INF 551 – Spring 2020

Homework #3: SQL

Due: April 1, Wednesday (11:59pm)
100 points

Please use Python 3.7 for all assignments.

We will use the "world" database (https://dev.mysql.com/doc/index-other.html) for this homework. Note that it is not the "world-x" database. Please download the SQL script from the above link. Create and populate a "world" database in your MySQL server (either on EC2 or locally on your laptop).

1. [40 points] Write a Python script "export.py" that connects to the "world" database on your MySQL server (using user name "inf551" and password "inf551") and exports the tables in CSV files, one file per table. Name the file after the table: city.csv, country.csv, and countrylanguage.csv. The files should have the same formats as the ones provided to you in the first homework. In other words, it should have a header (1st line) and followed by rows in the table, with columns quoted if values are strings and separated by comma. For example, here shows the content of city.csv.

   ```
   # ID, Name, CountryCode, District, Population
   '1', 'Kabul', 'AFG', 'Kabol', '1780000'
   '2', 'Qandahar', 'AFG', 'Qandahar', '237500'
   '3', 'Herat', 'AFG', 'Herat', '186800'
   …
   ```

   Note that your "export.py" should work in general, that is, it can export contents of databases other than "world". Other databases will be used to test your "export.py".

   You are required to use the "information_schema" which stores the metadata about the databases (to see details, execute "use information_schema" and "show tables"). In particular, use "tables" and "columns" in this database to find out which tables the world database has and a list of columns for each table in that database.

   Execution format: python export.py "world"

   You may use python SQL connector library for this task.

   Q1 Submission: export.py

2. [30 points] Write an SQL query/view for each of the following tasks.
   a. Find the 10 most populated cities in the United States (country name). Output names of such cities and their populations in the descending order of populations.
   b. Find name, continent, and population of countries whose name contains "united" and population is at least 1 million. (note the "like" operator in SQL is NOT case sensitive).

c. Find names of countries which do not have any languages recorded (in the country language table).
    i. Using subquery
    ii. Using outer join

d. Find names of countries which have at least 10 unofficial languages. Output the countries by the descending order of the number of such languages.

e. Create a view “LangCnt” that lists country code and the number of languages (both official and unofficial). Use the view to find the names of countries which has the largest number of both official and unofficial languages.

f. Find the top 10 languages by populations of countries where they are official languages, but only the countries whose populations are over 1 million are counted. For example, English may be spoked in 5 countries as official language and 4 of them each has a population of 2 million and one 500K. Then the total population count for English would 8 million.

Q2 Submission: submit a SQL script for each sub-question, named q2-a.sql, q2-b.sql, q3-ci, q3-cii, etc. The script contains the SQL query/view. Also submit a text file showing all the output for each query.

3. [30 points] Write a Python script “top10.py” (that does not use SQL query, so no SQL connected should be used) that takes the country.csv and countrylanguage.csv (these are in the same format as produced in question 1) as the input, and answers the query in question q2.f (i.e., finding top 10 languages). It should output the same result as the SQL query answer of the question.

Include a discussion (in the comments of your code) on the complexity and efficiency of your algorithm. If particular, assume country table has m rows and countrylanguage has n rows. What is the complexity of your algorithm? Can you further improve its efficiency?

Execution format: python top10.py country.csv countrylanguage.csv

Q3 Submission: top10.py

Final Submission: Please include all three submissions above in one folder and zip it as Firstname_Lastname.zip. Please note wrong submission format or execution format will result in the deduction of 5 points.