Homework 3: Data Modeling and SQL

DSCI 551 – Fall 2024

Due: 11:59pm, October 21, 2024, Monday

Points: 100

Consider a university dataset with the following csv files:

- students.csv (note every student must be assigned to one and only one advisor)
- courses.csv
- enrollments.csv

Assumptions:

- Note advisors are also instructors. Also instructor ids are the same as advisor ids.
- All attributes in the above tables are of **string** types (char/varchar/text) except for id (including all types of ids) and credit hours which are integers.
- The tables you design must contain **all the attributes** in the csv file.
- Your code should run **without error** and satisfy all requirements stated in the question. It's only required for you to define PK, FK and data type for each attribute. You also need to decide when to use "**NOT NULL**".
- You can assume that your code will be tested on a MySQL database called **dsci551**.

1. [10 points] Explain the unnecessary redundancies of data stored in these csv files.
2. [20 points] Create an ER model to capture the information in these csv files. Draw the model, making sure to indicate key attributes, attributes (if any) and multiplicity of relationships, etc. The model should remove the data redundancies observed in the previous question.
3. [10 points] Convert the above ER model into relations/tables. For each relation, write MySQL create table statement to create the relation. Explain if the tables have the redundancies observed earlier in the comments. Make sure primary keys and foreign keys of relations are properly defined.
4. [10 points] Write insert statements to add sample data in the provided CSV file into the relations.
5. [50 points] Using the tables obtained in step 3, for each of the following questions, write an SQL query to answer the question.
    (1) Find full names of students advised by Dr. Smith. Note output one full name (first name followed by a space and the last name) for each such student, e.g., John doe. You can assume that there is only one advisor called Dr. Smith.
    (2) Find names of advisors who advise only one student. Do not use aggregate and group by.
    (3) Find names of advisors who advise only one student. Do use aggregate and group by.
    (4) Find (full) names of students who enroll in at least two different courses. Do not use aggregate and group by.
    (5) Find names of students who enroll in at least two different courses. Do use aggregate and group by.
    (6) Find names of the most popular courses, ranked by the number of students taking the course. Note that there may be more than one such course.

(7) For each student by id, find out his/her total credit hours.
(8) Using outer join, find ids of students who did not take any courses.
(9) Using subquery, find ids of students who did not take any courses.
(10) Find names of students who took both courses 101 and 103.

Submission details:

1. Submit only 5 files:
   - one **.pdf** file containing the explanation for **Q1** and the rendered/hand-drawn ER model for **Q2** called **Q1_2.pdf**
   - one **.sql** file for Q3 called **Q3.sql**
   - one **.sql** file for Q4 called **Q4.sql**
   - one **.sql** file for Q5 called **Q5.sql (The order of your SQL statements in the Q5.sql should be consistent with the order of the questions.)**
2. All the 10 queries of Q5 should be present in the Q5.sql file. Any code other than queries in Q5.sql should be in comments.
3. All different SQL statements in the .sql file need to be separated by **semicolons(;)**.
4. **Only create** table statements for Q3; **Only insert** statements for Q4; **Only select** statements for Q5.(Do not add statements like DROP TABLE IF EXISTS xxx).