PRACTICAL-3

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

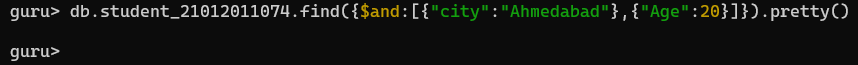
* PRACTICE QUESTION

**1.Find all the student details whose city is Ahmedabad and age is 20.**

**INPUT:**

db.student\_21012011074.find({$and:[{"city":"Ahmedabad"},{"Age":20}]}).pretty()

**Output :-**

****

**2.Display enrolment number of students whose enrolment number is greater than 3 or age is 20.**

**INPUT:**

**db.student\_21012011074.find({$or:[{en\_no:{$gt:3}},{"Age":20}]}).pretty()**

**OUTPUT:**

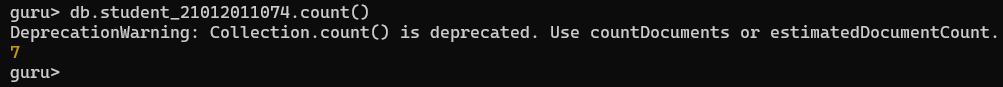
****

**Q.3 Count the number of documents whose age is > 20.**

**INPUT:**

**db.student\_21012011074.count()**

**OUTPUT:**

****

**Q.4 Only display first 5 records of student collection.**

INPUT:

db.student\_21012011074.find().limit(5)

OUTPUT:

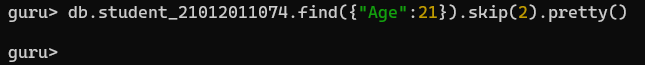
****

**Q.5 Display records of students whose age is 21, skip first 2 records.**

INPUT:

db.student\_21012011074.find({"Age":21}).skip(2).pretty()

OUTPUT:

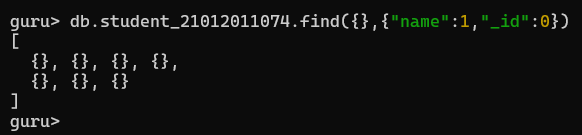
****

**Q.6 Sort & display records of students based on ascending order of name.**

**INPUT:**

db.student\_21012011074.find({},{"name":1,"\_id":0})

**OUTPUT:**

****

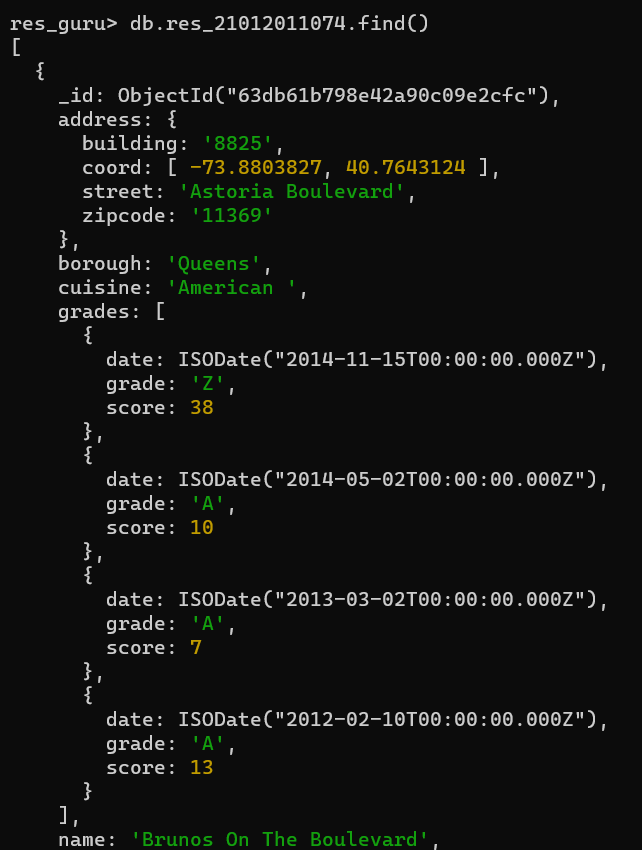
* **Exercise Question**

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

INPUT:

db.res\_21012011074.find()

OUTPUT:

****

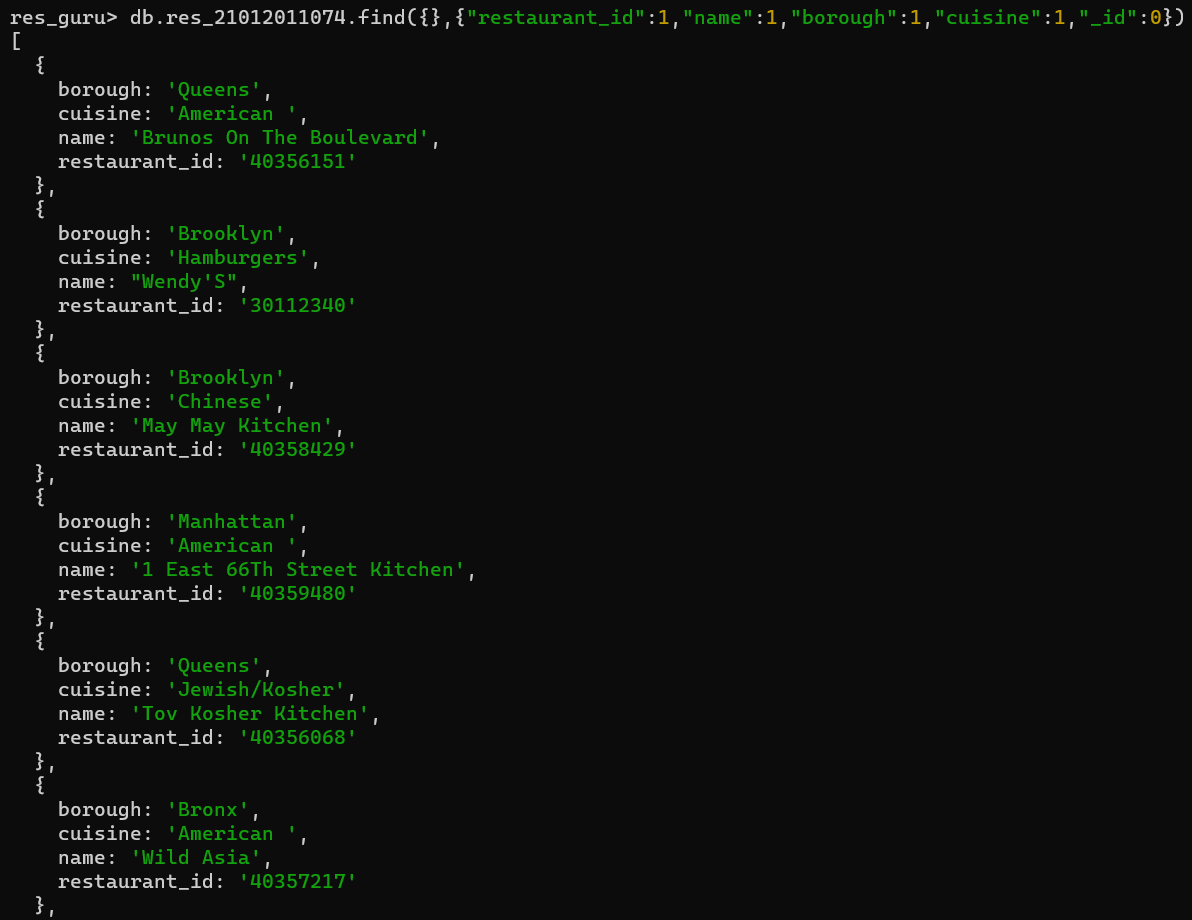
And Continue……

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

INPUT;

db.res\_21012011074.find({},{"restaurant\_id":1,"name":1,"borough":1,"cuisine":1,"\_id":0})

OUTPUT:

****

**And Continue……..**

1. 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)

INPUT:

db.res\_21012011074.find({},{"restaurant\_id":1,"name":1,"borough":1,"address.zipcode":1,"\_id":0})

OUTPUT:

****

**And Continue…….**

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

INPUT:

db.res\_21012011074.find({"borough":"Bronx"}).limit(5)

OUTPUT:

****

**And Continue…..**

1. Write a MongoDB query to display the next 5 restaurants after skipping first 5 which are in the borough Bronx. (USING SKIP)

INPUT:

db.res\_21012011074.find({"borough":"Bronx"},{"\_id":0}).skip(5)

OUTPUT:

****

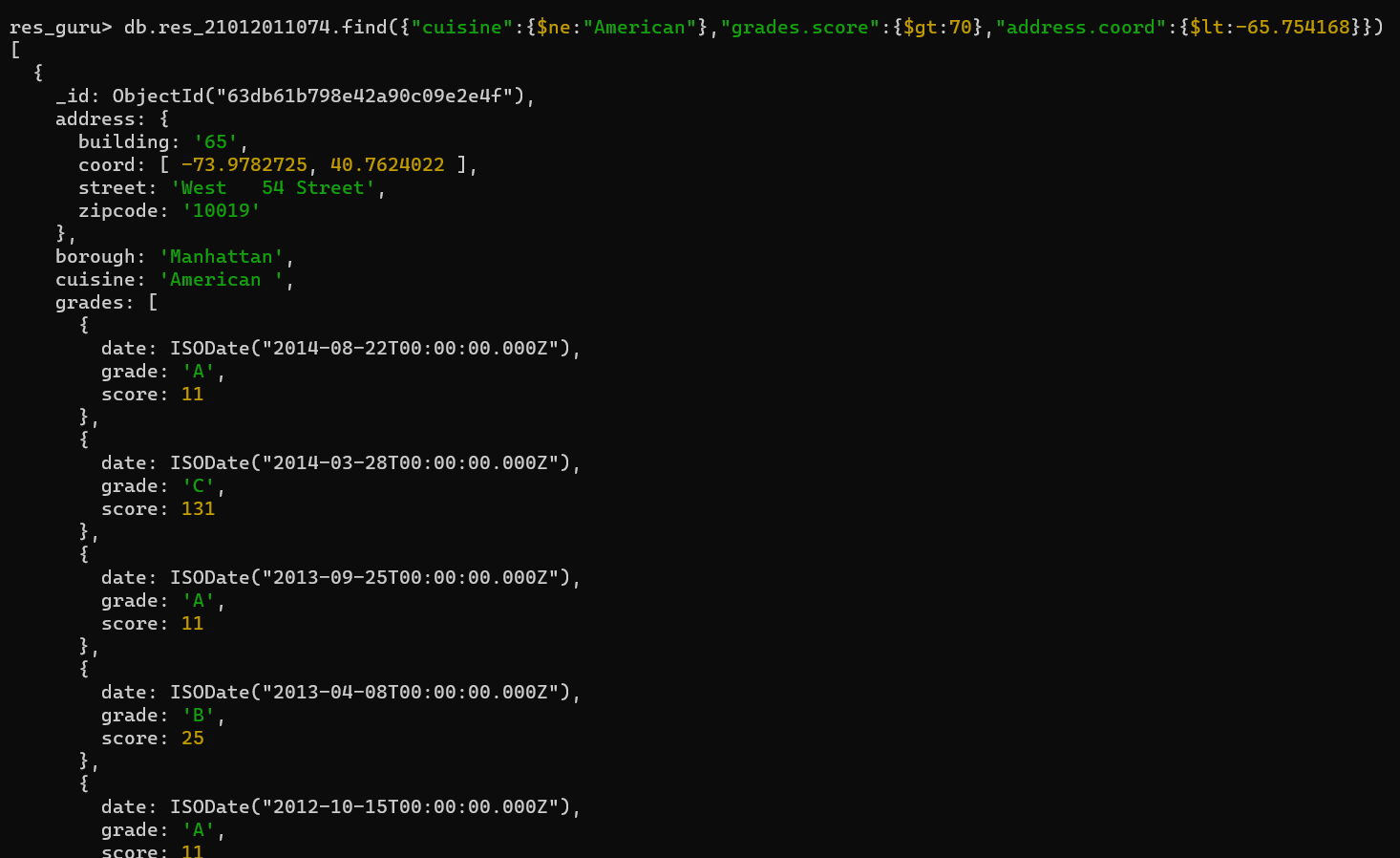
**And Continue……**

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)

INPUT:

db.res\_21012011074.find({"cuisine":{$ne:"American"},"grades.score":{$gt:70},"address.coord":{$lt:-65.754168}})

OUTPUT:

****

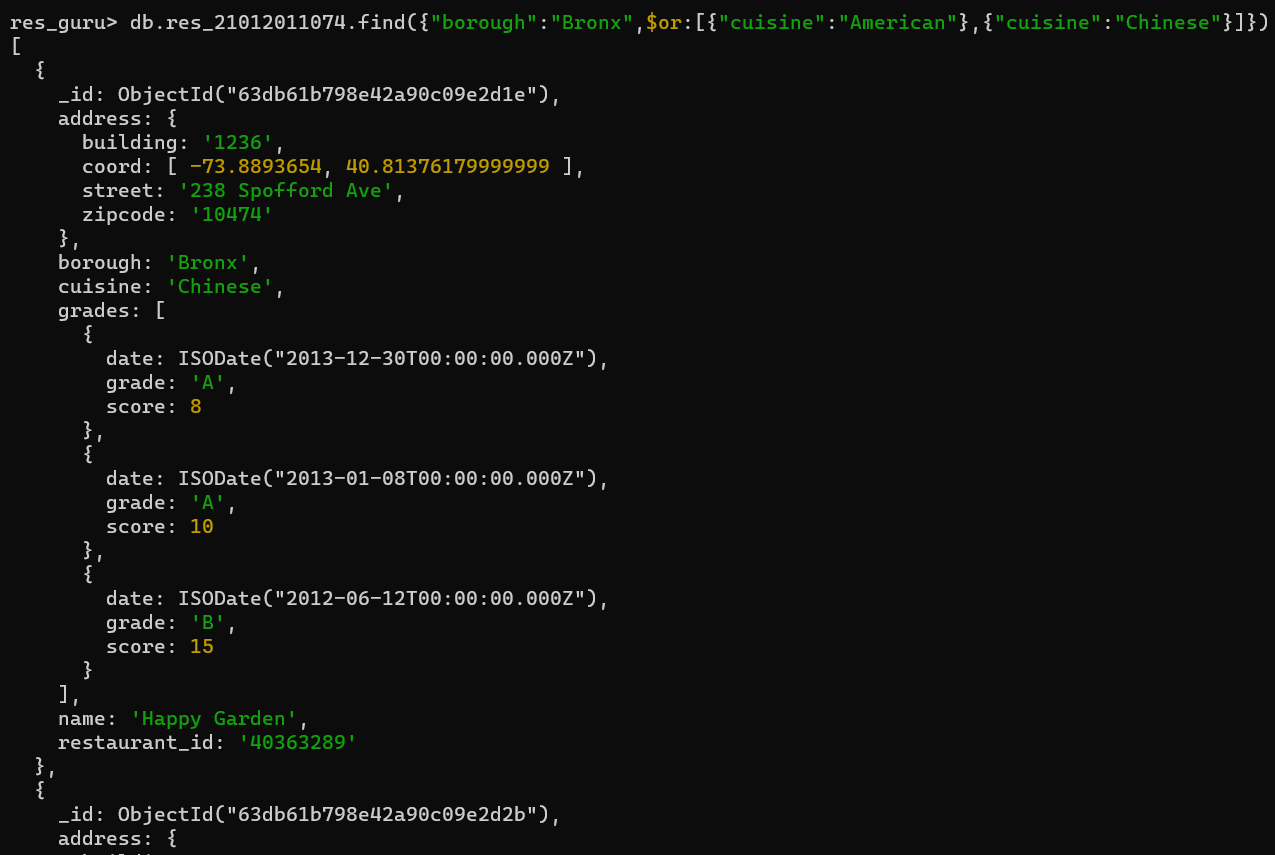
**And Continue…….**

1. Write a MongoDB query to find the restaurants which belong to the borough Bronx and prepared either American or Chinese dish. (USING OR)

INPUT:

db.res\_21012011074.find({"borough":"Bronx",$or:[{"cuisine":"American"},{"cuisine":"Chinese"}]})

OUTPUT:

****

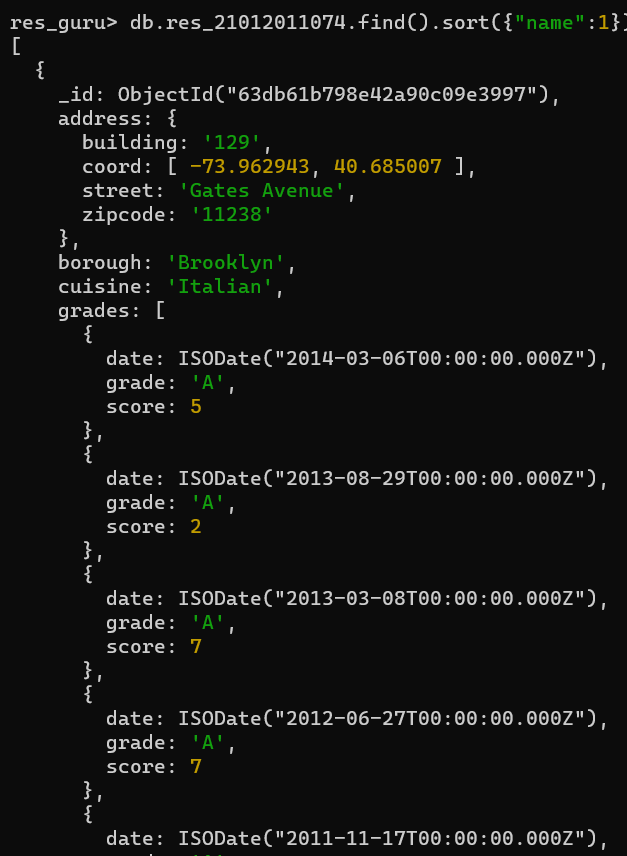
**And Continue…….**

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

INPUT:

db.res\_21012011074.find().sort({"name":1})

OUTPUT:

****