CSC2008: Database Systems
Lab 4: Remote Database Access to Raspberry Pi

Objectives
To learn remote database access and operation to Raspberry Pi.

Deliverables
You are required to finish the lab tasks listed below and submit your report. All tasks shall be completed.

Format: **ONE (1) report in PDF.**
Filename: <lab id>_<student id>_<name>_Lab4.pdf,
e.g., **P0_2102008_ZhangWei_Lab4.pdf.**
Venue: Dropbox in xSiTe.
Deadline: the end of the week (Sunday, 11:30pm) when the lab is conducted.
Penalty: A penalty of **10% per day** for late submission will be imposed. A penalty of 100% for this lab will be imposed for the 1st time plagiarism and a penalty of 100% for ALL the labs for the 2nd time plagiarism.

Hardware, Software, and References
Raspberry Pi.
MySQL Workbench.

Background
In the last lab, you learnt how to set up a database in Raspberry and how to perform some basic database operations. Consider a realistic setting, you may not have the privilege to perform local access and operation in your edge devices, e.g., you do not even have a specific keyboard or screen. Thus, in this lab, we would like to practice the database access control and operations and network settings in a remote computer, e.g., your laptop.

(Note: You are encouraged to go beyond our lecture materials to explore the latest and advanced database technologies.)

Tasks:
First, please finish the below tasks in your Raspberry Pi.

Q1. Log into your Raspberry Pi MariaDB using the root account (hope you did not forget your root password). Create a **new user** with your name (e.g., csc2008wei) that applies to **every IP address** with a **password.** (Note: Having an account applies to every IP address is not a good practice in many realistic applications. Here we do this for simplicity, as you may lack the IP information in this stage.)

Q2. Grant privileges to the user you just created to access MariaDB. Be noted that your system is still in a temporary state and you shall commit the action to make it effective.
(Note: You may grant the privileges to all the databases for simplicity but be noted that it may not be a good practice in many realistic applications.)

Q3. By default, your Raspberry Pi database is (partially) not listening to the external signals, for resource-saving and security considerations. To enable remote access, you have to update the range of IP addresses to listen.
(Hint: You may look for the configurations in /etc/mysql/mariadb.conf.d/50-server.cnf.)

Q4. Double-check if your Raspberry Pi database is running. If so, check the **IP address** of your Raspberry Pi.

Q5. Check the port number for your RaspberryPi MariaDB environment.

Now, we have the necessary information in Raspberry Pi and we can move to a remote computer, e.g., our laptop.

Q6. Open MySQL Workbench. (You shall download and install first if you have not done so.)

Q7. Set up a new connection in MySQL Workbench with the Raspberry Pi's information (e.g., username, password, IP, and port). You may use TCP/IP for the connection.

Q8. Check if the connection has been successfully established. If so, you are able to perform remote access and control to your Raspberry Pi database already.

Q9. In your MySQL Workbench, perform some basic SQL operations we tested in the previous labs. For example, you can create a new database and a new table in the database. You can insert some rows into the table and update them. Finally, you can delete some rows, your table, and your new test database.
(Note: Just one screenshot in your MySQL Workbench here is enough to demonstrate you can perform remote access and operations.)

Q10. (optional) You may perform some tests and profile analysis to see if the database operation speed is affected by remote access. Such tests can also be part of your team project.

--- END ---