

TEMA8

PL/SQL. PROCEDIMIENTOS Y FUNCIONES

EJERCICIOS SOLUCIONADOS

Bases de Datos
CFGs DAW

Raquel Torres
raquel.torres@ceedcv.es
Versión:180507.1412

UD08. PL/SQL.PROCEDIMIENTOS Y FUNCIONES

1. EJERCICIOS

1.1 Ejercicio 1

Crea un procedimiento almacenado llamado *calcula_salario* que obtenga el salario del número de empleado pasado como primer parámetro de entrada y si el salario obtenido es menor o igual que un valor máximo (segundo parámetro de entrada-salida de este procedimiento) se le suma 100.

Además escribe un ejemplo de ejecución de este procedimiento desde un bloque PL-SQL.

```
CREATE OR REPLACE PROCEDURE calcula_salario (emple IN NUMBER, vmax IN OUT
NUMBER)
IS
    b NUMBER:=0;
BEGIN
    SELECT salario INTO b FROM empleado WHERE num_emp=emple;
    IF b <= vmax THEN
        b:=b+100;
        -- Otra interpretación podría ser que cambia el salario en la BD, en ese caso
        -- UPDATE empleado SET SALARIO=b WHERE num_emple=emple;
    END IF;
    vmax:=b; --porque vmax es variable de entrada-salida
END;
/

--Para ejecutar el procedimiento desde un bloque:

SET SERVEROUTPUT ON
DECLARE
    salario NUMBER:=500;
BEGIN
    dbms_output.put_line ('El salario máximo es ' || salario);
    calcula_salario(7369, salario);
    dbms_output.put_line ('El salario final del empleado 7369 es ' || salario);END;
/
```

1.2 Ejercicio 2

Crea un procedimiento almacenado llamado *revision_salario* que almacene en la siguiente tabla *Salario_Demasiado_Alto* los números de los empleados que tienen un salario muy alto.

Para ello, crea primero la siguiente tabla:

```
CREATE TABLE SALARIO_DEMASIADO_ALTO (NUM_EMP NUMBER (4,0));
```

El procedimiento debe tener como parámetros el nombre y el número del empleado a consultar, y deberá obtener el salario actual del mismo. En el caso que este salario sea mayor que 3000 euros se insertará el numero del empleado en la tabla SALARIO_DEMASIADO_ALTO y si es menor o igual a 3000 se le subirá un 20% el salario.

Escribe un ejemplo de ejecución de este procedimiento desde un bloque PL-SQL donde compares el salario inicial del empleado con el salario final y sólo muestres por pantalla el nombre y número de los empleados que no le subes el salario explicando que tienen el mismo salario porque es demasiado alto.

```
CREATE OR REPLACE PROCEDURE revision_salario (emple IN VARCHAR2, n_emp IN
NUMBER, sal_emp OUT NUMBER)
IS
BEGIN
    SELECT SALARIO INTO sal_emp
    FROM EMPLEADO WHERE NOMBRE = emple AND num_emp=n_emp;
    IF sal_emp > 3000 THEN
        INSERT INTO SALARIO_DEMASIADO_ALTO (num_emp)
        VALUES(n_emp);
    ELSE
        sal_emp:=sal_emp*1.2;
        UPDATE EMPLEADO SET SALARIO=sal_emp WHERE
        NOMBRE = emple AND num_emp=n_emp;
    END IF;
END;
/

--Para ejecutar el procedimiento desde un bloque:

SET ServerOutput ON;
DECLARE
    salario_ini NUMBER(7,2);
    nombre_emp VARCHAR2(10);
    num_empleado NUMBER(4);
    salario_final NUMBER(7,2);
BEGIN
    nombre_emp := 'MARTINEZ';
    num_empleado := 7782;
```

```
SELECT salario, nombre, num_emp INTO salario_ini, nombre_emp,
num_empleado
FROM empleado
WHERE nombre = nombre_emp AND num_emp= num_empleado;
dbms_output.put_line('El salario inicial es: ' || salario_ini);
revision_salario (nombre_emp, num_empleado, salario_final);
IF salario_ini = salario_final THEN
    dbms_output.put_line ('El salario del empleado con número ' ||
num_empleado || ' con nombre ' || nombre_emp || ' permanece igual a '
|| salari_ini || ' porque es un salario demasiado alto');
ELSE
    dbms_output.put_line ('El salario del empleado con número ' ||
num_empleado || ' con nombre ' || nombre_emp || ' después de la subida es
' || salario_final);
END IF;
END;
/
```

1.3 Ejercicio 3

Crea una función llamada `GET_TOTAL_EMPLEADOS` que devuelva el número total de empleados existentes y escribe un ejemplo de una llamada a la función desde un bloque PL/SQL.

```
CREATE FUNCTION GET_TOTAL_EMPLEADOS
RETURN NUMBER
IS
    V_TOTAL NUMBER:=0;
BEGIN
    SELECT COUNT(*) INTO V_TOTAL FROM EMPLEADO;
RETURN V_TOTAL;
END;
/

--Llamada a una función:

SET SERVEROUTPUT ON;
BEGIN
```

```
DBMS_OUTPUT.PUT_LINE('El numero total de empleados : ' ||  
GET_TOTAL_EMPLEADOS());  
END;
```

1.4 Ejercicio 4

Crea una función que tome como entrada el nombre del departamento y devuelva cuántos empleados pertenecen a él.

Luego crea un ejemplo de ejecución de dicha función dentro de un bloque PL-SQL en el que muestres por pantalla el número de empleados del departamento de ventas.

```
CREATE FUNCTION GET_EMPLEADOS_DPT (pnombre_dpt VARCHAR2)  
RETURN NUMBER  
IS  
    V_TOTAL_DPT NUMBER:=0;  
BEGIN  
    SELECT COUNT(*) INTO V_TOTAL_DPT FROM EMPLEADO E, DPTO D  
    WHERE D.NUM_DPTO=E.NUM_DPTO AND  
    D.NOMBRE_DPTO=pnombre_dpt;  
    RETURN V_TOTAL_DPT;  
END;  
/  
  
--Llamada a una función:  
  
SET SERVEROUTPUT ON;  
BEGIN  
    DBMS_OUTPUT.PUT_LINE('El departamento de Ventas tiene ' ||  
    GET_EMPLEADOS_DPT('VENTAS') || ' empleados');  
END;  
/
```