




DEVELOPER'S PORTFOLIO

TASK - 2



Introduction about Developer's Portfolio

Developers build portfolio websites as full-stack developer sample projects to showcase their skills and impress clients. As a student or professional learning web development, you must practise making portfolio websites to gain knowledge and experience in efficient front end web development technology.



Campus Name :- Ajeenkya DY Patil University Pune
Name :- Devendra Singh
Batch :- Bachelor's In Computer Application (BCA)
Year :- 1st year
URN :- 2022-B-09072004C



About Task – 2

Do database modelling, create models and create various APIs

- Design schema for all the data to be stored
- Start MongoDB local server and point the backend to the server
- Define all the routes for the backend

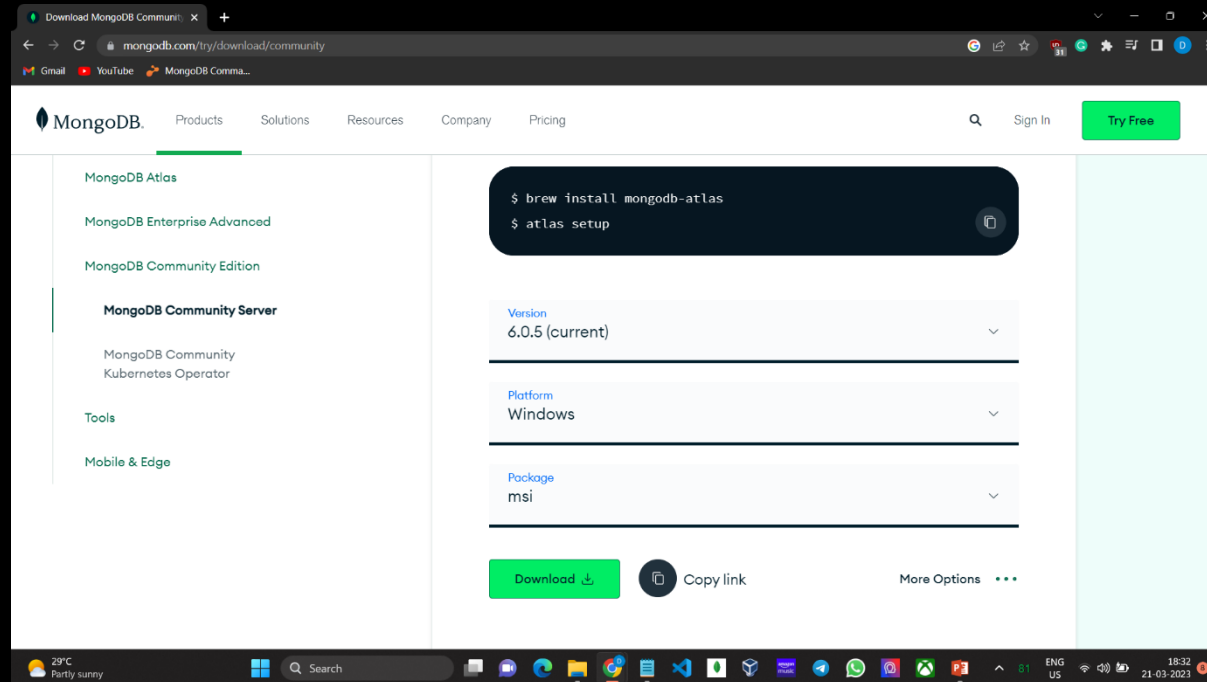


Introduction to MongoDB

MongoDB is an open source NoSQL database management program. NoSQL (Not only SQL) is used as an alternative to traditional relational databases. NoSQL databases are quite useful for working with large sets of distributed data. MongoDB is a tool that can manage document-oriented information, store or retrieve information.

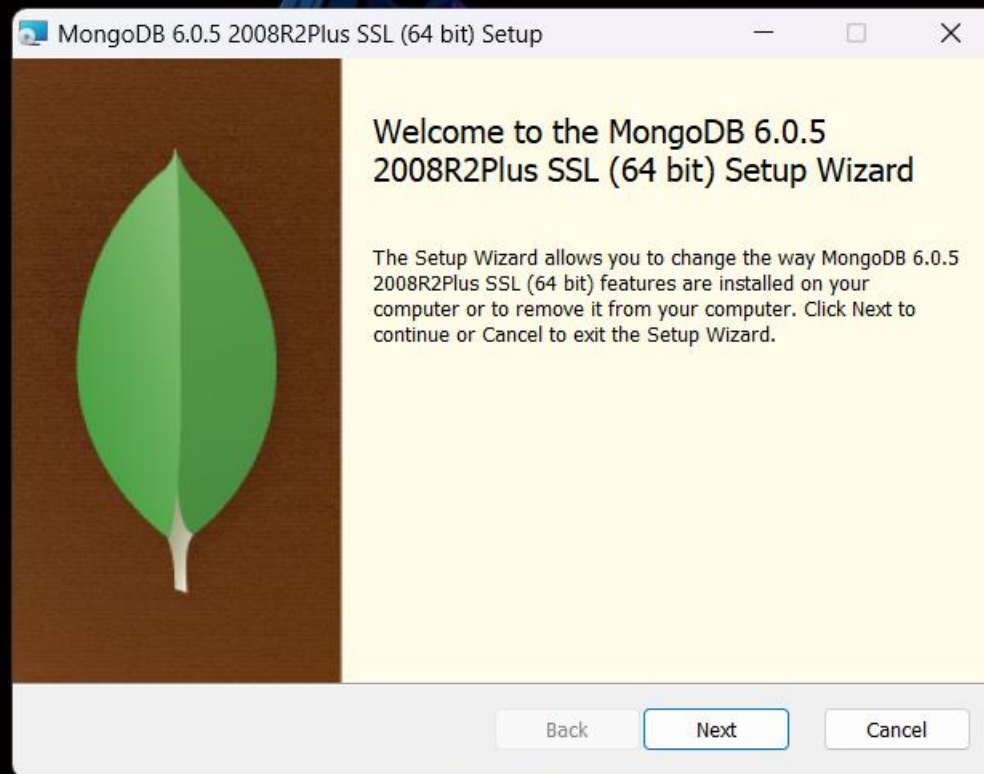
Steps for installing MongoDB

Step 1 – Go to this website - <https://www.mongodb.com/try/download/community>
Download the setup by clicking on download button
Open the MongoDB setup file.



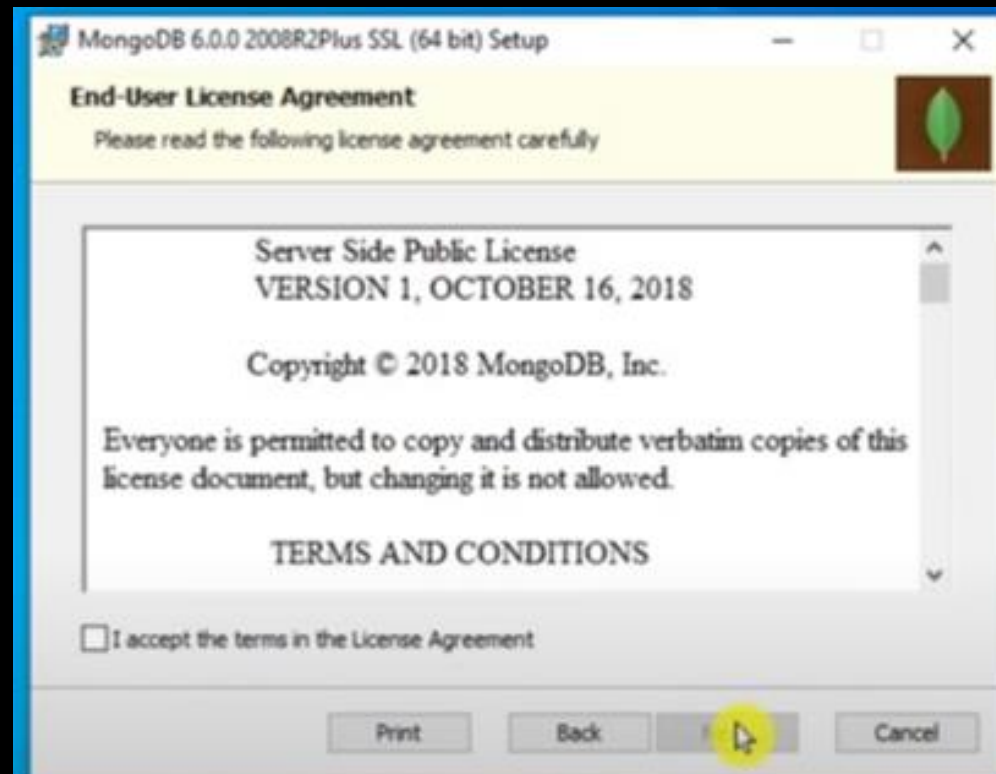
Steps for installing MongoDB

Step 2 – Installation window will pop up now click on “Next” button.



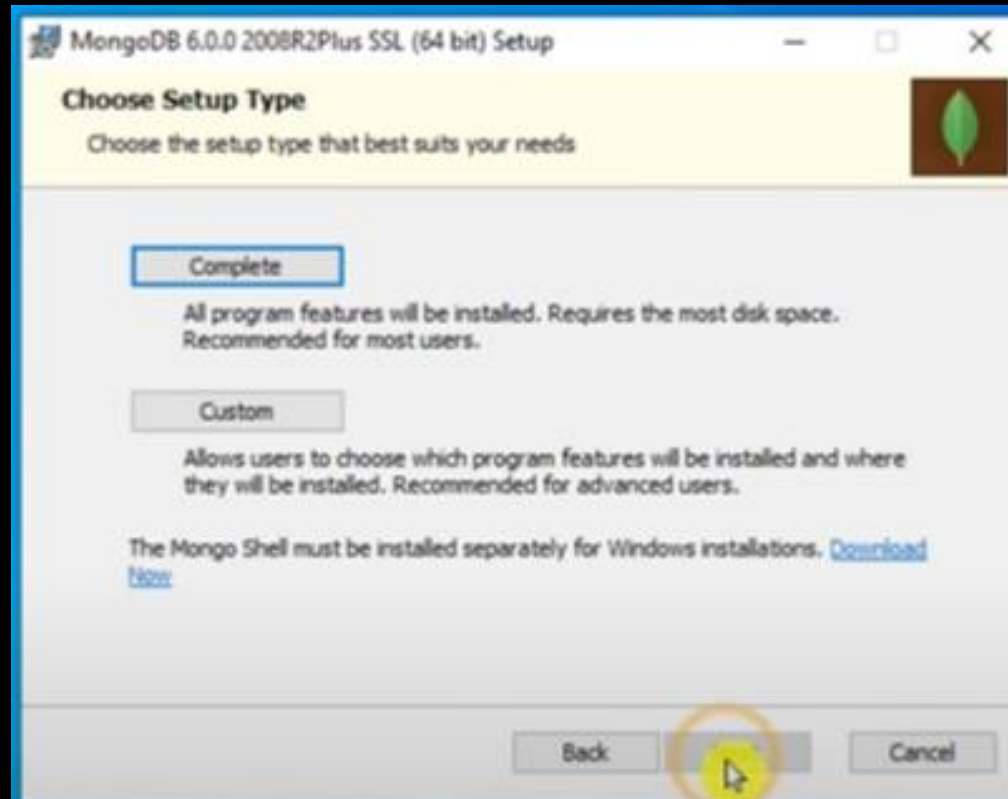
Steps for installing MongoDB

Step 3 - Now accept the “Terms” by clicking on checkbox and click on “Next”.



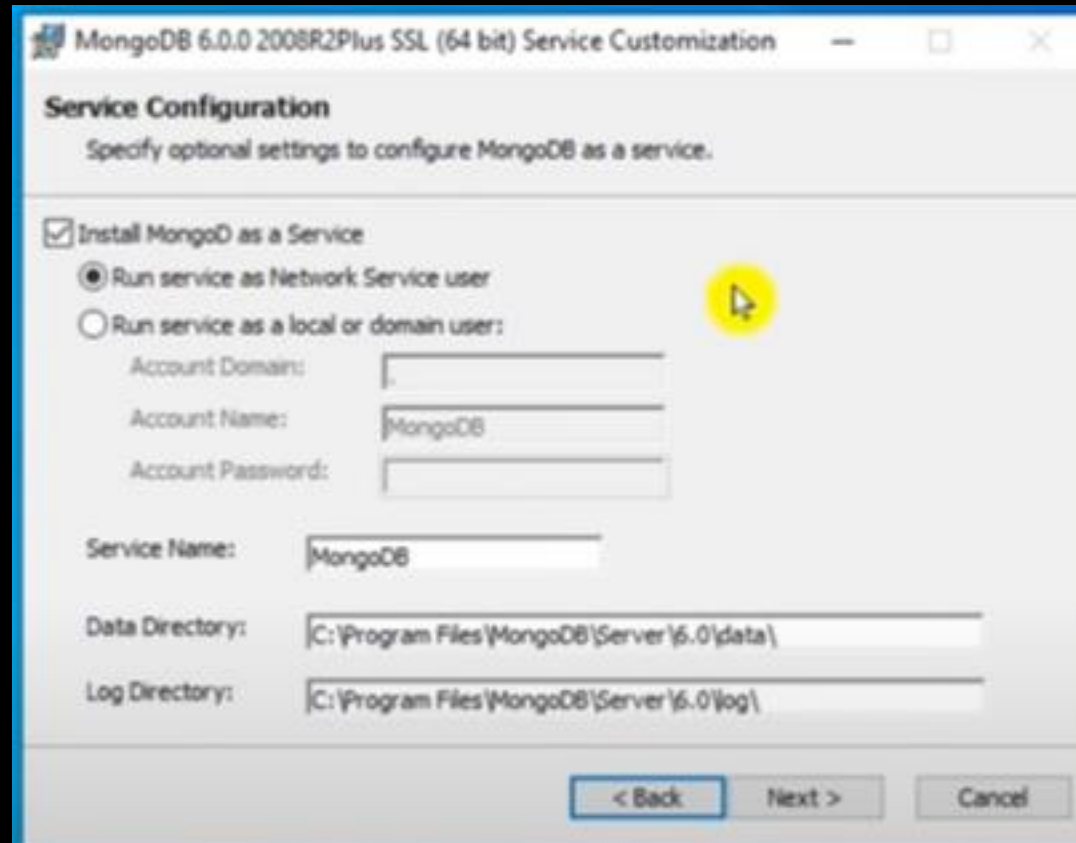
Steps for installing MongoDB

Step 4 - Now Select "Complete" Setup Type.



Steps for installing MongoDB

Step 5 - Now choose the “Directory” where you want to install MongoDB and click on “Next”.



MongoDB 6.0.0 2008R2Plus SSL (64 bit) Service Customization

Service Configuration
Specify optional settings to configure MongoDB as a service.

☒ Install MongoDB as a Service

☒ Run service as Network Service user

☐ Run service as a local or domain user:

Account Domain:

Account Name:

Account Password:

Service Name:

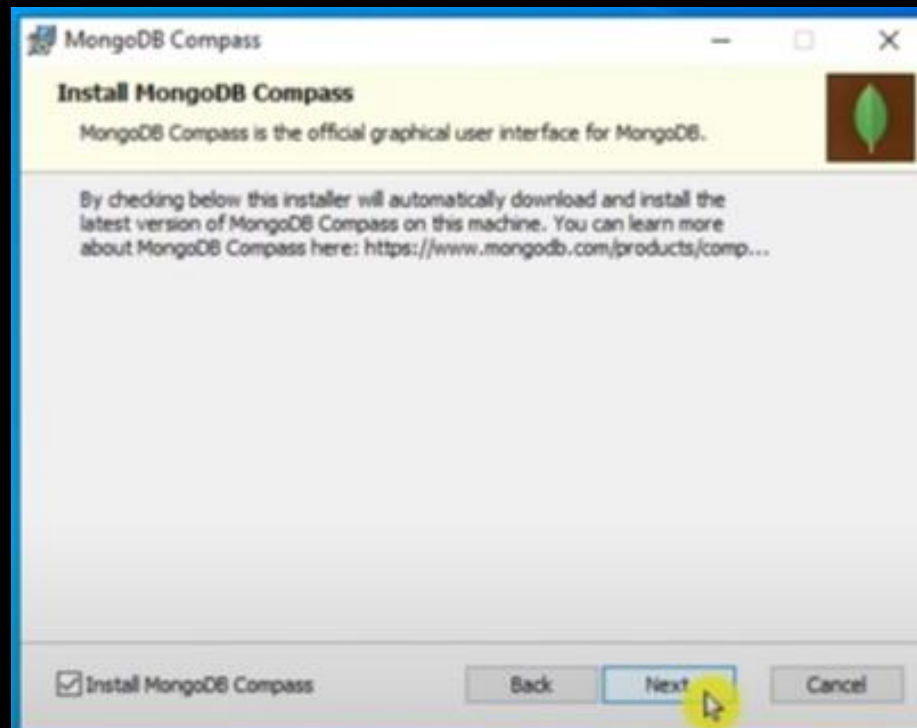
Data Directory:

Log Directory:

< Back Next > Cancel

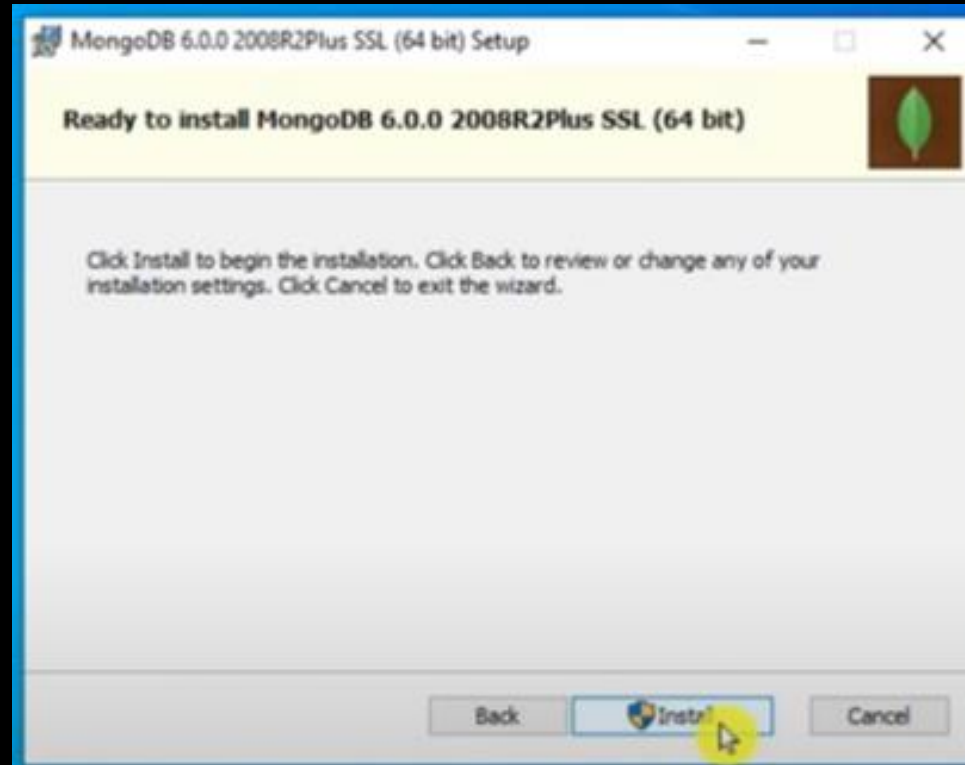
Steps for installing MongoDB

Step 6 - If you want to install “MongoDB Compass” then tick the checkbox or else click on “Next”.



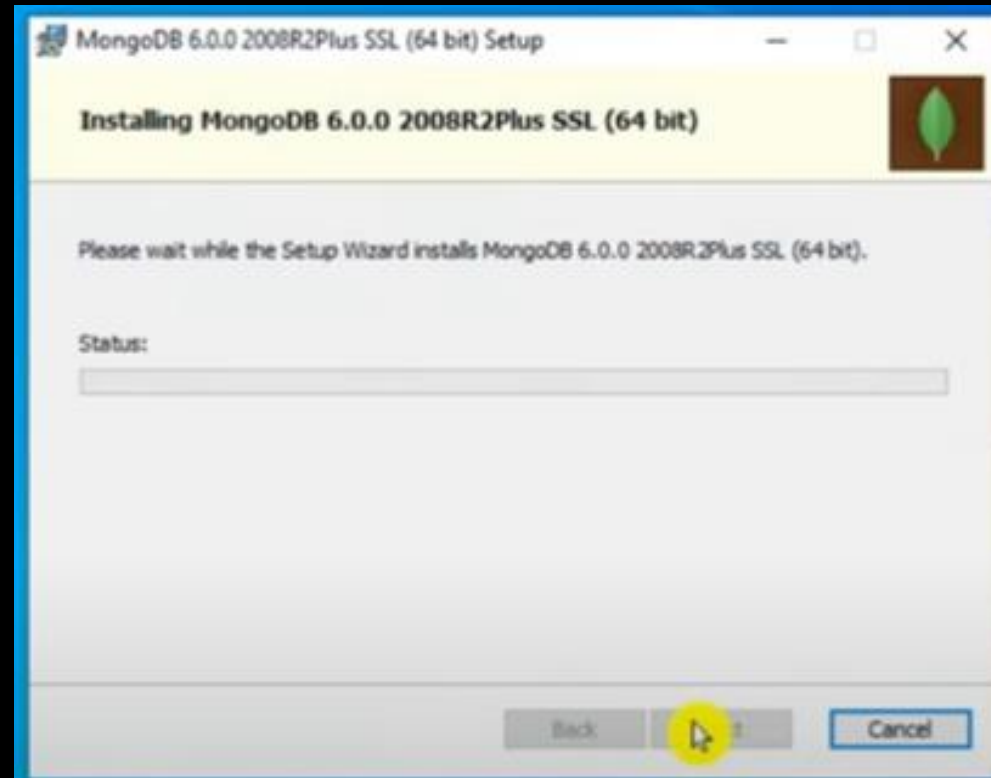
Steps for installing MongoDB

Step 7 - Now click on “Install”.



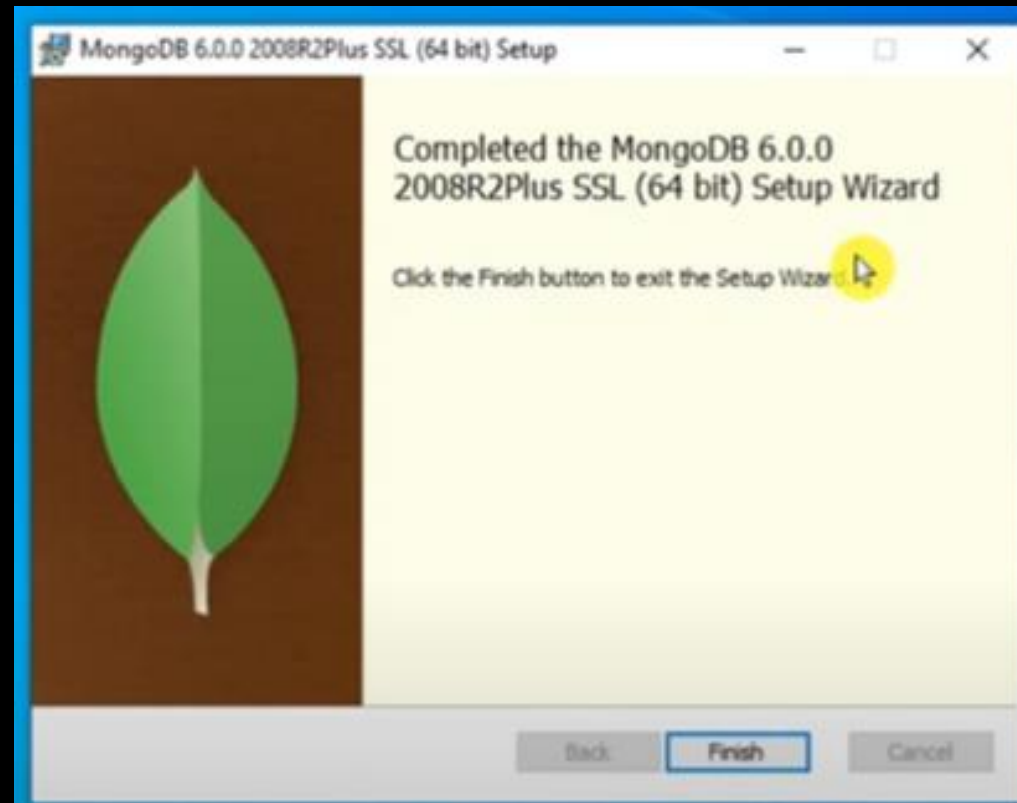
Steps for installing MongoDB

Step 8 - The installation process will start.



Steps for installing MongoDB

Step 9 – After installation click on “Finish”.

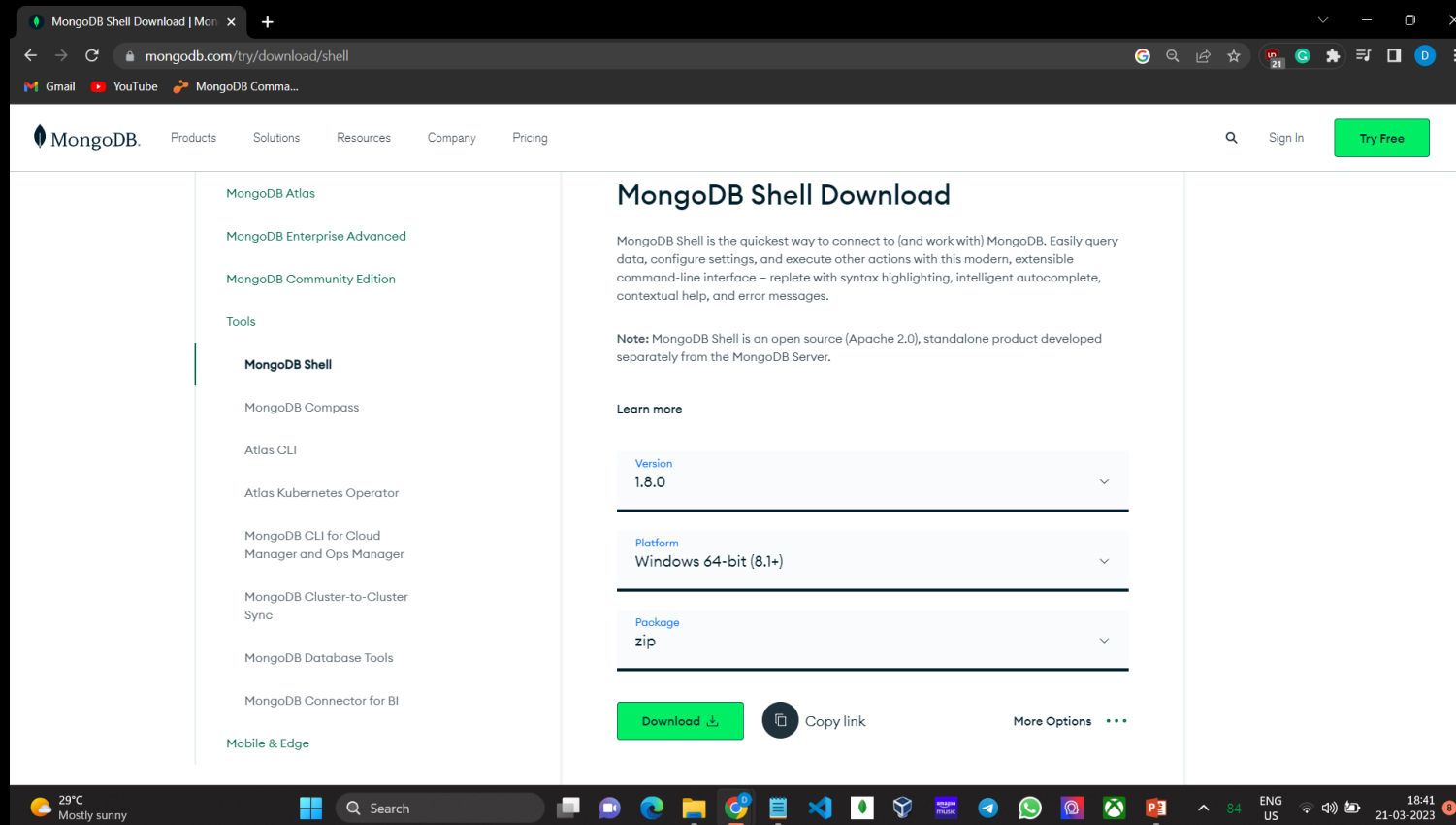




**MongoDB installation is done now we will install MongoDB
Server So that we can connect it to the server and use
MonogoDB Commands.**

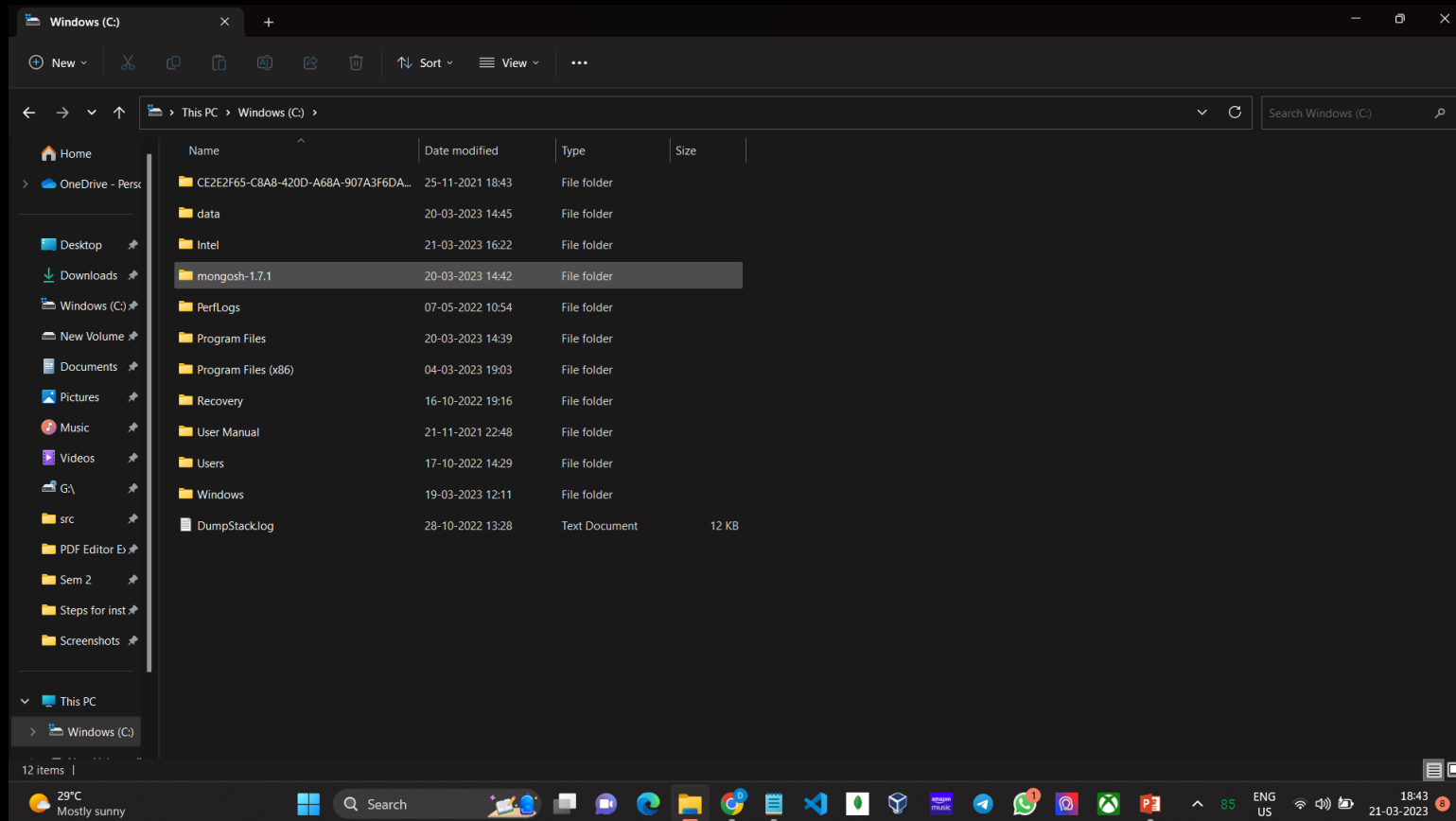
Steps for installing MongoDB Server

Step 1 - Go to this website - <https://www.mongodb.com/try/download/shell> and click on “Download” button



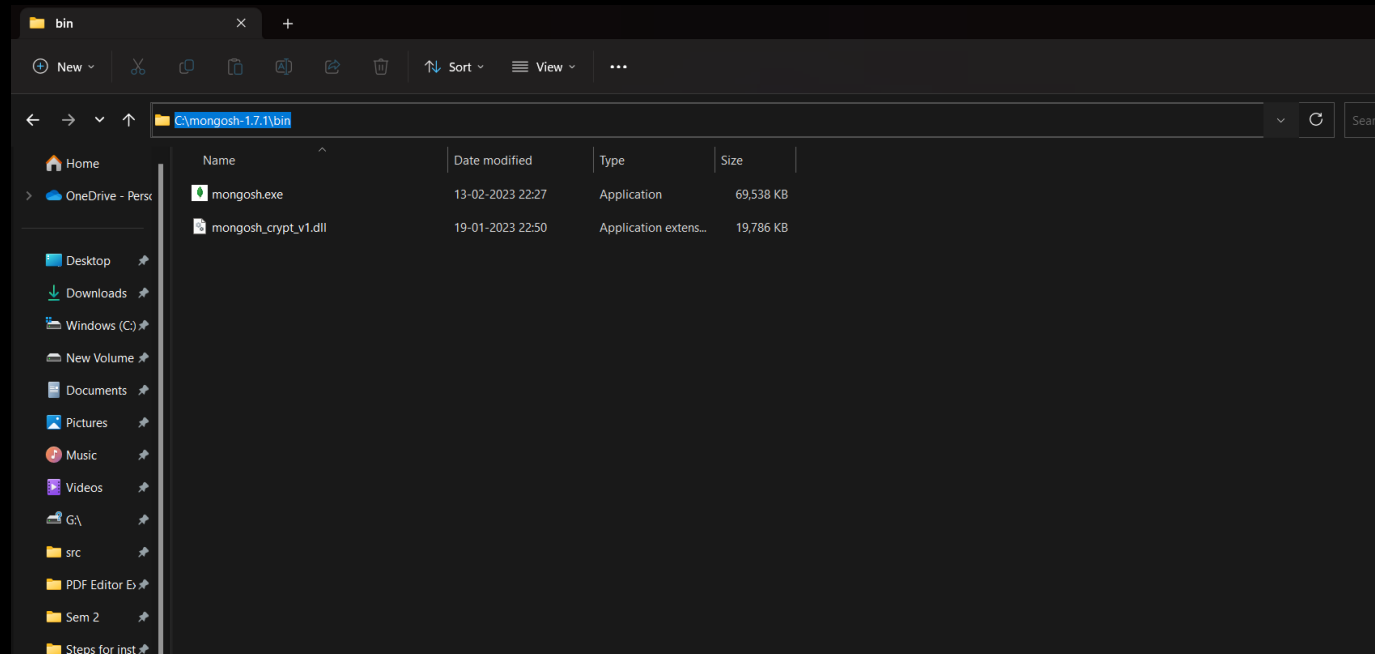
Steps for installing MongoDB Server

Step 2 - After downloading the file, extract it and paste it in “C” drive.



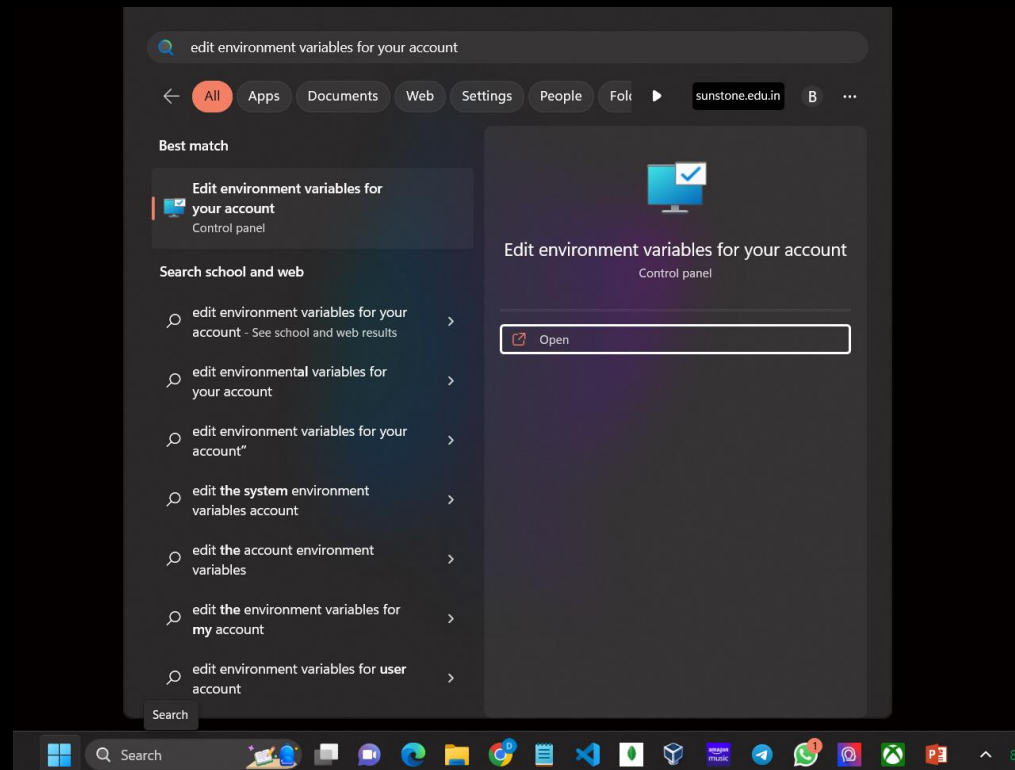
Steps for installing MongoDB Server

Step 3 - Now open the bin folder and copy the “path”.



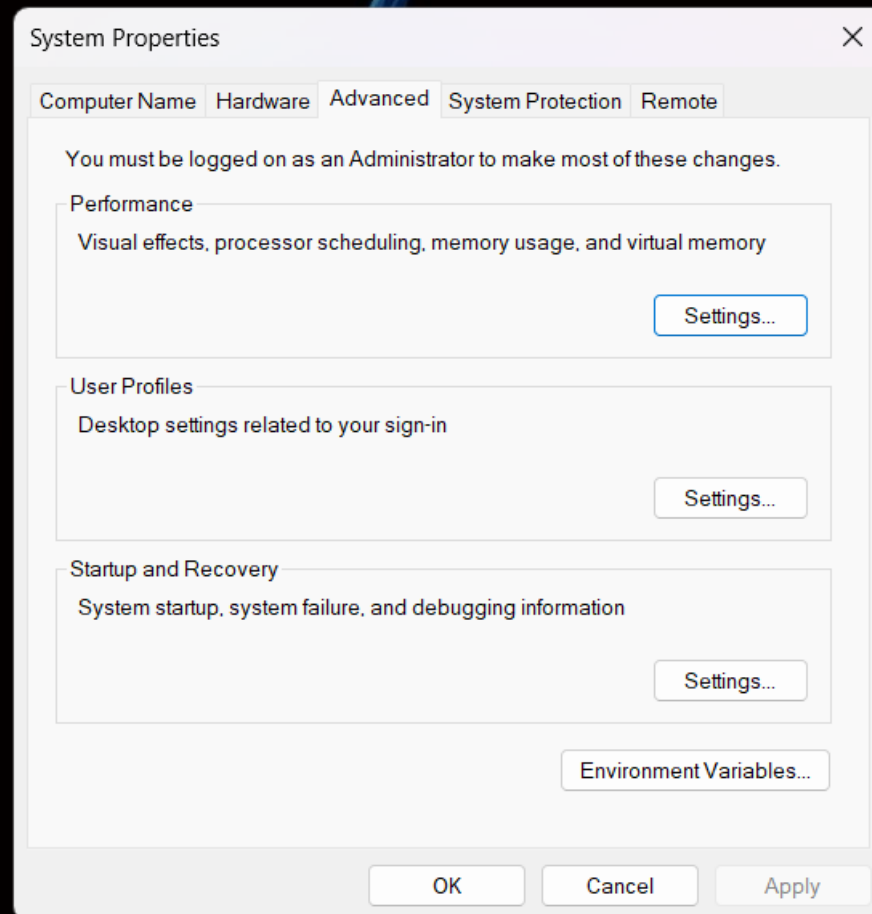
Steps for installing MongoDB Server

Step 4 - Now search for “Edit Environment variables for your account”.



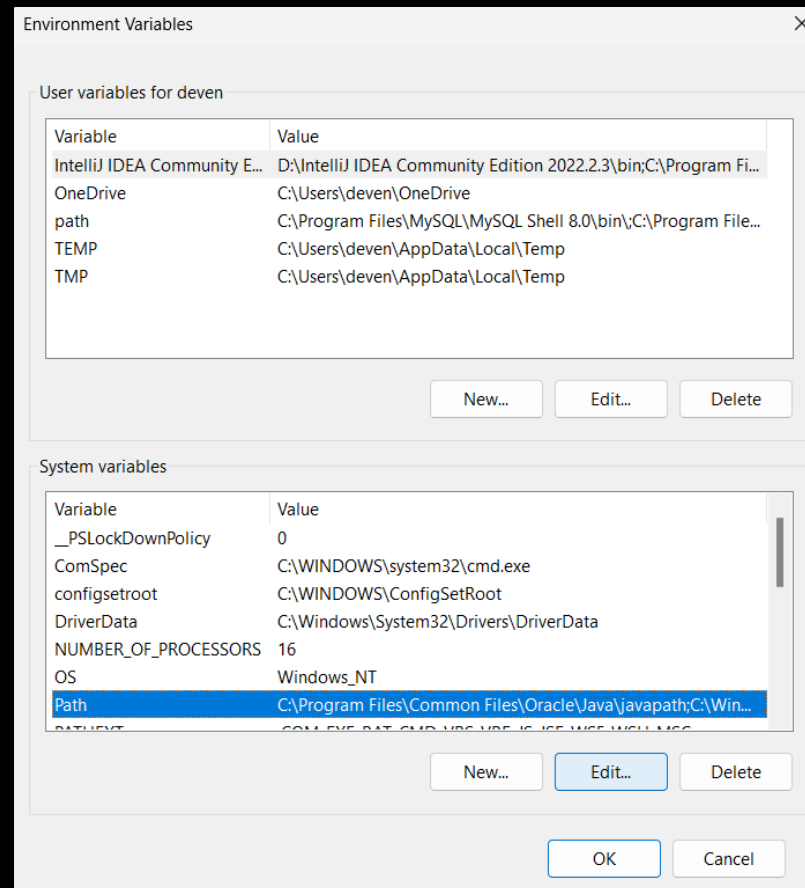
Steps for installing MongoDB Server

Step 5 - This window will pop up now click on “Environment Variables”.



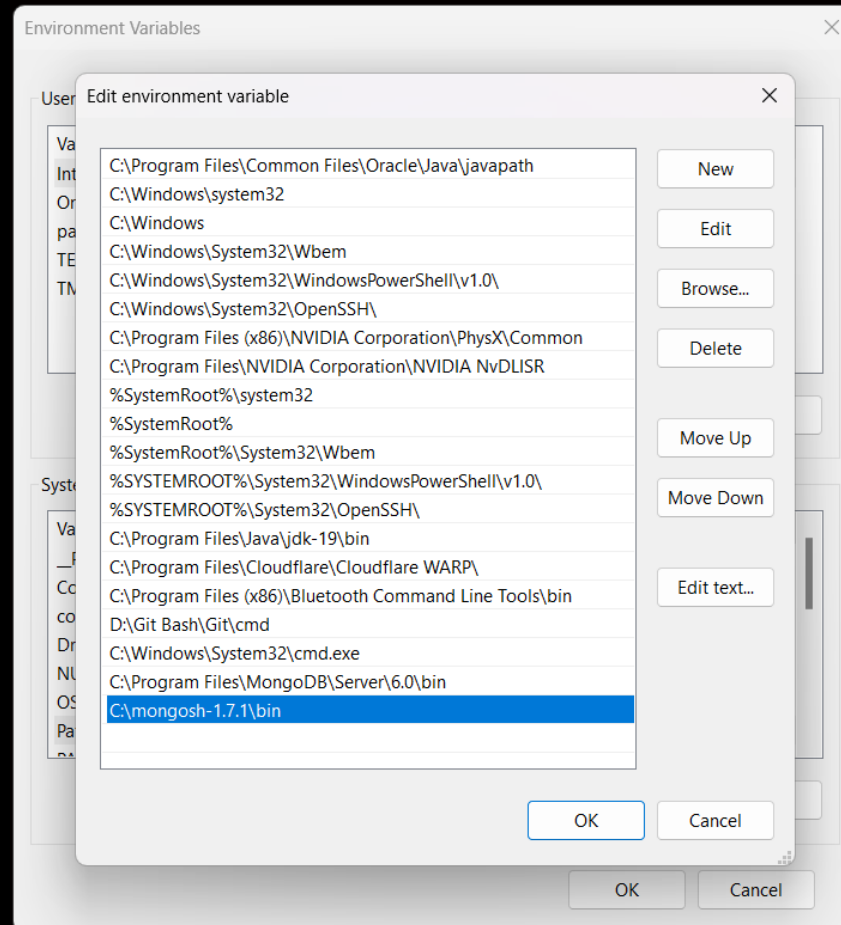
Steps for installing MongoDB Server

Step 6 - Now click on “Path” and “Edit” option.



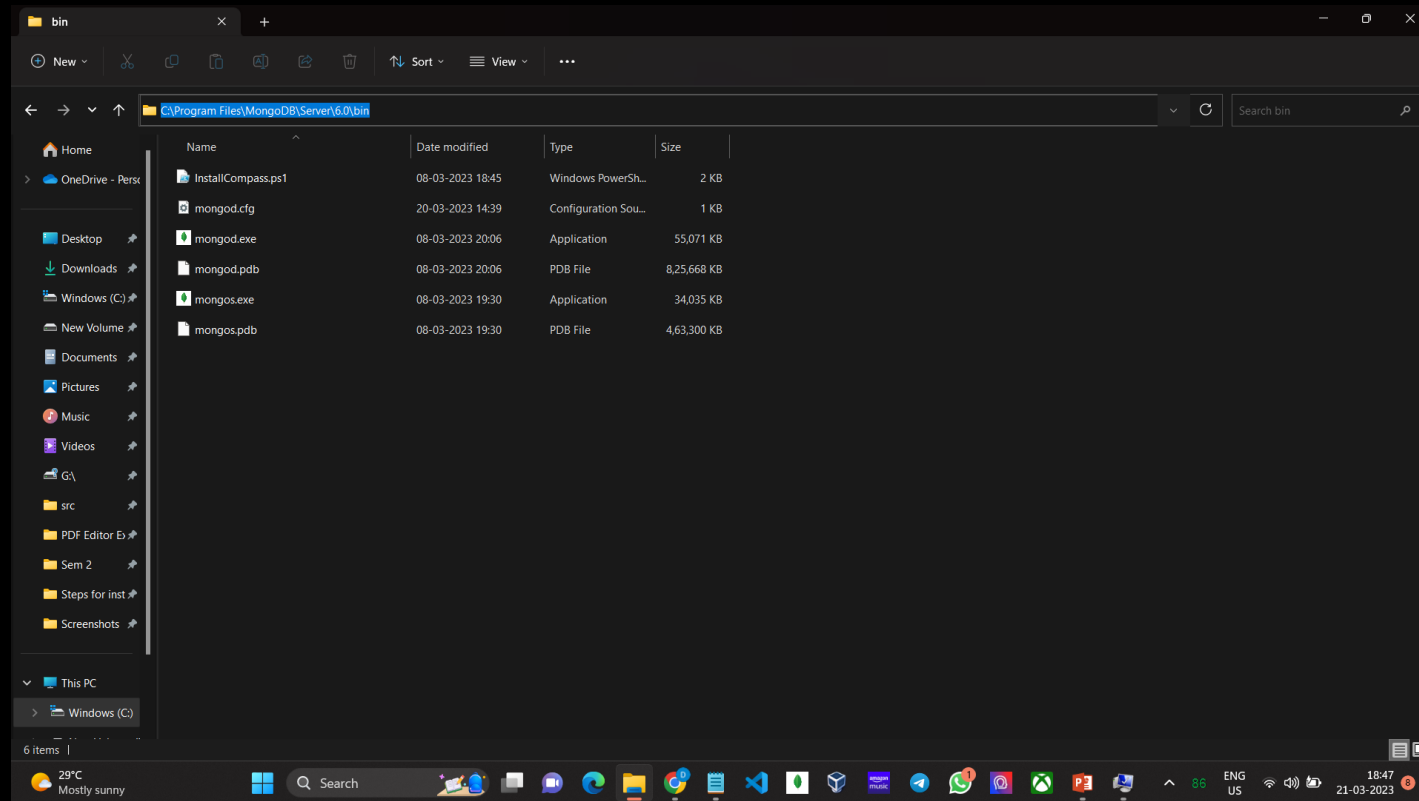
Steps for installing MongoDB Server

Step 7 - Now click on “New” option and paste the copied path and click on “OK”.



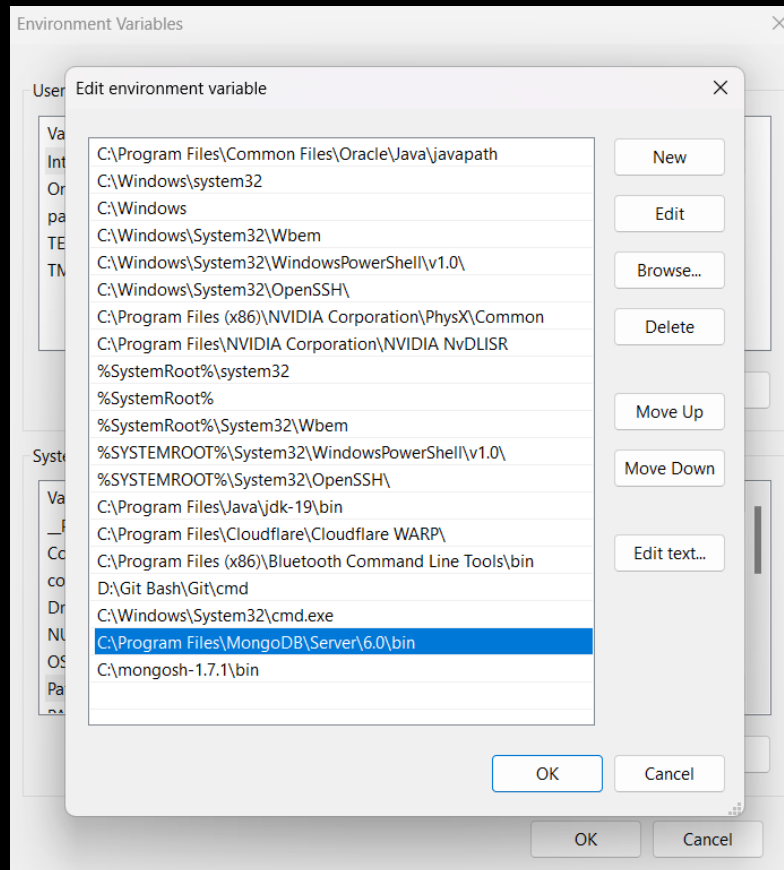
Steps for installing MongoDB Server

Step 8 - Now go to the directory where you installed MongoDB and copy its bin folder path.



Steps for installing MongoDB Server

Step 9 - Now again come to “Environment variables”, click on “New” and paste the path.



Steps for installing MongoDB Server

Step 10 - Now open "CMD" and type "Mongod --version" for checking whether the MongoDB is properly installed or not.

```
Command Prompt
Microsoft Windows [Version 10.0.22621.1413]
(c) Microsoft Corporation. All rights reserved.

C:\Users\deven>mongod --version
db version v6.0.5
Build Info: {
  "version": "6.0.5",
  "gitVersion": "c9a99c120371d4d4c52cbb15dac34a36ce8d3b1d",
  "modules": [],
  "allocator": "tcmalloc",
  "environment": {
    "distmod": "windows",
    "distarch": "x86_64",
    "target_arch": "x86_64"
  }
}
```

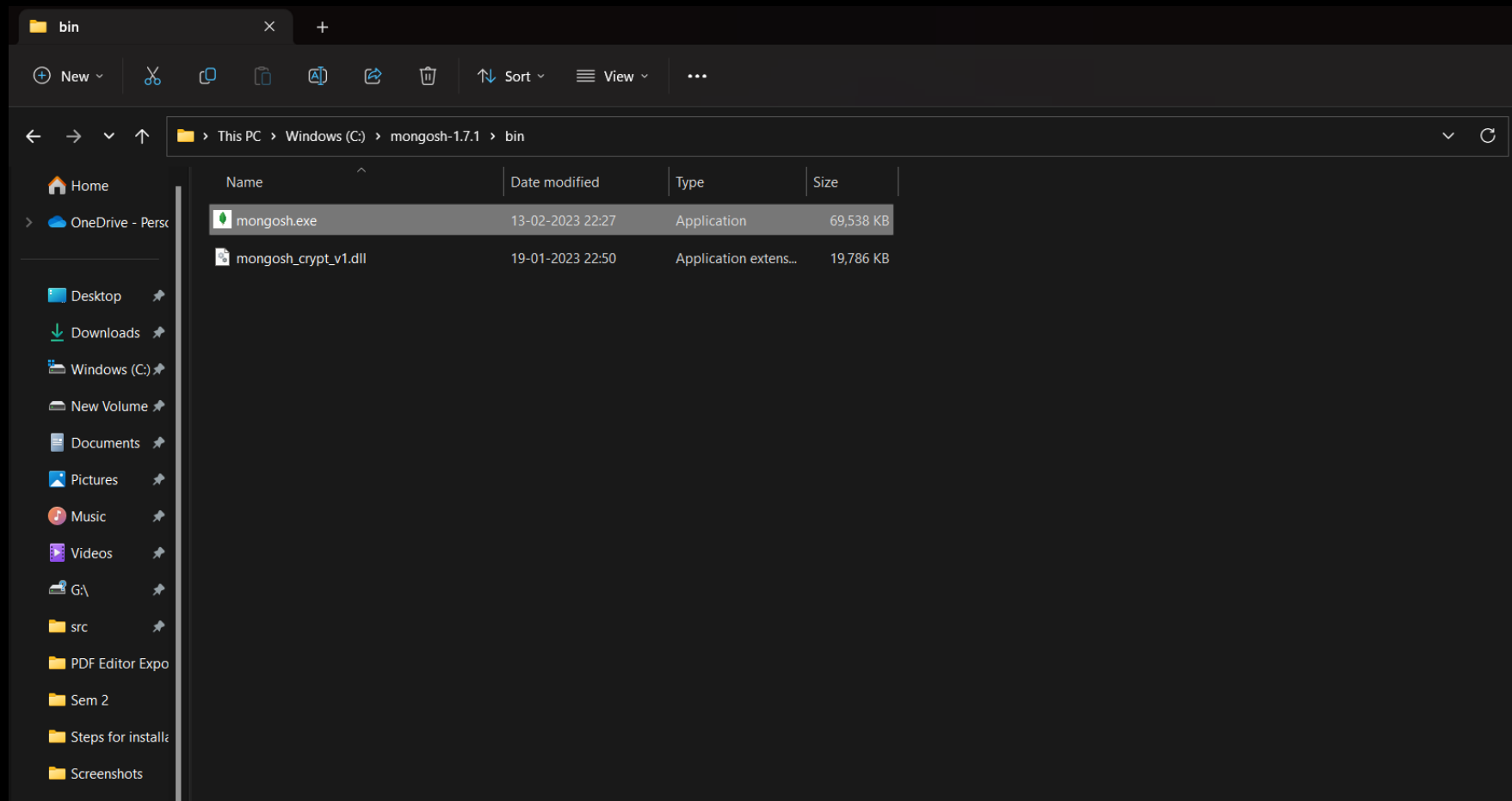
Steps for installing MongoDB Server

Step 11 - Now type “Mongod” and enter to start the server.

```
Command Prompt - mongod x + v
C:\Users\deven>mongod
{"t":{"$date":"2023-03-21T18:49:45.071+05:30"},"s":"I", "c":"CONTROL", "id":23285, "ctx":"","msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2023-03-21T18:49:45.071+05:30"},"s":"I", "c":"NETWORK", "id":4915701, "ctx":"","msg":"Initialized wire specification", "attr":{"spec":{"incomingExternalClient":{"minWireVersion":0,"maxWireVersion":17},"incomingInternalClient":{"minWireVersion":0,"maxWireVersion":17},"outgoing":{"minWireVersion":6,"maxWireVersion":17},"isInternalClient":true}}}}
{"t":{"$date":"2023-03-21T18:49:46.009+05:30"},"s":"I", "c":"NETWORK", "id":4648602, "ctx":"thread1", "msg":"Implicit TCP FastOpen in use."}
{"t":{"$date":"2023-03-21T18:49:46.011+05:30"},"s":"I", "c":"REPL", "id":5123008, "ctx":"thread1", "msg":"Successfully registered PrimaryOnlyService", "attr":{"service":"TenantMigrationDonorService","namespace":"config.tenantMigrationDonors"}}
{"t":{"$date":"2023-03-21T18:49:46.011+05:30"},"s":"I", "c":"REPL", "id":5123008, "ctx":"thread1", "msg":"Successfully registered PrimaryOnlyService", "attr":{"service":"TenantMigrationRecipientService","namespace":"config.tenantMigrationRecipients"}}
{"t":{"$date":"2023-03-21T18:49:46.011+05:30"},"s":"I", "c":"REPL", "id":5123008, "ctx":"thread1", "msg":"Successfully registered PrimaryOnlyService", "attr":{"service":"ShardSplitDonorService","namespace":"config.tenantSplitDonors"}}
{"t":{"$date":"2023-03-21T18:49:46.012+05:30"},"s":"I", "c":"CONTROL", "id":5945603, "ctx":"thread1", "msg":"Multi threading initialized"}
{"t":{"$date":"2023-03-21T18:49:46.013+05:30"},"s":"I", "c":"CONTROL", "id":4615611, "ctx":"initandlisten", "msg":"MongoDB starting", "attr":{"pid":10852,"port":27017,"dbPath":"C:/data/db/", "architecture":"64-bit", "host":"Dev"}}
{"t":{"$date":"2023-03-21T18:49:46.013+05:30"},"s":"I", "c":"CONTROL", "id":23398, "ctx":"initandlisten", "msg":"Target operating system minimum version", "attr":{"targetMinOS":"Windows 7/Windows Server 2008 R2"}}
{"t":{"$date":"2023-03-21T18:49:46.013+05:30"},"s":"I", "c":"CONTROL", "id":23403, "ctx":"initandlisten", "msg":"Build Info", "attr":{"buildInfo":{"version":"6.0.5","gitVersion":"c9a99c120371dd4dc52cbb15dac34a36ce8d3b1d","modules":[],"allocator":"tcmalloc","environment":{"distmod":"windows","distarch":"x86_64","target_arch":"x86_64"}}}}
{"t":{"$date":"2023-03-21T18:49:46.013+05:30"},"s":"I", "c":"CONTROL", "id":51765, "ctx":"initandlisten", "msg":"Operating System", "attr":{"os":{"name":"Microsoft Windows 10","version":"10.0 (build 22621)"}}}
{"t":{"$date":"2023-03-21T18:49:46.014+05:30"},"s":"I", "c":"CONTROL", "id":21951, "ctx":"initandlisten", "msg":"Options set by command line", "attr":{"options":{}}}
{"t":{"$date":"2023-03-21T18:49:46.023+05:30"},"s":"I", "c":"STORAGE", "id":22270, "ctx":"initandlisten", "msg":"Storage engine to use detected by data files", "attr":{"dbpath":"C:/data/db/", "storageEngine":"WiredTiger"}}
{"t":{"$date":"2023-03-21T18:49:46.024+05:30"},"s":"I", "c":"STORAGE", "id":22315, "ctx":"initandlisten", "msg":"Opening WiredTiger", "attr":{"config":"create,cache_size=7530M,session_max=33000,eviction=(threads_min=4,threads_max=4),config_base=false,statistics=(fast),log=(enabled=true,remove=true,path=journal,compressor=snappy),builtin_extension_config=(zstd=(compression_level=6)),file_manager=(close_idle_time=600,close_scan_interval=10,close_handle_minimum=2000),statistics_log=(wait=0),json_output=(error,message),verbose=[recovery_progress:1,checkpoint_progress:1,compact_progress:1,backup:0,checkpoint:0,compact:0,evict:0,history_store:0,recovery:0,fts:0,salvage:0,tiered:0,timestamp:0,transaction:0,verify:0,log:0]"}
{"t":{"$date":"2023-03-21T18:49:46.400+05:30"},"s":"I", "c":"STORAGE", "id":4795906, "ctx":"initandlisten", "msg":"WiredTiger opened", "attr":{"durationMillis":376}}
{"t":{"$date":"2023-03-21T18:49:46.400+05:30"},"s":"I", "c":"RECOVERY", "id":23987, "ctx":"initandlisten", "msg":"WiredTiger recoveryTimestamp", "attr":{"recoveryTimestamp":{"timestamp":{"t":0,"i":0}}}}
{"t":{"$date":"2023-03-21T18:49:46.427+05:30"},"s":"W", "c":"CONTROL", "id":22120, "ctx":"initandlisten", "msg":"Access control is not enabled for the database. Read and write access to data and configuration is unrestricted", "tags":["startupWarnings"]}
{"t":{"$date":"2023-03-21T18:49:46.427+05:30"},"s":"W", "c":"CONTROL", "id":22140, "ctx":"initandlisten", "msg":"This server is bound to localhost. Remote systems will be unable to connect to this server. Start the server with --bind_ip <address> to specify which IP addresses it should serve responses from, or with --bind_ip_all to bind to all interfaces. If this behavior is desired, start the server with --bind_ip 127.0.0.1 to disable this warning", "tags":["startupWarnings"]}
{"t":{"$date":"2023-03-21T18:49:46.440+05:30"},"s":"I", "c":"NETWORK", "id":4915702, "ctx":"initandlisten", "msg":"Updated wire specification", "attr":{"oldSpec":{"incomingExternalClient":{"minWireVersion":0,"maxWireVersion":17},"incomingInternalClient":{"minWireVersion":0,"maxWireVersion":17},"outgoing":{"minWireVersion":6,"maxWireVersion":17},"isInternalClient":true},"newSpec":{"incomingExternalClient":{"minWireVersion":0,"maxWireVersion":17},"incomingInternalClient":{"minWireVersion":17,"maxWireVersion":17},"outgoing":{"minWireVersion":17,"maxWireVersion":17},"isInternalClient":true}}}}
{"t":{"$date":"2023-03-21T18:49:46.441+05:30"},"s":"I", "c":"REPL", "id":5853300, "ctx":"initandlisten", "msg":"current featureCompatibilityVersion value", "attr":{"featureCompatibilityVersion":"6.0","context":"startup"}}
{"t":{"$date":"2023-03-21T18:49:46.441+05:30"},"s":"I", "c":"STORAGE", "id":5071100, "ctx":"initandlisten", "msg":"Clearing temp directory"}
{"t":{"$date":"2023-03-21T18:49:46.442+05:30"},"s":"I", "c":"CONTROL", "id":20536, "ctx":"initandlisten", "msg":"Flow Control is enabled on this deployment"}
{"t":{"$date":"2023-03-21T18:49:46.720+05:30"},"s":"I", "c":"FTDC", "id":20625, "ctx":"initandlisten", "msg":"Initializing full-time diagnostic data capture", "attr":{"dataDirectory":"C:/data/db/diagnostic.data"}}
{"t":{"$date":"2023-03-21T18:49:46.732+05:30"},"s":"I", "c":"REPL", "id":6015317, "ctx":"initandlisten", "msg":"Setting new configuration state", "attr":{"newState":"ConfigReplicationDisabled","oldState":"ConfigPreStart"}}
{"t":{"$date":"2023-03-21T18:49:46.732+05:30"},"s":"I", "c":"STORAGE", "id":22262, "ctx":"initandlisten", "msg":"Timestamp monitor starting"}
{"t":{"$date":"2023-03-21T18:49:46.735+05:30"},"s":"I", "c":"NETWORK", "id":23015, "ctx":"listener", "msg":"Listening on", "attr":{"address":"127.0.0.1"}}
{"t":{"$date":"2023-03-21T18:49:46.736+05:30"},"s":"I", "c":"NETWORK", "id":23016, "ctx":"listener", "msg":"Waiting for connections", "attr":{"port":27017,"ssl":"off"}}
```

Steps for installing MongoDB Server

Step 12 - After starting server you can open mongosh.exe.



Steps for installing MongoDB Server

Step 13 - Now after mongosh.exe opens press “Enter” to use the MongoDB Database.

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.7.1
Please enter a MongoDB connection string (Default: mongodb://localhost/):

Current Mongosh Log ID: 6419af54c38410abbda70dfd
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.7.1
Using MongoDB:      6.0.5
Using Mongosh:      1.7.1

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

-----
The server generated these startup warnings when booting
2023-03-21T16:22:45.292+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
-----

-----
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.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
-----

test> |
```


Now After Installation of MongoDB lets start with few Commands

1. To show all the Databases.

Use → **show dbs**

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.7.1
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.7.1
Using MongoDB:      6.0.5
Using Mongosh:      1.7.1

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

-----
The server generated these startup warnings when booting
2023-03-21T16:22:45.292+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
-----

-----
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.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
-----

test> show dbs
Devendra 72.00 KiB
admin    40.00 KiB
config   60.00 KiB
local    72.00 KiB
test>
```


Now After Installation of MongoDB lets start with few Commands

2. To create or use existing Database.

Use → **use database_name**

```
mongosh mongodb://127.0.0.1 × + v
test> show dbs
Devendra      8.00 KiB
admin         40.00 KiB
config        108.00 KiB
local         72.00 KiB
test> use Devendra
switched to db Devendra
Devendra> |
```

Now After Installation of MongoDB lets start with few Commands

3. To create Collection.

Use → **db.createCollection("collection_name")**

```
test> show dbs
Devendra  72.00 KiB
admin      40.00 KiB
config    60.00 KiB
local      72.00 KiB
test> use Devendra
switched to db Devendra
Devendra> db.createCollection("Emp")
{ ok: 1 }
Devendra> |
```

Now After Installation of MongoDB lets start with few Commands

4. To show all Collections in Database.

Use → **show collections** OR **db.getCollectionNames()**

```
test> show dbs
Devendra 72.00 KiB
admin     40.00 KiB
config    60.00 KiB
local     72.00 KiB
test> use Devendra
switched to db Devendra
Devendra> db.createCollection("Emp")
{ ok: 1 }
Devendra> show collections
Emp
Devendra> db.getCollectionNames()
[ 'Emp' ]
Devendra> |
```

Now After Installation of MongoDB lets start with few Commands

5. To insert Document in Collection.

Use →

For single Document → **db.collection_name.insert({field1:'value1',field2:'value2'})**

For Multiple Document → **db.collection_name.insertMany([{field1:"value1"},{field2:"value2"}])**

```
test> show dbs
Devendra 72.00 KiB
admin     40.00 KiB
config    60.00 KiB
local     72.00 KiB
test> use Devendra
switched to db Devendra
Devendra> db.createCollection("Emp")
{ ok: 1 }
Devendra> show collections
Emp
Devendra> db.getCollectionNames()
[ 'Emp' ]
Devendra> db.Emp.insert({Name:'Devendra Singh',Address:'Pune'})
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("641a9ad0404033f62ddd54fb") }
}
Devendra> db.Emp.insertMany([{Name:'Rohan',Address:'Delhi'},{Name:'Ashok',Address:'Mumbai'}])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("641a9b19404033f62ddd54fc"),
    '1': ObjectId("641a9b19404033f62ddd54fd")
  }
}
Devendra> |
```

Now After Installation of MongoDB lets start with few Commands

6. To get the list of Documents in Collection.

Use → **db.collection_name.find()**

```
mongosh mongodb://127.0.0.1:27021/ > use test
test> db.createCollection('Emp')
test> db.Emp.insertMany([
  { Name: 'Devendra Singh', Address: 'Pune' },
  { Name: 'Rohan', Address: 'Delhi' },
  { Name: 'Ashok', Address: 'Mumbai' }
])
test> db.Emp.find()
{
  "_id" : ObjectId("641a9ad0404033f62ddd54fb"),
  "Name" : "Devendra Singh",
  "Address" : "Pune"
},
{
  "_id" : ObjectId("641a9b19404033f62ddd54fc"),
  "Name" : "Rohan",
  "Address" : "Delhi"
},
{
  "_id" : ObjectId("641a9b19404033f62ddd54fd"),
  "Name" : "Ashok",
  "Address" : "Mumbai"
}
]
```

Now After Installation of MongoDB lets start with few Commands

7. To update the Document in Collection.

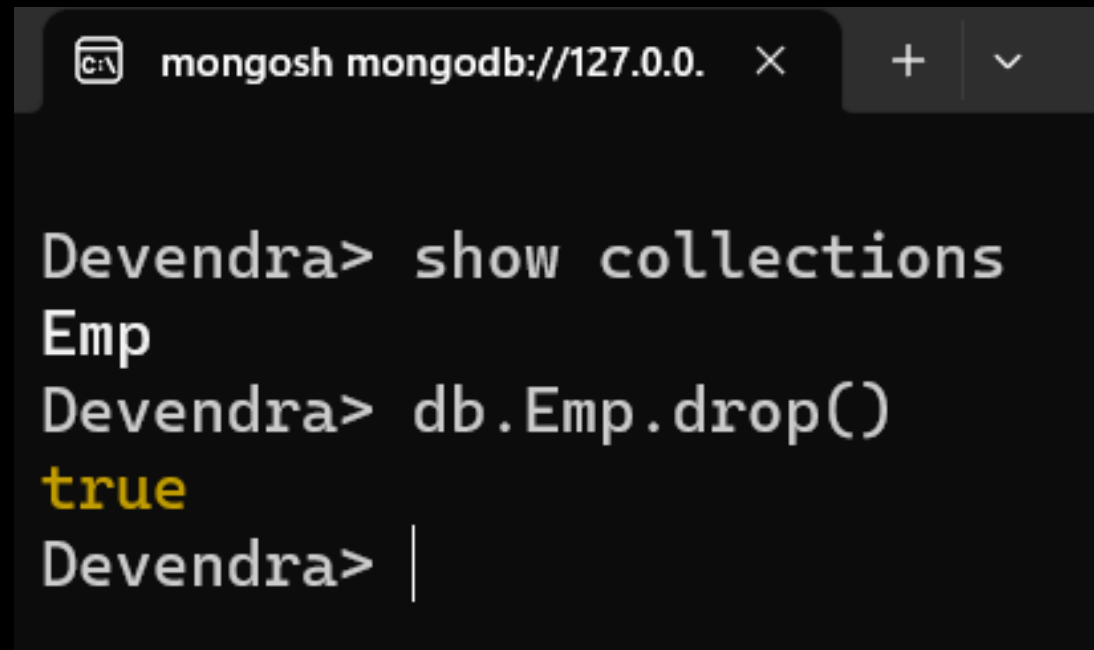
Use → **db.collection_name.update(selection_criteria,updated_data)**

```
{
  _id: ObjectId("641a9b19404033f62ddd54fc"),
  Name: 'Rohan',
  Address: 'Delhi'
},
{
  _id: ObjectId("641a9b19404033f62ddd54fd"),
  Name: 'Ashok',
  Address: 'Mumbai'
}
}
Devendra> db.Emp.update({'Name': 'Rohan'}, {$set: {'Name': 'Parth'}})
DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Devendra> db.Emp.find().pretty()
[
  {
    _id: ObjectId("641a9ad0404033f62ddd54fb"),
    Name: 'Devendra Singh',
    Address: 'Pune'
  },
  {
    _id: ObjectId("641a9b19404033f62ddd54fc"),
    Name: 'Parth',
    Address: 'Delhi'
  },
]
```

Now After Installation of MongoDB lets start with few Commands

8. To Delete or Drop Collection.

Use → **db.collection_name.drop()**

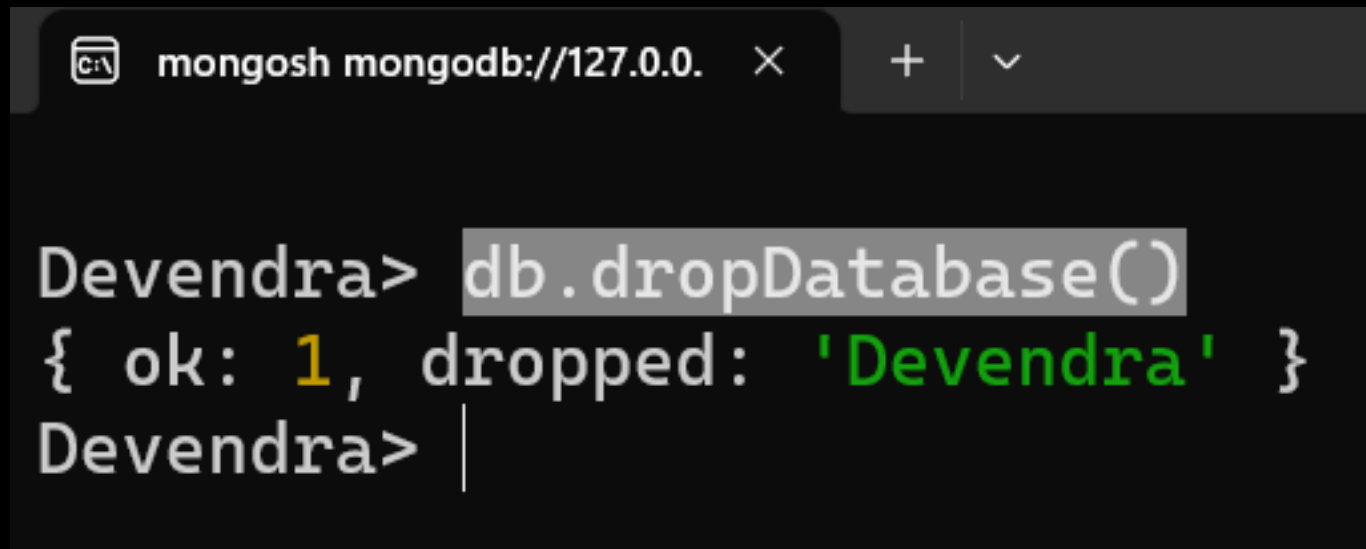


```
mongosh mongodb://127.0.0.1
Devendra> show collections
Emp
Devendra> db.Emp.drop()
true
Devendra> |
```


Now After Installation of MongoDB lets start with few Commands

9. To Delete or Drop Database.

Use → **db.dropDatabase()**



```
mongosh mongodb://127.0.0.1
Devendra> db.dropDatabase()
{ ok: 1, dropped: 'Devendra' }
Devendra> |
```



SUMMARY

- We learned about how to install MongoDB and MongoDB Server.
- We learned about how to set up MongoDB server and how to start it.
- We learned about few commands to work with MongoDB.



My Git Hub Link

<https://github.com/DevendraSingh7465>

The background features a solid black field. At the top, there is a decorative, wavy, translucent shape that transitions through a color gradient from yellow and orange on the left to green and blue on the right.

Thank You