

Taller 2 Unidad 2

Sistemas Gestores de Bases de datos

Los objetivos generales de esta actividad son:

- Crear una base de datos en PostgreSQL.
- Crear tablas e inyectar datos.
- Realizar consultas para extraer información útil de la base de datos.

Tabla - Estudiantes			
Atributo	Tipo	Tamaño	Condicion
cod_e	numeric	11	PK
nom_e	character varying	40	
dir	character varying	30	
tel	numeric	10,0	
fec	date		

Tabla - Asignaturas			
Atributo	Tipo	Tamaño	Condicion
cod_a	smallint	11	PK
nom_a	character varying	45	
ih	numeric	1	

Tabla - Profesores			
Atributo	Tipo	Tamaño	Condicion
id_p	numeric	10	PK
nom_p	character varying	45	
profesion	character varying	40	

Tabla - Libros			
Atributo	Tipo	Tamaño	Condicion
id_p	numeric	10	PK
nom_p	character varying	45	
profesion	character varying	40	

Tabla - Imparte			
Atributo	Tipo	Tamaño	Condicion
id_p	numeric	10	
id_a	smallint		
horario	time		

Tabla - Inscribe			
Atributo	Tipo	Tamaño	Condicion
id_p	numeric	10	
id_a	smallint		
horario	time		
cod_e	numeric	11	
n1	numeric	1,00	
n2		1,00	
n3		1,00	

Tabla - Libros			
Atributo	Tipo	Tamaño	Condicion
isbn	numeric	10	PK
titulo	character varying	45	

Tabla - Autores			
Atributo	Tipo	Tamaño	Condicion
isbn	numeric	10	
nom_a	character varying	45	

Tabla - Ejemplares			
Atributo	Tipo	Tamaño	Condicion
isbn	numeric	10	
num_e	numeric	2	

Tabla - Referencia			
Atributo	Tipo	Tamaño	Condicion
cod_a	smallint		
isbn	numeric	10	

Tabla - Presta			
Atributo	Tipo	Tamaño	Condicion
cod_e	numeric	11	
num_e	numeric	2	
isbn	numeric	10	
f_p	date		
f_d	date		

Información de atributos

Tabla Estudiantes

- cod_e: Código estudiante
- nom_e: Nombres estudiante
- dir: Dirección.
- tel: Teléfono.
- fec: Fecha nacimiento.

Tabla asignaturas

- cod_a: Código asignatura.
- nom_a: Nombre Asignatura.
- ih: Intensidad Horaria.

Tabla Profesores

- id_p: Cédula o identificación del profesor.
- nom_p: Nombre del profesor.
- profesión: Profesión del profesor.

Tabla libros

- isbn: ISBN o código de publicación del libro
- título: Nombre o título de libro.

Tabla autores

- isbn: ISBN o código de publicación del libro
- nom_a: Nombre del autor

Tabla ejemplares

- isbn: ISBN o código de publicación del libro
- num_e: Numero ejemplar del libro.

Tabla imparte

- id_p: Identificador del profesor
- id_a: Id de la asignatura.
- horario: Horario de la clase.

Tabla inscribe

- id_p: identificador profesor.
- id_a: identificadora asignatura
- horario: horario de clase.
- cod_e: código del estudiante

- n1: NOTA1
- n2: NOTA2
- n3: NOTA3

Tabla referencia

- cod_a: código asignatura
- isbn: ISBN o código de publicación del libro

Tabla presta

- cod_e: código estudiante
- num_e: número ejemplar del libro
- isbn: ISBN o código de publicación del libro
- f_p: fecha préstamo de libro
- f_d: fecha devolución libro.

1. Creación de una base de datos

Para crear una base de datos en Postgres, es necesario que abra la ventana de Comandos de SQL desde el PGAdmin y después ingrese el siguiente comando: *CREATE DATABASE "Taller2_Practico_BD"*

La nueva base de datos se creará y se verá reflejada en la lista de bases de datos de postgres.

2. Creación de tablas

Para crear las tablas utilice el comando **CREATE TABLE**, y configure los atributos que esta tendrá. En el siguiente enlace puede encontrar los tipos de datos soportados por postgres <https://www.postgresql.org/docs/current/datatype.html>

Para crear la tabla de estudiantes, haga lo siguiente:

```
CREATE TABLE estudiantes
(
cod_e numeric(11) NOT NULL,
nom_e character varying(40) NOT NULL,
dir character varying(30) NOT NULL,
tel numeric(10,0) NULL,
fec date NOT NULL,
CONSTRAINT pk_estudiantes PRIMARY KEY (cod_e)
);
```

Con esta misma instrucción debe crear la totalidad de las tablas que conforman la base de datos. Igualmente, para esta etapa del proceso solo debemos definir las llaves primarias (PK) para los atributos que conforme el modelo se indica.

3. Inserción de datos

Para insertar datos en las tablas previamente creadas, se utiliza el comando INSERT. Este comando permite especificar la tabla sobre la cual se realizará la inserción de información y los valores que se insertarán.

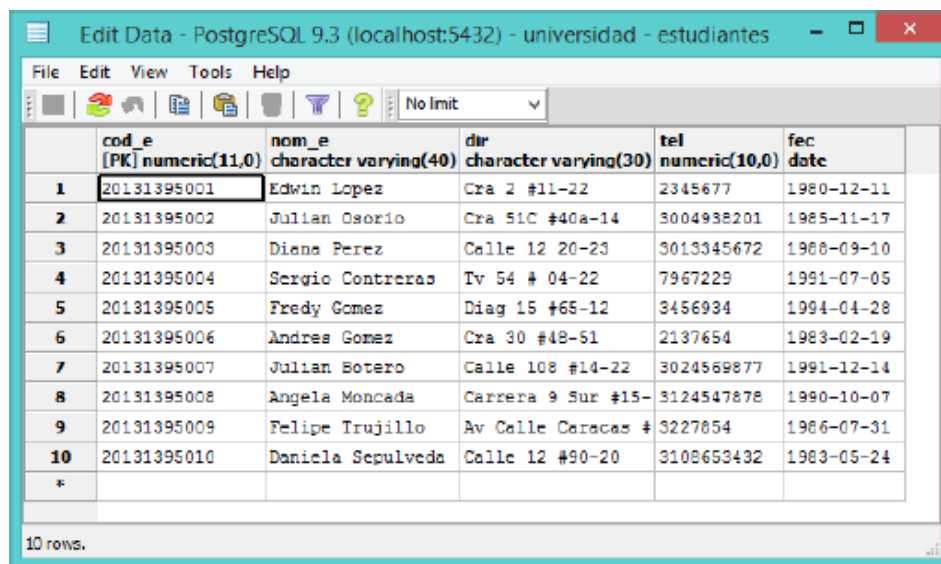
Para realizar la inserción de datos realice lo siguiente (ej.):

```
INSERT INTO autores values (1244678201,'Peter Neil');
```

Realice la inserción de los datos de acuerdo con las

gráficas:

Estudiantes



	cod_e [PK] numeric(11,0)	nom_e character varying(40)	dir character varying(30)	tel numeric(10,0)	fec date
1	20131395001	Edwin Lopez	Cra 2 #11-22	2345677	1980-12-11
2	20131395002	Julian Osorio	Cra 51C #40a-14	3004938201	1985-11-17
3	20131395003	Diana Perez	Calle 12 20-23	3013345672	1988-09-10
4	20131395004	Sergio Contreras	Tv 54 # 04-22	7967229	1991-07-05
5	20131395005	Fredy Gomez	Diag 15 #65-12	3456934	1994-04-28
6	20131395006	Andres Gomez	Cra 30 #48-51	2137654	1983-02-19
7	20131395007	Julian Botero	Calle 108 #14-22	3024569877	1991-12-14
8	20131395008	Angela Moncada	Carrera 9 Sur #15-	3124547878	1990-10-07
9	20131395009	Felipe Trujillo	Av Calle Caracas #	3227854	1986-07-31
10	20131395010	Daniela Sepulveda	Calle 12 #90-20	3108653432	1983-05-24
*					

10 rows.

4. Creación de índices

Para crear un índice se utiliza el comando CREATE INDEX. Este comando permite mejorar los tiempos en la consulta de grandes volúmenes de información, ya que permite ordenar y organizar de una manera más eficiente la información para ser consultada. En este enlace puede encontrar más información sobre el comando

(<https://www.postgresql.org/docs/14/sql-createindex.html>)

Para crear un índice realice lo siguiente:

```
CREATE INDEX IX_ESTUDIANTES ON estudiantes (cod_e);
```

Cree índices para las tablas asignaturas, autores y libros.

5. Consulte la información.

Para consultar la información se utiliza el comando SELECT. Este comando permite obtener la información almacenada en las tablas para su fácil análisis o para combinarla con otra información útil.

Utilice el siguiente comando para obtener toda la información de la tabla estudiantes:

```
SELECT * FROM estudiantes;
```

Utilice este comando para revisar las otras tablas de la base de datos.

En el siguiente link puede encontrar información sobre los diferentes comandos y opciones para consultar la información:

<https://www.postgresql.org/docs/current/queries.html>

Vamos a realizar algunas consultas:

5.1. Consultar el código del estudiante, su nombre y su teléfono.

```
SELECT cod_e as CODIGO, nom_e as NOMBRE, tel as TELEFONO FROM estudiantes;
```

5.2. Consultar todas las asignaturas con su nombre e intensidad horaria.

5.3. Consultar los nombres de los libros prestados y el estudiante que lo presto.

```
SELECT E.nom_e as NOMBRE, L.titulo as LIBRO  
FROM presta as P, libros as L, estudiantes as E  
WHERE P.cod_e=E.cod_e and  
P.isbn=L.ISBN
```

5.4. Consultar las materias impartidas y el profesor que imparte cada materia.

5.5. Consultar los alumnos cuya nota3 es mayor a 4. Muestre el nombre del alumno, la materia y la nota3. Ordene alfabéticamente.

```
SELECT A.nom_a as ASIGNATURA, E.nom_e as ESTUDIANTE, l.n3 as NOTA3
```

```
FROM inscribe as I, estudiantes as E, asignaturas as A
WHERE I.id_a=A.cod_a and
I.cod_e=E.cod_e and
I.n3>4
ORDER BY E.nom_e
```

5.6. Muestre los alumnos inscritos con su asignatura. Ordene alfabéticamente por asignatura.

5.7. Muestre el número de estudiantes por materia

```
SELECT A.nom_a as ASIGNATURA, count(I.id_a) as INSCRITOS
FROM inscribe as I, asignaturas as A
WHERE I.id_a=A.cod_a
GROUP BY I.id_a, A.nom_a
ORDER BY A.nom_a
```

5.8. Consultar los libros referenciados por las asignaturas: mostrar el nombre de la asignatura y el nombre del libro. Ordena por nombre de libro

5.9. Muestre la nota definitiva de cada estudiante

```
SELECT E.nom_e, A.nom_a, ((I.n1*0.35)+(I.n2*0.35)+I.n3*0.30) as DEFINITIVA
FROM estudiantes as E, asignaturas as A, inscribe as I
WHERE I.cod_e=E.cod_e and
I.id_a=A.cod_a
ORDER BY E.nom_e,A.nom_a ASC
```