CS482/502 Database Management Systems I

Project: Phase 2 (80 points)

The purpose of this project phase is to create an interface (web interface OR GUI - you can choose any that suits you) for the MySQL database that was created in Phase 1 to learn different types of data insertion techniques. You can use any programming language to finish this Phase. We will be using MySQL database for this phase as well.

Task 1:

- – The user should be able to insert a text file that contains the data for the corresponding table. E.g. If the name of the text file is *Players.txt*, then the data in the text file should be inserted into the *Players* table. The format of the textfile will be:
  Column_value1, *Column_value2*,..., , *Column_valueN*
  where a table consists of N columns. The text file will not contain any headers.
  - – You need to implement three different types of Insertion techniques:
    - ∗ **Single Insertion**: This is the most naive way to insert data. For each tuple in the file, you are supposed to call a single *Insert into* command.
    - ∗ **Multiple-row Insert Syntax**: You need to use the Multiple-row Insert Syntax in this option. Check `https://dev.mysql.com/doc/refman/8.0/en/optimizing-innodb-bulk-data-loading.html` for further details.
    - ∗ **Load Data Syntax**: In this option, you need to use the *Load Data Infile* syntax. Check `https://dev.mysql.com/doc/refman/8.0/en/optimizing-innodb-bulk-data-loading.html` for further details.
- For the *Players* table, create 3 different test datasets: dataset 1 should contain 10,000 tuples, dataset 2 should contain 100,000 tuples, and dataset 3 should contain 1,000,000 tuples.
- Create a report showing the performance of these 3 test datasets for the 3 different types of insertion techniques. Your report should include charts (and subsequent analysis) showing the performance of these 3 test datasets for the 3 different types of insertions.
- The *Position* column can have only one of the following values: *QB*, *RB*, *WR*. Ensure that this constraint is taken care of in your SQL commands.

Task 2: Your GUI should additionally contain functionality to perform the following operations:

- *Delete.* Given a table name, delete data from the table.
- *Retrieve all data.* Given a table name, retrieve all data from the table and present it to the user.
- *Average.* Given a table name and a column name, return the average of the column. Here the assumption is that the column domain will be integers.

We will be assuming that the input table names and column names already exist in the database (i.e. you do not need to worry about handling cases where the input table names and/or column names do not exist in the database).

Handing in & Grading:

- You have to submit your assignment electronically (through Canvas). Printed copies are not accepted. Combine and compress all source files to a .zip file with name "GroupNumber_Phase2.zip".
- Only one submission per group is needed.
- See course syllabus for policies on late submission and plagiarism.