***PROFESSIONAL DATA ENGINEER** - Google Cloud Platform*

TITLE: "Introduction to Google Cloud SQL - Console"

Author:

- *Name: "Vignesh Sekar"*
- *Designation: "Multi Cloud Architect"*
- *Tags: [Google Cloud, DataEngineer, Python, PySpark, SQL, BigData]*

Task 1 - Create a Cloud SQL instance

1. Click on the menu icon in the top left of the screen to see the Navigation menu (Navigation menu icon).
2. In the left menu of the Console, click on SQL.
3. Click Create Instance.
4. Click Choose PostgreSQL.
5. Create your instance with the following settings:
6. Enter myinstance for Instance ID.
7. Enter a password for the postgres user. Save or remember this password, you'll need it in the next section.
8. Click Create Instance.

Task 2 - Connect to your instance using the psql client in the Cloud Shell

1. In the Cloud Console, click Cloud Shell (Cloud Shell icon) in the upper right corner.Then click Continue if prompted.
2. In Cloud Shell, display the instance details:
   - export CLOUD_SQL_INSTANCE=mypostgresinstance
   - gcloud sql instances describe $CLOUD_SQL_INSTANCE

3. At the Cloud Shell prompt, connect to your Cloud SQL instance by running:

    - gcloud sql connect mypostgresinstance --user=postgres

4. Enter your postgres password.You should now see the psql prompt.

5. Display information about current connection

    - \conninfo

6. Create Database.

    - CREATE DATABASE book;

7. In psql, change to the orders database:

    - \c book;

8. List all databases on the sql server.

    - \l

9. To see all the tables in the database.

    - \dt

10. Create a customize schema

    - CREATE SCHEMA studentbook

11. To see table's field formats.

    - \dt .;
    - SELECT schema_name FROM information_schema.schemata;

12. To delete a db.

    - DROP DATABASE book;
    - DROP DATABASE IF EXISTS book;

13. Create the table "guestbook".

    - CREATE TABLE guestbook (guestName VARCHAR(255), content VARCHAR(255), entryID SERIAL PRIMARY KEY);

14. Show all the tables to confirm that the "guestbook" table has been created

    - \dt;

15. To delete a table.

    - DROP TABLE guestbook;
    - DROP TABLE IF EXISTS guestbook;

16. Insert a row with all the column values

    - INSERT INTO guestbook (guestName, content) values ('first guest', 'I got here!');
    - INSERT INTO guestbook (guestName, content) values ('second guest', 'Me too!');

17. Create view as "studentdetails"

    - CREATE VIEW studentbook.winners AS SELECT guestName FROM guestbook ;

18. To view data

    - SELECT * FROM guestbook;
    - select * from studentbook.winners;

19. Drop view from postgresql

    - DROP VIEW studentbook.winners;

20. To return columns and column information.

- SELECT column_name, data_type FROM information_schema.columns WHERE table_name = 'guestbook';

NOTE:

1. Now in Psql you could run commands such as:

    - ? -----> list all the commands
    - \l -----> list databases
    - \conninfo -----> display information about current connection
    - \c [DBNAME] -----> connect to new database, e.g., \c template1
    - \dt -----> list tables of the public schema
    - \dt .* -----> list tables of certain schema, e.g., \dt public.*
    - \dt . -----> list tables of all schemas
    - Then you can run SQL statements, e.g., SELECT * FROM my_table;(Note: a statement must be terminated with semicolon ;)
    - \q -----> quit psql