

**2017**

Entrega ejercicio PLSQL

Bases de datos

Andrei García Cuadra

# Implementar un procedimiento que reciba un importe y visualice el desglose del cambio en unidades monetarias de 1, 5, 10, 25, 50, 100, 200, 500, 1000, 2000, 5000 Ptas. en orden inverso al que aparecen aquí enumeradas.

Bloque anónimo:

DECLARE

BEGIN

DESGLOSAR(5573);

END;

Procedimiento:

CREATE OR REPLACE PROCEDURE DESGLOSAR

(

IMPORTE IN NUMBER

)

AS

var\_valorActual NUMBER;

var\_monedas5000 NUMBER;

var\_monedas2000 NUMBER;

var\_monedas1000 NUMBER;

var\_monedas500 NUMBER;

var\_monedas200 NUMBER;

var\_monedas100 NUMBER;

var\_monedas50 NUMBER;

var\_monedas25 NUMBER;

var\_monedas10 NUMBER;

var\_monedas5 NUMBER;

var\_monedas1 NUMBER;

BEGIN

var\_valorActual := IMPORTE;

var\_monedas5000 := 0;

var\_monedas2000 := 0;

var\_monedas1000 := 0;

var\_monedas500 := 0;

var\_monedas200 := 0;

var\_monedas100 := 0;

var\_monedas50 := 0;

var\_monedas25 := 0;

var\_monedas10 := 0;

var\_monedas5 := 0;

var\_monedas1 := 0;

FOR loop\_counter IN REVERSE 0..IMPORTE

LOOP

IF MOD(var\_valorActual,5000) = 0 THEN

var\_valorActual := var\_valorActual-5000;

var\_monedas5000 := var\_monedas5000+1;

ELSE

IF MOD(var\_valorActual,2000) = 0 THEN

var\_valorActual := var\_valorActual-2000;

var\_monedas2000 := var\_monedas2000+1;

ELSE

IF MOD(var\_valorActual,1000) = 0 THEN

var\_valorActual := var\_valorActual-1000;

var\_monedas1000 := var\_monedas1000+1;

ELSE

IF MOD(var\_valorActual,500) = 0 THEN

var\_valorActual := var\_valorActual-500;

var\_monedas500 := var\_monedas500+1;

ELSE

IF MOD(var\_valorActual,200) = 0 THEN

var\_valorActual := var\_valorActual-200;

var\_monedas200 := var\_monedas200+1;

ELSE

IF MOD(var\_valorActual,100) = 0 THEN

var\_valorActual := var\_valorActual-100;

var\_monedas100 := var\_monedas100+1;

ELSE

IF MOD(var\_valorActual,50) = 0 THEN

var\_valorActual := var\_valorActual-50;

var\_monedas50 := var\_monedas50+1;

ELSE

IF MOD(var\_valorActual,25) = 0 THEN

var\_valorActual := var\_valorActual-25;

var\_monedas25 := var\_monedas25+1;

ELSE

IF MOD(var\_valorActual,10) = 0 THEN

var\_valorActual := var\_valorActual-10;

var\_monedas10 := var\_monedas10+1;

ELSE

IF MOD(var\_valorActual,5) = 0 THEN

var\_valorActual := var\_valorActual-5;

var\_monedas5 := var\_monedas5+1;

ELSE

IF MOD(var\_valorActual,1) = 0 THEN

var\_valorActual := var\_valorActual-1;

var\_monedas1 := var\_monedas1+1;

END IF;

END IF;

END IF;

END IF;

END IF;

END IF;

END IF;

END IF;

END IF;

END IF;

END IF;

END LOOP;

DBMS\_OUTPUT.PUT\_LINE('Desglose del pago correspondiente a ' || IMPORTE

|| ' Monedas de 5000 pesetas: ' || var\_monedas5000

|| ' Monedas de 2000 pesetas: ' || var\_monedas2000

|| ' Monedas de 1000 pesetas: ' || var\_monedas1000

|| ' Monedas de 500 pesetas: ' || var\_monedas500

|| ' Monedas de 200 pesetas: ' || var\_monedas200

|| ' Monedas de 100 pesetas: ' || var\_monedas100

|| ' Monedas de 50 pesetas: ' || var\_monedas50

|| ' Monedas de 25 pesetas: ' || var\_monedas25

|| ' Monedas de 10 pesetas: ' || var\_monedas10

|| ' Monedas de 5 pesetas: ' || var\_monedas5

|| ' Monedas de 1 pesetas: ' || var\_monedas1

);

END DESGLOSAR;

1. Codificar un procedimiento que permita borrar un empleado cuyo número se pasará en la llamada.

Bloque anónimo:

DECLARE

BEGIN

BORRAREMPLEADO(5573);

END;

Procedimiento:

CREATE OR REPLACE PROCEDURE BORRAREMPLEADO

(

IDEMPLEADO IN NUMBER

) AS

var\_nombreEmpleado VARCHAR2(255);

BEGIN

SELECT ename into var\_nombreEmpleado FROM employees WHERE empno=IDEMPLEADO;

DELETE FROM employees WHERE empno=IDEMPLEADO;

DBMS\_OUTPUT.PUT\_LINE('Se eliminó correctamente al empleado' || var\_nombreEmpleado);

1. Escribir un procedimiento que modifique la localidad de un departamento. El procedimiento recibirá como parámetros el número del departamento y la localidad nueva.

Bloque anónimo:

DECLARE

BEGIN

MODIFICARLOCALIDADDPTO(2, ‘Leganés’);

END;

Procedimiento:

CREATE OR REPLACE PROCEDURE MODIFICARLOCALIDADDPTO

(

DNO IN NUMBER

, NUEVALOCALIDAD IN VARCHAR2

) AS

var\_localidadAntigua VARCHAR2(255);

BEGIN

SELECT dname into var\_localidadAntigua FROM departments WHERE deptno=DNO;

UPDATE departments SET location=NUEVALOCALIDAD WHERE deptno=DNO;

DBMS\_OUTPUT.PUT\_LINE('Se modificó correctamente del departamento ' || DNO || '. La antigua localización era ' || var\_localidadAntigua || ' y ahora es ' || NUEVALOCALIDAD);

END MODIFICARLOCALIDADDPTO;

1. Crear un procedimiento que reciba como parámetros el número de un empleado y, por referencia, se le pase un parámetro de tipo localidad del departamento del empleado. En la llamada ese parámetro irá sin valor. El procedimiento recogerá la localidad del departamento si la tuviere. En un bloque anónimo, llamaremos al procedimiento y evaluaremos dicha localidad, si tiene valor, lo sacaremos por pantalla, si no tiene nada, diremos que está vacío.

Bloque anónimo:

DECLARE

LOCALIDADVACIA VARCHAR2(255);

BEGIN

SELECT location into LOCALIDADVACIA FROM departments WHERE deptno=1;

EVALUARLOCALIDAD(5573,'Estoy vacio');

END;

Procedimiento:

CREATE OR REPLACE PROCEDURE EVALUARLOCALIDAD

(

ENO IN NUMBER

, LOCALIDADVACIA OUT VARCHAR2

) AS

BEGIN

IF LOCALIDADVACIA IS NOT NULL THEN

DBMS\_OUTPUT.PUT\_LINE('La localidad tiene el valor ' || LOCALIDADVACIA);

ELSE

DBMS\_OUTPUT.PUT\_LINE('La localidad no tiene el valor.');

END IF;

END EVALUARLOCALIDAD;

DBMS\_OUTPUT.PUT\_LINE('Se modificó correctamente del departamento ' || DNO || '. La antigua localización era ' || var\_localidadAntigua || ' y ahora es ' || NUEVALOCALIDAD);

END MODIFICARLOCALIDADDPTO;