Using psql for Question 2 of Tutorial 4

This handout describes how to attempt Question 2 of Tutorial 4 using `psql`. Your submission for this tutorial will not be graded as this is just for your own practice and to get you familiar with the submission process.

1 How to set up

1. Login to `sunfire` server.
2. Download the file http://www.comp.nus.edu.sg/∼cs2102/cs2102-tut4.zip as follows.

   ```
   $ cd ~
   $ wget http://www.comp.nus.edu.sg/~cs2102/cs2102-tut4.zip
   $ unzip cs2102-tut4.zip
   ```

   The unzipped directory cs2102-tut4/ contains the following files.

   - setup0.sh, setup1.sh - bash scripts to update environment variables
   - check.sh - bash script to display the values of your `PostgreSQL` environment variables.
   - tut4.sql - template SQL script for you to fill in your answers for Question 2.
   - data/ - directory containing 5 CSV files for a database instance.
   - loaddata.sql - SQL script to create database schema and load database instance.
   - solution-files/ - directory containing the solution output files for the provided database instance.
   - test.sql - bash script to compare the outputs of your SQL answers against the provided solution files.

3. Execute setup0.sh if your assigned `PostgreSQL` server is psql0; otherwise, execute setup1.sh.

   ```
   $ cd ~/cs2102-tut4
   $ bash setupX.sh
   $ source ~/.bash_profile
   $ echo $PATH
   ```

   You should see the directory names `/home/course/cs2102/bin` and `/usr/local/postgres/10-pgdg/bin/64` included in the value of `PATH`. Note that you do not need to repeat this step for subsequent logins.

If you have not already configured your ~/.pgpass file, a message will be displayed explaining how to configure this file.

4. Check your PostgreSQL environment variable configurations by executing the check.sh script. The following illustrates the output for a user with PostgreSQL account “alice” assigned to PostgreSQL server “psql0”.

   ```
   $ cd ~/cs2102-tut4
   $ bash check.sh
   PGUSER=alice
   PGHOST=psql0
   PGDATABASE=cs2102
   PGSQL_EDITOR=/usr/local/bin/vim
   ```

   If the values shown are not the expected values, edit the file ~/.bash_profile with the necessary changes.

5. Create the database schema and load the provided database instance as follows.

   ```
   $ psql < loaddata.sql
   ```

2 How to prepare & check your answers

Edit the provided SQL script tut4.sql to fill in your answers for Question 2. Your answer to each SQL query will be specified as a view.

For each part, say part X, a view has been defined named qX in tut4.sql with a dummy definition consisting of a single line (e.g., SELECT 1) that has the comment “replace this line”. To input your answer for part X, simply search for the view definition qX in tut4.sql, and replace the single line commented with “replaced this line” with your SQL answer.

You can compare the outputs of your SQL answers against the provided solutions in solution-files/ as follows.

```
$ bash test.sh
```

The test.sh script will first execute your tut4.sql script to create all your view definitions. Next, for each view definition qX, it will generate the output of your view qX in a file named your-solution/qX.txt and compare it against the corresponding solution file named solution-files/qX.txt using the diff command. The diff command will create an output file named diff-dir/qX.txt which will be empty if your solution file matches the provided solution file; otherwise, the output file will contain the differences between the two solution files. Refer to https://en.wikipedia.org/wiki/Diff_utility for an overview of how to interpret the output created by the diff command.

Edit tut4.sql and run test.sh again if necessary to fix any errors.

3 How to submit

Submit your SQL script `tut4.sql` as follows.

```
$ submit-tut4
You submitted tut4.sql (size: 1622 bytes, #lines: 51) on Friday, February 9, 2018 12:31:13 PM SGT
```

You may make multiple submissions using the `submit-tut4` command and only the latest submission will be recorded.