1. Use restaurants
2. db.createCollection("restaurants");
3. db.restaurants.insert({
   ... "address": {
   ... "building": "1007",
   ... "street": "HighStreet",
   ... "zipcode": "TW31HA"
   ... },
   ... "borough": "Hounslow",
   ... "cuisine": "Bakery",
   ... "Orders": [
   ... { "date": { "$date": 1393804800000 }, "rating": "A", "stars": 2 },
   ... { "date": { "$date": 1378857600000 }, "rating": "A", "stars": 6 },
   ... { "date": { "$date": 1358985600000 }, "rating": "A", "stars": 10 },
   ... { "date": { "$date": 1322006400000 }, "rating": "C", "stars": 9 },
   ... { "date": { "$date": 1299715200000 }, "rating": "B", "stars": 14 }
   ...],
   ... "name": "Just Bake Shop",
   ... "restaurant_id": "5075445"
   ... });
4. db.restaurants.insertMany([{"address": {"building": "1007", "coord": [-73.856077, 40.848447], "street": "Morris Park Ave", "zipcode": "10462"}, "borough": "Bronx", "cuisine": "Bakery", "grades": [{"date": {"$date": 1393804800000}, "grade": "A", "score": 2}, {"date": {"$date": 1378857600000}, "grade": "A", "score": 6}, {"date": {"$date": 1358985600000}, "grade": "A", "score": 10}, {"date": {"$date": 1322006400000}, "grade": "A", "score": 9}, {"date": {"$date": 1299715200000}, "grade": "B", "score": 14}], "name": "Morris Park Bake Shop", "restaurant_id": "30075445"},

   {"address": {"building": "469", "coord": [-73.961704, 40.662942], "street": "Flatbush Avenue", "zipcode": "11225"}, "borough": "Brooklyn", "cuisine": "Hamburgers", "grades": [{"date": {"$date": 1419897600000}, "grade": "A", "score": 8}, {"date": {"$date": 1404172800000}, "grade": "B", "score": 23}, {"date": {"$date": 1367280000000}, "grade": "A", "score": 12}, {"date": {"$date": 1336435200000}, "grade": "A", "score": 12}], "name": "Wendy'S", "restaurant_id": "30112340"},

   {"address": {"building": "351", "coord": [-73.98513559999999, 40.7676919], "street": "West 57 Street", "zipcode": "10019"}, "borough": "Manhattan", "cuisine": "Irish", "grades": [{"date": {"$date": 1409961600000}, "grade": "A", "score": 2}, {"date": {"$date": 1374451200000}, "grade": "A", "score": 11}, {"date": {"$date": 1343692800000}, "grade": "A", "score": 12}, {"date":

{"$date": 1325116800000}, "grade": "A", "score": 12}], "name": "Dj Reynolds Pub And Restaurant", "restaurant_id": "30191841"},

{"address": {"building": "2780", "coord": [-73.98241999999999, 40.579505], "street": "Stillwell Avenue", "zipcode": "11224"}, "borough": "Brooklyn", "cuisine": "American ", "grades": [{"date": {"$date": 1402358400000}, "grade": "A", "score": 5}, {"date": {"$date": 1370390400000}, "grade": "A", "score": 7}, {"date": {"$date": 1334275200000}, "grade": "A", "score": 12}, {"date": {"$date": 1318377600000}, "grade": "A", "score": 12}], "name": "Riviera Caterer", "restaurant_id": "40356018"},

{"address": {"building": "97-22", "coord": [-73.8601152, 40.7311739], "street": "63 Road", "zipcode": "11374"}, "borough": "Queens", "cuisine": "Jewish/Kosher", "grades": [{"date": {"$date": 1416787200000}, "grade": "Z", "score": 20}, {"date": {"$date": 1358380800000}, "grade": "A", "score": 13}, {"date": {"$date": 1343865600000}, "grade": "A", "score": 13}, {"date": {"$date": 1323907200000}, "grade": "B", "score": 25}], "name": "Tov Kosher Kitchen", "restaurant_id": "40356068"},

{"address": {"building": "8825", "coord": [-73.8803827, 40.7643124], "street": "Astoria Boulevard", "zipcode": "11369"}, "borough": "Queens", "cuisine": "American ", "grades": [{"date": {"$date": 1416009600000}, "grade": "Z", "score": 38}, {"date": {"$date": 1398988800000}, "grade": "A", "score": 10}, {"date": {"$date": 1362182400000}, "grade": "A", "score": 7}, {"date": {"$date": 1328832000000}, "grade": "A", "score": 13}], "name": "Brunos On The Boulevard", "restaurant_id": "40356151"},

{"address": {"building": "2206", "coord": [-74.1377286, 40.6119572], "street": "Victory Boulevard", "zipcode": "10314"}, "borough": "Staten Island", "cuisine": "Jewish/Kosher", "grades": [{"date": {"$date": 1412553600000}, "grade": "A", "score": 9}, {"date": {"$date": 1400544000000}, "grade": "A", "score": 12}, {"date": {"$date": 1365033600000}, "grade": "A", "score": 12}, {"date": {"$date": 1327363200000}, "grade": "A", "score": 9}], "name": "Kosher Island", "restaurant_id": "40356442"},

{"address": {"building": "7114", "coord": [-73.9068506, 40.6199034], "street": "Avenue U", "zipcode": "11234"}, "borough": "Brooklyn", "cuisine": "Delicatessen", "grades": [{"date": {"$date": 1401321600000}, "grade": "A", "score": 10}, {"date": {"$date": 1389657600000}, "grade": "A", "score": 10}, {"date": {"$date": 1375488000000}, "grade": "A", "score": 8}, {"date": {"$date": 1342569600000}, "grade": "A", "score": 10}, {"date": {"$date": 1331251200000}, "grade": "A", "score": 13}, {"date": {"$date": 1318550400000}, "grade": "A", "score": 9}], "name": "Wilken'S Fine Food", "restaurant_id": "40356483"},

{"address": {"building": "6409", "coord": [-74.00528899999999, 40.628886], "street": "11 Avenue", "zipcode": "11219"}, "borough": "Brooklyn", "cuisine": "American ", "grades": [{"date": {"$date": 1405641600000}, "grade": "A", "score": 12}, {"date": {"$date": 1375142400000}, "grade": "A", "score": 12}, {"date": {"$date": 1360713600000}, "grade": "A", "score": 11}, {"date": {"$date": 1345075200000}, "grade": "A", "score": 2}, {"date": {"$date": 1313539200000}, "grade": "A", "score": 11}], "name": "Regina Caterers", "restaurant_id": "40356649"},
{"address": {"building": "1839", "coord": [-73.9482609, 40.6408271], "street": "Nostrand Avenue", "zipcode": "11226"}, "borough": "Brooklyn", "cuisine": "Ice Cream, Gelato, Yogurt, Ices", "grades": [{"date": {"$date": 1405296000000}, "grade": "A", "score": 12}, {"date": {"$date": 1373414400000}, "grade": "A", "score": 8}, {"date": {"$date": 1341964800000}, "grade": "A", "score": 5}, {"date": {"$date": 1329955200000}, "grade": "A", "score": 8}], "name": "Taste The Tropics Ice Cream", "restaurant_id": "40356731"},
{"address": {"building": "2300", "coord": [-73.8786113, 40.8502883], "street": "Southern Boulevard", "zipcode": "10460"}, "borough": "Bronx", "cuisine": "American ", "grades": [{"date": {"$date": 1401235200000}, "grade": "A", "score": 11}, {"date": {"$date": 1371600000000}, "grade": "A", "score": 4}, {"date": {"$date": 1339718400000}, "grade": "A", "score": 3}], "name": "Wild Asia", "restaurant_id": "40357217"}]);
{"address": {"building": "7715", "coord": [-73.9973325, 40.61174889999999], "street": "18 Avenue", "zipcode": "11214"}, "borough": "Brooklyn", "cuisine": "American ", "grades": [{"date": {"$date": 1397606400000}, "grade": "A", "score": 5}, {"date": {"$date": 1366675200000}, "grade": "A", "score": 2}, {"date": {"$date": 1335225600000}, "grade": "A", "score": 5}, {"date": {"$date": 1323993600000}, "grade": "A", "score": 2}], "name": "C & C Catering Service", "restaurant_id": "40357437"}
{"address": {"building": "1269", "coord": [-73.871194, 40.6730975], "street": "Sutter Avenue", "zipcode": "11208"}, "borough": "Brooklyn", "cuisine": "Chinese", "grades": [{"date": {"$date": 1410825600000}, "grade": "B", "score": 21}, {"date": {"$date": 1377648000000}, "grade": "A", "score": 7}, {"date": {"$date": 1364860800000}, "grade": "C", "score": 56}, {"date": {"$date": 1344988800000}, "grade": "B", "score": 27}, {"date": {"$date": 1332892800000}, "grade": "B", "score": 27}], "name": "May May Kitchen", "restaurant_id": "40358429"}

QUERIES

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

```
db.restaurants.find();
```

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});
```

3. Write a MongoDB query to display the fields restaurant_id, name, borough and cuisine, but exclude the field _id for all the documents in the collection restaurant.

```
db.restaurants.find({},{"restaurant_id" : 1,"name":1,"borough":1,"cuisine" :1,"_id":0});
```

4. 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.

5. Write a MongoDB query to display all the restaurant which is in the borough Harrow.

```
db.restaurants.find({"borough": "Harrow"});
```

6. Write a MongoDB query to display the first 5 restaurant which is in the borough Croydon.

```
db.restaurants.find({"borough": "Bronx"}).limit(5);
```

7. Write a MongoDB query to display the next 5 restaurants after skipping first 5 which are in the borough Hounslow.

```
db.restaurants.find({"borough":"Hounslow"}).skip(5).limit(5);
```

8. Write a MongoDB query to find the restaurants who achieved a score more than 90.

```
db.restaurants.find({"grades.score" : {"$gt" : 90}})
```

9. Write a MongoDB query to find the restaurants that achieved a score, more than 80 but less than 100.

```
db.restaurants.find({$and : [{"grades.score" : {"$gt" : 90}},{"grades.score" : {"$lt" : 100}}]})
```

10. Write a MongoDB query to find the restaurants which locate in Hounslow.

```
db.restaurants.find({"borough":"Hounslow"});
```

11. Write a MongoDB query to find the restaurants that do not prepare any cuisine of 'American' and their grade star more than 70 and located in Harrow.

```
db.rest.find({$and : [{"cuisine" : {$ne : "American "}}, {"borough":"Harrow"}, {"grades.score" : {$gt : 70}}]})
```

12. Write a MongoDB query to find the restaurants which do not prepare any cuisine of 'American' and achieved a score more than 70 and and located in Harrow..
Note : Do this query without using $and operator.

```
db.restaurants.find({ "cuisine" : {$ne : "American "},"grades.score" :{$gt: 70},
"borough":"Harrow"});
```

13. Write a MongoDB query to find the restaurants which do not prepare any
cuisine of 'American ' and achieved a grade point 'A' not belongs to the borough
Brooklyn. The document must be displayed according to the cuisine in
descending order.

```
db.restaurants.find({$and : [{"cuisine" : {$ne : "American "}}, {"grades.grade"
: "A"}, {"borough" : {$ne : "Brooklyn "}}]}).sort({cuisine : -1})
```

14. Write a MongoDB query to find the restaurant Id, name, borough and cuisine
for those restaurants which contain 'Wil' as first three letters for its name.

```
db.rest.find({"name" : { $regex: /^Wil.*/}}, {_id:0, restaurant_id:1, name:1,
borough:1, cuisine:1})
```

15. Write a MongoDB query to find the restaurant Id, name, borough and cuisine
for those restaurants which contain 'ces' as last three letters for its name.

```
db.restaurants.find({"name" : { $regex: /.*ces$/}}, {_id:0, restaurant_id:1,
name:1, borough:1, cuisine:1})
```

16. Write a MongoDB query to find the restaurant Id, name, borough and cuisine
for those restaurants which contain 'Reg' as three letters somewhere in its name.

```
db.rest.find({"name" : { $regex: /Reg/}}, {_id:0, restaurant_id:1, name:1,
borough:1, cuisine:1})
```

17. Write a MongoDB query to find the restaurants which belong to the borough Hounslow and prepared either American or Chinese

```
db.rest.find({borough: "Bronx", cuisine: {$in: ["American ","Chinese"]}},
{_id:0, restaurant_id:1, name:1, borough:1, cuisine:1})
```

18. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which belong to the borough Windsor or Harrow or WestMinister.

```
db.restaurants.find({"borough" :{$in :["Windsor ","Harrow","WestMinister"]}},
{"restaurant_id" : 1,"name":1,"borough":1,"cuisine" :1});
```

19. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which are not belonging to the borough Harrow or Hounslow or Ealing.

```
db.rest.find( {borough: {$nin: ["Harrow","Hounslow ","Ealing"]}} , {_id:0,
restaurant_id:1, name:1, borough:1, cuisine:1})
```

20. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which achieved a score which is not more than 10.

```
db.restaurants.find({"grades.score": {$lte: 10}}, {_id:0, restaurant_id:1,
name:1, borough:1, cuisine:1})
```

{ "borough" : "Bronx", "cuisine" : "Bakery", "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "borough" : "Manhattan", "cuisine" : "Irish", "name" : "Dj Reynolds Pub And Restaurant", "restaurant_id" : "30191841" }
{ "borough" : "Brooklyn", "cuisine" : "American ", "name" : "Riviera Caterer", "restaurant_id" : "40356018" }
{ "borough" : "Queens", "cuisine" : "American ", "name" : "Brunos On The Boulevard", "restaurant_id" : "40356151" }

{ "borough" : "Staten Island", "cuisine" : "Jewish/Kosher", "name" : "Kosher Island", "restaurant_id" : "40356442" }

{ "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Wilken'S Fine Food", "restaurant_id" : "40356483" }

{ "borough" : "Brooklyn", "cuisine" : "American ", "name" : "Regina Caterers", "restaurant_id" : "40356649" }

{ "borough" : "Brooklyn", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Taste The Tropics Ice Cream", "restaurant_id" : "40356731" }

{ "borough" : "Bronx", "cuisine" : "American ", "name" : "Wild Asia", "restaurant_id" : "40357217" }

21. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which prepared dish except 'American' and 'Chinees' or restaurant's name begins with letter 'Wil'.

```
db.rest.find({$nor: [{cuisine: {$in: ["American ","Chinese"]}},{name:
/^Wil.*/}]},{_id:0, restaurant_id:1, name:1, borough:1, cuisine:1})
```

22. Write a MongoDB query to find the restaurant Id, name, and grades for those restaurants which achieved a grade of "A" and scored 11 on an ISODate "2014-08-11T00:00:00Z" among many of survey dates.

```
db.restaurants.find({
"grades.date":ISODate("2014-08-11T00:00:00Z"),"grades.grade":"A"
,"grades.score" : 11}, {"restaurant_id" : 1,"name":1,"grades":1} );
```

23. Write a MongoDB query to find the restaurant Id, name and grades for those restaurants where the 2nd element of grades array contains a grade of "A" and score 9 on an ISO Date "2014-08-11T00:00:00Z".

```
db.restaurants.find({$and: [{"grades.1.grade":"A"}, {"grades.1.score":
9}, {"grades.1.date": ISODate("2014-08-11T00:00:00Z")}]},{_id:0,
restaurant_id:1, name:1, grades:1}).pretty()
```

24. Write a MongoDB query to find the restaurant Id, name, address and geographical location for those restaurants where 2nd element of coord array contains a value which is more than 42 and upto 52.

```
db.restaurants.find({$and : [{"address.coord.1": {$gt : 42}},{"address.coord.1": {$lte : 52}}]}, {_id:0, restaurant_id:1, name:1, address:1})
```

25. Write a MongoDB query to arrange the name of the restaurants in ascending order along with all the columns.

```
db.restaurants.find({},{_id:0, name:1}).sort( {name: 1})
```

```
{ "name" : "Brunos On The Boulevard" }
{ "name" : "Dj Reynolds Pub And Restaurant" }
{ "name" : "Just Bake Shop" }
{ "name" : "Kosher Island" }
{ "name" : "Morris Park Bake Shop" }
{ "name" : "Regina Caterers" }
{ "name" : "Riviera Caterer" }
{ "name" : "Taste The Tropics Ice Cream" }
{ "name" : "Tov Kosher Kitchen" }
{ "name" : "Wendy'S" }
{ "name" : "Wild Asia" }
{ "name" : "Wilken'S Fine Food" }
```

26. Write a MongoDB query to arrange the name of the restaurants in descending along with all the columns.

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

{ "_id" : ObjectId("6331268eca37aeff25a2e14a"), "address" : { "building" : "7114", "coord" : [-73.9068506, 40.6199034], "street" : "Avenue U", "zipcode" : "11234" }, "borough" : "Brooklyn", "cuisine" : "Delicatessen", "grades" : [{ "date" : { "$date" : 1401321600000 }, "grade" : "A", "score" : 10 }, { "date" : {

"$date" : 1389657600000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1375488000000 }, "grade" : "A", "score" : 8 }, { "date" : { "$date" : 1342569600000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1331251200000 }, "grade" : "A", "score" : 13 }, { "date" : { "$date" : 1318550400000 }, "grade" : "A", "score" : 9 }], "name" : "Wilken'S Fine Food", "restaurant_id" : "40356483" }
{ "_id" : ObjectId("6331268eca37aeff25a2e14d"), "address" : { "building" : "2300", "coord" : [-73.8786113, 40.8502883], "street" : "Southern Boulevard", "zipcode" : "10460" }, "borough" : "Bronx", "cuisine" : "American ", "grades" : [{ "date" : { "$date" : 1401235200000 }, "grade" : "A", "score" : 11 }, { "date" : { "$date" : 1371600000000 }, "grade" : "A", "score" : 4 }, { "date" : { "$date" : 1339718400000 }, "grade" : "A", "score" : 3 }], "name" : "Wild Asia", "restaurant_id" : "40357217" }
{ "_id" : ObjectId("6331268eca37aeff25a2e144"), "address" : { "building" : "469", "coord" : [-73.961704, 40.662942], "street" : "Flatbush Avenue", "zipcode" : "11225" }, "borough" : "Brooklyn", "cuisine" : "Hamburgers", "grades" : [{ "date" : { "$date" : 1419897600000 }, "grade" : "A", "score" : 8 }, { "date" : { "$date" : 1404172800000 }, "grade" : "B", "score" : 23 }, { "date" : { "$date" : 1367280000000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1336435200000 }, "grade" : "A", "score" : 12 }], "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "_id" : ObjectId("6331268eca37aeff25a2e147"), "address" : { "building" : "97-22", "coord" : [-73.8601152, 40.7311739], "street" : "63 Road", "zipcode" : "11374" }, "borough" : "Queens", "cuisine" : "Jewish/Kosher", "grades" : [{ "date" : { "$date" : 1416787200000 }, "grade" : "Z", "score" : 20 }, { "date" : { "$date" : 1358380800000 }, "grade" : "A", "score" : 13 }, { "date" : { "$date" : 1343865600000 }, "grade" : "A", "score" : 13 }, { "date" : { "$date" : 1323907200000 }, "grade" : "B", "score" : 25 }], "name" : "Tov Kosher Kitchen", "restaurant_id" : "40356068" }
{ "_id" : ObjectId("6331268eca37aeff25a2e14c"), "address" : { "building" : "1839", "coord" : [-73.9482609, 40.6408271], "street" : "Nostrand Avenue", "zipcode" : "11226" }, "borough" : "Brooklyn", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "grades" : [{ "date" : { "$date" : 1405296000000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1373414400000 }, "grade" : "A", "score" : 8 }, { "date" : { "$date" : 1341964800000 }, "grade" : "A", "score" : 5 }, { "date" : { "$date" : 1329955200000 }, "grade" : "A", "score" : 8 }], "name" : "Taste The Tropics Ice Cream", "restaurant_id" : "40356731" }
{ "_id" : ObjectId("6331268eca37aeff25a2e146"), "address" : { "building" : "2780", "coord" : [-73.98241999999999, 40.579505], "street" : "Stillwell Avenue", "zipcode" : "11224" }, "borough" : "Brooklyn", "cuisine" : "American ",

"grades" : [{ "date" : { "$date" : 1402358400000 }, "grade" : "A", "score" : 5 }, { "date" : { "$date" : 1370390400000 }, "grade" : "A", "score" : 7 }, { "date" : { "$date" : 1334275200000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1318377600000 }, "grade" : "A", "score" : 12 }], "name" : "Riviera Caterer", "restaurant_id" : "40356018" }
{ "_id" : ObjectId("6331268eca37aeff25a2e14b"), "address" : { "building" : "6409", "coord" : [-74.00528899999999, 40.628886], "street" : "11 Avenue", "zipcode" : "11219" }, "borough" : "Brooklyn", "cuisine" : "American ", "grades" : [{ "date" : { "$date" : 1405641600000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1375142400000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1360713600000 }, "grade" : "A", "score" : 11 }, { "date" : { "$date" : 1345075200000 }, "grade" : "A", "score" : 2 }, { "date" : { "$date" : 1313539200000 }, "grade" : "A", "score" : 11 }], "name" : "Regina Caterers", "restaurant_id" : "40356649" }
{ "_id" : ObjectId("6331268eca37aeff25a2e143"), "address" : { "building" : "1007", "coord" : [-73.856077, 40.848447], "street" : "Morris Park Ave", "zipcode" : "10462" }, "borough" : "Bronx", "cuisine" : "Bakery", "grades" : [{ "date" : { "$date" : 1393804800000 }, "grade" : "A", "score" : 2 }, { "date" : { "$date" : 1378857600000 }, "grade" : "A", "score" : 6 }, { "date" : { "$date" : 1358985600000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1322006400000 }, "grade" : "A", "score" : 9 }, { "date" : { "$date" : 1299715200000 }, "grade" : "B", "score" : 14 }], "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6331268eca37aeff25a2e149"), "address" : { "building" : "2206", "coord" : [-74.1377286, 40.6119572], "street" : "Victory Boulevard", "zipcode" : "10314" }, "borough" : "Staten Island", "cuisine" : "Jewish/Kosher", "grades" : [{ "date" : { "$date" : 1412553600000 }, "grade" : "A", "score" : 9 }, { "date" : { "$date" : 1400544000000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1365033600000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1327363200000 }, "grade" : "A", "score" : 9 }], "name" : "Kosher Island", "restaurant_id" : "40356442" }
{ "_id" : ObjectId("6331239bca37aeff25a2e142"), "address" : { "building" : "1007", "street" : "HighStreet", "zipcode" : "TW31HA" }, "borough" : "Hounslow", "cuisine" : "Bakery", "Orders" : [{ "date" : { "$date" : 1393804800000 }, "rating" : "A", "stars" : 2 }, { "date" : { "$date" : 1378857600000 }, "rating" : "A", "stars" : 6 }, { "date" : { "$date" : 1358985600000 }, "rating" : "A", "stars" : 10 }, { "date" : { "$date" : 1322006400000 }, "rating" : "C", "stars" : 9 }, { "date" : { "$date" : 1299715200000 }, "rating" : "B", "stars" : 14 }], "name" : "Just Bake Shop", "restaurant_id" : "5075445" }

{ "_id" : ObjectId("6331268eca37aeff25a2e145"), "address" : { "building" : "351", "coord" : [-73.98513559999999, 40.7676919], "street" : "West   57 Street", "zipcode" : "10019" }, "borough" : "Manhattan", "cuisine" : "Irish", "grades" : [{ "date" : { "$date" : 1409961600000 }, "grade" : "A", "score" : 2 }, { "date" : { "$date" : 1374451200000 }, "grade" : "A", "score" : 11 }, { "date" : { "$date" : 1343692800000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1325116800000 }, "grade" : "A", "score" : 12 }], "name" : "Dj Reynolds Pub And Restaurant", "restaurant_id" : "30191841" }
{ "_id" : ObjectId("6331268eca37aeff25a2e148"), "address" : { "building" : "8825", "coord" : [-73.8803827, 40.7643124], "street" : "Astoria Boulevard", "zipcode" : "11369" }, "borough" : "Queens", "cuisine" : "American ", "grades" : [{ "date" : { "$date" : 1416009600000 }, "grade" : "Z", "score" : 38 }, { "date" : { "$date" : 1398988800000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1362182400000 }, "grade" : "A", "score" : 7 }, { "date" : { "$date" : 1328832000000 }, "grade" : "A", "score" : 13 }], "name" : "Brunos On The Boulevard", "restaurant_id" : "40356151" }

27. Write a MongoDB query to arranged the name of the cuisine in ascending order and for that same cuisine borough should be in descending order.on

```
db.restaurants.find().sort({"cuisine":1,"borough" : -1,} );
```

{ "_id" : ObjectId("6331239bca37aeff25a2e142"), "address" : { "building" : "1007", "street" : "HighStreet", "zipcode" : "TW31HA" }, "borough" : "Hounslow", "cuisine" : "Bakery", "Orders" : [{ "date" : { "$date" : 1393804800000 }, "rating" : "A", "stars" : 2 }, { "date" : { "$date" : 1378857600000 }, "rating" : "A", "stars" : 6 }, { "date" : { "$date" : 1358985600000 }, "rating" : "A", "stars" : 10 }, { "date" : { "$date" : 1322006400000 }, "rating" : "C", "stars" : 9 }, { "date" : { "$date" : 1299715200000 }, "rating" : "B", "stars" : 14 }], "name" : "Just Bake Shop", "restaurant_id" : "5075445" }
{ "_id" : ObjectId("6331268eca37aeff25a2e143"), "address" : { "building" : "1007", "coord" : [-73.856077, 40.848447], "street" : "Morris Park Ave", "zipcode" : "10462" }, "borough" : "Bronx", "cuisine" : "Bakery", "grades" : [{ "date" : { "$date" : 1393804800000 }, "grade" : "A", "score" : 2 }, { "date" : { "$date" : 1378857600000 }, "grade" : "A", "score" : 6 }, { "date" : { "$date" : 1358985600000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" :

1322006400000 }, "grade" : "A", "score" : 9 }, { "date" : { "$date" : 1299715200000 }, "grade" : "B", "score" : 14 }], "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6331268eca37aeff25a2e144"), "address" : { "building" : "469", "coord" : [-73.961704, 40.662942], "street" : "Flatbush Avenue", "zipcode" : "11225" }, "borough" : "Brooklyn", "cuisine" : "Hamburgers", "grades" : [{ "date" : { "$date" : 1419897600000 }, "grade" : "A", "score" : 8 }, { "date" : { "$date" : 1404172800000 }, "grade" : "B", "score" : 23 }, { "date" : { "$date" : 1367280000000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1336435200000 }, "grade" : "A", "score" : 12 }], "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "_id" : ObjectId("6331268eca37aeff25a2e145"), "address" : { "building" : "351", "coord" : [-73.98513559999999, 40.7676919], "street" : "West   57 Street", "zipcode" : "10019" }, "borough" : "Manhattan", "cuisine" : "Irish", "grades" : [{ "date" : { "$date" : 1409961600000 }, "grade" : "A", "score" : 2 }, { "date" : { "$date" : 1374451200000 }, "grade" : "A", "score" : 11 }, { "date" : { "$date" : 1343692800000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1325116800000 }, "grade" : "A", "score" : 12 }], "name" : "Dj Reynolds Pub And Restaurant", "restaurant_id" : "30191841" }
{ "_id" : ObjectId("6331268eca37aeff25a2e146"), "address" : { "building" : "2780", "coord" : [-73.98241999999999, 40.579505], "street" : "Stillwell Avenue", "zipcode" : "11224" }, "borough" : "Brooklyn", "cuisine" : "American ", "grades" : [{ "date" : { "$date" : 1402358400000 }, "grade" : "A", "score" : 5 }, { "date" : { "$date" : 1370390400000 }, "grade" : "A", "score" : 7 }, { "date" : { "$date" : 1334275200000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1318377600000 }, "grade" : "A", "score" : 12 }], "name" : "Riviera Caterer", "restaurant_id" : "40356018" }
{ "_id" : ObjectId("6331268eca37aeff25a2e147"), "address" : { "building" : "97-22", "coord" : [-73.8601152, 40.7311739], "street" : "63 Road", "zipcode" : "11374" }, "borough" : "Queens", "cuisine" : "Jewish/Kosher", "grades" : [{ "date" : { "$date" : 1416787200000 }, "grade" : "Z", "score" : 20 }, { "date" : { "$date" : 1358380800000 }, "grade" : "A", "score" : 13 }, { "date" : { "$date" : 1343865600000 }, "grade" : "A", "score" : 13 }, { "date" : { "$date" : 1323907200000 }, "grade" : "B", "score" : 25 }], "name" : "Tov Kosher Kitchen", "restaurant_id" : "40356068" }
{ "_id" : ObjectId("6331268eca37aeff25a2e148"), "address" : { "building" : "8825", "coord" : [-73.8803827, 40.7643124], "street" : "Astoria Boulevard", "zipcode" : "11369" }, "borough" : "Queens", "cuisine" : "American ", "grades" : [{ "date" : { "$date" : 1416009600000 }, "grade" : "Z", "score" : 38 }, { "date" : { "$date" : 1398988800000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" :

1362182400000 }, "grade" : "A", "score" : 7 }, { "date" : { "$date" : 1328832000000 }, "grade" : "A", "score" : 13 }], "name" : "Brunos On The Boulevard", "restaurant_id" : "40356151" }
{ "_id" : ObjectId("6331268eca37aeff25a2e149"), "address" : { "building" : "2206", "coord" : [-74.1377286, 40.6119572], "street" : "Victory Boulevard", "zipcode" : "10314" }, "borough" : "Staten Island", "cuisine" : "Jewish/Kosher", "grades" : [{ "date" : { "$date" : 1412553600000 }, "grade" : "A", "score" : 9 }, { "date" : { "$date" : 1400544000000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1365033600000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1327363200000 }, "grade" : "A", "score" : 9 }], "name" : "Kosher Island", "restaurant_id" : "40356442" }
{ "_id" : ObjectId("6331268eca37aeff25a2e14a"), "address" : { "building" : "7114", "coord" : [-73.9068506, 40.6199034], "street" : "Avenue U", "zipcode" : "11234" }, "borough" : "Brooklyn", "cuisine" : "Delicatessen", "grades" : [{ "date" : { "$date" : 1401321600000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1389657600000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1375488000000 }, "grade" : "A", "score" : 8 }, { "date" : { "$date" : 1342569600000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1331251200000 }, "grade" : "A", "score" : 13 }, { "date" : { "$date" : 1318550400000 }, "grade" : "A", "score" : 9 }], "name" : "Wilken'S Fine Food", "restaurant_id" : "40356483" }

28. Write a MongoDB query to know whether all the addresses contains the street or not.

```
db.restaurants.find({"address.street" : { $exists : true } }  );
```

{ "_id" : ObjectId("6331239bca37aeff25a2e142"), "address" : { "building" : "1007", "street" : "HighStreet", "zipcode" : "TW31HA" }, "borough" : "Hounslow", "cuisine" : "Bakery", "Orders" : [{ "date" : { "$date" : 1393804800000 }, "rating" : "A", "stars" : 2 }, { "date" : { "$date" : 1378857600000 }, "rating" : "A", "stars" : 6 }, { "date" : { "$date" : 1358985600000 }, "rating" : "A", "stars" : 10 }, { "date" : { "$date" : 1322006400000 }, "rating" : "C", "stars" : 9 }, { "date" : { "$date" : 1299715200000 }, "rating" : "B", "stars" : 14 }], "name" : "Just Bake Shop", "restaurant_id" : "5075445" }
{ "_id" : ObjectId("6331268eca37aeff25a2e143"), "address" : { "building" : "1007", "coord" : [-73.856077, 40.848447], "street" : "Morris Park Ave",

"zipcode" : "10462" }, "borough" : "Bronx", "cuisine" : "Bakery", "grades" : [{ "date" : { "$date" : 1393804800000 }, "grade" : "A", "score" : 2 }, { "date" : { "$date" : 1378857600000 }, "grade" : "A", "score" : 6 }, { "date" : { "$date" : 1358985600000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1322006400000 }, "grade" : "A", "score" : 9 }, { "date" : { "$date" : 1299715200000 }, "grade" : "B", "score" : 14 }], "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6331268eca37aeff25a2e144"), "address" : { "building" : "469", "coord" : [-73.961704, 40.662942], "street" : "Flatbush Avenue", "zipcode" : "11225" }, "borough" : "Brooklyn", "cuisine" : "Hamburgers", "grades" : [{ "date" : { "$date" : 1419897600000 }, "grade" : "A", "score" : 8 }, { "date" : { "$date" : 1404172800000 }, "grade" : "B", "score" : 23 }, { "date" : { "$date" : 1367280000000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1336435200000 }, "grade" : "A", "score" : 12 }], "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "_id" : ObjectId("6331268eca37aeff25a2e145"), "address" : { "building" : "351", "coord" : [-73.98513559999999, 40.7676919], "street" : "West   57 Street", "zipcode" : "10019" }, "borough" : "Manhattan", "cuisine" : "Irish", "grades" : [{ "date" : { "$date" : 1409961600000 }, "grade" : "A", "score" : 2 }, { "date" : { "$date" : 1374451200000 }, "grade" : "A", "score" : 11 }, { "date" : { "$date" : 1343692800000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1325116800000 }, "grade" : "A", "score" : 12 }], "name" : "Dj Reynolds Pub And Restaurant", "restaurant_id" : "30191841" }
{ "_id" : ObjectId("6331268eca37aeff25a2e146"), "address" : { "building" : "2780", "coord" : [-73.98241999999999, 40.579505], "street" : "Stillwell Avenue", "zipcode" : "11224" }, "borough" : "Brooklyn", "cuisine" : "American ", "grades" : [{ "date" : { "$date" : 1402358400000 }, "grade" : "A", "score" : 5 }, { "date" : { "$date" : 1370390400000 }, "grade" : "A", "score" : 7 }, { "date" : { "$date" : 1334275200000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1318377600000 }, "grade" : "A", "score" : 12 }], "name" : "Riviera Caterer", "restaurant_id" : "40356018" }
{ "_id" : ObjectId("6331268eca37aeff25a2e147"), "address" : { "building" : "97-22", "coord" : [-73.8601152, 40.7311739], "street" : "63 Road", "zipcode" : "11374" }, "borough" : "Queens", "cuisine" : "Jewish/Kosher", "grades" : [{ "date" : { "$date" : 1416787200000 }, "grade" : "Z", "score" : 20 }, { "date" : { "$date" : 1358380800000 }, "grade" : "A", "score" : 13 }, { "date" : { "$date" : 1343865600000 }, "grade" : "A", "score" : 13 }, { "date" : { "$date" : 1323907200000 }, "grade" : "B", "score" : 25 }], "name" : "Tov Kosher Kitchen", "restaurant_id" : "40356068" }

{ "_id" : ObjectId("6331268eca37aeff25a2e148"), "address" : { "building" : "8825", "coord" : [-73.8803827, 40.7643124], "street" : "Astoria Boulevard", "zipcode" : "11369" }, "borough" : "Queens", "cuisine" : "American ", "grades" : [{ "date" : { "$date" : 1416009600000 }, "grade" : "Z", "score" : 38 }, { "date" : { "$date" : 1398988800000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1362182400000 }, "grade" : "A", "score" : 7 }, { "date" : { "$date" : 1328832000000 }, "grade" : "A", "score" : 13 }], "name" : "Brunos On The Boulevard", "restaurant_id" : "40356151" }
{ "_id" : ObjectId("6331268eca37aeff25a2e149"), "address" : { "building" : "2206", "coord" : [-74.1377286, 40.6119572], "street" : "Victory Boulevard", "zipcode" : "10314" }, "borough" : "Staten Island", "cuisine" : "Jewish/Kosher", "grades" : [{ "date" : { "$date" : 1412553600000 }, "grade" : "A", "score" : 9 }, { "date" : { "$date" : 1400544000000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1365033600000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1327363200000 }, "grade" : "A", "score" : 9 }], "name" : "Kosher Island", "restaurant_id" : "40356442" }
{ "_id" : ObjectId("6331268eca37aeff25a2e14a"), "address" : { "building" : "7114", "coord" : [-73.9068506, 40.6199034], "street" : "Avenue U", "zipcode" : "11234" }, "borough" : "Brooklyn", "cuisine" : "Delicatessen", "grades" : [{ "date" : { "$date" : 1401321600000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1389657600000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1375488000000 }, "grade" : "A", "score" : 8 }, { "date" : { "$date" : 1342569600000 }, "grade" : "A", "score" : 10 }, { "date" : { "$date" : 1331251200000 }, "grade" : "A", "score" : 13 }, { "date" : { "$date" : 1318550400000 }, "grade" : "A", "score" : 9 }], "name" : "Wilken'S Fine Food", "restaurant_id" : "40356483" }
{ "_id" : ObjectId("6331268eca37aeff25a2e14b"), "address" : { "building" : "6409", "coord" : [-74.00528899999999, 40.628886], "street" : "11 Avenue", "zipcode" : "11219" }, "borough" : "Brooklyn", "cuisine" : "American ", "grades" : [{ "date" : { "$date" : 1405641600000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1375142400000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1360713600000 }, "grade" : "A", "score" : 11 }, { "date" : { "$date" : 1345075200000 }, "grade" : "A", "score" : 2 }, { "date" : { "$date" : 1313539200000 }, "grade" : "A", "score" : 11 }], "name" : "Regina Caterers", "restaurant_id" : "40356649" }
{ "_id" : ObjectId("6331268eca37aeff25a2e14c"), "address" : { "building" : "1839", "coord" : [-73.9482609, 40.6408271], "street" : "Nostrand Avenue", "zipcode" : "11226" }, "borough" : "Brooklyn", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "grades" : [{ "date" : { "$date" : 1405296000000 }, "grade" : "A", "score" : 12 }, { "date" : { "$date" : 1373414400000 }, "grade" : "A", "score" : 8

}, { "date" : { "$date" : 1341964800000 }, "grade" : "A", "score" : 5 }, { "date" : { "$date" : 1329955200000 }, "grade" : "A", "score" : 8 }], "name" : "Taste The Tropics Ice Cream", "restaurant_id" : "40356731" }
{ "_id" : ObjectId("6331268eca37aeff25a2e14d"), "address" : { "building" : "2300", "coord" : [-73.8786113, 40.8502883], "street" : "Southern Boulevard", "zipcode" : "10460" }, "borough" : "Bronx", "cuisine" : "American ", "grades" : [{ "date" : { "$date" : 1401235200000 }, "grade" : "A", "score" : 11 }, { "date" : { "$date" : 1371600000000 }, "grade" : "A", "score" : 4 }, { "date" : { "$date" : 1339718400000 }, "grade" : "A", "score" : 3 }], "name" : "Wild Asia", "restaurant_id" : "40357217" }

29. Write a MongoDB query which will select all documents in the restaurants collection where the coord field value is Double.

db.restaurants.find({"address.coord": {$type: "double"}}, {_id:0, address:1})

{ "address" : { "building" : "1007", "coord" : [-73.856077, 40.848447], "street" : "Morris Park Ave", "zipcode" : "10462" } }
{ "address" : { "building" : "469", "coord" : [-73.961704, 40.662942], "street" : "Flatbush Avenue", "zipcode" : "11225" } }
{ "address" : { "building" : "351", "coord" : [-73.98513559999999, 40.7676919], "street" : "West   57 Street", "zipcode" : "10019" } }
{ "address" : { "building" : "2780", "coord" : [-73.98241999999999, 40.579505], "street" : "Stillwell Avenue", "zipcode" : "11224" } }
{ "address" : { "building" : "97-22", "coord" : [-73.8601152, 40.7311739], "street" : "63 Road", "zipcode" : "11374" } }
{ "address" : { "building" : "8825", "coord" : [-73.8803827, 40.7643124], "street" : "Astoria Boulevard", "zipcode" : "11369" } }
{ "address" : { "building" : "2206", "coord" : [-74.1377286, 40.6119572], "street" : "Victory Boulevard", "zipcode" : "10314" } }
{ "address" : { "building" : "7114", "coord" : [-73.9068506, 40.6199034], "street" : "Avenue U", "zipcode" : "11234" } }
{ "address" : { "building" : "6409", "coord" : [-74.00528899999999, 40.628886], "street" : "11 Avenue", "zipcode" : "11219" } }
{ "address" : { "building" : "1839", "coord" : [-73.9482609, 40.6408271], "street" : "Nostrand Avenue", "zipcode" : "11226" } }
{ "address" : { "building" : "2300", "coord" : [-73.8786113, 40.8502883], "street" : "Southern Boulevard", "zipcode" : "10460" } }

30. Write a MongoDB query which will select the restaurant Id, name and grades for those restaurants which returns 0 as a remainder after dividing the score by 7.

db.restaurant.find({"grades": {$elemMatch: {"score": {$mod: [7,0]}}}},{_id:0, restaurant_id:1, name:1, grades:1})

31. Write a MongoDB query to find the restaurant name, borough, longitude and attitude and cuisine for those restaurants which contains 'mon' as three letters somewhere in its name.

db.restaurant.find({name: {$regex: /mon/}},{_id:0, name:1, borough:1, "address.coord":1, cuisine:1})

32. Write a MongoDB query to find the restaurant name, borough, longitude and latitude and cuisine for those restaurants which contain 'Mad' as first three letters of its name.

db.rest.find({name: {$regex: /^Mad.*/}},{_id:0, name:1, borough:1, "address.coord":1, cuisine:1})