What is MongoDB?

MongoDB is a popular NoSQL database system that stores data in a flexible document format instead of rigid tables with rows and columns, like traditional relational databases.

Key Features of MongoDB:

Flexible Document storage:

- Unlike rigid tables, MongoDB stores data in JSON_like doucuments.
- This allows for more natural modelling of complex data structures and easier adaptation as data evolves.

Scalability for Growing Data:

- As your application grows and the amount of data you store increases, MongoDB can easily scale horizontally by adding more servers.

High Performance:

- Optimized for fast reads and writes, MongoDB excels in real-time applications where data needs to be accessed and updated quickly.

NoSQL database:

- Offers advantages over relational databases for certain use cases, like ease of development and flexibility.

Developer-Friendly:

- The use of JSON-like documents and a familiar query language (similar to SQL) makes MongoDB easy to learn and use for developers.

MongoDB is a popular choice for various applications :

- **Web applications:** Storing user data, content, and preferences.

- **Mobile apps:** Managing user data and application state on the backend.
- **Real-time analytics:** Storing and analyzing data streams from sensors or applications.
- **Content management systems:** Storing and managing content in a flexible way.

How to Install MongoDB on Windows?

Requirements to Install MongoDB on Windows:

- MongoDB versions 4.4 and later only support 64-bit versions of Windows.
- If you're using an older version (e.g., MongoDB 7.0), supported Windows versions include:

  Windows Server 2022 (64-bit)

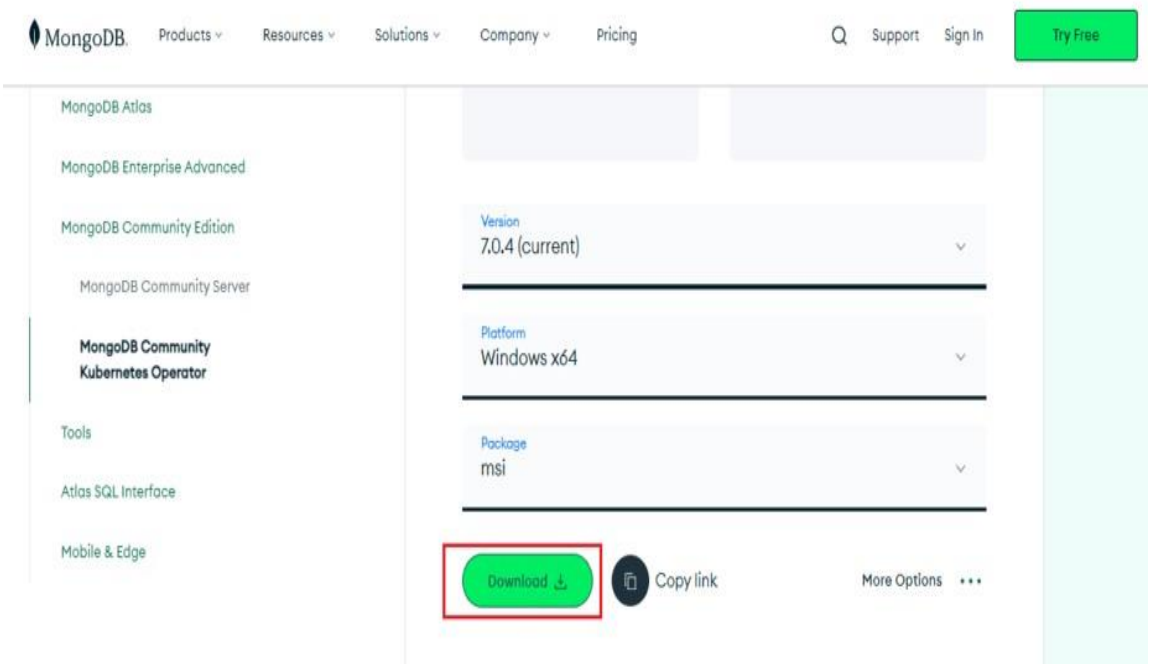
  Windows Server 2019 (64-bit)

  Windows 11 (64-bit)

Steps to Install MongoDB on windows using MSI:

To install MongoDB on Windows, first, download the MongoDB server and then install the MongoDB shell.

STEP1: Go to the MongoDB Download Center to download the MongoDB Community Server.

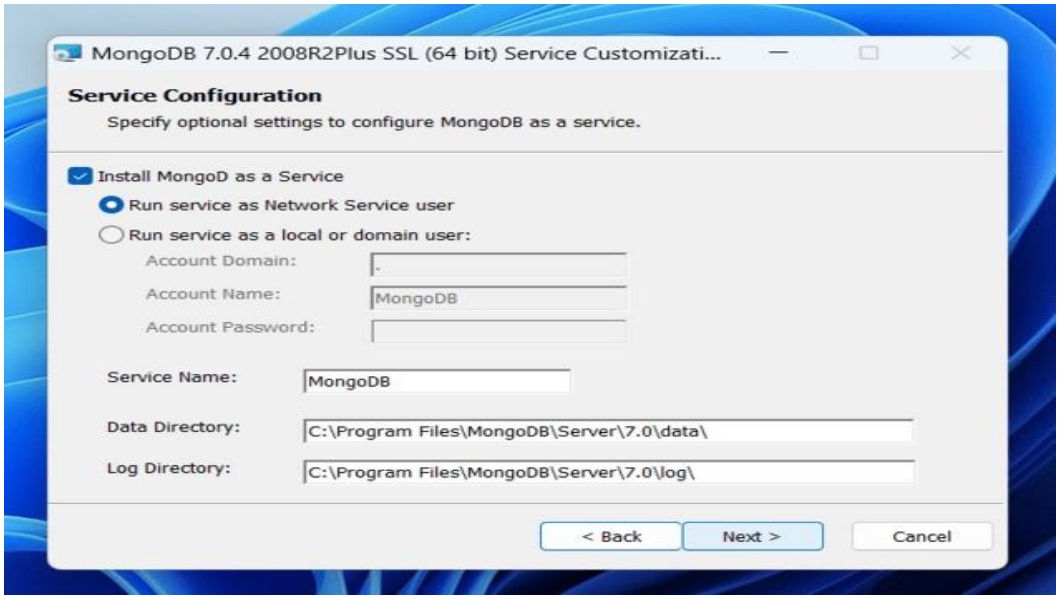


select any version, Windows, and package according to your requirement. For Windows, we need to choose:

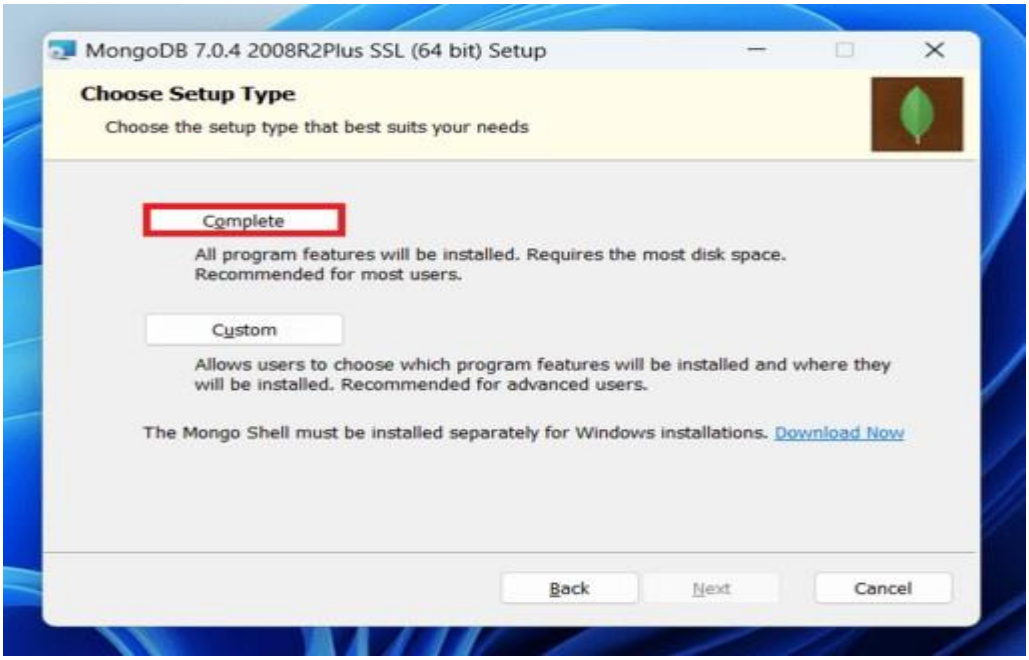
Version: 7.0.4
OS: Windows x64
Package: msi

Step2: When the download is complete open the msi file and click the next button in the startup screen:

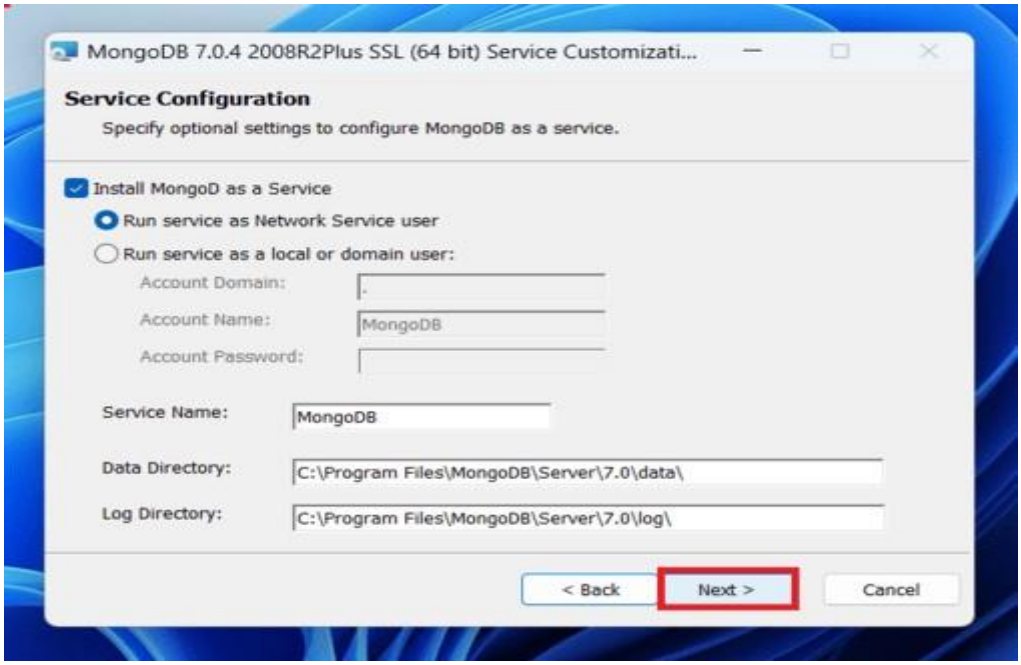
Step3: Now accept the End-User License Agreement and click the next button:



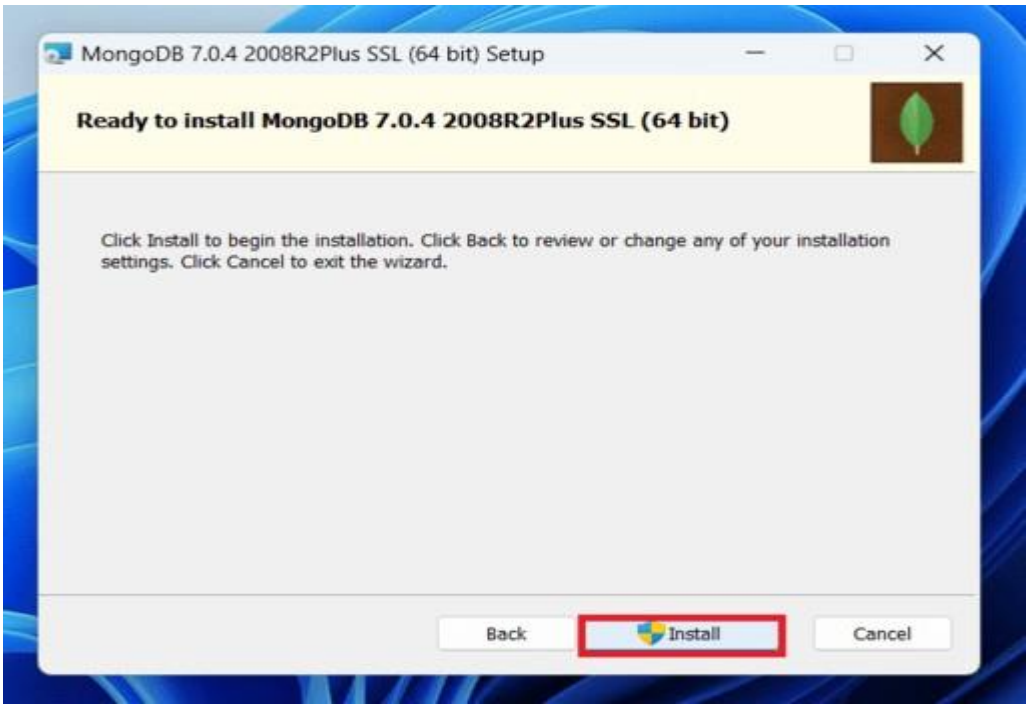
Step4: Now select the complete option to install all the program features. Here, if you can want to install only selected program features and want to select the location of the installation, then use the Custom option:



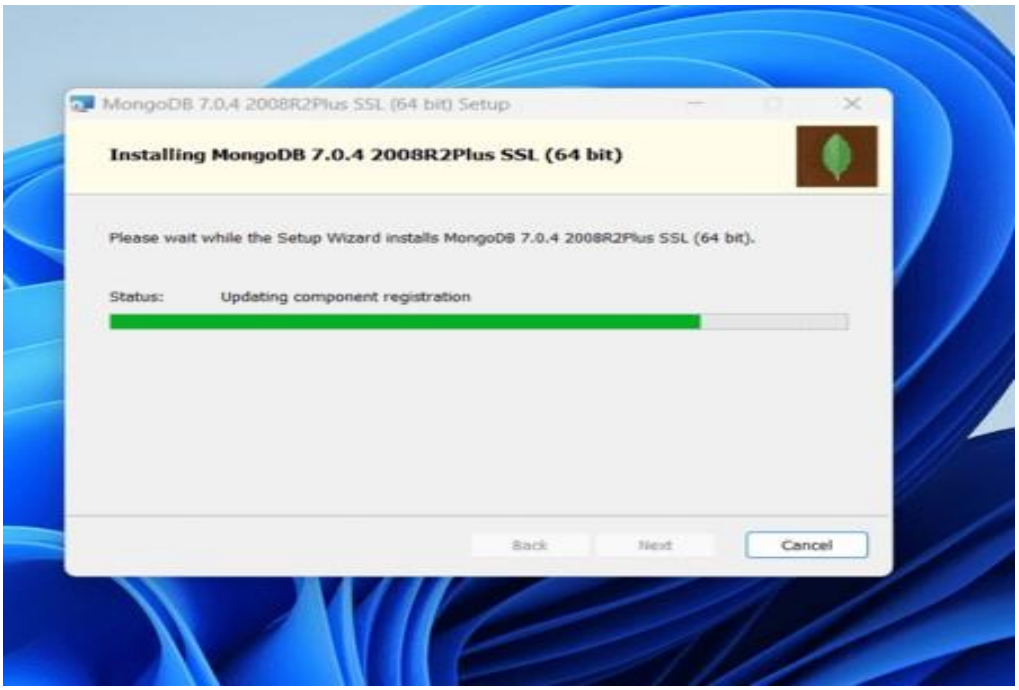
Step5: Select "Run service as Network Service user" and copy the path of the data directory. Click Next:



Step6: Click the Install button to start the MongoDB installation process:



Step:7: After clicking on the install button installation of MongoDB begins:



Step 8: Now click the Finish button to complete the MongoDB installation process:

MongoDB Mongo shell Features:

MongoDB Mongo shell is the default client for the MongoDB database server. It's a command-line interface (CLI), where the input and output are all console-based. The Mongo shell is a good tool to manipulate small sets of data.

Here are the top features that Mongo shell offers:

- Run all MongoDB queries from the Mongo shell.
- Manipulate data and perform administration operations.
- Mongo shell uses JavaScript and a related API to issue commands.
- See previous commands in the mongo shell with up and down arrow keys.
- View possible command completions using the tab button after partially entering a command.
- Print error messages, so you know what went wrong with your commands.

MongoDB has recently introduced a new mongo shell known as **mongosh**. It has some additional features, such as extensibility and embeddability—that is, the ability to use it inside other products such as VS Code.

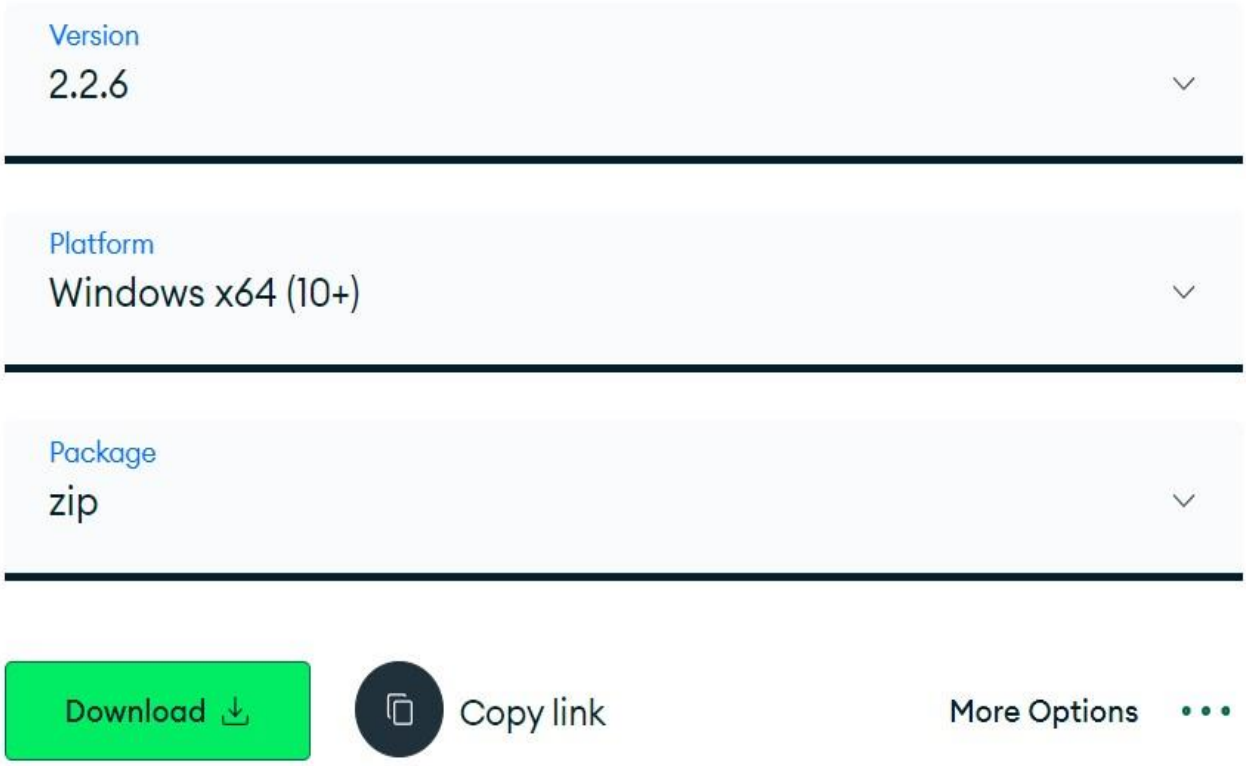
Installing the mongo Shell

MongoDB Shell is the quickest way to connect to (and work with) MongoDB. Easily query data, configure settings, and execute other actions with this modern, extensible commandline interface — replete with syntax highlighting, intelligent autocomplete, contextual help, and error messages.

Note: MongoDB Shell is an open source (Apache 2.0), standalone product developed separately from the MongoDB Server.

Steps to Download MongoDB Shell

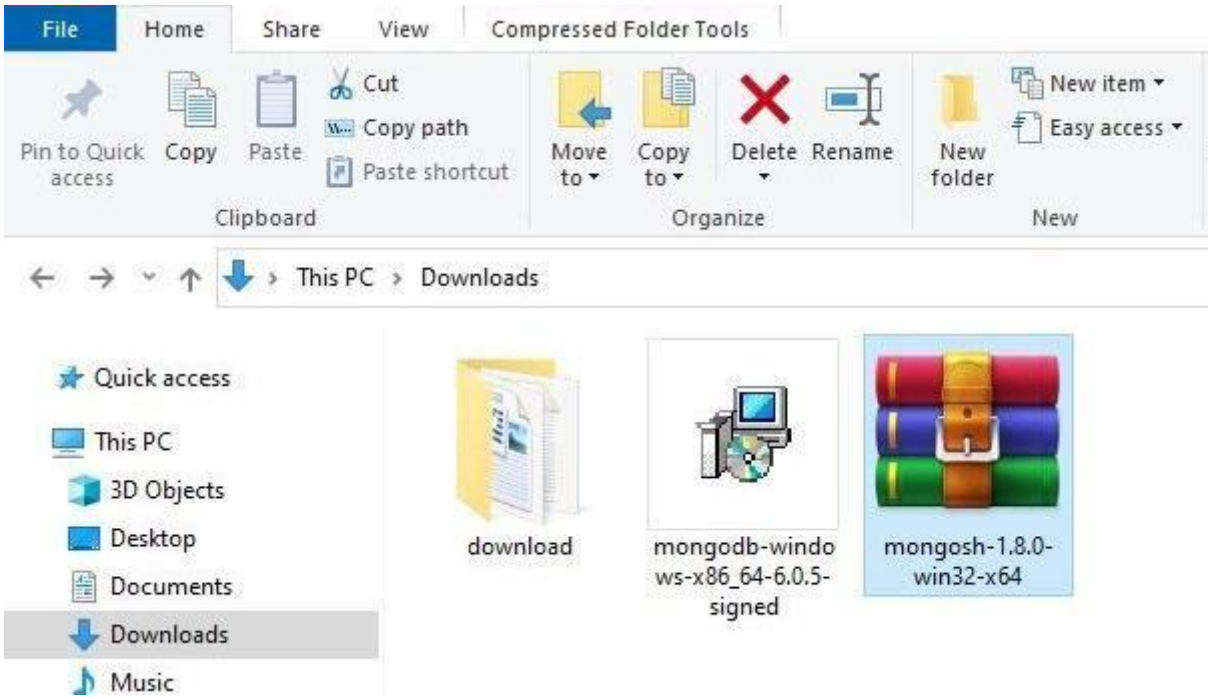
Step1: Open the MongoDB Shell download page and Open the MongoDB Download Center.



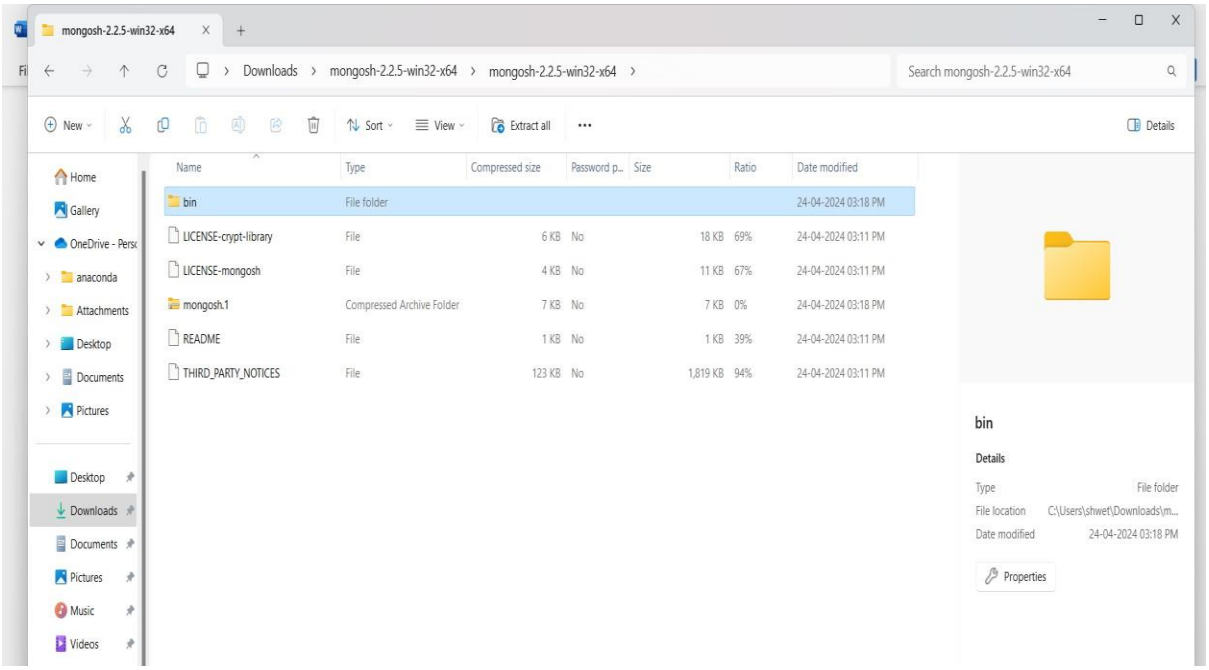
Select any version, Windows, and package according to your requirement. For Windows, we need to choose:

Version: 2.2.6
OS: Windows x64
Package: zip

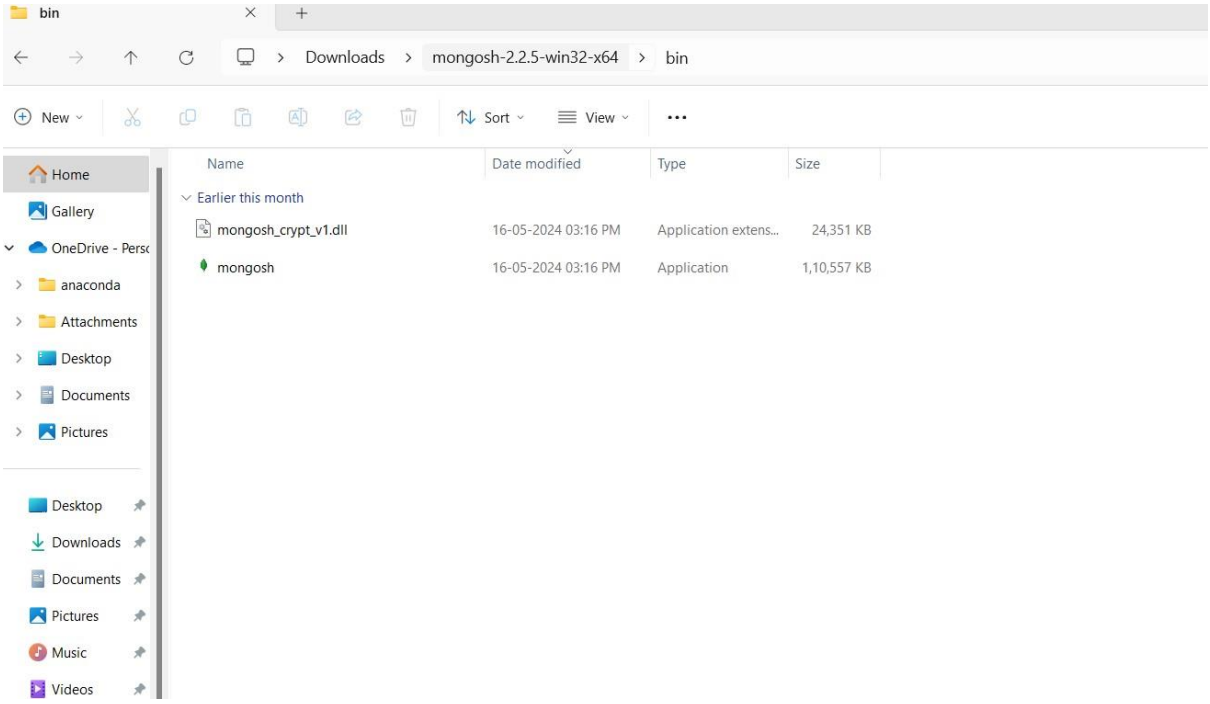
Step2: Locate the downloaded .zip file in your Downloads folder (or wherever you saved it).Right-click on the mongosh-2.2.6-win32-x64 file and select "Extract Here"



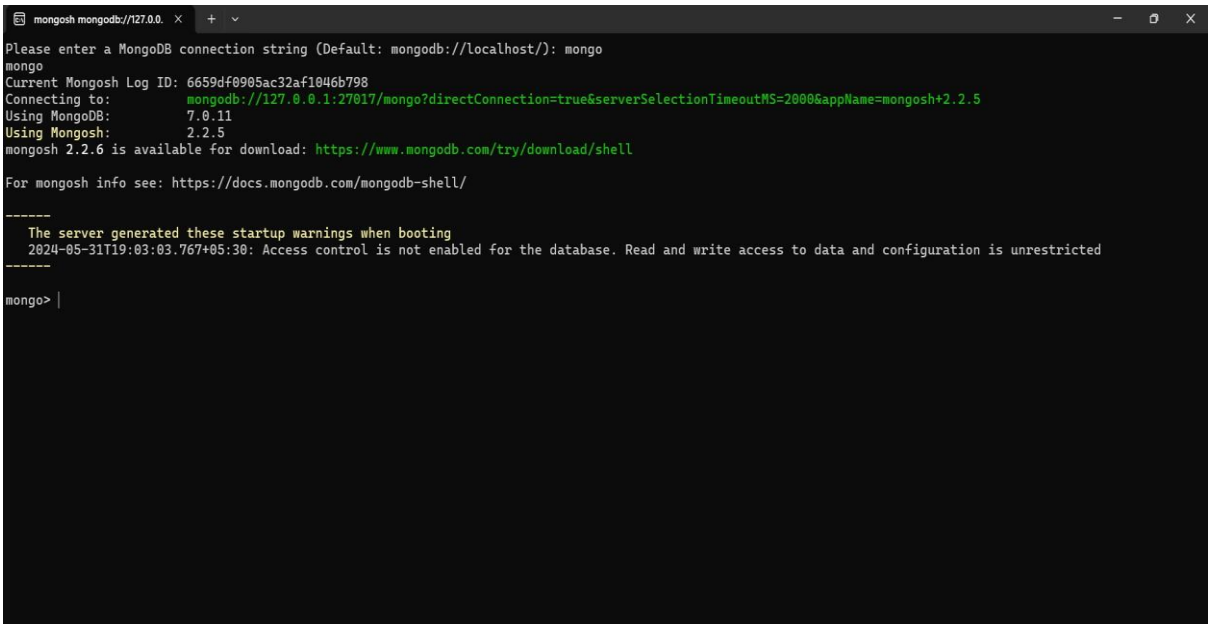
Step3: Click on mongosh-2.2.6-win32-x64 file folder and select bin folder.



Step4: Select mongosh application and complete the MongoDB Shell installation process



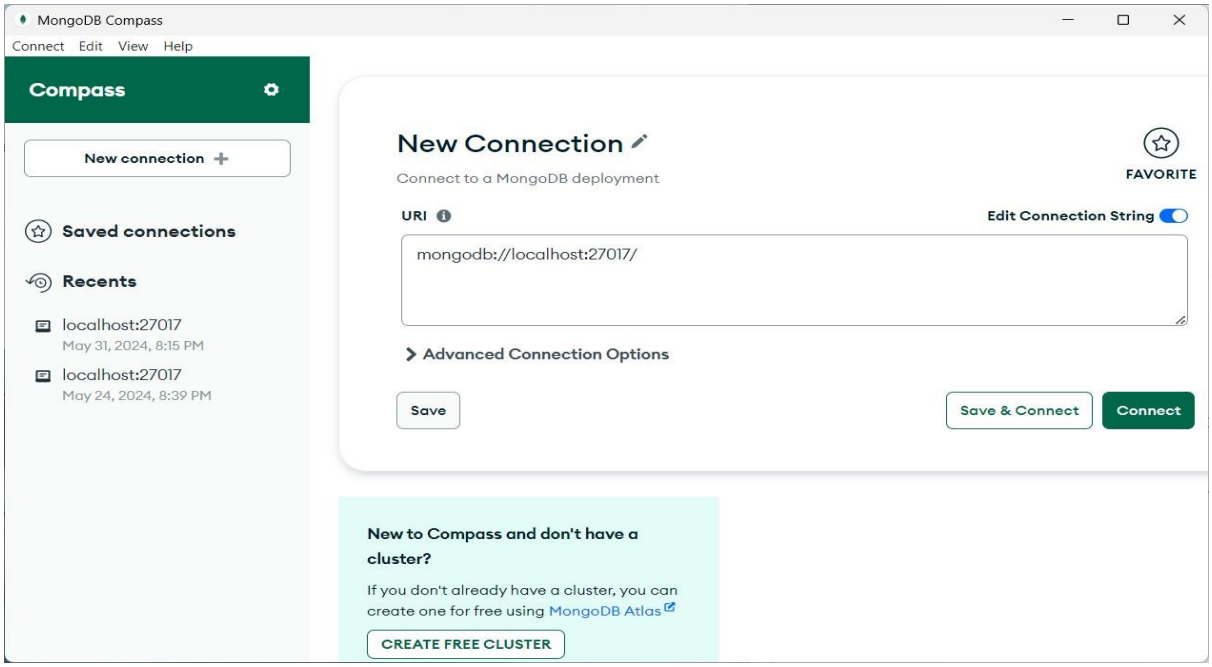
Step5: Next Terminal Window Will Open. Then Run the command `mongo` to start the shell.



MongoDB Compass:

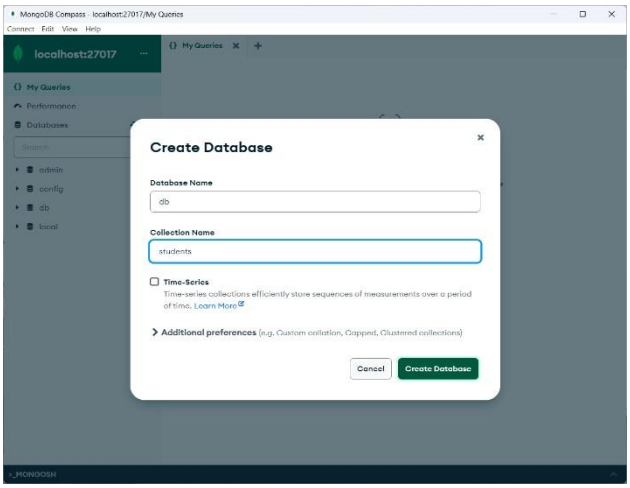
Connect to MongoDB database

Open MongoDB Compass and click on connect button.



Select the databases and click on create Database.

Give Database Name and Collection Name to create Database and click on create Database ○ **Database:** A container that holds multiple collections of related data. **Collection:** Similar to a table in a relational database, a collection stores documents.



How to Import data in MongoDb Compass?

In Compass, it has always been quite easy to import data – from JSON and CSV files

Click on Import Button

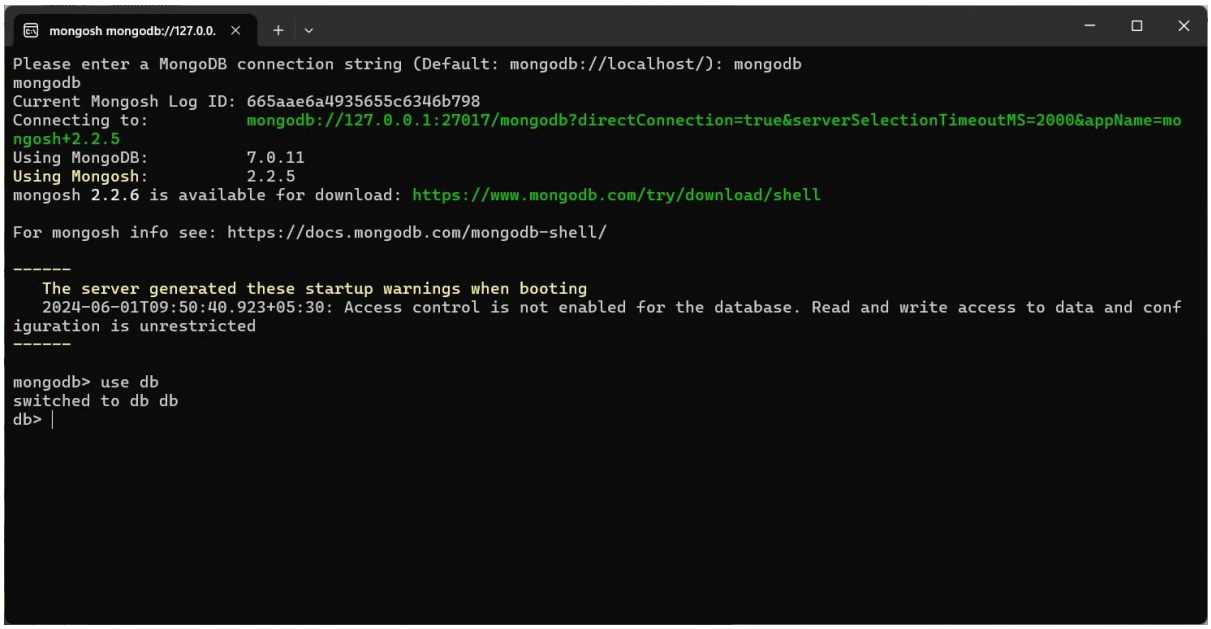
Select the exported JSON or CSV file you want to import.
Select the connection names you want to import. Click Import.

After connections, Few commands to test:

Switching Databases:

By default, the `db` points to the "test" database when you start the Mongo shell. To switch to a different database named "db", you can use the `use` command followed by the database name

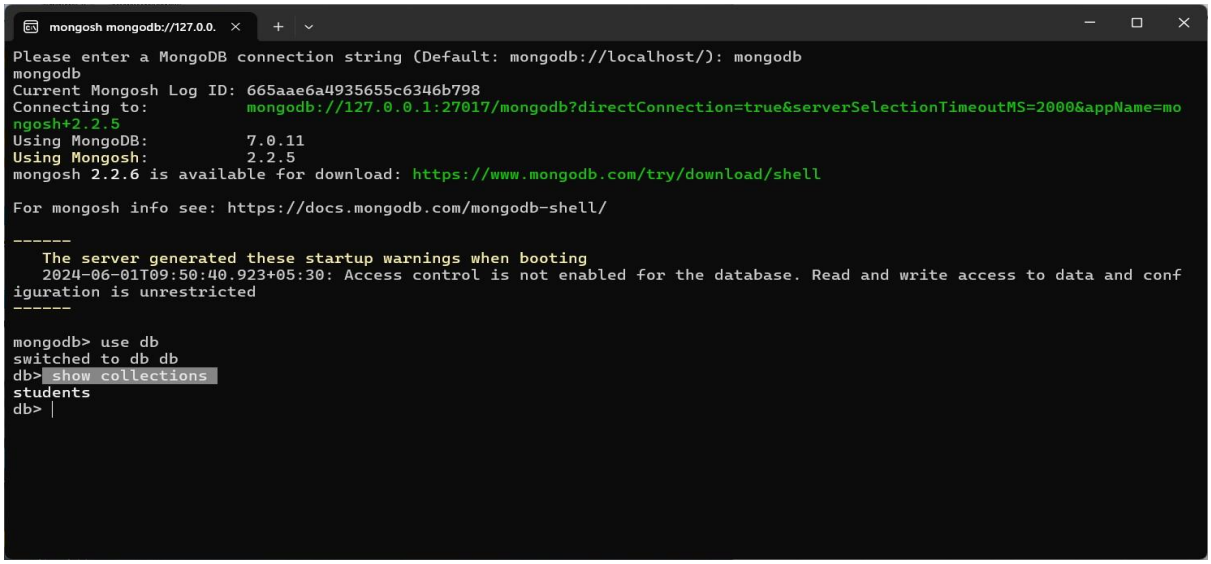
use db



Accessing Collections in the Current Database:

The `db` variable acts as a reference point for accessing collections within the current database you're connected to.

show collections



db.color.insertOne({"color" : "red"}) // Insert a record to collection. Create collection if not exit.

Documents:

In Mongodb, documents are the fundamental unit of data storage. They act like containers holding information in a flexible schema. Here's a breakdown of what documents are in MongoDB

Example:

```
_id: ObjectId('6650add5ec3192109fecd31a')
name : "Student 948"
age : 19
courses : "['English', 'Computer Science', 'Physics', 'Mathematics']"
gpa : 3.44
home_city : "City 2"
blood_group : "O+"
is_hotel_resident : true
```

- **_id** is a special field that uniquely identifies each document by default.
- Other fields like **name, age**, and **courses** hold specific data about the document.
- Nested documents like **home_city** can be used to group related information.

Benefits:

- The flexible schema of documents allows you to store data without rigid table structures like in relational databases.
- Documents can efficiently represent complex data hierarchies.

Documents can be represented as JSON

In MongoDB, documents are the fundamental unit of data storage. Each document represents a record and uses a JSON-like structure. This structure allows for storing various data types within a document and enables easy manipulation and querying of data.

So, while JSON is not exclusive to documents, it serves as a powerful and widely used way to represent them, especially in NoSQL databases like MongoDB.

Collections:

In MongoDB, collections act like containers that hold documents, which are essentially JSON-like structures that store your actual data. You can think of collections as similar to tables in relational databases, but with some key differences.

Examples:

- _id ○ name ○ email
- orders

Database:

- MongoDB groups collections into databases.
- A single instance of MongoDB can host several databases, each grouping together zero or more collections.
- A database has its own permissions, and each database is stored in separate files on disk.

Datatype:

MongoDB stores data using a format called BSON (Binary JSON), which extends JSON with additional data types. Here's a rundown of the common data types you'll encounter in MongoDB

Basic Types:

- **String:** The most common type, used for textual data. It must be valid UTF-8.
- **Integer:** Stores whole numbers, can be 32-bit or 64-bit depending on your server configuration.
- **Double:** Stores floating-point numbers with higher precision.
- **Boolean:** Represents true or false values.

Complex Types:

- **Array:** Stores an ordered list of values of various data types.
- **Object:** Used for embedding documents within documents, helpful for representing hierarchical data.
- **Date:** Stores dates and times as milliseconds since the Unix epoch.
- **ObjectId:** A unique 12-byte identifier automatically generated for each document when inserted.

Other Data Types:

- **Null:** Represents missing or undefined values.
- **Binary:** Stores raw binary data like images or files.
- **Code:** Allows storing JavaScript code within a document for server-side execution.
- **Symbol:** Similar to strings but less common, often used for specific data types in some programming languages.
- **Timestamp:** Represents a specific point in time with nanosecond precision.
- **Decimal128:** Stores high-precision decimal numbers.