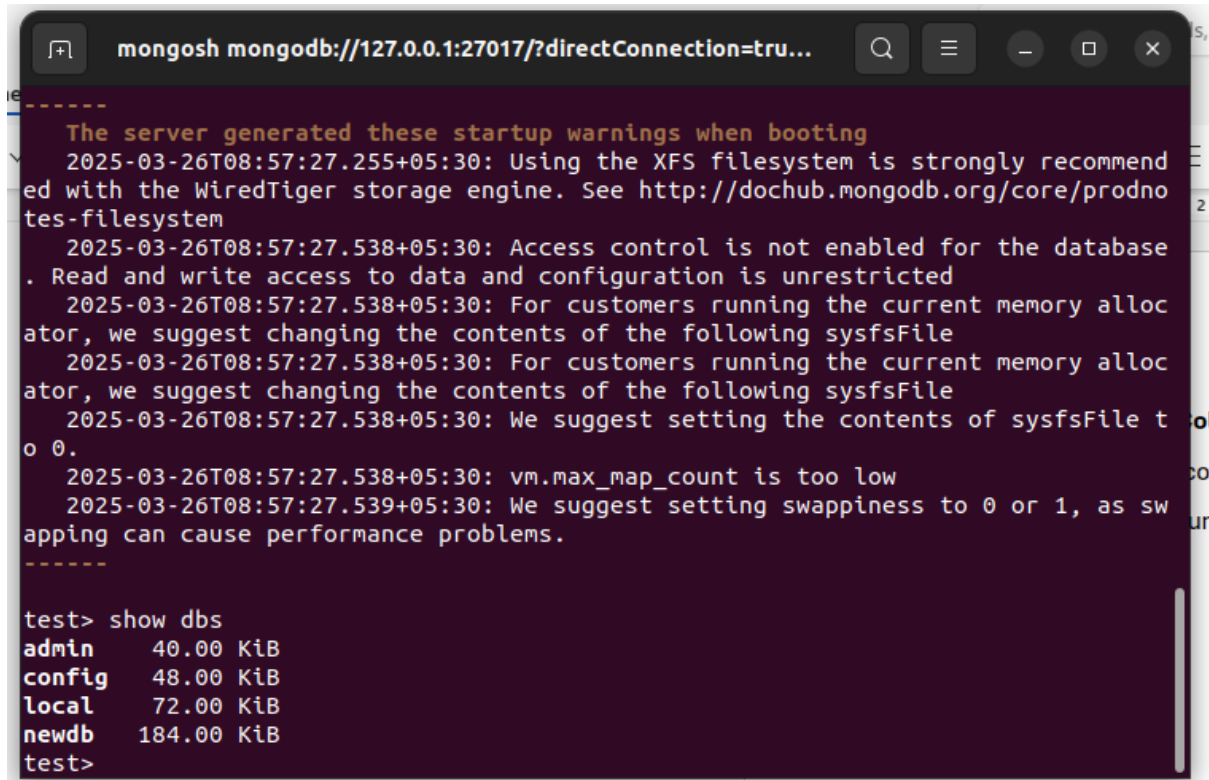


What is a Database and Collection in MongoDB?

Database: A container for collections, similar to a database in SQL.

Collection: A group of documents, like a table in SQL.



```
mongosh mongodb://127.0.0.1:27017/?directConnection=tru...

-----
The server generated these startup warnings when booting
2025-03-26T08:57:27.255+05:30: Using the XFS filesystem is strongly recommend
ed with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodno
tes-filesystem
2025-03-26T08:57:27.538+05:30: Access control is not enabled for the database
. Read and write access to data and configuration is unrestricted
2025-03-26T08:57:27.538+05:30: For customers running the current memory alloc
ator, we suggest changing the contents of the following sysfsFile
2025-03-26T08:57:27.538+05:30: For customers running the current memory alloc
ator, we suggest changing the contents of the following sysfsFile
2025-03-26T08:57:27.538+05:30: We suggest setting the contents of sysfsFile t
o 0.
2025-03-26T08:57:27.538+05:30: vm.max_map_count is too low
2025-03-26T08:57:27.539+05:30: We suggest setting swappiness to 0 or 1, as sw
apping can cause performance problems.
-----

test> show dbs
admin      40.00 KiB
config     48.00 KiB
local      72.00 KiB
newdb      184.00 KiB
test>
```

Show dbs – Create or switch to a database

db.createCollection("newdb") - Create a collection

Document in MongoDB

A document is the basic unit of data storage in MongoDB. It is a JSON-like structure that contains field-value pairs.

```
{
  "_id": ObjectId("507f1f77bcf86cd799439011"),
  "name": "Brindha",
}
```

Documents -

A **document** is the basic unit of data storage in MongoDB. It is a JSON-like structure that contains field-value pairs.

```
newdb> db.newproducts.find()
```

```
[
  { _id: ObjectId('67e3a4198ea3f781676b140b'), name: 'Laptop' },
  {
    _id: ObjectId('67e3a56a8ea3f781676b140c'),
    name: 'Laptop',
    price: 34300
  },
  {
    _id: ObjectId('67e3a5908ea3f781676b140d'),
    name: 'Laptop',
    price: 34300
  },
  {
    _id: ObjectId('67e3bfc58ea3f781676b140f'),
    name: 'Laptop',
    price: 34300
  },
  {
    _id: ObjectId('67e3bfc58ea3f781676b1410'),
    name: 'Laptop1',
    price: 34300
  },
  {
    _id: ObjectId('67e3bff68ea3f781676b1411'),
    name: 'Laptop1',
    price: 34300
  }
]
```

What is BSON?

- BSON (Binary JSON) is the internal storage format of MongoDB.
- It supports additional data types like Date, Decimal128, and Binary.

```
{_id: new ObjectId('67e3a4198ea3f781676b140b'), name: 'Laptop' },  
{_id: new ObjectId('67e3a56a8ea3f781676b140c'), name: 'Laptop', price: 34300 },  
{_id: new ObjectId('67e3a5908ea3f781676b140d'), name: 'Laptop', price: 34300 },  
{_id: new ObjectId('67e3bfc58ea3f781676b140f'), name: 'Laptop', price: 34300 },  
{_id: new ObjectId('67e3bfc58ea3f781676b1410'), name: 'Laptop1', price: 34300 },  
{_id: new ObjectId('67e3bff68ea3f781676b1411'), name: 'Laptop1', price: 34300 };
```

Bson:

```
[  
  {_id: ObjectId("67e3a4198ea3f781676b140b"), name: "Laptop" },  
  {_id: ObjectId("67e3a56a8ea3f781676b140c"), name: "Laptop", price:  
    NumberInt(34300) },  
  {_id: ObjectId("67e3a5908ea3f781676b140d"), name: "Laptop", price:  
    NumberInt(34300) },  
  {_id: ObjectId("67e3bfc58ea3f781676b140f"), name: "Laptop", price:  
    NumberInt(34300) },  
  {_id: ObjectId("67e3bfc58ea3f781676b1410"), name: "Laptop1", price:  
    NumberInt(34300) },  
  {_id: ObjectId("67e3bff68ea3f781676b1411"), name: "Laptop1", price:  
    NumberInt(34300) }  
]
```

MongoDB installation:

In Linux:

sudo apt install -y mongodb-org

After installation, start the MongoDB service:

sudo systemctl start mongod

To check mongodb running status

`sudo systemctl status mongod`

Version

`mongod --version`

Connect to shell

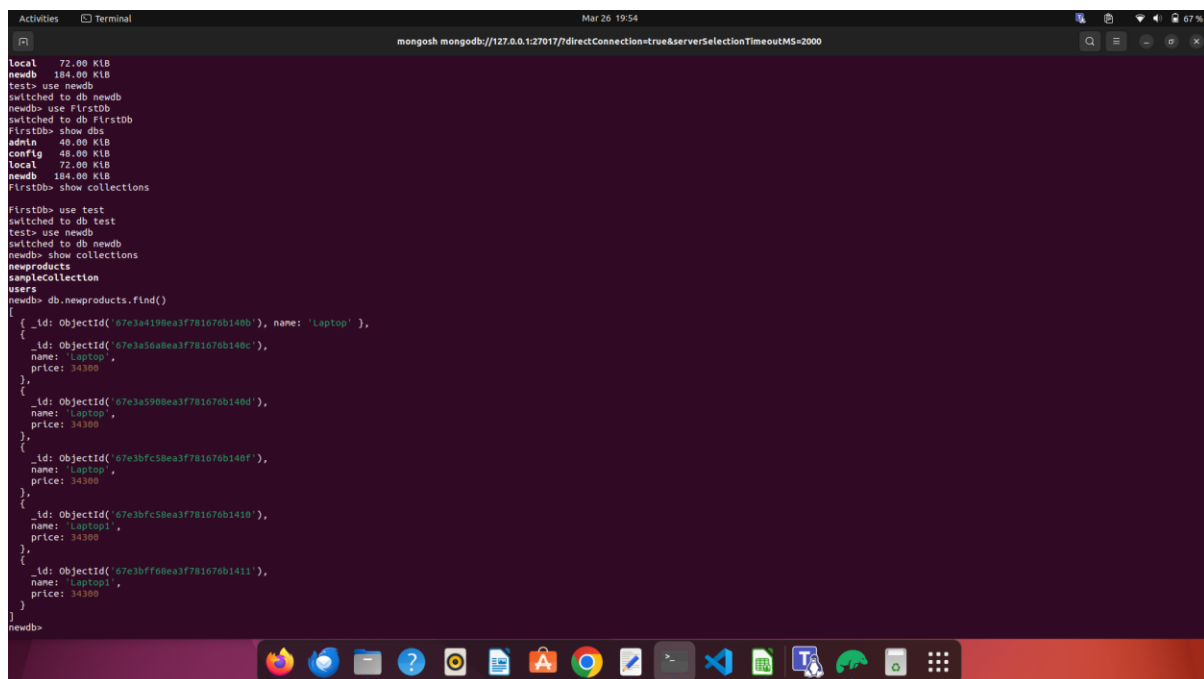
Mongosh

Restart the mongodb shell

`sudo systemctl restart mongod`

Commands;

`Db.newproducts.find()` - Retrieves all the data from the collection



```
local 72.00 KiB
newdb 184.00 KiB
test> use newdb
switched to db.newdb
newdb> use FirstDb
switched to db.FirstDb
FirstDb> show dbs
admin 48.00 KiB
config 48.00 KiB
local 72.00 KiB
newdb 184.00 KiB
FirstDb> show collections
FirstDb> use test
switched to db.test
test> use newdb
switched to db.newdb
newdb> show collections
newproducts
sampleCollection
users
newdb> db.newproducts.find()
[
  {
    _id: ObjectId('67e3a4198ea3f781676b148b'), name: 'Laptop' },
  {
    _id: ObjectId('67e3a56a8ea3f781676b148c'),
    name: 'Laptop',
    price: 34300
  },
  {
    _id: ObjectId('67e3a5988ea3f781676b148d'),
    name: 'Laptop',
    price: 34300
  },
  {
    _id: ObjectId('67e3bfc58ea3f781676b148f'),
    name: 'Laptop',
    price: 34300
  },
  {
    _id: ObjectId('67e3bfc58ea3f781676b1410'),
    name: 'Laptop1',
    price: 34300
  },
  {
    _id: ObjectId('67e3bff68ea3f781676b1411'),
    name: 'Laptop1',
    price: 34300
  }
]
```

How to Connect to MongoDB using a Connection String?

Local Connection

mongodb://localhost:27017

Remote Connection

mongodb://username:password@host:port/database

Example in Node.js:

```
const mongoose = require('mongoose');
mongoose.connect("mongodb://localhost:27017/newdb", { useNewUrlParser: true,
useUnifiedTopology: true})
  .then(() => console.log ("Connected to MongoDB"))
  .catch (err => console.log(err));
```

Database & Collection in MongoDB

Database in MongoDB

```
use newdb;
```

Create collection

```
db.createCollection("users");
```

Inserting Data into a Collection (Creates Collection If Not Exists)

```
db.newproducts.insertOne({name: "Mobile", price:25,000.00});
```

