CIS560 Class Project – Database Application

Project progress report - due November 22nd

In this assignment, you will first implement your database in MySql and then demonstrate its functionality with 10 queries, as detailed below.

Database implementation

This includes creating the tables in your database schema and populating each table with 5 to 10 records of realistic data. Additionally, you will need to support all necessary constraints and triggers to ensure referential integrity within your database.

More specific requirements:

1. Provide one SQL script file, calld table.sql, containing all of your CREATE TABLE statements. Make sure you use the InnoDB storage engine when you create your tables.
2. Remember to specify a PRIMARY KEY for each relation in your schema. If the relation has any keys in addition to the PRIMARY KEY, specify the additional keys as UNIQUE. If there are no additional keys, add a comment to that effect.
3. For each referential integrity constraint that should hold in your schema, specify the constraint using a REFERENCING clause within the appropriate CREATE TABLE statement. You may use the default option for handling referential integrity violations (violations will generate an error).
4. Write data modification commands to illustrate the following scenarios and submit a file viols.sql containing all five commands:
    a. An INSERT command creating a key violation
    b. An UPDATE command creating a key violation
    c. An INSERT command creating a referential integrity violation
    d. A DELETE command creating a referential integrity violation
    e. An UPDATE command creating a referential integrity violation
5. Create at least two "interesting" triggers for your database application. Submit a file trigs.sql containing the CREATE TRIGGER statements. Also submit documentation to show proper creation and functionality of each trigger.
6. Submit proof of the records in your tables by providing your SQL script file, called records.sql, used for uploading data into the database. In addition, you should also discuss how you will gather/generate additional data for your project.

Queries and reports

Write at least 10 queries (seven question-type queries and three report-type queries) that are "interesting," substantially-different, and structurally-different, i.e., operations that require some of the more complex SQL constructs such as subqueries, aggregates, set operators, etc.

Note: Ultimately, you will need to support your queries through your project interface, but SQL queries run from command line, phpmyadmin or some equivalent environment (depending on the specific DBMS you are using) should suffice for now.

Here are some guidelines and specific requirements:

- The purpose of the queries and reports is to demonstrate how your database might be used, and, therefore, should not be the output of a single table. Ask yourself what would be most useful for the target user of the database.
- Remember that, while queries are supposed to answer a particular question, the purpose of a report is to provide a summary of more of the content of the database. Sometimes the difference really comes down to quantity: while a query generates a single tuple or a small set of tuples, a report may generate dozens or more tuples. In general, a report is designed to be more human readable and to present information in tabular form. (See the original General Project Description document for some specific examples.)
- For the SQL queries demonstrated, ensure that there is at least one *join*, at least one *nested query*, at least one appropriate use of an *aggregate function* (such as min(), max(), avg(), sum(), etc.) within a select statement; at least one group-by, and at least one compound where condition (with at least a couple of sub-conditions that are different from *join* conditions). Note that these requirements ask for structurally different queries (these queries will be part of the 10 required queries).
- For the SQL reports demonstrated, at least two of your reports should be based on queries involving more than one table. You may employ concatenation to put together a full name or an address into a more readable form. You may also employ more than one query to construct a report. For instance, you may have a table with sales of a particular item or set of items, and then also compute the sales total.
- For now, you can run the queries using small sample tables; however, for your final presentation, you'll need to make sure that you have enough data for an interesting demonstration. At that time you will need to create at least 5 indexes (in addition to indexes created by default for primary keys) that will increase the performance for some of your queries.

Submit a script containing your 10+ queries and a screenshot of the results of each query. Make sure that your script is well commented and it clearly describes your queries and the rationale for each of them. Furthermore, the results should be clearly labeled, well-organized and easy to read.

Other requirements

1. Report other accomplishments on the project beyond database implementation, queries and reports. If you have no other accomplishments to report, make a note stating that.
2. Describe any problems you might have encountered in this phase of the project and how you have addressed such problems. If you didn't encounter any problems, make a note stating that.
3. Finally, write a short note about how your team has worked together on this assignment, and problems you have encountered, if any. Be specific about what student has worked on what part so far. If two or more students have worked together on a part, state that explicitly.

Note: Submit the required files and documentation using the course Dropbox (preferably all in one archive).