COEN 280 - Database Systems
Fall 2021
Homework Assignment 3
Due: Tuesday, Nov 23 @11:59pm
Demo: TBD

In your course project you would develop a data analysis application for Yelp.com's business review data. The emphasis would be on the database infrastructure of the application.

In 2013, Yelp.com has announced the "Yelp Dataset Challenge" and invited students to use this data in an innovative way and break ground in research. In this project you would query this dataset to extract useful information for local businesses and individual users.

The Yelp data is available in JSON format. The original Yelp dataset includes 42,153 businesses, 252,898 users, and 1,125,458 reviews from Phoenix (AZ), Las Vegas (NV), Madison (WI) in United States and Waterloo (ON) and Edinburgh (ON) in Canada. (http://www.yelp.com/dataset_challenge/). In your project you will use a smaller and simplified dataset. This simplified dataset includes only **20,544** businesses, the reviews that are written for those businesses only, and the users that wrote those reviews.

The Yelp JSON files that you will use in this project are available on Camino.
(Note: Please make sure to use the dataset available on Camin, not the one from the Yelp.com website)

See Appendix-A for an overview of the Yelp Academic Dataset.

Overview & Requirements:

You would develop a target application which runs queries on the Yelp data and extracts useful information. The primary users for this application will be potential customers seeking for businesses and users that match their search criteria. Your application will have a user interface that provides the user the available business categories (main, sub-categories), the attributes associated with each business category along with business review and yelp user information associated with each business category. Using this application the user will search for the businesses from various business categories that have the properties (attributes) the user is looking for.

Faceted search has become a popular technique in commercial search applications, particularly for online retailers and libraries. It is a technique for accessing information organized according to a faceted classification system, allowing users to explore a collection of information by applying multiple **filters**. Faceted search is the dynamic clustering of items or search results into categories that let users drill into search results (or even skip searching entirely) by any value in any field. Users can then "drill down" by applying specific constraints to the search results. Look at https://react.rocks/tag/Faceted_Search for some examples.

In this application, the user can filter the search results using available business attributes (i.e. facets) such as category, sub-category, attributes, reviews, stars and votes. Each time the user clicks on a facet value; the set of results is reduced to only the items that have that value. Additional clicks continue to narrow down the search—the previous facet values are remembered and applied again.

You will be designing your application a standalone Java application.

Example screenshots of a possible application are available in Appendix-B. In evaluating your work, instructor's primary focus will be primarily on how you design your database and how efficiently you can search the database and pull out the information. However, your GUI should provide the basic functionality for easy browsing of the business categories and attributes (as illustrated in Appendix-B). Creativity is encouraged!

Project Details:

0. Part 0

Install Oracle Database 11gR2 or later. Consult the instructions provided on Camino under Assignment 3. You will be installing a docket container with an Oracle Database install on this environment that you will be using for your assignment.

I. Part 1

- Download the Yelp dataset from Camino. Look at each JSON file and understand what information the JSON objects provide. Pay attention to the data items in JSON objects that you will need for your application (For example, categories, attributes,…etc.)
- You may have to modify your database design from Homework 2 to model the database for the described application scenario on page-1. Your database schema doesn't necessarily need to include all the data items provided in the JSON files. Your schema should be precise but yet complete. It should be designed in such a way that all queries/data retrievals on/from the database run efficiently and effectively.
- Produce DDL SQL statements for creating the corresponding tables in a relational DBMS. Note the constraints, including key constraints, referential integrity constraints, not NULL constraints, etc. needed for the relational schema to capture and enforce the semantics of your ER design.
- Populate your database with the Yelp data. Generate INSERT statements for your tables and run those to insert data into your DB.
- After you populated your database, created indexes on frequently accessed columns of its tables using CREATE INDEX statement. This will help speed up query execution times. You have some flexibility about which indexes to choose.

II. Part 2

Implement the application for searching local businesses as explained in section "Overview & Requirements". In this milestone you would:

- Write the SQL queries to search your database.
- Establish connectivity with the DBMS.
- Embed/execute queries in/from the code. Retrieve query results and parse the returned results to generate the output that will be displayed on the GUI.
- **Business Search:** Implement a GUI where the user can search for movies that match the criteria given.
    - Browse through main categories for the businesses (See Appendix C); select the business attributes that user wants to search for;
      note: The list of the main categories is given in Appendix-C. All other categories that appear in the business objects are sub-categories. Such a distinction is made for easier browsing of the business categories.
    - The usage flow of the GUI is as follows:
        1) Once the application is loaded, main categories are loaded from the backend database. Note that, selection of business main categories (single or multiple) is required. For instance, assume that use selects *Restaurants* as the main category.
        2) The subcategories matching the main category selection will be listed under subcategories column. Since user selected *Restaurants* in previous step, only sub-categories values that its main category is *Restaurants* should appear in the sub-categories panel. Note how faceted search work here. After step 1, the set of results is reduced to only the businesses that belong to *Restaurants* category. The user can select desired sub-categories values. This attribute is optional in building the query. User might not select a sub-category at all. Assume that use selects *Mediterranean* as the sub-category value.
        3) Attribute column is the next selection. Similar to No 2, the set of results is reduced to only the subcategories **AND** main category selection. The user can select desired attribute values (single or multiple selections). This attribute is optional in building the query. User might not select a attribute values at all. Since user selected *Restaurants*, and *Mediterranean* in previous steps, only attribute values that appeared in business with main-category = Restaurants **AND** sub-category = *Mediterranean*, should appear in the attribute selection panel. Assume that user

selects *Outdoor Sitting* as the desired attribute.

4) Review column is the next selection. You can specify review duration (from/to) and enter the star and vote values into the text box. The attributes under the Review column are also optional.

- The application should be able to search for the businesses that have either all the specified values (AND condition) or that have any of the values specified (OR condition). For example:
  - if user selected AND condition, and selected *Restaurants* and *Cafes* as main categories, sub-categories of businesses that have *Restaurants* **AND** *Cafes* as main categories, should be listed in the next panel.
  - If user selected OR condition, and selected *Restaurants* and *Cafes* as main categories, sub-categories of businesses that have *Restaurants* **OR** *Cafes* as main categories, should be listed in the next panel.

Note that the relation between facets (or business characteristics) is always **AND**. However, the relation between values of one facet can be set to be OR or AND.

Example:
Consider the below example on the AND/OR selection. Assume the following example:

BusinessID	Category	Sub-category
1	restaurant	Mediterranean
2	restaurant	Mexican
3	restaurant	Mediterranean

Suppose User selects Restaurant as main category and both Mediterranean and Mexican as sub-category. Also user selects AND from the "Search for" drop down menu. This means that attributes of businesses that are (Restaurant, Mediterranean) AND (Restaurant, Mexican) should appear in the attribute column.
Look for the conjunction of attributes between business 1, 2, 3 that follow the above rule.
Per above example the following attributes should show in the attribute panel since they are common between all three businesses: (remember that user selected AND from the "Search for" drop down menu)

Ambience_Good_True
Price_Range_1_False

Suppose User selects Restaurant as main category and both Mediterranean and Mexican as sub-category.
Also user selects OR from the "Search for" drop down menu.
This means that attributes of business that are (Restaurant, Mediterranean) OR (Restaurant, Mexican) should appear in the attribute column. So, you have to look for disjunction of attributes between business 1, 2, 3 that follow the above rule.
Per above example, what shows in attribute panel is:

Music_Loud_True
Ambience_Good_True
Parking_Street_False
Price_Range_1_False
Music_Loud_False

- Select a certain business in the search results and list the following for that business(s): Business, City, State, Stars
- select a certain business in the search results and list all the reviews for that business. (*note: The review list should also include the names of the users who provided those reviews*)

- **User Search:** Implement a GUI where the user can perform a search for users that match the criteria given
    - The usage flow of the GUI is as follows:
        1) User can specify user search using attributes such as member_since, review_count, number of friends, average stars and number of friends.
        2) Clicking on "Execute User Query" will show user matches, yelping_since and average_stars
        3) select a certain user in the search results and list all the reviews given by that user.

Note: The user can do either a Business search or User search (not both) at any given time through the GUI.
Note: All data displayed on the GUI should be kept in the database and should be retrieved from it when needed. You are not allowed to create internal data structures to store data.

Required .sql files:
You are required to create two .sql files:

1. createdb.sql: This file should create all required tables. In addition, it should include constraints, indexes, and any other DDL statements you might need for your application.
2. dropdb.sql: This file should drop all tables and the other objects once created by your createdb.sql file.

Required Java Programs:
You are required to implement two Java programs:

1. populate.java: This program should get the names of the input files as command line parameters and populate them into your database. It should be executed as:
   "> java populate yelp_business.json yelp_review.json yelp_checkin.json yelp_user.json".
   Note that every time you run this program, it should remove the previous data in your tables; otherwise the tables will have redundant data.
2. hw3.java: This program should provide a GUI, similar to figure 1, to query your database.

References:

1. Yelp Dataset Challenge, http://www.yelp.com/dataset_challenge/
2. Samples for users of the Yelp Academic Database, https://github.com/Yelp/dataset-examples

Appendix-A

Yelp's Academic Dataset

Yelp has made available a dataset which contains user reviews for 42,153 businessesin Phoenix (AZ), Las Vegas (NV), Madison (WI) in United States and Waterloo (ON) and Edinburgh (ON) in Canada. The purpose was to provide a real-world data set to promote research in various areas of research. The dataset includes 5 types of data objects: *business*, *review*, *user*, *tip*, and *check-in*. Every object contains a 'type' field, which tells whether it is a *business*, a *user*, or a *review*. *Business* objects contain basic information about local businesses. *Review* objects contain the details of the reviews by users for the businesses. *Review*'s user_id associates the reviews with the *user* objects. Similarly, *review*'s business_id associates each review with the *businesses*.

The fields of objects are given below:

Business Objects

Business objects contain basic information about local businesses.

```
{
     'business_id': (encrypted business id),
     'full_address': (localized address),
     'hours': (the days of the week when business is open; the opening and closing times on those days)
     'open': True / False (corresponds to closed, not business hours),
     'categories': (categories associated with the business)
     'city': (city),
     'state': (state),
     'latitude': latitude,
     'longitude': longitude,
     'review_count': review count,
     'name': (business name),
     'neighborhoods': [(hood names)],
     'stars': (star rating, rounded to half-stars),
     'attributes': (business properties),
     'type': 'business'
}
```

Review Objects

Review objects contain the review text, the star rating, and information on votes Yelp users have cast on the review. Use user_id to associate this review with others by the same user. Use business_id to associate this review with others of the same business.

```
{
     'votes': {
         'useful': (count of useful votes),
         'funny': (count of funny votes),
         'cool': (count of cool votes)
       }
     'user_id': (the identifier of the authoring user),
     'review_id': (the identifier of the reviewed business),
     'stars': (star rating, integer 1-5),
     'date': (date, formatted like '2011-04-19'),
     'text': (review text),
     'type': 'review',
     'business_id': (the identifier of the reviewed business)
}
```

User Objects

User objects contain aggregate information about a single user across all of Yelp (including businesses and reviews not in this dataset).

```
{
     'yelping_since': (the date when user account was created)
     'votes': {
         'useful': (count of useful votes across all reviews),
         'funny': (count of funny votes across all reviews),
         'cool': (count of cool votes across all reviews)
```

```
    }
    'review_count': (review count),
    'name': (first name, last initial, like 'Matt J.'),
    'user_id': (unique user identifier),
    'friends': (friends of the user),
    'fans': (number fans of the user),
    'average_stars': (floating point average, like 4.31),
    'type': 'user',
    'compliments': (comments from other users),
    'elite': ()
}
```

Checkin

```
{
    'type': 'checkin',
    'business_id': (encrypted business id),
    'checkin_info': {
        '0-0': (number of checkins from 00:00 to 01:00 on all Sundays),
        '1-0': (number of checkins from 01:00 to 02:00 on all Sundays),
        ...
        '14-4': (number of checkins from 14:00 to 15:00 on all Thursdays),
        ...
        '23-6': (number of checkins from 23:00 to 00:00 on all Saturdays)
    } # if there was no checkin for a hour-day block it will not be in the list
}
```

Tip

```
{
    'user_id': (encrypted user id),
    'text': (),
    'business_id': (encrypted user id),
    'likes': (),
    'date': (),
    'type': 'tip'
}
```

Usage of this dataset is governed by the Academic Dataset Terms of Use.

s

Appendix-B

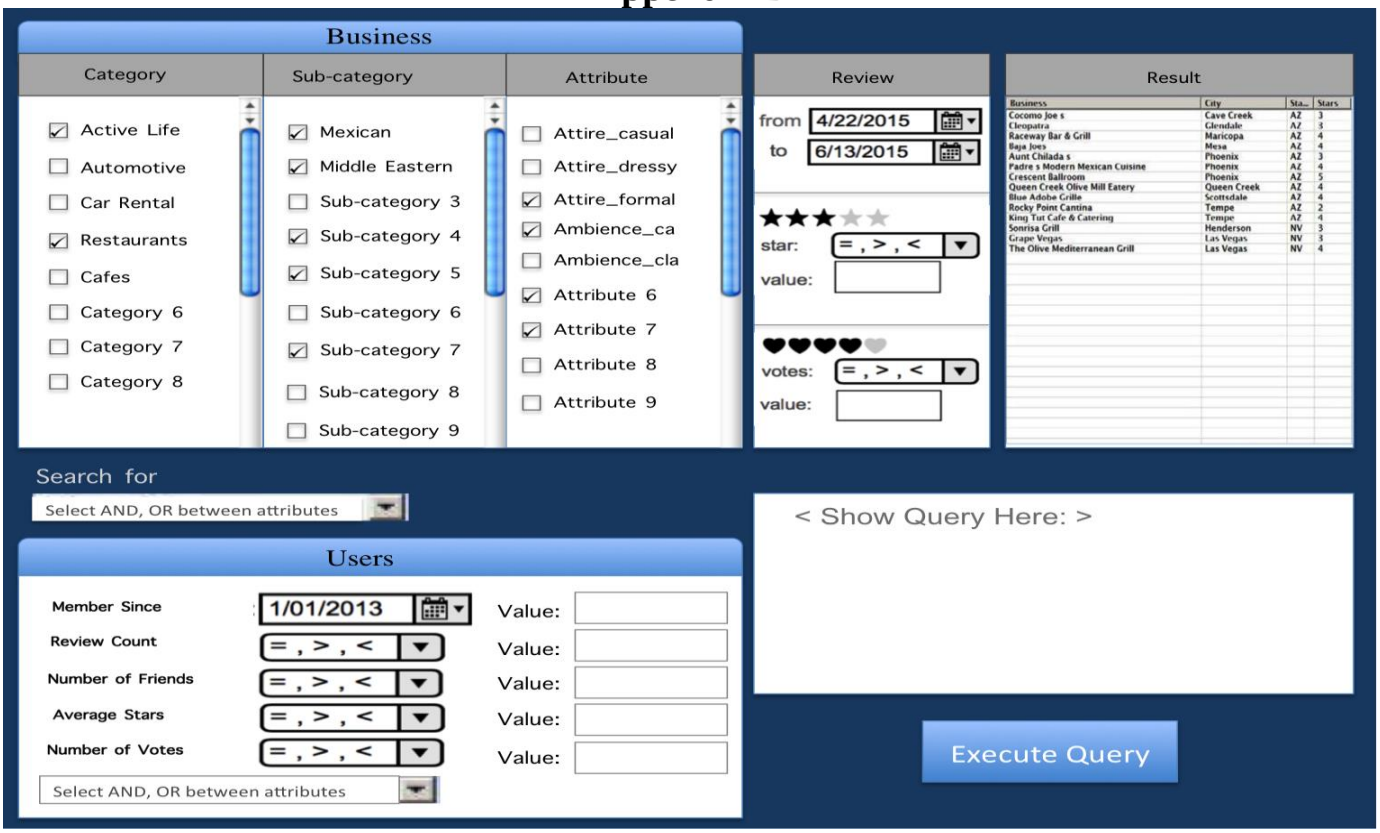

Figure 1- Yelp Application Main UI (Business Search)

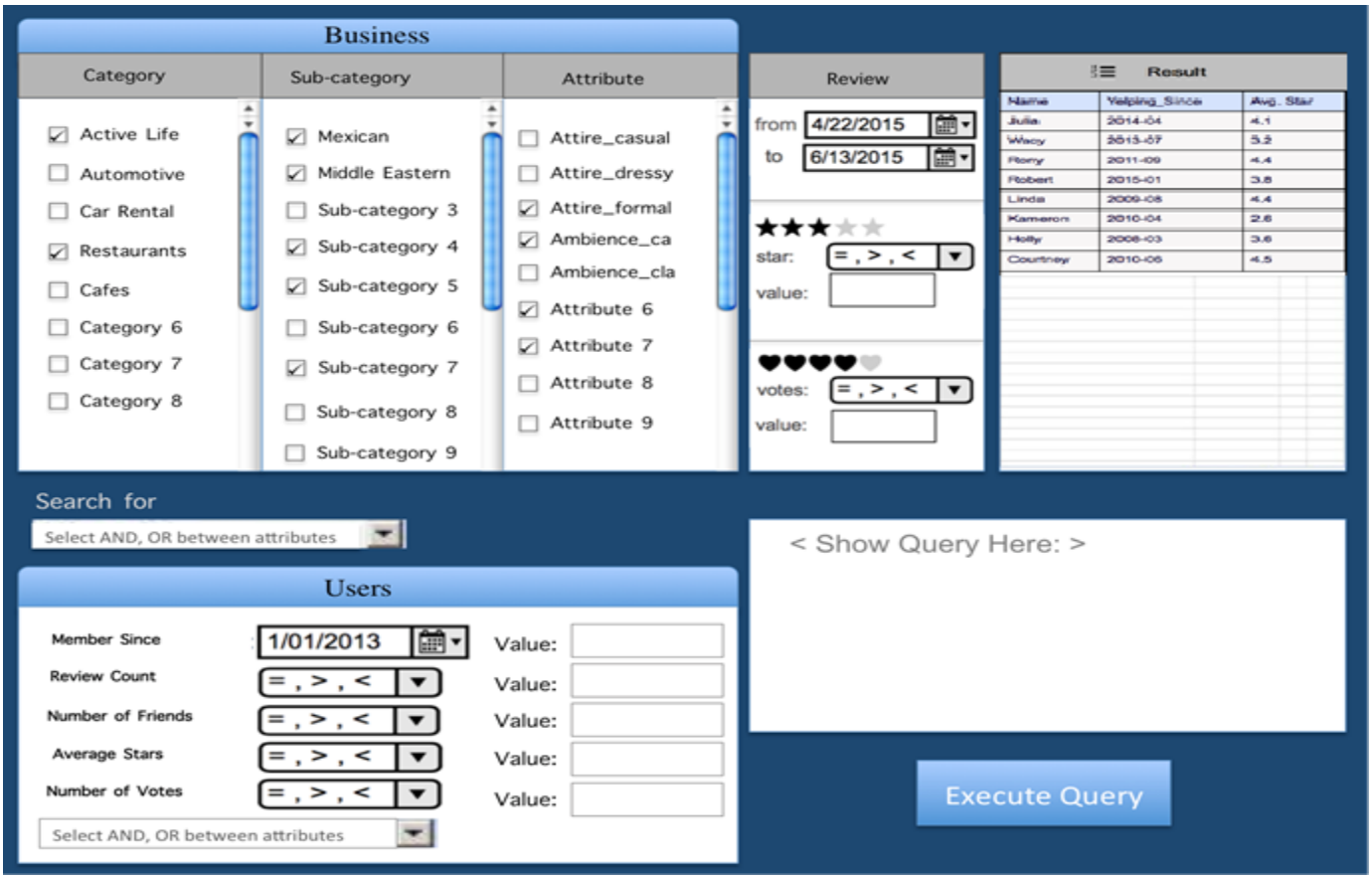

Figure 2- Yelp Application Main UI (User Search)

Appendix-C

Main Business Categories

1. Active Life
2. Arts & Entertainment
3. Automotive
4. Car Rental
5. Cafes
6. Beauty & Spas
7. Convenience Stores
8. Dentists
9. Doctors
10. Drugstores
11. Department Stores
12. Education
13. Event Planning & Services
14. Flowers & Gifts
15. Food
16. Health & Medical
17. Home Services
18. Home & Garden
19. Hospitals
20. Hotels & Travel
21. Hardware Stores
22. Grocery
23. Medical Centers
24. Nurseries & Gardening
25. Nightlife
26. Restaurants
27. Shopping
28. Transportation