**Universidad Tecnológica de Panamá**

**Facultad de Ingeniería de Sistemas Computacionales**

**Licenciatura en Ingeniería de Sistemas de Información**

**Departamento de Sistemas de Información**

Sistemas de Bases de Datos II

Equipo #2

Laboratorio #4: “Implementación Bloques anónimos”

Facilitador:

Ing. Henry Lezcano

Integrantes grupo 2:

Karen Cabrera 20-14-5403

Medardo Logreira 8-879-1679

Sebastian Zamora 20-15-4392

Georlenys Martinez 8-926-2256

Grupo:

1IF131

II Semestre, 2020

**UNIVERSIDAD TECNOLOGICA DE PANAMA**

**FACULTAD DE INGENIERIA DE SISTEMAS COMPUTACIONES**

**LICENCIATURA EN INGENIERIA DE SISTEMAS DE INFORMACION**

**SISTEMAS DE BASE DE DATOS II**

**LABORATORIO NO.4**

1. Desarrolle un bloque anónimo que capture (&captura) el nombre de una ciudad española y mande a línea de comando el nombre del equipo que representa la ciudad. El ejercicio será para 3 ciudades. Utilice la estructura del CASE por la estructura de control IF-THEN-ELSE. No se permiten las mismas ciudades por equipo.

**--primer problema**

declare

v\_ciudad varchar2(20);

v\_equipo varchar2(30);

v\_estadio varchar2(30);

BEGIN

CASE

v\_ciudad := &capture;

WHEN v\_ciudad= 'Vitoria' THEN v\_equipo := 'Deportivo Alavés';

dbms\_output.put\_line('El equipo de esta ciudad es: '|| v\_equipo);

WHEN v\_ciudad= 'Vigo' THEN v\_equipo := 'R. C. Celta de Vigo';

dbms\_output.put\_line('El equipo de esta ciudad es: '|| v\_equipo);

WHEN v\_ciudad ='Bilbao' THEN v\_equipo := 'Athletic Club';

dbms\_output.put\_line('El equipo de esta ciudad es: '|| v\_equipo);´

WHEN v\_ciudad= 'Huesca' THEN v\_equipo := 'S. D. Huesca';

dbms\_output.put\_line('El equipo de esta ciudad es: '|| v\_equipo);

WHEN v\_ciudad= 'Barcelona' THEN v\_equipo := 'F. C. Barcelona';

dbms\_output.put\_line('El equipo de esta ciudad es: '|| v\_equipo);

WHEN v\_ciudad ='Villarreal' THEN v\_equipo := 'Villarreal C. F.';

dbms\_output.put\_line('El equipo de esta ciudad es: '|| v\_equipo);´

WHEN v\_ciudad= 'Éibar' THEN v\_equipo := 'S. D. Eibar';

dbms\_output.put\_line('El equipo de esta ciudad es: '|| v\_equipo);

WHEN v\_ciudad= 'Getafe' THEN v\_equipo := 'Getafe C. F.';

dbms\_output.put\_line('El equipo de esta ciudad es: '|| v\_equipo);

WHEN v\_ciudad ='Valladolid' THEN v\_equipo := 'Real Valladolid';

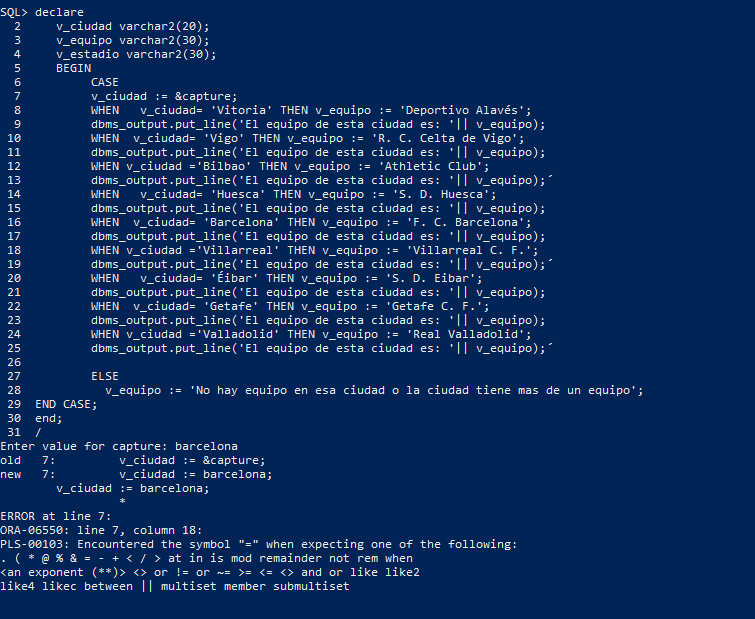
dbms\_output.put\_line('El equipo de esta ciudad es: '|| v\_equipo);´

ELSE

v\_equipo := 'No hay equipo en esa ciudad o la ciudad tiene mas de un equipo';

END CASE;

end;

/

1. Desarrolle un bloque anónimo que cargue en una relación o tabla de base de datos llamada estudiante con el número de estudiante, cedula, nombre y calificación final. Luego que realice una consulta a esta tabla de estudiante para mostrar en la línea de comando el nombre del estudiante con la calificación final obtenida.

/\*segundo punto\*/

--creando la tabla

create table Estudiantes (

num\_estudiante number(3) not null,

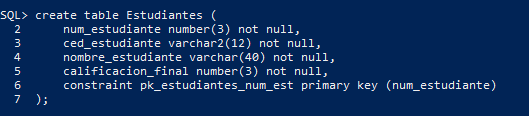
ced\_estudiante varchar2(12) not null,

nombre\_estudiante varchar(40) not null,

calificacion\_final number(3) not null,

constraint pk\_estudiantes\_num\_est primary key (num\_estudiante)

);



**--insertando valores**

INSERT INTO ESTUDIANTES VALUES (001,'8-950-240',‘Luisa Martin’, 91);

INSERT INTO ESTUDIANTES VALUES (002,'20-345-100',‘Sebastian Zamora’, 67);

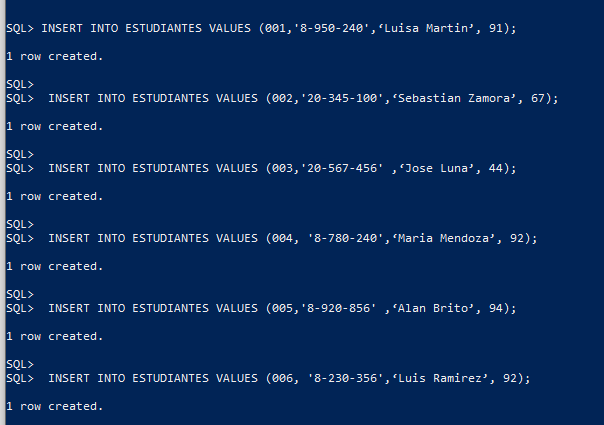
INSERT INTO ESTUDIANTES VALUES (003,'20-567-456' ,‘Jose Luna’, 44);

INSERT INTO ESTUDIANTES VALUES (004, '8-780-240',‘Maria Mendoza’, 92);

INSERT INTO ESTUDIANTES VALUES (005,'8-920-856' ,‘Alan Brito’, 94);

INSERT INTO ESTUDIANTES VALUES (006, '8-230-356',‘Luis Ramirez’, 92);

-



**--Levantando el servidor**

set serveroutput on;



DECLARE -- Declaración de las variables para la tabla

v\_salida\_nombre\_est Estudiantes.nombre\_estudiante%TYPE;

v\_salida\_calif\_final Estudiantes.calificacion\_final%TYPE;

BEGIN --Ahora se insertando las variables a la tabla

FOR v\_counter in 1..6 LOOP

select nombre\_estudiante, calificacion\_final into v\_salida\_nombre\_est, v\_salida\_calif\_final

from Estudiantes

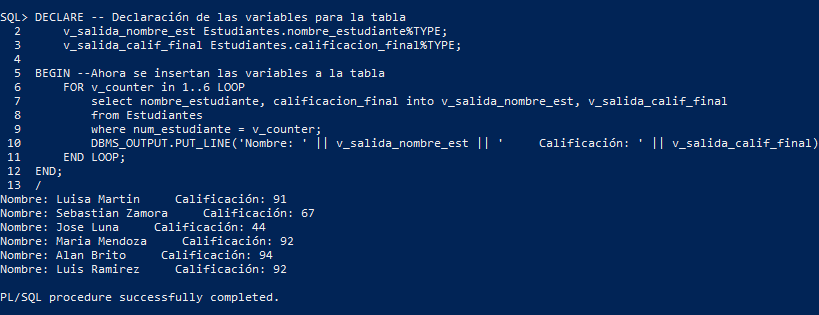
where num\_estudiante = v\_counter;

DBMS\_OUTPUT.PUT\_LINE('Nombre: ' || v\_salida\_nombre\_est || ' Calificación: ' || v\_salida\_calif\_final);-- imprimiendo la tabla

END LOOP;

END;

/



1. Desarrolle un bloque anónimo que capture un numero entero y determine si este número es primo o no lo es, adicionalmente muestre el resultado en la línea de comando.

-- tercer problema

set serveroutput on; --levantamos el servidor

DECLARE-- declaramos las variables

primo char(2) := 'si';

numero integer;

BEGIN

numero := to\_number('&captureunnumero');-- capturamos el numero

dbms\_output.put\_line('El numero ingresado por el usuario es: '|| numero);

for i in 2..numero-1

loop

if mod(numero,i)=0 then

primo := 'no';

end if;

end loop;

if primo='si' then

dbms\_output.put\_line('el numero es primo ');

else

dbms\_output.put\_line('el numero no es primo');

end if;

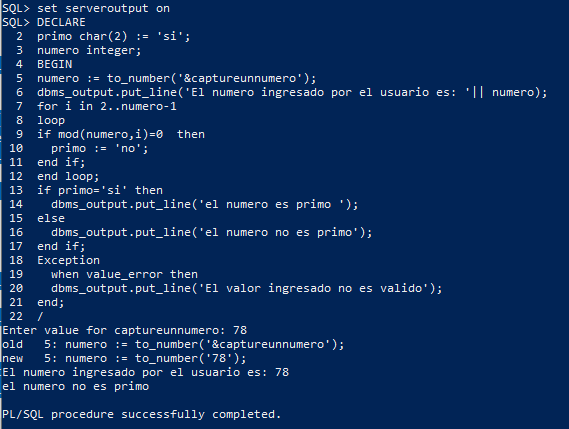
Exception -- creamos una excepcion

when value\_error then

dbms\_output.put\_line('El valor ingresado no es valido');

end;

/



1. Desarrolle un bloque anónimo que implemente un proceso de repetición para almacenar en una relación de base de datos llamada cumpleaños la identificación que corresponde al contador que controla el ciclo de repetición, nombre y día de cumpleaños de 5 estudiantes de su grupo. Luego un bloque adicional que me permita capturar la identificación y haga una consulta a la relación cumpleaños para conocer el nombre y el día de cumpleaños en línea de comando.

--CREANDO LA TABLA

Create table Cumple(

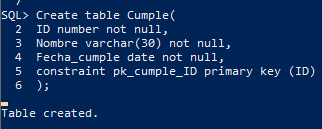
ID number not null,

Nombre varchar(30) not null,

Fecha\_cumple date not null,

constraint pk\_cumple\_ID primary key (ID)

);



--creamos el ciclo que almacene la informacion

DECLARE

V\_ID number;

Nombre Cumple.Nombre%TYPE;

Fecha\_cumple Cumple.Fecha\_cumple%TYPE;

BEGIN

FOR V\_ID IN 1..5 LOOP

IF V\_ID = 1 THEN

Nombre :='&Nombre';

Fecha\_cumple :=to\_date('&fecha','DD/MM/YYYY');

INSERT INTO Cumple VALUES(V\_ID,Nombre,Fecha\_cumple);

ELSIF V\_ID = 2 THEN

Nombre :='&Nombre';

Fecha\_cumple :=to\_date('&fecha','DD/MM/YYYY');

INSERT INTO Cumple VALUES(V\_ID,Nombre,Fecha\_cumple);

ELSIF V\_ID = 3 THEN

Nombre :='&Nombre';

Fecha\_cumple :=to\_date('&fecha','DD/MM/YYYY');

INSERT INTO Cumple VALUES(V\_ID,Nombre,Fecha\_cumple);

ELSIF V\_ID = 4 THEN

Nombre :='&Nombre';

Fecha\_cumple :=to\_date('&fecha','DD/MM/YYYY');

INSERT INTO Cumple VALUES(V\_ID,Nombre,Fecha\_cumple);

ELSE

Nombre :='&Nombre';

Fecha\_cumple :=to\_date('&fecha','DD/MM/YYYY');

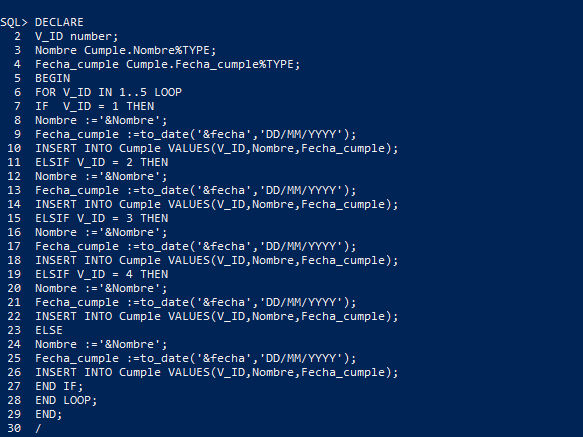
INSERT INTO Cumple VALUES(V\_ID,Nombre,Fecha\_cumple);

END IF;

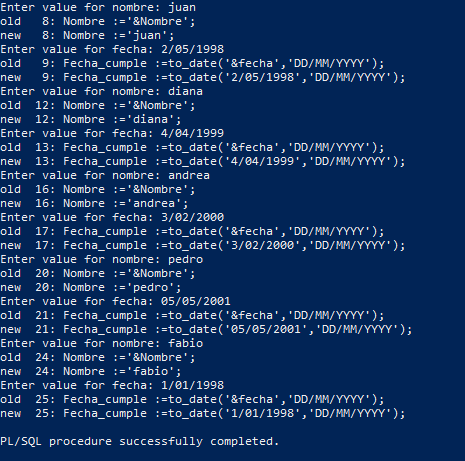
END LOOP;

END;

/



--almacenamos datos de prueba –



**--segundo bloque- bloque de consulta**

DECLARE

v\_ID Cumple.ID%TYPE;

v\_Nombre Cumple.Nombre%TYPE;

v\_Fecha\_cumple Cumple.Fecha\_cumple%TYPE;

BEGIN

v\_ID := &v\_ID;

Select Nombre,Fecha\_cumple INTO v\_Nombre,v\_Fecha\_cumple

From cumple

Where ID=v\_ID;

DBMS\_OUTPUT.PUT\_LINE('Nombre: ' || v\_Nombre || ' Fecha de cumpleaños:' || v\_Fecha\_cumple);

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('Estudiante no existe en la BD');

END;

