**Desarrollo de aplicaciones multiplataforma**

**Acceso a datos**

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

**Mohamed El Younousi Bentalha**

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 VARCHAR2(10),

Nombre VARCHAR2(10),

Cargo VARCHAR2(9),

fechaIng DATE,

sueldo NUMBER(9),

comision NUMBER(9),

anticipo NUMBER(9),

MAP 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

MAP MEMBER FUNCTION sueldo\_liquido RETURN NUMBER IS

BEGIN

-- Calcular sueldo líquido

RETURN sueldo + comision - anticipo;

END sueldo\_liquido;

MEMBER PROCEDURE aumento\_sueldo(aumento NUMBER) IS

BEGIN

-- Aplicar aumento de sueldo

sueldo := sueldo + aumento;

END aumento\_sueldo;

END;

/

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

**RESPUESTA:**

ALTER TYPE empleado ADD MEMBER PROCEDURE setAnticipo(anticipo NUMBER);

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

**RESPUESTA:**

CREATE OR REPLACE TYPE BODY empleado AS

MAP MEMBER FUNCTION sueldo\_liquido RETURN NUMBER IS

BEGIN

RETURN sueldo + comision - anticipo;

END sueldo\_liquido;

MEMBER PROCEDURE aumento\_sueldo(aumento NUMBER) IS

BEGIN

sueldo := sueldo + aumento;

END aumento\_sueldo;

MEMBER PROCEDURE setAnticipo(anticipo NUMBER) IS

BEGIN

self.anticipo := anticipo;

END setAnticipo;

END;

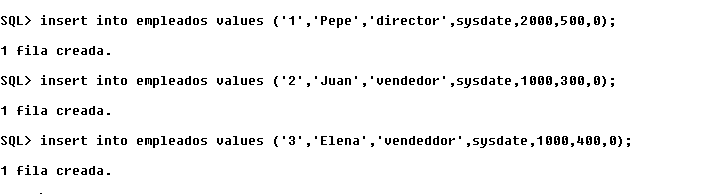
/

1. Crear una tabla empleados de tipo empleado.

**RESPUESTA:**

CREATE TABLE empleados OF empleado;

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



**RESPUESTA:**

insert into empleados values ('1','Pepe','director',sysdate,2000,500,0);

insert into empleados values ('2','Juan','vendedor',sysdate,1000,300,0);

insert into empleados values ('3','Elena','vendedor',sysdate,1000,400,0);

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:**

DECLARE

v\_rut VARCHAR2(10) := '1';

v\_sueldo\_anterior NUMBER;

v\_sueldo\_liquido\_anterior NUMBER;

v\_sueldo\_nuevo NUMBER;

v\_sueldo\_liquido\_nuevo NUMBER;

BEGIN

-- Obtener el sueldo y sueldo líquido actual del empleado con Rut=1

SELECT e.Sueldo, e.sueldo\_liquido() INTO v\_sueldo\_anterior, v\_sueldo\_liquido\_anterior

FROM empleados e

WHERE e.Rut = v\_rut;

-- Mostrar el sueldo y sueldo líquido actual

DBMS\_OUTPUT.PUT\_LINE('Antes del aumento:');

DBMS\_OUTPUT.PUT\_LINE('Sueldo de ' || v\_rut || ': ' || v\_sueldo\_anterior);

DBMS\_OUTPUT.PUT\_LINE('Sueldo líquido de ' || v\_rut || ': ' || v\_sueldo\_liquido\_anterior);

-- Aumentar el sueldo en 400 euros

UPDATE empleados

SET Sueldo = Sueldo + 400

WHERE Rut = v\_rut;

-- Obtener el nuevo sueldo y sueldo líquido

SELECT e.Sueldo, e.sueldo\_liquido() INTO v\_sueldo\_nuevo, v\_sueldo\_liquido\_nuevo

FROM empleados e

WHERE e.Rut = v\_rut;

-- Mostrar el sueldo y sueldo líquido después del aumento

DBMS\_OUTPUT.PUT\_LINE('Después del aumento:');

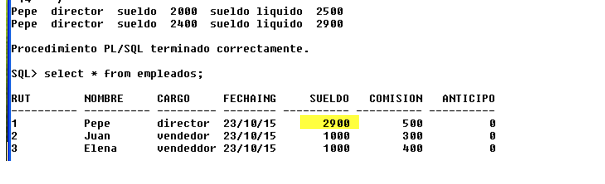
DBMS\_OUTPUT.PUT\_LINE('Sueldo de ' || v\_rut || ': ' || v\_sueldo\_nuevo);

DBMS\_OUTPUT.PUT\_LINE('Sueldo líquido de ' || v\_rut || ': ' || v\_sueldo\_liquido\_nuevo);

END;

/

1. Persistir en la tabla empleados el sueldo aumentado



**RESPUESTA:**

DECLARE

v\_rut VARCHAR2(10) := '1';

v\_sueldo\_anterior NUMBER;

v\_sueldo\_liquido\_anterior NUMBER;

v\_sueldo\_nuevo NUMBER;

v\_sueldo\_liquido\_nuevo NUMBER;

BEGIN

-- Obtener el sueldo y sueldo líquido actual del empleado con Rut=1

SELECT e.Sueldo, e.sueldo\_liquido() INTO v\_sueldo\_anterior, v\_sueldo\_liquido\_anterior

FROM empleados e

WHERE e.Rut = v\_rut;

-- Mostrar el sueldo y sueldo líquido actual

DBMS\_OUTPUT.PUT\_LINE('Antes del aumento:');

DBMS\_OUTPUT.PUT\_LINE('Sueldo de ' || v\_rut || ': ' || v\_sueldo\_anterior);

DBMS\_OUTPUT.PUT\_LINE('Sueldo líquido de ' || v\_rut || ': ' || v\_sueldo\_liquido\_anterior);

-- Aumentar el sueldo en 400 euros

UPDATE empleados

SET Sueldo = Sueldo + 400

WHERE Rut = v\_rut;

-- Obtener el nuevo sueldo y sueldo líquido

SELECT e.Sueldo, e.sueldo\_liquido() INTO v\_sueldo\_nuevo, v\_sueldo\_liquido\_nuevo

FROM empleados e

WHERE e.Rut = v\_rut;

-- Mostrar el sueldo y sueldo líquido después del aumento

DBMS\_OUTPUT.PUT\_LINE('Después del aumento:');

DBMS\_OUTPUT.PUT\_LINE('Sueldo de ' || v\_rut || ': ' || v\_sueldo\_nuevo);

DBMS\_OUTPUT.PUT\_LINE('Sueldo líquido de ' || v\_rut || ': ' || v\_sueldo\_liquido\_nuevo);

**-- Persistir el sueldo aumentado en la tabla empleados**

**COMMIT;**

END;

/

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

**RESPUESTA:**

DECLARE

CURSOR c\_empleados IS

SELECT

e.Rut,

e.Nombre,

e.Cargo,

e.Sueldo,

e.sueldo\_liquido() AS Sueldo\_Liquido

FROM

empleados e;

v\_rut empleados.Rut%TYPE;

v\_nombre empleados.Nombre%TYPE;

v\_cargo empleados.Cargo%TYPE;

v\_sueldo empleados.Sueldo%TYPE;

v\_sueldo\_liquido NUMBER;

BEGIN

OPEN c\_empleados;

LOOP

FETCH c\_empleados INTO v\_rut, v\_nombre, v\_cargo, v\_sueldo, v\_sueldo\_liquido;

EXIT WHEN c\_empleados%NOTFOUND;

-- Mostrar información

DBMS\_OUTPUT.PUT\_LINE('Rut: ' || v\_rut);

DBMS\_OUTPUT.PUT\_LINE('Nombre: ' || v\_nombre);

DBMS\_OUTPUT.PUT\_LINE('Cargo: ' || v\_cargo);

DBMS\_OUTPUT.PUT\_LINE('Sueldo: ' || v\_sueldo);

DBMS\_OUTPUT.PUT\_LINE('Sueldo Líquido: ' || v\_sueldo\_liquido);

DBMS\_OUTPUT.PUT\_LINE('---------------------');

END LOOP;

CLOSE c\_empleados;

END;

/