**EJERCICIOS CURSORES:**

1. Escribir un bloque PL que utilice un cursor explicito para visualizar el nombre y la localidad de todos los departamentos.

DECLARE

CURSOR c\_departments IS

SELECT department\_name, location

FROM departments;

BEGIN

FOR dept\_rec IN c\_departments

LOOP

DBMS\_OUTPUT.PUT\_LINE('Nombre del departamento: ' || dept\_rec.department\_name || ', Localidad: ' || dept\_rec.location);

END LOOP;

END;

/

1. Visualizar los apellidos de los empleados pertenecientes al departamento 20 numerándolos secuencialmente. Utilizar %ROWCOUNT los números secuenciales.

1.SANCHEZ

2.JIMENEZ

3.GIL

4.ALONSO

5.FERNANDEZ

DECLARE

v\_counter NUMBER := 1;

BEGIN

FOR emp\_rec IN (SELECT last\_name FROM employees WHERE department\_id = 20)

LOOP

DBMS\_OUTPUT.PUT\_LINE(v\_counter || '.' || emp\_rec.last\_name);

v\_counter := v\_counter + 1;

END LOOP;

END;

/

1. Visualizar los empleados de un departamento mediante un procedimiento y utilizando variables de acoplamiento.

CREATE OR REPLACE PROCEDURE mostrar\_empleados(dept\_id IN NUMBER)

IS

BEGIN

FOR emp\_rec IN (SELECT employee\_id, last\_name FROM employees WHERE department\_id = dept\_id)

LOOP

DBMS\_OUTPUT.PUT\_LINE('ID: ' || emp\_rec.employee\_id || ', Apellido: ' || emp\_rec.last\_name);

END LOOP;

END mostrar\_empleados;

/

1. Escribir un bloque PL que visualice el apellido, el oficio y la comisión de los empleados que supere los 500€. Utilizando CURSOR FOR........LOOP.

DECLARE

CURSOR c\_high\_commission IS

SELECT last\_name, job\_id, commission\_pct

FROM employees

WHERE commission\_pct > 0.5;

BEGIN

FOR emp\_rec IN c\_high\_commission

LOOP

DBMS\_OUTPUT.PUT\_LINE('Apellido: ' || emp\_rec.last\_name || ', Oficio: ' || emp\_rec.job\_id || ', Comisión: ' || emp\_rec.commission\_pct);

END LOOP;

END;

/

1. Escribir un bloque PL que visualice el apellido y la fecha de alta de todos los empleados ordenados por fecha de alta. Utilizando CURSOR FOR........LOOP.

DECLARE

CURSOR c\_employees IS

SELECT last\_name, hire\_date

FROM employees

ORDER BY hire\_date;

BEGIN

FOR emp\_rec IN c\_employees

LOOP

DBMS\_OUTPUT.PUT\_LINE('Apellido: ' || emp\_rec.last\_name || ', Fecha de Alta: ' || emp\_rec.hire\_date);

END LOOP;

END;

/

1. Desarrollar un procedimiento que visualice el apellido y la fecha de alta de todos los empleados ordenados por apellido.

CREATE OR REPLACE PROCEDURE mostrar\_empleados\_apellido

IS

BEGIN

FOR emp\_rec IN (SELECT last\_name, hire\_date FROM employees ORDER BY last\_name)

LOOP

DBMS\_OUTPUT.PUT\_LINE('Apellido: ' || emp\_rec.last\_name || ', Fecha de Alta: ' || emp\_rec.hire\_date);

END LOOP;

END mostrar\_empleados\_apellido;

/

1. Codificar un procedimiento que muestre el nombre de cada departamento y el número de empleados que tiene.

CREATE OR REPLACE PROCEDURE mostrar\_num\_empleados\_por\_departamento

IS

BEGIN

FOR dept\_rec IN (SELECT department\_name, COUNT(\*) AS num\_empleados FROM employees GROUP BY department\_name)

LOOP

DBMS\_OUTPUT.PUT\_LINE('Departamento: ' || dept\_rec.department\_name || ', Número de Empleados: ' || dept\_rec.num\_empleados);

END LOOP;

END mostrar\_num\_empleados\_por\_departamento;

/

1. Escribir un procedimiento que visualice el apellido y el salario de los cinco empleados que tienen el salario más alto.

REY 4100

NEGRO 3005

FERNANDEZ3000

GIL 3000

JIMENEZ 2900

CEREZO 2885

MUÑOZ 1690

SALA 1625

MARTIN 1600

ARROYO 1500

ALONSO 1430

TOVAR 1350

JIMENO 1335

SANCHEZ 1040

La salida seria:

**REY\*4100**

**NEGRO\*3005**

**FERNANDEZ\*3000**

**GIL\*3000**

**JIMENEZ\*2900**

**CREATE OR REPLACE PROCEDURE mostrar\_top\_salarios**

**IS**

**BEGIN**

**FOR emp\_rec IN (SELECT last\_name, salary FROM employees ORDER BY salary DESC FETCH FIRST 5 ROWS ONLY)**

**LOOP**

**DBMS\_OUTPUT.PUT\_LINE(emp\_rec.last\_name || '\*' || emp\_rec.salary);**

**END LOOP;**

**END mostrar\_top\_salarios;**

**/**

1. Codificar un procedimiento que visualice los dos empleados que ganan menos de cada oficio.

FERNANDEZANALISTA 3000

GIL ANALISTA 3000

CEREZO DIRECTOR 2885

JIMENEZ DIRECTOR 2900

NEGRO DIRECTOR 3005

SANCHEZ EMPLEADO 1040

JIMENO EMPLEADO 1335

ALONSO EMPLEADO 1430

MUÑOZ EMPLEADO 1690

REY PRESIDENTE 4100

TOVAR VENDEDOR 1350

ARROYO VENDEDOR 1500

MARTIN VENDEDOR 1600

SALA VENDEDOR 1625

La salida seria:

**ANALISTA\*FERNANDEZ\*3000**

**ANALISTA\*GIL\*3000**

**DIRECTOR\*CEREZO\*2885**

**DIRECTOR\*JIMENEZ\*2900**

**EMPLEADO\*SANCHEZ\*1040**

**EMPLEADO\*JIMENO\*1335**

**PRESIDENTE\*REY\*4100**

**VENDEDOR\*TOVAR\*1350**

**VENDEDOR\*ARROYO\*1500**

**CREATE OR REPLACE PROCEDURE mostrar\_dos\_menores\_salarios\_por\_oficio**

**IS**

**BEGIN**

**FOR job\_rec IN (SELECT DISTINCT job\_id FROM employees)**

**LOOP**

**FOR emp\_rec IN (SELECT last\_name FROM employees WHERE job\_id = job\_rec.job\_id ORDER BY salary FETCH FIRST 2 ROWS ONLY)**

**LOOP**

**DBMS\_OUTPUT.PUT\_LINE(job\_rec.job\_id || '\*' || emp\_rec.last\_name);**

**END LOOP;**

**END LOOP;**

**END mostrar\_dos\_menores\_salarios\_por\_oficio;**

**/**

**LOS MAS INTERESANTES SON EL 8 Y 9**