**- Procedimiento que dado un codigo de un empleado mostrar en pantalla el código de empleado, nombre y apellidos de todos sus subordinados.**

**CREATE OR REPLACE PROCEDURE LISTADO\_SUBORDINADOS (CODIGO EMPLOYEES.EMPLOYEE\_ID%TYPE) IS**

**CURSOR SUBORDINADOS IS**

**SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME**

**FROM EMPLOYEES**

**WHERE MANAGER\_ID=CODIGO;**

**NOMBRE EMPLOYEES.FIRST\_NAME%TYPE;**

**APELLIDO EMPLOYEES.LAST\_NAME%TYPE;**

**CODIGOEMP EMPLOYEES.EMPLOYEE\_ID%TYPE;**

**BEGIN**

**OPEN SUBORDINADOS;**

**LOOP**

**FETCH SUBORDINADOS INTO CODIGOEMP, NOMBRE, APELLIDO;**

**EXIT WHEN SUBORDINADOS%NOTFOUND;**

**DBMS\_OUTPUT.PUT\_LINE(CODIGOEMP||' '||NOMBRE||' '||APELLIDO);**

**END LOOP;**

**CLOSE SUBORDINADOS;**

**END;**

**EXECUTE LISTADO\_SUBORDINADOS(101);**

**Versión 2:**

**CREATE OR REPLACE PROCEDURE LISTADO\_SUBORDINADOS2 (CODIGO EMPLOYEES.EMPLOYEE\_ID%TYPE) IS**

**CURSOR SUBORDINADOS IS**

**SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME**

**FROM EMPLOYEES**

**WHERE MANAGER\_ID=CODIGO;**

**REGISTRO SUBORDINADOS%ROWTYPE;**

**BEGIN**

**OPEN SUBORDINADOS;**

**LOOP**

**FETCH SUBORDINADOS INTO REGISTRO;**

**EXIT WHEN SUBORDINADOS%NOTFOUND;**

**DBMS\_OUTPUT.PUT\_LINE(REGISTRO.EMPLOYEE\_ID||' '||REGISTRO.FIRST\_NAME||' '||REGISTRO.LAST\_NAME);**

**END LOOP;**

**CLOSE SUBORDINADOS;**

**END;**

**Versión 3:**

**CREATE OR REPLACE PROCEDURE LISTADO\_SUBORDINADOS3 (CODIGO EMPLOYEES.EMPLOYEE\_ID%TYPE) IS**

**CURSOR SUBORDINADOS IS**

**SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME**

**FROM EMPLOYEES**

**WHERE MANAGER\_ID=CODIGO;**

**BEGIN**

**FOR REGISTRO IN SUBORDINADOS LOOP**

**DBMS\_OUTPUT.PUT\_LINE(REGISTRO.EMPLOYEE\_ID||' '||REGISTRO.FIRST\_NAME||' '||REGISTRO.LAST\_NAME);**

**END LOOP;**

**END;**

**- Procedimiento que reciba nombre de departamento y emita:**

**a)Nombre y apellidos de todos sus empleados.**

**b)Número de empleados total.**

**c)Suma salarial de sus empleados.**

**Versión corta:**

**CREATE OR REPLACE PROCEDURE INFO\_DEPARTAMENTOS2 (NOMBRE DEPARTMENTS.DEPARTMENT\_NAME%TYPE) IS**

**CURSOR EMPLEADOS\_DEPT IS**

**SELECT FIRST\_NAME, LAST\_NAME, SALARY**

**FROM EMPLOYEES JOIN DEPARTMENTS USING (DEPARTMENT\_ID)**

**WHERE DEPARTMENT\_NAME=NOMBRE;**

**TOTAL\_EMPLEADOS NUMBER(4):=0;**

**SUM\_SALARIAL NUMBER(9,2):=0;**

**BEGIN**

**FOR REGISTRO IN EMPLEADOS\_DEPT LOOP**

**DBMS\_OUTPUT.PUT\_LINE(REGISTRO.FIRST\_NAME||' '||REGISTRO.LAST\_NAME);**

**SUM\_SALARIAL:=SUM\_SALARIAL+REGISTRO.SALARY;**

**TOTAL\_EMPLEADOS:=TOTAL\_EMPLEADOS+1;**

**END LOOP;**

**DBMS\_OUTPUT.PUT\_LINE('Total empleados: '||TOTAL\_EMPLEADOS);**

**DBMS\_OUTPUT.PUT\_LINE('Suma salarial: '||SUM\_SALARIAL);**

**END;**

**Versión larga:**

**CREATE OR REPLACE PROCEDURE INFO\_DEPARTAMENTOS (NOMBRE DEPARTMENTS.DEPARTMENT\_NAME%TYPE) IS**

**CURSOR EMPLEADOS\_DEPT IS**

**SELECT FIRST\_NAME, LAST\_NAME**

**FROM EMPLOYEES JOIN DEPARTMENTS USING (DEPARTMENT\_ID)**

**WHERE DEPARTMENT\_NAME=NOMBRE;**

**TOTAL\_EMPLEADOS NUMBER(2);**

**SUM\_SALARIAL NUMBER(9,2);**

**BEGIN**

**SELECT COUNT(\*) INTO TOTAL\_EMPLEADOS**

**FROM EMPLOYEES JOIN DEPARTMENTS USING (DEPARTMENT\_ID)**

**WHERE DEPARTMENT\_NAME=NOMBRE;**

**SELECT SUM(SALARY) INTO SUM\_SALARIAL**

**FROM EMPLOYEES JOIN DEPARTMENTS USING (DEPARTMENT\_ID)**

**WHERE DEPARTMENT\_NAME=NOMBRE;**

**FOR REGISTRO IN EMPLEADOS\_DEPT LOOP**

**DBMS\_OUTPUT.PUT\_LINE(REGISTRO.FIRST\_NAME||' '||REGISTRO.LAST\_NAME);**

**END LOOP;**

**DBMS\_OUTPUT.PUT\_LINE('Total empleados: '||TOTAL\_EMPLEADOS);**

**DBMS\_OUTPUT.PUT\_LINE('Suma salarial: '||SUM\_SALARIAL);**

**END;**

**- Dado el nombre de una localidad, mostrar un listado de empleados indicando al final el nombre del empleado con mayor salario de todos.**

**CREATE OR REPLACE PROCEDURE INFO\_LOCALIDAD (CIUDAD LOCATIONS.CITY%TYPE) IS**

**CURSOR EMPLEADOS IS**

**SELECT FIRST\_NAME, LAST\_NAME, SALARY**

**FROM EMPLOYEES JOIN DEPARTMENTS USING (DEPARTMENT\_ID)**

**JOIN LOCATIONS USING (LOCATION\_ID)**

**WHERE CITY=CIUDAD**

**ORDER BY SALARY;**

**REGISTRO2 EMPLEADOS%ROWTYPE;**

**BEGIN**

**FOR REGISTRO IN EMPLEADOS LOOP**

**DBMS\_OUTPUT.PUT\_LINE(REGISTRO.FIRST\_NAME||' '||REGISTRO.LAST\_NAME);**

**REGISTRO2:=REGISTRO;**

**END LOOP;**

**DBMS\_OUTPUT.PUT\_LINE('EMPLEADOS CON MAYOR SALARIO: ');**

**FOR REGISTRO IN EMPLEADOS LOOP**

**IF REGISTRO.SALARY=REGISTRO2.SALARY THEN**

**DBMS\_OUTPUT.PUT\_LINE(REGISTRO.FIRST\_NAME||' '||REGISTRO.LAST\_NAME);**

**END IF;**

**END LOOP;**

**END;**

**EXECUTE INFO\_LOCALIDAD('Seattle');´**

**- Escribir procedimiento que dada una ciudad muestre todos aquellos empleados cuyo salario está por encima de la media salarial de la ciudad.**

**Versión 1:**

**CREATE OR REPLACE PROCEDURE INFO\_LOCALIDAD2 (CIUDAD LOCATIONS.CITY%TYPE) IS**

**CURSOR EMPLEADOS IS**

**SELECT FIRST\_NAME, LAST\_NAME, SALARY**

**FROM EMPLOYEES JOIN DEPARTMENTS USING (DEPARTMENT\_ID)**

**JOIN LOCATIONS USING (LOCATION\_ID)**

**WHERE CITY=CIUDAD AND SALARY > (SELECT AVG(SALARY)**

**FROM EMPLOYEES JOIN DEPARTMENTS USING (DEPARTMENT\_ID)**

**JOIN LOCATIONS USING (LOCATION\_ID)**

**WHERE CITY=CIUDAD)**

**ORDER BY SALARY;**

**BEGIN**

**FOR REGISTRO IN EMPLEADOS LOOP**

**DBMS\_OUTPUT.PUT\_LINE(REGISTRO.FIRST\_NAME||' '||REGISTRO.LAST\_NAME||' '||REGISTRO.SALARY);**

**END LOOP;**

**END;**

**Versión 2:**

**CREATE OR REPLACE PROCEDURE INFO\_LOCALIDAD3 (CIUDAD LOCATIONS.CITY%TYPE) IS**

**CURSOR EMPLEADOS IS**

**SELECT FIRST\_NAME, LAST\_NAME, SALARY**

**FROM EMPLOYEES JOIN DEPARTMENTS USING (DEPARTMENT\_ID)**

**JOIN LOCATIONS USING (LOCATION\_ID)**

**WHERE CITY=CIUDAD;**

**MEDIA\_SALARIAL EMPLOYEES.SALARY%TYPE;**

**BEGIN**

**SELECT AVG(SALARY) INTO MEDIA\_SALARIAL**

**FROM EMPLOYEES JOIN DEPARTMENTS USING (DEPARTMENT\_ID)**

**JOIN LOCATIONS USING (LOCATION\_ID)**

**WHERE CITY=CIUDAD;**

**FOR REGISTRO IN EMPLEADOS LOOP**

**IF REGISTRO.SALARY > MEDIA\_SALARIAL THEN**

**DBMS\_OUTPUT.PUT\_LINE(REGISTRO.FIRST\_NAME||' '||REGISTRO.LAST\_NAME);**

**END IF;**

**END LOOP;**

**END;**