Using psql

Print to PDF

Let us understand how to use `psql` utility to perform database operations.

- `psql` is command line utility to connect to the Postgres database server. It is typically used for the following by advanced Database users:
  - Manage Databases
  - Manage Tables
  - Load data into tables for testing purposes
- We need to have at least Postgres Client installed on the server from which you want to use psql to connect to Postgres Server.
- If you are on the server where **Postgres Database Server** is installed, `psql` will be automatically available.
- We can run `sudo -u postgres psql -U postgres` from the server provided you have sudo permissions on the server. Otherwise we need to go with `psql -U postgres -W` which will prompt for the password.
- **postgres** is the super user for the postgres server and hence typically developers will not have access to it in non development environments.
- As a developer, we can use following command to connect to a database setup on postgres server using user credentials.

```
psql  -h <host_ip_or_dns_alias> -d <db_name> -U <user_name> -W

# Here is the example to connect to itversity_sms_db using itversity_sms_user
psql -h localhost -p 5432 -d itversity_sms_db -U itversity_sms_user -W
```

- We typically use `psql` to troubleshoot the issues in non development servers. IDEs such as **SQL Alchemy** might be better for regular usage as part of development and unit testing process.
- For this course, we will be primarily using Jupyter based environment for practice.
- However, we will go through some of the important commands to get comfortable with `psql`.
  - Listing Databases - `\l`
  - Switching to a Database - `\c <DATABASE_NAME>`
  - Get help for `psql` - `\?`
  - Listing tables - `\d`
  - Create table - `CREATE TABLE t (i SERIAL PRIMARY KEY)`
  - Get details related to a table - `\d <table_name>`
  - Running Scripts - `\i <SCRIPT_PATH>`
  - You will go through some of the commands over a period of time.

By Durga Gadiraju