Programming Assignment 1

Contact person: Ilir Jusufi - ilir.jusufi@lnu.se

For this assignment you will need to use Python (version 3.6 or above) and MySQL.

All answers should be your own. You are allowed to work in groups of two. Make sure you include your names in the report when you submit.

NOTE: Follow the submission and implementation guidelines carefully. If your implementation does not adhere to the requirements presented in this document you will **get 0 points!**

Tasks

1. Parse the files and import the data in the database

In the moodle you will find a *zip* file with some *csv* files in it. Your task is to parse the files and insert them into the created database. Your first steps should be to analyze the files and see how they are related to each other in order to design a database schema for the given data. Give it time, as a wrong design might create issues with solving the later tasks of this assignment.

Several important implementation details must be followed at this point:

1. You will need to use the Connector/Python for this assignment. This is mandatory! You can use the following command to do so: (mandatory)
    a.
    ```
    shell> pip install mysql-connector-python
    ```
2. Your code should check if the database with your last name exists. If you are working alone it should be named *Lastname*. If you are doing the assignment in a group it should have both *Lastname1_Lastname2*. (10 points)
3. In case the database exists, the program should ask the user for input i.e. show the *main menu* (explained in the next Task). If it does not exist, it should be created using the instructions in Step 1, create the appropriate tables and insert the data from the *csv* files. (10 points)
4. At the top of your python file (after the *import* statements), you should include the connection details. For example: (mandatory)
    a.
    ```
    cnx = mysql.connector.connect(user='root', password='root',

                                  host='127.0.0.1',

                                  database='jusufi')
    ```
5. Do not forget to comment on your code. (10 points)

2. Queries

After the database has been created and the data inserted, the program should show an interactive prompt. Users should see a main menu as the one displayed in the picture. Each number should perform a certain action and run appropriate queries as described below.

1. **List all planets.** After pressing number *1* followed by enter the program should list the names of all the planets. Users will need to click any key to return to the main menu. (10 points)
2. **Search for planet details.** After pressing number *2* followed by enter the program should ask the user to enter the name of the planet. After the user enters the data and presses enter all the details of the planet should be displayed. Users will need to click any key to return to the main menu. Users will need to click any key to return to the main menu. (10 points)
3. **Search for species with height higher than given number.** After pressing number *3* followed by enter the program should ask the user to enter the average height of the species. The program should list all the species fulfilling the given condition. Users will need to click any key to return to the main menu. (10 points)
4. **What is the most likely desired climate of the given species?** After pressing number *4* followed by enter the program should ask the user to enter the name of the species. After the user enters the data and presses enter the question should be answered. (20 points)
5. **What is the average lifespan per species classification?** After pressing number *5* followed by enter the program should list the names of species classification and their average lifespan. Users will need to click any key to return to the main menu. (20 points)

Submission and deadlines

Your submission should include solutions to all assignments above.

Submit your python file in moodle.

Deadline: End of day 2021-02-24.

IMPORTANT: In case you have a question, try to find the answer in the assignment forum. If the answer/question is not there, ask your question in the forum first instead of writing a direct email to the contact person. This will be the fastest way to get a reply.

GRADING: The lowest score to pass the assignment is 60. The highest score is 100. The parts marked with mandatory tag need to be done. Failure to follow them or any submission instruction will render your contribution to 0 points. If the assignment is unable to run due to error regarding your code, you will not receive any points.