CS 564, Fall 2019: Assignment 2 - SQL

Due Date: September 30, 2019 by 11:59PM
Project Grade Weight: 10% of the total grade

Introduction

In this assignment, you will use SQLite3 to perform various queries on a real database. You can find documentation on SQLite3 at http://www.sqlite.org/. SQLite3 is not as functional as PostgreSQL, MySQL or commercial relational DBMSs, but it is much easier to use and program. That is why there are many more installations of SQLite than PostgreSQL or any other traditional database management system (DBMS). Also, you can easily install SQLite3 on your own machine, but make sure your final code runs on SQLite3 installed on the CS **snares-XX.cs.wisc.edu** machines.

For this assignment, you will use the TPC-H benchmark database. This particular database is used to test and compare database engine speed and performance on a standardized dataset with a set of provided queries. Because of this, the actual data that is stored in the TPC-H tables may not be very coherent (e.g. the "comments" columns in the tables have random words, not complete sentences). A specified "scaling factor" parameter used in the creation of the data allows us to make TPC-H databases of all sizes, but benchmark results have been provided for TPC-H databases as large as 10 TB! The one you'll be working with in this assignment is considerably smaller-- only about 100 MB.

Most substantial datasets are published with a bulky document describing the data. Here is what that looks like for this dataset: http://www.tpc.org/tpc_documents_current_versions/pdf/tpc-h_v2.18.0.pdf. Starting on page 13 of this document, you will find the schema description for this dataset.

We have provided you the TPC-H database file, available here. The database, tables, and indexes have all been created for you, so all you're required to do is query the database. More specifically, your job is to write the following queries in separate files, as described below:

File name(s)	Query
`Query1.txt`	Produce a list of the 20 largest orders in the database, where size is defined as $\Sigma(quantity * (extended\ price - discount))$. Order the output by decreasing order size. ~~Please **DO NOT** use a "limit" clause to select the first 20 rows. Instead, think of other ways to filter out the bottom rows.~~ Include the customer name, order number, and order size in the output table.
`Query2.txt` AND `Query2_nest.txt`	Produce an alphabetical list of all nations in the database, along with the total order volume within each nation (supplier nation equal customer nation). The total order volume can be defined as the sum of order sizes. Write two queries, one using a nested query and one without a nested query. In each output table include the country name and the order volume.
`Query3.txt`	Produce a list of all nations in the database. The output table should include the name of the seller nation, each nation's total order volume from nations within their own region, and the total order volume from nations in a different region. Sort the nations by their total order volume within their own region in descending order.

Query4.txt	Find a list of all suppliers with orders from more than 615 different customers. Include the supplier name and customer count in the output table and order the rows by decreasing customer count.

For grading purposes, please place each column in the order they were specified in the queries. For example: query 1 should output a table where the first column is the customer name, the second is the order number, and the third is the order size. In addition, **include your student ID number as a comment somewhere within your SQL code in each query file.**

Note: To check the query output, we will run:

```
sqlite3 TPC-H.db < Query1.txt
```

Your text file must only output your final desired output. All of these queries should be able to be written in a single SQL query. However, you are permitted to use a "with" clause in your query when it's convenient to do so.

Copy your query results to a file called `results.txt`.

Submission Instructions

Only the following files are required for this assignment: `Query1.txt`, `Query2.txt`, `Query2_nest.txt`, `Query3.txt`, `Query4.txt`, `results.txt`. For your more complicated queries, please use comments in your SQL files. When you're finished, please follow these instructions to submit the project:

1) Place all six required files in a directory.

2) Name this directory using the format: `<netID>_P2` (`e.g.klklassy_P2`).

2) Run:

```
tar -czvf <netID>_P2.tar.gz /path-to-project/<netID>_P2
```

3) Submit the tar file.

4) To check, you can uncompress the tar file. (run: `tar –xzvf <netID>_P2.tar.gz`).

I know that some students were having troubles in the last project with nested folders being included in the tar file. If you're having this problem, you can use the following:

1) `cd /path/to/project/directory`
2) `mkdir <netID>_P2`
3) `mv *.txt <netID>_P2`
4) `tar -czvf <netID>_P2.tar.gz <netID>_P2`

If you do not adhere to this standard, 5 points will be deducted. To hand in your work, please go to the Canvas: Assignment 2 SQL page to upload your files. Your files must be uploaded by the deadline stated on the first page.