

# MONGO DB (part-1)

- open cmd or gitbash.....

**Write 'mongosh'**

```
C:\Users\Vandan Raval>mongosh
Current Mongosh Log ID: 6470a48d2c8f8acc1022d055
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.9.1
Using MongoDB:      6.0.6
Using Mongosh:       1.9.1

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

-----
The server generated these startup warnings when booting
2023-05-26T17:01:09.084+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
-----
```

- Show database using...

**'show dbs' or we can also write 'show databases'**

```
test> show dbs;
admin      40.00 KiB
config    108.00 KiB
local      72.00 KiB
test> show databases
admin      40.00 KiB
config    108.00 KiB
local      72.00 KiB
test> _
```

- Use a particular database....

Command is → **'use dbname'**

```
test> use admin
switched to db admin
admin> use local;
switched to db local
local>
```

- Print all collection in the current database....

So first you have to select the database using **'use dbname'**

When database selected then you can show collections using

**'show collections;'**

```
local> show collections;
startup_log
local>
```

- To print all the document from current collection you can use this syntax.....

**'db.collection\_name.find();'**

```
admin> use local;
switched to db local
local> show collections;
startup_log
local> db.startup_log.find();
[
  {
    _id: 'VandanRaaval-2002-1685100544631',
    hostname: 'VandanRaaval-2002',
    startTime: ISODate("2023-05-26T11:29:04.000Z"),
    startTimeLocal: 'Fri May 26 16:59:04.632',
    cmdline: {
      config: 'C:\\Program Files\\MongoDB\\Server\\6.0\\bin\\mongod.cfg',
      net: { bindIp: '127.0.0.1', port: 27017 },
      service: true,
      storage: {
        dbPath: 'C:\\Program Files\\MongoDB\\Server\\6.0\\data',
        journal: { enabled: true }
      },
      systemLog: {
        destination: 'file',
        logAppend: true,
        path: 'C:\\Program Files\\MongoDB\\Server\\6.0\\log\\mongod.log'
      }
    },
    pid: Long("14476"),
    buildinfo: {
      version: '6.0.6',
      gitVersion: '26b4851a412cc8b9b4a18cdb6cd0f9f642e06aa7',
      targetMinOS: 'Windows 7/Windows Server 2008 R2',
      modules: [],
      allocator: 'tcmalloc',

```

- To create a database we can write **'use'** command creates a new database if there is no already present db with the same name, otherwise if there is a db with the same name it just select.....

**'use newdbname';**

- ✓ after creating a db ,if we try to do show db then it will not list our newly created database....
- ✓ so you will have to first create a collection under the database...if mongo db sees that there is no valid collection added in the database, and the db is empty ,it doesn't display.

```
local> show dbs;
admin      40.00 KiB
config    108.00 KiB
local      72.00 KiB
local> use First_demo;
switched to db First_demo
First_demo> show dbs
admin      40.00 KiB
config    108.00 KiB
local      72.00 KiB
First_demo> _
```

- **SO how to add new collection....?**

You can use a this syntax....

**db.createCollection("name\_of\_the\_collection")**

- make sure you can we execute this command after use **some\_db\_name.**

```
local> use First_demo
switched to db First_demo
First_demo> show dbs
admin      40.00 KiB
config    108.00 KiB
local      72.00 KiB
First_demo> db.createCollection("userDetails")
{ ok: 1 }
First_demo> show dbs
First_demo   8.00 KiB
admin        40.00 KiB
config       108.00 KiB
local        72.00 KiB
First_demo> _
```

- In normal RDBMS, while create a table, we have to define what will be the column of the table. Why we are not defining properties of a collection in mongodb??

This is because, mongodb dose not restrict us by any means for defining documents of a collection.

Two documents of the same collection can posses different types of properties....

- How to add new record in collection??

```
db.collection_name.insertOne({key1:"value1", key2:"value2",  
key3:"value3", key4:"value4"});
```

```
First_demo> db.createCollection("userDetails")  
{ ok: 1 }  
First_demo> show dbs  
First_demo 8.00 KiB  
admin       40.00 KiB  
config      108.00 KiB  
local       72.00 KiB  
First_demo> db.userDetails.insertOne({name:"vandan Raval",age:21,DOB:"04-08-2002",education:"persuing BE in computer Engineering"});  
{ acknowledged: true,  
  insertedId: ObjectId("6470bd4c2c8f8acc1022d056")  
}  
First_demo> db.userDetails.find()  
{  
  {  
    _id: ObjectId("6470bd4c2c8f8acc1022d056"),  
    name: 'vandan Raval',  
    age: 21,  
    DOB: '04-08-2002',  
    education: 'persuing BE in computer Engineering'  
  }  
}  
First_demo>
```

```
First_demo> db.createCollection("userDetails")
```

```
{ ok: 1 }
```

```
First_demo> show dbs
```

```
First_demo 8.00 KiB
```

```
admin 40.00 KiB
```

```
config 108.00 KiB
```

```
local 72.00 KiB
```

```
First_demo> db.userDetails.insertOne({name:"vandan Raval",age:21,DOB:"04-08-2002",education:"persuing BE in computer Engineering"});
```

```
{ acknowledged: true,  
  insertedId: ObjectId("6470bd4c2c8f8acc1022d056")  
}
```

```
First_demo> db.userDetails.find()
```

```
[  
  {  
    _id: ObjectId("6470bd4c2c8f8acc1022d056"),  
    name: 'vandan Raval',  
    age: 21,  
    DOB: '04-08-2002',  
    education: 'persuing BE in computer Engineering'  
  }  
]
```

```
First_demo>
```