



Boletim zDeveloper

sample code

```
*****
* COBOL ENTERPRISE/SQL - COMPILAR COM O PRE-COMPILADOR DB2 VRS 8 *
*****
* VRS001 30/05/08 - TULIO L. - IMPLANTACAO. *
*****
```

```
IDENTIFICATION DIVISION.
*****
```

```
PROGRAM-ID. TESTDSN8.
AUTHOR. TULIO LAZARINI.
DATE-WRITTEN. 30/05/2008.
DATE-COMPILED. 30/05/2008.
REMARKS. ESTE PROGRAMA EFETUA FETCHES E INSERTS COM
ROWSETS, TESTANDO A HABILIDADE DO DB2 V8 EM
EXECUTA-LOS.
```

```
*****
ENVIRONMENT DIVISION.
*****
```

```
*****
CONFIGURATION SECTION.
*****
```

```
SOURCE-COMPUTER. IBM-SYSTEM-Z9-109-S54.
OBJECT-COMPUTER. IBM-SYSTEM-Z9-109-S54.
SPECIAL-NAMES.
DECIMAL-POINT IS COMMA.
```

```
*****
DATA DIVISION.
*****
```

```
*****
WORKING-STORAGE SECTION.
*****
```

```
*****
CONTADORES E CONTROLE DO GET DIAGNOSTICS ---*
*****
```

```
01 CEE3ABD-RC PIC S9(09) COMP VALUE 888.
01 CEE3ABD-CLEANUP PIC S9(09) COMP VALUE 1.
01 I PIC S9(04) COMP.
01 J PIC S9(04) COMP.
01 WS-ROWERR PIC S9(09) COMP-3.
01 WS-SQLCODE PIC S9(09) COMP.
01 WS-SQLSTATE PIC X(05).
01 WS-SQLERRD1 PIC S9(09) COMP.
01 WS-SQLERRD2 PIC S9(09) COMP.
01 WS-SQLERRD3 PIC S9(09) COMP.
```

```
*-----*
*--- BOOKS DAS TABELAS ---*
*-----*
```

```
01 WS-AREA-TAB-EMP.
03 WS-EMPNO PIC X(06) OCCURS 20 TIMES.
03 WS-FIRSTNME PIC X(12). OCCURS 20 TIMES.
49 WS-FIRSTNME-LEN PIC S9(04) COMP SYNCHRONIZED.
49 WS-FIRSTNME-TEXT PIC X(01).
03 WS-MIDINIT PIC X(01) OCCURS 20 TIMES.
03 WS-LASTNAME PIC X(01) OCCURS 20 TIMES.
49 WS-LASTNAME-LEN PIC S9(04) COMP SYNCHRONIZED.
49 WS-LASTNAME-TEXT PIC X(15).
03 WS-WORKDEPT PIC X(03) OCCURS 20 TIMES.
03 WS-IWORKDEPT PIC S9(04) COMP OCCURS 20 TIMES.
03 WS-PHONENO PIC X(04) OCCURS 20 TIMES.
03 WS-IPHONENO PIC S9(04) COMP OCCURS 20 TIMES.
03 WS-HIREDATE PIC X(10) OCCURS 20 TIMES.
03 WS-IHIREDATE PIC S9(04) COMP OCCURS 20 TIMES.
03 WS-JOB PIC X(08) OCCURS 20 TIMES.
03 WS-IJOB PIC S9(04) COMP OCCURS 20 TIMES.
03 WS-EDLEVEL PIC S9(04) COMP OCCURS 20 TIMES.
03 WS-IEDLEVEL PIC S9(04) COMP OCCURS 20 TIMES.
03 WS-SEX PIC X(01) OCCURS 20 TIMES.
03 WS-ISEX PIC S9(04) COMP OCCURS 20 TIMES.
03 WS-BIRTHDATE PIC X(10) OCCURS 20 TIMES.
03 WS-IBIRTHDATE PIC S9(04) COMP OCCURS 20 TIMES.
03 WS-SALARY PIC S9(07)V99 COMP-3 OCCURS 20 TIMES.
03 WS-ISALARY PIC S9(04) COMP OCCURS 20 TIMES.
03 WS-BONUS PIC S9(07)V99 COMP-3 OCCURS 20 TIMES.
03 WS-IBONUS PIC S9(04) COMP OCCURS 20 TIMES.
03 WS-COMM PIC S9(07)V99 COMP-3 OCCURS 20 TIMES.
03 WS-ICOMM PIC S9(04) COMP OCCURS 20 TIMES.
```

```
*****
DEFINICAO DE CURSOR COM ROWSET ---*
*****
```

```
EXEC SQL
DECLARE CSRI CURSOR WITH ROWSET POSITIONING FOR
SELECT
EMPNO,
FIRSTNME,
MIDINIT,
LASTNAME,
WORKDEPT,
PHONENO,
HIREDATE,
JOB,
EDLEVEL,
```

```

SEX,
BIRTHDATE,
SALARY,
BONUS,
COMM
FROM
DSNUSER.EMP
END-EXEC.
*
EXEC SQL
INCLUDE SQLCA
END-EXEC.
*
*****
PROCEDURE DIVISION.
*****
*-----*
000000-ROTINA-PRINCIPAL SECTION.
*-----*
*
PERFORM 100000-LEITURA-DADOS.
PERFORM 200000-INSERT.
PERFORM 100000-LEITURA-DADOS.
*
000099-ENCERRA.
STOP RUN.
*
*-----*
100000-LEITURA-DADOS SECTION.
*-----*
*
EXEC SQL
OPEN CSR1
END-EXEC.
*
DISPLAY '<< EXIBICAO DOS DADOS DA TABELA EMP >>'.
DISPLAY ' '.
DISPLAY 'EMPNO '
DISPLAY 'FIRSTNME '
DISPLAY 'MID '
DISPLAY 'LASTNAME '
DISPLAY 'DEP '
DISPLAY 'PHONE '
DISPLAY 'HIREDATE '
DISPLAY 'JOB '
DISPLAY 'EDLV '
DISPLAY 'SEX '
DISPLAY 'BIRTHDATE '
DISPLAY 'SALARY '
DISPLAY 'BONUS '
DISPLAY 'COMM '
DISPLAY '===== '
DISPLAY '===== '
DISPLAY '===== '
*
PERFORM UNTIL SQLCODE EQUAL +100
*
INITIALIZE WS-AREA-TAB-EMP
EXEC SQL
FETCH NEXT ROWSET
FROM CSR1
FOR 20 ROWS

```

```

INTO :WS-EMPNO
:WS-FIRSTNME
:WS-MIDINIT
:WS-LASTNAME
:WS-WORKDEPT :WS-IWORKDEPT
:WS-PHONENO :WS-IPHONENO
:WS-HIREDATE :WS-IHIREDATE
:WS-JOB :WS-IJOB
:WS-EDLEVEL :WS-IEDLEVEL
:WS-SEX :WS-ISEX
:WS-BIRTHDATE :WS-IBIRTHDATE
:WS-SALARY :WS-ISALARY
:WS-BONUS :WS-IBONUS
:WS-COMM :WS-ICOMM
END-EXEC
*
IF (SQLCODE LESS ZERO)
PERFORM 999098-ERRO-DB2
END-IF
*
PERFORM VARYING I FROM 1 BY 1 UNTIL I GREATER SQLERRD(3)
*
DISPLAY WS-EMPNO (I) ' '
DISPLAY WS-FIRSTNME-TEXT(I) ' '
DISPLAY WS-MIDINIT (I) ' '
DISPLAY WS-LASTNAME-TEXT(I) ' '
DISPLAY WS-WORKDEPT (I) ' '
DISPLAY WS-PHONENO (I) ' '
DISPLAY WS-HIREDATE (I) ' '
DISPLAY WS-JOB (I) ' '
DISPLAY WS-EDLEVEL (I) ' '
DISPLAY WS-SEX (I) ' '
DISPLAY WS-BIRTHDATE (I) ' '
DISPLAY WS-SALARY (I) ' '
DISPLAY WS-BONUS (I) ' '
DISPLAY WS-COMM (I) ' '
*
END-PERFORM
*
END-PERFORM.
*
EXEC SQL
CLOSE CSR1
END-EXEC.
*
100099-SAI.
EXIT.
*
*-----*
200000-INSERT SECTION.
*-----*
*
MOVE '000840' TO WS-EMPNO (1).
MOVE 5 TO WS-FIRSTNME-LEN (1).
MOVE 'TULIO' TO WS-FIRSTNME-TEXT(1).
MOVE 'A' TO WS-MIDINIT (1).
MOVE 8 TO WS-LASTNAME-LEN (1).
MOVE 'LAZARINI' TO WS-LASTNAME-TEXT(1).
MOVE 'C01' TO WS-WORKDEPT (1).
MOVE '1558' TO WS-PHONENO (1).
MOVE '30.11.1998' TO WS-HIREDATE (1).
MOVE 'ANALYST' TO WS-JOB (1).
MOVE 16 TO WS-EDLEVEL (1).

```

```

MOVE 'M' TO WS-SEX (1).
MOVE '15.02.1978' TO WS-BIRTHDATE (1).
MOVE 23800 TO WS-SALARY (1).
MOVE 500 TO WS-BONUS (1).
MOVE 1904 TO WS-COMM (1).
*
MOVE '000850' TO WS-EMPNO (2).
MOVE 3 TO WS-FIRSTNME-LEN (2).
MOVE 'ANA' TO WS-FIRSTNME-TEXT (2).
MOVE 'P' TO WS-MIDINIT (2).
MOVE 10 TO WS-LASTNAME-LEN (2).
MOVE 'CASAGRANDE' TO WS-LASTNAME-TEXT (2).
MOVE 'D11' TO WS-WORKDEPT (2).
MOVE '6090' TO WS-PHONENO (2).
MOVE '15.03.2000' TO WS-HIREDATE (2).
MOVE 'DESIGNER' TO WS-JOB (2).
MOVE 17 TO WS-EDLEVEL (2).
MOVE 'F' TO WS-SEX (2).
MOVE '11.03.1980' TO WS-BIRTHDATE (2).
MOVE 21340 TO WS-SALARY (2).
MOVE 500 TO WS-BONUS (2).
MOVE 1707 TO WS-COMM (2).
*
MOVE '000860' TO WS-EMPNO (3).
MOVE 6 TO WS-FIRSTNME-LEN (3).
MOVE 'SILTON' TO WS-FIRSTNME-TEXT (3).
MOVE 'R' TO WS-MIDINIT (3).
MOVE 9 TO WS-LASTNAME-LEN (3).
MOVE 'SCHNEIDER' TO WS-LASTNAME-TEXT (3).
MOVE 'C01' TO WS-WORKDEPT (3).
MOVE '1498' TO WS-PHONENO (3).
MOVE '06.07.1983' TO WS-HIREDATE (3).
MOVE 'ANALYST' TO WS-JOB (3).
MOVE 18 TO WS-EDLEVEL (3).
MOVE 'M' TO WS-SEX (3).
MOVE '16.04.1968' TO WS-BIRTHDATE (3).
MOVE 28420 TO WS-SALARY (3).
MOVE 600 TO WS-BONUS (3).
MOVE 2274 TO WS-COMM (3).
*
DISPLAY '<< INSERCAO DE TRES REGISTROS NA TABELA EMP >>'.
DISPLAY ' '.
*
EXEC SQL
  INSERT INTO DSNUSER.EMP
    VALUES (:WS-EMPNO ,
            :WS-FIRSTNME ,
            :WS-MIDINIT ,
            :WS-LASTNAME ,
            :WS-WORKDEPT ,
            :WS-PHONENO ,
            :WS-HIREDATE ,

```

```

            :WS-JOB ,
            :WS-EDLEVEL ,
            :WS-SEX ,
            :WS-BIRTHDATE ,
            :WS-SALARY ,
            :WS-BONUS ,
            :WS-COMM)
  FOR 3 ROWS NOT ATOMIC CONTINUE ON SQLEXCEPTION
END-EXEC.
*
EXEC SQL
  GET DIAGNOSTICS :J = NUMBER
END-EXEC.
*
PERFORM VARYING I FROM 1 BY 1 UNTIL I GREATER J
  EXEC SQL
    GET DIAGNOSTICS CONDITION :I
      :WS-ROWERR = DB2_ROW_NUMBER ,
      :WS-SQLCODE = DB2_RETURNED_SQLCODE ,
      :WS-SQLSTATE = RETURNED_SQLSTATE ,
      :WS-SQLERRD1 = DB2_SQLERRD1 ,
      :WS-SQLERRD2 = DB2_SQLERRD2 ,
      :WS-SQLERRD3 = DB2_SQLERRD3
  END-EXEC
  IF (WS-SQLCODE NOT EQUAL ZERO)
    DISPLAY 'ERRO NA LINHA ' WS-ROWERR
      , SQLCODE=' WS-SQLCODE
      , SQLSTATE=' WS-SQLSTATE
      , SQLERRD1=' WS-SQLERRD1
      , SQLERRD2=' WS-SQLERRD2
      , SQLERRD3=' WS-SQLERRD3
  END-IF
END-PERFORM.
*
200099-SAI.
EXIT.
*
*-----*
999900-ERROS SECTION.
*-----*
*
999098-ERRO-DB2.
*-----*
DISPLAY '888 ' CTE-PROG ' 098 ERRO NO ACESSO AO DB2'.
DISPLAY '888 ' CTE-PROG ' 098 SQLERRD(3) - ' SQLERRD(3).
DISPLAY '888 ' CTE-PROG ' 098 SQLCODE - ' SQLCODE.
DISPLAY '888 ' CTE-PROG ' 098 SQLSTATE - ' SQLSTATE.
DISPLAY '888 ' CTE-PROG ' 098 SQLERRP - ' SQLERRP.
DISPLAY '888 ' CTE-PROG ' 098 SQLCODE - ' SQLCODE.
CALL 'CEE3ABD' USING CEE3ABD-RC, CEE3ABD-CLEANUP.
*

```