Assignment No.12

Lab Exercise:-

1.Create Database Restaurants.

```
> use Restaurants
switched to db Restaurants
>
```

2.Create Collection REST1

```
> db.createCollection("REST1")
{ "ok" : 1 }
>
```

3.Insert documents

```
> db.REST1.insertOne({
...   "address": {
...     "building": "1007",
...     "coord": [ -73.856077, 40.848447 ],
...     "street": "Morris Park Ave",
...     "zipcode": "10462"
...   },
...   "borough": "Bronx",
...   "cuisine": "Bakery",
...   "grades": [
...     { "date": new Date(1393804800000), "grade": "A", "score": 2 },
...     { "date": new Date(1378857600000), "grade": "A", "score": 6 },
...     { "date": new Date(1358985600000), "grade": "A", "score": 10 },
...     { "date": new Date(1322006400000), "grade": "A", "score": 9 },
...     { "date": new Date(1299715200000), "grade": "B", "score": 14 }
...   ],
...   "name": "Morris Park Bake Shop",
...   "restaurant_id": "30075445"
... })
{
      "acknowledged" : true,
      "insertedId" : ObjectId("66ff72739ca237ba9c82073a")
}
```

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

```
> db.REST1.find().pretty()
{
      "_id" : ObjectId("66ff72739ca237ba9c82073a"),
      "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"
}
>
```


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


```
> db.REST1.find({}, { restaurant_id: 1, name: 1, borough: 1, cuisine: 1, _id: 0 }).pretty()
{
    "borough" : "Bronx",
    "cuisine" : "Bakery",
    "name" : "Morris Park Bake Shop",
    "restaurant_id" : "30075445"
}
>
```


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


```
> db.REST1.find({ borough: "Bronx" }).pretty()
{
    "_id" : ObjectId("66ff72739ca237ba9c82073a"),
    "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"
}
>
```


7. 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.

```
> db.REST1.find({}, { restaurant_id: 1, name: 1, borough: 1, "address.zipcode": 1, _id: 0 }).pretty()
{
    "address" : {
        "zipcode" : "10462"
    },
    "borough" : "Bronx",
    "name" : "Morris Park Bake Shop",
    "restaurant_id" : "30075445"
}
>
```

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

```
> db.REST1.find({ "grades.score": { $gt: 90 } }).pretty()
```

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

```
> db.REST1.find({ "grades.score": { $gt: 80, $lt: 100 } }).pretty()

>
```