

**[Use Case 1]**

Suppose I (Charles), Ting and Alan are taking this class, and there is the sample database of our info stored into MySQL.

**[Note]:**

- 1) The following Q1~Q8 are somewhat related! It has to be finished in the previous steps, otherwise, we cannot move onto the next one. Please read through Q1~Q8 first! Not doing the whole homework one-by-one!
- 2) For some of the questions, I need screenshots. You can just directly pasted into Word files or submit image files. Either way.
- 3) For some of the questions, I need SQL code
- 4) For Android App, I need source code. Please zip it up into a zip file for the whole Android Studio folder
- 5) There are some of the detail you don't need to deal with. For example, (Lat, Long), it is because I might like to re-use this project into some other courses in the future. For example, Mobile Apps Development.
- 6) Some details you might need to deal with. For example, how to deal with such kind of floating points, for example (Lat, Long) ? I don't really care about anything related with losing its precision due to the casting, ..., etc.

**[Database name] Team\_Member; Primary key: Student\_ID**

**[Table1 name] TeamMember; Primary key: Student\_ID**

Student_ID	First_Name	Last_Name
1	Charles	Yu
2	Ting	Cao
3	Alan	Garvey

**[Table2 name] SecretPlace**

Student_ID	Lat	Long
1	40.21672516295085	-92.57955453022424
1	40.19751402638587	-92.56949237863425
2	40.19575800686065	-92.58376106777236
3	40.19379440711938	-92.58579646943969
3	40.19453756370904	-92.58445139940807
3	40.21672516295085	-92.57955453022424

### Q1: schema

Use the MySQL Workbench to setup the original database, Team\_Member, like we mentioned above. (including 2 tables and those rows populated)

[Submit]: I need a screenshot here. No need for SQL code

### Q2: Query

I want to make a query from the MySQL workbench, I want to show ALL of us (3 students, so far), the columns are listed as {Student\_ID, First\_Name, Last\_Name, Lat, Long}

In this question, I need the SQL query to show how you do that?

[Submit]:

- a) I need screenshots here
- b) I need SQL query

### Q3: PHP and Query

We need to setup Apache, PHP, MySQL. Make sure that your PHP can make the query to the MySQL database and show the result onto the HTML webpage (in your browser). I will assume your PHP webpage, Apache, MySQL are all installed onto the SAME machine. I need the PHP source file

[Submit]:

- a) I need screenshots here. In your webpage, you need to show me the JSON file returned
- b) I need your PHP source code

### Q4: SQLite Studio

Still remember our “android\_connetion\_demo” database in our class demo?

Create a database using the name exactly THE SAME (Team\_Member) as we described above.

Then, create 1 table named: “QueryResult”, into the Team\_Member database (Yes! Just one!)

The 5 columns are listed as {Student\_ID, First\_Name, Last\_Name, Lat, Long}

Do not populate any data at this step.

This table will contain exactly, 5 columns.

[Submit]:

- a) Show me the screenshot will be good enough.

#### **Q5: Android Query the MySQL**

Develop an Android app and use the volley (or you can use “retrofit”), along with your PHP file to query the MySQL. The query command is the same! We just want to have everything from 2 tables and the result is in 5 columns:

{Student\_ID, First\_Name, Last\_Name, Lat, Long}

The Android App can be in the physical phone or in the emulator.

After we get the data, 5 columns and 6 rows, we can move onto Q6 for data insertion

[Submit]:

- a) Nothing to submit here. Just getting ready for your final source code!

#### **Q6: Android write data into SQLite**

The Android app can write the data “from MySQL database” into SQLite now!

Once we have the JSON data (from Q5), we can write the tuples into SQLite database as we created in Q4. So basically, it will be 6 rows (tuples) in total. All of them, have to be inserted into 1 table.

[Submit]:

- a) Nothing to submit here. Just getting ready for your final source code!

[Note]

1. When we are doing the development job, we can drag and drop the SQLite database file (created in Q4) onto the Android Studio, under the “/assets” folder (you need to create the folder)
2. The sample code about “how to write data” into SQLite database will be provided in the Homework3 as an attached zip file.
3. The way my installation the SQLite database is to install that with the Android App. You can try (optional) to install to some other PATH (folder). Try this link:  
<https://stackoverflow.com/questions/14376807/read-write-string-from-to-a-file-in-android>

#### **Q7: Android Create table, insert data into SQLite**

Put your team member’s info into the SQLite, same database file, by creating the 2nd table in SQLite programmatically. You can just “hard-code” that in the query string and execute the insertion in the Android App. The 2nd table is isolated without any relationship with the 1<sup>st</sup> table in the SQLite.

You remember the 1<sup>st</sup> table in SQLite is “QueryResult”, with the schema: {Student\_ID, First\_Name, Last\_Name, Lat, Long}

But this table is called “OurMember” which is isolated

[Table3 name] OurMember

Student_ID	First_Name	Last_Name

[Submit]:

- a) Nothing to submit here. Just getting ready for your final source code!

Q8: Show everything onto your UI

In your App, the UI needs to show up all the data either you can show that onto “dynamically” created “EditText” Java class or like my demo, “Toast” Java class

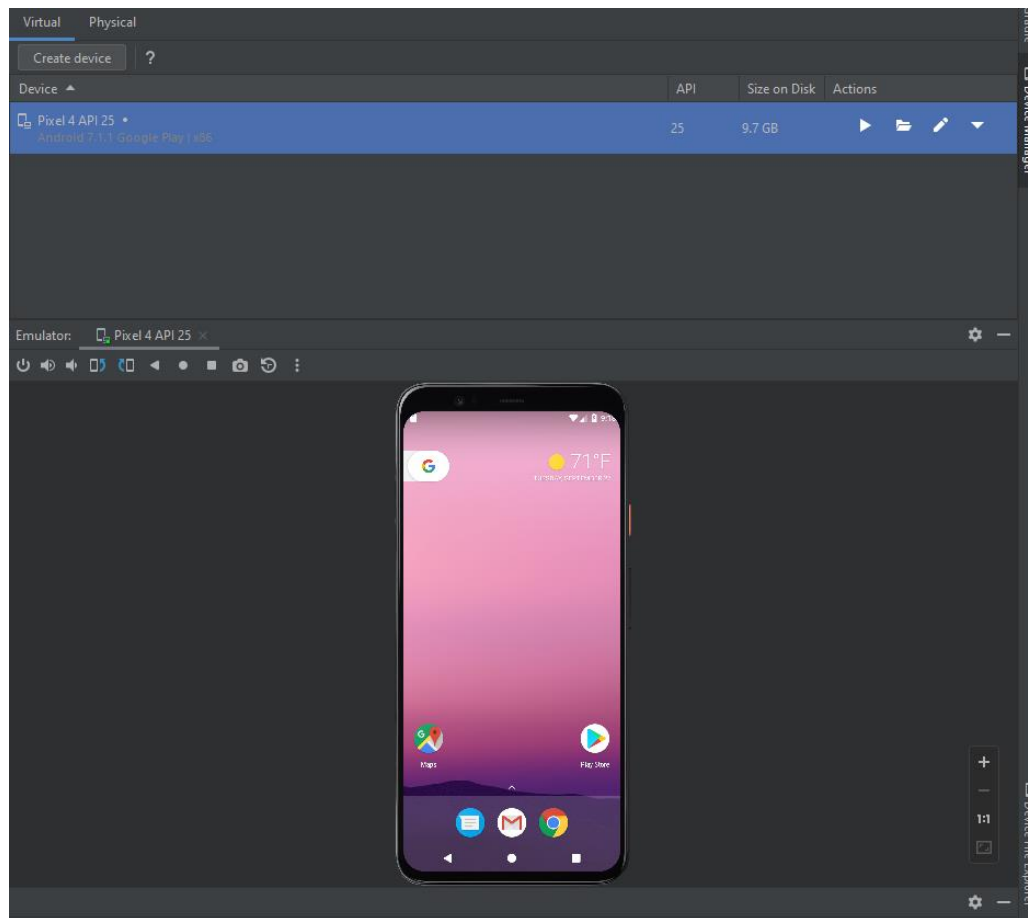
[Submit]:

- a) I need your Android Studio project zip into a file here!

Q9: Save the SQLite file as a copy to our computer / laptop

Now, we can suppose your SQLite file is DONE!

There is a thing in the Android Studio called “Device File Explorer”.

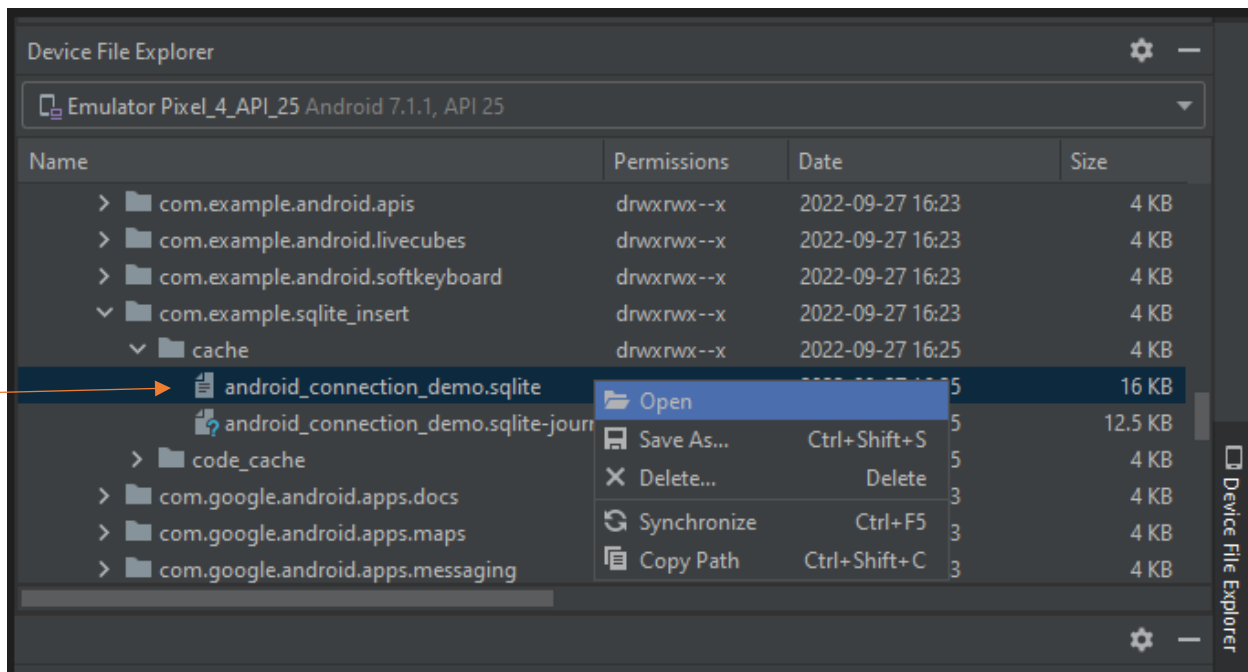


In our example, there is a path: `/data/data/com.example.sqlite_insert/cache`

Then, you can find your fully populated database while, whole database.

Right click the file and “Save As”. You can save a new copy onto your desktop / laptop computer.

If you are in the “middle of debugging / developing”, you can use this way to “save as” the db file to the desktop / laptop computer at any time! Then, you can use the SQLite Studio to observe if the data is correctly populated?



**[Submit]:**

- a) I need your SQLite database file as well as screenshot. You can open it up in SQLite and take the screenshot