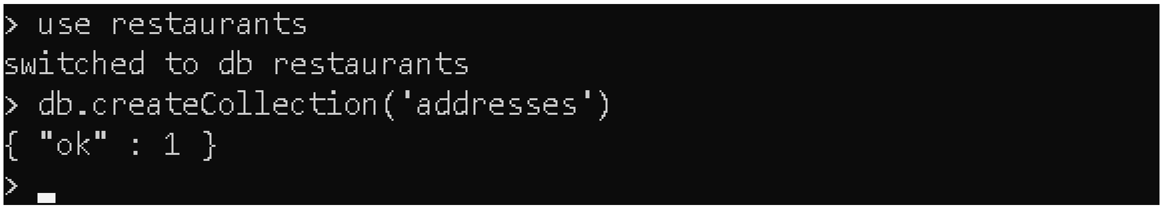
Assignment 3
MongoDB – Complex Queries

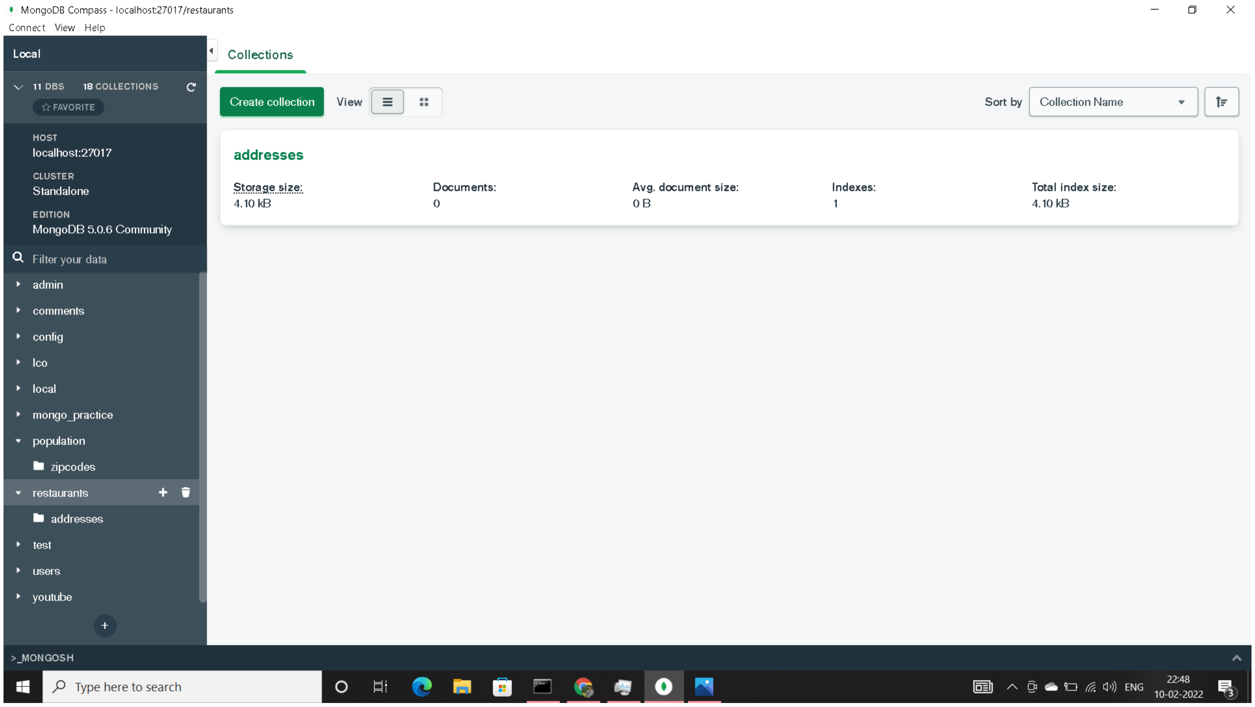
SHAIK MOHSIN S

Mongo DB Exercises - With the Restaurants Data Set

1. Download the restaurants.zip file
2. Unzip the file, you will see restaurants.json file
3. Run the mongod server
4. Run the following command to import the json file provided. It will load the
json file into the mongodb with database name - restaurants, collections name - addresses



```
> use restaurants
switched to db restaurants
> db.createCollection('addresses')
{ "ok" : 1 }
> _
```



MongoDB Compass - localhost:27017/restaurants

HOST
localhost:27017

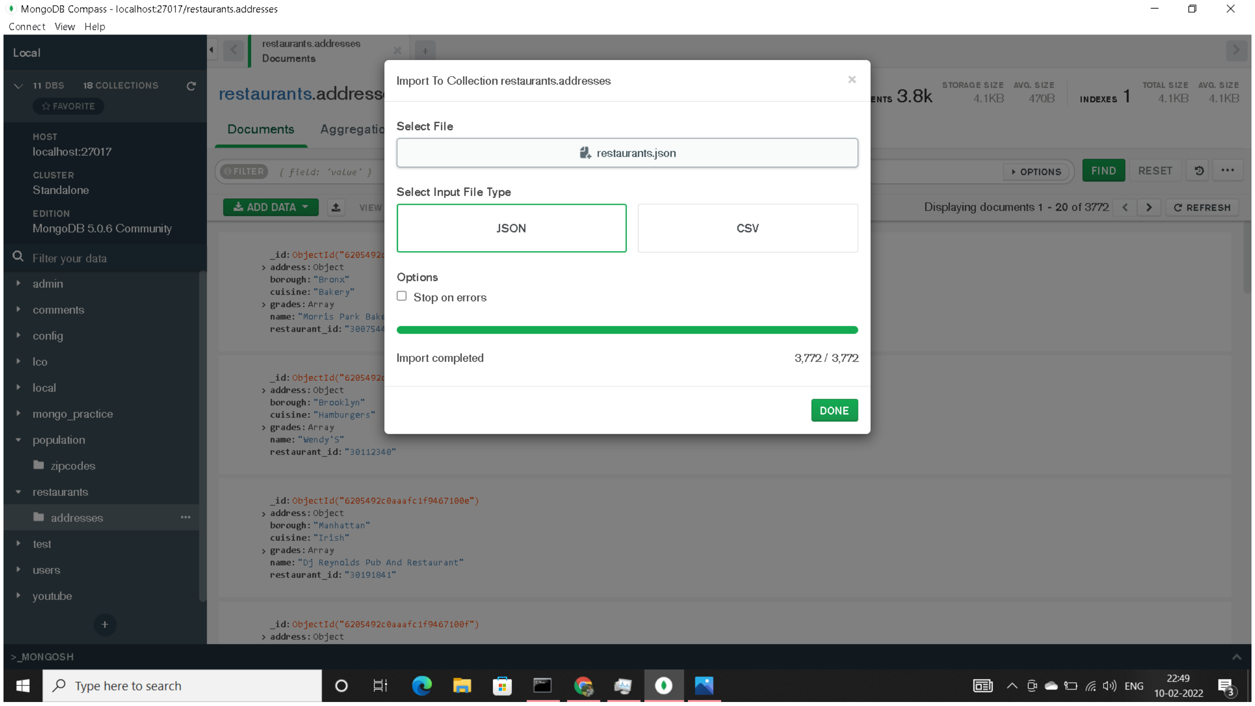
CLUSTER
Standalone

EDITION
MongoDB 5.0.6 Community

addresses

Storage size: 4.10 kB
Documents: 0
Avg. document size: 0 B
Indexes: 1
Total index size: 4.10 kB

Sort by Collection Name



Import To Collection restaurants.addresses

Select File

restaurants.json

Select Input File Type

JSON

CSV

Options

□ Stop on errors

Import completed 3,772 / 3,772

DONE

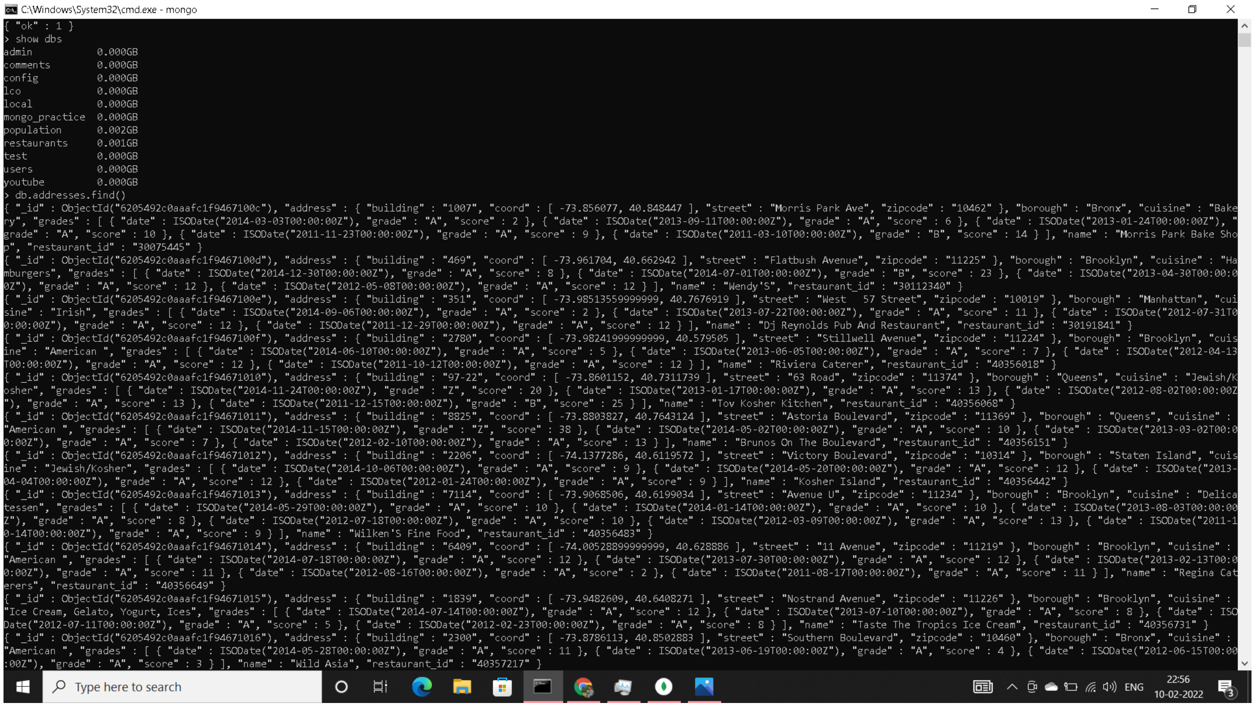
```
_id: ObjectId("6205492c
> address: Object
  borough: "Bronx"
  cuisine: "Bakery"
> grades: Array
  name: "Morris Park Bake
  restaurant_id: "3007544

_id: ObjectId("6205492c
> address: Object
  borough: "Brooklyn"
  cuisine: "Hamburgers"
> grades: Array
  name: "Wendy'S"
  restaurant_id: "30112340"

_id: ObjectId("6205492c0aaafc1f9467100e")
> address: Object
  borough: "Manhattan"
  cuisine: "Irish"
> grades: Array
  name: "Dj Reynolds Pub And Restaurant"
  restaurant_id: "30191841"

_id: ObjectId("6205492c0aaafc1f9467100f")
> address: Object
```

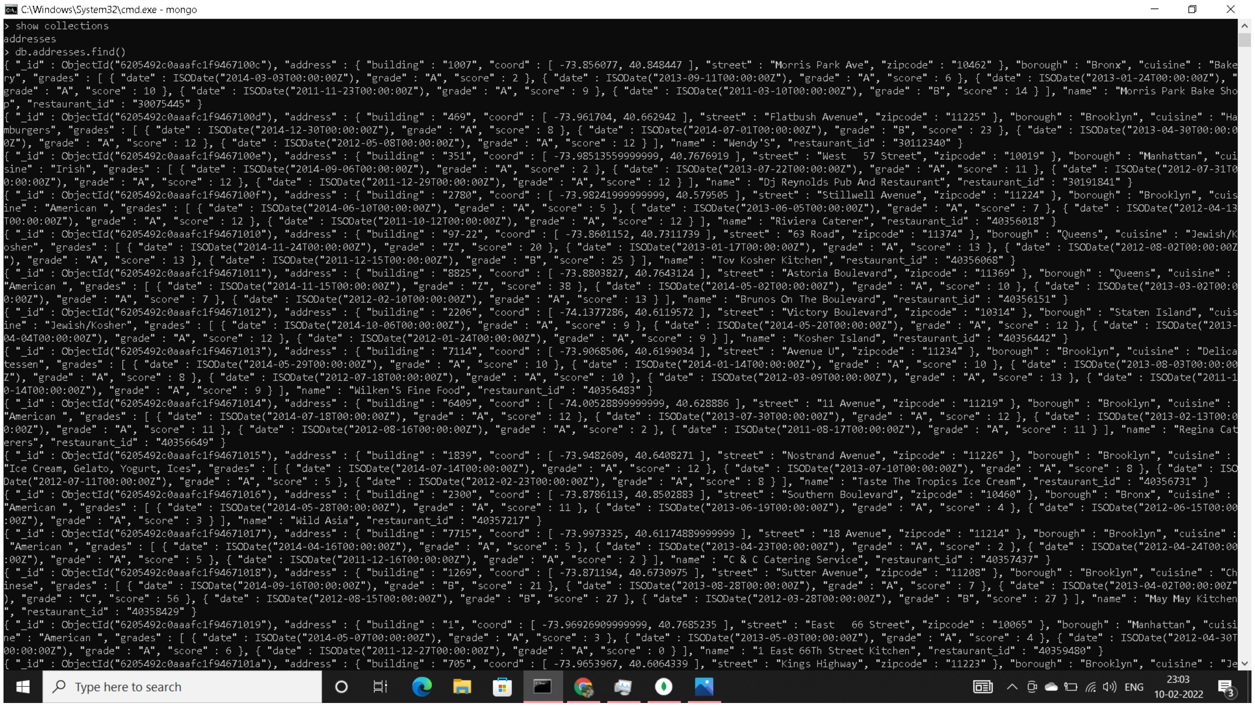
5. Run mongo shell command
6. show databases
7. use restaurants
8. db.addresses.find() should print entire json data



```
{ "ok" : 1 }
> show dbs
admin           0.000GB
comments        0.000GB
config          0.000GB
lco             0.000GB
local           0.000GB
mongo_practice  0.000GB
population      0.002GB
restaurants     0.001GB
test            0.000GB
users           0.000GB
youtube         0.000GB
> db.addresses.find()
{ "_id" : ObjectId("6205492c0aaafc1f9467100c"), "address" : { "building" : "1007", "coord" : [ -73.856077, 40.848447 ], "street" : "Morris Park Ave", "zipcode" : "10462" }, "borough" : "Bronx", "cuisine" : "Bake
ry", "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 Sho
p", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100d"), "address" : { "building" : "469", "coord" : [ -73.961704, 40.662942 ], "street" : "Flatbush Avenue", "zipcode" : "11225" }, "borough" : "Brooklyn", "cuisine" : "Ha
mburgers", "grades" : [ { "date" : ISODate("2014-12-30T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2014-07-01T00:00:00Z"), "grade" : "B", "score" : 23 }, { "date" : ISODate("2013-04-30T00:00:0
0Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-05-08T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100e"), "address" : { "building" : "351", "coord" : [ -73.98513559999999, 40.7676919 ], "street" : "West   57 Street", "zipcode" : "10019" }, "borough" : "Manhattan", "cui
sine" : "Irish", "grades" : [ { "date" : ISODate("2014-09-06T00:00:00Z"), "grade" : "A", "score" : 2 }, { "date" : ISODate("2013-07-22T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2012-07-31T0
0:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2011-12-29T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "Dj Reynolds Pub And Restaurant", "restaurant_id" : "30191841" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100f"), "address" : { "building" : "2780", "coord" : [ -73.98241999999999, 40.579505 ], "street" : "Stillwell Avenue", "zipcode" : "11224" }, "borough" : "Brooklyn", "cuis
ine" : "American ", "grades" : [ { "date" : ISODate("2014-06-10T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2013-06-05T00:00:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2012-04-13
T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2011-10-12T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "Riviera Caterer", "restaurant_id" : "40356018" }
{ "_id" : ObjectId("6205492c0aaafc1f94671010"), "address" : { "building" : "97-22", "coord" : [ -73.8601152, 40.7311739 ], "street" : "63 Road", "zipcode" : "11374" }, "borough" : "Queens", "cuisine" : "Jewish/K
osher", "grades" : [ { "date" : ISODate("2014-11-24T00:00:00Z"), "grade" : "Z", "score" : 20 }, { "date" : ISODate("2013-01-17T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2012-08-02T00:00:00Z
"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2011-12-15T00:00:00Z"), "grade" : "B", "score" : 25 } ], "name" : "Tov Kosher Kitchen", "restaurant_id" : "40356068" }
{ "_id" : ObjectId("6205492c0aaafc1f94671011"), "address" : { "building" : "8825", "coord" : [ -73.8803827, 40.7643124 ], "street" : "Astoria Boulevard", "zipcode" : "11369" }, "borough" : "Queens", "cuisine" :
"American ", "grades" : [ { "date" : ISODate("2014-11-15T00:00:00Z"), "grade" : "Z", "score" : 38 }, { "date" : ISODate("2014-05-02T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2013-03-02T00:0
0:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2012-02-10T00:00:00Z"), "grade" : "A", "score" : 13 } ], "name" : "Brunos On The Boulevard", "restaurant_id" : "40356151" }
{ "_id" : ObjectId("6205492c0aaafc1f94671012"), "address" : { "building" : "2206", "coord" : [ -74.1377286, 40.6119572 ], "street" : "Victory Boulevard", "zipcode" : "10314" }, "borough" : "Staten Island", "cuis
ine" : "Jewish/Kosher", "grades" : [ { "date" : ISODate("2014-10-06T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2014-05-20T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-
04-04T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-01-24T00:00:00Z"), "grade" : "A", "score" : 9 } ], "name" : "Kosher Island", "restaurant_id" : "40356442" }
{ "_id" : ObjectId("6205492c0aaafc1f94671013"), "address" : { "building" : "7114", "coord" : [ -73.9068506, 40.6199034 ], "street" : "Avenue U", "zipcode" : "11234" }, "borough" : "Brooklyn", "cuisine" : "Delica
tessen", "grades" : [ { "date" : ISODate("2014-05-29T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2014-01-14T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2013-08-03T00:00:00
Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2012-07-18T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2012-03-09T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2011-1
0-14T00:00:00Z"), "grade" : "A", "score" : 9 } ], "name" : "Wilken'S Fine Food", "restaurant_id" : "40356483" }
{ "_id" : ObjectId("6205492c0aaafc1f94671014"), "address" : { "building" : "6409", "coord" : [ -74.00528899999999, 40.628886 ], "street" : "11 Avenue", "zipcode" : "11219" }, "borough" : "Brooklyn", "cuisine" :
"American ", "grades" : [ { "date" : ISODate("2014-07-18T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-07-30T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-02-13T00:0
0:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2012-08-16T00:00:00Z"), "grade" : "A", "score" : 2 }, { "date" : ISODate("2011-08-17T00:00:00Z"), "grade" : "A", "score" : 11 } ], "name" : "Regina Cat
erers", "restaurant_id" : "40356649" }
{ "_id" : ObjectId("6205492c0aaafc1f94671015"), "address" : { "building" : "1839", "coord" : [ -73.9482609, 40.6408271 ], "street" : "Nostrand Avenue", "zipcode" : "11226" }, "borough" : "Brooklyn", "cuisine" :
"Ice Cream, Gelato, Yogurt, Ices", "grades" : [ { "date" : ISODate("2014-07-14T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-07-10T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISO
Date("2012-07-11T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2012-02-23T00:00:00Z"), "grade" : "A", "score" : 8 } ], "name" : "Taste The Tropics Ice Cream", "restaurant_id" : "40356731" }
{ "_id" : ObjectId("6205492c0aaafc1f94671016"), "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" : "40357217" }
```

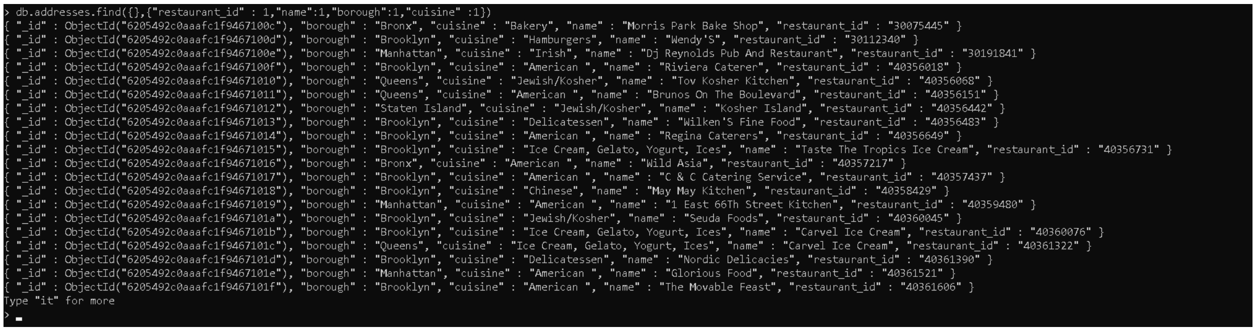
Exercise Questions

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



```
> show collections
addresses
> db.addresses.find()
{ "_id" : ObjectId("6205492c0aaafc1f9467100c"), "address" : { "building" : "1007", "coord" : [ -73.856077, 40.848447 ], "street" : "Morris Park Ave", "zipcode" : "10462" }, "borough" : "Bronx", "cuisine" : "Bake
ry", "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 Sho
p", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100d"), "address" : { "building" : "469", "coord" : [ -73.961704, 40.662942 ], "street" : "Flatbush Avenue", "zipcode" : "11225" }, "borough" : "Brooklyn", "cuisine" : "Ha
mburgers", "grades" : [ { "date" : ISODate("2014-12-30T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2014-07-01T00:00:00Z"), "grade" : "B", "score" : 23 }, { "date" : ISODate("2013-04-30T00:00:0
0Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-05-08T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100e"), "address" : { "building" : "351", "coord" : [ -73.98513559999999, 40.7676919 ], "street" : "West   57 Street", "zipcode" : "10019" }, "borough" : "Manhattan", "cui
sine" : "Irish", "grades" : [ { "date" : ISODate("2014-09-06T00:00:00Z"), "grade" : "A", "score" : 2 }, { "date" : ISODate("2013-07-22T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2012-07-31T0
0:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2011-12-29T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "Dj Reynolds Pub And Restaurant", "restaurant_id" : "30191841" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100f"), "address" : { "building" : "2780", "coord" : [ -73.98241999999999, 40.579505 ], "street" : "Stillwell Avenue", "zipcode" : "11224" }, "borough" : "Brooklyn", "cuis
ine" : "American ", "grades" : [ { "date" : ISODate("2014-06-10T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2013-06-05T00:00:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2012-04-13
T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2011-10-12T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "Riviera Caterer", "restaurant_id" : "40356018" }
{ "_id" : ObjectId("6205492c0aaafc1f94671010"), "address" : { "building" : "97-22", "coord" : [ -73.8601152, 40.7311739 ], "street" : "63 Road", "zipcode" : "11374" }, "borough" : "Queens", "cuisine" : "Jewish/K
osher", "grades" : [ { "date" : ISODate("2014-11-24T00:00:00Z"), "grade" : "Z", "score" : 20 }, { "date" : ISODate("2013-01-17T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2012-08-02T00:00:00Z
"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2011-12-15T00:00:00Z"), "grade" : "B", "score" : 25 } ], "name" : "Tov Kosher Kitchen", "restaurant_id" : "40356068" }
{ "_id" : ObjectId("6205492c0aaafc1f94671011"), "address" : { "building" : "8825", "coord" : [ -73.8803827, 40.7643124 ], "street" : "Astoria Boulevard", "zipcode" : "11369" }, "borough" : "Queens", "cuisine" :
"American ", "grades" : [ { "date" : ISODate("2014-11-15T00:00:00Z"), "grade" : "Z", "score" : 38 }, { "date" : ISODate("2014-05-02T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2013-03-02T00:0
0:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2012-02-10T00:00:00Z"), "grade" : "A", "score" : 13 } ], "name" : "Brunos On The Boulevard", "restaurant_id" : "40356151" }
{ "_id" : ObjectId("6205492c0aaafc1f94671012"), "address" : { "building" : "2206", "coord" : [ -74.1377286, 40.6119572 ], "street" : "Victory Boulevard", "zipcode" : "10314" }, "borough" : "Staten Island", "cuis
ine" : "Jewish/Kosher", "grades" : [ { "date" : ISODate("2014-10-06T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2014-05-20T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-
04-04T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-01-24T00:00:00Z"), "grade" : "A", "score" : 9 } ], "name" : "Kosher Island", "restaurant_id" : "40356442" }
{ "_id" : ObjectId("6205492c0aaafc1f94671013"), "address" : { "building" : "7114", "coord" : [ -73.9068506, 40.6199034 ], "street" : "Avenue U", "zipcode" : "11234" }, "borough" : "Brooklyn", "cuisine" : "Delica
tessen", "grades" : [ { "date" : ISODate("2014-05-29T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2014-01-14T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2013-08-03T00:00:00
Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2012-07-18T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2012-03-09T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2011-1
0-14T00:00:00Z"), "grade" : "A", "score" : 9 } ], "name" : "Wilken'S Fine Food", "restaurant_id" : "40356483" }
{ "_id" : ObjectId("6205492c0aaafc1f94671014"), "address" : { "building" : "6409", "coord" : [ -74.00528899999999, 40.628886 ], "street" : "11 Avenue", "zipcode" : "11219" }, "borough" : "Brooklyn", "cuisine" :
"American ", "grades" : [ { "date" : ISODate("2014-07-18T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-07-30T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-02-13T00:0
0:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2012-08-16T00:00:00Z"), "grade" : "A", "score" : 2 }, { "date" : ISODate("2011-08-17T00:00:00Z"), "grade" : "A", "score" : 11 } ], "name" : "Regina Cat
erers", "restaurant_id" : "40356649" }
{ "_id" : ObjectId("6205492c0aaafc1f94671015"), "address" : { "building" : "1839", "coord" : [ -73.9482609, 40.6408271 ], "street" : "Nostrand Avenue", "zipcode" : "11226" }, "borough" : "Brooklyn", "cuisine" :
"Ice Cream, Gelato, Yogurt, Ices", "grades" : [ { "date" : ISODate("2014-07-14T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-07-10T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISO
Date("2012-07-11T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2012-02-23T00:00:00Z"), "grade" : "A", "score" : 8 } ], "name" : "Taste The Tropics Ice Cream", "restaurant_id" : "40356731" }
{ "_id" : ObjectId("6205492c0aaafc1f94671016"), "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" : "40357217" }
{ "_id" : ObjectId("6205492c0aaafc1f94671017"), "address" : { "building" : "7715", "coord" : [ -73.9973325, 40.61174889999999 ], "street" : "18 Avenue", "zipcode" : "11214" }, "borough" : "Brooklyn", "cuisine" :
 "American ", "grades" : [ { "date" : ISODate("2014-04-16T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2013-04-23T00:00:00Z"), "grade" : "A", "score" : 2 }, { "date" : ISODate("2012-04-24T00:00
:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2011-12-16T00:00:00Z"), "grade" : "A", "score" : 2 } ], "name" : "C & C Catering Service", "restaurant_id" : "40357437" }
{ "_id" : ObjectId("6205492c0aaafc1f94671018"), "address" : { "building" : "1269", "coord" : [ -73.871194, 40.6730975 ], "street" : "Sutter Avenue", "zipcode" : "11208" }, "borough" : "Brooklyn", "cuisine" : "Ch
inese", "grades" : [ { "date" : ISODate("2014-09-16T00:00:00Z"), "grade" : "B", "score" : 21 }, { "date" : ISODate("2013-08-28T00:00:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2013-04-02T00:00:00Z"
), "grade" : "C", "score" : 56 }, { "date" : ISODate("2012-08-15T00:00:00Z"), "grade" : "B", "score" : 27 }, { "date" : ISODate("2012-03-28T00:00:00Z"), "grade" : "B", "score" : 27 } ], "name" : "May May Kitchen
", "restaurant_id" : "40358429" }
{ "_id" : ObjectId("6205492c0aaafc1f94671019"), "address" : { "building" : "1", "coord" : [ -73.96926909999999, 40.7685235 ], "street" : "East   66 Street", "zipcode" : "10065" }, "borough" : "Manhattan", "cuisi
ne" : "American ", "grades" : [ { "date" : ISODate("2014-05-07T00:00:00Z"), "grade" : "A", "score" : 3 }, { "date" : ISODate("2013-05-03T00:00:00Z"), "grade" : "A", "score" : 4 }, { "date" : ISODate("2012-04-30T
00:00:00Z"), "grade" : "A", "score" : 6 }, { "date" : ISODate("2011-12-27T00:00:00Z"), "grade" : "A", "score" : 0 } ], "name" : "1 East 66Th Street Kitchen", "restaurant_id" : "40359480" }
{ "_id" : ObjectId("6205492c0aaafc1f9467101a"), "address" : { "building" : "705", "coord" : [ -73.9653967, 40.6064339 ], "street" : "Kings Highway", "zipcode" : "11223" }, "borough" : "Brooklyn", "cuisine" : "Je
```

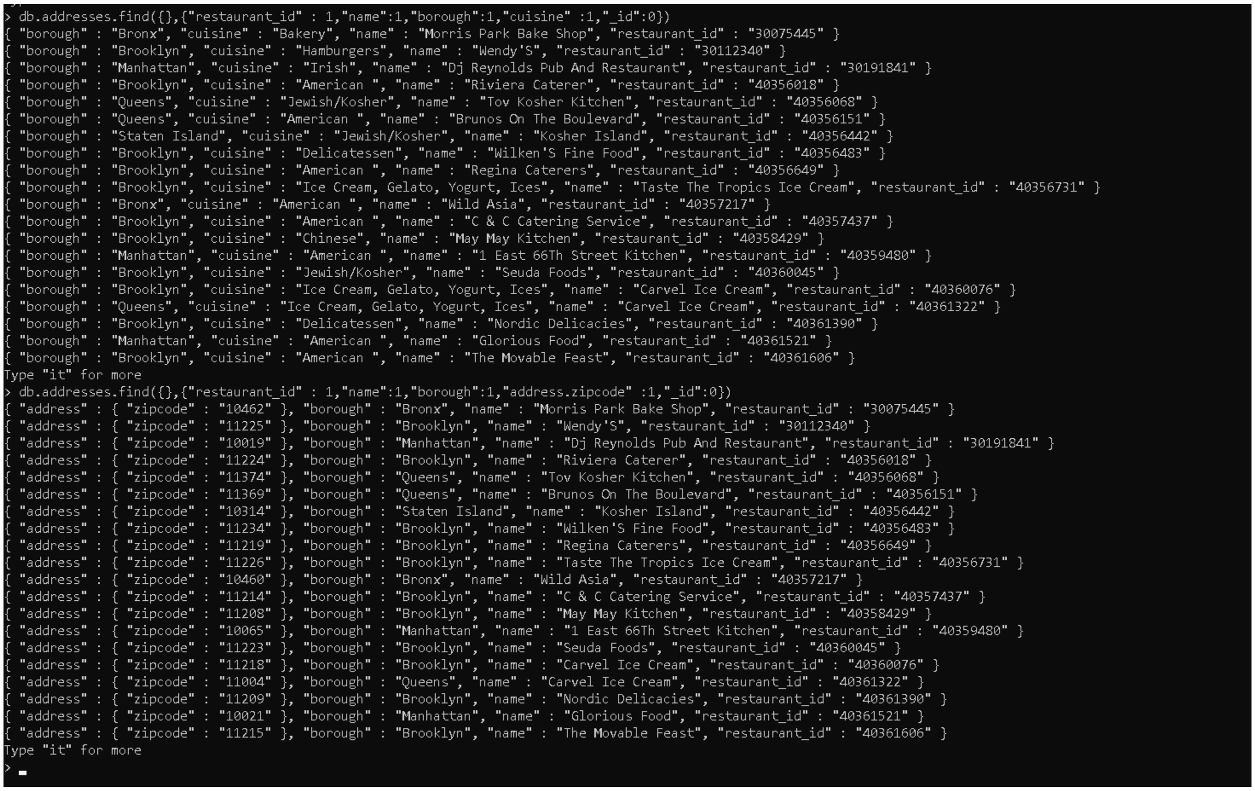
2. Write a MongoDB query to display the fields restaurant_id, name, borough and cuisine for all the documents in the collection addresses.



```
> db.addresses.find({},{"restaurant_id" : 1,"name":1,"borough":1,"cuisine" :1})
{ "_id" : ObjectId("6205492c0aaafc1f9467100c"), "borough" : "Bronx", "cuisine" : "Bakery", "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100d"), "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100e"), "borough" : "Manhattan", "cuisine" : "Irish", "name" : "Dj Reynolds Pub And Restaurant", "restaurant_id" : "30191841" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100f"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "Riviera Caterer", "restaurant_id" : "40356018" }
{ "_id" : ObjectId("6205492c0aaafc1f94671010"), "borough" : "Queens", "cuisine" : "Jewish/Kosher", "name" : "Tov Kosher Kitchen", "restaurant_id" : "40356068" }
{ "_id" : ObjectId("6205492c0aaafc1f94671011"), "borough" : "Queens", "cuisine" : "American ", "name" : "Brunos On The Boulevard", "restaurant_id" : "40356151" }
{ "_id" : ObjectId("6205492c0aaafc1f94671012"), "borough" : "Staten Island", "cuisine" : "Jewish/Kosher", "name" : "Kosher Island", "restaurant_id" : "40356442" }
{ "_id" : ObjectId("6205492c0aaafc1f94671013"), "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Wilken'S Fine Food", "restaurant_id" : "40356483" }
{ "_id" : ObjectId("6205492c0aaafc1f94671014"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "Regina Caterers", "restaurant_id" : "40356649" }
{ "_id" : ObjectId("6205492c0aaafc1f94671015"), "borough" : "Brooklyn", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Taste The Tropics Ice Cream", "restaurant_id" : "40356731" }
{ "_id" : ObjectId("6205492c0aaafc1f94671016"), "borough" : "Bronx", "cuisine" : "American ", "name" : "Wild Asia", "restaurant_id" : "40357217" }
{ "_id" : ObjectId("6205492c0aaafc1f94671017"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "C & C Catering Service", "restaurant_id" : "40357437" }
{ "_id" : ObjectId("6205492c0aaafc1f94671018"), "borough" : "Brooklyn", "cuisine" : "Chinese", "name" : "May May Kitchen", "restaurant_id" : "40358429" }
{ "_id" : ObjectId("6205492c0aaafc1f94671019"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "1 East 66Th Street Kitchen", "restaurant_id" : "40359480" }
{ "_id" : ObjectId("6205492c0aaafc1f9467101a"), "borough" : "Brooklyn", "cuisine" : "Jewish/Kosher", "name" : "Seuda Foods", "restaurant_id" : "40360045" }
{ "_id" : ObjectId("6205492c0aaafc1f9467101b"), "borough" : "Brooklyn", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Carvel Ice Cream", "restaurant_id" : "40360076" }
{ "_id" : ObjectId("6205492c0aaafc1f9467101c"), "borough" : "Queens", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "Carvel Ice Cream", "restaurant_id" : "40361322" }
{ "_id" : ObjectId("6205492c0aaafc1f9467101d"), "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Nordic Delicacies", "restaurant_id" : "40361390" }
{ "_id" : ObjectId("6205492c0aaafc1f9467101e"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Glorious Food", "restaurant_id" : "40361521" }
{ "_id" : ObjectId("6205492c0aaafc1f9467101f"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "The Movable Feast", "restaurant_id" : "40361606" }
Type "it" for more
> 
```

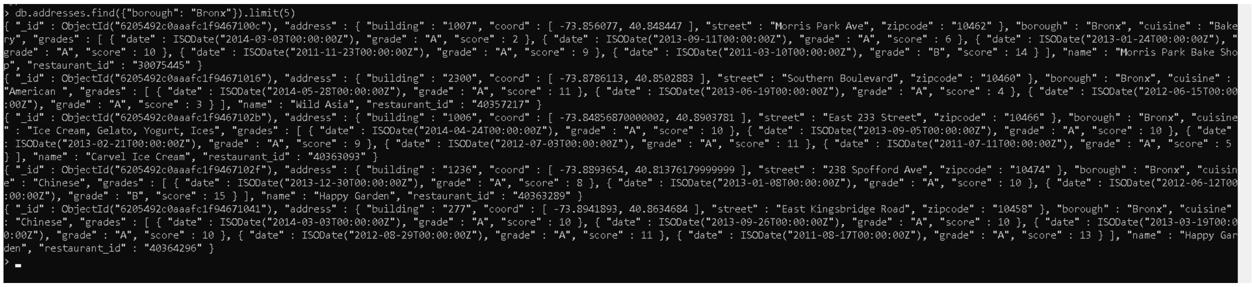
3. Write a MongoDB query to display the fields restaurant_id, name, borough and cuisine, but exclude the field _id for all the documents in the collection addresses.

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 addresses.



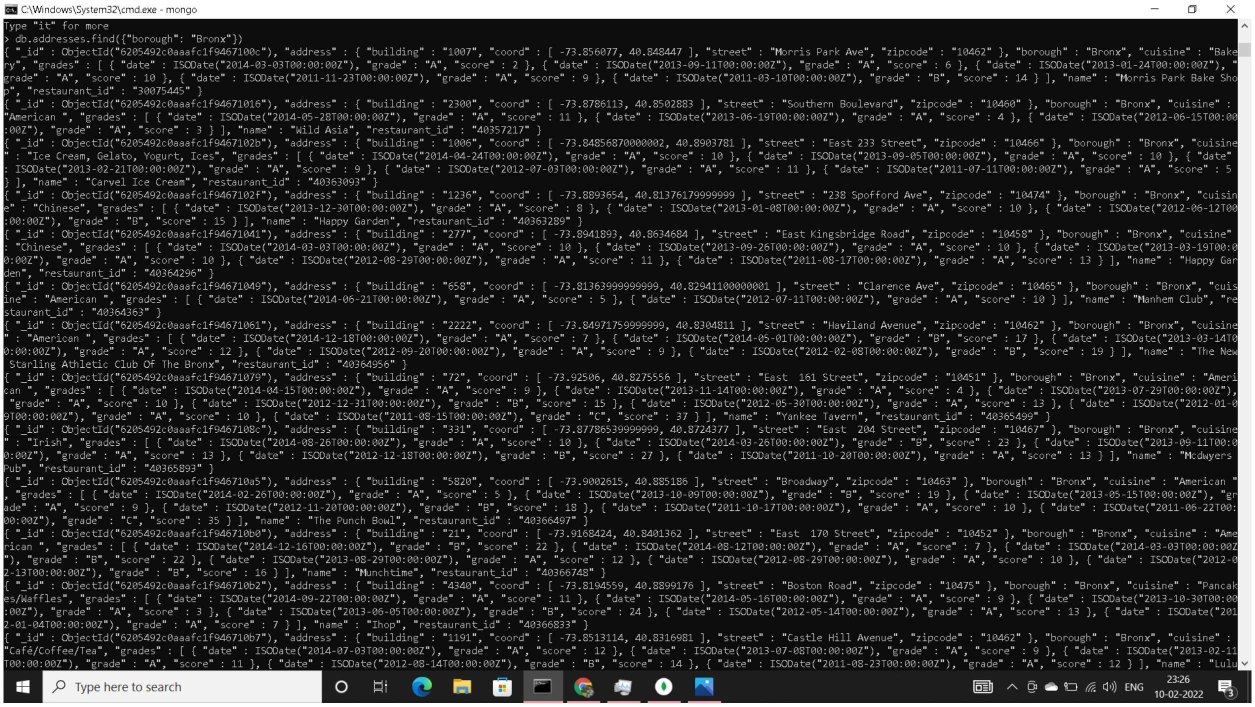
```
> db.addresses.find({},{"restaurant_id" : 1,"name":1,"borough":1,"cuisine" :1,"_id":0})
{ "borough" : "Bronx", "cuisine" : "Bakery", "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "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
> db.addresses.find({},{"restaurant_id" : 1,"name":1,"borough":1,"address.zipcode" :1,"_id":0})
{ "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.



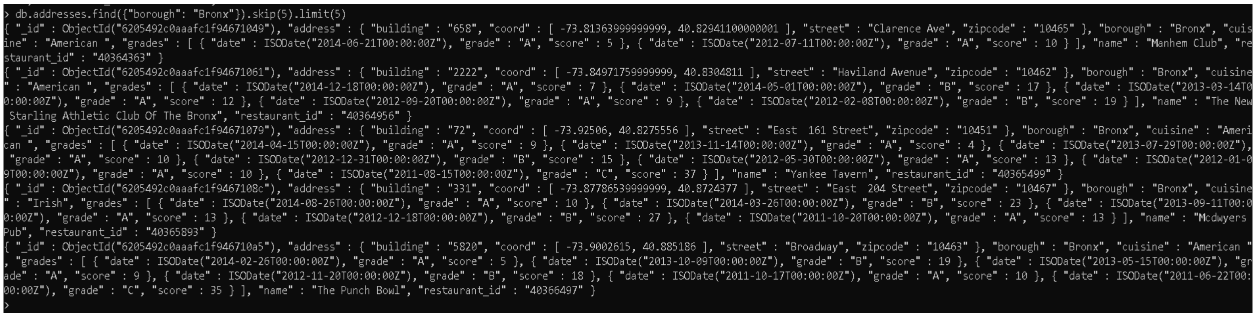
```
> db.addresses.find({"borough": "Bronx"}).limit(5)
{ "_id" : ObjectId("6205492c0aaafc1f9467100c"), "address" : { "building" : "1007", "coord" : [ -73.856077, 40.848447 ], "street" : "Morris Park Ave", "zipcode" : "10462" }, "borough" : "Bronx", "cuisine" : "Bake
ry", "grades" : [ { "date" : ISODate("2014-03-03T00:00:00Z"), "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 Sho
p", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6205492c0aaafc1f94671016"), "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" : "40357217" }
{ "_id" : ObjectId("6205492c0aaafc1f9467102b"), "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" : ISODate("2012-07-03T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2011-07-11T00:00:00Z"), "grade" : "A", "score" : 5
} ], "name" : "Carvel Ice Cream", "restaurant_id" : "40363093" }
{ "_id" : ObjectId("6205492c0aaafc1f9467102f"), "address" : { "building" : "1236", "coord" : [ -73.8893654, 40.81376179999999 ], "street" : "238 Spofford Ave", "zipcode" : "10474" }, "borough" : "Bronx", "cuisin
e" : "Chinese", "grades" : [ { "date" : ISODate("2013-12-30T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2013-01-08T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2012-06-12T00
:00:00Z"), "grade" : "B", "score" : 15 } ], "name" : "Happy Garden", "restaurant_id" : "40363289" }
{ "_id" : ObjectId("6205492c0aaafc1f94671041"), "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("2013-03-19T00:0
0:00Z"), "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 Gar
den", "restaurant_id" : "40364296" }
> 
```

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



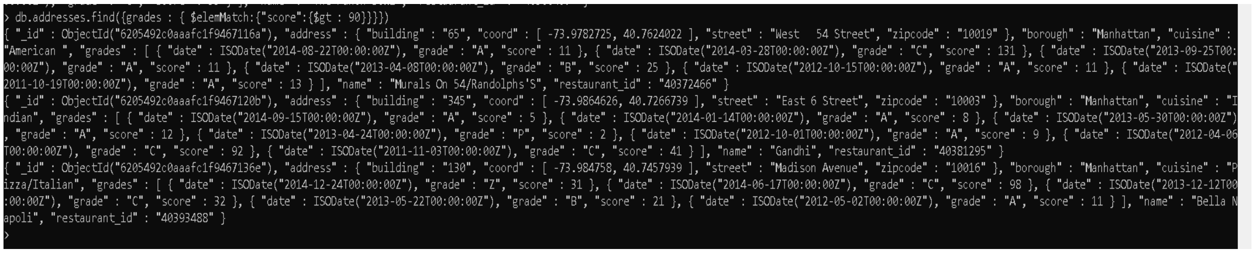
```
Type "it" for more
> db.addresses.find({"borough": "Bronx"})
{ "_id" : ObjectId("6205492c0aaafc1f9467100c"), "address" : { "building" : "1007", "coord" : [ -73.856077, 40.848447 ], "street" : "Morris Park Ave", "zipcode" : "10462" }, "borough" : "Bronx", "cuisine" : "Bake
ry", "grades" : [ { "date" : 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 Sho
p", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6205492c0aaafc1f94671016"), "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" : "40357217" }
{ "_id" : ObjectId("6205492c0aaafc1f9467102b"), "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" : ISODate("2012-07-03T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2011-07-11T00:00:00Z"), "grade" : "A", "score" : 5
} ], "name" : "Carvel Ice Cream", "restaurant_id" : "40363093" }
{ "_id" : ObjectId("6205492c0aaafc1f9467102f"), "address" : { "building" : "1236", "coord" : [ -73.8893654, 40.81376179999999 ], "street" : "238 Spofford Ave", "zipcode" : "10474" }, "borough" : "Bronx", "cuisin
e" : "Chinese", "grades" : [ { "date" : ISODate("2013-12-30T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2013-01-08T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2012-06-12T00
:00:00Z"), "grade" : "B", "score" : 15 } ], "name" : "Happy Garden", "restaurant_id" : "40363289" }
{ "_id" : ObjectId("6205492c0aaafc1f94671041"), "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("2013-03-19T00:0
0:00Z"), "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 Gar
den", "restaurant_id" : "40364296" }
{ "_id" : ObjectId("6205492c0aaafc1f94671049"), "address" : { "building" : "658", "coord" : [ -73.81363999999999, 40.82941100000001 ], "street" : "Clarence Ave", "zipcode" : "10465" }, "borough" : "Bronx", "cuis
ine" : "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", "re
staurant_id" : "40364363" }
{ "_id" : ObjectId("6205492c0aaafc1f94671061"), "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("2014-05-01T00:00:00Z"), "grade" : "B", "score" : 17 }, { "date" : ISODate("2013-03-14T0
0:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-09-20T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2012-02-08T00:00:00Z"), "grade" : "B", "score" : 19 } ], "name" : "The New
 Starling Athletic Club Of The Bronx", "restaurant_id" : "40364956" }
{ "_id" : ObjectId("6205492c0aaafc1f94671079"), "address" : { "building" : "72", "coord" : [ -73.92506, 40.8275556 ], "street" : "East  161 Street", "zipcode" : "10451" }, "borough" : "Bronx", "cuisine" : "Ameri
can ", "grades" : [ { "date" : ISODate("2014-04-15T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2013-11-14T00:00:00Z"), "grade" : "A", "score" : 4 }, { "date" : ISODate("2013-07-29T00:00:00Z"),
 "grade" : "A", "score" : 10 }, { "date" : ISODate("2012-12-31T00:00:00Z"), "grade" : "B", "score" : 15 }, { "date" : ISODate("2012-05-30T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2012-01-0
9T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2011-08-15T00:00:00Z"), "grade" : "C", "score" : 37 } ], "name" : "Yankee Tavern", "restaurant_id" : "40365499" }
{ "_id" : ObjectId("6205492c0aaafc1f9467108c"), "address" : { "building" : "331", "coord" : [ -73.87786539999999, 40.8724377 ], "street" : "East  204 Street", "zipcode" : "10467" }, "borough" : "Bronx", "cuisine
" : "Irish", "grades" : [ { "date" : ISODate("2014-08-26T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2014-03-26T00:00:00Z"), "grade" : "B", "score" : 23 }, { "date" : ISODate("2013-09-11T00:0
0:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2012-12-18T00:00:00Z"), "grade" : "B", "score" : 27 }, { "date" : ISODate("2011-10-20T00:00:00Z"), "grade" : "A", "score" : 13 } ], "name" : "Mcdwyers
Pub", "restaurant_id" : "40365893" }
{ "_id" : ObjectId("6205492c0aaafc1f946710a5"), "address" : { "building" : "5820", "coord" : [ -73.9002615, 40.885186 ], "street" : "Broadway", "zipcode" : "10463" }, "borough" : "Bronx", "cuisine" : "American "
, "grades" : [ { "date" : ISODate("2014-02-26T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2013-10-09T00:00:00Z"), "grade" : "B", "score" : 19 }, { "date" : ISODate("2013-05-15T00:00:00Z"), "gr
ade" : "A", "score" : 9 }, { "date" : ISODate("2012-11-20T00:00:00Z"), "grade" : "B", "score" : 18 }, { "date" : ISODate("2011-10-17T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2011-06-22T00:
00:00Z"), "grade" : "C", "score" : 35 } ], "name" : "The Punch Bowl", "restaurant_id" : "40366497" }
{ "_id" : ObjectId("6205492c0aaafc1f946710b0"), "address" : { "building" : "21", "coord" : [ -73.9168424, 40.8401362 ], "street" : "East  170 Street", "zipcode" : "10452" }, "borough" : "Bronx", "cuisine" : "Ame
rican ", "grades" : [ { "date" : ISODate("2014-12-16T00:00:00Z"), "grade" : "B", "score" : 22 }, { "date" : ISODate("2014-08-12T00:00:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2014-03-03T00:00:00Z
"), "grade" : "B", "score" : 22 }, { "date" : ISODate("2013-08-29T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-08-29T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2012-0
2-13T00:00:00Z"), "grade" : "B", "score" : 16 } ], "name" : "Munchtime", "restaurant_id" : "40366748" }
{ "_id" : ObjectId("6205492c0aaafc1f946710b2"), "address" : { "building" : "4340", "coord" : [ -73.8194559, 40.8899176 ], "street" : "Boston Road", "zipcode" : "10475" }, "borough" : "Bronx", "cuisine" : "Pancak
es/Waffles", "grades" : [ { "date" : ISODate("2014-09-22T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2014-05-16T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2013-10-30T00:00
:00Z"), "grade" : "A", "score" : 3 }, { "date" : ISODate("2013-06-05T00:00:00Z"), "grade" : "B", "score" : 24 }, { "date" : ISODate("2012-05-14T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("201
2-01-04T00:00:00Z"), "grade" : "A", "score" : 7 } ], "name" : "Ihop", "restaurant_id" : "40366833" }
{ "_id" : ObjectId("6205492c0aaafc1f946710b7"), "address" : { "building" : "1191", "coord" : [ -73.8513114, 40.8316981 ], "street" : "Castle Hill Avenue", "zipcode" : "10462" }, "borough" : "Bronx", "cuisine" :
"Café/Coffee/Tea", "grades" : [ { "date" : ISODate("2014-07-03T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-07-08T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2013-02-11
T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2012-08-14T00:00:00Z"), "grade" : "B", "score" : 14 }, { "date" : ISODate("2011-08-23T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "Lulu
```

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



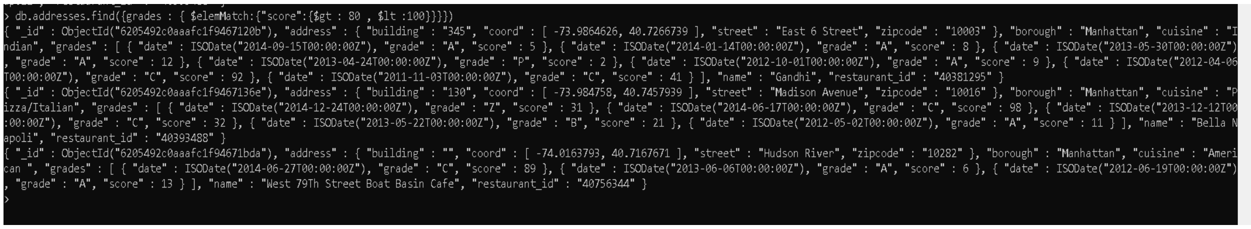
```
> db.addresses.find({"borough": "Bronx"}).skip(5).limit(5)
{ "_id" : ObjectId("6205492c0aaafc1f94671049"), "address" : { "building" : "658", "coord" : [ -73.81363999999999, 40.82941100000001 ], "street" : "Clarence Ave", "zipcode" : "10465" }, "borough" : "Bronx", "cuis
ine" : "American ", "grades" : [ { "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", "re
staurant_id" : "40364363" }
{ "_id" : ObjectId("6205492c0aaafc1f94671061"), "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("2014-05-01T00:00:00Z"), "grade" : "B", "score" : 17 }, { "date" : ISODate("2013-03-14T0
0:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-09-20T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2012-02-08T00:00:00Z"), "grade" : "B", "score" : 19 } ], "name" : "The New
 Starling Athletic Club Of The Bronx", "restaurant_id" : "40364956" }
{ "_id" : ObjectId("6205492c0aaafc1f94671079"), "address" : { "building" : "72", "coord" : [ -73.92506, 40.8275556 ], "street" : "East  161 Street", "zipcode" : "10451" }, "borough" : "Bronx", "cuisine" : "Ameri
can ", "grades" : [ { "date" : ISODate("2014-04-15T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2013-11-14T00:00:00Z"), "grade" : "A", "score" : 4 }, { "date" : ISODate("2013-07-29T00:00:00Z"),
 "grade" : "A", "score" : 10 }, { "date" : ISODate("2012-12-31T00:00:00Z"), "grade" : "B", "score" : 15 }, { "date" : ISODate("2012-05-30T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2012-01-0
9T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2011-08-15T00:00:00Z"), "grade" : "C", "score" : 37 } ], "name" : "Yankee Tavern", "restaurant_id" : "40365499" }
{ "_id" : ObjectId("6205492c0aaafc1f9467108c"), "address" : { "building" : "331", "coord" : [ -73.87786539999999, 40.8724377 ], "street" : "East  204 Street", "zipcode" : "10467" }, "borough" : "Bronx", "cuisine
" : "Irish", "grades" : [ { "date" : ISODate("2014-08-26T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2014-03-26T00:00:00Z"), "grade" : "B", "score" : 23 }, { "date" : ISODate("2013-09-11T00:0
0:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2012-12-18T00:00:00Z"), "grade" : "B", "score" : 27 }, { "date" : ISODate("2011-10-20T00:00:00Z"), "grade" : "A", "score" : 13 } ], "name" : "Mcdwyers
Pub", "restaurant_id" : "40365893" }
{ "_id" : ObjectId("6205492c0aaafc1f946710a5"), "address" : { "building" : "5820", "coord" : [ -73.9002615, 40.885186 ], "street" : "Broadway", "zipcode" : "10463" }, "borough" : "Bronx", "cuisine" : "American "
, "grades" : [ { "date" : ISODate("2014-02-26T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2013-10-09T00:00:00Z"), "grade" : "B", "score" : 19 }, { "date" : ISODate("2013-05-15T00:00:00Z"), "gr
ade" : "A", "score" : 9 }, { "date" : ISODate("2012-11-20T00:00:00Z"), "grade" : "B", "score" : 18 }, { "date" : ISODate("2011-10-17T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2011-06-22T00:
00:00Z"), "grade" : "C", "score" : 35 } ], "name" : "The Punch Bowl", "restaurant_id" : "40366497" }
>
```

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



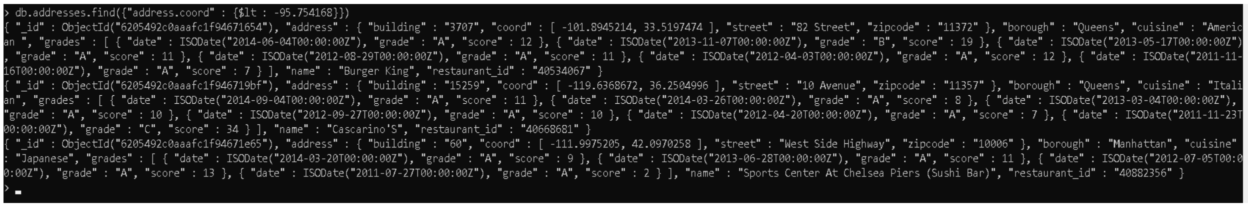
```
> db.addresses.find({grades : { $elemMatch:{"score":{$gt : 90}}}})
{ "_id" : ObjectId("6205492c0aaafc1f9467116a"), "address" : { "building" : "65", "coord" : [ -73.9782725, 40.7624022 ], "street" : "West   54 Street", "zipcode" : "10019" }, "borough" : "Manhattan", "cuisine" :
"American ", "grades" : [ { "date" : ISODate("2014-08-22T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2014-03-28T00:00:00Z"), "grade" : "C", "score" : 131 }, { "date" : ISODate("2013-09-25T00:
00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2013-04-08T00:00:00Z"), "grade" : "B", "score" : 25 }, { "date" : ISODate("2012-10-15T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("
2011-10-19T00:00:00Z"), "grade" : "A", "score" : 13 } ], "name" : "Murals On 54/Randolphs'S", "restaurant_id" : "40372466" }
{ "_id" : ObjectId("6205492c0aaafc1f9467120b"), "address" : { "building" : "345", "coord" : [ -73.9864626, 40.7266739 ], "street" : "East 6 Street", "zipcode" : "10003" }, "borough" : "Manhattan", "cuisine" : "I
ndian", "grades" : [ { "date" : ISODate("2014-09-15T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2014-01-14T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2013-05-30T00:00:00Z")
, "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-04-24T00:00:00Z"), "grade" : "P", "score" : 2 }, { "date" : ISODate("2012-10-01T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2012-04-06
T00:00:00Z"), "grade" : "C", "score" : 92 }, { "date" : ISODate("2011-11-03T00:00:00Z"), "grade" : "C", "score" : 41 } ], "name" : "Gandhi", "restaurant_id" : "40381295" }
{ "_id" : ObjectId("6205492c0aaafc1f9467136e"), "address" : { "building" : "130", "coord" : [ -73.984758, 40.7457939 ], "street" : "Madison Avenue", "zipcode" : "10016" }, "borough" : "Manhattan", "cuisine" : "P
izza/Italian", "grades" : [ { "date" : ISODate("2014-12-24T00:00:00Z"), "grade" : "Z", "score" : 31 }, { "date" : ISODate("2014-06-17T00:00:00Z"), "grade" : "C", "score" : 98 }, { "date" : ISODate("2013-12-12T00
:00:00Z"), "grade" : "C", "score" : 32 }, { "date" : ISODate("2013-05-22T00:00:00Z"), "grade" : "B", "score" : 21 }, { "date" : ISODate("2012-05-02T00:00:00Z"), "grade" : "A", "score" : 11 } ], "name" : "Bella N
apoli", "restaurant_id" : "40393488" }
>
```

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



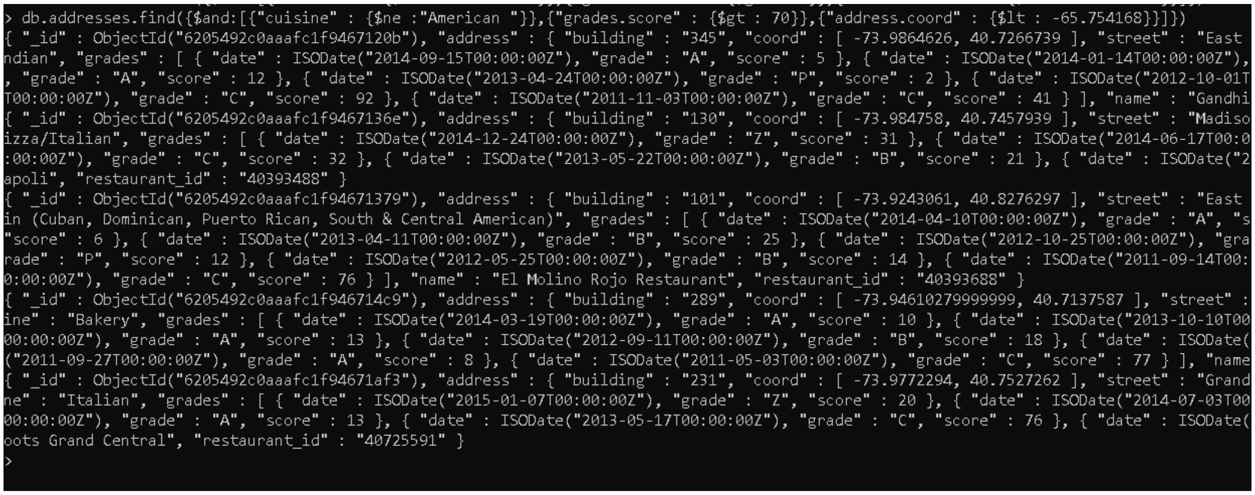
```
> db.addresses.find({grades : { $elemMatch:{"score":{$gt : 80 , $lt :100}}}})
{ "_id" : ObjectId("6205492c0aaafc1f9467120b"), "address" : { "building" : "345", "coord" : [ -73.9864626, 40.7266739 ], "street" : "East 6 Street", "zipcode" : "10003" }, "borough" : "Manhattan", "cuisine" : "I
ndian", "grades" : [ { "date" : ISODate("2014-09-15T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2014-01-14T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2013-05-30T00:00:00Z")
, "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-04-24T00:00:00Z"), "grade" : "P", "score" : 2 }, { "date" : ISODate("2012-10-01T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2012-04-06
T00:00:00Z"), "grade" : "C", "score" : 92 }, { "date" : ISODate("2011-11-03T00:00:00Z"), "grade" : "C", "score" : 41 } ], "name" : "Gandhi", "restaurant_id" : "40381295" }
{ "_id" : ObjectId("6205492c0aaafc1f9467136e"), "address" : { "building" : "130", "coord" : [ -73.984758, 40.7457939 ], "street" : "Madison Avenue", "zipcode" : "10016" }, "borough" : "Manhattan", "cuisine" : "P
izza/Italian", "grades" : [ { "date" : ISODate("2014-12-24T00:00:00Z"), "grade" : "Z", "score" : 31 }, { "date" : ISODate("2014-06-17T00:00:00Z"), "grade" : "C", "score" : 98 }, { "date" : ISODate("2013-12-12T00
:00:00Z"), "grade" : "C", "score" : 32 }, { "date" : ISODate("2013-05-22T00:00:00Z"), "grade" : "B", "score" : 21 }, { "date" : ISODate("2012-05-02T00:00:00Z"), "grade" : "A", "score" : 11 } ], "name" : "Bella N
apoli", "restaurant_id" : "40393488" }
{ "_id" : ObjectId("6205492c0aaafc1f94671bda"), "address" : { "building" : "", "coord" : [ -74.0163793, 40.7167671 ], "street" : "Hudson River", "zipcode" : "10282" }, "borough" : "Manhattan", "cuisine" : "Ameri
can ", "grades" : [ { "date" : ISODate("2014-06-27T00:00:00Z"), "grade" : "C", "score" : 89 }, { "date" : ISODate("2013-06-06T00:00:00Z"), "grade" : "A", "score" : 6 }, { "date" : ISODate("2012-06-19T00:00:00Z")
, "grade" : "A", "score" : 13 } ], "name" : "West 79Th Street Boat Basin Cafe", "restaurant_id" : "40756344" }
>
```

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



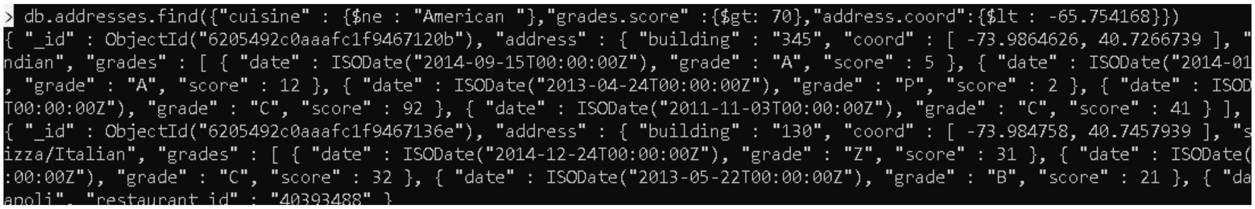
```
> db.addresses.find({"address.coord" : {$lt : -95.754168}})
{ "_id" : ObjectId("6205492c0aaafc1f94671654"), "address" : { "building" : "3707", "coord" : [ -101.8945214, 33.5197474 ], "street" : "82 Street", "zipcode" : "11372" }, "borough" : "Queens", "cuisine" : "Americ
an ", "grades" : [ { "date" : ISODate("2014-06-04T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-11-07T00:00:00Z"), "grade" : "B", "score" : 19 }, { "date" : ISODate("2013-05-17T00:00:00Z")
, "grade" : "A", "score" : 11 }, { "date" : ISODate("2012-08-29T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2012-04-03T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2011-11-
16T00:00:00Z"), "grade" : "A", "score" : 7 } ], "name" : "Burger King", "restaurant_id" : "40534067" }
{ "_id" : ObjectId("6205492c0aaafc1f946719bf"), "address" : { "building" : "15259", "coord" : [ -119.6368672, 36.2504996 ], "street" : "10 Avenue", "zipcode" : "11357" }, "borough" : "Queens", "cuisine" : "Itali
an", "grades" : [ { "date" : ISODate("2014-09-04T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2014-03-26T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2013-03-04T00:00:00Z"),
"grade" : "A", "score" : 10 }, { "date" : ISODate("2012-09-27T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2012-04-20T00:00:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2011-11-23T
00:00:00Z"), "grade" : "C", "score" : 34 } ], "name" : "Cascarino'S", "restaurant_id" : "40668681" }
{ "_id" : ObjectId("6205492c0aaafc1f94671e65"), "address" : { "building" : "60", "coord" : [ -111.9975205, 42.0970258 ], "street" : "West Side Highway", "zipcode" : "10006" }, "borough" : "Manhattan", "cuisine"
: "Japanese", "grades" : [ { "date" : ISODate("2014-03-20T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2013-06-28T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2012-07-05T00:0
0:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2011-07-27T00:00:00Z"), "grade" : "A", "score" : 2 } ], "name" : "Sports Center At Chelsea Piers (Sushi Bar)", "restaurant_id" : "40882356" }
>
```

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.



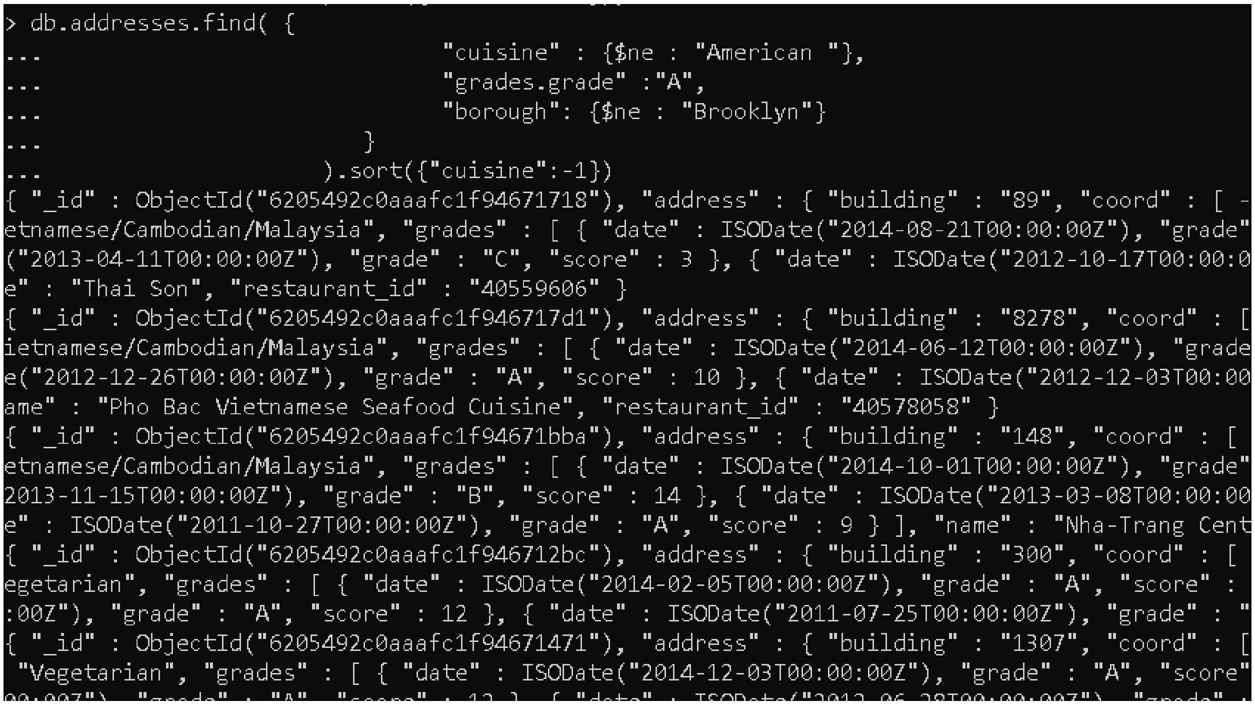
```
> db.addresses.find({$and:[{"cuisine" : {$ne :"American "}},{"grades.score" : {$gt : 70}},{"address.coord" : {$lt : -65.754168}}]})
{ "_id" : ObjectId("6205492c0aaafc1f9467120b"), "address" : { "building" : "345", "coord" : [ -73.9864626, 40.7266739 ], "street" : "East
ndian", "grades" : [ { "date" : ISODate("2014-09-15T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2014-01-14T00:00:00Z"),
, "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-04-24T00:00:00Z"), "grade" : "P", "score" : 2 }, { "date" : ISODate("2012-10-01T
T00:00:00Z"), "grade" : "C", "score" : 92 }, { "date" : ISODate("2011-11-03T00:00:00Z"), "grade" : "C", "score" : 41 } ], "name" : "Gandhi
{ "_id" : ObjectId("6205492c0aaafc1f9467136e"), "address" : { "building" : "130", "coord" : [ -73.984758, 40.7457939 ], "street" : "Madiso
izza/Italian", "grades" : [ { "date" : ISODate("2014-12-24T00:00:00Z"), "grade" : "Z", "score" : 31 }, { "date" : ISODate("2014-06-17T00:0
:00:00Z"), "grade" : "C", "score" : 32 }, { "date" : ISODate("2013-05-22T00:00:00Z"), "grade" : "B", "score" : 21 }, { "date" : ISODate("2
apoli", "restaurant_id" : "40393488" }
{ "_id" : ObjectId("6205492c0aaafc1f94671379"), "address" : { "building" : "101", "coord" : [ -73.9243061, 40.8276297 ], "street" : "East
in (Cuban, Dominican, Puerto Rican, South & Central American)", "grades" : [ { "date" : ISODate("2014-04-10T00:00:00Z"), "grade" : "A", "s
"score" : 6 }, { "date" : ISODate("2013-04-11T00:00:00Z"), "grade" : "B", "score" : 25 }, { "date" : ISODate("2012-10-25T00:00:00Z"), "gra
rade" : "P", "score" : 12 }, { "date" : ISODate("2012-05-25T00:00:00Z"), "grade" : "B", "score" : 14 }, { "date" : ISODate("2011-09-14T00:
0:00:00Z"), "grade" : "C", "score" : 76 } ], "name" : "El Molino Rojo Restaurant", "restaurant_id" : "40393688" }
{ "_id" : ObjectId("6205492c0aaafc1f946714c9"), "address" : { "building" : "289", "coord" : [ -73.94610279999999, 40.7137587 ], "street" :
ine" : "Bakery", "grades" : [ { "date" : ISODate("2014-03-19T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2013-10-10T00
00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2012-09-11T00:00:00Z"), "grade" : "B", "score" : 18 }, { "date" : ISODate(
("2011-09-27T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2011-05-03T00:00:00Z"), "grade" : "C", "score" : 77 } ], "name
{ "_id" : ObjectId("6205492c0aaafc1f94671af3"), "address" : { "building" : "231", "coord" : [ -73.9772294, 40.7527262 ], "street" : "Grand
ne" : "Italian", "grades" : [ { "date" : ISODate("2015-01-07T00:00:00Z"), "grade" : "Z", "score" : 20 }, { "date" : ISODate("2014-07-03T00
00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2013-05-17T00:00:00Z"), "grade" : "C", "score" : 76 }, { "date" : ISODate(
oots Grand Central", "restaurant_id" : "40725591" }
>
```

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.



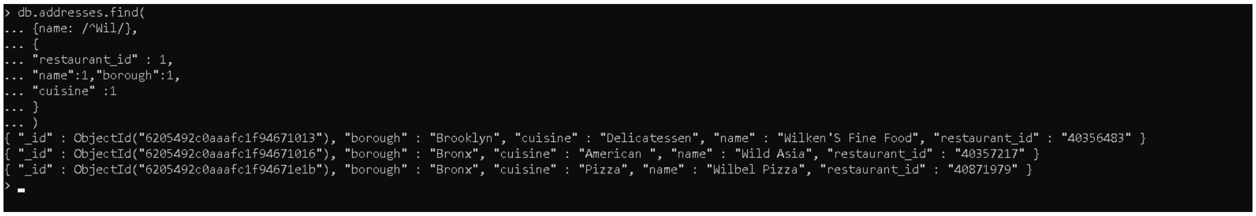
```
> db.addresses.find({"cuisine" : {$ne : "American "},"grades.score" :{$gt: 70},"address.coord":{$lt : -65.754168}})
{ "_id" : ObjectId("6205492c0aaafc1f9467120b"), "address" : { "building" : "345", "coord" : [ -73.9864626, 40.7266739 ], "
ndian", "grades" : [ { "date" : ISODate("2014-09-15T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2014-01
, "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-04-24T00:00:00Z"), "grade" : "P", "score" : 2 }, { "date" : ISOD
T00:00:00Z"), "grade" : "C", "score" : 92 }, { "date" : ISODate("2011-11-03T00:00:00Z"), "grade" : "C", "score" : 41 } ],
{ "_id" : ObjectId("6205492c0aaafc1f9467136e"), "address" : { "building" : "130", "coord" : [ -73.984758, 40.7457939 ], "s
izza/Italian", "grades" : [ { "date" : ISODate("2014-12-24T00:00:00Z"), "grade" : "Z", "score" : 31 }, { "date" : ISODate(
:00:00Z"), "grade" : "C", "score" : 32 }, { "date" : ISODate("2013-05-22T00:00:00Z"), "grade" : "B", "score" : 21 }, { "da
apoli", "restaurant_id" : "40393488" }
```

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.



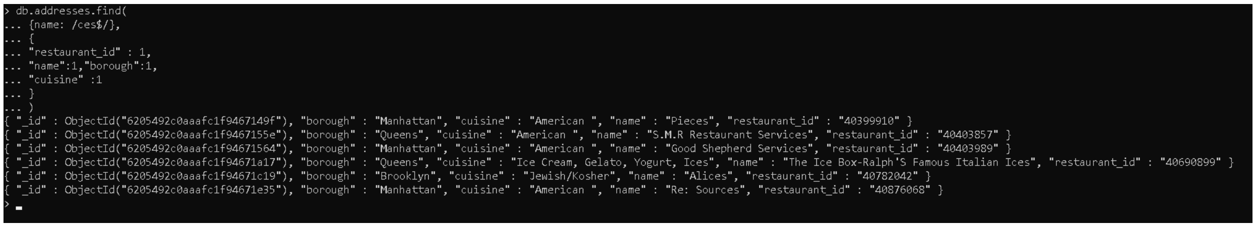
```
> db.addresses.find( {
...                             "cuisine" : {$ne : "American "},
...                             "grades.grade" :"A",
...                             "borough": {$ne : "Brooklyn"}
...                         }
...                     ).sort({"cuisine":-1})
{ "_id" : ObjectId("6205492c0aaafc1f94671718"), "address" : { "building" : "89", "coord" : [ -
etnamese/Cambodian/Malaysia", "grades" : [ { "date" : ISODate("2014-08-21T00:00:00Z"), "grade"
("2013-04-11T00:00:00Z"), "grade" : "C", "score" : 3 }, { "date" : ISODate("2012-10-17T00:00:0
e" : "Thai Son", "restaurant_id" : "40559606" }
{ "_id" : ObjectId("6205492c0aaafc1f946717d1"), "address" : { "building" : "8278", "coord" : [
ietnamese/Cambodian/Malaysia", "grades" : [ { "date" : ISODate("2014-06-12T00:00:00Z"), "grade
e("2012-12-26T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2012-12-03T00:00
ame" : "Pho Bac Vietnamese Seafood Cuisine", "restaurant_id" : "40578058" }
{ "_id" : ObjectId("6205492c0aaafc1f94671bba"), "address" : { "building" : "148", "coord" : [
etnamese/Cambodian/Malaysia", "grades" : [ { "date" : ISODate("2014-10-01T00:00:00Z"), "grade"
2013-11-15T00:00:00Z"), "grade" : "B", "score" : 14 }, { "date" : ISODate("2013-03-08T00:00:00
e" : ISODate("2011-10-27T00:00:00Z"), "grade" : "A", "score" : 9 } ], "name" : "Nha-Trang Cent
{ "_id" : ObjectId("6205492c0aaafc1f946712bc"), "address" : { "building" : "300", "coord" : [
egetarian", "grades" : [ { "date" : ISODate("2014-02-05T00:00:00Z"), "grade" : "A", "score" :
:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2011-07-25T00:00:00Z"), "grade" : "
{ "_id" : ObjectId("6205492c0aaafc1f94671471"), "address" : { "building" : "1307", "coord" : [
 "Vegetarian", "grades" : [ { "date" : ISODate("2014-12-03T00:00:00Z"), "grade" : "A", "score"
00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-06-28T00:00:00Z"), "grade" :
```

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.



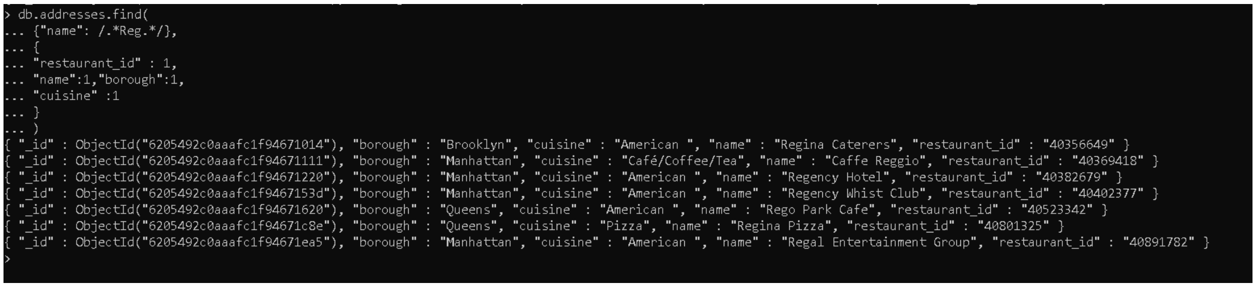
```
> db.addresses.find(
... {name: /^Wil/},
... {
... "restaurant_id" : 1,
... "name":1,"borough":1,
... "cuisine" :1
... }
... )
{ "_id" : ObjectId("6205492c0aaafc1f94671013"), "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Wilken'S Fine Food", "restaurant_id" : "40356483" }
{ "_id" : ObjectId("6205492c0aaafc1f94671016"), "borough" : "Bronx", "cuisine" : "American ", "name" : "Wild Asia", "restaurant_id" : "40357217" }
{ "_id" : ObjectId("6205492c0aaafc1f94671e1b"), "borough" : "Bronx", "cuisine" : "Pizza", "name" : "Wilbel Pizza", "restaurant_id" : "40871979" }
> ▬
```

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.



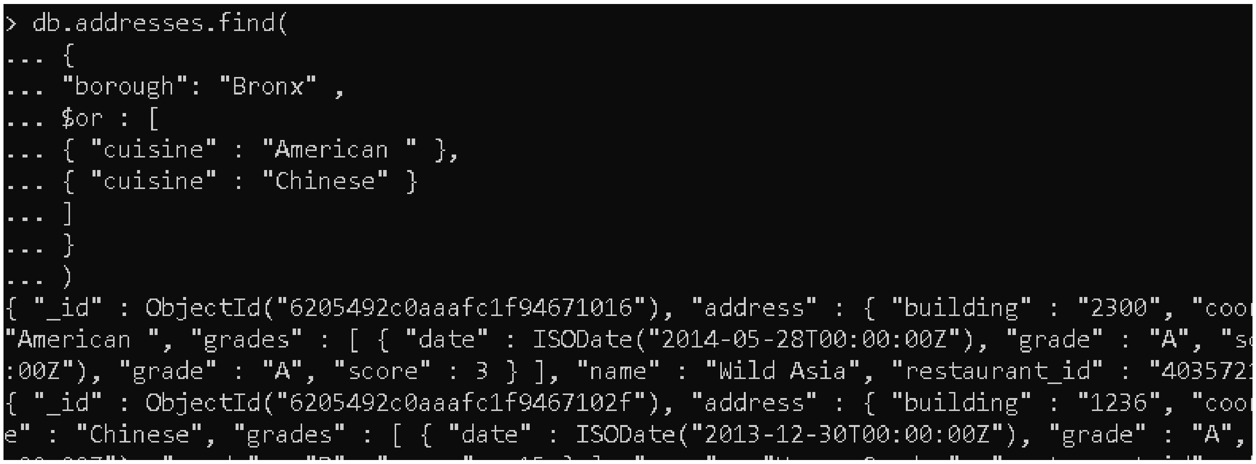
```
> db.addresses.find(
... {name: /ces$/},
... {
... "restaurant_id" : 1,
... "name":1,"borough":1,
... "cuisine" :1
... }
... )
{ "_id" : ObjectId("6205492c0aaafc1f9467149f"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Pieces", "restaurant_id" : "40399910" }
{ "_id" : ObjectId("6205492c0aaafc1f9467155e"), "borough" : "Queens", "cuisine" : "American ", "name" : "S.M.R Restaurant Services", "restaurant_id" : "40403857" }
{ "_id" : ObjectId("6205492c0aaafc1f94671564"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Good Shepherd Services", "restaurant_id" : "40403989" }
{ "_id" : ObjectId("6205492c0aaafc1f94671a17"), "borough" : "Queens", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "name" : "The Ice Box-Ralph'S Famous Italian Ices", "restaurant_id" : "40690899" }
{ "_id" : ObjectId("6205492c0aaafc1f94671c19"), "borough" : "Brooklyn", "cuisine" : "Jewish/Kosher", "name" : "Alices", "restaurant_id" : "40782042" }
{ "_id" : ObjectId("6205492c0aaafc1f94671e35"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Re: Sources", "restaurant_id" : "40876068" }
> ▬
```

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.



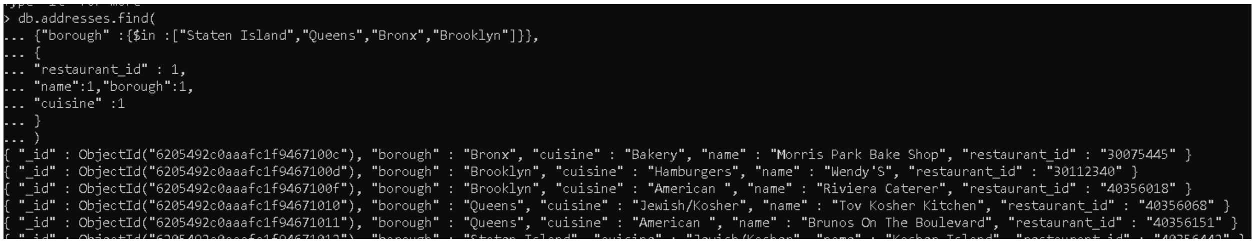
```
> db.addresses.find(
... {"name": /.*Reg.*/},
... {
... "restaurant_id" : 1,
... "name":1,"borough":1,
... "cuisine" :1
... }
... )
{ "_id" : ObjectId("6205492c0aaafc1f94671014"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "Regina Caterers", "restaurant_id" : "40356649" }
{ "_id" : ObjectId("6205492c0aaafc1f94671111"), "borough" : "Manhattan", "cuisine" : "Café/Coffee/Tea", "name" : "Caffe Reggio", "restaurant_id" : "40369418" }
{ "_id" : ObjectId("6205492c0aaafc1f94671220"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Regency Hotel", "restaurant_id" : "40382679" }
{ "_id" : ObjectId("6205492c0aaafc1f9467153d"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Regency Whist Club", "restaurant_id" : "40402377" }
{ "_id" : ObjectId("6205492c0aaafc1f94671620"), "borough" : "Queens", "cuisine" : "American ", "name" : "Rego Park Cafe", "restaurant_id" : "40523342" }
{ "_id" : ObjectId("6205492c0aaafc1f94671c8e"), "borough" : "Queens", "cuisine" : "Pizza", "name" : "Regina Pizza", "restaurant_id" : "40801325" }
{ "_id" : ObjectId("6205492c0aaafc1f94671ea5"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Regal Entertainment Group", "restaurant_id" : "40891782" }
>
```

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



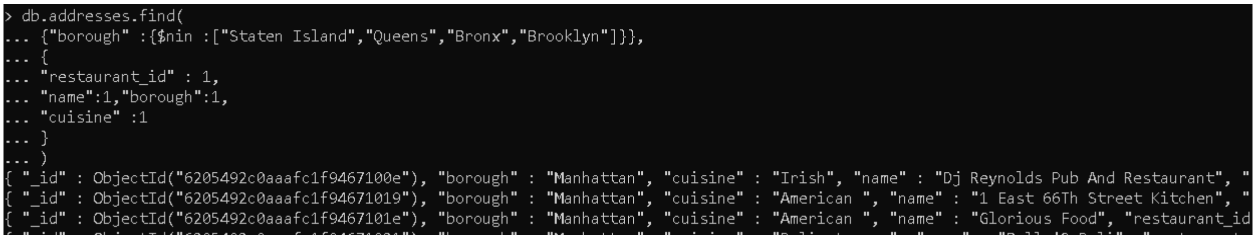
```
> db.addresses.find(
... {
... "borough": "Bronx" ,
... $or : [
... { "cuisine" : "American " },
... { "cuisine" : "Chinese" }
... ]
... }
... )
{ "_id" : ObjectId("6205492c0aaafc1f94671016"), "address" : { "building" : "2300", "coo
"American ", "grades" : [ { "date" : ISODate("2014-05-28T00:00:00Z"), "grade" : "A", "s
:00Z"), "grade" : "A", "score" : 3 } ], "name" : "Wild Asia", "restaurant_id" : "403572
{ "_id" : ObjectId("6205492c0aaafc1f9467102f"), "address" : { "building" : "1236", "coo
e" : "Chinese", "grades" : [ { "date" : ISODate("2013-12-30T00:00:00Z"), "grade" : "A",
```

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.



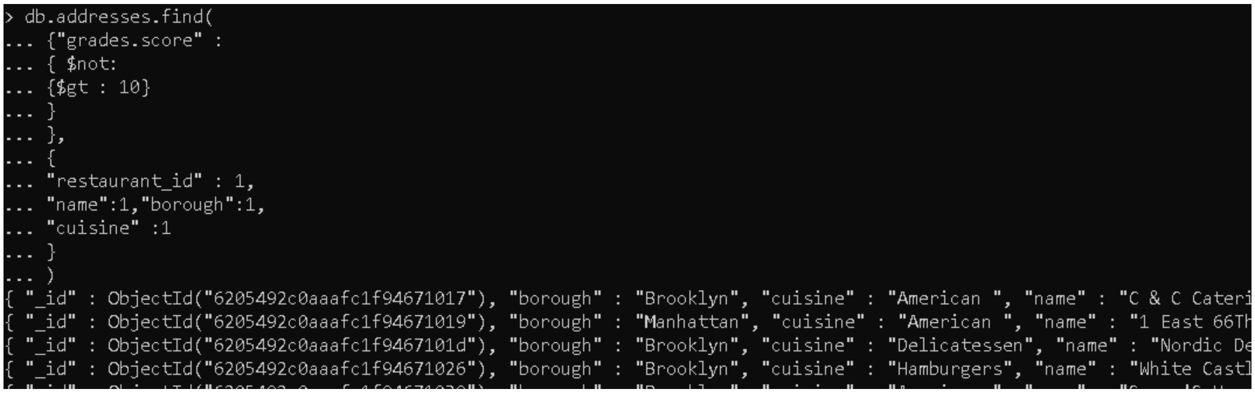
```
> db.addresses.find(
... {"borough" :{$in :["Staten Island","Queens","Bronx","Brooklyn"]}},
... {
... "restaurant_id" : 1,
... "name":1,"borough":1,
... "cuisine" :1
... }
... )
{ "_id" : ObjectId("6205492c0aaafc1f9467100c"), "borough" : "Bronx", "cuisine" : "Bakery", "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100d"), "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100f"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "Riviera Caterer", "restaurant_id" : "40356018" }
{ "_id" : ObjectId("6205492c0aaafc1f94671010"), "borough" : "Queens", "cuisine" : "Jewish/Kosher", "name" : "Tov Kosher Kitchen", "restaurant_id" : "40356068" }
{ "_id" : ObjectId("6205492c0aaafc1f94671011"), "borough" : "Queens", "cuisine" : "American ", "name" : "Brunos On The Boulevard", "restaurant_id" : "40356151" }
```

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.



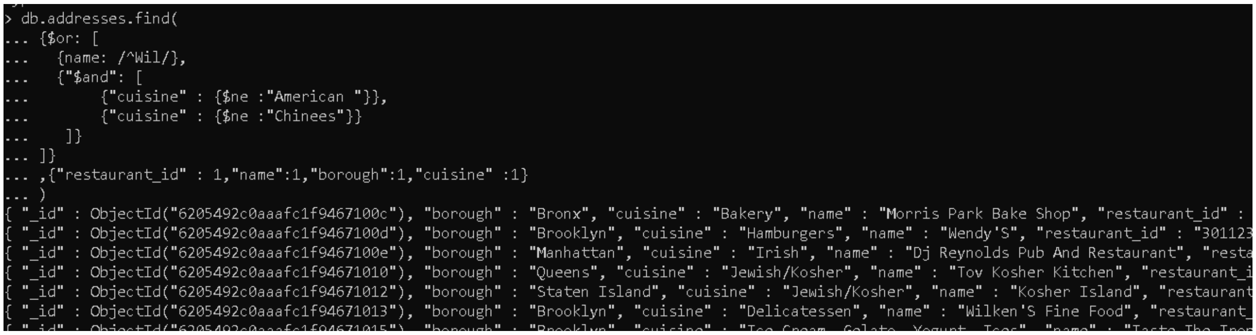
```
> db.addresses.find(
... {"borough" :{$nin :["Staten Island","Queens","Bronx","Brooklyn"]}},
... {
... "restaurant_id" : 1,
... "name":1,"borough":1,
... "cuisine" :1
... }
... )
{ "_id" : ObjectId("6205492c0aaafc1f9467100e"), "borough" : "Manhattan", "cuisine" : "Irish", "name" : "Dj Reynolds Pub And Restaurant", "
{ "_id" : ObjectId("6205492c0aaafc1f94671019"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "1 East 66Th Street Kitchen", "
{ "_id" : ObjectId("6205492c0aaafc1f9467101e"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "Glorious Food", "restaurant_id
```

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.



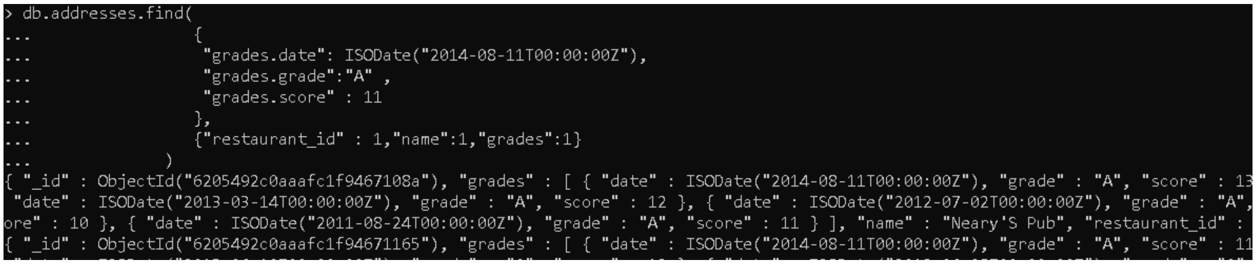
```
> db.addresses.find(
... {"grades.score" :
... { $not:
... {$gt : 10}
... }
... },
... {
... "restaurant_id" : 1,
... "name":1,"borough":1,
... "cuisine" :1
... }
... )
{ "_id" : ObjectId("6205492c0aaafc1f94671017"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "C & C Cateri
{ "_id" : ObjectId("6205492c0aaafc1f94671019"), "borough" : "Manhattan", "cuisine" : "American ", "name" : "1 East 66Th
{ "_id" : ObjectId("6205492c0aaafc1f9467101d"), "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Nordic De
{ "_id" : ObjectId("6205492c0aaafc1f94671026"), "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "White Castl
```

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'.



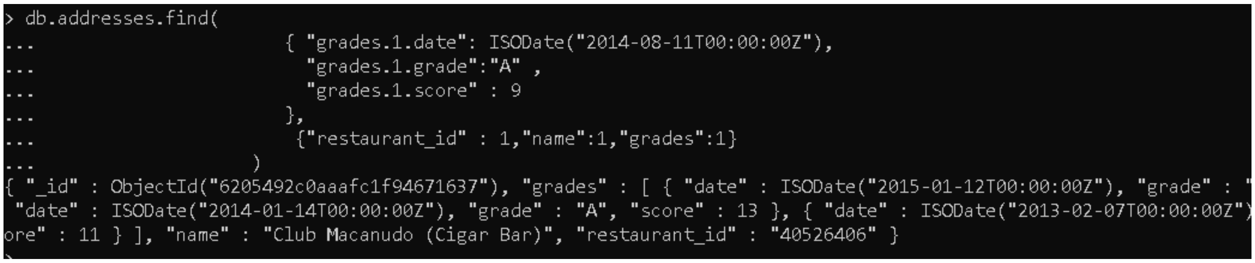
```
> db.addresses.find(
... {$or: [
...    {name: /^Wil/},
...    {"$and": [
...         {"cuisine" : {$ne :"American "}},
...         {"cuisine" : {$ne :"Chinees"}}
...     ]}
... ]}
... ,{"restaurant_id" : 1,"name":1,"borough":1,"cuisine" :1}
... )
{ "_id" : ObjectId("6205492c0aaafc1f9467100c"), "borough" : "Bronx", "cuisine" : "Bakery", "name" : "Morris Park Bake Shop", "restaurant_id" :
{ "_id" : ObjectId("6205492c0aaafc1f9467100d"), "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "Wendy'S", "restaurant_id" : "301123
{ "_id" : ObjectId("6205492c0aaafc1f9467100e"), "borough" : "Manhattan", "cuisine" : "Irish", "name" : "Dj Reynolds Pub And Restaurant", "resta
{ "_id" : ObjectId("6205492c0aaafc1f94671010"), "borough" : "Queens", "cuisine" : "Jewish/Kosher", "name" : "Tov Kosher Kitchen", "restaurant_i
{ "_id" : ObjectId("6205492c0aaafc1f94671012"), "borough" : "Staten Island", "cuisine" : "Jewish/Kosher", "name" : "Kosher Island", "restaurant
{ "_id" : ObjectId("6205492c0aaafc1f94671013"), "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Wilken'S Fine Food", "restaurant_
```

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..



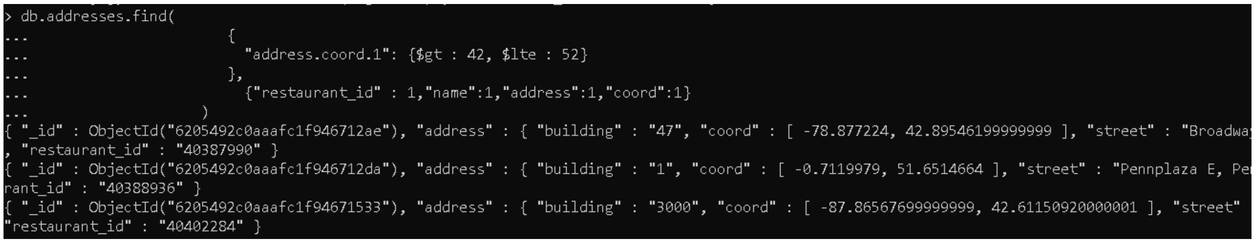
```
> db.addresses.find(
...                {
...                 "grades.date": ISODate("2014-08-11T00:00:00Z"),
...                 "grades.grade":"A" ,
...                 "grades.score" : 11
...                },
...                {"restaurant_id" : 1,"name":1,"grades":1}
...             )
{ "_id" : ObjectId("6205492c0aaafc1f9467108a"), "grades" : [ { "date" : ISODate("2014-08-11T00:00:00Z"), "grade" : "A", "score" : 13
 "date" : ISODate("2013-03-14T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-07-02T00:00:00Z"), "grade" : "A",
ore" : 10 }, { "date" : ISODate("2011-08-24T00:00:00Z"), "grade" : "A", "score" : 11 } ], "name" : "Neary'S Pub", "restaurant_id" :
{ "_id" : ObjectId("6205492c0aaafc1f94671165"), "grades" : [ { "date" : ISODate("2014-08-11T00:00:00Z"), "grade" : "A", "score" : 11
```

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"



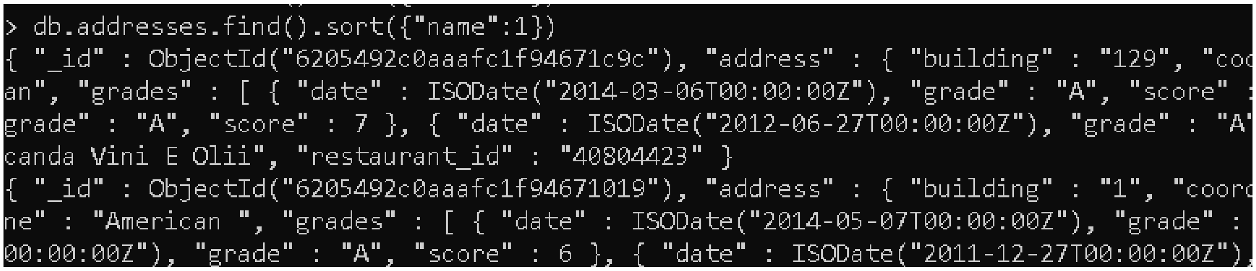
```
> db.addresses.find(
...                        { "grades.1.date": ISODate("2014-08-11T00:00:00Z"),
...                          "grades.1.grade":"A" ,
...                          "grades.1.score" : 9
...                        },
...                         {"restaurant_id" : 1,"name":1,"grades":1}
...                     )
{ "_id" : ObjectId("6205492c0aaafc1f94671637"), "grades" : [ { "date" : ISODate("2015-01-12T00:00:00Z"), "grade" : "
 "date" : ISODate("2014-01-14T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2013-02-07T00:00:00Z")
ore" : 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..



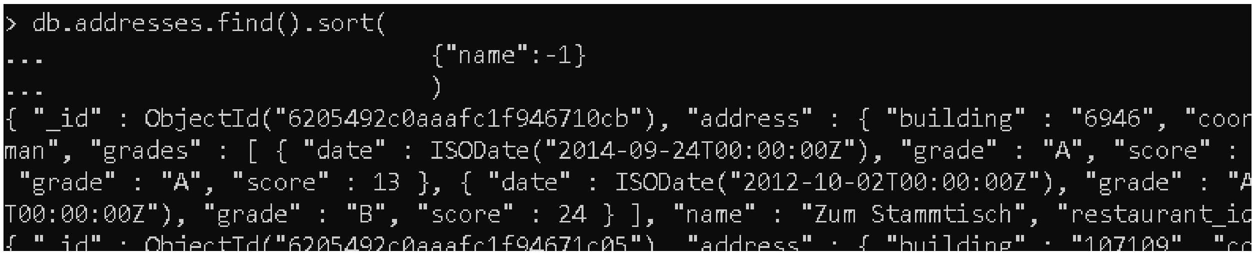
```
> db.addresses.find(
...                        {
...                          "address.coord.1": {$gt : 42, $lte : 52}
...                        },
...                          {"restaurant_id" : 1,"name":1,"address":1,"coord":1}
...                     )
{ "_id" : ObjectId("6205492c0aaafc1f946712ae"), "address" : { "building" : "47", "coord" : [ -78.877224, 42.89546199999999 ], "street" : "Broadwa
, "restaurant_id" : "40387990" }
{ "_id" : ObjectId("6205492c0aaafc1f946712da"), "address" : { "building" : "1", "coord" : [ -0.7119979, 51.6514664 ], "street" : "Pennplaza E, Pe
rant_id" : "40388936" }
{ "_id" : ObjectId("6205492c0aaafc1f94671533"), "address" : { "building" : "3000", "coord" : [ -87.86567699999999, 42.61150920000001 ], "street"
"restaurant_id" : "40402284" }
```

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



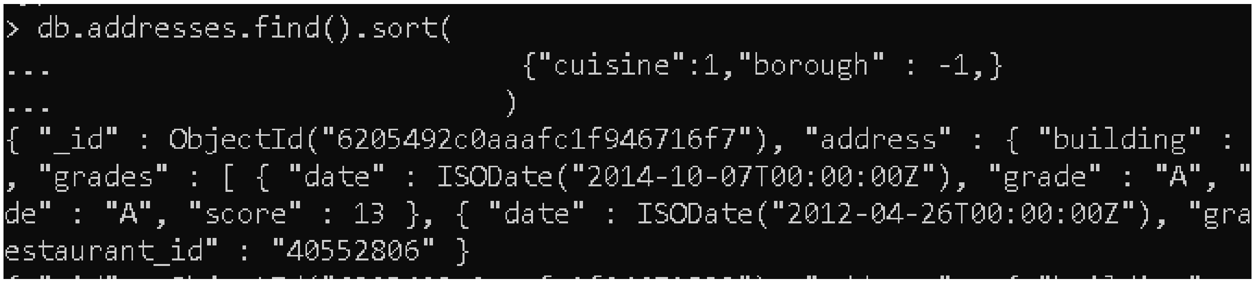
```
> db.addresses.find().sort({"name":1})
{ "_id" : ObjectId("6205492c0aaafc1f94671c9c"), "address" : { "building" : "129", "coo
an", "grades" : [ { "date" : ISODate("2014-03-06T00:00:00Z"), "grade" : "A", "score" :
grade" : "A", "score" : 7 }, { "date" : ISODate("2012-06-27T00:00:00Z"), "grade" : "A"
canda Vini E Olii", "restaurant_id" : "40804423" }
{ "_id" : ObjectId("6205492c0aaafc1f94671019"), "address" : { "building" : "1", "coord
ne" : "American ", "grades" : [ { "date" : ISODate("2014-05-07T00:00:00Z"), "grade" :
00:00:00Z"), "grade" : "A", "score" : 6 }, { "date" : ISODate("2011-12-27T00:00:00Z"),
```

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



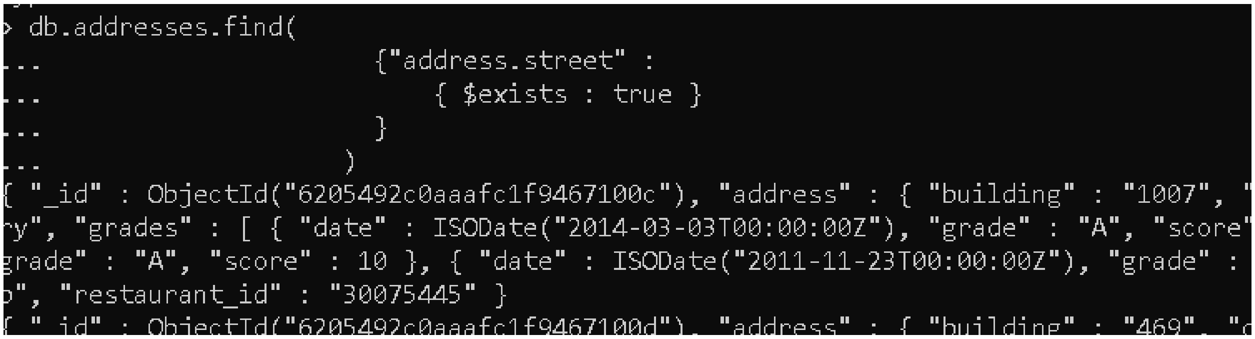
```
> db.addresses.find().sort(
...                             {"name":-1}
...                             )
{ "_id" : ObjectId("6205492c0aaafc1f946710cb"), "address" : { "building" : "6946", "coor
man", "grades" : [ { "date" : ISODate("2014-09-24T00:00:00Z"), "grade" : "A", "score" :
 "grade" : "A", "score" : 13 }, { "date" : ISODate("2012-10-02T00:00:00Z"), "grade" : "A
T00:00:00Z"), "grade" : "B", "score" : 24 } ], "name" : "Zum Stammtisch", "restaurant_id
{ "_id" : ObjectId("6205492c0aaafc1f94671c05"), "address" : { "building" : "107109", "co
```

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.



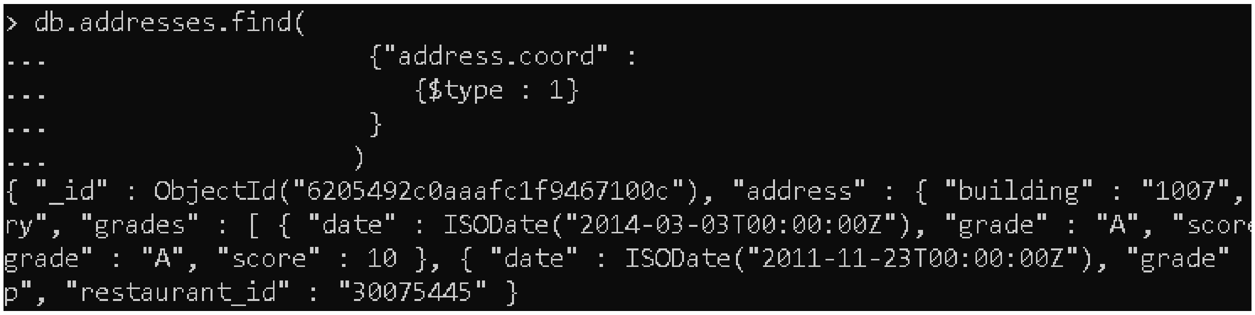
```
> db.addresses.find().sort(
...                             {"cuisine":1,"borough" : -1,}
...                           )
{ "_id" : ObjectId("6205492c0aaafc1f946716f7"), "address" : { "building" :
, "grades" : [ { "date" : ISODate("2014-10-07T00:00:00Z"), "grade" : "A", "
de" : "A", "score" : 13 }, { "date" : ISODate("2012-04-26T00:00:00Z"), "gra
estaurant_id" : "40552806" }
```

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



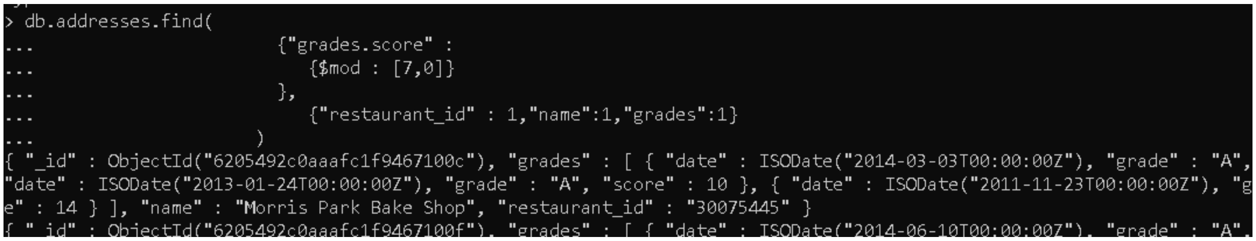
```
> db.addresses.find(
...                         {"address.street" :
...                             { $exists : true }
...                         }
...                       )
{ "_id" : ObjectId("6205492c0aaafc1f9467100c"), "address" : { "building" : "1007", "
ry", "grades" : [ { "date" : ISODate("2014-03-03T00:00:00Z"), "grade" : "A", "score"
grade" : "A", "score" : 10 }, { "date" : ISODate("2011-11-23T00:00:00Z"), "grade" :
p", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100d"), "address" : { "building" : "469", "c
```

29. Write a MongoDB query which will select all documents in the restaurants collection where the coord field value is Double.



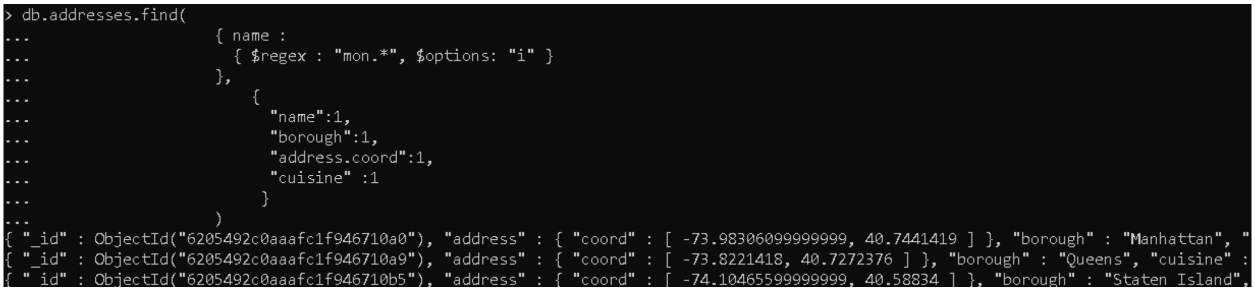
```
> db.addresses.find(
...                     {"address.coord" :
...                        {$type : 1}
...                     }
...                   )
{ "_id" : ObjectId("6205492c0aaafc1f9467100c"), "address" : { "building" : "1007",
ry", "grades" : [ { "date" : ISODate("2014-03-03T00:00:00Z"), "grade" : "A", "scor
grade" : "A", "score" : 10 }, { "date" : ISODate("2011-11-23T00:00:00Z"), "grade"
p", "restaurant_id" : "30075445" }
```

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.



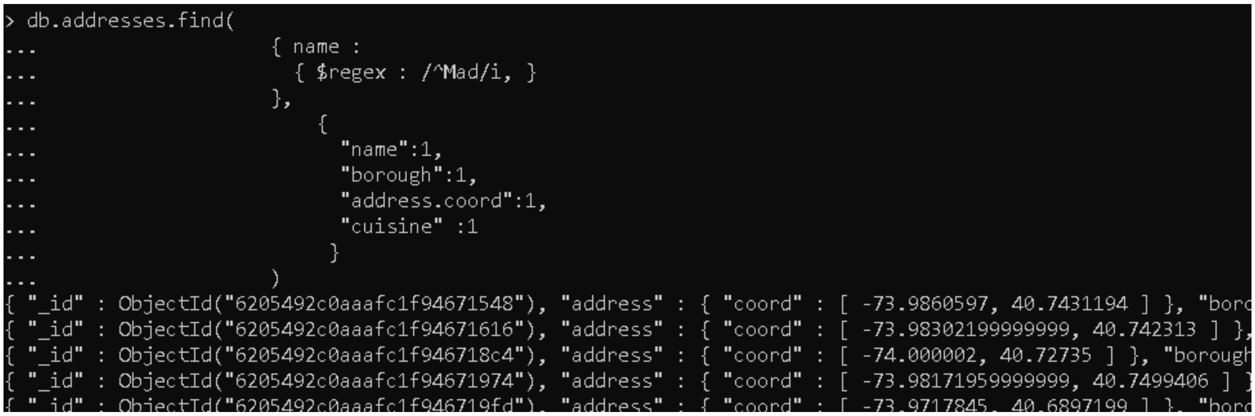
```
> db.addresses.find(
...                     {"grades.score" :
...                        {$mod : [7,0]}
...                     },
...                        {"restaurant_id" : 1,"name":1,"grades":1}
...                   )
{ "_id" : ObjectId("6205492c0aaafc1f9467100c"), "grades" : [ { "date" : ISODate("2014-03-03T00:00:00Z"), "grade" : "A",
"date" : ISODate("2013-01-24T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2011-11-23T00:00:00Z"), "g
e" : 14 } ], "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6205492c0aaafc1f9467100f"), "grades" : [ { "date" : ISODate("2014-06-10T00:00:00Z"), "grade" : "A",
```

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
...                        }
...                   )
{ "_id" : ObjectId("6205492c0aaafc1f946710a0"), "address" : { "coord" : [ -73.98306099999999, 40.7441419 ] }, "borough" : "Manhattan", "
{ "_id" : ObjectId("6205492c0aaafc1f946710a9"), "address" : { "coord" : [ -73.8221418, 40.7272376 ] }, "borough" : "Queens", "cuisine" :
{ "_id" : ObjectId("6205492c0aaafc1f946710b5"), "address" : { "coord" : [ -74.10465599999999, 40.58834 ] }, "borough" : "Staten Island",
```

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
...                         }
...                    )
{ "_id" : ObjectId("6205492c0aaafc1f94671548"), "address" : { "coord" : [ -73.9860597, 40.7431194 ] }, "boro
{ "_id" : ObjectId("6205492c0aaafc1f94671616"), "address" : { "coord" : [ -73.98302199999999, 40.742313 ] },
{ "_id" : ObjectId("6205492c0aaafc1f946718c4"), "address" : { "coord" : [ -74.000002, 40.72735 ] }, "borough
{ "_id" : ObjectId("6205492c0aaafc1f94671974"), "address" : { "coord" : [ -73.98171959999999, 40.7499406 ] }
{ "_id" : ObjectId("6205492c0aaafc1f946719fd"), "address" : { "coord" : [ -73.9717845, 40.6897199 ] }, "boro
```