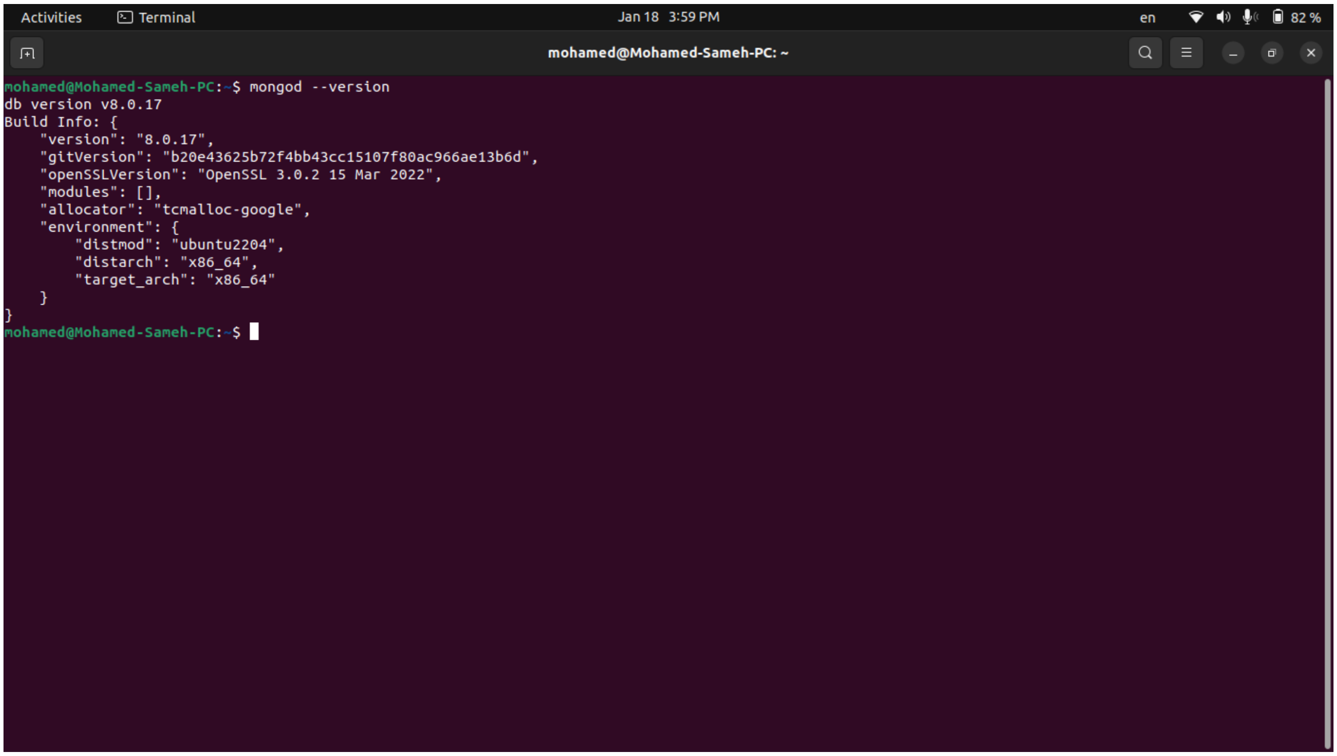
Mohamed Sameh Elkholy

MongoDB - Lab 1

1. Installation

Install MongoDB server and mongosh client



```
mohamed@Mohamed-Sameh-PC:~$ mongod --version
db version v8.0.17
Build Info: {
    "version": "8.0.17",
    "gitVersion": "b20e43625b72f4bb43cc15107f80ac966ae13b6d",
    "openSSLVersion": "OpenSSL 3.0.2 15 Mar 2022",
    "modules": [],
    "allocator": "tcmalloc-google",
    "environment": {
        "distmod": "ubuntu2204",
        "distarch": "x86_64",
        "target_arch": "x86_64"
    }
}
mohamed@Mohamed-Sameh-PC:~$
```

2. Add MongoDB to System Path

1. Go to Edit System environment variables
2. Navigate to Environment variables
3. Inside system variables, choose path
4. Add `C:\Program Files\MongoDB\Server\4.0\bin`
5. Apply and exit

3. Start MongoDB Server

Open server on Windows shell using command:

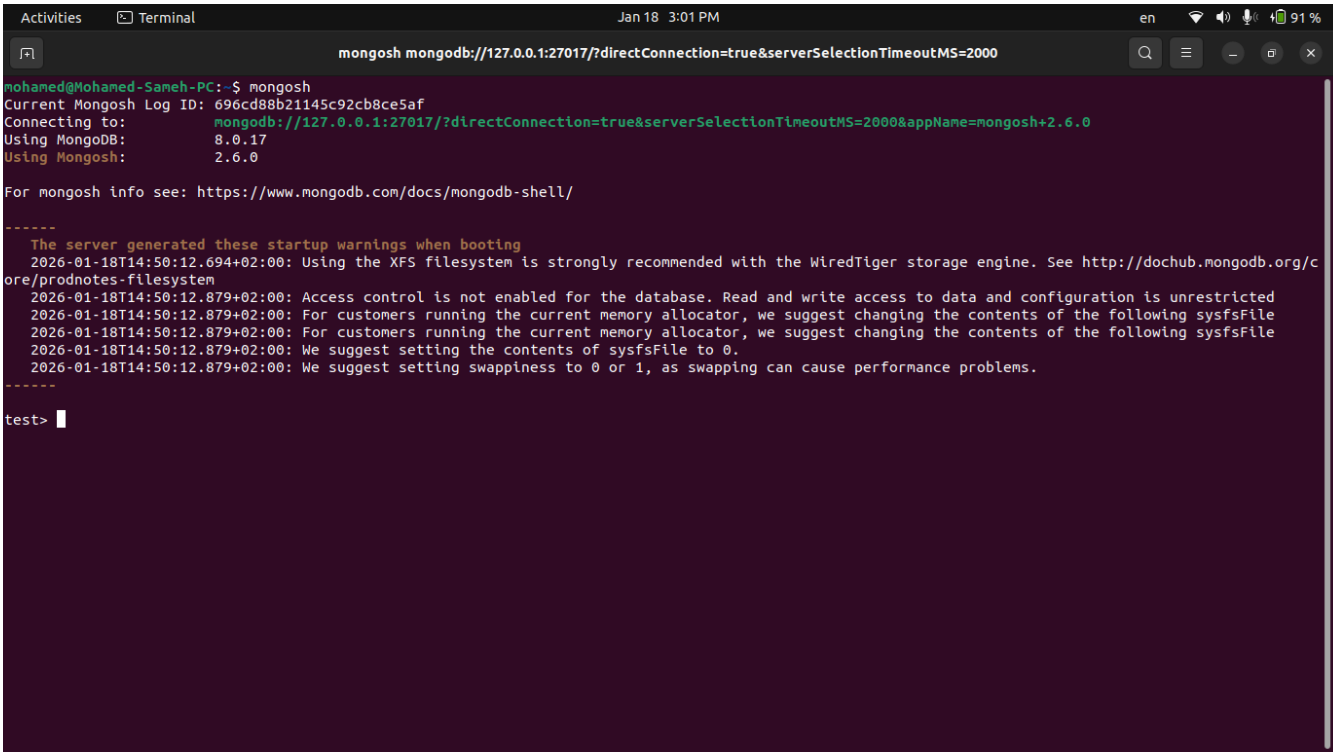
```
mongod
```

4. Start MongoDB Client

Open another shell for client using command:

```
mongosh
```

Or use noSqlBoost Tool



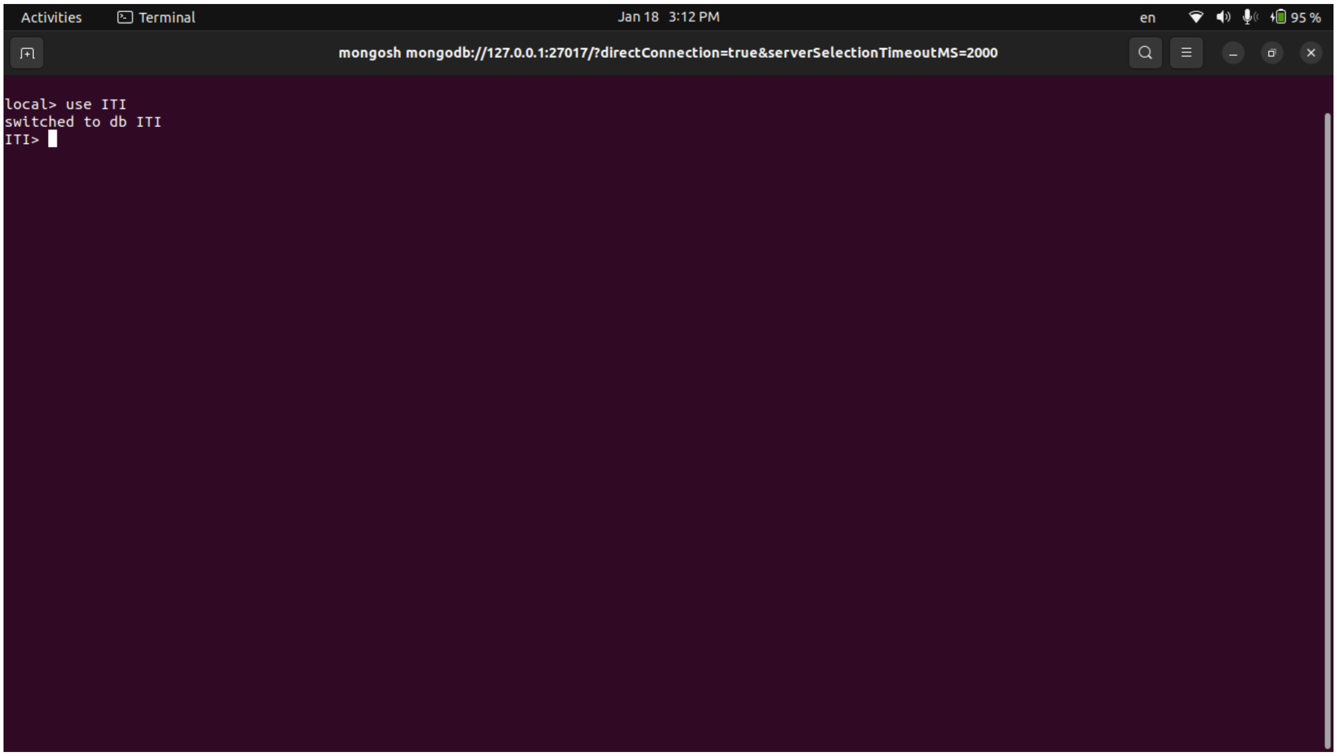
Activities
Terminal
Jan 18 3:01 PM
en
91 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
mohamed@Mohamed-Sameh-PC:~$ mongosh
Current Mongosh Log ID: 696cd88b21145c92cb8ce5af
Connecting to:          mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.6.0
Using MongoDB:          8.0.17
Using Mongosh:          2.6.0

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

------
   The server generated these startup warnings when booting
   2026-01-18T14:50:12.694+02:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/c
ore/prodnotes-filesystem
   2026-01-18T14:50:12.879+02:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
   2026-01-18T14:50:12.879+02:00: For customers running the current memory allocator, we suggest changing the contents of the following sysfsFile
   2026-01-18T14:50:12.879+02:00: For customers running the current memory allocator, we suggest changing the contents of the following sysfsFile
   2026-01-18T14:50:12.879+02:00: We suggest setting the contents of sysfsFile to 0.
   2026-01-18T14:50:12.879+02:00: We suggest setting swappiness to 0 or 1, as swapping can cause performance problems.
------

test> 
```

5. Create Database

Create database with name ITI by typing:

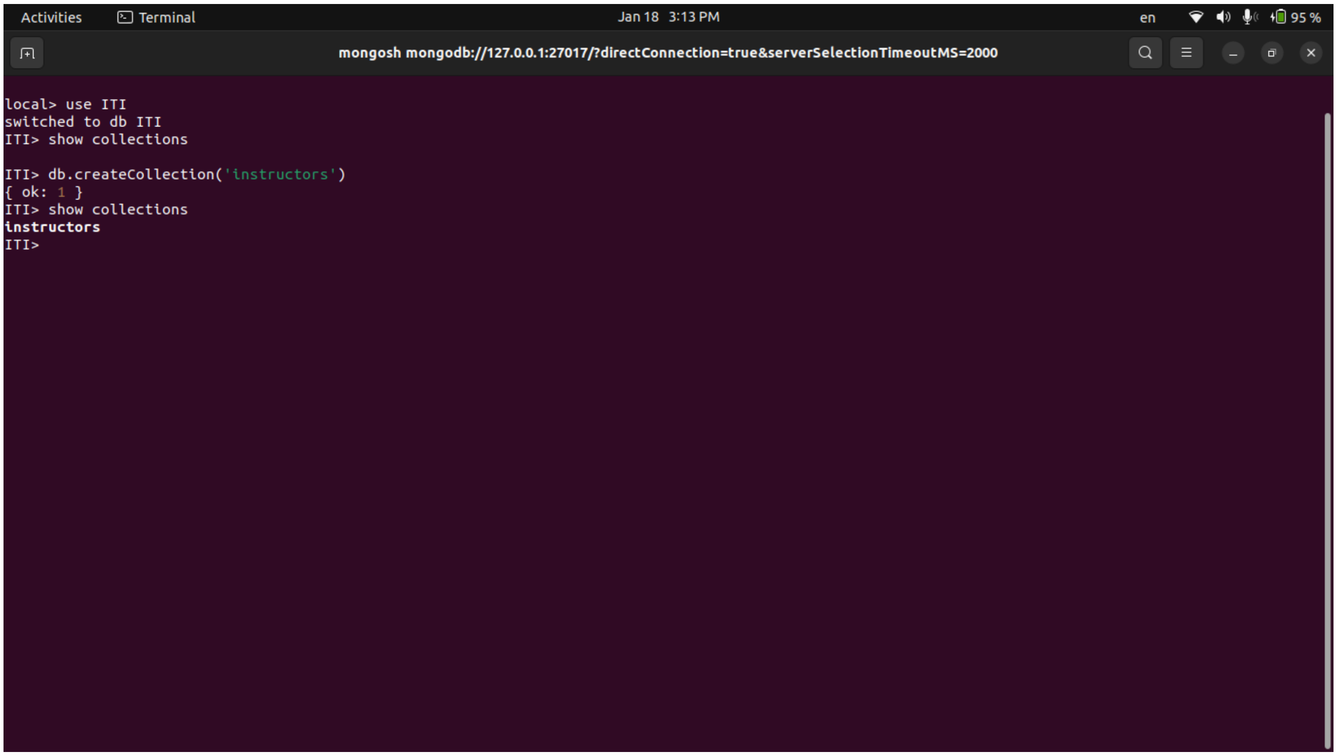
```
use ITI
```


Activities
Terminal
Jan 18 3:12 PM
en
95 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
local> use ITI
switched to db ITI
ITI> 
```

6. Create Collection and Insert Data

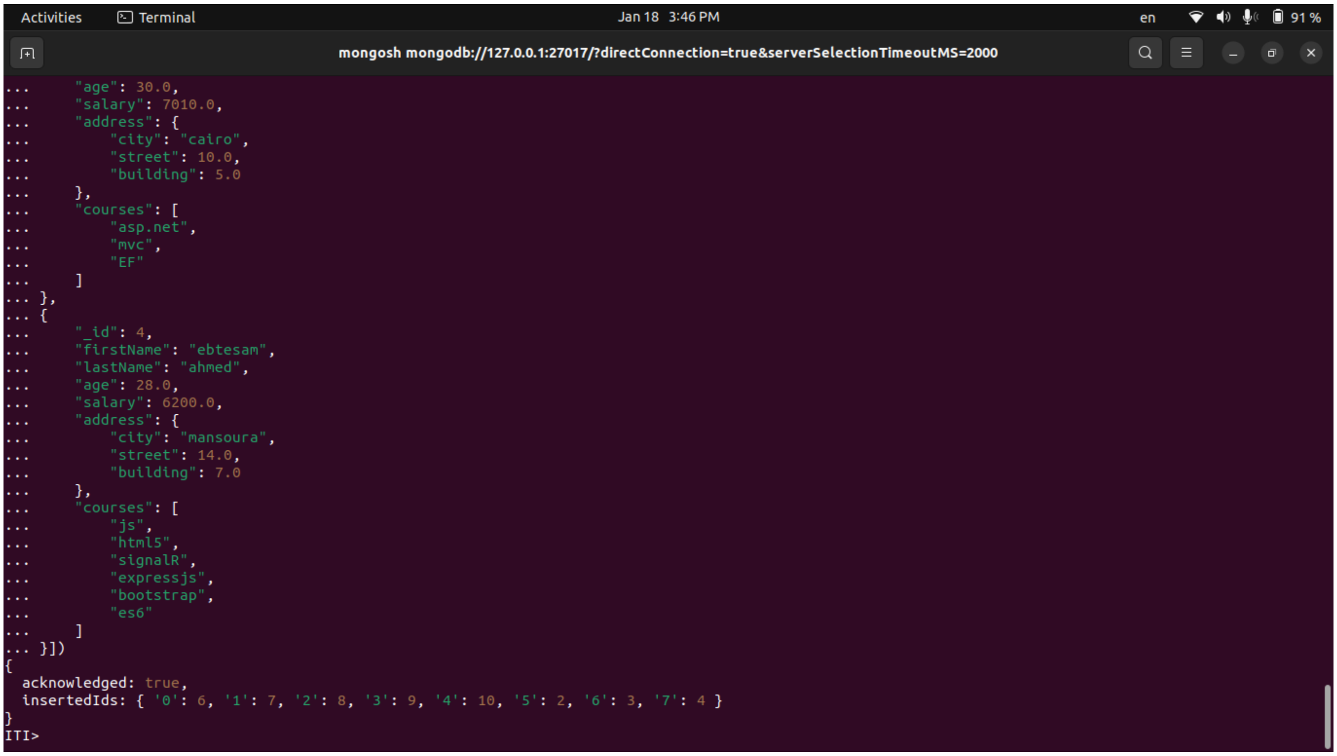
Start creating new collection named instructors and start with following inserts:


Activities
Terminal
Jan 18 3:13 PM
en
95 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
local> use ITI
switched to db ITI
ITI> show collections

ITI> db.createCollection('instructors')
{ ok: 1 }
ITI> show collections
instructors
ITI>
```

a. Insert Many

Insert many using array contained with lab folder instructors.txt file

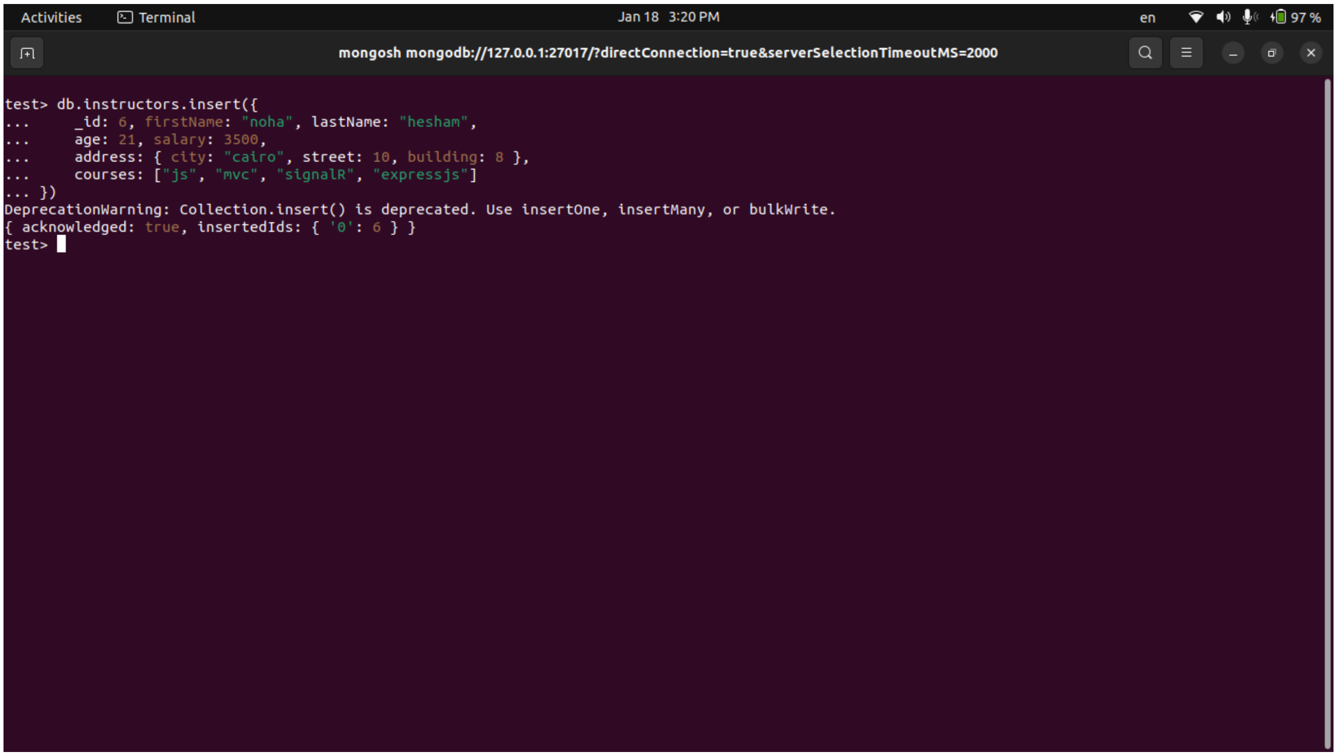


Activities
Terminal
Jan 18 3:46 PM
en
91 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
...     "age": 30.0,
...     "salary": 7010.0,
...     "address": {
...         "city": "cairo",
...         "street": 10.0,
...         "building": 5.0
...     },
...     "courses": [
...         "asp.net",
...         "mvc",
...         "EF"
...     ]
... },
... {
...     "_id": 4,
...     "firstName": "ebtesam",
...     "lastName": "ahmed",
...     "age": 28.0,
...     "salary": 6200.0,
...     "address": {
...         "city": "mansoura",
...         "street": 14.0,
...         "building": 7.0
...     },
...     "courses": [
...         "js",
...         "html5",
...         "signalR",
...         "expressjs",
...         "bootstrap",
...         "es6"
...     ]
... }])
{
  acknowledged: true,
  insertedIds: { '0': 6, '1': 7, '2': 8, '3': 9, '4': 10, '5': 2, '6': 3, '7': 4 }
}
ITI>
```

b. Insert Your Own Data

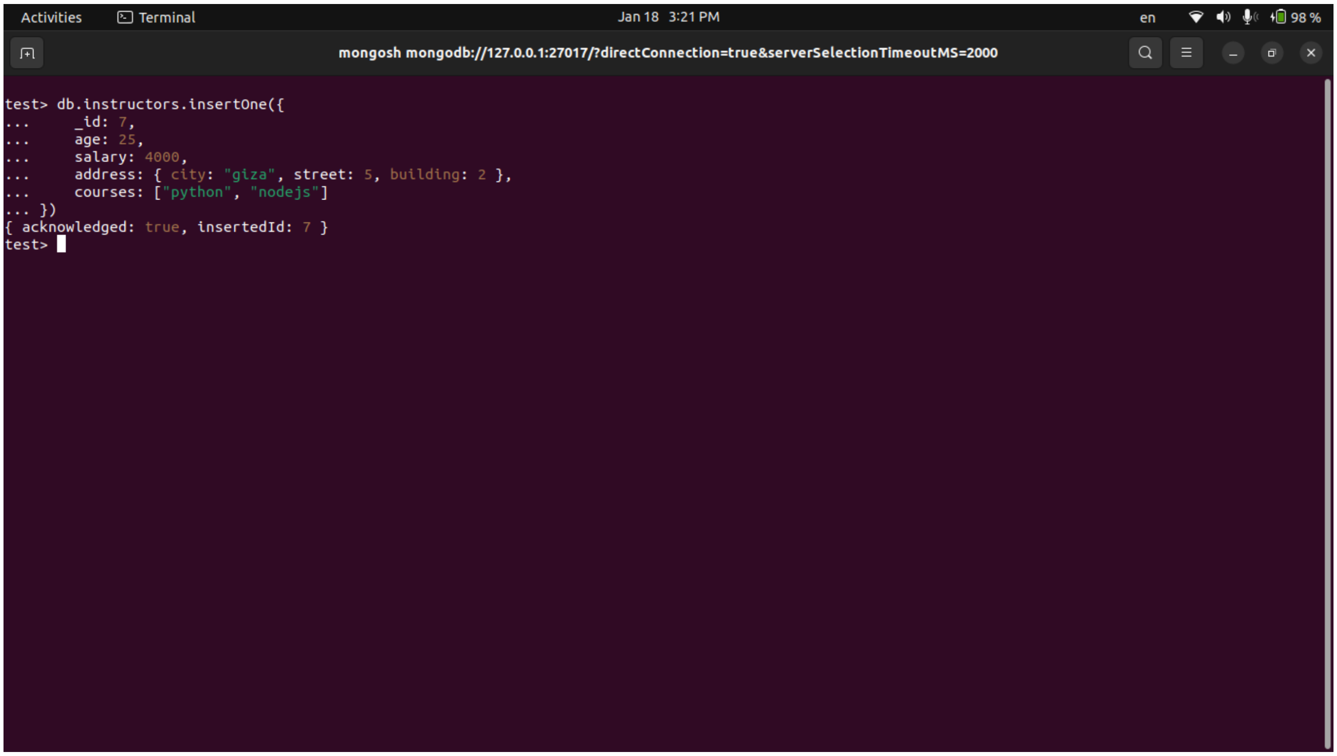
Insert your own data


Activities
Terminal
Jan 18 3:20 PM
en
97 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
test> db.instructors.insert({
...     _id: 6, firstName: "noha", lastName: "hesham",
...     age: 21, salary: 3500,
...     address: { city: "cairo", street: 10, building: 8 },
...     courses: ["js", "mvc", "signalR", "expressjs"]
... })
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{ acknowledged: true, insertedIds: { '0': 6 } }
test> 
```

c. Insert Without Required Fields

Insert instructor without firstName and lastName (mongo will raise an error or not?)


Activities
Terminal
Jan 18 3:21 PM
en
98 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

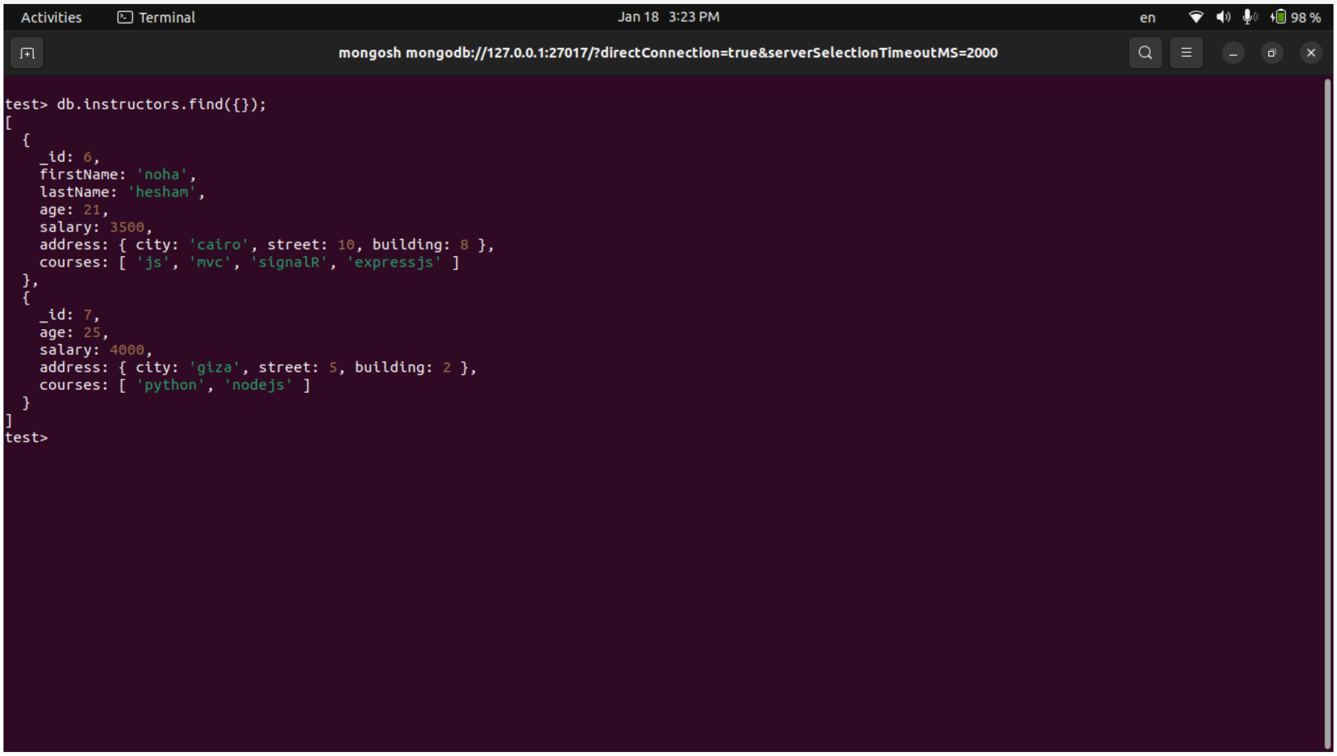
```
test> db.instructors.insertOne({
...     _id: 7,
...     age: 25,
...     salary: 4000,
...     address: { city: "giza", street: 5, building: 2 },
...     courses: ["python", "nodejs"]
... })
{ acknowledged: true, insertedId: 7 }
test> 
```

7. Queries

Try the following queries (all queries should be run using find or findOne):

a. Find all documents with empty object

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

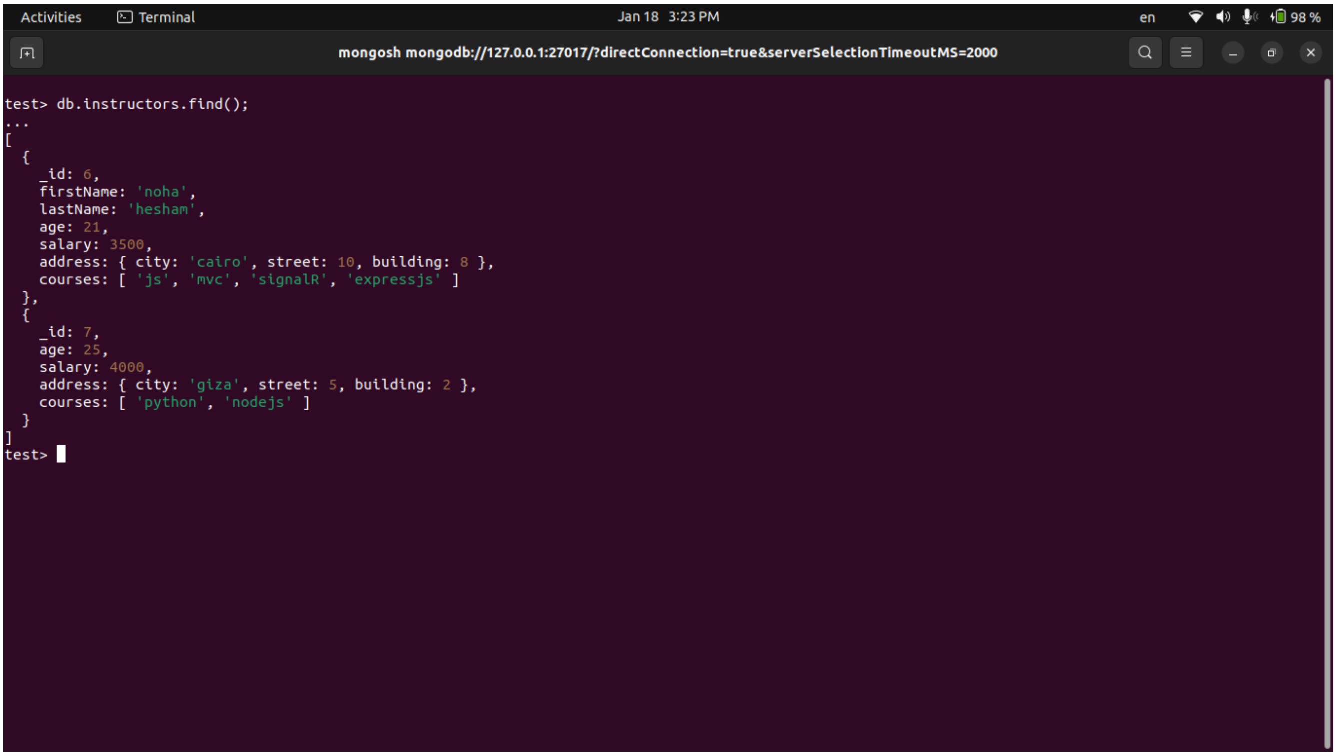


Activities
Terminal
Jan 18 3:23 PM
en
98 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
test> 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,
    age: 25,
    salary: 4000,
    address: { city: 'giza', street: 5, building: 2 },
    courses: [ 'python', 'nodejs' ]
  }
]
test>
```

b. Find all documents

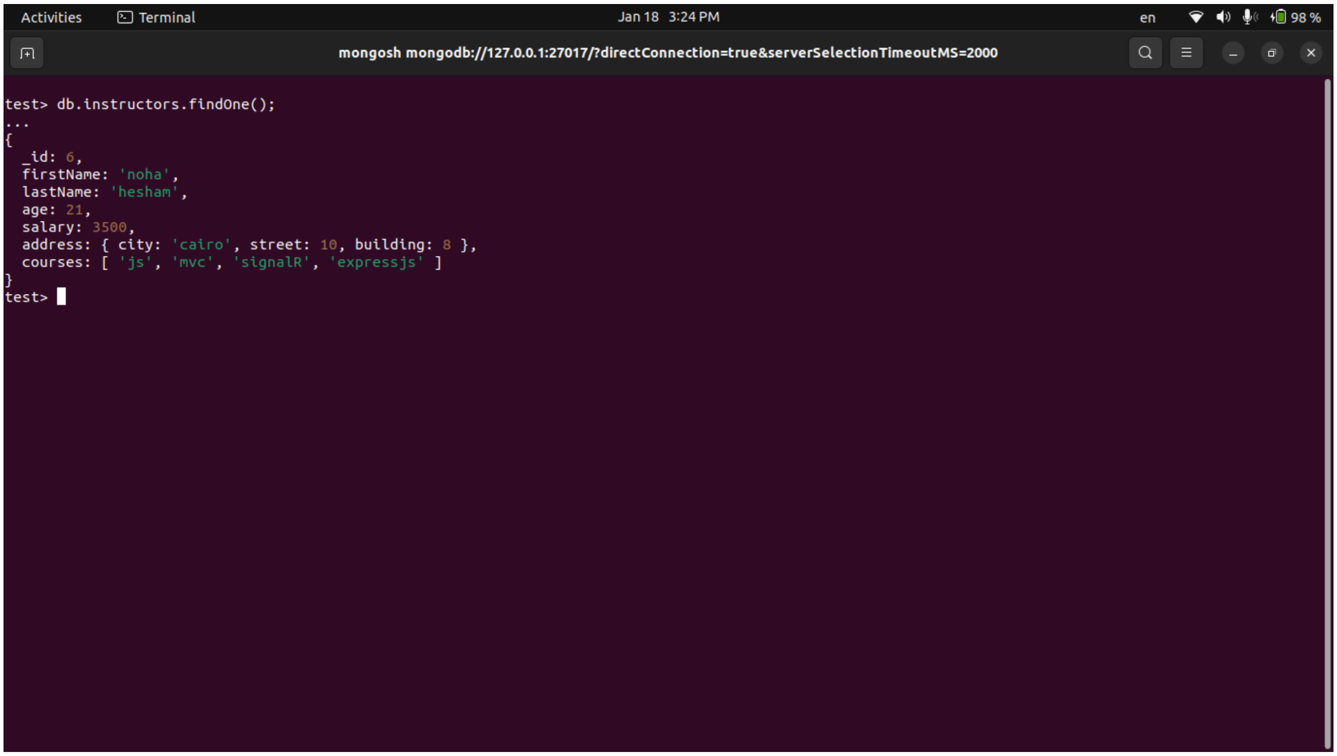
```
db.instructors.find();
```


Activities
Terminal
Jan 18 3:23 PM
en
98 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
test> 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,
    age: 25,
    salary: 4000,
    address: { city: 'giza', street: 5, building: 2 },
    courses: [ 'python', 'nodejs' ]
  }
]
test> 
```

c. Find one document

```
db.instructors.findOne();
```

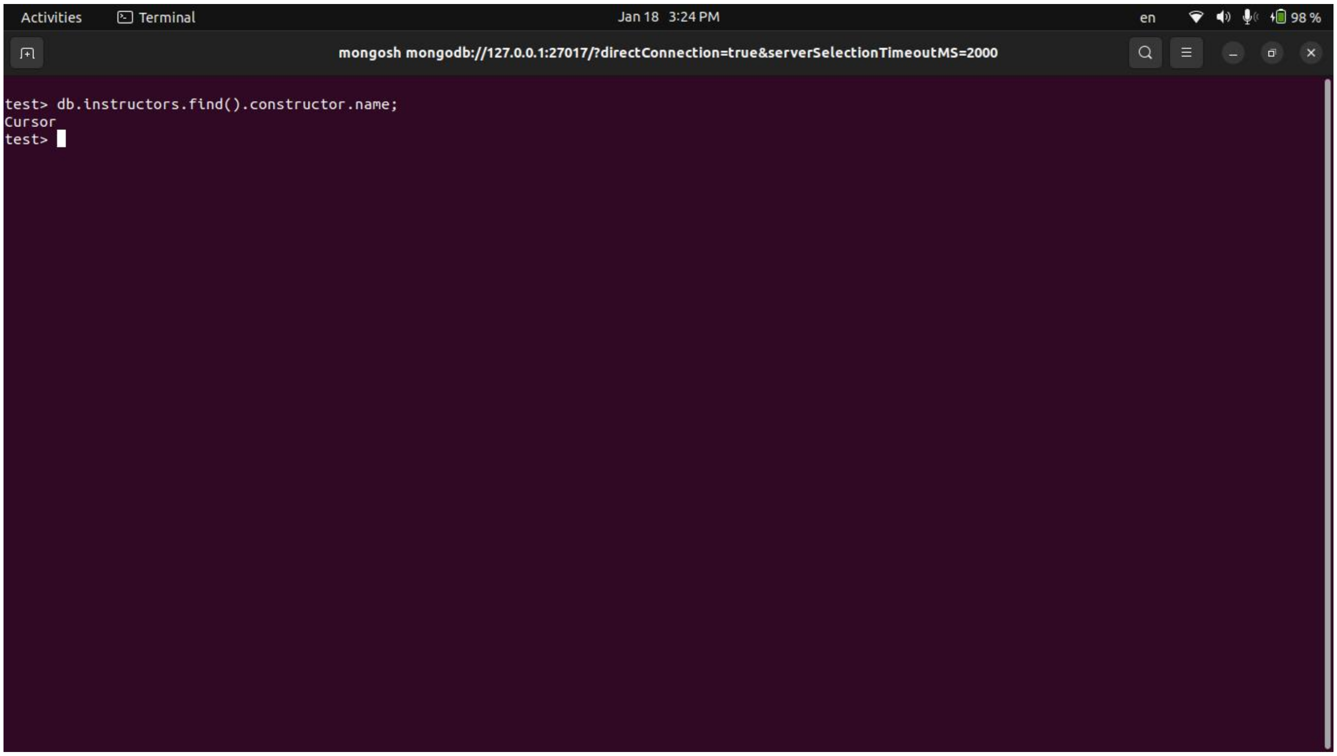


Activities
Terminal
Jan 18 3:24 PM
en
98 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
test> 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' ]
}
test> 
```

d. Get constructor name

```
db.instructors.find().constructor.name;
```

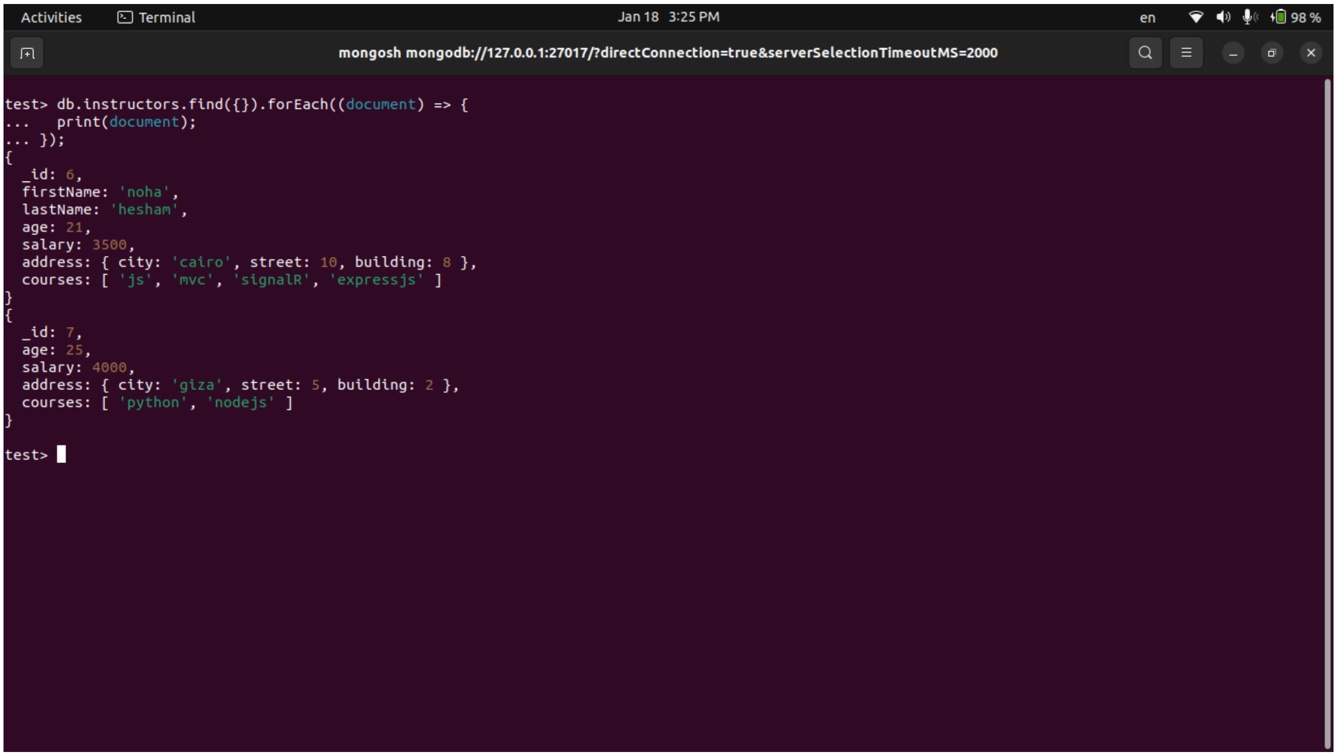


Activities
Terminal
Jan 18 3:24 PM
en
98 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
test> db.instructors.find().constructor.name;
Cursor
test> 
```

e. ForEach loop to print documents

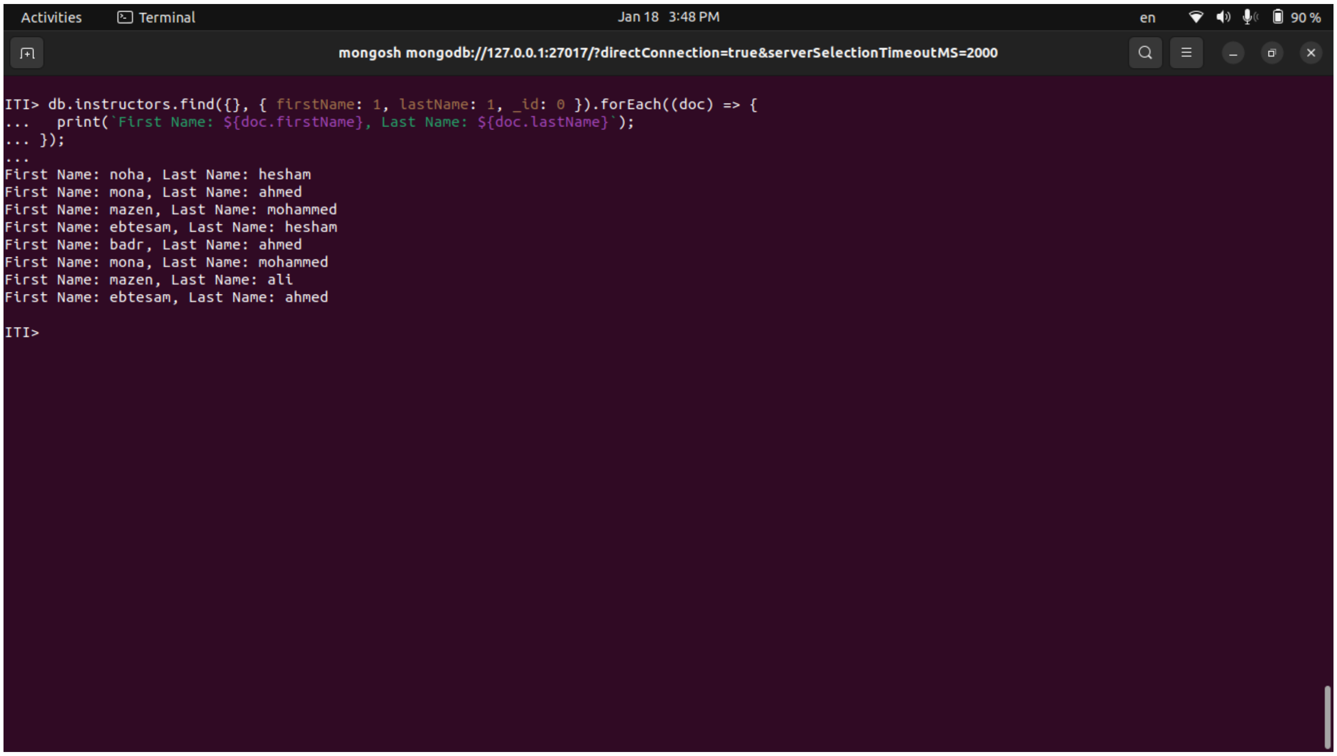
```
db.instructors.find({}).forEach((document) => {
   print(document);
});
```


Activities
Terminal
Jan 18 3:25 PM
en
98 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
test> db.instructors.find({}).forEach((document) => {
...    print(document);
... });
{
  _id: 6,
  firstName: 'noha',
  lastName: 'hesham',
  age: 21,
  salary: 3500,
  address: { city: 'cairo', street: 10, building: 8 },
  courses: [ 'js', 'mvc', 'signalR', 'expressjs' ]
}
{
  _id: 7,
  age: 25,
  salary: 4000,
  address: { city: 'giza', street: 5, building: 2 },
  courses: [ 'python', 'nodejs' ]
}

test> 
```

f. Display only firstName and lastName

Now from step e, try to display only firstName and lastName for each instructor?!


Activities
Terminal
Jan 18 3:48 PM
en
90 %
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
ITI> db.instructors.find({}, { firstName: 1, lastName: 1, _id: 0 }).forEach((doc) => {
...    print(`First Name: ${doc.firstName}, Last Name: ${doc.lastName}`);
... });
...
First Name: noha, Last Name: hesham
First Name: mona, Last Name: ahmed
First Name: mazen, Last Name: mohammed
First Name: ebtesam, Last Name: hesham
First Name: badr, Last Name: ahmed
First Name: mona, Last Name: mohammed
First Name: mazen, Last Name: ali
First Name: ebtesam, Last Name: ahmed

ITI> 
```