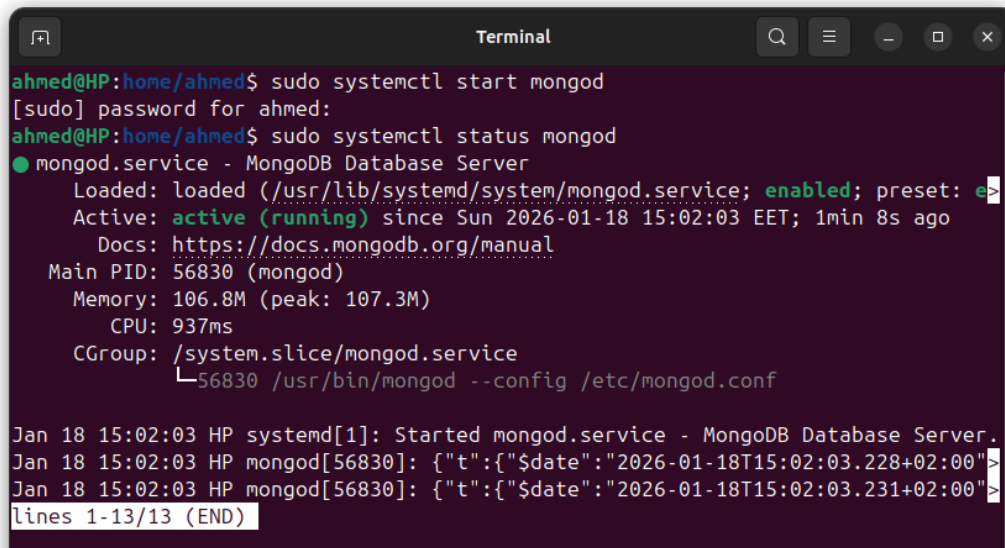


Ahmed Ehab Ahmed

MongoDB Lab1

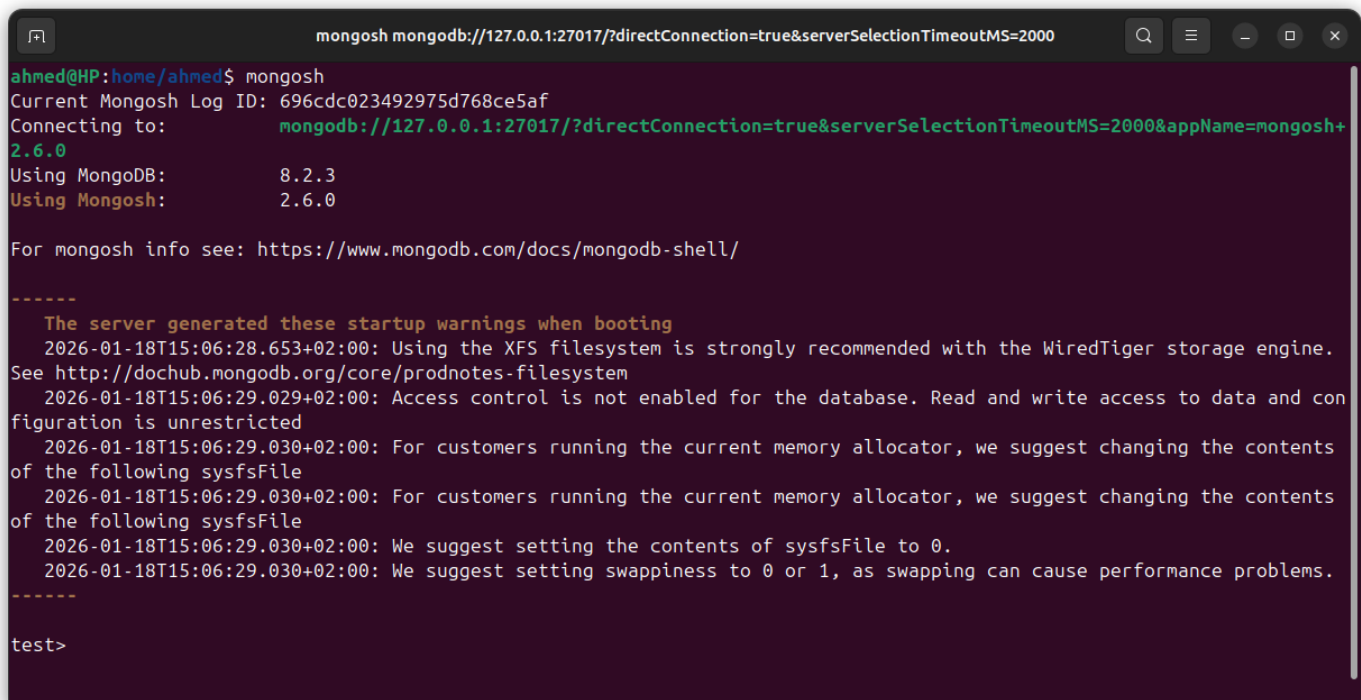
- 1- Install mongodb server and mongosh client
- 2- Add mongo DB folder to your system variables path
- 3- Open server (on windows shell using command “mongod”);



```
ahmed@HP:home/ahmed$ sudo systemctl start mongod
[sudo] password for ahmed:
ahmed@HP:home/ahmed$ sudo systemctl status mongod
● mongod.service - MongoDB Database Server
   Loaded: loaded (/usr/lib/systemd/system/mongod.service; enabled; preset: e>
   Active: active (running) since Sun 2026-01-18 15:02:03 EET; 1min 8s ago
     Docs: https://docs.mongodb.org/manual
    Main PID: 56830 (mongod)
      Memory: 106.8M (peak: 107.3M)
         CPU: 937ms
    CGroup: /system.slice/mongod.service
            └─56830 /usr/bin/mongod --config /etc/mongod.conf

Jan 18 15:02:03 HP systemd[1]: Started mongod.service - MongoDB Database Server.
Jan 18 15:02:03 HP mongod[56830]: {"t":{"$date":"2026-01-18T15:02:03.228+02:00"}>
Jan 18 15:02:03 HP mongod[56830]: {"t":{"$date":"2026-01-18T15:02:03.231+02:00"}>
lines 1-13/13 (END)
```

- 4- Open another shell for client using command “mongosh”



```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
ahmed@HP:home/ahmed$ mongosh
Current Mongosh Log ID: 696cdc023492975d768ce5af
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+
2.6.0
Using MongoDB:      8.2.3
Using Mongosh:      2.6.0

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

-----
  The server generated these startup warnings when booting
  2026-01-18T15:06:28.653+02:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine.
See http://dochub.mongodb.org/core/prodnotes-filesystem
  2026-01-18T15:06:29.029+02:00: Access control is not enabled for the database. Read and write access to data and con
figuration is unrestricted
  2026-01-18T15:06:29.030+02:00: For customers running the current memory allocator, we suggest changing the contents
of the following sysfsFile
  2026-01-18T15:06:29.030+02:00: For customers running the current memory allocator, we suggest changing the contents
of the following sysfsFile
  2026-01-18T15:06:29.030+02:00: We suggest setting the contents of sysfsFile to 0.
  2026-01-18T15:06:29.030+02:00: We suggest setting swappiness to 0 or 1, as swapping can cause performance problems.
-----
test>
```

- 5- Create database with name ITI ☹ by typing use ITI
- 6- start create new collection named instructors

```
test> use ITI
switched to db ITI
ITI> db.createCollection("instructors")
{ ok: 1 }
ITI>
```

- a- Insert many using array contained with lab folder instructors.txt file.
- b- Insert your own data
- c- Insert instructor without firstName and LastName (mongo will raise an error or not ?)

```
ITI> db.instructors.insertMany(instructorsArray)
{
  acknowledged: true,
  insertedIds: { '0': 6, '1': 7, '2': 8, '3': 9, '4': 10, '5': 2, '6': 3, '7': 4 }
}
ITI> db.instructors.insertOne({firstName: "Ahmed", lastName: "Ehab", age: 22, salary: 1200})
{
  acknowledged: true,
  insertedId: ObjectId('696ce08b12c3d5d6fc8ce5b0')
}
ITI> db.instructors.insertOne({age: 22, salary: 1200})
{
  acknowledged: true,
  insertedId: ObjectId('696ce0b212c3d5d6fc8ce5b1')
}
ITI>
```

- 7- Try the following queries: (all queries should be run using find or findOne)
- a- db.instructors.find({});

```
ITI> db.instructors.find({})
[
  {
    _id: 6,
    firstName: 'noha',
    lastName: 'hesham',
    age: 21,
    salary: 3500,
    address: { city: 'cairo', street: 10, building: 8 },
    courses: [ 'js', 'mvc', 'signalR', 'expressjs' ]
  },
  {
    _id: 7,
    firstName: 'mona',
    lastName: 'ahmed',
    age: 21,
```

b- db.instructors.find();

```
ITI> db.instructors.find()
[
  {
    _id: 6,
    firstName: 'noha',
    lastName: 'hesham',
    age: 21,
    salary: 3500,
    address: { city: 'cairo', street: 10, building: 8 },
    courses: [ 'js', 'mvc', 'signalR', 'expressjs' ]
  },
  {
    _id: 7,
    firstName: 'mona',
    lastName: 'ahmed',
    age: 21,
    salary: 3600,
    address: { city: 'cairo', street: 20, building: 8 },
    courses: [ 'js', 'mvc', 'signalR', 'expressjs' ]
  }
]
```

c- db.instructors.findOne();

d- db.instructors.find().constructor.name

```
ITI> db.instructors.findOne();
{
  _id: 6,
  firstName: 'noha',
  lastName: 'hesham',
  age: 21,
  salary: 3500,
  address: { city: 'cairo', street: 10, building: 8 },
  courses: [ 'js', 'mvc', 'signalR', 'expressjs' ]
}
ITI> db.instructors.find().constructor.name
Cursor
ITI> 
```

e- db.instructors.find({}).forEach((document)=>{
 print(document); })

f- display only firstName and lastName for each instructor?!

```
ITI> db.instructors.find({}).forEach((document)=>{
...   print(document.firstName, document.lastName); })
noha hesham
mona ahmed
mazen mohammed
ebtesam hesham
badr ahmed
mona mohammed
mazen ali
ebtesam ahmed
Ahmed Ehab
ITI> 
```