

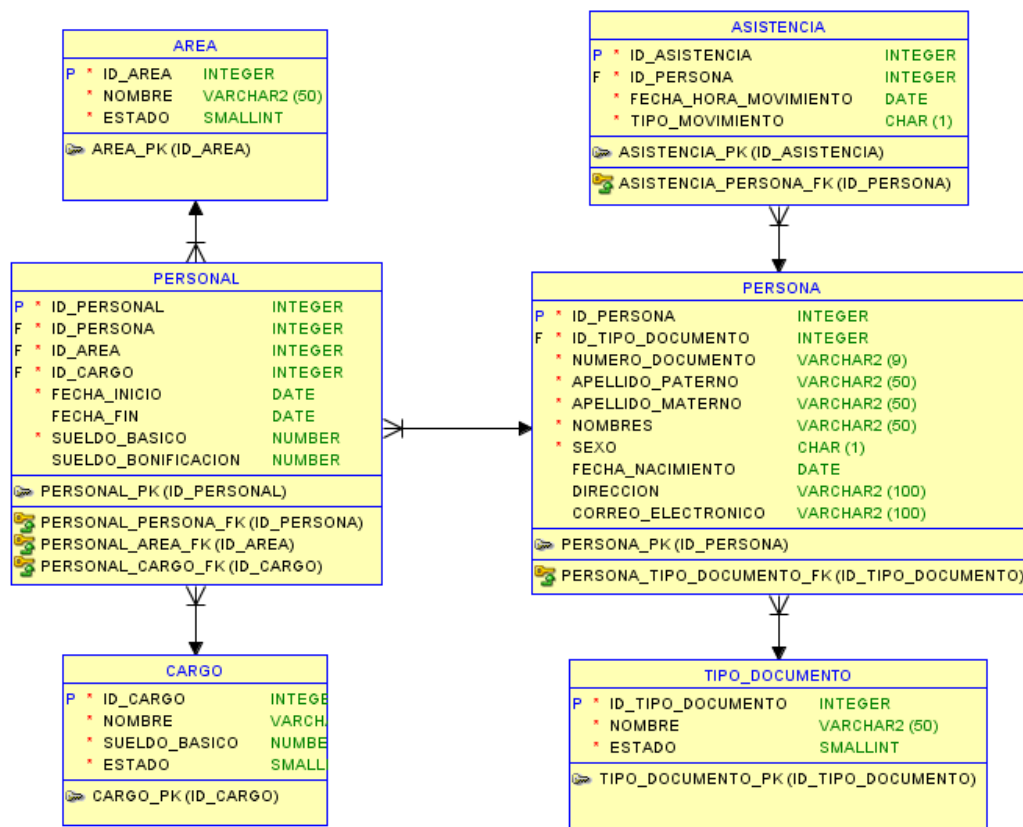
PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA
Bases de Datos
4ta. Práctica Dirigida
(Primer Semestre 2021)

Nota: Para este laboratorio, puede tomar como referencia la guía del laboratorio anterior. No se olvide de crear una conexión usando los siguientes datos de configuración:

- Nombre de conexión (Name): LAB4_DIRIGIDA
- Usuario: system
- Contraseña: Contraseña inscrita en la instalación de Oracle Database Express Edition.

Tema propuesto

La empresa Rent SAC es una empresa emergente (startup) recién formada que ha conseguido financiamiento y colaboración de personas de distintas partes del mundo. Desea mantener el registro de su personal en un Sistema de Recursos Humanos elaborado por su propio personal. Para este fin, se elaboró una base de datos cuyo modelo de datos se muestra a continuación:



Ejecute el contenido del archivo **INF246_2021-1_LAB4_Dir_Inicial.sql**, disponible en la tarea de la parte dirigida del presente laboratorio en PAIDEIA. Esto permitirá crear las tablas de la base de datos e insertar algunos datos en ellas.

Ejercicio 1

Elaborar una función que reciba como parámetro un sexo (valores 'F' o 'M') y que devuelva la cantidad de personas que tienen dicho sexo. Si el valor del parámetro no es correcto la función deberá retornar -1.

```
create or replace function f_contar_personas_por_sexo( v_sexo char )
return number
as
    v_cantidad number;
begin
    if v_sexo <> 'M' and v_sexo <> 'F' then
        return -1;
    else
        select count(*)
        into v_cantidad
        from PERSONA
        where sexo = v_sexo;
        return v_cantidad;
    end if;
end;
```

set SERVEROUTPUT ON

declare

v_cantidad number;

v_sexo char;

begin

v_sexo := 'F';

v_cantidad := f_contar_personas_por_sexo(v_sexo);

if v_cantidad <> -1 then

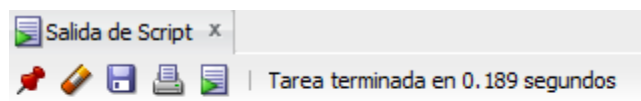
dbms_output.put_line('Hay ' || v_cantidad || ' personas de sexo ' || v_sexo);

else

dbms_output.put_line('Sexo no válido');

end if;

end;



Hay 3 personas de sexo F

Procedimiento PL/SQL terminado correctamente.

Ejercicio 2

- a) Elaborar una función que reciba como parámetro una fecha y que devuelva dicha fecha en forma de frase. Por ejemplo, si la fecha que recibe la función es '21/6/2021', está devolverá la frase '21 de junio de 2021'.
- b) Elaborar una función que reciba como parámetro una fecha de nacimiento y que devuelva la edad correspondiente.

```
create or replace function f_fecha_frase( v_fecha date )
return varchar2
is
    v_d number;
    v_m number;
    v_a number;
    v_dia varchar2(2);
    v_mes varchar2(20);
    v_año varchar2(4);
    v_frase varchar2(30);
begin
    v_d := extract( day from v_fecha );
    v_m := extract( month from v_fecha );
    v_a := extract( year from v_fecha );
    case v_m
        when 1 then v_mes := 'enero';
        when 2 then v_mes := 'febrero';
        when 3 then v_mes := 'marzo';
        when 4 then v_mes := 'abril';
        when 5 then v_mes := 'mayo';
        when 6 then v_mes := 'junio';
        when 7 then v_mes := 'julio';
        when 8 then v_mes := 'agosto';
        when 9 then v_mes := 'setiembre';
        when 10 then v_mes := 'octubre';
        when 11 then v_mes := 'noviembre';
        else v_mes := 'diciembre';
    end case;
    v_dia := to_char( v_d );
    v_año := to_char( v_a );
    v_frase := v_dia || ' de ' || v_mes || ' de ' || v_año;
    return v_frase;
end;
```

```
create or replace function f_obtener_edad( v_fecha_nacimiento date )
return number
is
    v_meses number;
    v_edad number;
```

```

begin
    v_meses := months_between( sysdate, v_fecha_nacimiento );
    v_edad := trunc( v_meses / 12 );
    return v_edad;
end;

```

```

select apellido_paterno || ' ' || apellido_materno || ' ' || nombres as Empleado,
       f_fecha_frase( fecha_nacimiento) as Fecha_Nacimiento,
       f_obtener_edad( fecha_nacimiento ) as Edad
from PERSONA;

```

Resultado de la Consulta x

Todas las Filas Recuperadas: 12 en 0.011 segundos

	EMPLEADO	FECHA_NACIMIENTO	EDAD
1	RODRIGUEZ ORTEGA DANIEL MANUEL	26 de diciembre de 1958	62
2	CHAVEZ TAVERA BRUNO MARTIN	16 de setiembre de 1967	53
3	SANCHEZ CARLOS JULIO ROBERTO	9 de mayo de 1966	55
4	CARDOSO MARTINEZ FRANCISCO MIGUEL	31 de mayo de 1980	41
5	RODRIGUEZ RUIZ EDDIE FRANCISCO	28 de marzo de 1975	46
6	PISCONTE DE LA CRUZ SANDRA VIVIANA	1 de noviembre de 1988	32
7	APARICIO MANCO CARLA PATRICIA	26 de octubre de 1981	39
8	DEZA RIVERA ERICK TIMOTEO	12 de abril de 1979	42
9	ROMERO RAMIREZ DIEGO ANTONIO	14 de enero de 1989	32
10	TELLO MELENDEZ JOSE ALBERTO	6 de diciembre de 1992	28
11	VERA CORDOVA JORGE FELIX	26 de octubre de 1980	40
12	RODRIGUEZ VERA FRANCISCA	1 de agosto de 1996	24

Ejercicio 3

Elaborar un procedimiento que reciba como parámetro el nombre de un cargo y que muestre en pantalla la cantidad de personas que han tenido o tienen dicho cargo.

```

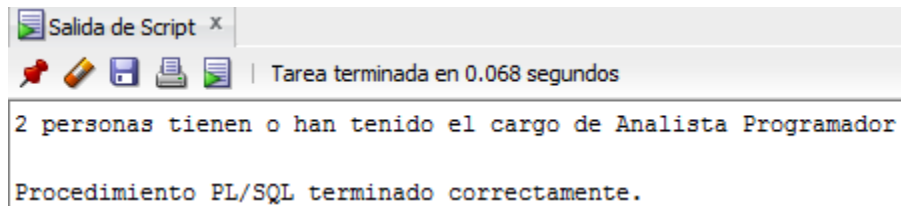
create or replace procedure p_cantidad_personas_cargo( v_nom_cargo
                                                         varchar2 )
is
    v_cantidad number;
begin
    select count(*) into v_cantidad
    from PERSONAL P, CARGO C
    where P.id_cargo = C.id_cargo and
          C.nombre = v_nom_cargo;
    dbms_output.put_line( v_cantidad || ' personas tienen o han tenido
                           el cargo de ' || v_nom_cargo );
end;

```

```

set SERVEROUTPUT ON
exec p_cantidad_personas_cargo( 'Analista Programador' );

```



```
Salida de Script x
Tarea terminada en 0.068 segundos

2 personas tienen o han tenido el cargo de Analista Programador

Procedimiento PL/SQL terminado correctamente.
```

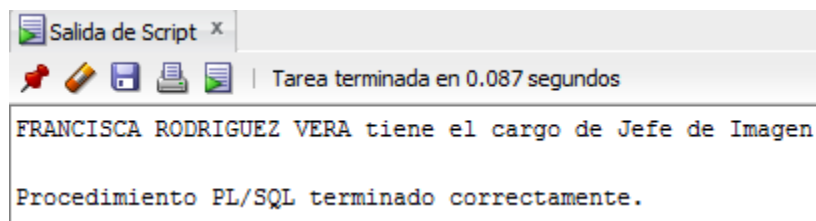
Ejercicio 4

Elaborar un procedimiento que reciba como parámetros el apellido paterno, apellido materno y nombre de una persona; y que muestre el cargo de la persona en la empresa. Si la persona no existe o no tiene un cargo asignado se debe mostrar un mensaje apropiado.

```
create or replace procedure p_mostrar_cargo_persona
( v_ape_paterno varchar2,
  v_ape_materno varchar2,
  v_nombre varchar2 )
is
  v_nom_cargo varchar2(50);
begin
  select nombre into v_nom_cargo
  from PERSONA P, PERSONAL PR, CARGO C
  where P.id_persona = PR.id_persona and
        PR.id_cargo = C.id_cargo and
        P.apellido_paterno = v_ape_paterno and
        P.apellido_materno = v_ape_materno and
        P.nombres = v_nombre;
  dbms_output.put_line( v_nombre || ' ' || v_ape_paterno || ' ' ||
                        v_ape_materno || ' tiene el cargo de ' || v_nom_cargo );
exception
  when NO_DATA_FOUND then
    dbms_output.put_line( 'Persona no existe o no tiene cargo
                          asignado' );
end;
```

set SERVEROUTPUT ON

exec p_mostrar_cargo_persona('RODRIGUEZ', 'VERA', 'FRANCISCA');



```
Salida de Script x
Tarea terminada en 0.087 segundos

FRANCISCA RODRIGUEZ VERA tiene el cargo de Jefe de Imagen

Procedimiento PL/SQL terminado correctamente.
```

exec p_mostrar_cargo_persona('MORALES', 'TORRES', 'ELVIRA');

```
Salida de Script x
Tarea terminada en 0.09 segundos

Persona no existe o no tiene cargo asignado

Procedimiento PL/SQL terminado correctamente.
```

Ejercicio 5

Elaborar un procedimiento que reciba como parámetros el nombre de un área de la empresa y un monto de dinero. En base a ello se debe incrementar en ese monto el sueldo básico al personal que pertenezca a esa área cuya fecha de fin de contrato sea indefinida.

```
create or replace procedure p_incrementar_sueldo_basico
( v_nom_area varchar2, v_monto number )
is
  v_id_area number;
begin
  select id_area into v_id_area
  from AREA where nombre = v_nom_area;
  update PERSONAL
  set sueldo_basico = sueldo_basico + v_monto
  where id_area = v_id_area and fecha_fin is null;
exception
  when NO_DATA_FOUND then
    dbms_output.put_line( 'Area ' || v_nom_area || ' no existe' );
end;
```

Por ejemplo, el área de Imagen Institucional tiene id_area con valor 4. Observamos el sueldo del personal que trabaja en esta área:

```
select * from PERSONAL where id_area = 4;
```

Resultado de la Consulta x								
SQL Todas las Filas Recuperadas: 2 en 0.004 segundos								
	ID_PERSONAL	ID_PERSONA	ID_AREA	ID_CARGO	FECHA_INICIO	FECHA_FIN	SUELDO_BASICO	SUELDO_BONIFICACION
1	10	11	4	11	01/01/20	30/06/22	3500	(null)
2	11	12	4	4	01/01/20	(null)	13000	(null)

Ejecutamos el procedimiento:

```
exec p_incrementar_sueldo_basico( 'Imagen Institucional' , 200 );
```

```
Salida de Script x
Tarea terminada en 0.075 segundos

Procedimiento PL/SQL terminado correctamente.
```

Observamos nuevamente el sueldo del personal de dicha área:

```
select * from PERSONAL where id_area = 4;
```

Resultado de la Consulta x								
SQL Todas las Filas Recuperadas: 2 en 0.003 segundos								
	ID_PERSONAL	ID_PERSONA	ID_AREA	ID_CARGO	FECHA_INICIO	FECHA_FIN	SUELDO_BASICO	SUELDO_BONIFICACION
1	10	11	4	11	01/01/20	30/06/22	3500	(null)
2	11	12	4	4	01/01/20	(null)	13200	(null)

Se observa que se ha incrementado 200 soles solo a la persona que pertenece a esta área y que tiene fecha de fin de contrato indefinida (Fecha_Fin = null).

Ejercicio 6

Elaborar un procedimiento que devuelva dos valores: el menor sueldo básico y la cantidad de personas que tienen ese sueldo. Solo se deben considerar a aquellas personas cuyo contrato tiene una fecha de finalización.

```
create or replace procedure p_obtener_menor_sueldo_basico(
                                v_menor_sueldo out number,
                                v_cant_menor_sueldo out number )
is
begin
    select min(sueldo_basico) into v_menor_sueldo
    from PERSONAL
    where fecha_fin is not null;
    select count(*) into v_cant_menor_sueldo
    from PERSONAL
    where sueldo_basico = v_menor_sueldo and fecha_fin is not null;
end;
```

```
set SERVEROUTPUT ON
```

```
declare
```

```
    v_menor_sueldo number;
```

```
    v_cant_menor_sueldo number;
```

```
begin
```

```
    p_obtener_menor_sueldo_basico( v_menor_sueldo, v_cant_menor_sueldo );
```

```
    dbms_output.put_line( 'El menor sueldo basico es ' || v_menor_sueldo );
```

```
    dbms_output.put_line( 'Hay ' || v_cant_menor_sueldo || ' personas con ese sueldo' );
```

```
end;
```

```
Salida de Script x
Tarea terminada en 0.162 segundos

El menor sueldo basico es 3500
Hay 2 personas con ese sueldo

Procedimiento PL/SQL terminado correctamente.
```