

Práctica 3

El Shell de PostgreSQL

Objetivo

En esta práctica se introduce el manejo de arreglos y conjuntos de valores, así como datos definidos por el usuario en el diseño de bases de datos PostgreSQL.

Introducción

Una base de datos orientada a objetos como en el caso de PostgreSQL, ofrece herramientas como soporte para objetos complejos (conjuntos, arreglos, listas), la posibilidad de extender los tipos de datos, Polimorfismo, Herencia y Recursión.

Para conectarse a una base de datos PostgreSQL, se requiere tener acceso al Shell conocido como `psql`. Conociendo los comandos básicos es posible conocer el contenido de la base de dato. Para obtener el listado de las bases de datos existentes en el sistema, usar el comando `\l`. Para listar tablas de la base de datos actual se usa `\dt` y para obtener información sobre cualquier objeto de la base de datos es posible utilizar el comando `\d`.

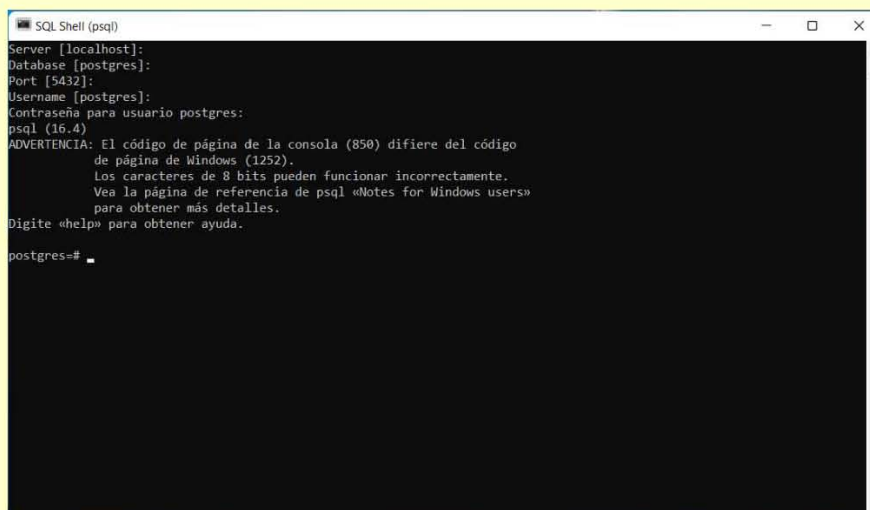
Para obtener una completa referencia de los comandos disponibles es posible usar el comando `\?`.

Equipo necesario

Una computadora con sistema operativo **Windows**, que cuente con el Shell de **PostgreSQL** conocido como ***psql***.

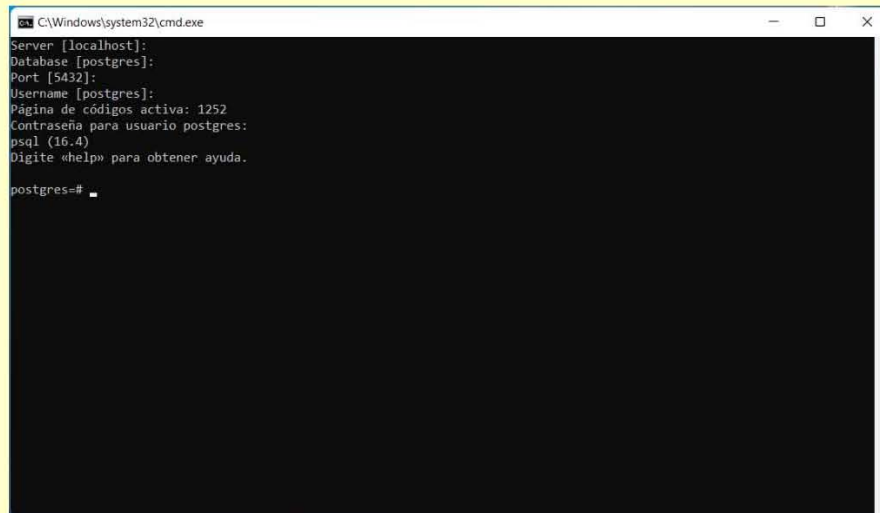
Metodología

1. Oprima el botón de **Inicio** de Windows y en la lista de todos los programas localice la carpeta **PostgreSQL 16**, ahí encontrará el acceso directo para ejecutar la herramienta **SQL Shell (*psql*)**.
2. Se abre la ventana de la consola de PostgreSQL, primero solicita indicar el servidor, que en este caso es **localhost**, oprima **enter**.
3. Ahora solicita indicar la base de datos que se va a abrir, por lo pronto deje el valor por omisión **postgres**, oprima **enter**.



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Contraseña para usuario postgres:
psql (16.4)
ADVERTENCIA: El código de página de la consola (850) difiere del código
de página de Windows (1252).
Los caracteres de 8 bits pueden funcionar incorrectamente.
Vea la página de referencia de psql «Notes for Windows users»
para obtener más detalles.
Digite «help» para obtener ayuda.
postgres=#
```

4. A continuación, solicita el número del puerto, deje el valor de 5432, oprima **enter**.
5. Ahora solicita indicar el usuario, deje el valor de **postgres**, que es el Superusuario registrado y oprima **enter**.
6. Finalmente solicita la contraseña, escriba la contraseña que definió en la primera práctica y oprima **enter**.
7. Ahora ya tiene el acceso a la consola de **psql**, tal como se muestra en la imagen de la siguiente página.
8. La primera cosa que podrá observar es que la consola no utiliza los caracteres latinos que emplea Windows, sino la página de códigos 850, esto puede causar conflictos.
9. Para resolver este problema, se requiere hacer lo siguiente:
10. Cierre la consola, tecleando el comando **exit**.
11. En su computadora localice el archivo llamado **runpsql.bat**, que se ubica en la siguiente carpeta:
`C:\Program Files\PostgreSQL\16\scripts`
12. Cópielo en la carpeta de **Documentos**, para poderlo modificar.
13. Abra el archivo con el **Bloc de notas**.
14. Debajo de la línea comentada: **REM Run psql**, agregue la siguiente línea:
`"cmd.exe" /c chcp 1252`
15. Guarde el archivo modificado y cópielo en la ubicación original.
16. Vuelva a entrar a la consola de **psql**, y ahora no deberá aparecer el mensaje de advertencia:



```
C:\Windows\system32\cmd.exe
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Página de códigos activa: 1252
Contraseña para usuario postgres:
psql (16.4)
Digite «help» para obtener ayuda.
postgres=#
```

17. Una vez dentro, el sistema le mostrará el siguiente prompt:

```
postgres=#
```

18. Esto indica que está activo el esquema llamado **postgres**.

19. Para listar las bases de datos que existen en el **tablespace** del usuario, escriba:

```
postgres=# \l
```

20. Ahí aparece la base de datos **Biblioteca** que creó en la práctica anterior, para seleccionarla escriba:

```
postgres=# \c Biblioteca
```

21. Además del aviso de cambio exitoso, podrá notar que el **prompt** también ha cambiado, para ver las tablas que hay en esta base de datos escriba:

```
Biblioteca=# \dt
```

22. Obtenga una imagen completa de la pantalla, mostrando la ejecución de estas últimas sentencias y consérvela como evidencia.

23. Enliste los registros de la tabla **Editoriales**, mediante la siguiente sentencia (Recuerde que el empleo de comillas se requiere dependiendo de la forma en que se hayan definido los nombres):

```
Biblioteca=# SELECT * FROM "Editoriales";
```

24. Una característica de PostgreSQL es que permite manejar Colecciones o Conjuntos de elementos de un tipo determinado, como ejemplo, ejecute la siguiente modificación de la tabla **Editoriales**:

```
Biblioteca=# ALTER TABLE "Editoriales" ADD COLUMN "Teléfonos"
CHAR(12)[];
```

25. Ahora podrá agregar dos números telefónicos para la Editorial **Debolsillo**:

```
Biblioteca=# UPDATE "Editoriales" SET "Teléfonos"[1]='833 163
```



```
2953', "Teléfonos"[2]='833 231 6634' WHERE  
"Nombre"='Debolsillo';
```

26. Y otros dos para la Editorial **Gandhi Editores**:

```
Biblioteca=# UPDATE "Editoriales" SET "Teléfonos"[1]='933 180  
0799', "Teléfonos"[2]='933 343 1365' WHERE "Nombre"='Gandhi  
Editores';
```

27. Revise los cambios:

```
Biblioteca=# SELECT * FROM "Editoriales";
```

28. Obtenga una imagen completa de la pantalla, mostrando la ejecución de estas últimas sentencias y consérvela como evidencia.

29. **PostgreSQL** permite que los usuarios pueden definir nuevos tipos además de lo que ya cuenta, usando el comando **CREATE TYPE**, como ejemplo, defina el tipo direc:

```
Biblioteca=# CREATE TYPE direc AS (Calle TEXT, Numero INT,  
Colonia TEXT, Ciudad TEXT);
```

30. Ahora, agregue una columna en la tabla **Autores**, llamada **Dirección**, con este tipo de datos:

```
Biblioteca=# ALTER TABLE Autores ADD COLUMN "Dirección" direc;
```

31. Ahora ingrese la dirección el Autor **Aurelio Baldor**.

```
Biblioteca=# UPDATE Autores SET "Dirección".Calle='Hidalgo',  
"Dirección".Numero=243,"Dirección".Colonia='Los Pinos',  
"Dirección".Ciudad='Morelia' WHERE Nombre='Aurelio Baldor';
```

32. Y luego agregue la dirección para la autora **Isabel Allende**, empleando un modo diferente:

```
Biblioteca=# UPDATE Autores SET "Dirección" = ROW('Morelos',  
417, 'Alameda', 'Zamora') WHERE Nombre='Isabel Allende';
```

33. Revise los cambios:

```
Biblioteca=# SELECT * FROM Autores;
```

34. Obtenga una imagen completa de la pantalla, mostrando la ejecución de estas últimas sentencias y consérvela como evidencia.

35. Se puede hacer una consulta separando cada elemento de la dirección en una columna distinta, observe que debe escribir el nombre **"Dirección"** entre paréntesis:

```
Biblioteca=# SELECT Nombre, ("Dirección").calle,  
("Dirección").Numero,("Dirección").colonia,("Dirección").ciudad  
FROM Autores;
```

36. Para realizar consultas de semejanza, generalmente se emplea la cláusula **LIKE**, como en este ejemplo:

```
Biblioteca=# SELECT Nombre, "Dirección" FROM Autores WHERE  
Nombre LIKE '%A%';
```

37. Pero también se puede emplear la cláusula ***SIMILAR TO*** para hacer la misma consulta:

```
Biblioteca=# SELECT Nombre, "Dirección" FROM Autores WHERE  
Nombre SIMILAR TO '%A%';
```

38. Una tercera alternativa de esta consulta es usando una expresión regular y el símbolo tilde (~):

```
Biblioteca=# SELECT Nombre, "Dirección" FROM Autores WHERE  
Nombre ~ 'A';
```

39. Obtenga una imagen completa de la pantalla, mostrando la ejecución de estas últimas consultas y consérvela como evidencia.

40. Ahora en la tabla ***Editoriales***, agregue otra columna, esta vez usando el tipo ***SERIAL***, es decir, autonumérico:

```
Biblioteca=# ALTER TABLE "Editoriales" ADD COLUMN Clave SERIAL;
```

41. Observe el resultado de este agregar esta columna, asignando un número de clave a cada editorial:

```
Biblioteca=# SELECT * FROM "Editoriales";
```

42. Ahora liste la estructura de esta tabla, mediante el siguiente comando:

```
Biblioteca=# \d "Editoriales"
```

43. Obtenga una imagen completa de la pantalla, mostrando la ejecución de estas últimas consultas y consérvela como evidencia.

44. Puede ver que se ha definido un valor por omisión para la columna Clave basado de una secuencia auxiliar (***Editoriales_Clave_seq***) que se generó con este objetivo.

45. Ahora, debe agregar una columna en la tabla ***libros*** de tipo MONEY, monetario con valor por omisión de diez pesos:

```
Biblioteca=# ALTER TABLE Libros ADD COLUMN Precio MONEY DEFAULT  
10;
```

46. También va a definir un tipo de dato enumerado, para indicar el idioma en que está escrito cada libro, enumerando los más comunes:

```
Biblioteca=# CREATE TYPE lengua AS ENUM ('Español', 'Inglés',  
'Francés');
```

47. Ahora, agregue una nueva columna en la tabla ***libros***, llamada ***Idioma***, para indicar la lengua en que está escrito el libro, asignando por omisión el idioma ***Español***:

```
Biblioteca=# ALTER TABLE Libros ADD COLUMN Idioma lengua  
DEFAULT 'Español';
```


48. A continuación, muestre la estructura de esta tabla modificada:

```
Biblioteca=# \d Libros
```

49. Obtenga una imagen completa de la pantalla, mostrando la ejecución de estas últimas consultas y consérvela como evidencia.

50. Ahora, escriba la sentencia para eliminar la llave primaria (**PkEditoriales**) de la tabla **Editoriales**, tome en cuenta que existe una llave foránea que la vincula con la tabla **libros**, por lo que hay que indicar la eliminación en cascada, para que se eliminen ambas.

51. Ahora, proceda a crear la nueva llave principal, ahora sobre la columna **Clave**, con el mismo nombre (**PkEditoriales**).

52. Hay que volver a crear la llave foránea que fue eliminada de la tabla **libros**, para ello, primero es necesario indicar que la columna **Nombre** de la tabla **Editoriales** es de valor único (UNIQUE), definiendo la restricción **UkNombre**, para que sea equivalente a una llave primaria.

53. Entonces, vuelva a crear la llave foránea (**FkEditorial**) de la tabla **libros**.

54. Revise como quedó la estructura de la tabla **Editoriales**:

```
Biblioteca=# \d "Editoriales"
```

55. Obtenga una imagen completa de la pantalla, mostrando la ejecución de estas últimas sentencias y consérvela como evidencia.

56. Para cerrar la consola, escriba:

```
Biblioteca=# \q
```

57. Fin de la Práctica.

Evidencias

El alumno deberá enviar al instructor **un documento PDF** con todas las evidencias solicitadas durante la realización de la práctica.

Sugerencias didácticas

El instructor deberá atender a los alumnos que tengan dificultades en la interpretación y la realización de las instrucciones de la práctica.

Resultados

Se aprendió a crear distintos tipos de datos por el usuario, así como el manejo de arreglos y conjuntos de datos mediante **PostgreSQL**, modificando una tabla, agregando columnas de diversos tipos, y añadiendo diversos valores.

Bibliografía

- <https://www.postgresql.org/docs/current/app-psql.html>