# Homework Assignment #1
# Database Systems – Fall 2023/2024

- 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).
  - Use the database that you have created in the previous assignment

## SQL Queries

### 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.
- Any assumptions you made as notes (with preceding ‘#’). If you think that the question can be understood in more than one way, explain according to which interpretation you solved it.
- Documentation - **mandatory**.
- 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="covid_db",
       port='3307',
   )
   cursor = mydb.cursor()
   cursor.execute("""
      ## PUT YOUR QUERY ##
   """)
   print(', '.join(str(row) for row in cursor.fetchall()))
```

## Important Notes

- Your query must return the answer to the question exactly; no more and no less attributes or rows.
- 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).

## Example Format

```
# Any assumptions you made regarding q1 should be written here...

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

# Questions

1. Find the difference between the number of total cases in February (across all years) and in March(February-March).

2. Write a query to find dates where 2 locations had the same amount of new cases, only if this amount is greater than 1000, your query should consider only the first 1000 rows from the table. The result should contain the date, amount of new cases, and the 2 locations. Beware of duplicates- the same two locations in a different order.

3. Write a query to find the average amount of new tests, on days where the positive_rate is higher than average (the vaccinations table is dirty- only consider rows where the new_tests and positive_rate attributes aren’t empty(<>”)).

4. Write a query to find the days, where overall, there were more new vaccinations than 20000000 in the continent Asia. The answer should contain both the date and the total amount of new vaccinations as “daily_vaccinations”

5. Write a query to find the amount of new cases, in location that have in some time the highest population, and the average of new cases in this location are greater than 3.

6. Write a query to find the continents in covid deaths table, where the overall amount of new cases is greater than the average number over all the continents, and remove from this calculations the locations in these continents that start with ‘A’ character
7. Write a query such that For each location in covid_deaths, will find:
    a. The monthly average of new cases(called “avg”)
    b. The max monthly amount of new cases(called “max_monthly”)
    c. The min monthly amount of new cases(called “min_monthly”)

   The result should present the location, and all 3 amounts.

8. Write a query that returns the continents which average new cases is greater that the average of new cases of locations where the average population is lower than the global average population

### *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, and the *python-dotenv* 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.