**MySql and Cloud Spanner: create instance, create a table and load data**

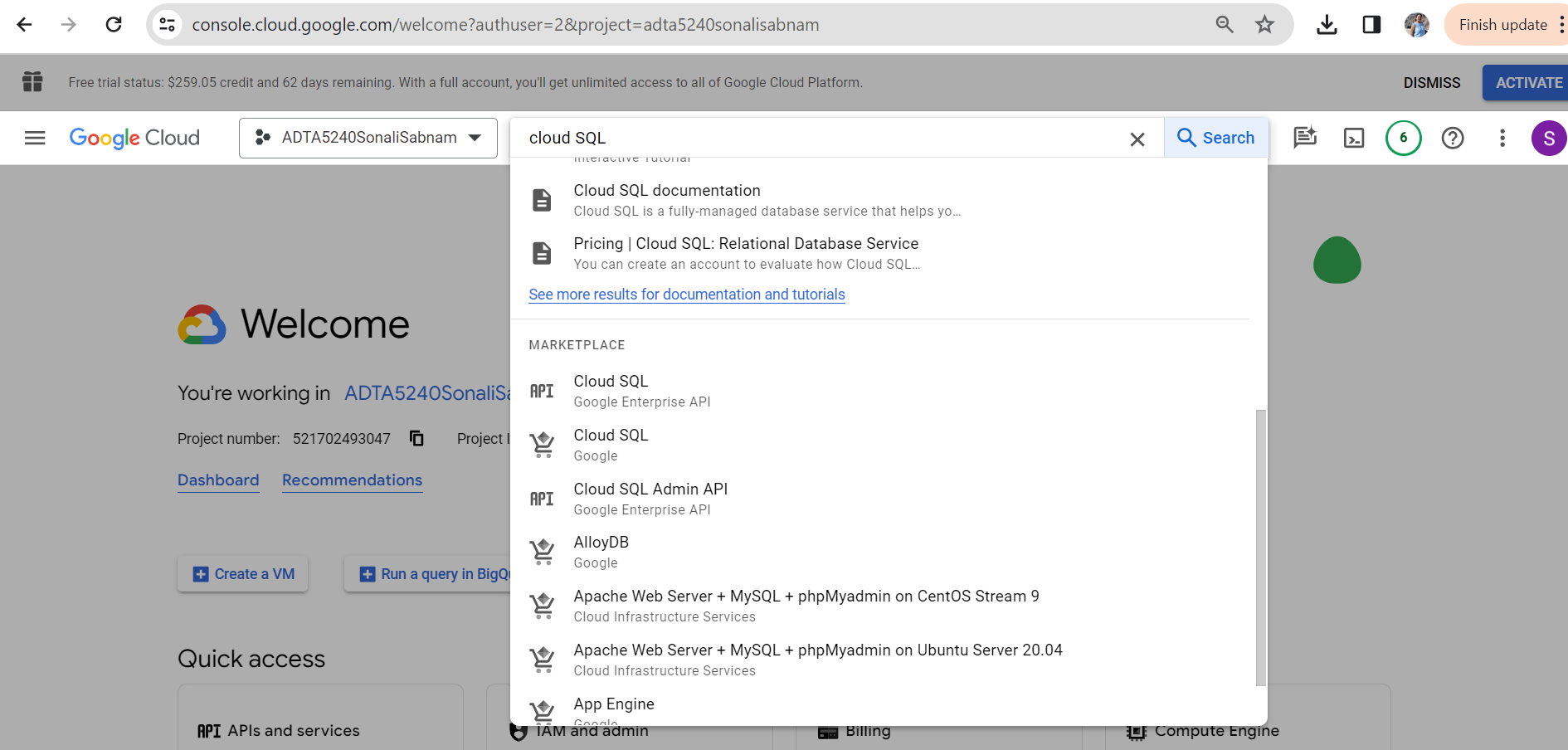
**TASK 1: Create instance for MySql, create database, table and load data**

**Objective:**

* To create Cloud MYSQL instance, table and load data.

**Step 1: Enable Cloud SQL Admin API**

* Login to google console using GMAIL/google ID.
* Select your project and in the search bar type “Cloud SQL”, the below screen is shown.



* Click on “Cloud SQL Admin API”
* The below screen get displayed.

**A screenshot of a computer

Description automatically generated**

* Click on “ENABLE”

A screenshot of a computer

Description automatically generated

* Now, the Cloud SQL Admin API is enabled.

**Step 2: Create a Cloud SQL instance**

* In the search bar type “SQL”
* The below screen gets displayed.

A screenshot of a computer

Description automatically generated

* Click on “SQL”
* The below screen gets displayed.

A screenshot of a computer

Description automatically generated

* Click on “CREATE INSTANCE WITH YOUR FREE CREDITS”.
* The below screen gets displayed.

A screenshot of a computer

Description automatically generated

* Select “MySQL”.
* The below screen gets displayed.

A screenshot of a computer

Description automatically generated

* Provide an instance ID, password.
* Select “Enterprise” for Cloud SQL edition.

A screenshot of a computer

Description automatically generated

* Select “Sandbox” as the environment.
* Select zonal availability as “Single Zone”.
* Click on “CREATE INSTANCE”.
* The MySql instance is now created and it is running.

A screenshot of a computer

Description automatically generated

**Step 3: Connect to MYSQL shell and login to the MYSQL instance**

* In the above screen, scroll down and click on “OPEN CLOUD SHELL”.
* The below screen gets displayed.

**A screenshot of a computer

Description automatically generated**

* Hit Enter and authorize.
* Type your password and hit enter.

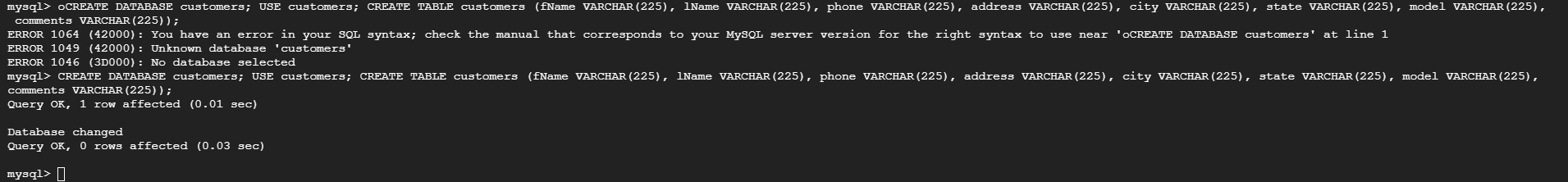
A screenshot of a computer

Description automatically generated

* Now you are connected to the MYSQL instance.

**Step 4: Create new database and add a table**

* Type the below query to create Database “customers” and add table “customers”.Hit enter after typing the code.
  + CREATE DATABASE customers; USE customers; CREATE TABLE customers (fName VARCHAR(225), lName VARCHAR(225), phone VARCHAR(225), address VARCHAR(225), city VARCHAR(225), state VARCHAR(225), model VARCHAR(225), comments VARCHAR(225));

****

* The query gets executed successfully and the database and table created.

**Step 5: Enter data into newly created database and table**

* To enter data into the table, execute the below query.

INSERT INTO customers (fName, lName, phone,address, city, state, model, comments) values ("Tony", "Barone", "555-676-7778","1018 State Street", "Houston", "TX", "A-1237", "This is the best product I have ever purchased.");

INSERT INTO customers (fName, lName, phone,address, city, state, model, comments) values ("Helen", "Smith", "777-879-0098", "889 Elm Road", "St. Louis", "MO", "H-435", "I would never buy this product again!");

INSERT INTO customers (fName, lName, phone, address, city, state, model, comments) values ("Susan", "Heller", "876-888-6795", "879 Main Street", "Los Angeles", "CA", "K-8887", "All good");

INSERT INTO customers (fName, lName, phone, address, city, state, model, comments) values ("Betsy", "Clark", "555-887-1098", "45 West 54th Ave.", "Topeka", "KS", "Z-2", "No issues");

* The query gets executed successfully and the data gets inserted into the table.

A black and white screen

Description automatically generated

**Step 6: Retrieve the data**

* Execute the below query to retrieve the data from the table.

SELECT \* FROM customers;

* The below screen displays the data that we inserted in the previous steps.

A black and white screen with white text

Description automatically generated

**Step 7: Stop the instance**

* Close the terminal by clicking the “X” button.
* On the Cloud SQL Instance page click on “STOP”

A screenshot of a computer

Description automatically generated

* The cloud SQL instance is now stopped.

A screenshot of a computer

Description automatically generated

**TASK 2: Create instance for Cloud Spanner, create dataset, table and load data**

**Objective:**

* To understand Cloud Spanner to create instance, database and schema.
* Put data into the database and run SQL on the same.
* Cleanup.

**Step 1: Enable Cloud Spanner API**

* In the search bar type “Cloud Spanner API”
* The below screen is displayed.

A screenshot of a computer

Description automatically generated

* Click on Cloud Spanner API.

A screenshot of a web page

Description automatically generated

* Click on “ENABLE”
* Cloud Spanner API is now enabled.

A screenshot of a computer

Description automatically generated

**Step 2: Create an instance**

* Activate the shell from the top right corner of the page.

**A screenshot of a computer

Description automatically generated**

* Type the below command to create the instance.

gcloud spanner instances create sonali-instance --config=regional-us-central1 --description="sonali instance" --nodes=2

* The cloud spanner instance is now created.

A black screen with white text

Description automatically generated

* To set the default instance, execute the below command.

gcloud config set spanner/instance sonali-instance

* The default instance is now set.

A black screen with white text

Description automatically generated

**Step 3: Create Database and Tables**

* To create a database, execute the below command.

gcloud spanner databases create example-db

A black screen with white text

Description automatically generated

* To create table Player1, execute the below command.

gcloud spanner databases ddl update example-db --ddl='CREATE TABLE Players1 ( PlayersId INT64 NOT NULL, FirstName STRING(1024), LastName STRING(1024) ) PRIMARY KEY (PlayersId)'

A black screen with white text

Description automatically generated

* To create table PlayerStats1, execute the below code.

gcloud spanner databases ddl update example-db --ddl='CREATE TABLE PlayerStats1 ( PlayersId INT64 NOT NULL, GameId INT64 NOT NULL, OpposingTeam STRING(MAX), PointsScore INT64, OpposingScore INT64) PRIMARY KEY (PlayersId, GameId), INTERLEAVE IN PARENT Players1 ON DELETE CASCADE'

A black screen with white text

Description automatically generated

**Step 4: Write Data to the tables**

* Execute the below commands to insert/write Player data into the Players1 table

gcloud spanner rows insert --instance=sonali-instance --database=example-db --table=Players1 --data=PlayersId=1,FirstName='Lebron',LastName='James'

gcloud spanner rows insert --instance=sonali-instance --database=example-db --table=Players1 --data=PlayersId=2,FirstName=’Stephen’,LastName=‘Curry’

gcloud spanner rows insert --instance=sonali-instance --database=example-db --table=Players1 --data=PlayersId=3,FirstName=‘Kevin’,LastName=‘Durant’

**A black and white image

Description automatically generated**

* Execute the below code to enter the Game details.

gcloud spanner rows insert --instance=sonali-instance --database=example-db --table=PlayerStats1 --data=PlayersId=3,GameId=1,OpposingTeam='Lakers',PointsScore=130,OpposingScore=112

gcloud spanner rows insert --instance=sonali-instance --database=example-db --table=PlayerStats1 --data=PlayersId=3,GameId=2,OpposingTeam='Golden State

Warriors',PointsScore=98,OpposingScore=109

gcloud spanner rows insert --instance=sonali-instance --database=example-db --table=PlayerStats1 --data=PlayersId=1,GameId=3,OpposingTeam='Suns',PointsScore=113,OpposingScore=123

A black and white image of a screen

Description automatically generated

* Now, we have the tables and the data ready.
* To close the terminal click on the ‘X’ at the top right corner of the terminal.

**Step 5: Query data**

* In the search bar type “Spanner”.

**A screenshot of a computer

Description automatically generated**

* Click on your instance.
* The below screen shows the instance. The database is also listed here.

**A screenshot of a computer

Description automatically generated**

* Click on the database.
* The below screen shows.

A screenshot of a computer

Description automatically generated

* Click on “Spanner Studio” from the left pane.
* The below screen shows.

A screenshot of a computer

Description automatically generated

* Once we expand the explorer items, we can see the tables and data that we created in the previous step.
* Click on the “+” button in the right pane.
* Write the below query to select the data from Players1 table.

select \* from Players1

A screenshot of a computer

Description automatically generated

* The above screen displays the data in the table.
* Write the below query to get data from the PlayerStats1 table

select \* from PlayerStats1

* The query displays the below data.

A screenshot of a computer

Description automatically generated

**Step 6: Delete the database and the instance**

* To avoid unnecessary charges, we need to drop the database and the instance.
* Activate the cloud shell from the top right corner.
* Set the instance with the below command.

gcloud config set spanner/instance sonali-instance

A screen shot of a computer

Description automatically generated

* Execute the below command to delete the database and Enter ‘Y’.

gcloud spanner databases delete example-db

A screen shot of a computer code

Description automatically generated

* Execute the below command to delete the instance and type ‘Y’ when asked.

gcloud spanner instances delete sonali-instance

A screen shot of a computer program

Description automatically generated

* With this, the instance, table and the data are deleted. We can not access them again. We must create new ones for the next time.

**Conclusion:**

With this exercise, we performed the below listed tasks

* Created a Cloud MySQL instance
* Created MySql database
* Added table to the database
* Inserted data to the tables
* Verified if the data was successfully loaded.
* Created a Cloud Spanner Instance
* Created a database
* Added table to the database
* Inserted data into the database
* Verified the data is present in the tables
* cleanup