Command line PostgreSQL: Takeaways ⤴

Syntax

- Starting the PostgreSQL command line tool:

  ```
  psql
  ```

- Exiting the PostGreSQL command line tool:

  ```
  \q
  ```

- Creating a database:

  ```
  CREATE DATABASE dbName;
  ```

- Listing databases:

  ```
  \l
  ```

- Listing all tables in the current database:

  ```
  \dt
  ```

- Listing the users that have access to the database:

  ```
  \du
  ```

- Connecting to a specified database:

  ```
  psql -d dbName
  ```

- Creating a user:

  ```
  CREATE ROLE userName;
  ```

- Allowing a user to login to PostgreSQL and run queries:

  ```
  CREATE ROLE userName WITH LOGIN;
  ```

- Creating a password for a user:

  ```
  CREATE ROLE userName WITH LOGIN PASSWORD `password`:
  ```

- Allowing a user to create databases:

  ```
  CREATE ROLE userName WITH CREATEDB LOGIN PASSWORD 'password`;
  ```

- Allowing a user to create other users:

  ```
  CREATE ROLE userName WITH CREATEROLE LOGIN PASSWORD 'password`;
  ```

- Making the user a superuser:

  ```
  CREATE ROLE userName WITH LOGIN PASSWORD 'password` SUPERUSER;
  ```

- Granting a user permissions to access a table:

  ```
  GRANT SELECT ON tableName TO userName;
  ```

- Granting a user complete control of a table:

  ```
  GRANT ALL PRIVILEGES ON tableName to userName;
  ```

- Displaying what privileges have been granted to users:

  ```
  \dp tableName
  ```

- Removing permissions from a user:

  ```
  REVOKE SELECT ON tableName FROM userName;
  ```

- Removing all permissions from a user:

  ```
  REVOKE ALL PRIVILEGES on tableName FROM userName;
  ```

Concepts

- The PostgreSQL command line tool is called `psql` .

- `psql` connects to a running PostgreSQL server process, which enables you to:

  - Run queries.
  - Manage users and permissions.
  - Manage databases.
  - See PostgreSQL system information.

- Queries in `psql` must end with a semicolon (`;`) or they won't be performed.

- When users are created, they don't have any ability, or permissions, to access tables in existing databases.

- You can grant or revoke multiple permissions by separating them with commas.

- You can grant or revoke users ability to use the `SELECT`, `INSERT`, `UPDATE`, or `DELETE` clauses on a table.
- A superuser can perform any function in a database.

Resources

- [psql documentation](#)
- [17 Practical psql commands](#)