CS 775/875 hw3 2/22/2021 Due date: 3/1/2021

Instructions:

1. Electronic submission: You assignment is **due by 11:00 PM, 3/1**.

   Submit instructions: From your agate home directory:

   (a) create a directory called *hw3*. Write all your queries in *hw3*.

   (b) copy or create the *culinary.db* file to directory *hw3*.

   (c) Each query should be in a separate file named **q***i*, where $i = 1, 2, ..., 6$. You should have **q1, q2, ..., q6** corresponding to each query, by order.

   (d) From directory *hw3*, submit your queries using the command:
   $\sim cs775/submit\ 2\ q*$

   (e) 3 is the assignment number. Note that if you want to resubmit, then you need to use 3a, 3b, 3c,...... for assignment number (not 2).

   (f) We have had submission problems in the past. In order to ensure that you get credit for you work, make a tar file of your final submission using the command
   *tar -zcpvf hw3.tar q**
   Do not touch *hw3.tar* until you get back your graded assignment. The tar file keeps a dated copy of submitted files in your directory.

2. Your answers must be written in the format specified by **RA: A Relational Algebra Interpreter**, developed by Professor Yang of Duke University. Information about the RA tool is available on Prof. Yang's website: https://users.cs.duke.edu/∼junyang/radb/

3. The TA will be grading your assignment by using the following command:

   radb culinary.db -i q*i* -o outq*i*

   for each query.

   Please note that the TA may use a different instance of the database while grading.

4. Late policy: 1 day late: 2 points off, 2 days late: 4 points off; $>$ 2 days late: will not be graded.

5. The relevant reading material is from Chapter 8 and class notes.

Queries:
To test some of the queries, you may have to add or delete data to/from the database.

1. (6 points) **q1**: Retrieve the descriptions of the highest level for each course. (Example: If there are 4 levels for a course, I want the description of the course at Level 4.)

   Result has schema (school, course, level, description).

2. (6 points) **q2**: Retrieve staff members who never participate in a course whose description contains the text 'Pasta' either as students or on duty.

   Result has schema (ssn, name)

3. (6 points) **q3**: Retrieve student pairs where the first student takes at least one course the second takes.

   Result has schema (name1, name2).

   Do not print the same pair twice. For example, do not print (joe, mary) since (mary, joe) appears in the output.

   ```
   (name1:string, name2:string)
   ----------------------------------------------------------------------
   mary, joe
   steve, joe
   steve, mary
   tommy, steve
   ----------------------------------------------------------------------
   4 tuples returned
   ```

4. (6 points) **q4**: List courses that are held in only one location.

   Result has schema: (code, classdate, location)

5. (6 points) **q5**: List locations that have only held one course (on one or more dates).

   Result has schema: (code, location)

   For the sample database, the output is

   ```
   (code:string, location:string, classdate:date)
   ----------------------------------------------------------------------
   cs1, Boston, 2016-01-01
   cs2, New York, 2016-01-02
   cs2, Toronto, 2016-01-11
   cs3, Chicago, 2016-01-03
   cs4, Seattle, 2016-01-04
   cs4, Hamburg, 2016-01-13
   cs5, San Diego, 2016-01-05
   cs5, Geneva, 2016-01-14
   fc1, Dijon, 2015-11-19
   hb1, Houston, 2016-01-17
   jp1, Paris, 2016-01-06
   jp1, London, 2016-01-15
   jp2, Nantes, 2016-01-07
   jp2, Dublin, 2016-01-16
   ```

```
jp2, Dublin, 2016-02-16
ln1, Naples, 2016-01-09
ln1, Rome, 2016-01-18
ln1, Naples, 2016-02-09
----------------------------------------------------------------------
18 tuples returned
```

Multiple courses have been held in Berlin, so Berlin is not listed.