

APPENDIX D - ADA EXAMPLES

This chapter covers the following topics:

- Example 1
 - Example 2
 - Example 3
-

Example 1

```
with TYPES, ADABAS_GENERIC_CALLS, TEXT_IO ;
use TYPES, TEXT_IO ;
--
-- 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-VEAR-MAKE >
-- CLASS = 'C'
procedure AEX1 is
  START_MODEL : STRING (1..20) := "MERCEDES-BENZ ";
  START_YEAR_MAKE : STRING (1..2) := "86" ;
  START_MODEL_YEAR_MAKE : STRING(1..22) := START_MODEL &
                                           START_YEAR_MAKE ;

  FILLE1 : STRING(1..20) := " PERSONNEL-ID ";
  FILLE2 : STRING(1..17) := " NAME ";
  FILLE3 : STRING(1..18) := " FIRST-NAME ";
  FILLE4 : STRING(1..6) := "BIRTH ";
  FILLE5 : STRING(1..3) := "SEX" ;
  HEADER : STRING(1..64) := FILLE1 & FILLE2 & FILLE3 & FILLE4
                           & FILLE5 ;
  HEADER2: STRING(1..64) := (1..64 => '*');
  SPACE_LINE : STRING(1..80) := (1..80 => ' ');
  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 "100000001" AND "19999999"
    AND VEHICLES.MODEL-YEAR-MAKE > :START_MODEL_YEAR_MAKE
    AND VEHICLES.CLASS = "C"
  END-EXEC

begin

  EXEC ADABAS
  OPEN EMPL
  END-EXEC
```

```

    PUT_LINE (HEADER) ;
    PUT_LINE (HEADER2) ;
    PUT_LINE (SPACE_LINE) ;

    EXEC ADABAS
    FETCH EMPL
    END-EXEC

while ADACODE /= 3 loop
    PUT_LINE (" " & EMPLOYEES.PERSONNEL_ID & " " & EMPLOYEES.NAME &
              " " & EMPLOYEES.FIRST_NAME & " " & EMPLOYEES.BIRTH & " "
              & EMPLOYEES.SEX ) ;

    EXEC ADABAS
    FETCH EMPL
    END-EXEC

end loop ;

    EXEC ADABAS
    CLOSE EMPL
    END-EXEC

    EXEC ADABAS
    DBCLOSE
    END-EXEC
end AEX1 ;

```

Example 2

```

with TYPES, ADABAS_GENERIC_CALLS, TEXT_IO ;
use  TYPES, TEXT_IO ;
--
--  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.

procedure AEX2 is
    PERSONNEL_NUMBER : STRING(1..8) := "20007100" ;
    EMPLOYEE_ISN : INTEGER := 0 ;

    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

begin
--
--  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 EMPLOYEES_1.QUANTITY = 1 then

        EMPLOYEE_ISN := EMPLOYEES_1.ISN ;
--
--        DELETE EMPLOYEE
--
--        EXEC ADABAS
        DELETE
        FROM EMPLOYEES
        WHERE ISN = :EMPLOYEE_ISN
        END-EXEC
--
--        READ VEHICLES-FILE
--
--        EXEC ADABAS
        OPEN VEH1
        END-EXEC

        EXEC ADABAS
        FETCH VEH1
        END-EXEC

    while ADACODE /= 3 AND
        VEHICLES.PERSONNEL_ID = PERSONNEL_NUMBER loop
        if VEHICLES.CLASS = "P" then
            EXEC ADABAS
            DELETE
            FROM VEHICLES
            WHERE CURRENT OF VEH1
            END-EXEC
            PUT_LINE ("PRIVATE CAR" & VEHICLES.REG_NUM &
                "HAS BEEN DELETED");
        else
            VEHICLES.PERSONNEL_ID := VEHICLES.PERSONNEL_ID (1..1)
                & " " & " ";

            EXEC ADABAS
            UPDATE VEHICLES
            WHERE CURRENT OF VEH1
            END-EXEC
            PUT_LINE ( "COMPANY CAR " & VEHICLES.REG_NUM &
                " HAS BEEN UPDATED" ) ;
        end if ;
        EXEC ADABAS
        FETCH VEH1
        END-EXEC
    end loop ;

    EXEC ADABAS
    CLOSE VEH1
    END-EXEC

```

```

        EXEC ADABAS
        COMMIT WORK
        END-EXEC

else
    PUT_LINE ( "NO EMPLOYEES FOUND WITH PERSONNEL-ID " &
               PERSONNEL_NUMBER ) ;
end if ;
EXEC ADABAS
DBCLOSE
END-EXEC
end AEX2 ;

```

Example 3

```

with TYPES, ADABAS_GENERIC_CALLS, TEXT_IO ;
use TYPES, TEXT_IO ;
-- 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 FROM THE LAST DEPARTMENT
-- WHOSE SALARY UPDATE HAD BEEN COMPLETED BEFORE THE ABEND
-- OCCURED.

```

```

procedure AEX3 is
    type COMMIT_DATA_1 is
        RECORD
            COMMIT_DEPARTMENT : STRING(1..6) := "      " ;
            COMMIT_SUM : INTEGER := 0 ;
            COMMIT_FIL : STRING(1..490) := (1..490 => ' ');
        end record ;
    COMMIT_DATA : COMMIT_DATA_1 ;
    COMMIT_DATA_2 : STRING(1..500);
    for COMMIT_DATA use at COMMIT_DATA_2'ADDRESS;
    START_DEPT : STRING(1..6) := "      " ;
    J : INTEGER := 0 ;
    NEW_SALARY : INTEGER := 0 ;
    INCREASE : INTEGER := 0 ;
    SUM_DEPARTMENT : INTEGER := 0 ;
    SUM_TOTAL : INTEGER := 0 ;
    FILLE1 : STRING(1..10) := " DEPARTMENT" ;
    FILLE2 : STRING(1..15) := (1..15 => ' ') ;
    FILLE3 : STRING(1..15) := "SALARY INCREASE " ;
    HEADER : STRING(1..40) := FILLE1 & FILLE2 & FILLE3 ;
    HEADER2 : STRING(1..40) := (1..40 => '*') ;
    SPACE_LINE : STRING(1..40) := (1..40 => ' ') ;

```

```

        EXEC ADABAS
        BEGIN DECLARE SECTION
        END-EXEC

```

```

        EXEC ADABAS
        HISTOGRAM
        DECLARE EMP1 CURSOR FOR
        SELECT DEPT
        FROM EMPLOYEES EMPLOYEES_1
        WHERE DEPT GE :COMMIT_DATA.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
        ORDER BY DEPT
        OPTIONS HOLD
        END-EXEC

begin

        EXEC ADABAS
        CONNECT 'INCREASE'
        UPD=EMPLOYEES
        AND USERDATA INTO :COMMIT_DATA_2
        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.COMMIT_DEPARTMENT /= "      " then

            PUT_LINE (" LAST PROGRAM RUN TERMINATED ABNORMALLY ") ;
            PUT_LINE (" LAST DEPARTMENT WAS: " &
                COMMIT_DATA.COMMIT_DEPARTMENT) ;

            EXEC ADABAS
            FETCH EMP1
            END-EXEC

        end if ;

        START_DEPT := EMPLOYEES_1.DEPT ;

        EXEC ADABAS
        OPEN EMP2
        END-EXEC

        PUT_LINE(HEADER) ;
        PUT_LINE(HEADER2) ;
        PUT_LINE(SPACE_LINE) ;

        while ADACODE /= 3 loop
--
--  THE EMPLOYEES FILE WILL BE READ UNTIL ALL RECORDS FOR THE
--  DEPARTMENT HAVE BEEN PROCESSED AND THE SALARY HAS BEEN
--  UPDATED.
--
            J := 1 ;

```

```

while J <= EMPLOYEES_1.QUANTITY loop
    EXEC ADABAS
    FETCH EMP2
    END-EXEC
    J := J + 1 ;
--    THE SALARY INCREASE CAN BE EXECUTED WHEN THE COUNT OF THE
--    PERIODIC GROUP IS LESS THAN 40.
    if EMPLOYEES.C_INCOME < 40 then
        INCREASE := (EMPLOYEES.SALARY(1) * 4)/100 ;
        NEW_SALARY := EMPLOYEES.SALARY(1) + INCREASE ;
        EMPLOYEES.SALARY(2..40) := EMPLOYEES.SALARY(1..39) ;
        EMPLOYEES.SALARY(1) := NEW_SALARY ;
        EXEC ADABAS
        UPDATE EMPLOYEES
        WHERE CURRENT OF EMP2
        END-EXEC
        SUM_DEPARTMENT := SUM_DEPARTMENT + INCREASE ;
        SUM_TOTAL := SUM_TOTAL + INCREASE ;
    else
        PUT_LINE("UPDATE PERSON " & EMPLOYEES.PERSONNEL_ID &
            "NOT POSSIBLE") ;
    end if ;
end loop ;
PUT_LINE("    " & EMPLOYEES.DEPT & "                                " &
    INTEGER'IMAGE(SUM_DEPARTMENT)) ;
SUM_DEPARTMENT := 0 ;
COMMIT_DATA.COMMIT_DEPARTMENT := EMPLOYEES.DEPT ;
COMMIT_DATA.COMMIT_SUM := SUM_TOTAL;
    EXEC ADABAS
    COMMIT WORK
    USERDATA = :COMMIT_DATA_2
    END-EXEC

    EXEC ADABAS
    FETCH EMP1
    END-EXEC
end loop ;

    EXEC ADABAS
CLOSE EMP1
    END-EXEC
    EXEC ADABAS
CLOSE EMP2
    END-EXEC
PUT_LINE(SPACE_LINE) ;
SPACE_LINE(1..50) := (1..50 => '-' ) ;
PUT_LINE(SPACE_LINE) ;
SPACE_LINE(1..50) := (1..50 => ' ' ) ;
PUT_LINE(SPACE_LINE) ;
PUT_LINE("TOTAL SALARY INCREASE : " & INTEGER'IMAGE(SUM_TOTAL)) ;
COMMIT_DATA.COMMIT_DEPARTMENT := "    " ;

    EXEC ADABAS
DBCLOSE
USERDATA = :COMMIT_DATA_2
    END-EXEC
end AEX3 ;

```