

Assignment #04

Submitted by: Basharat Hussain

Roll no: P17-6102

Subject: Database Systems

Submitted to: Ms. Safia Fatima

MongoDB CURD Operations

```
dark-vamp@darkvamp: ~/Desktop/Database Systems/Dataset
dark-vamp@darkvamp:~/Desktop/Database Systems/Dataset$ sudo mongoimport --db good_db --collection restaurants --file assignment.json
[sudo] password for dark-vamp:
2021-07-03T00:45:40.760+0500    connected to: mongodb://localhost/
2021-07-03T00:45:42.098+0500    25359 document(s) imported successfully. 0 document(s) failed to import.
dark-vamp@darkvamp:~/Desktop/Database Systems/Dataset$
```

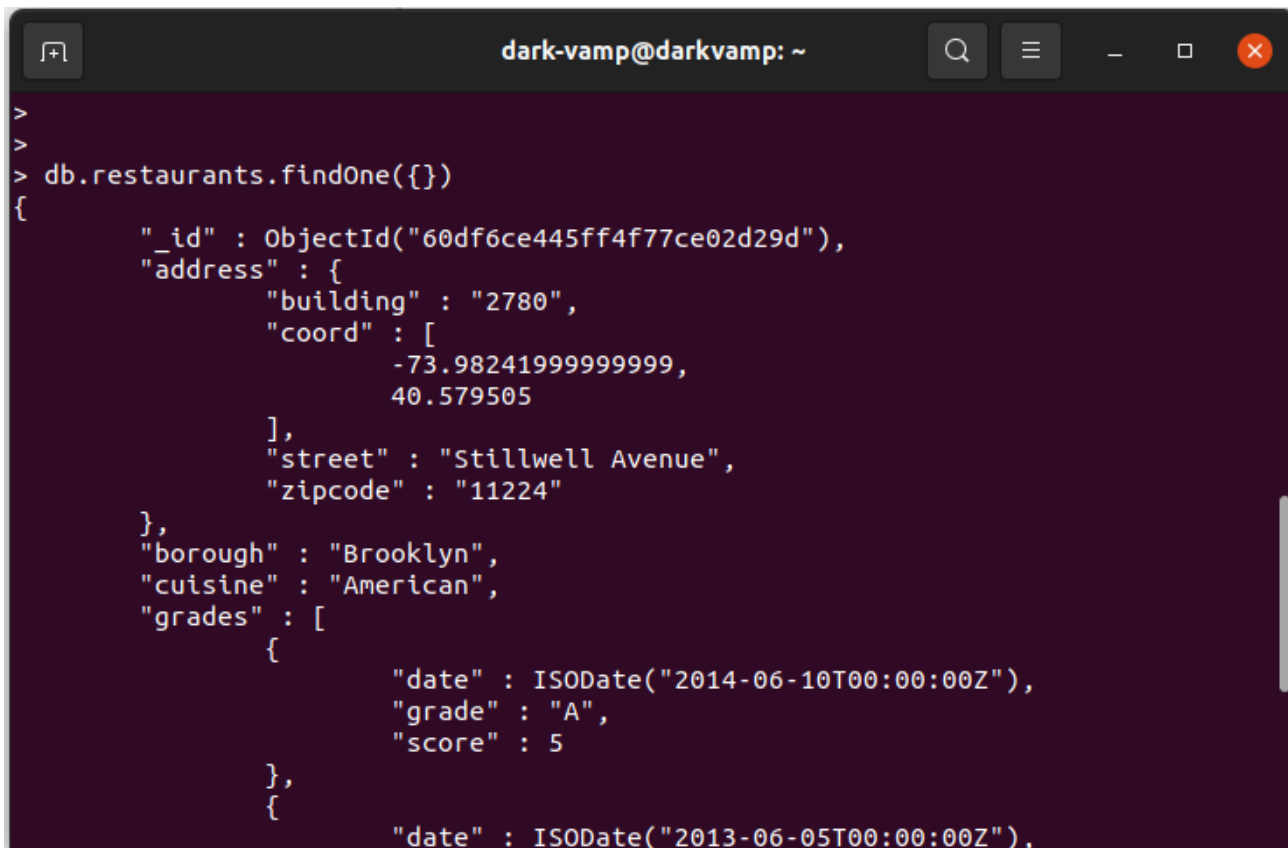
```
dark-vamp@darkvamp: ~
URL accessible to you
    and anyone you share the URL with. MongoDB may use this information to make product
    improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
>
> show dbs
admin      0.000GB
config     0.000GB
good_db    0.004GB
local      0.000GB
> use good_db
switched to db good_db
> show collections
restaurants
> db.restaurants.count()
25359
>
```

1.Read Operation

1 . Retrieve the first document of the collection restaurants only.

> db.restaurants.findOne({})

A terminal window with a dark purple background. The title bar shows 'dark-vamp@darkvamp: ~'. The command prompt is '>'. The user enters 'db.restaurants.findOne({})'. The output is a JSON document representing a restaurant in Brooklyn.

```
>
>
> db.restaurants.findOne({})
{
  "_id" : ObjectId("60df6ce445ff4f77ce02d29d"),
  "address" : {
    "building" : "2780",
    "coord" : [
      -73.98241999999999,
      40.579505
    ],
    "street" : "Stillwell Avenue",
    "zipcode" : "11224"
  },
  "borough" : "Brooklyn",
  "cuisine" : "American",
  "grades" : [
    {
      "date" : ISODate("2014-06-10T00:00:00Z"),
      "grade" : "A",
      "score" : 5
    },
    {
      "date" : ISODate("2013-06-05T00:00:00Z"),
```

2 . Retrieve all those restaurants named "Wendy'S", serving the cuisine of "Hamburgers" and in the borough of "Bronx".

> db.restaurants.find({ \$and: [{ name: "Wendy'S" , cuisine: "Hamburgers" ,borough: "Bronx" }] }).pretty()

```
dark-vamp@darkvamp: ~  
>  
>  
>  
> db.restaurants.find( { $and: [ { name: "Wendy'S" , cuisine: "Hamburgers" ,borough: "Bronx" } ] } ).pretty()  
{  
  "_id" : ObjectId("60df6ce445ff4f77ce02d507"),  
  "address" : {  
    "building" : "3636",  
    "coord" : [  
      -73.8421735,  
      40.8794229  
    ],  
    "street" : "Boston Road",  
    "zipcode" : "10469"  
  },  
  "borough" : "Bronx",  
  "cuisine" : "Hamburgers",  
  "grades" : [  
    {  
      "date" : ISODate("2014-11-17T00:00:00Z"),  
      "grade" : "A",  
      "score" : 9  
    },  
  ],  
}
```

3 . Retrieve all those restaurants having a grade of 'A', score of '2', on the date of"2014-05-08T00:00:00Z".

```
> db.restaurants.find({"grades.date":ISODate("2014-05-08T00:00:00Z"),  
"grades.grade": "A","grades.score":2 } ).pretty()
```

```
dark-vamp@darkvamp: ~  
>  
>  
>  
> db.restaurants.find({"grades.date":ISODate("2014-05-08T00:00:00Z"), "grades.gr  
ade": "A","grades.score":2 } ).pretty()  
{  
  "_id" : ObjectId("60df6ce445ff4f77ce02d55f"),  
  "address" : {  
    "building" : "969",  
    "coord" : [  
      -73.962756,  
      40.769241  
    ],  
    "street" : "Lexington Avenue",  
    "zipcode" : "10021"  
  },  
  "borough" : "Manhattan",  
  "cuisine" : "Italian",  
  "grades" : [  
    {  
      "date" : ISODate("2014-05-08T00:00:00Z"),  
      "grade" : "A",  
      "score" : 11  
    },  
    {  
      "date" : ISODate("2014-05-08T00:00:00Z"),  
      "grade" : "A",  
      "score" : 11  
    }  
  ]  
}
```

2.Update Operations

1 . Update the name of restaurant to “FAST” who’s restaurant ID is “50008124”.

```
> db.restaurants.updateOne({"restaurant_id": "50008124"},{$set:  
{"name": "FAST"} })
```

```
> db.restaurants.find({"restaurant_id": "50008124"}).pretty()
```

```
dark-vamp@darkvamp: ~  
> db.restaurants.updateOne({"restaurant_id": "50008124"},{  
... .. $set:{"name": "FAST"} })  
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }  
>  
> db.restaurants.find({"restaurant_id": "50008124"})  
{ "_id" : ObjectId("60df6ce545ff4f77ce03280a"), "address" : { "building" : "3179", "coord" : [ -73.877865, 40.680265 ], "street" : "Atlantic Avenue", "zipcode" : "11208" }, "borough" : "Brooklyn", "cuisine" : "Hamburgers", "grades" : [ { "date" : ISODate("2014-11-18T00:00:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2014-05-08T00:00:00Z"), "grade" : "A", "score" : 2 } ], "name" : "FAST", "restaurant_id" : "50008124" }  
>  
> db.restaurants.find({"restaurant_id": "50008124"}).pretty()  
uncaught exception: TypeError: db.restaurants.find(...).pretty is not a function :  
@(shell):1:1  
>  
>  
> db.restaurants.find({"restaurant_id": "50008124"}).pretty()  
{  
  "_id" : ObjectId("60df6ce545ff4f77ce03280a"),  
  "address" : {  
    "building" : "3179",  
    "coord" : [  
      -73.877865,  
      40.680265  
    ],  
    "street" : "Atlantic Avenue",  
    "zipcode" : "11208"  
  },  
  "borough" : "Brooklyn",  
  "cuisine" : "Hamburgers",  
  "grades" : [  
    {  
      "date" : ISODate("2014-11-18T00:00:00Z"),  
      "grade" : "A",  
      "score" : 7  
    },  
    {  
      "date" : ISODate("2014-05-08T00:00:00Z"),  
      "grade" : "A",  
      "score" : 2  
    }  
  ],  
  "name" : "FAST",  
  "restaurant_id" : "50008124"  
}
```

2 . Find out all the restaurants named “Wendy'S Old Fashioned Hamburgers” and update their cuisine to “Desi”.

```
> db.restaurants.update({"name": "Wendy'S Old Fashioned Hamburgers"},{$set:{"cuisine": "DESI log"}})
```

```
> db.restaurants.find({"cuisine": "DESI log"}).pretty()
```

```
dark-vamp@darkvamp: ~  
}>  
> db.restaurants.update({"name": "Wendy'S Old Fashioned Hamburgers"},{$set:{"cuisine": "DESI log"}})  
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 0 })  
>  
>  
> db.restaurants.find({"cuisine": "DESI log"}).pretty()  
{  
  "_id" : ObjectId("60df6ce545ff4f77ce03280e"),  
  "address" : {  
    "building" : "3939",  
    "coord" : [  
      -73.9415434,  
      40.8391424  
    ],  
    "street" : "Broadway",  
    "zipcode" : "10032"  
  },  
  "borough" : "Manhattan",  
  "cuisine" : "DESI log",  
  "grades" : [  
    f
```

3 . There are many “Mcdonald’s” in the borough of “Staten Island”. Update all cuisines of Mcdonald’s to “Pizzas”.

```
> db.restaurants.update({"borough": "Staten Island",  
"cuisine": "Mcdonald's"},{$set:{"cuisine": "Pizzas"}} )
```


3 . Find all Burger King restaurants, serving the cuisine of Hamburgers in the borough of Queens and delete all those entries only.

```
> db.restaurants.find( {"name": "Burger King", "cuisine":  
"Hamburgers", "borough": "Queens"}).pretty().count()  
20
```

```
> db.restaurants.remove( {"name": "Burger King", "cuisine":  
"Hamburgers", "borough":"Queens"})  
WriteResult({ "nRemoved" : 20 })
```

```
dark-vamp@darkvamp: ~  
> db.restaurants.find( {"name": "Burger King", "cuisine": "Hamburgers", "borough": "Queens"}).pretty().count()  
20  
> db.restaurants.remove( {"name": "Burger King", "cuisine": "Hamburgers", "borough": "Queens"})  
WriteResult({ "nRemoved" : 20 })
```


4.Creating your own document

```
> db.restaurants.insert({"address":{"building": "108-88","coord" :
[ 73.86210299999911,40.749987], "street": "silicon valley",
"zipcode" : "55000"},"borough" : "Queens","cuisine" : "puppooooo",
{"date" : ISODate("2021-08-19T00:00:00Z"),"grade" :[{ "F", "score" : 11
}], "name" : "Libs kitchen","restaurant_id" : "6102"})
```

```
dark-vamp@darkvamp: ~  
: [{ "F", "score" : 11 }], "name" : "Libs kitchen", "restaurant_id" : "6102" }  
...  
...  
> db.restaurants.insert({"name": "Libs kitchen"})  
WriteResult({ "nInserted" : 1 })  
> db.restaurants.find({"name": "Libs kitchen"})  
{ "_id" : ObjectId("60df79722a355ab4582b17fd"), "name" : "Libs kitchen" }  
> db.restaurants.find({"address": {"building": "108-88", "coord" : [ 73.8621029999  
9911, 40.749987], "street": "silicon valley", "zipcode" : "55000"}, "borough" : "Qu  
eens", "cuisine" : "puppoooo", {"date" : ISODate("2021-08-19T00:00:00Z"), "grade" :  
[ { "F", "score" : 11 } ], "name" : "Libs kitchen", "restaurant_id" : "6102" })).prett  
y()  
...  
...
```