EE477 Database and Big Data Systems, Spring 2021
HW0*

Due date: 04/02/2021 (11:59pm)

This is a guide for you to get started with Google Cloud SQL. Cloud SQL is a fully-managed database service that makes it easy to set up, maintain, manage, and administer your relational databases on Google Could Platform(GCP). Google Cloud SQL supports either MySQL or PostgreSQL for the database language, and we will use MySQL.

Section 1 explains how to create and connect to a Cloud MySQL instance and perform basic SQL opertaions on GCP. For more details, please follow the link for *Quickstart for Cloud SQL for MySQL*: https://cloud.google.com/sql/docs/mysql/quickstart

Section 2 explains several things to keep in mind when using Cloud SQL on GCP.

Section 3 is the actual homework assignment. There are no deliverables for sections 1 and 2. In section 3, you are asked to write and submit your own simple MySQL code for each problem.

Submission instructions: Use KAIST KLMS to submit your homeworks. Your submission should be one gzipped tar file whose name is `YourStudentID_hw0.tar.gz`. For example, if your student ID is 20210000, and it is for homework #0, please name the file as `20210000_hw0.tar.gz`. You can also use these extensions: tar, gz, zip, tar.zip. Do not use other options not mentioned here.

Your zip file should contain three things; one PDF file for writeup queries of both [Task 1] and [Task 2] (`hw0.pdf`), one folder named `YourStudentID_hw0`(put all [Task 2] output files {hw0_1.csv, hw0_2.csv, ..., hw0_9.csv}), and the Ethics Oath pdf file. Do not include Korean letters in any file name or directory name when you submit. **If you violate any format (file name or extension) error, we will deduct 1 point per mistake.**

Submitting writeup: Prepare answers to the homework questions into a single PDF file. You can use the following template. Please write as succinctly as possible.

Ethics Oath: For every homework submission, please fill out and submit the **PDF** version of this document that pledges your honor that you did not violate any ethics rules required by this course and KAIST. You can either scan a printed version into a PDF file or make the Word document into a PDF file after filling it out. Please sign on the document and submit it along with your other files.

*Material adapted from Google Cloud Manual and Simon Fraser University CMPT354.

1 Setting up Google SQL for MySQL

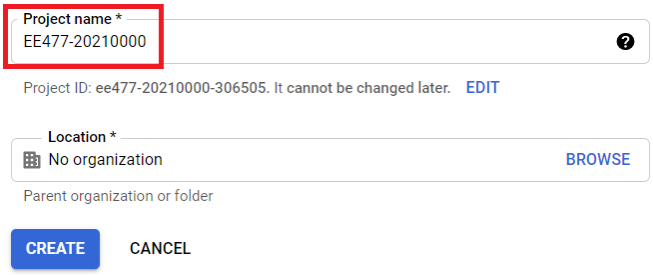
- Google Cloud Platform Setting.

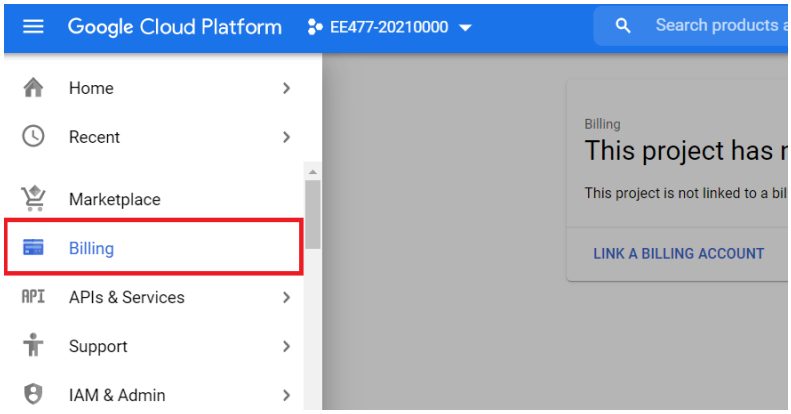
  (1) Create a Gmail account if you do not already have one.

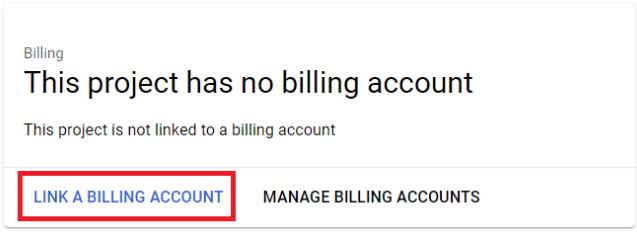
  (2) Send your Gmail address to the head TA through this form: Google Survey Link.

  (3) The head TA will then add you as a EE477 billing account user. It will be updated at every 7:00pm.

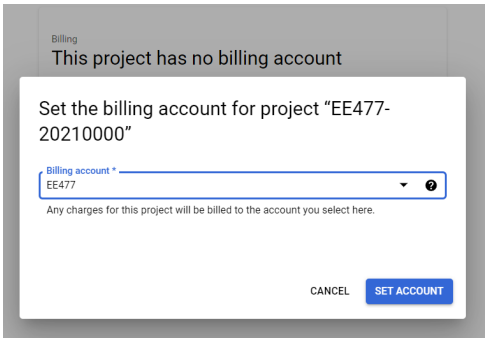
  (4) Click GO TO CONSOLE on Google Cloud webpage: https://cloud.google.com/gcp/ and create your own GCP project and name it `EE477-YourStudentID` (e.g., EE477-20210000). Please use this naming format strictly because we would like to identify the owner of each project. Then, click **Link a billing account**. We recommend Chrome for using GCP.







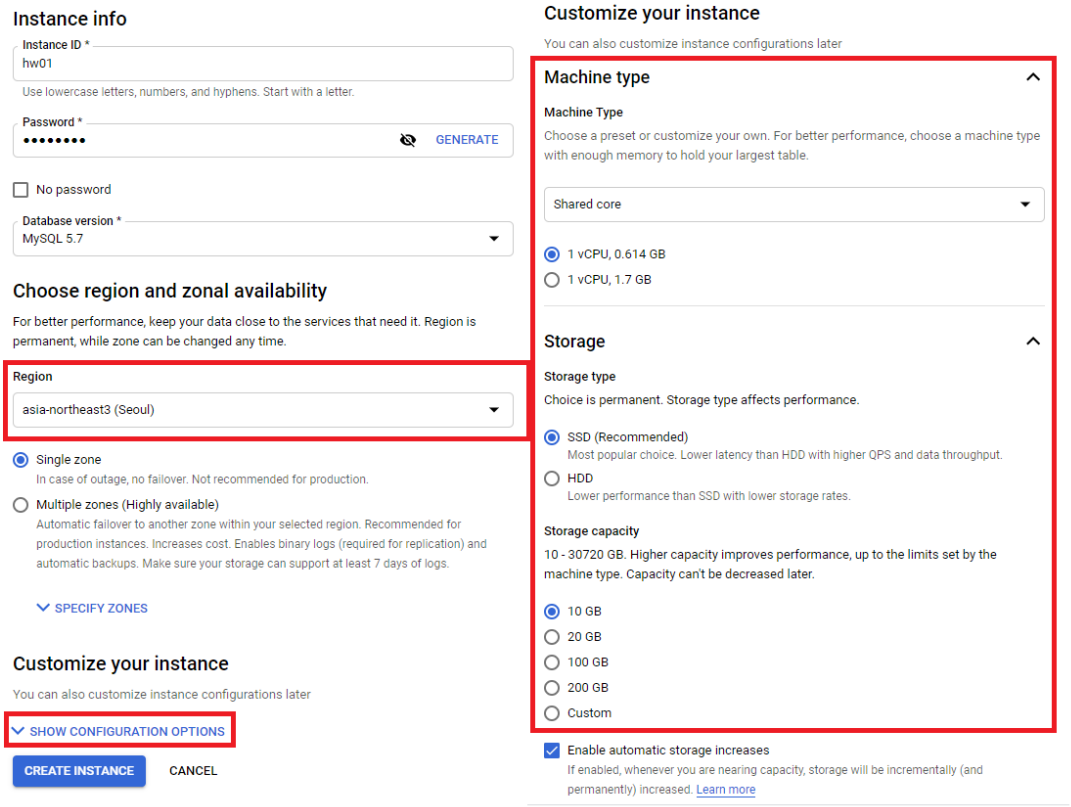
  (5) Set **EE477** for your billing account (**after step (3)**). Since **EE477** is a shared account and has limited credit, please read Section 2 carefully.



- Create a Cloud SQL instance.

  (1) Click explore menu and then SQL to create Cloud SQL instance on your GCP project.

  (2) Create Instance > MySQL > ENABLE API > Enter your instance ID and password > Region is asia-anything > Show configuration options > Change machine and storage type as shown in the following figures > Create. It may take a few minutes.

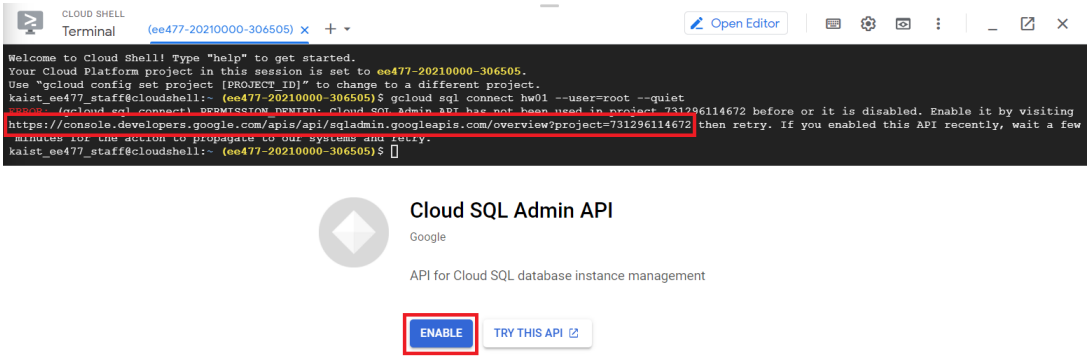


- Connect to your instance using the MySQL client. There are several options for connecting to SQL instance. We recommend you to use one of below two methods, and please refer to the following link for other options:
  https://cloud.google.com/sql/docs/mysql/external-connection-methods

  (1) The simplest way is using Activate Cloud Shell icon in the upper right corner.

(2) Install Google Cloud SDK on your local machine and connect to your google account by following instructions in https://cloud.google.com/sdk/docs/.

Then, type `"gcloud sql connect YOUR_INSTANCE_ID --user=root --quiet"` (For the first time you connect, you have to enable Cloud SQL Admin API.)



- Simple MySQL code example.

  (1) Create a SQL database on your Cloud SQL instance.

  ```
  CREATE DATABASE EE477;
  USE EE477;
  ```

  (2) Create table in database and insert sample data.

  ```
  CREATE TABLE TA(name VARCHAR(10), studentid INT);
  INSERT INTO TA(name, studentid) VALUES
  ('Roh', 20210001), ('Tae', 20210002), ('Heo', 20210003);
  ```

  (3) Show contents of table.

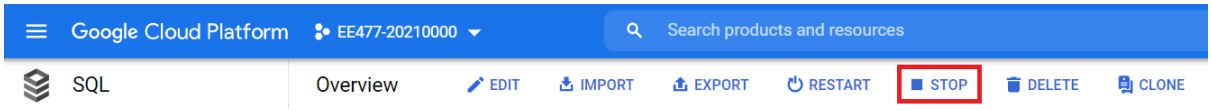
  ```
  SELECT * FROM TA;
  ```

  You should see;

  ```
  +--------+-----------+
  | name   | studentid |
  +--------+-----------+
  | Roh    |  20210001 |
  | Tae    |  20210002 |
  | Heo    |  20210003 |
  +--------+-----------+
  ```

2 Directions for using GCP credit

We will be offering a shared billing account for using GCP credit. Total charges will be determined by instance usage time as well as type of machine and storage. Because total credit is limited, we strongly recommend you to **STOP** an instance when not in use. Please refer to Cloud SQL for MySQL Pricing for detailed information.



3 Write your own MySQL Job

Now, you will go through how to use MySQL to store and query a database. In this task, you will learn the followings:

- Create a table and insert data
- Select certain rows or columns
- Join two tables
- Use expressions

There may be more than one way to solve the problem, and providing one of the correct answers is good enough. Please refer to several documents including following sites to understand how to use MySQL(5.7 version in GCP).

- http://www.mysqltutorial.org/basic-mysql-tutorial.aspx
- https://dev.mysql.com/doc/

[Task 1] Using SQL to create a database

(1) Create an empty database named *coursys*

(2) Create a table named *grades*

The *grades* table has four columns and six rows as shown below.

```
studentid, course, mark, credit
20210001,  EE477,  90,   3.5
20210001,  EE412,  85,   4
20210001,  EE209,  79.5, 5
20210002,  EE477,  95,   3.5
20210002,  EE209,  59,   5
20210002,  EE412,  70,   4
```

Please write an SQL query to create the *grades* table and insert the above data (six rows) into the *grades* table. Note that the table has to meet the following requirements.

- studentid - integer
- course - char(10)
- mark and credit - double
- (studentid, course) is Primary Key
- studentid references students.id

(3) Create a table named *students*

The *students* table has four columns and two rows as shown below.

```
id,        name, gender, age
20210001,  Roh,  Female, 26
20210002,  Tae,  Male,   23
```

Please write an SQL query to create the *students* table and insert the above data (2 rows) into the *students* table. Note that the table has to meet the following requirements.

- id, age - integer
- name - varchar(30)
- gender - char(6)
- id is a primary key
- name cannot be NULL

[Task 2] Using SQL to query a database

(1) Please write an SQL query to show all rows in the *grades* table

(2) Please write an SQL query to show the rows whose course is "EE477" in the *grades* table

(3) Please write an SQL query to show the rows whose mark is larger than 60 and credit is no smaller than 4 in the *grades* table

(4) Please write an SQL query to show the rows whose course starts with "EE4" in the *grades* table.

(5) Please write an SQL query to show studentid, course and mark of all rows in the *grades* table

(6) Please write an SQL query to show distinct course of all rows in the *grades* table

(7) Please write an SQL query to show studentid, course and markpoint of all rows in the grades table. markpoint is defined as markpoint = mark * credit.

(8) Please write an SQL query to find the students who have taken "EE477" and show their name, mark.

(9) Please write an SQL query to compute lettergrade of each row in the *grades* table, and show studentid, course and lettergrade of all rows in the *grades* table. lettergrade is computed as follows (use a CASE Expression):

- If mark $\geq$ 90, then lettergrade = "A"
- If 80 $\leq$ mark $<$ 90, then lettergrade = "B"
- If 70 $\leq$ mark $<$ 80, then lettergrade = "C"
- If 60 $\leq$ mark $<$ 70, then lettergrade = "D"

- If mark < 60, then lettergrade = "F"

Export Query Output

For each question, output your query results into a CSV file on Google Storage, and download the CSV file to your local machine for submission. To use Google Storage, create a bucket using either a console or cloud sdk.

- Console : https://cloud.google.com/storage/docs/quickstart-console
- Cloud SDK : https://cloud.google.com/storage/docs/quickstart-gsutil

After creating a bucket, you can export the output of each [Task 2] problem in CSV format to the bucket: https://cloud.google.com/sql/docs/mysql/import-export/exporting. Then, download each CSV file to your local folder for submission. The download method depends on your choice (Console or Cloud SDK) and is described in the links above.