

# Homework 8: 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 writing 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. `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	category_id	rating	price
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	RRtWTpa15xa...	White Lotus F...	7217 W Libert...	48103	1	5.0	\$\$
2	Xcuu9bTjW62...	Dom Bakeries	1305 Washte...	48197	0	4.5	\$
3	JXcDdqg4RZh...	NeoPapalis	500 E William...	48104	2	4.5	\$
4	5yrNbYde_Pm...	Aamani's Smo...	2529 Dexter ...	48103	3	4.5	\$\$\$\$
5	qWMfIzMymW...	Zingerman's B...	3711 Plaza Dr...	48108	0	4.5	\$\$
6	8Ww_4J_4pq...	Wolverine Sta...	2019 W Stadi...	48103	5	4.5	\$\$
7	-h_zeuIMCW...	Anthony's Go...	1508 N Maple...	48103	7	4.5	\$\$
8	Lb3kPdkKFJpc...	Pizza Perfect	332 S Ford Bl...	48198	7	4.5	\$
9	p4Nad3u6PD0...	Tippins Market	4845 Ann Arb...	48103	11	4.5	\$\$
10	s8x9YIRRASt8...	Joe's Pizza	1107 S Univer...	48104	7	4.5	\$\$\$\$
11	a-mgI_xrcFd...	Jolly Pumpkin...	2319 Bishop ...	48130	13	4.5	\$\$
12	EtjYm9PGiPZk...	Stadium Market	1423 E Stadiu...	48113	11	4.5	\$\$
13	hZwum9mA8f...	Wings N Things	3220 Broad St...	48130	16	4.5	\$\$\$\$
14	tXR0D1Fa-cR...	Little Caesars...	1783 Washte...	48197	7	4.5	\$
15	Wiry4DuTOnt...	Stadium Deli ...	1956 S Indust...	48104	6	4.5	\$
16	4REtzXpQYy8...	Mani Osteria ...	341 E Liberty ...	48104	4	4.0	\$\$
17	fQ8c9S6jitKS5...	Zingerman's ...	422 Detroit St...	48104	6	4.0	\$\$
18	yNIYH9041m...	Aventura	216 E Washin...	48104	8	4.0	\$\$

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

Expected output for restaurants of price “\$\$\$\$”:

```
("Aamani's Smokehouse & Pizza", '2529 Dexter Ave, Ann Arbor'),
("Joe's Pizza", '1107 S University Ave, Ann Arbor'),
('Wings N Things', '3220 Broad St, Dexter')
```

3. **getRestaurantsByZipcodeBelowRatingAndByPrice() function:** The function takes five arguments as input: the zip code value, the rating value, the price, the database cursor, and database connection object. It selects all the restaurants with a certain zip code, rating less 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, address, rating, and price.

Expected Output for restaurants with zip code "48104", rating  $\leq 4.0$  and price = '\$':

```
('Mr Spots', '808 S State St, Ann Arbor', 4.0, '$'),  
( 'South U Pizza', '1106 S University Ave, Ann Arbor', 3.5, '$'),  
( 'New York Pizza Depot', '605 E William St, Ann Arbor', 3.5, '$'),  
( 'Pizza Bob's', '814 S State St, Ann Arbor', 3.5, '$'),  
( "Blaze Fast Fire'd Pizza", '3500 Washtenaw Ave, Ann Arbor', 3.5, '$'),  
( 'Backroom Pizza', '605 Church St, Ann Arbor', 3.5, '$')
```

4. **getRestaurantsAboveRatingOfCategory() function:** The function takes four arguments as input: a rating, a category, the database cursor, and database connection object. It returns a list of tuples for all of the restaurant names that match that category and have a rating greater than or equal to the rating passed to the function. Each tuple in the list should contain the restaurant name, address and rating.

Note: You have to use JOIN for this task.

Expected Output for rating  $\geq 3.0$  and category "Bakeries":

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

## Grading Rubric

1. setUpRestaurantTable() - 25 points
  - a. 10 points for entering all 50 restaurants in the table
  - b. 5 points for creating all 7 columns in the table
  - c. 10 points for using the correct type for each column
2. getRestaurantsByPrice() - 10 points
  - a. 5 points for returning the correct number of restaurants by zip code
  - b. 5 points for returning the two columns: restaurant name and address
3. getRestaurantsByZipcodeBelowRatingAndByPrice() - 10 points
  - a. 5 points for returning a list of tuples of restaurants by zip code and above or equal to a rating.
  - b. 5 points for returning all three columns: restaurant name, address, rating.
4. getRestaurantsAboveRatingOfCategory() - 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 and rating.

### 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',)
```