



# MongoDB

- 1. Introduction to MongoDB***
- 2. Installation***
- 3. MongoDB Shell vs MongoDB Server***
- 4. MongoDB CRUD Operations***
- 5. Create Database***
- 6. Create Collections***
- 7. Drop Database***
- 8. Drop Collections***
- 9. Operators in MongoDB***
- 10. Query Documents***
- 11. Insert Documents***
- 12. Update Documents***



# MongoDB

- 13. Delete Documents***
- 14. Working with Arrays***
- 15. Importing Data In MongoDB***
- 16. Projection***
- 17. Sorting Records***
- 18. Limiting Records***
- 19. Indexing***
- 20. Aggregating Documents***
- 21. Atomic Queries***
- 22. Data Modeling in MongoDB***
- 23. What are MongoDB Drivers?***
- 24. Data Types***



# MongoDB

***25. GridFS***

***26. MongoDB Compass***

***27. Replication***

***28. Sharding***



# What is MongoDB?

MongoDB is a NoSQL cross-platform, open-source, document oriented database written in C++.

It is a document-based, general-purpose, distributed database with scalability and flexibility.

It doesn't use tables and rows to store its data, but instead *collections* of *documents*.

MongoDB is a schema-less database, so we don't need to specify the number or type of columns before inserting our data.

***MongoDb is a NoSQL document database***



# Objects in Mongo Db

SQL Server

- **Data Base**
- **Table**
- **Row**
- **Column**

Mongo Db

- **Data Base**
- **Collection**
- **Document**
- **Field**



# Key Features of Mongo Db

1. Ad-hoc Queries - Queries which are not known while designing the structure
2. Aggregation – perform a operation on the grouped data
3. Schema Less Database – different documents can have different fields in the same collection
4. Document oriented – the data is stored in the form of documents which is identified with a unique ID.
5. Indexing - index the data to improve the search query performance
6. High Performance – Provides high availability, scalability & better query response
7. Grid FS – stores the large data in to different documents
8. Sharding – partition the very large data into smaller parts known as shards
9. Replication - creates a copy of the database on multiple machines which increases the **availability** & accessibility



# SQL Vs NoSQL Db

SQL	MongoDB
Matured or stable	Its new and updated frequently
It follows tabular structure	It follows document structure
It needs a proper schema	Its flexible in nature
Managing complex relations among different tables is easy	Its not that great in complex managing relationship
Its scales vertically	Horizontally scalable
Suggestible for structured data	Can be used for non structured data





# Key Components of Mongo Db

- **\_id** – this is a mandatory field in every document which represents a unique key (Primary Key). If we create a document without **\_id** field MongoDB will automatically create it.
- **Collection** – A collection is just like table in SQL which contains a group of documents.
- **Data Base** – Just like in SQL we have database similarly we have database in MongoDB which is container for collections.
- **Document** – A record in MongoDB is created in a collection., which contains fields.
- **Field** – it is a pair of Name /key & value.



# Key Components of Mongo Db

```
MongoDB Enterprise > use test
switched to db test
MongoDB Enterprise > db.student.find().pretty();
{
  "_id" : ObjectId("608e9e7c8182197faa80a4c6"),
  "rno" : 1,
  "name" : "ajay",
  "age" : 13
}
MongoDB Enterprise > _
```

Database

Collection

Auto ID field

Key : Value Pair

Document



# Format to create a document

Start & End with curly braces { }

Separate Key & value with colon :

Separate Key & value pair with comma ,

“Keys” also referred to as “fields” must be given in quotation marks “”

```
db.books.insert({"name" : "Rajat  
Sharma", "Addr": "Sector 5 Dwarka", "city" : "Delhi", "DOB"  
: "13-Jan-1990"})
```



# Create a document

```
db.test.insert(  
    {"name" : "Rajat Sharma",  
     "Addr":"Sector 5 Dwarka",  
     "city" : "Delhi",  
     "Phone" : "99999003456"  
     "DOB" : "13-Jan-1990"}  
)
```

Pros of JSON : Friendly, Readable, Familiar

Cons of JSON : Text Based, Space Consuming, Limited

Data is stored in BSON format

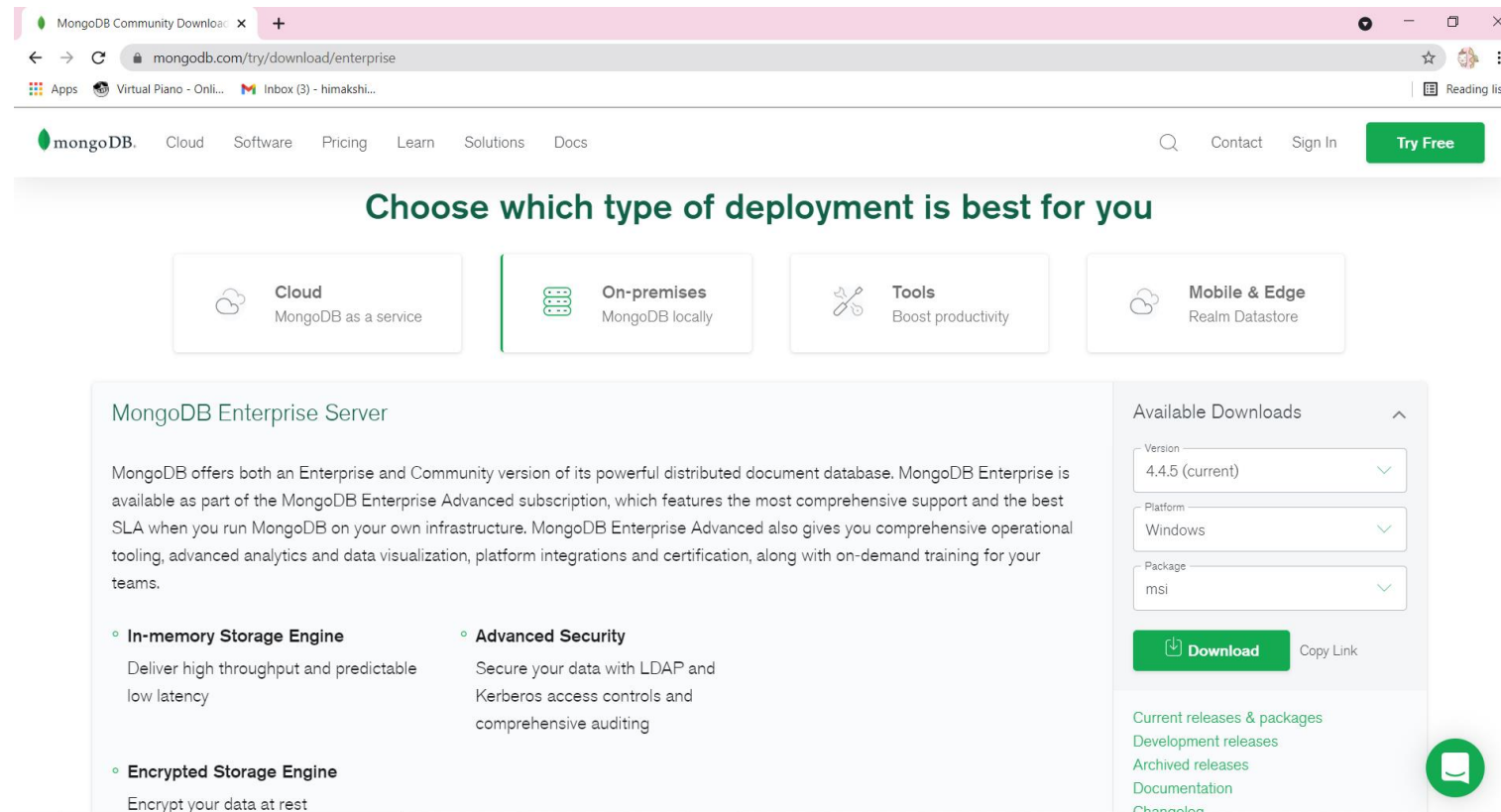


# Installing Mongo Db

We need to install MongoDB EnterpriseServer.

The installation can be done using a ZIP or msi file.

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



The screenshot shows the MongoDB website's download page for Enterprise. The browser's address bar displays 'mongodb.com/try/download/enterprise'. The page features a navigation bar with links to 'Cloud', 'Software', 'Pricing', 'Learn', 'Solutions', and 'Docs'. A 'Try Free' button is visible in the top right. The main heading is 'Choose which type of deployment is best for you', followed by four options: 'Cloud' (MongoDB as a service), 'On-premises' (MongoDB locally), 'Tools' (Boost productivity), and 'Mobile & Edge' (Realm Datastore). The 'On-premises' option is selected. Below this, the 'MongoDB Enterprise Server' section provides a description of the Enterprise version and lists its features: 'In-memory Storage Engine' (high throughput and low latency), 'Encrypted Storage Engine' (data at rest encryption), 'Advanced Security' (LDAP and Kerberos access controls), and 'comprehensive auditing'. On the right, the 'Available Downloads' section allows users to select the version (4.4.5), platform (Windows), and package type (msi). A 'Download' button and a 'Copy Link' option are provided. At the bottom right, there are links for 'Current releases & packages', 'Development releases', 'Archived releases', 'Documentation', and 'Changelog'.

MongoDB Community Download x +

mongodb.com/try/download/enterprise

Apps Virtual Piano - Onli... Inbox (3) - himakshi...

Reading list

mongoDB. Cloud Software Pricing Learn Solutions Docs

Search Contact Sign In Try Free

### Choose which type of deployment is best for you

- Cloud  
MongoDB as a service
- On-premises**  
MongoDB locally
- Tools  
Boost productivity
- Mobile & Edge  
Realm Datastore

#### MongoDB Enterprise Server

MongoDB offers both an Enterprise and Community version of its powerful distributed document database. MongoDB Enterprise is available as part of the MongoDB Enterprise Advanced subscription, which features the most comprehensive support and the best SLA when you run MongoDB on your own infrastructure. MongoDB Enterprise Advanced also gives you comprehensive operational tooling, advanced analytics and data visualization, platform integrations and certification, along with on-demand training for your teams.

- In-memory Storage Engine**  
Deliver high throughput and predictable low latency
- Encrypted Storage Engine**  
Encrypt your data at rest
- Advanced Security**  
Secure your data with LDAP and Kerberos access controls and comprehensive auditing

#### Available Downloads

Version: 4.4.5 (current) ✓

Platform: Windows ✓

Package: msi ✓

**Download** Copy Link

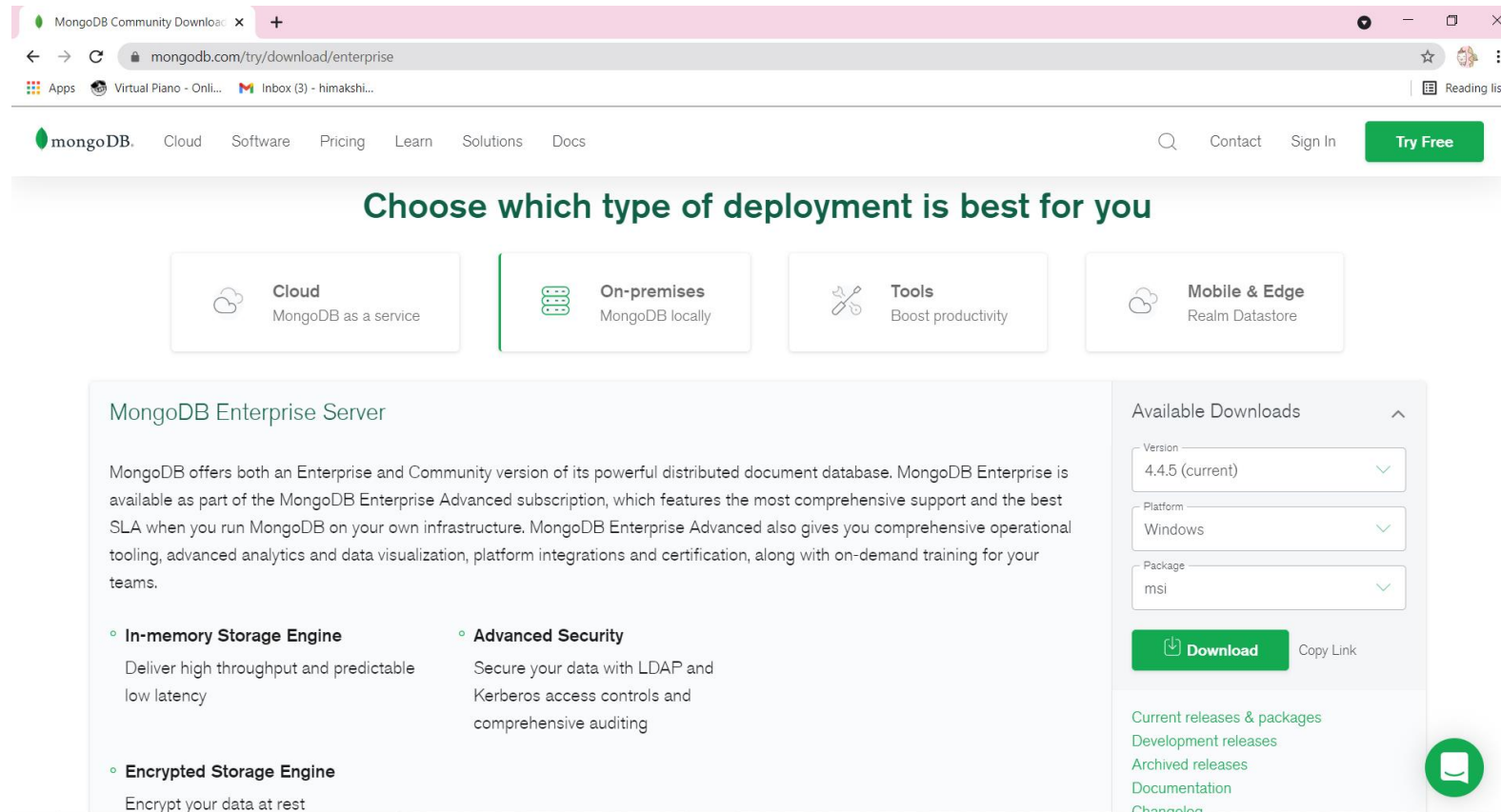
[Current releases & packages](#)  
[Development releases](#)  
[Archived releases](#)  
[Documentation](#)  
[Changelog](#)



# Installing Mongo Db

Installation through msi file.

`mongodb-windows-x86_64-enterprise-4.4.5-signed`



The screenshot shows the MongoDB Community Download page for Enterprise Server. The page is viewed in a web browser with the URL `mongodb.com/try/download/enterprise`. The navigation bar includes links for Cloud, Software, Pricing, Learn, Solutions, and Docs, along with a search icon, Contact, Sign In, and a Try Free button. The main heading is "Choose which type of deployment is best for you". Below this, there are four deployment options: Cloud (MongoDB as a service), On-premises (MongoDB locally), Tools (Boost productivity), and Mobile & Edge (Realm Datastore). The On-premises option is selected. The main content area is titled "MongoDB Enterprise Server" and describes the Enterprise version. It lists features: In-memory Storage Engine, Encrypted Storage Engine, and Advanced Security. The right sidebar shows "Available Downloads" with filters for Version (4.4.5 (current)), Platform (Windows), and Package (msi). A Download button and a Copy Link button are present. At the bottom of the sidebar, there are links for Current releases & packages, Development releases, Archived releases, Documentation, and Changelog. A chat icon is visible in the bottom right corner.

MongoDB Community Download

mongodb.com/try/download/enterprise

Apps Virtual Piano - Onli... Inbox (3) - himakshi... Reading list

mongodb Cloud Software Pricing Learn Solutions Docs Contact Sign In Try Free

### Choose which type of deployment is best for you

- Cloud  
MongoDB as a service
- On-premises  
MongoDB locally**
- Tools  
Boost productivity
- Mobile & Edge  
Realm Datastore

#### MongoDB Enterprise Server

MongoDB offers both an Enterprise and Community version of its powerful distributed document database. MongoDB Enterprise is available as part of the MongoDB Enterprise Advanced subscription, which features the most comprehensive support and the best SLA when you run MongoDB on your own infrastructure. MongoDB Enterprise Advanced also gives you comprehensive operational tooling, advanced analytics and data visualization, platform integrations and certification, along with on-demand training for your teams.

- In-memory Storage Engine**  
Deliver high throughput and predictable low latency
- Encrypted Storage Engine**  
Encrypt your data at rest
- Advanced Security**  
Secure your data with LDAP and Kerberos access controls and comprehensive auditing

#### Available Downloads

Version: 4.4.5 (current)

Platform: Windows

Package: msi

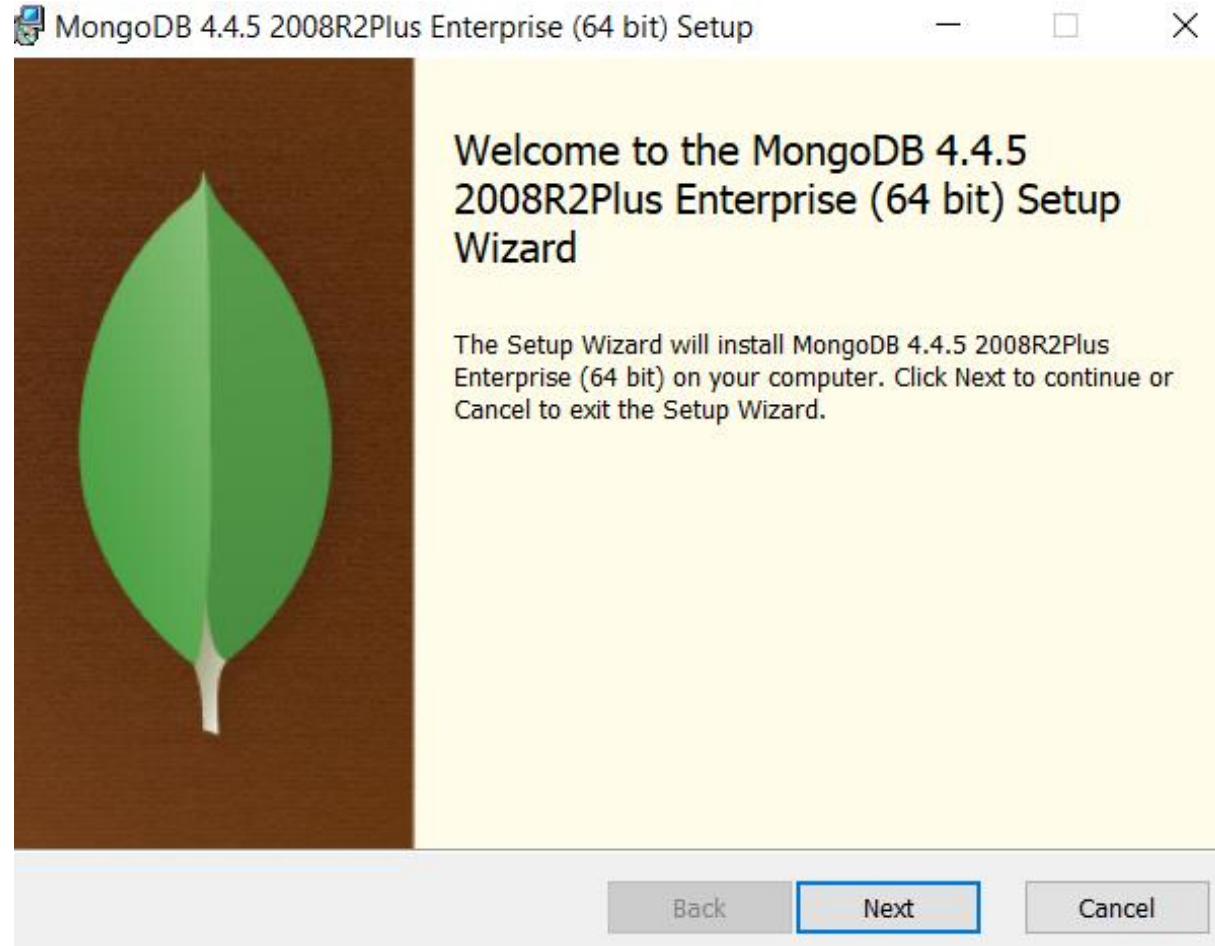
[Copy Link](#)

[Current releases & packages](#)  
[Development releases](#)  
[Archived releases](#)  
[Documentation](#)  
[Changelog](#)



# Installing Mongo Db

Double click on the downloaded file to initiate the installation.



# Installing Mongo Db

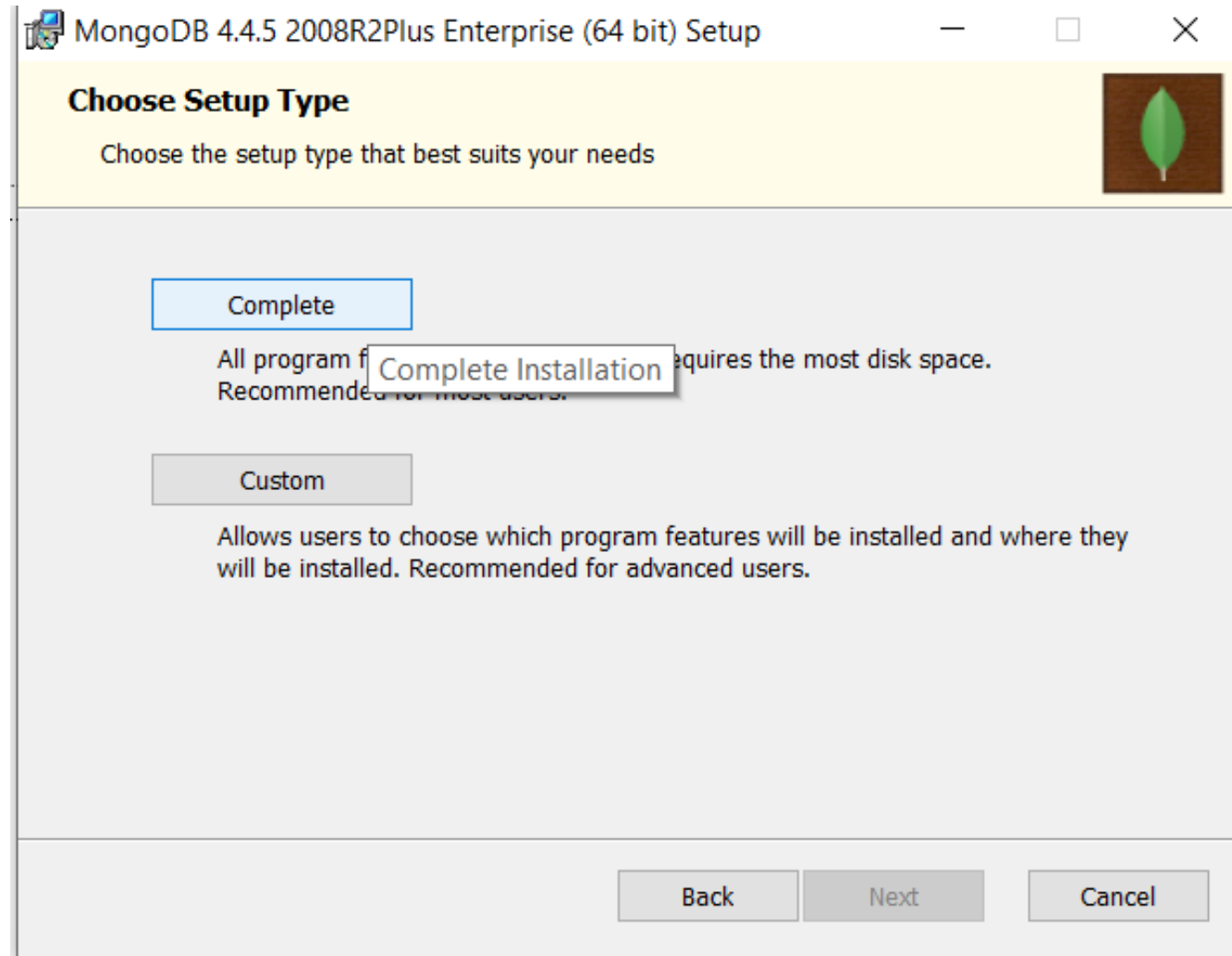
Accept the license agreement





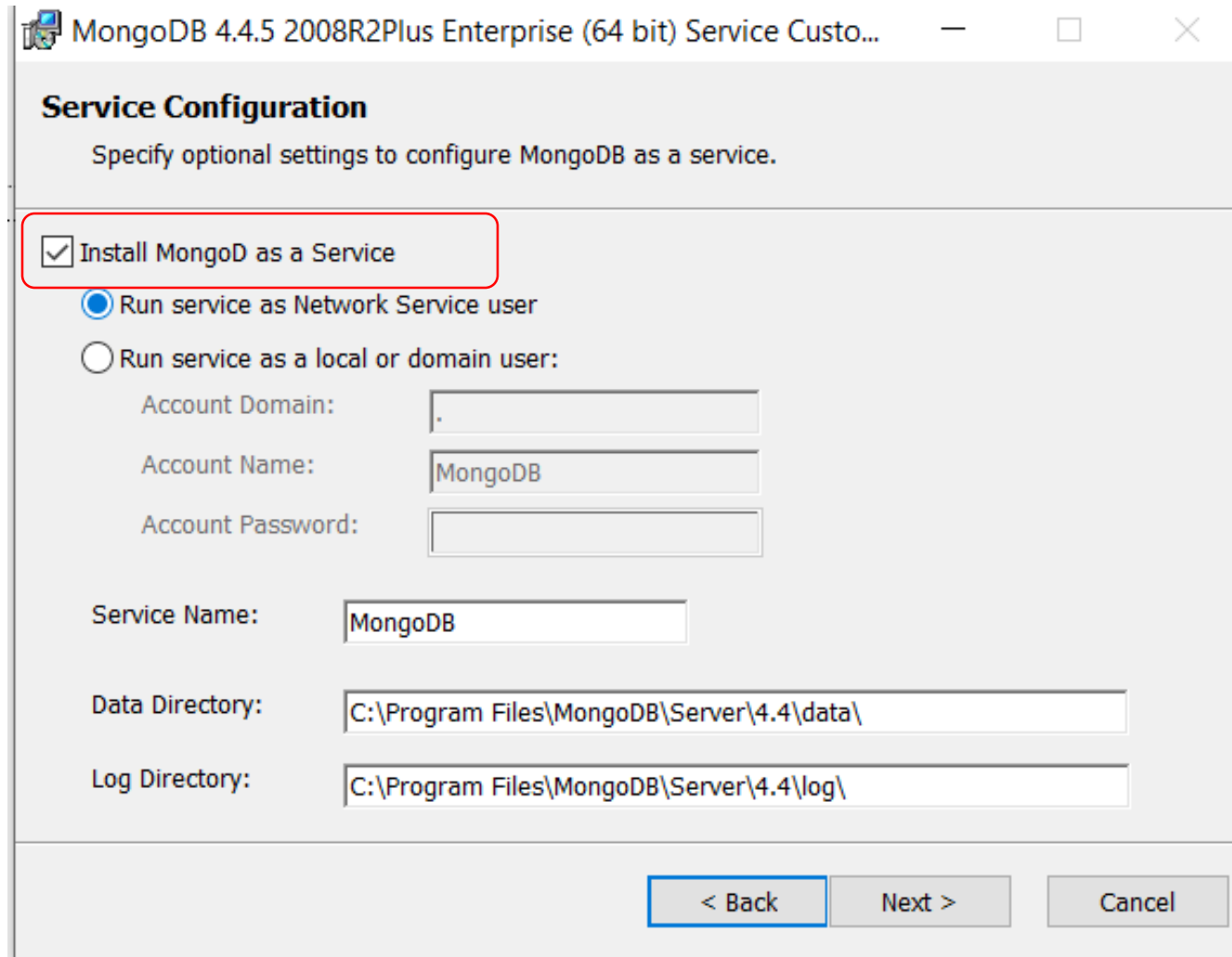
# Installing Mongo Db

Choose Complete Installation



# Installing Mongo Db

Select data location. Ensure Install MongoDB as a service is checked



MongoDB 4.4.5 2008R2Plus Enterprise (64 bit) Service Custo...

### 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:

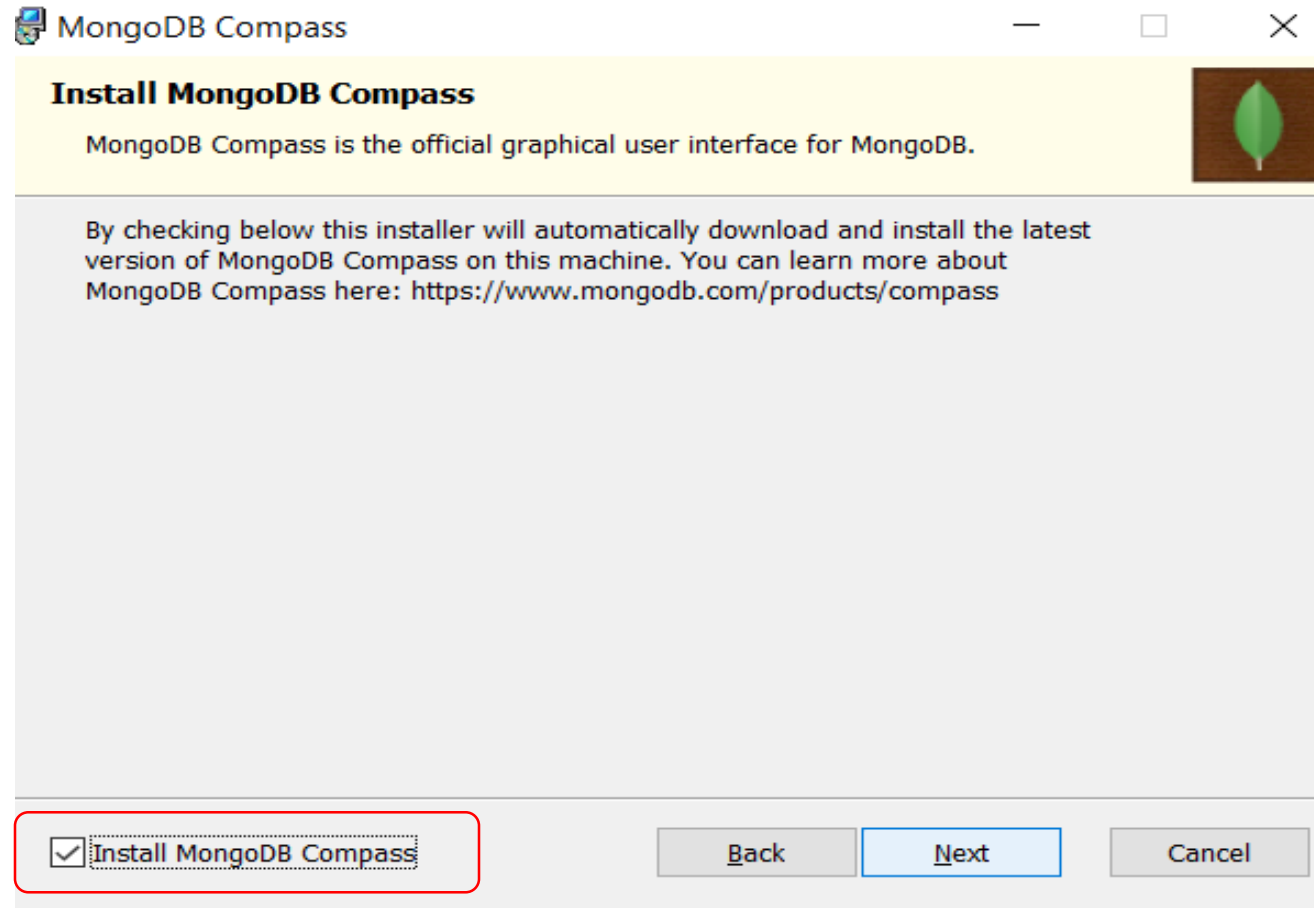
Data Directory:

Log Directory:

< Back    Next >    Cancel

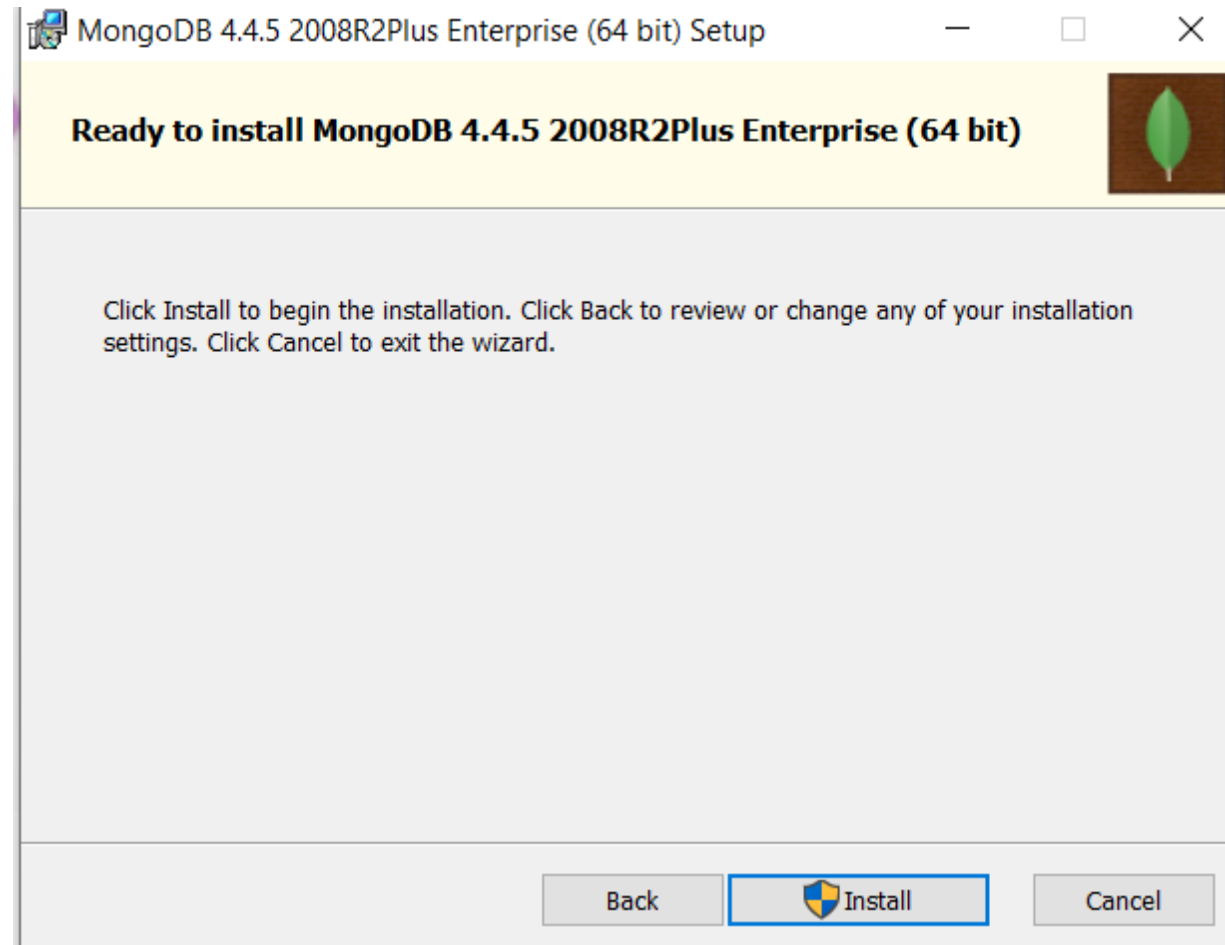
# Installing Mongo Db

Clear checkbox for Install Compass

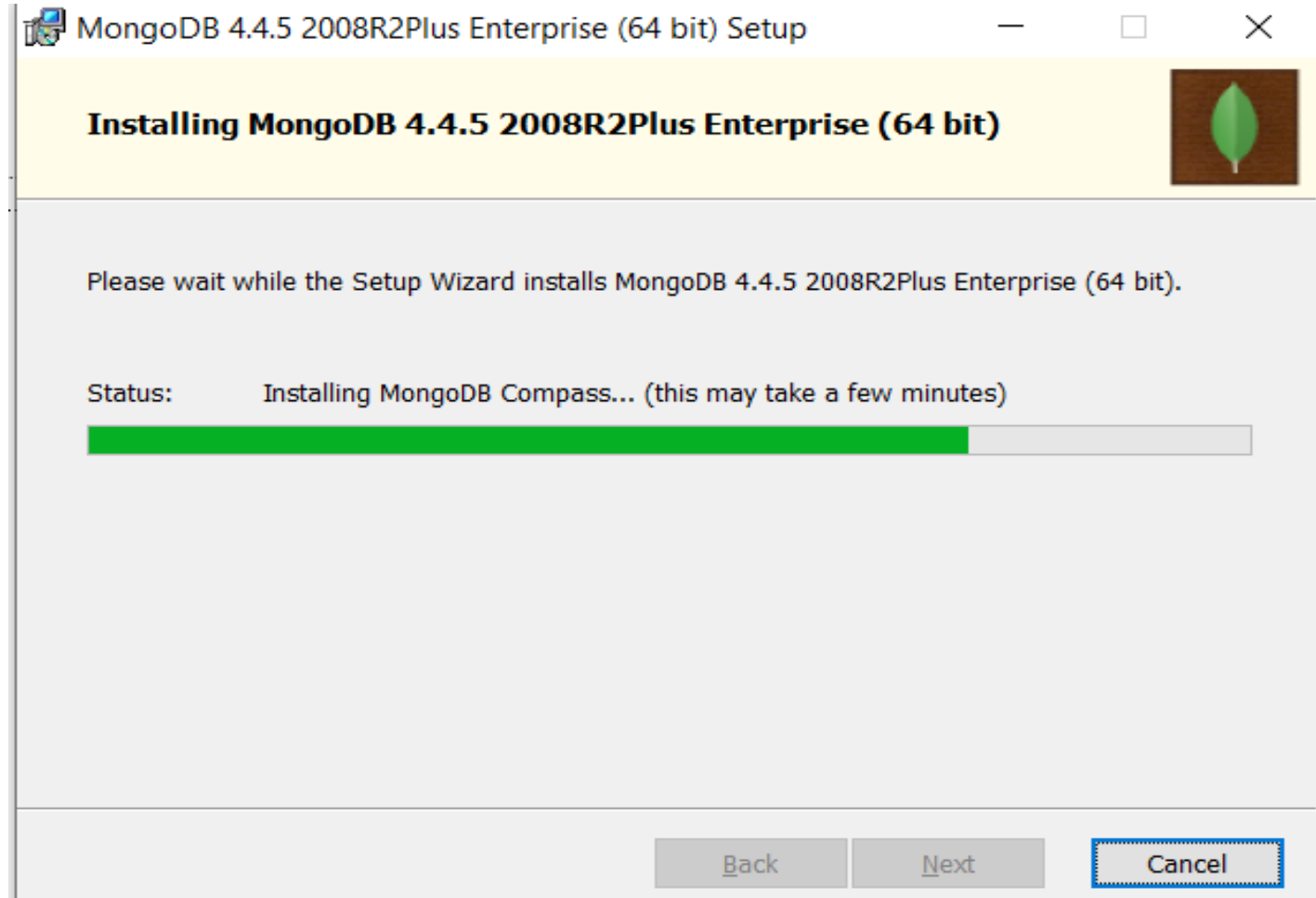


# Installing Mongo Db

Click Install to begin the installation

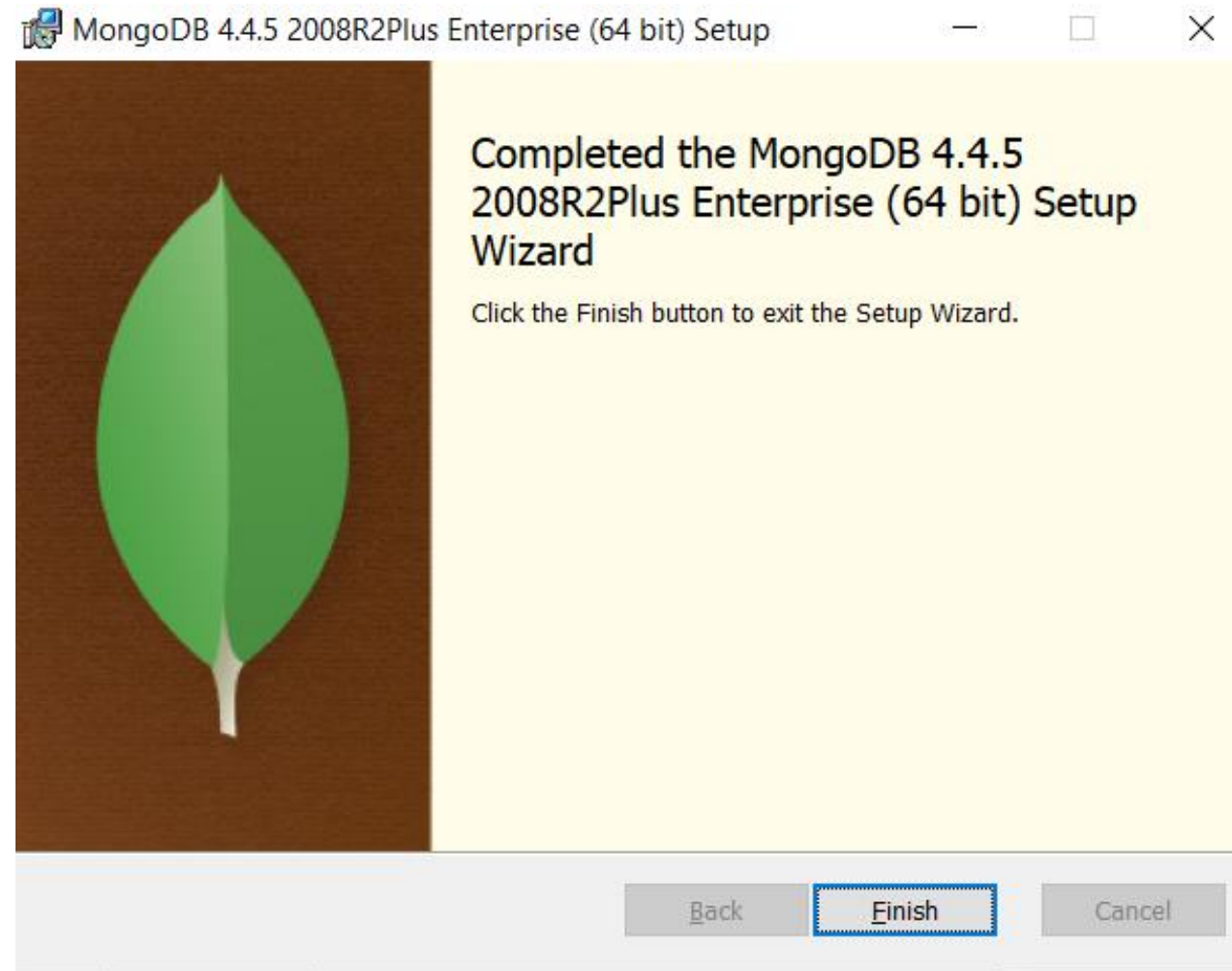


# Installing Mongo Db



# Installing Mongo Db

Click Finish to complete the installation



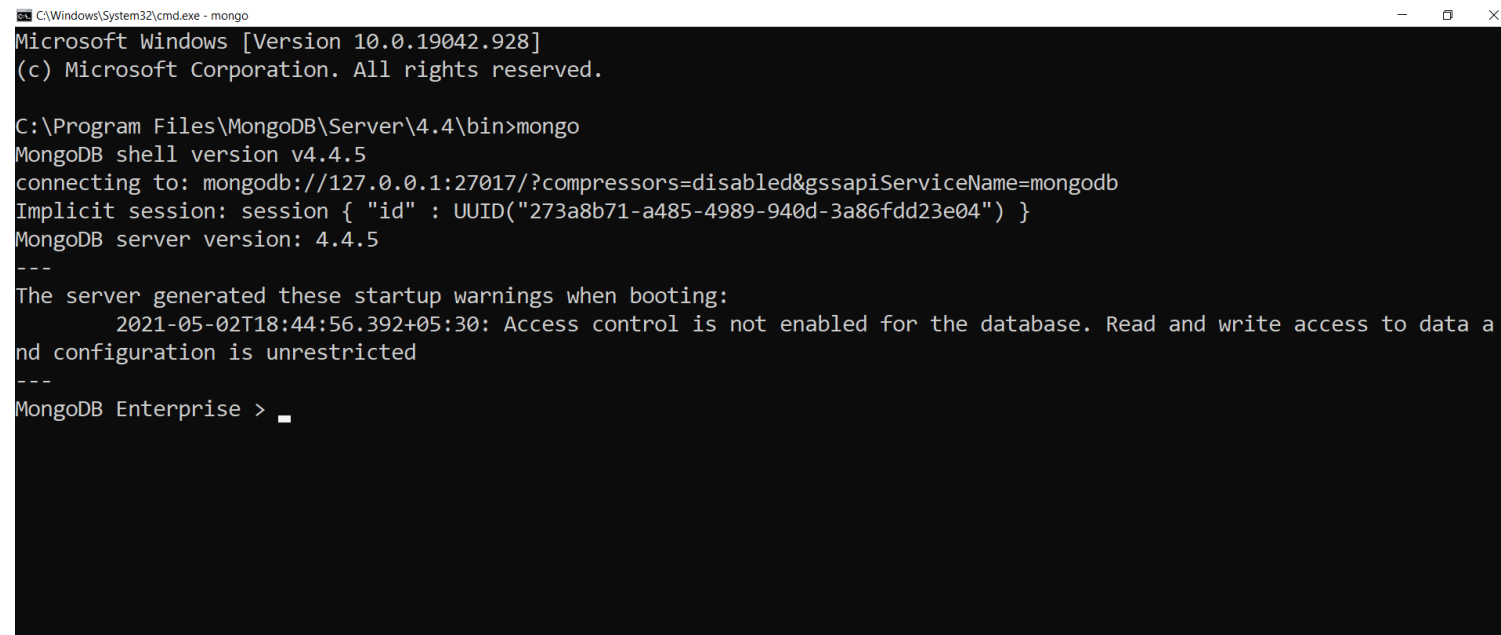
# Open Mongo Db

To run MongoDB shell browse to data location.

C:\Program Files\MongoDB\Server\4.4\bin

In the address bar run CMD command . It will take you to command prompt.

Type in mongo and press enter to get into MongoDB shell

A screenshot of a Windows command prompt window titled "C:\Windows\System32\cmd.exe - mongo". The window shows the execution of the 'mongo' command from the directory 'C:\Program Files\MongoDB\Server\4.4\bin'. The output displays the MongoDB shell version (v4.4.5), the connection details (connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb), the implicit session information, and the MongoDB server version (4.4.5). It also shows startup warnings generated by the server, including a message about access control not being enabled for the database. The prompt ends with "MongoDB Enterprise >".

```
C:\Windows\System32\cmd.exe - mongo
Microsoft Windows [Version 10.0.19042.928]
(c) Microsoft Corporation. All rights reserved.

C:\Program Files\MongoDB\Server\4.4\bin>mongo
MongoDB shell version v4.4.5
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("273a8b71-a485-4989-940d-3a86fdd23e04") }
MongoDB server version: 4.4.5
---
The server generated these startup warnings when booting:
  2021-05-02T18:44:56.392+05:30: Access control is not enabled for the database. Read and write access to data a
nd configuration is unrestricted
---
MongoDB Enterprise > _
```

# Installing Mongo Db

Installation using a ZIP file.

Download MongoDB zip package

- Create Directory Structure for MongoDB Server  
C:\data\db
- Start MongoDB Server
- Start MongoDB Shell





# Installing Mongo Db

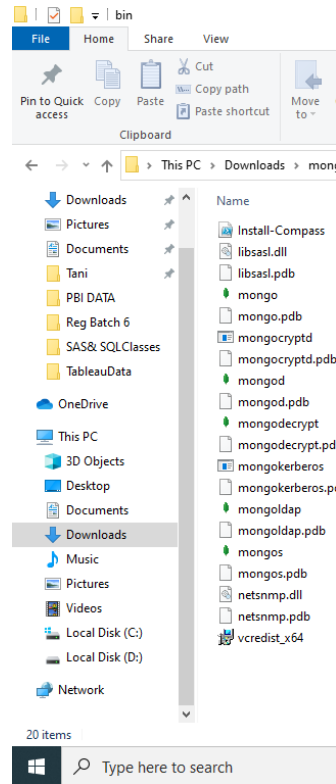
## Unzip the downloaded folder

Under the bin folder we need to execute below application files from the command prompt:

## Mongod – Server file

# Mongo – Shell

Note : the downloaded file should be in C: drive



# Installing Mongo Db

Go to the command prompt and execute the database file (mongod). This will run MongoDB server

```
Select C:\Windows\System32\cmd.exe - mongod
Microsoft Windows [Version 10.0.18363.1440]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Raj\Downloads\mongodb-win32-x86_64-enterprise-windows-4.4.5\bin>mongod
{"t":{"$date":"2021-04-28T13:17:03.634+05:30"},"s":"I", "c":"CONTROL", "id":23285, "ctx":"main","msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0 spec
ify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2021-04-28T13:17:03.638+05:30"},"s":"W", "c":"ASIO", "id":22601, "ctx":"main","msg":"No TransportLayer configured during NetworkInterface startup"}
{"t":{"$date":"2021-04-28T13:17:03.638+05:30"},"s":"I", "c":"NETWORK", "id":4648602, "ctx":"main","msg":"Implicit TCP FastOpen in use."}
{"t":{"$date":"2021-04-28T13:17:03.644+05:30"},"s":"W", "c":"ASIO", "id":22601, "ctx":"main","msg":"No TransportLayer configured during NetworkInterface startup"}
{"t":{"$date":"2021-04-28T13:17:03.647+05:30"},"s":"I", "c":"STORAGE", "id":4615611, "ctx":"initandlisten","msg":"MongoDB starting","attr":{"pid":13364,"port":27017,"
dbPath":"C:/data/db/","architecture":"64-bit","host":"DESKTOP-GIHIQ7D"}}
{"t":{"$date":"2021-04-28T13:17:03.647+05:30"},"s":"I", "c":"CONTROL", "id":23398, "ctx":"initandlisten","msg":"Target operating system minimum version","attr":{"ta
rgetMinOS":"Windows 7/Windows Server 2008 R2"}}
{"t":{"$date":"2021-04-28T13:17:03.647+05:30"},"s":"I", "c":"CONTROL", "id":23403, "ctx":"initandlisten","msg":"Build Info","attr":{"buildInfo":{"version":"4.4.5","
gitVersion":"ff5cb77101b052fa02da43b8538093486cf9b3f7","modules":["enterprise"],"allocator":"tcmalloc","environment":{"distmod":"windows","distarch":"x86_64","target_ar
ch":"x86_64"}}}}
{"t":{"$date":"2021-04-28T13:17:03.648+05:30"},"s":"I", "c":"CONTROL", "id":51765, "ctx":"initandlisten","msg":"Operating System","attr":{"os":{"name":"Microsoft Wi
ndows 10","version":"10.0 (build 18363)}}}
{"t":{"$date":"2021-04-28T13:17:03.648+05:30"},"s":"I", "c":"CONTROL", "id":21951, "ctx":"initandlisten","msg":"Options set by command line","attr":{"options":{}}}
{"t":{"$date":"2021-04-28T13:17:03.664+05:30"},"s":"I", "c":"STORAGE", "id":22315, "ctx":"initandlisten","msg":"Opening WiredTiger","attr":{"config":"create,cache_s
ize=3515M,session_max=33000,eviction=(threads_min=4,threads_max=4),config_base=false,statistics=(fast),log=(enabled=true,archive=true,path=journal,compressor=snappy),fi
le_manager=(close_idle_time=100000,close_scan_interval=10,close_handle_minimum=250),statistics_log=(wait=0),verbose=[recovery_progress,checkpoint_progress,compact_progr
ess],"}}}
{"t":{"$date":"2021-04-28T13:17:04.079+05:30"},"s":"I", "c":"STORAGE", "id":22430, "ctx":"initandlisten","msg":"WiredTiger message","attr":{"message":"[1619596024:7
9068][13364:140715815624352], txn-recover: [WT_VERB_RECOVERY | WT_VERB_RECOVERY_PROGRESS] Set global recovery timestamp: (0, 0)"}
{"t":{"$date":"2021-04-28T13:17:04.080+05:30"},"s":"I", "c":"STORAGE", "id":22430, "ctx":"initandlisten","msg":"WiredTiger message","attr":{"message":"[1619596024:8
0079][13364:140715815624352], txn-recover: [WT_VERB_RECOVERY | WT_VERB_RECOVERY_PROGRESS] Set global oldest timestamp: (0, 0)"}
{"t":{"$date":"2021-04-28T13:17:04.177+05:30"},"s":"I", "c":"STORAGE", "id":4795906, "ctx":"initandlisten","msg":"WiredTiger opened","attr":{"durationMillis":512}}
{"t":{"$date":"2021-04-28T13:17:04.178+05:30"},"s":"I", "c":"RECOVERY", "id":23987, "ctx":"initandlisten","msg":"WiredTiger recoveryTimestamp","attr":{"recoveryTimes
tamp":{"$timestamp":{"t":0,"i":0}}}}
{"t":{"$date":"2021-04-28T13:17:04.406+05:30"},"s":"I", "c":"STORAGE", "id":4366408, "ctx":"initandlisten","msg":"No table logging settings modifications are required
for existing WiredTiger tables","attr":{"loggingEnabled":true}}
{"t":{"$date":"2021-04-28T13:17:04.409+05:30"},"s":"I", "c":"STORAGE", "id":22262, "ctx":"initandlisten","msg":"Timestamp monitor starting"}
{"t":{"$date":"2021-04-28T13:17:04.513+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":{"$date":"2021-04-28T13:17:04.514+05:30"},"s":"W", "c":"CONTROL", "id":22140, "ctx":"initandlisten","msg":"This server is bound to localhost. Remote systems wi
ll be unable to connect to this server. Start the server with --bind_ip <address> to specify which IP addresses it should serve responses from, or with --bind_ip_all to
bind to all interfaces. If this behavior is desired, start the server with --bind_ip 127.0.0.1 to disable this warning","tags":["startupWarnings"]}
{"t":{"$date":"2021-04-28T13:17:04.524+05:30"},"s":"I", "c":"STORAGE", "id":20320, "ctx":"initandlisten","msg":"createCollection","attr":{"namespace":"admin.system.
version","uuidDisposition":"provided","uuid":{"$uuid":{"$uuid":"493d4c9b-ade8-4ec2-847a-90962feb23b9"}}, "options":{"$uuid":{"$uuid":"493d4c9b-ade8-4ec2-847a-90962feb23b9"}
}}}}
{"t":{"$date":"2021-04-28T13:17:04.703+05:30"},"s":"I", "c":"INDEX", "id":20345, "ctx":"initandlisten","msg":"Index build: done building","attr":{"buildUUID":null}}
```



# Installing Mongo Db

Open command prompt in another window and execute Shell file (mongo).

```
Microsoft Windows [Version 10.0.18363.1440]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Raj\Downloads\mongodb-win32-x86_64-enterprise-windows-4.4.5\bin>mongo
MongoDB shell version v4.4.5
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("44fb60cc-2e0f-40b2-bf16-0ca336efbf6c") }
MongoDB server version: 4.4.5
---
The server generated these startup warnings when booting:
  2021-04-28T13:29:41.984+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
  2021-04-28T13:29:41.987+05:30: 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
---
MongoDB Enterprise > _
```



Now you are in MongoDB shell.

# Using MongoDB commands

- > show dbs – display list of available database ;
- > use test – to create a database.
- > db – to check the current database.

**Note: Newly created database will not be displayed until it has some collections.**



# Creating Database

- > show dbs – display list of available database ;
- > use test – to create a database
- > db – display the name of current database

```
>db.stu.insertOne({  
  "rno" : 1,  
  "name" : "amit" ,  
  "age" : 14});
```

```
>db.stu.find();
```



# Dropping Database

> db.dropDatabase(); - drop the current database

> show dbs;



# Create Collections

`db.createCollection(name,options)` method is used to create a collection

```
> use mydb;
```

```
> db.createCollection("stu");
```

```
> show collections;
```

```
> db.createCollection("stu2",{capped : true, size : 50000,  
max : 5});
```



# Create Collections

We do not need to create a collection, MongoDB creates a collection automatically.

```
> db.stu3.insertOne(  
    {"rno" : 1,  
     "name" : "ajay kumar",  
     "age" : 14}  
);  
  
> show collections;
```





# Drop Collections

`db.collection.drop()` is used to drop a collection.

```
> db.stu2.drop();
```

```
> show collections;
```



# Inserting Documents

`db.collection.insertOne()` is used to insert a single document into a collection.

```
>db.stu.insertOne(  
    {"rno" : 1,  
     "name" : "kapil sharma",  
     "age" : 15}  
);  
  
> db.stu.find();
```



# Inserting Documents

`db.collection.insertMany()` is used to insert multiple document into a collection.

```
> db.stu3.insertMany([  
    {"rno" : 2,  
     "name" : "aman sharma",  
     "age" : 16},  
    {"rno" : 3,  
     "name" : "gaurav kapoor",  
     "age" : 14}  
]);
```



# Query Documents

find() method is used to query the document. It displays the documents in non structured way.

```
db.collectionname.find({});
```

> db.stu3.find({}); - non formatted output

> db.stu3.find({}).pretty(); - formatted output



# Query Documents

Specifying the keys:

```
> db.salary.find({"DEPT" : "HR"});
```

\$and: operator – used to specify multiple keys

```
> db.collectionname.find({  
    $and: [  
        {key : value1}, {key : value2}]  
    });
```



# Query Documents

Specifying the keys:

```
>db.salary.find( {$and:
```

```
    [ {"DEPT" : "HR"},  
      {"DESI" : "ASSOCIATE"}]  
    });
```

\$or: operator



# Query Documents

Specifying the keys:

\$or: operator

```
>db.collectionname.find({
```

```
    $or: [
```

```
        {key : value1}, {key : value2}]
```

```
    });
```

```
>db.salary.find( {
```

```
    $or: [
```

```
        {"DEPT" : "HR"},
```

```
        {"DESI" : "ASSOCIATE"}]
```

```
    }
```

```
});
```



# Update Documents

```
>db.collectionname.updateOne(<filter>,<update>,<options>);
```

```
>db.collectionname.updateMany(<filter>,<update>,<options>);
```

```
>db.collectionname.replaceOne(<filter>,<update>,<options>);
```

```
> db.emp.updateOne({"EID" : 1025},  
                    {$set: {"ADDRESS" : "B302 PRAGYA  
APARTMENTS, DWARKA, DELHI", "PHONE" : 9899245970} }  
                    );
```

```
> db.salary.updateMany({$and: [{"DEPT" : "HR"}, {"DESI" :  
"ASSOCIATE"}]}}, {$set: {"DESI" : "SR. ASSOCIATE"}});
```





# Update Documents

```
>db.collectionname.updateMany(<filter>,<update>,<options>);
```

```
> db.salary.updateMany(  
    {$and: [{"DEPT" : "HR"},  
            {"DESI" : "ASSOCIATE"}]},  
    {$set: {"DESI" : "SR. ASSOCIATE"}}  
);
```

```
> db.stu2.updateMany(  
    {},  
    {"$set" : {"class" : "8th"}});
```



# Update Documents

```
>db.collectionname.replaceOne(<filter>,<update>,<options>);
```

```
> db.stu.replaceOne(  
    {"rno" : 2},  
    {"rno" : 3 ,"name" : "ajay kumar",  
    "age" : 16}  
    );
```

Note: \_id remains same the document has been replaced



# ASSIGNMENT



- Create a database demo
- Create a EMP collection.
- Insert 8 documents containing in EMP collection containing eid, name, city, doj, dept, desi
- Display the documents in formatted manner.
- Show the documents containing “HR” dept.
- Show the documents for “OPS” managers.
- Promote all the associates as Sr.Associates