



28 August

```
PS C:\Users\prach> mongod --version
db version v7.0.14
Build Info: {
  "version": "7.0.14",
  "gitVersion": "ce59cfc6a3c5e5c067dca0d30697edd68d4f5188",
  "modules": [],
  "allocator": "tcmalloc",
  "environment": {
    "distmod": "windows",
    "distarch": "x86_64",
    "target_arch": "x86_64"
  }
}
```

To check version of database.

```
PS C:\Users\prach> mongosh --version
2.3.0
```

To check version of shell

```

PS C:\Users\prach> mongosh
Current Mongosh Log ID: 66d1830afcb7ad1f632710bb
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true
&serverSelectionTimeoutMS=2000&appName=mongosh+2.3.0
Using MongoDB:      7.0.14
Using Mongosh:      2.3.0

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

-----
  The server generated these startup warnings when booting
  2024-08-30T10:42:12.462+05:30: Access control is not enabled for the
  database. Read and write access to data and configuration is unrestrict
  ed
  -----

test>

```

Mongo Shell is started

```

test> use Avengers
switched to db Avengers
Avengers> |

```

to create a database named Avengers.

```

Avengers> show dbs
admin      40.00 KiB
config     12.00 KiB
local      72.00 KiB

```

To show all databases

```

Avengers> db.createCollection("OG6")
{ ok: 1 }

```

To create a new empty collection "OG6".

▼ Collection

It is a fundamental organizational unit used to store documents.

```

Avengers> db.myNewCollection.insertOne({name:"IronMan"})
{
  acknowledged: true,
  insertedId: ObjectId('66d184b7fcb7ad1f632710bc')
}

```

Added a single data in the Collection.

```

Avengers> db.user.insertMany([ {name: "IronMan", dialogue:"I am Iron Man",
ogname: "Tony Stark"}, {name: "Captain America", dialogue:"I can do this all day",
ogname: "Steve Rogers"}, {name: "Hulk", dialogue: "Smash! Smash! Smash!!!",
ogname: "Bruce Banner"}])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('66d186c0fcb7ad1f632710bd'),
    '1': ObjectId('66d186c0fcb7ad1f632710be'),
    '2': ObjectId('66d186c0fcb7ad1f632710bf')
  }
}

```

```

Avengers> db.villain.insertMany([
... {name: "Loki", dialogue:"I am Loki from Asgard and I am burdened with a glorious purpose",
ogname: "God Of Mischief"},
... {name:"Thanos", dialogue:"jb tak log apne kl ko bhula nhi sakte tb tk vo apne aane waale kl ko apna nhi skte",
ogname: "Mad Titan"}])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('66d18834fcb7ad1f632710c0'),
    '1': ObjectId('66d18834fcb7ad1f632710c1')
  }
}

```

```

Avengers> db.guardians.insertMany([
... {name: "GROOT", dialogue:"I AM GROOT", ogname:"GROOT"},
... {name: "Rocket", dialogue:"Yeh bandook kitne mein dega nhi toh yeh haath de de",
ogname:"Rocket Raccon"}])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('66d188e3fcb7ad1f632710c2'),
    '1': ObjectId('66d188e3fcb7ad1f632710c3')
  }
}

```

```

Avengers> db.villain.updateMany(
...  {},
...  {$set: {subject: "Data Science"}}
...  )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 2,
  modifiedCount: 2,
  upsertedCount: 0
}

```

```

Avengers> db.villain.updateMany( {}, { $set: { ogname: "God Of Stories"
} } )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 2,
  modifiedCount: 2,
  upsertedCount: 0
}

```

to change one object in a collection
 set → changes the given value to derired value
 {} → match all docs

```

Avengers> db.villain.updateMany(
...  {},
...  {$set: {address: "Asgard"}}
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 2,
  modifiedCount: 2,
  upsertedCount: 0
}

```

added a field in a collection

```

Avengers> db.villain.find()
[
  {
    _id: ObjectId('66d18834fcb7ad1f632710c0'),
    name: 'Loki',
    dialogue: 'I am Loki from Asgard and I am burdened with a glorious purpose',
    ogname: 'God Of Stories',
    subject: 'Data Science',
    address: 'Asgard'
  },
  {
    _id: ObjectId('66d18834fcb7ad1f632710c1'),
    name: 'Thanos',
    dialogue: 'jb tak log apne kl ko bhula nhi sakte tb tk vo apne aane waale kl ko apna nhi skte',
    ogname: 'God Of Stories',
    subject: 'Data Science',
    address: 'Asgard'
  }
]

```

returns all documents in the specified collection

```

Avengers> db.villain.findOne()
{
  _id: ObjectId('66d18834fcb7ad1f632710c0'),
  name: 'Loki',
  dialogue: 'I am Loki from Asgard and I am burdened with a glorious purpose',
  ogname: 'God Of Stories',
  subject: 'Data Science',
  address: 'Asgard'
}

```

returns first document from the collection

```

Avengers> db.villain.find({ name: "Loki"})
[
  {
    _id: ObjectId('66d18834fcb7ad1f632710c0'),
    name: 'Loki',
    dialogue: 'I am Loki from Asgard and I am burdened with a glorious purpose',
    ogname: 'God Of Stories',
    subject: 'Data Science',
    address: 'Asgard'
  }
]

```

to search by a specific query

▼ ISODate()

It generates the current date and time

▼ insertOne()

It is used to insert a single document into the collection. It takes

```

Avengers> db.villain.insertOne({date: ISODate()})
{
  acknowledged: true,
  insertedId: ObjectId('66d19396fcb7ad1f632710c4')
}

```

Query Selectors

```

Avengers> db.student.updateMany(
...   {marks: { $gte: 91 } },
...   { $set: { grade: "A++" } }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}

```

```

Avengers> db.student.updateMany(
...   { marks: { $gte: 86, $lte: 90 } },
...   { $set: { grade: "A+" } }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}

```

one more example