MongoDB
ASSIGNMENT 3

Exercise Questions-

```
1. db.addresses.find().pretty()
2. db.addresses.find({},{"restaurant_id":1,"name":1,"borough":1,"cuisine":1}).pretty()
3. db.addresses.find({},{"restaurant_id":1,"name":1,"borough":1,"cuisine":1,_id:0}).pr
   etty()
4. db.addresses.find({},{"restaurant_id":1,"name":1,"borough":1,"zipcode":1,_id:0}).pretty()
5. db.addresses.find({"borough":"Bronx"}).limit(5).pretty()
6. db.addresses.find({"borough":"Bronx"}).pretty()
7. db.addresses.find({"borough":"Bronx"}).skip(5).limit(5).pretty()
8. db.addresses.find({"grades.score":{$gt:90}}).pretty()
9. db.addresses.find({$and:[{"grades.score":{$gt:80}},{"grades.score":{$lt:100}}]}).pretty()
10. db.addresses.find( {"address.coord":{$lt:-95.754168} } ).pretty()
11. db.addresses.find({$and:[{"cusine":{$ne:"American"}},{"grades.score":{$gt:70}},{"address
    .coord":{$lt:-65.754168}}]}).pretty();
12. db.addresses.find({$and:[{"cusine":{$ne:"American"}},{"grades.score":{$gt:70}},{"address
    .coord":{$lt:-65.754168}}]}).pretty();
13. db.addresses.find({$and:[{"cuisine":{$ne:"American"}},{"grades.grade":"A"},{"borough":{$
    ne:"Brooklyn"}}]}).sort({“cuisine”:-1}).pretty();
14. db.addresses.find( { name: { $regex: /^Wil/}
    },{name:1,borough:1,"address.coord":1}).pretty()
15. db.addresses.find( { name: { $regex: /ces$/}
    },{name:1,borough:1,"address.coord":1}).pretty()
16. db.addresses.find( { name: { $regex: /Reg/}
    },{name:1,borough:1,"address.coord":1}).pretty()
17. db.addresses.find({borough:"Bronx",$or:[{cuisine:"American"},{cuisine:"Chinese"}]}).prett
    y()
18. db.addresses.find({$or:[{borough:"Staten Island"},{borough:"Queens"},{borough:"Bronxor
    Brooklyn"}]}).pretty()
19.  db.addresses.find({$or:[{borough:{$ne:"Staten
    Island"}},{borough:{$ne:"Queens"}},{borough:{$ne:"Bronxor Brooklyn"}}]}).pretty()
20. db.addresses.find({"grades.score":{$lte:10}},{"restaurant_id":1,"name":1,"borough":1,"cui
    sine":1}).pretty()
21. db.addresses.find({$or:[{"cuisine":{$ne:"American"}},{"cuisine":{$ne:"Chinese"}},{"name":
    { $regex: /^Wil/} }]}).sort({"cuisine":-1}).pretty();
22. db.addresses.find({$and:[{"grades.date":ISODate("2014-08-11T00:00:00Z")},{"grades.gr
    ade":"A"},{"grades.score":11}]},{“restaurant_id”:1,”name”:1,”grades.grade”:1}).pretty()
23. db.restaurants.find(  { "grades.1.date": ISODate("2014-08-11T00:00:00Z"),
    "grades.1.grade":"A" ,  "grades.1.score" : 9}, {"restaurant_id":1,"name":1,"grades":} );
```

24. db.addresses.find({"address.coord.1":{$gt:42,$lte:52}},{"restaurant_id":1,"name":1,"address.coord":1}).pretty()
25. db.addresses.find().sort({name:1}).pretty()
26. db.addresses.find().sort({name:-1}).pretty()
27. db.addresses.find().sort({name:1,borough:-1}).pretty()
28. db.addresses.find({ "address.street": { $exists: false } }).pretty() finds all the documents in which street is not present
29. {"address.coord": {$type : 1}});
30. db.restaurants.find({"grades.score" : {$mod : [7,0] }, {"restaurant_id" : 1,"name":1,"grades":1})
31. db.addresses.find({name:/mon/},{name:1,borough:1,"address.coord":1}).pretty()
32. db.addresses.find({ name: { $regex: /^Mon/} },{name:1,borough:1,"address.coord":1}).pretty()