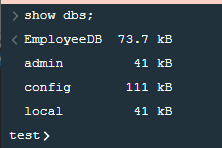
MongoDB commands for database

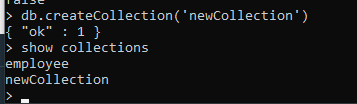
1. show dbs – show all databases.



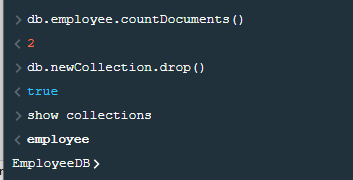
1. use EmployeeDB – To switch to new database or create one.
2. db – List out the current db name.
3. db.dropDatabase() – To delete or drop the current database.

MongoDB commands for Collections

1. show collections – to list out the collections in the current database.
2. db.createCollection(‘newCollection’) – Creating a new collection inside a database.

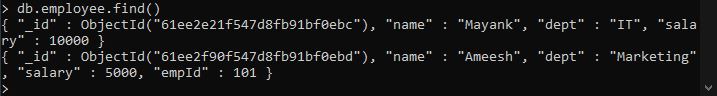


1. db.newCollection.drop() – To drop a collection

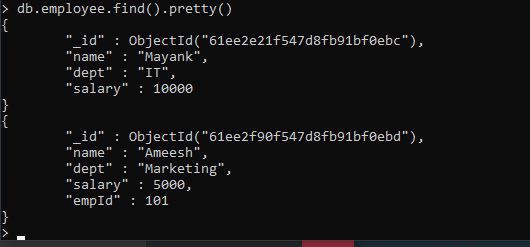


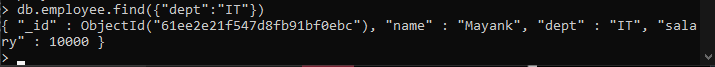
MongoDB commands for documents

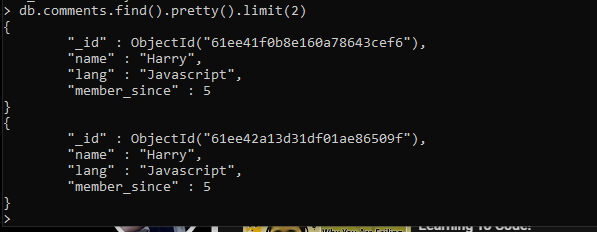
1. db.<Collection Name>.insert({'name':'Harry', 'lang':'Javascript','member\_since':5}) – Insert document in a collection
2. db.comments.insertMany([{'name':'Harry', 'lang':'Javascript','member\_since':5},{'name':'Mayank', 'lang':'C++','member\_since':3},{'name':'Swapnil', 'lang':'Python','member\_since':1}]) – Insert many documents at once.
3. db.<Collection name>.find() – Display all documents inside the collection.



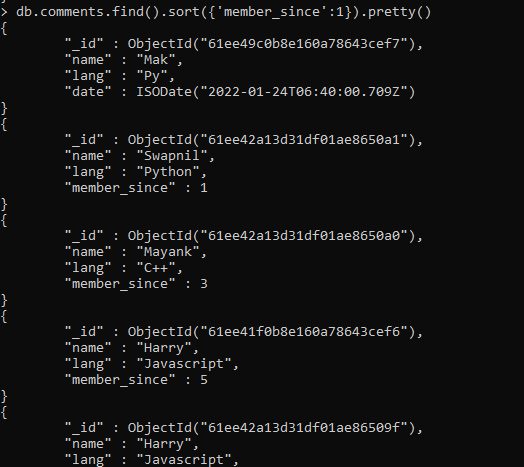
1. db.<Collection Name>.find().pretty() – Gives the output in a prettified manner



1. db.<Collection Name>.find({“dept”:”IT”}) – To find the document with the given JSON. 
2. db.employee.find().count() – Count the number of documents inside the collection.
3. db.<collection\_name>.findOne({key: 'value'}) - Find the first row matching the object
4. db.comments.find().pretty().limit(2) – Limit the output rows



1. db.comments.find().limit(2).count() – It will count the total number of rows in the output without being limited. So the answer will not be 2 but whatever is the count of the find function. So limit function only limits the output to be displayed.
2. Sort according to a field

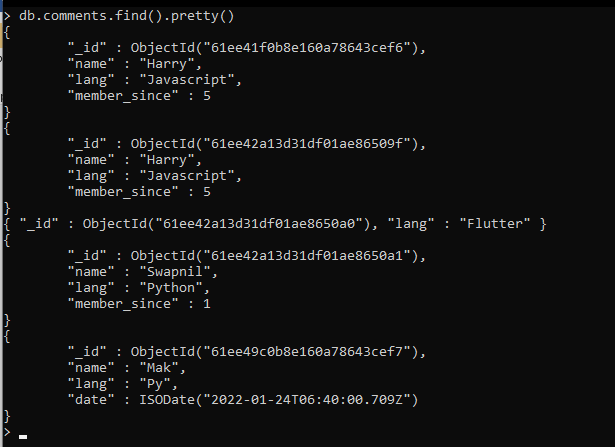


For descending order, give -1 instead of 1.

1. db.comments.findOne({‘name’:’Harry’}) – Find the first row matching the object
2. db.comments.find({'member\_since':{$gt:3}}).pretty() – find documents having field value greater than 3.

$gt – greater than, $gte – greater than equal to, $lt – less than, $lte – less than equal to

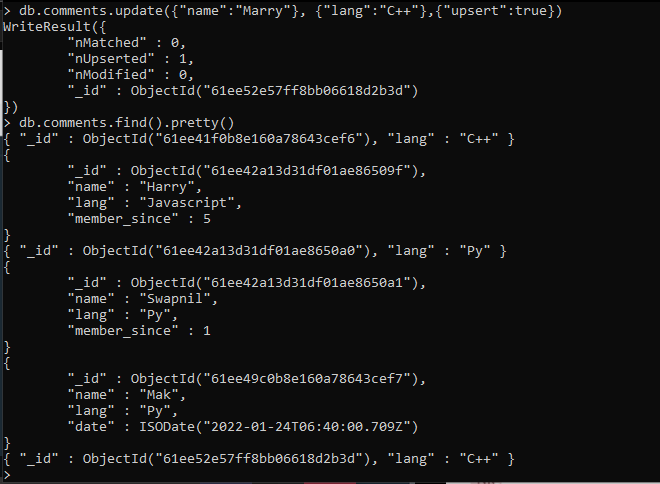
1. db.comments.update({"name":"Mayank"},{"lang":"Flutter"}) – To update the document. This will overwrite all the fields inside the matching document with the second argument. By default it only updates the first matching document.



You can see the document of Mayank has been overwritten and it contains only the json that we provided.

1. db.comments.update({"name":"Swapnil"}, {$set:{"lang":"Py"}}) – In the document where name is swapnil, the lang will be updated to Py.

22. db.comments.update({"name":"Marry"}, {"lang":"C++"},{"upsert":true}) – Now “Marry” is not there so a new document will be added as upsert is set to true.



Notes: https://www.codewithharry.com/blogpost/mongodb-cheatsheet

Sir Notes

#Mango Db Commands Rows

#show all the row in collections  
>db.<collection\_name>.find()

#insert single row

>db.<collection\_name>.insert({});

db.employee.insert(  
{  
"name": "ramesh",  
"dept":"Account",  
"salary":5000  
});

#insert multiple rows  
>db.<collection\_name>.insertMany([{},{},{}])

db.employee.insertMany([  
{  
"name": "c",  
"dept":"c",  
"salary":2000  
},  
{  
"name": "python",  
"dept":"python",  
"salary":1000  
}  
]);

#Show all Rows in a Collection (Prettified)

>db.<collection\_name>.find().pretty()

#Find the first row matching the object

db.<collection\_name>.findOne({key: 'value'})

#Search in a MongoDb Database

db.<collection\_name>.find({key:'value'})

#Limit the number of rows in output  
>db.comments.find().limit(2)

#Count the number of rows in the output

>db.comments.find().count()