Assignment 3

In this assignment, you will be building a single page application that make use of APIs published by Yelp https://www.yelp.ca/developers/documentation/v3/get_started. You will use this APIs to get restaurants' information and reviews in Vancouver, BC.

Authentication

You need to sign up at https://www.yelp.ca/developers/v3/manage_app to obtain an API key (registration is free).
Read https://www.yelp.com/developers/documentation/v3/authentication to understand how the API key is used as authentication for the queries made to Yelp. Below is a sample of how you will add the API key into the header of the API call. Replace 'xxxxx' with your private API key.

```
var request = require('request');
var options = {
  url:
'https://api.yelp.com/v3/businesses/search?term=restaurants&locale=en_CA
&location=vancouver,bc,canada.........',
  headers: {
    'Authorization': 'Bearer xxxxx'
  }
};
request(options, function(error, response, body) {
  if (error) {
    return console.error(error);
  } else {
    console.log('data from API call = ' +JSON.stringify(body));
```

Requirements

You will implement both the backend and frontend for this website. The backend will be implemented using Node.js with Express as the Web Framework while the frontend will be implemented using jQuery, javascript and basic HTML.

Backend

The backend has 2 endpoints:

1. GET /restaurants/<selected_cuisine>
2. GET /reviews/<id>

The backend will also provide an index.html as a landing page for the users.

The 1st endpoint **GET /restaurants/<selected_cuisine>** will be calling the Yelp API https://www.yelp.com/developers/documentation/v3/business_search, passing in the selected

cuisine as the value for *categories*. The location and locale will be limited to Vancouver and CA respectively. So the query URL should look like this:
https://api.yelp.com/v3/businesses/search?term=restaurants&categories=chinese&locale=en_CA&location=vancouver,bc,canada&....

This endpoint will return the following data from the API response to the UI:

- name
- url
- review_count
- rating
- price
- location

Other requirements:

- The result size from Yelp will be limited to the top 10 restaurants
- The user can sort the result by best_match, rating or review_count
- The user can filter the result by price

The 2nd endpoint **GET /reviews/<id>** will be calling the Yelp API https://www.yelp.com/developers/documentation/v3/business_reviews. This API call will return 3 reviews of the restaurant specified by the provided ***id***.
This endpoint will return the ***text*** of the reviews from the API response to the UI.

Frontend

The frontend is the implementation of **index.html** which will have 3 dropdown lists and a submit button.

1. The **Cuisine** dropdown list has the following options:
   a. italian
   b. mexican
   c. chinese
   d. japanese
2. The **sort by** dropdown list has the following options:
   a. best_match
   b. rating
   c. review_count
3. The **price filter** dropdown list has the following options:
   a. Inexpensive
   b. Moderate
   c. Pricey
   d. Ultra High-End

On click of the submit button, a call will be made to **GET /restaurants/<selected_cuisine>** endpoint. The sort by and price filter will be pass as query parameters.

The result from the query can be rendered to an unordered list on the same page below the dropdown lists.

Alongside with the display for each restaurant, you will provide a "*see reviews*" link which on click will call **GET /reviews/<id>** to get reviews of restaurant. The reviews of the restaurant will be appended to expanded rows below the restaurant itself. The user can collapse the reviews of the restaurant by clicking a "*close reviews"* link. When the user clicks on the review of another restaurant, all expanded reviews of other restaurants need to be collapsed. Look into using jQuery effect to expand and collapse the reviews of the restaurants.