

Introduction to Database Systems: CS312

Compass, MongoDB's DB Interface

Oliver Bonham-Carter

1 April 2019

Getting started with Mongo

Setup a data directory (if you have not already done so)

```
mkdir ~/mongodbData
```

Start the Mongo server with data directory as a parameter

Note: Control-C to exit.

```
mongod --dbpath ~/mongodbData/
```

With new another terminal, start the Mongo client

```
mongo
```

Find databases or collections, from Mongo's client

```
show dbs
```

```
show collections
```

Begin a new database, from Mongo's client

```
use myDB
```

How Do I Compare Values?

Operation	MongoDB	RDBMS
Equality	<code>db.employees.find({"salary":"5000"})</code>	<code>where 'salary' = '5000'</code>
Less Than	<code>db.employees.find({"age":{"\$lt":30}})</code>	<code>where age < 30</code>
Less Than Equals	<code>db.employees.find({"age":{"\$lte":30}})</code>	<code>where age <= 30</code>
Greater Than	<code>db.employees.find({"age":{"\$gt":30}})</code>	<code>where age > 30</code>
Greater Than Equals	<code>db.employees.find({"age":{"\$gte":30}})</code>	<code>where age >= 30</code>
Not Equals	<code>db.employees.find({"age":{"\$ne":30}})</code>	<code>where age != 30</code>

Details where age less than 30

```
db.employee.find({"age":{"$lt":30}},{"email":1}).pretty()
```

```
> db.employee.find({"age":{"$lt":30}},{"email":1}).pretty()
{
  "_id" : "5c89c73ec4863a49668b3784",
  "email" : "belinda.guzman@beadzza.org"
}
{ "_id" : "5c89c73e5e7b4f23f0a32660", "email" : "cora.keith@combogen.io" }
```

How Do I Compare Values?

Details where age greater than 30

```
db.employee.find({"age":{"$gt":30}},{"email":1}).pretty()
```

```
> db.employee.find({"age":{"$gt":30}},{"email":1}).pretty()
{
  "_id" : "5c89c73ea91e2b221611b950",
  "email" : "pennington.moss@radiantix.tv"
}
{
  "_id" : "5c89c73ea080dbf02bd40f0e",
  "email" : "rosemary.french@prosure.co.uk"
}
```

Details where age not equal to 30

```
db.employee.find({"age":{"$gt":30}},{"email":1}).pretty()
```

```
> db.employee.find({"age":{"$ne":27}},{"email":1}).pretty()
{
  "_id" : "5c89c73ea91e2b221611b950",
  "email" : "pennington.moss@radiantix.tv"
}
{
  "_id" : "5c89c73ea080dbf02bd40f0e",
  "email" : "rosemary.french@prosure.co.uk"
}
{ "_id" : "5c89c73e5e7b4f23f0a32660", "email" : "cora.keith@combogen.io" }
```

Cool! Let's See That in Action!



BTW: This graphic found as a result of image searching, "The Coolest Thing." Apologies.

Getting started with Compass

Setup a data directory (if you have not already done so)

```
mkdir ~/mongodbData
```

Start the Mongo server with data directory as a parameter

Note: Control-C to exit.

```
mongod --dbpath ~/mongodbData/
```

Launch the Compass client

Look-up the icon for *Compass* on your Ubuntu machine.
Launch this icon.

Find databases or collections, from Mongo's client

```
show dbs  
show collections
```

Begin a new database, from Mongo's client

```
use myDB
```

Compass Main screen

Connect to the server

Connect to Host

Hostname

Port

SRV Record ☐

Authentication

Replica Set Name

Read Preference

SSL

SSH Tunnel

Favorite Name ⓘ

CREATE FAVORITE

CONNECT

Compass Databases Menu

What DBs (database groups) are on my server?

MongoDB Compass Community - localhost27017

Connect View Help

localhost27017 STANDALONE MongoDB 3.6.3 Community

My Cluster

4 DBS 3 COLLECTIONS

filter

- admin
- config
- local
- myDB

Databases

CREATE DATABASE

Database Name	Storage Size	Collections	Indexes
admin	16.0KB	0	1
config	4.0KB	0	2
local	16.0KB	1	1
myDB	32.0KB	2	2

Click *myDB*

Note: We have already established this DB during another class but, you could create a new DB by typing, use `myOtherDB`.

Compass Collections Menu

What Collections are on my server?

My Cluster localhost:27017 STANDALONE MongoDB 3.6.3 Community

4 DBS 3 COLLECTIONS

filter

- > admin
- > config
- > local
- ▼ myDB
 - employee
 - people

CREATE COLLECTION

Collection Name ^	Documents	Avg. Document Size	Total Document Size	Num. Indexes	Total Index Size
employee	4	1.2 KB	4.7 KB	1	16.0 KB
people	5	59.0 B	295.0 B	1	16.0 KB

Click *employee*

Note: We have already established this collection during another class.

Schema

How do I see how my data is organized?

myDB.employee

DOCUMENTS 4 TOTAL SIZE 4.8KB AVG. SIZE 1.2KB INDEXES 1 TOTAL SIZE 16.0KB AVG. SIZE 16.0KB

Documents Aggregations **Schema** Explain Plan Indexes Validation

FILTER {} **OPTIONS** **ANALYZE** **RESET** ...

PROJECT {eyeColor:1,company:1,"name.first":1,"name.last":1,_id:0,frinds:1} **LIMIT** 0

The schema content is outdated and no longer in sync with the documents view. Press "Analyze" again to see the schema for the current query.

Field	Type	Value
_id	string	5c89c73ea088dbf62bd40f0e
		5c89c73e5e7b4f23f0a32660
		5c89c73ec4863a49668b3784
		5c89c73ea91e2b221611b950
about	string	Quis do laborum non aliquip laboris pariatur ipsum eiusmod. Veniam sint deserunt Lorem ut veniam excepteur Lorem velit. Cupidatat proident aliqua laborum ullamco.
		Qui aliqua magna eiusmod fugiat irure dolor tempor magna amet tempor labore sint. Incidunt qui fugiat proident laborum adipisicing culpa labore mollit

We get a look at how MongoDB organizes its data in this tab.
Good for debugging.

Query!!

How do I run a query?

myDB.employee

Documents Aggregations Schema Explain Plan

FILTER {}

PROJECT {eyeColor:1,company:1,"name.first":1,"name.last":1}

SORT

VIEW **LIST** **TABLE**

```
_id: "5c89c73ea91e2b221611b950"  
eyeColor: "brown"  
▶ name: Object  
company: "RADIANTIX"
```

```
_id: "5c89c73ec4863a49668b3784"  
eyeColor: "green"  
▶ name: Object  
company: "BEADZZA"
```

We have already established this collection from the file `syntheticData_small.json` during another class.

Query!!

Awesome! How do I run another query?

myDB.employee

Documents

Aggregations

Schema

Expla

FILTER {}

PROJECT {eyeColor:1,company:1,"name.first":1,"name.last":1,_id:0,friends:1}

SORT

VIEW

LIST

TABLE

```
eyeColor: "brown"
  ▶ name: Object
    company: "RADIANTIX"
  ▶ friends: Array
```

```
eyeColor: "green"
  ▶ name: Object
    company: "BEADZZA"
  ▼ friends: Array
    0: Object
      id: 0
      name: "Bryan Frederick"
    1: Object
      id: 1
      name: "Navarro Rutledge"
    2: Object
      id: 2
      name: "Jacklyn Eaton"
```

Query!!

Cool! How do I run yet another query?

myDB.employee

Documents

Explain Plan

Indexes

FILTER {}

OPTIONS

FIND

RESET



PROJECT

{ "email":1, "balance":1, "_id":0 }

SORT

{ field: -1 }

SKIP 0

LIMIT 0

VIEW

LIST

TABLE

Displaying documents 1 - 4 of 4



balance: "\$3,294.70"
email: "pennington.moss@radiantix.tv"

balance: "\$1,949.33"
email: "belinda.guzman@beadzza.org"

balance: "\$2,532.27"
email: "rosemary.french@prosure.co.uk"

balance: "\$2,359.76"
email: "cora.keith@combogen.io"

Query As a Table!!

Can I display my query as a table?

myDB.employee

Documents

Explain Plan

Indexes

FILTER {}

▼ OPTIONS

FIND

RESET



PROJECT {"email":1,"balance":1,"_id":0}

SORT { field: -1 }

SKIP 0

LIMIT 0

VIEW

LIST

TABLE

Displaying documents 1 - 4 of 4



🏠 employee

	balance String	email String
1	"\$3,294.70"	"pennington.moss@radiantix.tv"
2	"\$1,949.33"	"belinda.guzman@beadzza.org"
3	"\$2,532.27"	"rosemary.french@prosure.co.uk"
4	"\$2,359.76"	"cora.keith@combogen.io"



myDB.employee

Indexes

©

LIMIT 0

"pennington.moss@radiantix.tv"

Consider this...

THINK

- Can you create and populate a completely new MongoDB database?
- Can you write sophisticated queries in your database to isolate meaningful information from the data?