CMPSC 380
Principles of Database Systems
Fall 2014

Laboratory Assignment Five
Using the SQL Data Manipulation Language

Introduction

After creating and visualizing a relational database schema, it is important to define the data manipulation language (DML) statements that are needed to support interaction with the database. In this laboratory assignment, you will create a wide variety of DML commands—like the SQL `select`, `insert`, `update`, and `delete`—and executed them against a SQLite database.

Adding and Removing Relational Data

The SQL `insert` and `delete` commands allow you to add and remove relational data, respectively. To learn how these commands support the modification of a database, please review Sections 3.9.1 and 3.9.2 of your textbook. Then, you should return to one of the relational databases that you visualized for the previous laboratory assignment. Now, you can use the `sqlite3` command-line interactive shell to instantiate this schema as a SQLite database. Once you are sure that the schema was correctly created in the database, you should implement and execute at least four `insert` statements for each of the tables in your chosen schema.

As you are populating your database with data, please make sure that you take into account the integrity constraints defined for the schema. In addition to running the four `insert` statements that will conform to the integrity constraints, you should try to run at least one `insert` that will be rejected by SQLite because of the fact that it violates the database's constraints. Please recall from a previous classroom discussion that, depending upon the design of your schema, you may need to configure SQLite to fully enforce the integrity constraints. Now, you should run at least one `delete` statement and observe how it changes the state of the database. As you are running all of these `insert` and `delete` commands, you should carefully record the "before" and "after" state of the database and comment on why the state changed the way that it did.

Updating Data in a Relational Database

Section 3.9.3 explains the operation of the SQL `update` statement. For each of the tables in your chosen schema, you should run at least one `update` statement that will yield data that adheres to the integrity constraints. Again, as you run the `update`s, please take care to record the "before" and "after" state of the database, in addition to commenting on why the `update` changed the database in the way that it did. If possible, try to execute a wide variety of different `update` operations.

Querying a Relational Database

The SQL `select` statement allows you to query the database, thereby discovering what data values are stored inside of it. To learn more about the basic structure of SQL queries, please review Sections 3.3 and 4.1 of your textbook. Then, you should write at least ten `select` statements that interact with a wide variety of the tables in the relational database. Whenever possible, these queries should

also use various features of the SQL language, such as queries on multiple tables, aggregation, and joins. As you are running your queries, you should consider the pseudo-ER diagrams that you produced in the previous laboratory assignment to ensure that you understand how tables are connected. Please see the course instructor if you cannot run the `select` statements correctly.

The General Form of SQL Commands

After executing a wide variety of SQL commands, you should start to develop a better understand of their general forms. For instance, can you generalize all of the `select` statements that you ran? To learn more about the general form of SQL commands, please visit the "SQL As Understood by SQLite" page that is available at: `https://www.sqlite.org/lang.html`. As you study this page, you will notice that it contains "railroad diagrams" that are like flow charts for SQL commands. If these diagrams suggest that one of the data manipulation language statements has a feature that you have not yet tried, please go back to the `sqlite3` interactive shell and try it now.

Summary of the Required Deliverables

As you complete this laboratory assignment, you should record all of your notes, DML commands, and outputs in files stored in your Git version control repository. Additionally, this assignment invites you to submit one signed and printed version of the following deliverables:

1. A commentary on the general structure of `select`, `update`, `insert`, and `delete` statements.

2. Convincing evidence to demonstrate that your `create table` statements ran correctly.

3. A description of how you configured SQLite to run correctly for your schema.

4. The final listing of the `create table` statements for your database schema.

5. A complete listing of the `update`, `insert`, and `delete` statements that you ran.

6. A complete listing of the `select` statements that you used to query the database.

7. For each command, the before and after state for each modified table, highlighting all changes.

8. A reflection on the challenges that you faced when completing this laboratory assignment.

In adherence to the Honor Code, students should complete this assignment on an individual basis. While it is appropriate for students in this class to have high-level conversations about the assignment, it is necessary to distinguish carefully between the student who discusses the principles underlying a problem with others and the student who produces assignments that are identical to, or merely variations on, someone else's work. Deliverables that are nearly identical to the work of others will be taken as evidence of violating Allegheny College's Honor Code.