Universidad Tecnológica de Panamá

Facultad de Ingeniería de Sistemas Computacionales

Licenciatura en Ingeniería de Sistemas de Información

Departamento de Sistemas de Información, Control y Evaluación de Recursos Informáticos

Sistemas de Bases de Datos II

Equipo #1

Laboratorio #4: “Implementación de Bloques PL/SQL”

Facilitador:

Ing. Henry Lezcano

Integrantes:

Aguilar, Milagros 3-740-771

Atencio, Anel 8-950-868

Márquez, Paola 8-949-1108

Rojas, Reynaldo 8-950-792

Grupo:

1IF131

II Semestre, 2020

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.

DECLARE

-- Declaración de las variables

v\_ciudad varchar(25);

BEGIN

-- Sentencia para captura

v\_ciudad:= '&n\_ciudad';

dbms\_output.put\_line('Nombre del equipo:');

-- Inicio del case

CASE v\_ciudad

when 'Valencia' then

dbms\_output.put\_line('Valencia CF');

when 'Madrid' then

dbms\_output.put\_line('Atlético de Madrid');

when 'Sevilla' then

dbms\_output.put\_line('Real Betis');

else

dbms\_output.put\_line('Ciudad no válida.');

END CASE;

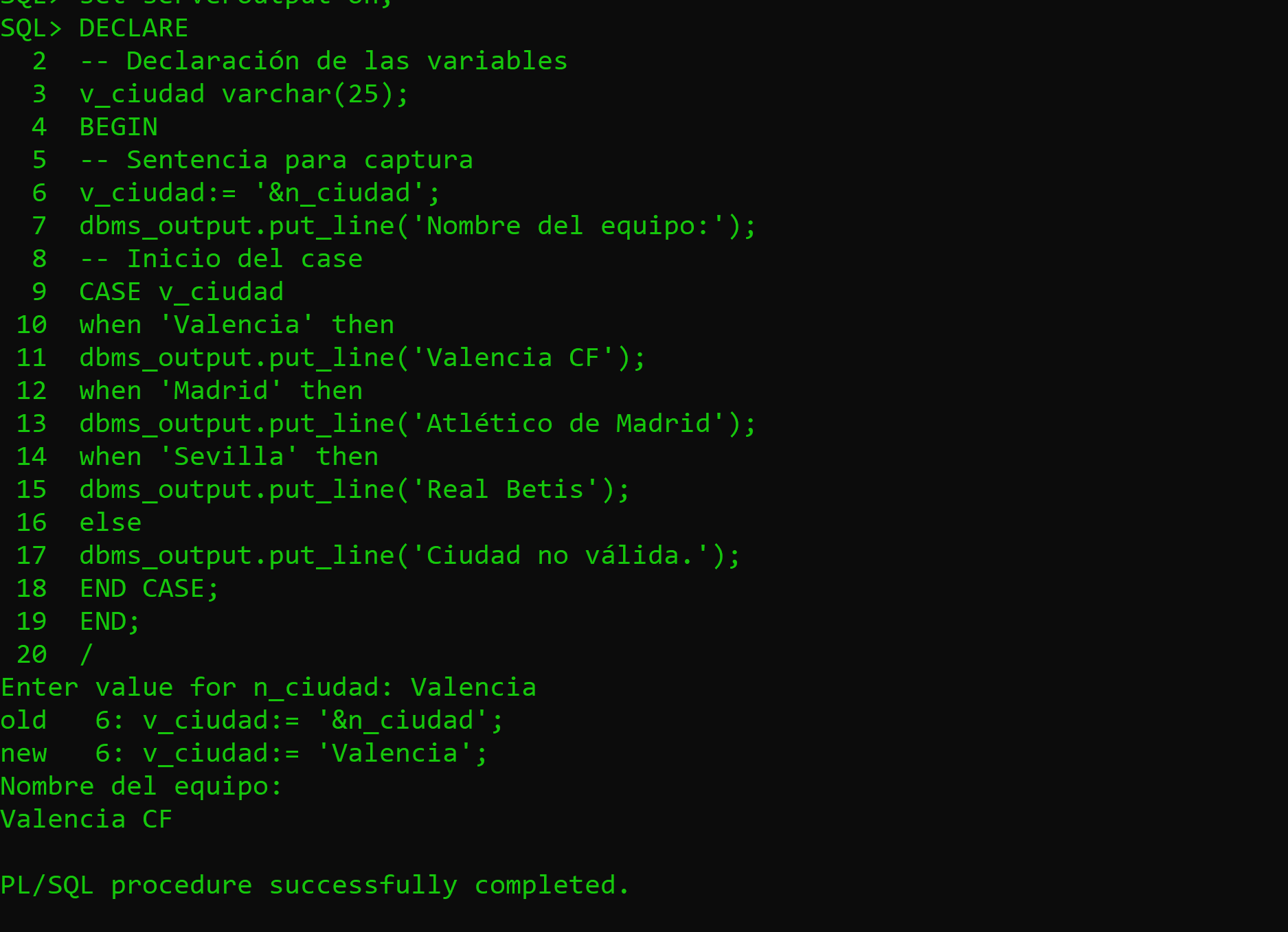
EXCEPTION

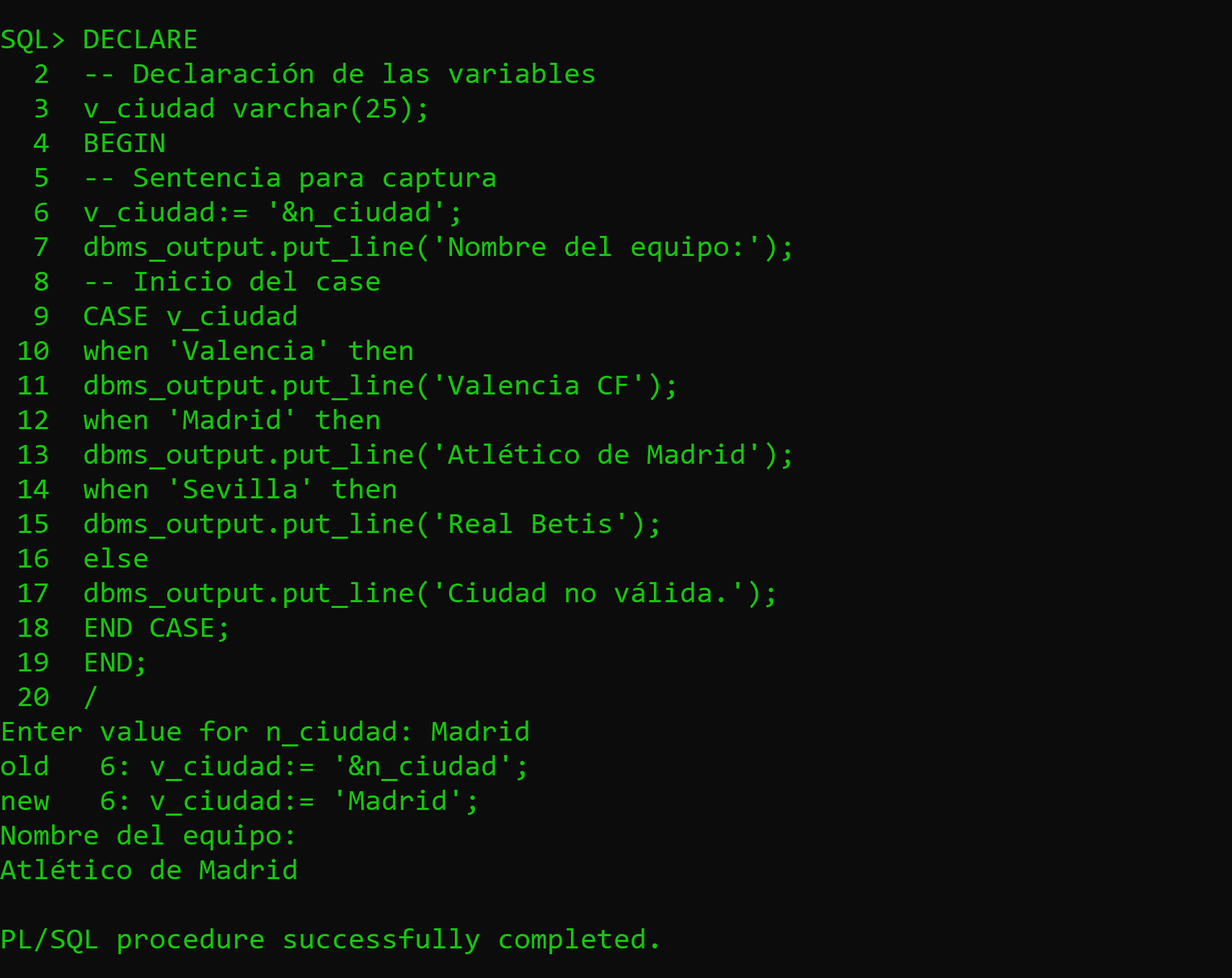
when no\_data\_found then

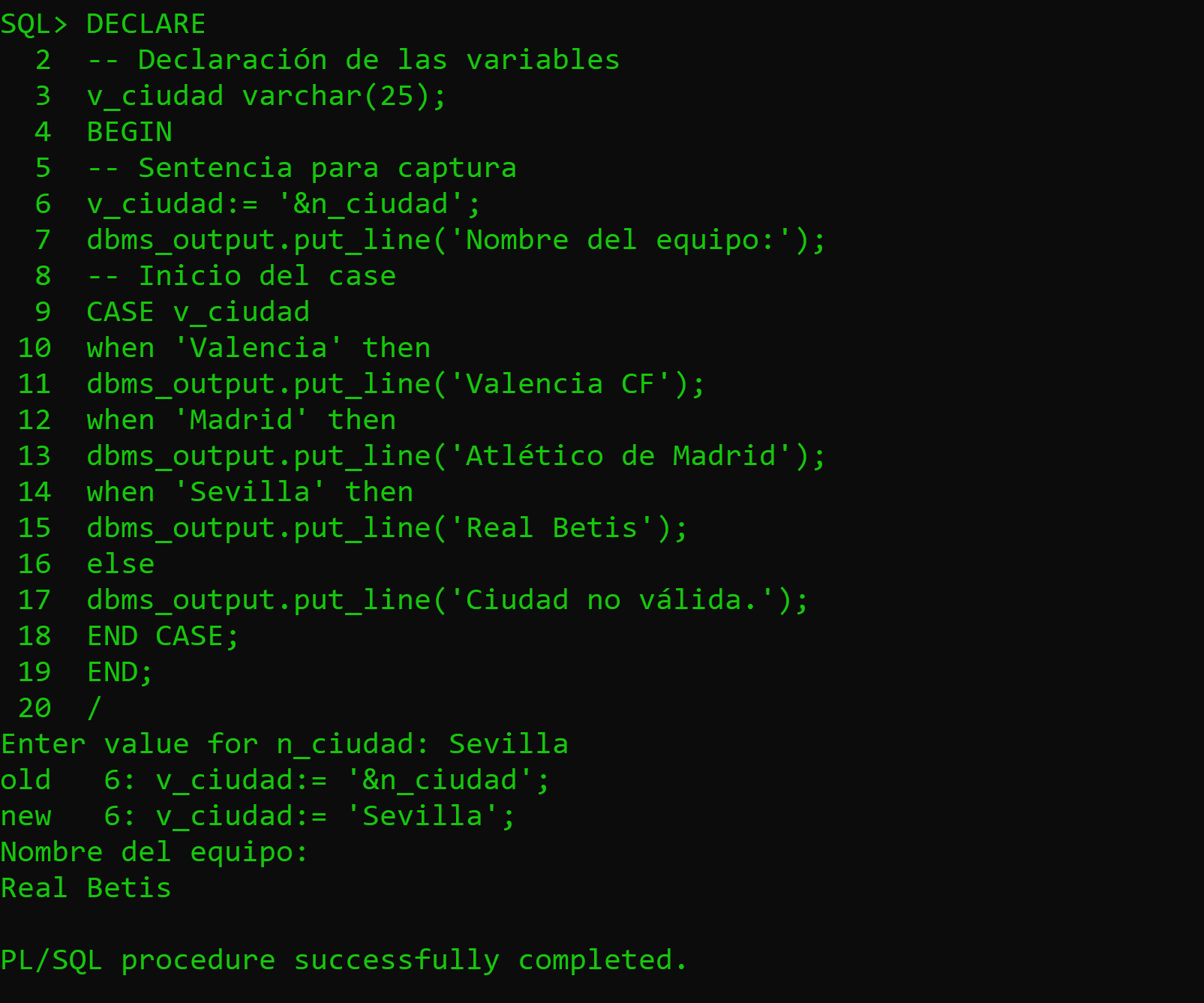
dbms\_output.put\_line('Error. No se insertó ningún dato');

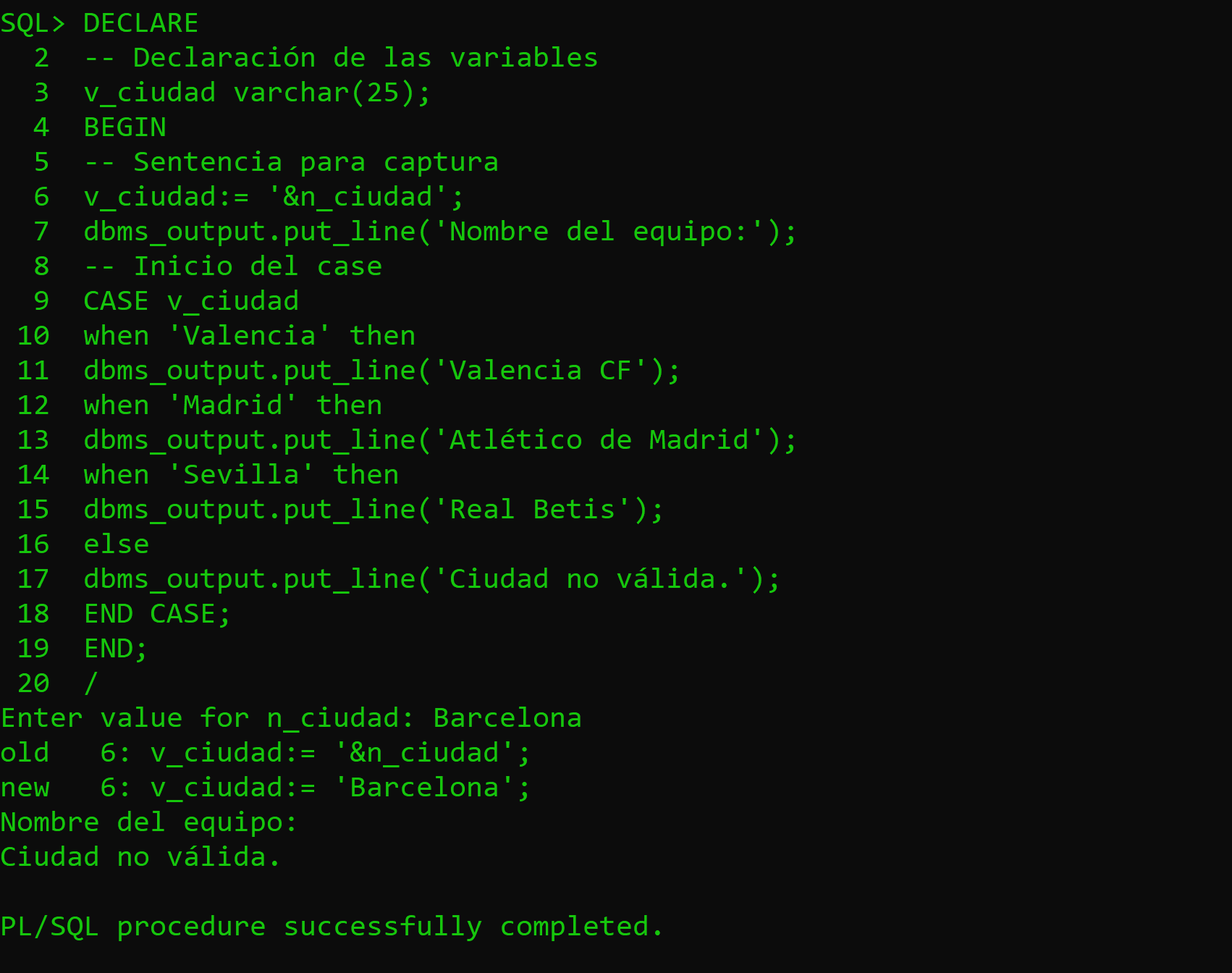
END;

**PROGRAMA EN EJECUCIÓN**









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.

create table Estudiantes (

num\_est number(5) not null,

ced\_est varchar2(12) not null,

nombre\_est varchar(50) not null,

calif\_final number(3) not null,

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

);

insert into Estudiantes values (00001, '8-950-100', 'Marcos Gonzalez', 61);

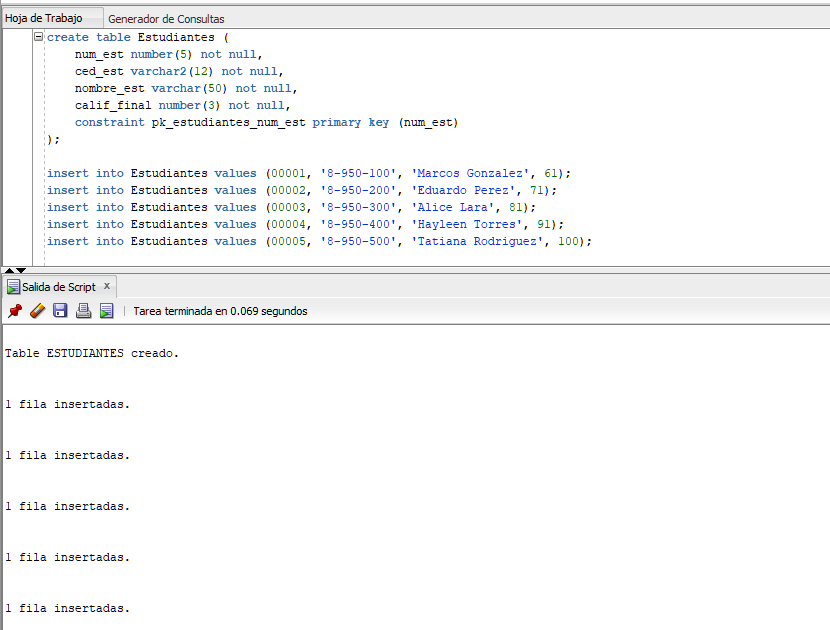
insert into Estudiantes values (00002, '8-950-200', 'Eduardo Perez', 71);

insert into Estudiantes values (00003, '8-950-300', 'Alice Lara', 81);

insert into Estudiantes values (00004, '8-950-400', 'Hayleen Torres', 91);

insert into Estudiantes values (00005, '8-950-500', 'Tatiana Rodriguez', 100);

**CREACIÓN DE LA TABLA**



set serveroutput on;

DECLARE -- Declaración de las variables para la tabla

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

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

BEGIN --Ahora se insertan las variables a la tabla

FOR v\_counter in 1..5 LOOP

select nombre\_est, calif\_final into v\_salida\_nombre\_est, v\_salida\_calif\_final

from Estudiantes

where num\_est = v\_counter;

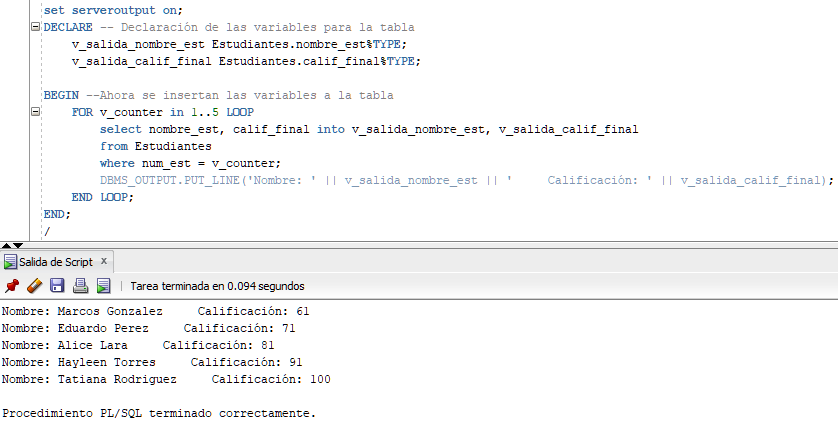
DBMS\_OUTPUT.PUT\_LINE('Nombre: ' || v\_salida\_nombre\_est || ' Calificación: ' || v\_salida\_calif\_final);

END LOOP;

END;

/

**PROGRAMA EN EJECUCIÓN**



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.

DECLARE

V\_Num number;

Div number :=0;

Negative EXCEPTION;

BEGIN

V\_num := '&Numb';

IF V\_Num <=0 THEN

RAISE Negative;

END IF;

FOR i IN 1..V\_Num LOOP

IF mod (V\_Num, i)=0 THEN

Div:=Div+1;

END IF;

END LOOP;

IF Div = 2 THEN

DBMS\_OUTPUT.PUT\_LINE ('Es un numero primo');

ELSE

DBMS\_OUTPUT.PUT\_LINE ('No es un número primo');

END IF;

EXCEPTION

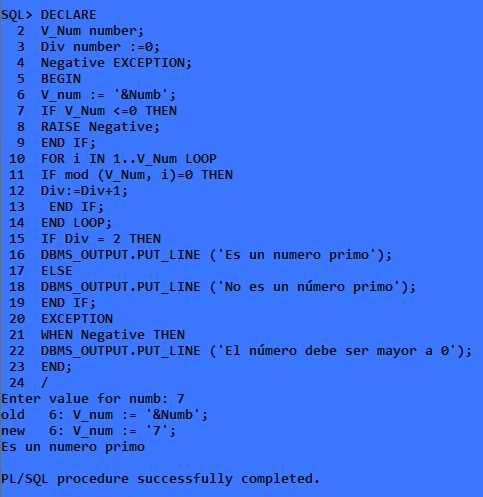
WHEN Negative THEN

DBMS\_OUTPUT.PUT\_LINE ('El número debe ser mayor a 0');

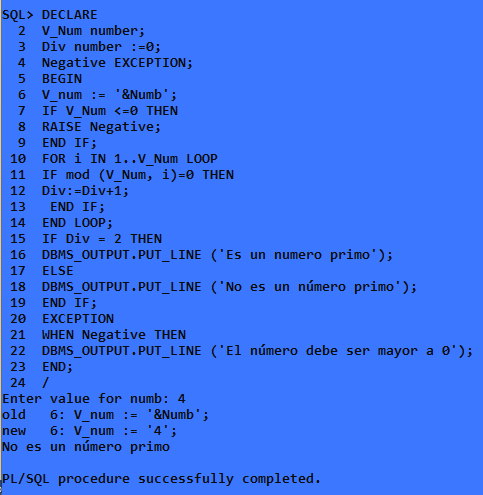
END;

/

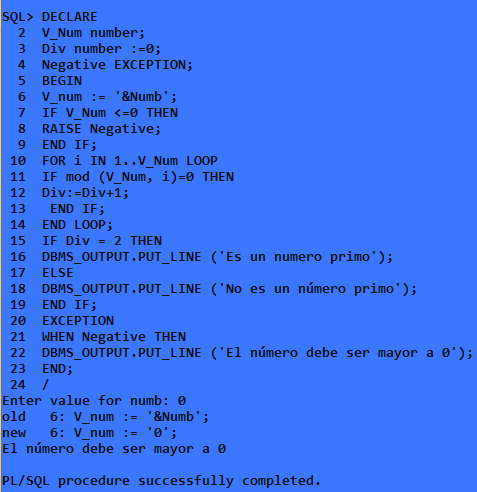
**PROGRAMA EJECUTÁNDOSE CON NÚMERO PRIMO**



**PROGRAMA EJECUTÁNDOSE CON NÚMERO QUE NO ES PRIMO**



**EXCEPTION DEL PROGRAMA**



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.

**CREACÍON DE LA TABLA QUE VAMOS A UTILIZAR**

SQL> Create table cumpleanos (

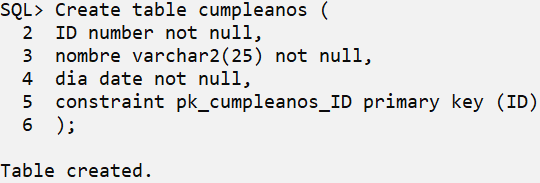
2 ID number not null,

3 nombre varchar2(25) not null,

4 dia date not null,

5 constraint pk\_cumpleanos\_ID primary key (ID)

6 );



**CÓDIGO DEL PRIMER BLOQUE ANÓNIMO**

Declare

IDC number;

nombre cumpleanos.nombre%TYPE;

dia cumpleanos.dia%TYPE;

Begin

FOR IDC in 1..5 LOOP

IF IDC = 1 THEN

nombre :='&Nombre';

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

INSERT INTO cumpleanos VALUES(IDC,nombre,dia);

ELSIF IDC = 2 THEN

nombre :='&Nombre';

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

INSERT INTO cumpleanos VALUES(IDC,nombre,dia);

ELSIF IDC = 3 THEN

nombre :='&Nombre';

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

INSERT INTO cumpleanos VALUES(IDC,nombre,dia);

ELSIF IDC = 4 THEN

nombre :='&Nombre';

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

INSERT INTO cumpleanos VALUES(IDC,nombre,dia);

ELSIF IDC = 5 THEN

nombre :='&Nombre';

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

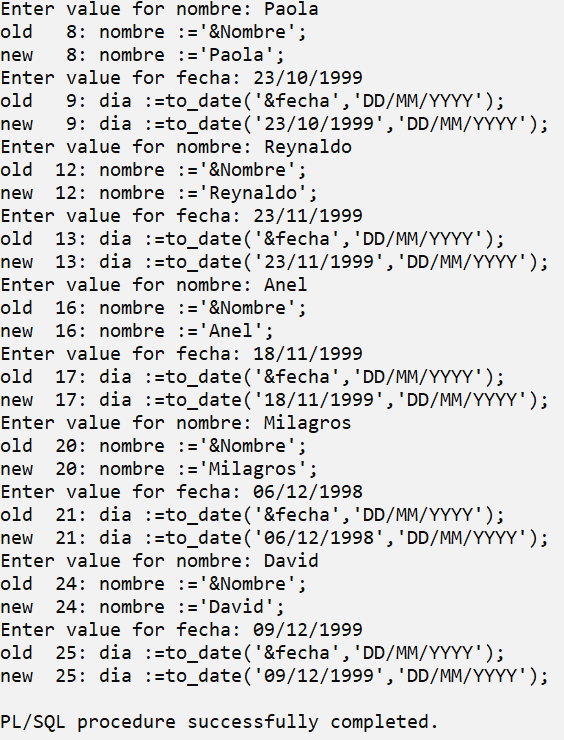
INSERT INTO cumpleanos VALUES(IDC,nombre,dia);

END IF;

END LOOP;

END;

**PROGRAMA EJECUTÁNDOSE**



**CÓDIGO DEL SEGUNDO BLOQUE ANÓNIMO**

DECLARE

IDC cumpleanos.ID%TYPE;

nombre1 cumpleanos.nombre%TYPE;

dia1 cumpleanos.dia%TYPE;

BEGIN

IDC := &IDC;

Select nombre,dia INTO nombre1, dia1

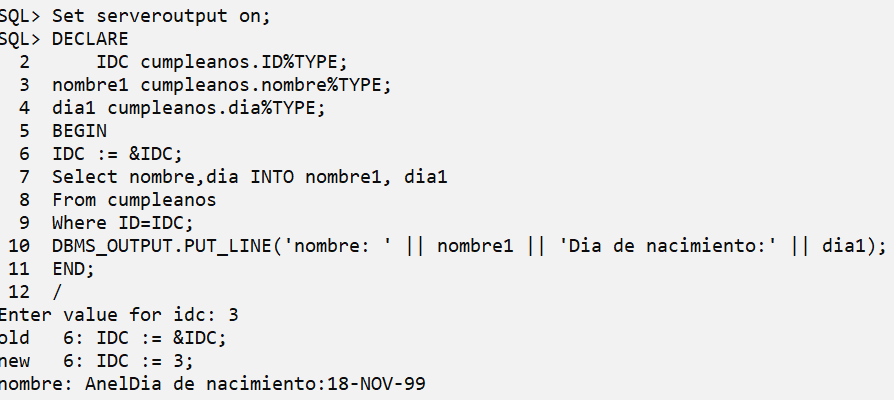
From cumpleanos

Where ID=IDC;

DBMS\_OUTPUT.PUT\_LINE('nombre: ' || nombre1 || 'Dia de nacimiento: ' || dia1);

END;

**CÓDIGO EJECUTÁNDOSE**



***No sabemos que sucedía anteriormente que no imprimía así que colocamos el comando que se ubica antes del ‘Declare’.***