

## EJERCICIOS PL/SQL

1.- Codificar un bloque anónimo que visualice el empleado (nombre, apellido sueldo) que tenga el sueldo mayor. Realizar todas las comprobaciones posibles, es decir, si hay más de un empleado con ese sueldo se visualizará un mensaje “Hay xx empleados con sueldo yy”

```
set serveroutput on
declare
nombre emple.nombre%type;
apellido emple.apellido%type;
varSalario emple.salario%type;
maxSalario emple.salario%type;
n int:=0;
begin
select max(salario) into maxSalario from emple;
select count(*) into n from emple where salario = maxSalario;
if n=1 then
select nombre,apellido,salario into nombre,apellido,varSalario from emple where salario =
maxSalario;
  dbms_output.put_line(apellido||' , '||nombre||'=> '||varSalario);
else
dbms_output.put_line('hay '||n||' registros con '||maxSalario||' salario');
end if;
end;
```

2.- Visualizar el nombre del departamento, el código y el número de empleados del departamento que tenga más de 4 empleados.

```
set serveroutput on
declare

dept depart.dept_no%type;
dept_nombre depart.dnombre%type;
n int:=0;
begin

select d.dept_no, count(*)  into dept, n from depart d, emple e where e.dept_no=d.dept_no group
by d.dept_no having count(*)>4;
dbms_output.put_line('hay '||n||' empleados del departamento '||dept);
```

### EXCEPTION

```
when too_many_rows then dbms_output.put_line('hay mas de un registro');
when no_data_found then dbms_output.put_line('No hay registros');
```

```
end;
```

3.- Visualizar el departamento (nombre, dept\_no) que tiene mayor numero de empleados.

```
set serveroutput on
declare

dept depart.dept_no%type;
dept_nombre depart.dnombre%type;
n int:=0;
begin
```

```

select dept_no,dnombre into dept,dept_nombre from depart where dept_no =(select dept_no from
emple group by dept_no having count(emp_no) =(select max(count(emp_no)) from emple group by
dept_no));
dbms_output.put_line('El departamento con mayor numero de empleados es '||dept||' '||
dept_nombre);

```

#### EXCEPTION

```

when too_many_rows then dbms_output.put_line('hay mas de un registro');
when no_data_found then dbms_output.put_line('No hay registros');

end;

```

4.- Visualizar los datos de los departamentos, incluso los que no tienen empleados (No se puede hacer un bloque anónimo porque todavía no hemos visto cursores, hacerlo como una consulta desde la línea de comandos).

En mysql:

```
select distinct d.* from depart d left outer join emple e on d.dept_no=e.dept_no;
```

En Oracle:

```
select distinct d.* from depart d,emple e where d.dept_no=e.dept_no (+);
```

5.- Visualizar el numero de empleados de cada departamento incluyendo los que no tienen empleados. (línea de comandos)

```

Select d.dept_no,dnombre,count(emp_no) from emple e, depart d where d.dept_no=e.dept_no (+)
group by d.dept_no,dnombre;

```

6.- El número máximo de empleados que hay en algún departamentos.

```
Select max(count(*)) from emple group by dept_no;
```

7.- El número máximo de empleados que hay en algún departamentos, junto con el nombre del mismo.

```

Select d.dept_no,dnombre,count(emp_no) from emple e, depart d where d.dept_no=e.dept_no
group by d.dept_no,dnombre having count(emp_no)=(select max(count(emp_no)) from emple e,
depart d where d.dept_no=e.dept_no group by d.dept_no,dnombre);

```

8.- Se lee un numero por teclado y se visualiza si es un 1 => lunes ...7=> domingo, en otro caso error

```

set serveroutput on;
declare
dia int:='&DameDia';
cad varchar2(20);
begin
case dia
when 1 then cad:='lunes';
when 2 then cad:='martes';
when 3 then cad:='miercoles';
when 4 then cad:='jueves';
when 5 then cad:='viernes';
when 6 then cad:='sabado';
when 7 then cad:='domingo';

```

```

else
cad:='ERROR';
end case;
dbms_output.put_line(cad);
end;

set serveroutput on;
declare
dia int:='&DameDia';
cad varchar2(20);
begin
case
when dia >=1 and dia <6 then cad:='dia de la semana';
when dia =6 or dia=7 then cad:='fin de semana';
else
cad:='ERROR';
end case;
dbms_output.put_line(cad);
end;

```

9.- Realizar un bloque anónimo que calcule el importe de una factura sabiendo que el IVA a aplicar es del 12% y que si el importe bruto de la factura es superior a 50.000 € se debe realizar un descuento del 5%.

```

declare
iva constant real:=0.12;
importe_bruto real;
importe_netto real;
begin
importe_bruto:='&importe_bruto';
importe_bruto:=importe_bruto + (iva * importe_bruto);
if (importe_bruto>50000)
then
importe_netto:=importe_bruto - (importe_bruto*0.05);
dbms_output.put_line('el importe neto de la factura con descuento es '||importe_netto||' euros');
else
importe_netto:=importe_bruto;
dbms_output.put_line('el importe neto de la factura sin descuento es '||importe_netto||' euros');
end if;
end;
/

```

10.- Realizar un bloque pl/sql que nos pida un valor numérico por teclado y nos diga si dicho número es primo o no. Un número es primo si solo es divisible por el mismo y por la unidad.

```

declare
n number(3);
sw integer;
i integer;
begin
n:='&numero';
i:=2;
sw:=1;

```

```

while i<=n-1 loop
    sw:=n mod i;
    if sw=0 then
        exit;
    end if;
    i:=i+1;
end loop;
if sw = 1 then
    DBMS_OUTPUT.PUT_LINE(n||' es primo');
else
    DBMS_OUTPUT.PUT_LINE(n||' no es primo');
end if;
end;
/

```

**Otra forma:**

```

declare
n number(3);
    cont integer:=0;
    i integer;
begin
    n:='&numero';
    for var in 1..n
    loop
    if mod(n,var)=0 then
        cont:=cont+1;
    end if;
    end loop;
    if cont = 2 then
        DBMS_OUTPUT.PUT_LINE(n||' es primo');
    else
        DBMS_OUTPUT.PUT_LINE(n||' no es primo');
    end if;
end;

```

11.- Realizar un bloque pl/sql que nos pida 2 valores numéricos enteros por teclado y calcule la multiplicación de dichos números por sumas sucesivas.

```

declare
    num1 number(3):=&num1;
    num2 number(3):=&num2;
    resul number(3):=0;
    cont number(3):=0;
begin
    while cont<num2 loop
        resul:=resul+num1;
        cont:=cont+1;
    end loop;
    DBMS_OUTPUT.PUT_LINE('EL RESULTADO ES '||resul);
END;

```

12.- Realizar un bloque pl/sql que nos pida dos números enteros por teclado y calcule su división por restas sucesivas.

```

declare
  dividendo number(3):=&dividendo;
  divisor number(3):=&divisor;
  cociente number(3):=0;
  resto number(3):=0;
begin
  while dividendo>divisor loop
    cociente:=cociente+1;
    dividendo:=dividendo-divisor;
  end loop;
  resto:=dividendo;
  DBMS_OUTPUT.PUT_LINE('el resultado es=' || cociente||' de cociente y '|| resto||' de resto');
end;
/

```

13.- Hallar el factorial de un numero leído por teclado.

```

set serveroutput on;
declare
  n int:=&dameNumero;
  f int:=1;
begin
  if n<0
  then
    DBMS_OUTPUT.PUT_LINE('no existen factoriales de numeros negativos');
  else
    while (n>0)
    loop
      if (n!=1)
      then
        dbms_output.put(n||'*');
      else
        dbms_output.put(n||'=');
      end if;
      f:=f*n;
      n:=n-1;
    end loop;
    dbms_output.put_line(f);
  end if;
end;

```

14.- Visualizar los números entre dos números a y b leídos (Con todos lo tipos de bucles).

```

a) set serveroutput on;
declare
  n1 int:=&dameNumero1;
  n2 int:=&dameNumero2;
  aux int;
begin
  if (n1>n2) then
    aux:=n1;
    n1:=n2;
    n2:=aux;
  end if;

```

```
dbms_output.put_line(' Los numeros entre '||n1||' y '||n2||' son ');
for n in n1..n2
loop
dbms_output.put(n||' ');
```

```
end loop;
dbms_output.put_line("");
end;
```

B) set serveroutput on;

```
declare
n1 int:=&dameNumero1;
n2 int:=&dameNumero2;
aux int;
begin
if (n1>n2) then
aux:=n1;
n1:=n2;
n2:=aux;
end if;
```

```
dbms_output.put_line(' Los numeros entre '||n1||' y '||n2||' son ');
while (n1<=n2)
loop
dbms_output.put(n1||' ');
n1:=n1+1;
end loop;
dbms_output.put_line("");
end;
```

C)set serveroutput on;

```
declare
n1 int:=&dameNumero1;
n2 int:=&dameNumero2;
```

```
aux int;
begin
if (n1>n2) then
aux:=n1;
n1:=n2;
n2:=aux;
end if;
```

```
dbms_output.put_line(' Los numeros entre '||n1||' y '||n2||' son ');
loop
dbms_output.put(n1||' ');
n1:=n1+1;
if n1 >n2 then
exit;
end if;
end loop;
dbms_output.put_line("");
end;
```

15.- Realizar un bloque pl/sql que calcule la suma de los números enteros hasta un número dado introducido por el usuario

```
declare
    num integer:=&num;
    suma integer;
    i integer;
begin
    suma:=0;
    i:=1;
    while i<=num loop
        suma:=suma + i;
        i:=i+1;
    end loop;
    DBMS_OUTPUT.PUT_LINE('la suma de los ' ||num ||' primeros enteros es ' ||suma);
end;
/
```

16.- Realizar un bloque pl/sql que calcule las tablas de multiplicar del 1 al 9

```
declare
    mult real;
    num integer;
    i integer;
    j integer;
begin
    num:=9;
    i:=1;
    while i<=num loop
        j:=1;
        while j<=10 loop
            mult:=i*j;
            DBMS_OUTPUT.PUT_LINE(i ||'x' || j || '=' || mult);
            j:=j+1;
        end loop;
        DBMS_OUTPUT.PUT_LINE(' ');
        DBMS_OUTPUT.PUT_LINE(' ');
        i:=i+1;
    end loop;
end;
```

**Otra forma:**

```
begin
for i in 1..9 loop
    for j in 1..10 loop
        dbms_output.put_line(i||'*'||j||'='||(i*j));
    end loop;
dbms_output.put_line("");
end loop;
end;
```

17.- Para un número de empleado dado , incrementar el salario en un 10%.

declare

numEmple emple.emp\_no%type:=&NumeroEmpleado;

vsalario emple.salario%type;

n int;

begin

select count(\*) into n from emple where emp\_no=numEmple;

if n!=0 then

select emp\_no,salario into numEmple,vsalario from emple where emp\_no=numEmple;

dbms\_output.put('se modifica el sueldo de '||numEmple||' de '||vsalario||' a ');

vsalario:=vsalario+vsalario\*0.1;

update emple set salario=vsalario where emp\_no=numEmple;

dbms\_output.put\_line(vsalario);

else

dbms\_output.put\_line('no existe el empleado '||numEmple);

end if;

end;

**Otra forma:**

declare

numEmple emple.emp\_no%type:=&NumeroEmpleado;

vsalario emple.salario%type;

n int;

begin

select emp\_no,salario into numEmple,vsalario from emple where emp\_no=numEmple;

dbms\_output.put('se modifica el sueldo de '||numEmple||' de '||vsalario||' a ');

vsalario:=vsalario+vsalario\*0.1;

update emple set salario=vsalario where emp\_no=numEmple;

dbms\_output.put\_line(vsalario);

dbms\_output.put\_line('SE HAN MODIFICADO '||SQL%ROWCOUNT||' REGISTROS');

exception



```

when no_data_found then dbms_output.put_line('codigo de error: '||sqlcode||' mensaje '||sqlerrm);
dbms_output.put_line('no existe el empleado '||numEmple);
end;

```

18.- Aceptar tres valores que correspondan al número, nombre y localidad de un departamento. Realizar un bloque PL/SQL que inserte en la tabla DEPT los valores aceptados. Controlar los siguientes errores:

- Si el departamento ya existe, insertar en la tabla MENSAJES\_ERROR un mensaje apropiado.
- Si se producen otros errores, insertar en la tabla MENSAJES\_ERROR el número y el mensaje del error producido.

```

declare
numDept depart.dept_no%type:=&DameDept_no;
nombre depart.dnombre%type:='&DameNombre';
dir depart.tdir%type:='&Dameletra';
n int;
begin
select count(*) into n from depart where dept_no=numDept;
if n !=0 then
    insert into mensajes_error VALUES ('EL DEPARTAMENTO '||NUMdEPT||' YA EXISTE');
ELSE
    INSERT INTO DEPART (DEPT_NO,DNOMBRE,TDIR) VALUES (numDept,nombre,dir);
dbms_output.put_line('Se inserta '||numDept||' '||nombre||' '||dir);
end if;
exception
when value_error then dbms_output.put_line('ERROR tamaño '||sqlcode||sqlerrm);
INSERT INTO MENSAJES_ERROR VALUES ('ERROR TAMAÑO ');
when others then dbms_output.put_line('ERROR '||sqlcode||sqlerrm);
dbms_output.put_line('ERROR '||sqlcode||sqlerrm);
end;

```