Carnegie Mellon University in Qatar
15415 - Spring 2020

Recitation 4

1 More on PostgreSQL...

1. Log in to your assigned VM
2. Connect to PostgreSQL, create an empty database and connect to it:

   ```
   sudo -u postgres psql
   CREATE DATABASE Recitation4;
   \connect Recitation4;
   ```

3. Under the database, create four tables; namely: *Student*, *Faculty*, *Class*, and *Enrolled*. You can either enter the `SQL CREATE` statements via the command-line or put them all in a file (aka a SQL script). For convenience, you can download the SQL script from the course website and import it:

   ```
   \i '<SQL script path>';
   ```

4. To check that the tables were created, we run the following command: `\dt;`
5. Populate the tables by inserting tuples. Again, you may enter your SQL INSERT statements via the command-line. However, we shall show you how to import existing data into the tables. In the tgz that you got from the website, there are four CSV (Comma Separated Values) files; namely: `student.csv`, `faculty.csv`, `class.csv`, and `enrolled.csv`, each of which should be imported into the respective table. To do so, use the copy command as exemplified below:

   ```
   copy <table_name> from '<filename>' with CSV # EXAMPLE

   copy student from 'C:/Users/ZFK/Recitation4/Student.csv' with CSV;
   ```

6. Write your first query!

   ```
   SELECT * from student;
   ```

7. Close the connection to the database:

   ```
   \q;
   ```

2 Writing SQL Queries

Consider the following relation schemas:

Student (sid: integer, sname: string, major: string
standing: string, age: integer

Class (name: string, meets_at: string, room: string, fid: integer)

Faculty (fid: integer, fname: string, deptid: integer)

Enrolled (sid: integer, cname: string)

The meaning of these relations is straightforward. For example, Enrolled records student-class pairs such that the student is *Enrolled* in the class. A student's standing refers to the student's year, and can take on the values FR (Freshman), SO (Sophomore), JR (Junior), and SR (Senior).

Write SQL queries to achieve the following requirements. Note that no duplicates should be produced in any of the answers.

1. Find all Juniors who are enrolled in a class taught by any faculty whose surname begins with the letter **T**. Print the students and faculty names.
2. For all the standings except JR, print the standing and the average age of students in that group.
3. Find the names of all students who have a conflict; i.e. they are enrolled in two classes that meet at the same time.