Homework Assignment #1
Database Systems – Semester A 2024/2025

This assignment consists of two parts:

part A - This part is technical, and you need to follow the instructions in order to set up the environment on which you'll run the database.

Part B - Contains questions that you are required to answer and submit on Lamda.

Part A - Installation

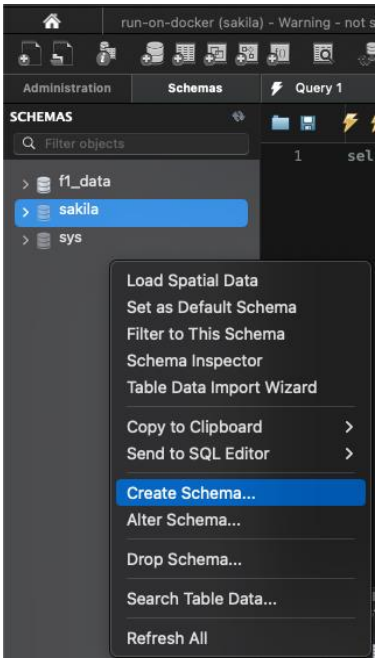
Installation

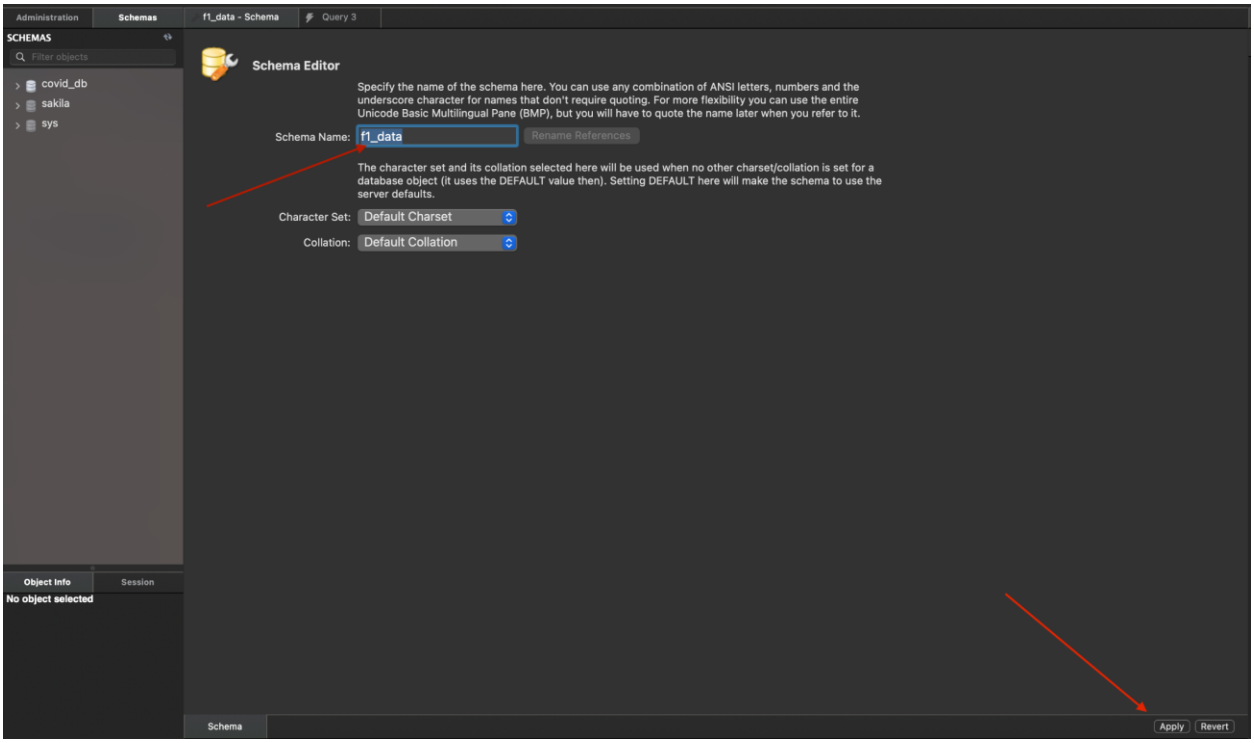
Before you start

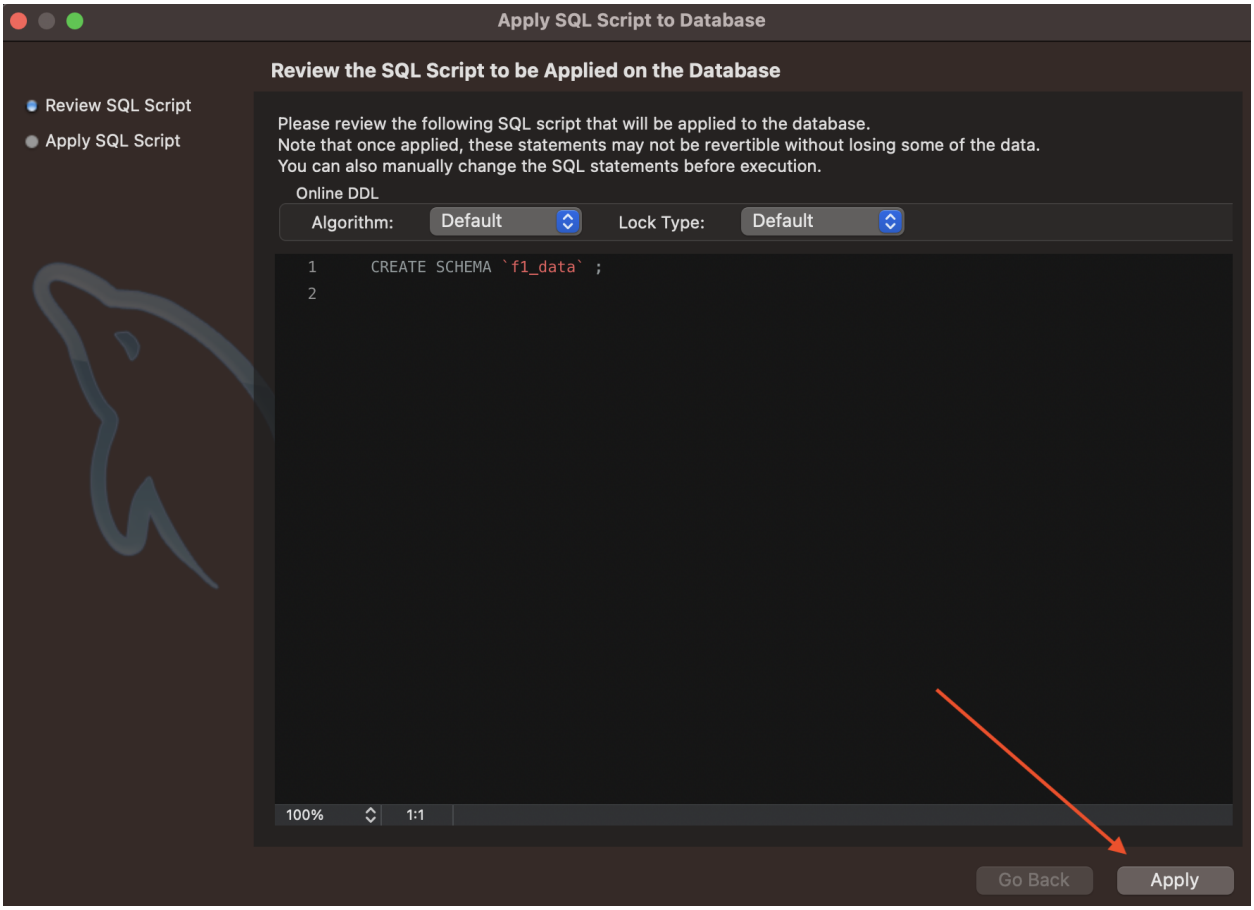
- Make sure you followed Exercise 0 before this exercise.
- Download the files from moodle.

Then, you can either follow this youtube using MySql workbench, or follow these instructions:

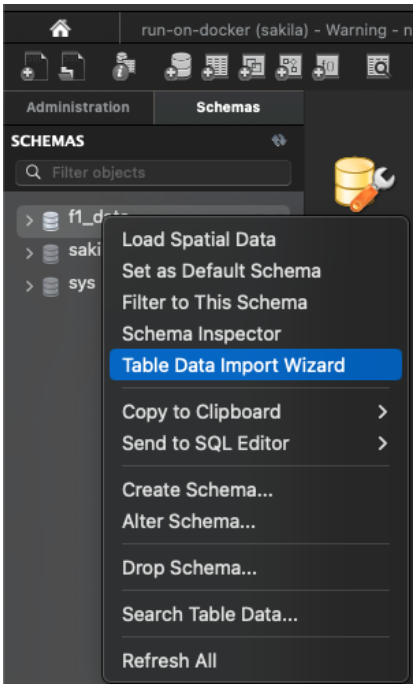
1. Load your SQL workbench using docker as we learned in class
2. Create a new schema and call it "f1_data" in your workbench

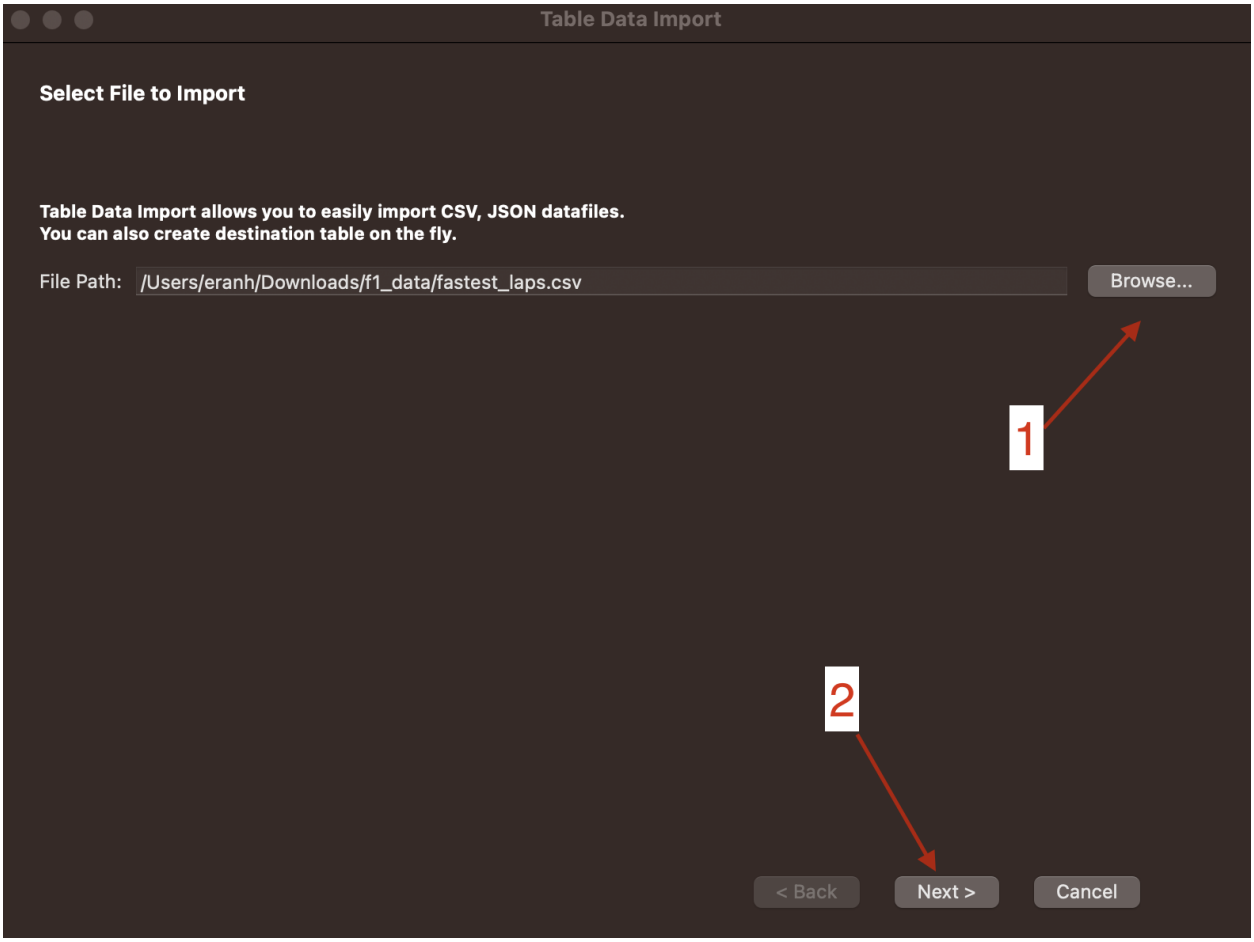




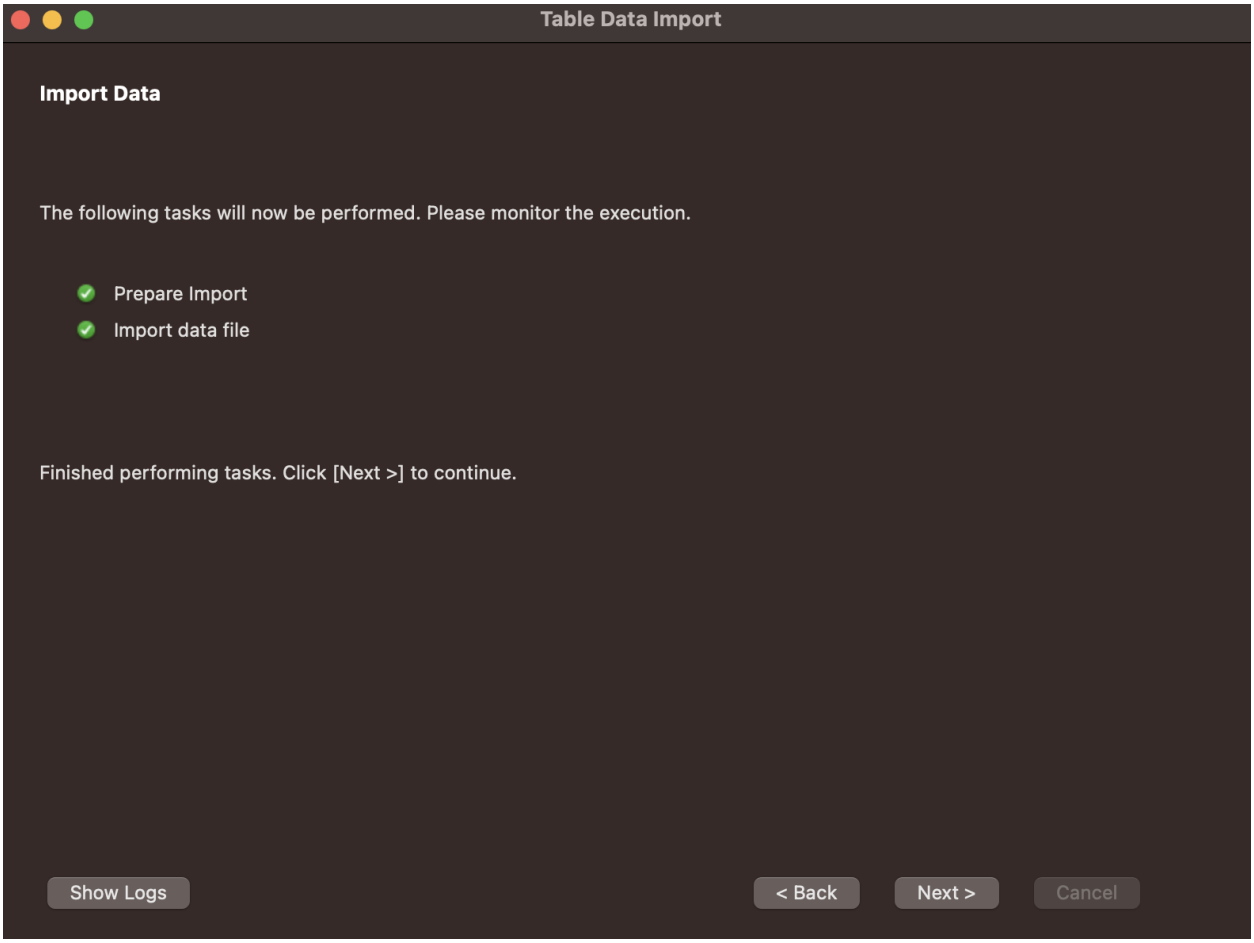


3. Load each of the CSV tables to the schema



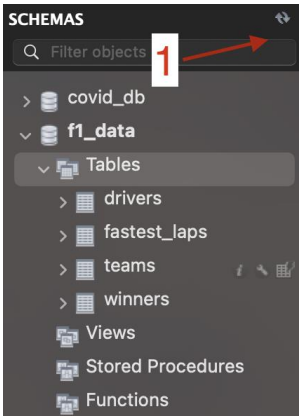


Keep pressing next until you see this screen:



Do this again for each table in moodle.

At the end, you should see all tables. You might have to click the refresh button to see them.



Credit: the data's original source is kaggle .

Part B - SQL Queries

- Your final submission should contain a zip file containing all submission requirements and a details.txt file that should contain the ID and name of each submitting student, such as:

first_name last_name 123456789
first_name last_name 987654321

- The zip file should be named with your ID, such as “123456789.zip”. If multiple students are submitting, then separate the ID numbers with underscore, such as “123456789_987654321.zip”.
- In this assignment, your zip file needs to contain 8 .py files, make sure you don’t add any other unnecessary files to the zip.
- Submission is through the course website and can be either individually or in pairs – if you submit in pairs only one of the two should submit and that student will also be the one to receive feedback. A submission for three people is not permitted, if you are unable to find a pair then you should submit by yourself (it is not necessary to ask for permissions to submit by yourself).

Objective

Understanding and experiencing writing and executing SQL queries.

Data

All the questions in this part will assume the data in part A.

Requirements

For each question you are required to submit a single file called **q{n}.py** (for example q1.py, q2.py etc.) that will include the following:

- The SQL query you used.
- Documentation - **mandatory**. Use ‘#’ to add comments.
- Each of the python files need to be in this format:

```
import mysql.connector

if __name__ == '__main__':
   mydb = mysql.connector.connect(
        host="localhost",
        user="root",
        password="root",
        database="f1_data",
        port='3307',
   )
```

```
cursor = mydb.cursor()
cursor.execute("""
   ## PUT YOUR QUERY ##
""")
print(', '.join(str(row) for row in cursor.fetchall()))
```

Important Notes

- The format is important because the testing is automatic. If the automatic script fails because you changed the format, you will not get any points for the question.
- Do not use commands that have not yet been learned, or will not be learned at all - a query that uses such commands, **all points will be deducted**.
- Please make sure that the queries are well formatted (use tabs and newlines, parentheses etc.) to make them readable (See the example format).
- You should assume that the data we are using is different from yours, do not make any assumptions on the data.
    - For example, if we ask who was the winner in 2023, do not write a query which contains the value “Max Verstaphen”, because you already know he won. We might use an alternated version of the data, where the winner is someone else.

Example Format

```
SELECT      film_id, title
FROM        film
WHERE       length > 10
ORDER BY    title ASC
```

Tips

- Some columns have a space ‘ ‘ in their names. For example, `Grand Prix`. You need to use backticks when referencing these columns. Examples:
    - SELECT `Grand Prix` FROM winners;
    - SELECT w.`Grand Prix` FROM winners AS w;
- In some questions you should use the function YEAR, which returns the YEAR out of a date column.
- In the **fastest_laps** table, notice the time format is “mm:ss”.
  Use the following query to transform the fastest_laps Time column to minute format
    - SELECT *, MINUTE(STR_TO_DATE(Time, '%i:%s.%f')) FROM fastest_laps

Questions

1. How many drivers start with the letter A? Return one numerical cell with the result

2. How many drivers end with the letter N? Return one numerical cell with the result

3. Find the driver who won in one of the f1 contests in the year 2000, and had the most amount of laps. Print his name and the minimum of his fastest lap time. The answer should contain the driver name and the minimum time as “min_time”.

4. What is the total amount of wins in 2001, for the team who had the most wins in 2000? Return a single numerical cell.

5. Write a query to find the average amount of points for each car, for cars whose fastest lap time was less than 2 minutes. Order the average points in descending order. The answer should contain both the car and the average points as “avg_pts”.

6. Find all pairs of Grand Prix tournaments that have the exact same amount of laps, and that this amount is greater than 100. Do not return duplicates. Return the following columns: GP1, GP2, Laps. Make sure that GP1 is alphabetically before GP2 (for example, <*Apple, Banana*> and not <*Banana, Apple*>).

7. Return drivers who **won** with Alfa Romeo or drivers who have a Japanese (JPN) nationality. Sort the results alphabetically. Do not return duplicates. Return a table with one column “driver”.`

8. Find the difference between the number of total points of the 'Ferrari' car team and the 'Mercedes' car team. The result should contain one cell with the result called “diff”.

9. Write a query such that for each nationality of the drivers, will find:
    a. The average points of all drivers from each nationality
    b. The minimum fastest laps from each country ever
    c. The latest date where this nationality won an F1 race
    d. Note: The answer should contain the Nationality, the average of points as “avg_pts”, the fastest lap time as “min_time” and the last date as “latest”

Running python programs

To start the program, running the following line from the command line should work:

```
>  python q1.py
```

External python packages

Your code should run only with the *MySQL Connector/Python* package. If you want to use any other library, you first need to get approval by writing a message in the moodle course forum. Use only the moodle course forum, so we can approve or reject it once for all the students.