OF W-DEPEND

SET LOAD-EMPL-DEP-YES OF W-SW TO TRUE

PERFORM XN000-LOAD-DEPENDENT

END-IF

PERFORM LN000-OBTAIN-PARTIC-DATA

PERFORM LP000-LOAD-TABLE-DATA

PERFORM XH000-GET-CBR-EVT-RULE

IF CBR-INIT-QUALIFIED OF W-EVENT(WPARTIC-IDX)

OR CBR-INIT-OPT-PREP-ERROR OF W-EVENT(WPARTIC-IDX)

PERFORM LS000-PREPARE-OPTIONS

END-IF

IF CBR-INIT-ELECT-ENTERED OF W-EVENT(WPARTIC-IDX)

OR CBR-INIT-ELECT-ERROR OF W-EVENT(WPARTIC-IDX)

PERFORM LT000-PROCESS-ELECTION

END-IF

IF CBR-SCND-QUALIFIED OF W-EVENT(WPARTIC-IDX)

PERFORM LU000-EXTEND-COBRA-COVERAGE

END-IF

PERFORM XW000-STORE-COBRA-DATA

MOVE EMPLID OF SELECT-DATA OF S-PRCSPAR

TO EMPLID OF W-CNTL

MOVE BENEFIT-RCD-NO OF SELECT-DATA OF S-PRCSPAR

TO BENEFIT-RCD-NO OF W-CNTL

MOVE COBRA-EVENT-ID OF SELECT-DATA OF S-PRCSPAR

TO COBRA-EVENT-ID OF W-CNTL

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-PRCSPAR

TO COBRA-EVENT-DT OF W-CNTL

MOVE BENEFIT-PROGRAM OF SELECT-DATA OF S-PRCSPAR

TO BENEFIT-PROGRAM OF W-CNTL

PERFORM TD000-CHECKPOINT-RUN-UNIT

IF RESELECT-YES OF W-SW

PERFORM LG000-SELECT-PARTIC-DATA

END-IF

PERFORM LJ000-FETCH-PROCESS-PARTIC

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

PERFORM LV000-DISC-PROCESS-PARTIC

PERFORM LY000-TERM-PROCESS-PARTIC

.

PROCESS-PARTICIPANT-EXIT.

/*****

* *

LD000-INIT-PROCESS-PARTIC SECTION.

LD000.

* *

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'COBRA Participant Processing Started at '

TIME-OUT OF W-WK '.'

SET COBRA-PHASE-PARTIC OF W-CNTL TO TRUE

SET PROGRESS-REPORT-PARTIC OF W-WK TO TRUE

PERFORM TA000-START-RUN-UNIT

MOVE SPACE TO BENEFIT-PROGRAM OF BATBL

.

INIT-PROCESS-PARTIC-EXIT.

/*****

* *

LG000-SELECT-PARTIC-DATA SECTION.

LG000.

* *

PERFORM LG100-SELECT-PROCESS-PARTIC

PERFORM LG200-SELECT-PRCSPAR-PLAN

PERFORM LG300-SELECT-PARTIC-OPTN

PERFORM LG400-SELECT-PARTIC-DPND

PERFORM XC200-FETCH-PARTIC-PLN3

PERFORM LN210-FETCH-PARTIC-OPTN

PERFORM LN310-FETCH-PARTIC-DPND

.

SELECT-PARTIC-DATA-EXIT.

/*****

* *

LG100-SELECT-PROCESS-PARTIC SECTION.

LG100.

* *

MOVE BENEFIT-PROGRAM OF W-CNTL

TO BENEFIT-PROGRAM OF BIND-DATA OF S-PRCSPAR

MOVE BENEFIT-PROGRAM OF W-CNTL

TO BENEFIT-PROGRAM-2 OF BIND-DATA OF S-PRCSPAR

BENEFIT-PROGRAM-3 OF BIND-DATA OF S-PRCSPAR

BENEFIT-PROGRAM-4 OF BIND-DATA OF S-PRCSPAR

MOVE COBRA-EVENT-DT OF W-CNTL

TO COBRA-EVENT-DT OF BIND-DATA OF S-PRCSPAR

MOVE COBRA-EVENT-DT OF W-CNTL

TO COBRA-EVENT-DT-2 OF BIND-DATA OF S-PRCSPAR

MOVE EMPLID OF W-CNTL TO EMPLID OF BIND-DATA OF S-PRCSPAR

MOVE EMPLID OF W-CNTL TO EMPLID-2 OF BIND-DATA OF S-PRCSPAR

EMPLID-3 OF BIND-DATA OF S-PRCSPAR

EMPLID-4 OF BIND-DATA OF S-PRCSPAR

MOVE BENEFIT-RCD-NO OF W-CNTL

TO BENEFIT-RCD-NO OF BIND-DATA OF S-PRCSPAR

MOVE BENEFIT-RCD-NO OF W-CNTL

TO BENEFIT-RCD-NO-2 OF BIND-DATA OF S-PRCSPAR

MOVE COBRA-EVENT-ID OF W-CNTL

TO COBRA-EVENT-ID OF BIND-DATA OF S-PRCSPAR

MOVE COBRA-EVENT-ID OF W-CNTL

TO COBRA-EVENT-ID-2 OF BIND-DATA OF S-PRCSPAR

MOVE DEPENDENT-BENEF OF W-CNTL

TO DEPENDENT-BENEF OF BIND-DATA OF S-PRCSPAR

DEPENDENT-BENEF-2 OF BIND-DATA OF S-PRCSPAR

DEPENDENT-BENEF-3 OF BIND-DATA OF S-PRCSPAR

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PRCSPAR OF W-CNTL

SQL-STMT OF S-PRCSPAR

BIND-SETUP OF S-PRCSPAR

BIND-DATA OF S-PRCSPAR

SELECT-SETUP OF S-PRCSPAR

SELECT-DATA OF S-PRCSPAR

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-PROCESS-PARTIC' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-PROCESS-PARTIC-EXIT.

/*****

*

*

LG200-SELECT-PRCSPAR-PLAN SECTION.

LG200.

*

*

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PARPLN3 OF W-CNTL

SQL-STMT OF S-PARPLN3

BIND-SETUP OF S-PRCSPAR

BIND-DATA OF S-PRCSPAR

SELECT-SETUP OF S-PARPLN3

SELECT-DATA OF S-PARPLN3

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-PRCSPAR-PLAN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-PRCSPAR-PLAN-EXIT.

/*****

* *

LG300-SELECT-PARTIC-OPTN SECTION.

LG300.

* *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PAROPTN OF W-CNTL

SQL-STMT OF S-PAROPTN

BIND-SETUP OF S-PRCSPAR

BIND-DATA OF S-PRCSPAR

SELECT-SETUP OF S-PAROPTN

SELECT-DATA OF S-PAROPTN

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-PARTIC-OPTN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-PARTIC-OPTN-EXIT.

/*****

* *

LG400-SELECT-PARTIC-DPND SECTION.

LG400.

* *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PARDPND OF W-CNTL

SQL-STMT OF S-PARDPND

BIND-SETUP OF S-PRCSPAR

BIND-DATA OF S-PRCSPAR

SELECT-SETUP OF S-PARDPND

SELECT-DATA OF S-PARDPND

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-PARTIC-DPND' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-PARTIC-DPND-EXIT.

/*****

* *

LJ000-FETCH-PROCESS-PARTIC SECTION.

LJ000.

* *

INITIALIZE SELECT-DATA OF S-PRCSPAR

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-PRCSPAR OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-PROCESS-PARTIC' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-PROCESS-PARTIC-EXIT.

/*****

* *

LM000-SAVE-PROCESS-PARTIC SECTION.

LM000.

* *

MOVE ZERO TO WPARTIC-COUNT OF W-EVENT

ADD 1 TO WPARTIC-COUNT OF W-EVENT

SET WPARTIC-IDX TO WPARTIC-COUNT OF W-EVENT

INITIALIZE WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

SET INSERT-NO OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO TRUE

SET UPDATE-NO OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO TRUE

MOVE EMPLID OF SELECT-DATA OF S-PRCSPAR
TO EMPLID OF W-EVENT

MOVE BENEFIT-RCD-NO OF SELECT-DATA OF S-PRCSPAR
TO BENEFIT-RCD-NO OF W-EVENT

MOVE COBRA-EVENT-ID OF SELECT-DATA OF S-PRCSPAR
TO COBRA-EVENT-ID OF W-EVENT

MOVE EVT-PROCESS-STATUS OF S-PRCSPAR
TO CBR-PROCESS-STATUS OF WEVENT-DATA OF W-EVENT

MOVE EVT-QUALIFY-STATUS OF S-PRCSPAR
TO CBR-QUALIFY-STATUS OF WEVENT-DATA OF W-EVENT

MOVE CBR-EVT-REPRCS-IND OF S-PRCSPAR
TO CBR-EVT-REPRCS-IND OF W-EVENT

MOVE CBR-COV-OVERRIDE OF S-PRCSPAR
TO CBR-COV-OVERRIDE OF WEVENT-DATA OF W-EVENT

MOVE CBR-ALT-COV-TRM-DT OF S-PRCSPAR
TO CBR-ALT-COV-TRM-DT OF WEVENT-DATA OF W-EVENT

MOVE CBR-COV-AS-OF-DT OF S-PRCSPAR
TO CBR-COV-AS-OF-DT OF WEVENT-DATA OF W-EVENT

MOVE DEPENDENT-BENEF OF SELECT-DATA OF S-PRCSPAR
TO DEPENDENT-BENEF OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-EVENT-DT OF SELECT-DATA OF S-PRCSPAR

TO COBRA-EVENT-DT OF W-EVENT
MOVE EMPL-RCD-NO OF SELECT-DATA OF S-PRCSPAR
TO EMPL-RCD-NO OF W-EVENT
MOVE SCHED-ID OF SELECT-DATA OF S-PRCSPAR
TO SCHED-ID OF W-EVENT
MOVE EVENT-ID OF SELECT-DATA OF S-PRCSPAR
TO EVENT-ID OF W-EVENT
MOVE COBRA-EVENT-CLASS OF SELECT-DATA OF S-PRCSPAR
TO COBRA-EVENT-CLASS OF W-EVENT
MOVE COBRA-EVENT-CLASS OF SELECT-DATA OF S-PRCSPAR
TO COBRA-EVENT-CLASS OF W-CNTL
MOVE BENEFIT-PROGRAM OF SELECT-DATA OF S-PRCSPAR
TO BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)
MOVE CBR-QUALIFY-STATUS OF S-PRCSPAR
TO CBR-QUALIFY-STATUS OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
MOVE CBR-PROCESS-STATUS OF S-PRCSPAR
TO CBR-PROCESS-STATUS OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
MOVE CBR-PAR-REPRCS-IND OF S-PRCSPAR
TO CBR-PAR-REPRCS-IND OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
MOVE CBR-INIT-EVENT-STS OF S-PRCSPAR
TO CBR-INIT-EVENT-STS OF W-EVENT(WPARTIC-IDX)
MOVE CBR-SCND-EVENT-STS OF S-PRCSPAR
TO CBR-SCND-EVENT-STS OF W-EVENT(WPARTIC-IDX)

MOVE CBR-INIT-NOTIFY-DT OF S-PRCSPAR

TO CBR-INIT-NOTIFY-DT OF W-EVENT(WPARTIC-IDX)

MOVE CBR-SCND-NOTIFY-DT OF S-PRCSPAR

TO CBR-SCND-NOTIFY-DT OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-ELECT OF S-PRCSPAR

TO COBRA-ELECT OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-ELECT-DT OF S-PRCSPAR

TO COBRA-ELECT-DT OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-WAIVE-DT OF S-PRCSPAR

TO COBRA-WAIVE-DT OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-REVOKE-DT OF S-PRCSPAR

TO COBRA-REVOKE-DT OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-EMPLID OF S-PRCSPAR

TO COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

MOVE BAS-ASSIGN-STATUS OF S-PRCSPAR

TO BAS-ASSIGN-STATUS OF W-EVENT(WPARTIC-IDX)

MOVE CBR-TERM-NOTIFY-DT OF S-PRCSPAR

TO CBR-TERM-NOTIFY-DT OF W-EVENT(WPARTIC-IDX)

.

SAVE-PROCESS-PARTIC-EXIT.

/*****

* *

LN000-OBTAIN-PARTIC-DATA SECTION.

LN000.

* *

PERFORM LN100-OBTAIN-PARTIC-PLAN-DATA

PERFORM LN200-OBTAIN-PARTIC-OPTN-DATA

PERFORM LN300-OBTAIN-PARTIC-DPND-DATA

.

OBTAIN-PARTIC-DATA-EXIT.

/*****

* *

LN100-OBTAIN-PARTIC-PLAN-DATA SECTION.

LN100.

* *

MOVE ZERO TO WPLAN-COUNT OF W-EVENT

PERFORM UNTIL EMPLID OF SELECT-DATA OF S-PARPLN3

= HIGH-VALUE

OR EMPLID OF SELECT-DATA OF S-PARPLN3

NOT = EMPLID OF SELECT-DATA OF S-PRCSPAR

OR BENEFIT-RCD-NO OF SELECT-DATA OF S-PARPLN3

NOT = BENEFIT-RCD-NO OF SELECT-DATA

OF S-PRCSPAR

OR COBRA-EVENT-ID OF SELECT-DATA OF S-PARPLN3

NOT = COBRA-EVENT-ID OF SELECT-DATA

OF S-PRCSPAR

OR DEPENDENT-BENEF OF SELECT-DATA OF S-PARPLN3

NOT = DEPENDENT-BENEF OF SELECT-DATA

OF S-PRCSPAR

ADD 1 TO WPLAN-COUNT OF W-EVENT

PERFORM XC300-SAVE-PARTIC-PLN3

PERFORM XC200-FETCH-PARTIC-PLN3

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

OBTAIN-PARTIC-PLAN-DATA-EXIT.

/*****

*

*

LN200-OBTAIN-PARTIC-OPTN-DATA SECTION.

LN200.

*

MOVE ZERO TO WOPTN-COUNT OF W-EVENT

PERFORM UNTIL EMPLID OF SELECT-DATA OF S-PAROPTN

= HIGH-VALUE

OR EMPLID OF SELECT-DATA OF S-PAROPTN

NOT = EMPLID OF SELECT-DATA OF S-PRCSPAR

OR BENEFIT-RCD-NO OF SELECT-DATA OF S-PAROPTN

NOT = BENEFIT-RCD-NO OF SELECT-DATA

OF S-PRCSPAR

OR COBRA-EVENT-ID OF SELECT-DATA OF S-PAROPTN

NOT = COBRA-EVENT-ID OF SELECT-DATA

OF S-PRCSPAR

OR DEPENDENT-BENEF OF SELECT-DATA OF S-PAROPTN

NOT = DEPENDENT-BENEF OF SELECT-DATA

OF S-PRCSPAR

ADD 1 TO WOPTN-COUNT OF W-EVENT

PERFORM LN220-SAVE-PARTIC-OPTN

PERFORM LN210-FETCH-PARTIC-OPTN

END-PERFORM

OBTAIN-PARTIC-OPTN-DATA-EXIT.

/*****

* *

LN210-FETCH-PARTIC-OPTN SECTION.

LN210.

* *

INITIALIZE SELECT-DATA OF S-PAROPTN

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-PAROPTN OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

MOVE 'FETCH-PARTIC-OPTN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

FETCH-PARTIC-OPTN-EXIT.

/*****

* *

LN220-SAVE-PARTIC-OPTN SECTION.

LN220.

* *

SET WOPTN-IDX TO WOPTN-COUNT OF W-EVENT

INITIALIZE WOPTN-DATA OF W-EVENT(WOPTN-IDX)

SET INSERT-NO OF WOPTN-DATA OF W-EVENT(WOPTN-IDX) TO TRUE

MOVE DEPENDENT-BENEF OF SELECT-DATA OF S-PAROPTN

TO DEPENDENT-BENEF OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

MOVE PLAN-TYPE OF SELECT-DATA OF S-PAROPTN

TO PLAN-TYPE OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

MOVE OPTION-ID OF SELECT-DATA OF S-PAROPTN

TO OPTION-ID OF W-EVENT(WOPTN-IDX)

MOVE BENEFIT-PLAN OF SELECT-DATA OF S-PAROPTN

TO BENEFIT-PLAN OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

MOVE COVRG-CD OF SELECT-DATA OF S-PAROPTN

TO COVRG-CD OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
MOVE OPTION-CD OF SELECT-DATA OF S-PAROPTN
TO OPTION-CD OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
MOVE CBR-MM-COST OF SELECT-DATA OF S-PAROPTN
TO CBR-MM-COST OF W-EVENT(WOPTN-IDX)
MOVE CBR-DISABL-MM-COST OF SELECT-DATA OF S-PAROPTN
TO CBR-DISABL-MM-COST OF W-EVENT(WOPTN-IDX)

SAVE-PARTIC-OPTN-EXIT.

/*****

* *

LN300-OBTAIN-PARTIC-DPND-DATA SECTION.

LN300.

*

MOVE ZERO TO WDPND-COUNT OF W-EVENT

PERFORM UNTIL EMPLID OF SELECT-DATA OF S-PARDPND

= HIGH-VALUE

OR EMPLID OF SELECT-DATA OF S-PARDPND

NOT = EMPLID OF SELECT-DATA OF S-PRCSPAR

OR BENEFIT-RCD-NO OF SELECT-DATA OF S-PARDPND

NOT = BENEFIT-RCD-NO OF SELECT-DATA

OF S-PRCSPAR

OR COBRA-EVENT-ID OF SELECT-DATA OF S-PARDPND

NOT = COBRA-EVENT-ID OF SELECT-DATA

OF S-PRCSPAR

OR DEPENDENT-BENEF OF SELECT-DATA OF S-PARDPND

NOT = DEPENDENT-BENEF OF SELECT-DATA

OF S-PRCSPAR

IF WDPND-COUNT-MAX OF W-EVENT

DISPLAY 'Max # of Dependents Exceeded'

MOVE 'OBTAIN-PARTIC-DPND-DATA'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO WDPND-COUNT OF W-EVENT

PERFORM LN320-SAVE-PARTIC-DPND

PERFORM LN310-FETCH-PARTIC-DPND

END-PERFORM

.

OBTAIN-PARTIC-DPND-DATA-EXIT.

/*****

* *

LN310-FETCH-PARTIC-DPND SECTION.

LN310.

* *

INITIALIZE SELECT-DATA OF S-PARDPND

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-PARDPND OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

MOVE 'FETCH-PARTIC-DPND' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

FETCH-PARTIC-DPND-EXIT.

/*****

* *

LN320-SAVE-PARTIC-DPND SECTION.

LN320.

* *

SET WDPND-IDX TO WDPND-COUNT OF W-EVENT

INITIALIZE WDPND-DATA OF W-EVENT(WDPND-IDX)

SET UPDATE-NO OF WDPND-DATA OF W-EVENT(WDPND-IDX) TO TRUE

MOVE DEPENDENT-BENEF OF SELECT-DATA OF S-PARDPND

TO DEPENDENT-BENEF OF WDPND-DATA

OF W-EVENT(WDPND-IDX)

MOVE PLAN-TYPE OF SELECT-DATA OF S-PARDPND

TO PLAN-TYPE OF WDPND-DATA OF W-EVENT(WDPND-IDX)

MOVE COBRA-DEP-BENEF OF SELECT-DATA OF S-PARDPND

TO COBRA-DEP-BENEF OF WDPND-DATA

OF W-EVENT(WDPND-IDX)

MOVE COBRA-ERROR OF SELECT-DATA OF S-PARDPND

TO COBRA-ERROR OF WDPND-DATA OF W-EVENT(WDPND-IDX)

MOVE HLTH-PROVIDER-ID OF S-PARDPND

TO HLTH-PROVIDER-ID OF WDPND-DATA

OF W-EVENT(WDPND-IDX)

MOVE PREVIOUSLY-SEEN OF S-PARDPND
TO PREVIOUSLY-SEEN OF WDPND-DATA
OF W-EVENT(WDPND-IDX)

MOVE OTH-INSURANCE-IND OF S-PARDPND
TO OTH-INSURANCE-IND OF WDPND-DATA
OF W-EVENT(WDPND-IDX)

MOVE OTH-INSURANCE-NAME OF S-PARDPND
TO OTH-INSURANCE-NAME OF WDPND-DATA
OF W-EVENT(WDPND-IDX)

.

SAVE-PARTIC-DPND-EXIT.

/*****

* *

LP000-LOAD-TABLE-DATA SECTION.

LP000.

* *

IF BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)
NOT = BENEFIT-PROGRAM OF BATBL

SET INIT-RT-ARRAY-YES OF BADED TO TRUE
MOVE ZERO TO CALIX-COUNT OF CALTB

MOVE ZERO TO CAL-COUNT OF CALTB
MOVE BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)
TO BENEFIT-PROGRAM OF BATBL
MOVE COBRA-EVENT-DT OF W-EVENT TO EVENT-DT OF BATBL
SET ACTION-LOAD-PGM-DEFN OF BATBL TO TRUE
PERFORM RA000-TABLE-ACCESS
MOVE HIGH-VALUE TO ELGIX-MAX-EFFDT OF PDEFN
MOVE HIGH-VALUE TO GEOIX-MAX-EFFDT OF PDEFN
MOVE HIGH-VALUE TO WCBR-EVT-MAX-EFFDT OF W-CBR-EVT
MOVE HIGH-VALUE TO WCVGCD-MAX-EFFDT OF W-CVGCD
MOVE COBRA-SURCHARGE OF PDEFN
TO COBRA-SURCHARGE OF W-CBRDEFN
MOVE COBRA-DISABL-SURCG OF PDEFN
TO COBRA-DISABL-SURCG OF W-CBRDEFN
PERFORM XO500-LOAD-DEP-RULE-PLANDF
END-IF

IF BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)
= BENEFIT-PROGRAM OF BATBL
AND COBRA-EVENT-DT OF W-EVENT
< EFFDT OF PGMDF-DATA OF PDEFN

MOVE BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)
TO BENEFIT-PROGRAM OF BATBL
MOVE COBRA-EVENT-DT OF W-EVENT TO EVENT-DT OF BATBL
SET ACTION-LOAD-PGM-DEFN OF BATBL TO TRUE

PERFORM RA000-TABLE-ACCESS

MOVE HIGH-VALUE TO ELGIX-MAX-EFFDT OF PDEFN

MOVE HIGH-VALUE TO GEOIX-MAX-EFFDT OF PDEFN

MOVE HIGH-VALUE TO WCBR-EVT-MAX-EFFDT OF W-CBR-EVT

MOVE HIGH-VALUE TO WCVGCD-MAX-EFFDT OF W-CVGCD

MOVE COBRA-SURCHARGE OF PDEFN

TO COBRA-SURCHARGE OF W-CBRDEFN

MOVE COBRA-DISABL-SURCG OF PDEFN

TO COBRA-DISABL-SURCG OF W-CBRDEFN

PERFORM XO500-LOAD-DEP-RULE-PLANDF

END-IF

IF BENADMIN-YES OF W-CNTL

IF COBRA-EVENT-DT OF W-EVENT

< ELGIX-MAX-EFFDT OF PDEFN

MOVE COBRA-EVENT-DT OF W-EVENT TO EVENT-DT OF BATBL

SET ACTION-LOAD-ELIG OF BATBL TO TRUE

PERFORM RA000-TABLE-ACCESS

END-IF

IF COBRA-EVENT-DT OF W-EVENT

< GEOIX-MAX-EFFDT OF PDEFN

MOVE COBRA-EVENT-DT OF W-EVENT TO EVENT-DT OF BATBL

SET ACTION-LOAD-GEO OF BATBL TO TRUE

PERFORM RA000-TABLE-ACCESS

END-IF

END-IF

SET INIT-PARTIC-YES OF BADED TO TRUE

IF COBRA-EVENT-DT OF W-EVENT

< WCVGCD-MAX-EFFDT OF W-CVGCD

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT OF BIND-DATA OF S-CVGCD

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT-2 OF BIND-DATA OF S-CVGCD

PERFORM XJ000-LOAD-COVRG-CD-TBL

END-IF

IF COBRA-EVENT-DT OF W-EVENT

< WCBR-EVT-MAX-EFFDT OF W-CBR-EVT

PERFORM XF000-LOAD-CBR-EVT-RULES

END-IF

.

LOAD-TABLE-DATA-EXIT.

/*****

* *

LS000-PREPARE-OPTIONS SECTION.

LS000.

* *

SET WDEPEND-IDX TO 1

SEARCH WDEPEND-DATA

WHEN DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

= DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

CONTINUE

END-SEARCH

MOVE COBRA-EVENT-DT OF W-EVENT TO SEARCH-EFFDT OF W-DEPEND

PERFORM XN700-SRCH-DEP-EFF-DATA

PERFORM XL000-GET-ELIG-PARAMS

MOVE BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)

TO BENEFIT-PROGRAM OF PARTC

PERFORM LS100-PREPARE-BASE-OPTN

IF VALID-OPT-PREP-YES OF W-SW

IF BAS-ASSIGNED OF W-EVENT(WPARTIC-IDX)

PERFORM LS200-PREPARE-BENADMIN-OPTN

ELSE

PERFORM VARYING WOPTN-IDX FROM 1 BY 1

UNTIL WOPTN-IDX > WOPTN-COUNT OF W-EVENT

PERFORM LS221-CALC-MONTHLY-COST

SET INSERT-YES OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX) TO TRUE

END-PERFORM

END-IF

SET CBR-INIT-OPTION-PREPARED OF W-EVENT(WPARTIC-IDX)

TO TRUE

SET UPDATE-YES OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

TO TRUE

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

MOVE SPACE TO BENEFIT-PLAN OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE SPACE TO COVRG-CD OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE SPACE TO OPTION-CD OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
SET UPDATE-YES OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX) TO TRUE

END-IF

END-PERFORM

ELSE

SET CBR-INIT-OPT-PREP-ERROR OF W-EVENT(WPARTIC-IDX)
TO TRUE

SET UPDATE-YES OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO TRUE

END-IF

.

PREPARE-OPTIONS-EXIT.

/*****

*

*

LS100-PREPARE-BASE-OPTN SECTION.

LS100.

*

*

SET VALID-OPT-PREP-YES OF W-SW TO TRUE

SET ERROR-NO OF W-SW TO TRUE

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

IF EMPLOYEE OF W-DEPEND(WDEPEFF-IDX)

OR SPOUSE OF W-DEPEND(WDEPEFF-IDX)

OR SAME-SEX-SPOUSE OF W-DEPEND(WDEPEFF-IDX)

OR EX-SPOUSE OF W-DEPEND(WDEPEFF-IDX)

PERFORM LS110-BUILD-BASE-OPTN-SET

ELSE

PERFORM LS120-BUILD-BASE-OPTN-ONLY

END-IF

END-IF

IF ERROR-YES OF W-SW

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF PYMSG
MOVE DEPENDENT-BENEF OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
TO DEPENDENT-BENEF OF PYMSG
MOVE PLAN-TYPE OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX) TO MSGDATA1 OF PYMSG
MOVE BENEFIT-PLAN OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX) TO MSGDATA2 OF PYMSG
MOVE COVRG-CD OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX) TO MSGDATA3 OF PYMSG
PERFORM ZM000-MESSAGE
SET COBRA-ERROR-YES OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX) TO TRUE
SET VALID-OPT-PREP-NO OF W-SW TO TRUE
SET ERROR-NO OF W-SW TO TRUE

END-IF

END-PERFORM

.

PREPARE-BASE-OPTN-EXIT.

/*****

*

*

LS110-BUILD-BASE-OPTN-SET SECTION.

LS110.

*

*

IF PLAN-TYPE-HEALTH OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM LS111-DEFINE-CUR-COVRG-CD

END-IF

IF ERROR-NO OF W-SW

PERFORM LS121-CHECK-BENOPTN-MATCH

SET PLANDF-IDX TO 1

SEARCH PLANDF-DATA

AT END

DISPLAY

'Cobra Plan Not Defined in Pgm/PlanType/Effdt: '

BENEFIT-PROGRAM OF PGMDF-DATA

/'

PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

/'

EFFDT OF PGMDF-DATA

DISPLAY

'Emplid: ' EMPLID OF W-EVENT
MOVE 'BUILD-BASE-OPTN-SET'
TO ERR-SECTION OF SQLRT
SET RTNCD-USER-ERROR OF SQLRT TO TRUE
PERFORM ZZ000-SQL-ERROR

WHEN PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
= PLAN-TYPE OF PLANDF-DATA OF PDEFN(PLANDF-IDX)

CONTINUE

END-SEARCH

SET OPTNDF-IDX TO OPTNDF-START OF PDEFN(PLANDF-IDX)
SET OPTIONS-FOUND-NO OF W-SW TO TRUE

PERFORM VARYING OPTNDF-IDX FROM OPTNDF-IDX BY 1
UNTIL OPTNDF-IDX > OPTNDF-END OF PDEFN(PLANDF-IDX)

IF ((BENEFIT-PLAN OF WPLAN-DATA OF
W-EVENT(WPLAN-IDX)
= BENEFIT-PLAN OF ODEFN(OPTNDF-IDX))
AND OPTN-FOUND-YES OF W-SW)
OR OPTN-FOUND-NO OF W-SW

IF PLAN-TYPE-HEALTH OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

PERFORM LS112-COMPARE-COVRG-CD

ELSE

SET VALID-OPTION-YES OF W-SW TO TRUE

SET OPTIONS-FOUND-YES OF W-SW TO TRUE

END-IF

IF VALID-OPTION-YES OF W-SW

PERFORM LS300-INIT-PARTIC-OPTN

MOVE PLAN-TYPE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

MOVE BENEFIT-PLAN OF ODEFN(OPTNDF-IDX)

TO BENEFIT-PLAN OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

MOVE COVRG-CD OF ODEFN(OPTNDF-IDX)

TO COVRG-CD OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

MOVE OPTION-CD OF ODEFN(OPTNDF-IDX)

TO OPTION-CD OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

MOVE OPTION-ID OF ODEFN(OPTNDF-IDX)

TO OPTION-ID OF W-EVENT(WOPTN-IDX)

END-IF

END-IF

END-PERFORM

IF OPTIONS-FOUND-NO OF W-SW

SET ERROR-YES OF W-SW TO TRUE

SET MSGID-BENEFIT-OPTS-NOT-FOUND OF PYMSG TO TRUE

END-IF

END-IF

.

BUILD-BASE-OPTN-SET-EXIT.

/*****

* *

LS111-DEFINE-CUR-COVRG-CD SECTION.

LS111.

* *

SET VALID-COVRG-CD-NO OF W-SW TO TRUE

SET WCVGCD-IDX TO 1

SEARCH WCVGCD-DATA

AT END

CONTINUE

WHEN WCVGCD-IDX > WCVGCD-COUNT OF W-CVGCD

CONTINUE

WHEN COVRG-CD OF W-CVGCD(WCVGCD-IDX)

= COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

SET VALID-COVRG-CD-YES OF W-SW TO TRUE

MOVE CBR-COVRG-SET OF W-CVGCD(WCVGCD-IDX)

TO CBR-COVRG-SET OF W-EVENT(WPLAN-IDX)

MOVE MIN-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

TO MIN-NUM-OF-DEPS OF W-EVENT(WPLAN-IDX)

MOVE MAX-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

TO MAX-NUM-OF-DEPS OF W-EVENT(WPLAN-IDX)

MOVE SPOUSE-COVERAGE OF W-CVGCD(WCVGCD-IDX)

TO SPOUSE-COVERAGE OF W-EVENT(WPLAN-IDX)

END-SEARCH

IF VALID-COVRG-CD-NO OF W-SW

SET MSGID-COVG-CODE-INVALID OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

.

DEFINE-CUR-COVRG-CD-EXIT.

/*****

* *

LS112-COMPARE-COVRG-CD SECTION.

LS112.

* *

SET VALID-OPTION-NO OF W-SW TO TRUE

SET WCVGCD-IDX TO 1

SEARCH WCVGCD-DATA

AT END

CONTINUE

WHEN WCVGCD-IDX > WCVGCD-COUNT OF W-CVGCD

CONTINUE

WHEN COVRG-CD OF W-CVGCD(WCVGCD-IDX)

= COVRG-CD OF ODEFN(OPTNDF-IDX)

IF (SPOUSE-COVERAGE-NOT-ALLOWED OF W-EVENT(WPLAN-IDX)

AND NOT SPOUSE-COVERAGE-NOT-ALLOWED

OF W-CVGCD(WCVGCD-IDX))

OR (SPOUSE-COVERAGE-ALLOWED OF W-EVENT(WPLAN-IDX)

AND (SPOUSE-COVERAGE-REQUIRED

OF W-CVGCD(WCVGCD-IDX))

AND (SPOUSE OF W-DEPEND(WDEPEFF-IDX)

OR SAME-SEX-SPOUSE

OF W-DEPEND(WDEPEFF-IDX)

OR EX-SPOUSE

OF W-DEPEND(WDEPEFF-IDX)))

OR (SPOUSE-COVERAGE-REQUIRED

OF W-EVENT(WPLAN-IDX)

AND SPOUSE-COVERAGE-REQUIRED

OF W-CVGCD(WCVGCD-IDX)

AND (SPOUSE OF W-DEPEND(WDEPEFF-IDX)

OR SAME-SEX-SPOUSE

OF W-DEPEND(WDEPEFF-IDX)

OR EX-SPOUSE

OF W-DEPEND(WDEPEFF-IDX)))

OR (EMPLOYEE OF W-DEPEND(WDEPEFF-IDX)
 AND (SPOUSE-COVERAGE-SPOUSE-ONLY
 OF W-CVGCD(WCVGCD-IDX)
 OR SPOUSE-COVERAGE-DEPEND-ONLY
 OF W-CVGCD(WCVGCD-IDX)
 OR SPOUSE-COVERAGE-SPSDEP-ONLY
 OF W-CVGCD(WCVGCD-IDX)))
 OR ((SPOUSE OF W-DEPEND(WDEPEFF-IDX)
 OR SAME-SEX-SPOUSE
 OF W-DEPEND(WDEPEFF-IDX)
 OR EX-SPOUSE
 OF W-DEPEND(WDEPEFF-IDX))
 AND OTHR-DEP-IN-CVGCD-NO
 OF W-CVGCD(WCVGCD-IDX))
 OR (MAX-NUM-OF-DEPS OF W-EVENT(WPLAN-IDX)
 < MAX-NUM-OF-DEPS
 OF W-CVGCD(WCVGCD-IDX))
 OR (SPOUSE-COVERAGE-REQUIRED
 OF W-EVENT(WPLAN-IDX)
 AND MAX-NUM-OF-DEPS OF W-EVENT(WPLAN-IDX)
 = 1
 AND NOT SPOUSE-COVERAGE-REQUIRED
 OF W-CVGCD(WCVGCD-IDX)
 AND MAX-NUM-OF-DEPS
 OF W-CVGCD(WCVGCD-IDX) = 1)

SET VALID-OPTION-NO OF W-SW TO TRUE

ELSE

IF OPTN-FOUND-YES OF W-SW

IF CBR-COVRG-SET OF W-CVGCD(WCVGCD-IDX)

= CBR-COVRG-SET OF W-EVENT(WPLAN-IDX)

SET VALID-OPTION-YES OF W-SW TO TRUE

SET OPTIONS-FOUND-YES OF W-SW TO TRUE

ELSE

SET VALID-OPTION-NO OF W-SW TO TRUE

END-IF

ELSE

SET VALID-OPTION-YES OF W-SW TO TRUE

SET OPTIONS-FOUND-YES OF W-SW TO TRUE

END-IF

END-IF

END-SEARCH

.

COMPARE-COVRG-CD-EXIT.

/*****

*

*

LS120-BUILD-BASE-OPTN-ONLY SECTION.

LS120.

*

*

*SEARCH COVERAGE CODE TABLE TO IDENTIFY THE CURRENT COVERAGE CODE

*SET FOR THE CURRENT ENROLLMENT COVERAGE CODE

PERFORM LS122-DEFINE-CUR-COVR-SET

IF ERROR-NO OF W-SW

SET WCVGCD-IDX TO 1

*SEARCH COVERAGE CODE TABLE AGAIN TO IDENTIFY AN EMPLOYEE ONLY

*COVERAGE CODE WITH A COVERAGE CODE SET THAT MATCHES THE CURRENT

*ENROLLMENT COVERAGE CODE SET. IF NO MATCH ON COVERAGE SET IS

*FOUND THEN THE OPTION FOUND SWITCH IS SET TO NO FOR PROCESSING

*FURTHER DOWN STREAM IN THIS PARAGRAPH.

SEARCH WCVGCD-DATA

AT END

SET OPTN-FOUND-NO OF W-SW TO TRUE

WHEN WCVGCD-IDX > WCVGCD-COUNT OF W-CVGCD

SET OPTN-FOUND-NO OF W-SW TO TRUE

WHEN MIN-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX) = ZERO

AND CBR-COVRG-SET OF W-CVGCD(WCVGCD-IDX)
= CBR-COVRG-SET OF W-EVENT(WPLAN-IDX)

MOVE COVRG-CD OF W-CVGCD(WCVGCD-IDX)
TO COVRG-CD OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

PERFORM LS121-CHECK-BENOPTN-MATCH

END-SEARCH

SET PLANDF-IDX TO 1

SEARCH PLANDF-DATA

AT END

DISPLAY

'Cobra Plan Not Defined in Pgm/PlanType/Effdt: '

BENEFIT-PROGRAM OF PGMDF-DATA

'/'

PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

/'

EFFDT OF PG MDF-DATA

DISPLAY

'Emplid: ' EMPLID OF W-EVENT

MOVE 'BUILD-BASE-OPTION-ONLY'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

WHEN PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

= PLAN-TYPE OF PLANDF-DATA OF PDEFN(PLANDF-IDX)

CONTINUE

END-SEARCH

SET OPTNDF-IDX TO OPTNDF-START OF PDEFN(PLANDF-IDX)

SET OPTIONS-FOUND-NO OF W-SW TO TRUE

PERFORM VARYING OPTNDF-IDX FROM OPTNDF-IDX BY 1

UNTIL OPTNDF-IDX > OPTNDF-END OF PDEFN(PLANDF-IDX)

*IF BENEFIT PLANS DO NOT MATCH HERE AND WE KNOW WE HAVE FOUND A

*MATCH PREVIOUSLY WHERE BENEFIT PLAN, COVERAGE CODE AND COVERAGE

*SET OF THE ENROLLMENT MATCHED ON THE PLAN DEFINITION TABL

*THEN SKIP THIS PLAN

IF ((BENEFIT-PLAN OF WPLAN-DATA OF
W-EVENT(WPLAN-IDX)
= BENEFIT-PLAN OF ODEFN(OPTNDF-IDX))
AND OPTN-FOUND-YES OF W-SW)
OR OPTN-FOUND-NO OF W-SW

PERFORM LS123-FIND-EMPLOYEE-ONLY-COVRG

IF VALID-OPTION-YES OF W-SW

PERFORM LS300-INIT-PARTIC-OPTN
MOVE PLAN-TYPE OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
TO PLAN-TYPE OF WOPTN-DATA
OF W-EVENT(WOPTN-IDX)
MOVE BENEFIT-PLAN OF ODEFN(OPTNDF-IDX)
TO BENEFIT-PLAN OF WOPTN-DATA
OF W-EVENT(WOPTN-IDX)
MOVE COVRG-CD OF ODEFN(OPTNDF-IDX)
TO COVRG-CD OF WOPTN-DATA
OF W-EVENT(WOPTN-IDX)
MOVE OPTION-CD OF ODEFN(OPTNDF-IDX)
TO OPTION-CD OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

MOVE OPTION-ID OF ODEFN(OPTNDF-IDX)

TO OPTION-ID OF W-EVENT(WOPTN-IDX)

END-IF

END-IF

END-PERFORM

IF OPTIONS-FOUND-NO OF W-SW

SET ERROR-YES OF W-SW TO TRUE

SET MSGID-EMPL-ONLY-CVG-NOT-FOUND

OF PYMSG TO TRUE

END-IF

END-IF

.

BUILD-BASE-OPTN-ONLY-EXIT.

/*****

*

*

LS121-CHECK-BENOPTN-MATCH SECTION.

LS121.

*

*

SET OPTN-FOUND-NO OF W-SW TO TRUE

SET PLANDF-IDX TO 1

SEARCH PLANDF-DATA

AT END

DISPLAY

'Cobra Plan Not Defined in Pgm/PlanType/Effdt: '

BENEFIT-PROGRAM OF PGMDF-DATA

/'

PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

/'

EFFDT OF PGMDF-DATA

DISPLAY

'Emplid: ' EMPLID OF W-EVENT

MOVE 'CHECK-BENOPTN-MATCH'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

WHEN PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

= PLAN-TYPE OF PLANDF-DATA OF PDEFN(PLANDF-IDX)

CONTINUE

END-SEARCH

SET OPTNDF-IDX TO OPTNDF-START OF PDEFN(PLANDF-IDX)

SEARCH OPTNDF-DATA

AT END

CONTINUE

WHEN OPTNDF-IDX > OPTNDF-END OF PDEFN(PLANDF-IDX)

CONTINUE

WHEN BENEFIT-PLAN OF OPTNDF-DATA(OPTNDF-IDX)

= BENEFIT-PLAN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

AND COVRG-CD OF OPTNDF-DATA(OPTNDF-IDX)

= COVRG-CD OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

SET OPTN-FOUND-YES OF W-SW TO TRUE

END-SEARCH

CHECK-BENOPTN-MATCH-EXIT.

/*****

* *

LS122-DEFINE-CUR-COVR-SET SECTION.

LS122.

* *

SET WCVGCD-IDX TO 1

SEARCH WCVGCD-DATA

AT END

SET VALID-COVRG-CD-NO OF W-SW TO TRUE

WHEN WCVGCD-IDX > WCVGCD-COUNT OF W-CVGCD

SET VALID-COVRG-CD-NO OF W-SW TO TRUE

WHEN COVRG-CD OF W-CVGCD(WCVGCD-IDX)

= COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

SET VALID-COVRG-CD-YES OF W-SW TO TRUE

MOVE CBR-COVRG-SET OF W-CVGCD(WCVGCD-IDX)
TO CBR-COVRG-SET OF W-EVENT(WPLAN-IDX)

END-SEARCH

IF VALID-COVRG-CD-NO OF W-SW

SET MSGID-COVRG-CODE-INVALID OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

.

DEFINE-CUR-COVR-SET-EXIT.

/*****

* *

LS123-FIND-EMPLOYEE-ONLY-COVRG SECTION.

LS123.

* *

SET WCVGCD-IDX TO 1

SEARCH WCVGCD-DATA

AT END

CONTINUE

WHEN WCVGCD-IDX > WCVGCD-COUNT OF W-CVGCD

CONTINUE

WHEN COVRG-CD OF W-CVGCD(WCVGCD-IDX)
= COVRG-CD OF ODEFN(OPTNDF-IDX)

IF MIN-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX) = ZERO

IF OPTN-FOUND-YES OF W-SW

IF CBR-COVRG-SET OF W-CVGCD(WCVGCD-IDX)
= CBR-COVRG-SET OF W-EVENT(WPLAN-IDX)

SET VALID-OPTION-YES OF W-SW TO TRUE

SET OPTIONS-FOUND-YES OF W-SW TO TRUE

ELSE

SET VALID-OPTION-NO OF W-SW TO TRUE

END-IF

ELSE

SET VALID-OPTION-YES OF W-SW TO TRUE

SET OPTIONS-FOUND-YES OF W-SW TO TRUE

END-IF

ELSE

SET VALID-OPTION-NO OF W-SW TO TRUE

END-IF

END-SEARCH

.

FIND-EMPLOYEE-ONLY-COVRG-EXIT.

/*****

* *

LS200-PREPARE-BENADMIN-OPTN SECTION.

LS200.

* *

MOVE ZERO TO COBRA-COUNT OF PARTC

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

IF COBRA-COUNT-MAX OF PARTC

DISPLAY 'Max # of COBRA Plan Exceeded'

MOVE 'PREPARE-BENADMIN-OPTN'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO COBRA-COUNT OF PARTC

SET COBRA-IDX TO COBRA-COUNT OF PARTC

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF COBRA-DATA

OF PARTC(COBRA-IDX)

END-IF

END-PERFORM

SET BAS-TYPE-COBRA OF ADMIN TO TRUE

SET PROCESS-OPTION-OTHERS OF ADMIN TO TRUE

CALL 'PSPBAOPT' USING SQLRT

ADMIN

PARTC

POPTN

CRELT

EHIST

BATBL

BADED

PDEFN

ODEFN

CALTB

BCLTB

GEOTB

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid: ' EMPLID OF W-EVENT

MOVE 'PREPARE-OPTIONS(PSPBAOPT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

PERFORM LS210-COMPARE-BENADMIN-ELIG

IF BENADMIN-ELIG-OPTN-NO OF W-SW

SET MSGID-BAS-ELIG-OPTN-NONE OF PYMSG TO TRUE

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF PYMSG

MOVE DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

TO DEPENDENT-BENEF OF PYMSG

MOVE PLAN-TYPE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO MSGDATA1 OF PYMSG

PERFORM ZM000-MESSAGE

PERFORM LS220-SET-BASE-OPTN

ELSE

IF BASE-OPTN-MATCH-NO OF W-SW

SET BENADMIN-ELIG-CVGCD-NO OF W-SW

TO TRUE

PERFORM LS230-SET-BENADMIN-OPTN

IF BENADMIN-ELIG-CVGCD-NO OF W-SW

SET MSGID-BAS-ELIG-OPTN-NONE OF PYMSG

TO TRUE

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF PYMSG
MOVE DEPENDENT-BENEF OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
TO DEPENDENT-BENEF
OF PYMSG
MOVE PLAN-TYPE OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
TO MSGDATA1 OF PYMSG
PERFORM ZM000-MESSAGE
PERFORM LS220-SET-BASE-OPTN
END-IF
END-IF
END-IF
END-IF
END-PERFORM

.
PREPARE-BENADMIN-OPTN-EXIT.

/******
* *
LS210-COMPARE-BENADMIN-ELIG SECTION.
LS210.
* *

SET BENADMIN-ELIG-OPTN-NO OF W-SW TO TRUE

SET PLAN-IDX TO 1

SEARCH PLAN-DATA OF PARTC

AT END

CONTINUE

WHEN PLAN-IDX > PLAN-COUNT OF PARTC

CONTINUE

WHEN PLAN-TYPE OF PLAN-DATA OF PARTC(PPLAN-IDX)

= PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

SET BENADMIN-ELIG-OPTN-YES OF W-SW TO TRUE

SET BASE-OPTN-MATCH-NO OF W-SW TO TRUE

SET OPTN-IDX TO 1

SEARCH OPTN-DATA

AT END

CONTINUE

WHEN OPTN-IDX > OPTN-COUNT OF POPTN

CONTINUE

WHEN PLAN-TYPE OF POPTN(OPTN-IDX)

= PLAN-TYPE OF PLAN-DATA

OF PARTC(PLAN-IDX)

PERFORM VARYING OPTN-IDX FROM OPTN-IDX BY 1

UNTIL PLAN-TYPE OF OPTN-DATA

OF POPTN(OPTN-IDX)

NOT = PLAN-TYPE OF PLAN-DATA

OF PARTC(PLAN-IDX)

OR OPTN-IDX

> OPTN-COUNT OF POPTN

PERFORM LS211-CHECK-BASE-OPTN-MATCH

END-PERFORM

END-SEARCH

END-SEARCH

.

PREPARE-BENADMIN-ELIG-EXIT.

/*****

*

*

LS211-CHECK-BASE-OPTN-MATCH SECTION.

LS211.

*

*

SET WOPTN-IDX TO 1

SEARCH WOPTN-DATA

AT END

CONTINUE

WHEN WOPTN-IDX > WOPTN-COUNT OF W-EVENT

CONTINUE

WHEN PLAN-TYPE OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

= PLAN-TYPE OF POPTN(OPTN-IDX)

AND BENEFIT-PLAN OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

= BENEFIT-PLAN

OF POPTN(OPTN-IDX)

AND COVRG-CD OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

= COVRG-CD OF POPTN(OPTN-IDX)

PERFORM LS232-GET-MONTHLY-COST

SET INSERT-YES OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

TO TRUE

SET BASE-OPTN-MATCH-YES OF W-SW TO TRUE

END-SEARCH

.

CHECK-BASE-OPTN-MATCH-EXIT.

/*****

*

*

LS220-SET-BASE-OPTN SECTION.

LS220.

*

*

SET WOPTN-IDX TO 1

SEARCH WOPTN-DATA

AT END

CONTINUE

WHEN WOPTN-IDX > WOPTN-COUNT OF W-EVENT

CONTINUE

WHEN PLAN-TYPE OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
= PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM VARYING WOPTN-IDX FROM WOPTN-IDX BY 1

UNTIL PLAN-TYPE

OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

NOT = PLAN-TYPE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

OR WOPTN-IDX

> WOPTN-COUNT OF W-EVENT

PERFORM LS221-CALC-MONTHLY-COST

SET INSERT-YES OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX) TO TRUE

END-PERFORM

END-SEARCH

.

SET-BASE-OPTN-EXIT.

/*****

* *

LS221-CALC-MONTHLY-COST SECTION.

LS221.

* *

INITIALIZE DCALC-DATA OF BADED

MOVE PLAN-TYPE OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
TO PLAN-TYPE OF BADED

MOVE OPTION-ID OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
TO OPTION-ID OF BADED

MOVE BENEFIT-PLAN OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
TO BENEFIT-PLAN OF BADED

MOVE COVRG-CD OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
TO COVRG-CD OF BADED

SET RATE-QUALIFIER-TOTAL OF BADED TO TRUE

MOVE CBR-RATE-PCT OF PARTC TO CBR-RATE-PCT OF BADED

MOVE CBR-DIS-RATE-PCT OF PARTC TO CBR-DIS-RATE-PCT OF BADED

MOVE COBRA-EVENT-DT OF W-EVENT TO EVENT-DT OF PARTC

SET ERROR-NO OF W-SW TO TRUE

MOVE ZERO TO CBR-MM-COST OF WOPTN-DATA
OF W-EVENT(WOPTN-IDX)

MOVE ZERO TO CBR-DISABL-MM-COST OF WOPTN-DATA
OF W-EVENT(WOPTN-IDX)

PERFORM LS225-SET-COVERED-PERSONS

SET PLANDF-IDX TO 1

SEARCH PLANDF-DATA

AT END

DISPLAY

'Cobra Plan Not Defined in Pgm/PlanType/Effdt: '

BENEFIT-PROGRAM OF PG MDF-DATA

/'

PLAN-TYPE OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

/'

EFFDT OF PG MDF-DATA

DISPLAY

'Emplid: ' EMPLID OF W-EVENT

MOVE 'CALC-MONTHLY-COST'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

WHEN PLAN-TYPE OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

= PLAN-TYPE OF PLANDF-DATA OF PDEFN(PLANDF-IDX)

CONTINUE

END-SEARCH

SET OPTNDF-IDX TO OPTNDF-START OF PDEFN(PLANDF-IDX)

SEARCH OPTNDF-DATA

WHEN BENEFIT-PLAN OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

= BENEFIT-PLAN OF ODEFN(OPTNDF-IDX)

AND COVRG-CD OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

= COVRG-CD OF ODEFN(OPTNDF-IDX)

CONTINUE

END-SEARCH

PERFORM VARYING COSTDF-IDX

FROM COSTDF-START OF ODEFN(OPTNDF-IDX) BY 1

UNTIL COSTDF-IDX > COSTDF-END OF ODEFN(OPTNDF-IDX)

OR ERROR-YES OF W-SW

MOVE COST-ID OF PDEFN(COSTDF-IDX) TO COST-ID OF BADED

CALL 'PSPBADED' USING SQLRT

ADMIN

BADED

PARTC

IF RTNCD-ERROR OF SQLRT

MOVE 'CALC-MONTHLY-COST(PSPBADED)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ADD DEDN-AMT-B-TAX OF BADED

TO CBR-MM-COST OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

ADD DEDN-AMT-A-TAX OF BADED

TO CBR-MM-COST OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

ADD DEDN-AMT-NT-TAX OF BADED

TO CBR-MM-COST OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

ADD CBR-DIS-DEDUCT-AMT OF BADED

TO CBR-DISABL-MM-COST OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

END-PERFORM

.

CALC-MONTHLY-COST-EXIT.

/*****

* *

LS225-SET-COVERED-PERSONS SECTION.

LS225.

* *

*

* Load the set of potential covered dependents so they are
* available to the DedCalc routines, since the rate may require
* individual biographics.

*

MOVE ZERO TO COVERED-DPND-CNT OF BADED

IF PLAN-TYPE-HEALTH OF BADED

* Get Coverage Code for which cost is being determined...

SET WCVGCD-IDX TO 1

SEARCH WCVGCD-DATA

WHEN COVRG-CD OF W-CVGCD(WCVGCD-IDX)

= COVRG-CD OF BADED

CONTINUE

END-SEARCH

- * Step through each potentially coverable dependent...

PERFORM VARYING WCURDB-IDX FROM 1 BY 1

UNTIL WCURDB-IDX > WCURDB-COUNT OF W-CURELT

OR COVERED-DPND-CNT-MAX OF BADED

- * Get the dependents biographics...

SET WDEPEND-IDX TO 1

SEARCH WDEPEND-DATA

WHEN DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

= DEPENDENT-BENEF OF W-CURELT(WCURDB-IDX)

SET WDEPEFF-IDX

TO WDEPEFF-START

OF W-DEPEND(WDEPEND-IDX)

END-SEARCH

- * Filter dependent against current coverage code...

IF (EMPLOYEE OF W-DEPEND(WDEPEFF-IDX)

AND NOT (SPOUSE-COVERAGE-SPOUSE-ONLY

OF W-CVGCD(WCVGCD-IDX)

OR SPOUSE-COVERAGE-SPSDEP-ONLY

OF W-CVGCD(WCVGCD-IDX)))

OR ((SPOUSE OF W-DEPEND(WDEPEFF-IDX)
 OR SAME-SEX-SPOUSE OF W-DEPEND(WDEPEFF-IDX)
 OR EX-SPOUSE OF W-DEPEND(WDEPEFF-IDX))
 AND (SPOUSE-COVERAGE-ALLOWED
 OF W-CVGCD(WCVGCD-IDX)
 OR SPOUSE-COVERAGE-REQUIRED
 OF W-CVGCD(WCVGCD-IDX)
 OR SPOUSE-COVERAGE-SPOUSE-ONLY
 OF W-CVGCD(WCVGCD-IDX)
 OR SPOUSE-COVERAGE-SPSDEP-ONLY
 OF W-CVGCD(WCVGCD-IDX)))
 OR (CHILD OF W-DEPEND(WDEPEFF-IDX)
 AND (MAX-NUM-OF-DEPS
 OF W-CVGCD(WCVGCD-IDX) > 1
 OR (MAX-NUM-OF-DEPS
 OF W-CVGCD(WCVGCD-IDX) = 1
 AND NOT (SPOUSE-COVERAGE-SPOUSE-ONLY
 OF W-CVGCD(WCVGCD-IDX)
 OR SPOUSE-COVERAGE-REQUIRED
 OF W-CVGCD(WCVGCD-IDX))))))

* Add dependent to list...

IF COVERED-DPND-CNT-MAX OF BADED

SET MSGID-MAX-CVD-PERS-EXCEEDED OF PYMSG
 TO TRUE

MOVE PLAN-TYPE OF BADED
TO MSGDATA1 OF PYMSG
MOVE BENEFIT-PLAN OF BADED
TO MSGDATA2 OF PYMSG
MOVE COVERED-DPND-CNT OF BADED
TO MSGDATA3-INT OF PYMSG
PERFORM ZZ000-SQL-ERROR
END-IF

ADD 1 TO COVERED-DPND-CNT OF BADED
SET BADPND-IDX TO COVERED-DPND-CNT OF BADED

MOVE DEPENDENT-BENEF OF W-CURELT(WCURDB-IDX)
TO DEPENDENT-BENEF OF COVERED-DPND-CACHE
OF BADED(BADPND-IDX)
END-IF
END-PERFORM
END-IF

.

SET-COVERED-PERSONS-EXIT.

/*****

*

*

LS230-SET-BENADMIN-OPTN SECTION.

LS230.

*

*

SET OPTN-IDX TO 1

SEARCH OPTN-DATA

AT END

CONTINUE

WHEN OPTN-IDX > OPTN-COUNT OF POPTN

CONTINUE

WHEN PLAN-TYPE OF POPTN(OPTN-IDX)

= PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM VARYING OPTN-IDX FROM OPTN-IDX BY 1

UNTIL PLAN-TYPE

OF OPTN-DATA OF POPTN(OPTN-IDX)

NOT = PLAN-TYPE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

OR OPTN-IDX

> OPTN-COUNT OF POPTN

PERFORM LS231-CHECK-BASE-CVGCD-MATCH

IF BASE-CVGCD-MATCH-YES OF W-SW

PERFORM LS300-INIT-PARTIC-OPTN

MOVE PLAN-TYPE OF OPTN-DATA

OF POPTN(OPTN-IDX)

TO PLAN-TYPE OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

MOVE BENEFIT-PLAN OF POPTN(OPTN-IDX)

TO BENEFIT-PLAN OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

MOVE COVRG-CD OF POPTN(OPTN-IDX)

TO COVRG-CD OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

MOVE OPTION-CD OF POPTN(OPTN-IDX)

TO OPTION-CD OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

MOVE OPTION-ID OF POPTN(OPTN-IDX)

TO OPTION-ID OF W-EVENT(WOPTN-IDX)

PERFORM LS232-GET-MONTHLY-COST

SET INSERT-YES OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX) TO TRUE

END-IF

END-PERFORM

END-SEARCH

.

SET-BENADMIN-OPTN-EXIT.

/*****

*

*

LS231-CHECK-BASE-CVGCD-MATCH SECTION.

LS231.

*

*

SET BASE-CVGCD-MATCH-NO OF W-SW TO TRUE

SET WOPTN-IDX TO 1

SEARCH WOPTN-DATA

AT END

CONTINUE

WHEN WOPTN-IDX > WOPTN-COUNT OF W-EVENT

CONTINUE

WHEN PLAN-TYPE OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

= PLAN-TYPE OF POPTN(OPTN-IDX)

AND COVRG-CD OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

= COVRG-CD OF POPTN(OPTN-IDX)

SET BASE-CVGCD-MATCH-YES OF W-SW TO TRUE

SET BENADMIN-ELIG-CVGCD-YES OF W-SW TO TRUE

END-SEARCH

.

CHECK-BASE-CVGCD-MATCH-EXIT.

/*****

* *

LS232-GET-MONTHLY-COST SECTION.

LS232.

* *

SET COST-IDX TO 1

SEARCH COST-DATA

WHEN PLAN-TYPE OF COST-DATA OF PARTC(COST-IDX)
= PLAN-TYPE OF POPTN(OPTN-IDX)
AND OPTION-ID OF COST-DATA OF PARTC(COST-IDX)
= OPTION-ID OF POPTN(OPTN-IDX)

COMPUTE CBR-MM-COST OF WOPTN-DATA
OF W-EVENT(WOPTN-IDX)

= DEDN-AMT-B-TAX OF COST-DATA
OF PARTC(COST-IDX)

+ DEDN-AMT-A-TAX OF COST-DATA
OF PARTC(COST-IDX)

+ DEDN-AMT-NT-TAX OF COST-DATA
OF PARTC(COST-IDX)

MOVE CBR-DIS-DEDUCT-AMT OF COST-DATA
OF PARTC(COST-IDX)

TO CBR-DISABL-MM-COST OF WOPTN-DATA
OF W-EVENT(WOPTN-IDX)

END-SEARCH

.

GET-MONTHLY-COST-EXIT.

/*****

*

*

LS300-INIT-PARTIC-OPTN SECTION.

LS300.

*

*

IF WOPTN-COUNT-MAX OF W-EVENT

DISPLAY 'Maximum CBR-Partic-Optn Entries Exceeded'

MOVE 'INIT-PARTIC-OPTN' TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO WOPTN-COUNT OF W-EVENT

SET WOPTN-IDX TO WOPTN-COUNT OF W-EVENT

INITIALIZE WOPTN-DATA OF W-EVENT(WOPTN-IDX)

MOVE DEPENDENT-BENEF OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

TO DEPENDENT-BENEF OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

SET INSERT-NO OF WOPTN-DATA OF W-EVENT(WOPTN-IDX) TO TRUE

.

INIT-PARTIC-OPTN-EXIT.

/*****

*

*

LT000-PROCESS-ELECTION SECTION.

LT000.

*

*

SET STORE-HR-YES OF W-SW TO TRUE

SET STORE-BP-PGM-YES OF W-SW TO TRUE

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

AND (COBRA-ELECT-ELECT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

OR COBRA-ELECT-WAIVE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX))

IF COBRA-ELECT-WAIVE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

AND CBR-PROCESS-OPEN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

SET CBR-PROCESS-CLOSED OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX) TO TRUE

SET UPDATE-YES OF WPLAN-DATA

```
        OF W-EVENT(WPLAN-IDX) TO TRUE
    SET VALID-ELECTION-YES OF W-SW TO TRUE
ELSE
    IF COBRA-ELECT-ELECT OF WPLAN-DATA
        OF W-EVENT(WPLAN-IDX)
        AND CBR-PROCESS-OPEN OF WPLAN-DATA
            OF W-EVENT(WPLAN-IDX)
        AND CBR-ENROLL-NO OF WPLAN-DATA
            OF W-EVENT(WPLAN-IDX)

        PERFORM LT100-VALIDATE-ELECTION
    END-IF
END-IF

    IF VALID-ELECTION-YES OF W-SW

        PERFORM LT200-ENROLL-ELECTION
    ELSE
        SET COBRA-ERROR-YES OF WPLAN-DATA
            OF W-EVENT(WPLAN-IDX) TO TRUE
    END-IF
END-IF

END-PERFORM

PERFORM LT300-UPDATE-PARTIC-STATUS
SET RESET-EVENT-STATUS-YES OF W-SW TO TRUE
```

PROCESS-ELECTION-EXIT.

/*****

* *

LT100-VALIDATE-ELECTION SECTION.

LT100.

* *

PERFORM LT110-INIT-ELECTION

PERFORM LT120-VALIDATE-OPTION-CD

IF PLAN-TYPE-HEALTH OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

IF VALID-OPTION-CD-YES OF W-SW

PERFORM LT130-VALIDATE-DEPENDENT

IF VALID-DEPENDENT-YES OF W-SW

OR DEPENDENT-COUNT OF W-EVENT(WPLAN-IDX)

= ZERO

PERFORM LT140-VALIDATE-COVRG-CD

END-IF

END-IF

END-IF

.

VALIDATE-ELECTION-EXIT.

/*****

* *

LT110-INIT-ELECTION SECTION.

LT110.

* *

SET UPDATE-YES OF WPLAN-DATA OF W-EVENT(WPLAN-IDX) TO TRUE

SET INSERT-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX) TO TRUE

SET COBRA-ERROR-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO TRUE

MOVE SPACE

TO BENEFIT-PLAN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE SPACE

TO COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE ZERO TO DEPENDENT-COUNT OF W-EVENT(WPLAN-IDX)

SET DEP-SPOUSE-NO OF W-EVENT(WPLAN-IDX) TO TRUE

SET ERROR-NO OF W-SW TO TRUE

SET VALID-ELECTION-YES OF W-SW TO TRUE

.

INIT-ELECTION-EXIT.

/*****

* *

LT120-VALIDATE-OPTION-CD SECTION.

LT120.

* *

SET VALID-OPTION-CD-NO OF W-SW TO TRUE

SET ERROR-NO OF W-SW TO TRUE

IF OPTION-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX) = SPACE

SET MSGID-COBRA-ELECTION-MISSING OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

ELSE

SET WOPTN-IDX TO 1

SEARCH WOPTN-DATA

AT END

SET MSGID-OPTION-CD-INVALID OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

WHEN WOPTN-IDX > WOPTN-COUNT OF W-EVENT

SET MSGID-OPTION-CD-INVALID OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

WHEN PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

= PLAN-TYPE OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

AND OPTION-CD OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

= OPTION-CD OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

SET VALID-OPTION-CD-YES OF W-SW TO TRUE

MOVE BENEFIT-PLAN OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

TO BENEFIT-PLAN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COVRG-CD OF WOPTN-DATA

OF W-EVENT(WOPTN-IDX)

TO COVRG-CD OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

END-SEARCH

END-IF

IF ERROR-YES OF W-SW

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF PYMSG

MOVE DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

TO DEPENDENT-BENEF OF PYMSG

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO MSGDATA1 OF PYMSG

MOVE OPTION-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO MSGDATA2 OF PYMSG

PERFORM ZM000-MESSAGE

SET COBRA-ERROR-YES OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO TRUE

SET VALID-ELECTION-NO OF W-SW TO TRUE

END-IF

.

VALIDATE-OPTION-CD-EXIT.

/*****

*

*

LT130-VALIDATE-DEPENDENT SECTION.

LT130.

*

*

SET VALID-DEPENDENT-YES OF W-SW TO TRUE

PERFORM VARYING WDPND-IDX FROM 1 BY 1

UNTIL WDPND-IDX > WDPND-COUNT OF W-EVENT

IF PLAN-TYPE OF WDPND-DATA OF W-EVENT(WDPND-IDX)

= PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM XO900-SEARCH-DEP-RULE

ADD 1 TO DEPENDENT-COUNT OF W-EVENT(WPLAN-IDX)

PERFORM LT131-SRCH-DEPEND-BENEF

END-IF

END-PERFORM

.

VALIDATE-DEPENDENT-EXIT.

/*****

*

*

LT131-SRCH-DEPEND-BENEF SECTION.

LT131.

*

*

SET ERROR-NO OF W-SW TO TRUE

SET WDEPEND-IDX TO 1

SEARCH WDEPEND-DATA

AT END

SET MSGID-DPND-NOT-FOUND OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

WHEN WDEPEND-IDX > WDEPEND-COUNT OF W-DEPEND

SET MSGID-DPND-NOT-FOUND OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

WHEN COBRA-DEP-BENEF OF W-EVENT(WDPND-IDX)

= DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

MOVE COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

TO SEARCH-EFFDT OF W-DEPEND

PERFORM XN700-SRCH-DEP-EFF-DATA

IF SPOUSE OF W-DEPEND(WDEPEFF-IDX) OR
SAME-SEX-SPOUSE OF W-DEPEND(WDEPEFF-IDX)
SET DEP-SPOUSE-YES OF W-EVENT(WPLAN-IDX)
TO TRUE
END-IF

PERFORM LT132-VALIDATE-HTH-DEP

END-SEARCH

IF ERROR-YES OF W-SW

MOVE COBRA-EVENT-ID OF W-EVENT
TO COBRA-EVENT-ID OF PYMSG
MOVE DEPENDENT-BENEF OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
TO DEPENDENT-BENEF OF PYMSG
MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO MSGDATA1 OF PYMSG
MOVE OPTION-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO MSGDATA2 OF PYMSG
MOVE COBRA-DEP-BENEF OF W-EVENT(WDPND-IDX)
TO MSGDATA3 OF PYMSG
PERFORM ZM000-MESSAGE
SET UPDATE-YES OF WDPND-DATA OF W-EVENT(WDPND-IDX)
TO TRUE

SET COBRA-ERROR-YES OF WDPND-DATA OF W-EVENT(WDPND-IDX)
TO TRUE
SET VALID-DEPENDENT-NO OF W-SW TO TRUE
SET VALID-ELECTION-NO OF W-SW TO TRUE
ELSE

IF COBRA-ERROR-YES OF WDPND-DATA OF W-EVENT(WDPND-IDX)

SET UPDATE-YES OF WDPND-DATA OF W-EVENT(WDPND-IDX)
TO TRUE
SET COBRA-ERROR-NO OF WDPND-DATA
OF W-EVENT(WDPND-IDX) TO TRUE
END-IF
END-IF

.

SRCH-DEPEND-BENEF-EXIT.

/*****

* *

LT132-VALIDATE-HTH-DEP SECTION.

LT132.

* *

IF DT-OF-DEATH OF W-DEPEND (WDEPEND-IDX)

NOT = SPACE

AND DT-OF-DEATH OF W-DEPEND (WDEPEND-IDX)

<= COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

SET MSGID-DPND-DECEASED OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

IF EX-SPOUSE OF W-DEPEND (WDEPEFF-IDX)

SET MSGID-DPND-EX-SPOUSE OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

IF MARRIED-DEP-NOT-ALLOWED OF W-DEPRULE

AND MARRIED OF W-DEPEND(WDEPEFF-IDX)

AND NOT SPOUSE OF W-DEPEND(WDEPEFF-IDX)

AND NOT SAME-SEX-SPOUSE OF W-DEPEND(WDEPEFF-IDX)

SET MSGID-DPND-MARRIED OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

IF ((DISABLED-YES OF W-DEPEND(WDEPEFF-IDX)

AND EXCL-DISABLED-NO OF W-DEPRULE)

OR DISABLED-NO

OF W-DEPEND(WDEPEFF-IDX))

AND CHILD OF W-DEPEND(WDEPEFF-IDX)

MOVE COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

TO AGE-AS-OF-DT OF W-CNTL

PERFORM XQ000-DEPENDENT-AGE-CHECK

IF OVERAGE-S-YES OF W-SW

SET MSGID-DPND-OVERAGE-S OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

IF OVERAGE-NS-YES OF W-SW

SET MSGID-DPND-OVERAGE-NS OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

END-IF

.

VALIDATE-HEALTH-DPND-EXIT.

/*****

*

*

LT140-VALIDATE-COVRG-CD SECTION.

LT140.

*

*

SET ERROR-NO OF W-SW TO TRUE

SET WCVGCD-IDX TO 1

SEARCH WCVGCD-DATA

AT END

SET MSGID-CVGCD-NOT-FOUND OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

WHEN WCVGCD-IDX > WCVGCD-COUNT OF W-CVGCD

SET MSGID-CVGCD-NOT-FOUND OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

WHEN COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

= COVRG-CD OF W-CVGCD(WCVGCD-IDX)

IF DEP-SPOUSE-NO OF W-EVENT(WPLAN-IDX)

AND SPOUSE-COVERAGE-REQUIRED

OF W-CVGCD(WCVGCD-IDX)

SET MSGID-SPOUSE-REQD OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

IF CC-CTL-TOTAL-YES OF W-CVGCD(WCVGCD-IDX)

IF DEPENDENT-COUNT OF W-EVENT(WPLAN-IDX)

> MAX-NUM-TOTAL OF W-CVGCD(WCVGCD-IDX)

SET MSGID-MAX-DPND OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

IF DEPENDENT-COUNT OF W-EVENT(WPLAN-IDX)

< MIN-NUM-TOTAL OF W-CVGCD(WCVGCD-IDX)

SET MSGID-MIN-DPND OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

ELSE

IF DEPENDENT-COUNT OF W-EVENT(WPLAN-IDX)

> MAX-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

SET MSGID-MAX-DPND OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

IF DEPENDENT-COUNT OF W-EVENT(WPLAN-IDX)

< MIN-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

SET MSGID-MIN-DPND OF PYMSG TO TRUE

SET ERROR-YES OF W-SW TO TRUE

END-IF

END-IF

END-SEARCH

IF ERROR-YES OF W-SW

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF PYMSG

MOVE DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

TO DEPENDENT-BENEF OF PYMSG

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO MSGDATA1 OF PYMSG

MOVE OPTION-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO MSGDATA2 OF PYMSG

MOVE COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO MSGDATA3 OF PYMSG

PERFORM ZM000-MESSAGE

SET COBRA-ERROR-YES OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO TRUE

SET VALID-ELECTION-NO OF W-SW TO TRUE

END-IF

.

VALIDATE-COVRG-CD-EXIT.

/*****

* *

LT200-ENROLL-ELECTION SECTION.

LT200.

* *

IF STORE-HR-YES OF W-SW

PERFORM LT210-STORE-HR-DATA

IF WPLAN-IDX = WPLAN-COUNT OF W-EVENT

SET STORE-HR-NO OF W-SW TO TRUE

END-IF

END-IF

IF STORE-BP-PGM-YES OF W-SW

IF COBRA-EMPLID OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

NOT = SPACE

PERFORM LT230-STORE-BP-PGM-DATA

END-IF

IF WPLAN-IDX = WPLAN-COUNT OF W-EVENT

SET STORE-BP-PGM-NO OF W-SW TO TRUE

END-IF

END-IF

IF COBRA-EMPLID OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

NOT = SPACE

IF PLAN-TYPE-HEALTH OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM LT240-STORE-HTH-DATA

ELSE

PERFORM LT250-STORE-FSA-DATA

END-IF

END-IF

IF COBRA-ELECT-ELECT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

SET CBR-ENROLLED OF W-EVENT(WPLAN-IDX) TO TRUE

ELSE

IF COBRA-ELECT-WAIVE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

SET CBR-ENROLL-NO OF W-EVENT(WPLAN-IDX) TO TRUE

END-IF

END-IF

SET CBR-PROCESS-CLOSED OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO TRUE

.

ENROLL-ELECTION-EXIT.

/*****

* *

LT210-STORE-HR-DATA SECTION.

LT210.

* *

PERFORM XL100-GET-EMPL-HR-DATA

SET WDEPEND-IDX TO 1

SEARCH WDEPEND-DATA

WHEN DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

= DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

CONTINUE

END-SEARCH

MOVE COBRA-EVENT-DT OF W-EVENT TO SEARCH-EFFDT OF W-DEPEND

PERFORM XN700-SRCH-DEP-EFF-DATA

IF COBRA-ELECT-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

IF COBRA-EMPLID OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

= SPACE

PERFORM LT211-GET-COBRA-EMPLID-LAST

PERFORM LT212-ASSIGN-COBRA-EMPLID

SET CBR-NONEE-CREATE-ALL OF W-EVENT TO TRUE

PERFORM LT223-INSERT-NONEE-TRIGGER

ELSE

IF NOT EMPLOYEE OF W-DEPEND(WDEPEFF-IDX)

SET CBR-NONEE-CREATE-JOB OF W-EVENT TO TRUE

PERFORM LT223-INSERT-NONEE-TRIGGER

END-IF

END-IF

END-IF

.

STORE-HR-DATA-EXIT.

/*****

*

*

LT211-GET-COBRA-EMPLID-LAST SECTION.

LT211.

*

*

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-INSTALL

BIND-SETUP OF S-INSTALL

BIND-DATA OF S-INSTALL

SELECT-SETUP OF S-INSTALL

SELECT-DATA OF S-INSTALL

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-COBRA-EMPLID-LAST(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-INSTALL

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-COBRA-EMPLID-LAST(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

MOVE COBRA-EMPLID-LAST OF S-INSTALL

TO COBRA-EMPLID-LAST OF W-CNTL

ADD 1 TO COBRA-EMPLID-LAST OF W-CNTL

MOVE COBRA-EMPLID-LAST OF W-CNTL

TO COBRA-EMPLID-LAST OF U-INSTALL

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF U-INSTALL

BIND-SETUP OF U-INSTALL

BIND-DATA OF U-INSTALL

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-COBRA-EMPLID-LAST(UPDATE)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

GET-COBRA-EMPLID-LAST-EXIT.

/*****

*

*

LT212-ASSIGN-COBRA-EMPLID SECTION.

LT212.

*

*

MOVE COBRA-EMPLID OF W-CNTL

TO COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-EMPLID OF W-CNTL

TO COBRA-EMPLID OF W-DEPEND(WDEPEND-IDX)

PERFORM XT000-UPDATE-DEP-BEN

.

ASSIGN-COBRA-EMPLID-EXIT.

/*****

* *

LT214C-GET-EMPL-ADDRESS SECTION.

LT214C.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF S-ADDRESS

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT-1 OF S-ADDRESS

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT-2 OF S-ADDRESS

MOVE ADDRESS-TYPE OF W-DEPEND(WDEPEND-IDX)

TO ADDRESS-TYPE OF S-ADDRESS

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-ADDRESS OF W-CNTL

SQL-STMT OF S-ADDRESS

BIND-SETUP OF S-ADDRESS

BIND-DATA OF S-ADDRESS

SELECT-SETUP OF S-ADDRESS

SELECT-DATA OF S-ADDRESS

IF RTNCD-ERROR OF SQLRT

```
MOVE 'GET-ADDRESS-DATA(SELECT)'  
    TO ERR-SECTION OF SQLRT  
PERFORM ZZ000-SQL-ERROR  
END-IF
```

```
INITIALIZE SELECT-DATA OF S-ADDRESS
```

```
CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT  
    SQLRT  
    SQL-CURSOR OF S-ADDRESS OF W-CNTL  
IF RTNCD-ERROR OF SQLRT
```

```
IF RTNCD-END OF SQLRT
```

```
SET RTNCD-OK OF SQLRT TO TRUE  
SET MSGID-ADDRESS-NOT-FOUND OF PYMSG  
    TO TRUE
```

```
MOVE EFFDT-1 OF S-ADDRESS  
    TO MSGDATA1 OF PYMSG  
MOVE ADDRESS-TYPE OF S-ADDRESS  
    TO MSGDATA2 OF PYMSG
```

```
PERFORM ZM000-MESSAGE
```

```
* MOVE 'USA'
```

* TO COUNTRY OF EMPL-ADDRESS-SBR OF I-PERADDR

ELSE

DISPLAY 'Emplid/CBREvtID: '

EMPLID OF S-ADDRESS '/'

EFFDT-1 OF S-ADDRESS '/'

ADDRESS-TYPE OF S-ADDRESS

MOVE 'GET-ADDRESS-DATA(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

* ELSE

* MOVE EMPL-ADDRESS-SBR OF S-ADDRESS

* TO EMPL-ADDRESS-SBR OF I-PERADDR

* END-IF

.

GET-EMPL-ADDRESS-EXIT.

/*****

* *

LT223-INSERT-NONEE-TRIGGER SECTION.

LT223.

*

*

INITIALIZE BIND-DATA OF I-NONEE

MOVE OPRID OF SQLRT TO OPRID OF I-NONEE

MOVE BATCH-RUN-ID OF SQLRT TO BATCH-RUN-ID OF I-NONEE

MOVE EMPLID OF W-EVENT TO EMPLID OF I-NONEE

MOVE DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

TO DEPENDENT-BENEF OF I-NONEE

MOVE JOB-EMPL-RCD OF EMPL-HR-DATA OF W-EVENT

TO JOB-EMPL-RCD OF I-NONEE

MOVE JOB-EFFDT OF EMPL-HR-DATA OF W-EVENT

TO JOB-EFFDT OF I-NONEE

MOVE JOB-EFFSEQ OF EMPL-HR-DATA OF W-EVENT

TO JOB-EFFSEQ OF I-NONEE

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

TO COBRA-EMPLID OF I-NONEE

IF CBR-COV-OVERRIDE-YES

MOVE CBR-COV-AS-OF-DT OF W-EVENT

TO COBRA-EVENT-DT OF I-NONEE

ELSE

MOVE COBRA-EVENT-DT OF W-EVENT

TO COBRA-EVENT-DT OF I-NONEE

END-IF

MOVE CBR-NONEE-CREATE OF W-EVENT

TO CBR-NONEE-CREATE OF I-NONEE

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF I-NONEE OF W-CNTL

SQL-STMT OF I-NONEE

BIND-SETUP OF I-NONEE

BIND-DATA OF I-NONEE

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-DUPL OF SQLRT

DISPLAY 'Emplid: '

EMPLID OF I-NONEE

MOVE 'INSERT-NONEE-TRIGGER' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

INSERT-NONEE-TRIGGER-EXIT.

/*****

*

*

LT230-STORE-BP-PGM-DATA SECTION.

LT230.

*

*

INITIALIZE BIND-DATA OF I-BP-PGM

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

TO EMPLID OF I-BP-PGM

MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF I-BP-PGM

IF CBR-COV-OVERRIDE-YES

MOVE CBR-COV-AS-OF-DT OF W-EVENT TO EFFDT OF I-BP-PGM

ELSE

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT OF I-BP-PGM

END-IF

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF I-BP-PGM

MOVE BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)

TO BENEFIT-PROGRAM OF I-BP-PGM

ELSE

MOVE CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-ID OF I-BP-PGM

MOVE INIT-BENEFIT-PGM OF W-EVENT(WPLAN-IDX)

TO BENEFIT-PROGRAM OF I-BP-PGM

END-IF

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF I-BP-PGM OF W-CNTL

SQL-STMT OF I-BP-PGM

BIND-SETUP OF I-BP-PGM

BIND-DATA OF I-BP-PGM

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-DUPL OF SQLRT

MOVE 'STORE-BP-PGM' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

STORE-BP-PGM-EXIT.

/*****

* *

LT240-STORE-HTH-DATA SECTION.

LT240.

* *

PERFORM LT241-INSERT-HTH-ELECT

PERFORM VARYING WDPND-IDX FROM 1 BY 1

UNTIL WDPND-IDX > WDPND-COUNT OF W-EVENT

IF PLAN-TYPE OF WDPND-DATA OF W-EVENT(WDPND-IDX)

= PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM LT243-INSERT-HEALTH-DEP

END-IF

END-PERFORM

IF COBRA-ELECT-ELECT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

AND CBR-PROCESS-OPEN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

AND CBR-ENROLL-NO OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

PERFORM LT242-INSERT-HTH-TERM

END-IF

.

STORE-HEALTH-DATA-EXIT.

/*****

* *

LT241-INSERT-HTH-ELECT SECTION.

LT241.

* *

INITIALIZE BIND-DATA OF I-HTH

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

TO EMPLID OF I-HTH

MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF I-HTH

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF I-HTH

ELSE

MOVE CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-ID OF I-HTH

END-IF

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF I-HTH

IF COBRA-REVOKE-DT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

NOT = SPACE

AND COBRA-REVOKE-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

> COVERAGE-BEGIN-DT

OF W-EVENT(WPLAN-IDX)

MOVE COBRA-REVOKE-DT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO COVERAGE-BEGIN-DT OF I-HTH

MOVE COBRA-REVOKE-DT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO EFFDT OF I-HTH

ELSE

MOVE COVERAGE-BEGIN-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO COVERAGE-BEGIN-DT OF I-HTH

MOVE COVERAGE-BEGIN-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO EFFDT OF I-HTH

END-IF

MOVE SPACE TO DEDUCTION-END-DT OF I-HTH

MOVE SPACE TO COVERAGE-END-DT OF I-HTH

MOVE COBRA-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO COVERAGE-ELECT OF I-HTH

MOVE COBRA-ELECT-DT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO COVERAGE-ELECT-DT OF I-HTH

MOVE BENEFIT-PLAN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO BENEFIT-PLAN OF I-HTH
MOVE COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO COVRG-CD OF I-HTH
SET PRETAX-CD-YES OF I-HTH TO TRUE
MOVE HLTH-PROVIDER-ID OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO HLTH-PROVIDER-ID OF I-HTH
MOVE PREVIOUSLY-SEEN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO PREVIOUSLY-SEEN OF I-HTH
MOVE OTH-INSURANCE-IND OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO OTH-INSURANCE-IND OF I-HTH
MOVE OTH-INSURANCE-NAME OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO OTH-INSURANCE-NAME OF I-HTH

PERFORM XX100-INSERT-BP-PLN
PERFORM XX200-INSERT-HEALTH-BENEFIT

IF BILLING-YES OF W-CNTL
AND COBRA-ELECT-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

INITIALIZE BIINT

SET BILL-FUNCTION-ADD OF BIINT TO TRUE
MOVE EMPLID OF I-HTH TO EMPLID OF BIINT
MOVE EMPL-RCD-NO OF I-HTH TO EMPL-RCD-NO OF BIINT
MOVE COBRA-EVENT-ID OF I-HTH TO COBRA-EVENT-ID OF BIINT
MOVE PLAN-TYPE OF I-HTH TO PLAN-TYPE OF BIINT

SET BILLING-STATUS-ACTIVE OF BIINT TO TRUE

SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE

IF WAIVE-COBRA-SURCHG-NO OF W-CBR-EVT(WCBR-EVT-IDX)

 MOVE COBRA-SURCHARGE OF W-CBRDEFN

 TO RATE-PERCENT OF BIINT

ELSE

 MOVE ZERO TO RATE-PERCENT OF BIINT

END-IF

MOVE ZERO TO RATE-AMOUNT OF BIINT

MOVE ZERO TO EVENT-ID OF BIINT

MOVE COVERAGE-BEGIN-DT OF I-HTH TO BILL-EFFDT OF BIINT

PERFORM XX500-NOTIFY-BILLING

IF DISABL-CVG-BEG-DT OF W-EVENT(WPLAN-IDX) NOT= SPACE

 AND WAIVE-COBRA-SURCHG-NO

 OF W-CBR-EVT(WCBR-EVT-IDX)

 MOVE DISABL-CVG-BEG-DT OF W-EVENT(WPLAN-IDX)

 TO BILL-EFFDT OF BIINT

 MOVE COBRA-DISABL-SURCG OF W-CBRDEFN

 TO RATE-PERCENT OF BIINT

 PERFORM XX500-NOTIFY-BILLING

END-IF

END-IF

.

INSERT-HTH-ELECT-EXIT.

/*****

* *

LT242-INSERT-HTH-TERM SECTION.

LT242.

* *

MOVE COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

TO BEGIN-DT OF DTWRK

MOVE 1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK TO EFFDT OF I-HTH

MOVE END-DT OF DTWRK TO COVERAGE-BEGIN-DT OF I-HTH

SET COVERAGE-ELECT-TERM OF I-HTH TO TRUE

MOVE SPACE TO BENEFIT-PLAN OF I-HTH

MOVE SPACE TO COVRG-CD OF I-HTH

SET PRETAX-CD-YES OF I-HTH TO TRUE

MOVE SPACE TO HLTH-PROVIDER-ID OF I-HTH

SET PREVIOUSLY-SEEN-NO OF I-HTH TO TRUE

SET OTH-INSURANCE-NO OF I-HTH TO TRUE

MOVE SPACE TO OTH-INSURANCE-NAME OF I-HTH

PERFORM XX200-INSERT-HEALTH-BENEFIT

IF BILLING-YES OF W-CNTL

AND COBRA-ELECT-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

INITIALIZE BIINT

SET BILL-FUNCTION-ADD OF BIINT TO TRUE

MOVE EMPLID OF I-HTH TO EMPLID OF BIINT

MOVE EMPL-RCD-NO OF I-HTH TO EMPL-RCD-NO OF BIINT

MOVE COBRA-EVENT-ID OF I-HTH TO COBRA-EVENT-ID OF BIINT

MOVE PLAN-TYPE OF I-HTH TO PLAN-TYPE OF BIINT

SET BILLING-STATUS-INACTIVE OF BIINT TO TRUE

SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE

MOVE ZERO TO RATE-PERCENT OF BIINT

MOVE ZERO TO RATE-AMOUNT OF BIINT

MOVE ZERO TO EVENT-ID OF BIINT

MOVE COVERAGE-BEGIN-DT OF I-HTH TO BILL-EFFDT OF BIINT

PERFORM XX500-NOTIFY-BILLING

END-IF

INSERT-HTH-TERM-EXIT.

/*****

*

*

LT243-INSERT-HEALTH-DEP SECTION.

LT243.

*

*

INITIALIZE BIND-DATA OF I-HTH-DEP

MOVE EMPLID OF I-HTH TO EMPLID OF I-HTH-DEP

MOVE EMPL-RCD-NO OF I-HTH TO EMPL-RCD-NO OF I-HTH-DEP

MOVE COBRA-EVENT-ID OF I-HTH TO COBRA-EVENT-ID OF I-HTH-DEP

MOVE PLAN-TYPE OF I-HTH TO PLAN-TYPE OF I-HTH-DEP

MOVE EFFDT OF I-HTH TO EFFDT OF I-HTH-DEP

MOVE COBRA-DEP-BENEF OF W-EVENT(WDPND-IDX)

TO DEPENDENT-BENEF OF I-HTH-DEP

MOVE HLTH-PROVIDER-ID OF WDPND-DATA OF W-EVENT(WDPND-IDX)

TO HLTH-PROVIDER-ID OF I-HTH-DEP

MOVE PREVIOUSLY-SEEN OF WDPND-DATA OF W-EVENT(WDPND-IDX)

TO PREVIOUSLY-SEEN OF I-HTH-DEP

MOVE OTH-INSURANCE-IND OF WDPND-DATA OF W-EVENT(WDPND-IDX)

TO OTH-INSURANCE-IND OF I-HTH-DEP

MOVE OTH-INSURANCE-NAME OF WDPND-DATA OF W-EVENT(WDPND-IDX)
TO OTH-INSURANCE-NAME OF I-HTH-DEP

PERFORM XX300-INSERT-HEALTH-DEPENDENT

.

INSERT-HEALTH-DEP-EXIT.

/*****

* *

LT250-STORE-FSA-DATA SECTION.

LT250.

* *

PERFORM LT251-INSERT-FSA-ELECT

PERFORM LT252-INSERT-FSA-TERM

.

STORE-FSA-DATA-EXIT.

/*****

* *

LT251-INSERT-FSA-ELECT SECTION.

LT251.

*

*

INITIALIZE BIND-DATA OF I-FSA

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

TO EMPLID OF I-FSA

MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF I-FSA

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF I-FSA

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF I-FSA

IF COBRA-REVOKE-DT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

NOT = SPACE

AND COBRA-REVOKE-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

> COVERAGE-BEGIN-DT

OF W-EVENT(WPLAN-IDX)

MOVE COBRA-REVOKE-DT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO COVERAGE-BEGIN-DT OF I-FSA

MOVE COBRA-REVOKE-DT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO EFFDT OF I-FSA

ELSE

MOVE COVERAGE-BEGIN-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)
TO COVERAGE-BEGIN-DT OF I-FSA
MOVE COVERAGE-BEGIN-DT OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
TO EFFDT OF I-FSA
END-IF

MOVE SPACE TO DEDUCTION-END-DT OF I-FSA
MOVE SPACE TO COVERAGE-END-DT OF I-FSA
MOVE COBRA-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO COVERAGE-ELECT OF I-FSA
MOVE COBRA-ELECT-DT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO COVERAGE-ELECT-DT OF I-FSA
MOVE BENEFIT-PLAN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO BENEFIT-PLAN OF I-FSA
MOVE EMPL-CONTRIBUTN-AMT OF W-EVENT(WPLAN-IDX)
TO EMPL-CONTRIBUTN-AMT OF I-FSA
MOVE ANN-EX-CREDIT-FSA OF W-EVENT(WPLAN-IDX)
TO ANN-EX-CREDIT-FSA OF I-FSA
MOVE ANNUAL-PLEDGE OF W-EVENT(WPLAN-IDX)
TO ANNUAL-PLEDGE OF I-FSA
MOVE FSA-SUB-AMT-YTD OF W-EVENT(WPLAN-IDX)
TO FSA-SUB-AMT-YTD OF I-FSA
MOVE FSA-APR-AMT-YTD OF W-EVENT(WPLAN-IDX)
TO FSA-APR-AMT-YTD OF I-FSA
MOVE FSA-PD-AMT-YTD OF W-EVENT(WPLAN-IDX)

TO FSA-PD-AMT-YTD OF I-FSA

SET FSA-ACCT-STATUS-ACTIVE OF I-FSA TO TRUE

SET CARRYFORWARD-CLAIMS OF I-FSA TO TRUE

PERFORM XX100-INSERT-BP-PLN

PERFORM XX400-INSERT-FSA-BENEFIT

IF BILLING-YES OF W-CNTL

 AND COBRA-ELECT-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

 INITIALIZE BIINT

 SET BILL-FUNCTION-ADD OF BIINT TO TRUE

 MOVE EMPLID OF I-FSA TO EMPLID OF BIINT

 MOVE EMPL-RCD-NO OF I-FSA TO EMPL-RCD-NO OF BIINT

 MOVE COBRA-EVENT-ID OF I-FSA TO COBRA-EVENT-ID OF BIINT

 MOVE PLAN-TYPE OF I-FSA TO PLAN-TYPE OF BIINT

 SET BILLING-STATUS-ACTIVE OF BIINT TO TRUE

 SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE

IF WAIVE-COBRA-SURCHG-NO OF W-CBR-EVT(WCBR-EVT-IDX)

 MOVE COBRA-SURCHARGE OF W-CBRDEFN

 TO RATE-PERCENT OF BIINT

ELSE

 MOVE ZERO TO RATE-PERCENT OF BIINT

END-IF

MOVE ZERO TO EVENT-ID OF BIINT

MOVE ZERO TO RATE-AMOUNT OF BIINT

MOVE COVERAGE-BEGIN-DT OF I-FSA TO BILL-EFFDT OF BIINT

PERFORM XX500-NOTIFY-BILLING

IF DISABL-CVG-BEG-DT OF W-EVENT(WPLAN-IDX) NOT= SPACE

AND WAIVE-COBRA-SURCHG-NO

OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE DISABL-CVG-BEG-DT OF W-EVENT(WPLAN-IDX)

TO BILL-EFFDT OF BIINT

MOVE COBRA-DISABL-SURCG OF W-CBRDEFN

TO RATE-PERCENT OF BIINT

PERFORM XX500-NOTIFY-BILLING

END-IF

END-IF

.

INSERT-FSA-ELECT-EXIT.

/*****

*

*

LT252-INSERT-FSA-TERM SECTION.

LT252.

*

*

MOVE COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

TO BEGIN-DT OF DTWRK

MOVE 1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK TO EFFDT OF I-FSA

MOVE END-DT OF DTWRK TO COVERAGE-BEGIN-DT OF I-FSA

SET COVERAGE-ELECT-TERM OF I-FSA TO TRUE

MOVE SPACE TO BENEFIT-PLAN OF I-FSA

MOVE ZERO TO EMPL-CONTRIBUTN-AMT OF I-FSA

MOVE ZERO TO ANN-EX-CREDIT-FSA OF I-FSA

MOVE ZERO TO ANNUAL-PLEDGE OF I-FSA

MOVE ZERO TO FSA-SUB-AMT-YTD OF I-FSA

MOVE ZERO TO FSA-APR-AMT-YTD OF I-FSA

MOVE ZERO TO FSA-PD-AMT-YTD OF I-FSA

SET FSA-ACCT-STATUS-TERMINATED OF I-FSA TO TRUE

PERFORM XX400-INSERT-FSA-BENEFIT

IF BILLING-YES OF W-CNTL

AND COBRA-ELECT-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

INITIALIZE BIINT
SET BILL-FUNCTION-ADD OF BIINT TO TRUE
MOVE EMPLID OF I-FSA TO EMPLID OF BIINT
MOVE EMPL-RCD-NO OF I-FSA TO EMPL-RCD-NO OF BIINT
MOVE COBRA-EVENT-ID OF I-FSA
TO COBRA-EVENT-ID OF BIINT
MOVE PLAN-TYPE OF I-FSA TO PLAN-TYPE OF BIINT
SET BILLING-STATUS-INACTIVE OF BIINT TO TRUE
SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE
MOVE ZERO TO RATE-PERCENT OF BIINT
MOVE ZERO TO RATE-AMOUNT OF BIINT
MOVE ZERO TO EVENT-ID OF BIINT
MOVE COVERAGE-BEGIN-DT OF I-FSA
TO BILL-EFFDT OF BIINT
PERFORM XX500-NOTIFY-BILLING
END-IF

.

INSERT-FSA-TERM-EXIT.

/*****

* *

LT300-UPDATE-PARTIC-STATUS SECTION.

LT300.

* *

```
SET CBR-PROCESS-CLOSED OF WPARTIC-DATA
  OF W-EVENT(WPARTIC-IDX) TO TRUE
MOVE ZERO TO INIT-PLAN-COUNT OF W-COUNT
MOVE ZERO TO INIT-ERROR-COUNT OF W-COUNT
MOVE ZERO TO INIT-ENROLL-COUNT OF W-COUNT

PERFORM VARYING WPLAN-IDX FROM 1 BY 1
  UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

IF CBR-PROCESS-OPEN OF WPLAN-DATA
  OF W-EVENT(WPLAN-IDX)

  SET CBR-PROCESS-OPEN OF WPARTIC-DATA
    OF W-EVENT(WPARTIC-IDX) TO TRUE
END-IF

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

  ADD 1 TO INIT-PLAN-COUNT OF W-COUNT

  IF COBRA-ERROR-YES OF WPLAN-DATA
    OF W-EVENT(WPLAN-IDX)

    ADD 1 TO INIT-ERROR-COUNT OF W-COUNT
```

END-IF

IF CBR-ENROLLED OF W-EVENT(WPLAN-IDX)
OR COBRA-ELECT-WAIVE OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

ADD 1 TO INIT-ENROLL-COUNT OF W-COUNT

END-IF

END-IF

END-PERFORM

IF INIT-ERROR-COUNT OF W-COUNT > ZERO

SET CBR-INIT-ELECT-ERROR OF W-EVENT(WPARTIC-IDX)
TO TRUE

ELSE

IF INIT-ENROLL-COUNT OF W-COUNT
= INIT-PLAN-COUNT OF W-COUNT

SET CBR-INIT-ENROLL-COMPLETE OF W-EVENT(WPARTIC-IDX)
TO TRUE

ELSE

SET CBR-INIT-ELECT-ENROLLED OF W-EVENT(WPARTIC-IDX)
TO TRUE

END-IF

END-IF

SET UPDATE-YES OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO TRUE

.

UPDATE-PARTIC-STATUS-EXIT.

/*****

* *

LU000-EXTEND-COBRA-COVERAGE SECTION.

LU000.

* *

SET WPLAN-IDX TO 1

SEARCH WPLAN-DATA

WHEN DEPENDENT-BENEF OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

= DEPENDENT-BENEF OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

CONTINUE

END-SEARCH

PERFORM VARYING WPLAN-IDX FROM WPLAN-IDX BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

IF COBRA-EVENT-SECONDARY OF W-EVENT(WPLAN-IDX)

PERFORM LU100-GET-INITIAL-PLAN

IF INIT-PLAN-ENROLLED OF W-EVENT(WPLAN-IDX)

PERFORM LU200-EXTEND-INDIVIDUAL-COVRG

IF BILLING-YES OF W-CNTL

AND COBRA-ELECT-ELECT

OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM LU300-ADJUST-BILLING

END-IF

ELSE

SET COBRA-ELECT-ELECT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO TRUE

MOVE PROCESS-DT OF W-CNTL

TO COBRA-ELECT-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

PERFORM LU400-INSERT-INDIVIDUAL-COVRG

END-IF

SET UPDATE-YES OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO TRUE

SET SET-SCND-EVT-YES OF W-EVENT(WPLAN-IDX) TO TRUE

END-IF

END-PERFORM

SET CBR-SCND-COVRG-EXTENDED OF W-EVENT(WPARTIC-IDX) TO TRUE

SET UPDATE-YES OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

TO TRUE

.

EXTEND-COBRA-COVERAGE-EXIT.

/*****

*

*

LU100-GET-INITIAL-PLAN SECTION.

LU100.

*

*

MOVE EMPLID OF W-EVENT TO EMPLID OF S-INITPLN

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF S-INITPLN

MOVE CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-ID OF S-INITPLN

MOVE DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO DEPENDENT-BENEF OF S-INITPLN

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF S-INITPLN

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-INITPLN

BIND-SETUP OF S-INITPLN

BIND-DATA OF S-INITPLN

SELECT-SETUP OF S-INITPLN

SELECT-DATA OF S-INITPLN

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid/BenRcd#/CBREvtID/DepBenef/PlnType: '

EMPLID OF S-INITPLN '/'

BENEFIT-RCD-NO OF S-INITPLN '/'

COBRA-EVENT-ID OF S-INITPLN '/'

DEPENDENT-BENEF OF S-INITPLN '/'

PLAN-TYPE OF S-INITPLN '/'

MOVE 'GET-INITIAL-PLAN(SELECT)'

TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

INITIALIZE SELECT-DATA OF S-INITPLN

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT
SQLRT
SQL-CURSOR-COMMON OF SQLRT
IF RTNCD-ERROR OF SQLRT

MOVE 'GET-INITIAL-PLAN(FETCH)'
TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

MOVE CBR-ENROLL-STATUS OF S-INITPLN
TO INIT-PLAN-ENROLL-SW OF W-EVENT(WPLAN-IDX)

IF COBRA-TERM-DT OF S-INITPLN = SPACE

MOVE COVERAGE-END-DT OF S-INITPLN
TO INIT-COVRG-END-DT OF W-EVENT(WPLAN-IDX)

ELSE

MOVE COBRA-TERM-DT OF S-INITPLN TO BEGIN-DT OF DTWRK
MOVE -1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO INIT-COVRG-END-DT OF W-EVENT(WPLAN-IDX)

END-IF

MOVE DISABL-CVG-BEG-DT OF S-INITPLN

TO INIT-DISABL-CVG-DT OF W-EVENT(WPLAN-IDX)

MOVE COVERAGE-BEGIN-DT OF S-INITPLN

TO INIT-COVRG-BEGIN-DT OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-ID OF S-INITPLN

TO INIT-COBRA-EVENT-ID OF W-EVENT(WPLAN-IDX)

.

GET-INITIAL-PLAN-EXIT.

/*****

*

*

LU200-EXTEND-INDIVIDUAL-COVRG SECTION.

LU200.

*

*

MOVE COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

TO BEGIN-DT OF DTWRK

MOVE 1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK TO EFFDT OF U-HTHTERM

MOVE END-DT OF DTWRK TO COVERAGE-BEGIN-DT OF U-HTHTERM

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

TO EMPLID OF U-HTHTERM

MOVE BENEFIT-RCD-NO OF W-EVENT

TO EMPL-RCD-NO OF U-HTHTERM

MOVE CBR-INIT-EVENT-ID OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-ID OF U-HTHTERM

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF U-HTHTERM

MOVE INIT-COVRG-END-DT OF W-EVENT(WPLAN-IDX)

TO BEGIN-DT OF DTWRK

MOVE 1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK TO EFFDT-2 OF U-HTHTERM

IF PLAN-TYPE-HEALTH OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM XY700-UPDATE-COBRA-HTH-TERM

PERFORM LU210-GET-CUR-COBRA-HTH-COVRG

ELSE

IF PLAN-TYPE-FSA OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM XY800-UPDATE-COBRA-FSA-TERM

END-IF

END-IF

SET COBRA-COVERAGE-EXTENDED OF W-EVENT(WPLAN-IDX) TO TRUE

.

EXTEND-INDIVIDUAL-COVRG-EXIT.

/*****

* *

LU210-GET-CUR-COBRA-HTH-COVRG SECTION.

LU210.

* *

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

TO EMPLID OF S-CBRHTH
MOVE BENEFIT-RCD-NO OF W-EVENT
TO EMPL-RCD-NO OF S-CBRHTH
MOVE CBR-INIT-EVENT-ID OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
TO COBRA-EVENT-ID OF S-CBRHTH
MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO PLAN-TYPE OF S-CBRHTH
MOVE INIT-BENEFIT-PGM OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO BENEFIT-PROGRAM OF S-CBRHTH
MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT OF S-CBRHTH
MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT-2 OF S-CBRHTH

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT
SQLRT
SQL-CURSOR-COMMON OF SQLRT
SQL-STMT OF S-CBRHTH
BIND-SETUP OF S-CBRHTH
BIND-DATA OF S-CBRHTH
SELECT-SETUP OF S-CBRHTH
SELECT-DATA OF S-CBRHTH
IF RTNCD-ERROR OF SQLRT

MOVE 'GET-CUR-COBRA-HTH-COVRG(SELECT)'
TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-CBRHTH

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-CUR-COBRA-HTH-COVRG(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

ELSE

MOVE BENEFIT-PLAN OF S-CBRHTH

TO BENEFIT-PLAN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COVRG-CD OF S-CBRHTH

TO COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE OPTION-CD OF S-CBRHTH

TO OPTION-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

END-IF

.

GET-CUR-COBRA-HTH-COVRG-EXIT.

/*****

*

LU300-ADJUST-BILLING SECTION.

LU300.

*

MOVE ZERO TO RATE-PERCENT OF W-SAVE

IF WAIVE-COBRA-SURCHG-NO OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE COBRA-SURCHARGE OF W-CBRDEFN

TO RATE-PERCENT OF W-SAVE

END-IF

PERFORM LU310-GET-INIT-BILLING

IF RATE-PERCENT OF S-BI-EFDT NOT = RATE-PERCENT OF W-SAVE

PERFORM LU320-INSERT-BILLING-CHANGE

END-IF

IF INIT-DISABL-CVG-DT OF W-EVENT(WPLAN-IDX) > SPACE

AND INIT-DISABL-CVG-DT OF W-EVENT(WPLAN-IDX)

> COBRA-EVENT-DT OF W-EVENT

PERFORM LU330-DELETE-DISABLE-BILLING

END-IF

.

ADJUST-BILLING-EXIT.

/*****

*/

LU310-GET-INIT-BILLING SECTION.

LU310.

*/

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

TO EMPLID OF S-BI-EFDT

MOVE BENEFIT-RCD-NO OF W-EVENT

TO EMPL-RCD-NO OF S-BI-EFDT

MOVE INIT-COBRA-EVENT-ID OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-ID OF S-BI-EFDT

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF S-BI-EFDT

MOVE INIT-COVRG-BEGIN-DT OF W-EVENT(WPLAN-IDX)

TO EFFDT OF S-BI-EFDT

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-BI-EFDT

BIND-SETUP OF S-BI-EFDT

BIND-DATA OF S-BI-EFDT

SELECT-SETUP OF S-BI-EFDT

SELECT-DATA OF S-BI-EFDT

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-INIT-BILLING(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-BI-EFDT

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid/EmplRcd#/CBREvtID/PInTyp/Effdt: '

EMPLID OF S-BI-EFDT '/'

EMPL-RCD-NO OF S-BI-EFDT

COBRA-EVENT-ID OF S-BI-EFDT '/'

PLAN-TYPE OF S-BI-EFDT

EFFDT OF S-BI-EFDT

MOVE 'GET-INIT-BILLING(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

GET-INIT-BILLING-EXIT.

/*****

*/

LU320-INSERT-BILLING-CHANGE SECTION.

LU320.

*/

INITIALIZE BIINT

SET BILLING-STATUS-ACTIVE OF BIINT TO TRUE

MOVE COBRA-EVENT-DT OF W-EVENT TO BILL-EFFDT OF BIINT

SET BILL-FUNCTION-ADD OF BIINT TO TRUE

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

TO EMPLID OF BIINT

MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF BIINT

MOVE CBR-INIT-EVENT-ID OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-ID OF BIINT

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF BIINT

SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE

MOVE RATE-PERCENT OF W-SAVE TO RATE-PERCENT OF BIINT

MOVE ZERO TO RATE-AMOUNT OF BIINT

MOVE ZERO TO EVENT-ID OF BIINT

PERFORM XX500-NOTIFY-BILLING

.

INSERT-BILLING-CHANGE-EXIT.

/*****

*

LU330-DELETE-DISABLE-BILLING SECTION.

LU330.

*

INITIALIZE BIINT

SET BILL-FUNCTION-DELETE OF BIINT TO TRUE

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

TO EMPLID OF BIINT

MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF BIINT

MOVE CBR-INIT-EVENT-ID OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-ID OF BIINT

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF BIINT

SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE

SET BILLING-STATUS-ACTIVE OF BIINT TO TRUE

SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE

MOVE ZERO TO RATE-PERCENT OF BIINT

MOVE ZERO TO RATE-AMOUNT OF BIINT

MOVE ZERO TO EVENT-ID OF BIINT

MOVE DISABL-CVG-BEG-DT OF W-EVENT(WPLAN-IDX)

TO BILL-EFFDT OF BIINT

PERFORM XX500-NOTIFY-BILLING

.

DELETE-DISABLE-BILLING-EXIT.

/*****

*

*

LU400-INSERT-INDIVIDUAL-COVRG SECTION.

LU400.

*

*

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT OF BIND-DATA OF S-CVGCD

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT-2 OF BIND-DATA OF S-CVGCD

PERFORM XJ000-LOAD-COVRG-CD-TBL

SET OPTN-FOUND-NO OF W-SW TO TRUE

PERFORM VARYING WCVGCD-IDX FROM 1 BY 1

UNTIL WCVGCD-IDX > WCVGCD-COUNT OF W-CVGCD

OR OPTN-FOUND-YES OF W-SW

IF MAX-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

= ZERO

AND CBR-COVRG-SET OF W-CVGCD(WCVGCD-IDX)

= CBR-COVRG-SET OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

PERFORM LU410-CHECK-BENADMIN-OPTN

END-IF

END-PERFORM

IF OPTN-FOUND-NO OF W-SW

DISPLAY 'Employee Only Coverage Not Found For '

'Emplid/Pgm/PlanType/BenefitPlan: '

DISPLAY COBRA-EMPLID OF W-EVENT(WPARTIC-IDX) ' / '

BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX) ' / '

PLAN-TYPE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX) ' / '

BENEFIT-PLAN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX) ' . '

MOVE 'INSERT-INDIVIDUAL-COVERAGE'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

ELSE

MOVE BENEFIT-PLAN OF S-BENOPTN

TO BENEFIT-PLAN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COVRG-CD OF S-BENOPTN

TO COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE OPTION-CD OF S-BENOPTN

TO OPTION-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM LT210-STORE-HR-DATA

PERFORM LT230-STORE-BP-PGM-DATA

PERFORM LT240-STORE-HTH-DATA

SET COBRA-COVERAGE-INSERTED OF W-EVENT(WPLAN-IDX)

TO TRUE

END-IF

.

INSERT-INDIVIDUAL-COVRG-EXIT.

/*****

* *

LU410-CHECK-BENADMIN-OPTN SECTION.

LU410.

* *

MOVE INIT-BENEFIT-PGM OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO BENEFIT-PROGRAM OF S-BENOPTN

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF S-BENOPTN

MOVE BENEFIT-PLAN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO BENEFIT-PLAN OF S-BENOPTN

MOVE COVRG-CD OF W-CVGCD(WCVGCD-IDX)

TO COVRG-CD OF S-BENOPTN

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT OF S-BENOPTN

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-BENOPTN

BIND-SETUP OF S-BENOPTN

BIND-DATA OF S-BENOPTN

SELECT-SETUP OF S-BENOPTN

SELECT-DATA OF S-BENOPTN

IF RTNCD-ERROR OF SQLRT

MOVE 'CHECK-BENADMIN-OPTN(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-BENOPTN

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

IF NOT RTNCD-END OF SQLRT

DISPLAY 'Emplid/BenPgm/PlnTyp/BenPln/CvgCd/EvtDt: '

DISPLAY COBRA-EMPLID OF W-EVENT(WPARTIC-IDX) '/'

BENEFIT-PROGRAM OF S-BENOPTN '/'

PLAN-TYPE OF S-BENOPTN '/'

BENEFIT-PLAN OF S-BENOPTN '/'

COVRG-CD OF S-BENOPTN '/'

EFFDT OF S-BENOPTN

MOVE 'CHECK-BENADMIN-OPTN(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

SET OPTN-FOUND-YES OF W-SW TO TRUE

END-IF

.

CHECK-BENADMIN-OPTN-EXIT.

/*****

*

*

LV000-DISC-PROCESS-PARTIC SECTION.

LV000.

*

*

PERFORM NA000-DISC-DEDCALC-CURSORS

PERFORM QA000-DISC-CURSORS

IF SQL-CURSOR OF S-PRCSPAR OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PRCSPAR OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-PROCESS-PARTIC(S-PRCSPAR)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-PAROPTN OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PAROPTN OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-PROCESS-PARTIC(S-PAROPTN)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-PARDPND OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PARDPND OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-PROCESS-PARTIC(S-PARDPND)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF U-PARDPND OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF U-PARDPND OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-PROCESS-PARTIC(U-PARDPND)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF I-PAROPTN OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF I-PAROPTN OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-PROCESS-PARTIC(I-PAROPTN)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

DISC-PROCESS-PARTIC-EXIT.

/*****

*

*

LY000-TERM-PROCESS-PARTIC SECTION.

LY000.

*

*

SET COBRA-PHASE-COMPLETE OF W-CNTL TO TRUE

PERFORM TG000-FINISH-RUN-UNIT

PERFORM ZZ999-GET-DISPLAY-TIME

DISPLAY 'COBRA Participant Processing Ended at '
TIME-OUT OF W-WK '.'

.

TERM-PROCESS-PARTIC-EXIT.

/*****

*

*

NA000-DISC-DEDCALC-CURSORS SECTION.

NA000.

*

*

IF SQL-CURSOR OF S-DAYTD OF ADMIN NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
SQLRT

SQL-CURSOR OF S-DAYTD OF ADMIN

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-DEDCALC-CURSORS(S-DAYTD)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-DEDTX OF ADMIN NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-DEDTX OF ADMIN

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-DEDCALC-CURSORS(S-DEDTX)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-DPLIF OF ADMIN NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-DPLIF OF ADMIN

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-DEDCALC-CURSORS(S-DPLIF)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-EBRET OF ADMIN NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-EBRET OF ADMIN

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-DEDCALC-CURSORS(S-EBRET)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-JOBS OF ADMIN NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-JOBS OF ADMIN

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-DEDCALC-CURSORS(S-JOBS)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-PENRT OF ADMIN NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PENRT OF ADMIN

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-DEDCALC-CURSORS(S-PENRT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-SVCPCT OF ADMIN NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-SVCPCT OF ADMIN

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-DEDCALC-CURSORS(S-SVCPCT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-SVCST OF ADMIN NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-SVCST OF ADMIN

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-DEDCALC-CURSORS(S-SVCST)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

DISC-DEDCALC-CURSORS-EXIT.

/*****

*

*

QA000-DISC-CURSORS SECTION.

QA000.

*

*

IF SQL-CURSOR OF S-CBR-EVT OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-CBR-EVT OF W-CNTL

IF RTNCD-ERROR OF SQLRT

MOVE 'DISC-CURSORS(S-CBR-EVT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

IF SQL-CURSOR OF S-STDNT OF W-CNTL NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR OF S-STDNT OF W-CNTL

IF RTNCD-ERROR OF SQLRT

```
        MOVE 'DISC-CURSORS(S-STDNT)'
          TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
      END-IF
    END-IF

    IF SQL-CURSOR OF S-PERDATA OF W-CNTL NOT = ZERO

      CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF S-PERDATA OF W-CNTL
      IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(S-PERDATA)'
          TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
      END-IF
    END-IF

    IF SQL-CURSOR OF S-SMOKER OF W-CNTL NOT = ZERO

      CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF S-SMOKER OF W-CNTL
      IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(S-SMOKER)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-ADDRESS OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
                        SQLRT
                        SQL-CURSOR OF S-ADDRESS OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(S-ADDRESS)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-EMPL-HR OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
                        SQLRT
                        SQL-CURSOR OF S-EMPL-HR OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(S-EMPL-HR)'  
            TO ERR-SECTION OF SQLRT  
        PERFORM ZZ000-SQL-ERROR  
    END-IF  
END-IF  
  
IF SQL-CURSOR OF S-ELGDATA OF W-CNTL NOT = ZERO  
  
    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT  
        SQLRT  
        SQL-CURSOR OF S-ELGDATA OF W-CNTL  
    IF RTNCD-ERROR OF SQLRT  
  
        MOVE 'DISC-CURSORS(S-ELGDATA)'  
            TO ERR-SECTION OF SQLRT  
        PERFORM ZZ000-SQL-ERROR  
    END-IF  
END-IF  
  
IF SQL-CURSOR OF S-DEPEND OF W-CNTL NOT = ZERO  
  
    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT  
        SQLRT  
        SQL-CURSOR OF S-DEPEND OF W-CNTL  
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(S-DEPEND)'  
            TO ERR-SECTION OF SQLRT  
        PERFORM ZZ000-SQL-ERROR  
    END-IF  
END-IF  
  
IF SQL-CURSOR OF S-DEP-NID OF W-CNTL NOT = ZERO  
  
    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT  
        SQLRT  
        SQL-CURSOR OF S-DEP-NID OF W-CNTL  
    IF RTNCD-ERROR OF SQLRT  
  
        MOVE 'DISC-CURSORS(S-DEP-NID)'  
            TO ERR-SECTION OF SQLRT  
        PERFORM ZZ000-SQL-ERROR  
    END-IF  
END-IF  
  
IF SQL-CURSOR OF S-BP-PGM OF W-CNTL NOT = ZERO  
  
    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT  
        SQLRT  
        SQL-CURSOR OF S-BP-PGM OF W-CNTL  
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(S-BP-PGM)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-HTH OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF S-HTH OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(S-HTH)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-HTH-DEP OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF S-HTH-DEP OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(S-HTH-DEP)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-FSA OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF S-FSA OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(S-FSA)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-HTHTERM OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF S-HTHTERM OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(S-HTHTERM)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-FSATERM OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF S-FSATERM OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(S-FSATERM)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-DEPTERM OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF S-DEPTERM OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```



```
        MOVE 'DISC-CURSORS(S-DEPTERM)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-HTHWAIV OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF S-HTHWAIV OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(S-HTHWAIV)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-FSAWAIV OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF S-FSAWAIV OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(S-FSAWAIV)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF U-DEPEND OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF U-DEPEND OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(U-DEPEND)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF U-HTHTERM OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF U-HTHTERM OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(U-HTHTERM)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF U-FSATERM OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF U-FSATERM OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(U-FSATERM)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF U-VDPBEN OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF U-VDPBEN OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(U-VDPBEN)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF I-BP-PGM OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF I-BP-PGM OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(I-BP-PGM)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF I-BP-PLN OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF I-BP-PLN OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(I-BP-PLN)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF I-HTH OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING  ACTION-DISCONNECT OF SQLRT
                        SQLRT
                        SQL-CURSOR OF I-HTH OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(I-HTH)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF I-HTH-DEP OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING  ACTION-DISCONNECT OF SQLRT
                        SQLRT
                        SQL-CURSOR OF I-HTH-DEP OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(I-HTH-DEP)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF I-FSA OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF I-FSA OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(I-FSA)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-PAR1 OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF S-PAR1 OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(S-PAR1)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-PAR2 OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
                        SQLRT
                        SQL-CURSOR OF S-PAR2 OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(S-PAR2)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-PARPLN1 OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
                        SQLRT
                        SQL-CURSOR OF S-PARPLN1 OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(S-PARPLN1)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-PARPLN2 OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
                        SQLRT
                        SQL-CURSOR OF S-PARPLN2 OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(S-PARPLN2)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF S-PARPLN3 OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
                        SQLRT
                        SQL-CURSOR OF S-PARPLN3 OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```



```
        MOVE 'DISC-CURSORS(S-PARPLN3)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF U-EVT OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF U-EVT OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(U-EVT)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF U-RESEVT OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF U-RESEVT OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(U-RESEVT)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF U-PAR OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF U-PAR OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(U-PAR)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF U-PARPLN OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF U-PARPLN OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

```
        MOVE 'DISC-CURSORS(U-PARPLN)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF U-PARDPND OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF U-PARDPND OF W-CNTL
    IF RTNCD-ERROR OF SQLRT

        MOVE 'DISC-CURSORS(U-PARDPND)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
END-IF

IF SQL-CURSOR OF I-NONEE OF W-CNTL NOT = ZERO

    CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT
        SQLRT
        SQL-CURSOR OF I-NONEE OF W-CNTL
    IF RTNCD-ERROR OF SQLRT
```

MOVE 'DISC-CURSORS(I-NONEE)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

DISC-CURSORS-EXIT.

/*****

* *

RA000-TABLE-ACCESS SECTION.

RA000.

* *

CALL 'PSPBATBL' USING SQLRT

ADMIN

BATBL

PDEFN

ODEFN

GEOTB

IF RTNCD-ERROR OF SQLRT

MOVE 'TABLE-ACCESS(PSPBATBL)' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

TABLE-ACCESS-EXIT.

/*****

* *

TA000-START-RUN-UNIT SECTION.

TA000.

* *

PERFORM XA000-GET-START-TIME

MOVE ZERO TO PROGRESS-COUNT OF W-WK

MOVE ZERO TO PROGRESS-TOTAL OF W-WK

.

START-RUN-UNIT-EXIT.

/*****

* *

TD000-CHECKPOINT-RUN-UNIT SECTION.

TD000.

*

*

ADD 1 TO PROGRESS-COUNT OF W-WK

ADD 1 TO PROGRESS-TOTAL OF W-WK

IF PROGRESS-COUNT OF W-WK = PROGRESS-REPORT OF W-WK

PERFORM UA000-REPORT-PROGRESS

MOVE ZERO TO PROGRESS-COUNT OF W-WK

END-IF

SET RESELECT-NO OF W-SW TO TRUE

IF CHKPT-INTERVAL OF W-CNTL NOT = ZERO

PERFORM XD000-GET-ELAPSED-TIME

IF TIME-ELAPSED OF W-WK

> (CHKPT-INTERVAL OF W-CNTL * 60)

PERFORM UD000-TAKE-CHECKPOINT

PERFORM XA000-GET-START-TIME

IF CURSOR-NORMAL OF SQLRT

SET RESELECT-YES OF W-SW TO TRUE

END-IF

END-IF

END-IF

.

CHECKPOINT-RUN-UNIT-EXIT.

/*****

* *

TG000-FINISH-RUN-UNIT SECTION.

TG000.

* *

PERFORM WA000-CLEAR-RESTART-KEYS

PERFORM XD000-GET-ELAPSED-TIME

PERFORM UA000-REPORT-PROGRESS

PERFORM UD000-TAKE-CHECKPOINT

* PERFORM WX000-DISCONNECT-COMMON-CURSOR

.

FINISH-RUN-UNIT-EXIT.

/*****

* *

UA000-REPORT-PROGRESS SECTION.

UA000.

* *

MOVE PROGRESS-COUNT OF W-WK TO DISPLAY-COUNT-1 OF W-WK

MOVE PROGRESS-TOTAL OF W-WK TO DISPLAY-COUNT-2 OF W-WK

DISPLAY ' Progress Report: '

DISPLAY-COUNT-1 OF W-WK

' Records Processed for a Total of '

DISPLAY-COUNT-2 OF W-WK

.

REPORT-PROGRESS-EXIT.

/*****

* *

UD000-TAKE-CHECKPOINT SECTION.

UD000.

* *

PERFORM WD000-SAVE-RESTART

PERFORM WZ000-COMMIT

DIVIDE TIME-ELAPSED OF W-WK BY 60

GIVING DISPLAY-MM OF W-WK

REMAINDER DISPLAY-SS OF W-WK

MOVE PROGRESS-TOTAL OF W-WK TO DISPLAY-COUNT-2 OF W-WK

DISPLAY '>>> Commit/Checkpoint Taken after '

DISPLAY-TIME OF W-WK ' Minutes; '

'Rcds Processed ' DISPLAY-COUNT-2 OF W-WK

.

TAKE-CHECKPOINT-EXIT.

/*****

* *

WA000-CLEAR-RESTART-KEYS SECTION.

WA000.

* *

MOVE SPACE TO BENEFIT-PROGRAM OF W-CNTL

MOVE SPACE TO EMPLID OF W-CNTL

MOVE SPACE TO COBRA-EVENT-DT OF W-CNTL

MOVE SPACE TO DEPENDENT-BENEF OF W-CNTL

MOVE SPACE TO COBRA-ACTION OF W-CNTL

MOVE ZERO TO BENEFIT-RCD-NO OF W-CNTL

MOVE ZERO TO EMPL-RCD-NO OF W-CNTL

MOVE ZERO TO COBRA-EVENT-ID OF W-CNTL

.

CLEAR-RESTART-KEYS-EXIT.

/*****

* *

WD000-SAVE-RESTART SECTION.

WD000.

* *

MOVE COBRA-PHASE OF W-CNTL TO COBRA-PHASE OF U-RESTART

MOVE OVERAGE-PROCESS OF W-CNTL

TO OVERAGE-PROCESS OF U-RESTART

MOVE BENEFIT-PROGRAM OF W-CNTL

TO BENEFIT-PROGRAM OF U-RESTART

MOVE EMPLID OF W-CNTL TO EMPLID OF U-RESTART

MOVE EMPL-RCD-NO OF W-CNTL TO EMPL-RCD-NO OF U-RESTART

MOVE BENEFIT-RCD-NO OF W-CNTL

TO BENEFIT-RCD-NO OF U-RESTART

MOVE COBRA-EVENT-DT OF W-CNTL

TO COBRA-EVENT-DT OF U-RESTART

MOVE COBRA-EVENT-ID OF W-CNTL

TO COBRA-EVENT-ID OF U-RESTART
MOVE DEPENDENT-BENEF OF W-CNTL
TO DEPENDENT-BENEF OF U-RESTART
MOVE COBRA-ACTION OF W-CNTL
TO COBRA-ACTION OF U-RESTART
MOVE OPRID OF SQLRT TO OPRID OF U-RESTART
MOVE BATCH-RUN-ID OF SQLRT TO BATCH-RUN-ID OF U-RESTART

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT
SQLRT
SQL-CURSOR-COMMON OF SQLRT
SQL-STMT OF U-RESTART
BIND-SETUP OF U-RESTART
BIND-DATA OF U-RESTART
IF RTNCD-ERROR OF SQLRT

MOVE 'SAVE-RESTART' TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

.

SAVE-RESTART-EXIT.

/*****

*

*

WV000-CLEAR-MESSAGES SECTION.

WV000.

*

*

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF D-MSG

BIND-SETUP OF D-MSG

BIND-DATA OF D-MSG

IF RTNCD-ERROR OF SQLRT

MOVE 'CLEAR-MESSAGES' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

CLEAR-MESSAGES-EXIT.

/*****

*

*

WX000-DISCONNECT-COMMON-CURSOR SECTION.

WX000.

*

*

IF SQL-CURSOR-COMMON OF SQLRT NOT = ZERO

CALL 'PTPSQLRT' USING ACTION-DISCONNECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

MOVE 'DISCONNECT-COMMON-CURSOR'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

DISCONNECT-COMMON-CURSOR-EXIT.

/*****

* *

WZ000-COMMIT SECTION.

WZ000.

* *

CALL 'PTPSQLRT' USING ACTION-COMMIT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

MOVE 'COMMIT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

COMMIT-EXIT.

/*****

*

*

XA000-GET-START-TIME SECTION.

XA000.

*

*

PERFORM XZ000-GET-CLOCK-TIME

MOVE TIME-CLOCK OF W-WK TO TIME-START OF W-WK

.

GET-START-TIME-EXIT.

/*****

* *

XD000-GET-ELAPSED-TIME SECTION.

XD000.

* *

PERFORM XZ000-GET-CLOCK-TIME

COMPUTE TIME-ELAPSED OF W-WK

= TIME-CLOCK OF W-WK

- TIME-START OF W-WK

IF TIME-ELAPSED OF W-WK < ZERO

COMPUTE TIME-ELAPSED OF W-WK

= TIME-ELAPSED OF W-WK

+ (24

* 60

* 60)

END-IF

.

GET-ELAPSED-TIME-EXIT.

/*****

* *

XF000-LOAD-CBR-EVT-RULES SECTION.

XF000.

* *

MOVE ZERO TO WCBR-EVT-COUNT OF W-CBR-EVT

MOVE ZERO TO WCBR-BEN-COUNT OF W-CBR-EVT

PERFORM XF100-SELECT-CBR-EVT-RULES

PERFORM XF200-FETCH-CBR-EVT-RULES

PERFORM UNTIL RTNCD-END OF SQLRT

IF WCBR-EVNT-COUNT-MAX OF W-CBR-EVT

DISPLAY 'Max # of COBRA Event Rules Exceeded'

MOVE 'LOAD-CBR-EVT-RULES' TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO WCBR-EVT-COUNT OF W-CBR-EVT

PERFORM XF300-SAVE-CBR-EVT-RULES

PERFORM XF400-SELECT-CBR-EVT-BEN

PERFORM XF500-FETCH-CBR-EVT-BEN

PERFORM UNTIL RTNCD-END OF SQLRT

IF WCBR-BEN-COUNT-MAX OF W-CBR-EVT

DISPLAY 'Max # of COBRA Event Benefs Exceeded'

MOVE 'LOAD-CBR-EVT-RULES'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO WCBR-BEN-COUNT OF W-CBR-EVT

PERFORM XF600-SAVE-CBR-EVT-BEN

PERFORM XF500-FETCH-CBR-EVT-BEN

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

PERFORM XF200-FETCH-CBR-EVT-RULES

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

LOAD-CBR-EVT-RULES-EXIT.

/*****

*

*

XF100-SELECT-CBR-EVT-RULES SECTION.

XF100.

*

*

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT OF BIND-DATA OF S-CBR-EVT

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT-2 OF BIND-DATA OF S-CBR-EVT

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-CBR-EVT OF W-CNTL

SQL-STMT OF S-CBR-EVT

BIND-SETUP OF S-CBR-EVT

BIND-DATA OF S-CBR-EVT

SELECT-SETUP OF S-CBR-EVT

SELECT-DATA OF S-CBR-EVT

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-CBR-EVT-RULES' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-CBR-EVT-RULES-EXIT.

/*****

* *

XF200-FETCH-CBR-EVT-RULES SECTION.

XF200.

* *

INITIALIZE SELECT-DATA OF S-CBR-EVT

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-CBR-EVT OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

DISPLAY 'CBREvtEffDt: ' EFFDT OF BIND-DATA OF S-CBR-EVT

MOVE 'FETCH-CBR-EVT-RULES' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

FETCH-CBR-EVT-RULES-EXIT.

/*****

* *

XF300-SAVE-CBR-EVT-RULES SECTION.

XF300.

* *

SET WCBR-EVT-IDX TO WCBR-EVT-COUNT OF W-CBR-EVT

INITIALIZE WCBR-EVT-DATA OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE COBRA-EVENT-CLASS OF S-CBR-EVT

TO COBRA-EVENT-CLASS OF WCBR-EVT-DATA

OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE EFFDT OF SELECT-DATA OF S-CBR-EVT

TO EFFDT OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE MM-COVERAGE OF S-CBR-EVT

TO MM-COVERAGE OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE ADDL-MM-DISABLED OF S-CBR-EVT

TO ADDL-MM-DISABLED OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE USE-BAS-EVENT-RULE OF S-CBR-EVT

TO USE-BAS-EVENT-RULE OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE COBRA-PD-BEGINS-CD OF S-CBR-EVT

TO COBRA-PD-BEGINS-CD OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE INCL-ALT-COVRG OF S-CBR-EVT
TO INCL-ALT-COVRG OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE EVENT-NOTIFY-DAYS OF S-CBR-EVT
TO EVENT-NOTIFY-DAYS OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE RESPONSE-DAYS OF S-CBR-EVT
TO RESPONSE-DAYS OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE PAYMENT-GRACE-DAYS OF S-CBR-EVT
TO PAYMENT-GRACE-DAYS OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE WAIVE-COBRA-SURCHG OF S-CBR-EVT
TO WAIVE-COBRA-SURCHG OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE SECOND-EVENT-ROLE OF S-CBR-EVT
TO SECOND-EVENT-ROLE OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE SECOND-EVENT-MONTH OF S-CBR-EVT
TO SECOND-EVENT-MONTH OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE SECOND-EVENT-MODE OF S-CBR-EVT
TO SECOND-EVENT-MODE OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE BENEFIT-LOST-LEVEL OF S-CBR-EVT
TO BENEFIT-LOST-LEVEL OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE EFF-STATUS OF SELECT-DATA OF S-CBR-EVT
TO EFF-STATUS OF WCBR-EVT-DATA(WCBR-EVT-IDX)

IF EFFDT OF SELECT-DATA OF S-CBR-EVT
 > WCBR-EVT-MAX-EFFDT OF W-CBR-EVT
 OR WCBR-EVT-COUNT OF W-CBR-EVT = 1

MOVE EFFDT OF SELECT-DATA OF S-CBR-EVT

TO WCBR-EVT-MAX-EFFDT OF W-CBR-EVT

END-IF

.

SAVE-CBR-EVT-RULES-EXIT.

/*****

* *

XF400-SELECT-CBR-EVT-BEN SECTION.

XF400.

* *

MOVE COBRA-EVENT-CLASS OF S-CBR-EVT

TO COBRA-EVENT-CLASS OF BIND-DATA OF S-CBR-BEN

MOVE EFFDT OF SELECT-DATA OF S-CBR-EVT

TO EFFDT OF BIND-DATA OF S-CBR-BEN

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-CBR-BEN

BIND-SETUP OF S-CBR-BEN

BIND-DATA OF S-CBR-BEN

SELECT-SETUP OF S-CBR-BEN

SELECT-DATA OF S-CBR-BEN

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-CBR-EVT-BEN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

CBR-EVT-BEN-EXIT.

/*****

* *

XF500-FETCH-CBR-EVT-BEN SECTION.

XF500.

* *

INITIALIZE SELECT-DATA OF S-CBR-BEN

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-CBR-EVT-BEN' TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

.

FETCH-CBR-EVT-BEN-EXIT.

/*****

* *

XF600-SAVE-CBR-EVT-BEN SECTION.

XF600.

* *

SET WCBR-BEN-IDX TO WCBR-BEN-COUNT OF W-CBR-EVT
INITIALIZE WCBR-BEN-DATA OF W-CBR-EVT(WCBR-BEN-IDX)
MOVE COBRA-EVENT-CLASS OF SELECT-DATA OF S-CBR-BEN
TO COBRA-EVENT-CLASS OF WCBR-BEN-DATA
OF W-CBR-EVT(WCBR-BEN-IDX)
MOVE COVERED-PERSON-TYP OF S-CBR-BEN
TO COVERED-PERSON-TYP OF W-CBR-EVT(WCBR-BEN-IDX)

.

SAVE-CBR-EVT-BEN-EXIT.

/*****

* *

XH000-GET-CBR-EVT-RULE SECTION.

XH000.

* *

SET WCBR-EVT-IDX TO 1

SEARCH WCBR-EVT-DATA

WHEN WCBR-EVT-IDX > WCBR-EVT-COUNT OF W-CBR-EVT

DISPLAY 'COBRA Event Rules Not Found For: '

EMPLID OF W-EVENT '/'

COBRA-EVENT-CLASS OF W-CNTL '/'

COBRA-EVENT-DT OF W-EVENT '.'

MOVE 'GET-CBR-EVT-RULE' TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

WHEN COBRA-EVENT-CLASS OF W-CNTL

= COBRA-EVENT-CLASS OF WCBR-EVT-DATA

OF W-CBR-EVT(WCBR-EVT-IDX)

AND COBRA-EVENT-DT OF W-EVENT

>= EFFDT OF WCBR-EVT-DATA
OF W-CBR-EVT(WCBR-EVT-IDX)

CONTINUE

END-SEARCH

.

GET-CBR-EVT-RULE-EXIT.

/*****

*

*

XJ000-LOAD-COVRG-CD-TBL SECTION.

XJ000.

*

*

INITIALIZE W-CVG

MOVE ZERO TO WCVGCD-COUNT OF W-CVGCD

MOVE ZERO TO COVERED-PERS-TYP-CNT OF W-WK

PERFORM XJ100-SELECT-CVGCD

PERFORM XJ200-FETCH-CVGCD

MOVE COVRG-CD OF S-CVGCD TO COVRG-CD OF W-CVG

SET EE-FOUND-NO OF W-SW TO TRUE

SET OTHR-DEP-FOUND-NO OF W-SW TO TRUE

PERFORM UNTIL RTNCD-END OF SQLRT

IF COVRG-CD OF S-CVGCD NOT = COVRG-CD OF W-CVG

PERFORM XJ300-SAVE-CVGCD

ELSE

PERFORM XJ400-MIN-MAX-CVRD

END-IF

PERFORM XJ200-FETCH-CVGCD

END-PERFORM

IF RTNCD-END OF SQLRT

PERFORM XJ300-SAVE-CVGCD

END-IF

SET RTNCD-OK OF SQLRT TO TRUE

.

LOAD-COVRG-CD-TBL-EXIT.

/*****

* * *

XJ100-SELECT-CVGCD SECTION.

XJ100.

* * *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-CVGCD

BIND-SETUP OF S-CVGCD

BIND-DATA OF S-CVGCD

SELECT-SETUP OF S-CVGCD

SELECT-DATA OF S-CVGCD

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-CVGCD' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-CVGCD-EXIT.

/*****

* *

XJ200-FETCH-CVGCD SECTION.

XJ200.

* *

INITIALIZE SELECT-DATA OF S-CVGCD

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-CVGCD' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-CVGCD-EXIT.

/*****

* *

XJ300-SAVE-CVGCD SECTION.

XJ300.

* *

IF WCVGCD-COUNT-MAX OF W-CVGCD

```
    DISPLAY 'Max # of Coverage Code Exceeded'

    MOVE 'SAVE-CVGCD' TO ERR-SECTION OF SQLRT

    SET RTNCD-USER-ERROR OF SQLRT TO TRUE

    PERFORM ZZ000-SQL-ERROR

END-IF


ADD 1 TO WCVGCD-COUNT OF W-CVGCD


SET WCVGCD-IDX TO WCVGCD-COUNT OF W-CVGCD


INITIALIZE WCVGCD-DATA OF W-CVGCD(WCVGCD-IDX)


MOVE COVRG-CD OF W-CVG TO COVRG-CD OF W-CVGCD(WCVGCD-IDX)


MOVE CBR-COVRG-SET OF W-CVG
    TO CBR-COVRG-SET OF W-CVGCD(WCVGCD-IDX)


MOVE CC-CTL-TOTAL-FLG OF W-CVG
    TO CC-CTL-TOTAL-FLG OF W-CVGCD(WCVGCD-IDX)

MOVE MIN-NUM-TOTAL OF W-CVG
    TO MIN-NUM-TOTAL OF W-CVGCD(WCVGCD-IDX)

MOVE MAX-NUM-TOTAL OF W-CVG
    TO MAX-NUM-TOTAL OF W-CVGCD(WCVGCD-IDX)

MOVE MIN-NUM-CVRD-CH OF W-CVG
    TO MIN-NUM-CHILD OF W-CVGCD(WCVGCD-IDX)

MOVE MAX-NUM-CVRD-CH OF W-CVG
```

TO MAX-NUM-CHILD OF W-CVGCD(WCVGCD-IDX)

- * IF OTHER DEPENDENTS OTHER THAN SPOUSE, EXSPOUSE OR EMPLOYEE
 - * IS FOUND OR EMPLOYEE ONLY COVERAGE THEN SET 'OTHER DEP IN COVG
 - * CD' TO YES, THIS WILL BE USED TO EXCLUDE THOSE OPTIONS IN
 - * LS112-COMPARE-COVG-CD FOR CHECKING SPOUSE/EXSP COBRA COVERAGE
- IF OTHR-DEP-FOUND-YES OF W-SW
- OR (EE-FOUND-YES AND COVERED-PERS-TYP-CNT OF W-WK = 1)

SET OTHR-DEP-IN-CVGCD-YES OF W-CVGCD(WCVGCD-IDX) TO TRUE

ELSE

SET OTHR-DEP-IN-CVGCD-NO OF W-CVGCD(WCVGCD-IDX) TO TRUE

END-IF

SET OTHR-DEP-FOUND-NO OF W-SW TO TRUE

MOVE ZERO TO COVERED-PERS-TYP-CNT OF W-WK

IF CC-CTL-TOTAL-YES OF W-CVGCD(WCVGCD-IDX)

AND EE-FOUND-YES OF W-SW

SUBTRACT 1 FROM MIN-NUM-TOTAL OF W-CVGCD(WCVGCD-IDX)

IF MAX-NUM-TOTAL OF W-CVGCD(WCVGCD-IDX) < 99

SUBTRACT 1 FROM MAX-NUM-TOTAL OF W-CVGCD(WCVGCD-IDX)

END-IF

END-IF

SET EE-FOUND-NO OF W-SW TO TRUE

MOVE EFFDT OF W-CVG

TO EFFDT OF W-CVGCD(WCVGCD-IDX)

IF CC-CTL-TOTAL-YES OF W-CVGCD(WCVGCD-IDX)

MOVE MIN-NUM-TOTAL OF W-CVGCD(WCVGCD-IDX)

TO MIN-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

MOVE MAX-NUM-TOTAL OF W-CVGCD(WCVGCD-IDX)

TO MAX-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

ELSE

MOVE SUM-MIN-NUM-CVRD OF W-CVG

TO MIN-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

IF SUM-MAX-NUM-CVRD OF W-CVG > 99

MOVE '99' TO MAX-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

ELSE

MOVE SUM-MAX-NUM-CVRD OF W-CVG

TO MAX-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

END-IF

IF SUM-MAX-NUM-CVRD OF W-CVG = 2

AND SUM-MIN-NUM-CVRD OF W-CVG = 0

MOVE '1' TO MIN-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

MOVE '1' TO MAX-NUM-OF-DEPS OF W-CVGCD(WCVGCD-IDX)

END-IF

END-IF

IF COVERED-PERSON-TYP OF W-CVG = 'SP' OR

COVERED-PERSON-TYP OF W-CVG = 'SS'

IF MIN-NUM-CVRD-SP OF W-CVG = 0

AND MAX-NUM-CVRD-SP OF W-CVG = 0

MOVE 'N' TO SPOUSE-COVERAGE OF W-CVGCD(WCVGCD-IDX)

END-IF

IF MIN-NUM-CVRD-SP OF W-CVG = 1

AND MAX-NUM-CVRD-SP OF W-CVG = 1

MOVE 'R' TO SPOUSE-COVERAGE OF W-CVGCD(WCVGCD-IDX)

END-IF

IF MIN-NUM-CVRD-SP OF W-CVG = 0

AND MAX-NUM-CVRD-SP OF W-CVG = 1

MOVE 'A' TO SPOUSE-COVERAGE OF W-CVGCD(WCVGCD-IDX)

END-IF

ELSE

IF MIN-NUM-CVRD-SP OF W-CVG = 0

AND MAX-NUM-CVRD-SP OF W-CVG = 0

MOVE 'N' TO SPOUSE-COVERAGE OF W-CVGCD(WCVGCD-IDX)

END-IF

END-IF

IF (COVERED-PERSON-TYP OF W-CVG = 'SP' OR

COVERED-PERSON-TYP OF W-CVG = 'SS')

AND CC-CTL-TOTAL-YES OF W-CVGCD(WCVGCD-IDX)

MOVE 'A' TO SPOUSE-COVERAGE OF W-CVGCD(WCVGCD-IDX)

END-IF

INITIALIZE W-CVG

IF COVRG-CD OF S-CVGCD NOT = SPACES

PERFORM XJ400-MIN-MAX-CVRD

END-IF

.

SAVE-CVGCD-EXIT.

/*****

* *

XJ400-MIN-MAX-CVRD SECTION.

XJ400.

* *

MOVE COVRG-CD OF S-CVGCD TO COVRG-CD OF W-CVG

MOVE CBR-COVRG-SET OF S-CVGCD

TO CBR-COVRG-SET OF W-CVG

MOVE CC-CTL-TOTAL-FLG OF SELECT-DATA OF S-CVGCD

TO CC-CTL-TOTAL-FLG OF W-CVG

MOVE MIN-NUM-TOTAL OF SELECT-DATA OF S-CVGCD

TO MIN-NUM-TOTAL OF W-CVG
MOVE MAX-NUM-TOTAL OF SELECT-DATA OF S-CVGCD
TO MAX-NUM-TOTAL OF W-CVG

IF COVERED-PERSON-TYP OF S-CVGCD = 'SP' OR
COVERED-PERSON-TYP OF S-CVGCD = 'SS'
MOVE COVERED-PERSON-TYP OF S-CVGCD
TO COVERED-PERSON-TYP OF W-CVG
MOVE MIN-NUM-COVERED OF S-CVGCD
TO MIN-NUM-CVRD-SP OF W-CVG
MOVE MAX-NUM-COVERED OF S-CVGCD
TO MAX-NUM-CVRD-SP OF W-CVG
END-IF

IF EFFDT OF SELECT-DATA OF S-CVGCD
> EFFDT OF W-CVG
MOVE EFFDT OF SELECT-DATA OF S-CVGCD
TO EFFDT OF W-CVG
MOVE EFFDT OF SELECT-DATA OF S-CVGCD
TO WCVGCD-MAX-EFFDT OF W-CVGCD
END-IF

IF COVERED-PERSON-TYP OF S-CVGCD NOT = 'EE'
COMPUTE SUM-MIN-NUM-CVRD OF W-CVG
= SUM-MIN-NUM-CVRD OF W-CVG
+ MIN-NUM-COVERED OF S-CVGCD

COMPUTE SUM-MAX-NUM-CVRD OF W-CVG

= SUM-MAX-NUM-CVRD OF W-CVG

+ MAX-NUM-COVERED OF S-CVGCD

ELSE

SET EE-FOUND-YES OF W-SW TO TRUE

END-IF

IF COVERED-PERSON-TYP OF S-CVGCD NOT = 'SP'

AND COVERED-PERSON-TYP OF S-CVGCD NOT = 'SS'

AND COVERED-PERSON-TYP OF S-CVGCD NOT = 'XS'

AND COVERED-PERSON-TYP OF S-CVGCD NOT = 'EE'

AND COVERED-PERSON-TYP OF S-CVGCD NOT = SPACE

SET OTHR-DEP-FOUND-YES OF W-SW TO TRUE

COMPUTE MIN-NUM-CVRD-CH OF W-CVG

= MIN-NUM-CVRD-CH OF W-CVG

+ MIN-NUM-COVERED OF S-CVGCD

COMPUTE MAX-NUM-CVRD-CH OF W-CVG

= MAX-NUM-CVRD-CH OF W-CVG

+ MAX-NUM-COVERED OF S-CVGCD

END-IF

ADD 1 TO COVERED-PERS-TYP-CNT OF W-WK

.

MIN-MAX-CVRD-EXIT.

/*****

* *

XL000-GET-ELIG-PARAMS SECTION.

XL000.

* *

PERFORM XL100-GET-EMPL-HR-DATA

INITIALIZE PARTC

MOVE EMPLID OF W-EVENT TO EMPLID OF PARTC

MOVE COBRA-EVENT-DT OF W-EVENT TO EVENT-DT OF PARTC

MOVE COBRA-EVENT-ID OF W-EVENT TO COBRA-EVENT-ID OF PARTC

MOVE DEPENDENT-BENEF OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO DEPENDENT-BENEF OF PARTIC-DATA OF PARTC

MOVE BENEFIT-RCD-NO OF W-EVENT
TO BENEFIT-RCD-NO OF PARTIC-DATA OF PARTC

MOVE EMPL-RCD-NO OF W-EVENT
TO EMPL-RCD-NO OF PARTIC-DATA OF PARTC

MOVE SPACE TO BENEFIT-PROGRAM OF PARTIC-DATA OF PARTC

MOVE SERVICE-DT OF W-EVENT
TO SERVICE-DT OF PARTIC-DATA OF PARTC

MOVE TERMINATION-DT OF W-EVENT

TO TERMINATION-DT OF PARTIC-DATA OF PARTC

MOVE BIRTHDATE OF W-EVENT

TO BIRTHDATE OF HR-DATA OF PARTC

MOVE SEX OF W-EVENT

TO SEX OF HR-DATA OF PARTC

MOVE SMOKER OF W-EVENT

TO SMOKER OF HR-DATA OF PARTC

MOVE HIGHLY-COMP-EMPL-C OF W-EVENT

TO HIGHLY-COMP-EMPL-C OF HR-DATA OF PARTC

MOVE STATE OF W-DEPEND(WDEPEND-IDX)

TO STATE OF HR-DATA OF PARTC

MOVE POSTAL OF W-DEPEND(WDEPEND-IDX)

TO POSTAL OF HR-DATA OF PARTC

MOVE COUNTRY OF W-DEPEND(WDEPEND-IDX)

TO COUNTRY OF HR-DATA OF PARTC

PERFORM XL150-LOAD-ELIGIBILITY-DATA

IF WAIVE-COBRA-SURCHG-NO OF W-CBR-EVT(WCBR-EVT-IDX)

MOVE COBRA-SURCHARGE OF PDEFN TO CBR-RATE-PCT OF PARTC

MOVE COBRA-DISABL-SURCG OF PDEFN

TO CBR-DIS-RATE-PCT OF PARTC

ELSE

MOVE ZERO TO RATE-PERCENT OF BIINT

END-IF

.

GET-ELIG-PARAMS-EXIT.

/*****

* *

XL100-GET-EMPL-HR-DATA SECTION.

XL100.

* *

IF EVENT-ID OF W-EVENT NOT = ZERO

PERFORM XL110-SETUP-BENADMIN

PERFORM XL130-SELECT-EMPL-HR-DATA

INITIALIZE SELECT-DATA OF S-EMPL-HR

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-EMPL-HR OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

PERFORM XL120-SETUP-BASEBEN

PERFORM XL130-SELECT-EMPL-HR-DATA

INITIALIZE SELECT-DATA OF S-EMPL-HR

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-EMPL-HR

OF W-CNTL

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid: ' EMPLID OF W-EVENT

MOVE 'GET-EMPL-HR-DATA(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

DISPLAY 'Emplid: ' EMPLID OF W-EVENT

MOVE 'GET-EMPL-HR-DATA(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

ELSE

PERFORM XL120-SETUP-BASEBEN

PERFORM XL130-SELECT-EMPL-HR-DATA

INITIALIZE SELECT-DATA OF S-EMPL-HR

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-EMPL-HR OF W-CNTL

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid: ' EMPLID OF W-EVENT

MOVE 'GET-EMPL-HR-DATA(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

MOVE SERVICE-DT OF S-EMPL-HR TO SERVICE-DT OF W-EVENT

MOVE TERMINATION-DT OF S-EMPL-HR

TO TERMINATION-DT OF W-EVENT

MOVE UNION-CD OF S-EMPL-HR TO UNION-CD OF W-EVENT

MOVE DEPTID OF S-EMPL-HR TO DEPTID OF W-EVENT

MOVE JOBCODE OF S-EMPL-HR TO JOBCODE OF W-EVENT

MOVE EMPL-STATUS OF S-EMPL-HR TO EMPL-STATUS OF W-EVENT

MOVE ACTION OF S-EMPL-HR TO ACTION OF W-EVENT

MOVE ACTION-DT OF S-EMPL-HR TO ACTION-DT OF W-EVENT

MOVE ACTION-REASON OF S-EMPL-HR TO ACTION-REASON OF W-EVENT

MOVE SETID-LOCATION OF S-EMPL-HR

TO SETID-LOCATION OF W-EVENT

MOVE LOCATION OF S-EMPL-HR TO LOCATION OF W-EVENT

MOVE TAX-LOCATION-CD OF S-EMPL-HR

TO TAX-LOCATION-CD OF W-EVENT

MOVE REG-TEMP OF S-EMPL-HR TO REG-TEMP OF W-EVENT

MOVE FULL-PART-TIME OF S-EMPL-HR

TO FULL-PART-TIME OF W-EVENT

MOVE OFFICER-CD OF S-EMPL-HR TO OFFICER-CD OF W-EVENT

MOVE COMPANY OF S-EMPL-HR TO COMPANY OF W-EVENT

MOVE PAYGROUP OF S-EMPL-HR TO PAYGROUP OF W-EVENT

MOVE BAS-GROUP-ID OF S-EMPL-HR TO BAS-GROUP-ID OF W-EVENT

MOVE REG-REGION OF S-EMPL-HR TO REG-REGION OF W-EVENT

MOVE ELIG-CONFIG1 OF S-EMPL-HR TO ELIG-CONFIG1 OF W-EVENT

MOVE ELIG-CONFIG2 OF S-EMPL-HR TO ELIG-CONFIG2 OF W-EVENT

MOVE ELIG-CONFIG3 OF S-EMPL-HR TO ELIG-CONFIG3 OF W-EVENT

MOVE ELIG-CONFIG4 OF S-EMPL-HR TO ELIG-CONFIG4 OF W-EVENT

MOVE ELIG-CONFIG5 OF S-EMPL-HR TO ELIG-CONFIG5 OF W-EVENT

MOVE ELIG-CONFIG6 OF S-EMPL-HR TO ELIG-CONFIG6 OF W-EVENT

MOVE ELIG-CONFIG7 OF S-EMPL-HR TO ELIG-CONFIG7 OF W-EVENT

MOVE ELIG-CONFIG8 OF S-EMPL-HR TO ELIG-CONFIG8 OF W-EVENT

MOVE ELIG-CONFIG9 OF S-EMPL-HR TO ELIG-CONFIG9 OF W-EVENT

MOVE BEN-STATUS OF S-EMPL-HR TO BEN-STATUS OF W-EVENT

MOVE EMPL-TYPE OF S-EMPL-HR TO EMPL-TYPE OF W-EVENT

MOVE STD-HOURS OF S-EMPL-HR TO STD-HOURS OF W-EVENT

MOVE STD-HRS-FREQUENCY OF S-EMPL-HR
 TO STD-HRS-FREQUENCY OF W-EVENT
 MOVE STDHRS-FREQ-FACTOR OF S-EMPL-HR
 TO STDHRS-FREQ-FACTOR OF W-EVENT
 MOVE FLSA-STATUS OF S-EMPL-HR TO FLSA-STATUS OF W-EVENT
 MOVE SETID-SALARY OF S-EMPL-HR TO SETID-SALARY OF W-EVENT
 MOVE SAL-ADMIN-PLAN OF S-EMPL-HR
 TO SAL-ADMIN-PLAN OF W-EVENT
 MOVE GRADE OF S-EMPL-HR TO GRADE OF W-EVENT
 MOVE BUSINESS-UNIT OF S-EMPL-HR TO BUSINESS-UNIT OF W-EVENT
 MOVE EMPL-CLASS OF S-EMPL-HR TO EMPL-CLASS OF W-EVENT
 MOVE SETID-DEPT OF S-EMPL-HR TO SETID-DEPT OF W-EVENT
 MOVE SETID-JOBCODE OF S-EMPL-HR TO SETID-JOBCODE OF W-EVENT
 MOVE PAY-SYSTEM-FLG OF S-EMPL-HR
 TO PAY-SYSTEM-FLG OF W-EVENT
 MOVE BENEFIT-SYSTEM OF S-EMPL-HR
 TO BENEFIT-SYSTEM OF W-EVENT

IF GOVERNMENT OF ADMIN	FEDMERG
AND US-FEDERAL-GOVT OF ADMIN	FEDMERG
MOVE GVT-HIRE-DATE OF S-EMPL-HR	FED0006
TO GVT-HIRE-DATE OF W-EVENT	FED0006
* MOVE GVT-ELIG-FEHB OF S-EMPL-HR	FED0005
* TO GVT-ELIG-FEHB OF W-EVENT	FED0005

* MOVE GVT-RETIRE-PLAN OF S-EMPL-HR FED0005
* TO GVT-RETIRE-PLAN OF W-EVENT FED0005
END-IF FEDMERG

MOVE JOB-EMPL-RCD OF S-EMPL-HR
TO JOB-EMPL-RCD OF EMPL-HR-DATA OF W-EVENT
MOVE JOB-EFFDT OF S-EMPL-HR
TO JOB-EFFDT OF EMPL-HR-DATA OF W-EVENT
MOVE JOB-EFFSEQ OF S-EMPL-HR
TO JOB-EFFSEQ OF EMPL-HR-DATA OF W-EVENT

.
GET-EMPL-HR-DATA-EXIT.

/*****

* *

XL110-SETUP-BENADMIN SECTION.

XL110.

* *

SET SQL-STMT-BENADMIN OF S-EMPL-HR TO TRUE
SET EMPLID OF BIND-SETUP-BENADMIN OF S-EMPL-HR TO TRUE
SET BENEFIT-RCD-NO OF BIND-SETUP-BENADMIN OF S-EMPL-HR
TO TRUE

SET SCHED-ID OF BIND-SETUP-BENADMIN OF S-EMPL-HR
TO TRUE

SET EVENT-ID OF BIND-SETUP-BENADMIN OF S-EMPL-HR
TO TRUE

SET STDHRS-FREQ-EFFDT OF BIND-SETUP-BENADMIN OF S-EMPL-HR
TO TRUE

SET DELIMCHAR OF BIND-SETUP-BENADMIN OF S-EMPL-HR
TO TRUE

MOVE EMPLID OF W-EVENT
TO EMPLID OF BIND-DATA-BENADMIN OF S-EMPL-HR

MOVE BENEFIT-RCD-NO OF W-EVENT
TO BENEFIT-RCD-NO OF BIND-DATA-BENADMIN
OF S-EMPL-HR

MOVE SCHED-ID OF W-EVENT
TO SCHED-ID OF BIND-DATA-BENADMIN OF S-EMPL-HR

MOVE EVENT-ID OF W-EVENT
TO EVENT-ID OF BIND-DATA-BENADMIN OF S-EMPL-HR

MOVE COBRA-EVENT-DT OF W-EVENT TO STDHRS-FREQ-EFFDT
OF BIND-DATA-BENADMIN OF S-EMPL-HR

SET DELIMCHAR OF BIND-DATA-BENADMIN OF S-EMPL-HR
TO TRUE

.

SETUP-BENADMIN-EXIT.

/*****

* *

XL120-SETUP-BASEBEN SECTION.

XL120.

* *

SET SQL-STMT-BASEBEN OF S-EMPL-HR TO TRUE

SET EMPLID OF BIND-SETUP-BASEBEN OF S-EMPL-HR TO TRUE

SET EMPL-RCD-NO OF BIND-SETUP-BASEBEN OF S-EMPL-HR
TO TRUE

SET EFFDT OF BIND-SETUP-BASEBEN OF S-EMPL-HR
TO TRUE

SET STDHRS-FREQ-EFFDT OF BIND-SETUP-BASEBEN OF S-EMPL-HR
TO TRUE

SET DELIMCHAR OF BIND-SETUP-BASEBEN OF S-EMPL-HR
TO TRUE

MOVE EMPLID OF W-EVENT
TO EMPLID OF BIND-DATA-BASEBEN OF S-EMPL-HR

MOVE EMPL-RCD-NO OF W-EVENT
TO EMPL-RCD-NO OF BIND-DATA-BASEBEN OF S-EMPL-HR

MOVE COBRA-EVENT-DT OF W-EVENT
TO EFFDT OF BIND-DATA-BASEBEN OF S-EMPL-HR

MOVE COBRA-EVENT-DT OF W-EVENT TO STDHRS-FREQ-EFFDT
OF BIND-DATA-BASEBEN OF S-EMPL-HR

SET DELIMCHAR OF BIND-DATA-BASEBEN OF S-EMPL-HR
TO TRUE

.

SETUP-BASEBEN-EXIT.

/*****

* *

XL130-SELECT-EMPL-HR-DATA SECTION.

XL130.

* *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT
SQLRT
SQL-CURSOR OF S-EMPL-HR OF W-CNTL
SQL-STMT OF S-EMPL-HR
BIND-SETUP OF S-EMPL-HR
BIND-DATA OF S-EMPL-HR
SELECT-SETUP OF S-EMPL-HR
SELECT-DATA OF S-EMPL-HR

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-EMPL-HR-DATA(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-EMPL-HR-DATA-EXIT.

/*****

*

*

XL150-LOAD-ELIGIBILITY-DATA SECTION.

XL150.

*

*

* Populate JOB-DATA OF PARTC with eligibility data from all jobs.*

MOVE ZERO TO JOB-COUNT OF PARTC

PERFORM XL160-SELECT-ELIGIBILITY-DATA

PERFORM XL170-FETCH-ELIGIBILITY-DATA

PERFORM UNTIL RTNCD-END OF SQLRT

IF JOB-COUNT-MAX OF PARTC

DISPLAY 'Max # of Active Concurrent Jobs Exceeded'

'EMPLID: ' EMPLID OF PARTC

', EVENT-DT: ' EVENT-DT OF PARTC

MOVE 'LOAD-ELIGIBILITY-DATA'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO JOB-COUNT OF PARTC

SET JOB-IDX TO JOB-COUNT OF PARTC

PERFORM XL180-LOAD-JOB-DATA

PERFORM XL170-FETCH-ELIGIBILITY-DATA

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

SET JOB-IDX TO 1

SEARCH JOB-DATA OF PARTC

AT END

DISPLAY 'No Primary Job in Effect on the day '

'prior to the Event Date'

DISPLAY 'EMPLID: ' EMPLID OF PARTC

', BENEFIT_RCD_NBR: ' BENEFIT-RCD-NO
 OF PARTIC-DATA OF PARTC
 ', EVENT-DT: ' EVENT-DT OF PARTC
MOVE 'LOAD-ELIGIBILITY-DATA'
 TO ERR-SECTION OF SQLRT
SET RTNCD-USER-ERROR OF SQLRT TO TRUE
PERFORM ZZ000-SQL-ERROR

WHEN JOB-IDX > JOB-COUNT OF PARTC

 DISPLAY 'No Primary Job in Effect on the day '
 'prior to the Event Date'

 DISPLAY 'EMPLID: ' EMPLID OF PARTC

 ', BENEFIT_RCD_NBR: ' BENEFIT-RCD-NO
 OF PARTIC-DATA OF PARTC
 ', EVENT-DT: ' EVENT-DT OF PARTC
MOVE 'LOAD-ELIGIBILITY-DATA'
 TO ERR-SECTION OF SQLRT
SET RTNCD-USER-ERROR OF SQLRT TO TRUE
PERFORM ZZ000-SQL-ERROR

WHEN PRIMARY-JOB-YES OF JOB-DATA OF PARTC(JOB-IDX)

 CONTINUE

END-SEARCH

.

XL150-EXIT.

/*****

*

*

XL160-SELECT-ELIGIBILITY-DATA SECTION.

XL160.

*

*

MOVE EMPLID OF PARTC

TO EMPLID OF BIND-DATA OF S-ELGDATA

MOVE EVENT-DT OF PARTC

TO EFFDT OF BIND-DATA OF S-ELGDATA

MOVE EVENT-DT OF PARTC

TO EFFDT-2 OF BIND-DATA OF S-ELGDATA

MOVE EVENT-DT OF PARTC

TO EFFDT-3 OF BIND-DATA OF S-ELGDATA

MOVE EVENT-DT OF PARTC

TO EFFDT-4 OF BIND-DATA OF S-ELGDATA

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-ELGDATA OF W-CNTL

SQL-STMT OF S-ELGDATA

BIND-SETUP OF S-ELGDATA

BIND-DATA OF S-ELGDATA

SELECT-SETUP OF S-ELGDATA

SELECT-DATA OF S-ELGDATA

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT ELIGIBILITY DATA' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

XL160-EXIT.

/*****

* *

XL170-FETCH-ELIGIBILITY-DATA SECTION.

XL170.

* *

INITIALIZE SELECT-DATA OF S-ELGDATA

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-ELGDATA OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-ELIGIBILITY-DATA' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

XL170-EXIT.

/*****

* *

XL180-LOAD-JOB-DATA SECTION.

XL180.

* *

INITIALIZE JOB-DATA OF PARTC(JOB-IDX)

MOVE BENEFIT-RCD-NO OF S-ELGDATA

TO BENEFIT-RCD-NO OF JOB-DATA OF PARTC(JOB-IDX)

MOVE EMPL-RCD-NO OF S-ELGDATA

TO EMPL-RCD-NO OF JOB-DATA OF PARTC(JOB-IDX)

MOVE EFFDT OF SELECT-DATA OF S-ELGDATA

TO EFFDT OF JOB-DATA OF PARTC(JOB-IDX)

MOVE EFFSEQ OF S-ELGDATA

TO EFFSEQ OF JOB-DATA OF PARTC(JOB-IDX)

MOVE EMPL-STATUS OF S-ELGDATA

TO EMPL-STATUS OF JOB-DATA OF PARTC(JOB-IDX)

MOVE BEN-STATUS OF S-ELGDATA

TO BEN-STATUS OF JOB-DATA OF PARTC(JOB-IDX)

MOVE EMPL-TYPE OF S-ELGDATA

TO EMPL-TYPE OF JOB-DATA OF PARTC(JOB-IDX)

MOVE EMPL-CLASS OF S-ELGDATA

TO EMPL-CLASS OF JOB-DATA OF PARTC(JOB-IDX)

MOVE REG-TEMP OF S-ELGDATA

TO REG-TEMP OF JOB-DATA OF PARTC(JOB-IDX)

MOVE FULL-PART-TIME OF S-ELGDATA

TO FULL-PART-TIME OF JOB-DATA OF PARTC(JOB-IDX)

MOVE OFFICER-CD OF S-ELGDATA

TO OFFICER-CD OF JOB-DATA OF PARTC(JOB-IDX)

MOVE COMPANY OF S-ELGDATA

TO COMPANY OF JOB-DATA OF PARTC(JOB-IDX)

MOVE PAYGROUP OF S-ELGDATA

TO PAYGROUP OF JOB-DATA OF PARTC(JOB-IDX)

MOVE SETID-LOCATION OF S-ELGDATA

TO SETID-LOCATION OF JOB-DATA OF PARTC(JOB-IDX)

MOVE LOCATION OF S-ELGDATA

TO LOCATION OF JOB-DATA OF PARTC(JOB-IDX)

MOVE STD-HOURS OF S-ELGDATA

TO STD-HOURS OF JOB-DATA OF PARTC(JOB-IDX)

MOVE FLSA-STATUS OF S-ELGDATA

TO FLSA-STATUS OF JOB-DATA OF PARTC(JOB-IDX)

MOVE SETID-SALARY OF S-ELGDATA

TO SETID-SALARY OF JOB-DATA OF PARTC(JOB-IDX)

MOVE SAL-ADMIN-PLAN OF S-ELGDATA

TO SAL-ADMIN-PLAN OF JOB-DATA OF PARTC(JOB-IDX)

MOVE GRADE OF S-ELGDATA

TO GRADE OF JOB-DATA OF PARTC(JOB-IDX)

MOVE REG-REGION OF S-ELGDATA

TO REG-REGION OF JOB-DATA OF PARTC(JOB-IDX)

MOVE BUSINESS-UNIT OF S-ELGDATA

TO BUSINESS-UNIT OF JOB-DATA OF PARTC(JOB-IDX)

MOVE COMP-FREQUENCY OF S-ELGDATA

TO COMP-FREQUENCY OF JOB-DATA OF PARTC(JOB-IDX)

MOVE COMP-FREQ-TYPE OF S-ELGDATA

TO COMP-FREQ-TYPE OF JOB-DATA OF PARTC(JOB-IDX)

MOVE COMP-FREQ-FACTOR OF S-ELGDATA

TO COMP-FREQ-FACTOR OF JOB-DATA OF PARTC(JOB-IDX)

MOVE STD-HRS-FREQUENCY OF S-ELGDATA

TO STD-HRS-FREQUENCY OF JOB-DATA OF PARTC(JOB-IDX)

MOVE STD-HRS-FREQ-TYPE OF S-ELGDATA

TO STD-HRS-FREQ-TYPE OF JOB-DATA OF PARTC(JOB-IDX)

MOVE STDHRS-FREQ-FACTOR OF S-ELGDATA

TO STDHRS-FREQ-FACTOR OF JOB-DATA OF PARTC(JOB-IDX)

MOVE COMPRATE OF S-ELGDATA

TO COMPRATE OF JOB-DATA OF PARTC(JOB-IDX)

MOVE ANNUAL-RT OF S-ELGDATA

TO ANNUAL-RT OF JOB-DATA OF PARTC(JOB-IDX)

MOVE HOURLY-RT OF S-ELGDATA

TO HOURLY-RT OF JOB-DATA OF PARTC(JOB-IDX)

MOVE ANNL-BENEF-BASE-RT OF S-ELGDATA

TO ANNL-BENEF-BASE-RT OF JOB-DATA OF PARTC(JOB-IDX)

MOVE UNION-CD OF S-ELGDATA

TO UNION-CD OF JOB-DATA OF PARTC(JOB-IDX)

MOVE ELIG-CONFIG1 OF S-ELGDATA

TO ELIG-CONFIG1 OF JOB-DATA OF PARTC(JOB-IDX)

MOVE ELIG-CONFIG2 OF S-ELGDATA

TO ELIG-CONFIG2 OF JOB-DATA OF PARTC(JOB-IDX)

MOVE ELIG-CONFIG3 OF S-ELGDATA

TO ELIG-CONFIG3 OF JOB-DATA OF PARTC(JOB-IDX)

MOVE ELIG-CONFIG4 OF S-ELGDATA

TO ELIG-CONFIG4 OF JOB-DATA OF PARTC(JOB-IDX)

MOVE ELIG-CONFIG5 OF S-ELGDATA

TO ELIG-CONFIG5 OF JOB-DATA OF PARTC(JOB-IDX)

MOVE ELIG-CONFIG6 OF S-ELGDATA

TO ELIG-CONFIG6 OF JOB-DATA OF PARTC(JOB-IDX)

MOVE ELIG-CONFIG7 OF S-ELGDATA

TO ELIG-CONFIG7 OF JOB-DATA OF PARTC(JOB-IDX)

MOVE ELIG-CONFIG8 OF S-ELGDATA

TO ELIG-CONFIG8 OF JOB-DATA OF PARTC(JOB-IDX)

MOVE ELIG-CONFIG9 OF S-ELGDATA

TO ELIG-CONFIG9 OF JOB-DATA OF PARTC(JOB-IDX)

MOVE FTE OF S-ELGDATA

TO FTE OF JOB-DATA OF PARTC(JOB-IDX)

MOVE SERVICE-DT OF S-ELGDATA

TO SERVICE-DT OF JOB-DATA OF PARTC(JOB-IDX)

MOVE TERMINATION-DT OF S-ELGDATA

TO TERMINATION-DT OF JOB-DATA OF PARTC(JOB-IDX)

MOVE COBRA-ACTION OF S-ELGDATA

TO COBRA-ACTION OF JOB-DATA OF PARTC(JOB-IDX)

MOVE PRIMARY-JOB-IND OF S-ELGDATA

TO PRIMARY-JOB-IND OF JOB-DATA OF PARTC(JOB-IDX)

MOVE PRIMARY-FLAG1 OF S-ELGDATA

TO PRIMARY-FLAG1 OF JOB-DATA OF PARTC(JOB-IDX)

MOVE PRIMARY-FLAG2 OF S-ELGDATA

TO PRIMARY-FLAG2 OF JOB-DATA OF PARTC(JOB-IDX)

IF PRIMARY-JOB-YES OF S-ELGDATA

AND BENEFIT-RCD-NO OF S-ELGDATA

= BENEFIT-RCD-NO OF PARTIC-DATA OF PARTC

- * If we've already found a primary job for this BRN,
- * report an error, otherwise load primary job data

SET JOB-IDX2 TO 1

SEARCH JOB-DATA OF PARTC VARYING JOB-IDX2

AT END

SET PRIMARY-JOB-YES OF JOB-DATA OF PARTC(JOB-IDX)
TO TRUE

PERFORM XL190-LOAD-PRIMARY-JOB-DATA

WHEN JOB-IDX2 > JOB-COUNT OF PARTC

SET PRIMARY-JOB-YES OF JOB-DATA OF PARTC(JOB-IDX)
TO TRUE

PERFORM XL190-LOAD-PRIMARY-JOB-DATA

WHEN PRIMARY-JOB-YES OF JOB-DATA OF PARTC(JOB-IDX2)

AND BENEFIT-RCD-NO OF JOB-DATA

OF PARTC(JOB-IDX2)

= BENEFIT-RCD-NO OF S-ELGDATA

AND JOB-IDX2 NOT = JOB-IDX

SET PRIMARY-JOB-NO OF JOB-DATA OF PARTC(JOB-IDX)
TO TRUE

SET MSGID-TOO-MANY-PRIMARY-JOBS-EL OF PYMSG
TO TRUE

MOVE BENEFIT-RCD-NO OF PARTC-DATA OF PARTC
TO MSGDATA1 OF PYMSG

MOVE EMPL-RCD-NO OF JOB-DATA OF PARTC(JOB-IDX2)

TO MSGDATA2 OF PYMSG
MOVE EMPL-RCD-NO OF S-ELGDATA
TO MSGDATA3 OF PYMSG

PERFORM ZM000-MESSAGE

END-SEARCH

END-IF

.

XL180-EXIT.

/*****

*

*

XL190-LOAD-PRIMARY-JOB-DATA SECTION.

XL190.

*

*

* Populate HR-DATA and PARTIC-DATA of PARTC with certain data *

* from the Primary Job. *

MOVE SERVICE-DT OF S-ELGDATA

TO SERVICE-DT OF PARTIC-DATA OF PARTC

MOVE TERMINATION-DT OF S-ELGDATA

TO TERMINATION-DT OF PARTC-DATA OF PARTC
MOVE UNION-CD OF S-ELGDATA

TO UNION-CD OF HR-DATA OF PARTC
MOVE COMPANY OF S-ELGDATA

TO COMPANY OF HR-DATA OF PARTC
MOVE PAYGROUP OF S-ELGDATA

TO PAYGROUP OF HR-DATA OF PARTC
MOVE STD-HOURS OF S-ELGDATA

TO STD-HOURS OF HR-DATA OF PARTC
MOVE COMP-FREQUENCY OF S-ELGDATA

TO COMP-FREQUENCY OF HR-DATA OF PARTC
MOVE COMP-FREQ-TYPE OF S-ELGDATA

TO COMP-FREQ-TYPE OF HR-DATA OF PARTC
MOVE COMP-FREQ-FACTOR OF S-ELGDATA

TO COMP-FREQ-FACTOR OF HR-DATA OF PARTC
MOVE STD-HRS-FREQUENCY OF S-ELGDATA

TO STD-HRS-FREQUENCY OF HR-DATA OF PARTC
MOVE STD-HRS-FREQ-TYPE OF S-ELGDATA

TO STD-HRS-FREQ-TYPE OF HR-DATA OF PARTC
MOVE STDHRS-FREQ-FACTOR OF S-ELGDATA

TO STDHRS-FREQ-FACTOR OF HR-DATA OF PARTC
MOVE COMPRATE OF S-ELGDATA

TO COMPRATE OF HR-DATA OF PARTC
MOVE ANNUAL-RT OF S-ELGDATA

TO ANNUAL-RT OF HR-DATA OF PARTC
MOVE HOURLY-RT OF S-ELGDATA

TO HOURLY-RT OF HR-DATA OF PARTC
MOVE ANNL-BENEF-BASE-RT OF S-ELGDATA
TO ANNL-BENEF-BASE-RT OF HR-DATA OF PARTC

.

XL190-EXIT.

/*****

* *

XM000-OBTAIN-CUR-ELECT-DATA SECTION.

XM000.

* *

PERFORM XM100-OBTAIN-BP-PGM-DATA

PERFORM XM200-OBTAIN-HTH-DATA

PERFORM XM300-OBTAIN-FSA-DATA

.

OBTAIN-BP-PGM-DATA-EXIT.

/*****

* *

XM100-OBTAIN-BP-PGM-DATA SECTION.

XM100.

*

*

PERFORM XM110-SELECT-BP-PGM

PERFORM XM120-FETCH-BP-PGM

PERFORM XM130-SAVE-BP-PGM

.

OBTAIN-BP-PGM-DATA-EXIT.

/*****

*

*

XM110-SELECT-BP-PGM SECTION.

XM110.

*

*

MOVE EMPLID OF W-EVENT TO EMPLID OF S-BP-PGM

MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF S-BP-PGM

SET BENPRG-FOUND-NO OF W-SW TO TRUE

IF CBR-COV-OVERRIDE-YES OF W-EVENT

MOVE CBR-ALT-COV-TRM-DT OF W-EVENT

TO EFFDT OF BIND-DATA OF S-BP-PGM

MOVE CBR-ALT-COV-TRM-DT OF W-EVENT

TO EFFDT-2 OF BIND-DATA OF S-BP-PGM

ELSE

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT OF BIND-DATA OF S-BP-PGM

MOVE COBRA-EVENT-DT OF W-EVENT

TO EFFDT-2 OF BIND-DATA OF S-BP-PGM

END-IF

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-BP-PGM OF W-CNTL

SQL-STMT OF S-BP-PGM

BIND-SETUP OF S-BP-PGM

BIND-DATA OF S-BP-PGM

SELECT-SETUP OF S-BP-PGM

SELECT-DATA OF S-BP-PGM

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-BP-PGM' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-BP-PGM-EXIT.

/*****

* *

XM120-FETCH-BP-PGM SECTION.

XM120.

* *

INITIALIZE SELECT-DATA OF S-BP-PGM

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-BP-PGM OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

MOVE 'FETCH-BP-PGM' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

SET BENPRG-FOUND-YES OF W-SW TO TRUE

END-IF

.

FETCH-BP-PGM-EXIT.

/*****

* *

XM130-SAVE-BP-PGM SECTION.

XM130.

* *

MOVE BENEFIT-PROGRAM OF S-BP-PGM

TO BENEFIT-PROGRAM OF W-CURELT

MOVE ZERO TO COBRA-EVENT-ID OF W-CURELT

.

SAVE-BP-PGM-EXIT.

/*****

* *

XM200-OBTAIN-HTH-DATA SECTION.

XM200.

* *

MOVE ZERO TO WCUR-COUNT OF W-CURELT

PERFORM XM210-SELECT-HTH

PERFORM XM220-FETCH-HTH

PERFORM UNTIL RTNCD-END OF SQLRT

IF WCUR-COUNT-MAX OF W-CURELT

DISPLAY 'Max # of Health Options Exceeded'

MOVE 'OBTAIN-HTH-DATA' TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO WCUR-COUNT OF W-CURELT

PERFORM XM230-SAVE-HTH

PERFORM XM240-SELECT-HTH-DEP

PERFORM XM250-FETCH-HTH-DEP

PERFORM UNTIL RTNCD-END OF SQLRT

ADD 1 TO WCURDB-COUNT OF W-CURELT

PERFORM XM260-SAVE-HTH-DEP

PERFORM XM250-FETCH-HTH-DEP

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

PERFORM XM220-FETCH-HTH

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

OBTAIN-HTH-DATA-EXIT.

/*****

*

*

XM210-SELECT-HTH SECTION.

XM210.

*

*

MOVE EMPLID OF W-EVENT TO EMPLID OF S-HTH

MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF S-HTH

MOVE COBRA-EVENT-ID OF W-CURELT TO COBRA-EVENT-ID OF S-HTH

IF CBR-COV-OVERRIDE-YES OF W-EVENT

MOVE CBR-ALT-COV-TRM-DT OF W-EVENT TO BEGIN-DT OF DTWRK

MOVE -1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO COVERAGE-END-DT OF BIND-DATA OF S-HTH

MOVE CBR-ALT-COV-TRM-DT OF W-EVENT

TO COVERAGE-BEGIN-DT OF BIND-DATA OF S-HTH

MOVE CBR-ALT-COV-TRM-DT OF W-EVENT

TO COVERAGE-BEGIN-DT-2 OF BIND-DATA OF S-HTH

ELSE

MOVE COBRA-EVENT-DT OF W-EVENT TO BEGIN-DT OF DTWRK

MOVE -1 TO DAYS OF DTWRK

SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO COVERAGE-END-DT OF BIND-DATA OF S-HTH

MOVE COBRA-EVENT-DT OF W-EVENT

TO COVERAGE-BEGIN-DT OF BIND-DATA OF S-HTH

MOVE COBRA-EVENT-DT OF W-EVENT

TO COVERAGE-BEGIN-DT-2 OF BIND-DATA OF S-HTH

END-IF

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-HTH OF W-CNTL

SQL-STMT OF S-HTH

BIND-SETUP OF S-HTH

BIND-DATA OF S-HTH

SELECT-SETUP OF S-HTH

SELECT-DATA OF S-HTH

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-HTH' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-HTH-EXIT.

/*****

*

*

XM220-FETCH-HTH SECTION.

XM220.

*

*

INITIALIZE SELECT-DATA OF S-HTH

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-HTH OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

DISPLAY 'Emplid/BenRcd#/CBREvtID: '

EMPLID OF S-HTH '/'

EMPL-RCD-NO OF S-HTH '/'

COBRA-EVENT-ID OF S-HTH

MOVE 'FETCH-HTH' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-HTH-EXIT.

/*****

* *

XM230-SAVE-HTH SECTION.

XM230.

* *

SET WCUR-IDX TO WCUR-COUNT OF W-CURELT

INITIALIZE WCUR-DATA OF W-CURELT(WCUR-IDX)

MOVE PLAN-TYPE OF S-HTH

TO PLAN-TYPE OF WCUR-DATA OF W-CURELT(WCUR-IDX)

MOVE EFFDT OF SELECT-DATA OF S-HTH

TO EFFDT OF WCUR-DATA OF W-CURELT(WCUR-IDX)
MOVE DEDUCTION-END-DT OF SELECT-DATA OF S-HTH
TO DEDUCTION-END-DT OF W-CURELT(WCUR-IDX)
MOVE COVERAGE-BEGIN-DT OF SELECT-DATA OF S-HTH
TO COVERAGE-BEGIN-DT OF W-CURELT(WCUR-IDX)
MOVE COVERAGE-END-DT OF SELECT-DATA OF S-HTH
TO COVERAGE-END-DT OF W-CURELT(WCUR-IDX)
MOVE BENEFIT-PLAN OF S-HTH
TO BENEFIT-PLAN OF WCUR-DATA OF W-CURELT(WCUR-IDX)
MOVE COVRG-CD OF S-HTH
TO COVRG-CD OF WCUR-DATA OF W-CURELT(WCUR-IDX)
MOVE COVERAGE-ELECT OF S-HTH
TO COVERAGE-ELECT OF W-CURELT(WCUR-IDX)
MOVE HLTH-PROVIDER-ID OF S-HTH
TO HLTH-PROVIDER-ID OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
MOVE PREVIOUSLY-SEEN OF S-HTH
TO PREVIOUSLY-SEEN OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
MOVE OTH-INSURANCE-IND OF S-HTH
TO OTH-INSURANCE-IND OF WCUR-DATA
OF W-CURELT(WCUR-IDX)
MOVE OTH-INSURANCE-NAME OF S-HTH
TO OTH-INSURANCE-NAME OF WCUR-DATA
OF W-CURELT(WCUR-IDX)

SAVE-HTH-EXIT.

/*****

* *

XM240-SELECT-HTH-DEP SECTION.

XM240.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF S-HTH-DEP

MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF S-HTH-DEP

MOVE COBRA-EVENT-ID OF W-CURELT

TO COBRA-EVENT-ID OF S-HTH-DEP

MOVE PLAN-TYPE OF S-HTH TO PLAN-TYPE OF S-HTH-DEP

MOVE BENEFIT-NO OF S-HTH TO BENEFIT-NO OF S-HTH-DEP

MOVE EFFDT OF SELECT-DATA OF S-HTH TO EFFDT OF S-HTH-DEP

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-HTH-DEP OF W-CNTL

SQL-STMT OF S-HTH-DEP

BIND-SETUP OF S-HTH-DEP

BIND-DATA OF S-HTH-DEP

SELECT-SETUP OF S-HTH-DEP

SELECT-DATA OF S-HTH-DEP

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-HTH-DEP' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-HTH-DEP-EXIT.

/*****

* *

XM250-FETCH-HTH-DEP SECTION.

XM250.

* *

*****/

INITIALIZE SELECT-DATA OF S-HTH-DEP

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-HTH-DEP OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

DISPLAY 'Emplid/EmplRcd#/CBREvtID: '

EMPLID OF S-HTH-DEP '/'

EMPL-RCD-NO OF S-HTH-DEP '/'

COBRA-EVENT-ID OF S-HTH-DEP

DISPLAY 'PlnTyp/BenPln/EffDt: '

PLAN-TYPE OF S-HTH-DEP '/'

BENEFIT-NO OF S-HTH-DEP '/'

EFFDT OF S-HTH-DEP

MOVE 'FETCH-HTH-DEP' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-HTH-DEP-EXIT.

/*****

*

*

XM260-SAVE-HTH-DEP SECTION.

XM260.

*

*

SET WCURDB-IDX TO WCURDB-COUNT OF W-CURELT

INITIALIZE WCURDB-DATA OF W-CURELT(WCURDB-IDX)

MOVE PLAN-TYPE OF S-HTH-DEP

TO PLAN-TYPE OF WCURDB-DATA OF W-CURELT(WCURDB-IDX)
MOVE DEPENDENT-BENEF OF S-HTH-DEP
TO DEPENDENT-BENEF OF W-CURELT(WCURDB-IDX)

MOVE HLTH-PROVIDER-ID OF S-HTH-DEP
TO HLTH-PROVIDER-ID OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)

MOVE PREVIOUSLY-SEEN OF S-HTH-DEP
TO PREVIOUSLY-SEEN OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)

MOVE OTH-INSURANCE-IND OF S-HTH-DEP
TO OTH-INSURANCE-IND OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)

MOVE OTH-INSURANCE-NAME OF S-HTH-DEP
TO OTH-INSURANCE-NAME OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)

MOVE BENEFIT-PLAN OF S-HTH
TO BENEFIT-PLAN OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)

MOVE COVRG-CD OF S-HTH TO COVRG-CD OF WCURDB-DATA
OF W-CURELT(WCURDB-IDX)

SET OVERAGE-NO OF W-CURELT(WCURDB-IDX) TO TRUE

.

SAVE-HTH-DEP-EXIT.

/*****

* *

XM300-OBTAIN-FSA-DATA SECTION.

XM300.

* *

PERFORM XM310-SELECT-FSA

PERFORM XM320-FETCH-FSA

PERFORM UNTIL RTNCD-END OF SQLRT

IF WCUR-COUNT-MAX OF W-CURELT

DISPLAY 'Max # of FSA Options Exceeded'

MOVE 'OBTAIN-FSA-DATA' TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO WCUR-COUNT OF W-CURELT

PERFORM XM330-SAVE-FSA

SET RTNCD-OK OF SQLRT TO TRUE

PERFORM XM320-FETCH-FSA

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

OBTAIN-FSA-DATA-EXIT.

/*****

* *

XM310-SELECT-FSA SECTION.

XM310.

* *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-FSA OF W-CNTL

SQL-STMT OF S-FSA

BIND-SETUP OF S-HTH

BIND-DATA OF S-HTH

SELECT-SETUP OF S-FSA

SELECT-DATA OF S-FSA

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-FSA' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

SELECT-FSA-EXIT.

/*****

* *

XM320-FETCH-FSA SECTION.

XM320.

* *

INITIALIZE SELECT-DATA OF S-FSA

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-FSA OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

DISPLAY 'Emplid/EmplRcd#/CBREvtID: '

EMPLID OF S-HTH-DEP '/'

EMPL-RCD-NO OF S-HTH-DEP '/'

COBRA-EVENT-ID OF S-HTH-DEP

DISPLAY 'PlnTyp/BenPln/EffDt: '

PLAN-TYPE OF S-HTH-DEP '/'

BENEFIT-NO OF S-HTH-DEP '/'

EFFDT OF S-HTH-DEP

MOVE 'FETCH-FSA' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-FSA-EXIT.

/*****

* *

XM330-SAVE-FSA SECTION.

XM330.

* *

SET WCUR-IDX TO WCUR-COUNT OF W-CURELT

INITIALIZE WCUR-DATA OF W-CURELT(WCUR-IDX)

MOVE PLAN-TYPE OF S-FSA

TO PLAN-TYPE OF WCUR-DATA OF W-CURELT(WCUR-IDX)

MOVE EFFDT OF SELECT-DATA OF S-FSA

TO EFFDT OF WCUR-DATA OF W-CURELT(WCUR-IDX)

MOVE DEDUCTION-END-DT OF SELECT-DATA OF S-FSA

TO DEDUCTION-END-DT OF W-CURELT(WCUR-IDX)

MOVE COVERAGE-BEGIN-DT OF SELECT-DATA OF S-FSA
TO COVERAGE-BEGIN-DT OF W-CURELT(WCUR-IDX)
MOVE COVERAGE-END-DT OF SELECT-DATA OF S-FSA
TO COVERAGE-END-DT OF W-CURELT(WCUR-IDX)
MOVE BENEFIT-PLAN OF S-FSA
TO BENEFIT-PLAN OF WCUR-DATA OF W-CURELT(WCUR-IDX)
MOVE COVERAGE-ELECT OF S-FSA
TO COVERAGE-ELECT OF W-CURELT(WCUR-IDX)
MOVE EMPL-CONTRBUTN-AMT OF S-FSA
TO EMPL-CONTRBUTN-AMT OF W-CURELT(WCUR-IDX)
MOVE ANN-EX-CREDIT-FSA OF S-FSA
TO ANN-EX-CREDIT-FSA OF W-CURELT(WCUR-IDX)
MOVE ANNUAL-PLEDGE OF S-FSA
TO ANNUAL-PLEDGE OF W-CURELT(WCUR-IDX)
MOVE FSA-SUB-AMT-YTD OF S-FSA
TO FSA-SUB-AMT-YTD OF W-CURELT(WCUR-IDX)
MOVE FSA-APR-AMT-YTD OF S-FSA
TO FSA-APR-AMT-YTD OF W-CURELT(WCUR-IDX)
MOVE FSA-PD-AMT-YTD OF S-FSA
TO FSA-PD-AMT-YTD OF W-CURELT(WCUR-IDX)

.

SAVE-FSA-EXIT.

/*****

*

*

XN000-LOAD-DEPENDENT SECTION.

XN000.

*

*

IF LOAD-EMPL-DEP-YES OF W-SW

ADD 1 TO WDEPEND-COUNT OF W-DEPEND

PERFORM XN400-LOAD-EMPLOYEE-DEPENDENT

PERFORM XN500-SET-MIN-MAX-COVRG-END-DT

END-IF

MOVE SPACES TO WDEPEND-BREAK OF W-DEPEND

PERFORM XN100-SELECT-DEPENDENT

PERFORM XN200-FETCH-DEPENDENT

PERFORM UNTIL RTNCD-END OF SQLRT

IF WDEPEND-COUNT-MAX OF W-DEPEND

DISPLAY 'Maximum Number of Depend-Benefs Exceeded'

DISPLAY 'Emplid/DepCnt: '

EMPLID OF W-EVENT

WDEPEND-COUNT OF W-DEPEND

MOVE 'LOAD-DEPENDENT' TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE
PERFORM ZZ000-SQL-ERROR
END-IF

IF DEPENDENT-BENEF OF S-DEPEND
NOT = WDEPEND-BREAK OF W-DEPEND

ADD 1 TO WDEPEND-COUNT OF W-DEPEND
PERFORM XN300-SAVE-DEPENDENT
PERFORM XN600-LOAD-DEPENDENT-NID
END-IF

IF WDEPEFF-COUNT-MAX OF W-DEPEND

DISPLAY 'Maximum Depend-Benef History Rows Exceeded'
DISPLAY 'Emplid/HistCnt: '
EMPLID OF W-EVENT
WDEPEFF-COUNT OF W-DEPEND
MOVE 'LOAD-DEPENDENT' TO ERR-SECTION OF SQLRT
SET RTNCD-USER-ERROR OF SQLRT TO TRUE
PERFORM ZZ000-SQL-ERROR
END-IF

ADD 1 TO WDEPEFF-COUNT OF W-DEPEND
PERFORM XN350-SAVE-DEP-EFF-DATA
PERFORM XN500-SET-MIN-MAX-COVRG-END-DT

PERFORM XN200-FETCH-DEPENDENT

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

LOAD-DEPENDENT-EXIT.

/*****

*

*

XN100-SELECT-DEPENDENT SECTION.

XN100.

*

*

MOVE COBRA-EVENT-DT OF W-EVENT

TO WDEPEND-ASOFDT OF W-DEPEND

MOVE EMPLID OF W-EVENT TO EMPLID OF S-DEPEND

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT-1 OF S-DEPEND

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT-2 OF S-DEPEND

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT-3 OF S-DEPEND

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-DEPEND OF W-CNTL

SQL-STMT OF S-DEPEND

BIND-SETUP OF S-DEPEND

BIND-DATA OF S-DEPEND

SELECT-SETUP OF S-DEPEND

SELECT-DATA OF S-DEPEND

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-DEPENDENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-DEPENDENT-EXIT.

/*****

* * *

XN200-FETCH-DEPENDENT SECTION.

XN200.

* * *

INITIALIZE SELECT-DATA OF S-DEPEND

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-DEPEND OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF NOT RTNCD-END OF SQLRT

DISPLAY 'Emplid: ' EMPLID OF S-DEPEND

MOVE 'FETCH-DEPENDENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

FETCH-DEPENDENT-EXIT.

/*****

* *

XN300-SAVE-DEPENDENT SECTION.

XN300.

* *

MOVE DEPENDENT-BENEF OF S-DEPEND

TO WDEPEND-BREAK OF W-DEPEND

SET WDEPEND-IDX TO WDEPEND-COUNT OF W-DEPEND

INITIALIZE WDEPEND-DATA OF W-DEPEND(WDEPEND-IDX)
MOVE DEPENDENT-BENEF OF S-DEPEND
 TO DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)
MOVE SAME-ADDRESS-EMPL OF S-DEPEND
 TO SAME-ADDRESS-EMPL OF W-DEPEND(WDEPEND-IDX)
IF SAME-ADDRESS-EMPL-NO OF S-DEPEND
 MOVE DEP-ADDRESS-SBR OF S-DEPEND
 TO DEP-ADDRESS-SBR OF W-DEPEND(WDEPEND-IDX)
END-IF
MOVE ADDRESS-TYPE OF S-DEPEND
 TO ADDRESS-TYPE OF W-DEPEND(WDEPEND-IDX)
MOVE BIRTHDATE OF S-DEPEND
 TO BIRTHDATE OF W-DEPEND(WDEPEND-IDX)
MOVE DT-OF-DEATH OF S-DEPEND
 TO DT-OF-DEATH OF W-DEPEND(WDEPEND-IDX)
MOVE MEDICARE-ENTLD-DT OF S-DEPEND
 TO MEDICARE-ENTLD-DT OF W-DEPEND(WDEPEND-IDX)
MOVE COBRA-ACTION OF S-DEPEND
 TO COBRA-ACTION OF W-DEPEND(WDEPEND-IDX)
MOVE COBRA-EVENT-DT OF S-DEPEND
 TO COBRA-EVENT-DT OF W-DEPEND(WDEPEND-IDX)
MOVE COBRA-EMPLID OF S-DEPEND
 TO COBRA-EMPLID OF W-DEPEND(WDEPEND-IDX)
MOVE WDEPEFF-COUNT OF W-DEPEND
 TO WDEPEFF-START OF W-DEPEND(WDEPEND-IDX)
ADD 1 TO WDEPEFF-START OF W-DEPEND(WDEPEND-IDX)

SAVE-DEPENDENT-EXIT.

/*****

* *

XN350-SAVE-DEP-EFF-DATA SECTION.

XN350.

* *

SET WDEPEFF-IDX TO WDEPEFF-COUNT OF W-DEPEND

INITIALIZE WDEPEFF-DATA OF W-DEPEND(WDEPEFF-IDX)

MOVE DEPENDENT-BENEF OF S-DEPEND

TO DEPENDENT-BEN OF W-DEPEND(WDEPEFF-IDX)

MOVE EFFDT OF S-DEPEND

TO EFFDT OF W-DEPEND(WDEPEFF-IDX)

MOVE COVERED-PERSON-TYP OF S-DEPEND

TO COVERED-PERSON-TYP OF W-DEPEND(WDEPEFF-IDX)

MOVE DEP-BENEF-TYPE OF S-DEPEND

TO DEP-BENEF-TYPE OF W-DEPEND(WDEPEFF-IDX)

MOVE MAR-STATUS OF S-DEPEND

TO MAR-STATUS OF W-DEPEND(WDEPEFF-IDX)

MOVE SEX OF S-DEPEND TO SEX OF W-DEPEND(WDEPEFF-IDX)

MOVE STUDENT OF S-DEPEND

TO STUDENT-STATUS OF W-DEPEND(WDEPEFF-IDX)

MOVE DISABLED OF S-DEPEND

TO DISABLED OF W-DEPEND(WDEPEFF-IDX)

MOVE SMOKER OF S-DEPEND

TO SMOKER OF W-DEPEND(WDEPEFF-IDX)

SET AGE-LIMIT-NO OF W-DEPEND(WDEPEFF-IDX) TO TRUE

IF EFF-STATUS-ACTIVE OF SELECT-DATA OF S-DEPEND

MOVE AGE-LIMIT-FLG OF S-DEPEND

TO AGE-LIMIT-FLG OF W-DEPEND(WDEPEFF-IDX)

END-IF

.

SAVE-DEP-EFF-DATA-EXIT.

/*****

*

*

XN400-LOAD-EMPLOYEE-DEPENDENT SECTION.

XN400.

*

*

PERFORM XN410-GET-PERSONAL-DATA

PERFORM XN411-GET-SMOKER-DATA

PERFORM XN420-CHECK-DISABLED-EMP

SET WDEPEND-IDX TO WDEPEND-COUNT OF W-DEPEND
INITIALIZE WDEPEND-DATA OF W-DEPEND(WDEPEND-IDX)
MOVE '00' TO DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)
MOVE EMPLID OF W-EVENT
TO COBRA-EMPLID OF W-DEPEND(WDEPEND-IDX)
MOVE BIRTHDATE OF W-EVENT
TO BIRTHDATE OF W-DEPEND(WDEPEND-IDX)
MOVE MEDICARE-ENTLD-DT OF W-EVENT
TO MEDICARE-ENTLD-DT OF W-DEPEND(WDEPEND-IDX)
MOVE POSTAL OF W-EVENT TO POSTAL OF W-DEPEND(WDEPEND-IDX)
MOVE STATE OF W-EVENT TO STATE OF W-DEPEND(WDEPEND-IDX)
MOVE COUNTRY OF W-EVENT TO COUNTRY OF W-DEPEND(WDEPEND-IDX)
MOVE WDEPEFF-COUNT OF W-DEPEND
TO WDEPEFF-START OF W-DEPEND(WDEPEND-IDX)
ADD 1 TO WDEPEFF-START OF W-DEPEND(WDEPEND-IDX)

ADD 1 TO WDEPEFF-COUNT OF W-DEPEND
SET WDEPEFF-IDX TO WDEPEFF-COUNT OF W-DEPEND
INITIALIZE WDEPEFF-DATA OF W-DEPEND(WDEPEFF-IDX)

MOVE '00' TO DEPENDENT-BEN OF W-DEPEND(WDEPEFF-IDX)
MOVE COBRA-EVENT-DT OF W-EVENT
TO EFFDT OF W-DEPEND(WDEPEFF-IDX)
SET EMPLOYEE OF W-DEPEND(WDEPEFF-IDX) TO TRUE
MOVE DISABLED OF W-EVENT
TO DISABLED OF W-DEPEND(WDEPEFF-IDX)

LOAD-EMPLOYEE-DEPENDENT-EXIT.

/*****

* *

XN410-GET-PERSONAL-DATA SECTION.

XN410.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF S-PERDATA

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT OF S-PERDATA

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT-2 OF S-PERDATA

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT-3 OF S-PERDATA

MOVE COBRA-EVENT-DT OF W-EVENT TO EFFDT-4 OF S-PERDATA

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PERDATA OF W-CNTL

SQL-STMT OF S-PERDATA

BIND-SETUP OF S-PERDATA

BIND-DATA OF S-PERDATA

SELECT-SETUP OF S-PERDATA

SELECT-DATA OF S-PERDATA

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-PERSONAL-DATA(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-PERDATA

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-PERDATA OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

DISPLAY 'Emplid/CBREvtID: '

EMPLID OF S-PERDATA '/'

EFFDT OF S-PERDATA

MOVE 'GET-PERSONAL-DATA(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

MOVE BIRTHDATE OF S-PERDATA TO BIRTHDATE OF W-EVENT
SET DISABLED-NO OF W-EVENT TO TRUE
MOVE MEDICARE-ENTLD-DT OF S-PERDATA
TO MEDICARE-ENTLD-DT OF W-EVENT

MOVE SEX OF S-PERDATA TO SEX OF W-EVENT
MOVE HIGHLY-COMP-EMPL-C OF S-PERDATA
TO HIGHLY-COMP-EMPL-C OF W-EVENT

MOVE COUNTRY OF S-PERDATA TO COUNTRY OF W-EVENT
MOVE ADDRESS1 OF S-PERDATA TO ADDRESS1 OF W-EVENT
MOVE ADDRESS2 OF S-PERDATA TO ADDRESS2 OF W-EVENT
MOVE ADDRESS3 OF S-PERDATA TO ADDRESS3 OF W-EVENT
MOVE ADDRESS4 OF S-PERDATA TO ADDRESS4 OF W-EVENT
MOVE CITY OF S-PERDATA TO CITY OF W-EVENT

MOVE NUM1 OF S-PERDATA TO NUM1 OF W-EVENT
MOVE NUM2 OF S-PERDATA TO NUM2 OF W-EVENT

MOVE HOUSE-TYPE OF S-PERDATA TO HOUSE-TYPE OF W-EVENT
MOVE ADDR-FIELD1 OF S-PERDATA TO ADDR-FIELD1 OF W-EVENT
MOVE ADDR-FIELD2 OF S-PERDATA TO ADDR-FIELD2 OF W-EVENT
MOVE ADDR-FIELD3 OF S-PERDATA TO ADDR-FIELD3 OF W-EVENT
MOVE COUNTY OF S-PERDATA TO COUNTY OF W-EVENT

MOVE STATE OF S-PERDATA TO STATE OF W-EVENT
MOVE POSTAL OF S-PERDATA TO POSTAL OF W-EVENT

MOVE GEO-CODE OF S-PERDATA TO GEO-CODE OF W-EVENT
MOVE IN-CITY-LIMIT OF S-PERDATA TO IN-CITY-LIMIT OF W-EVENT

GET-PERSONAL-DATA-EXIT.

/*****

* *

XN411-GET-SMOKER-DATA SECTION.

XN411.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF S-SMOKER

MOVE COBRA-EVENT-DT OF W-EVENT TO SMOKER-DT OF S-SMOKER

MOVE COBRA-EVENT-DT OF W-EVENT TO SMOKER-DT-1 OF S-SMOKER

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-SMOKER OF W-CNTL

SQL-STMT OF S-SMOKER

BIND-SETUP OF S-SMOKER

BIND-DATA OF S-SMOKER

SELECT-SETUP OF S-SMOKER

SELECT-DATA OF S-SMOKER

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-SMOKER-DATA(SELECT)'

```
        TO ERR-SECTION OF SQLRT
    PERFORM ZZ000-SQL-ERROR
END-IF

INITIALIZE SELECT-DATA OF S-SMOKER

CALL 'PTPSQLRT' USING  ACTION-FETCH OF SQLRT
                        SQLRT
                        SQL-CURSOR OF S-SMOKER OF W-CNTL
IF RTNCD-ERROR OF SQLRT

    IF RTNCD-END OF SQLRT

        SET RTNCD-OK OF SQLRT TO TRUE
    ELSE
        DISPLAY 'Emplid/CBREvtID: '
            EMPLID OF S-SMOKER '/'
            SMOKER-DT OF S-SMOKER
        MOVE 'GET-SMOKER-DATA(FETCH)'
            TO ERR-SECTION OF SQLRT
        PERFORM ZZ000-SQL-ERROR
    END-IF
ELSE
    MOVE SMOKER OF S-SMOKER TO SMOKER OF W-EVENT
END-IF
```

GET-SMOKER-DATA-EXIT.

/*****

*

*

XN420-CHECK-DISABLED-EMP SECTION.

XN420.

*

*

MOVE EMPLID OF W-EVENT TO EMPLID OF S-DISABLE

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-DISABLE OF W-CNTL

SQL-STMT OF S-DISABLE

BIND-SETUP OF S-DISABLE

BIND-DATA OF S-DISABLE

SELECT-SETUP OF S-DISABLE

SELECT-DATA OF S-DISABLE

IF RTNCD-ERROR OF SQLRT

MOVE 'CHECK-DISABLED-EMP(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

IF RTNCD-OK OF SQLRT

INITIALIZE SELECT-DATA OF S-DISABLE

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-DISABLE OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

MOVE 'CHECK-DISABLED-EMP(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

IF DISABLED-YES OF S-DISABLE

OR DISABLED-VET-YES OF S-DISABLE

SET DISABLED-YES OF W-EVENT TO TRUE

END-IF

END-IF

END-IF

.

CHECK-DISABLED-EMP-EXIT.

/*****

* *

XN500-SET-MIN-MAX-COVRG-END-DT SECTION.

XN500.

* *

IF EMPLOYEE OF W-DEPEND(WDEPEFF-IDX)

AND MEDICARE-ENTLD-DT OF W-EVENT NOT = SPACE

MOVE MEDICARE-ENTLD-DT OF W-EVENT

TO BEGIN-DT OF DTWRK

MOVE 36 TO MONTHS OF DTWRK

SET OPTION-ADD-MONTHS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO DEP-MIN-COVRG-END-DT OF W-EVENT

ELSE

MOVE LOW-VALUE TO DEP-MIN-COVRG-END-DT OF W-EVENT
END-IF

IF (EMPLOYEE OF W-DEPEND(WDEPEFF-IDX)
OR SPOUSE OF W-DEPEND(WDEPEFF-IDX)
OR SAME-SEX-SPOUSE OF W-DEPEND(WDEPEFF-IDX)
OR EX-SPOUSE OF W-DEPEND(WDEPEFF-IDX))
AND (MEDICARE-ENTLD-DT
OF W-DEPEND(WDEPEND-IDX) NOT = SPACE)

MOVE MEDICARE-ENTLD-DT OF W-DEPEND(WDEPEND-IDX)
TO MAX-COVRG-END-DT OF W-DEPEND(WDEPEND-IDX)
ELSE
MOVE HIGH-VALUE
TO MAX-COVRG-END-DT OF W-DEPEND(WDEPEND-IDX)
END-IF

.

SET-MIN-MAX-COVRG-END-DT-EXIT.

/*****

* *

XN600-LOAD-DEPENDENT-NID SECTION.

XN600.

* *

PERFORM XN610-SELECT-DEPENDENT-NID

PERFORM XN620-FETCH-DEPENDENT-NID

PERFORM UNTIL RTNCD-END OF SQLRT

IF WDEPNID-COUNT-MAX OF W-DEPNID

DISPLAY 'Maximum Number of Dep-Benef-NIDs Exceeded'

MOVE 'LOAD-DEPENDENT-NID' TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

END-IF

ADD 1 TO WDEPNID-COUNT OF W-DEPNID

PERFORM XN630-SAVE-DEPENDENT-NID

PERFORM XN620-FETCH-DEPENDENT-NID

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

LOAD-DEPEND-BENEF-EXIT.

/*****

* *

XN610-SELECT-DEPENDENT-NID SECTION.

XN610.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF S-DEP-NID

MOVE DEPENDENT-BENEF OF S-DEPEND

TO DEPENDENT-BENEF OF S-DEP-NID

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-DEP-NID OF W-CNTL

SQL-STMT OF S-DEP-NID

BIND-SETUP OF S-DEP-NID

BIND-DATA OF S-DEP-NID

SELECT-SETUP OF S-DEP-NID

SELECT-DATA OF S-DEP-NID

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-DEPENDENT-NID' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

SELECT-DEPENDENT-NID-EXIT.

/*****

*

*

XN620-FETCH-DEPENDENT-NID SECTION.

XN620.

*

*

INITIALIZE SELECT-DATA OF S-DEP-NID

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-DEP-NID OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF NOT RTNCD-END OF SQLRT

DISPLAY 'Emplid/DepBenef: '

EMPLID OF S-DEP-NID '/'

DEPENDENT-BENEF OF S-DEP-NID

MOVE 'FETCH-DEPENDENT-NID' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

FETCH-DEPENDENT-EXIT.

/*****

* *

XN630-SAVE-DEPENDENT-NID SECTION.

XN630.

* *

SET WDEPNID-IDX TO WDEPNID-COUNT OF W-DEPNID

INITIALIZE WDEPNID-DATA OF W-DEPNID(WDEPNID-IDX)

MOVE DEPENDENT-BENEF OF S-DEP-NID

TO DEPENDENT-BENEF OF W-DEPNID(WDEPNID-IDX)

MOVE NID-COUNTRY OF S-DEP-NID

TO NID-COUNTRY OF W-DEPNID(WDEPNID-IDX)

MOVE NATIONAL-ID-TYPE OF S-DEP-NID

TO NATIONAL-ID-TYPE OF W-DEPNID(WDEPNID-IDX)

MOVE NATIONAL-ID OF S-DEP-NID

TO NATIONAL-ID OF W-DEPNID(WDEPNID-IDX)

MOVE PRIMARY-NID OF S-DEP-NID

TO PRIMARY-NID OF W-DEPNID(WDEPNID-IDX)

SAVE-DEPENDENT-NID-EXIT.

/*****

* *

XN700-SRCH-DEP-EFF-DATA SECTION.

XN700.

* *

- * This routine assumes the caller has already established the
- * pointer to the desired Dependent through W-DEPEND(WDEPEND-IDX)
- * and has set the target effective date in SEARCH-EFFDT. If the
- * routine fails to find an appropriate EFFDT it returns the
- * lowest date available.

SET ERROR-NO OF W-SW TO TRUE

SET WDEPEFF-IDX TO WDEPEFF-START OF W-DEPEND(WDEPEND-IDX)

SEARCH WDEPEFF-DATA

AT END

SET ERROR-YES OF W-SW TO TRUE

WHEN WDEPEFF-IDX > WDEPEFF-COUNT OF W-DEPEND

SET ERROR-YES OF W-SW TO TRUE

WHEN DEPENDENT-BEN OF W-DEPEND(WDEPEFF-IDX)

NOT = DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

SET ERROR-YES OF W-SW TO TRUE

WHEN EFFDT OF W-DEPEND(WDEPEFF-IDX)

<= SEARCH-EFFDT OF W-DEPEND

CONTINUE

END-SEARCH

IF ERROR-YES OF W-SW

SET ERROR-NO OF W-SW TO TRUE

SET WDEPEFF-IDX DOWN BY 1

END-IF

.

SRCH-DEP-EFF-DATA-EXIT.

/*****

* *

XO100-GET-MIN-DEP-RULE SECTION.

XO100.

* GET MINIMUM DEP RULE FOR BEN PROG THAT WILL GIVE EARLIEST OVER *

* AGE DATE FROM ALL COBRA PLAN TYPES AND SAVE IT FOR OVERAGE PROC*

* *

PERFORM XO800-DEFAULT-DEP-RULE

MOVE BENEFIT-PROGRAM OF W-CBRDEFN

TO BENEFIT-PROGRAM OF BIND-DATA OF S-DEPRUL

MOVE EFFDT OF W-CBRDEFN

TO EFFDT OF BIND-DATA OF S-DEPRUL

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-DEPRUL

BIND-SETUP OF S-DEPRUL

BIND-DATA OF S-DEPRUL

SELECT-SETUP OF S-DEPRUL

SELECT-DATA OF S-DEPRUL

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-DEP-RULE' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-DEPRUL

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

IF NOT RTNCD-END OF SQLRT

MOVE 'FETCH-DEP-RULE' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

MOVE CORR SELECT-DATA OF S-DEPRUL TO W-MIN-DEPRULE

END-IF

SET RTNCD-OK OF SQLRT TO TRUE

.

LOAD-DEP-RULE-EXIT.

/*****

* *

XO500-LOAD-DEP-RULE-PLANDF SECTION.

XO500.

* *

PERFORM VARYING PLANDF-IDX FROM 1 BY 1

UNTIL PLANDF-IDX > PLANDF-COUNT OF PDEFN

SET WCBRPLN-IDX TO PLANDF-IDX

MOVE PLAN-TYPE OF PLANDF-DATA(PLANDF-IDX)

TO PLAN-TYPE

OF WCBRPLN-DATA OF W-CBRDEFN(WCBRPLN-IDX)

MOVE DEP-RULE-ID OF PLANDF-DATA(PLANDF-IDX)

TO DEP-RULE-ID

OF WCBRPLN-DATA OF W-CBRDEFN(WCBRPLN-IDX)

MOVE DEP-AGE-LIMIT OF PLANDF-DATA(PLANDF-IDX)

TO DEP-AGE-LIMIT
OF WCBRPLN-DATA OF W-CBRDEFN(WCBRPLN-IDX)
MOVE STUDENT-AGE-LIMIT OF PLANDF-DATA(PLANDF-IDX)
TO STUDENT-AGE-LIMIT
OF WCBRPLN-DATA OF W-CBRDEFN(WCBRPLN-IDX)
MOVE EXCL-DISABLED-AGE OF PLANDF-DATA(PLANDF-IDX)
TO EXCL-DISABLED-AGE
OF WCBRPLN-DATA OF W-CBRDEFN(WCBRPLN-IDX)
MOVE DEPENDENT-MARRIAGE OF PLANDF-DATA(PLANDF-IDX)
TO DEPENDENT-MARRIAGE
OF WCBRPLN-DATA OF W-CBRDEFN(WCBRPLN-IDX)
MOVE OVERAGE-PD-BEG-CD OF PLANDF-DATA(PLANDF-IDX)
TO OVERAGE-PD-BEG-CD
OF WCBRPLN-DATA OF W-CBRDEFN(WCBRPLN-IDX)
END-PERFORM

.

LOAD-DEP-RULE-PLANDF-EXIT.

/*****

*

*

XO800-DEFAULT-DEP-RULE SECTION.

XO800.

*

*

MOVE 99 TO DEP-AGE-LIMIT OF W-DEPRULE
MOVE 99 TO STUDENT-AGE-LIMIT OF W-DEPRULE
MOVE 'Y' TO EXCL-DISABLED-AGE OF W-DEPRULE
MOVE 'N' TO DEPENDENT-MARRIAGE OF W-DEPRULE
MOVE 'E' TO OVERAGE-PD-BEG-CD OF W-DEPRULE

.

DEFAULT-DEP-RULE-EXIT.

/*****

* *

XO900-SEARCH-DEP-RULE SECTION.

XO900.

* *

PERFORM XO800-DEFAULT-DEP-RULE

MOVE 0 TO DEPRUL-PTR

SET WCBRPLN-IDX TO 1

SEARCH WCBRPLN-DATA

AT END

DISPLAY 'Max # Dep-Rule-Id Exceeded PSPCOBRA, Plan Type '

PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
MOVE 'SEARCH-DEP-RULE' TO ERR-SECTION OF SQLRT
SET RTNCD-USER-ERROR OF SQLRT TO TRUE
PERFORM ZZ000-SQL-ERROR

WHEN WCBRPLN-IDX > WCBRPLN-COUNT OF W-CBRDEFN

CONTINUE

WHEN PLAN-TYPE

OF WCBRPLN-DATA OF W-CBRDEFN(WCBRPLN-IDX)
= PLAN-TYPE
OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

SET DEPRUL-PTR TO WCBRPLN-IDX

END-SEARCH

IF DEPRUL-PTR NOT = 0

MOVE DEP-AGE-LIMIT

OF WCBRPLN-DATA OF W-CBRDEFN(DEPRUL-PTR)
TO DEP-AGE-LIMIT OF W-DEPRULE

MOVE STUDENT-AGE-LIMIT

OF WCBRPLN-DATA OF W-CBRDEFN(DEPRUL-PTR)
TO STUDENT-AGE-LIMIT OF W-DEPRULE

MOVE EXCL-DISABLED-AGE

OF WCBRPLN-DATA OF W-CBRDEFN(DEPRUL-PTR)
TO EXCL-DISABLED-AGE OF W-DEPRULE
MOVE DEPENDENT-MARRIAGE

OF WCBRPLN-DATA OF W-CBRDEFN(DEPRUL-PTR)
TO DEPENDENT-MARRIAGE OF W-DEPRULE
MOVE OVERAGE-PD-BEG-CD

OF WCBRPLN-DATA OF W-CBRDEFN(DEPRUL-PTR)
TO OVERAGE-PD-BEG-CD OF W-DEPRULE

END-IF

.

SEARCH-DEP-RULE-EXIT.

/*****

* *

XO950-SEARCH-DEP-PLNDFN SECTION.

XO950.

* *

PERFORM XO800-DEFAULT-DEP-RULE

MOVE 0 TO DEPRUL-PTR

SET PLANDF-IDX TO 1

SEARCH PLANDF-DATA

AT END

DISPLAY 'Max # Dep-Rule-Id Exceeded PSPCOBRA, Plan Type '

PLAN-TYPE OF PLANDF-DATA(PLANDF-IDX)

MOVE 'SEARCH-DEP-PLNDFN' TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

WHEN PLANDF-IDX > PLANDF-COUNT OF PDEFN

CONTINUE

WHEN PLAN-TYPE OF PLANDF-DATA(PLANDF-IDX)

= PLAN-TYPE OF W-SAVE

SET DEPRUL-PTR TO PLANDF-IDX

END-SEARCH

IF DEPRUL-PTR NOT= 0

MOVE DEP-AGE-LIMIT OF PLANDF-DATA(DEPRUL-PTR)

TO DEP-AGE-LIMIT OF W-DEPRULE

MOVE STUDENT-AGE-LIMIT OF PLANDF-DATA(DEPRUL-PTR)

TO STUDENT-AGE-LIMIT OF W-DEPRULE
MOVE EXCL-DISABLED-AGE OF PLANDF-DATA(DEPRUL-PTR)
TO EXCL-DISABLED-AGE OF W-DEPRULE
MOVE DEPENDENT-MARRIAGE OF PLANDF-DATA(DEPRUL-PTR)
TO DEPENDENT-MARRIAGE OF W-DEPRULE
MOVE OVERAGE-PD-BEG-CD OF PLANDF-DATA(DEPRUL-PTR)
TO OVERAGE-PD-BEG-CD OF W-DEPRULE
END-IF

.

SEARCH-DEP-PLNDFN.

/*****

* *

XP000-QUALIFY-DEPENDENT SECTION.

XP000.

* *

SET WCBR-BEN-IDX TO 1

SEARCH WCBR-BEN-DATA

AT END

SET QUALIFIED-NO OF W-DEPEND(WDEPEND-IDX) TO TRUE

WHEN WCBR-BEN-IDX > WCBR-BEN-COUNT OF W-CBR-EVT

SET QUALIFIED-NO OF W-DEPEND(WDEPEND-IDX) TO TRUE

WHEN COBRA-EVENT-CLASS OF W-CNTL

= COBRA-EVENT-CLASS OF WCBR-BEN-DATA

OF W-CBR-EVT(WCBR-BEN-IDX)

AND COVERED-PERSON-TYP OF W-DEPEND(WDEPEFF-IDX)

= COVERED-PERSON-TYP OF WCBR-BEN-DATA

OF W-CBR-EVT(WCBR-BEN-IDX)

SET QUALIFIED-YES OF W-DEPEND(WDEPEND-IDX) TO TRUE

END-SEARCH

IF COBRA-PHASE-QUALIFY OF W-CNTL

AND (COBRA-EVENT-DEPENDENT

OR COBRA-EVENT-OVERAGE OF W-EVENT)

AND QUALIFIED-YES OF W-DEPEND(WDEPEND-IDX)

IF COBRA-EVENT-CLASS OF W-CNTL

NOT = COBRA-ACTION OF W-DEPEND(WDEPEND-IDX)

AND COBRA-EVENT-DT OF W-EVENT

NOT = COBRA-EVENT-DT

OF W-DEPEND(WDEPEND-IDX)

SET QUALIFIED-NO OF W-DEPEND(WDEPEND-IDX) TO TRUE
ELSE

IF COBRA-EVENT-DEPENDENT OF W-EVENT
AND MARRIED-DEP-ALLOWED OF W-DEPRULE

SET QUALIFIED-NO OF W-DEPEND(WDEPEND-IDX)
TO TRUE

END-IF

END-IF

END-IF

.

QUALIFY-DEPENDENT-EXIT.

/*****

* *

XQ000-DEPENDENT-AGE-CHECK SECTION.

XQ000.

* *

SET OVERAGE-S-NO OF W-SW TO TRUE

SET OVERAGE-NS-NO OF W-SW TO TRUE

IF BIRTHDATE OF W-DEPEND(WDEPEND-IDX) NOT = SPACE
MOVE AGE-AS-OF-DT OF W-CNTL TO END-DT OF DTWRK
MOVE BIRTHDATE OF W-DEPEND(WDEPEND-IDX)
TO BEGIN-DT OF DTWRK
SET OPTION-AGE OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE YEARS OF DTWRK
TO DEPENDENT-AGE OF W-DEPEND(WDEPEND-IDX)

IF STUDENT OF W-DEPEND(WDEPEFF-IDX)
PERFORM XQ100-STUDENT-AGE-CHECK
ELSE
PERFORM XQ200-NON-STUDENT-AGE-CHECK
END-IF

IF OVERAGE-S-YES OF W-SW OR OVERAGE-NS-YES OF W-SW

PERFORM XQ400-SET-OVERAGE-COVG-BEG
END-IF

ELSE
SET MSGID-CBR-NO-DEP-BIRTHDATE TO TRUE
MOVE COBRA-EVENT-ID OF W-EVENT
TO COBRA-EVENT-ID OF PYMSG
MOVE DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

TO DEPENDENT-BENEF OF PYMSG

MOVE EMPLID OF W-EVENT TO EMPLID OF PYMSG

PERFORM ZM000-MESSAGE

END-IF

.

DEPENDENT-AGE-CHECK-EXIT.

/*****

*

*

XQ100-STUDENT-AGE-CHECK SECTION.

XQ100.

*

*

IF DEPENDENT-AGE OF W-DEPEND(WDEPEND-IDX)

>= STUDENT-AGE-LIMIT OF W-DEPRULE

SET OVERAGE-S-YES OF W-SW TO TRUE

MOVE BIRTHDATE OF W-DEPEND(WDEPEND-IDX)

TO BEGIN-DT OF DTWRK

COMPUTE MONTHS OF DTWRK

= 12

* STUDENT-AGE-LIMIT OF W-DEPRULE

SET OPTION-ADD-MONTHS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE END-DT OF DTWRK

TO OVERAGE-DT OF W-SAVE

END-IF

.

STUDENT-AGE-CHECK-EXIT.

/*****

* *

XQ200-NON-STUDENT-AGE-CHECK SECTION.

XQ200.

* *

IF DEPENDENT-AGE OF W-DEPEND(WDEPEND-IDX)

>= DEP-AGE-LIMIT OF W-DEPRULE

SET OVERAGE-NS-YES OF W-SW TO TRUE

MOVE BIRTHDATE OF W-DEPEND(WDEPEND-IDX)

TO BEGIN-DT OF DTWRK

COMPUTE MONTHS OF DTWRK

= 12

* DEP-AGE-LIMIT OF W-DEPRULE

SET OPTION-ADD-MONTHS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

PERFORM XQ300-NON-STUDENT-TRANSITION

IF END-DT OF DTWRK

< STUDENT-STATUS-DT OF W-DEPEND(WDEPEFF-IDX)

AND STUDENT-STATUS-DT OF W-DEPEND(WDEPEFF-IDX)

NOT = SPACE

MOVE STUDENT-STATUS-DT OF W-DEPEND(WDEPEFF-IDX)

TO END-DT OF DTWRK

END-IF

MOVE END-DT OF DTWRK TO OVERAGE-DT OF W-SAVE

END-IF

.

NON-STUDENT-AGE-CHECK-EXIT.

/*****

*

*

XQ300-NON-STUDENT-TRANSITION SECTION.

XQ300.

*

*

- * Determine when an individual who is currently a non-student lost
- * their student status (if they ever held such status).
- * (Loss of student status may supercede an earlier overage event.)

MOVE EMPLID OF W-EVENT TO EMPLID OF S-STDNT

MOVE DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)

TO DEPENDENT-BENEF OF S-STDNT

MOVE AGE-AS-OF-DT OF W-CNTL TO EFFDT-1 OF S-STDNT

MOVE AGE-AS-OF-DT OF W-CNTL TO EFFDT-2 OF S-STDNT

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-STDNT OF W-CNTL

SQL-STMT OF S-STDNT

BIND-SETUP OF S-STDNT

BIND-DATA OF S-STDNT

SELECT-SETUP OF S-STDNT

SELECT-DATA OF S-STDNT

IF RTNCD-ERROR OF SQLRT

MOVE 'NON-STDNT-TRANS(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-STDNT

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-STDNT OF W-CNTL

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

DISPLAY 'Emplid/DepBenef: '

EMPLID OF S-STDNT '/'

DEPENDENT-BENEF OF S-STDNT

MOVE 'NON-STDNT-TRANS(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

ELSE

MOVE STUDENT-STATUS-DT OF S-STDNT

TO STUDENT-STATUS-DT

OF W-DEPEND(WDEPEFF-IDX)

END-IF

NON-STUDENT-TRANSITION-EXIT.

/*****

* *

XQ400-SET-OVERAGE-COVG-BEG SECTION.

XQ400.

* Based on Dependent rules, Coverage Begin may be set other than *

* Overage Event Date and allow Alternate Coverage with grace *

* period, e.g. 1st day of next month following Overage Event Date*

* *

IF OVERAGE-PD-BEG-ON-AFT-MM OF W-DEPRULE

MOVE OVERAGE-DT OF W-SAVE TO BEGIN-DT OF DTWRK

MOVE 1 TO MONTHS OF DTWRK

SET OPTION-ADD-MONTHS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING DTWRK

MOVE 1 TO MONTH-DAY OF END-DT OF DTWRK

MOVE END-DT OF DTWRK

TO OVERAGE-DT OF W-SAVE

END-IF

.

SET-OVERAGE-COVG-BEG-EXIT.

/*****

* *

XS000-LOAD-OVERAGE-ACTIVITY SECTION.

XS000.

* *

SET WOVGACT-IDX TO 1

SEARCH WOVGACT-DATA

AT END

DISPLAY 'Maximum # of COBRA Activity Exceeded'

MOVE 'LOAD-OVERAGE-ACTIVITY'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

WHEN WOVGACT-IDX > WOVGACT-COUNT OF W-OVGACT

ADD 1 TO WOVGACT-COUNT OF W-OVGACT
SET WOVGACT-IDX TO WOVGACT-COUNT
MOVE OVERAGE-DT OF W-DEPEND(WDEPEND-IDX)
TO COBRA-EVENT-DT OF W-OVGACT(WOVGACT-IDX)

WHEN OVERAGE-DT OF W-DEPEND(WDEPEND-IDX)
= COBRA-EVENT-DT OF W-OVGACT(WOVGACT-IDX)

CONTINUE

END-SEARCH

.

LOAD-OVERAGE-ACTIVITY-EXIT.

/*****

* *

XT000-UPDATE-DEP-BEN SECTION.

XT000.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF U-DEPEND
MOVE DEPENDENT-BENEF OF W-DEPEND(WDEPEND-IDX)
TO DEPENDENT-BENEF OF U-DEPEND
MOVE COBRA-ACTION OF W-DEPEND(WDEPEND-IDX)

TO COBRA-ACTION OF U-DEPEND
MOVE COBRA-EVENT-DT OF W-DEPEND(WDEPEND-IDX)
TO COBRA-EVENT-DT OF U-DEPEND
MOVE COBRA-EMPLID OF W-DEPEND(WDEPEND-IDX)
TO COBRA-EMPLID OF U-DEPEND

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT
SQLRT
SQL-CURSOR OF U-DEPEND OF W-CNTL
SQL-STMT OF U-DEPEND
BIND-SETUP OF U-DEPEND
BIND-DATA OF U-DEPEND

IF RTNCD-ERROR OF SQLRT

DISPLAY 'Emplid/DepBenef'
EMPLID OF U-DEPEND '/'
DEPENDENT-BENEF OF U-DEPEND

MOVE 'UPDATE-DEPENDENT-BENEF' TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

.

UPDATE-DEP-BEN-EXIT.

/*****

* *

XE000-OBTAIN-PARTIC1 SECTION.

XE000.

* *

PERFORM XE100-SELECT-PARTIC1

PERFORM XE200-FETCH-PARTIC1

PERFORM UNTIL RTNCD-END OF SQLRT

ADD 1 TO WPARTIC-COUNT OF W-EVENT

PERFORM XE300-SAVE-PARTIC1

PERFORM XE200-FETCH-PARTIC1

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

OBTAIN-PARTIC1-EXIT.

/*****

* *

XE100-SELECT-PARTIC1 SECTION.

XE100.

*

*

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PAR1 OF W-CNTL

SQL-STMT OF S-PAR1

BIND-SETUP OF S-PAR1

BIND-DATA OF S-PAR1

SELECT-SETUP OF S-PAR1

SELECT-DATA OF S-PAR1

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-PARTIC1' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-PARTIC1-EXIT.

/*****

*

*

XE200-FETCH-PARTIC1 SECTION.

XE200.

*

*

INITIALIZE SELECT-DATA OF S-PAR1

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-PAR1 OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-PARTIC1' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-PARTIC1-EXIT.

/*****

*

*

XE300-SAVE-PARTIC1 SECTION.

XE300.

*

*

SET WPARTIC-IDX TO WPARTIC-COUNT OF W-EVENT
INITIALIZE WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
SET INSERT-NO OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO TRUE
SET UPDATE-NO OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO TRUE
MOVE DEPENDENT-BENEF OF SELECT-DATA OF S-PAR1
TO DEPENDENT-BENEF OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
MOVE CBR-QUALIFY-STATUS OF S-PAR1
TO CBR-QUALIFY-STATUS OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
MOVE CBR-PROCESS-STATUS OF S-PAR1
TO CBR-PROCESS-STATUS OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
MOVE CBR-INIT-EVENT-STS OF S-PAR1
TO CBR-INIT-EVENT-STS OF W-EVENT(WPARTIC-IDX)
MOVE CBR-SCND-EVENT-STS OF S-PAR1
TO CBR-SCND-EVENT-STS OF W-EVENT(WPARTIC-IDX)
MOVE CBR-INIT-NOTIFY-DT OF S-PAR1
TO CBR-INIT-NOTIFY-DT OF W-EVENT(WPARTIC-IDX)
MOVE CBR-SCND-NOTIFY-DT OF S-PAR1
TO CBR-SCND-NOTIFY-DT OF W-EVENT(WPARTIC-IDX)
MOVE COBRA-ELECT OF S-PAR1
TO COBRA-ELECT OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-ELECT-DT OF S-PAR1
TO COBRA-ELECT-DT OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-WAIVE-DT OF S-PAR1
TO COBRA-WAIVE-DT OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-REVOKE-DT OF S-PAR1
TO COBRA-REVOKE-DT OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-EMPLID OF S-PAR1
TO COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

MOVE BENEFIT-PROGRAM OF S-PAR1
TO BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)

MOVE BAS-ASSIGN-STATUS OF S-PAR1
TO BAS-ASSIGN-STATUS OF W-EVENT(WPARTIC-IDX)

MOVE CBR-TERM-NOTIFY-DT OF S-PAR1
TO CBR-TERM-NOTIFY-DT OF W-EVENT(WPARTIC-IDX)

MOVE CBR-PAR-REPRCS-IND OF S-PAR1
TO CBR-PAR-REPRCS-IND OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

.

SAVE-PARTIC1-EXIT.

/*****

*

*

XU000-OBTAIN-PARTIC2 SECTION.

XU000.

*

*

PERFORM XU100-SELECT-PARTIC2

PERFORM XU200-FETCH-PARTIC2

PERFORM UNTIL RTNCD-END OF SQLRT

ADD 1 TO WPARTIC-COUNT OF W-EVENT

PERFORM XU300-SAVE-PARTIC2

PERFORM XU200-FETCH-PARTIC2

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

OBTAIN-PARTIC2-EXIT.

/*****

*

*

XU100-SELECT-PARTIC2 SECTION.

XU100.

*

*

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PAR2 OF W-CNTL

SQL-STMT OF S-PAR2

BIND-SETUP OF S-PAR2

BIND-DATA OF S-PAR2

SELECT-SETUP OF S-PAR2

SELECT-DATA OF S-PAR2

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-PARTIC2' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-PARTIC2-EXIT.

/*****

* *

XU200-FETCH-PARTIC2 SECTION.

XU200.

* *

INITIALIZE SELECT-DATA OF S-PAR2

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-PAR2 OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-PARTI2C' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-PARTIC2-EXIT.

/*****

* *

XU300-SAVE-PARTIC2 SECTION.

XU300.

* *

SET WPARTIC-IDX TO WPARTIC-COUNT OF W-EVENT

INITIALIZE WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

SET INSERT-NO OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO TRUE

SET UPDATE-NO OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO TRUE

MOVE DEPENDENT-BENEF OF SELECT-DATA OF S-PAR2
TO DEPENDENT-BENEF OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

MOVE CBR-QUALIFY-STATUS OF S-PAR2
TO CBR-QUALIFY-STATUS OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

MOVE CBR-PROCESS-STATUS OF S-PAR2
TO CBR-PROCESS-STATUS OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

MOVE CBR-INIT-EVENT-STS OF S-PAR2
TO CBR-INIT-EVENT-STS OF W-EVENT(WPARTIC-IDX)

MOVE CBR-SCND-EVENT-STS OF S-PAR2
TO CBR-SCND-EVENT-STS OF W-EVENT(WPARTIC-IDX)

MOVE CBR-INIT-NOTIFY-DT OF S-PAR2
TO CBR-INIT-NOTIFY-DT OF W-EVENT(WPARTIC-IDX)

MOVE CBR-SCND-NOTIFY-DT OF S-PAR2
TO CBR-SCND-NOTIFY-DT OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-ELECT OF S-PAR2
TO COBRA-ELECT OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-ELECT-DT OF S-PAR2
TO COBRA-ELECT-DT OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-WAIVE-DT OF S-PAR2

TO COBRA-WAIVE-DT OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-REVOKE-DT OF S-PAR2

TO COBRA-REVOKE-DT OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

MOVE COBRA-EMPLID OF S-PAR2

TO COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

MOVE BENEFIT-PROGRAM OF S-PAR2

TO BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)

MOVE BAS-ASSIGN-STATUS OF S-PAR2

TO BAS-ASSIGN-STATUS OF W-EVENT(WPARTIC-IDX)

MOVE CBR-TERM-NOTIFY-DT OF S-PAR2

TO CBR-TERM-NOTIFY-DT OF W-EVENT(WPARTIC-IDX)

MOVE CBR-PAR-REPRCS-IND OF S-PAR2

TO CBR-PAR-REPRCS-IND OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

.

SAVE-PARTIC2-EXIT.

/*****

*

*

XB000-OBTAIN-PARTIC-PLN2 SECTION.

XB000.

*

*

PERFORM XB100-SELECT-PARTIC-PLN2

PERFORM XB200-FETCH-PARTIC-PLN2

PERFORM UNTIL RTNCD-END OF SQLRT

ADD 1 TO WPLAN-COUNT OF W-EVENT

PERFORM XB300-SAVE-PARTIC-PLN2

PERFORM XB200-FETCH-PARTIC-PLN2

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

OBTAIN-PARTIC-PLN2-EXIT.

/*****

*

*

XB100-SELECT-PARTIC-PLN2 SECTION.

XB100.

*

*

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PARPLN2 OF W-CNTL

SQL-STMT OF S-PARPLN2

BIND-SETUP OF S-PARPLN2

BIND-DATA OF S-PARPLN2

SELECT-SETUP OF S-PARPLN2

SELECT-DATA OF S-PARPLN2

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-PARTIC-PLN2' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-PARTIC-PLN2-EXIT.

/*****

* *

XB200-FETCH-PARTIC-PLN2 SECTION.

XB200.

* *

INITIALIZE SELECT-DATA OF S-PARPLN2

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-PARPLN2 OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-PARTIC-PLN2' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-PARTIC-PLN2-EXIT.

/*****

* *

XB300-SAVE-PARTIC-PLN2 SECTION.

XB300.

* *

SET WPLAN-IDX TO WPLAN-COUNT OF W-EVENT

INITIALIZE WPLAN-DATA OF W-EVENT(WPLAN-IDX)

SET INSERT-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX) TO TRUE
SET UPDATE-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX) TO TRUE
SET CLEAR-SCND-EVT-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO TRUE

MOVE DEPENDENT-BENEF OF SELECT-DATA OF S-PARPLN2
TO DEPENDENT-BENEF OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

MOVE PLAN-TYPE OF SELECT-DATA OF S-PARPLN2
TO PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-TYPE OF S-PARPLN2
TO COBRA-EVENT-TYPE OF W-EVENT(WPLAN-IDX)

MOVE CBR-INIT-EVENT-ID OF S-PARPLN2
TO CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

MOVE INIT-BENEFIT-PGM OF S-PARPLN2
TO INIT-BENEFIT-PGM OF W-EVENT(WPLAN-IDX)

MOVE CBR-SCND-EVENT-ID OF S-PARPLN2
TO CBR-SCND-EVENT-ID OF W-EVENT(WPLAN-IDX)

MOVE CBR-PROCESS-STATUS OF S-PARPLN2
TO CBR-PROCESS-STATUS OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

MOVE CBR-ENROLL-STATUS OF S-PARPLN2
TO CBR-ENROLL-STATUS OF W-EVENT(WPLAN-IDX)

MOVE CBR-PLN-REPRCS-IND OF S-PARPLN2
TO CBR-PLN-REPRCS-IND
OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COVERAGE-BEGIN-DT OF S-PARPLN2

TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)
MOVE DISABL-CVG-BEG-DT OF S-PARPLN2
TO DISABL-CVG-BEG-DT OF W-EVENT(WPLAN-IDX)
MOVE COVERAGE-END-DT OF S-PARPLN2
TO COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)
MOVE COBRA-ELECT-END-DT OF S-PARPLN2
TO COBRA-ELECT-END-DT OF W-EVENT(WPLAN-IDX)
MOVE COBRA-ELECT OF S-PARPLN2
TO COBRA-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
MOVE COBRA-ELECT-DT OF S-PARPLN2
TO COBRA-ELECT-DT OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE COBRA-WAIVE-DT OF S-PARPLN2
TO COBRA-WAIVE-DT OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE COBRA-REVOKE-DT OF S-PARPLN2
TO COBRA-REVOKE-DT OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE COBRA-TERM-DT OF S-PARPLN2
TO COBRA-TERM-DT OF W-EVENT(WPLAN-IDX)
MOVE COBRA-TERM-REASON OF S-PARPLN2
TO COBRA-TERM-REASON OF W-EVENT(WPLAN-IDX)
MOVE OPTION-CD OF S-PARPLN2
TO OPTION-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
MOVE BENEFIT-PLAN OF S-PARPLN2
TO BENEFIT-PLAN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COVRG-CD OF S-PARPLN2

TO COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE EMPL-CONTRBUTN-AMT OF S-PARPLN2

TO EMPL-CONTRBUTN-AMT OF W-EVENT(WPLAN-IDX)

MOVE ANN-EX-CREDIT-FSA OF S-PARPLN2

TO ANN-EX-CREDIT-FSA OF W-EVENT(WPLAN-IDX)

MOVE ANNUAL-PLEDGE OF S-PARPLN2

TO ANNUAL-PLEDGE OF W-EVENT(WPLAN-IDX)

MOVE FSA-SUB-AMT-YTD OF S-PARPLN2

TO FSA-SUB-AMT-YTD OF W-EVENT(WPLAN-IDX)

MOVE FSA-APR-AMT-YTD OF S-PARPLN2

TO FSA-APR-AMT-YTD OF W-EVENT(WPLAN-IDX)

MOVE FSA-PD-AMT-YTD OF S-PARPLN2

TO FSA-PD-AMT-YTD OF W-EVENT(WPLAN-IDX)

MOVE HLTH-PROVIDER-ID OF S-PARPLN2

TO HLTH-PROVIDER-ID OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE PREVIOUSLY-SEEN OF S-PARPLN2

TO PREVIOUSLY-SEEN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE OTH-INSURANCE-IND OF S-PARPLN2

TO OTH-INSURANCE-IND OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE OTH-INSURANCE-NAME OF S-PARPLN2

TO OTH-INSURANCE-NAME OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COBRA-ERROR OF S-PARPLN2

TO COBRA-ERROR OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

.

SAVE-PARTIC-PLN2-EXIT.

/*****

* *

XC200-FETCH-PARTIC-PLN3 SECTION.

XC200.

* *

INITIALIZE SELECT-DATA OF S-PARPLN3

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-PARPLN3 OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-PARTIC-PLN3' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

FETCH-PARTIC-PLN3-EXIT.

/*****

* *

XC300-SAVE-PARTIC-PLN3 SECTION.

XC300.

* *

SET WPLAN-IDX TO WPLAN-COUNT OF W-EVENT

INITIALIZE WPLAN-DATA OF W-EVENT(WPLAN-IDX)

SET INSERT-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX) TO TRUE

SET UPDATE-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX) TO TRUE

SET CLEAR-SCND-EVT-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO TRUE

MOVE DEPENDENT-BENEF OF SELECT-DATA OF S-PARPLN3

TO DEPENDENT-BENEF OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE PLAN-TYPE OF SELECT-DATA OF S-PARPLN3

TO PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-TYPE OF S-PARPLN3

TO COBRA-EVENT-TYPE OF W-EVENT(WPLAN-IDX)

MOVE CBR-INIT-EVENT-ID OF S-PARPLN3

TO CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

MOVE INIT-BENEFIT-PGM OF S-PARPLN3
TO INIT-BENEFIT-PGM OF W-EVENT(WPLAN-IDX)

MOVE CBR-SCND-EVENT-ID OF S-PARPLN3
TO CBR-SCND-EVENT-ID OF W-EVENT(WPLAN-IDX)

MOVE CBR-PROCESS-STATUS OF S-PARPLN3
TO CBR-PROCESS-STATUS OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

MOVE CBR-ENROLL-STATUS OF S-PARPLN3
TO CBR-ENROLL-STATUS OF W-EVENT(WPLAN-IDX)

MOVE CBR-PLN-REPRCS-IND OF S-PARPLN3
TO CBR-PLN-REPRCS-IND
OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COVERAGE-BEGIN-DT OF S-PARPLN3
TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

MOVE DISABL-CVG-BEG-DT OF S-PARPLN3
TO DISABL-CVG-BEG-DT OF W-EVENT(WPLAN-IDX)

MOVE COVERAGE-END-DT OF S-PARPLN3
TO COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

MOVE COBRA-ELECT-END-DT OF S-PARPLN3
TO COBRA-ELECT-END-DT OF W-EVENT(WPLAN-IDX)

MOVE COBRA-ELECT OF S-PARPLN3
TO COBRA-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COBRA-ELECT-DT OF S-PARPLN3
TO COBRA-ELECT-DT OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

MOVE COBRA-WAIVE-DT OF S-PARPLN3

TO COBRA-WAIVE-DT OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

MOVE COBRA-REVOKE-DT OF S-PARPLN3
TO COBRA-REVOKE-DT OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

MOVE COBRA-TERM-DT OF S-PARPLN3
TO COBRA-TERM-DT OF W-EVENT(WPLAN-IDX)

MOVE COBRA-TERM-REASON OF S-PARPLN3
TO COBRA-TERM-REASON OF W-EVENT(WPLAN-IDX)

MOVE OPTION-CD OF S-PARPLN3
TO OPTION-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE BENEFIT-PLAN OF S-PARPLN3
TO BENEFIT-PLAN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COVRG-CD OF S-PARPLN3
TO COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE EMPL-CONTRIBUTN-AMT OF S-PARPLN3
TO EMPL-CONTRIBUTN-AMT OF W-EVENT(WPLAN-IDX)

MOVE ANN-EX-CREDIT-FSA OF S-PARPLN3
TO ANN-EX-CREDIT-FSA OF W-EVENT(WPLAN-IDX)

MOVE ANNUAL-PLEDGE OF S-PARPLN3
TO ANNUAL-PLEDGE OF W-EVENT(WPLAN-IDX)

MOVE FSA-SUB-AMT-YTD OF S-PARPLN3
TO FSA-SUB-AMT-YTD OF W-EVENT(WPLAN-IDX)

MOVE FSA-APR-AMT-YTD OF S-PARPLN3
TO FSA-APR-AMT-YTD OF W-EVENT(WPLAN-IDX)

MOVE FSA-PD-AMT-YTD OF S-PARPLN3

TO FSA-PD-AMT-YTD OF W-EVENT(WPLAN-IDX)
MOVE HLTH-PROVIDER-ID OF S-PARPLN3
TO HLTH-PROVIDER-ID OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE PREVIOUSLY-SEEN OF S-PARPLN3
TO PREVIOUSLY-SEEN OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE OTH-INSURANCE-IND OF S-PARPLN3
TO OTH-INSURANCE-IND OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE OTH-INSURANCE-NAME OF S-PARPLN3
TO OTH-INSURANCE-NAME OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)
MOVE COBRA-ERROR OF S-PARPLN3
TO COBRA-ERROR OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

.

SAVE-PARTIC-PLN3-EXIT.

/*****

* *

XV000-OBTAIN-PARTIC-PLN1 SECTION.

XV000.

* *

PERFORM XV100-SELECT-PARTIC-PLN1

PERFORM XV200-FETCH-PARTIC-PLN1

PERFORM UNTIL RTNCD-END OF SQLRT

ADD 1 TO WPLAN-COUNT OF W-EVENT

PERFORM XV300-SAVE-PARTIC-PLN1

PERFORM XV200-FETCH-PARTIC-PLN1

END-PERFORM

SET RTNCD-OK OF SQLRT TO TRUE

.

OBTAIN-PARTIC-PLAN-EXIT.

/*****

* *

XV100-SELECT-PARTIC-PLN1 SECTION.

XV100.

* *

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR OF S-PARPLN1 OF W-CNTL

SQL-STMT OF S-PARPLN1

BIND-SETUP OF S-PARPLN1

BIND-DATA OF S-PARPLN1

SELECT-SETUP OF S-PARPLN1

SELECT-DATA OF S-PARPLN1

IF RTNCD-ERROR OF SQLRT

MOVE 'SELECT-PARTIC-PLN1' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SELECT-PARTIC-PLAN-EXIT.

/*****

*

*

XV200-FETCH-PARTIC-PLN1 SECTION.

XV200.

*

*

*****/

INITIALIZE SELECT-DATA OF S-PARPLN1

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR OF S-PARPLN1 OF W-CNTL

IF RTNCD-ERROR OF SQLRT

AND NOT RTNCD-END OF SQLRT

MOVE 'FETCH-PARTIC-PLN1' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

FETCH-PARTIC-PLN1-EXIT.

/*****

* * *

XV300-SAVE-PARTIC-PLN1 SECTION.

XV300.

* * *

SET WPLAN-IDX TO WPLAN-COUNT OF W-EVENT

INITIALIZE WPLAN-DATA OF W-EVENT(WPLAN-IDX)

SET INSERT-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX) TO TRUE

SET UPDATE-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX) TO TRUE

SET CLEAR-SCND-EVT-NO OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO TRUE

MOVE DEPENDENT-BENEF OF SELECT-DATA OF S-PARPLN1

TO DEPENDENT-BENEF OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE PLAN-TYPE OF SELECT-DATA OF S-PARPLN1

TO PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-TYPE OF S-PARPLN1

TO COBRA-EVENT-TYPE OF W-EVENT(WPLAN-IDX)

MOVE CBR-INIT-EVENT-ID OF S-PARPLN1

TO CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

MOVE INIT-BENEFIT-PGM OF S-PARPLN1

TO INIT-BENEFIT-PGM OF W-EVENT(WPLAN-IDX)

MOVE CBR-SCND-EVENT-ID OF S-PARPLN1

TO CBR-SCND-EVENT-ID OF W-EVENT(WPLAN-IDX)

MOVE CBR-PROCESS-STATUS OF S-PARPLN1

TO CBR-PROCESS-STATUS OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE CBR-ENROLL-STATUS OF S-PARPLN1

TO CBR-ENROLL-STATUS OF W-EVENT(WPLAN-IDX)

MOVE CBR-PLN-REPRCS-IND OF S-PARPLN1

TO CBR-PLN-REPRCS-IND

OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COVERAGE-BEGIN-DT OF S-PARPLN1

TO COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

MOVE DISABL-CVG-BEG-DT OF S-PARPLN1

TO DISABL-CVG-BEG-DT OF W-EVENT(WPLAN-IDX)

MOVE COVERAGE-END-DT OF S-PARPLN1

TO COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

MOVE COBRA-ELECT-END-DT OF S-PARPLN1

TO COBRA-ELECT-END-DT OF W-EVENT(WPLAN-IDX)

MOVE COBRA-ELECT OF S-PARPLN1

TO COBRA-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COBRA-ELECT-DT OF S-PARPLN1

TO COBRA-ELECT-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COBRA-WAIVE-DT OF S-PARPLN1

TO COBRA-WAIVE-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COBRA-REVOKE-DT OF S-PARPLN1

TO COBRA-REVOKE-DT OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COBRA-TERM-DT OF S-PARPLN1

TO COBRA-TERM-DT OF W-EVENT(WPLAN-IDX)

MOVE COBRA-TERM-REASON OF S-PARPLN1

TO COBRA-TERM-REASON OF W-EVENT(WPLAN-IDX)

MOVE OPTION-CD OF S-PARPLN1

TO OPTION-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE BENEFIT-PLAN OF S-PARPLN1

TO BENEFIT-PLAN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE COVRG-CD OF S-PARPLN1

TO COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

MOVE EMPL-CONTRIBUTN-AMT OF S-PARPLN1

TO EMPL-CONTRIBUTN-AMT OF W-EVENT(WPLAN-IDX)

MOVE ANN-EX-CREDIT-FSA OF S-PARPLN1

TO ANN-EX-CREDIT-FSA OF W-EVENT(WPLAN-IDX)

MOVE ANNUAL-PLEDGE OF S-PARPLN1

TO ANNUAL-PLEDGE OF W-EVENT(WPLAN-IDX)

MOVE FSA-SUB-AMT-YTD OF S-PARPLN1

TO FSA-SUB-AMT-YTD OF W-EVENT(WPLAN-IDX)

MOVE FSA-APR-AMT-YTD OF S-PARPLN1

TO FSA-APR-AMT-YTD OF W-EVENT(WPLAN-IDX)

MOVE FSA-PD-AMT-YTD OF S-PARPLN1

TO FSA-PD-AMT-YTD OF W-EVENT(WPLAN-IDX)

MOVE HLTH-PROVIDER-ID OF S-PARPLN1

TO HLTH-PROVIDER-ID OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE PREVIOUSLY-SEEN OF S-PARPLN1

TO PREVIOUSLY-SEEN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE OTH-INSURANCE-IND OF S-PARPLN1

TO OTH-INSURANCE-IND OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE OTH-INSURANCE-NAME OF S-PARPLN1

TO OTH-INSURANCE-NAME OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

MOVE COBRA-ERROR OF S-PARPLN1

TO COBRA-ERROR OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

.

SAVE-PARTIC-PLN1-EXIT.

/*****

*

*

XW000-STORE-COBRA-DATA SECTION.

XW000.

*

*

PERFORM XW100-STORE-EVENT

PERFORM XW200-STORE-PARTIC

PERFORM XW300-STORE-PARTIC-PLAN

PERFORM XW400-STORE-PARTIC-OPTN

PERFORM XW500-STORE-PARTIC-DPND

IF RESET-EVENT-STATUS-YES OF W-SW

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF BIND-DATA OF S-PAR1

PERFORM XW700-RESET-EVENT-STATUS

SET RESET-EVENT-STATUS-NO OF W-SW TO TRUE

END-IF

IF REPROCESS-PARTIC-YES OF W-SW

AND CBR-REPROCESS-VOID OF W-SW

```
MOVE COBRA-EVENT-ID OF S-REPRPAR
    TO COBRA-EVENT-ID OF BIND-DATA OF S-PAR1
PERFORM XW700-RESET-EVENT-STATUS
SET REPROCESS-PARTIC-NO OF W-SW TO TRUE
SET CBR-REPROCESS-NONE OF W-SW TO TRUE
END-IF

IF REPROCESS-PLAN-YES OF W-SW
    AND (CBR-REPROCESS-VOID OF W-SW
        OR CBR-REPROCESS-ELECT OF W-SW)
MOVE COBRA-EVENT-ID OF S-REPRPLN
    TO COBRA-EVENT-ID OF BIND-DATA OF S-PARPLN2
MOVE DEPENDENT-BENEF OF S-REPRPLN
    TO DEPENDENT-BENEF OF BIND-DATA OF S-PARPLN2
PERFORM XW600-RESET-PARTIC-STATUS

IF CBR-REPROCESS-VOID OF W-SW

    MOVE COBRA-EVENT-ID OF S-REPRPLN
        TO COBRA-EVENT-ID OF BIND-DATA
            OF S-PAR1

    PERFORM XW700-RESET-EVENT-STATUS
    SET REPROCESS-PLAN-NO OF W-SW TO TRUE
    SET CBR-REPROCESS-NONE OF W-SW TO TRUE
END-IF
END-IF
```

.

STORE-COBRA-DATA-EXIT.

/*****

* *

XW100-STORE-EVENT SECTION.

XW100.

* *

IF INSERT-YES OF WEVENT-DATA OF W-EVENT

PERFORM XW110-INSERT-EVENT

ELSE

IF UPDATE-YES OF WEVENT-DATA OF W-EVENT

PERFORM XW120-UPDATE-EVENT

END-IF

END-IF

.

STORE-EVENT-EXIT.

/*****

* *

XW110-INSERT-EVENT SECTION.

XW110.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF I-EVT

MOVE BENEFIT-RCD-NO OF W-EVENT TO BENEFIT-RCD-NO OF I-EVT

MOVE COBRA-EVENT-ID OF W-EVENT TO COBRA-EVENT-ID OF I-EVT

MOVE COBRA-EVENT-DT OF W-EVENT

TO COBRA-EVENT-DT OF I-EVT

MOVE COBRA-EVENT-CLASS OF W-EVENT

TO COBRA-EVENT-CLASS OF I-EVT

MOVE SCHED-ID OF W-EVENT TO SCHED-ID OF I-EVT

MOVE EVENT-ID OF W-EVENT TO EVENT-ID OF I-EVT

MOVE EMPL-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF I-EVT

MOVE CBR-ACTION-SOURCE OF W-EVENT

TO CBR-ACTION-SOURCE OF I-EVT

MOVE CBR-COV-OVERRIDE OF W-EVENT

TO CBR-COV-OVERRIDE OF I-EVT

MOVE CBR-ALT-COV-TRM-DT OF W-EVENT

TO CBR-ALT-COV-TRM-DT OF I-EVT

MOVE CBR-COV-AS-OF-DT OF W-EVENT

TO CBR-COV-AS-OF-DT OF I-EVT

MOVE CBR-QUALIFY-STATUS OF WEVENT-DATA OF W-EVENT
TO CBR-QUALIFY-STATUS OF I-EVT

MOVE CBR-PROCESS-STATUS OF WEVENT-DATA OF W-EVENT
TO CBR-PROCESS-STATUS OF I-EVT

MOVE CBR-EVT-REPRCS-IND OF W-EVENT
TO CBR-EVT-REPRCS-IND OF I-EVT

MOVE CBR-EVENT-CONFLICT OF W-EVENT
TO CBR-EVENT-CONFLICT OF I-EVT

MOVE BAS-DATA-CHG OF W-EVENT TO BAS-DATA-CHG OF I-EVT

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF I-EVT OF W-CNTL

SQL-STMT OF I-EVT

BIND-SETUP OF I-EVT

BIND-DATA OF I-EVT

IF RTNCD-ERROR OF SQLRT

MOVE 'INSERT-EVENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

INSERT-EVENT-EXIT.

/*****

* *

XW120-UPDATE-EVENT SECTION.

XW120.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF U-EVT

MOVE BENEFIT-RCD-NO OF W-EVENT TO BENEFIT-RCD-NO OF U-EVT

MOVE COBRA-EVENT-ID OF W-EVENT TO COBRA-EVENT-ID OF U-EVT

MOVE CBR-PROCESS-STATUS OF WEVENT-DATA OF W-EVENT

TO CBR-PROCESS-STATUS OF U-EVT

MOVE CBR-QUALIFY-STATUS OF WEVENT-DATA OF W-EVENT

TO CBR-QUALIFY-STATUS OF U-EVT

MOVE CBR-EVT-REPRCS-IND OF W-EVENT

TO CBR-EVT-REPRCS-IND OF U-EVT

MOVE CBR-EVENT-CONFLICT OF W-EVENT

TO CBR-EVENT-CONFLICT OF U-EVT

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF U-EVT OF W-CNTL

SQL-STMT OF U-EVT

BIND-SETUP OF U-EVT

BIND-DATA OF U-EVT

IF RTNCD=ERROR OF SQLRT

MOVE 'UPDATE-EVENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

UPDATE-EVENT-EXIT.

/*****

* *

XW130-UPDATE-RESET-EVENT SECTION.

XW130.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF U-RESEVT

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF U-RESEVT

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF U-RESEVT

MOVE CBR-PROCESS-STATUS OF WEVENT-DATA OF W-EVENT

TO CBR-PROCESS-STATUS OF U-RESEVT

MOVE CBR-QUALIFY-STATUS OF WEVENT-DATA OF W-EVENT

TO CBR-QUALIFY-STATUS OF U-RESEVT

MOVE CBR-EVT-REPRCS-IND OF W-EVENT
TO CBR-EVT-REPRCS-IND OF U-RESEVT

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT
SQLRT
SQL-CURSOR OF U-RESEVT OF W-CNTL
SQL-STMT OF U-RESEVT
BIND-SETUP OF U-RESEVT
BIND-DATA OF U-RESEVT

IF RTNCD-ERROR OF SQLRT

MOVE 'UPDATE-RESET-EVENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

UPDATE-EVENT-EXIT.

/*****

* *

XW200-STORE-PARTIC SECTION.

XW200.

* *

PERFORM VARYING WPARTIC-IDX FROM 1 BY 1

UNTIL WPARTIC-IDX > WPARTIC-COUNT OF W-EVENT

IF INSERT-YES OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

PERFORM XW210-INSERT-PARTIC

ELSE

IF UPDATE-YES OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

PERFORM XW220-UPDATE-PARTIC

END-IF

END-IF

END-PERFORM

.

STORE-COBRA-DATA-EXIT.

/*****

* * *

XW210-INSERT-PARTIC SECTION.

XW210.

* * *

MOVE EMPLID OF W-EVENT TO EMPLID OF I-PAR

MOVE BENEFIT-RCD-NO OF W-EVENT TO BENEFIT-RCD-NO OF I-PAR

MOVE COBRA-EVENT-ID OF W-EVENT TO COBRA-EVENT-ID OF I-PAR
MOVE COBRA-EVENT-DT OF W-EVENT TO COBRA-EVENT-DT OF I-PAR
MOVE DEPENDENT-BENEF OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO DEPENDENT-BENEF OF I-PAR
MOVE CBR-QUALIFY-STATUS OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
TO CBR-QUALIFY-STATUS OF I-PAR
MOVE CBR-PROCESS-STATUS OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
TO CBR-PROCESS-STATUS OF I-PAR
MOVE CBR-INIT-EVENT-STS OF W-EVENT(WPARTIC-IDX)
TO CBR-INIT-EVENT-STS OF I-PAR
MOVE CBR-SCND-EVENT-STS OF W-EVENT(WPARTIC-IDX)
TO CBR-SCND-EVENT-STS OF I-PAR
MOVE COBRA-ELECT OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO COBRA-ELECT OF I-PAR
MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)
TO COBRA-EMPLID OF I-PAR
MOVE BENEFIT-PROGRAM OF W-EVENT(WPARTIC-IDX)
TO BENEFIT-PROGRAM OF I-PAR
MOVE BAS-ASSIGN-STATUS OF W-EVENT(WPARTIC-IDX)
TO BAS-ASSIGN-STATUS OF I-PAR
MOVE CBR-PAR-REPRCS-IND OF W-EVENT(WPARTIC-IDX)
TO CBR-PAR-REPRCS-IND OF I-PAR
MOVE SPACE TO CBR-INIT-NOTIFY-DT OF I-PAR
MOVE SPACE TO CBR-SCND-NOTIFY-DT OF I-PAR

MOVE SPACE TO COBRA-ELECT-DT OF I-PAR
MOVE SPACE TO COBRA-WAIVE-DT OF I-PAR
MOVE SPACE TO COBRA-REVOKE-DT OF I-PAR
MOVE SPACE TO CBR-TERM-NOTIFY-DT OF I-PAR

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF I-PAR OF W-CNTL

SQL-STMT OF I-PAR

BIND-SETUP OF I-PAR

BIND-DATA OF I-PAR

IF RTNCD-ERROR OF SQLRT

MOVE 'INSERT-PARTIC' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

INSERT-PARTIC-EXIT.

/*****

* * *

XW220-UPDATE-PARTIC SECTION.

XW220.

* * *

MOVE EMPLID OF W-EVENT TO EMPLID OF U-PAR

MOVE BENEFIT-RCD-NO OF W-EVENT TO BENEFIT-RCD-NO OF U-PAR

MOVE COBRA-EVENT-ID OF W-EVENT TO COBRA-EVENT-ID OF U-PAR

MOVE DEPENDENT-BENEF OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)
TO DEPENDENT-BENEF OF U-PAR

MOVE CBR-QUALIFY-STATUS OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
TO CBR-QUALIFY-STATUS OF U-PAR

MOVE CBR-PROCESS-STATUS OF WPARTIC-DATA
OF W-EVENT(WPARTIC-IDX)
TO CBR-PROCESS-STATUS OF U-PAR

MOVE CBR-PAR-REPRCS-IND OF W-EVENT(WPARTIC-IDX)
TO CBR-PAR-REPRCS-IND OF U-PAR

MOVE CBR-INIT-EVENT-STS OF W-EVENT(WPARTIC-IDX)
TO CBR-INIT-EVENT-STS OF U-PAR

MOVE CBR-SCND-EVENT-STS OF W-EVENT(WPARTIC-IDX)
TO CBR-SCND-EVENT-STS OF U-PAR

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)
TO COBRA-EMPLID OF U-PAR

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT
SQLRT
SQL-CURSOR OF U-PAR OF W-CNTL
SQL-STMT OF U-PAR

BIND-SETUP OF U-PAR

BIND-DATA OF U-PAR

IF RTNCD-ERROR OF SQLRT

MOVE 'UPDATE-PARTIC' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

UPDATE-PARTIC-EXIT.

/*****

*

*

XW300-STORE-PARTIC-PLAN SECTION.

XW300.

*

*

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

IF INSERT-YES OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM XW310-INSERT-PARTIC-PLAN

ELSE

IF UPDATE-YES OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM XW340-UPDATE-PARTIC-PLAN

END-IF

END-IF

IF SET-SCND-EVT-YES OF W-EVENT(WPLAN-IDX)

PERFORM XW320-SET-SCND-EVT-ID

END-IF

IF CLEAR-SCND-EVT-YES OF W-EVENT(WPLAN-IDX)

PERFORM XW330-CLEAR-SCND-EVT-ID

END-IF

END-PERFORM

.

STORE-PARTIC-PLAN-EXIT.

/*****

* * *

XW310-INSERT-PARTIC-PLAN SECTION.

XW310.

*

*

MOVE EMPLID OF W-EVENT TO EMPLID OF I-PARPLN

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF I-PARPLN

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF I-PARPLN

SET WPARTIC-IDX TO 1

SEARCH WPARTIC-DATA

WHEN DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

= DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

CONTINUE

END-SEARCH

MOVE BENEFIT-PROGRAM OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

TO BENEFIT-PROGRAM OF I-PARPLN

MOVE COBRA-EVENT-DT OF W-EVENT

TO COBRA-EVENT-DT OF I-PARPLN

MOVE DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO DEPENDENT-BENEF OF I-PARPLN

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF I-PARPLN

MOVE COBRA-EVENT-TYPE OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-TYPE OF I-PARPLN

MOVE CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

TO CBR-INIT-EVENT-ID OF I-PARPLN

MOVE CBR-SCND-EVENT-ID OF W-EVENT(WPLAN-IDX)

TO CBR-SCND-EVENT-ID OF I-PARPLN

MOVE INIT-BENEFIT-PGM OF W-EVENT(WPLAN-IDX)

TO INIT-BENEFIT-PGM OF I-PARPLN

MOVE CBR-PROCESS-STATUS OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO CBR-PROCESS-STATUS OF I-PARPLN

MOVE CBR-ENROLL-STATUS OF W-EVENT(WPLAN-IDX)

TO CBR-ENROLL-STATUS OF I-PARPLN

MOVE CBR-PLN-REPRCS-IND OF W-EVENT(WPLAN-IDX)

TO CBR-PLN-REPRCS-IND OF I-PARPLN

MOVE COVERAGE-BEGIN-DT OF W-EVENT(WPLAN-IDX)

TO COVERAGE-BEGIN-DT OF I-PARPLN

MOVE DISABL-CVG-BEG-DT OF W-EVENT(WPLAN-IDX)

TO DISABL-CVG-BEG-DT OF I-PARPLN

MOVE COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

TO COVERAGE-END-DT OF I-PARPLN

MOVE COBRA-ELECT-END-DT OF W-EVENT(WPLAN-IDX)

TO COBRA-ELECT-END-DT OF I-PARPLN

MOVE COBRA-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO COBRA-ELECT OF I-PARPLN

MOVE OPTION-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO OPTION-CD OF I-PARPLN

MOVE BENEFIT-PLAN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO BENEFIT-PLAN OF I-PARPLN

MOVE COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO COVRG-CD OF I-PARPLN

MOVE EMPL-CONTRIBUTN-AMT OF W-EVENT(WPLAN-IDX)

TO EMPL-CONTRIBUTN-AMT OF I-PARPLN

MOVE ANN-EX-CREDIT-FSA OF W-EVENT(WPLAN-IDX)

TO ANN-EX-CREDIT-FSA OF I-PARPLN

MOVE ANNUAL-PLEDGE OF W-EVENT(WPLAN-IDX)

TO ANNUAL-PLEDGE OF I-PARPLN

MOVE FSA-SUB-AMT-YTD OF W-EVENT(WPLAN-IDX)

TO FSA-SUB-AMT-YTD OF I-PARPLN

MOVE FSA-APR-AMT-YTD OF W-EVENT(WPLAN-IDX)

TO FSA-APR-AMT-YTD OF I-PARPLN

MOVE FSA-PD-AMT-YTD OF W-EVENT(WPLAN-IDX)

TO FSA-PD-AMT-YTD OF I-PARPLN

MOVE COBRA-ERROR OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO COBRA-ERROR OF I-PARPLN

MOVE SPACE TO COBRA-ELECT-DT OF I-PARPLN

MOVE SPACE TO COBRA-WAIVE-DT OF I-PARPLN

MOVE SPACE TO COBRA-REVOKE-DT OF I-PARPLN

MOVE SPACE TO COBRA-TERM-DT OF I-PARPLN

MOVE SPACE TO HLTH-PROVIDER-ID OF I-PARPLN

SET PREVIOUSLY-SEEN-NO OF I-PARPLN TO TRUE

SET OTH-INSURANCE-NO OF I-PARPLN TO TRUE
MOVE SPACE TO OTH-INSURANCE-NAME OF I-PARPLN
SET COBRA-NOT-TERMINATED OF I-PARPLN TO TRUE

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT
SQLRT
SQL-CURSOR OF I-PARPLN OF W-CNTL
SQL-STMT OF I-PARPLN
BIND-SETUP OF I-PARPLN
BIND-DATA OF I-PARPLN

IF RTNCD-ERROR OF SQLRT

MOVE 'INSERT-PARTIC-PLAN' TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

.

INSERT-PARTIC-PLAN-EXIT.

/*****

* *

XW320-SET-SCND-EVT-ID SECTION.

XW320.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF U-SETSCND

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF U-SETSCND

MOVE CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-ID OF U-SETSCND

MOVE DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO DEPENDENT-BENEF OF U-SETSCND

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF U-SETSCND

MOVE COBRA-EVENT-ID OF W-EVENT

TO CBR-SCND-EVENT-ID OF U-SETSCND

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF U-SETSCND

BIND-SETUP OF U-SETSCND

BIND-DATA OF U-SETSCND

IF RTNCD-ERROR OF SQLRT

MOVE 'SET-SCND-EVT-ID' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

SET-SCND-EVT-ID-EXIT.

/*****

*

*

XW330-CLEAR-SCND-EVT-ID SECTION.

XW330.

*

*

MOVE ZERO TO CBR-SCND-EVENT-ID OF U-CLRSCND

MOVE EMPLID OF W-EVENT TO EMPLID OF U-CLRSCND

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF U-CLRSCND

MOVE CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-ID OF U-CLRSCND

MOVE DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO DEPENDENT-BENEF OF U-CLRSCND

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF U-CLRSCND

MOVE COBRA-EVENT-ID OF W-EVENT

TO CBR-SCND-EVT-ID-2 OF U-CLRSCND

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF U-CLRSCND

BIND-SETUP OF U-CLRSCND

BIND-DATA OF U-CLRSCND

IF RTNCD-ERROR OF SQLRT

MOVE 'CLEAR-SCND-EVT-ID' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

CLEAR-SCND-EVT-ID-EXIT.

/*****

*

*

XW340-UPDATE-PARTIC-PLAN SECTION.

XW340.

*

*

MOVE EMPLID OF W-EVENT TO EMPLID OF U-PARPLN

MOVE EMPLID OF W-EVENT TO EMPLID OF D-VHTH

MOVE EMPLID OF W-EVENT TO EMPLID OF U-VDPBEN

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF U-PARPLN

MOVE EMPL-RCD-NO OF W-EVENT

TO EMPL-RCD-NO OF D-VHTH
MOVE COBRA-EVENT-ID OF W-EVENT
TO COBRA-EVENT-ID OF U-PARPLN
MOVE COBRA-EVENT-DT OF W-EVENT
TO EFFDT OF D-VHTH
MOVE DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO DEPENDENT-BENEF OF U-PARPLN
MOVE DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO DEPENDENT-BENEF OF U-VDPBEN
MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO PLAN-TYPE OF U-PARPLN
MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO PLAN-TYPE OF D-VHTH
MOVE CBR-PROCESS-STATUS OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO CBR-PROCESS-STATUS OF U-PARPLN
MOVE CBR-PLN-REPRCS-IND OF W-EVENT(WPLAN-IDX)
TO CBR-PLN-REPRCS-IND OF U-PARPLN
MOVE CBR-ENROLL-STATUS OF W-EVENT(WPLAN-IDX)
TO CBR-ENROLL-STATUS OF U-PARPLN
MOVE COBRA-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO COBRA-ELECT OF U-PARPLN
MOVE COBRA-ELECT-DT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO COBRA-ELECT-DT OF U-PARPLN
MOVE COBRA-TERM-REASON OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO COBRA-TERM-REASON OF U-PARPLN
MOVE COBRA-TERM-DT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO COBRA-TERM-DT OF U-PARPLN
MOVE OPTION-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO OPTION-CD OF U-PARPLN
MOVE BENEFIT-PLAN OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO BENEFIT-PLAN OF U-PARPLN
MOVE COVRG-CD OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO COVRG-CD OF U-PARPLN
MOVE COBRA-ERROR OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
TO COBRA-ERROR OF U-PARPLN

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF U-PARPLN OF W-CNTL

SQL-STMT OF U-PARPLN

BIND-SETUP OF U-PARPLN

BIND-DATA OF U-PARPLN

IF RTNCD-ERROR OF SQLRT

MOVE 'UPDATE-PARTIC-PLAN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

IF COBRA-EVENT-OVERAGE OF W-EVENT

AND CBR-PROCESS-VOID OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM XY940-UPDATE-VDEPEND-BENEF

PERFORM XY910-DELETE-COBRAVD-HTH

PERFORM XY920-DELETE-COBRAVD-HTH-DEP

PERFORM XY930-DELETE-COBRAVD-FSA

END-IF

.

UPDATE-PARTIC-PLAN-EXIT.

/*****

* *

XW400-STORE-PARTIC-OPTN SECTION.

XW400.

* *

PERFORM VARYING WOPTN-IDX FROM 1 BY 1

UNTIL WOPTN-IDX > WOPTN-COUNT OF W-EVENT

IF INSERT-YES OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

PERFORM XW410-INSERT-PARTIC-OPTN

END-IF

END-PERFORM

.

STORE-PARTIC-OPTN-EXIT.

/*****

* *

XW410-INSERT-PARTIC-OPTN SECTION.

XW410.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF I-PAROPTN

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF I-PAROPTN

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF I-PAROPTN

SET WPARTIC-IDX TO 1

SEARCH WPARTIC-DATA

WHEN DEPENDENT-BENEF OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)

= DEPENDENT-BENEF OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

CONTINUE

END-SEARCH

MOVE COBRA-EVENT-DT OF W-EVENT
TO COBRA-EVENT-DT OF I-PAROPTN

MOVE DEPENDENT-BENEF OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
TO DEPENDENT-BENEF OF I-PAROPTN

MOVE PLAN-TYPE OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
TO PLAN-TYPE OF I-PAROPTN

MOVE OPTION-ID OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
TO OPTION-ID OF I-PAROPTN

MOVE BENEFIT-PLAN OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
TO BENEFIT-PLAN OF I-PAROPTN

MOVE COVRG-CD OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
TO COVRG-CD OF I-PAROPTN

MOVE OPTION-CD OF WOPTN-DATA OF W-EVENT(WOPTN-IDX)
TO OPTION-CD OF I-PAROPTN

MOVE CBR-MM-COST OF W-EVENT(WOPTN-IDX)
TO CBR-MM-COST OF I-PAROPTN

MOVE CBR-DISABL-MM-COST OF W-EVENT(WOPTN-IDX)
TO CBR-DISABL-MM-COST OF I-PAROPTN

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT
SQLRT
SQL-CURSOR OF I-PAROPTN OF W-CNTL
SQL-STMT OF I-PAROPTN
BIND-SETUP OF I-PAROPTN
BIND-DATA OF I-PAROPTN

IF RTNCD-ERROR OF SQLRT

MOVE 'INSERT-CBR-PARTIC-OPTN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

INSERT-PARTIC-OPTN-EXIT.

/*****

* *

XW500-STORE-PARTIC-DPND SECTION.

XW500.

* *

PERFORM VARYING WDPND-IDX FROM 1 BY 1

UNTIL WDPND-IDX > WDPND-COUNT OF W-EVENT

IF UPDATE-YES OF WDPND-DATA OF W-EVENT(WDPND-IDX)

PERFORM XW510-UPDATE-PARTIC-DPND

END-IF

END-PERFORM

STORE-PARTIC-DPND-EXIT.

/*****

*

*

XW510-UPDATE-PARTIC-DPND SECTION.

XW510.

*

*

MOVE EMPLID OF W-EVENT TO EMPLID OF U-PARDPND

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF U-PARDPND

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF U-PARDPND

MOVE DEPENDENT-BENEF OF WDPND-DATA OF W-EVENT(WDPND-IDX)

TO DEPENDENT-BENEF OF U-PARDPND

MOVE PLAN-TYPE OF WDPND-DATA OF W-EVENT(WDPND-IDX)

TO PLAN-TYPE OF U-PARDPND

MOVE COBRA-ERROR OF WDPND-DATA OF W-EVENT(WDPND-IDX)

TO COBRA-ERROR OF U-PARDPND

MOVE COBRA-DEP-BENEF OF WDPND-DATA OF W-EVENT(WDPND-IDX)

TO COBRA-DEP-BENEF OF U-PARDPND

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF U-PARDPND OF W-CNTL

SQL-STMT OF U-PARDPND

BIND-SETUP OF U-PARDPND

BIND-DATA OF U-PARDPND

IF RTNCD-ERROR OF SQLRT

MOVE 'UPDATE-PARTIC-DPND' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

UPDATE-PARTIC-DPND-EXIT.

/*****

* *

XW600-RESET-PARTIC-STATUS SECTION.

XW600.

* *

MOVE ZERO TO WPARTIC-COUNT OF W-EVENT

MOVE ZERO TO WPLAN-COUNT OF W-EVENT

MOVE ZERO TO WOPTN-COUNT OF W-EVENT

MOVE ZERO TO WDPND-COUNT OF W-EVENT

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF S-PARPLN2

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF BIND-DATA OF S-PARPLN2

PERFORM XB000-OBTAIN-PARTIC-PLN2

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF S-PAR2

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF BIND-DATA OF S-PAR2

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF BIND-DATA OF S-PAR2

MOVE DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO DEPENDENT-BENEF OF BIND-DATA OF S-PAR2

PERFORM XU000-OBTAIN-PARTIC2

SET CBR-PROCESS-CLOSED OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX) TO TRUE

MOVE ZERO TO INIT-PLAN-COUNT OF W-COUNT

MOVE ZERO TO INIT-ERROR-COUNT OF W-COUNT

MOVE ZERO TO INIT-ENROLL-COUNT OF W-COUNT

MOVE ZERO TO INIT-NOT-ENROLL-COUNT OF W-COUNT

MOVE ZERO TO INIT-ENTERED-COUNT OF W-COUNT

MOVE ZERO TO INIT-ENROLL-COUNT OF W-COUNT

MOVE ZERO TO SCND-PLAN-COUNT OF W-COUNT

MOVE ZERO TO NOT-DETERM-PLAN-COUNT OF W-COUNT

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

IF CBR-PROCESS-OPEN OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

SET CBR-PROCESS-OPEN OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX) TO TRUE

END-IF

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

ADD 1 TO INIT-PLAN-COUNT OF W-COUNT

IF COBRA-ERROR-YES OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

ADD 1 TO INIT-ERROR-COUNT OF W-COUNT

END-IF

IF CBR-ENROLLED OF W-EVENT(WPLAN-IDX)
OR COBRA-ELECT-WAIVE OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

ADD 1 TO INIT-ENROLL-COUNT OF W-COUNT
ELSE

IF COBRA-ELECT-NONE OF WPLAN-DATA
OF W-EVENT(WPLAN-IDX)

ADD 1 TO INIT-NOT-ENROLL-COUNT OF W-COUNT
ELSE

ADD 1 TO INIT-ENTERED-COUNT OF W-COUNT
END-IF

END-IF

ELSE

IF COBRA-EVENT-SECONDARY OF W-EVENT(WPLAN-IDX)

ADD 1 TO SCND-PLAN-COUNT OF W-COUNT
ELSE

ADD 1 TO NOT-DETERM-PLAN-COUNT OF W-COUNT
END-IF

END-IF

END-PERFORM

IF INIT-PLAN-COUNT OF W-COUNT = ZERO

SET CBR-INIT-NOT-QUALIFIED OF W-EVENT(WPARTIC-IDX)

TO TRUE

ELSE

IF INIT-ERROR-COUNT OF W-COUNT > ZERO

SET CBR-INIT-ELECT-ERROR OF W-EVENT(WPARTIC-IDX)

TO TRUE

ELSE

IF INIT-ENROLL-COUNT OF W-COUNT

= INIT-PLAN-COUNT OF W-COUNT

SET CBR-INIT-ENROLL-COMPLETE

OF W-EVENT(WPARTIC-IDX)

TO TRUE

ELSE

IF INIT-ENROLL-COUNT OF W-COUNT > ZERO

IF INIT-ENTERED-COUNT OF W-COUNT > ZERO

SET CBR-INIT-ELECT-ENTERED

OF W-EVENT(WPARTIC-IDX) TO TRUE

ELSE

SET CBR-INIT-ELECT-ENROLLED

OF W-EVENT(WPARTIC-IDX) TO TRUE

END-IF

ELSE

IF CBR-INIT-QUALIFY-PENDING

OF W-EVENT(WPARTIC-IDX)

IF NOT-DETERM-PLAN-COUNT OF W-COUNT

= ZERO

SET CBR-INIT-QUALIFIED

OF W-EVENT(WPARTIC-IDX)

TO TRUE

END-IF

END-IF

END-IF

END-IF

END-IF

END-IF

IF SCND-PLAN-COUNT OF W-COUNT = ZERO

SET CBR-SCND-NOT-QUALIFIED OF W-EVENT(WPARTIC-IDX)

TO TRUE

END-IF

IF NOT-DETERM-PLAN-COUNT OF W-COUNT = ZERO

IF CBR-INIT-NOT-QUALIFIED OF W-EVENT(WPARTIC-IDX)

AND CBR-SCND-NOT-QUALIFIED

OF W-EVENT(WPARTIC-IDX)

SET CBR-NOT-QUALIFIED OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX) TO TRUE

ELSE

SET CBR-QUALIFIED OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX) TO TRUE

END-IF

END-IF

SET UPDATE-YES OF WPARTIC-DATA OF W-EVENT(WPARTIC-IDX)

TO TRUE

PERFORM XW220-UPDATE-PARTIC

.

RESET-PARTIC-STATUS-EXIT.

/*****

*

*

XW700-RESET-EVENT-STATUS SECTION.

XW700.

*

*

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF S-PAR1

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF BIND-DATA OF S-PAR1

PERFORM XE000-OBTAIN-PARTIC1

SET CBR-PROCESS-CLOSED OF WEVENT-DATA OF W-EVENT TO TRUE

SET CBR-QUALIFIED OF WEVENT-DATA OF W-EVENT TO TRUE

SET CBR-QUALIFY-ERROR-NO OF W-SW TO TRUE

SET CBR-QUALIFY-PENDING-NO OF W-SW TO TRUE

SET CBR-REPROCESS-NONE OF WEVENT-DATA OF W-EVENT TO TRUE

PERFORM VARYING WPARTIC-IDX FROM 1 BY 1

UNTIL WPARTIC-IDX > WPARTIC-COUNT OF W-EVENT

IF CBR-PROCESS-OPEN OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

SET CBR-PROCESS-OPEN OF WEVENT-DATA OF W-EVENT

TO TRUE

END-IF

IF CBR-QUALIFY-ERROR OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

SET CBR-QUALIFY-ERROR-YES OF W-SW TO TRUE

END-IF

IF CBR-QUALIFY-PENDING OF WPARTIC-DATA

OF W-EVENT(WPARTIC-IDX)

SET CBR-QUALIFY-PENDING-YES OF W-SW TO TRUE

END-IF

END-PERFORM

IF CBR-QUALIFY-ERROR-YES OF W-SW

SET CBR-QUALIFY-ERROR OF WEVENT-DATA OF W-EVENT TO TRUE

ELSE

IF CBR-QUALIFY-PENDING-YES OF W-SW

SET CBR-QUALIFY-PENDING OF WEVENT-DATA OF W-EVENT

TO TRUE

END-IF

END-IF

PERFORM XW710-UPDATE-RESET-EVENT

.

RESET-EVENT-STATUS-EXIT.

/*****

* *

XW710-UPDATE-RESET-EVENT SECTION.

XW710.

* *

MOVE EMPLID OF W-EVENT TO EMPLID OF U-RESEVT

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF U-RESEVT

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF U-RESEVT

MOVE CBR-PROCESS-STATUS OF WEVENT-DATA OF W-EVENT

TO CBR-PROCESS-STATUS OF U-RESEVT

MOVE CBR-QUALIFY-STATUS OF WEVENT-DATA OF W-EVENT

TO CBR-QUALIFY-STATUS OF U-RESEVT

MOVE CBR-EVT-REPRCS-IND OF W-EVENT

TO CBR-EVT-REPRCS-IND OF U-RESEVT

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF U-RESEVT OF W-CNTL

SQL-STMT OF U-RESEVT

BIND-SETUP OF U-RESEVT

BIND-DATA OF U-RESEVT

IF RTNCD-ERROR OF SQLRT

MOVE 'UPDATE-RESET-EVENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

UPDATE-RESET-EVENT-EXIT.

/*****

* *

XX100-INSERT-BP-PLN SECTION.

XX100.

* *

INITIALIZE BIND-DATA OF I-BP-PLN

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

```
        TO EMPLID OF I-BP-PLN

MOVE BENEFIT-RCD-NO OF W-EVENT TO EMPL-RCD-NO OF I-BP-PLN

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

    MOVE COBRA-EVENT-ID OF W-EVENT
        TO COBRA-EVENT-ID OF I-BP-PLN
ELSE
    MOVE CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)
        TO COBRA-EVENT-ID OF I-BP-PLN
END-IF

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)
    TO PLAN-TYPE OF I-BP-PLN

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT
    SQLRT
    SQL-CURSOR OF I-BP-PLN OF W-CNTL
    SQL-STMT OF I-BP-PLN
    BIND-SETUP OF I-BP-PLN
    BIND-DATA OF I-BP-PLN

IF RTNCD-ERROR OF SQLRT
    AND NOT RTNCD-DUPL OF SQLRT

    MOVE 'INSERT-BP-PLN' TO ERR-SECTION OF SQLRT
    PERFORM ZZ000-SQL-ERROR
END-IF
```


INSERT-BP-PLN-EXIT.

/*****

* *

XX200-INSERT-HEALTH-BENEFIT SECTION.

XX200.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF I-HTH OF W-CNTL

SQL-STMT OF I-HTH

BIND-SETUP OF I-HTH

BIND-DATA OF I-HTH

IF RTNCD-ERROR OF SQLRT

IF RTNCD-DUPL OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

MOVE 'INSERT-HEALTH-BENEFIT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

INSERT-HEALTH-BENEFIT-EXIT.

/*****

* *

XX300-INSERT-HEALTH-DEPENDENT SECTION.

XX300.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF I-HTH-DEP OF W-CNTL

SQL-STMT OF I-HTH-DEP

BIND-SETUP OF I-HTH-DEP

BIND-DATA OF I-HTH-DEP

IF RTNCD-ERROR OF SQLRT

MOVE 'INSERT-HEALTH-DEPENDENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

INSERT-HEALTH-DEPENDENT-EXIT.

/*****

* *

XX400-INSERT-FSA-BENEFIT SECTION.

XX400.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF I-FSA OF W-CNTL

SQL-STMT OF I-FSA

BIND-SETUP OF I-FSA

BIND-DATA OF I-FSA

IF RTNCD-ERROR OF SQLRT

IF RTNCD-DUPL OF SQLRT

SET RTNCD-OK OF SQLRT TO TRUE

ELSE

MOVE 'INSERT-FSA-BENEFIT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

INSERT-FSA-BENEFIT-EXIT.

/*****

* *

XX500-NOTIFY-BILLING SECTION.

XX500.

* *

CALL 'PSPBIINT' USING SQLRT

ADMIN

BIINT

IF RTNCD-ERROR OF SQLRT

MOVE 'NOTIFY-BILLING(PSPBIINT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

NOTIFY-BILLING-EXIT.

/*****

* *

XX600-UPDATE-HEALTH-BENEFIT SECTION.

XX600.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF U-XHTH

SQL-STMT OF U-XHTH

BIND-SETUP OF U-XHTH

BIND-DATA OF U-XHTH

IF RTNCD-ERROR OF SQLRT

MOVE 'UPDATE-HEALTH-BENEFIT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

UPDATE-HEALTH-BENEFIT-EXIT.

/*****

* *

XX610-DELETE-HEALTH-DEPENDENT SECTION.

XX610.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF D-XDEP

SQL-STMT OF D-XDEP

BIND-SETUP OF D-XDEP

BIND-DATA OF D-XDEP

IF RTNCD-ERROR OF SQLRT

MOVE 'DELETE-HEALTH-DEPENDENT' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-HEALTH-DEPENDENT-EXIT.

/*****

* *

XY000-DELETE-COBRA-BENEFIT SECTION.

XY000.

* *

IF REPROCESS-PLAN-YES OF W-SW

PERFORM VARYING WPARTIC-IDX FROM 1 BY 1

UNTIL WPARTIC-IDX > WPARTIC-COUNT OF W-EVENT

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

TO EMPLID OF D-BP-PGM

MOVE BENEFIT-RCD-NO OF W-EVENT

TO EMPL-RCD-NO OF D-BP-PGM

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF D-BP-PGM

PERFORM XY100-DELETE-COBRA-BP-PGM

END-PERFORM

END-IF

PERFORM VARYING WPLAN-IDX FROM 1 BY 1

UNTIL WPLAN-IDX > WPLAN-COUNT OF W-EVENT

SET WPARTIC-IDX TO 1

SEARCH WPARTIC-DATA

WHEN DEPENDENT-BENEF OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

= DEPENDENT-BENEF

OF WPARTIC-DATA(WPARTIC-IDX)

CONTINUE

END-SEARCH

MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

TO EMPLID OF D-BP-PLN

MOVE BENEFIT-RCD-NO OF W-EVENT

TO EMPL-RCD-NO OF D-BP-PLN

IF COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

MOVE COBRA-EVENT-ID OF W-EVENT

TO COBRA-EVENT-ID OF D-BP-PLN

ELSE

MOVE CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-ID OF D-BP-PLN

END-IF

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF D-BP-PLN

IF (COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

AND CBR-ENROLLED OF W-EVENT(WPLAN-IDX))

OR (COBRA-EVENT-SECONDARY OF W-EVENT(WPLAN-IDX)

AND COBRA-COVERAGE-INSERTED

OF W-EVENT(WPLAN-IDX))

OR ((COBRA-EVENT-INITIAL OF W-EVENT(WPLAN-IDX)

OR COBRA-EVENT-SECONDARY

OF W-EVENT(WPLAN-IDX))

AND CBR-ENROLL-NO OF W-EVENT(WPLAN-IDX)

AND COBRA-ELECT-WAIVE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX))

PERFORM XY200-DELETE-COBRA-BP-PLN

IF PLAN-TYPE-HEALTH OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

PERFORM XY300-DELETE-COBRA-HTH

PERFORM XY400-DELETE-COBRA-HTH-DEP

END-IF

IF PLAN-TYPE-FSA OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM XY500-DELETE-COBRA-FSA

END-IF

IF BILLING-YES OF W-CNTL

AND COBRA-ELECT-ELECT

OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

INITIALIZE BIINT

SET BILL-FUNCTION-DELETE OF BIINT TO TRUE

MOVE EMPLID OF D-BP-PLN TO EMPLID OF BIINT

MOVE EMPL-RCD-NO OF D-BP-PLN

TO EMPL-RCD-NO OF BIINT

MOVE COBRA-EVENT-ID OF D-BP-PLN

TO COBRA-EVENT-ID OF BIINT

MOVE PLAN-TYPE OF D-BP-PLN

TO PLAN-TYPE OF BIINT

SET BILLING-STATUS-INACTIVE OF BIINT TO TRUE

SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE

MOVE ZERO TO RATE-PERCENT OF BIINT

MOVE ZERO TO RATE-AMOUNT OF BIINT

MOVE ZERO TO EVENT-ID OF BIINT

MOVE SPACE TO BILL-EFFDT OF BIINT

PERFORM XX500-NOTIFY-BILLING

END-IF

END-IF

IF COBRA-EVENT-SECONDARY OF W-EVENT(WPLAN-IDX)

AND COBRA-COVERAGE-EXTENDED OF W-EVENT(WPLAN-IDX)

PERFORM XY600-GET-INIT-COVRG-END-DT

IF COBRA-TERM-DT OF S-INITEND = SPACE

MOVE COVERAGE-END-DT OF S-INITEND

TO BEGIN-DT OF DTWRK

MOVE 1 TO DAYS OF DTWRK

```
SET OPTION-ADD-DAYS OF DTWRK TO TRUE

CALL 'PTPDTWRK' USING  DTWRK

MOVE END-DT OF DTWRK TO EFFDT OF U-HTHTERM

MOVE END-DT OF DTWRK

      TO COVERAGE-BEGIN-DT OF U-HTHTERM

ELSE

      MOVE COBRA-TERM-DT OF S-INITEND

            TO EFFDT OF U-HTHTERM

END-IF


MOVE COBRA-EMPLID OF W-EVENT(WPARTIC-IDX)

      TO EMPLID OF U-HTHTERM

MOVE BENEFIT-RCD-NO OF W-EVENT

      TO EMPL-RCD-NO OF U-HTHTERM

MOVE CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

      TO COBRA-EVENT-ID OF U-HTHTERM

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

      TO PLAN-TYPE OF U-HTHTERM


IF COBRA-TERM-DT OF W-EVENT(WPLAN-IDX) = SPACE


      MOVE COVERAGE-END-DT OF W-EVENT(WPLAN-IDX)

            TO BEGIN-DT OF DTWRK

      MOVE 1 TO DAYS OF DTWRK

      SET OPTION-ADD-DAYS OF DTWRK TO TRUE

      CALL 'PTPDTWRK' USING  DTWRK
```

MOVE END-DT OF DTWRK TO EFFDT-2 OF U-HTHTERM

ELSE

MOVE COBRA-TERM-DT OF W-EVENT(WPLAN-IDX)

TO EFFDT-2 OF U-HTHTERM

END-IF

IF PLAN-TYPE-HEALTH OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX)

PERFORM XY700-UPDATE-COBRA-HTH-TERM

END-IF

IF PLAN-TYPE-FSA OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

PERFORM XY800-UPDATE-COBRA-FSA-TERM

END-IF

END-IF

END-PERFORM

.

DELETE-COBRA-BENEFIT-EXIT.

/*****

*

*

XY100-DELETE-COBRA-BP-PGM SECTION.

XY100.

*

*

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF D-BP-PGM OF W-CNTL

SQL-STMT OF D-BP-PGM

BIND-SETUP OF D-BP-PGM

BIND-DATA OF D-BP-PGM

IF RTNCD-ERROR OF SQLRT

MOVE 'DELETE-COBRA-BP-PGM' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-COBRA-BP-PGM-EXIT.

/*****

*

*

XY200-DELETE-COBRA-BP-PLN SECTION.

XY200.

*

*

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF D-BP-PLN OF W-CNTL

SQL-STMT OF D-BP-PLN

BIND-SETUP OF D-BP-PLN

BIND-DATA OF D-BP-PLN

IF RTNCD-ERROR OF SQLRT

MOVE 'DELETE-COBRA-BP-PLN' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-COBRA-BP-PLN-EXIT.

/*****

* *

XY300-DELETE-COBRA-HTH SECTION.

XY300.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF D-HTH OF W-CNTL

SQL-STMT OF D-HTH

BIND-SETUP OF D-BP-PLN

BIND-DATA OF D-BP-PLN

IF RTNCD-ERROR OF SQLRT

MOVE 'DELETE-COBRA-HTH' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-COBRA-HTH-EXIT.

/*****

*

*

XY400-DELETE-COBRA-HTH-DEP SECTION.

XY400.

*

*

*****/

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF D-HTH-DEP OF W-CNTL

SQL-STMT OF D-HTH-DEP

BIND-SETUP OF D-BP-PLN

BIND-DATA OF D-BP-PLN

IF RTNCD-ERROR OF SQLRT

MOVE 'DELETE-COBRA-HTH-DEP' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-COBRA-HTH-DEP-EXIT.

/*****

* *

XY500-DELETE-COBRA-FSA SECTION.

XY500.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF D-FSA OF W-CNTL

SQL-STMT OF D-FSA

BIND-SETUP OF D-BP-PLN

BIND-DATA OF D-BP-PLN

IF RTNCD-ERROR OF SQLRT

MOVE 'DELETE-COBRA-FSA' TO ERR-SECTION OF SQLRT
PERFORM ZZ000-SQL-ERROR
END-IF

DELETE-COBRA-FSA-EXIT.

/*****

* *

XY600-GET-INIT-COVRG-END-DT SECTION.

XY600.

* *

MOVE EMPLID OF W-EVENT

TO EMPLID OF BIND-DATA OF S-INITEND

MOVE BENEFIT-RCD-NO OF W-EVENT

TO BENEFIT-RCD-NO OF BIND-DATA OF S-INITEND

MOVE CBR-INIT-EVENT-ID OF W-EVENT(WPLAN-IDX)

TO COBRA-EVENT-ID OF BIND-DATA OF S-INITEND

MOVE DEPENDENT-BENEF OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO DEPENDENT-BENEF OF BIND-DATA OF S-INITEND

MOVE PLAN-TYPE OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

TO PLAN-TYPE OF S-INITEND

CALL 'PTPSQLRT' USING ACTION-SELECT OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF S-INITEND

BIND-SETUP OF S-INITEND

BIND-DATA OF S-INITEND

SELECT-SETUP OF S-INITEND

SELECT-DATA OF S-INITEND

IF RTNCD-ERROR OF SQLRT

MOVE 'GET-INIT-COVRG-END-DT(SELECT)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

INITIALIZE SELECT-DATA OF S-INITEND

CALL 'PTPSQLRT' USING ACTION-FETCH OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

IF RTNCD-ERROR OF SQLRT

IF RTNCD-END OF SQLRT

DISPLAY 'Initial Event Not Found For '

'Emplid/Rcd#/CBREvtID/PlanType/INITENDID: '

DISPLAY EMPLID OF W-EVENT ' / '

BENEFIT-RCD-NO OF W-EVENT ' / '

COBRA-EVENT-ID OF W-EVENT ' / '

PLAN-TYPE OF WPLAN-DATA

OF W-EVENT(WPLAN-IDX) ' / '

CBR-INIT-EVENT-ID

OF W-EVENT(WPLAN-IDX) ' . '

MOVE 'GET-INIT-COVRG-END-DT(FETCH)'

TO ERR-SECTION OF SQLRT

SET RTNCD-USER-ERROR OF SQLRT TO TRUE

PERFORM ZZ000-SQL-ERROR

ELSE

MOVE 'GET-INIT-COVRG-END-DT(FETCH)'

TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

END-IF

.

GET-INIT-COVRG-END-DT-EXIT.

/*****

*

*

XY700-UPDATE-COBRA-HTH-TERM SECTION.

XY700.

*

*

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF U-HTHTERM OF W-CNTL

SQL-STMT OF U-HTHTERM

BIND-SETUP OF U-HTHTERM

BIND-DATA OF U-HTHTERM

IF RTNCD-ERROR OF SQLRT

MOVE 'UPDATE-COBRA-HTH-TERM' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

IF BILLING-YES OF W-CNTL

AND COBRA-ELECT-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

INITIALIZE BIINT

SET BILL-FUNCTION-DELETE OF BIINT TO TRUE

MOVE EMPLID OF U-HTHTERM TO EMPLID OF BIINT

MOVE EMPL-RCD-NO OF U-HTHTERM TO EMPL-RCD-NO OF BIINT

MOVE COBRA-EVENT-ID OF U-HTHTERM

TO COBRA-EVENT-ID OF BIINT

MOVE PLAN-TYPE OF U-HTHTERM TO PLAN-TYPE OF BIINT
SET BILLING-STATUS-INACTIVE OF BIINT TO TRUE
SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE
MOVE ZERO TO RATE-PERCENT OF BIINT
MOVE ZERO TO RATE-AMOUNT OF BIINT
MOVE ZERO TO EVENT-ID OF BIINT
MOVE EFFDT-2 OF U-HTHTERM TO BILL-EFFDT OF BIINT
PERFORM XX500-NOTIFY-BILLING

INITIALIZE BIINT

SET BILL-FUNCTION-ADD OF BIINT TO TRUE
MOVE EMPLID OF U-HTHTERM TO EMPLID OF BIINT
MOVE EMPL-RCD-NO OF U-HTHTERM TO EMPL-RCD-NO OF BIINT
MOVE COBRA-EVENT-ID OF U-HTHTERM

TO COBRA-EVENT-ID OF BIINT

MOVE PLAN-TYPE OF U-HTHTERM TO PLAN-TYPE OF BIINT
SET BILLING-STATUS-INACTIVE OF BIINT TO TRUE
SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE
MOVE ZERO TO RATE-PERCENT OF BIINT
MOVE ZERO TO RATE-AMOUNT OF BIINT
MOVE ZERO TO EVENT-ID OF BIINT
MOVE EFFDT OF U-HTHTERM TO BILL-EFFDT OF BIINT
PERFORM XX500-NOTIFY-BILLING

END-IF

UPDATE-COBRA-HTH-TERM-EXIT.

/*****

* *

XY800-UPDATE-COBRA-FSA-TERM SECTION.

XY800.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF U-FSATERM OF W-CNTL

SQL-STMT OF U-FSATERM

BIND-SETUP OF U-HTHTERM

BIND-DATA OF U-HTHTERM

IF RTNCD-ERROR OF SQLRT

MOVE 'UPDATE-COBRA-FSA-TERM' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

IF BILLING-YES OF W-CNTL

AND COBRA-ELECT-ELECT OF WPLAN-DATA OF W-EVENT(WPLAN-IDX)

INITIALIZE BIINT

SET BILL-FUNCTION-DELETE OF BIINT TO TRUE

MOVE EMPLID OF U-HTHTERM TO EMPLID OF BIINT

MOVE EMPL-RCD-NO OF U-HTHTERM TO EMPL-RCD-NO OF BIINT

MOVE COBRA-EVENT-ID OF U-HTHTERM

TO COBRA-EVENT-ID OF BIINT

MOVE PLAN-TYPE OF U-HTHTERM TO PLAN-TYPE OF BIINT

SET BILLING-STATUS-INACTIVE OF BIINT TO TRUE

SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE

MOVE ZERO TO RATE-PERCENT OF BIINT

MOVE ZERO TO RATE-AMOUNT OF BIINT

MOVE ZERO TO EVENT-ID OF BIINT

MOVE EFFDT-2 OF U-HTHTERM TO BILL-EFFDT OF BIINT

PERFORM XX500-NOTIFY-BILLING

INITIALIZE BIINT

SET BILL-FUNCTION-ADD OF BIINT TO TRUE

MOVE EMPLID OF U-HTHTERM TO EMPLID OF BIINT

MOVE EMPL-RCD-NO OF U-HTHTERM TO EMPL-RCD-NO OF BIINT

MOVE COBRA-EVENT-ID OF U-HTHTERM

TO COBRA-EVENT-ID OF BIINT

MOVE PLAN-TYPE OF U-HTHTERM TO PLAN-TYPE OF BIINT

SET BILLING-STATUS-INACTIVE OF BIINT TO TRUE

SET RATE-QUALIFIER-TOTAL OF BIINT TO TRUE

MOVE ZERO TO RATE-PERCENT OF BIINT

MOVE ZERO TO RATE-AMOUNT OF BIINT

MOVE ZERO TO EVENT-ID OF BIINT

MOVE EFFDT OF U-HTHTERM TO BILL-EFFDT OF BIINT

PERFORM XX500-NOTIFY-BILLING

END-IF

.

UPDATE-COBRA-FSA-TERM-EXIT.

/*****

*

*

XY910-DELETE-COBRAVD-HTH SECTION.

XY910.

*

*

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF D-VHTH

BIND-SETUP OF D-VHTH

BIND-DATA OF D-VHTH

IF RTNCD-ERROR OF SQLRT

MOVE 'DELETE-COBRAVD-HTH' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-COBRAVD-HTH-EXIT.

/*****

*

*

XY920-DELETE-COBRAVD-HTH-DEP SECTION.

XY920.

*

*

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF D-VHTH-DP

BIND-SETUP OF D-VHTH

BIND-DATA OF D-VHTH

IF RTNCD-ERROR OF SQLRT

MOVE 'DELETE-COBRAVD-HTH-DEP' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-COBRAVD-HTH-DEP-EXIT.

/*****

* *

XY930-DELETE-COBRAVD-FSA SECTION.

XY930.

* *

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR-COMMON OF SQLRT

SQL-STMT OF D-VFSA

BIND-SETUP OF D-VHTH

BIND-DATA OF D-VHTH

IF RTNCD-ERROR OF SQLRT

MOVE 'DELETE-COBRAVD-FSA' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

DELETE-COBRAVD-FSA-EXIT.

/*****

*

*

XY940-UPDATE-VDEPEND-BENEF SECTION.

XY940.

*

*

CALL 'PTPSQLRT' USING ACTION-UPDATE OF SQLRT

SQLRT

SQL-CURSOR OF U-VDPBEN OF W-CNTL

SQL-STMT OF U-VDPBEN

BIND-SETUP OF U-VDPBEN

BIND-DATA OF U-VDPBEN

IF RTNCD-ERROR OF SQLRT

MOVE 'UPDATE-VDEPEND-BENEF' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

UPDATE-VDEPEND-BENEF-EXIT.

/*****

* *

XZ000-GET-CLOCK-TIME SECTION.

XZ000.

* *

ACCEPT TIME-COLLECT OF W-WK FROM TIME

COMPUTE TIME-CLOCK OF W-WK ROUNDED

= (((TIME-COLLECT-HR OF W-WK

* 60)

+ TIME-COLLECT-MM OF W-WK)

* 60)

+ TIME-COLLECT-SS OF W-WK

+ (TIME-COLLECT-HH OF W-WK

/ 100.00)

.

GET-CLOCK-TIME-EXIT.

/*****

* *

ZM000-MESSAGE SECTION.

ZM000.

* *

SET PAYROLL-STEP-COBRA OF PYMSG TO TRUE

MOVE EMPLID OF W-EVENT TO EMPLID OF PYMSG

MOVE BENEFIT-RCD-NO OF W-EVENT TO BENEFIT-RCD-NO OF PYMSG

CALL 'PSPPYMSG' USING SQLRT

PYMSG

IF RTNCD-ERROR OF SQLRT

MOVE 'MESSAGE' TO ERR-SECTION OF SQLRT

PERFORM ZZ000-SQL-ERROR

END-IF

.

MESSAGE-EXIT.

/*****

* *

ZZ000-SQL-ERROR SECTION.

ZZ000.

* *

* SQL ERROR PROCESSING *

* *

CALL 'PTPSQLRT' USING ACTION-ERROR OF SQLRT
SQLRT

PERFORM ZZ999-GET-DISPLAY-TIME
DISPLAY 'COBRA did not finish at '
TIME-OUT OF W-WK ' !!!!!'

COPY PSCRTNCD.

.
SQL-ERROR-EXIT.
STOP RUN.

/*****

* *

ZZ999-GET-DISPLAY-TIME SECTION.

ZZ999.

* *

ACCEPT TIME-OUT OF W-WK FROM TIME
INSPECT TIME-OUT OF W-WK CONVERTING SPACE TO ':'
INSPECT TIME-OUT OF W-WK CONVERTING '/' TO '.'

.

GET-DISPLAY-TIME-EXIT.