

EECS118

Fall 2019

Mini-Project 1 - Database Programming

Assigned on: 10/03/2019

In this project, you will learn how to write a Python program that can interact with a database. The project is due by 11:59PM, 10/17/2019.

Deliverables

1. Source code of your Python program that can interact with MySQL.
2. Output of your program.

STEP 0 – Install Python 3.6.8

All projects in this class will be based on Python 3.6.8 (other versions will not work). You can download the executable installer from [here](#).

STEP 1 – Install Python MySQL Client PyMySQL

To connect MySQL from Python, you can install PyMySQL library from pip. To do so, use the command “pip install pymysql”.

We provide some code snippets below. You can start to modify the code to do this project and provide the requested functions described at the end of this document.

- Sample Code -

In the beginning of the py file, import the library you need:

```
import pymysql
```

To Create a Connection with MySQL Server for the database “sample_python”:

```
db = pymysql.connect(host='localhost',  
                    user='root',  
                    passwd='xxxxxx',  
                    db='sample_python')  
cur = db.cursor()
```

In the host part, ‘localhost’ is the address of the server (127.0.0.1 means the localhost which is your own computer; if you want to connect to other computers just change the address), and “sample_python” is the database name. This code snippet uses user id “root” and password “xxxxxx” to login.

To Perform SQL Queries:

```
sql="SELECT * FROM customer"  
cur.execute(sql)
```

To Get the Results:

```
for row in cur.fetchall():      # cur.fetchone() gets one result at a time
```

```
print(row)
```

To Add a Tuple into a Table:

```
sql = ("INSERT INTO bank.customer(customer_ID, customer_name)
VALUES(default, %s)" # %s is a place holder for inserting a variable here
val = (cust_name) # customer name is stored in variable cust_name
cur.execute(sql, val)
cur.commit() #use commit to save the changes you made to the database
```

Note that:

- “default” means to use the default value for customer_ID, which is auto incremented.

Use `db.close()` to end the connection.

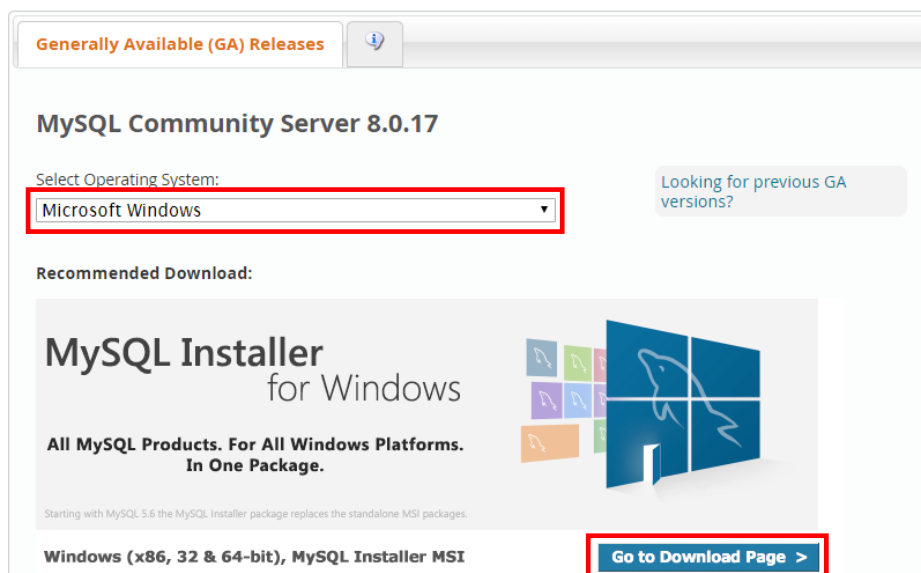
If you are new to SQL queries, you may find [w3schools](https://www.w3schools.com/sql/) helpful.

STEP 2 – Install MySQL Workbench

Go to MySQL download page to download MySQL Community Server 8.0
<https://dev.mysql.com/downloads/mysql/>

Select the correct installer based on your system

◀ MySQL Community Server



MySQL Community Downloads

MySQL Installer

Generally Available (GA) Releases

MySQL Installer 8.0.17

Select Operating System:
Microsoft Windows

Looking for previous GA versions?

Windows (x86, 32-bit), MSI Installer (mysql-installer-web-community-8.0.17.0.msi)	8.0.17	18.5M	Download
Windows (x86, 32-bit), MSI Installer (mysql-installer-community-8.0.17.0.msi)	8.0.17	393.4M	Download

We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.

Choose Developer Default setup type.

MySQL Installer
Adding Community

License Agreement
Choosing a Setup Type
Installation
Installation Complete

Choosing a Setup Type

Please select the Setup Type that suits your use case.

☒ **Developer Default**
Installs all products needed for MySQL development purposes.

☐ **Server only**
Installs only the MySQL Server product.

☐ **Client only**
Installs only the MySQL Client products, without a server.

☐ **Full**
Installs all included MySQL products and features.

☐ **Custom**
Manually select the products that should be installed on the system.

Setup Type Description
Installs the MySQL Server and the tools required for MySQL application development. This is useful if you intend to develop applications for an existing server.
This Setup Type includes:
* MySQL Server
* MySQL Shell
The new MySQL client application to manage MySQL Servers and InnoDB cluster instances.
* MySQL Router
High availability router daemon for InnoDB cluster setups to be installed on application nodes.
* MySQL Workbench
The GUI application to develop for and manage the server.
* MySQL for Excel

< Back Next > Cancel

Choose Standard MySQL Server

MySQL Installer

MySQL Server 8.0.17

High Availability

Type and Networking

Authentication Method

Accounts and Roles

Windows Service

Logging Options

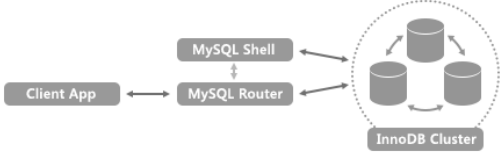
Advanced Options

Apply Configuration

High Availability

☒ **Standalone MySQL Server / Classic MySQL Replication**
Choose this option to run the MySQL instance as a standalone database server with the opportunity to configure classic replication later. With this option, you can provide your own high-availability solution, if required.

☐ **InnoDB Cluster**
The InnoDB cluster technology provides an out-of-the-box high availability (HA) solution for MySQL using Group Replication.



Note: [InnoDB cluster](#) requires a minimum of three MySQL server instances to provide a fully automated HA solution. Members of a cluster should be located such that network communication latency between servers is low.

Next > Cancel

Choose a type of computer; you can select developer or server.
The default port is 3306 unless you want to change it.

MySQL Installer

MySQL Server 8.0.17

High Availability

Type and Networking

Authentication Method

Accounts and Roles

Windows Service

Apply Configuration

Type and Networking

Server Configuration Type

Choose the correct server configuration type for this MySQL Server installation. This setting will define how much system resources are assigned to the MySQL Server instance.

Config Type: Development Computer

Connectivity

Use the following controls to select how you would like to connect to this server.

☒ TCP/IP Port: 3306 X Protocol Port: 33060

☒ Open Windows Firewall ports for network access

☐ Named Pipe Pipe Name: MYSQL

☐ Shared Memory Memory Name: MYSQL

Advanced Configuration

Select the check box below to get additional configuration pages where you can set advanced and logging options for this server instance.

☐ Show Advanced and Logging Options

< Back Next > Cancel

Choose “Use strong password for authentication”

MySQL Installer

MySQL Server 8.0.17

High Availability

Type and Networking

Authentication Method

Accounts and Roles


Windows Service

Apply Configuration

Authentication Method

☒ **Use Strong Password Encryption for Authentication (RECOMMENDED)**

MySQL 8 supports a new authentication based on improved stronger SHA256-based password methods. It is recommended that all new MySQL Server installations use this method going forward.

 Attention: This new authentication plugin on the server side requires new versions of connectors and clients which add support for this new 8.0 default authentication (caching_sha2_password authentication).

Currently MySQL 8.0 Connectors and community drivers which use libmysqlclient 8.0 support this new method. If clients and applications cannot be updated to support this new authentication method, the MySQL 8.0 Server can be configured to use the legacy MySQL Authentication Method below.

☐ **Use Legacy Authentication Method (Retain MySQL 5.x Compatibility)**

Using the old MySQL 5.x legacy authentication method should only be considered in the following cases:

- If applications cannot be updated to use MySQL 8 enabled Connectors and drivers.
- For cases where re-compilation of an existing application is not feasible.
- An updated, language specific connector or driver is not yet available.

Security Guidance: When possible, we highly recommend taking needed steps towards upgrading your applications, libraries, and database servers to the new stronger authentication. This new method will significantly improve your security.

< Back

Next >

Cancel

Come up with a password. check the password, then Next.

MySQL Installer

MySQL Server 8.0.17

High Availability

Type and Networking

Authentication Method

Accounts and Roles


Windows Service

Apply Configuration

Accounts and Roles

Root Account Password

Enter the password for the root account. Please remember to store this password in a secure place.

Current Root Password: 

< Back

Next >

Cancel

Add Windows service

MySQL Installer

MySQL Server 8.0.17

High Availability

Type and Networking

Authentication Method

Accounts and Roles

Windows Service

Apply Configuration

Windows Service

☒ Configure MySQL Server as a Windows Service

Windows Service Details
Please specify a Windows Service name to be used for this MySQL Server instance. A unique name is required for each instance.

Windows Service Name:

☒ Start the MySQL Server at System Startup

Run Windows Service as ...
The MySQL Server needs to run under a given user account. Based on the security requirements of your system you need to pick one of the options below.

☒ Standard System Account
Recommended for most scenarios.

☐ Custom User
An existing user account can be selected for advanced scenarios.

< Back

Next >

Cancel

There is no need to bootstrap MySQL Router with InnoDB cluster

MySQL Installer

MySQL Router 8.0.17

MySQL Router Configuration

MySQL Router Configuration

☐ Bootstrap MySQL Router for use with InnoDB cluster

This wizard can bootstrap MySQL Router to direct traffic between MySQL applications and a MySQL InnoDB cluster. Applications that connect to the router will be automatically directed to an available read/write or read-only member of the cluster.

The bootstrapping process requires a connection to the InnoDB cluster. In order to register the MySQL Router for monitoring, use the current Read/Write instance of the cluster.

Hostname:

Port:

Management User:

Password:

MySQL Router requires specification of a base port (between 80 and 65532). The first port is used for classic read/write connections. The other ports are computed sequentially after the first port. If any port is indicated to be in use, please change the base port.

Classic MySQL protocol connections to InnoDB cluster:

Read/Write:

Read Only:

MySQL X protocol connections to InnoDB cluster:

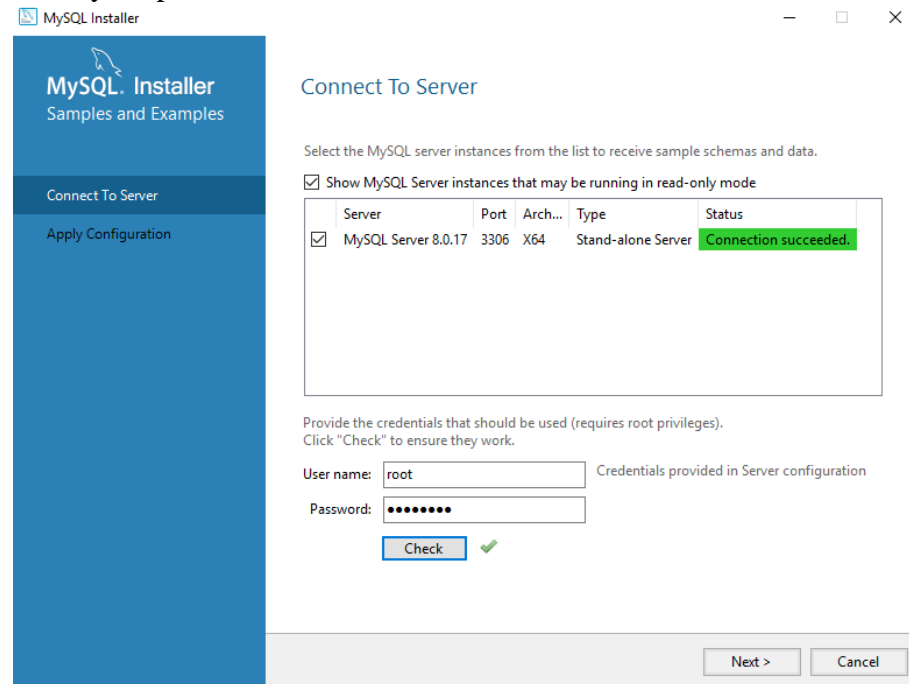
Read/Write:

Read Only:

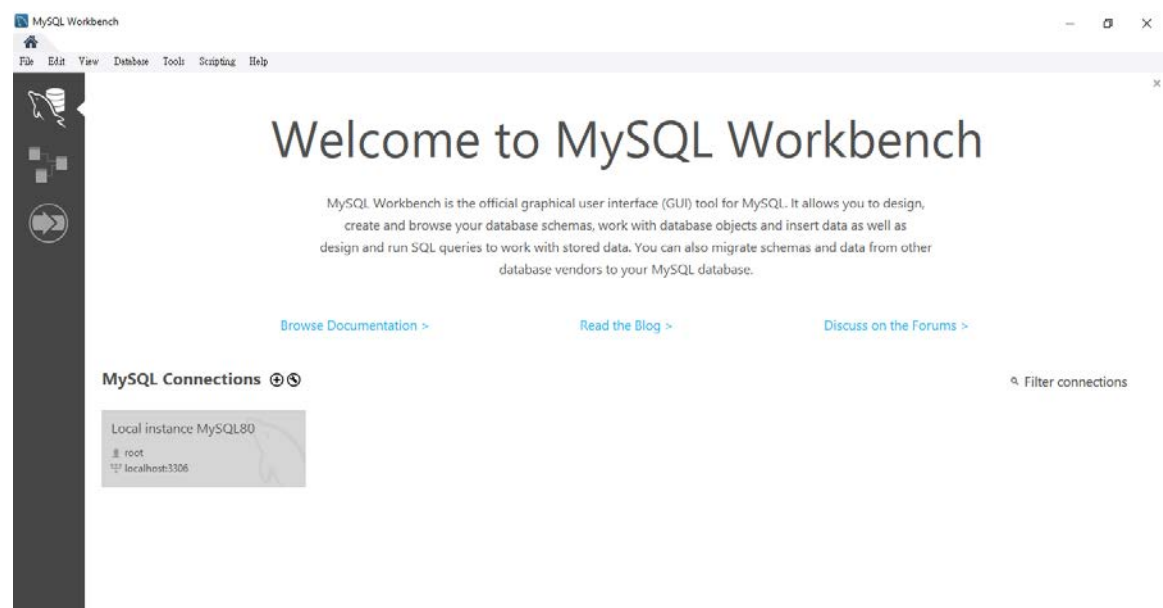
Finish

Cancel

Enter your password.



After the installation, run MySQL Workbench and you should see the following:



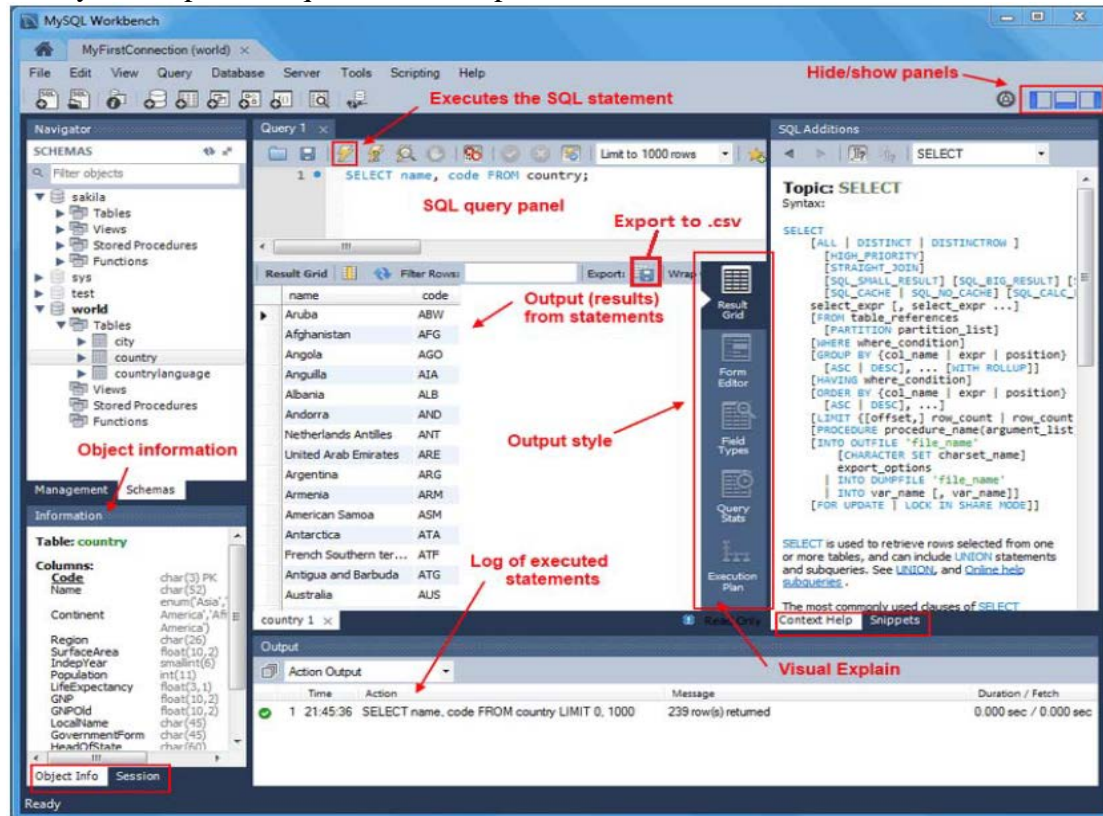
Double click the local instance to connect. Note that the port should be the same as the port when you installed MySQL (3306 in the case). If not, edit the connection by:

Windows and Linux: hover over the right side of a connection title and click the title.

OS X: hover over a connection title and click the little (i) in appears in the bottom right corner.

Type in your password, and connect to the database.

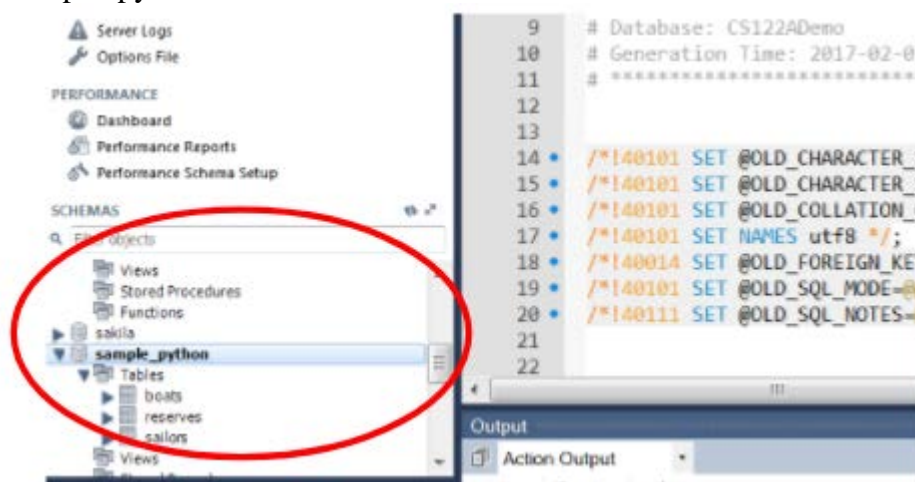
Now you can perform queries or manipulate database:



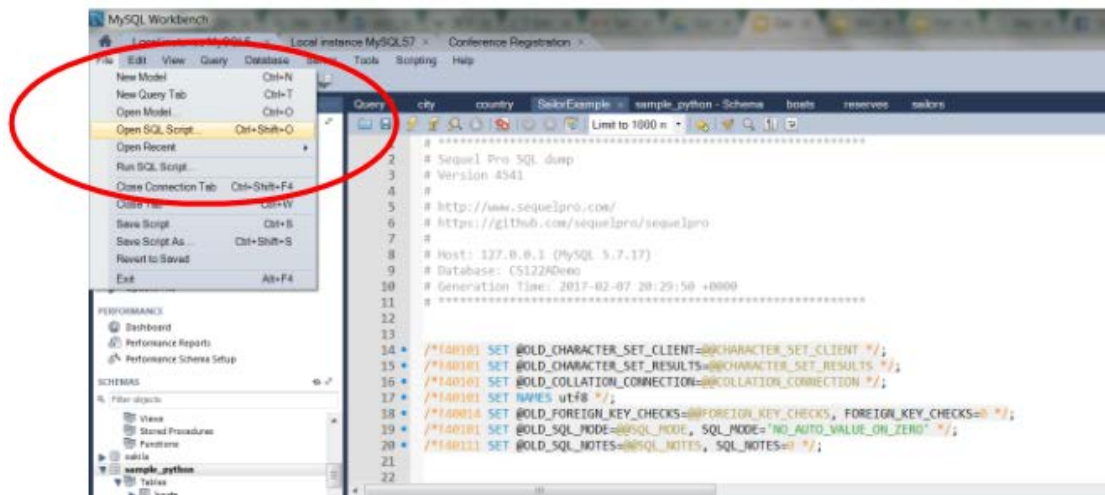
STEP 3 – Write the program

Use your localhost in MySQL as the server. Download the SQL file we prepared (project1_question.sql) [here](#).

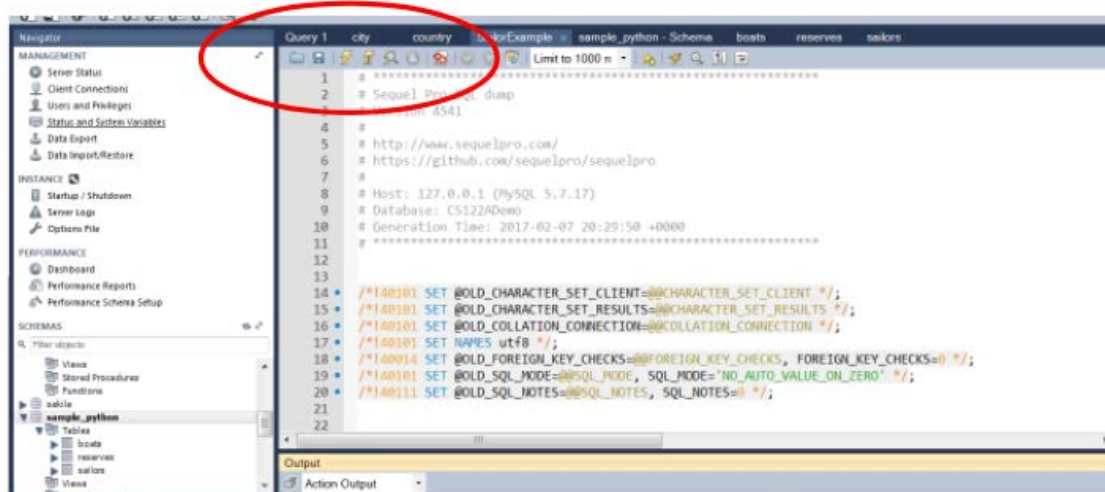
Open your MySQL. In the schemas area, right click to create a schema name it anything you prefer (here we use sample_python), and then double click on the “sample_python” to make it a default DB to be used.



Click File - Open SQL Script. Choose the Project1.sql you downloaded.



Click the “thunderbolt” to execute the script



Click the refresh button to refresh the schemas. Then check the schema "sample_python" to see if tables "question" and "result" are there. Then, the dataset is ready!

Remember the database is called sample_python or anything you named. Use username and password you created to login. Modify the corresponding URL (IP address and database name), user name and password in the code template.

There are two tables in the database “project1”:

- question (name, A, B)
- result (name, id2d, result)

Write a program to do the following functions in order:

1. In table “question”, there is already some data: A list of student names and two random numbers A and B. Your program should print out all the data in this table. Example output:

question:

```
Smith, John, 3156, 9327
Wang, David, 1357, 8642
...
```

2. Find your own name in table “question”, get the numbers A and B, then calculate $A * B + \text{the last 2 digits of your student id}$, e.g., if $A=3156$, $B=9327$, your student id is 12345678. The calculation result will be $3156*9327+78=29436090$. Insert your **name**, **last 2 digits of your student id** and **the result** to the “result” table. If you want to execute your program multiple times and doesn’t want to see errors of trying to insert duplicate entries, you may use “INSERT IGNORE INTO” statement, which will do nothing if there is already the same entry in the table.

Note: You won’t be able to use “DELETE” statement as we don’t want any accident, e.g. the data is all deleted by some student. It may be painful if you insert an incorrect result and want to modify it so make sure your result is correct before you insert. If you have to modify your result, you will need to use “UPDATE”. (UPDATE table_name SET column1=value1, column2=value2 ... WHERE ...)

3. Form a query to find your name and the result from the “result” table, print them out. Example output:

```
result:
Smith, John, 29436090
```

Put your outputs to a text file using the name “**output.txt**”, and then archive with your source code (better using the name **project1.py**) to the file “**118MP1-xxxxxxx.zip**”, xxxxxxxx being your student id, and turn it in on EEE dropbox under folder “**mp1**”. **Note: Not submitting the file under the correct folder may cause a deduction in your credit.**