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.restaurants.find({$or: [{name:/^Wil/},{"$and":[{"cuisine":{$ne:"American"}},{"cuisine":{$ne :"chinees"}}]}]},{"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 ISODate "2014-08-11T00:00:00Z".

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":1});

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({"address.coord.1":{$gt:42,$lte:55}},{"restaurant\_id":1,"name":1,"address":1,"coord":1});

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

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

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})

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.

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

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

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

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:1}});

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.restaurants.find({"grades.score":{$mod:[7,0]}},{"restaurant\_id":1,"name":1,"grades":1});