

SHIV PRATAP SINGH PARMAR (Assignment-7)

1. 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});
```

```
> db.restaurants.find({},{"restaurant_id" : 1,"name":1,"borough":1,"cuisine" :1,"_id":0});
{ "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" : "Jewish/Kosher", "name" : "Tov Kosher Kitchen", "restaurant_id" : "40356068" }
{ "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" }
{ "borough" : "Brooklyn", "cuisine" : "American ", "name" : "C & C Catering Service", "restaurant_id" : "40357437" }
{ "borough" : "Brooklyn", "cuisine" : "Chinese", "name" : "May May Kitchen", "restaurant_id" : "40358429" }
{ "borough" : "Manhattan", "cuisine" : "American ", "name" : "1 East 66Th Street Kitchen", "restaurant_id" : "40359480" }
{ "borough" : "Brooklyn", "cuisine" : "Jewish/Kosher", "name" : "Seuda Foods", "restaurant_id" : "40360045" }
{ "borough" : "Brooklyn", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Carvel Ice Cream", "restaurant_id" : "40360076" }
{ "borough" : "Queens", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Carvel Ice Cream", "restaurant_id" : "40361322" }
{ "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Nordic Delicacies", "restaurant_id" : "40361390" }
{ "borough" : "Manhattan", "cuisine" : "American ", "name" : "Glorious Food", "restaurant_id" : "40361521" }
{ "borough" : "Brooklyn", "cuisine" : "American ", "name" : "The Movable Feast", "restaurant_id" : "40361606" }
Type "it" for more
```

2. Write a MongoDB query to display all the restaurant which is in the borough Bronx

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

```
> db.restaurants.find({"borough": "Bronx"}).pretty();
{
  "_id" : ObjectId("5947bc1fb60c37c179375c3f"),
  "address" : {
    "building" : "1007",
    "coord" : [
      -73.856077,
      40.848447
    ],
    "street" : "Morris Park Ave",
    "zipcode" : "10462"
  },
  "borough" : "Bronx",
  "cuisine" : "Bakery",
  "grades" : [
    {
      "date" : ISODate("2014-03-03T00:00:00Z"),
      "grade" : "A",
      "score" : 2
    },
    {
      "date" : ISODate("2013-09-11T00:00:00Z"),
      "grade" : "A",
      "score" : 6
    },
    {
      "date" : ISODate("2013-01-24T00:00:00Z"),
      "grade" : "A",
      "score" : 10
    },
    {
      "date" : ISODate("2011-11-23T00:00:00Z"),
      "grade" : "A",
      "score" : 9
    },
    {
      "date" : ISODate("2011-03-10T00:00:00Z"),
      "grade" : "B",
      "score" : 14
    }
  ],
  "name" : "Morris Park Bake Shop",
  "restaurant_id" : "30075445"
}
{
  "_id" : ObjectId("5947bc1fb60c37c179375c49"),
  "address" : {
    "building" : "2300",
    "coord" : [
      -73.8786113,
      40.8502883
    ],
    "street" : "Southern Boulevard",
    "zipcode" : "10460"
  },
  "borough" : "Bronx",
  "cuisine" : "American ",
  "grades" : [
    {
      "date" : ISODate("2014-05-28T00:00:00Z"),

```

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

db.restaurants.find({ grades : { \$elemMatch : { "score":{ \$gt : 90 } } } });

```
> db.restaurants.find( { grades : { $elemMatch : { "score":{ $gt : 90 } } } } );
{
  "_id" : ObjectId("5947bc1fb60c37c179375d9d"),
  "address" : {
    "building" : "65",
    "coord" : [
      -73.9782725,
      40.7624022
    ],
    "street" : "West 54 Street",
    "zipcode" : "10019"
  },
  "borough" : "Manhattan",
  "cuisine" : "American ",
  "grades" : [
    {
      "date" : ISODate("2014-08-22T00:00:00Z"),
      "grade" : "A",
      "score" : 11
    },
    {
      "date" : ISODate("2014-03-28T00:00:00Z"),
      "grade" : "C",
      "score" : 131
    },
    {
      "date" : ISODate("2013-09-25T00:00:00Z"),
      "grade" : "A",
      "score" : 11
    },
    {
      "date" : ISODate("2013-04-08T00:00:00Z"),
      "grade" : "B",
      "score" : 25
    },
    {
      "date" : ISODate("2012-10-15T00:00:00Z"),
      "grade" : "A",
      "score" : 11
    },
    {
      "date" : ISODate("2011-10-19T00:00:00Z"),
      "grade" : "A",
      "score" : 13
    }
  ],
  "name" : "Murals On 54/Randolphs'S",
  "restaurant_id" : "40372466"
}
{
  "_id" : ObjectId("5947bc1fb60c37c179375e3e"),
  "address" : {
    "building" : "345",
    "coord" : [
      -73.9864626,
      40.7266739
    ],
    "street" : "East 6 Street",
    "zipcode" : "10003"
  },
  "borough" : "Manhattan",
  "cuisine" : "American ",
  "grades" : [
    {
      "date" : ISODate("2014-08-22T00:00:00Z"),
      "grade" : "A",
      "score" : 11
    },
    {
      "date" : ISODate("2014-03-28T00:00:00Z"),
      "grade" : "C",
      "score" : 131
    },
    {
      "date" : ISODate("2013-09-25T00:00:00Z"),
      "grade" : "A",
      "score" : 11
    },
    {
      "date" : ISODate("2013-04-08T00:00:00Z"),
      "grade" : "B",
      "score" : 25
    },
    {
      "date" : ISODate("2012-10-15T00:00:00Z"),
      "grade" : "A",
      "score" : 11
    },
    {
      "date" : ISODate("2011-10-19T00:00:00Z"),
      "grade" : "A",
      "score" : 13
    }
  ],
  "name" : "Murals On 54/Randolphs'S",
  "restaurant_id" : "40372466"
}
```

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

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

```
> db.restaurants.find({"borough": "Bronx"}).limit(5).pretty();
{
  "_id" : ObjectId("5947bc1fb60c37c179375c3f"),
  "address" : {
    "building" : "1007",
    "coord" : [
      -73.856077,
      40.848447
    ],
    "street" : "Morris Park Ave",
    "zipcode" : "10462"
  },
  "borough" : "Bronx",
  "cuisine" : "Bakery",
  "grades" : [
    {
      "date" : ISODate("2014-03-03T00:00:00Z"),
      "grade" : "A",
      "score" : 2
    },
    {
      "date" : ISODate("2013-09-11T00:00:00Z"),
      "grade" : "A",
      "score" : 6
    },
    {
      "date" : ISODate("2013-01-24T00:00:00Z"),
      "grade" : "A",
      "score" : 10
    },
    {
      "date" : ISODate("2011-11-23T00:00:00Z"),
      "grade" : "A",
      "score" : 9
    },
    {
      "date" : ISODate("2011-03-10T00:00:00Z"),
      "grade" : "B",
      "score" : 14
    }
  ],
  "name" : "Morris Park Bake Shop",
  "restaurant_id" : "30075445"
}
{
  "_id" : ObjectId("5947bc1fb60c37c179375c49"),
  "address" : {
    "building" : "2300",
    "coord" : [
      -73.8786113,
      40.8502883
    ],
    "street" : "Southern Boulevard",
    "zipcode" : "10460"
  },
  "borough" : "Bronx",
```

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

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

```
> db.restaurants.find( {"borough": "Bronx"} ).skip(5).limit(5).pretty();
{
  "_id" : ObjectId("5947bc1fb60c37c179375c7c"),
  "address" : {
    "building" : "658",
    "coord" : [
      -73.81363999999999,
      40.82941100000001
    ],
    "street" : "Clarence Ave",
    "zipcode" : "10465"
  },
  "borough" : "Bronx",
  "cuisine" : "American ",
  "grades" : [
    {
      "date" : ISODate("2014-06-21T00:00:00Z"),
      "grade" : "A",
      "score" : 5
    },
    {
      "date" : ISODate("2012-07-11T00:00:00Z"),
      "grade" : "A",
      "score" : 10
    }
  ],
  "name" : "Manhem Club",
  "restaurant_id" : "40364363"
}
{
  "_id" : ObjectId("5947bc1fb60c37c179375c94"),
  "address" : {
    "building" : "2222",
    "coord" : [
      -73.84971759999999,
      40.8304811
    ],
    "street" : "Haviland Avenue",
    "zipcode" : "10462"
  },
  "borough" : "Bronx",
  "cuisine" : "American ",
  "grades" : [
    {
      "date" : ISODate("2014-12-18T00:00:00Z"),
      "grade" : "A",
      "score" : 7
    },
    {
      "date" : ISODate("2014-05-01T00:00:00Z"),
      "grade" : "B",
      "score" : 17
    },
    {
      "date" : ISODate("2013-03-14T00:00:00Z"),
      "grade" : "A",
      "score" : 12
    },
    {
      "date" : ISODate("2012-09-20T00:00:00Z"),
      "grade" : "A",
      "score" : 8
    }
  ]
}
```

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


```
db.restaurants.find({grades : { $elemMatch:{ "score" : { $gt : 80 , $lt :100} } } });
```

```
db.restaurants.find({grades : { $elemMatch:{ "score" : { $gt : 80 , $lt :100} } } })
{ "_id" : ObjectId("5947bclfb60c37c179376288"), "address" : { "building" : "3707", "coord" : [ -73.984626, 40.726739 ], "street" : "East 6 Street", "zipcode" : "10003" }, "borough" : "Manhattan", "cuisine" : "Indian", "grades" : [ { "date" : ISODate("2014-09-15T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2014-01-14T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2013-05-30T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-04-24T00:00:00Z"), "grade" : "P", "score" : 2 }, { "date" : ISODate("2012-10-01T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2012-04-06T00:00:00Z"), "grade" : "C", "score" : 92 }, { "date" : ISODate("2011-11-03T00:00:00Z"), "grade" : "C", "score" : 41 } ], "name" : "Gandhi", "restaurant_id" : "40301295" },
{ "_id" : ObjectId("5947bclfb60c37c179375fal"), "address" : { "building" : "130", "coord" : [ -73.984758, 40.7457939 ], "street" : "Madison Avenue", "zipcode" : "10016" }, "borough" : "Manhattan", "cuisine" : "Pizza/Italian", "grades" : [ { "date" : ISODate("2014-12-24T00:00:00Z"), "grade" : "Z", "score" : 31 }, { "date" : ISODate("2014-06-17T00:00:00Z"), "grade" : "C", "score" : 98 }, { "date" : ISODate("2013-12-12T00:00:00Z"), "grade" : "C", "score" : 32 }, { "date" : ISODate("2013-05-22T00:00:00Z"), "grade" : "B", "score" : 21 }, { "date" : ISODate("2012-05-02T00:00:00Z"), "grade" : "A", "score" : 11 } ], "name" : "Bella Napoli", "restaurant_id" : "40393488" },
{ "_id" : ObjectId("5947bclfb60c37c179376810"), "address" : { "building" : "", "coord" : [ -74.0163793, 40.7167671 ], "street" : "Hudson River", "zipcode" : "10282" }, "borough" : "Manhattan", "cuisine" : "American", "grades" : [ { "date" : ISODate("2014-06-27T00:00:00Z"), "grade" : "C", "score" : 89 }, { "date" : ISODate("2013-06-06T00:00:00Z"), "grade" : "A", "score" : 6 }, { "date" : ISODate("2012-06-19T00:00:00Z"), "grade" : "A", "score" : 13 } ], "name" : "West 79th Street Boat Basin Cafe", "restaurant_id" : "40756344" }
```

7. Write a MongoDB query to find the restaurants which locate in latitude value less than -95.754168

```
db.restaurants.find( { "address.coord.0" : { $lt : -95.754168 } } );
```

```
> db.restaurants.find( { "address.coord.0" : { $lt : -95.754168 } }).pretty();
{
  "_id" : ObjectId("5947bclfb60c37c179376288"),
  "address" : {
    "building" : "3707",
    "coord" : [
      -101.8945214,
      33.5197474
    ],
    "street" : "82 Street",
    "zipcode" : "11372"
  },
  "borough" : "Queens",
  "cuisine" : "American ",
  "grades" : [
    {
      "date" : ISODate("2014-06-04T00:00:00Z"),
      "grade" : "A",
      "score" : 12
    },
    {
      "date" : ISODate("2013-11-07T00:00:00Z"),
      "grade" : "B",
      "score" : 19
    },
    {
      "date" : ISODate("2013-05-17T00:00:00Z"),
      "grade" : "A",
      "score" : 11
    },
    {
      "date" : ISODate("2012-08-29T00:00:00Z"),
      "grade" : "A",
      "score" : 11
    },
    {
      "date" : ISODate("2012-04-03T00:00:00Z"),
      "grade" : "A",
      "score" : 12
    },
    {
      "date" : ISODate("2011-11-16T00:00:00Z"),
      "grade" : "A",
      "score" : 7
    }
  ],
  "name" : "Burger King",
  "restaurant_id" : "40534067"
},
{
  "_id" : ObjectId("5947bclfb60c37c1793765f4"),
  "address" : {
    "building" : "15259",
    "coord" : [
      -119.6368672,
      36.2504996
    ],
    "street" : "10 Avenue",
    "zipcode" : "11357"
  },
  "borough" : "Queens",
  "cuisine" : "Italian",
  "grades" : [
    {
      "date" : ISODate("2014-06-04T00:00:00Z"),
```

8. 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

```
db.restaurants.find({$and: [ {"cuisine" : {$ne : "American " } }, {"grades.score" : {$gt : 70}}, {"address.coord.0" : {$lt : -65.754168}} ] });
```

```
db.restaurants.find({$and: [ {"cuisine" : {$ne : "American " } }, {"grades.score" : {$gt : 70}}, {"address.coord.0" : {$lt : -65.754168}} ] }).pretty();

{
  "_id" : ObjectId("5947bclfb60c37c179375e3e"),
  "address" : {
    "building" : "345",
    "coord" : [
      -73.9864626,
      40.7266739
    ],
    "street" : "East 6 Street",
    "zipcode" : "10003"
  },
  "borough" : "Manhattan",
  "cuisine" : "Indian",
  "grades" : [
    {
      "date" : ISODate("2014-09-15T00:00:00Z"),
      "grade" : "A",
      "score" : 5
    },
    {
      "date" : ISODate("2014-01-14T00:00:00Z"),
      "grade" : "A",
      "score" : 8
    },
    {
      "date" : ISODate("2013-05-30T00:00:00Z"),
      "grade" : "A",
      "score" : 12
    },
    {
      "date" : ISODate("2013-04-24T00:00:00Z"),
      "grade" : "P",
      "score" : 2
    },
    {
      "date" : ISODate("2012-10-01T00:00:00Z"),
      "grade" : "A",
      "score" : 9
    },
    {
      "date" : ISODate("2012-04-06T00:00:00Z"),
      "grade" : "C",
      "score" : 92
    },
    {
      "date" : ISODate("2011-11-03T00:00:00Z"),
      "grade" : "C",
      "score" : 41
    }
  ]
}
```

9. Write a MongoDB query to update the restaurant's grade to 'B' whose score is more than or equal to 10.

```
db.restaurants.update( { grades : { $elemMatch : { "score":{ $gte : 10 } } } }, { $set: { grades: {"grade": "B"} } }, { multi: true});
```

```
> db.restaurants.update({ { grades : { $elemMatch : { "score":{ $gte : 10 } } } }, { $set: { grades: {"grade": "B"} } });
2017-06-19T21:51:19.083+0530 E QUERY [thread1] SyntaxError: invalid property id @(shell):1:24
```

10. Write a MongoDB query to delete the restaurents in borough Bronx.

```
db.restaurants.remove( {"borough":"Bronx"} )
```

Create Google Doc and paste your solution queries there and also paste screenshots of each query and its result in that doc.

```
> db.restaurants.remove( {"borough":"Bronx"} );  
WriteResult({ "nRemoved" : 309 })  
> db.restaurants.find( {"borough":"Bronx"} );  
> |
```