Execute the following commands and in a table explain what each one is for
https://docs.mongodb.com/manual/reference/command/

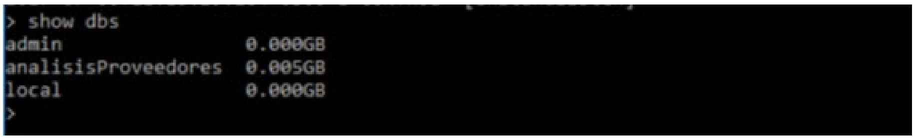
show dbs	If you want to check your databases list, use the command show dbs
use	If you want to use a database with name <mydb>, then use DATABASE statement

Read the reference manual and write down the function of the following:

aggregate	Performs aggregation tasks such as $group using an aggregation pipeline.
count	Counts the number of documents in a collection or a view.
distinct	Displays the distinct values found for a specified key in a collection or a view.
mapReduce	Performs map-reduce aggregation for large data sets.
delete	Deletes one or more documents.
find	Selects documents in a collection or a view.
findAndModify	Returns and modifies a single document.
getMore	Returns batches of documents currently pointed to by the cursor.
insert	Inserts one or more documents.
resetError	*Removed in MongoDB 5.0.* Resets the last error status.
update	Updates one or more documents.
planCacheClear	Removes cached query plan(s) for a collection.
planCacheClearFilters	Clears index filter(s) for a collection.
planCacheListFilters	Lists the index filters for a collection.
planCacheSetFilter	Sets an index filter for a collection.
authenticate	Starts an authenticated session using a username and password.
getnonce	This is an internal command to generate a one-time password for authentication.
logout	Terminates the current authenticated session.
createUser	Creates a new user.
dropAllUsersFromDatabase	Deletes all users associated with a database.
dropUser	Removes a single user.
grantRolesToUser	Grants a role and its privileges to a user.
revokeRolesFromUser	Removes a role from a user.
updateUser	Updates a user's data.
usersInfo	Returns information about the specified users.
createRole	Creates a role and specifies its privileges.
dropRole	Deletes the user-defined role.
dropAllRolesFromDatabase	Deletes all user-defined roles from a database.
grantPrivilegesToRole	Assigns privileges to a user-defined role.
grantRolesToRole	Specifies roles from which a user-defined role inherits privileges.

invalidateUserCache	Flushes the in-memory cache of user information, including credentials and roles.
revokePrivilegesFromRole	Removes the specified privileges from a user-defined role.
revokeRolesFromRole	Removes specified inherited roles from a user-defined role.
rolesInfo	Returns information for the specified role or roles.
create	Creates a collection or a view.
createIndexes	Builds one or more indexes for a collection.
currentOp	Returns a document that contains information on in-progress operations for the database instance.
drop	Removes the specified collection from the database.
dropDatabase	Removes the current database.
dropConnections	Drops outgoing connections to the specified list of hosts.
dropIndexes	Removes indexes from a collection.
listCollections	Returns a list of collections in the current database.
listDatabases	Returns a document that lists all databases and returns basic database statistics.
listIndexes	Lists all indexes for a collection.
renameCollection	Changes the name of an existing collection.
shutdown	Shuts down the mongod or mongos process.

Setup databases (Show dbs) :

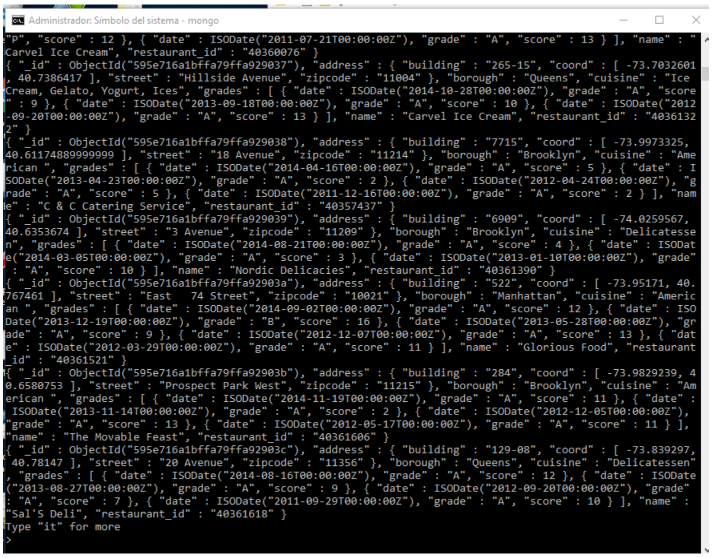


```
> show dbs
admin                  0.000GB
analisisProveedores    0.005GB
local                  0.000GB
>
```

Results query (Get providers ordered by borough and zipcode):

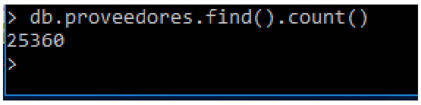
1. db.providers.insert({ "address" : { "street" : "2 Avenue", "zipcode" : "10075", "building" : "1480", "coord" : [- 73.9557413, 40.7720266] },
   "borough" : "Manhattan", "cuisine" : "Italian", "grades" : [{ "date" : ISODate("2014-10-01T00:00:00Z"), "grade" : "A", "score" : 11}, { "date" : ISODate("2014-01-16T00:00:00Z"),"grade" : "B", "score" : 17}],
   "name" : "Vella",

2. Display all the documents of the providers collection (db.providers.find())



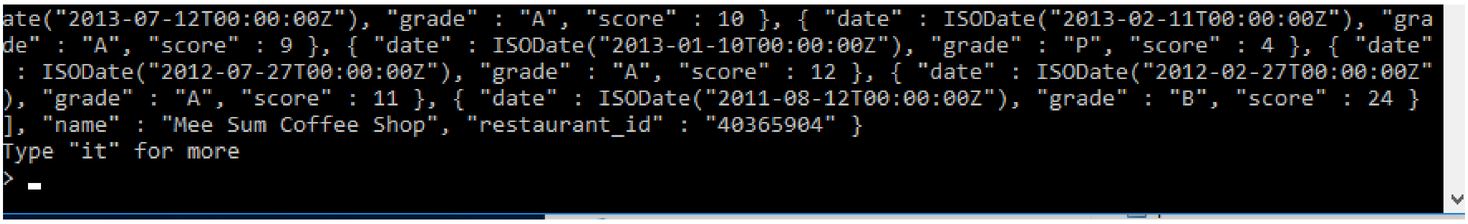
```
"P", "score" : 12 }, { "date" : ISODate("2011-07-21T00:00:00Z"), "grade" : "A", "score" : 13 } ], "name" : "
Carvel Ice Cream", "restaurant_id" : "40360076" }
{ "_id" : ObjectId("595e716a1bffa79ffa929037"), "address" : { "building" : "265-15", "coord" : [ -73.7032601
, 40.7386417 ], "street" : "Hillside Avenue", "zipcode" : "11004" }, "borough" : "Queens", "cuisine" : "Ice
Cream, Gelato, Yogurt, Ices", "grades" : [ { "date" : ISODate("2014-10-28T00:00:00Z"), "grade" : "A", "score
" : 9 }, { "date" : ISODate("2013-09-18T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2012
-09-20T00:00:00Z"), "grade" : "A", "score" : 13 } ], "name" : "Carvel Ice Cream", "restaurant_id" : "4036132
2" }
{ "_id" : ObjectId("595e716a1bffa79ffa929038"), "address" : { "building" : "7715", "coord" : [ -73.9973325,
40.61174889999999 ], "street" : "18 Avenue", "zipcode" : "11214" }, "borough" : "Brooklyn", "cuisine" : "Ame
rican ", "grades" : [ { "date" : ISODate("2014-04-16T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : I
SODate("2013-04-23T00:00:00Z"), "grade" : "A", "score" : 2 }, { "date" : ISODate("2012-04-24T00:00:00Z"), "g
rade" : "A", "score" : 5 }, { "date" : ISODate("2011-12-16T00:00:00Z"), "grade" : "A", "score" : 2 } ], "nam
e" : "C & C Catering Service", "restaurant_id" : "40357437" }
{ "_id" : ObjectId("595e716a1bffa79ffa929039"), "address" : { "building" : "6909", "coord" : [ -74.0259567,
40.6353674 ], "street" : "3 Avenue", "zipcode" : "11209" }, "borough" : "Brooklyn", "cuisine" : "Delicatesse
n", "grades" : [ { "date" : ISODate("2014-08-21T00:00:00Z"), "grade" : "A", "score" : 4 }, { "date" : ISODat
e("2014-03-05T00:00:00Z"), "grade" : "A", "score" : 3 }, { "date" : ISODate("2013-01-10T00:00:00Z"), "grade"
 : "A", "score" : 10 } ], "name" : "Nordic Delicacies", "restaurant_id" : "40361390" }
{ "_id" : ObjectId("595e716a1bffa79ffa92903a"), "address" : { "building" : "522", "coord" : [ -73.95171, 40.
767461 ], "street" : "East   74 Street", "zipcode" : "10021" }, "borough" : "Manhattan", "cuisine" : "Americ
an ", "grades" : [ { "date" : ISODate("2014-09-02T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISO
Date("2013-12-19T00:00:00Z"), "grade" : "B", "score" : 16 }, { "date" : ISODate("2013-05-28T00:00:00Z"), "gr
ade" : "A", "score" : 9 }, { "date" : ISODate("2012-12-07T00:00:00Z"), "grade" : "A", "score" : 13 }, { "dat
e" : ISODate("2012-03-29T00:00:00Z"), "grade" : "A", "score" : 11 } ], "name" : "Glorious Food", "restaurant
_id" : "40361521" }
{ "_id" : ObjectId("595e716a1bffa79ffa92903b"), "address" : { "building" : "284", "coord" : [ -73.9829239, 4
0.6580753 ], "street" : "Prospect Park West", "zipcode" : "11215" }, "borough" : "Brooklyn", "cuisine" : "Am
erican ", "grades" : [ { "date" : ISODate("2014-11-19T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" :
 ISODate("2013-11-14T00:00:00Z"), "grade" : "A", "score" : 2 }, { "date" : ISODate("2012-12-05T00:00:00Z"),
"grade" : "A", "score" : 13 }, { "date" : ISODate("2012-05-17T00:00:00Z"), "grade" : "A", "score" : 11 } ],
"name" : "The Movable Feast", "restaurant_id" : "40361606" }
{ "_id" : ObjectId("595e716a1bffa79ffa92903c"), "address" : { "building" : "129-08", "coord" : [ -73.839297,
 40.78147 ], "street" : "20 Avenue", "zipcode" : "11356" }, "borough" : "Queens", "cuisine" : "Delicatessen"
, "grades" : [ { "date" : ISODate("2014-08-16T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate
("2013-08-27T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2012-09-20T00:00:00Z"), "grade"
: "A", "score" : 7 }, { "date" : ISODate("2011-09-29T00:00:00Z"), "grade" : "A", "score" : 10 } ], "name" :
"Sal'S Deli", "restaurant_id" : "40361618" }
Type "it" for more
>
```

3. Count the number of documents stored in providers collection



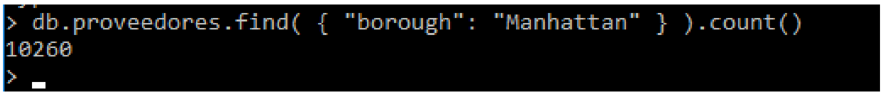
```
> db.proveedores.find().count()
25360
>
```

4. For establishing conditions: Find the restaurant providers from Manhattan
db.providers.find({ "borough": "Manhattan" })



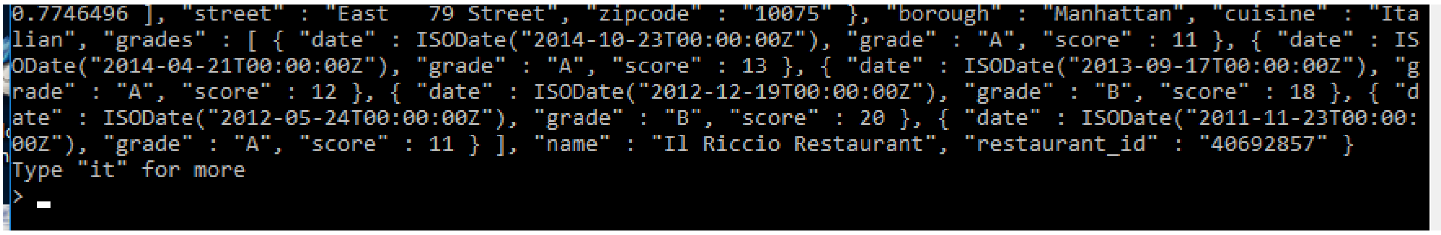
```
ate("2013-07-12T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2013-02-11T00:00:00Z"), "gra
de" : "A", "score" : 9 }, { "date" : ISODate("2013-01-10T00:00:00Z"), "grade" : "P", "score" : 4 }, { "date"
 : ISODate("2012-07-27T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-02-27T00:00:00Z"
), "grade" : "A", "score" : 11 }, { "date" : ISODate("2011-08-12T00:00:00Z"), "grade" : "B", "score" : 24 }
], "name" : "Mee Sum Coffee Shop", "restaurant_id" : "40365904" }
Type "it" for more
>
```

5. For counting documents: Get the number of restaurant providers from Manhattan



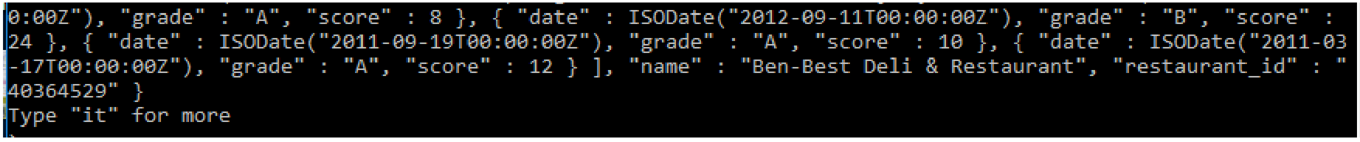
```
> db.proveedores.find( { "borough": "Manhattan" } ).count()
10260
>
```

6. For querying subdocument fields: Get the restaurant providers that have a zipcode 10075



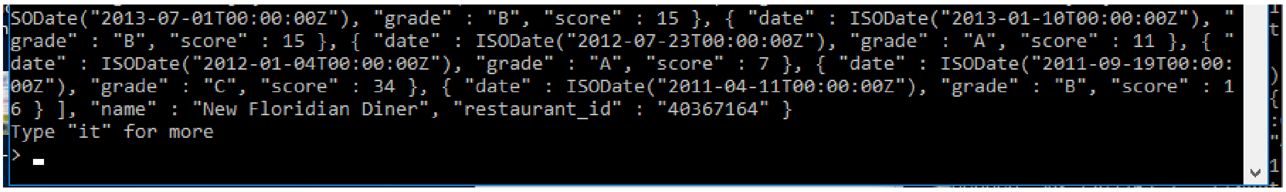
```
0.7746496 ], "street" : "East   79 Street", "zipcode" : "10075" }, "borough" : "Manhattan", "cuisine" : "Ita
lian", "grades" : [ { "date" : ISODate("2014-10-23T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : IS
ODate("2014-04-21T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2013-09-17T00:00:00Z"), "g
rade" : "A", "score" : 12 }, { "date" : ISODate("2012-12-19T00:00:00Z"), "grade" : "B", "score" : 18 }, { "d
ate" : ISODate("2012-05-24T00:00:00Z"), "grade" : "B", "score" : 20 }, { "date" : ISODate("2011-11-23T00:00:
00Z"), "grade" : "A", "score" : 11 } ], "name" : "Il Riccio Restaurant", "restaurant_id" : "40692857" }
Type "it" for more
>
```

7. - For querying array fields: Get the providers with grade B



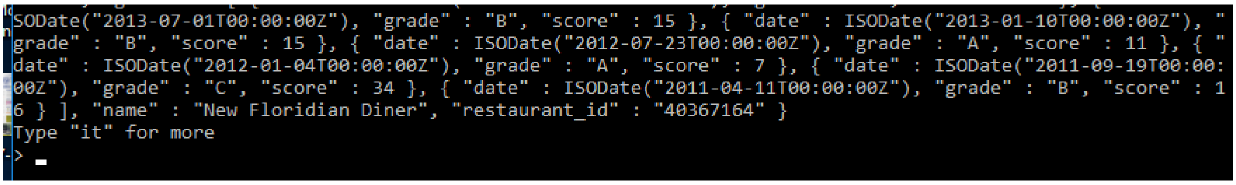
```
0:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2012-09-11T00:00:00Z"), "grade" : "B", "score" :
24 }, { "date" : ISODate("2011-09-19T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2011-03
-17T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "Ben-Best Deli & Restaurant", "restaurant_id" : "
40364529" }
Type "it" for more
```

8. Using operators: Get the providers that have a score greater than 30



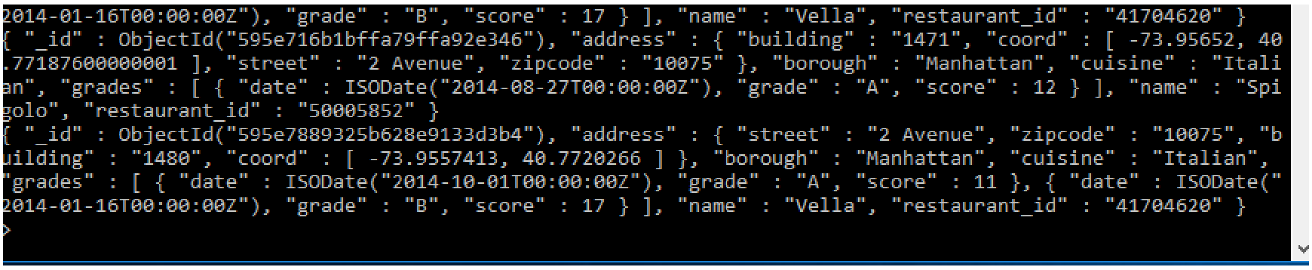
```
SODate("2013-07-01T00:00:00Z"), "grade" : "B", "score" : 15 }, { "date" : ISODate("2013-01-10T00:00:00Z"), "
grade" : "B", "score" : 15 }, { "date" : ISODate("2012-07-23T00:00:00Z"), "grade" : "A", "score" : 11 }, { "
date" : ISODate("2012-01-04T00:00:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2011-09-19T00:00:
00Z"), "grade" : "C", "score" : 34 }, { "date" : ISODate("2011-04-11T00:00:00Z"), "grade" : "B", "score" : 1
6 } ], "name" : "New Floridian Diner", "restaurant_id" : "40367164" }
Type "it" for more
>
```

Get the providers that have a score less than 10



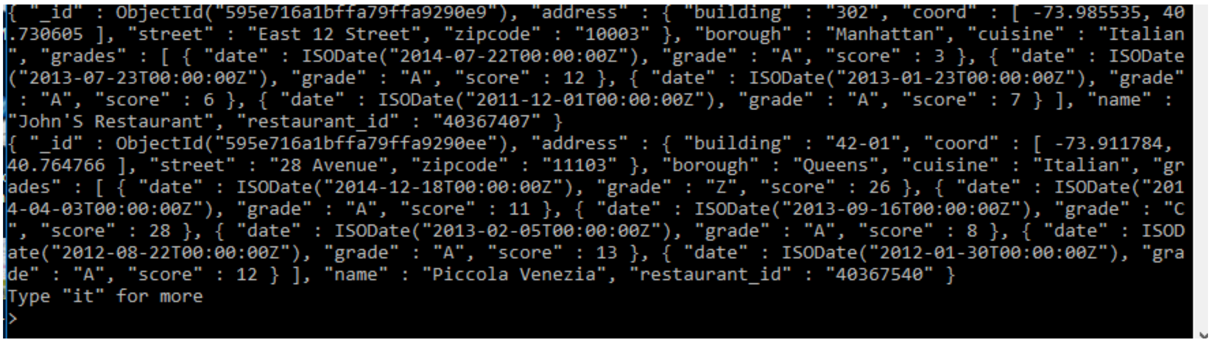
```
SODate("2013-07-01T00:00:00Z"), "grade" : "B", "score" : 15 }, { "date" : ISODate("2013-01-10T00:00:00Z"), "
grade" : "B", "score" : 15 }, { "date" : ISODate("2012-07-23T00:00:00Z"), "grade" : "A", "score" : 11 }, { "
date" : ISODate("2012-01-04T00:00:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2011-09-19T00:00:
00Z"), "grade" : "C", "score" : 34 }, { "date" : ISODate("2011-04-11T00:00:00Z"), "grade" : "B", "score" : 1
6 } ], "name" : "New Floridian Diner", "restaurant_id" : "40367164" }
Type "it" for more
>
```

9. Find providers of Italian cuisine and zipcode 10075



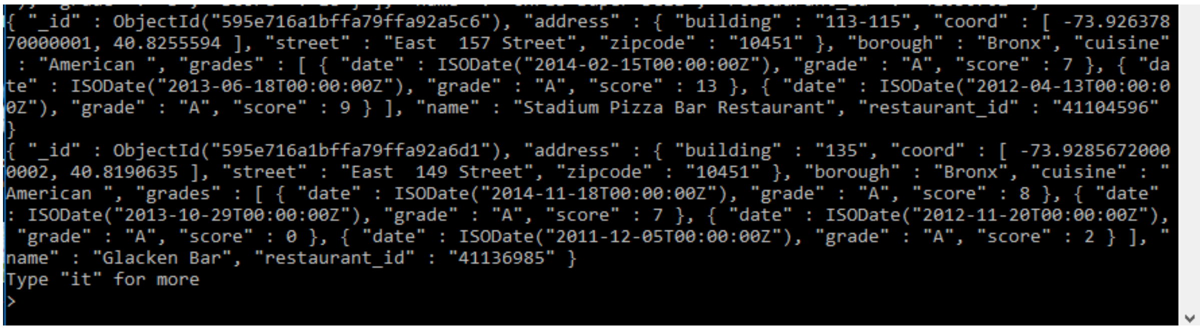
```
2014-01-16T00:00:00Z"), "grade" : "B", "score" : 17 } ], "name" : "Vella", "restaurant_id" : "41704620" }
{ "_id" : ObjectId("595e716b1bffa79ffa92e346"), "address" : { "building" : "1471", "coord" : [ -73.95652, 40
.77187600000001 ], "street" : "2 Avenue", "zipcode" : "10075" }, "borough" : "Manhattan", "cuisine" : "Itali
an", "grades" : [ { "date" : ISODate("2014-08-27T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "Spi
golo", "restaurant_id" : "50005852" }
{ "_id" : ObjectId("595e7889325b628e9133d3b4"), "address" : { "street" : "2 Avenue", "zipcode" : "10075", "b
uilding" : "1480", "coord" : [ -73.9557413, 40.7720266 ] }, "borough" : "Manhattan", "cuisine" : "Italian",
"grades" : [ { "date" : ISODate("2014-10-01T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("
2014-01-16T00:00:00Z"), "grade" : "B", "score" : 17 } ], "name" : "Vella", "restaurant_id" : "41704620" }
>
```

10. Find providers of Italian cuisine or those with zipcode 10075



```
{ "_id" : ObjectId("595e716a1bffa79ffa9290e9"), "address" : { "building" : "302", "coord" : [ -73.985535, 40
.730605 ], "street" : "East 12 Street", "zipcode" : "10003" }, "borough" : "Manhattan", "cuisine" : "Italian
", "grades" : [ { "date" : ISODate("2014-07-22T00:00:00Z"), "grade" : "A", "score" : 3 }, { "date" : ISODate
("2013-07-23T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-01-23T00:00:00Z"), "grade"
 : "A", "score" : 6 }, { "date" : ISODate("2011-12-01T00:00:00Z"), "grade" : "A", "score" : 7 } ], "name" :
"John'S Restaurant", "restaurant_id" : "40367407" }
{ "_id" : ObjectId("595e716a1bffa79ffa9290ee"), "address" : { "building" : "42-01", "coord" : [ -73.911784,
40.764766 ], "street" : "28 Avenue", "zipcode" : "11103" }, "borough" : "Queens", "cuisine" : "Italian", "gr
ades" : [ { "date" : ISODate("2014-12-18T00:00:00Z"), "grade" : "Z", "score" : 26 }, { "date" : ISODate("201
4-04-03T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2013-09-16T00:00:00Z"), "grade" : "C
", "score" : 28 }, { "date" : ISODate("2013-02-05T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISOD
ate("2012-08-22T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2012-01-30T00:00:00Z"), "gra
de" : "A", "score" : 12 } ], "name" : "Piccola Venezia", "restaurant_id" : "40367540" }
Type "it" for more
>
```

11. For ordering results: Get providers ordered by borough and zipcode



```
{ "_id" : ObjectId("595e716a1bffa79ffa92a5c6"), "address" : { "building" : "113-115", "coord" : [ -73.926378
70000001, 40.8255594 ], "street" : "East  157 Street", "zipcode" : "10451" }, "borough" : "Bronx", "cuisine"
 : "American ", "grades" : [ { "date" : ISODate("2014-02-15T00:00:00Z"), "grade" : "A", "score" : 7 }, { "da
te" : ISODate("2013-06-18T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2012-04-13T00:00:0
0Z"), "grade" : "A", "score" : 9 } ], "name" : "Stadium Pizza Bar Restaurant", "restaurant_id" : "41104596"
}
{ "_id" : ObjectId("595e716a1bffa79ffa92a6d1"), "address" : { "building" : "135", "coord" : [ -73.9285672000
0002, 40.8190635 ], "street" : "East  149 Street", "zipcode" : "10451" }, "borough" : "Bronx", "cuisine" : "
American ", "grades" : [ { "date" : ISODate("2014-11-18T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date"
: ISODate("2013-10-29T00:00:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2012-11-20T00:00:00Z"),
 "grade" : "A", "score" : 0 }, { "date" : ISODate("2011-12-05T00:00:00Z"), "grade" : "A", "score" : 2 } ], "
name" : "Glacken Bar", "restaurant_id" : "41136985" }
Type "it" for more
>
```