

COBOL IMS DB Sample Program

By Srinimf.com

```
*****
IDENTIFICATION DIVISION.
*****
PROGRAM-ID. IMSBATCH.
AUTHOR. ABC.
INSTALLATION. DEPARTMENT.
DATE-WRITTEN. DATE XX/XX/XX.
DATE-COMPILED.
*****
ENVIRONMENT DIVISION.
*****
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
SELECT OUTFILE ASSIGN TO UT-S-OUTFILE.
*****
DATA DIVISION.
*****
FILE SECTION.
FD OUTFILE
RECORD IS VARYING IN SIZE FROM 383 TO 22727 CHARACTERS RECORDING MODE IS V
BLOCK CONTAINS 0 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS O-PLT.
* 283
01 O-PLT.
05 O-PLT-DATA-AREA.
    10 O-PLTROOT-PREFIX PIC X(001).
    10 O-PLTROOT-NAME PIC X(008).
    10 O-PLTROOT-TITLE PIC X(050).
    10 O-PLTROOT-TERM-PRT-TABLE PIC X(200).
    10 O-PLT-SEQ PIC 9(008).
    10 O-PLTPE-SEQ-LAST PIC 9(008).
    10 O-SEGPROF-SEQ-LAST PIC 9(008).
* 44
05 O-PLTPE.
    10 O-PLTPE-KEY-SEQ PIC 9(03).
    10 O-PLTPE-TYPE PIC X(01) . 10 O-PLTPE-NAME PIC X(08).
    10 O-PLTPE-ACCESS PIC X(01).
    10 O-PLTPE-TITLE PIC X(30).
    10 O-PLTPE-COURTPRF PIC X(01).
* 56
05 O-SEGPROF OCCURS 400 TIMES DEPENDING ON W-SEGPROF-SEQ-LAST
INDEXED BY O-SEGPROF-INDEX.
    10 O-SEGPROF-SEQ PIC 9(08).
    10 O-SEGPROF-KEY-PREFIX PIC X(01).
```

```
10 O-SEGPROF-KEY-SUFFIX PIC X(08).
10 O-SEGPROF-PLT-NAME PIC X(08).
10 O-SEGPROF-NOTE PIC X(30).
10 O-SEGPROF-ACCESS PIC X(01).
WORKING-STORAGE SECTION.
01 W-TABLES.
05 W-PLT-DATA-AREA.
10 W-PLTROOT-PREFIX PIC X(001).
10 W-PLTROOT-NAME PIC X(008).
10 W-PLTROOT-TITLE PIC X(050).
10 W-PLTROOT-TERM-PRT-TABLE PIC X(200).
10 W-PLT-SEQ PIC 9(008) VALUE 0.
10 W-PLTPE-SEQ-LAST-LIT PIC X(019) VALUE ' W-PLTPE-SEQ-LAST= '.
10 W-PLTPE-SEQ-LAST PIC 9(008).
10 W-SEGPROF-SEQ-LAST-LIT PIC X(022) VALUE ' W-SEGPROF-SEQ-LAST= '.
10 W-SEGPROF-SEQ-LAST PIC 9(008).
05 W-PLTPE OCCURS 1 TO 400 TIMES DEPENDING ON W-PLTPE-SEQ-LAST
INDEXED BY W-PLTPE-INDEX.
10 W-PLTPE-TYPE PIC X(01) . 88 PLTPE-MENU VALUE 'M'.
88 PLTPE-PROGRAM VALUE 'P'.
10 W-PLTPE-NAME PIC X(08).
10 W-PLTPE-KEY-SEQ PIC 9(03).
10 W-PLTPE-ACCESS PIC X(01).
88 PLTPE-ACCESS-VALUE VALUES ARE '0' THRU '9'.
10 W-PLTPE-TITLE PIC X(30).
10 W-PLTPE-COURTPRF PIC X(01).
88 W-PLTPE-COURTPRF-YES VALUE 'Y'.
88 W-PLTPE-COURTPRF-NO VALUE 'N'.
05 W-SEGPROF OCCURS 1 TO 400 TIMES DEPENDING ON
W-SEGPROF-SEQ-LAST
INDEXED BY W-SEGPROF-INDEX.
10 W-SEGPROF-KEY-PREFIX PIC X(01).
10 W-SEGPROF-KEY-SUFFIX PIC X(08).
10 W-SEGPROF-PLT-NAME PIC X(08).
10 W-SEGPROF-NOTE PIC X(30).
10 W-SEGPROF-SECURITY-TABLE PIC X(400).
01 DLI-FUNCTION-CODES.
05 EQ PIC X(02) VALUE '='.
05 GE PIC X(02) VALUE '>='.
05 GT PIC X(02) VALUE '>'.
05 GU PIC X(04) VALUE 'GU '.
05 GHU PIC X(04) VALUE 'GHU '.
05 GN PIC X(04) VALUE 'GN '.
05 GHN PIC X(04) VALUE 'GHN '.
05 GNP PIC X(04) VALUE 'GNP '.
05 GHNP PIC X(04) VALUE 'GHNP'.
05 DLET PIC X(04) VALUE 'DLET'.
05 ISRT PIC X(04) VALUE 'ISRT'.
05 REPL PIC X(04) VALUE 'REPL'.
05 CHKP PIC X(04) VALUE 'CHKP'.
05 XRST PIC X(04) VALUE 'XRST'.
05 PCB PIC X(04) VALUE 'PCB '.
05 FUNCTION-CODE PIC X(04) VALUE ' '.
05 TERMINATE-DLI PIC X(04) VALUE 'TERM'.
05 PARM-COUNT PIC S9(08) COMP VALUE +1.
* 05 PSB-READ PIC X(8) VALUE 'MUPROFRD'.
```

```
01 DLI-STATUS-CODES.
05 GET-GOOD PIC X(02) VALUE SPACE.
05 GET-NOT-FOUND PIC X(02) VALUE 'GE'.
05 GET-END-DB PIC X(02) VALUE 'GB'.
05 GN-CROSS-BOUNDARY PIC X(02) VALUE 'GA'.
*****
THESE SSA'S ARE FROM THE COPY BOOK SECMSSA
*****
READ 01 = GU MENUROOT(PREFIX =P)
01 MENUROOT-SSA-PREFIX-P.
05 FILLER PIC X(17) VALUE 'MENUROOT(PREFIX '.
05 MENUROOT-SSA-OPER PIC X(02) VALUE ' ='.
05 MENUROOT-SSA-KEY.
    10 MENUROOT-SSA-PREFIX PIC X(01) VALUE 'P'.
05 FILLER PIC XX VALUE ')'.
01 PLTROOT-LIT PIC X(10) VALUE ' PLTROOT= '.
01 PLTROOT.
05 PLT-AREA.
    10 PLTROOT-KEY.
15 PLTROOT-PREFIX PIC X(1).
15 PLTROOT-NAME PIC X(08).
10 PLTROOT-TITLE PIC X(50).
10 PLTROOT-PFKEY-TABLE.
15 PLTROOT-PFKEY-RECORD
OCCURS 24 TIMES
INDEXED BY PLTROOT-PFKEY-INDEX.
20 PLTROOT-PFKEY PIC X(10).
88 VALID-PFKEY-VALUES VALUES ARE 'DISPLAY' 'UPDATE' 'FIND'
'BOTTOM' 'UP' 'DOWN'
'SWAP' 'SPLIT' 'MENU'
'PRINT' 'CLEAR' 'SOFTEXIT'
'LIST' 'TOP' 'RETURN'.
    10 PLTROOT-TERMINAL-PRINTER-TABLE.
15 PLTROOT-PRINTER-RECORD
OCCURS 10 TIMES
INDEXED BY PLTROOT-PRT-INDEX.
20 PLTROOT-PRINTER-LITERAL PIC X(20).
01 FILLER PIC X(500).
*****
READ 02 = GNP PLTPE I/O AREA = PLTPE
STATUS = GE END-OF-SEGMENTS
*****
01 UNQUAL-SSA-PLTPE PIC X(09) VALUE 'PLTPE '.
01 PLTPE-LIT PIC X(08) VALUE ' PLTPE= '.
01 PLTPE.
05 PLTPE-KEY.
10 PLTPE-KEY-SEQ PIC 9(03).
05 PLTPE-TYPE PIC X.
88 PLTPE-PROGRAM VALUE 'P'.
05 PLTPE-NAME PIC X(8).
05 PLTPE-TITLE PIC X(30).
05 PLTPE-ACCESS PIC X.
88 PLTPE-ACCESS-VALUE VALUES ARE '0' THRU '9'.
05 PLTPE-COURTPRF PIC X.
88 PLTPE-COURTPRF-YES VALUE 'Y'.
88 PLTPE-COURTPRF-NO VALUE 'N'.
```

```
01 FILLER PIC X(500).
*****
READ 03 = GU MENUROOT(PREFIX =S) THEN GN * I/O AREA = SEGPROF
*****
01 MENUROOT-SSA-PREFIX-S.
05 FILLER PIC X(19) VALUE 'MENUROOT(PREFIX ='.
05 MENUROOT-PREFIX-SSA-KEY PIC X VALUE 'S'.
05 FILLER PIC XX VALUE ') '.
01 SEGPROF-LIT PIC X(11) VALUE ' SEGPROF= '.
01 SEGPROF.
05 SEGPROF-KEY-AREA.
10 SEGPROF-KEY-PREFIX PIC X.
88 VALID-SEGPROF-KEY-PREFIX VALUE 'S'.
10 SEGPROF-KEY-SUFFIX PIC X(8).
05 SEGPROF-PLT-NAME PIC X(8).
05 SEGPROF-NOTE PIC X(30).
05 SEGPROF-SECURITY-TABLE PIC X(400).
05 SEGPROF-SECURITY-BYTE REDEFINES SEGPROF-SECURITY-TABLE OCCURS 400
TIMES.
10 SEGPROF-BYTE PIC X.
01 FILLER PIC X(500).
*****
PROGRAM WORK FIELDS
*****
01 W-ZERO PIC 9(008) VALUE 0.
01 READ-PLTROOT-COUNT-LIT PIC X(22) VALUE ' READ-PLTROOT-COUNT= '.
01 READ-PLTROOT-COUNT PIC 9(008) VALUE 0.
01 READ-PLTPE-COUNT-LIT PIC X(19) VALUE ' READ-PLTPE-COUNT= '.
01 READ-PLTPE-COUNT PIC 9(008) VALUE 0.
01 READ-PLTPE-COUNT-LIT PIC X(22) VALUE ' READ-SEGPROF-COUNT= '.
01 READ-SEGPROF-COUNT PIC 9(008) VALUE 0.
01 W-PLTPE-SEQ-MAX PIC 9(008) VALUE 400.
01 W-SEGPROF-SEQ-MAX PIC 9(008) VALUE 400.
01 READ-PLT-OPCODE PIC X(004) VALUE SPACES.
01 READ-SEC-OPCODE PIC X(004) VALUE SPACES.
01 D-PLTPE-INDEX PIC 9(008).
01 D-SEGPROF-INDEX PIC 9(008).
01 BEGIN-PLT PIC X(008) VALUE ' '.
01 W-SEGPROF-ACCESS-TABLE.
05 W-SEGPROF-ACCESS-BYTES PIC X(400).
05 W-SEGPROF-ACCESS-BYTE
REDEFINES W-SEGPROF-ACCESS-BYTES OCCURS 400 TIMES
INDEXED BY W-ACCESS-INDEX.
10 W-SEGPROF-BYTE PIC X.
*****
LINKAGE SECTION.
*****
DLI PROGRAM COMMUNICATION BLOCK FOR PLTROOT SEGMENT
AND PLTPE DEPENDENT SEGMENTS
*****
01 DLIPCB1.
05 PCB-DBD-NAME-1.
10 PCBDBDNM-1 PIC X(8).
05 PCB-LEVEL-1 PIC 9(2).
05 PCB-STATUS-CODE-1.
10 PCBSA-1 PIC XX.
```

```
05 PCB-PROCESSING-OPTIONS-1.
10 PCBPROPT-1 PIC X(4).
05 FILLER PIC S9(5) COMP.
05 PCB-SEGMENT-NAME-1.
10 PCBSEGNM-1 PIC X(8).
05 PCB-LENGTH-FEEDBACK-1.
10 PCBLKEYFB-1 PIC S9(5) COMP.
05 PCB-NUMBER-SEGMENTS-1 PIC S9(5) COMP.
05 PCB-KEY-FEEDBACK-1.
10 PCBKEYFB-1 PIC X(58).
*****
DLI PROGRAM COMMUNICATION BLOCK FOR SEGPROF SEGMENT
*****
01 DLIPCB2.
05 PCB-DBD-NAME-2.
10 PCBDBDNM-2 PIC X(8).
05 PCB-LEVEL-2 PIC 9(2).
05 PCB-STATUS-CODE-2.
10 PCBSA-2 PIC XX.
05 PCB-PROCESSING-OPTIONS-2.
10 PCBPROPT-2 PIC X(4).
05 FILLER PIC S9(5) COMP.
05 PCB-SEGMENT-NAME-2.
10 PCBSEGNM-2 PIC X(8).
05 PCB-LENGTH-FEEDBACK-2.
10 PCBLKEYFB-2 PIC S9(5) COMP.
05 PCB-NUMBER-SEGMENTS-2 PIC S9(5) COMP.
05 PCB-KEY-FEEDBACK-2.
10 PCBKEYFB-2 PIC X(58).
*****
PROGRAM INITIALIZATION
*****
PROCEDURE DIVISION.
ENTRY 'DLITCBL' USING DLIPCB1 DLIPCB2.
DISPLAY '*** START IMSBATCH'.
OPEN OUTPUT OUTFILE.
MOVE 0 TO O-PLT-SEQ.
MOVE GU TO READ-PLT-OPCODE.
MOVE GU TO READ-SEC-OPCODE.
*****
READ PLT MENUROOT SEGMENTS
*****
RTN-READ-PLT.
    CALL 'CBLTDLI' USING READ-PLT-OPCODE DLIPCB1
PLTROOT
    MENUROOT-SSA-PREFIX-P.
    IF PCBSA-1 EQUAL GET-GOOD
    GO TO RTN-READ-PLT-GOOD.
    GO TO END-OF-DATABASE.
    RTN-READ-PLT-GOOD.
    IF BEGIN-PLT EQUAL SPACES NEXT SENTENCE
ELSE
IF PLTROOT-NAME LESS THAN BEGIN-PLT DISPLAY '*** SKIPPING ' PLTROOT-NAME
MOVE GN TO READ-PLT-OPCODE
GO TO RTN-READ-PLT.
ADD 1 TO READ-PLTROOT-COUNT.
```

```
ADD 1 TO W-PLT-SEQ.
MOVE 0 TO W-PLTPE-SEQ-LAST.
MOVE 0 TO W-SEGPROF-SEQ-LAST.
MOVE GN TO READ-PLT-OPCODE.
MOVE PLTROOT-PREFIX TO W-PLTROOT-PREFIX.
MOVE PLTROOT-NAME TO W-PLTROOT-NAME.
INSPECT W-PLTROOT-NAME
REPLACING ALL LOW-VALUE BY SPACE.
MOVE PLTROOT-TITLE TO W-PLTROOT-TITLE.
INSPECT W-PLTROOT-TITLE
REPLACING ALL LOW-VALUE BY SPACE.
MOVE PLTROOT-TERMINAL-PRINTER-TABLE
                                TO W-PLTROOT-TERM-PRT-TABLE.
INSPECT W-PLTROOT-TERM-PRT-TABLE
REPLACING ALL LOW-VALUE BY SPACE.
* DISPLAY ' '.
* DISPLAY '*** RTN-READ-PLT = ' W-PLT-SEQ.
DISPLAY '*** PROCESSING ' W-PLTROOT-NAME.
*****
READ PLTPE DEPENDENT SEGMENTS
*****
RTN-READ-PLTPE.
CALL 'CBLTDLI' USING GNP
DLIPCB1
PLTPE
UNQUAL-SSA-PLTPE.
IF PCBSA-1 EQUAL GET-GOOD
GO TO RTN-READ-PLTPE-GOOD.
IF PCBSA-1 EQUAL GN-CROSS-BOUNDARY OR EQUAL GET-NOT-FOUND
OR EQUAL GET-END-DB
GO TO RTN-READ-PLTPE-END.
GO TO END-OF-DATABASE.
RTN-READ-PLTPE-GOOD.
IF W-PLTPE-SEQ-LAST EQUAL
    TO W-PLTPE-SEQ-MAX
DISPLAY '*** W-PLTPE-SEQ-MAX MAXIMUM ' W-PLTPE-SEQ-MAX ' COUNT EXCEEDED'
GO TO END-OF-DATABASE.
ADD 1 TO READ-PLTPE-COUNT.
ADD 1 TO W-PLTPE-SEQ-LAST.
SET W-PLTPE-INDEX TO W-PLTPE-SEQ-LAST.
MOVE PLTPE-TYPE TO W-PLTPE-TYPE(W-PLTPE-INDEX).
MOVE PLTPE-NAME TO W-PLTPE-NAME(W-PLTPE-INDEX).
INSPECT W-PLTPE-NAME(W-PLTPE-INDEX)
REPLACING ALL LOW-VALUE BY SPACE.
MOVE PLTPE-KEY-SEQ TO W-PLTPE-KEY-SEQ(W-PLTPE-INDEX).
MOVE PLTPE-ACCESS TO W-PLTPE-ACCESS(W-PLTPE-INDEX).
MOVE PLTPE-TITLE TO W-PLTPE-TITLE(W-PLTPE-INDEX).
INSPECT W-PLTPE-TITLE(W-PLTPE-INDEX)
REPLACING ALL LOW-VALUE BY SPACE.
MOVE PLTPE-COURTPRF TO W-PLTPE-COURTPRF(W-PLTPE-INDEX).
INSPECT W-PLTPE-COURTPRF(W-PLTPE-INDEX)
REPLACING ALL LOW-VALUE BY SPACE.
GO TO RTN-READ-PLTPE.
RTN-READ-PLTPE-END.
MOVE GU TO READ-SEC-OPCODE.
GO TO RTN-READ-SEGPROF.
```

```
*****
READ ALL SEGPROF SEGMENTS AND SELECT SEGMENTS THAT
MATCH THIS PLT
*****
RTN-READ-SEGPROF.
CALL 'CBLTDLI' USING READ-SEC-OPCODE DLIPCB2
SEGPROF
MENUROOT-SSA-PREFIX-S.
MOVE GN TO READ-SEC-OPCODE.
  IF PCBSA-2 EQUAL GET-GOOD
  GO TO RTN-READ-SEGPROF-GOOD.
  IF PCBSA-2 EQUAL GN-CROSS-BOUNDARY OR EQUAL GET-NOT-FOUND
  OR EQUAL GET-END-DB
  GO TO RTN-READ-SEGPROF-END.
  GO TO END-OF-DATABASE.
  RTN-READ-SEGPROF-GOOD.
  ADD 1 TO READ-SEGPROF-COUNT.
  IF SEGPROF-PLT-NAME NOT EQUAL TO W-PLTROOT-NAME GO TO RTN-READ-SEGPROF.
IF W-SEGPROF-SEQ-LAST GREATER THAN OR EQUAL TO W-SEGPROF-SEQ-MAX
DISPLAY '*** W-SEGPROF-SEQ-MAX MAXIMUM ' W-SEGPROF-SEQ-MAX ' COUNT EXCEEDED
'
GO TO END-OF-DATABASE.
ADD 1 TO W-SEGPROF-SEQ-LAST.
SET W-SEGPROF-INDEX TO W-SEGPROF-SEQ-LAST.
  MOVE SEGPROF-KEY-PREFIX TO
W-SEGPROF-KEY-PREFIX(W-SEGPROF-INDEX) .
  MOVE SEGPROF-KEY-SUFFIX TO
W-SEGPROF-KEY-SUFFIX(W-SEGPROF-INDEX) .
  INSPECT W-SEGPROF-KEY-SUFFIX(W-SEGPROF-INDEX) REPLACING ALL LOW-VALUE BY
SPACE.
  MOVE SEGPROF-PLT-NAME TO
W-SEGPROF-PLT-NAME(W-SEGPROF-INDEX) .
  INSPECT W-SEGPROF-PLT-NAME(W-SEGPROF-INDEX) REPLACING ALL LOW-VALUE BY
SPACE.
  MOVE SEGPROF-NOTE TO
W-SEGPROF-NOTE(W-SEGPROF-INDEX) .
  INSPECT W-SEGPROF-NOTE(W-SEGPROF-INDEX) REPLACING ALL LOW-VALUE BY SPACE.
  MOVE SEGPROF-SECURITY-TABLE TO
W-SEGPROF-SECURITY-TABLE(W-SEGPROF-INDEX) .
  INSPECT W-SEGPROF-SECURITY-TABLE(W-SEGPROF-INDEX) REPLACING ALL LOW-VALUE
BY SPACE.
  GO TO RTN-READ-SEGPROF.
  RTN-READ-SEGPROF-END.
  GO TO RTN-BUILD-RECORD.
*****
THE ROOT, DEPENDENT PLTPE AND OTHER SEGMENTS HAVE BEEN READ.
BUILD THE OUTPUT RECORD(S) .
*****
RTN-BUILD-RECORD.
  SET W-PLTPE-INDEX TO W-ZERO.
  SET W-SEGPROF-INDEX TO W-ZERO.
  RTN-BUILD-PLTPE.
  SET W-PLTPE-INDEX UP BY 1.
  IF W-PLTPE-INDEX GREATER THAN W-PLTPE-SEQ-LAST SET W-PLTPE-INDEX TO W-ZERO
  GO TO RTN-READ-PLT.
  RTN-BUILD-SEGPROF.
```

```
PERFORM RTN-MOVE-PLT THROUGH RTN-MOVE-PLT-END.
PERFORM RTN-MOVE-PLTPE THROUGH RTN-MOVE-PLTPE-END.
IF W-SEGPROF-SEQ-LAST GREATER THAN 0 PERFORM RTN-MOVE-SEGPROF THROUGH RTN-
MOVE-SEGPROF-END VARYING W-SEGPROF-INDEX FROM 1 BY 1 UNTIL W-SEGPROF-INDEX
GREATER THAN W-SEGPROF-SEQ-LAST
ELSE PERFORM RTN-MOVE-SEGPROF-DUMMY THROUGH RTN-MOVE-SEGPROF-DUMMY-END.
PERFORM RTN-WRITE-PLT THROUGH RTN-WRITE-PLT-END.
GO TO RTN-BUILD-PLTPE.
RTN-MOVE-PLT.
MOVE W-PLTROOT-PREFIX TO O-PLTROOT-PREFIX.
MOVE W-PLTROOT-NAME TO O-PLTROOT-NAME.
MOVE W-PLTROOT-TITLE TO O-PLTROOT-TITLE.
MOVE W-PLTROOT-TERM-PRT-TABLE TO O-PLTROOT-TERM-PRT-TABLE.
MOVE W-PLT-SEQ TO O-PLT-SEQ.
MOVE W-PLTPE-SEQ-LAST TO O-PLTPE-SEQ-LAST.
MOVE W-SEGPROF-SEQ-LAST TO O-SEGPROF-SEQ-LAST.
* DISPLAY ' '
* DISPLAY 'O-PLTROOT-SEQ = ' O-PLT-SEQ.
* DISPLAY 'O-PLTROOT-NAME = ' O-PLTROOT-NAME.
RTN-MOVE-PLT-END.
EXIT.
RTN-MOVE-PLTPE.
MOVE W-PLTPE-TYPE(W-PLTPE-INDEX) TO O-PLTPE-TYPE.
MOVE W-PLTPE-NAME(W-PLTPE-INDEX) TO O-PLTPE-NAME.
MOVE W-PLTPE-KEY-SEQ(W-PLTPE-INDEX) TO O-PLTPE-KEY-SEQ.
MOVE W-PLTPE-ACCESS(W-PLTPE-INDEX) TO O-PLTPE-ACCESS.
MOVE W-PLTPE-TITLE(W-PLTPE-INDEX) TO O-PLTPE-TITLE.
MOVE W-PLTPE-COURTPRF(W-PLTPE-INDEX) TO O-PLTPE-COURTPRF.
RTN-MOVE-PLTPE-END.
* DISPLAY 'O-PLTPE-NAME = ' O-PLTPE-NAME.
EXIT.
RTN-MOVE-SEGPROF.
SET O-SEGPROF-INDEX TO W-SEGPROF-INDEX.
MOVE W-SEGPROF-KEY-PREFIX(W-SEGPROF-INDEX) TO O-SEGPROF-KEY-PREFIX(O-
SEGPROF-INDEX).
MOVE W-SEGPROF-KEY-SUFFIX(W-SEGPROF-INDEX) TO O-SEGPROF-KEY-SUFFIX(O-
SEGPROF-INDEX).
MOVE W-SEGPROF-PLT-NAME(W-SEGPROF-INDEX) TO O-SEGPROF-PLT-NAME(O-SEGPROF-
INDEX).
MOVE W-SEGPROF-NOTE(W-SEGPROF-INDEX) TO O-SEGPROF-NOTE(O-SEGPROF-INDEX).
MOVE W-SEGPROF-SECURITY-TABLE(W-SEGPROF-INDEX) TO W-SEGPROF-ACCESS-TABLE.
SET W-ACCESS-INDEX TO W-PLTPE-KEY-SEQ(W-PLTPE-INDEX).
MOVE W-SEGPROF-BYTE(W-ACCESS-INDEX) TO O-SEGPROF-ACCESS(O-SEGPROF-INDEX).
ADD 1 TO O-SEGPROF-SEQ(O-SEGPROF-INDEX).
* DISPLAY 'O-SEGPROF-KEY-SUFFIX = ' O-SEGPROF-KEY-SUFFIX.
RTN-MOVE-SEGPROF-END.
EXIT.
RTN-MOVE-SEGPROF-DUMMY.
DISPLAY '*** DUMMY SECURITY FOR ' O-PLTROOT-NAME.
ADD 1 TO W-SEGPROF-SEQ-LAST.
SET O-SEGPROF-INDEX TO W-SEGPROF-SEQ-LAST.
MOVE W-SEGPROF-SEQ-LAST TO O-SEGPROF-SEQ-LAST.
MOVE 0 TO O-SEGPROF-SEQ(O-SEGPROF-INDEX).
MOVE 'S' TO O-SEGPROF-KEY-PREFIX(O-SEGPROF-INDEX).
MOVE 'DUMMY ' TO O-SEGPROF-KEY-SUFFIX(O-SEGPROF-INDEX).
MOVE 'DUMMY ' TO O-SEGPROF-PLT-NAME(O-SEGPROF-INDEX).
```

```
MOVE 'DUMMY ' TO O-SEGPROF-NOTE(O-SEGPROF-INDEX) .
MOVE 0 TO O-SEGPROF-ACCESS(O-SEGPROF-INDEX) .
RTN-MOVE-SEGPROF-DUMMY-END.
EXIT.
RTN-WRITE-PLT.
WRITE O-PLT.
RTN-WRITE-PLT-END.
EXIT.
```

```
*****
```

```
PROGRAM EXIT AFTER THE MAXIMUM NUMBER OF USER RECORDS HAVE
BEEN READ AND/OR THERE ARE NO MORE USER RECORDS TO READ
```

```
*****
```

```
END-OF-DATABASE.
```

```
DISPLAY '*** END-OF-DATABASE'.
DISPLAY 'DLIPCB1 = ' DLIPCB1.
DISPLAY 'DLIPCB2 = ' DLIPCB2.
CLOSE OUTFILE.
DISPLAY 'READ-PLTROOT-COUNT = ' READ-PLTROOT-COUNT.
DISPLAY 'READ-PLTPE-COUNT = ' READ-PLTPE-COUNT.
DISPLAY 'READ-SEGPROF-COUNT = ' READ-SEGPROF-COUNT.
DISPLAY '*** END-OF-IMSBATCH'.
GOBACK.
```