create or replace procedure sp\_principal(p\_fecha date)

is

v\_porc number;

v\_bono\_antiguedad number;

v\_comision number;

begin

for x in (select e.nombre\_emp,e.sueldo\_base\_emp,c.nombre\_comuna

,trunc((sysdate-e.fecing\_emp)/365) as antiguedad,e.numrut\_emp

from empleado e join comuna c on e.id\_comuna = c.id\_comuna)

loop

dbms\_output.put\_line('nombre:' || x.nombre\_emp);

v\_porc:=fn\_porc\_movilizacion(x.sueldo\_base\_emp, x.nombre\_comuna);

dbms\_output.put\_line('Valor de Movilizacion:' || v\_porc);

v\_bono\_antiguedad:= fn\_bonif\_annos\_contrato(x.antiguedad, x.sueldo\_base\_emp);

dbms\_output.put\_line('Bono Antiguedad:' || v\_bono\_antiguedad);

v\_comision:= fn\_comision(x.numrut\_emp,p\_fecha);

dbms\_output.put\_line('Comision:' || v\_comision);

end loop;

end;

set serveroutput on;

execute sp\_principal('01/04/2018');

------------------- funciones de apoyo

-- funcion que recupera porcentaje de movilizacion

create or replace function fn\_porc\_movilizacion(p\_sueldo number,p\_comuna varchar2)

return number

is

v\_porc number;

v\_adicional number:=0;

begin

v\_adicional:= fn\_adicional\_mov(p\_comuna);

select porc\_mov into v\_porc from porc\_movilizacion

where p\_sueldo between sueldo\_base\_inf and sueldo\_base\_sup;

RETURN (p\_sueldo\* (v\_porc/100))+v\_adicional;

exception

when others then

RETURN 0+v\_adicional;

end;

-----------------------------

create table porc\_movilizacion(

sec\_movilizacion number primary key,

sueldo\_base\_inf number,

sueldo\_base\_sup number,

porc\_mov number

);

select \* from porc\_movilizacion;

insert into porc\_movilizacion values(1, 100000, 300000,50);

insert into porc\_movilizacion values(2, 300001,1000000,30);

insert into porc\_movilizacion values(3,1000001,2500000,20);

insert into porc\_movilizacion values(4,2500001,8000000, 5);

declare

v\_porc number;

begin

v\_porc:=fn\_porc\_movilizacion(50000);

dbms\_output.put\_line('Porc movi:' || v\_porc);

end;

select numrut\_emp,sueldo\_base\_emp

,fn\_porc\_movilizacion(sueldo\_base\_emp)

from empleado;

---- adicional movilizacion

create or replace function fn\_adicional\_mov(p\_comuna varchar2)

return number

is

begin

if p\_comuna in('La Pintana', 'Cerro Navia','Peñalolén') then

return 25000;

end if;

if p\_comuna in('Melipilla','María Pinto','Curacaví','Talagante','Isla de Maipo','Paine') then

return 40000;

end if;

return 0;

end;

----------------------- prueba

declare

v\_adicional\_mov number;

begin

v\_adicional\_mov:=fn\_adicional\_mov('Melipilla');

dbms\_output.put\_line('Adicional Mov.:' || v\_adicional\_mov);

end;

------------------------ asignacion por años contratado (antiguedad)

create or replace function fn\_bonif\_annos\_contrato(p\_antiguedad number,p\_sueldo number)

return number

is

v\_porc number;

begin

select porc\_bonif into v\_porc from porc\_bonif\_annos\_contrato

where p\_antiguedad between annos\_inferior and annos\_superior;

RETURN p\_sueldo \* v\_porc;

exception

when others then

RETURN 0;

end;

----- probar

declare

v\_bonificacion number;

begin

v\_bonificacion:= fn\_bonif\_annos\_contrato(10,1500000);

dbms\_output.put\_line('Bono Antiguedad:' || v\_bonificacion);

end;

--------------------------------------- determinar comision venta

create or replace function fn\_comision(p\_rut varchar2,p\_fecha date)

return number

is

v\_comision number;

begin

select sum(cv.valor\_comision) into v\_comision from comision\_venta cv

join boleta b on cv.nro\_boleta = b.nro\_boleta

where b.numrut\_emp=p\_rut

and to\_char(b.fecha\_boleta,'MM/yyyy')=to\_char(p\_fecha,'MM/yyyy');

return v\_comision;

end;

-- probar

declare

v\_comision number;

begin

v\_comision:=fn\_comision('12113369','01/04/2018');

dbms\_output.put\_line('Suma Comisiones:' || v\_comision);

end;

select \* from boleta b

join comision\_venta cv on cv.nro\_boleta = b.nro\_boleta

order by b.numrut\_emp, b.fecha\_boleta;