Complex Assignment

Exercise Questions:

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

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> show dbs
mongo_practice   217 kB
population       1.68 MB
restaurants       655 kB
admin             336 kB
local            4.25 GB
Atlas atlas-maqh63-shard-0 [primary] restaurants> show collections
addresses
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find()
[
  {
    _id: ObjectId("6205f959ff06e556459af91c"),
    address: {
      building: '1007',
      coord: [ -73.856077, 40.848447 ],
      street: 'Morris Park Ave',
      zipcode: '10462'
    },
    borough: 'Bronx',
    cuisine: 'Bakery',
    grades: [
      {
        date: ISODate("2014-03-03T00:00:00.000Z"),
        grade: 'A',
        score: 2
      },
      {
        date: ISODate("2013-09-11T00:00:00.000Z"),
        grade: 'A',
        score: 6
      },
      {
        date: ISODate("2013-01-24T00:00:00.000Z"),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate("2011-11-23T00:00:00.000Z"),
        grade: 'A',
        score: 9
      },
      {
        date: ISODate("2011-03-10T00:00:00.000Z"),
        grade: 'B',
        score: 14
      }
    ],
```

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

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({},{restaurant_id:1,name:1,cuisine: 1,borough:1})
[
  {
    _id: ObjectId("6205f959ff06e556459af91c"),
    borough: 'Bronx',
    cuisine: 'Bakery',
    name: 'Morris Park Bake Shop',
    restaurant_id: '30075445'
  },
  {
    _id: ObjectId("6205f959ff06e556459af91d"),
    borough: 'Brooklyn',
    cuisine: 'Hamburgers',
    name: "Wendy'S",
    restaurant_id: '30112340'
  },
  {
    _id: ObjectId("6205f959ff06e556459af91e"),
    borough: 'Manhattan',
    cuisine: 'Irish',
    name: 'Dj Reynolds Pub And Restaurant',
    restaurant_id: '30191841'
  },
  {
    _id: ObjectId("6205f959ff06e556459af91f"),
    borough: 'Brooklyn',
    cuisine: 'American ',
    name: 'Riviera Caterer',
    restaurant_id: '40356018'
  },
  {
    _id: ObjectId("6205f959ff06e556459af920"),
    borough: 'Queens',
    cuisine: 'Jewish/Kosher',
    name: 'Tov Kosher Kitchen',
    restaurant_id: '40356068'
  },
  {
    _id: ObjectId("6205f959ff06e556459af921"),
    borough: 'Queens',
    cuisine: 'American ',
    name: 'Brunos On The Boulevard',
    restaurant_id: '40356151'
  },
```

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.

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({},{restaurant_id:1,name:1,cuisine: 1,borough: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'
  },
```

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.

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({},{restaurant_id:1,name:1,'address.zipcode' : 1,borough: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'
  },
```

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

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({borough:'Bronx'}).limit(5)
[
  {
    _id: ObjectId("6205f959ff06e556459af91c"),
    address: {
      building: '1007',
      coord: [ -73.856077, 40.848447 ],
      street: 'Morris Park Ave',
      zipcode: '10462'
    },
    borough: 'Bronx',
    cuisine: 'Bakery',
    grades: [
      {
        date: ISODate("2014-03-03T00:00:00.000Z"),
        grade: 'A',
        score: 2
      },
      {
        date: ISODate("2013-09-11T00:00:00.000Z"),
        grade: 'A',
        score: 6
      },
      {
        date: ISODate("2013-01-24T00:00:00.000Z"),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate("2011-11-23T00:00:00.000Z"),
        grade: 'A',
        score: 9
      },
      {
        date: ISODate("2011-03-10T00:00:00.000Z"),
        grade: 'B',
        score: 14
      }
    ],
    name: 'Morris Park Bake Shop',
    restaurant_id: '30075445'
  },
```

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

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({borough:'Bronx'}).pretty()
[
  {
    _id: ObjectId("6205f959ff06e556459af91c"),
    address: {
      building: '1007',
      coord: [ -73.856077, 40.848447 ],
      street: 'Morris Park Ave',
      zipcode: '10462'
    },
    borough: 'Bronx',
    cuisine: 'Bakery',
    grades: [
      {
        date: ISODate("2014-03-03T00:00:00.000Z"),
        grade: 'A',
        score: 2
      },
      {
        date: ISODate("2013-09-11T00:00:00.000Z"),
        grade: 'A',
        score: 6
      },
      {
        date: ISODate("2013-01-24T00:00:00.000Z"),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate("2011-11-23T00:00:00.000Z"),
        grade: 'A',
        score: 9
      },
      {
        date: ISODate("2011-03-10T00:00:00.000Z"),
        grade: 'B',
        score: 14
      }
    ],
    name: 'Morris Park Bake Shop',
    restaurant_id: '30075445'
  },
```

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

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({borough:'Bronx'}).skip(5).limit(5)
[
  {
    _id: ObjectId("6205f959ff06e556459af959"),
    address: {
      building: '658',
      coord: [ -73.81363999999999, 40.82941100000001 ],
      street: 'Clarence Ave',
      zipcode: '10465'
    },
    borough: 'Bronx',
    cuisine: 'American ',
    grades: [
      {
        date: ISODate("2014-06-21T00:00:00.000Z"),
        grade: 'A',
        score: 5
      },
      {
        date: ISODate("2012-07-11T00:00:00.000Z"),
        grade: 'A',
        score: 10
      }
    ],
    name: 'Manhem Club',
    restaurant_id: '40364363'
  },
  {
    _id: ObjectId("6205f959ff06e556459af971"),
    address: {
      building: '2222',
      coord: [ -73.84971759999999, 40.8304811 ],
      street: 'Haviland Avenue',
      zipcode: '10462'
    },
    borough: 'Bronx',
    cuisine: 'American ',
    grades: [
      {
        date: ISODate("2014-12-18T00:00:00.000Z"),
        grade: 'A',
        score: 7
      },
      {
        date: ISODate("2014-05-01T00:00:00.000Z"),
        grade: 'B',
        score: 17
      },
```

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

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({'grades.score':{$gt:90}})
[
  {
    _id: ObjectId("6205f959ff06e556459afa7a"),
    address: {
      building: '65',
      coord: [ -73.9782725, 40.7624022 ],
      street: 'West   54 Street',
      zipcode: '10019'
    },
    borough: 'Manhattan',
    cuisine: 'American ',
    grades: [
      {
        date: ISODate("2014-08-22T00:00:00.000Z"),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate("2014-03-28T00:00:00.000Z"),
        grade: 'C',
        score: 131
      },
      {
        date: ISODate("2013-09-25T00:00:00.000Z"),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate("2013-04-08T00:00:00.000Z"),
        grade: 'B',
        score: 25
      },
      {
        date: ISODate("2012-10-15T00:00:00.000Z"),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate("2011-10-19T00:00:00.000Z"),
        grade: 'A',
        score: 13
      }
    ],
    name: "Murals On 54/Randolphs'S",
    restaurant_id: '40372466'
  },
```

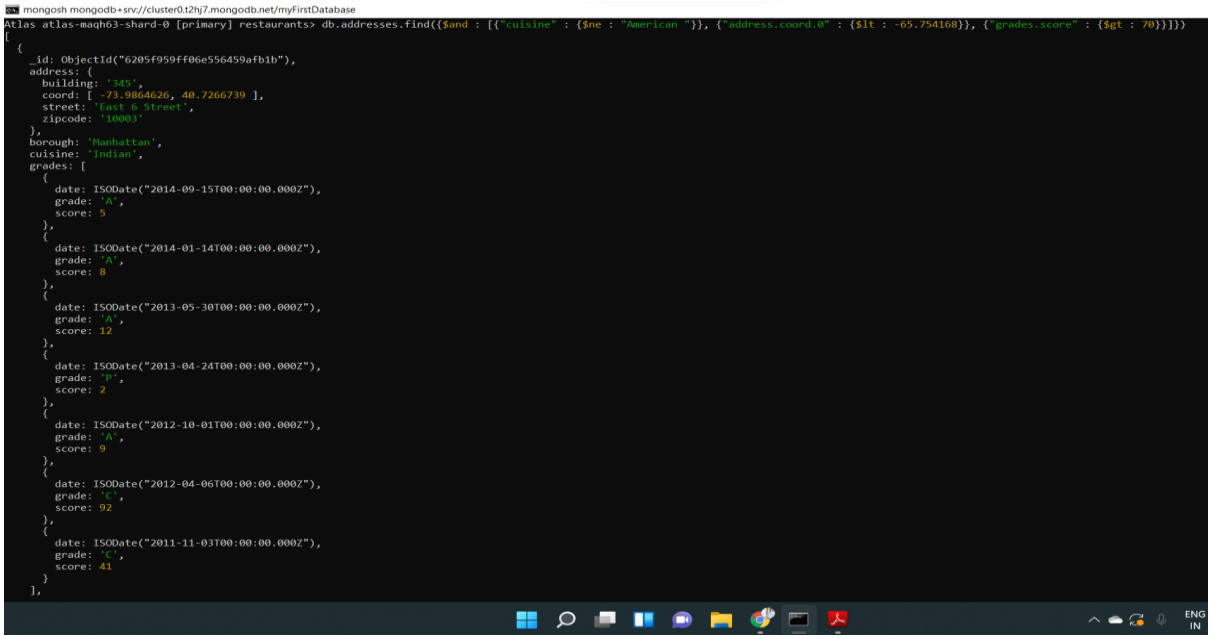
9. Write a MongoDB query to find the restaurants that achieved a score, more than 80 but less than 100.

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({$and:[{'grades.score':{$gt:80}},{'grades.score':{$lt:100}}]})
[     zipcode: '10016'
  { },
    _id: ObjectId("6205f959ff06e556459afa7a"),
    address: {Pizza/Italian',
      building: '65',
      coord: [ -73.9782725, 40.7624022 ],
      street: 'West   54 Street',T00:00:00.000Z"),
      zipcode: '10019'
    },  score: 31
    borough: 'Manhattan',
    cuisine: 'American ',
    grades: [ ISODate("2014-06-17T00:00:00.000Z"),
      { grade: 'C',
        date: ISODate("2014-08-22T00:00:00.000Z"),
        grade: 'A',
        score: 11
      },date: ISODate("2013-12-12T00:00:00.000Z"),
      { grade: 'C',
        date: ISODate("2014-03-28T00:00:00.000Z"),
        grade: 'C',
        score: 131
      },date: ISODate("2013-05-22T00:00:00.000Z"),
      { grade: 'B',
        date: ISODate("2013-09-25T00:00:00.000Z"),
        grade: 'A',
        score: 11
      },date: ISODate("2012-05-02T00:00:00.000Z"),
      { grade: 'A',
        date: ISODate("2013-04-08T00:00:00.000Z"),
        grade: 'B',
        score: 25
      },: 'Bella Napoli',
      {taurant_id: '40393488'
        date: ISODate("2012-10-15T00:00:00.000Z"),
        grade: 'A',
        score: 11("6205f95aff06e556459b04ea"),
      },ess: {
      {uilding: '',
        date: ISODate("2011-10-19T00:00:00.000Z"),
        grade: 'A',on River',
        score: 130282'
      }
    ],rough: 'Manhattan',
    name: "Murals On 54/Randolphs'S",
    restaurant_id: '40372466'
  },  {
```

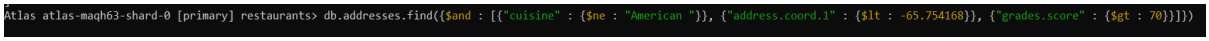
10. Write a MongoDB query to find the restaurants which locate in latitude value less than -95.754168

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({'address.coord':{$lt : -95.754168}})
[
  {
    _id: ObjectId("6205f95aff06e556459aff64"),
    address: {
      building: '3707',
      coord: [ -101.8945214, 33.5197474 ],
      street: '82 Street',
      zipcode: '11372'
    },
    borough: 'Queens',
    cuisine: 'American ',
    grades: [
      {
        date: ISODate("2014-06-04T00:00:00.000Z"),
        grade: 'A',
        score: 12
      },
      {
        date: ISODate("2013-11-07T00:00:00.000Z"),
        grade: 'B',
        score: 19
      },
      {
        date: ISODate("2013-05-17T00:00:00.000Z"),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate("2012-08-29T00:00:00.000Z"),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate("2012-04-03T00:00:00.000Z"),
        grade: 'A',
        score: 12
      },
      {
        date: ISODate("2011-11-16T00:00:00.000Z"),
        grade: 'A',
        score: 7
      }
    ],
    name: 'Burger King',
    restaurant_id: '40534067'
  },
```

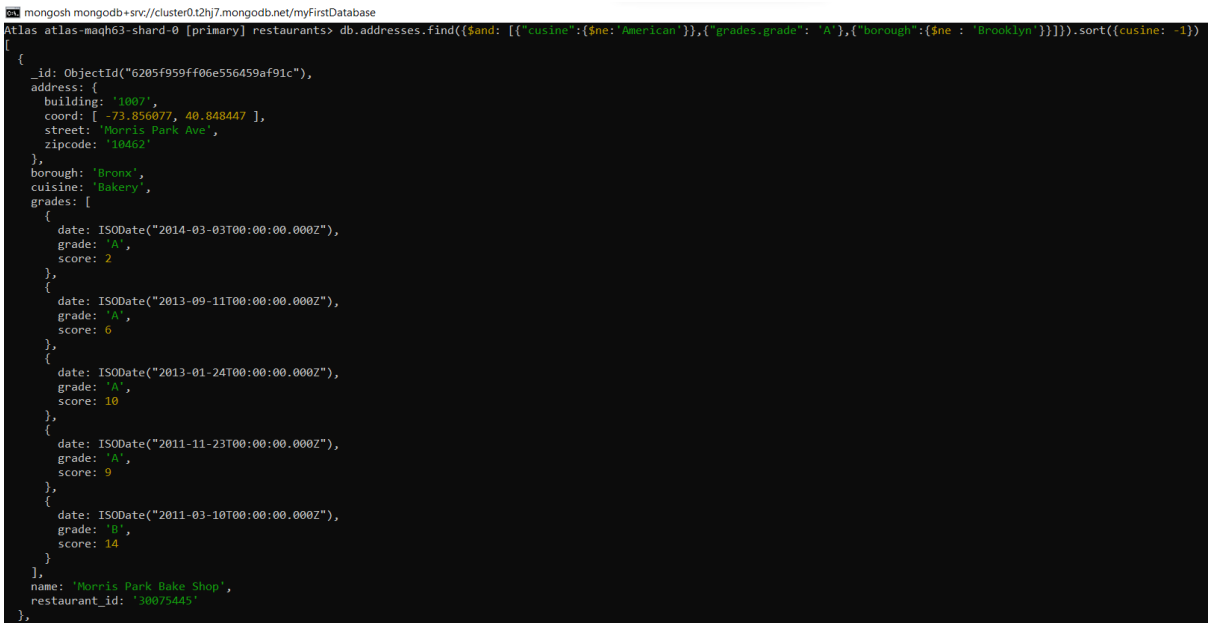
11. Write a MongoDB query to find the restaurants that do not prepare any cuisine of 'American' and their grade score more than 70 and latitude less than -65.754168.



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.

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase

Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({name:/^Wil/},{restaurant_id:1,name:1,borough:1,cuisine:1,_id:0})
[
  {
    borough: 'Brooklyn',
    cuisine: 'Delicatessen',
    name: "Wilken'S Fine Food",
    restaurant_id: '40356483'
  },
  {
    borough: 'Bronx',
    cuisine: 'American ',
    name: 'Wild Asia',
    restaurant_id: '40357217'
  },
  {
    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.

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({name:/ces$/},{restaurant_id:1,name:1,borough:1,cuisine:1,_id:0})
[
  {
    borough: 'Manhattan',
    cuisine: 'American ',
    name: 'Pieces',
    restaurant_id: '40399910'
  },
  {
    borough: 'Queens',
    cuisine: 'American ',
    name: 'S.M.R Restaurant Services',
    restaurant_id: '40403857'
  },
  {
    borough: 'Manhattan',
    cuisine: 'American ',
    name: 'Good Shepherd Services',
    restaurant_id: '40403989'
  },
  {
    borough: 'Queens',
    cuisine: 'Ice Cream, Gelato, Yogurt, Ices',
    name: "The Ice Box-Ralph'S Famous Italian Ices",
    restaurant_id: '40690899'
  },
  {
    borough: 'Brooklyn',
    cuisine: 'Jewish/Kosher',
    name: 'Alices',
    restaurant_id: '40782042'
  },
  {
    borough: 'Manhattan',
    cuisine: 'American ',
    name: 'Re: Sources',
    restaurant_id: '40876068'
  }
]
```

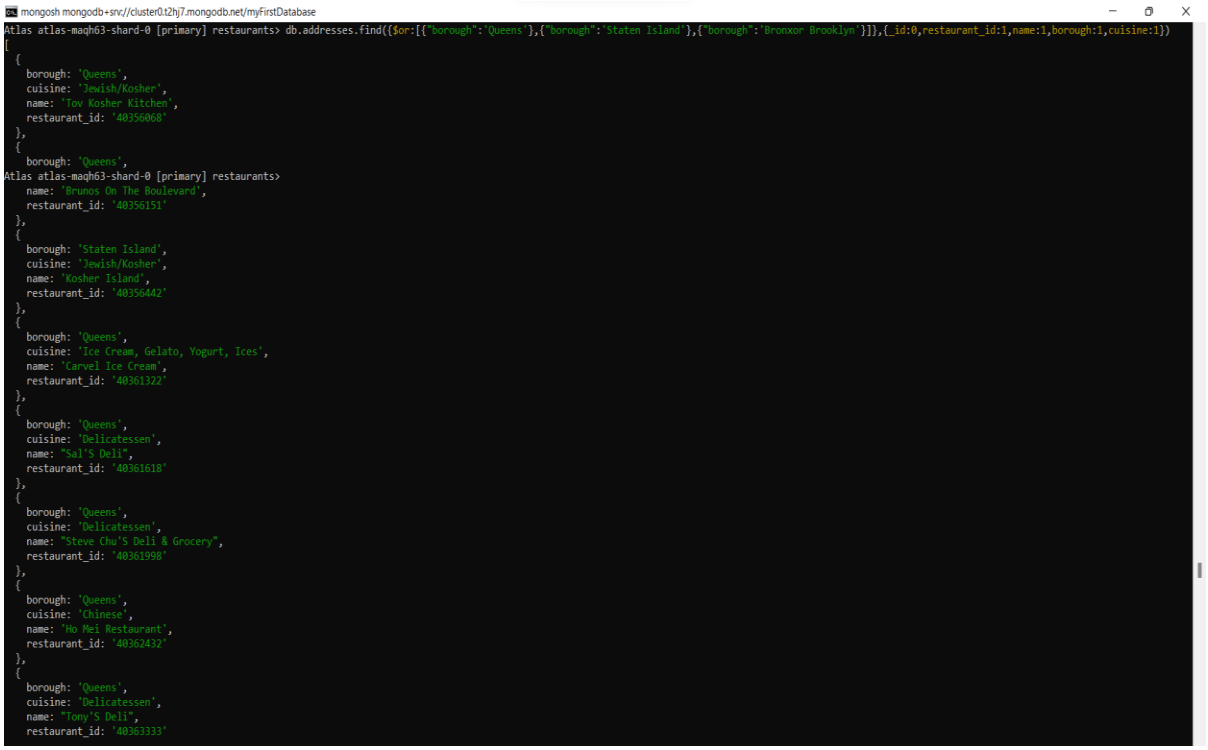
16. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which contain 'Reg' as three letters somewhere in its name.

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({name:/.*Reg.*/},{restaurant_id:1,name:1,borough:1,cuisine:1,_id:0})
[
  {
    borough: 'Brooklyn',
    cuisine: 'American ',
    name: 'Regina Caterers',
    restaurant_id: '40356649'
  },
  {
    borough: 'Manhattan',
    cuisine: 'Café/Coffee/Tea',
    name: 'Caffe Reggio',
    restaurant_id: '40369418'
  },
  {
    borough: 'Manhattan',
    cuisine: 'American ',
    name: 'Regency Hotel',
    restaurant_id: '40382679'
  },
  {
    borough: 'Manhattan',
    cuisine: 'American ',
    name: 'Regency Whist Club',
    restaurant_id: '40402377'
  },
  {
    borough: 'Queens',
    cuisine: 'American ',
    name: 'Rego Park Cafe',
    restaurant_id: '40523342'
  },
  {
    borough: 'Queens',
    cuisine: 'Pizza',
    name: 'Regina Pizza',
    restaurant_id: '40801325'
  },
  {
    borough: 'Manhattan',
    cuisine: 'American ',
    name: 'Regal Entertainment Group',
    restaurant_id: '40891782'
  }
]
```

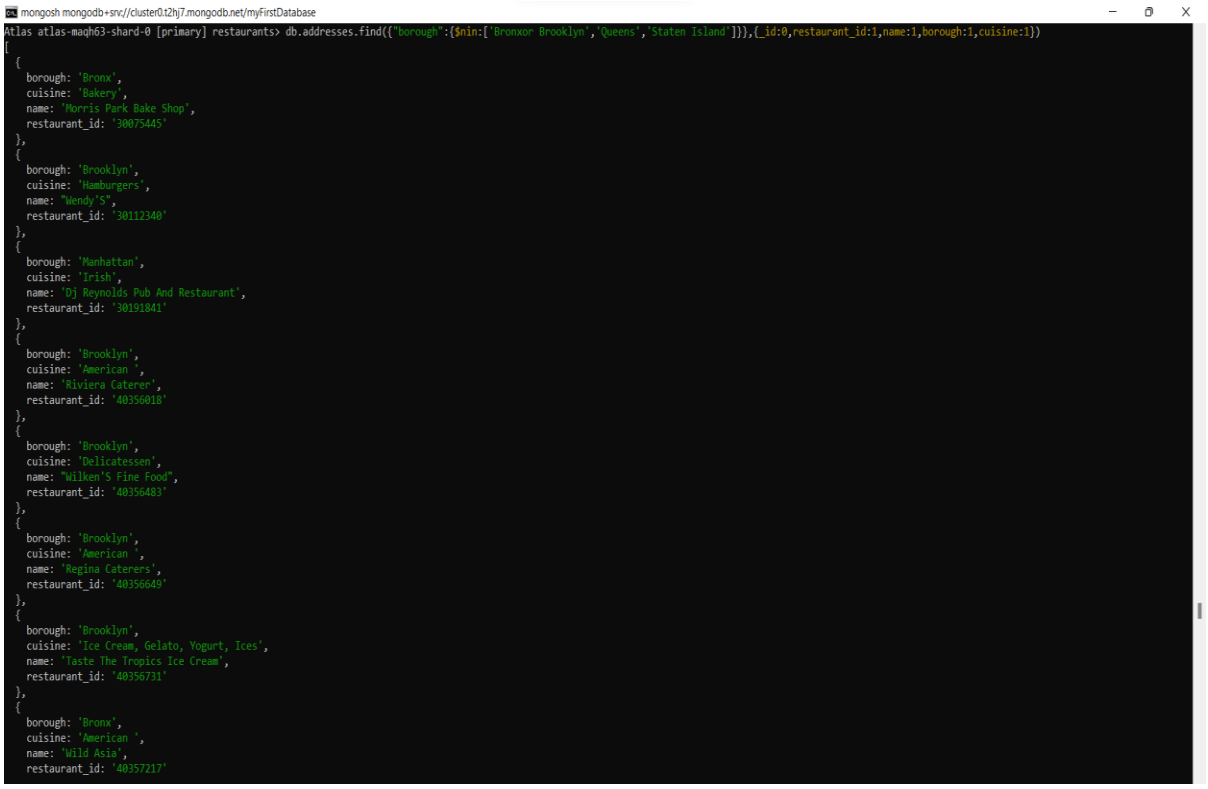
17. Write a MongoDB query to find the restaurants which belong to the borough Bronx and prepared either American or Chinese dish.

```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find({"borough":'Bronx',$or:[{"cuisine":'American'},{"cuisine":'Chinese'}]})
[
  {
    _id: ObjectId("6205f959ff06e556459af93f"),
    address: {
      building: '1236',
      coord: [ -73.8893654, 40.81376179999999 ],
      street: '238 Spofford Ave',
      zipcode: '10474'
    },
    borough: 'Bronx',
    cuisine: 'Chinese',
    grades: [
      {
        date: ISODate("2013-12-30T00:00:00.000Z"),
        grade: 'A',
        score: 8
      },
      {
        date: ISODate("2013-01-08T00:00:00.000Z"),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate("2012-06-12T00:00:00.000Z"),
        grade: 'B',
        score: 15
      }
    ],
    name: 'Happy Garden',
    restaurant_id: '40363289'
  },
```

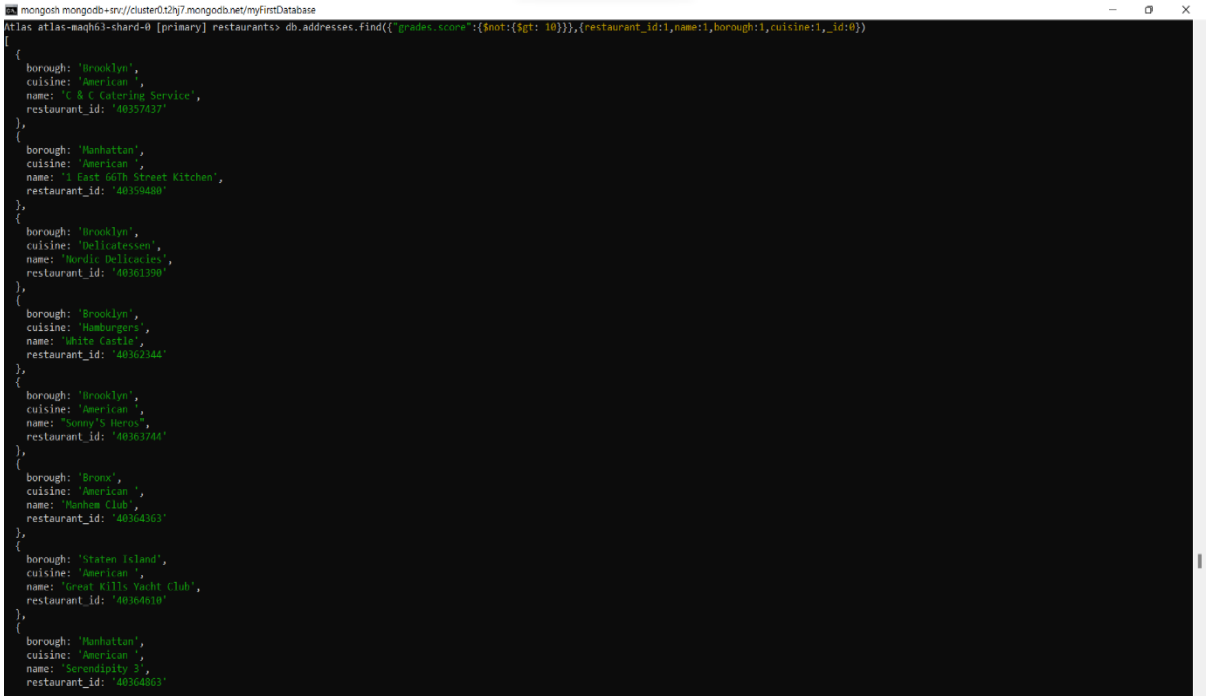
18. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which belong to the borough Staten Island or Queens or Bronxor Brooklyn.



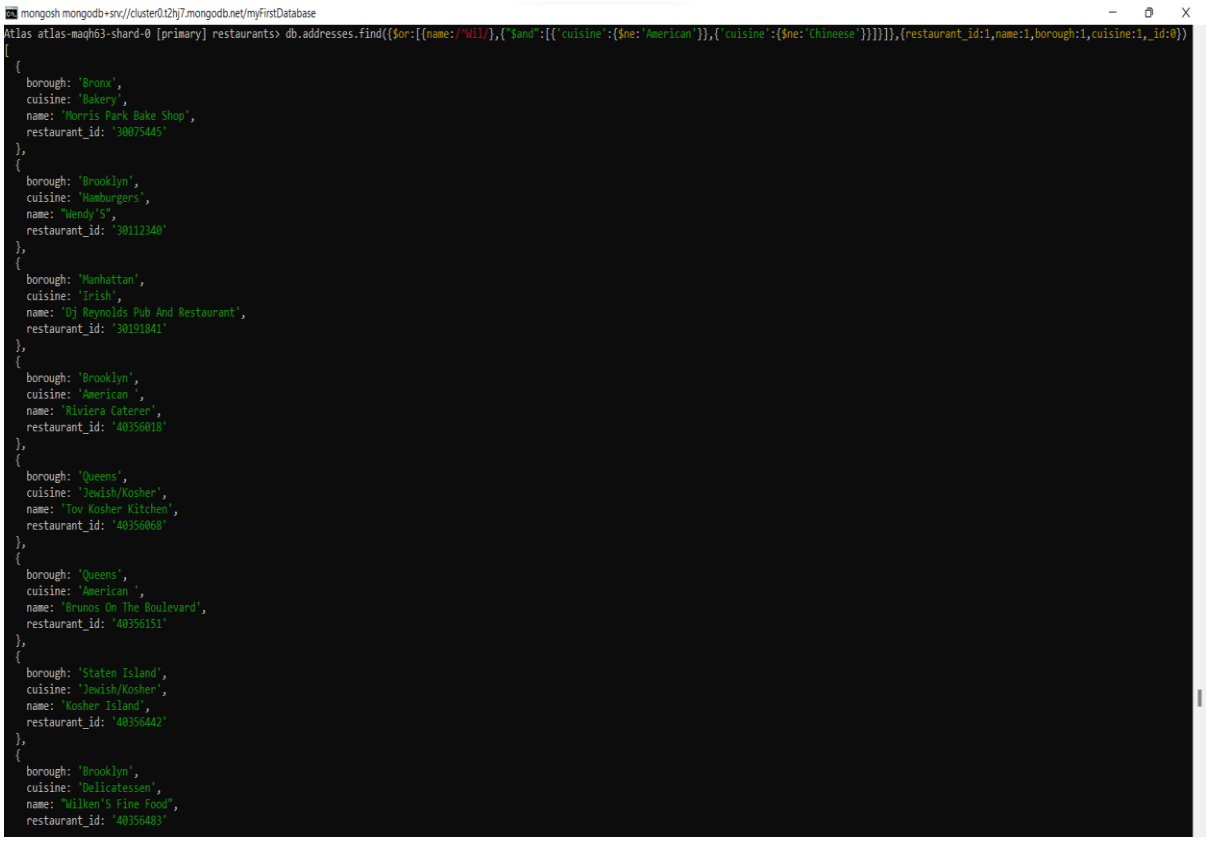
19. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which are not belonging to the borough Staten Island or Queens or Bronxor Brooklyn.



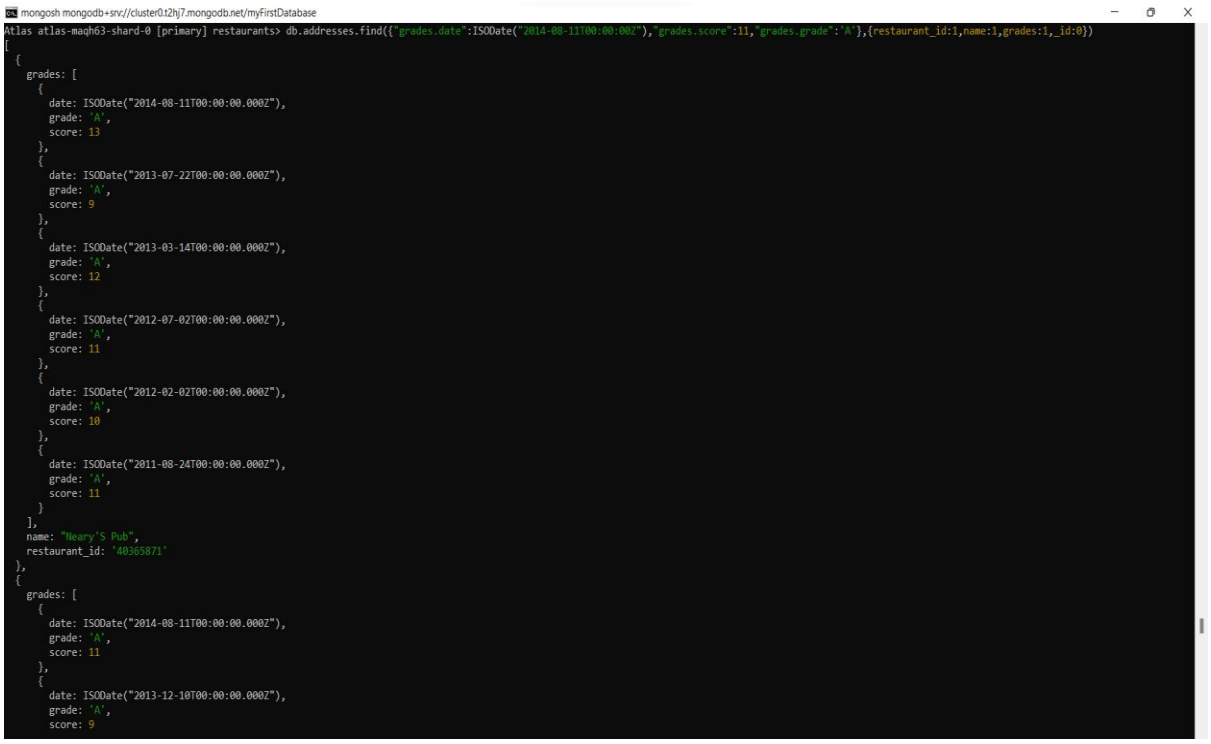
20. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which achieved a score which is not more than 10.



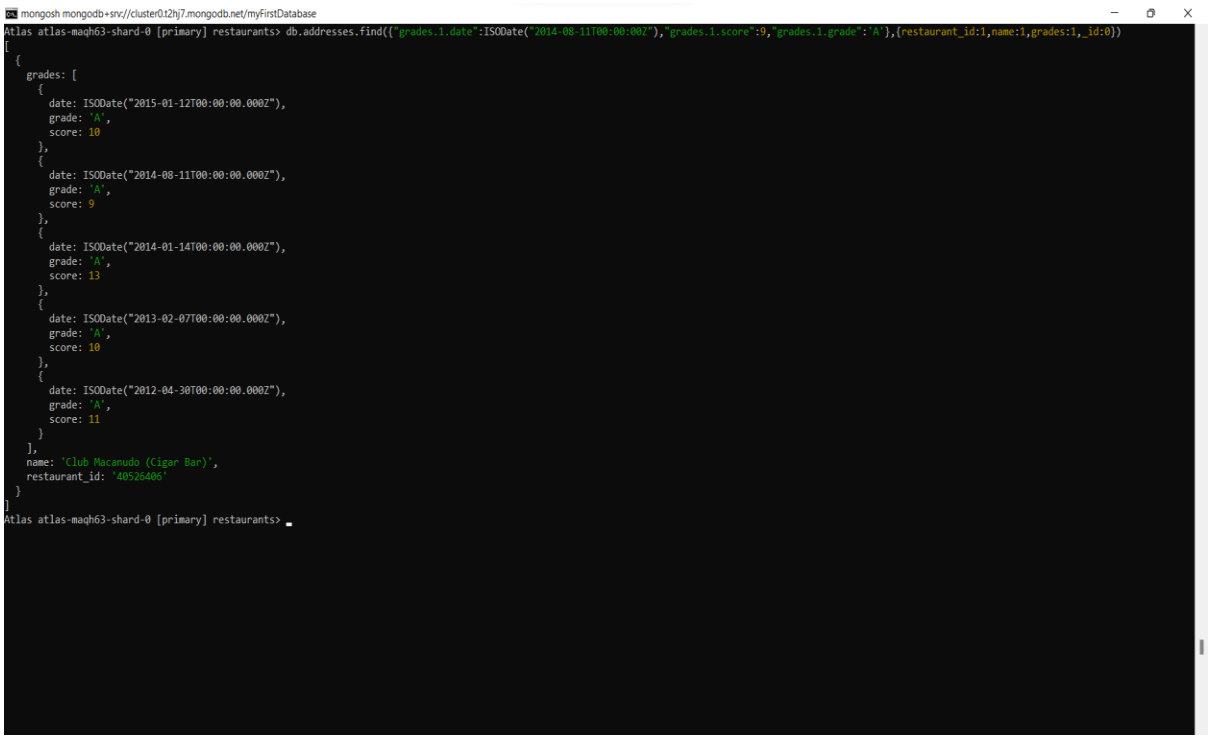
21. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which prepared dish except 'American' and 'Chinees' or restaurant's name begins with letter 'Wil'.



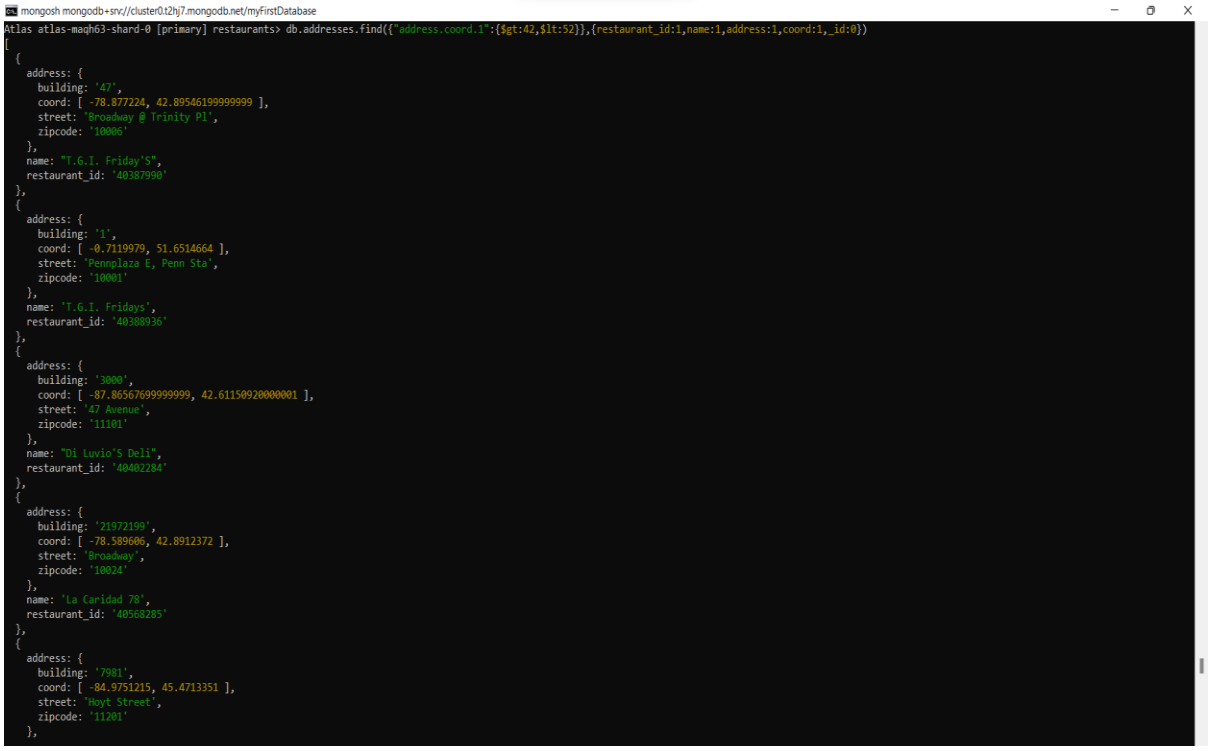
22. Write a MongoDB query to find the restaurant Id, name, and grades for those restaurants which achieved a grade of "A" and scored 11 on an ISODate "2014-08-11T00:00:00Z" among many of survey dates



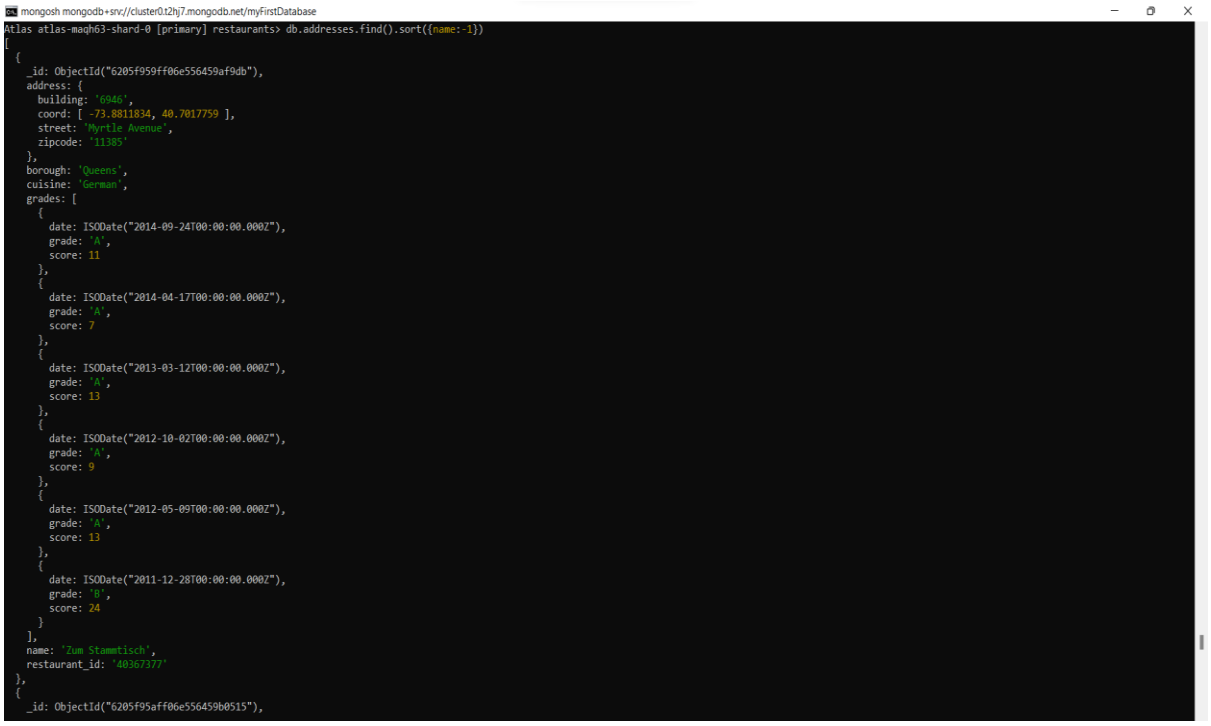
23. Write a MongoDB query to find the restaurant Id, name and grades for those restaurants where the 2nd element of grades array contains a grade of "A" and score 9 on an ISODate "2014-08-11T00:00:00Z"



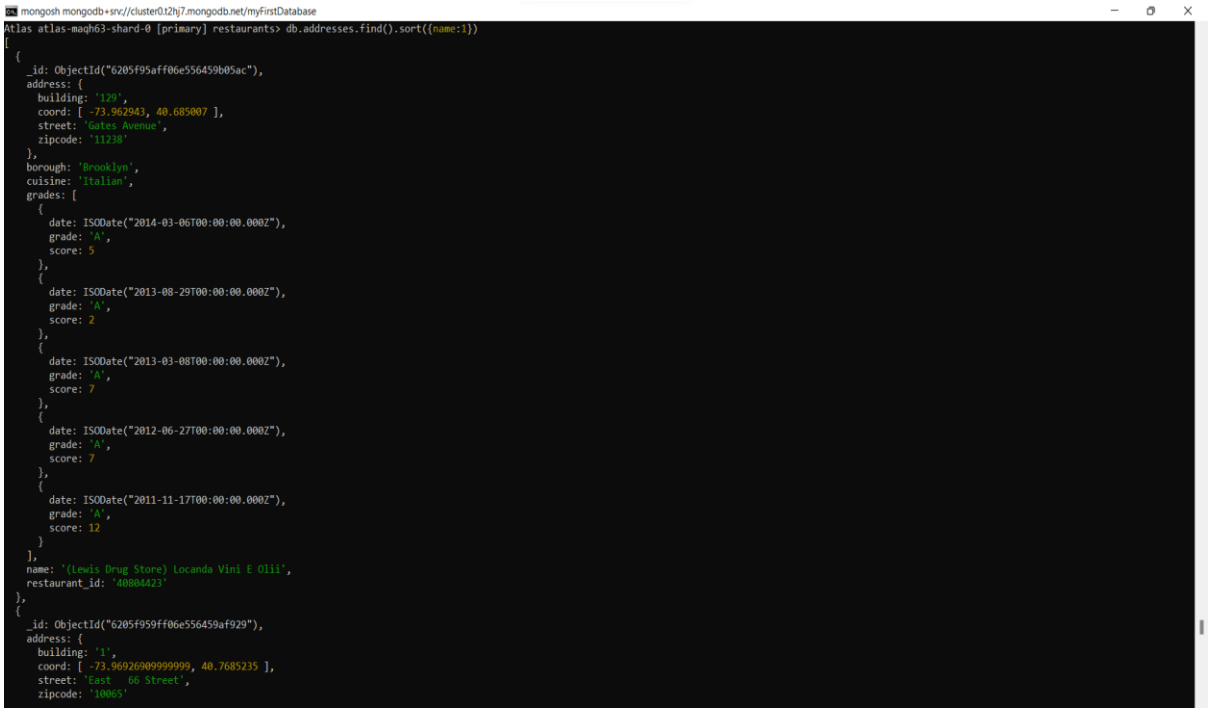
24. Write a MongoDB query to find the restaurant Id, name, address and geographical location for those restaurants where 2nd element of coord array contains a value which is more than 42 and upto 52.



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

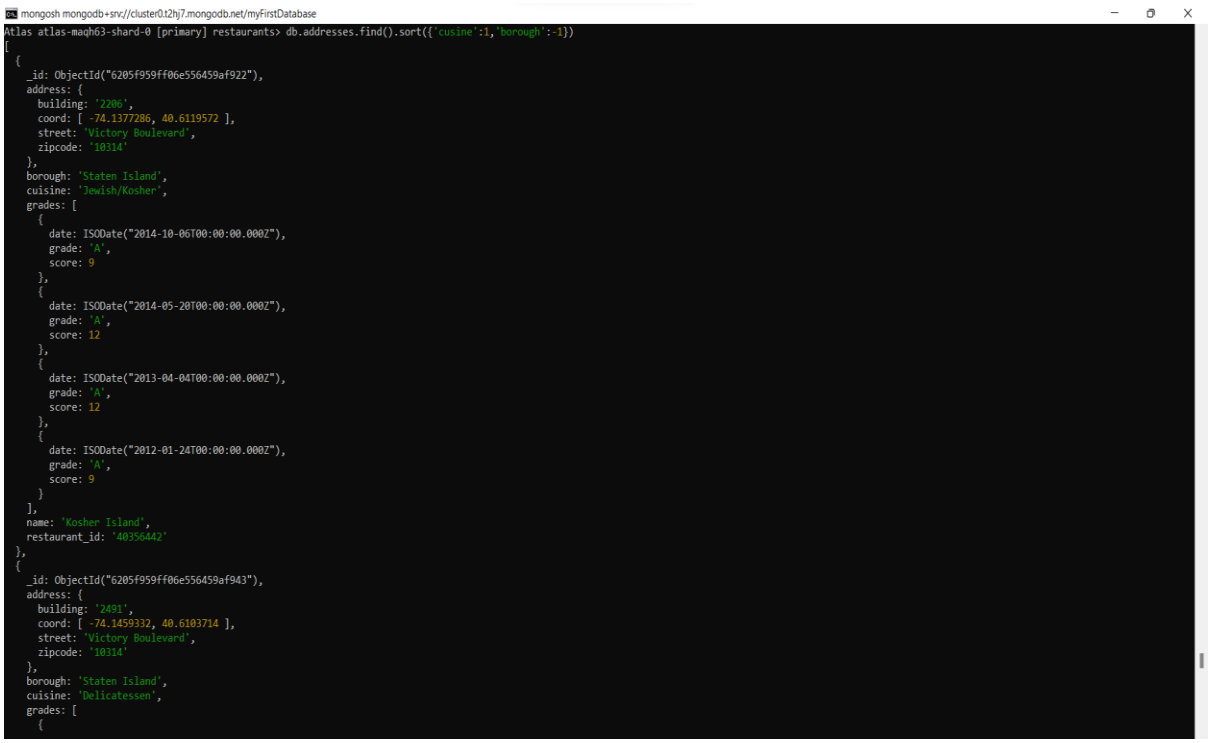


26. Write a MongoDB query to arrange the name of the restaurants in descending along with all the columns.



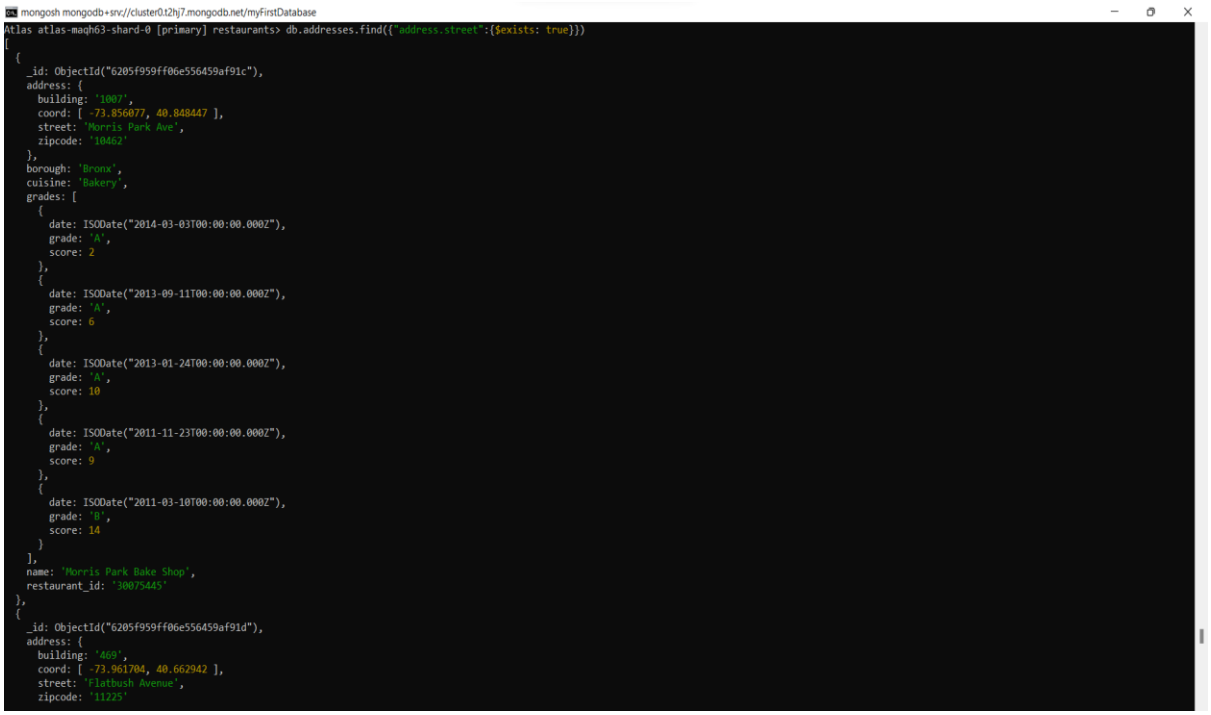
```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find().sort({name:1})
[
  {
    _id: ObjectId("6205f95aff06e556459b05ac"),
    address: {
      building: '129',
      coord: [ -73.962943, 40.685007 ],
      street: 'Gates Avenue',
      zipcode: '11238'
    },
    borough: 'Brooklyn',
    cuisine: 'Italian',
    grades: [
      {
        date: ISODate("2014-03-06T00:00:00.000Z"),
        grade: 'A',
        score: 5
      },
      {
        date: ISODate("2013-08-29T00:00:00.000Z"),
        grade: 'A',
        score: 2
      },
      {
        date: ISODate("2013-03-08T00:00:00.000Z"),
        grade: 'A',
        score: 7
      },
      {
        date: ISODate("2012-06-27T00:00:00.000Z"),
        grade: 'A',
        score: 7
      },
      {
        date: ISODate("2011-11-17T00:00:00.000Z"),
        grade: 'A',
        score: 12
      }
    ],
    name: '(Lewis Drug Store) Locanda Vini E Olii',
    restaurant_id: '40804423'
  },
  {
    _id: ObjectId("6205f959ff06e556459af929"),
    address: {
      building: '1',
      coord: [ -73.96926909999999, 40.7685235 ],
      street: 'East   66 Street',
      zipcode: '10065'
```

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.

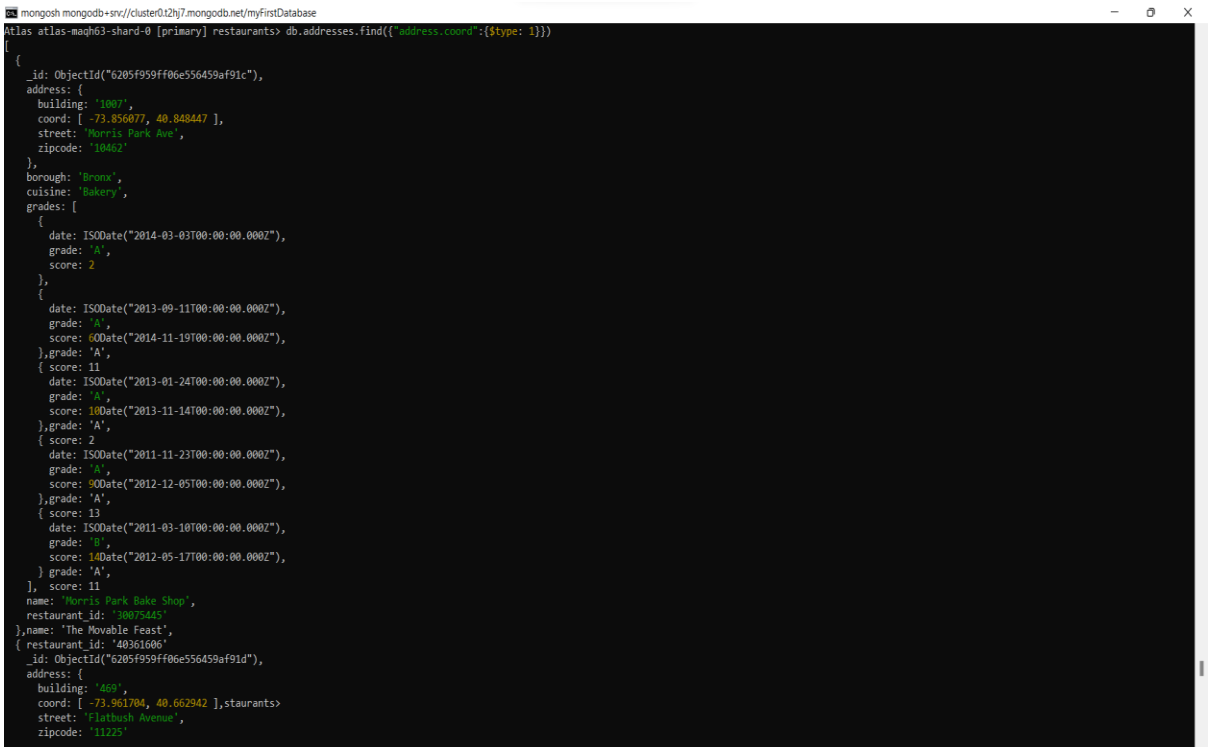


```
mongosh mongodb+srv://cluster0.t2hj7.mongodb.net/myFirstDatabase
Atlas atlas-maqh63-shard-0 [primary] restaurants> db.addresses.find().sort({'cusine':1,'borough':-1})
[
  {
    _id: ObjectId("6205f959ff06e556459af922"),
    address: {
      building: '2206',
      coord: [ -74.1377286, 40.6119572 ],
      street: 'Victory Boulevard',
      zipcode: '10314'
    },
    borough: 'Staten Island',
    cuisine: 'Jewish/Kosher',
    grades: [
      {
        date: ISODate("2014-10-06T00:00:00.000Z"),
        grade: 'A',
        score: 9
      },
      {
        date: ISODate("2014-05-20T00:00:00.000Z"),
        grade: 'A',
        score: 12
      },
      {
        date: ISODate("2013-04-04T00:00:00.000Z"),
        grade: 'A',
        score: 12
      },
      {
        date: ISODate("2012-01-24T00:00:00.000Z"),
        grade: 'A',
        score: 9
      }
    ],
    name: 'Kosher Island',
    restaurant_id: '40356442'
  },
  {
    _id: ObjectId("6205f959ff06e556459af943"),
    address: {
      building: '2491',
      coord: [ -74.1459332, 40.6103714 ],
      street: 'Victory Boulevard',
      zipcode: '10314'
    },
    borough: 'Staten Island',
    cuisine: 'Delicatessen',
    grades: [
      {
```

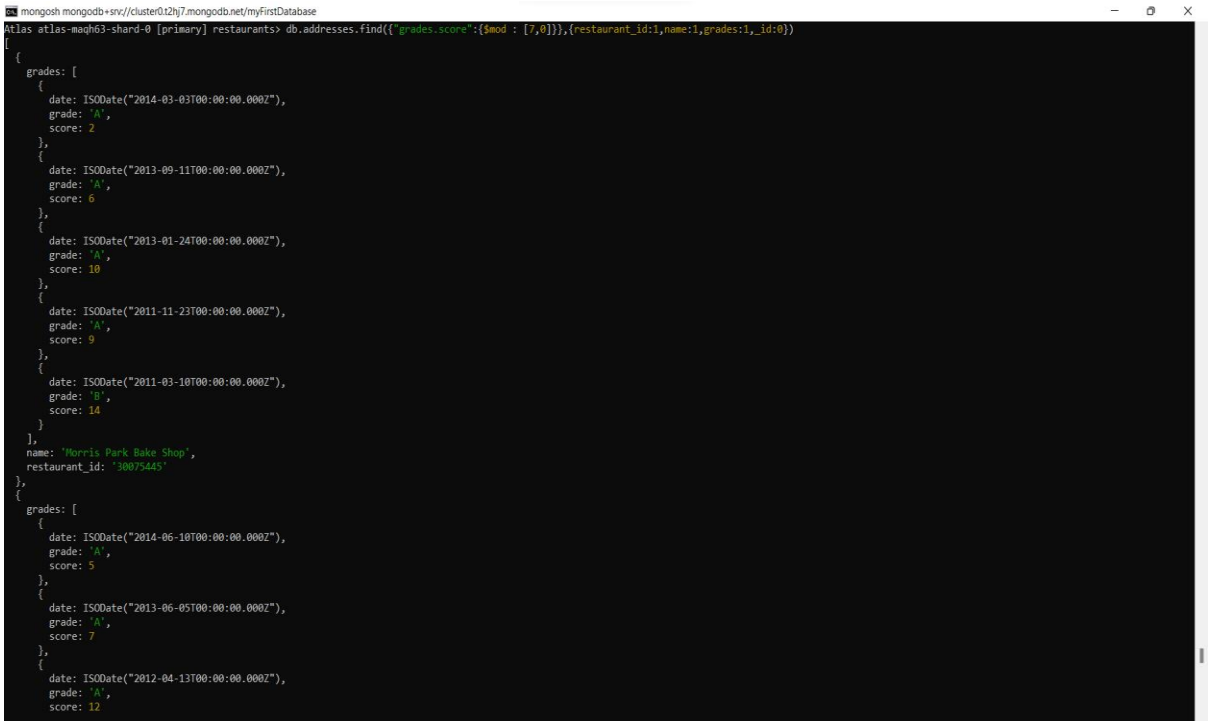
28. Write a MongoDB query to know whether all the addresses contains the street or not.



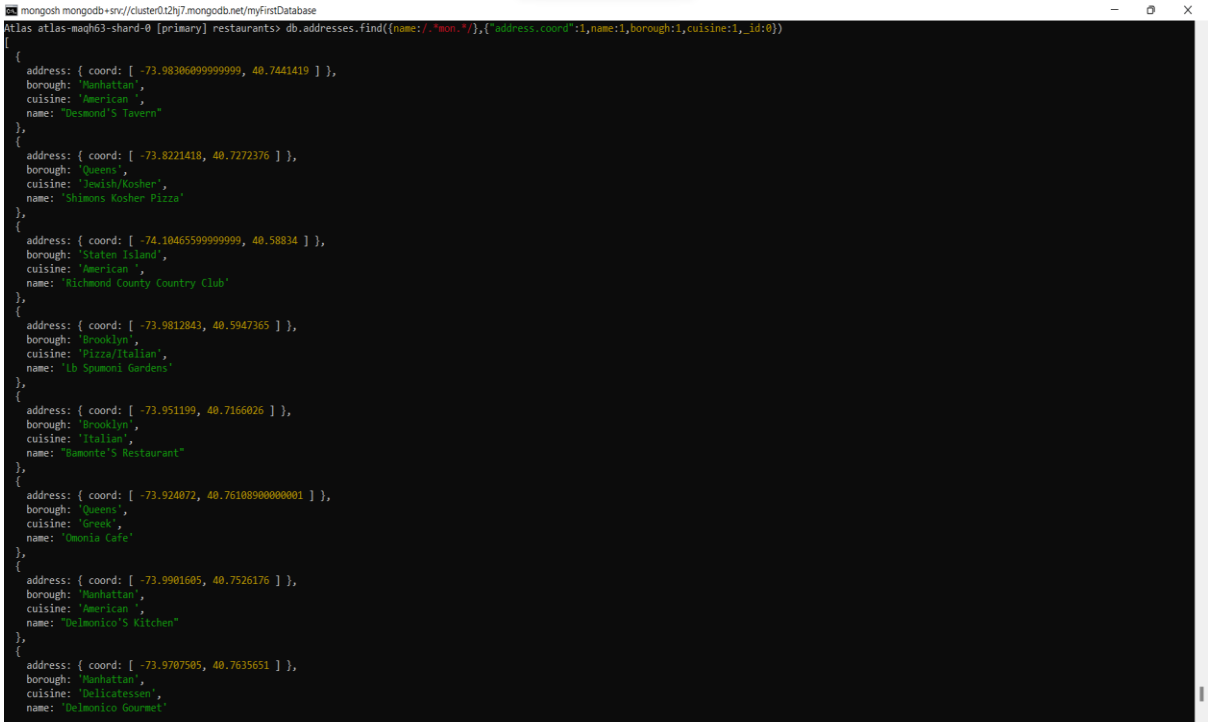
29. Write a MongoDB query which will select all documents in the restaurants collection where the coord field value is Double.



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.



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.



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.