****

**Práctica Bases de Datos Avanzadas**

**Consultas Avanzadas**

Breidy Núñez | 2017-5633

Felipe Ramírez | 2018-6104

Luis Franco | 2018-6342

**Prof.**  Leandro Fondeur

18/7/2019

I. Realice los siguientes ejercicios luego de estudiar el documento D64254GC11\_les05.ppt:

5.1 Construya un código de PL/SQL donde se inserte un nuevo registro en la tabla de empleados. El ID del empleado debe llenarse usando la secuencia creada para los fines. Genere una secuencia y asigne el valor directamente a una variable, tal como lo permite Oracle 11g. Recuerde asentar los cambios a la BD dentro del bloque de código.

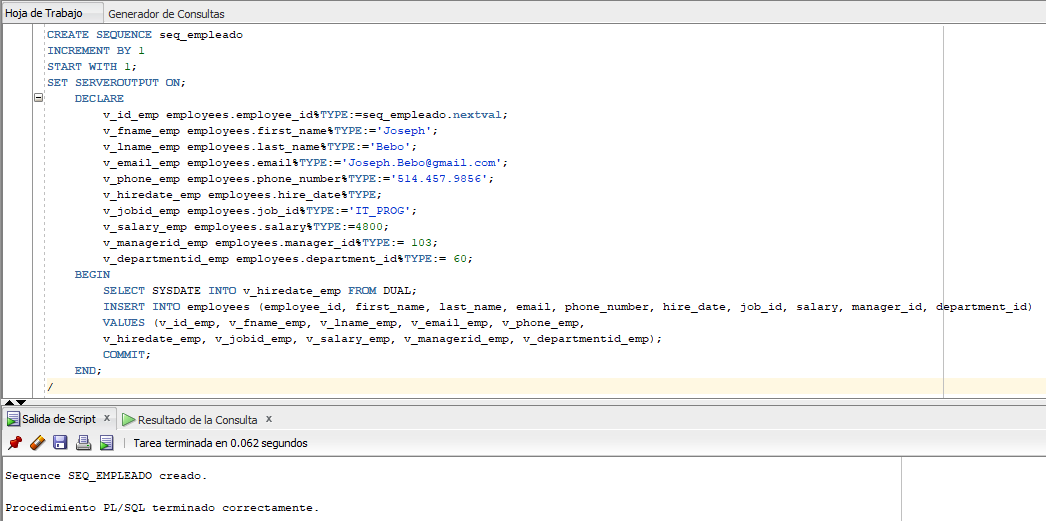
ID del empleado: Secuencia // Nombre: Joseph // Apellido: Bebo

Email: Joseph.Bebo@gmail.com // Teléfono: 514-457-9856

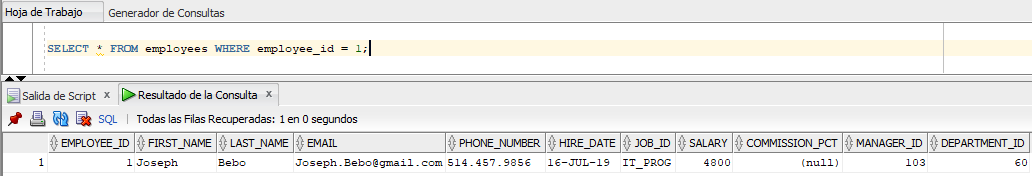
Fecha Ingreso: Fecha Actual (use la función) // Puesto: IT\_PROG

Salario: 4800 // Comisión: Ignorar // ID del Supervisor: 103

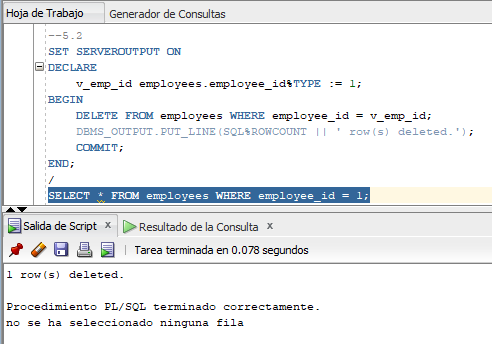
ID del Departamento: 60



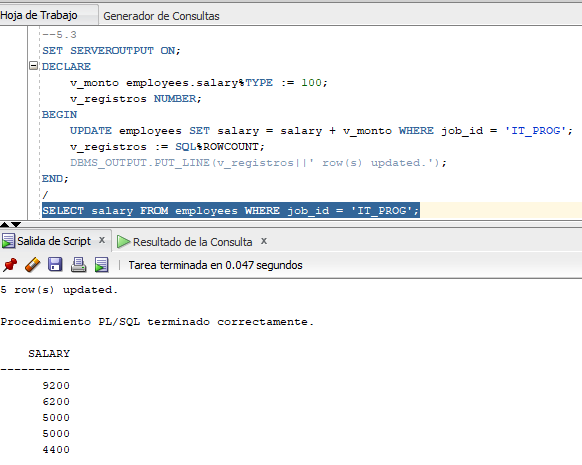
Fuera del bloque realice una consulta para mostrar el registro insertado.



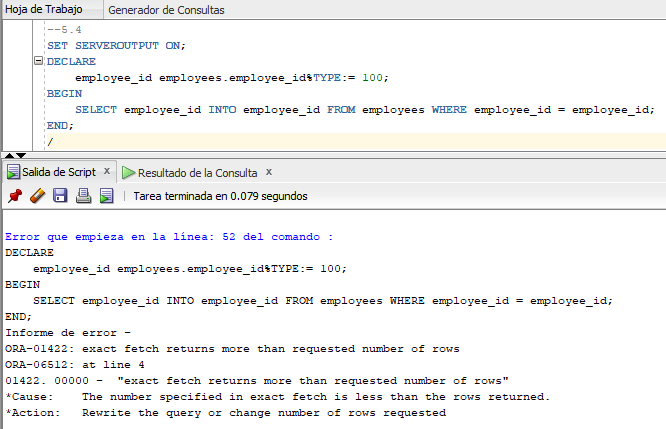
5.2 Construya un código PL/SQL que le pase un valor a una variable en la sección declarativa y que en el bloque del programa use dicho valor para eliminar un registro basado en el ID del empleado. Puede usar como valor el ID del empleado creado en el ejercicio 5.1. Muestre por pantalla la cantidad de registros borrados usando el atributo de cursor SQL correspondiente. Recuerde asentar los cambios. Fuera del bloque PL/SQL haga una consulta para validar que el empleado se haya borrado exitosamente.



5.3 Construya un código PL/SQL que asigne un monto a una variable y actualice los salarios sumando a los mismos el valor introducido en la variable, de todos los empleados que trabajan en la posición IT\_PROG. Asigne la cantidad de registros actualizados a una variable y muestre por pantalla el valor de dicha variable.



5.4 Construya un código PL/SQL que asigne un valor a una variable y realice una consulta por ID del empleado usando dicha variable. Declare la variable con el mismo nombre de la columna de ID del empleado. Diga cuál es el resultado de la corrida del programa y explique lo que ocurra.



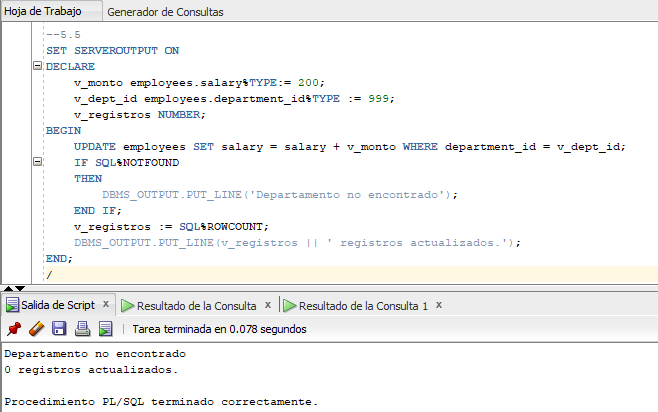
*El resultado del SELECT retorna más de una fila. Esto es debido a que en la cláusula WHERE existe una ambigüedad con el nombre de la variable y el nombre de la columna correspondiente a la tabla de la que se busca extraer el resultado. (Además, en PL/SQL las instrucciones INTO son obligatorias, ya que las instrucciones SELECT en PL/SQL se utilizan para introducir valores dentro de variables de acuerdo a un resultado arrojado por la consulta).*

5.5 Construya un código PL/SQL que pida un monto y un ID de departamento, y actualice los salarios sumando a los mismos el valor introducido, de todos los empleados que trabajan en el ID de departamento dado. Asigne la cantidad de registros actualizados a una variable. Use SQL%NOTFOUND para validar si haciendo la actualización no se llegó a actualizar ningún registro. Si no actualizó ningún registro, muestre el siguiente mensaje:

Departamento no encontrado

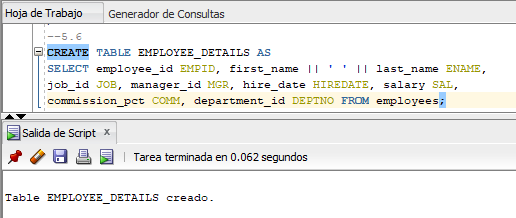
En cualquier escenario muestre el siguiente mensaje por pantalla:

<<cantidad de registros>> registros actualizados



5.6 Construya un código PL/SQL que lea información de una tabla y si los registros existen en otra tabla que actualice los valores en ésta, de lo contrario que inserte el registro. Para lograr el objetivo, realice las siguientes actividades:

1) Crear la tabla EMPLOYEE\_DETAILS usando un sub-query de la tabla EMPLOYEES, seleccionando el ID del empleado (EMPID), el nombre concatenado al apellido con un espacio de por medio (ENAME), el ID de la posición (JOB), el ID del supervisor (MGR), la fecha de contratación (HIREDATE), el salario (SAL), la comisión (COMM) y el ID del departamento (DEPTNO).



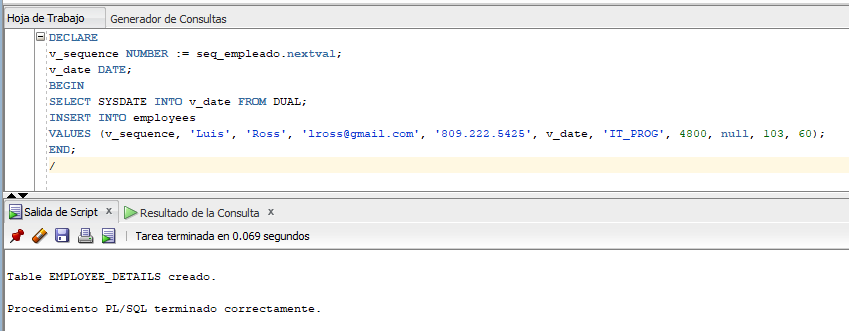
2) Insertar un nuevo registro en la tabla de empleados con los siguientes valores:

ID del empleado [usar la secuencia], Nombre [Luis], Apellido [Ross],

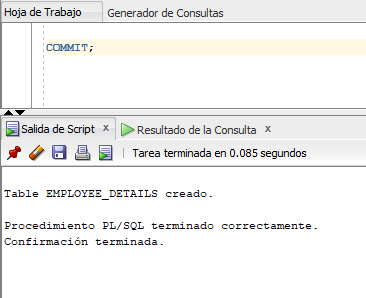
Correo [lross@gmail.com], Teléfono [809-222-5425], Fecha [fecha del sistema],

Posición [IT\_PROG], Salario [4800], Comisión [NULL], ID del supervisor [103],

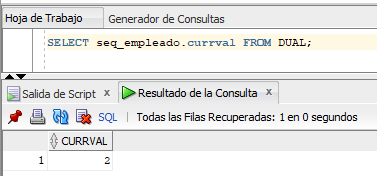
ID del departamento [60]



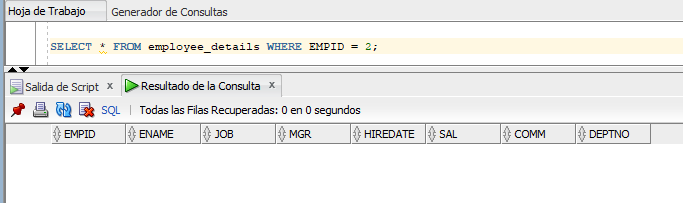
3) Asentar los cambios en la BD.



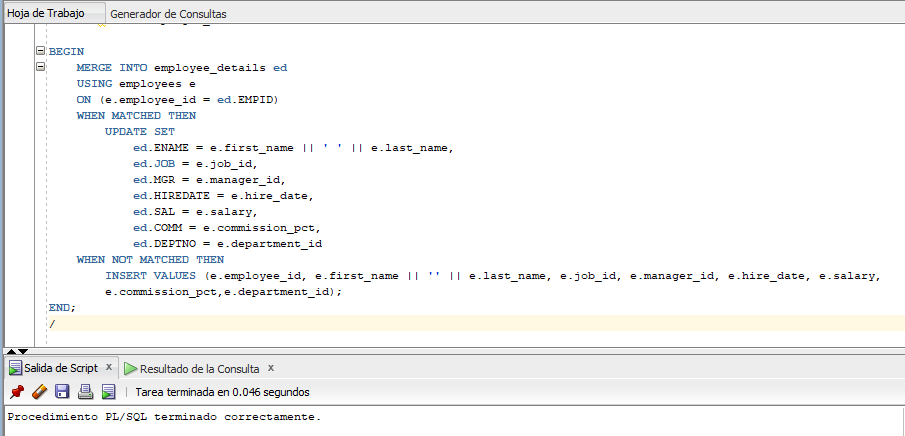
4) Revisar la secuencia actual que se usó para insertar el valor en el ID del empleado.



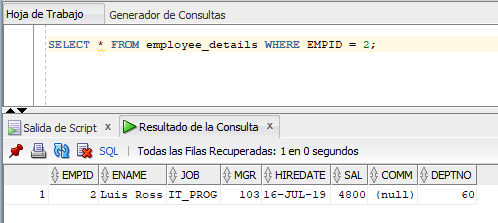
5) Revisar si existe un registro en la tabla EMPLOYEE\_DETAILS con el ID del empleado igual al número de secuencia retornado en el paso 4.



6) Construir un código PL/SQL que permita actualizar la tabla EMPLOYEE\_DETAILS con los nuevos cambios realizados en la tabla EMPLOYEES, mediante un MERGE.



7) Revisar nuevamente si existe un registro en la tabla EMPLOYEE\_DETAILS con el ID del empleado igual al número de secuencia retornado en el paso 4.

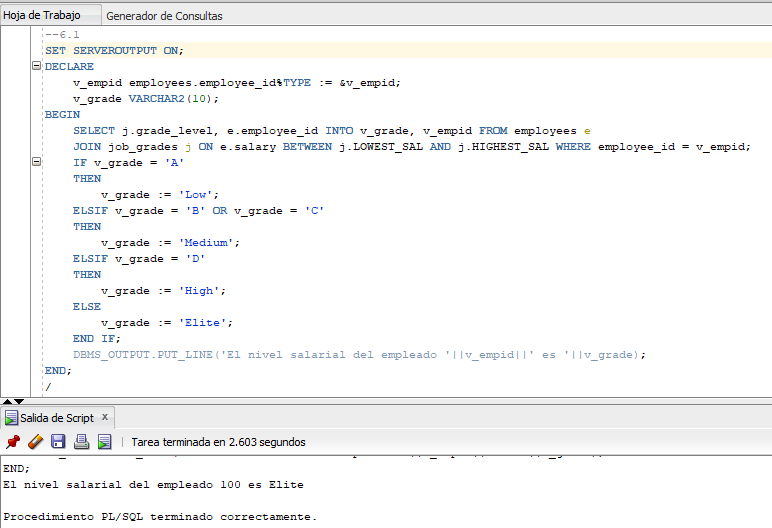


II. Realice los siguientes ejercicios luego de estudiar el documento D64254GC11\_les06.ppt:

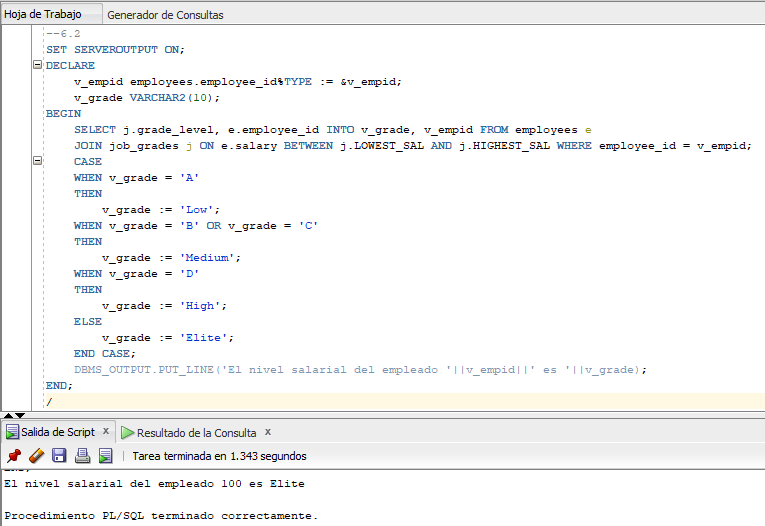
6.1 Construya un código PL/SQL que lea los datos (ID del empleado y nivel salarial) de un empleado que el usuario introduzca por pantalla. Relacione sus datos con la tabla JOB\_GRADES para conocer su nivel salarial. Si el nivel del empleado es 'A' se desea desplegar "Low", si es 'B' o 'C' que despliegue 'Medium', si es 'D' que despliegue 'High' y si no es ninguno de estos niveles, que despliegue 'Elite'. Muestre por pantalla

El nivel salarial del empleado <<ID del empleado>> es <<nivel salarial>>

Use la estructura IF.

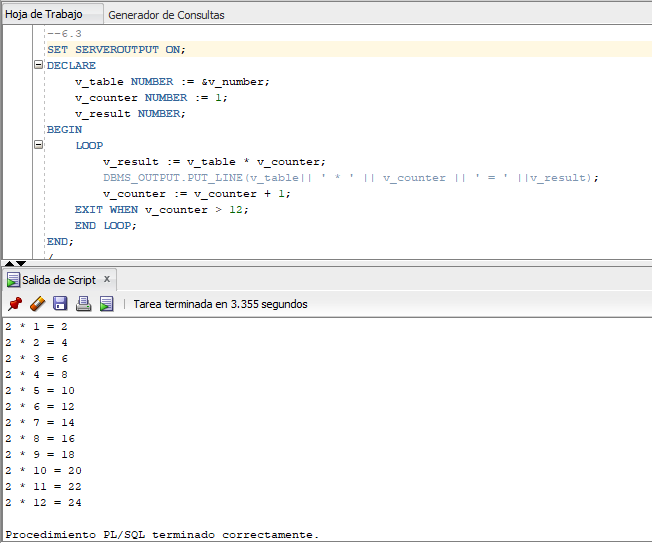


6.2 Reconstruya el código del ejercicio 6.1 y use la estructura CASE.

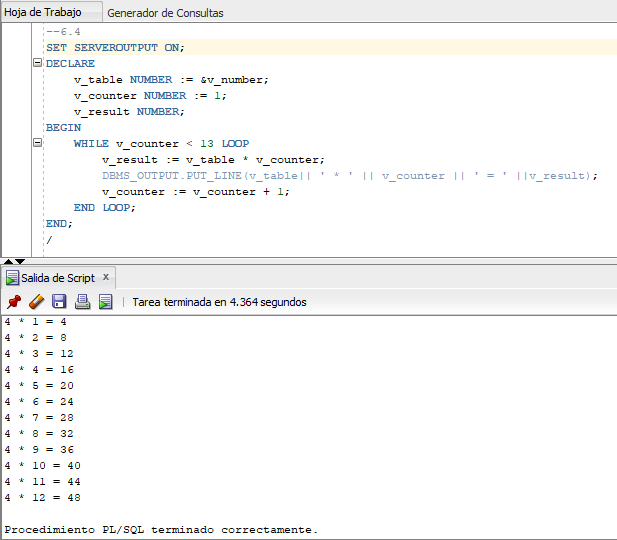


6.3 Construya un código PL/SQL para mostrar la tabla de multiplicar seleccionada por el usuario. Cuando el contador llegue a 12 salir del ciclo. Use un ciclo simple. Muestre en pantalla la información como:

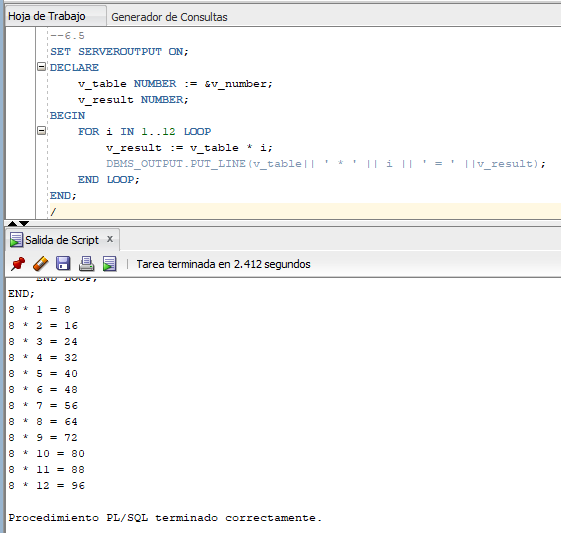
tabla \* contador = resultado



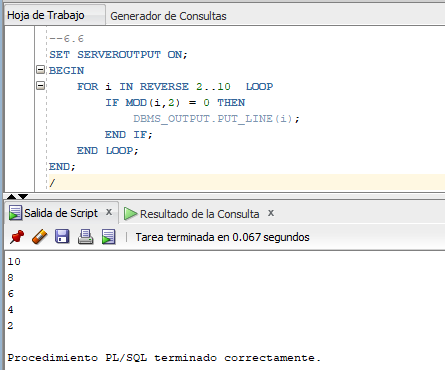
6.4 Reconstruya el código del ejercicio 6.3 usando el ciclo WHILE.



6.5 Reconstruya el código del ejercicio 6.3 usando el ciclo FOR.



6.6 Construya un código PL/SQL que muestre los números pares del 2 al 10 en reversa. Use el ciclo FOR.



6.7 Construya un código PL/SQL que utilizando ciclos FOR anidados genere las siguientes parejas de números:

0 1

1 1

2 1

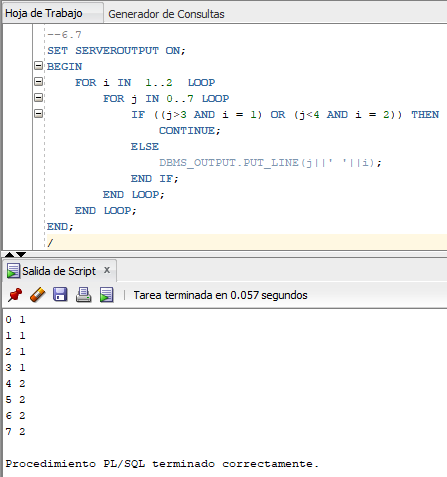
3 1

4 2

5 2

6 2

7 2



6.8 Construya un código PL/SQL que busque en la tabla de empleados el salario de un empleado introducido por el usuario. Crear un ciclo que dé cinco (5) iteraciones. Durante cada iteración se quiere que el salario se divida entre dos (2) y se muestre en pantalla el resultado. En otra sección del ciclo se desea mostrar en pantalla el número de iteración, pero sólo mientras el resultado de dividir el salario entre dos (2) sea >= al salario entre seis (6). Para este último requerimiento use CONTINUE.

