Homework 7: Databases

In this homework, you will be populating a database table with information from Yelp (we have provided the cache data in yelp_data.txt) and write code to fetch data from the table.

We have provided the code for the following:

1. To read the cache data (**readDataFromFile()** function)
2. To create the database and set up the connection and cursor (**setUpDatabase()** function)
3. To set up one of the tables, called Categories, in the database (**setUpCategoriesTable()** function): Run the starter code and then check the structure of the Categories table in the DB Browser.

When done with the assignment, your database will have two tables, including the one we have provided and the one that you will write code to create and fill.

We have also provided test cases that will pass if the functions are written correctly. You may not edit the test cases in any way.

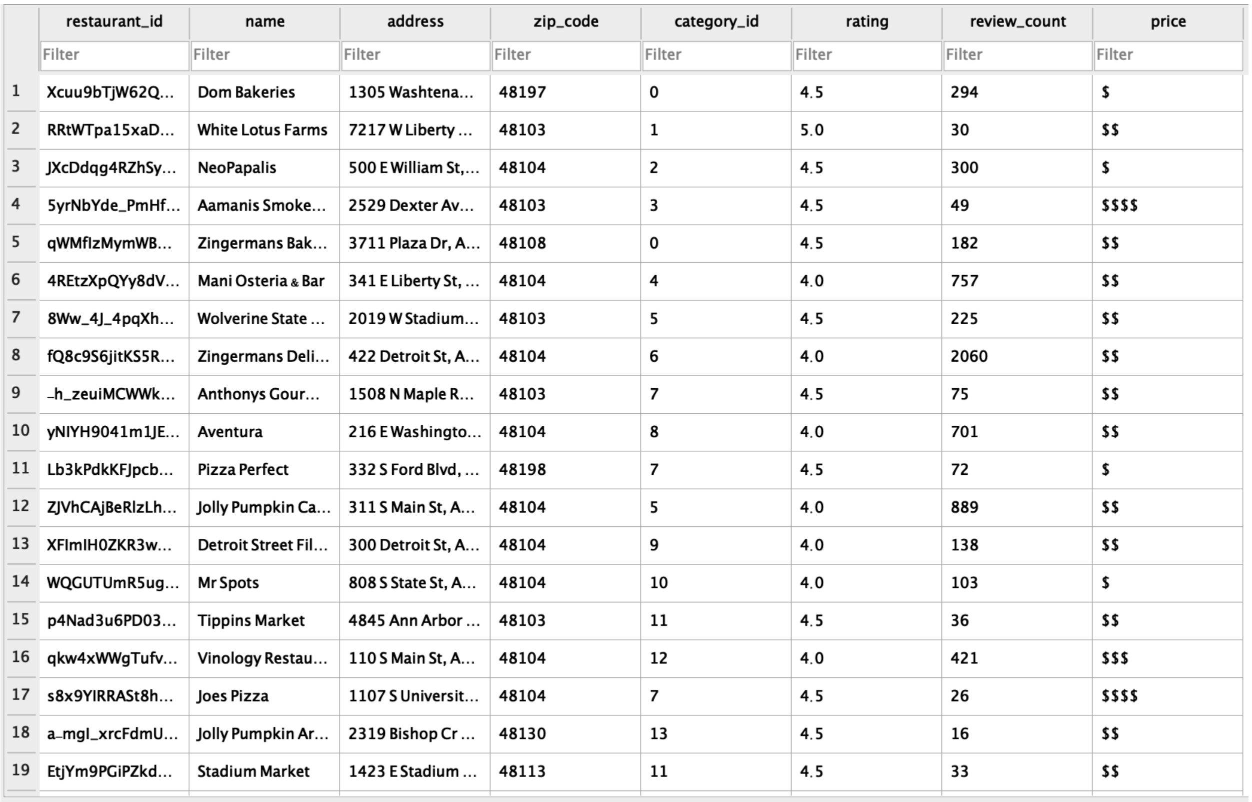
NOTE: It is okay for the extra credit test case to fail if you do not attempt the extra credit (**test_restaurants_of_type**)

Tasks

1. **setUpRestaurantTable() function:** The function takes three arguments as input: the JSON object, the database cursor, and database connection object. It loads all of the businesses in the JSON object into a table called Restaurants. The function does not return anything.
   The table should have the following columns:
   a. restaurant_id (datatype: text; primary key)
   b. name (datatype: text)
   c. address (datatype: text)
   d. zip_code (datatype: text)
   e. category_id (datatype: integer)
   f. rating (datatype: real)
   g. review_count(datatype: integer)
   h. price (datatype: text)

NOTE FOR PRICE: Some entries do not have a price value in their JSON file. If a restaurant does not have a price value, then the value for its price in the table should be entered as “$$$$”.

Expected Table in DB Browser:



	restaurant_id	name	address	zip_code	category_id	rating	review_count	price
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	Xcuu9bTjW62Q…	Dom Bakeries	1305 Washtena…	48197	0	4.5	294	$
2	RRtWTpa15xaD…	White Lotus Farms	7217 W Liberty …	48103	1	5.0	30	$$
3	JXcDdqg4RZhSy…	NeoPapalis	500 E William St,…	48104	2	4.5	300	$
4	5yrNbYde_PmHf…	Aamanis Smoke…	2529 Dexter Av…	48103	3	4.5	49	$$$$
5	qWMfIzMymWB…	Zingermans Bak…	3711 Plaza Dr, A…	48108	0	4.5	182	$$
6	4REtzXpQYy8dV…	Mani Osteria & Bar	341 E Liberty St, …	48104	4	4.0	757	$$
7	8Ww_4J_4pqXh…	Wolverine State …	2019 W Stadium…	48103	5	4.5	225	$$
8	fQ8c9S6jitKS5R…	Zingermans Deli…	422 Detroit St, A…	48104	6	4.0	2060	$$
9	_h_zeuiMCWWk…	Anthonys Gour…	1508 N Maple R…	48103	7	4.5	75	$$
10	yNIYH9041m1JE…	Aventura	216 E Washingto…	48104	8	4.0	701	$$
11	Lb3kPdkKFJpcb…	Pizza Perfect	332 S Ford Blvd, …	48198	7	4.5	72	$
12	ZJVhCAjBeRlzLh…	Jolly Pumpkin Ca…	311 S Main St, A…	48104	5	4.0	889	$$
13	XFImIH0ZKR3w…	Detroit Street Fil…	300 Detroit St, A…	48104	9	4.0	138	$$
14	WQGUTUmR5ug…	Mr Spots	808 S State St, A…	48104	10	4.0	103	$
15	p4Nad3u6PD03…	Tippins Market	4845 Ann Arbor …	48103	11	4.5	36	$$
16	qkw4xWWgTufv…	Vinology Restau…	110 S Main St, A…	48104	12	4.0	421	$$$
17	s8x9YIRRASt8h…	Joes Pizza	1107 S Universit…	48104	7	4.5	26	$$$$
18	a_mgl_xrcFdmU…	Jolly Pumpkin Ar…	2319 Bishop Cr …	48130	13	4.5	16	$$
19	EtjYm9PGiPZkd…	Stadium Market	1423 E Stadium …	48113	11	4.5	33	$$

(To find the category_id for each restaurant, you will have to look up the category of each restaurant in the Categories table we create for you. See **setUpCategoriesTable** for details)

2. **getRestaurantsByZip() function:** The function takes three arguments as input: a zip code, the database cursor, and database connection object. It selects all the restaurants for a particular zip code and returns a list of tuples. Each tuple contains the restaurant name, restaurant address and restaurant zip code.

   Expected output for restaurants of price “`48197`”:

   ```
   [('Dom Bakeries', '1305 Washtenaw Rd, Ypsilanti', '48197'),
   ('Little Caesars Pizza', '1783 Washtenaw Rd, Ypsilanti', '48197'),
   ('Mr Pizza', '1484 Washtenaw Rd, Ypsilanti', '48197')]
   ```

3. **getRestaurantsByZipcodeAboveRatingAndByPrice() function:** The function takes five arguments as input: the zip code, the rating, the price, the database cursor, and database connection object. It selects all the restaurants at a certain zip code, rating bigger than or equal to the rating passed to the function, with a particular price, and returns a list of tuples. Each tuple in the list contains the restaurant name, restaurant address, restaurant rating, and restaurant price.

   Expected Output for restaurants with zip code “48104”, rating >= 4.0 and price = ‘$’:

```
[('NeoPapalis', '500 E William St, Ann Arbor', 4.5, '$'),
('Mr Spots', '808 S State St, Ann Arbor', 4.0, '$'),
('Stadium Deli & Wine Shop', '1956 S Industrial Hwy, Ann Arbor', 4.5,
'$')]
```

4. **getRestaurantsAboveRatingAboveReviewsOfCategory() function:** The function takes five arguments as input: a rating, a review count, a category, the database cursor, and database connection object. It returns a list of tuples for all of the restaurants that match that category and have a rating greater than or equal to the rating passed to the function, and a review count greater than or equal to the review count passed to the function. Each tuple in the list should contain the restaurant name, restaurant address, restaurant rating, and restaurant review count.
   Note: You have to use a JOIN for this task.

   Expected Output for rating >= 4.5, review_count >= 100, and category “Bakeries”:

```
    [('Dom Bakeries', '1305 Washtenaw Rd, Ypsilanti', 4.5, 294),
    ("Zingerman's Bakehouse", '3711 Plaza Dr, Ann Arbor', 4.5,
182)]
```

Grading Rubric

1. **setUpRestaurantTable()** - 25 points
   a. 10 points for entering all 50 restaurants in the table
   b. 5 points for creating all 8 columns in the table
   c. 10 points for using the correct type for each column

2. **getRestaurantsByZip()** - 10 points
   a. 5 points for returning the correct number of restaurants by zip code
   b. 5 points for returning the three columns: restaurant name, address, and zip code

3. **getRestaurantsByZipcodeAboveRatingAndByPrice()** - 10 points

    a. 5 points for returning a list of tuples of restaurants at the passed zip code and above or equal to a rating and at price.
    b. 5 points for returning all four columns: restaurant name, address, rating, and price.

4. **getRestaurantsAboveRatingAboveReviewsOfCategory()** - 15 points
    a. 10 points for correctly using a JOIN to get the rows
    b. 5 points for correctly outputting a list of tuples with restaurant name, address, rating, and review count.

Git Commits
Make at least 3 git commits before the deadline. Each commit is worth 5 points. Please upload a link to your GitHub repository URL to Canvas.

Extra Credit - 6 points

getRestaurantsOfType() -- this function takes in 5 parameters: price, rating, category, the database cursor, and database connection object. It returns a list of all of the restaurant names that match the price, are greater than or equal to that rating, and match that category.

Expected output when searching for restaurants of price “$$”, rating 4.0 or above, and category “Pizza” [Name appears twice since there are two entries for Anthony’s Gourmet Pizza in the database]:

```
[("Anthony's Gourmet Pizza",),
("Anthony's Gourmet Pizza",),
('Red Rooster Pizzeria',)]
```

Write more tests for **getRestaurantsOfType()** in **test_restaurants_of_type_extra_credit**.
Write at least 3 assert statements.