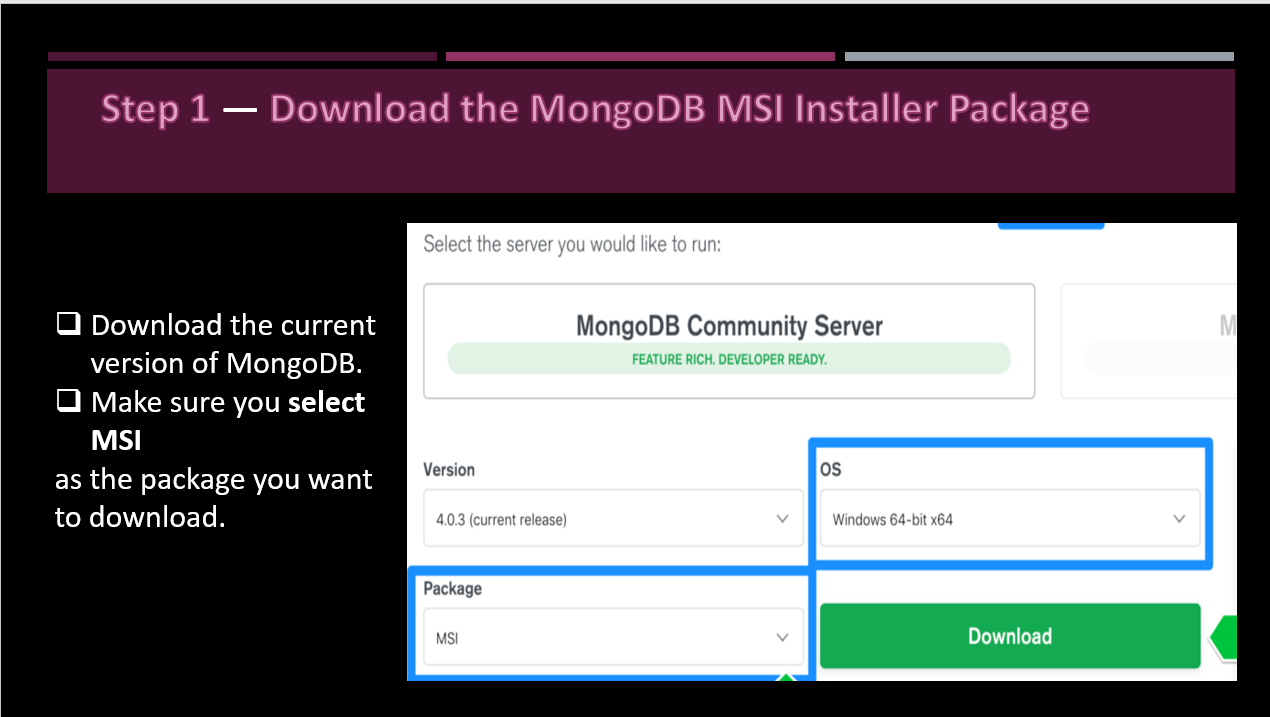
BIBD MINI PROJECT

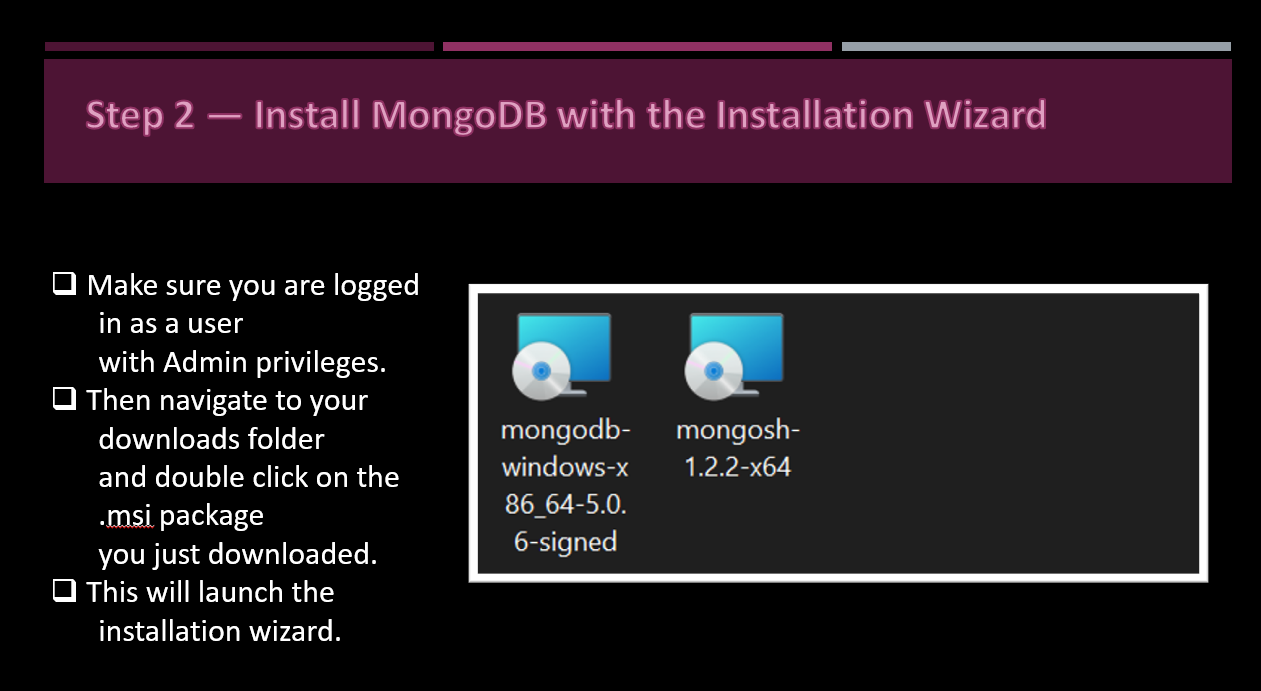
MongoDB

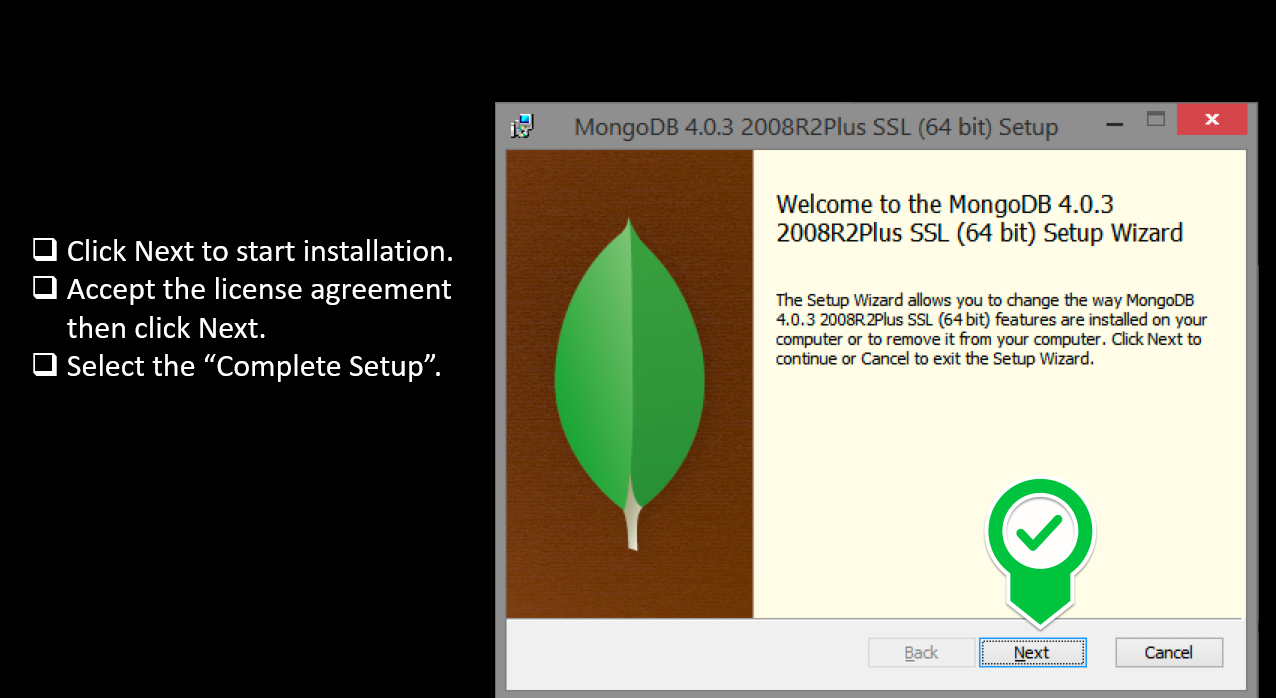
**What is MongoDB?**

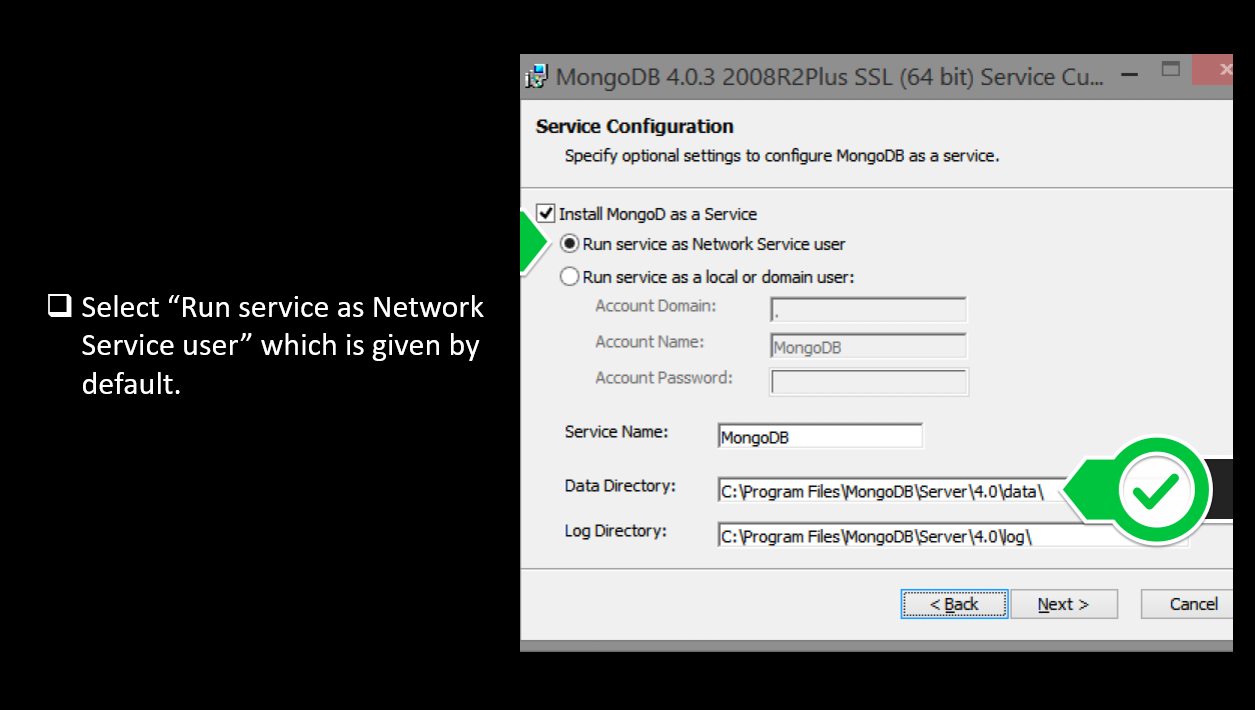
MongoDB is a **cross-platform document-oriented database program**. Classified as a NoSQL database program, MongoDB uses JSON -like documents with schema. MongoDB is developed by MongoDB Inc. and licensed under the Server-Side Public License.

Installation of MongoDB

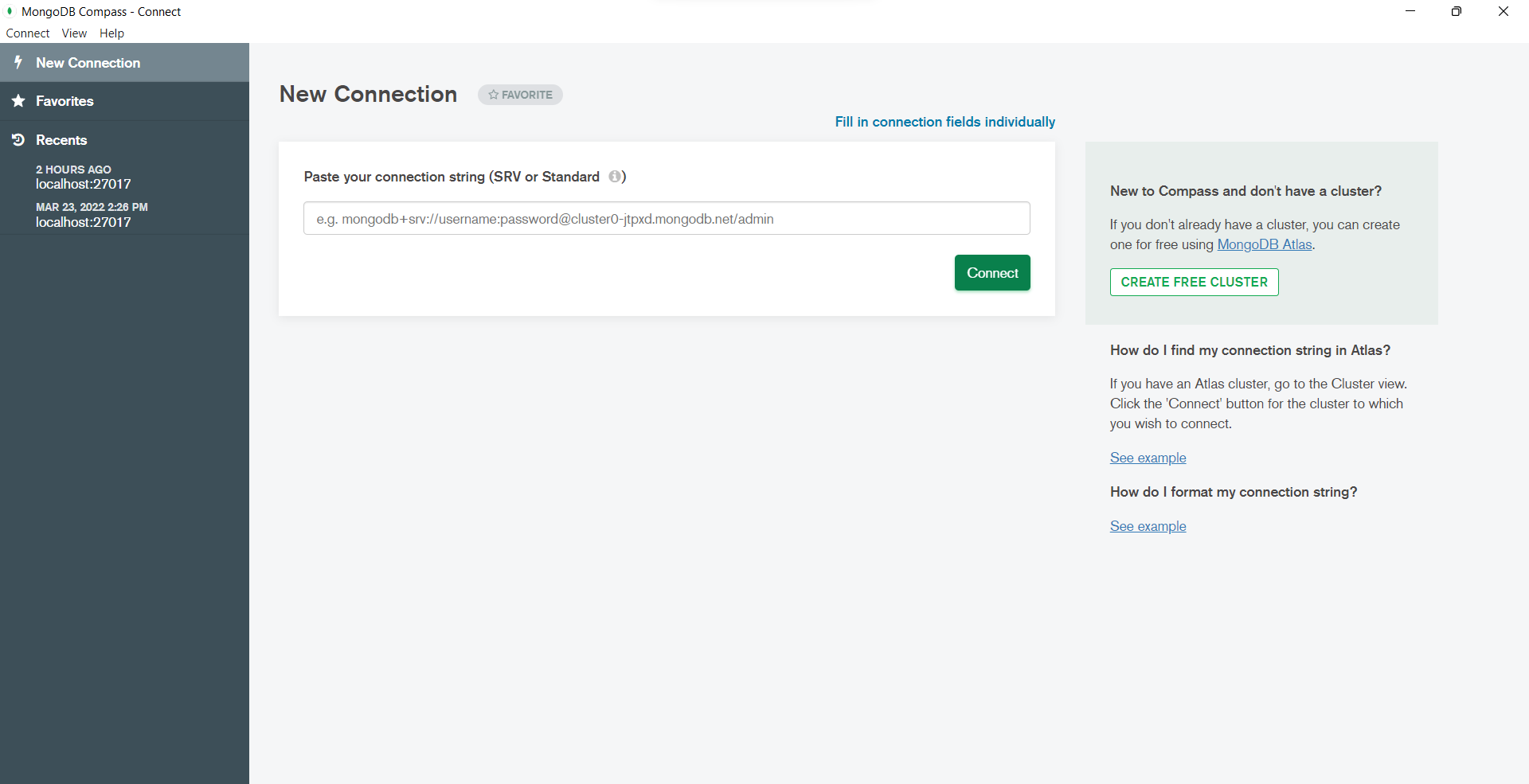




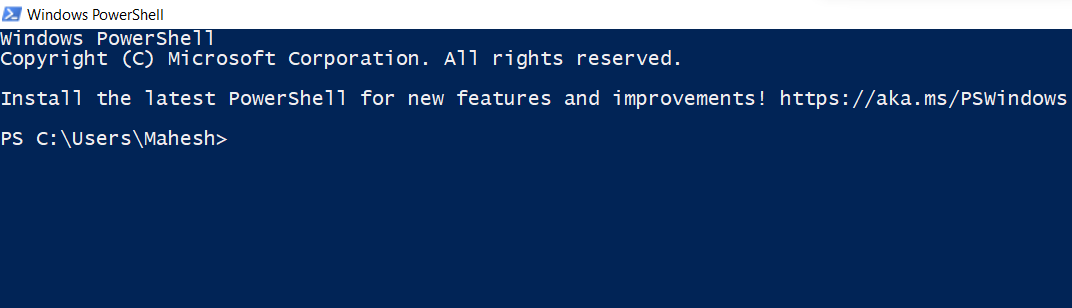




* After installing need to connect MongoDB Compass with Localhost so we can access the database



* After clicking on connect, localhost get connected then we can create database, collection & can also execute command by compass.
* But we will do the command on PowerShell.
* Open PowerShell



* Starting with MongoDB Command on powershell

View all databases

show dbs



Create a new or switch databases

use dbName



View current Database

db



Create a collection named 'comments'

db.createCollection('comments')



Insert One Row

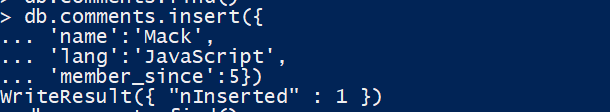
db.comments.insert({

'name': 'Mack',

'lang': 'JavaScript',

'member\_since': 5

})



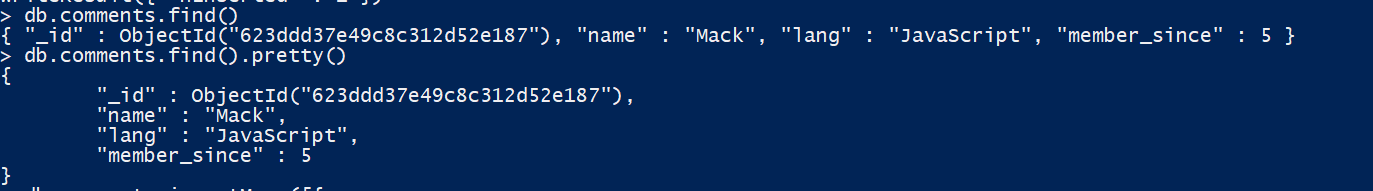
Show all Rows in a Collection

&

Show all Rows in a Collection (Prettified)

db.comments.find()

db.comments.find().pretty()



Insert many Rows

db.comments.insertMany([{

'name': 'Harry',

'lang': 'JavaScript',

'member\_since': 5

},

{'name': 'Rohan',

'lang': 'Python',

'member\_since': 3

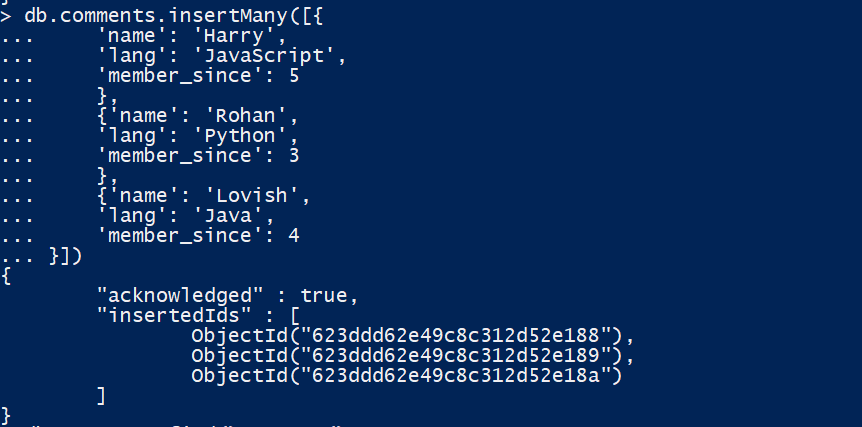
},

{'name': 'Lovish',

'lang': 'Java',

'member\_since': 4

}])



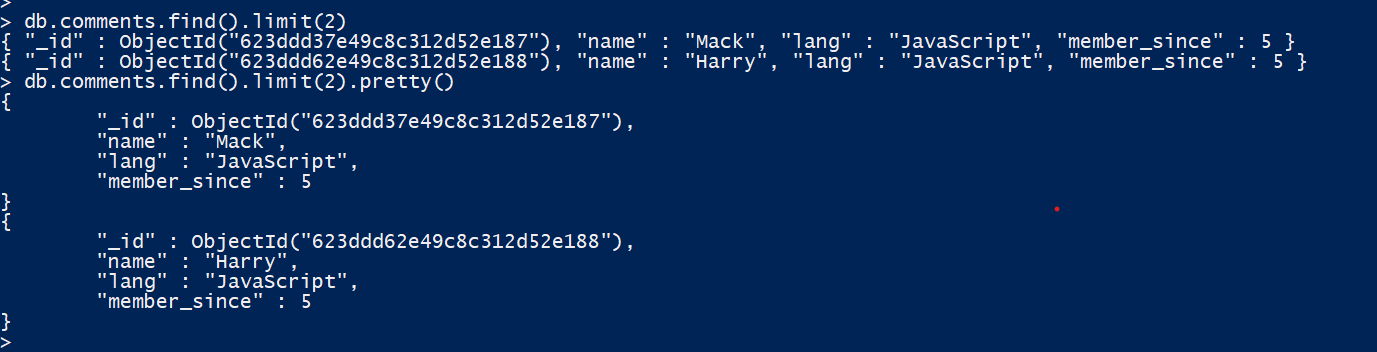
Search in a MongoDb Database

db.comments.find({lang:'Python'})



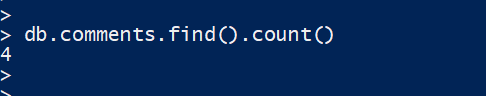
Limit the number of rows in output

db.comments.find().limit(2)



Count the number of rows in the output

db.comments.find().count()



Update a row & if upsert is true than new row is created

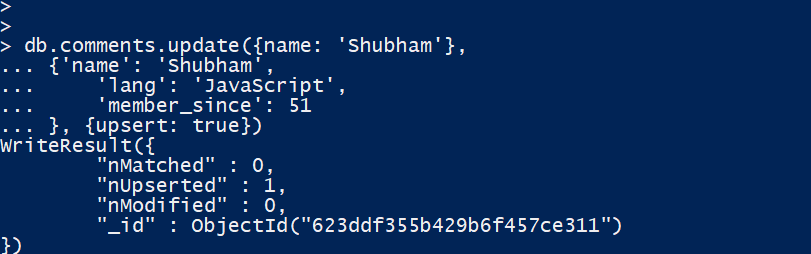
db.comments.update({name: 'Shubham'},

{'name': 'Shubham',

'lang': 'JavaScript',

'member\_since': 51

}, {upsert: true})



Mongodb Increment Operator

db.comments.update({name: 'Rohan'},

{$inc:{

member\_since: 2

}})

Mongodb Rename Operator

db.comments.update({name: 'Rohan'},

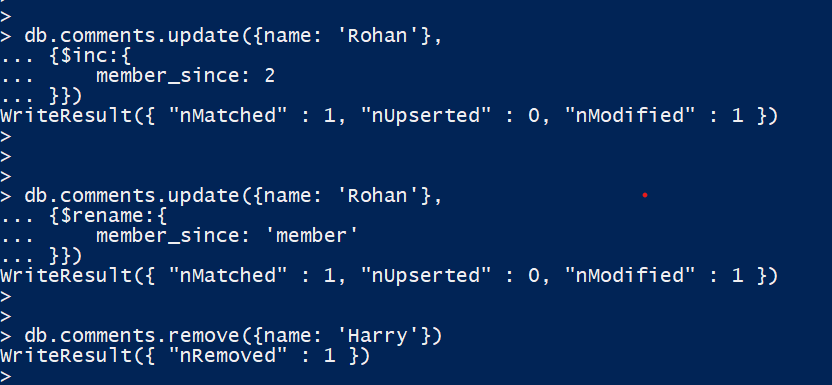
{$rename:{

member\_since: 'member'

}})

Delete Row

db.comments.remove({name: 'Harry'})



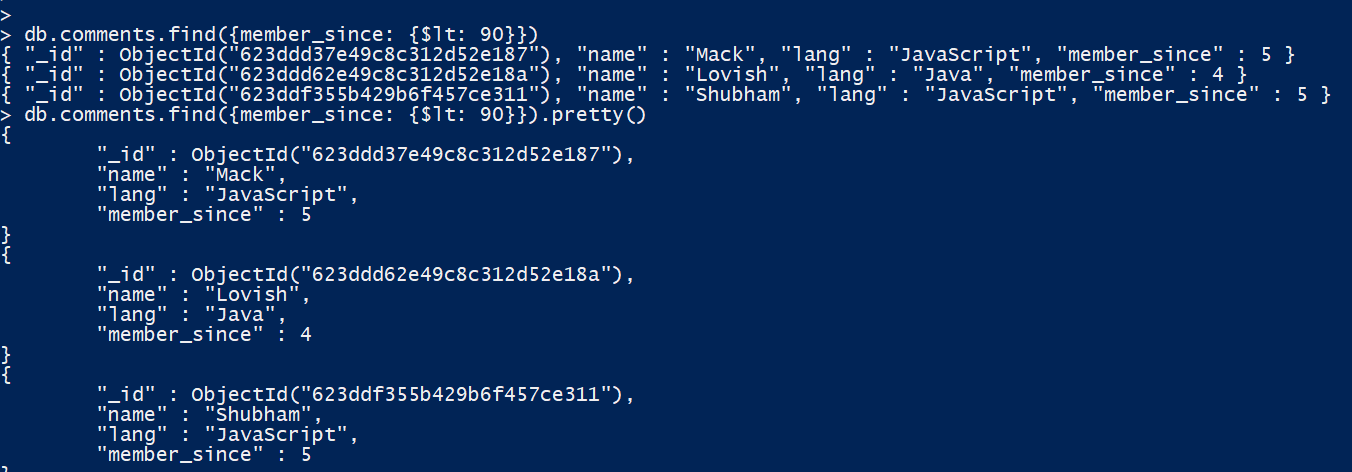
Less than/Greater than/ Less than or Eq/Greater than or Eq command

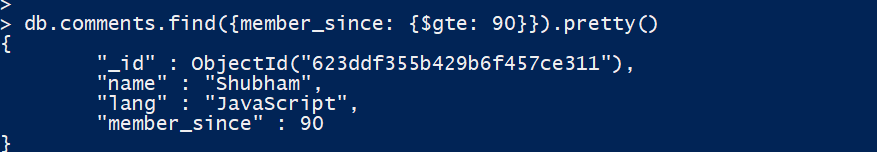
db.comments.find({member\_since: {$lt: 90}})

db.comments.find({member\_since: {$lte: 90}})

db.comments.find({member\_since: {$gt: 90}})

db.comments.find({member\_since: {$gte: 90}})





* Final collection data in the database

