Project 1
Yelp: Intro to SQL

Out: *September 25th, 2019*
Due: *October 2nd, 2019, 11:59 P.M.*

1 Introduction

In this assignment, you will learn how to use Java Databse Connectivity (JDBC) to interact with a database through Java and you will learn how to write simple queries using SQLite. You will compose and execute 5 queries on a database of Yelp business reviews.

2 Overview of the Data

You will be using a small subset of Yelp's Academic Dataset, which provides data and reviews of restaurants and businesses around 30 U.S universities, including Brown University. The TAs have already parsed the data into a SQLite database, which can be found in `/course/cs1270/pub/yelp/yelp.db`.

For more information about Yelp's Academic Dataset, see `https://www.yelp.com/academic_dataset`.

2.1 Database Schema

Below is the database schema for the databases's three tables (`business`, `review` and `user`) including the datatype for each column/field. For more information about datatypes in SQLite, refer to `http://www.sqlite.org/datatype3.html`.

2.1.1 business

```
'id': a unique identifier for this business (VARCHAR),
'name': the full business name (VARCHAR),
'full_address': localized address (VARCHAR),
'city': city (VARCHAR),
'state': state (VARCHAR),
'latitude': latitude (REAL),
'longitude': longitude (REAL),
'stars': star rating, rounded to half-stars (REAL),
'review_count': review count (INT),
'open': is the business still open for business 1 if yes, 0 if not? (INT),
'photo_url': photo url (VARCHAR)
```

2.1.2 review

```
'business_id': the identifier of the reviewed business (VARCHAR),
'user_id': the identifier of the authoring user (VARCHAR),
'stars': star rating, integer 1-5 (INT),
'text': review text (TEXT),
'useful_votes': count of useful votes (INT),
'funny_votes': count of funny votes (INT),
'cool_votes': count of cool votes (INT)
```

2.1.3 user

```
'id': unique user identifier (VARCHAR),
'name': first name, last initial, like 'Matt J.' (VARCHAR),
'review_count': review count (INT),
'useful_votes': count of useful votes across all reviews (INT),
'funny_votes': count of funny votes across all reviews (INT),
'cool_votes': count of cool votes across all reviews (INT)
```

3 Tools

3.1 SQLite

SQLite is installed on all Sunlab machines. It can be accessed from the command line using `sqlite3`. To load a pre-existing database (for example, the Yelp database), simply run

```
sqlite3 /course/cs1270/pub/yelp/yelp.db
```

For more information on using SQLite from the command line, see `http://www.sqlite.org/cli.html`

3.2 SQLite Online

SQLite Online allows you to view the tables in a database and execute test queries. You can use it to test your queries as you are writing them.

To connect the database to the SQLite Online:

1. Go to `https://sqliteonline.com/`.
2. File → Open DB.
3. Navigate to `/course/cs1270/pub/yelp`, and open `yelp.db`.

3.3 JDBC

Java Database Connectivity is an application programming interface (API) that allows you to interact with a database in Java. "It provides methods to query and update data in a database, and is oriented towards relational databases" (Wikipedia). It is a Java-specific implementation of the ODBC (Open Database Connectivity) API which other languges use to connect with databases. The JDBC SQLite driver is included as a part of your stencil code.

Learn how to use the JDBC by coming to the Yelp recitation, or from `https://www.tutorialspoint.com/jdbc/index.htm`.

4 Working on the Project

4.1 Getting Started

1. Your `~/.bashrc` file is a shell script that runs every time you open the terminal. Open your `~/.bashrc` in a text editor (atom, vim, etc.) Add the following line to your `~/.bashrc`

   ```
   export JAVA_HOME=‘‘/pro/java/linux/jdk1.8.0_40’’
   ```

   and save your changes.

2. Run `source ~/.bashrc` (you should only need to run this once).

3. To get started with the Java stencil, copy `/course/cs1270/pub/yelp/stencil.tgz` into your course directory, and unpack it with `tar -xvzf stencil.tgz`.

4.2 Importing into Eclipse

1. Right click the stencil.tgz file and select "extract here". That should create a directory named "yelp" inside your course directory.
2. Open Eclipse. From the top menu bar, navigate to File → New Java Project.
3. From there, uncheck "Use default location."
4. Browse to the yelp directory inside your course directory. Click Finish.

5 The GUI Application

5.1 How to Run

To launch the GUI, either run App.java in Eclipse or run with the command `/usr/bin/ant run` in the project directory.

5.2 How to Use

On the left pane, the application displays business information. Only seven businesses are shown. Note that the business names are clickable and once clicked, it will open your default browser to load an actual business page on Yelp.

In the middle (unlabeled) pane, the application displays reviewer information for the selected business. Only seven reviewers are shown.The "User Avg" button gives the average rating for all reviews written by the user. The user names are clickable and open your default browser to an actual user page on Yelp.

Note that the GIU is not meant to represent a normal DB table, where each row entry is associated with its row. In the GUI, if you click on a business in the first column, users associated with that business populate the entire second column.

On the right pane, the application displays reviews for the selected business. Only seven reviews are shown.

In the bottom panel, the three buttons serve to provide different ranking algorithms for displaying the businesses on the left pane. Queries 2 and 3 are omitted, because they are activated by selecting a business.

5.3 Demo

Students are highly encouraged to check out the demo before starting the assignment. The demo can be found in `/course/cs1270/pub/yelp/demo/demo.jar` and it can be run with the command `java -jar /course/cs1270/pub/yelp/demo/demo.jar`. You will compare the results of the demo to your own to check your accuracy.

6 Your Assignment

Your task is to set up a connection to the database and write five queries to answer the following questions. All queries should be composed of a single SQL statement. All of your Java code should be written in DBStudentController.java, a stencil file provided for you.

For each query method, you will have to do two things. First, you will have to write an appropriate SQL query and execute it to return a ResultSet. **You must use PreparedStatement to execute the queries.** PreparedStatement is an efficient way to execute queries, especially for the ones that require inputs. Second, you will need to extract the data from the ResultSet, load them into their appropriate data type (see the other classes for these data structures), and return them.

6.1 Queries

1. Return the businesses in Providence, RI that are open. Results should be sorted by review counts in descending order. Return top 7 businesses.

   Input: N/A

   Output: Six columns - the business id, name, full address, review count, photo url, and stars of the business.

2. Get the reviews for a particular business, given the business ID. Results should be sorted by the review's useful vote counts in descending order. Return top 7 reviews.

   Input: Business ID

   Output: Four columns - the user id, name of the user, stars of the review, and text of the review.

3. Find the average star rating across all reviews written by a particular user.

   Input: User ID

   Output: One column - the average star rating.

4. This problem consists of two parts. Part (a) is optional, but completing part (a) will keep you on the right track, and can potentially earn you partial credit.

   (a) First, write a sub-query using the WITH clause to find the 'elite users,' i.e. users who have written more than 10 reviews.

   (b) Now, in the main query, get the businesses in Providence, RI that have been reviewed by more than 5 'elite' users. Results should be ordered by the 'elite user' count in descending order. Return top 7 businesses.

   Input: N/A

   Output: Seven columns - the business id, business name, business full address, review count, photo url, stars, and the count of the 'elite' users for the particular business.

5. Get the businesses in Providence, RI that have the highest percentage of five star reviews, and have been reviewed at least 20 times. A WITH clause may be helpful here as well. (Note: You will need to cast the count of five star reviews as a FLOAT. What SQL clause might you use for this?) Results should be ordered by the percentage in descending order. Return top 7 businesses.

   Input: N/A

   Output: Seven columns - the business id, business name, business full address, review count, photo url, stars, and percentage of five star reviews

6.2 Hints

To return top n rows in SQLite, you can use the `LIMIT` clause (`http://www.sqlite.org/syntaxdiagrams.html#select-stmt`). For example, `SELECT * FROM business LIMIT 10;` will return the top 10 rows in the `business` table.

The WITH clause helps you to break down a complex query into linearly ordered (rather than nested) sub-queries. This makes your SQL code easier to write and read. (Another advantage, though not needed for Yelp, is that sub-queries defined by the WITH clause can be used multiple times in the query. This will become very handy in the ETL project.) A short tutorial on how to use the "WITH" clause can be found here: `https://www.geeksforgeeks.org/sql-with-clause/`

7 Handin

You can handin your project by running the following command from the directory containing all your files:

```
/course/cs1270/bin/cs127_handin yelp
```