3.3 Operations in MongoDB

This section will guide you to:

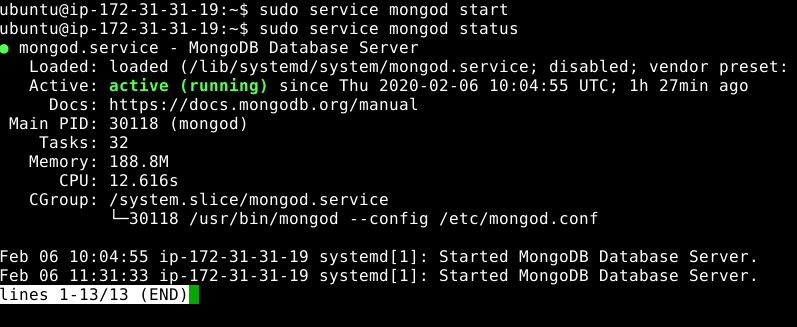
* Create a MongoDB database using command prompt
* Write a program to perform different operations on MongoDB

This lab has three subsections, namely:

* + 1. Setting up MongoDB server and shell
    2. Writing a program to perform different operations on MongoDB using command line prompt
    3. Pushing the code to GitHub repositories

**Step 3.3.1:** Setting up MongoDB server and shell

* Open the terminal
* Type **sudo service mongod start** to start the MongoDB server
* Type **sudo service mongod status** to check the status of the MongoDB server



* Type ctrl + c to exit the status
* Type **mongo --quiet** to start the mongo shell

**Step 3.2.2:** Writing a program to perform different operations on MongoDB using command line prompt

* Create a **generalStore** database using **use generalStore** command
* Write the code below resolving the syntactical errors
* Ordered bulk operation:

> show dbs 🡪showing the available databases

> use generalStore 🡪creating the database

> var bulk = db.items.initializeOrderedBulkOp();

> bulk.insert( { "\_id": 1, "item": "pen", "available": true, "soldQty":200})

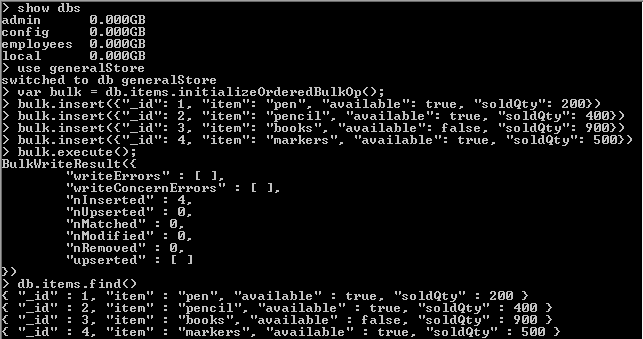
> bulk.insert( { "\_id": 2, "item": "pencil", "available": true, "soldQty":400})

> bulk.insert( { "\_id": 3, "item": "books", "available": false, "soldQty":900})

> bulk.insert( { "\_id": 4, "item": "markers", "available": true, "soldQty":500})

> bulk.execute()

> db.items.find()



* Unordered bulk operation:

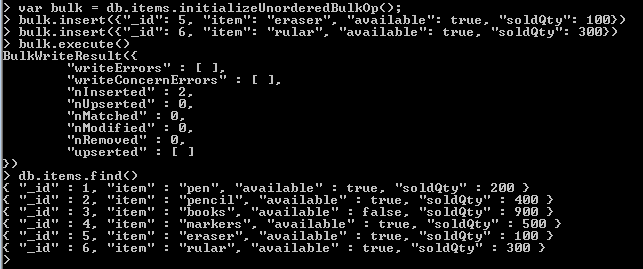
> var bulk = db.items.initializeUnorderedBulkOp();

> bulk.insert( { "\_id": 5, "item": "eraser", "available": true, "soldQty":100})

> bulk.insert( { "\_id": 6, "item": "rular", "available": true, "soldQty":300})

> bulk.execute()

> db.items.find()



* Insert operation:

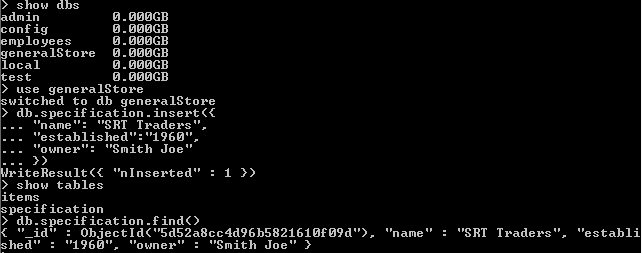
> show dbs

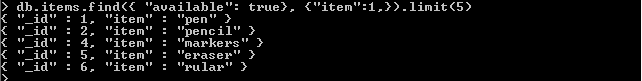
> use generalStore

> db.specification.insert( { "name": "SRT Traders", "established": "1960", "owner": "Smith Joe"} )

> show tables

> db.specification.find()





* Specify condition operation:

> db.items.find().pretty()

> db.items.find({})

> db.items.find( { "item": "pen"})



* $in and $lt operation:

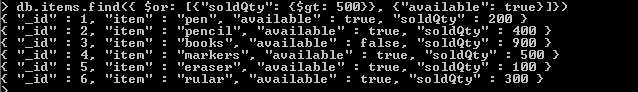
> db.items.find( { "available": {$in: [true, false]}} )

> db.items.find( { "available": true, "soldQty": {$lt: 900}})



* $or and $gt operations:

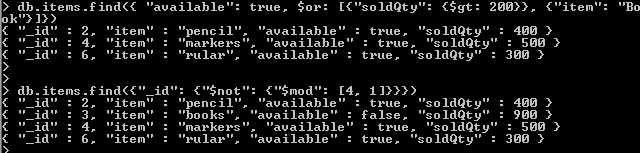
> db.items.find( { $or: [{ "soldQty": {$gt: 500}}, {"available": true}]} )



* OR and NOT operations:

> db.items.find( { "available": true, $or: [{ "soldQty": {$gt: 200}}, {"item": "Book"}]})

> db.items.find( { "\_id": {$not: {$mod: [4,1]}}})



* Regular expression operation:

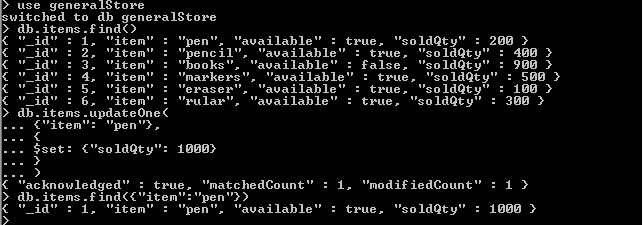
> db.items.find({ "item": /pe/i})

> db.items.find({ "item": /pen?/i })

* Update operation:

> db.items.updateOne({ "item": "pen" }, { $set: {"soldQty": 10000} })

> db.items.find({ "item": "pen" })



* Update operation using $inc:

> db.items.updateOne({ "item": "pencil" }, { $inc: {“soldQty”: 10} })

> db.items.find({ "item": "pen" })



* Remove operation:

> db.items.remove({ "item": "pencil" })

> db.items.drop()



**Step 3.3.3:** Pushing the code to GitHub repositories

Open your command prompt and navigate to the folder where you have created your files.

cd <folder path>

Initialize your repository using the following command:

git init

Add all the files to your git repository using the following command:

git add .

Commit the changes using the following command:

git commit . -m “Changes have been committed.”

Push the files to the folder you initially created using the following command:

git push -u origin master