

MongoDB Assignments:

Assignment:-

```
Microsoft Windows [Version 10.0.19043.1110]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Dell>mongo
MongoDB shell version v5.0.1
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("422257f9-9420-4f7d-ab5b-8108349a2093") }
MongoDB server version: 5.0.1

=====
Warning: the "mongo" shell has been superseded by "mongosh",
which delivers improved usability and compatibility. The "mongo" shell has been deprecated and will be removed in
an upcoming release.
We recommend you begin using "mongosh".
For installation instructions, see
https://docs.mongodb.com/mongodb-shell/install/
=====

The server generated these startup warnings when booting:
2021-08-01T14:23:36.198+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted

-----
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()

-----
> use population
switched to db population
> show collections
zipcode
> dp.zipcode.aggregate([{$match: {city:"ATLANTA",state:"GA"}}]).pretty();
uncaught exception: ReferenceError: dp is not defined :
@(_shell):1:1
> db.zipcode.aggregate([{$match: {city:"ATLANTA",state:"GA"}}]).pretty();
{
  "_id" : "30303",
  "city" : "ATLANTA",
  "loc" : [
    -84.388846,
    33.752504
  ],
  "pop" : 1845,
  "state" : "GA"
}
```

```
-----
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()

-----
> use population
switched to db population
> show collections
zipcode
> dp.zipcode.aggregate([{$match: {city:"ATLANTA",state:"GA"}}]).pretty();
uncaught exception: ReferenceError: dp is not defined :
@(_shell):1:1
> db.zipcode.aggregate([{$match: {city:"ATLANTA",state:"GA"}}]).pretty();
{
  "_id" : "30303",
  "city" : "ATLANTA",
  "loc" : [
    -84.388846,
    33.752504
  ],
  "pop" : 1845,
  "state" : "GA"
},
{
  "_id" : "30305",
  "city" : "ATLANTA",
  "loc" : [
    -84.385145,
    33.831963
  ],
  "pop" : 19122,
  "state" : "GA"
},
{
  "_id" : "30306",
  "city" : "ATLANTA",
  "loc" : [
    -84.351418,
    33.786827
  ],
  "pop" : 20081,
  "state" : "GA"
}
```



```

    33.823555
  },
  "pop": 17013,
  "state": "GA"
}
Type "it" for more
> db.zipcode.aggregate([{$match:{city:"ATLANTA"}},{$group:{_id:{city:"$city",zipcode:"$id"}},{$count:"zipcode"}}]
{
  "zipcode": 41
}
> db.zipcode.aggregate([{$match:{city:"ATLANTA"}},{$group:{_id:{city:"$city"}, "pop": {$sum:"$pop"}}}])
{
  "_id": { "city": "ATLANTA", "pop": 638846 }
}
> db.zipcode.aggregate([{$group:{_id:"$state",population:{$sum:"$pop"}}}])
{
  "_id": "DC", "population": 606968 }
{
  "_id": "WA", "population": 6016425 }
{
  "_id": "TX", "population": 4876457 }
{
  "_id": "DE", "population": 666168 }
{
  "_id": "ID", "population": 1006749 }
{
  "_id": "OR", "population": 2042321 }
{
  "_id": "IA", "population": 2776420 }
{
  "_id": "VA", "population": 6181479 }
{
  "_id": "SO", "population": 995397 }
{
  "_id": "HI", "population": 1515069 }
{
  "_id": "WY", "population": 1793146 }
{
  "_id": "MS", "population": 2573216 }
{
  "_id": "RI", "population": 1003218 }
{
  "_id": "FL", "population": 12686644 }
{
  "_id": "LA", "population": 4217595 }
{
  "_id": "NH", "population": 2350725 }
{
  "_id": "OK", "population": 6628637 }
{
  "_id": "OH", "population": 10846517 }
{
  "_id": "SC", "population": 3486701 }
{
  "_id": "VT", "population": 562758 }
}
Type "it" for more
> it
{
  "_id": "ND", "population": 638272 }
{
  "_id": "TX", "population": 16984601 }
{
  "_id": "HI", "population": 4372982 }
{
  "_id": "AK", "population": 544698 }
{
  "_id": "HI", "population": 1109252 }
{
  "_id": "PA", "population": 11881643 }
{
  "_id": "MI", "population": 4891769 }
{
  "_id": "WY", "population": 17990482 }
{
  "_id": "CT", "population": 3287216 }
{
  "_id": "IL", "population": 11427576 }
{
  "_id": "ND", "population": 4781379 }
{
  "_id": "ND", "population": 7730188 }
}

```

```

{
  "_id": "NE", "population": 5110648 }
{
  "_id": "KY", "population": 1578139 }
{
  "_id": "AZ", "population": 3675484 }
{
  "_id": "HI", "population": 3665228 }
{
  "_id": "OK", "population": 9295297 }
{
  "_id": "CO", "population": 3145585 }
{
  "_id": "CO", "population": 3293755 }
{
  "_id": "NV", "population": 1201833 }
{
  "_id": "CA", "population": 29754898 }
{
  "_id": "UT", "population": 1722858 }
}
> db.zipcode.aggregate([{$group:{_id:{state:"$state"},pop:{$sum:"$pop"}},{$sort:{pop:-1}}])
uncaught exception: Error: command failed: {
  "ok": 0,
  "errmsg": "Unrecognized pipeline stage name: 'sort'",
  "code": 40324,
  "codeName": "Location40324"
} with original command request: {
  "aggregate": "zipcode",
  "pipeline": [
    {
      "$group": {
        "_id": {
          "state": "$state"
        },
        "pop": {
          "$sum": "$pop"
        }
      }
    },
    {
      "$sort": {
        "pop": -1
      }
    }
  ],
  "cursor": {
    "isid": {
      "id": "UUID(\"a22257f9-9420-4f7d-ab5b-8108349a2093\")"
    }
  }
} on connection: connection to 127.0.0.1:27017 : aggregate failed :
getErrorMessage@src/mongo/shell/quit.js:25:13
assert@src/mongo/shell/assert.js:18:14

```



```

    },
    "lsid" : {
      "id" : UUID("422257f9-9420-4f7d-ab5b-8108349a2093")
    }
  } on connection: connection to 127.0.0.1:27017 : aggregate failed :
_getErrorWithCode@src/mongo/shell/utils.js:25:13
doassert@src/mongo/shell/assert.js:18:14
_assertCommandWorked@src/mongo/shell/assert.js:719:17
assert.commandWorked@src/mongo/shell/assert.js:811:16
DB.prototype._runAggregate@src/mongo/shell/db.js:276:5
DBCollection.prototype.aggregate@src/mongo/shell/collection.js:1058:12
@(shell):1:1
> db.zipcode.aggregate([{$group:{_id:{state:"$state"},pop:{$sum:"$pop"}},{$sort:{pop:-1}}])
{ "_id" : { "state" : "CA" }, "pop" : 29754890 }
{ "_id" : { "state" : "NY" }, "pop" : 17990402 }
{ "_id" : { "state" : "TX" }, "pop" : 16984601 }
{ "_id" : { "state" : "FL" }, "pop" : 12686644 }
{ "_id" : { "state" : "PA" }, "pop" : 11881643 }
{ "_id" : { "state" : "IL" }, "pop" : 11427576 }
{ "_id" : { "state" : "OH" }, "pop" : 10846517 }
{ "_id" : { "state" : "MI" }, "pop" : 9295297 }
{ "_id" : { "state" : "NJ" }, "pop" : 7730188 }
{ "_id" : { "state" : "NC" }, "pop" : 6628637 }
{ "_id" : { "state" : "GA" }, "pop" : 6478216 }
{ "_id" : { "state" : "VA" }, "pop" : 6181479 }
{ "_id" : { "state" : "MA" }, "pop" : 6016425 }
{ "_id" : { "state" : "IN" }, "pop" : 5544136 }
{ "_id" : { "state" : "NO" }, "pop" : 5110648 }
{ "_id" : { "state" : "WI" }, "pop" : 4891769 }
{ "_id" : { "state" : "TN" }, "pop" : 4876457 }
{ "_id" : { "state" : "WA" }, "pop" : 4866692 }
{ "_id" : { "state" : "MD" }, "pop" : 4781379 }
{ "_id" : { "state" : "HI" }, "pop" : 4372982 }
Type "it" for more
> it
{ "_id" : { "state" : "LA" }, "pop" : 4217595 }
{ "_id" : { "state" : "AL" }, "pop" : 4040587 }
{ "_id" : { "state" : "KY" }, "pop" : 3675484 }
{ "_id" : { "state" : "AZ" }, "pop" : 3665228 }
{ "_id" : { "state" : "SC" }, "pop" : 3486703 }
{ "_id" : { "state" : "CO" }, "pop" : 3293755 }
{ "_id" : { "state" : "CT" }, "pop" : 3287116 }
{ "_id" : { "state" : "OK" }, "pop" : 3145585 }
{ "_id" : { "state" : "OR" }, "pop" : 2842321 }
{ "_id" : { "state" : "IA" }, "pop" : 2776420 }
{ "_id" : { "state" : "MS" }, "pop" : 2573216 }

```

```

{ "_id" : { "state" : "SD" }, "pop" : 695397 }
{ "_id" : { "state" : "DE" }, "pop" : 666168 }
{ "_id" : { "state" : "ND" }, "pop" : 638272 }
{ "_id" : { "state" : "DC" }, "pop" : 606900 }
{ "_id" : { "state" : "VT" }, "pop" : 562758 }
{ "_id" : { "state" : "AK" }, "pop" : 544698 }
{ "_id" : { "state" : "WY" }, "pop" : 453528 }
> db.zipcode.aggregate([{$group:{_id:{state:"$state"},pop:{$sum:"$pop"}},{$sort:{pop:-1}},{$limit:3}}])
{ "_id" : { "state" : "CA" }, "pop" : 29754890 }
{ "_id" : { "state" : "NY" }, "pop" : 17990402 }
{ "_id" : { "state" : "TX" }, "pop" : 16984601 }
> db.zipcode.aggregate([{$group:{_id:{_id:"$id",city:"$city",state:"$state"},pop:{$sum:"$pop"}},{$sort:{_id:-1}}])
uncaught exception: SyntaxError: missing ] after element list
@(shell):1:88
> db.zipcode.aggregate([{$group:{_id:{_id:"$id",city:"$city",state:"$state"},pop:{$sum:"$pop"}},{$sort:{_id:-1}}])
{ "_id" : { "city" : "ZWOLLE", "state" : "LA" }, "pop" : 5325 }
{ "_id" : { "city" : "ZWINGLE", "state" : "IA" }, "pop" : 1320 }
{ "_id" : { "city" : "ZURICH", "state" : "KS" }, "pop" : 318 }
{ "_id" : { "city" : "ZUNI", "state" : "VA" }, "pop" : 2162 }
{ "_id" : { "city" : "ZUNI", "state" : "NM" }, "pop" : 7382 }
{ "_id" : { "city" : "ZUMBROTA", "state" : "MN" }, "pop" : 3535 }
{ "_id" : { "city" : "ZORTMAN", "state" : "MT" }, "pop" : 394 }
{ "_id" : { "city" : "ZOLFO SPRINGS", "state" : "FL" }, "pop" : 3515 }
{ "_id" : { "city" : "ZOE", "state" : "KY" }, "pop" : 190 }
{ "_id" : { "city" : "ZOARVILLE", "state" : "OH" }, "pop" : 2969 }
{ "_id" : { "city" : "ZIRCONIA", "state" : "NC" }, "pop" : 2350 }
{ "_id" : { "city" : "ZIONVILLE", "state" : "PA" }, "pop" : 1633 }
{ "_id" : { "city" : "ZIONSVILLE", "state" : "IN" }, "pop" : 2312 }
{ "_id" : { "city" : "ZIONSVILLE", "state" : "IN" }, "pop" : 9759 }
{ "_id" : { "city" : "ZION GROVE", "state" : "PA" }, "pop" : 2070 }
{ "_id" : { "city" : "ZION", "state" : "IL" }, "pop" : 24944 }
{ "_id" : { "city" : "ZION", "state" : "AR" }, "pop" : 1880 }
{ "_id" : { "city" : "ZIMMERMAN", "state" : "MN" }, "pop" : 5786 }
{ "_id" : { "city" : "ZIM", "state" : "MN" }, "pop" : 507 }
{ "_id" : { "city" : "ZILLAH", "state" : "WA" }, "pop" : 5266 }
Type "it" for more
> it
{ "_id" : { "city" : "ZIGZAG", "state" : "OR" }, "pop" : 1325 }
{ "_id" : { "city" : "ZIEGLERSVILLE", "state" : "PA" }, "pop" : 290 }
{ "_id" : { "city" : "ZIA PUEBLO", "state" : "NM" }, "pop" : 877 }
{ "_id" : { "city" : "ZEPHYRHILLS", "state" : "FL" }, "pop" : 39172 }
{ "_id" : { "city" : "ZEPHYR", "state" : "TX" }, "pop" : 433 }
{ "_id" : { "city" : "ZEONA", "state" : "SD" }, "pop" : 8 }
{ "_id" : { "city" : "ZENDA", "state" : "KS" }, "pop" : 200 }

```



```

{"_id": {"city": "YAZOO CITY", "state": "MS", "pop": 30411 }
{"_id": {"city": "YANKEY", "state": "WV", "pop": 17936 }
{"_id": {"city": "YATESVILLE", "state": "GA", "pop": 690 }
{"_id": {"city": "YATES CITY", "state": "IL", "pop": 1188 }
{"_id": {"city": "YATES CENTER", "state": "KS", "pop": 1402 }
{"_id": {"city": "YARNELL", "state": "AZ", "pop": 2859 }
{"_id": {"city": "YARMOUTH PORT", "state": "MA", "pop": 455 }
Type "it" for more
> db.zipcode.aggregate([{$group: {_id: {_id: "$id", city: "$city", state: "$state"}, pop: {$sum: "$pop"}}, {$sort: {pop: -1}}, {$limit: 3}}])
{"_id": {"city": "CHICAGO", "state": "IL", "pop": 2452177 }
{"_id": {"city": "BROOKLYN", "state": "NY", "pop": 2300504 }
{"_id": {"city": "LOS ANGELES", "state": "CA", "pop": 2102295 }
> db.zipcode.aggregate([{$group: {_id: {city: "TEXAS", pop: "$pop"}}, {$sort: {"_id.pop": -1}}, {$limit: 3}}])
uncaught exception: SyntaxError: missing ] after element list
@(:shell):1:68
> db.zipcode.aggregate([{$group: {_id: {city: "TEXAS", pop: "$pop"}}, {$sort: {"_id.pop": -1}}, {$limit: 3}}])
{"_id": {"city": "TEXAS", "pop": 112047 }
{"_id": {"city": "TEXAS", "pop": 111396 }
{"_id": {"city": "TEXAS", "pop": 106564 }
> db
population
> db.zipcode.aggregate([{$group: {_id: {state: "$state", city: "$city"}, AvgPop: {$avg: "$pop"}}}])
{"_id": {"state": "IN", "city": "PONETO", "AvgPop": 709 }
{"_id": {"state": "VA", "city": "COURTLAND", "AvgPop": 3570 }
{"_id": {"state": "IN", "city": "BOWLING GREEN", "AvgPop": 734 }
{"_id": {"state": "GA", "city": "BEMISS", "AvgPop": 32292 }
{"_id": {"state": "MN", "city": "WALKER", "AvgPop": 2431 }
{"_id": {"state": "SD", "city": "PEEVER", "AvgPop": 992 }
{"_id": {"state": "MD", "city": "MILLERSVILLE", "AvgPop": 16436 }
{"_id": {"state": "MN", "city": "EVAN", "AvgPop": 222 }
{"_id": {"state": "IL", "city": "CRETE", "AvgPop": 14372 }
{"_id": {"state": "OK", "city": "FITTSTOWN", "AvgPop": 759 }
{"_id": {"state": "OK", "city": "FORT TOWSON", "AvgPop": 2249 }
{"_id": {"state": "GA", "city": "COHUTTA", "AvgPop": 5052 }
{"_id": {"state": "ID", "city": "FAIRFIELD", "AvgPop": 668 }
{"_id": {"state": "MI", "city": "HUBBARDSTON", "AvgPop": 1407 }
{"_id": {"state": "IN", "city": "WINAMAC", "AvgPop": 6857 }
{"_id": {"state": "CA", "city": "LYNWOOD", "AvgPop": 61606 }
{"_id": {"state": "KY", "city": "OLDTOWN", "AvgPop": 690 }
{"_id": {"state": "WV", "city": "CAIRO", "AvgPop": 1379 }
{"_id": {"state": "IL", "city": "URBANA", "AvgPop": 46110 }
{"_id": {"state": "IN", "city": "GAS CITY", "AvgPop": 6975 }
Type "it" for more
> it
{"_id": {"state": "KS", "city": "OKETO", "AvgPop": 318 }
{"_id": {"state": "CA", "city": "NOVATO", "AvgPop": 18171 }
{"_id": {"state": "AR", "city": "REYDELL", "AvgPop": 150 }
{"_id": {"state": "IA", "city": "STOCKTON", "AvgPop": 777 }
{"_id": {"state": "NJ", "city": "NEWFOUNDLAND", "AvgPop": 1624 }
{"_id": {"state": "MO", "city": "HARTVILLE", "AvgPop": 3040 }
{"_id": {"state": "HI", "city": "WAIANAE", "AvgPop": 37518 }
{"_id": {"state": "AL", "city": "CORDOVA", "AvgPop": 5273 }
{"_id": {"state": "OH", "city": "KELLEYS ISLAND", "AvgPop": 172 }
{"_id": {"state": "KY", "city": "DAWSON SPRINGS", "AvgPop": 6728 }
{"_id": {"state": "OR", "city": "ZIGZAG", "AvgPop": 1325 }
{"_id": {"state": "MO", "city": "ARBELA", "AvgPop": 564 }
{"_id": {"state": "IL", "city": "BELLFLOWER", "AvgPop": 702 }
{"_id": {"state": "IL", "city": "DAVIS JUNCTION", "AvgPop": 1263 }

```

```

{"_id": {"city": "TEXAS", "pop": 106564 } }
> db
population
> db.zipcode.aggregate([{$group: {_id: {state: "$state", city: "$city"}, AvgPop: {$avg: "$pop"}}}])
{"_id": {"state": "IN", "city": "PONETO", "AvgPop": 709 }
{"_id": {"state": "VA", "city": "COURTLAND", "AvgPop": 3570 }
{"_id": {"state": "IN", "city": "BOWLING GREEN", "AvgPop": 734 }
{"_id": {"state": "GA", "city": "BEMISS", "AvgPop": 32292 }
{"_id": {"state": "MN", "city": "WALKER", "AvgPop": 2431 }
{"_id": {"state": "SD", "city": "PEEVER", "AvgPop": 992 }
{"_id": {"state": "MD", "city": "MILLERSVILLE", "AvgPop": 16436 }
{"_id": {"state": "MN", "city": "EVAN", "AvgPop": 222 }
{"_id": {"state": "IL", "city": "CRETE", "AvgPop": 14372 }
{"_id": {"state": "OK", "city": "FITTSTOWN", "AvgPop": 759 }
{"_id": {"state": "OK", "city": "FORT TOWSON", "AvgPop": 2249 }
{"_id": {"state": "GA", "city": "COHUTTA", "AvgPop": 5052 }
{"_id": {"state": "ID", "city": "FAIRFIELD", "AvgPop": 668 }
{"_id": {"state": "MI", "city": "HUBBARDSTON", "AvgPop": 1407 }
{"_id": {"state": "IN", "city": "WINAMAC", "AvgPop": 6857 }
{"_id": {"state": "CA", "city": "LYNWOOD", "AvgPop": 61606 }
{"_id": {"state": "KY", "city": "OLDTOWN", "AvgPop": 690 }
{"_id": {"state": "WV", "city": "CAIRO", "AvgPop": 1379 }
{"_id": {"state": "IL", "city": "URBANA", "AvgPop": 46110 }
{"_id": {"state": "IN", "city": "GAS CITY", "AvgPop": 6975 }
Type "it" for more
> it
{"_id": {"state": "KS", "city": "OKETO", "AvgPop": 318 }
{"_id": {"state": "CA", "city": "NOVATO", "AvgPop": 18171 }
{"_id": {"state": "AR", "city": "REYDELL", "AvgPop": 150 }
{"_id": {"state": "IA", "city": "STOCKTON", "AvgPop": 777 }
{"_id": {"state": "NJ", "city": "NEWFOUNDLAND", "AvgPop": 1624 }
{"_id": {"state": "MO", "city": "HARTVILLE", "AvgPop": 3040 }
{"_id": {"state": "HI", "city": "WAIANAE", "AvgPop": 37518 }
{"_id": {"state": "AL", "city": "CORDOVA", "AvgPop": 5273 }
{"_id": {"state": "OH", "city": "KELLEYS ISLAND", "AvgPop": 172 }
{"_id": {"state": "KY", "city": "DAWSON SPRINGS", "AvgPop": 6728 }
{"_id": {"state": "OR", "city": "ZIGZAG", "AvgPop": 1325 }
{"_id": {"state": "MO", "city": "ARBELA", "AvgPop": 564 }
{"_id": {"state": "IL", "city": "BELLFLOWER", "AvgPop": 702 }
{"_id": {"state": "IL", "city": "DAVIS JUNCTION", "AvgPop": 1263 }

```

Assignment:


```
CA Command Prompt - mongo
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()

---
> db
test
> use restaurants
switched to db restaurants
> use restaurant
switched to db restaurant
> show Collections
uncaught exception: Error: don't know how to show [Collections] :
shellHelper.show@src/mongo/shell/utils.js:1209:11
shellHelper@src/mongo/shell/utils.js:836:15
@(shellhelp2):1:1
> db.addresses.find()
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b5e"), "address" : { "building" : "1007", "coord" : [ -73.856077, 40.848447 ],
"street" : "Morris Park Ave", "zipcode" : "10462" }, "borough" : "Bronx", "cuisine" : "Bakery", "grades" : [ { "date" :
ISODate("2014-03-03T00:00:00Z"), "grade" : "A", "score" : 2 }, { "date" : ISODate("2013-09-11T00:00:00Z"), "grade" : "
", "score" : 6 }, { "date" : ISODate("2013-01-24T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2011-1
23T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2011-03-10T00:00:00Z"), "grade" : "B", "score" : 14 }
], "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b5f"), "address" : { "building" : "469", "coord" : [ -73.961704, 40.662942 ],
"street" : "Flatbush Avenue", "zipcode" : "11225" }, "borough" : "Brooklyn", "cuisine" : "Hamburgers", "grades" : [ { "d
te" : ISODate("2014-12-30T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2014-07-01T00:00:00Z"), "grade
" : "B", "score" : 23 }, { "date" : ISODate("2013-04-30T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("
012-05-08T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "Wendy'S", "restaurant_id" : "30112340" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b60"), "address" : { "building" : "351", "coord" : [ -73.98513559999999, 40.767
6919 ], "street" : "West 57 Street", "zipcode" : "10019" }, "borough" : "Manhattan", "cuisine" : "Irish", "grades" : [
```

```
Next Home Featured Profile
CA Command Prompt - mongo
0T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" : ISODate("2011-10-05T00:00:00Z"), "grade" : "A", "score" : 12 }
, "name" : "Palm Restaurant", "restaurant_id" : "40364355" }
Type "it" for more
> db
restaurant
> db.addresses.aggregate([{$project:{'restaurant_id':1,'name':1,'borough':1,'cuisine':1}}])
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b5e"), "borough" : "Bronx", "cuisine" : "Bakery", "name" : "Morris Park Bake S
op", "restaurant_id" : "30075445" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b5f"), "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "Wendy'S", "
restaurant_id" : "30112340" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b60"), "borough" : "Manhattan", "cuisine" : "Irish", "name" : "Dj Reynolds Pub
And Restaurant", "restaurant_id" : "30191841" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b61"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "Riviera Cate
er", "restaurant_id" : "40356018" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b62"), "borough" : "Queens", "cuisine" : "Jewish/Kosher", "name" : "Tov Kosher
Kitchen", "restaurant_id" : "40356068" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b63"), "borough" : "Queens", "cuisine" : "American ", "name" : "Brunos On The B
oulevard", "restaurant_id" : "40356151" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b64"), "borough" : "Staten Island", "cuisine" : "Jewish/Kosher", "name" : "Kosh
er Island", "restaurant_id" : "40356442" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b65"), "borough" : "Brooklyn", "cuisine" : "Delicatessen", "name" : "Wilken'S F
ine Food", "restaurant_id" : "40356483" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b66"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "Regina Catere
rs", "restaurant_id" : "40356649" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b67"), "borough" : "Brooklyn", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "
name" : "Taste The Tropics Ice Cream", "restaurant_id" : "40356731" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b68"), "borough" : "Bronx", "cuisine" : "American ", "name" : "Wild Asia", "res
taurant_id" : "40357217" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b69"), "borough" : "Brooklyn", "cuisine" : "American ", "name" : "C & C Caterin
g Service", "restaurant_id" : "40357437" }
```



```
Command Prompt - mongo
{ "borough" : "Queens", "cuisine" : "Delicatessen", "name" : "Tony'S Deli", "restaurant_id" : "40363333" }
{ "borough" : "Manhattan", "cuisine" : "Sandwiches/Salads/Mixed Buffet", "name" : "Lexler Deli", "restaurant_id" : "40363426" }
{ "borough" : "Staten Island", "cuisine" : "Delicatessen", "name" : "Bagels N Buns", "restaurant_id" : "40363427" }
Type "it" for more
> db
restaurant
> db.addresses.aggregate([{$project: {'restaurant_id':1,'name':1,'borough':1,'zip code':1,'_id':0}}])
uncaught exception: SyntaxError: missing : after property id :
@ (shell):1:53
> db
restaurant
> 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" : "3007545" }
{ "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" : "4036151" }
{ "address" : { "zipcode" : "10314" }, "borough" : "Staten Island", "name" : "Kosher Island", "restaurant_id" : "4035642" }
{ "address" : { "zipcode" : "11234" }, "borough" : "Brooklyn", "name" : "Wilken'S Fine Food", "restaurant_id" : "4035643" }
{ "address" : { "zipcode" : "11219" }, "borough" : "Brooklyn", "name" : "Regina Caterers", "restaurant_id" : "40356649" }
```

```
Command Prompt - mongo
{ "borough" : "Queens", "cuisine" : "Delicatessen", "name" : "Tony'S Deli", "restaurant_id" : "40363333" }
{ "borough" : "Manhattan", "cuisine" : "Sandwiches/Salads/Mixed Buffet", "name" : "Lexler Deli", "restaurant_id" : "40363426" }
{ "borough" : "Staten Island", "cuisine" : "Delicatessen", "name" : "Bagels N Buns", "restaurant_id" : "40363427" }
Type "it" for more
> db
restaurant
> db.addresses.aggregate([{$project: {'restaurant_id':1,'name':1,'borough':1,'zip code':1,'_id':0}}])
uncaught exception: SyntaxError: missing : after property id :
@ (shell):1:53
> db
restaurant
> 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" : "3007545" }
{ "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" : "4036151" }
{ "address" : { "zipcode" : "10314" }, "borough" : "Staten Island", "name" : "Kosher Island", "restaurant_id" : "4035642" }
{ "address" : { "zipcode" : "11234" }, "borough" : "Brooklyn", "name" : "Wilken'S Fine Food", "restaurant_id" : "4035643" }
{ "address" : { "zipcode" : "11219" }, "borough" : "Brooklyn", "name" : "Regina Caterers", "restaurant_id" : "40356649" }
```



```
Next Home Featured Profile
Command Prompt - mongo
{"address": { "zipcode": "10314" }, "borough": "Staten Island", "name": "Bagels N Buns", "restaurant_id": "40363427" }
Type "it" for more
> db
restaurant
> db.addresses.find({"borough":"Bronx"}).limit(5);
{ "_id": ObjectId("6108ed2e627ba6c0712e8b5e"), "address": { "building": "1007", "coord": [ -73.856077, 40.848447 ], "street": "Morris Park Ave", "zipcode": "10462" }, "borough": "Bronx", "cuisine": "Bakery", "grades": [ { "date": "2014-03-03T00:00:00Z", "grade": "A", "score": 2 }, { "date": "2013-09-11T00:00:00Z", "grade": "A", "score": 6 }, { "date": "2013-01-24T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2011-11-23T00:00:00Z", "grade": "A", "score": 9 }, { "date": "2011-03-10T00:00:00Z", "grade": "B", "score": 14 } ], "name": "Morris Park Bake Shop", "restaurant_id": "30075445" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b68"), "address": { "building": "2300", "coord": [ -73.8786113, 40.8502883 ], "street": "Southern Boulevard", "zipcode": "10460" }, "borough": "Bronx", "cuisine": "American", "grades": [ { "date": "2014-05-28T00:00:00Z", "grade": "A", "score": 11 }, { "date": "2013-06-19T00:00:00Z", "grade": "A", "score": 4 }, { "date": "2012-06-15T00:00:00Z", "grade": "A", "score": 3 } ], "name": "Wild Asides", "restaurant_id": "40357217" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b7d"), "address": { "building": "1006", "coord": [ -73.84856870000002, 40.8903781 ], "street": "East 233 Street", "zipcode": "10466" }, "borough": "Bronx", "cuisine": "Ice Cream, Gelato, Yogurt, Ices", "grades": [ { "date": "2014-04-24T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2013-02-21T00:00:00Z", "grade": "A", "score": 13-09-05T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2012-07-03T00:00:00Z", "grade": "A", "score": 11 }, { "date": "2011-07-11T00:00:00Z", "grade": "A", "score": 5 } ], "name": "Carvel Ice Cream", "restaurant_id": "40363093" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b81"), "address": { "building": "1236", "coord": [ -73.8893654, 40.8137617999999 ], "street": "238 Spofford Ave", "zipcode": "10474" }, "borough": "Bronx", "cuisine": "Chinese", "grades": [ { "date": "2013-12-30T00:00:00Z", "grade": "A", "score": 8 }, { "date": "2013-01-08T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2012-06-12T00:00:00Z", "grade": "B", "score": 15 } ], "name": "Happy Garden", "restaurant_id": "40363289" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b93"), "address": { "building": "277", "coord": [ -73.8941893, 40.8634684 ], "street": "East Kingsbridge Road", "zipcode": "10458" }, "borough": "Bronx", "cuisine": "Chinese", "grades": [ { "date": "2013-01-08T00:00:00Z", "grade": "A", "score": 13 } ], "name": "Happy Garden", "restaurant_id": "40364296" }
> db
restaurant
> db.addresses.find("borough":"Bronx")
uncaught exception: SyntaxError: missing ) after argument list :
@:(shell):1:27
> db.addresses.find({"borough":"Bronx"})
{ "_id": ObjectId("6108ed2e627ba6c0712e8b5e"), "address": { "building": "1007", "coord": [ -73.856077, 40.848447 ], "street": "Morris Park Ave", "zipcode": "10462" }, "borough": "Bronx", "cuisine": "Bakery", "grades": [ { "date": "2014-03-03T00:00:00Z", "grade": "A", "score": 2 }, { "date": "2013-09-11T00:00:00Z", "grade": "A", "score": 6 }, { "date": "2013-01-24T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2011-11-23T00:00:00Z", "grade": "A", "score": 9 }, { "date": "2011-03-10T00:00:00Z", "grade": "B", "score": 14 } ], "name": "Morris Park Bake Shop", "restaurant_id": "30075445" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b68"), "address": { "building": "2300", "coord": [ -73.8786113, 40.8502883 ], "street": "Southern Boulevard", "zipcode": "10460" }, "borough": "Bronx", "cuisine": "American", "grades": [ { "date": "2014-05-28T00:00:00Z", "grade": "A", "score": 11 }, { "date": "2013-06-19T00:00:00Z", "grade": "A", "score": 4 }, { "date": "2012-06-15T00:00:00Z", "grade": "A", "score": 3 } ], "name": "Wild Asides", "restaurant_id": "40357217" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b7d"), "address": { "building": "1006", "coord": [ -73.84856870000002, 40.8903781 ], "street": "East 233 Street", "zipcode": "10466" }, "borough": "Bronx", "cuisine": "Ice Cream, Gelato, Yogurt, Ices", "grades": [ { "date": "2014-04-24T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2013-02-21T00:00:00Z", "grade": "A", "score": 13-09-05T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2012-07-03T00:00:00Z", "grade": "A", "score": 11 }, { "date": "2011-07-11T00:00:00Z", "grade": "A", "score": 5 } ], "name": "Carvel Ice Cream", "restaurant_id": "40363093" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b81"), "address": { "building": "1236", "coord": [ -73.8893654, 40.8137617999999 ], "street": "238 Spofford Ave", "zipcode": "10474" }, "borough": "Bronx", "cuisine": "Chinese", "grades": [ { "date": "2013-12-30T00:00:00Z", "grade": "A", "score": 8 }, { "date": "2013-01-08T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2012-06-12T00:00:00Z", "grade": "B", "score": 15 } ], "name": "Happy Garden", "restaurant_id": "40363289" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b93"), "address": { "building": "277", "coord": [ -73.8941893, 40.8634684 ], "street": "East Kingsbridge Road", "zipcode": "10458" }, "borough": "Bronx", "cuisine": "Chinese", "grades": [ { "date": "2013-01-08T00:00:00Z", "grade": "A", "score": 13 } ], "name": "Happy Garden", "restaurant_id": "40364296" }
```

```
Command Prompt - mongo
re": 13 } ], "name": "Happy Garden", "restaurant_id": "40364296" }
> db
restaurant
> db.addresses.find("borough":"Bronx")
uncaught exception: SyntaxError: missing ) after argument list :
@:(shell):1:27
> db.addresses.find({"borough":"Bronx"})
{ "_id": ObjectId("6108ed2e627ba6c0712e8b5e"), "address": { "building": "1007", "coord": [ -73.856077, 40.848447 ], "street": "Morris Park Ave", "zipcode": "10462" }, "borough": "Bronx", "cuisine": "Bakery", "grades": [ { "date": "2014-03-03T00:00:00Z", "grade": "A", "score": 2 }, { "date": "2013-09-11T00:00:00Z", "grade": "A", "score": 6 }, { "date": "2013-01-24T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2011-11-23T00:00:00Z", "grade": "A", "score": 9 }, { "date": "2011-03-10T00:00:00Z", "grade": "B", "score": 14 } ], "name": "Morris Park Bake Shop", "restaurant_id": "30075445" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b68"), "address": { "building": "2300", "coord": [ -73.8786113, 40.8502883 ], "street": "Southern Boulevard", "zipcode": "10460" }, "borough": "Bronx", "cuisine": "American", "grades": [ { "date": "2014-05-28T00:00:00Z", "grade": "A", "score": 11 }, { "date": "2013-06-19T00:00:00Z", "grade": "A", "score": 4 }, { "date": "2012-06-15T00:00:00Z", "grade": "A", "score": 3 } ], "name": "Wild Asides", "restaurant_id": "40357217" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b7d"), "address": { "building": "1006", "coord": [ -73.84856870000002, 40.8903781 ], "street": "East 233 Street", "zipcode": "10466" }, "borough": "Bronx", "cuisine": "Ice Cream, Gelato, Yogurt, Ices", "grades": [ { "date": "2014-04-24T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2013-02-21T00:00:00Z", "grade": "A", "score": 13-09-05T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2012-07-03T00:00:00Z", "grade": "A", "score": 11 }, { "date": "2011-07-11T00:00:00Z", "grade": "A", "score": 5 } ], "name": "Carvel Ice Cream", "restaurant_id": "40363093" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b81"), "address": { "building": "1236", "coord": [ -73.8893654, 40.8137617999999 ], "street": "238 Spofford Ave", "zipcode": "10474" }, "borough": "Bronx", "cuisine": "Chinese", "grades": [ { "date": "2013-12-30T00:00:00Z", "grade": "A", "score": 8 }, { "date": "2013-01-08T00:00:00Z", "grade": "A", "score": 10 }, { "date": "2012-06-12T00:00:00Z", "grade": "B", "score": 15 } ], "name": "Happy Garden", "restaurant_id": "40363289" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8b93"), "address": { "building": "277", "coord": [ -73.8941893, 40.8634684 ], "street": "East Kingsbridge Road", "zipcode": "10458" }, "borough": "Bronx", "cuisine": "Chinese", "grades": [ { "date": "2013-01-08T00:00:00Z", "grade": "A", "score": 13 } ], "name": "Happy Garden", "restaurant_id": "40364296" }
```



```
Command Prompt - mongo
db
restaurant
db.addresses.find({"borough":"Bronx"}).skip(5).limit(5)
{ "_id" : ObjectId("6108ed2e627ba6c0712e8b9b"), "address" : { "building" : "658", "coord" : [ -73.81363999999999, 40.82911000000001 ], "street" : "Clarence Ave", "zipcode" : "10465", "borough" : "Bronx", "cuisine" : "American", "grades" : [ { "date" : ISODate("2014-06-21T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2012-07-11T00:00:00Z"), "grade" : "A", "score" : 10 } ], "name" : "Manhem Club", "restaurant_id" : "40364363" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8bb3"), "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("2013-03-14T00:00:00Z"), "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("6108ed2e627ba6c0712e8bcb"), "address" : { "building" : "72", "coord" : [ -73.92506, 40.8275556 ], "street" : "East 161 Street", "zipcode" : "10451", "borough" : "Bronx", "cuisine" : "American", "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("2012-01-09T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2011-08-15T00:00:00Z"), "grade" : "C", "score" : 37 } ], "name" : "Yankee Tavern", "restaurant_id" : "40365499" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8bde"), "address" : { "building" : "331", "coord" : [ -73.87786539999999, 40.8724377 ], "street" : "East 204 Street", "zipcode" : "10467", "borough" : "Bronx", "cuisine" : "Irish", "grades" : [ { "date" : ISODate("2014-08-22T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2014-03-26T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2013-09-11T00:00:00Z"), "grade" : "A", "score" : 23 }, { "date" : ISODate("2013-09-11T00:00:00Z"), "grade" : "B", "score" : 27 }, { "date" : ISODate("2011-10-20T00:00:00Z"), "grade" : "A", "score" : 13 } ], "name" : "McDwyers Pub", "restaurant_id" : "40365893" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8bf7"), "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" : "A", "score" : 9 }, { "date" : ISODate("2012-11-20T00:00:00Z"), "grade" : "A", "score" : 19 }, { "date" : ISODate("2013-05-15T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2012-11-20T00:00:00Z"), "grade" : "A", "score" : 19 } ], "name" : "McDwyers Pub", "restaurant_id" : "40365893" }
```

```
Next Home Featured Profile
Command Prompt - mongo
> db.addresses.find({"grades":{"elemMatch":{"score":{"$gt:90}}}})
{ "_id" : ObjectId("6108ed2e627ba6c0712e8bcb"), "address" : { "building" : "65", "coord" : [ -73.9712725, 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/Randolph's", "restaurant_id" : "40372466" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8d5d"), "address" : { "building" : "345", "coord" : [ -73.9864626, 40.7266739 ], "street" : "East 6 Street", "zipcode" : "10003", "borough" : "Manhattan", "cuisine" : "Indian", "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("2012-04-06T00:00:00Z"), "grade" : "C", "score" : 92 }, { "date" : ISODate("2011-11-03T00:00:00Z"), "grade" : "C", "score" : 41 } ], "name" : "Gandhi", "restaurant_id" : "40381295" }
{ "_id" : ObjectId("6108ed2e627ba6c0712e8ec0"), "address" : { "building" : "130", "coord" : [ -73.984758, 40.7457939 ], "street" : "Madison Avenue", "zipcode" : "10016", "borough" : "Manhattan", "cuisine" : "Pizza/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 Napoli", "restaurant_id" : "40393488" }
> db.addresses.find({"grades":{"elemMatch":{"score":{"$gt:80,$lt:100}}}})
{ "_id" : ObjectId("6108ed2e627ba6c0712e8d5d"), "address" : { "building" : "345", "coord" : [ -73.9864626, 40.7266739 ], "street" : "East 6 Street", "zipcode" : "10003", "borough" : "Manhattan", "cuisine" : "Indian", "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("2012-04-06T00:00:00Z"), "grade" : "C", "score" : 92 }, { "date" : ISODate("2011-11-03T00:00:00Z"), "grade" : "C", "score" : 41 } ], "name" : "Gandhi", "restaurant_id" : "40381295" }
```


Command Prompt - mongo

```
({"2011-07-27T00:00:00Z", "grade": "A", "score": 2 } ], "name": "Sports Center At Chelsea Piers (Sushi Bar)", "restaurant_id": "40882356" }
> db.addresses.find({"cuisine":{"ne:"American"},"grades.score":{"gt:70},"address.coord":{"lt:-65.754168}})
{ "_id": ObjectId("6108ed2e627ba6c0712e8cbc"), "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("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("6108ed2e627ba6c0712e8d5d"), "address": { "building": "345", "coord": [ -73.9864626, 40.7266739 ],
"street": "East 6 Street", "zipcode": "10003", "borough": "Manhattan", "cuisine": "Indian", "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("2012-04-06T00:00:00Z"), "grade": "C", "score": 92 }, { "date": ISODate("2011-11-03T00:00:00Z"), "grade": "C", "score": 41 } ], "name": "Gandhi", "restaurant_id": "40381295" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8deb"), "address": { "building": "14", "coord": [ -73.9883900, 40.748735 ],
"street": "East 23 Street", "zipcode": "10010", "borough": "Manhattan", "cuisine": "American", "grades": [ { "date": ISODate("2014-10-28T00:00:00Z"), "grade": "A", "score": 9 }, { "date": ISODate("2014-04-10T00:00:00Z"), "grade": "A", "score": 12 }, { "date": ISODate("2013-08-08T00:00:00Z"), "grade": "C", "score": 58 }, { "date": ISODate("2013-02-14T00:00:00Z"), "grade": "A", "score": 13 }, { "date": ISODate("2012-06-15T00:00:00Z"), "grade": "C", "score": 71 } ], "name": "Live Bait Bar & Restaurant", "restaurant_id": "40387237" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8ec0"), "address": { "building": "130", "coord": [ -73.984758, 40.7457939 ],
"street": "Madison Avenue", "zipcode": "10016", "borough": "Manhattan", "cuisine": "Pizza/Italian", "grades": [ { "date": ISODate("2014-12-24T00:00:00Z"), "grade": "A", "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 Napoli", "restaurant_id": "40393488" }
{ "_id": ObjectId("6108ed2e627ba6c0712e8ecb"), "address": { "building": "101", "coord": [ -73.9243061, 40.8276297 ],
```