**COMP 3133 – Full Stack Development II**

**Lab 03 MongoDB & Mongoose**

**Create NodeJS + Express + MongoDB + mongoose application to perform following operations**

**Note**:

* Create **GitHub** repository and commit all your code to GitHub
* Use **Postman** to test the result
* **Lab04 and Lab05 (Week05/Week06) work will be depended on this lab03**

**NodeJS Project**

1. Create NodeJS project having name **lab3\_restaurant\_database**
2. Install required modules to work with **express** and **mongoose**
3. Create Restaurant Schema in your project to perform mongoDB/mongoose query operations
4. Create REST API to return all restaurant details

* Select all the columns

[**http://localhost:3000/restaurants**](http://localhost:3000/restaurants)

**A screenshot of a computer

Description automatically generated**

1. Create REST API to return all restaurant details by cuisine

* Select all the columns

[**http://localhost:3000/restaurants/cuisine/Japanese**](http://localhost:3000/restaurants/cuisine/Japanese)

**A screenshot of a computer

Description automatically generated**

[**http://localhost:3000/restaurants/cuisine/Bakery**](http://localhost:3000/restaurants/cuisine/Bakery)

**A screenshot of a computer

Description automatically generated**

[**http://localhost:3000/restaurants/cuisine/Italian**](http://localhost:3000/restaurants/cuisine/Italian)

**A screenshot of a computer

Description automatically generated**

1. Create REST API to return the

* The selected columns must include id, cuisines, name, city, resturant\_id
* The sorting by the restaurant\_id in Ascending or Descending Order based on parameter passed.

[**http://localhost:3000/restaurants?sortBy=ASC**](http://localhost:3000/restaurants?sortBy=ASC)

**A screenshot of a computer

Description automatically generated**

[**http://localhost:3000/restaurants?sortBy=DESC**](http://localhost:3000/restaurants?sortBy=DESC)

**A screenshot of a computer

Description automatically generated**

1. Create REST API to return restaurants details where all cuisines are **equal** to Delicatessen **and** the city is **not equal** to Brooklyn

* The selected columns must include cuisines, name and city, but exclude id
* The sorting order must be Ascending Order on the name

**http://localhost:3000/restaurants/Delicatessen**

**A screenshot of a computer

Description automatically generated**

**MongoDB Atlas**

[**https://account.mongodb.com/account/login**](https://account.mongodb.com/account/login)

Creating Collections and Documents name **Restaurants**

Use the seed data script found here.   
<https://drive.google.com/file/d/1ON-ARc3cHHb45zi1att3StiP9Y37MIEG/view?usp=sharing>

**Reference**

* Lab work done during class
* <https://mongoosejs.com/docs/guide.html>
* <https://docs.mongodb.com/manual/reference/operator/query-comparison/>
* <https://docs.mongodb.com/manual/reference/operator/query-logical/>
* <https://techeplanet.com/express-path-parameter/>
* https://techeplanet.com/express-get-query-string-parameters/
* https://techeplanet.com/express-send-json-response/
* <https://kb.objectrocket.com/mongo-db/or-in-mongoose-1018>
* <https://kb.objectrocket.com/category/mongo-db>

**Submission**

* + **Screenshot(s)** (Postman) with appropriate title showing your REST API and output. You may add all the screenshots in word document and upload it**. *[Do not put screenshots in project/zip file]****.*
  + **Delete the node\_modules** folder as it has many files which are big in size and is not required for the submission.Create **a zip file** of your entire project folder and submit.
  + Your **GitHub** project link.