

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
```

```
VDNOMBRE DEPART.DNOMBRE%TYPE;
VDEPT_NO DEPART.DEPT_NO%TYPE;
NUMEMPLE INT;
```

```
BEGIN
SELECT DNOMBRE, D.DEPT_NO, COUNT(*) INTO VDNOMBRE,
VDEPT_NO,NUMEMPLE
FROM EMPL E, DEPART D
WHERE E.DEPT_NO=D.DEPT_NO
GROUP BY D.DEPT_NO, DNOMBRE HAVING COUNT(*) >4;
```

```
EXCEPTION
WHEN NO_DATA_FOUND THEN dbms_output.put_line('NO HAY DEPARTAMENTOS CON
MAS DE 4 EMPLEADOS');
WHEN TOO_MANY_ROWS THEN dbms_output.put_line('HAY MAS DE UN
DEPARTAMENTO CON MAS DE 4 EMPLEADOS');

END;
```

3) Visualizar el departamento (nombre, dept_no) que tiene mayor numero de empleados.

Visualizar el departamento (nombre, dept_no) que tiene mayor numero de empleados.

– PREVIO LÍNEA DE COMANDOS :

/*

```
select dept_no from emple group by dept_no having count(*) IN(
select count(*) from emple group by dept_no having count(*)
>=all(
```

```
select dept_no from emple group by dept_no having count(*) IN(
select count(*) from emple group by dept_no having count(*)
>=all(
```

```
select count(*) from emple group by dept_no))));
```

*/

DECLARE

VDEPT_NO EMPLE.DEPT_NO%TYPE;

VDNOMBRE DEPART.DNOMBRE%TYPE;

BEGIN

```
select D.dept_no,DNOMBRE INTO Vdept_no,VDNOMBRE  from
EMPLE E ,DEPART D WHERE e.dept_no=d.dept_no
group by D.dept_no,DNOMBRE  having count(*) IN(
select count(*) from emple group by dept_no having count(*) >=all(
select count(*) from emple group by dept_no))));
```

```
dbms_output.put_line('EL DEPARTAMENTO CON MAYOR NUMERO
DE EMPLEADOS ES '||VDNOMBRE||'('||Vdept_no||')');
```

EXCEPTION

```
WHEN TOO_MANY_ROWS THEN dbms_output.put_line('HAY MAS
DE UN DEPARTAMENTO CON EL MAXIMO NUMERO DE
EMPLEADOS');
```

```
when no_data_found then dbms_output.put_line('No hay registros');
```

end;

Otra FORMA:

```
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 (+);
```

(SÓLO EN ORACLE)

```
SELECT DISTINCT D.DEPT_NO,DNOMBRE,COUNT(EMP_NO)  
FROM EMPL E, DEPART D WHERE  
d.dept_no=e.dept_no(+) GROUP BY D.DEPT_NO,DNOMBRE;
```

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

```
SELECT DISTINCT D.DEPT_NO, DNOMBRE,COUNT(EMP_NO)  
FROM DEPART D LEFT OUTER JOIN EMPL E  
ON d.dept_no=e.dept_no GROUP BY D.DEPT_NO,DNOMBRE;
```

– OTRA FORMA (SÓLO EN ORACLE) :

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,

– PREVIO:

-- SELECT MAX(COUNT(EMP_NO)) FROM EMPLE GROUP BY DEPT_NO;

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

– PREVIO, DESDE LA LÍNEA DE COMANDOS:

-- SELECT DNOMBRE,D.DEPT_NO,COUNT(EMP_NO) FROM EMPLE E, DEPART D
-- WHERE E.DEPT_NO=D.DEPT_NO GROUP BY D.DEPT_NO, DNOMBRE;

SELECT DNOMBRE,D.DEPT_NO,COUNT(EMP_NO) FROM EMPLE E, DEPART D WHERE E.DEPT_NO=D.DEPT_NO GROUP BY D.DEPT_NO, DNOMBRE HAVING COUNT(EMP_NO) =(SELECT MAX(COUNT(EMP_NO)) FROM EMPLE GROUP BY DEPT_NO);

SET SERVEROUTPUT ON

DECLARE

VDNOMBRE DEPART.DNOMBRE%TYPE;

VDEPT_NO DEPART.DEPT_NO%TYPE;

CONT INT;

BEGIN

SELECT DNOMBRE,D.DEPT_NO,COUNT(EMP_NO)
INTO VDNOMBRE,VDEPT_NO,CONT
FROM EMPLE E, DEPART D
WHERE E.DEPT_NO=D.DEPT_NO
GROUP BY D.DEPT_NO, DNOMBRE
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: '||VDNOMBRE||' CON '||CONT||'
EMPLEADOS');
EXCEPTION
  WHEN TOO_MANY_ROWS THEN dbms_output.PUT_LINE('HAY
MAS DE UN DEPARTAMENTO CON EL NUMERO MAXIMO DE
EMPLEADOS');
END;

```

8.- Visualizar todos los datos del departamento 'Investigacion'.

```

SET SERVEROUTPUT ON
DECLARE
fila depart%rowtype;
deptBuscar varchar2(20):='&DAME_NOMBRE_DEP';
BEGIN
select * into fila from depart where upper(dnombre)=upper(deptBuscar);
dbms_output.put_line(fila.dept_no||' '||fila.direc||' '||fila.numce);
EXCEPTION
  WHEN no_data_found THEN dbms_output.PUT_LINE('no existe el
departamento '||deptBuscar);
END;

```

9.- 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.

PREVIAMENTE SE CREARÁ LA TABLA PARA LOS MENSAJES DE ERROR:

```

CREATE TABLE MENSAJES_ERROR (MENS VARCHAR2(200));

```

```

SET SERVEROUTPUT ON;
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 others then dbms_output.put_line('ERROR '||sqlcode||sqlerrm);
dbms_output.put_line('ERROR '||sqlcode||sqlerrm);
INSERT INTO MENSAJES_ERROR VALUES ('ERROR ');

end;

```

10.- 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;

```

11.- 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;
/

```

12.- 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;

```

13.- 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;

```

14.- 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;
/

```

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

```

set serveroutput on;
declare
n int:=&dameNumero;
f int:=1;
begin
    if num<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;

```

16.- 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;

```

17.- 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;
/

```

18.- Realizar un bloque pl/sql que calcule las tablas de multiplicar del 1 al 10 COMO UN CUADRADO.

```

declare
    mult real;
    num CONSTANT integer:=10;
    i integer;
    j integer;
begin

i:=1;
while i<=num loop
    j:=1;
    while j<=10 loop
        mult:=i*j;
DBMS_OUTPUT.PUT( mult||' ');
        j:=j+1;
    end loop;
    DBMS_OUTPUT.PUT_LINE(' ');
    i:=i+1;
end loop;
end;

```

19.- 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);
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;
```