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

db.restaurant.find( {} )

2. Display the fields restaurant\_id, name, borough and cuisine for all the documents in the collection restaurant.

db.restaurant.find({}, {restaurant\_id:1, name:1, cuisine:1, borough: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.restaurant.find({}, {restaurant\_id:1, \_id:0, name:1, cuisine:1, borough:1})

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.

db.restaurant.find({}, {restaurant\_id:1, \_id:0, name:1, "address.zipcode":1, borough:1})

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

db.restaurant.find({borough:"Bronx"})

6. Display the first 5 restaurant which is in the borough Brooklyn.

db.restaurant.find({borough:"Bronx"}).limit(5)

7. Display the next 5 restaurants after skipping first 5 which are in the borough Brooklyn.

db.restaurant.find({borough:"Bronx"}).limit(5).skip(5)

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

db.restaurant.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.restaurant.find({"grades.score": { $gt: 90 }, "grades.score": { $lt: 100}})

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

db.restaurant.find({"address.coord": { $lt: -95.754168}})

11. 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.restaurant.find({cuisine:{ $ne :"American "} ,"grades.score": { $gt: 70 },"address.coord": { $lt: -65.754168 }})

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 located in the longitude less than -65.754168.

db.restaurant.find({cuisine:{ $ne :"American "} ,"grades.score": { $gt: 70 },"address.coord": { $lt: [1 ,-65.754168] }})

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.restaurant.find({ cuisine:{ $ne :"American "}, borough:{ $ne :"Brooklyn"}, "grades.grade":"A" }).sort({borough: -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.restaurant.find({name:/^Wil/},{restaurant\_id: 1, borough: 1, name: 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.restaurant.find({name:/ces$/},{restaurant\_id: 1, borough: 1, name: 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.restaurant.find({name:/Reg/},{restaurant\_id: 1, borough: 1, name: 1, cuisine: 1})

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

db.restaurant.find( {borough: "Bronx", $or: [ { cuisine: "American "}, { cuisine: "Chinese"} ] } )

18. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which belong to the borough Staten Island or Queens or Bronxor Brooklyn

db.restaurant.find( {},{restaurant\_id: 1,cuisine: 1,borough: 1,name: 1,$or: [ { borough: "Bronx"}, { borough: "Brooklyn"}, { borough: "Queens"}, { borough: "Staten Island"}] } )

19. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which are not belonging to the borough Staten Island or Queens or Bronxor Brooklyn

db.restaurant.find({"borough": {$nin: ["Staten Island", "Queens", "Bronx", "Brooklyn"]}}, {"restuarant\_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.restaurant.find({"grades.score": { $gt: 70 }}, {"restuarant\_id": 1, "name": 1, "borough": 1, "cuisine": 1})