

Universidad Nacional de San Agustín
Escuela Profesional de Ciencia de la Computación
Base de Datos I – 2019.B
Práctica de Laboratorio 10 - Sección B: Procedimientos Almacenados y Funciones

I. OBJETIVOS

- Realizar la implementación de procedimientos almacenados y funciones.

II. ACTIVIDADES

Considere el siguiente análisis de Requisitos:

- Una Empresa mantiene la siguiente información:
- La empresa está organizada en departamentos. Cada departamento tiene un nombre distinto, un código distinto y un empleado en particular que la gerencia. Se registra la fecha inicial en que ese funcionario comenzó a gerenciar el departamento. Un departamento puede tener varios locales.
- Un departamento controla una serie de proyectos, cada uno de ellos con un nombre distinto, un código distinto y un local diferente.
- Almacenamos el nombre, número de DNI, dirección, sueldo, sexo (género) y fecha de nacimiento de cada empleado.
- Un empleado es designado para un departamento, pero puede trabajar en varios proyectos, que no necesariamente son controlados por el mismo departamento.
- Registramos el número actual de horas por semana que un empleado trabaja en cada proyecto. También, registramos el supervisor directo de cada empleado (que es otro empleado).
- También registramos los dependientes de cada empleado para fines del seguro. Para cada dependiente, almacenamos el nombre, sexo, fecha de nacimiento y parentesco con el empleado.

Realice y muestre los siguientes puntos:

1. Implemente la Base de Datos usando un Sistema Gestor de Base de Datos:

Tabla Proyecto:

	codigo [PK] integer	nombre character varying (100)	localp character varying (100)	dcodigo integer
1	1	Proyecto X	Oficina 317	5
2	2	Proyecto Y	Oficina 22	5
3	3	Proyecto Z	Oficina 544	1
4	10	Proyecto A	Oficina 51	4
5	20	Proyecto B	Oficina 67	1
6	30	Proyecto C	Oficina 385	4

Tabla Dependiente:

	edni [PK] integer	nombre [PK] character varying (200)	sexo character (1)	fecha_nacimiento date	parentesco character varying (50)
1	333445555	Ana	F	1967-04-03	Hija
2	333445555	Victor	M	1973-10-25	Hijo
3	333445555	Juana	F	1958-05-05	Conyugue
4	987654321	Igor	M	1952-02-29	Conyugue
5	123456789	Michael	M	1988-01-21	Hijo
6	123456789	Anabel	F	1998-12-31	Hija
7	123456789	Elizabeth	F	1957-05-05	Conyugue

Tabla Departamento_Local:

	dcodigo [PK] integer	dlocal [PK] character varying (50)
1	1	Lima
2	4	Arequipa
3	5	Puno
4	5	Trujillo
5	5	Cusco

Tabla Trabaja_en

	edni [PK] integer	pcodigo [PK] integer	horas integer
1	123456789	1	32
2	123456789	2	7
3	666884444	3	40
4	453453453	1	20
5	453453453	2	20
6	333445555	2	10
7	333445555	3	10
8	333445555	10	10
9	333445555	20	10
10	999887777	30	30
11	999887777	10	10
12	987987987	10	35
13	987987987	30	5
14	987654321	30	20

3. Aumentar el salario de todos los empleados en un 10% usando un SP.

bd1/postgres@localhost

Query Editor Query History

```
1 CREATE OR REPLACE PROCEDURE SP_Actualizar_Salarios()
2 LANGUAGE SQL
3 AS $$
4     UPDATE ehc_empleado SET salario = salario * 1.1;
5 $$;
```

bd1/postgres@localhost

Query Editor Query History

```
1 CALL SP_Actualizar_Salarios();
```

Data Output		Explain	Messages	Notifications						
	dni	nombres	papellido	sapellido	direccion	fecha_nacimiento	sexo	salario	super_dni	dcodigo
	[PK] integer	character varying (100)	character varying (100)	character varying (100)	character varying (150)	date	character (1)	numeric (10,2)	integer	integer
1	888665555	Francisco	Linares	Gomez	Calle Numero H 8	1957-11-10	M	6050.00	[null]	1
2	123456789	Juan	Perez	Rodriguez	Calle Numero A 1	1965-01-09	M	330.00	333445555	5
3	333445555	Frank	Velazquez	Flores	Calle Numero B 2	1955-12-08	M	4400.00	888665555	5
4	999887777	Alice	Jimenez	Portugal	Calle Numero C 3	1968-07-19	F	2750.00	987654321	4
5	987654321	Luisa	Santos	Ferrel	Calle Numero D 4	1951-06-20	F	473.00	888665555	4
6	666884444	Pedro	Lima	Maldonado	Calle Numero E 5	1952-09-15	M	1320.00	333445555	5
7	453453453	Daniela	Acco	Olvarez	Calle Numero F 6	1962-07-31	F	2750.00	333445555	5
8	987987987	Mateo	Vela	Marruecos	Calle Numero G 7	1979-03-29	M	2750.00	987654321	4

4. Modificar el supervisor de un determinado empleado usando un SP (recibe como parámetro el código del empleado y código del supervisor).

bd1/postgres@localhost

Query Editor
Query History

```

1 CREATE OR REPLACE PROCEDURE SP_Actualizar_Supervisor(dni_e INTEGER, dni_s INTEGER)
2 LANGUAGE SQL
3 AS $$
4     UPDATE ehc_empleado SET super_dni = dni_s WHERE dni = dni_e;
5 $$;
```

bd1/postgres@localhost

Query Editor
Query History

```

1 CALL SP_Actualizar_Supervisor(888665555, 987654321);
```

Data Output
Explain
Messages
Notifications

	dni [PK] integer	nombres character varying (100)	apellido character varying (100)	apellido character varying (100)	direccion character varying (150)	fecha_nacimiento date	sexo character (1)	salario numeric (10,2)	super_dni integer	dcodigo integer
1	123456789	Juan	Perez	Rodriguez	Calle Numero A 1	1965-01-09	M	330.00	333445555	5
2	333445555	Frank	Velazquez	Flores	Calle Numero B 2	1955-12-08	M	4400.00	888665555	5
3	999887777	Alice	Jimenez	Portugal	Calle Numero C 3	1968-07-19	F	2750.00	987654321	4
4	987654321	Luisa	Santos	Ferrel	Calle Numero D 4	1951-06-20	F	473.00	888665555	4
5	666884444	Pedro	Lima	Maldonado	Calle Numero E 5	1952-09-15	M	1320.00	333445555	5
6	453453453	Daniela	Acco	Olvarez	Calle Numero F 6	1962-07-31	F	2750.00	333445555	5
7	987987987	Mateo	Vela	Marruecos	Calle Numero G 7	1979-03-29	M	2750.00	987654321	4
8	888665555	Francisco	Linares	Gomez	Calle Numero H 8	1957-11-10	M	6050.00	987654321	1

5. Mostrar el DNI, nombre, primer apellido y fecha de nacimiento de todos los empleados que se encuentren entre las fechas de inicio y fin enviados como parámetros usando una función.

```
bd1/postgres@localhost
Query Editor  Query History
1 CREATE FUNCTION FU_Empleados_Filtro_Fechas(fecha_ini DATE, fecha_fin DATE)
2 RETURNS TABLE(dni INTEGER, nombre VARCHAR(50), papellido VARCHAR(100), fecha_nacimiento DATE) AS $$
3 SELECT dni, nombres, papellido, fecha_nacimiento FROM ehc_empleado WHERE fecha_nacimiento BETWEEN fecha_ini AND fecha_fin;
4 $$
5 LANGUAGE SQL;
```

bd1/postgres@localhost	
Query Editor	Query History
1 SELECT FU_Empleados_Filtro_Fechas('1965-01-01', '1969-12-31');	
Data Output Explain Messages Notifications	
fu_empleados_filtro_fechas record	
1	(123456789,Juan,Perez,1965-01-09)
2	(999887777,Alice,Jimenez,1968-07-19)

6. Mostrar el promedio de salario de todos los empleados usando una función.

```
bd1/postgres@localhost
Query Editor  Query History
1 CREATE OR REPLACE FUNCTION FU_Empleados_Promedio_Salario()
2 RETURNS DECIMAL(10,2) AS $$
3     SELECT ROUND(AVG(salario)::numeric,2) FROM ehc_empleado;
4 $$
5 LANGUAGE SQL;
```

```
bd1/postgres@localhost
Query Editor  Query History
1 SELECT FU_Empleados_Promedio_Salario();
```

	fu_empleados_promedio_salario	
	numeric	
1	2602.88	

III. EJERCICIOS

1. Considere 2 Sistemas de Base de Datos implementados con anterioridad y defina 2 Procedimientos Almacenados y 2 Funciones para cada uno de ellos.

IV. CONSIDERACIONES

1. Utilizar un Sistema Gestor de Base de Datos.
2. Entregue un solo archivo en formato .pdf que contenga la resolución de todos los ejercicios. El nombre del archivo debe tener el siguiente formato:
HinojosaCardenasE_BDI_2019B_SecB_Lab11.pdf
3. En el caso de que tenga acceso al aula virtual, subir su archivo hasta que termine la hora de este laboratorio. En caso contrario entregue el archivo al profesor vía USB u otro medio físico. Cuando tenga acceso al aula virtual, suba el mismo archivo entregado al profesor al aula virtual (no se considerará como retraso).

Dr. Edward Hinojosa Cárdenas