CS 284, Interim 2018, R. Brown

Homework 4 Due Friday, 1-19-18

XX5

A. SQL queries

- 1. For your book/patron database tables created in the previous database homework, write SQL queries using SELECT to determine the following.
 - a) Find all information about books by Tanenbaum.
 - b) Find the titles and authors of all books.
 - c) Find the call numbers of all novels.
 - d) Find the names of all patrons who have non-null phone numbers. Note. Instead of the = operator for this comparison (phone = null), the SQL is operator must be used for comparisons involving null, e.g., phone is null. See section 9.2 of Postgres documentation for more information.
 - e) Find the names and phone numbers of all patrons whose key values are 2 or more.
 - f) Find all information about patrons whose names start with S. (The Postgres documentation, Section 9.7, provides two approaches for matching patterns LIKE/SIMILAR TO and posix regular expressions (Posix is a standard for UNIX systems). You may use either approach.)

To deliver this work, create a file HW4A.sql that contains your queries, then use it to create a file HW4A.out that contains those queries and their results, and submit the latter file partname.out via git with the remainder of your homework for this assignment. See the recommendations below for creating HW4A.out.

Note: You can check a file of Postgres commands (such as HW4A.sql) against your database using the locally written command dopsql. First, make sure that you have blank lines between your queries or other Postgres commands (the program uses blank lines to determine where queries begin and end). Then, enter the following Linux command.

% ~cs284/.bin/dopsql filename > outfile

Here, filename is the name of your SQL file, e.g., HW4A.sql, and outfile is the name of a file for storing the output (use HW4A.out for this part). You can see the output on the screen by leaving off > outfile.

Note: Be sure to start your SQL file *filename* with a command

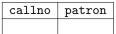
set search_path to username;

where *username* is your schema name.

B. SQL Joins

See recommendations concerning dopsql above for delivering this work.

- 1. Create a third table described below.
 - a) Table checkouts represents books checked out to patrons, and its fields are described by the following diagram.



The call number callno should be a key for book. The field patron should be a key pid for patron. (Be sure to include foreign key constraints.)

- b) Add rows to checkouts to express that Sue checked out Modern Operating Systems, Sam checked out War and Peace, and Chris checked out Structured Computer Organization.
- 2. Now write SQL join queries using SELECT to determine the following. Write two queries for each join task, one using a WHERE clause to specify the join condition and one using a JOIN clause. See text/sql.html#joins for examples.
 - a) Find the number of books checked out. (Use the aggregate function COUNT(*), which returns the number of rows matching SELECT criteria. A join might not be necessary for this query.)
 - b) Find the names of patrons who have checked out a book.
 - c) Find the patron names and book authors for all checkouts.
 - d) Find the names of patrons who have checked out a book with call number beginning with TAN.
 - e) Find the titles of books checked out by patrons who have no known phone number.
 - f) Find the call numbers for all books that are checked out for which either the book title or the patron name contains a letter u.

Create a file HW4B.out using dopsql with a file of your sql commands as before, and submit HW4B.out via git.

C. SQL queries

1. The sailor tables are defined in mca/pub/sailors.sql and have been entered in the public schema of the database mca_i18. Write SQL queries for the following searches listed in the page mca/queryegs.html, then test your queries using psql.

Query numbers: 1, 2, 3, 5, 6, 22, 25, 26, 27, 30

Notes:

• You may need to enter the SQL command

set search_path to public;

in order to access these tables.

• Submit this work by entering your queries in a file HW4C.sql, then producing HW4C.out via dopsql, and finally submitting HW4C.out via git.