



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 II.

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

UNIVERSIDAD TECNOLÓGICA DE PANAMA
FACULTAD DE INGENIERIA DE SISTEMAS COMPUTACIONES
LICENCIATURA EN INGENIERIA DE SISTEMAS DE INFORMACION

SISTEMAS DE BASE DE DATOS II
LABORATORIO NO.4

REALIZAR LAS IMPLEMENTACIONES DE CURSORES Y PROCEDIMIENTOS ALMACENADOS

La Sociedad de Ingenieros y Arquitectos de Panamá, requiere que los Ingenieros en Sistemas de Información implementen el proceso de cálculo de salario quincenal para sus colaboradores, actualmente ellos 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 que se les hará el pago (por lo tanto, debe haber un procedimiento almacenado que ejecute la tarea de carga del colaborador preliminarmente) y que dentro del proceso a implementar solicitado 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 es igual salario * 1.25%. Recuerde que esta pago del salario quincenal dentro del proceso.

Por último, muestra una vista de cumplimiento de lo solicitado:

Codigo	nombre	apellido	salario mensual	salario quincenal	seguro social	seguro educativo	Salario Neto
*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*

```
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-
JUL 21','A',1310.00);
agregar_col(4, 'Pablo','Escobar','9-784-1234','M','11-OCT-99','18-
JUL 21','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;

```

```
Statement processed.  
No es día de pago
```

Resultado mostrado cuando el día es del 15 al 30.

Vista de usuario:

CODIGO	NOMBRE	APELLIDO	SALARIO_MENSUAL	SALARIO_QUINCENAL	SEGURO_SOCIAL	SEGURO_EDUCATIVO	SALARIO_NETO
1	Carlos	Solis	800	400	39	5	356
2	Julia	Sanjur	920.55	460.28	44.88	5.75	409.65
3	Raul	Jimenez	1310	655	63.86	8.19	582.95
4	Pablo	Escobar	1510	755	73.61	9.44	671.95

[Download CSV](#)
4 rows selected.