PRACTICAL-3

NoSQL

AIM: Performing queries based on AND, OR, Limit, Sort and Projection and apply some queries to get specified output.

1. Write a MongoDB query to display all the documents in the collection restaurants.

# db.restaurants.find().pretty()



2. Write a MongoDB query to display the fields restaurant\_id, name, borough and cuisine for all the documents in the collection restaurant.

# db. restaurants.find({},{"restaurant\_id" : 1,"name" :1,"borough":1,"cuisine" :1}).pretty()



3. Write a MongoDB query to display the fields restaurant\_id, name, borough

and zip code, but exclude the field \_id for all the documents in the collection restaurant. (USING PROJECTION)

# db. restaurants.find().pretty().limit(5)



4. Write a MongoDB query to display the first 5 restaurant which is in the borough Bronx. (USING LIMIT)

# db. restaurants.find({borough:"Bronx"}).pretty().limit(5)



1. Write a MongoDB query to display the next 5 restaurants after skipping first 5 which are in the borough Bronx. (USING SKIP) db.restaurants.find({borough:"Bronx"}).pretty().limit(5).skip(5)



1. Write a MongoDB query to find the restaurants that do not prepare any cuisine of 'American' and their grade score more than 70 and latitude less than -65.754168. (USING AND)

# db. restaurants.find({$and:[{"cuisine":{$ne:"American "}},{"grades.score":{$gt:70}},{"address.coord":{$lt:-65.754168}}]}).pretty()



1. Write a MongoDB query to find the restaurants which belong to the borough Bronx and prepared either American or Chinese dish. (USING OR) db. restaurants.find({"borough": "Bronx" ,$or:[{"cuisine":"American "},{"cuisine":"Chinese"}]}).pretty()



1. Write a MongoDB query to arrange the name of the restaurants in ascending / descending order along with all the columns. (USING SORT)

db.restaurants.find().pretty().sort({"name":1});

