**UNIVERSIDAD TECNOLÓGICA DE PANAMÁ**

**FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES**

**DEPARTAMENTO DE SISTEMAS DE INFORMACIÓN, CONTROL Y EVALUACIÓN DE RECURSOS INFORMÁTICOS**

**LIC. EN ING. DE SISTEMAS DE INFORMACIÓN**

**SISTEMAS DE BASE DE DATOS II**

**Laboratorio N°5**

**Profesor:**

**Ing. Henry Lezcano**

**Integrantes:**

Cutire, Fernando (8-972-906)

Díaz, Gabriel (20-53-5198)

Escobar, Jorge (2-747-1772)

Feng, William (8-977-446)

**Grupo: 1IF131**

**Caso:**

La Mueblería Gratificante S. A. Necesita que los ingenieros en Sistemas implementen el proceso de cálculo de salario quincenal para sus colaboradores, actualmente cuentan con modelo físico que será proporcionado para esta tarea.

Colaboradores (id\_codcolaborador number pk, nombre varchar2(25), apellido varchar2(25), cedula varchar2(12), sexo char, fecha\_nacimiento date, fecha\_ingreso date, status char, salario\_mensual number(15,2))

salario\_quincenal ( id\_salario number pk , id\_codcolaborador number fk, fecha\_pago date, salario\_quincenal number(15,2), seguro\_social number(15,2), seguro\_educativo number(15,2), salario\_neto number(15,2))

Las fechas de pago son los 15 y 30 de cada mes. La información que mantienen en la base de datos incluye a colaboradores con status (A= Activo, R=Retirado y V=Vacaciones), las quincenas solo son pagadas a los colaboradores que están activos.

Se exige que en el proceso a implementar se utilice un cursor que busque en la base de datos la información de los colaboradores y que dentro del proceso a implementar cargue la tabla de los salarios quincenales con la información correspondiente a los cálculos del salario para cada colaborador.

Para calcular el seguro social, este es igual al salario \* 9.75%. Para el cálculo del seguro educativo este el igual salario \* 1.25%. Recuerde que este pago del salario quincenal dentro del proceso.

El Procedimiento Almacenado controla el proceso principal, las Funciones Almacenadas deberán ser diseñadas para el cálculo del seguro social y el seguro educativo correspondiente. La invocación la hará el usuario desde un bloque anónimo

Por último muestra una vista de cumplimiento de lo solicitado:



Código

set serveroutput on;

--TABLA COLABORADORES--

create table colaboradores (

id\_codcolaborador *number* not null,

nombre *varchar2*(25) not null,

apellido *VARCHAR2*(25)not NULL,

cedula *varchar2*(12) not NULL,

sexo *char* not NULL,

fecha\_nacimiento *date* not NULL,

fecha\_ingreso *date* not NULL,

status *char* not NULL,

salario\_mensual *number*(15,2) DEFAULT 0,

constraint n\_cedula unique (cedula),

constraint c\_sexo CHECK (sexo in ('F','M')),

constraint c\_status CHECK (status in ('A', 'V', 'R')),

constraint pk\_colab\_cod primary key (id\_codcolaborador)

);

--TABLA SALARIO QUINCENAL--

CREATE TABLE salario\_quincenal (

id\_salario *NUMBER* not null,

id\_codcolaborador *number* not NULL,

fecha\_pago *date* NOT NULL,

salario\_quincenal *number*(15,2) DEFAULT 0,

seguro\_social *number*(15,2) DEFAULT 0,

seguro\_educativo *number*(15,2) DEFAULT 0,

salario\_neto *NUMBER*(15,2) DEFAULT 0,

constraint pk\_id\_salario primary key (id\_salario),

constraint fk\_codcolaborador foreign key (id\_codcolaborador)

references colaboradores (id\_codcolaborador)

);

--VISTA DE COLABORADORES Y PAGO QUINCENAL--

CREATE VIEW Salario\_Quincenal\_vista AS

SELECT

c.id\_codcolaborador AS codigo,

CONCAT(CONCAT(c.nombre, ' ' ), c.apellido) AS colaborador,

c.salario\_mensual,

sq.salario\_quincenal, sq.seguro\_social,

sq.seguro\_educativo,

sq.salario\_neto AS salario\_neto

FROM colaboradores c

INNER JOIN salario\_quincenal sq ON c.id\_codcolaborador = sq.id\_codcolaborador

WHERE status = 'A'

ORDER BY c.id\_codcolaborador ASC;

--SECUENCIAS DE ID DE TABLAS COLABORADORES--

CREATE SEQUENCE sec\_id\_colaborador

INCREMENT BY 1

START WITH 1

MAXVALUE 99999

MINVALUE 1;

--SECUENCIA ID DE SALARIO\_QUINCENAL

CREATE SEQUENCE sec\_id\_salario

INCREMENT BY 1

START WITH 1

MAXVALUE 99999

MINVALUE 1;

-- PROCEDIMIENTO DE INSERCION DE USUARIO--

CREATE or REPLACE PROCEDURE insertNewColab(

p\_Nombre\_Colab IN colaboradores.nombre%TYPE,

p\_Apellido\_Colab IN colaboradores.apellido%TYPE,

p\_cedula\_colab IN colaboradores.cedula%TYPE,

p\_sexo\_colab IN COLABORADORES.SEXO%TYPE,

p\_fecha\_nac IN colaboradores.fecha\_nacimiento%TYPE,

p\_Status\_colab IN colaboradores.status%TYPE,

p\_SalarioM\_Colab IN colaboradores.salario\_mensual%TYPE)

IS

intSeqVal *number*(10);

v\_fechaIn *date*;

BEGIN

select SEC\_ID\_COLABORADOR.nextval into intSeqVal from dual;

select SYSDATE into v\_fechaIn from DUAL;

INSERT into COLABORADORES (id\_codcolaborador,nombre,apellido,cedula,sexo,

fecha\_nacimiento,fecha\_ingreso,status,salario\_mensual)

VALUES (intSeqVal, p\_Nombre\_Colab, p\_Apellido\_Colab, p\_cedula\_colab, p\_sexo\_colab,

to\_date(p\_fecha\_nac,'DD-MON-YY'),v\_fechaIn, p\_Status\_colab, p\_SalarioM\_Colab);

COMMIT;

EXCEPTION

WHEN DUP\_VAL\_ON\_INDEX THEN

DBMS\_OUTPUT.PUT\_LINE('💣 Error: El usuario ya existe la tabla de colaboradores');

END;

/

-- PARAMETROS: NOMBRE,APELLIDO,CEDULA,SEXO,FECHA NACIMIENTO,STATUS,SALARIO

EXECUTE insertNewColab('GABRIEL','DIAZ','7-125-778','M','14-FEB-21','A',8000.00);

EXECUTE insertNewColab('gabriel','diaz','20-53-5198','M','01-OCT-00','A',5000);

EXECUTE insertNewColab('william','feng','8-977-446','M','07-OCT-00','A',5000);

EXECUTE insertNewColab('jorge','escobar','2-747-1772','M','04-AUG-00','A',5000);

EXECUTE insertNewColab('Esperanza','Ordonez','8-972-906','F','06-JUL-00','A',5000);

EXECUTE insertNewColab('Carol','Santo','8-910-123','F','14-FEB-90','V',4000);

-- FUNCIÓN CALCULAR SALARIO QUINCENAL

CREATE OR REPLACE FUNCTION Cal\_salarioQuincenal (

p\_SalarioM\_colab COLABORADORES.salario\_mensual%TYPE)

RETURN *NUMBER* IS

V\_SALARIO\_QUINCENAL *NUMBER*;

v\_SalarioM\_Colab *NUMBER* := p\_SalarioM\_Colab;

BEGIN

-- Salario quincenal

V\_SALARIO\_QUINCENAL := v\_SalarioM\_Colab/2;

RETURN V\_SALARIO\_QUINCENAL;

EXCEPTION

WHEN ZERO\_DIVIDE THEN

DBMS\_OUTPUT.PUT\_LINE('División invalida');

END Cal\_salarioQuincenal;

/

-- FUNCIÓN CALCULAR SEGURO EDUCATIVO

CREATE OR REPLACE FUNCTION Cal\_seguroEducativo (

p\_SalarioM\_colab COLABORADORES.salario\_mensual%TYPE)

RETURN *NUMBER* IS

V\_SEGURO\_EDUCATIVO *NUMBER*;

v\_SalarioM\_Colab *NUMBER* := p\_SalarioM\_Colab;

BEGIN

-- Seguro educativo

V\_SEGURO\_EDUCATIVO := (v\_SalarioM\_Colab/2) \* 0.0125;

RETURN V\_SEGURO\_EDUCATIVO;

EXCEPTION

WHEN ZERO\_DIVIDE THEN

DBMS\_OUTPUT.PUT\_LINE('División invalida');

END Cal\_seguroEducativo;

/

-- FUNCIÓN CALCULAR SEGURO SOCIAL

CREATE OR REPLACE FUNCTION Cal\_seguroSocial (

p\_SalarioM\_colab COLABORADORES.salario\_mensual%TYPE)

RETURN *NUMBER* IS

V\_SEGURO\_SOCIAL *NUMBER*;

v\_SalarioM\_Colab *NUMBER* := p\_SalarioM\_Colab;

BEGIN

-- Seguro social

V\_SEGURO\_SOCIAL := (v\_SalarioM\_Colab/2) \* 0.0975;

RETURN V\_SEGURO\_SOCIAL;

EXCEPTION

WHEN ZERO\_DIVIDE THEN

DBMS\_OUTPUT.PUT\_LINE('División invalida');

END Cal\_seguroSocial;

/

-- FUNCION CALCULAR SALRIO NETO

CREATE OR REPLACE FUNCTION Cal\_salarioNeto (

p\_SalarioM\_colab COLABORADORES.salario\_mensual%TYPE)

RETURN *NUMBER* IS

V\_SALARIO\_NETO *NUMBER*;

v\_SalarioM\_Colab *NUMBER* := p\_SalarioM\_Colab;

BEGIN

-- Salario Neto

V\_SALARIO\_NETO := (v\_SalarioM\_Colab/2) - (v\_SalarioM\_Colab/2 \* 0.0975) - (v\_SalarioM\_Colab/2 \* 0.0125);

RETURN V\_SALARIO\_NETO;

WHEN ZERO\_DIVIDE THEN

DBMS\_OUTPUT.PUT\_LINE('División invalida');

END Cal\_salarioNeto;

/

-- PROCEDIMIENTO: CALCULO DE NOMINA --

CREATE OR REPLACE PROCEDURE Nomina(

P\_FECHAPAGO *DATE*)

AS

v\_Id\_Colab colaboradores.id\_codcolaborador%TYPE;

v\_SalarioM\_Colab colaboradores.salario\_mensual%TYPE;

v\_Status\_colab colaboradores.status%TYPE := 'A';

v\_intSeqVal *NUMBER*;

CURSOR c\_Salarios IS

SELECT id\_codcolaborador,

salario\_mensual

FROM COLABORADORES

WHERE status = v\_Status\_colab;

BEGIN

-- Este código se emplea los días quince y treinta de cada mes.

-- Si se quiere probar el código, cambiar los valores a el día en la que usted se encuentra

-- Por ejemplo: Si usted lo prueba el 6 de octubre, colocar 06 en el date

-- Ejem: IF to\_char(CURRENT\_DATE, 'dd') = '06' OR to\_char(CURRENT\_DATE, 'dd') = '30' THEN

IF to\_char(P\_FECHAPAGO, 'dd') = '15' OR to\_char(P\_FECHAPAGO, 'dd') = '30' THEN

OPEN c\_Salarios;

LOOP

FETCH c\_Salarios INTO

v\_Id\_Colab,

v\_SalarioM\_Colab;

EXIT

WHEN c\_salarios%NOTFOUND;

select SEC\_ID\_SALARIO.nextval into v\_intSeqVal from dual;

INSERT INTO salario\_quincenal (

id\_salario,id\_codcolaborador, fecha\_pago, salario\_quincenal,

seguro\_social, seguro\_educativo, salario\_neto)

VALUES (

v\_intSeqVal,

v\_Id\_Colab,

SYSDATE(),

Cal\_salarioQuincenal(v\_SalarioM\_Colab),

Cal\_seguroEducativo(v\_SalarioM\_Colab),

Cal\_seguroSocial(v\_SalarioM\_Colab),

Cal\_salarioNeto(v\_SalarioM\_Colab)

);

END LOOP;

CLOSE c\_Salarios;

COMMIT;

ELSE

DBMS\_OUTPUT.PUT\_LINE('💣 Error: Hoy no es día de pago.');

END IF;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('💣 Error: Este ID no existe.');

END NOMINA;

/

--LLAMADA AL PROCEDIMIENTO DE PAGO QUINCENAL

DECLARE

V\_FECHA\_PAGO *DATE*;

BEGIN

V\_FECHA\_PAGO := '&FECHA';

NOMINA(V\_FECHA\_PAGO);

DBMS\_OUTPUT.PUT\_LINE('Pago de nómina quincenal realizado correctamente');

END;

/

## Resultado

