



Universidad Nacional Autónoma de México
Facultad de Ingeniería

Bases de Datos

Tarea 3

Profesor: Ing. Fernando Arreola Franco

Grupo: 01

Tarea: 03

Alumna: Ramírez Flores Eslavica Monserrat

Fecha de entrega: 9/03/2021

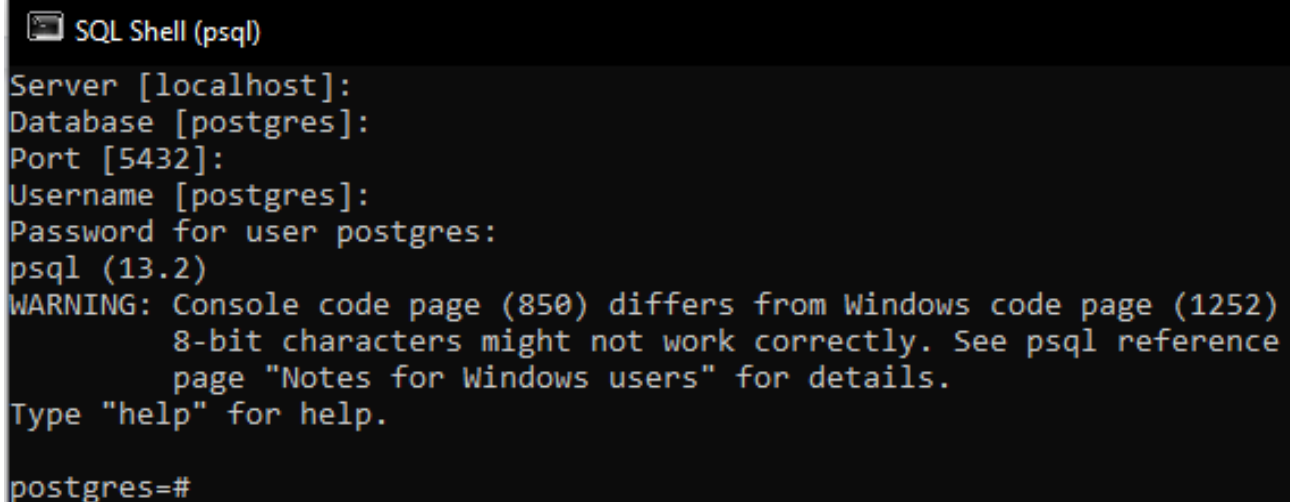
Parte 1

Crear un usuario con contraseña, que su cuenta sea válida por un mes y establecer un límite de conexiones.

Como se ha visto en clase de teoría, lo que se debe de realizar primero es crear un usuario y después asignarle una contraseña, ahora lo que cambia es encontrar algún comando para establecer un límite de conexiones.

Los pasos que se deben de realizar son los siguientes:

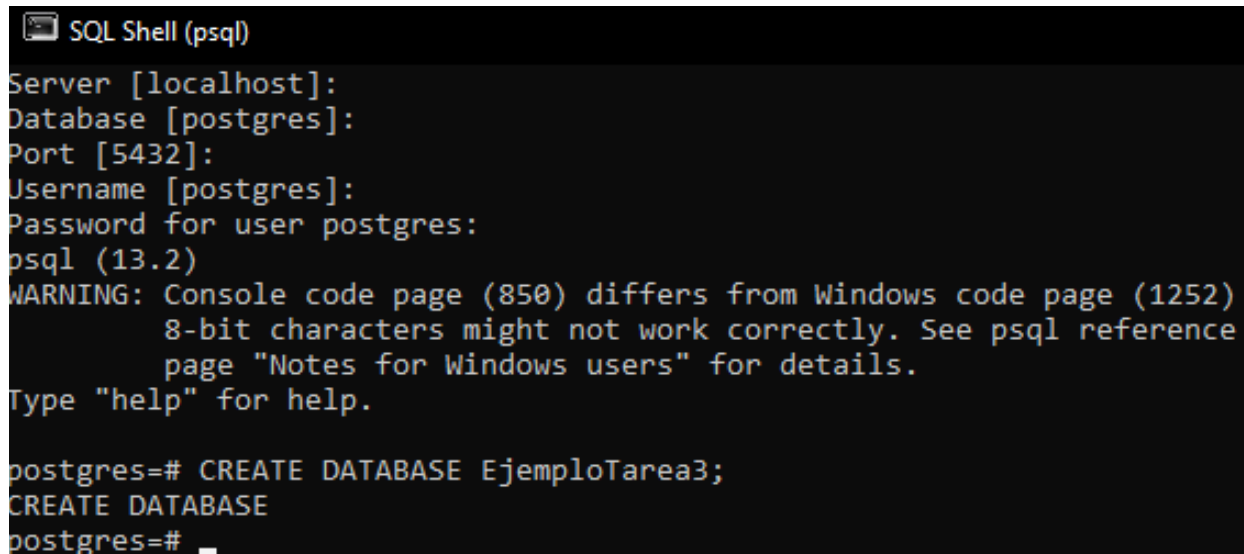
- Lo primero que se debe de realizar es entrar a SQL Shell y entrar con el usuario que hemos predeterminado:



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (13.2)
WARNING: Console code page (850) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

postgres=#
```

- Después crearemos una nueva bases de datos, en este caso le asigne el nombre *EjemploTarea3*, para realizar lo anterior se utilizó el comando **CREATE DATABASE EjemploTarea3**;
Donde después tarda un momento en crear la nueva bases de datos y la terminal nos lanzará un mensaje que ha sido creada, **CREATE DATABASE::**



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (13.2)
WARNING: Console code page (850) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

postgres=# CREATE DATABASE EjemploTarea3;
CREATE DATABASE
postgres=#
```

- Si queremos ver qué bases de datos tenemos creadas, basta con el comando `\l`, aquí encontraremos la nueva base de datos que creamos en el paso anterior:

```
postgres=# \l
```

Name	Owner	Encoding	Collate	Ctype	Access privileges
ejemplo	postgres	UTF8	English_United States.1252	English_United States.1252	
ejemplotarea3	postgres	UTF8	English_United States.1252	English_United States.1252	
grupo1	postgres	UTF8	English_United States.1252	English_United States.1252	
postgres	postgres	UTF8	English_United States.1252	English_United States.1252	
template0	postgres	UTF8	English_United States.1252	English_United States.1252	=c/postgres +
template1	postgres	UTF8	English_United States.1252	English_United States.1252	postgres=CTc/postgres +
					=c/postgres +
					postgres=CTc/postgres

```
(6 rows)

postgres=#
```

- A continuación se ponemos el comando `\c`, seguido del nombre de la base de datos que creamos para conectarnos y nos lanzará el siguiente mensaje:

```
(6 rows)

postgres=# \c ejemplotarea3
You are now connected to database "ejemplotarea3" as user "postgres".
ejemplotarea3=#
```

- Y para concluir creamos un usuario con su contraseña, seguido que su cuenta sea válida por un mes y estableciendo límites de conexiones, en este caso le daré hasta 250 conexiones al usuario: stormy.

El comando correspondiente será:

CREATE USER stormy WITH PASSWORD 'parte1' VALID UNTIL '2021-4-8' CONNECTION LIMIT 250;

```
postgres=# \c ejemplotarea3
You are now connected to database "ejemplotarea3" as user "postgres".
ejemplotarea3=# CREATE USER stormy WITH PASSWORD 'parte1' VALID UNTIL '2021-4-8' CONNECTION LIMIT 250;
ejemplotarea3=#
```

- Si queremos que fue creado el rol de forma exitosa, ponemos el comando `\du` y listo:

```
postgres=# \c ejemplotarea3
You are now connected to database "ejemplotarea3" as user "postgres".
ejemplotarea3=# CREATE USER stormy WITH PASSWORD 'parte1' VALID UNTIL '2021-4-8' CONNECTION LIMIT 250;
ejemplotarea3=# \du
```

Role name	Attributes	Member of
creatorbd	Cannot login	{}
creatorbr	Cannot login	{creatorbd}
postgres	Superuser, Create role, Create DB, Replication, Bypass RLS	{}
stormy	250 connections Password valid until 2021-04-08 00:00:00-05	{}

```
ejemplotarea3=#
```

Parte 2

Crear un role, asignar permisos de lectura, actualización y borrado en una tabla de nombre “estudiante”. Asignar dicho role al usuario del paso anterior.

- Creamos un nuevo role y este será para la parte 2, con el comando: **CREATE ROLE parte2;**
- Después creamos la tabla de nombre estudiante: **CREATE TABLE estudiante();**
- Para saber lo que llevamos hasta el momento y saber si fue creada la tabla con éxito basta con el comando **\du** y listo:
- Después para saber el contenido de la tabla estudiante ponemos el comando **\dp**

```
ejemplotarea3=# CREATE ROLE parte2;
CREATE ROLE
ejemplotarea3=# CREATE TABLE estudiante ();
CREATE TABLE
ejemplotarea3=# \du
```

Role name	List of roles Attributes	Member of
creadorbd	Cannot login	{}
creadorbr	Cannot login	{creadorbd}
parte2	Cannot login	{}
postgres	Superuser, Create role, Create DB, Replication, Bypass RLS	{}
stormy	250 connections	{}
	Password valid until 2021-04-08 00:00:00-05	

```
ejemplotarea3=# \dp
```

Schema	Name	Type	Access privileges	Column privileges	Policies
public	estudiante	table			

```
(1 row)
```

```
ejemplotarea3=#
```

- Después para agregar permisos al rol y rol a usuario:
- Usamos el comando **GRANT SELECT, UPDATE, DELETE ON estudiante to parte2;**
Es importante este comando ya que **GRANT** es el que se encarga de otorgar los privilegios a los roles.
SELECT es para asignar el permiso de lectura y **DELETE** para borrar.

```
ejemplotarea3=# GRANT SELECT, UPDATE, DELETE ON estudiante to parte2;
GRANT
ejemplotarea3=# \dp
```

Schema	Name	Type	Access privileges	Column privileges	Policies
public	estudiante	table	postgres=arwdDxt/postgres+ parte2=rwd/postgres		

(1 row)

```
ejemplotarea3=# GRANT parte2 TO stormy;
GRANT ROLE
ejemplotarea3=# \du
```

Role name	List of roles	Member of
	Attributes	
creatorbd	Cannot login	{}
creatorbr	Cannot login	{creatorbd}
parte2	Cannot login	{}
postgres	Superuser, Create role, Create DB, Replication, Bypass RLS	{}
stormy	250 connections	+ {parte2}
	Password valid until 2021-04-08 00:00:00-05	

```
ejemplotarea3=# _
```

Referencia consultada

- <https://www.postgresql.org/docs/>
- <https://www.postgresql.org/docs/9.3/sql-syntax.html>
- <https://www.postgresql.org/docs/current/>