

Practical 8

Aim: Implement an application that stores big data in Hbase / MongoDB and manipulate it using R / Python.

Theory:

MongoDB is an open-source and the leading NoSQL database. It is a document-oriented database that offers high performance, easy scalability, and high availability. It uses documents and collections to organize data rather than relations. This makes it an ideal database management system for the storage of unstructured data.

MongoDB uses replica sets to ensure there is a high availability of data. Each replica set is made up of two or more replicas of data. This gives its users the ability to access their data at any time. The replica sets also create fault tolerance. MongoDB scales well to accommodate more data. It uses the sharing technique to scale horizontally and meet the changing storage needs of its users. MongoDB was developed to help developers unleash the power of data and software.

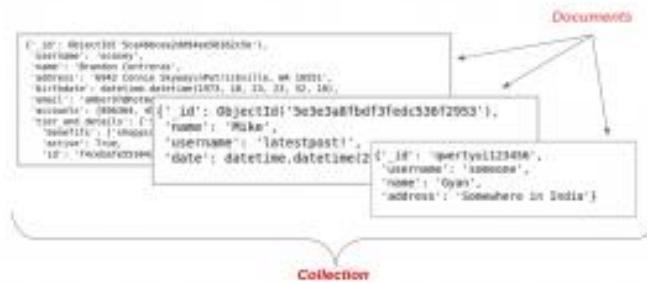
MongoDB is an unstructured database. It stores data in the form of documents. MongoDB is able to handle huge volumes of data very efficiently and is the most widely used NoSQL database as it offers rich query language and flexible and fast access to data.

The Architecture of a MongoDB Database

The information in MongoDB is stored in **documents**. Here, a document is analogous to **rows** in structured databases.

- Each document is a collection of key-value pairs
- Each key-value pair is called a **field**
- Every document has an **_id** field, which uniquely identifies the documents
- A document may also contain nested documents
- Documents may have a varying number of fields (they can be blank as well)

These documents are stored in a **collection**. A collection is literally a collection of documents in MongoDB. This is analogous to **tables** in traditional databases.



Unlike traditional databases, the data is generally stored in a single collection in MongoDB, so there is no concept of joins (except `$lookup` operator, which performs *left-outer-join* like operation). MongoDB has the nested document instead.

PyMongo is a Python library that enables us to connect with MongoDB. It allows us to perform basic operations on the MongoDB database.

We have chosen Python to interact with MongoDB because it is one of the most commonly used and considerably powerful languages for data science. PyMongo allows us to retrieve the data with dictionary-like syntax.

We can also use the dot notation to access MongoDB data. Its easy syntax makes our job a lot easier. Additionally, PyMongo's rich documentation is always standing there with a helping hand. We will use this library for accessing MongoDB.

Steps of the installation:

Step 1: Download MongoDB

Go to official website: MongoDB Community server

<https://www.mongodb.com/try/download/community>

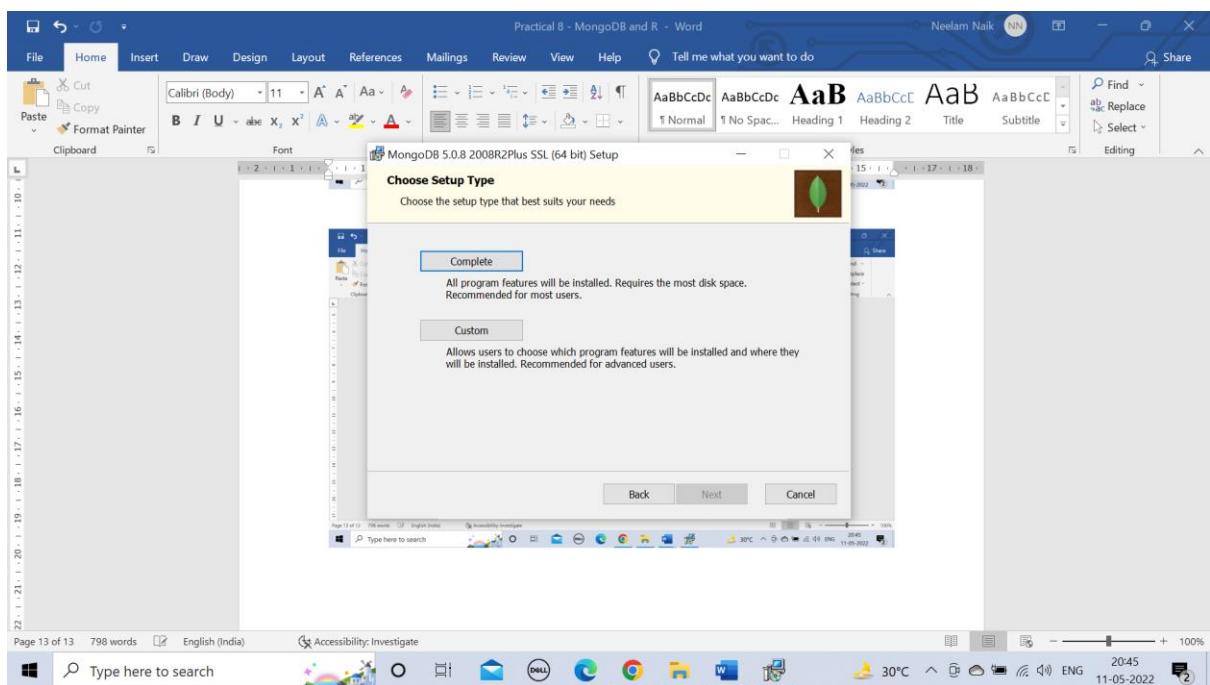
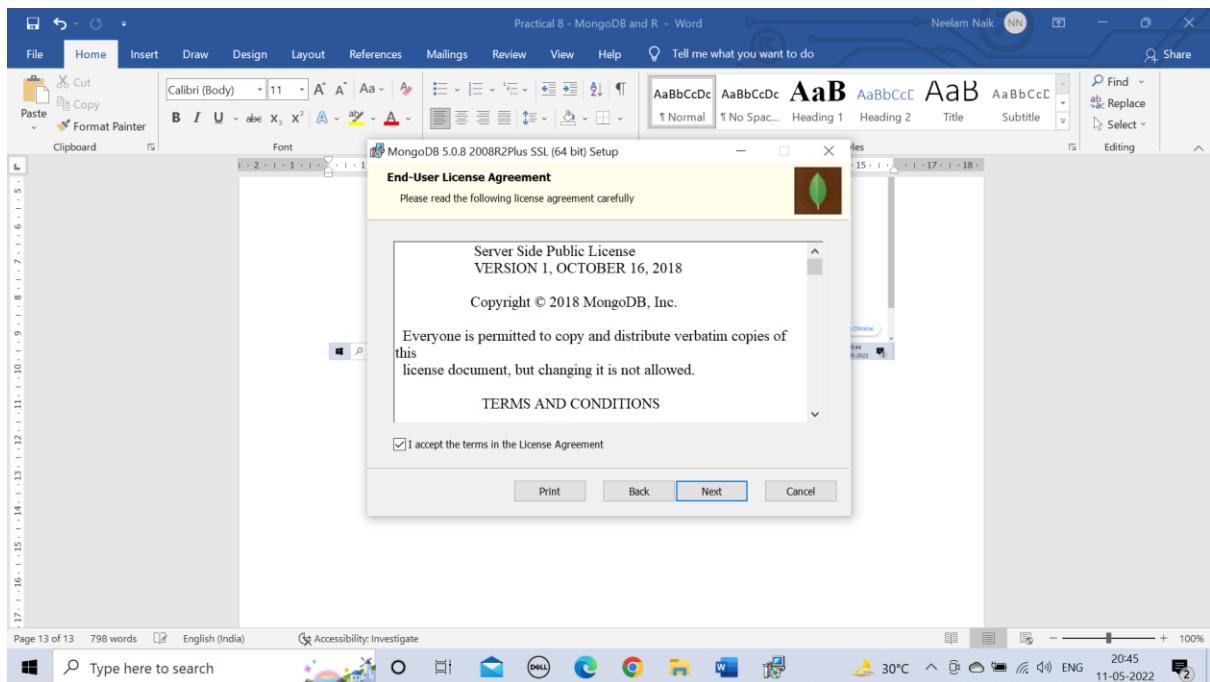
The screenshot shows the MongoDB website at [mongodb.com/try/download/community](https://www.mongodb.com/try/download/community). The main content area is titled "MongoDB Community Server". It describes the Community version as offering a flexible document data model and support for ad-hoc queries, secondary indexing, and real-time aggregations. It also mentions MongoDB Atlas as a managed service. A sidebar on the right lists "Available Downloads" for version 5.0.8 (current) on Windows in MSI format, with a large green "Download" button.

Click on download

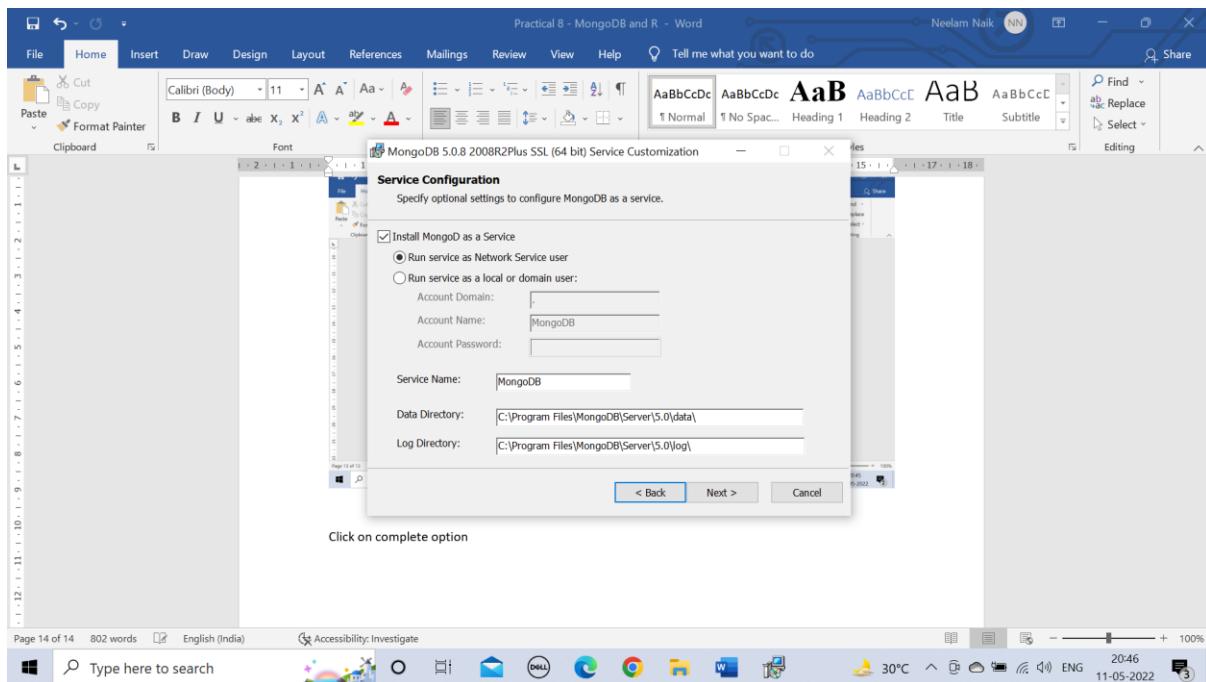
Step 2: Install MongoDB

It will download msi file. Click on it and Start the installation.

The screenshot shows a Windows 10 desktop with a Google Chrome window open. The window displays the "Welcome to the MongoDB 5.0.8 2008R2Plus SSL (64 bit) Setup Wizard". The setup wizard is a standard Windows-based application with a "Next" button visible. The desktop taskbar at the bottom shows various pinned icons and the date/time as 11-05-2022.

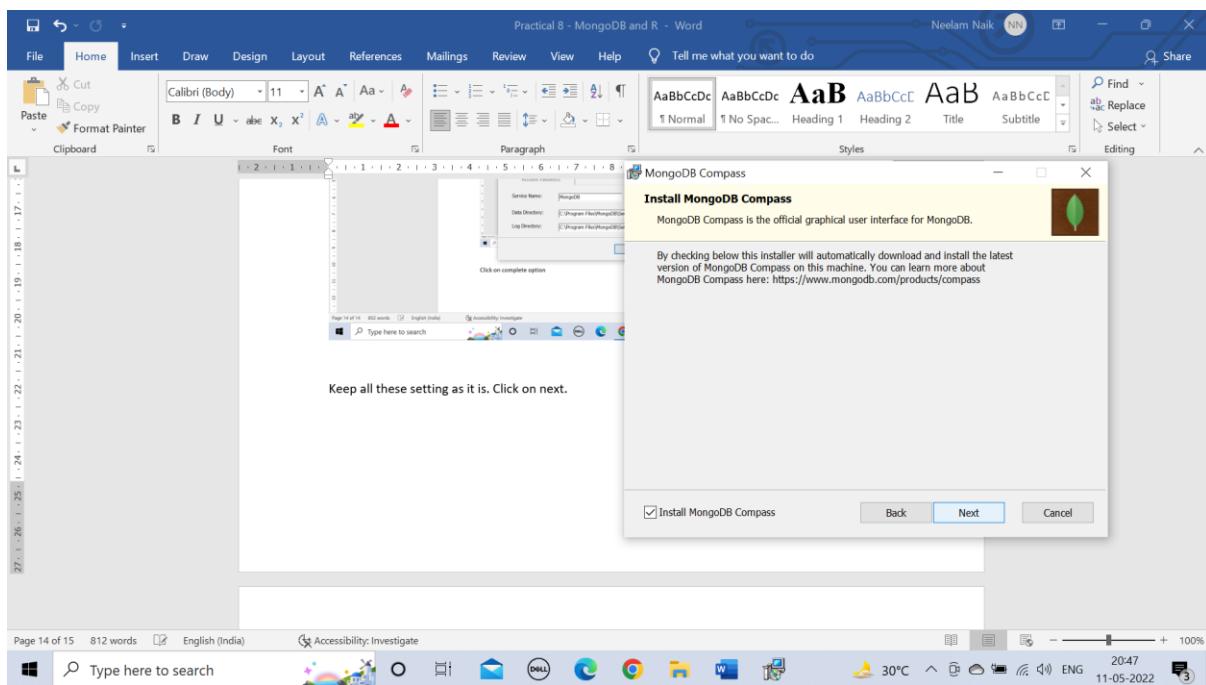


Click on complete option



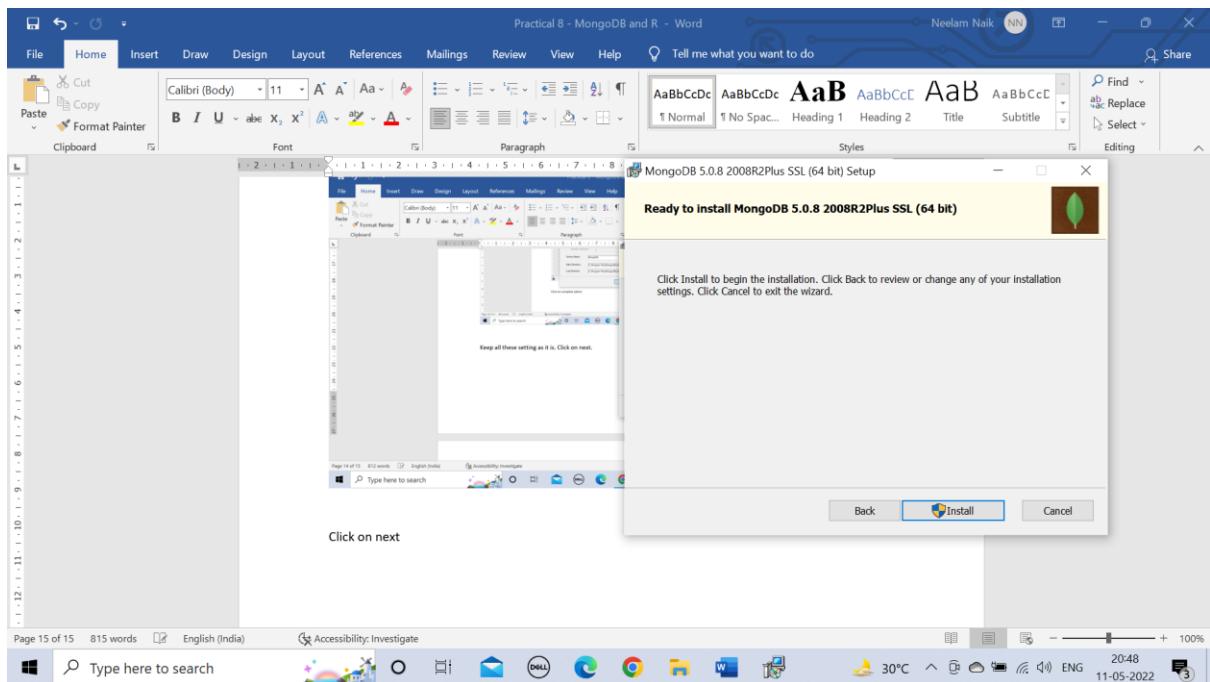
Click on complete option

Keep all these setting as it is. Click on next.

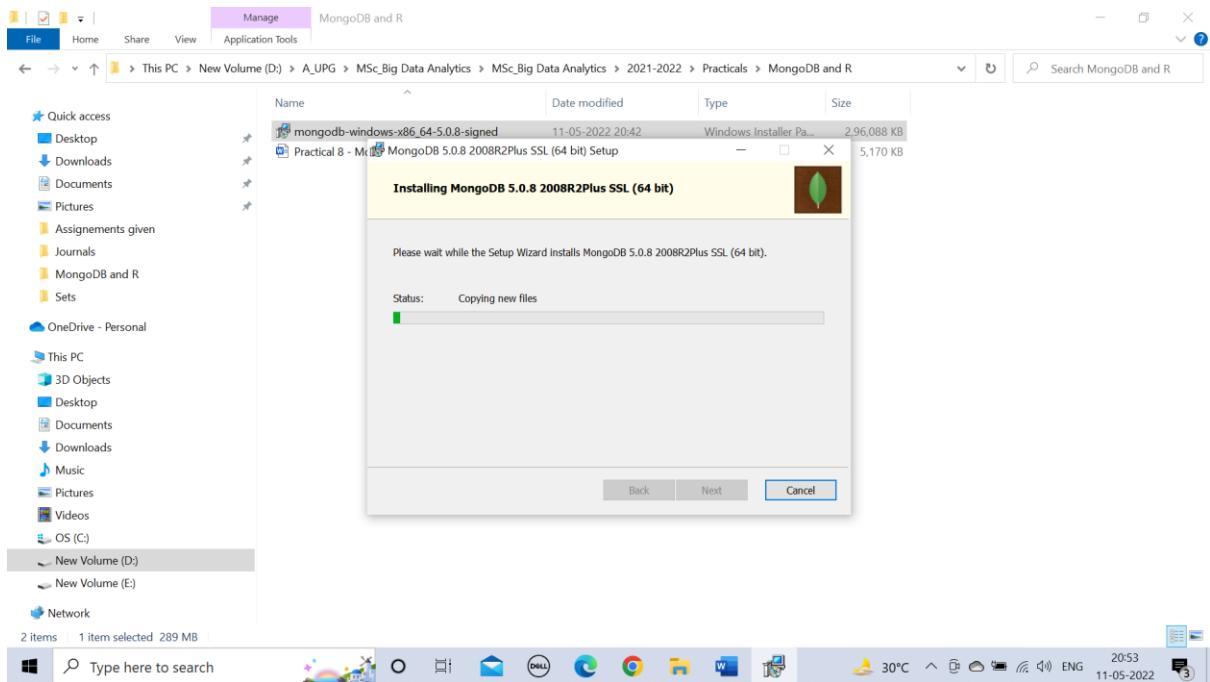


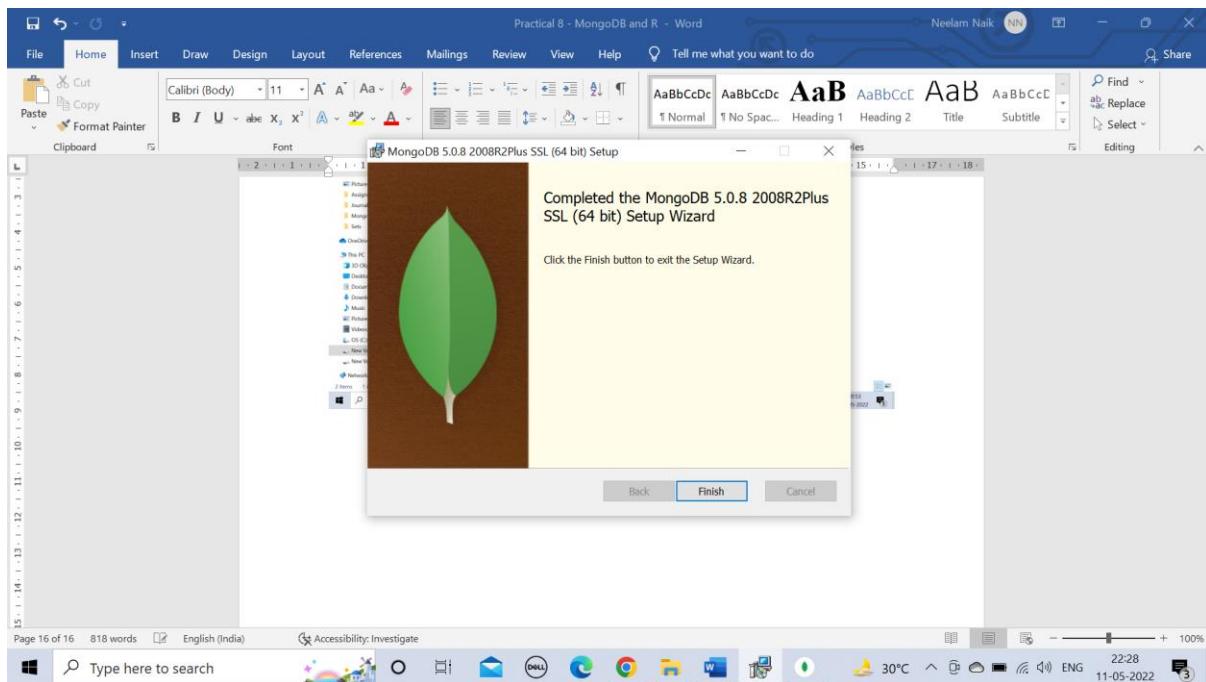
Keep all these setting as it is. Click on next.

Click on next



Click on install



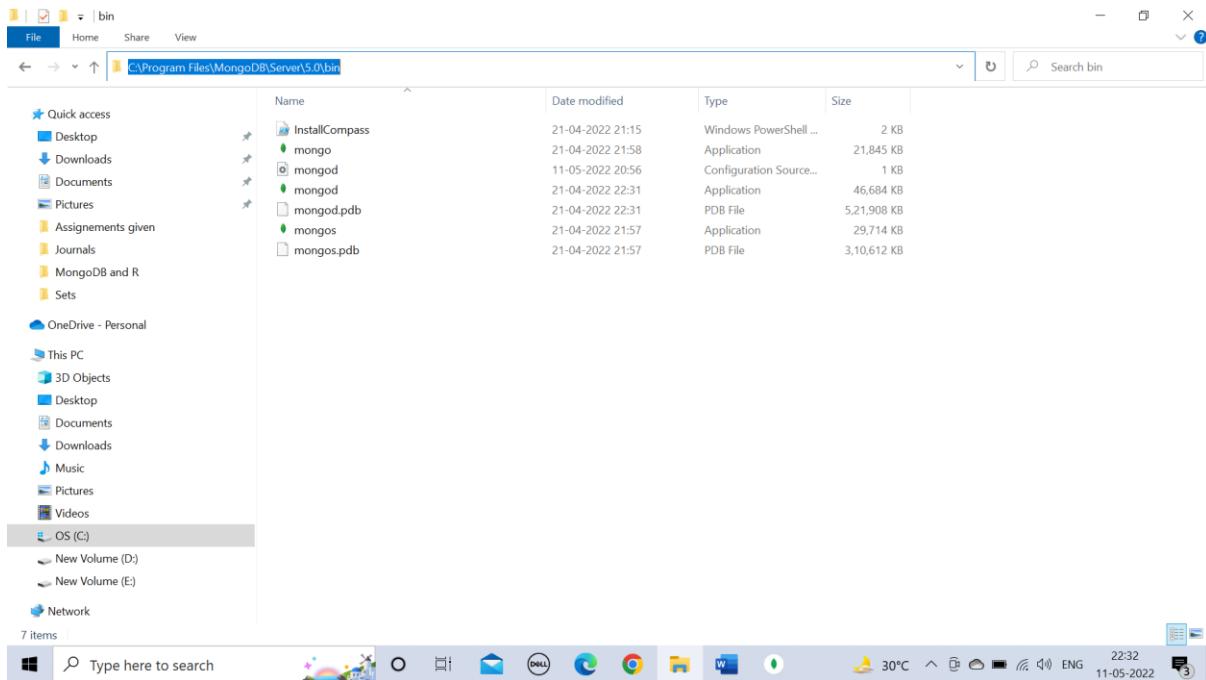


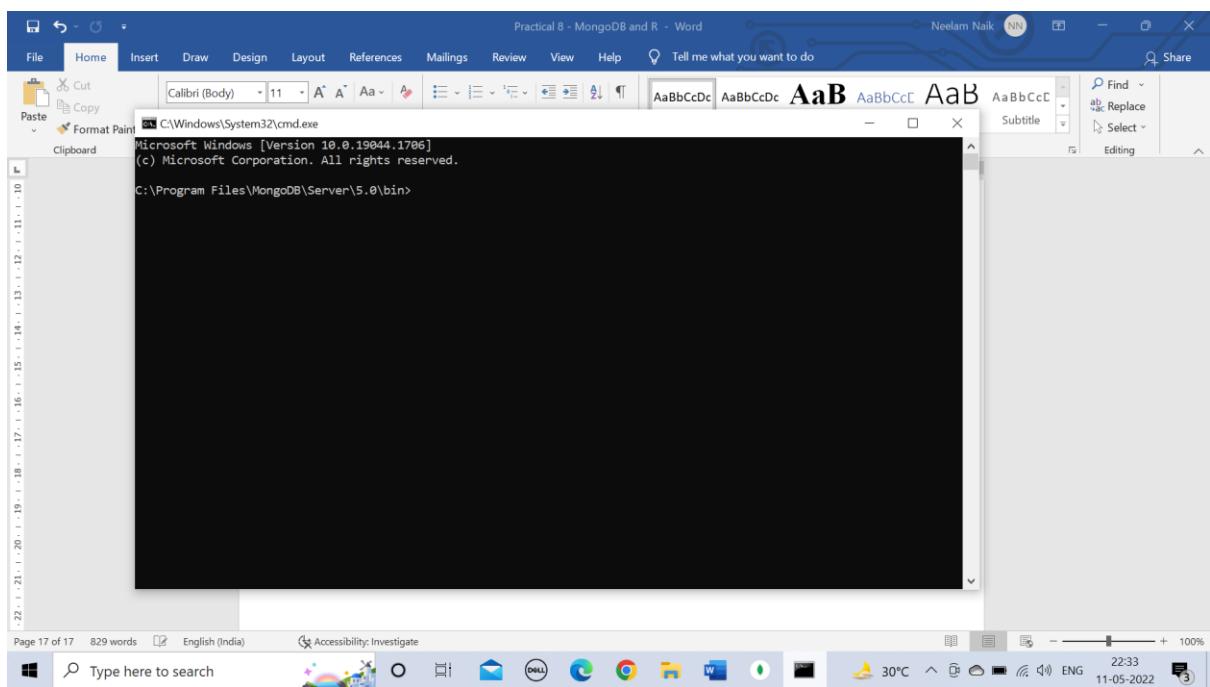
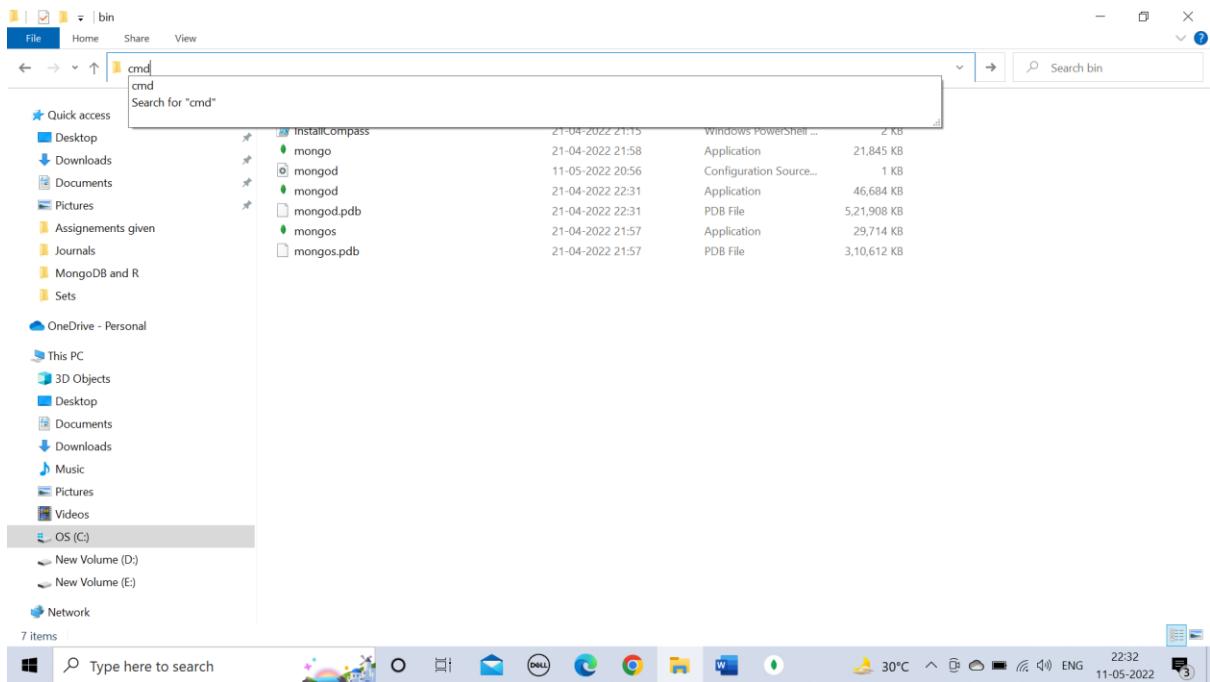
Step 3: Verify MongoDB Installation

Now go to

C:\Program Files\MongoDB\Server\5.0\bin

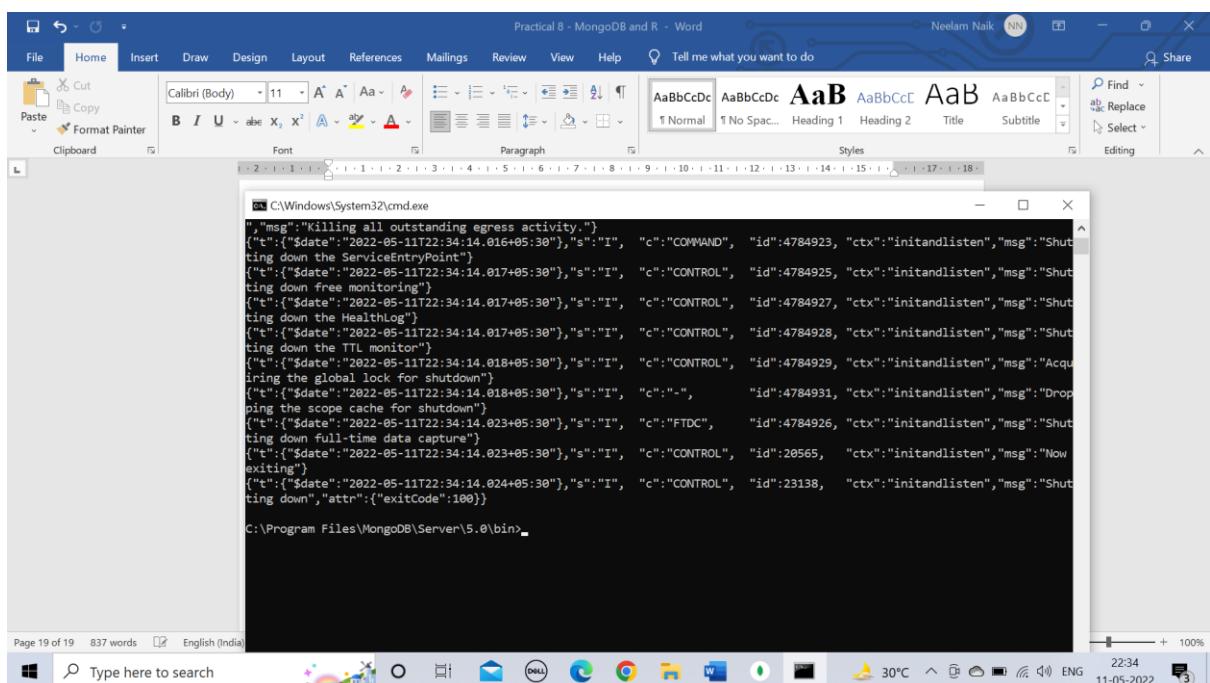
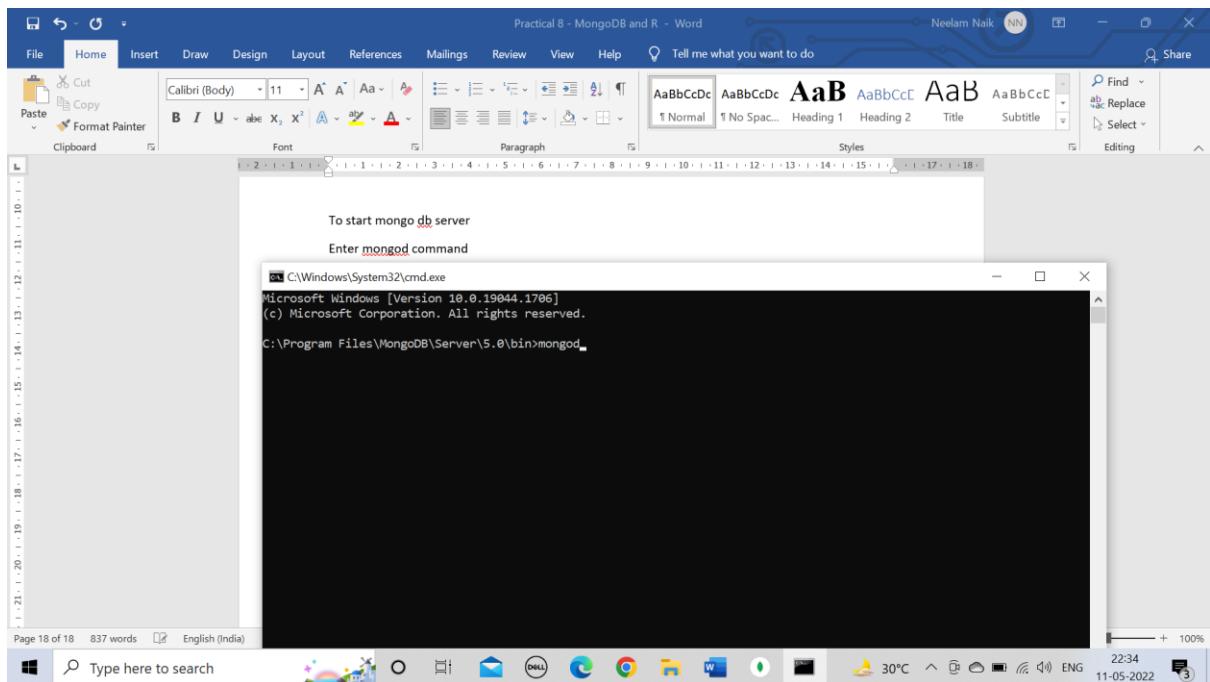
Start command prompt from this location





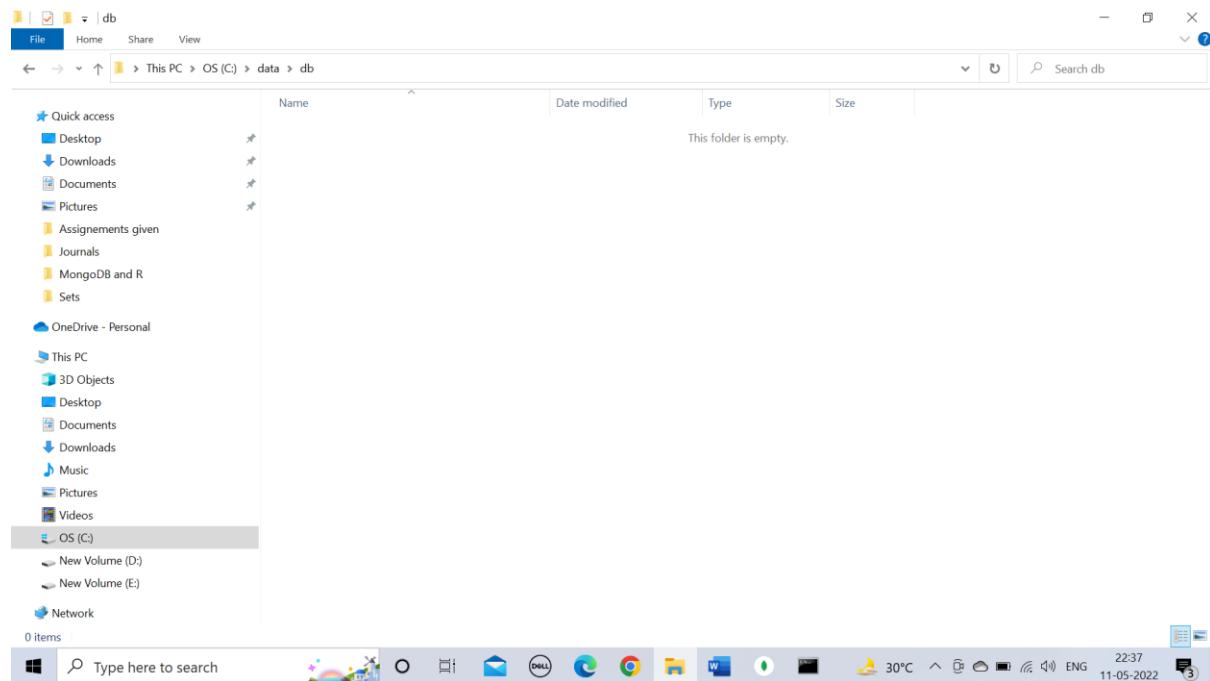
To start mongo db server

Enter mongod command

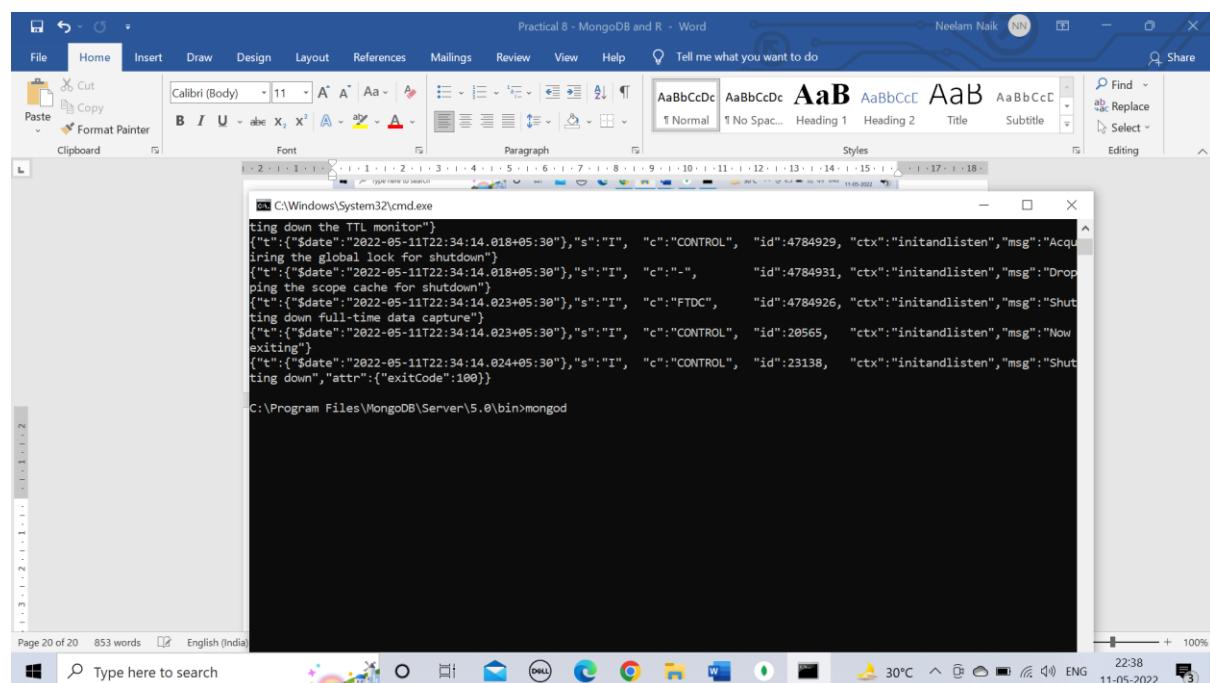


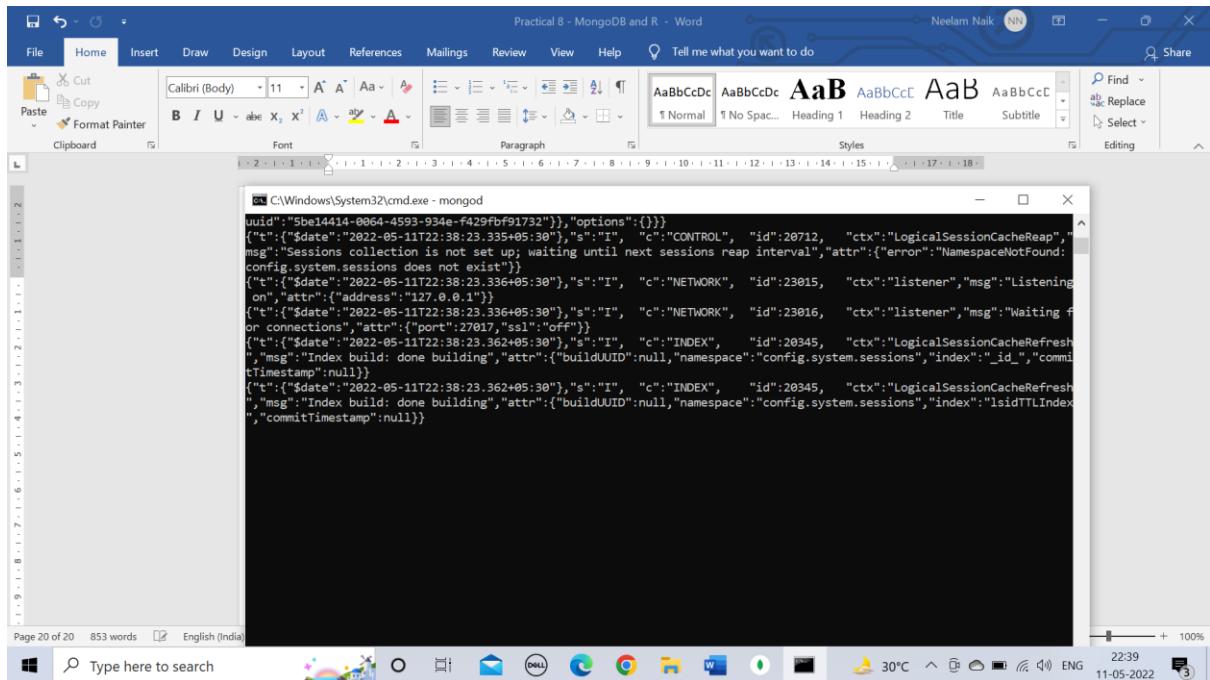
It says c:\data\db directory not found

So create c:\data\db



Run the above command mongod once again





Mongo daemon is started now

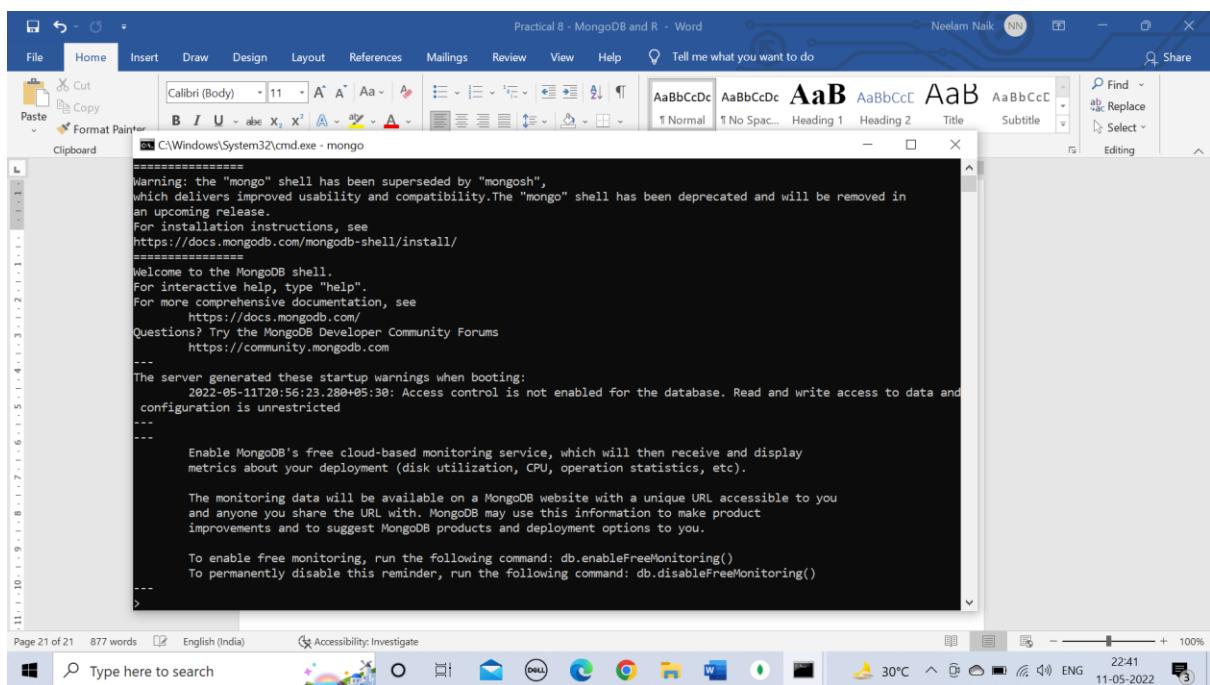
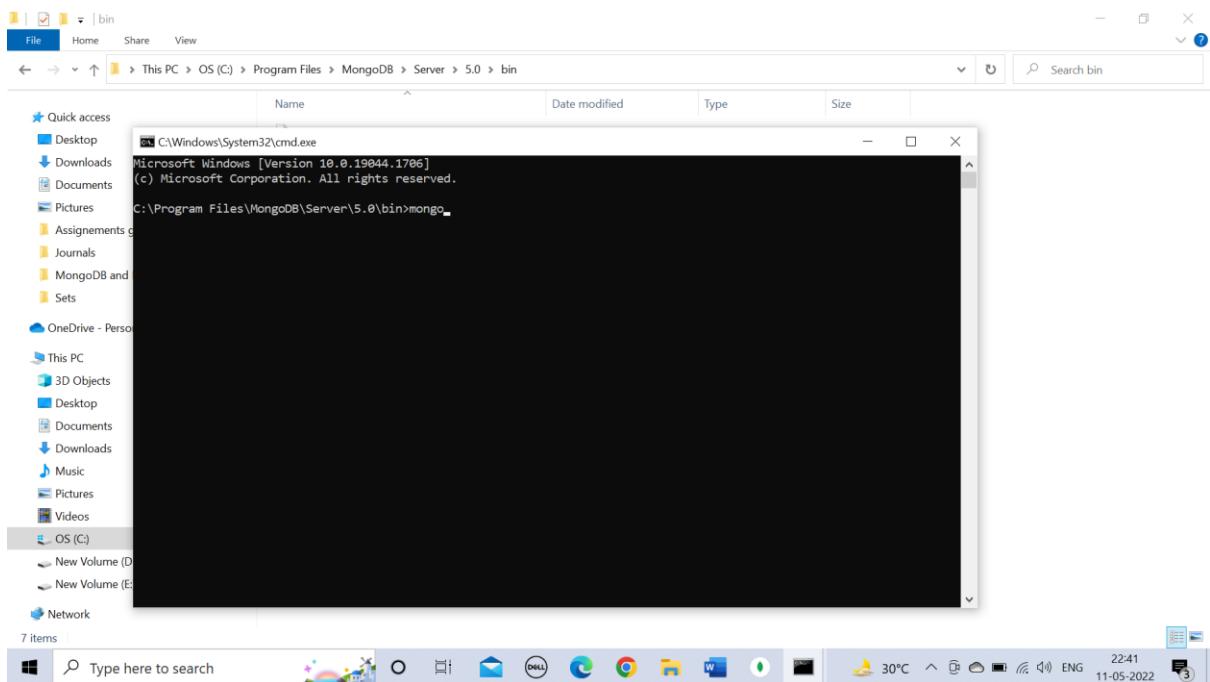
To open the mongo shell

Go to

C:\Program Files\MongoDB\Server\5.0\bin

Start command prompt from this location

Fire the command: mongo



Mongo shell is started

To see all the default databases:

>show dbs

Enable MongoDB's free cloud-based monitoring service, which will then receive and display metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique 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.

```

> show dbs
admin 0.000GB
config 0.000GB
local 0.000GB
>

```

ldest timestamp: (0, 0) , meta checkpoint timestamp: (0, 0) base write gen: 1"}}
{"t": {"\$date": "2022-05-11T22:41:23.218+05:30"}, "s": "I", "c": "STORAGE", "id": 22430, "ctx": "Checkpoint", "msg": "WiredTiger message", "attr": {"message": "[1652289083:218571][19936:140719219430736], WT_SESSION.checkpoint: [WT_VERB_CHECKPOINT_PROGRESS] saving checkpoint snapshot min: 39, snapshot max: 39 snapshot count: 0, o ldest timestamp: (0, 0) , meta checkpoint timestamp: (0, 0) base write gen: 1"}}
{"t": {"\$date": "2022-05-11T22:42:23.233+05:30"}, "s": "I", "c": "STORAGE", "id": 22430, "ctx": "Checkpoint", "msg": "WiredTiger message", "attr": {"message": "[1652289143:23156][19936:140719219430736], WT_SESSION.checkpoint: [WT_VERB_CHECKPOINT_PROGRESS] saving checkpoint snapshot min: 41, snapshot max: 41 snapshot count: 0, o ldest timestamp: (0, 0) , meta checkpoint timestamp: (0, 0) base write gen: 1"}}
{"t": {"\$date": "2022-05-11T22:43:23.253+05:30"}, "s": "I", "c": "STORAGE", "id": 22430, "ctx": "Checkpoint", "msg": "WiredTiger message", "attr": {"message": "[1652289203:253382][19936:140719219430736], WT_SESSION.checkpoint: [WT_VERB_CHECKPOINT_PROGRESS] saving checkpoint snapshot min: 43, snapshot max: 43 snapshot count: 0, o ldest timestamp: (0, 0) , meta checkpoint timestamp: (0, 0) base write gen: 1"}}

30°C 22:44 ENG 11-05-2022

To create new database named my_database:

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()

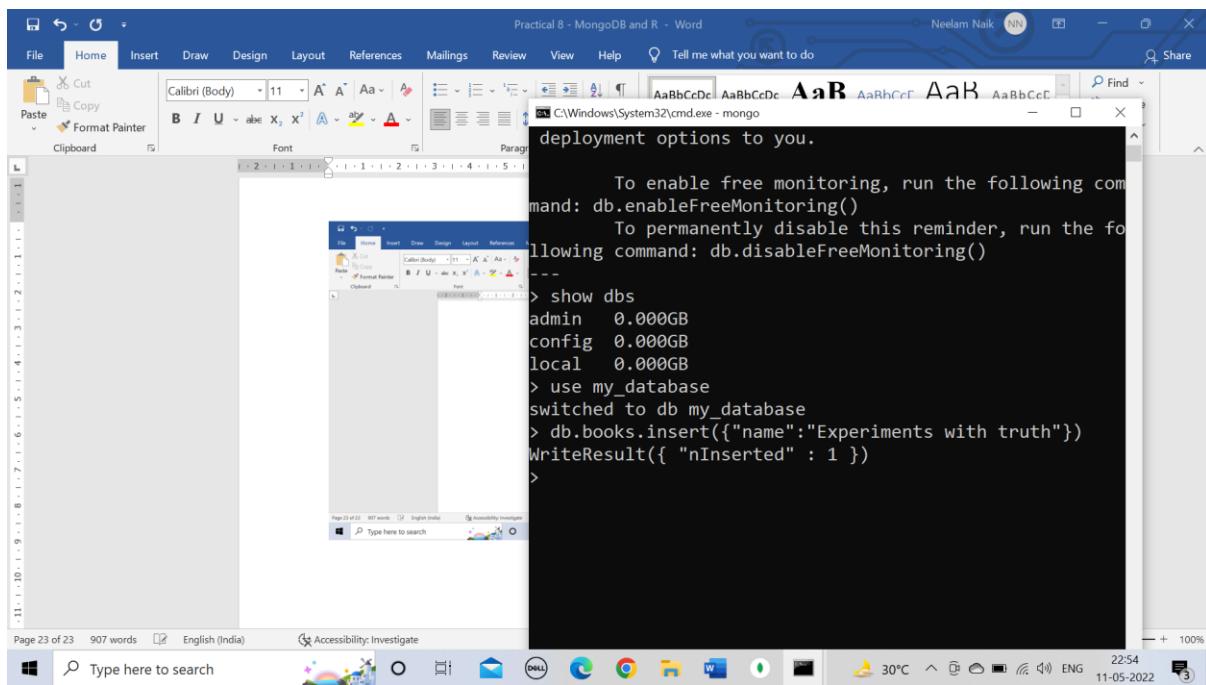
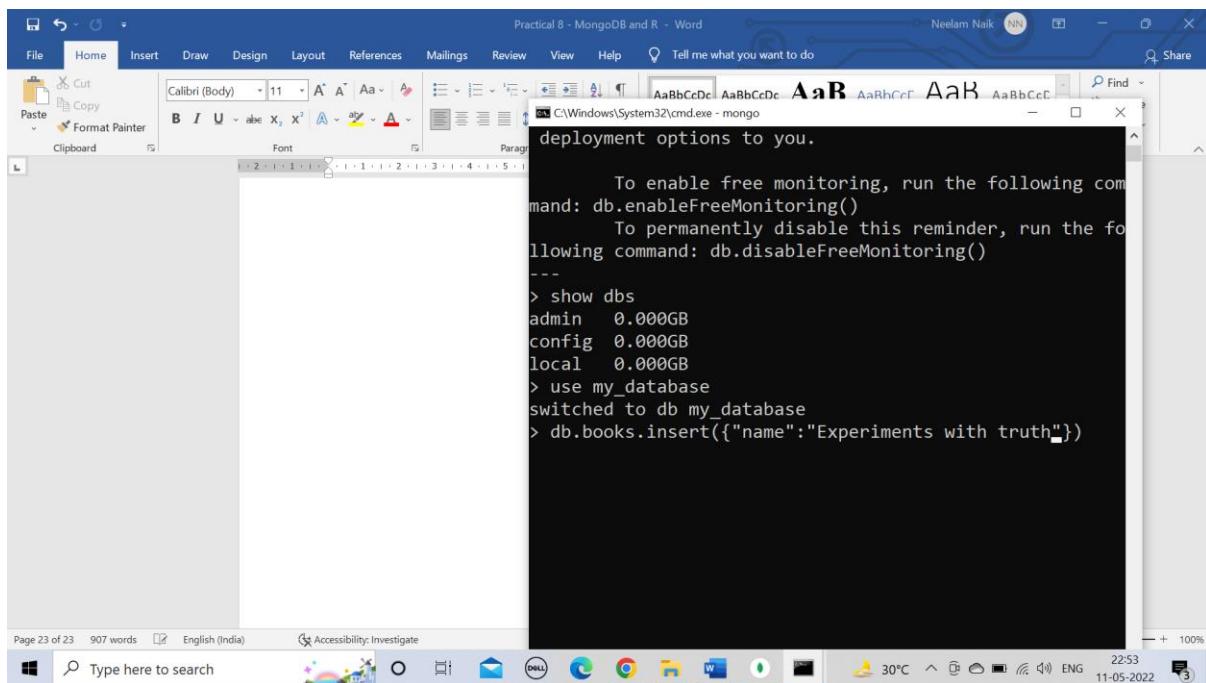
```

> show dbs
admin 0.000GB
config 0.000GB
local 0.000GB
> use my_database

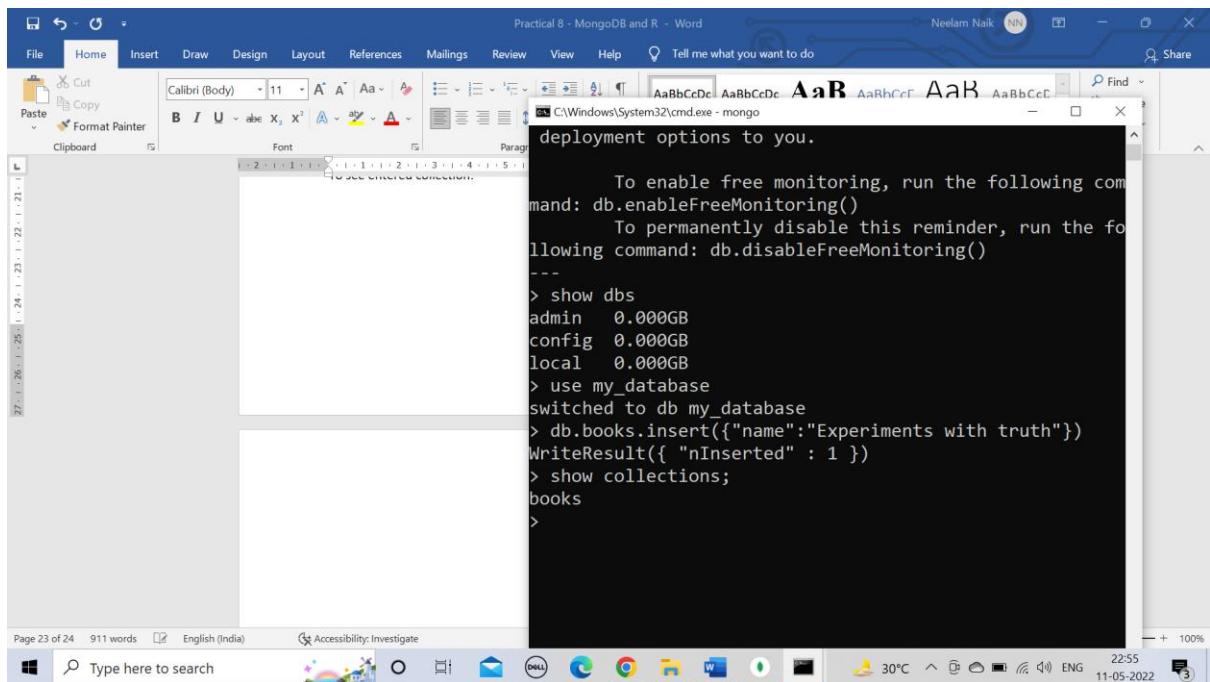
```

To create collection in the database:

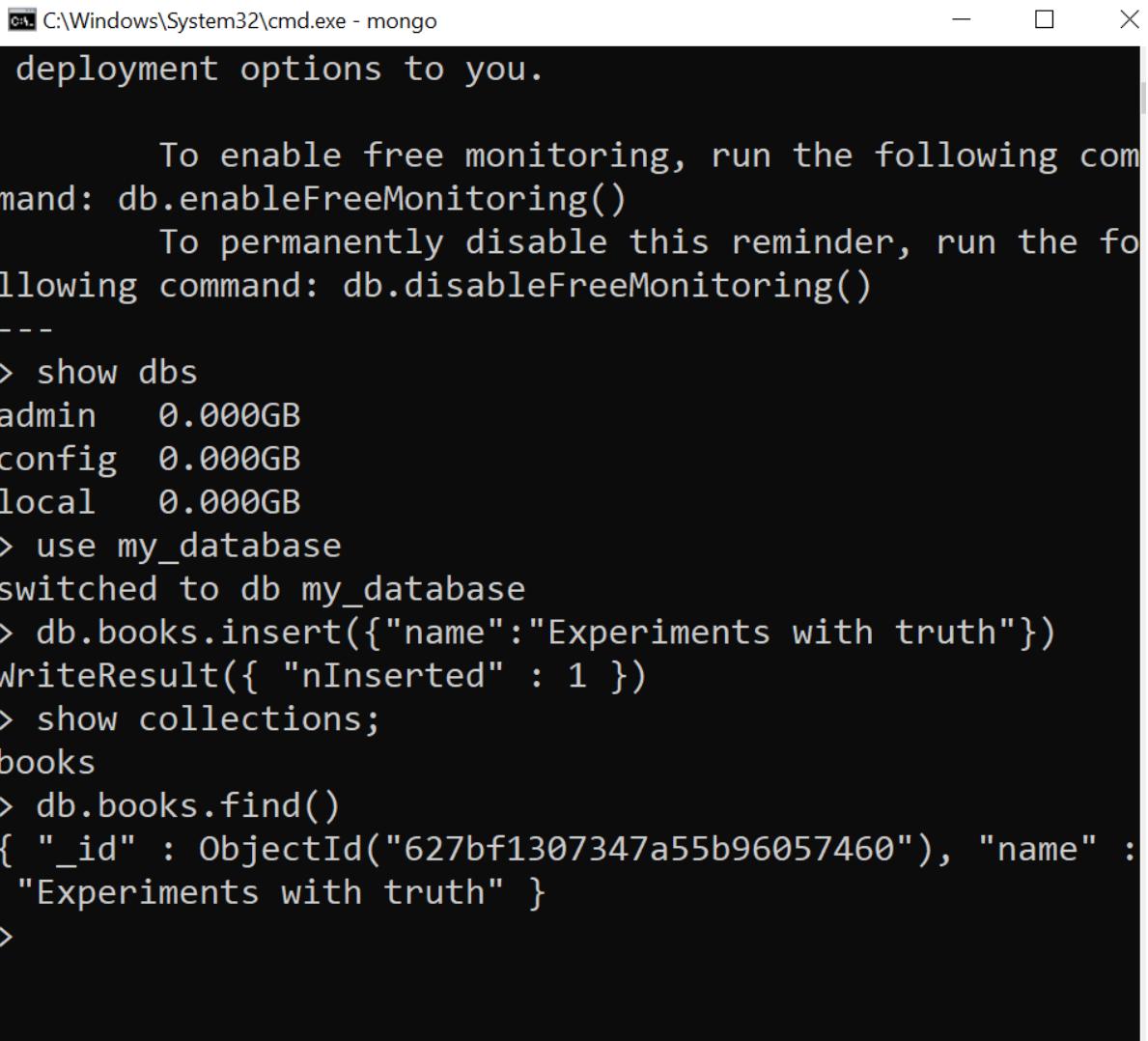
And insert json values into it:



To see entered collection:



To see all the documents in the collection:



```
C:\Windows\System32\cmd.exe - mongo
deployment options to you.

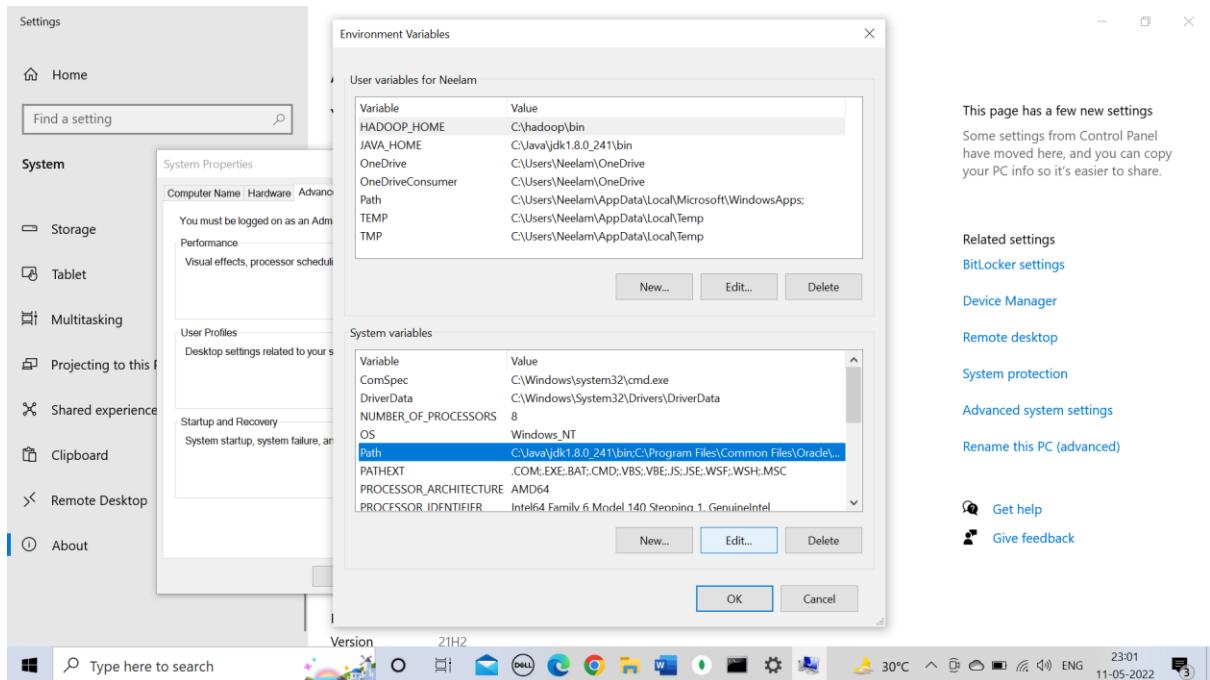
To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
> show dbs
admin    0.000GB
config   0.000GB
local    0.000GB
> use my_database
switched to db my_database
> db.books.insert({ "name": "Experiments with truth" })
WriteResult({ "nInserted" : 1 })
> show collections;
books
> db.books.find()
{ "_id" : ObjectId("627bf1307347a55b96057460"), "name" :
  "Experiments with truth" }
>
```

To set path of MongoDB server and shell

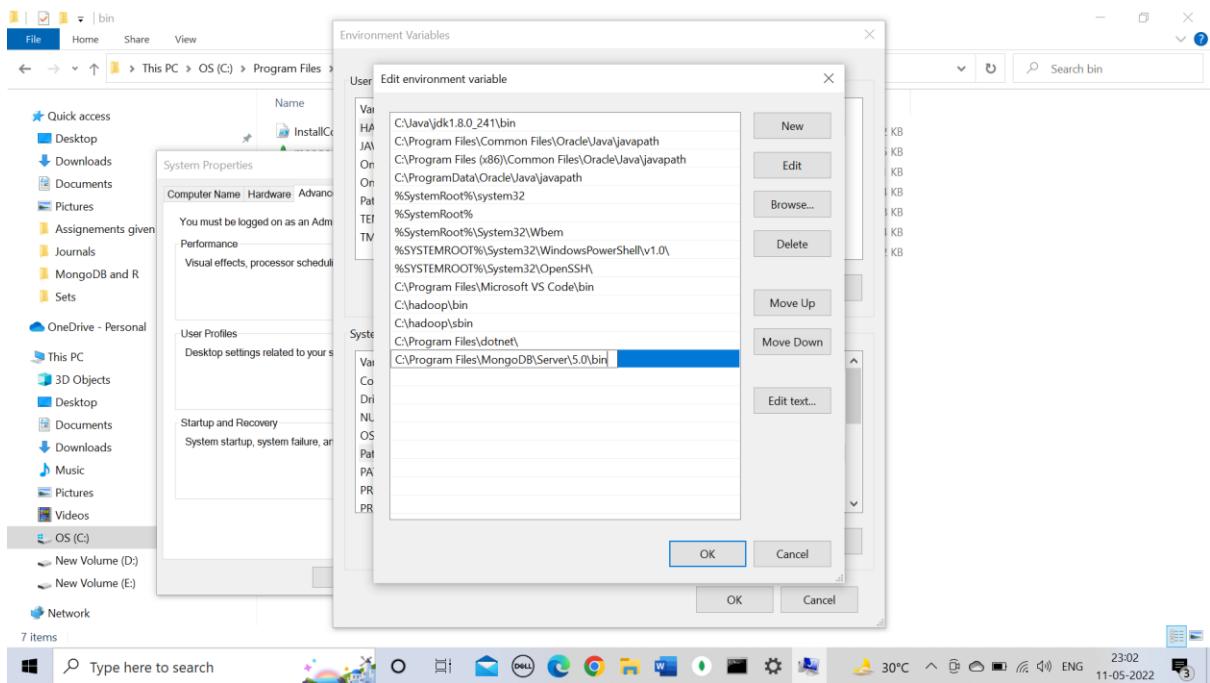
Copy in clipboard: C:\Program Files\MongoDB\Server\5.0\bin

Go to environment variables

Add mongodb path to system path variable



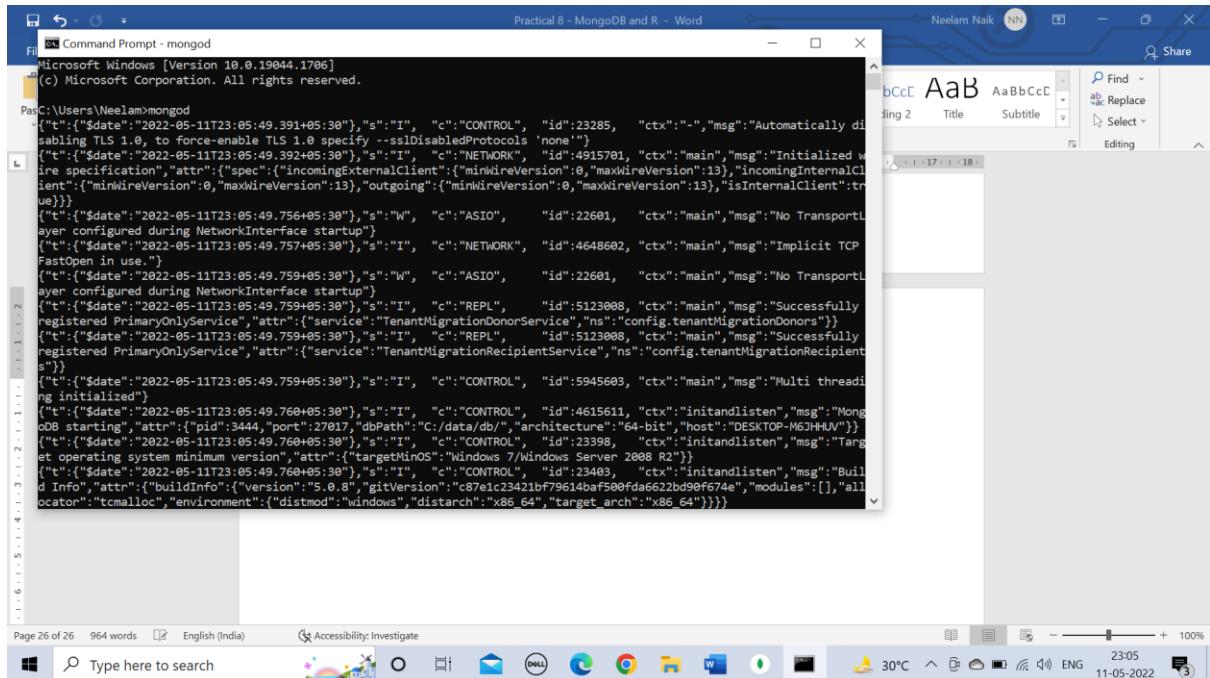
Click on New



Paste the path

Click on ok..ok

Run mongod and mongo command from command prompt once again from anywhere, it will start mongo server and mongo shell



```
C:\ Command Prompt - mongod
Microsoft Windows [Version 10.0.19044.1706]
(c) Microsoft Corporation. All rights reserved.

PS:C:\Users\Neelam> C:\Command Prompt - mongo
{"t":{"$date":2022-05-11T20:56:23.288Z}, "c": "Implicit session: session { \"id\" : UUID("33986e96-175b-406f-a7b7-b219cdf442ae") }", "m": "MongoDB server version: 5.0.8", "v": "MongoDB shell version v5.0.8", "r": "FastOpen in use, which delivers improved usability and compatibility. The \"mongo\" shell has been deprecated and will be removed in an upcoming release.", "w": "ayer configured dFor installation instructions, see https://docs.mongodb.com/mongodb-shell/install", "d": "registered Primary", "o": "The server generated these startup warnings when booting: 2022-05-11T20:56:23.288Z: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted", "n": "ng initialized", "l": "Enable MongoDB's free cloud-based monitoring service, which will then receive and display metrics about your deployment (disk utilization, CPU, operation statistics, etc).", "i": "The monitoring data will be available on a MongoDB website with a unique 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.", "s": "To enable free monitoring, run the following command: db.enableFreeMonitoring()", "u": "To permanently disable this reminder, run the following command: db.disableFreeMonitoring()"}>
```

To get the effect of running mongod as service, Restart your windows operating system.

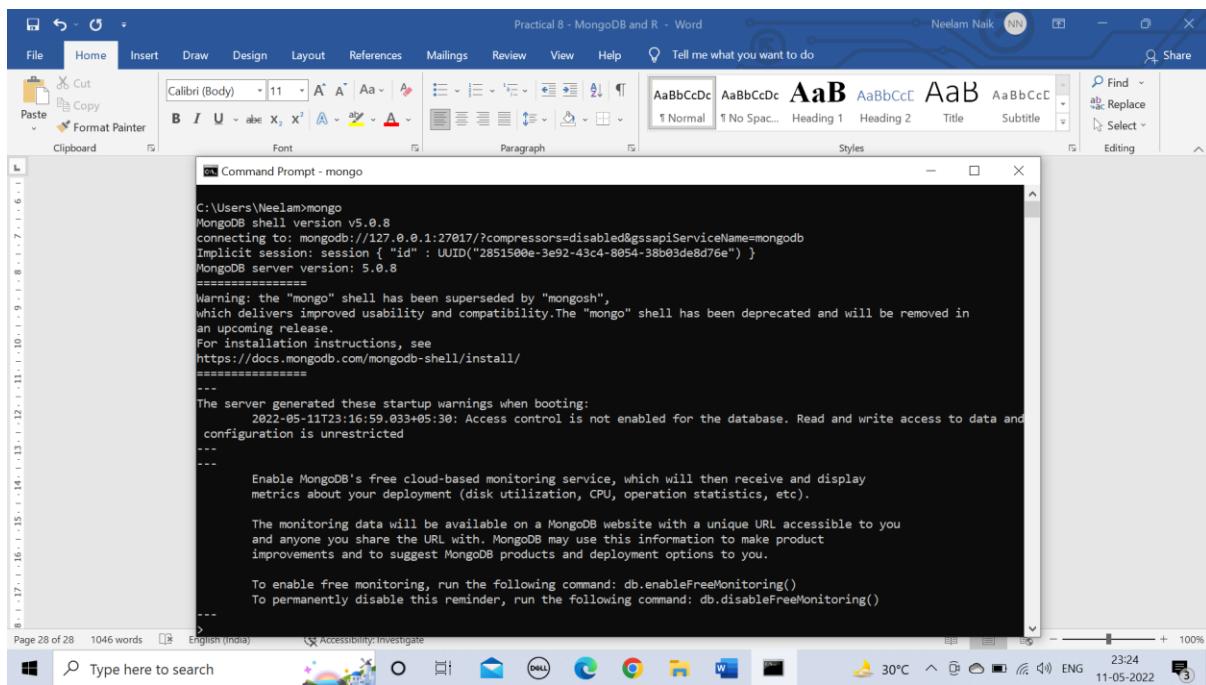
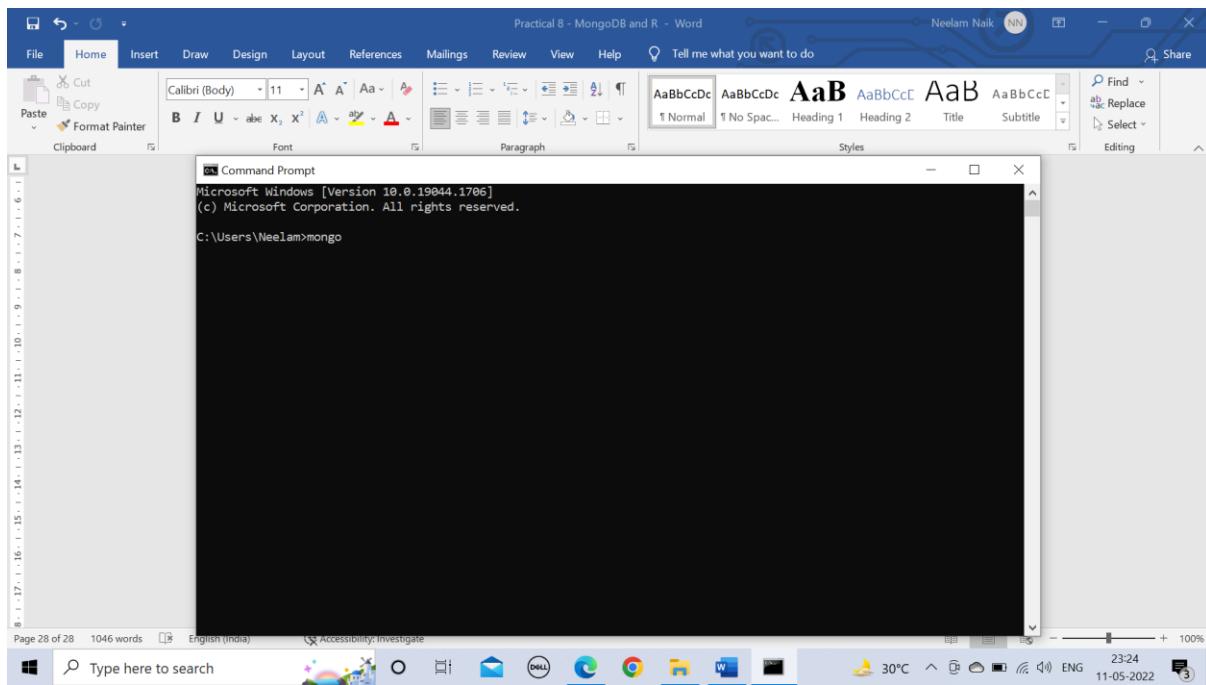
Restarted the machine

To check mongod service is running automatically:

Open command prompt

Give mongo command without running mongod command in another terminal.

It will start mongod server.



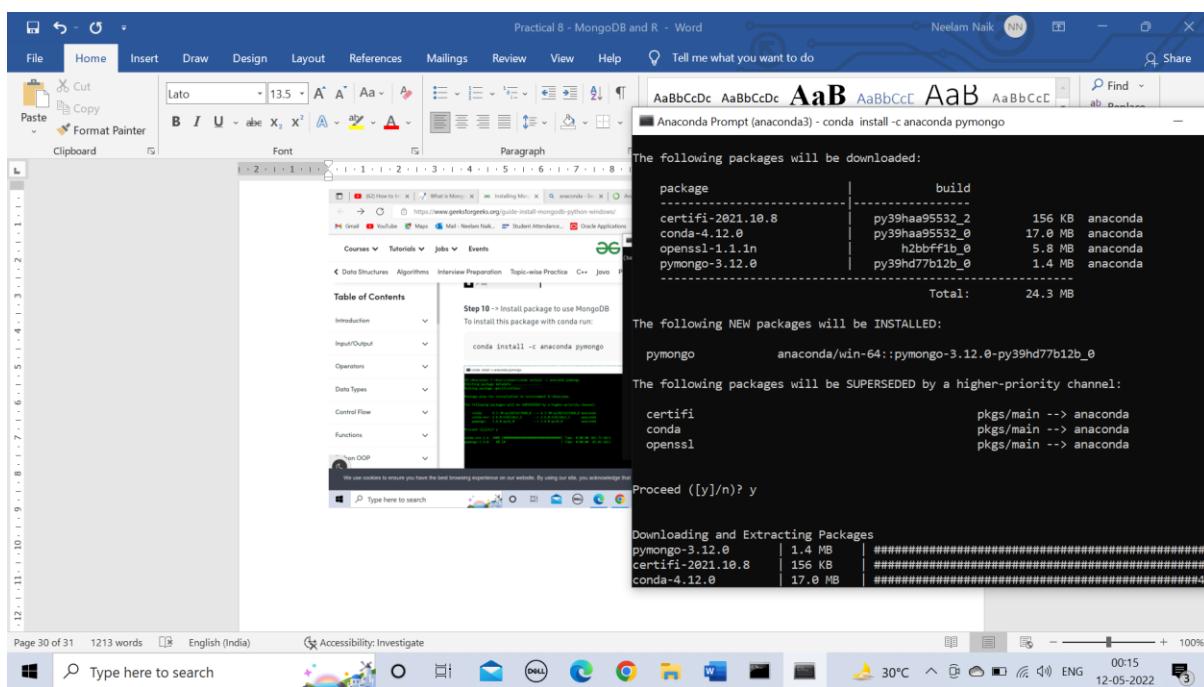
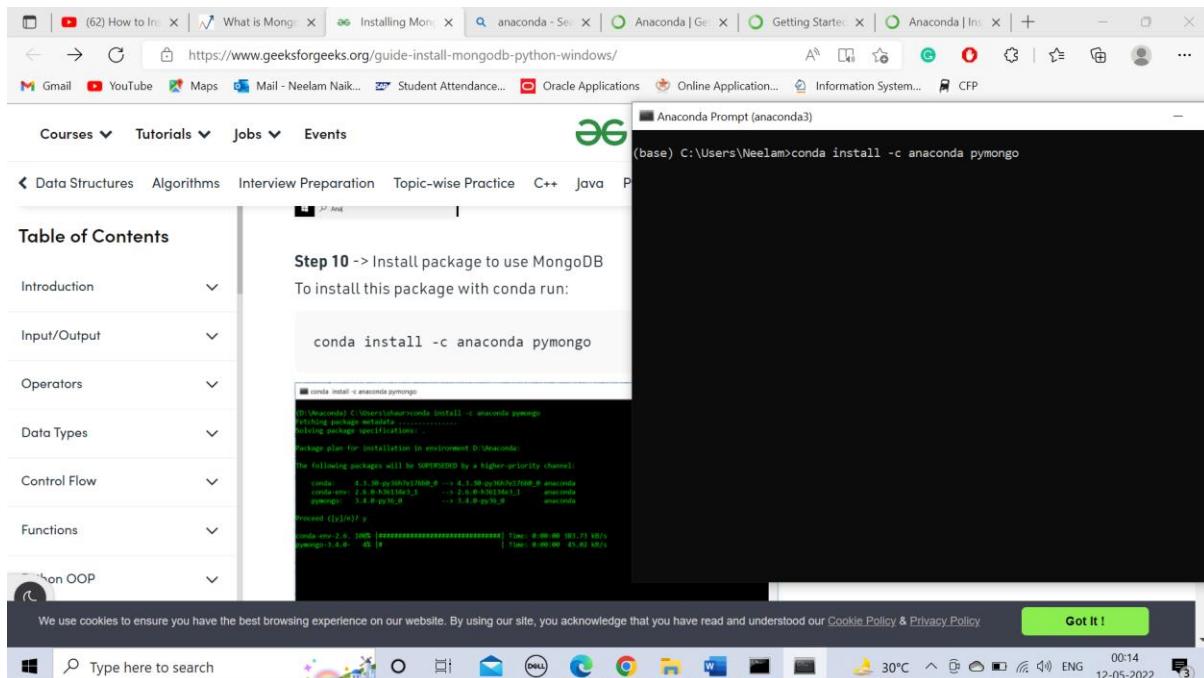
Step 4: Install MongoDB Python on Windows

We will be performing a few key basic operations on a MongoDB database in Python using the PyMongo library.

Install package to use MongoDB

To install this package with conda run:

```
conda install -c anaconda pymongo
```



Step 5: Verify MongoDB Python Connection

To retrieve the data from a MongoDB database, we will first connect to it. Write and execute the below code in your spider anaconda

```
import pymongo  
mongo_uri = "mongodb://localhost:27017/"  
client = pymongo.MongoClient(mongo_uri)
```

Let's see the available databases:

```
print(client.list_database_names())
```

We will use the my_database database for our purpose. Let's set the cursor to the same database:

```
db = client.my_database
```

connect to analysis database

The list_collection_names command shows the names of all the available collections:

```
print(db.list_collection_names())
```

Let's see the number of books we have. We will connect to the customers collection and then print the number of documents available in that collection:

```
table=db.books
```

```
print(table.count_documents({ })) #gives the number of documents in the table
```

Code:

```
import pymongo
```

```
mongo_uri = "mongodb://localhost:27017/"
```

```
client = pymongo.MongoClient(mongo_uri)
```

```
print(client.list_database_names())
```

```
db = client.my_database
```

```
print(db.list_collection_names())
```

```
table=db.books
```

```
zprint(table.count_documents({ }))
```

Output:

The screenshot shows the Spyder IDE interface with the following details:

- File Menu:** File Edit Search Source Run Debug Consoles Projects Tools View Help
- Toolbar:** Standard file operations like Open, Save, Print, etc.
- Code Editor:** The file `BDA_pract8.py` contains the following Python code:

```
1 import pymongo
2
3 mongo_uri = "mongodb://localhost:27017/"
4 client = pymongo.MongoClient(mongo_uri)
5 print(client.list_database_names())
6
7 db = client.my_database
8
9 print(db.list_collection_names())
10
11 table=db.books
12
13 print(table.count_documents({}))
```
- Console:** Shows the execution of the code. In [4] runs the file, and In [5] runs the `count_documents` method on the `books` collection.

```
In [4]: runfile('D:/A_UPG/MSc_Big Data Analytics/MSc_Big Data Analytics/2021-2022/Practicals/MongoDB and R/Program/BDA_pract8.py', wdir='D:/A_UPG/MSc_Big Data Analytics/MSc_Big Data Analytics/2021-2022/Practicals/MongoDB and R/Program')
['admin', 'config', 'local', 'my_database']
['books']
1

In [5]: table.count_documents({})
1
```
- Help:** A tooltip for the `Usage` button provides information on how to get help for objects.
- System Tray:** Shows the Windows taskbar with various pinned icons and system status like battery level, temperature (30°C), and date/time (12-05-2022).

References:

<https://www.analyticsvidhya.com/blog/2020/02/mongodb-in-python-tutorial-for-beginners-using-pymongo/>

https://www.youtube.com/watch?v=FwMwO8pXfq0&t=3s&ab_channel=ProgrammingKnowledge