**UNIVERSIDAD TECNOLÓGICA DE PANAMÁ**



**FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES**

**DEPARTAMENTO DE SISTEMAS DE INFORMACIÓN,**

**CONTROL Y EVALUACIÓN DE RECURSOS INFORMÁTICOS**

**LICENCIATURA EN INGENIERÍA DE SISTEMAS DE INFORMACIÓN**

**SISTEMAS DE BASES DE DATOS ll.**

**LABORATORIO 4**

**Profesor**

**Henry Lezcano**

**Estudiantes**

Solis, Michael 8-958-1219

Sánchez, Ana 8-967-832

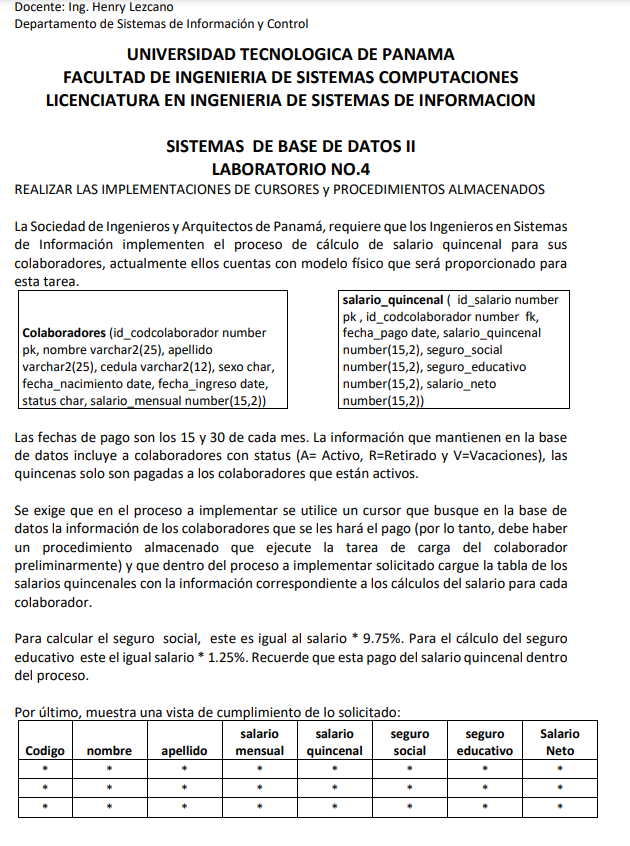
Tejada, Ángel 8-969-974

Samudio, Nedith 8-968-1471

Urriola, Vicente 8-892-2296

**Grupo**

**1IF131**



create table colaboradores(

id\_codcolaborador number,

nombre varchar2(25) not null,

apellido varchar2(25) not null,

cedula varchar2(12) not null unique,

sexo char not null check (sexo IN('M', 'F')),

fecha\_nacimiento date not null,

fecha\_ingreso date not null,

status char not null check (status IN('A', 'V', 'R')),

salario\_mensual number(15,2) default 0,

constraint pk\_colaboradores primary key (id\_codcolaborador)

);

create table salario\_quincenal(

id\_salario number primary key,

id\_codcolaborador1 number not null,

fecha\_pago date not null,

salario\_quincenal number(15,2) default 0 not null,

seguro\_social number(15,2) default 0 not null,

seguro\_educativo number(15,2) default 0 not null,

salario\_neto number(15,2) default 0 not null,

constraint fk\_colaborador\_ foreign key(id\_codcolaborador1) references

colaboradores(id\_codcolaborador)

);

create sequence salario\_sequence start with 1 increment by 1 cache 10 maxvalue

10000 nocycle;

create or replace procedure agregar\_col(

p\_id\_cod\_col colaboradores.id\_codcolaborador%TYPE,

p\_nombre colaboradores.nombre%TYPE,

p\_apellido colaboradores.apellido%TYPE,

p\_cedula colaboradores.cedula%TYPE,

p\_sexo colaboradores.sexo%TYPE,

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

p\_fecha\_ingreso colaboradores.fecha\_ingreso%TYPE,

p\_status colaboradores.status%TYPE,

p\_salario\_men colaboradores.salario\_mensual%TYPE)

as

begin

insert into colaboradores (id\_codcolaborador, nombre, apellido, cedula, sexo,

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

values (p\_id\_cod\_col, p\_nombre, p\_apellido, p\_cedula, p\_sexo, p\_fecha\_nac,

p\_fecha\_ingreso, p\_status, p\_salario\_men);

end agregar\_col;

begin

agregar\_col(1, 'Carlos','Solis','8-953-756','M','21-JUN-99','14-MAY-21','A',800.0);

agregar\_col(2, 'Julia','Sanjur','8-979-953','F','21-OCT-99','14-FEB-21','A',920.55);

agregar\_col(3, 'Raul','Jimenez','9-964-1234','M','11-OCT-98','14-JUL21','A',1310.00);

agregar\_col(4, 'Pablo','Escobar','9-784-1234','M','11-OCT-99','18-JUL21','A',1510.00);

end;

-- Función salario quincenal

create or replace function quincenal(p\_salario\_men IN colaboradores.salario\_mensual%type)

return number as

begin

return p\_salario\_men/2;

end quincenal;

-- función de seguro\_social

create or replace function seguro\_social(p\_salario\_men IN colaboradores.salario\_mensual%type)

return number as

begin

return ((p\_salario\_men/2) \* 0.0975);

end seguro\_social;

-- función de seguro\_educativo

create or replace function seguro\_educativo(p\_salario\_men IN colaboradores.salario\_mensual%type)

return number as

begin

return (p\_salario\_men/2 \* 0.0125);

end seguro\_educativo;

-- función de neto

create or replace function neto(p\_salario\_men IN colaboradores.salario\_mensual%type)

return number as

begin

return (p\_salario\_men/2) - (((p\_salario\_men/2) \* 0.0975) + (p\_salario\_men/2 \* 0.0125));

end neto;

declare

-- Declaración

v\_colaboradorID colaboradores.id\_codcolaborador%type;

v\_salarioMensual colaboradores.salario\_mensual%type;

v\_salarioQuincenal salario\_quincenal.salario\_quincenal%type;

v\_seguroSocial salario\_quincenal.seguro\_social%type;

v\_seguroEducativo salario\_quincenal.seguro\_educativo%type;

v\_salarioNeto salario\_quincenal.salario\_neto%type;

--Declarar Cursor

cursor c\_Colaboradores is

select id\_codcolaborador, salario\_mensual

from colaboradores

where status = 'A';

--Empieza el siguiente bloque

begin

--Ver si es día de pago

if to\_char(CURRENT\_DATE, 'dd') = '15' or to\_char(CURRENT\_DATE, 'dd')= '30'

then

open c\_Colaboradores;

loop

fetch c\_Colaboradores into v\_colaboradorID, v\_salarioMensual;

exit when c\_Colaboradores%NOTFOUND;

v\_salarioQuincenal := quincenal(v\_SalarioMensual);

v\_seguroSocial := seguro\_social(v\_salarioMensual);

v\_seguroEducativo := seguro\_educativo(v\_salarioMensual);

v\_salarioNeto := neto(v\_salarioMensual);

--v\_seguroSocial := v\_salarioQuincenal \* 0.0975;

--seguro\_social();

--v\_seguroEducativo := v\_salarioQuincenal \* 0.0125;

--v\_salarioNeto := v\_salarioQuincenal - (v\_seguroSocial + v\_seguroEducativo);

insert into salario\_quincenal values(salario\_sequence.nextval, v\_colaboradorID, CURRENT\_DATE, v\_salarioQuincenal, v\_seguroSocial, v\_seguroEducativo, v\_salarioNeto);

end loop;

close c\_Colaboradores;

else

dbms\_output.put\_line('No es día de pago');

end if;

exception

when NO\_DATA\_FOUND then

DBMS\_OUTPUT.PUT\_LINE('Este ID (identificador) no existe');

end;

create view Ver as

select c.id\_codcolaborador as codigo, c.nombre,

c.apellido , c.salario\_mensual, sk.salario\_quincenal,

sk.seguro\_social , sk.seguro\_educativo, sk.salario\_neto as salario\_Neto

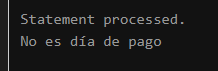
from colaboradores c

inner join salario\_quincenal sk on c.id\_codcolaborador = sk.id\_codcolaborador1

where status = 'A'

order by c.id\_codcolaborador asc;

select \* from ver;



Resultado mostrado cuando el día es del 15 al 30.

Vista de usuario:

