

APPENDIX F - COBOL EXAMPLES

This chapter covers the following topics:

- Example 1
- Example 2
- Example 3

Example 1

```

IDENTIFICATION DIVISION.
PROGRAM-ID. CEX1.
REMARKS.
* AN EXAMPLE OF SOFT COUPLING WITH A SEARCH CRITERION WHICH      *
* CONTAINS FIELDS TAKEN FROM TWO FILES. THE FIELDS PERSONNEL-ID *
* NAME, FIRST-NAME, BIRTH AND SEX (FROM THE MAIN FILE,          *
* PERSONNEL-ID) ARE PRINTED FOR RECORDS THAT SATISFY THE        *
* FOLLOWING CONDITION:                                          *
*   PERSONNEL-ID BETWEEN 10000001 AND 19999999                *
*   MODEL-YEAR-MAKE >                                         *
*   CLASS = 'C'                                               *
ENVIRONMENT DIVISION.

DATA DIVISION.
WORKING-STORAGE SECTION.
01  START-MODEL-YEAR-MAKE.
    02  START-MODEL          PIC X(20) VALUE 'MERCEDES-BENZ'.
    02  START-YEAR-MAKE     PIC 9(2)  VALUE 86.
*
01  HEADER.
    02  FILLER              PIC X(12) VALUE 'PERSONNEL-ID'.
    02  FILLER              PIC X(8)  VALUE SPACE.
    02  FILLER              PIC X(4)  VALUE 'NAME'.
    02  FILLER              PIC X(13) VALUE SPACE.
    02  FILLER              PIC X(10) VALUE 'FIRST NAME'.
    02  FILLER              PIC X(8)  VALUE SPACE.
    02  FILLER              PIC X(5)  VALUE 'BIRTH'.
    02  FILLER              PIC X(1)  VALUE SPACE.
    02  FILLER              PIC X(3)  VALUE 'SEX'.
01  HEADER2                PIC X(64) VALUE ALL '*'.
01  SPACE-LINE             PIC X(80) VALUE SPACE.
01  LINE1.
    02  FILLER              PIC X(2)  VALUE SPACE.
    02  PERSONNEL-NR       PIC X(8)  VALUE SPACE.
    02  FILLER              PIC X(3)  VALUE SPACE.
    02  LAST-NAME          PIC X(20) VALUE SPACE.
    02  FILLER              PIC X(1)  VALUE SPACE.
    02  F-NAME             PIC X(20) VALUE SPACE.
    02  FILLER              PIC X(1)  VALUE SPACE.
    02  BIRTHDAY           PIC X(6)  VALUE SPACE.
    02  FILLER              PIC X(1)  VALUE SPACE.
    02  KIND                PIC X(1)  VALUE SPACE.
*
    EXEC ADABAS
BEGIN DECLARE SECTION
END-EXEC

```

```

*
      EXEC ADABAS
      DECLARE EMPL CURSOR FOR
      SELECT PERSONNEL-ID, NAME, FIRST-NAME, BIRTH, SEX
      FROM EMPLOYEES, VEHICLES
      WHERE EMPLOYEES.PERSONNEL-ID = VEHICLES.PERSONNEL-ID
            AND PERSONNEL-ID BETWEEN '10000001' AND '19999999'
            AND VEHICLES.MODEL-YEAR-MAKE > :START-MODEL-YEAR-MAKE
            AND VEHICLES.CLASS = 'C'
      END-EXEC

PROCEDURE DIVISION.
*
      DISPLAY HEADER.
      DISPLAY HEADER2.
      DISPLAY SPACE-LINE.
*
      EXEC ADABAS
      OPEN EMPL
      END-EXEC
*
      EXEC ADABAS
      FETCH EMPL
      END-EXEC
*
      PERFORM READ-EMPLOYEES UNTIL ADACODE = 3.
*
      EXEC ADABAS
      CLOSE EMPL
      END-EXEC
*
      EXEC ADABAS
      DBCLOSE
      END-EXEC
*
      STOP RUN.
*
      READ-EMPLOYEES.
      MOVE PERSONNEL-ID TO PERSONNEL-NR.
      MOVE NAME TO LAST-NAME.
      MOVE FIRST-NAME TO F-NAME.
      MOVE BIRTH TO BIRTHDAY.
      MOVE SEX TO KIND.
      DISPLAY LINE1.
      MOVE SPACE TO LINE1.
*
      EXEC ADABAS
      FETCH EMPL
      END-EXEC

```

Example 2

```

IDENTIFICATION DIVISION.
PROGRAM-ID. CEX2.
REMARKS.
* DELETE AN EMPLOYEE RECORD AND RELEASE ALL CARS WHICH ARE
* ASSIGNED TO THIS EMPLOYEE. A PRIVATE CAR WILL BE DELETED
* AND A COMPANY CAR WILL BE MADE A POOL-CAR WHICH IS IDENTIFIED
* BY ITS PERSONNEL-ID CONTAINING ONLY THE COUNTRY CODE.
ENVIRONMENT DIVISION.
DATA DIVISION.

```

```

WORKING-STORAGE SECTION.
*
01  PERSONNEL-NUMBER          PIC X(8)  VALUE '20007100'.
01  EMPLOYEE-ISN             PIC 9(9)  COMP VALUE ZERO.
01  COUNTRY-NUMBER.
    02  COUNTRY-NO           PIC X(1)  VALUE SPACE.
    02  FILLER               PIC X(14) VALUE SPACE.
*
      EXEC ADABAS
BEGIN DECLARE SECTION
      END-EXEC
*
      EXEC ADABAS
READ LOGICAL
DECLARE VEH1 CURSOR FOR
SELECT REG-NUM, PERSONNEL-ID, CLASS
FROM VEHICLES
WHERE PERSONNEL-ID GE :PERSONNEL-NUMBER
OPTIONS HOLD
ORDER BY PERSONNEL-ID
      END-EXEC
*
PROCEDURE DIVISION.
*
*** FIND EMPLOYEE
*
      EXEC ADABAS
FIND
SELECT
FROM EMPLOYEES EMPLOYEES-1
WHERE PERSONNEL-ID = :PERSONNEL-NUMBER
OPTIONS HOLD
      END-EXEC
*
*** IF THE PERSONNEL-ID EXISTS DELETE THE EMPLOYEE AND READ THE
*** VEHICLES FILE
*
      IF QUANTITY OF EMPLOYEES-1 = 1
      MOVE ISN OF EMPLOYEES-1 TO EMPLOYEE-ISN
      PERFORM DELETE-EMPLOYEE
      PERFORM READ-VEHICLES-FILE
      ELSE
      DISPLAY
      'NO EMPLOYEE FOUND WITH PERSONNEL-ID ', PERSONNEL-NUMBER.
*
      EXEC ADABAS
DBCLOSE
      END-EXEC
*
      STOP RUN.
*
DELETE-EMPLOYEE.
      EXEC ADABAS
DELETE
FROM EMPLOYEES
WHERE ISN = :EMPLOYEE-ISN
      END-EXEC
*
      DISPLAY 'EMPLOYEE ', PERSONNEL-NUMBER, ' HAS BEEN DELETED'.
*
      READ-VEHICLES-FILE.

```

```

      EXEC ADABAS
OPEN VEH1
      END-EXEC
*
      EXEC ADABAS
      FETCH VEH1
      END-EXEC
*
      PERFORM READ-VEHICLES UNTIL ADACODE = 3 OR
          PERSONNEL-ID OF VEHICLES > PERSONNEL-NUMBER.
*
      EXEC ADABAS
      CLOSE VEH1
      END-EXEC
*
      EXEC ADABAS
      COMMIT WORK
      END-EXEC
*
      READ-VEHICLES.
      IF CLASS = 'P'
          PERFORM DELETE-PRIVATE-CAR
      ELSE
          PERFORM UPDATE-COMPANY-CAR.
*
      EXEC ADABAS
      FETCH VEH1
      END-EXEC
*
      DELETE-PRIVATE-CAR.
      EXEC ADABAS
      DELETE
      FROM VEHICLES
      WHERE CURRENT OF VEH1
      END-EXEC
      DISPLAY 'PRIVATE CAR ', REG-NUM, ' HAS BEEN DELETED'.
*
      UPDATE-COMPANY-CAR.
      MOVE PERSONNEL-ID OF VEHICLES TO COUNTRY-NUMBER.
      MOVE COUNTRY-NO TO PERSONNEL-ID OF VEHICLES.
*
      EXEC ADABAS
      UPDATE VEHICLES
      WHERE CURRENT OF VEH1
      END-EXEC
      DISPLAY 'COMPANY CAR ', REG-NUM, ' HAS BEEN UPDATED'.

```

Example 3

```

IDENTIFICATION DIVISION.
PROGRAM-ID. CEX3.
REMARKS.
* SALARY INCREASE.
* THIS PROGRAM INCREASES THE SALARY OF EVERY EMPLOYEE BY
* 4 PERCENT.
* THE DEPARTMENT, THE OVERALL AMOUNT OF PAY RISE FOR THE
* DEPARTMENT AND THE PAY RISE FOR ALL DEPARTMENTS WILL BE PRINTED
* OUT.

```

```
* THE PROGRAM IS RESTARTABLE. AFTER AN ABNORMAL TERMINATION THE
* PROGRAM EXECUTION WOULD RESTART WITH THE LAST DEPARTMENT
* WHOSE SALARY UPDATE HAD BEEN COMPLETED BEFORE THE ABEND
* OCCURED.
```

```
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
```

```
*
01 COMMIT-DATA.
   02 COMMIT-DEPARTMENT PIC X(6) VALUE SPACE.
   02 COMMIT-SUM PIC S9(10) COMP-3 VALUE +0.
01 START-DEPT PIC X(6) VALUE SPACE.
01 IND PIC 9(4) COMP VALUE 0.
01 I PIC 9(4) COMP VALUE 0.
01 J PIC 9(4) COMP VALUE 0.
01 NEW-SALARY PIC S9(9) COMP-3 VALUE +0.
01 INCREASE PIC S9(9) COMP-3 VALUE +0.
01 SUM-DEPARTMENT PIC S9(10) COMP-3 VALUE +0.
01 SUM-TOTAL PIC S9(11) COMP-3 VALUE +0.
*
01 HEADER.
   02 FILLER PIC X(10) VALUE 'DEPARTMENT'.
   02 FILLER PIC X(15) VALUE SPACE.
   02 FILLER PIC X(15) VALUE 'SALARY INCREASE'.
01 HEADER2 PIC X(40) VALUE ALL '*'.
01 SPACE-LINE PIC X(50) VALUE SPACE.
01 LINE1.
   02 FILLER PIC X(3) VALUE SPACE.
   02 DEPARTMENT PIC X(6) VALUE SPACE.
   02 FILLER PIC X(16) VALUE SPACE.
   02 SUM-DEPT PIC Z,ZZZ,ZZZ,ZZ9.
01 LAST-LINE.
   02 FILLER PIC X(21) VALUE 'TOTAL SALARY INCREASE'.
   02 FILLER PIC X(3) VALUE ' : '.
   02 TOTAL-SUM-DEPT PIC ZZ,ZZZ,ZZZ,ZZZ.
*
EXEC ADABAS
BEGIN DECLARE SECTION
END-EXEC
*
EXEC ADABAS
HISTOGRAM
DECLARE EMP1 CURSOR FOR
SELECT DEPT
FROM EMPLOYEES EMPLOYEES-1
WHERE DEPT GE :COMMIT-DEPARTMENT
GROUP BY DEPT
END-EXEC
*
EXEC ADABAS
READ LOGICAL
DECLARE EMP2 CURSOR FOR
SELECT PERSONNEL-ID, DEPT, SALARY, INCOME(COUNT)
FROM EMPLOYEES
WHERE DEPT GE :START-DEPT
OPTIONS HOLD
ORDER BY DEPT
END-EXEC
*
```

```

PROCEDURE DIVISION.
*
      EXEC ADABAS
      CONNECT 'INCREASE'
      UPD=EMPLOYEES
      AND USERDATA INTO :COMMIT-DATA
      END-EXEC
*
***  A HISTOGRAM STATEMENT IS USED TO ASCERTAIN THE NUMBER OF
***  EMPLOYEES PER DEPARTMENT
*
      EXEC ADABAS
      OPEN EMP1
      END-EXEC
*
      EXEC ADABAS
      FETCH EMP1
      END-EXEC
*
      IF COMMIT-DATA NOT = ' '
          PERFORM RESTART.
*
      MOVE DEPT OF EMPLOYEES-1 TO START-DEPT.
*
      EXEC ADABAS
      OPEN EMP2
      END-EXEC
*
      DISPLAY HEADER.
      DISPLAY HEADER2.
      DISPLAY SPACE-LINE.
      PERFORM HIST-EMPL UNTIL ADACODE = 3.
*
      EXEC ADABAS
      CLOSE EMP1
      END-EXEC
*
      EXEC ADABAS
      CLOSE EMP2
      END-EXEC
*
      DISPLAY SPACE-LINE.
      MOVE ALL '-' TO SPACE-LINE.
      DISPLAY SPACE-LINE.
      MOVE SPACES TO SPACE-LINE.
      DISPLAY SPACE-LINE.
      MOVE SUM-TOTAL TO TOTAL-SUM-DEPT.
      DISPLAY LAST-LINE.
      MOVE ' ' TO COMMIT-DATA.
*
      EXEC ADABAS
      DBCLOSE
      USERDATA = :COMMIT-DATA
      END-EXEC
*
      STOP RUN.
*
RESTART.
      DISPLAY 'LAST PROGRAM RUN TERMINATED ABNORMALLY'.
      DISPLAY 'LAST DEPARTMENT WAS: ', COMMIT-DEPARTMENT.
*

```

```

      EXEC ADABAS
      FETCH EMP1
      END-EXEC.
*
      HIST-EMPL.
*
***  THE EMPLOYEES FILE WILL BE READ UNTIL ALL RECORDS FOR THE
***  DEPARTMENT HAVE BEEN PROCESSED AND THE SALARY HAS BEEN
***  UPDATED.
*
      PERFORM READ-EMPL VARYING J FROM 1 BY 1 UNTIL
                                J > QUANTITY OF EMPLOYEES-1.
      MOVE DEPT OF EMPLOYEES TO DEPARTMENT.
      MOVE SUM-DEPARTMENT TO SUM-DEPT.
      MOVE ZERO TO SUM-DEPARTMENT.
      DISPLAY LINE1.
      MOVE SPACE TO LINE1.
*
      MOVE DEPT OF EMPLOYEES TO COMMIT-DEPARTMENT.
      MOVE SUM-TOTAL TO COMMIT-SUM.
      EXEC ADABAS
      COMMIT WORK
      USERDATA = :COMMIT-DATA
      END-EXEC
*
      EXEC ADABAS
      FETCH EMP1
      END-EXEC.
*
      READ-EMPL.
      EXEC ADABAS
      FETCH EMP2
      END-EXEC.
*
***  THE SALARY INCREASE CAN BE EXECUTED WHEN THE COUNT OF THE
***  PERIODIC GROUP IS LESS THAN 40.
*
      IF C-INCOME < 40
          PERFORM SALARY-INCREASE
      ELSE
          DISPLAY 'UPDATE PERSON ', PERSONNEL-ID, ' NOT POSSIBLE'.
*
      SALARY-INCREASE.
      COMPUTE INCREASE = SALARY(1) * 0.04.
      COMPUTE NEW-SALARY = SALARY(1) + INCREASE.
      ADD 1 C-INCOME OF EMPLOYEES GIVING IND.
      PERFORM INCREASE-IN-SALARY VARYING I FROM C-INCOME BY -1
                                UNTIL I = 0.
      MOVE NEW-SALARY TO SALARY(1).
*
      EXEC ADABAS

      UPDATE EMPLOYEES
      WHERE CURRENT OF EMP2
      END-EXEC
*
      COMPUTE SUM-DEPARTMENT = SUM-DEPARTMENT + INCREASE.
      COMPUTE SUM-TOTAL = SUM-TOTAL + INCREASE.

```

*

```
INCREASE-IN-SALARY.  
  MOVE SALARY(I) TO SALARY(IND).  
  SUBTRACT 1 FROM IND.
```