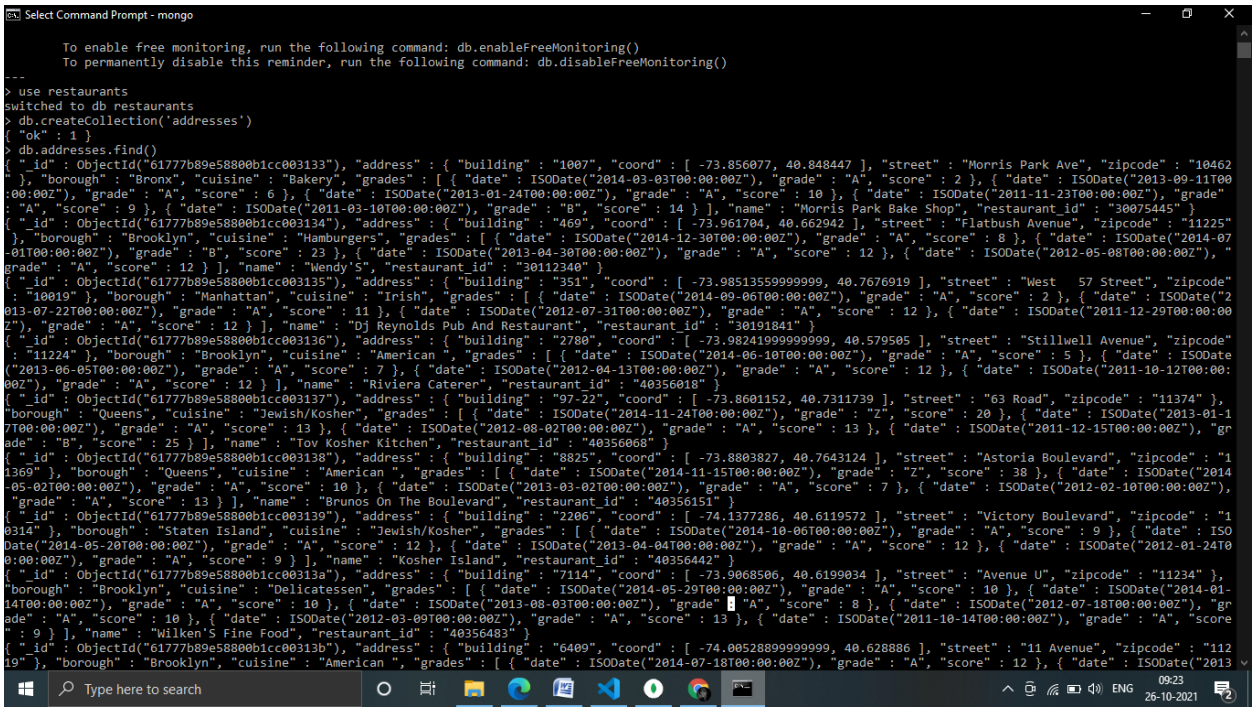
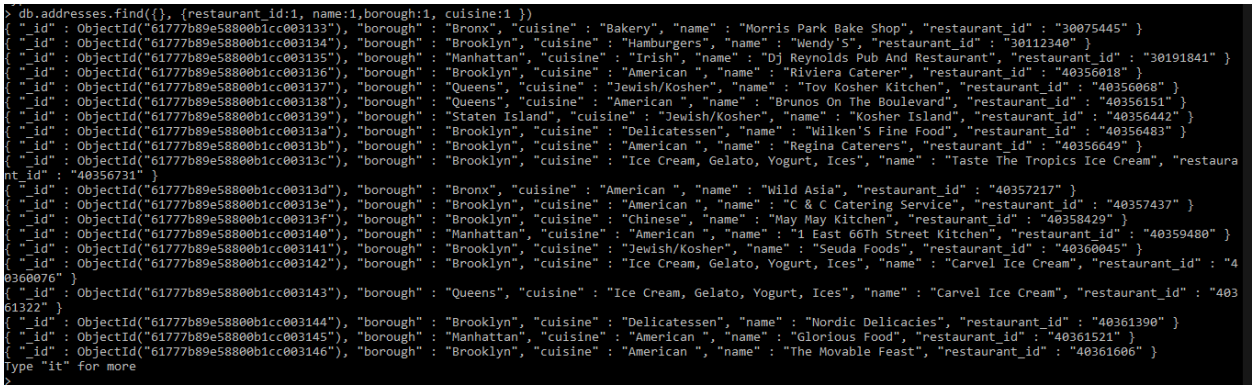
MongoDB – Complex Queries

ASSIGNMENT-3

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



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



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.addresses.find({}, {restaurant_id:1, _id:0, name:1,borough:1, cuisine:1 })e" : "Delicatessen", "name" : "Nordic Delicacies", "restaurant_id" : "40361390" }
{ "borough" : "Bronx", "cuisine" : "Bakery", "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }lorious Food", "restaurant_id" : "40361521" }
{ "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "Wendy'S", "restaurant_id" : "30112340" }" : "The Movable Feast", "restaurant_id" : "40361606" }
{ "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
>
```

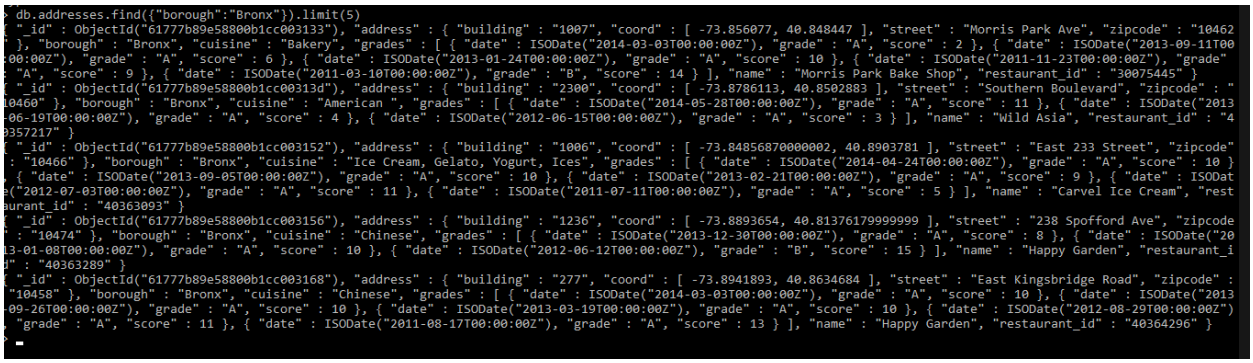

4. Write a MongoDB query to display the fields restaurant_id, name, borough and zip code, but exclude the field _id for all the documents in the collection restaurant.


```
> db.addresses.find({}, {restaurant_id:1, _id:0, name:1,borough:1, "address.zipcode":1 })
{ "address" : { "zipcode" : "10462" }, "borough" : "Bronx", "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "address" : { "zipcode" : "11225" }, "borough" : "Brooklyn", "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "address" : { "zipcode" : "10019" }, "borough" : "Manhattan", "name" : "Dj Reynolds Pub And Restaurant", "restaurant_id" : "30191841" }
{ "address" : { "zipcode" : "11224" }, "borough" : "Brooklyn", "name" : "Riviera Caterer", "restaurant_id" : "40356018" }
{ "address" : { "zipcode" : "11374" }, "borough" : "Queens", "name" : "Tov Kosher Kitchen", "restaurant_id" : "40356068" }
{ "address" : { "zipcode" : "11369" }, "borough" : "Queens", "name" : "Brunos On The Boulevard", "restaurant_id" : "40356151" }
{ "address" : { "zipcode" : "10314" }, "borough" : "Staten Island", "name" : "Kosher Island", "restaurant_id" : "40356442" }
{ "address" : { "zipcode" : "11234" }, "borough" : "Brooklyn", "name" : "Wilken'S Fine Food", "restaurant_id" : "40356483" }
{ "address" : { "zipcode" : "11219" }, "borough" : "Brooklyn", "name" : "Regina Caterers", "restaurant_id" : "40356649" }
{ "address" : { "zipcode" : "11226" }, "borough" : "Brooklyn", "name" : "Taste The Tropics Ice Cream", "restaurant_id" : "40356731" }
{ "address" : { "zipcode" : "10460" }, "borough" : "Bronx", "name" : "Wild Asia", "restaurant_id" : "40357217" }
{ "address" : { "zipcode" : "11214" }, "borough" : "Brooklyn", "name" : "C & C Catering Service", "restaurant_id" : "40357437" }
{ "address" : { "zipcode" : "11208" }, "borough" : "Brooklyn", "name" : "May May Kitchen", "restaurant_id" : "40358429" }
{ "address" : { "zipcode" : "10065" }, "borough" : "Manhattan", "name" : "1 East 66Th Street Kitchen", "restaurant_id" : "40359480" }
{ "address" : { "zipcode" : "11223" }, "borough" : "Brooklyn", "name" : "Seuda Foods", "restaurant_id" : "40360045" }
{ "address" : { "zipcode" : "11218" }, "borough" : "Brooklyn", "name" : "Carvel Ice Cream", "restaurant_id" : "40360076" }
{ "address" : { "zipcode" : "11004" }, "borough" : "Queens", "name" : "Carvel Ice Cream", "restaurant_id" : "40361322" }
{ "address" : { "zipcode" : "11209" }, "borough" : "Brooklyn", "name" : "Nordic Delicacies", "restaurant_id" : "40361390" }
{ "address" : { "zipcode" : "10021" }, "borough" : "Manhattan", "name" : "Glorious Food", "restaurant_id" : "40361521" }
{ "address" : { "zipcode" : "11215" }, "borough" : "Brooklyn", "name" : "The Movable Feast", "restaurant_id" : "40361606" }
Type "it" for more
```

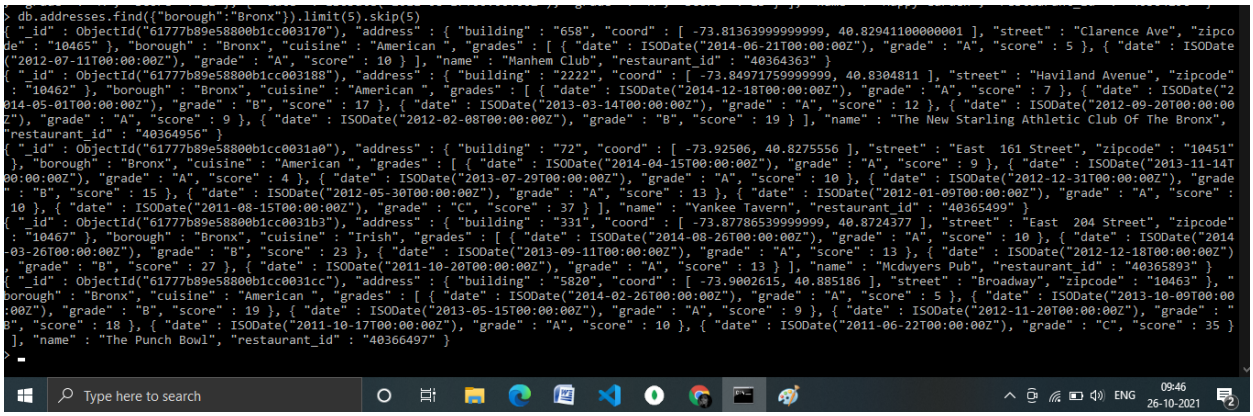

5. Write a MongoDB query to display the first 5 restaurant which is in the borough Bronx.


```
> db.addresses.find({"borough":"Bronx"})
{ "_id" : ObjectId("61777b89e58800b1cc003133"), "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("61777b89e58800b1cc00313d"), "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"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2013
-06-19T00:00:00Z"), "grade" : "A", "score" : 4 }, { "date" : ISODate("2012-06-15T00:00:00Z"), "grade" : "A", "score" : 3 } ], "name" : "Wild Asia", "restaurant_id" : "4
0357217" }
{ "_id" : ObjectId("61777b89e58800b1cc003152"), "address" : { "building" : "1006", "coord" : [ -73.84856870000002, 40.8903781 ], "street" : "East 233 Street", "zipcode"
: "10466" }, "borough" : "Bronx", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "grades" : [ { "date" : ISODate("2014-04-24T00:00:00Z"), "grade" : "A", "score" : 10 }
, { "date" : ISODate("2013-09-05T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2013-02-21T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODat
e("2012-07-03T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2011-07-11T00:00:00Z"), "grade" : "A", "score" : 5 } ], "name" : "Carvel Ice Cream", "rest
aurant_id" : "40363093" }
{ "_id" : ObjectId("61777b89e58800b1cc003156"), "address" : { "building" : "1236", "coord" : [ -73.8893654, 40.81376179999999 ], "street" : "238 Spofford Ave", "zipcode
" : "10474" }, "borough" : "Bronx", "cuisine" : "Chinese", "grades" : [ { "date" : ISODate("2013-12-30T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("20
13-01-08T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2012-06-12T00:00:00Z"), "grade" : "B", "score" : 15 } ], "name" : "Happy Garden", "restaurant_i
d" : "40363289" }
{ "_id" : ObjectId("61777b89e58800b1cc003168"), "address" : { "building" : "277", "coord" : [ -73.8941893, 40.8634684 ], "street" : "East Kingsbridge Road", "zipcode" :
"10458" }, "borough" : "Bronx", "cuisine" : "Chinese", "grades" : [ { "date" : ISODate("2014-03-03T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2013
-09-26T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2013-03-19T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2012-08-29T00:00:00Z")
, "grade" : "A", "score" : 11 }, { "date" : ISODate("2011-08-17T00:00:00Z"), "grade" : "A", "score" : 13 } ], "name" : "Happy Garden", "restaurant_id" : "40364296" }
{ "_id" : ObjectId("61777b89e58800b1cc003170"), "address" : { "building" : "658", "coord" : [ -73.81363999999999, 40.82941100000001 ], "street" : "Clarence Ave", "zipco
de" : "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("61777b89e58800b1cc003188"), "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("2
```

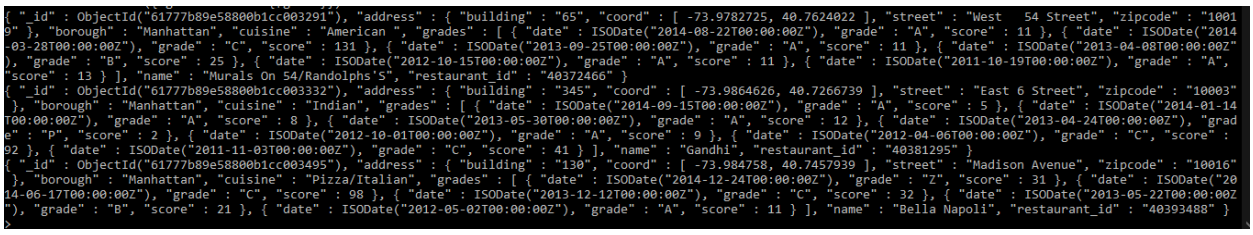

6. Write a MongoDB query to display all the restaurant which is in the borough Bronx.



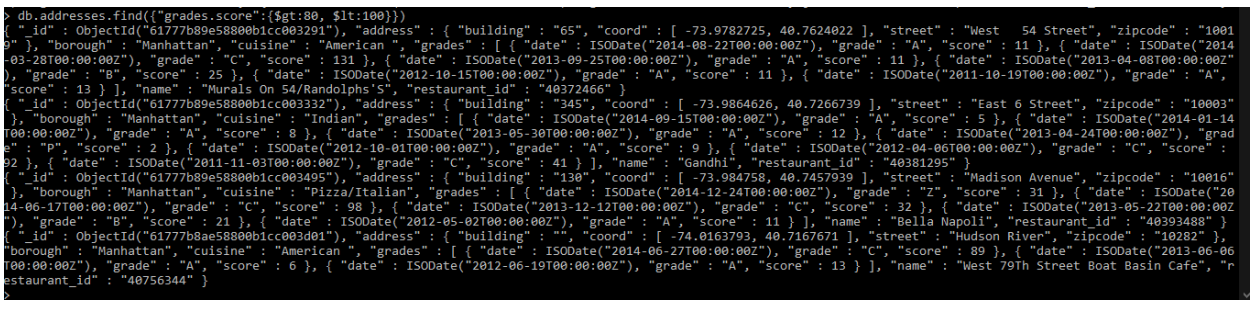
7. Write a MongoDB query to display the next 5 restaurants after skipping first 5 which are in the borough Bronx.



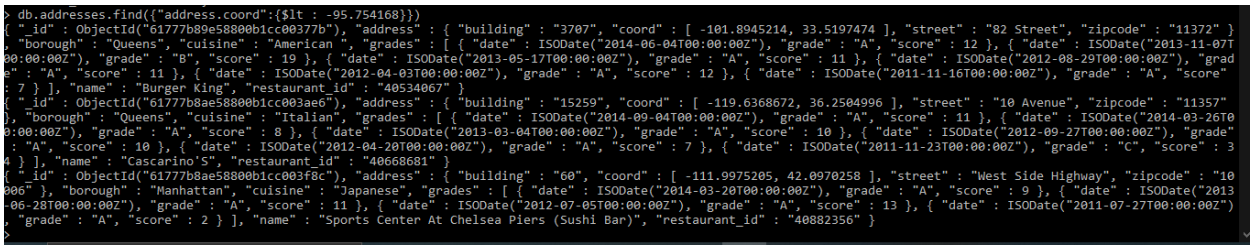
8. Write a MongoDB query to find the restaurants who achieved a score more than 90.



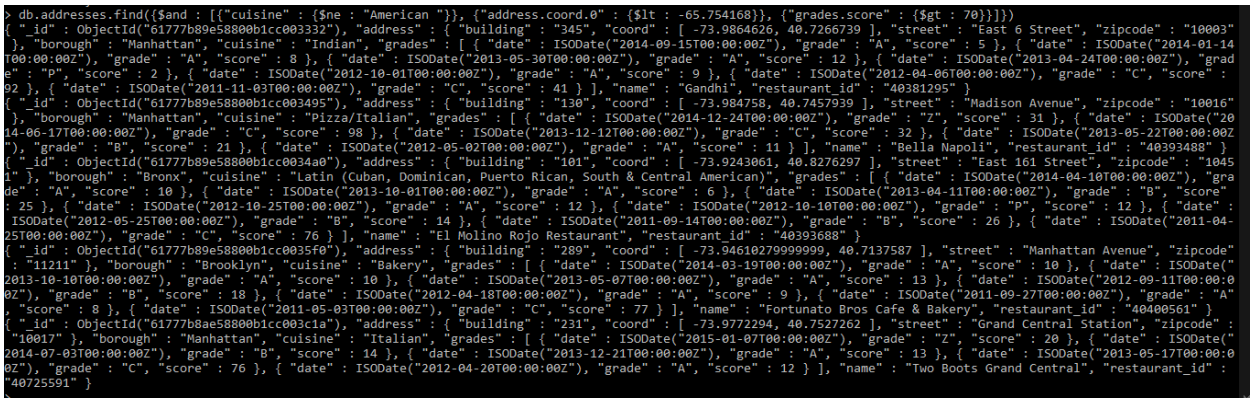
9. Write a MongoDB query to find the restaurants that achieved a score, more than 80 but less than 100.



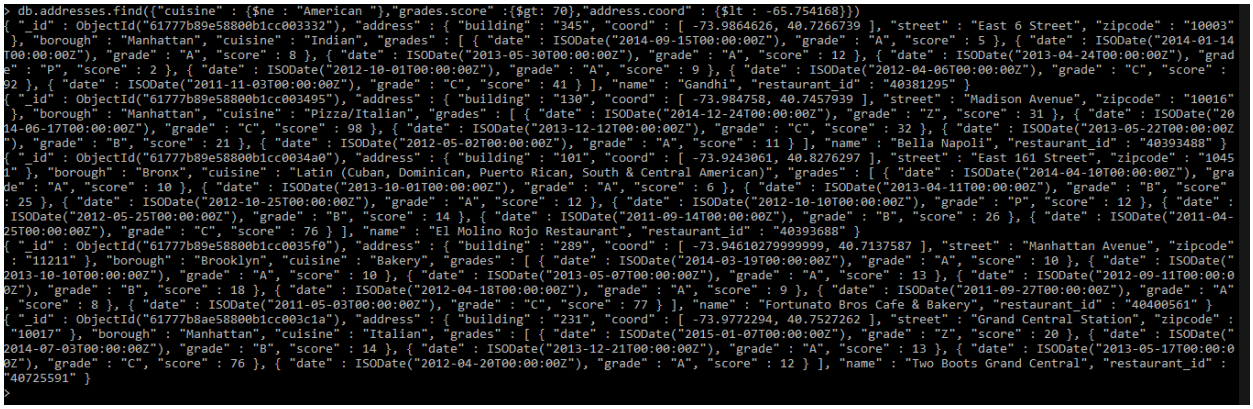
10. Write a MongoDB query to find the restaurants which locate in latitude value less than -95.754168.



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.

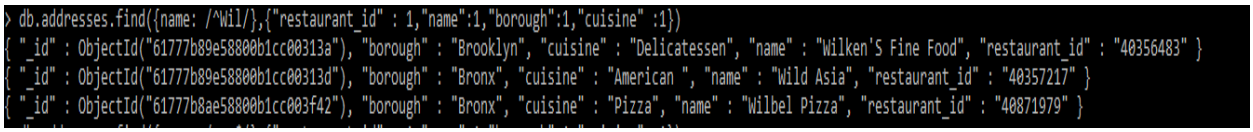


12. Write a MongoDB query to find the restaurants which do not prepare any cuisine of 'American' and achieved a score more than 70 and located in the longitude less than -65.754168.

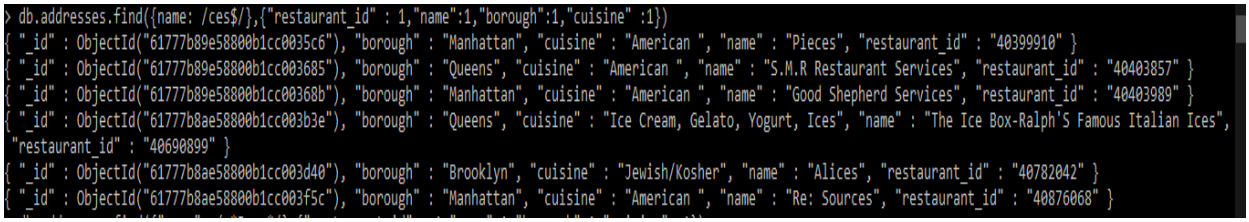


13. Write a MongoDB query to find the restaurants which do not prepare any cuisine of 'American ' and achieved a grade point 'A' not belongs to the borough Brooklyn. The document must be displayed according to the cuisine in descending order.

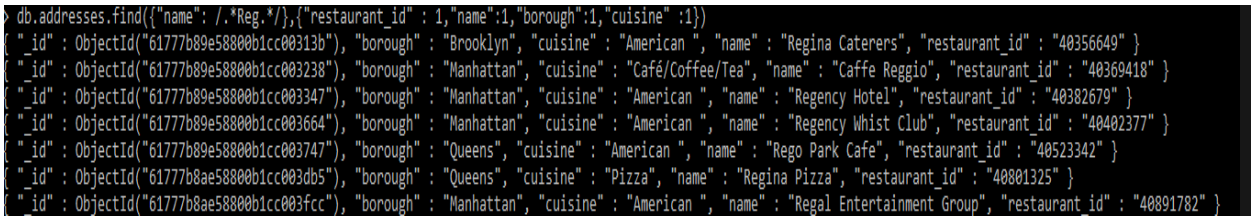
14. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which contain 'Wil' as first three letters for its name.



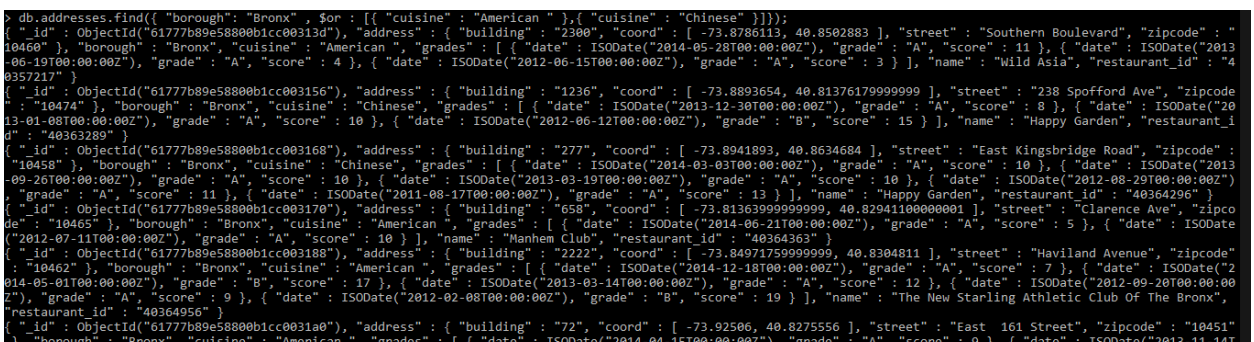
15. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which contain 'ces' as last three letters for its name.



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.



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



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.addresses.find({"borough" :{$in :["Staten Island","Queens","Bronx","Brooklyn"]}},{"restaurant_id" : 1,"name":1,"borough":1, "cuisine" :1});
{ "_id" : ObjectId("61777b89e58800b1cc003133"), "borough" : "Bronx", "cuisine" : "Bakery", "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("61777b89e58800b1cc003134"), "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "_id" : ObjectId("61777b89e58800b1cc003136"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "Riviera Caterer", "restaurant_id" : "40356018" }
{ "_id" : ObjectId("61777b89e58800b1cc003137"), "borough" : "Queens", "cuisine" : "Jewish/Kosher", "name" : "Tov Kosher Kitchen", "restaurant_id" : "40356068" }
{ "_id" : ObjectId("61777b89e58800b1cc003138"), "borough" : "Queens", "cuisine" : "American ", "name" : "Brunos On The Boulevard", "restaurant_id" : "40356151" }
{ "_id" : ObjectId("61777b89e58800b1cc003139"), "borough" : "Staten Island", "cuisine" : "Jewish/Kosher", "name" : "Kosher Island", "restaurant_id" : "40356442" }
{ "_id" : ObjectId("61777b89e58800b1cc00313a"), "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Wilken'S Fine Food", "restaurant_id" : "40356483" }
{ "_id" : ObjectId("61777b89e58800b1cc00313b"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "Regina Caterers", "restaurant_id" : "40356649" }
{ "_id" : ObjectId("61777b89e58800b1cc00313c"), "borough" : "Brooklyn", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Taste The Tropics Ice Cream", "restaura
nt_id" : "40356731" }
{ "_id" : ObjectId("61777b89e58800b1cc00313d"), "borough" : "Bronx", "cuisine" : "American ", "name" : "Wild Asia", "restaurant_id" : "40357217" }
{ "_id" : ObjectId("61777b89e58800b1cc00313e"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "C & C Catering Service", "restaurant_id" : "40357437" }
{ "_id" : ObjectId("61777b89e58800b1cc00313f"), "borough" : "Brooklyn", "cuisine" : "Chinese", "name" : "May May Kitchen", "restaurant_id" : "40358429" }
{ "_id" : ObjectId("61777b89e58800b1cc003141"), "borough" : "Brooklyn", "cuisine" : "Jewish/Kosher", "name" : "Seuda Foods", "restaurant_id" : "40360045" }
{ "_id" : ObjectId("61777b89e58800b1cc003142"), "borough" : "Brooklyn", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Carvel Ice Cream", "restaurant_id" : "4
0360076" }
{ "_id" : ObjectId("61777b89e58800b1cc003143"), "borough" : "Queens", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Carvel Ice Cream", "restaurant_id" : "403
61322" }
{ "_id" : ObjectId("61777b89e58800b1cc003144"), "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Nordic Delicacies", "restaurant_id" : "40361390" }
{ "_id" : ObjectId("61777b89e58800b1cc003146"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "The Movable Feast", "restaurant_id" : "40361606" }
{ "_id" : ObjectId("61777b89e58800b1cc003147"), "borough" : "Queens", "cuisine" : "Delicatessen", "name" : "Sal'S Deli", "restaurant_id" : "40361618" }
{ "_id" : ObjectId("61777b89e58800b1cc003149"), "borough" : "Queens", "cuisine" : "Delicatessen", "name" : "Steve Chu'S Deli & Grocery", "restaurant_id" : "40361998" }
{ "_id" : ObjectId("61777b89e58800b1cc00314d"), "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "White Castle", "restaurant_id" : "40362344" }
Type "it" for more
```

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.addresses.find({"borough" :{$nin :["Staten Island","Queens","Bronx","Brooklyn"]}},{"restaurant_id" : 1,"name":1,"borough":1,"cuisine" :1});
{ "_id" : ObjectId("61777b89e58800b1cc003135"), "borough" : "Manhattan", "cuisine" : "Irish", "name" : "Dj Reynolds Pub And Restaurant", "restaurant_id" : "30191841" }
{ "_id" : ObjectId("61777b89e58800b1cc003140"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "1 East 66Th Street Kitchen", "restaurant_id" : "40359480" }
{ "_id" : ObjectId("61777b89e58800b1cc003145"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Glorious Food", "restaurant_id" : "40361521" }
{ "_id" : ObjectId("61777b89e58800b1cc003148"), "borough" : "Manhattan", "cuisine" : "Delicatessen", "name" : "Bully'S Deli", "restaurant_id" : "40361708" }
{ "_id" : ObjectId("61777b89e58800b1cc00314a"), "borough" : "Manhattan", "cuisine" : "Chicken", "name" : "Harriet'S Kitchen", "restaurant_id" : "40362098" }
{ "_id" : ObjectId("61777b89e58800b1cc00314b"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "P & S Deli Grocery", "restaurant_id" : "40362264" }
{ "_id" : ObjectId("61777b89e58800b1cc00314c"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Angelika Film Center", "restaurant_id" : "40362274" }
{ "_id" : ObjectId("61777b89e58800b1cc00314f"), "borough" : "Manhattan", "cuisine" : "Turkish", "name" : "The Country Cafe", "restaurant_id" : "40362715" }
{ "_id" : ObjectId("61777b89e58800b1cc003151"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Downtown Deli", "restaurant_id" : "40363021" }
{ "_id" : ObjectId("61777b89e58800b1cc003155"), "borough" : "Manhattan", "cuisine" : "Bakery", "name" : "Olive'S", "restaurant_id" : "40363151" }
{ "_id" : ObjectId("61777b89e58800b1cc003157"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Cafe Metro", "restaurant_id" : "40363298" }
{ "_id" : ObjectId("61777b89e58800b1cc003159"), "borough" : "Manhattan", "cuisine" : "Sandwiches/Salads/Mixed Buffet", "name" : "Lexler Deli", "restaurant_id" : "403634
26" }
{ "_id" : ObjectId("61777b89e58800b1cc00315d"), "borough" : "Manhattan", "cuisine" : "Continental", "name" : "Lorenzo & Maria'S", "restaurant_id" : "40363630" }
{ "_id" : ObjectId("61777b89e58800b1cc00315e"), "borough" : "Manhattan", "cuisine" : "Pizza", "name" : "Domino'S Pizza", "restaurant_id" : "40363644" }
{ "_id" : ObjectId("61777b89e58800b1cc00315f"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Berkely", "restaurant_id" : "40363685" }
{ "_id" : ObjectId("61777b89e58800b1cc003163"), "borough" : "Manhattan", "cuisine" : "Pizza", "name" : "Domino'S Pizza", "restaurant_id" : "40363945" }
{ "_id" : ObjectId("61777b89e58800b1cc003164"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Spoon Bread Catering", "restaurant_id" : "40364179" }
{ "_id" : ObjectId("61777b89e58800b1cc00316a"), "borough" : "Manhattan", "cuisine" : "Chicken", "name" : "Texas Rotisserie", "restaurant_id" : "40364304" }
{ "_id" : ObjectId("61777b89e58800b1cc00316d"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Metropolitan Club", "restaurant_id" : "40364347" }
{ "_id" : ObjectId("61777b89e58800b1cc00316e"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Palm Restaurant", "restaurant_id" : "40364355" }
Type "it" for more
```

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.addresses.find({"grades.score" : { $not: {$gt : 10}}},{"restaurant_id" : 1,"name":1,"borough":1,"cuisine" :1});
{ "_id" : ObjectId("61777b89e58800b1cc00313e"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "C & C Catering Service", "restaurant_id" : "40357437" }
{ "_id" : ObjectId("61777b89e58800b1cc003140"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "1 East 66Th Street Kitchen", "restaurant_id" : "40359480" }
{ "_id" : ObjectId("61777b89e58800b1cc003144"), "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Nordic Delicacies", "restaurant_id" : "40361390" }
{ "_id" : ObjectId("61777b89e58800b1cc00314d"), "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "White Castle", "restaurant_id" : "40362344" }
{ "_id" : ObjectId("61777b89e58800b1cc003160"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "Sonny'S Heros", "restaurant_id" : "40363744" }
{ "_id" : ObjectId("61777b89e58800b1cc003170"), "borough" : "Bronx", "cuisine" : "American ", "name" : "Manhem Club", "restaurant_id" : "40364363" }
{ "_id" : ObjectId("61777b89e58800b1cc00317e"), "borough" : "Staten Island", "cuisine" : "American ", "name" : "Great Kills Yacht Club", "restaurant_id" : "40364610" }
{ "_id" : ObjectId("61777b89e58800b1cc003185"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Serendipity 3", "restaurant_id" : "40364863" }
{ "_id" : ObjectId("61777b89e58800b1cc003189"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "White Horse Tavern", "restaurant_id" : "40364958" }
{ "_id" : ObjectId("61777b89e58800b1cc003192"), "borough" : "Manhattan", "cuisine" : "Irish", "name" : "Dorrian'S Red Hand Restaurant", "restaurant_id" : "40365239" }
{ "_id" : ObjectId("61777b89e58800b1cc0031e3"), "borough" : "Manhattan", "cuisine" : "Mexican", "name" : "Mexico Lindo Restaurant", "restaurant_id" : "40367038" }
{ "_id" : ObjectId("61777b89e58800b1cc003203"), "borough" : "Brooklyn", "cuisine" : "Greek", "name" : "El Greco Diner", "restaurant_id" : "40367795" }
{ "_id" : ObjectId("61777b89e58800b1cc003209"), "borough" : "Bronx", "cuisine" : "Not Listed/Not Applicable", "name" : "The Lark'S Nest", "restaurant_id" : "40367946" }

{ "_id" : ObjectId("61777b89e58800b1cc00320a"), "borough" : "Bronx", "cuisine" : "Café/Coffee/Tea", "name" : "Terrace Cafe", "restaurant_id" : "40368018" }
{ "_id" : ObjectId("61777b89e58800b1cc00320b"), "borough" : "Bronx", "cuisine" : "African", "name" : "African Terrace", "restaurant_id" : "40368021" }
{ "_id" : ObjectId("61777b89e58800b1cc00320e"), "borough" : "Bronx", "cuisine" : "American ", "name" : "African Market (Baboon Cafe)", "restaurant_id" : "40368026" }
{ "_id" : ObjectId("61777b89e58800b1cc00320f"), "borough" : "Staten Island", "cuisine" : "Italian", "name" : "Roadhouse Restaurant", "restaurant_id" : "40368034" }
{ "_id" : ObjectId("61777b89e58800b1cc003231"), "borough" : "Manhattan", "cuisine" : "French", "name" : "Pergola Des Artistes", "restaurant_id" : "40369139" }
{ "_id" : ObjectId("61777b89e58800b1cc00323d"), "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "Mcdonald'S", "restaurant_id" : "40369535" }
{ "_id" : ObjectId("61777b89e58800b1cc003241"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "The River Cafe", "restaurant_id" : "40369608" }
Type "it" for more
```

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.addresses.find(
...     {$or: [
...       {name: /^Wil/},
...       {"$and": [
...            {"cuisine" : {$ne :"American "}},
...            {"cuisine" : {$ne :"Chinees"}}
...       ]}]},{"restaurant_id" : 1,"name":1,"borough":1,"cuisine" :1})
{ "_id" : ObjectId("61777b89e58800b1cc003133"), "borough" : "Bronx", "cuisine" : "Bakery", "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("61777b89e58800b1cc003134"), "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "_id" : ObjectId("61777b89e58800b1cc003135"), "borough" : "Manhattan", "cuisine" : "Irish", "name" : "Dj Reynolds Pub And Restaurant", "restaurant_id" : "30191841" }
{ "_id" : ObjectId("61777b89e58800b1cc003137"), "borough" : "Queens", "cuisine" : "Jewish/Kosher", "name" : "Tov Kosher Kitchen", "restaurant_id" : "40356068" }
{ "_id" : ObjectId("61777b89e58800b1cc003139"), "borough" : "Staten Island", "cuisine" : "Jewish/Kosher", "name" : "Kosher Island", "restaurant_id" : "40356442" }
{ "_id" : ObjectId("61777b89e58800b1cc00313a"), "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Wilken'S Fine Food", "restaurant_id" : "40356483" }
{ "_id" : ObjectId("61777b89e58800b1cc00313c"), "borough" : "Brooklyn", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Taste The Tropics Ice Cream", "restaura
nt_id" : "40356731" }
{ "_id" : ObjectId("61777b89e58800b1cc00313d"), "borough" : "Bronx", "cuisine" : "American ", "name" : "Wild Asia", "restaurant_id" : "40357217" }
{ "_id" : ObjectId("61777b89e58800b1cc00313f"), "borough" : "Brooklyn", "cuisine" : "Chinese", "name" : "May May Kitchen", "restaurant_id" : "40358429" }
{ "_id" : ObjectId("61777b89e58800b1cc003141"), "borough" : "Brooklyn", "cuisine" : "Jewish/Kosher", "name" : "Seuda Foods", "restaurant_id" : "40360045" }
{ "_id" : ObjectId("61777b89e58800b1cc003142"), "borough" : "Brooklyn", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Carvel Ice Cream", "restaurant_id" : "4
0360076" }
{ "_id" : ObjectId("61777b89e58800b1cc003143"), "borough" : "Queens", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Carvel Ice Cream", "restaurant_id" : "403
61322" }
{ "_id" : ObjectId("61777b89e58800b1cc003144"), "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Nordic Delicacies", "restaurant_id" : "40361390" }
{ "_id" : ObjectId("61777b89e58800b1cc003147"), "borough" : "Queens", "cuisine" : "Delicatessen", "name" : "Sal'S Deli", "restaurant_id" : "40361618" }
{ "_id" : ObjectId("61777b89e58800b1cc003148"), "borough" : "Manhattan", "cuisine" : "Delicatessen", "name" : "Bully'S Deli", "restaurant_id" : "40361708" }
{ "_id" : ObjectId("61777b89e58800b1cc003149"), "borough" : "Queens", "cuisine" : "Delicatessen", "name" : "Steve Chu'S Deli & Grocery", "restaurant_id" : "40361998" }
{ "_id" : ObjectId("61777b89e58800b1cc00314a"), "borough" : "Manhattan", "cuisine" : "Chicken", "name" : "Harriet'S Kitchen", "restaurant_id" : "40362098" }
{ "_id" : ObjectId("61777b89e58800b1cc00314d"), "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "White Castle", "restaurant_id" : "40362344" }
{ "_id" : ObjectId("61777b89e58800b1cc00314e"), "borough" : "Queens", "cuisine" : "Chinese", "name" : "Ho Mei Restaurant", "restaurant_id" : "40362432" }
{ "_id" : ObjectId("61777b89e58800b1cc00314f"), "borough" : "Manhattan", "cuisine" : "Turkish", "name" : "The Country Cafe", "restaurant_id" : "40362715" }
Type "it" for more
```

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.addresses.find(
...     {
...      "grades.date": ISODate("2014-08-11T00:00:00Z"),
...      "grades.grade":"A" ,
...      "grades.score" : 11
...     },
...     {"restaurant_id" : 1,"name":1,"grades":1})
{ "_id" : ObjectId("61777b89e58800b1cc0031b1"), "grades" : [ { "date" : ISODate("2014-08-11T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2013-07-22T0
0:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2013-03-14T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-07-02T00:00:00Z"), "grade"
: "A", "score" : 11 }, { "date" : ISODate("2012-02-02T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2011-08-24T00:00:00Z"), "grade" : "A", "score" :
11 } ], "name" : "Neary'S Pub", "restaurant_id" : "40365871" }
{ "_id" : ObjectId("61777b89e58800b1cc00328c"), "grades" : [ { "date" : ISODate("2014-08-11T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2013-12-10T0
0:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2013-06-10T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-06-08T00:00:00Z"), "grade"
: "A", "score" : 13 }, { "date" : ISODate("2012-01-25T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2011-09-13T00:00:00Z"), "grade" : "A", "score" : 1
2 } ], "name" : "Don Filippo Restaurant", "restaurant_id" : "40372417" }
{ "_id" : ObjectId("61777b89e58800b1cc003552"), "grades" : [ { "date" : ISODate("2014-08-11T00:00:00Z"), "grade" : "A", "score" : 6 }, { "date" : ISODate("2013-08-29T00
:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-04-25T00:00:00Z"), "grade" : "C", "score" : 2 }, { "date" : ISODate("2012-10-23T00:00:00Z"), "grade"
: "A", "score" : 13 }, { "date" : ISODate("2012-04-16T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2011-11-15T00:00:00Z"), "grade" : "A", "score" : 1
2 } ], "name" : "Mustang Sally'S Restaurant", "restaurant_id" : "40397374" }
{ "_id" : ObjectId("61777b89e58800b1cc00375e"), "grades" : [ { "date" : ISODate("2015-01-12T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2014-08-11T0
0:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2014-01-14T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2013-02-07T00:00:00Z"), "grade"
: "A", "score" : 10 }, { "date" : ISODate("2012-04-30T00:00:00Z"), "grade" : "A", "score" : 11 } ], "name" : "Club Macanudo (Cigar Bar)", "restaurant_id" : "40526406"
}
```

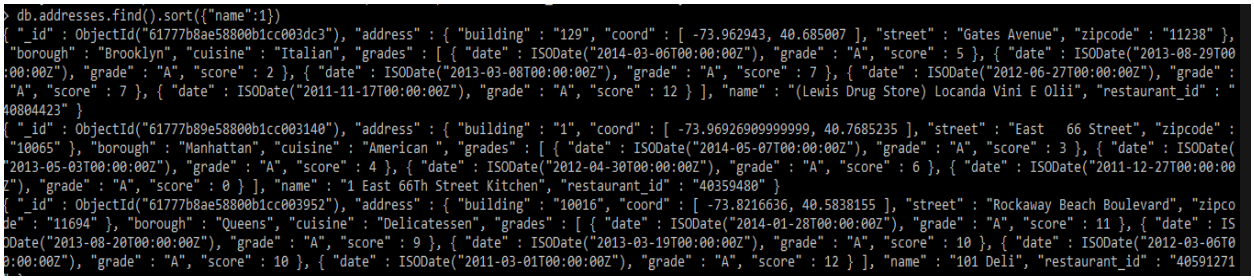
23. Write a MongoDB query to find the restaurant Id, name and grades for those restaurants where the 2nd element of grades array contains a grade of "A" and score 9 on an ISODate "2014-08-11T00:00:00Z"

```
> db.addresses.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}
... )
{ "_id" : ObjectId("61777b89e58800b1cc00375e"), "grades" : [ { "date" : ISODate("2015-01-12T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2014-08-11T0
0:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2014-01-14T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2013-02-07T00:00:00Z"), "grade"
: "A", "score" : 10 }, { "date" : ISODate("2012-04-30T00:00:00Z"), "grade" : "A", "score" : 11 } ], "name" : "Club Macanudo (Cigar Bar)", "restaurant_id" : "40526406"
}
>
```

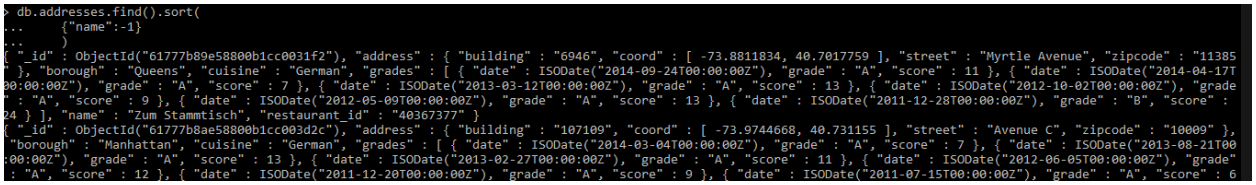
24. Write a MongoDB query to find the restaurant Id, name, address and geographical location for those restaurants where 2nd element of coord array contains a value which is more than 42 and upto 52.

```
> db.addresses.find(
...     {
...       "address.coord.1": {$gt : 42, $lte : 52}
...     },
...       {"restaurant_id" : 1,"name":1,"address":1,"coord":1}
... )
{ "_id" : ObjectId("61777b89e58800b1cc0033d5"), "address" : { "building" : "47", "coord" : [ -78.877224, 42.89546199999999 ], "street" : "Broadway @ Trinity Pl", "zipco
de" : "10006" }, "name" : "T.G.I. Friday'S", "restaurant_id" : "40387990" }
{ "_id" : ObjectId("61777b89e58800b1cc003401"), "address" : { "building" : "1", "coord" : [ -0.7119979, 51.6514664 ], "street" : "Pennplaza E, Penn Sta", "zipcode" : "1
0001" }, "name" : "T.G.I. Fridays", "restaurant_id" : "40388936" }
{ "_id" : ObjectId("61777b89e58800b1cc00365a"), "address" : { "building" : "3000", "coord" : [ -87.86567699999999, 42.61150920000001 ], "street" : "47 Avenue", "zipcode
" : "11101" }, "name" : "Di Luvio'S Deli", "restaurant_id" : "40402284" }
{ "_id" : ObjectId("61777b89e58800b1cc00388f"), "address" : { "building" : "21972199", "coord" : [ -78.589606, 42.8912372 ], "street" : "Broadway", "zipcode" : "10024"
}, "name" : "La Caridad 78", "restaurant_id" : "40568285" }
{ "_id" : ObjectId("61777b8ae58800b1cc003f61"), "address" : { "building" : "7981", "coord" : [ -84.9751215, 45.4713351 ], "street" : "Hoyt Street", "zipcode" : "11201"
}, "name" : "Bijan'S", "restaurant_id" : "40876618" }
{ "_id" : ObjectId("61777b8ae58800b1cc003f76"), "address" : { "building" : "0", "coord" : [ -88.0778799, 42.4154769 ], "street" : "& Grand Central", "zipcode" : "10017"
}, "name" : "Hyatt, Ny Central/Room Service", "restaurant_id" : "40879243" }
{ "_id" : ObjectId("61777b8ae58800b1cc003f8c"), "address" : { "building" : "60", "coord" : [ -111.9975205, 42.0970258 ], "street" : "West Side Highway", "zipcode" : "10
006" }, "name" : "Sports Center At Chelsea Piers (Sushi Bar)", "restaurant_id" : "40882356" }
```

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



26. Write a MongoDB query to arrange the name of the restaurants in descending along with all the columns.



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.addresses.find().sort(
    {"cuisine":1,"borough" : -1,}
   )
```

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

```
db.addresses.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.addresses.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.addresses.find(
    {"grades.score" :
       {$mod : [7,0]}
    },
       {"restaurant_id" : 1,"name":1,"grades":1}
  )
```

31. Write a MongoDB query to find the restaurant name, borough, longitude and attitude and cuisine for those restaurants which contains 'mon' as three letters somewhere in its name.

```
db.addresses.find(
```

```
    { name :
      { $regex : "mon.*", $options: "i" }
    },
        {
          "name":1,
          "borough":1,
          "address.coord":1,
          "cuisine" :1
         }
    )
```

32. Write a MongoDB query to find the restaurant name, borough, longitude and latitude and cuisine for those restaurants which contain 'Mad' as first three letters of its name.

```
db.addresses.find(
    { name :
      { $regex : /^Mad/i, }
    },
        {
          "name":1,
          "borough":1,
          "address.coord":1,
          "cuisine" :1
         }
    )
```