**Desarrollo de aplicaciones multiplataforma**

**Acceso a datos**

**Tema 2-Práctica 6: Modificar métodos**

1. Crear el tipo “empleado” con los atributos “Rut varchar(10)”, “Nombre varchar(10)”, “Cargo varchar(9)”, “fechaIng date”, “sueldo number(9)”, “somision number (9)”, y “anticipo number (9)”, la función “sueldo\_liquido”, que devuelve un number, y el procedimiento “aumento\_sueldo” que recibe como parámetro el aumento (number).

**RESPUESTA:**

**CREATE OR REPLACE TYPE EMPLEADO AS OBJECT(**

**RUT VARCHAR(10),**

**NOMBRE VARCHAR(10),**

**CARGO VARCHAR(9),**

**FECHAING DATE,**

**SUELDO NUMBER(9),**

**COMISION NUMBER(9),**

**ANTICIPO NUMBER(9),**

**MEMBER FUNCTION SUELDO\_LIQUIDO RETURN NUMBER,**

**MEMBER PROCEDURE AUMENTO\_SUELDO(AUMENTO NUMBER));**

**/**

1. Crear el body para dicho tipo desarrollando la función y el procedimiento. El sueldo líquido se calculará como “(sueldo +comisión )- anticipo”, y el aumento de sueldo se calculará como “sueldo+aumento”.

**RESPUESTA:**

**CREATE OR REPLACE TYPE BODY EMPLEADO AS**

**MEMBER FUNCTION SUELDO\_LIQUIDO RETURN NUMBER**

**IS**

**BEGIN**

**RETURN(SUELDO+COMISION)-ANTICIPO;**

**END;**

**MEMBER PROCEDURE AUMENTO\_SUELDO()**

**IS**

**BEGIN**

**SUELDO:=SUELDO+AUMENTO;**

**END;**

**END;**

**/**

1. Altera el tipo Empleado y añade el procedimiento “setAnticipo” que recibe como parámetro el anticipo de tipo number.

**RESPUESTA:**

**ALTER TYPE EMPLEADO REPLACE AS OBJECT(**

**RUT VARCHAR(10),**

**NOMBRE VARCHAR(10),**

**CARGO VARCHAR(9),**

**FECHAING DATE,**

**SUELDO NUMBER(9),**

**COMISION NUMBER(9),**

**ANTICIPO NUMBER(9),**

**MEMBER FUNCTION SUELDO\_LIQUIDO RETURN NUMBER,**

**MEMBER PROCEDURE AUMENTO\_SUELDO(AUMENTO NUMBER),**

**MEMBER PROCEDURE SETANTICIPO(ANTICIPO NUMBER));**

1. Crea el body para el nuevo método “setAnticipo”.

**RESPUESTA:**

**CREATE OR REPLACE TYPE BODY EMPLEADO AS**

**MEMBER FUNCTION SUELDO\_LIQUIDO RETURN NUMBER**

**IS**

**BEGIN**

**RETURN(SUELDO+COMISION)-ANTICIPO;**

**END;**

**MEMBER PROCEDURE AUMENTO\_SUELDO()**

**IS**

**BEGIN**

**SUELDO:=SUELDO+AUMENTO;**

**END;**

**MEMBER PROCEDURE SETANTICIPO(ANTICIPO NUMBER)**

**IS**

**BEGIN**

**SELF.ANTICIPO:=ANTICIPO;**

**END;**

**END;**

**/**

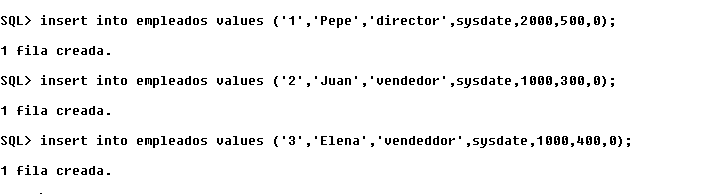
1. Crear una tabla empleados de tipo empleado.

**RESPUESTA:**

**DROP TABLE EMPLEADOS2:**

**CREATE TABLE EMPLEADOS2 OF EMPLEADO;**

1. Insertar dos o tres empleados, con estos datos.



**RESPUESTA:**

**INSERT INTO EMPLEADOS2 VALUES('1', 'Pepa','DIRECTORA',SYSDATE, 2000,500, 0);**

**INSERT INTO EMPLEADOS2 VALUES('2', 'Juana','COMERCIAL',SYSDATE, 1000,800, 0);**

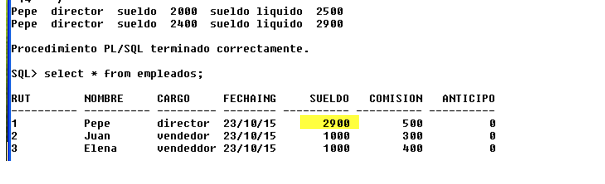
**INSERT INTO EMPLEADOS2 VALUES('3', 'Rosa','COMERCIAL',SYSDATE, 1000,800, 0);**

1. Crear un bloque PL/SQL para listar el sueldo líquido del empleado rut=1. Aumentarle el sueldo con 400 euros. Listar el sueldo aumentado. La salida será como sigue:



**RESPUESTA:**

1. Persistir en la tabla empleados el sueldo aumentado



**RESPUESTA:**

1. Sacar los sueldos y sus sueldos líquidos de todos los empleados. OJO: poner alias en las funciones dentro de cursores.

**RESPUESTA:**

**DECLARE**

**EMP EMPLEADO;**

**BEGIN**

**SELECT VALUE(E) INTO EMP FROM EMPLEADOS2 E WEHRE E.RUT='1';**

**DBMS\_OUTPUT.PUT\_LINE(**

**EMP.NOMBRE||' '||EMP.CARGO||' SUELDO: '||EMP.SUELDO||' SUELDO LIQUIDO :'||EMP.SUELDO\_LIQUIDO());**

**EMP.AUMENTO\_SUELDO(400);**

**DBMS\_OUTPUT.PUT\_LINE(**

**EMP.NOMBRE||' '||EMP.CARGO||' SUELDO: '||EMP.SUELDO||' SUELDO LIQUIDO :'||EMP.SUELDO\_LIQUIDO());**

**END;**

**/**