MongoDB_Lab1

Mariam khaled Saad

Open source

1 – open mongo shell and view the help

```
mongosh –help
```

2 – identify your current working database and show list of available databases

```
show dbs
```

3 – create a new database called iti and create a collection named “students”. Insert whatever data you want about yourself (include name and age in your details).

```
use iti
db.students.insert({name:"mariam",age:24})
```

4– show list of available databases. What did you notice ?

```
show dbs
```

5 – Insert un-structured or semi-structured data for 10 of your friends (include name and age in your details. The documents should have different types of data i.e. arrays, strings, documents, integers).

```
db.students.insertMany([
  {
    _id: 1,
    firstName: "mariam",
    lastName: "khaled",
    email: "mariam@abc.com",
     age:24,
    salary: 5000,
     tasks:["task1","task2"],
  },
  {
    _id: 2,
    firstName: "hager",
    lastName: "khaled",
    email: "hager@abc.com",
     age:24,
    salary: 8000,
```

```
    tasks:["task3","task2"],
},
{
   _id: 3,
   firstName: "alaa",
   lastName: "khaled",
   email: "alaa@abc.com",
    age:29,
   salary: 7500,
    tasks:["task1","task3"],
},
{
   _id: 4,
   firstName: "radwa",
   lastName: "J",
   email: "radwaj@abc.com",
    age:26,
   salary: 9000,
    tasks:["task1","task4"],
},
{
   _id: 5,
   firstName: "nada",
   lastName: "D",
   email: "nada@abc.com",
    age:25,
   salary: 4500,
    tasks:["task1","task2"],
},
{
   _id: 6,
   firstName: "ahmed",
   lastName: "B",
   email: "ahmed@abc.com",
    age:27,
   salary: 11000,
    tasks:["task4","task2"],
},
{
   _id: 7,
   firstName: "zoz",
   lastName: "B",
```

```
    email: "zoz@abc.com",
      age:28,
    salary: 11000,
      tasks:["task1","task3","task4"],
  },
  {
    _id: 8,
    firstName: "khaled",
    lastName: "B",
    email: "ahmed@abc.com",
      age:58,
    salary: 11000,
      tasks:["task3","task2"],
  },
  {
    _id: 9,
    firstName: "fatma",
    lastName: "B",
    email: "fatma@abc.com",
      age:58,
    salary: 11000,
      tasks:["task1","task2"],
  },

])
```

6 – Search for your object by name.

```
db.students.find( { firstName: "mariam" } )
```

7– Search for your friend(s) by age.

```
db.students.find({ "age" : 28})
```

8 – Search for all of your friends whose age is older than yours.

```
db.students.find({ "age" : { $gt: 24}})
```

9 – delete any of your friends by id.

```
db.students.remove({_id:ObjectId(5)})
```

10 – view all documents in students collection in a prettified format.
db.students.find().pretty()

11 – count all documents in students collection.
db.students.find().count()

---

part 2

1- Create database with name ems
db.createCollection("faculty")
2- Insert the following data into "faculty" collection

{ "name":"Krish", "age":35,"gender":"M","exp":10,subjects:["DS","C","OS"],"type":"Full Time","qualification":"M.Tech" },
{ "name":"Manoj", "age":38,"gender":"M","exp":12,subjects:["JAVA","DBMS"],"type":"Full Time", "qualification":"Ph.D"},
{ "name":"Anush", "age":32,"gender":"F","exp":8,subjects:["C","CPP"],"type":"Part Time","qualification":"M.Tech" },
{ "name":"Suresh", "age":40,"gender":"M","exp":9,subjects:["JAVA","DBMS","NETWORKING"],"type":"Full Time", "qualification":"Ph.D"},
{ "name":"Rajesh", "age":35,"gender":"M","exp":7,subjects:["DS","C","OS"],"type":"Full Time","qualification":"M.Tech" },
{ "name":"Mani", "age":38,"gender":"F","exp":10,subjects:["JAVA","DBMS","OS"],"type":"Part Time", "qualification":"Ph.D"},
{ "name":"Sivani", "age":32,"gender":"F","exp":8,subjects:["C","CPP","MATHS"],"type":"Part Time","qualification":"M.Tech" },
{ "name":"Nagesh", "age":39,"gender":"M","exp":11,subjects:["JAVA","DBMS","NETWORKING"],"type":"Full Time", "qualification":"Ph.D"},
{ "name":"Nagesh", "age":35,"gender":"M","exp":9,subjects:["JAVA",".Net","NETWORKING"],"type":"Full Time", "qualification":"Ph.D"},
{ "name":"Latha", "age":40,"gender":"F","exp":13,subjects:["MATHS"],"type":"Full Time", "qualification":"Ph.D"}

1. Get the details of all the faculty.

```
db.faculty.find()
```

2. Get the count of all faculty members.

```
db.faculty.find().count()
```

3. Get all the faculty members whose qualification is “Ph.D”.

```
db.faculty.find({qualification :"Ph.D"})
```

4. Get all the faculty members whose experience is between 8 to 12 years.

```
db.faculty.find({ "exp": {$gt:8, $lt:12}})
```

5. Get all the faculty members who teach “MATHS” or “NETWORKING”.

```
db.faculty.find({$or:[{subjects:"MATHS"},{subjects:"NETWORKING"}]})
```

6. Get all the faculty members who teach “MATHS” and whose age is more than 30 years and qualification must be “Ph.D”.

```
db.faculty.find({$and:[{subjects:"MATHS"},{age:{$gt:30}},{qualification:"Ph.D"}]})
```

7. Get all the faculty members who are working part-time or who teach “JAVA”.

```
db.faculty.find({$or:[{subjects:"JAVA"},{type:"Part Time"}]})
```

8. Add the following new faculty members:
{ "name":"Suresh Babu", "age":55,"gender":"M","exp":25,subjects: ["MATHS","DE"],"type":"Full Time", "qualification":"Ph.D"}

```
db.faculty.insert([
  { "name":"Suresh Babu", "age":55,"gender":"M","exp":25,subjects:
["MATHS","DE"],"type":"Full Time", "qualification":"Ph.D"}
])
```

9. Update the data of all faculty members by incrementing their age and exp by one year.

```
db.faculty.updateMany({}, { $inc: { age:1,exp:1 } })
```

10. Update the faculty “Sivani” with the following data: update qualification to “Ph.D” and type to “Full Time”.

```
db.faculty.update({name:"Sivani"}, { $set: {
   type:"Full Time",qualification:"Ph.D" }})
```

11. Update all faculty members who are teaching “MATHS” such that they should now also teach “PSK”.

```
db.faculty.updateMany({subjects:"MATHS"}, { $push: { subjects:"PSK"} })
```

12. Delete all faculty members whose age is more than 55 years.

```
db.faculty.deleteMany({age: {$gt: 55}})
```

13. Get only the name and qualification of all faculty members.

```
db.faculty.find({}, {name: 1, qualification:1 })
```

14. Get the name, qualification and exp of all faculty members and display the same in ascending order of exp.

```
db.faculty.find({}, {name: 1, qualification:1, exp: 1}).sort( { exp: 1 } )
```

15. Sort the faculty details by their age (descending order) and get the details of the first five faculty members only.

```
db.faculty.find().sort( { age: -1 } ).limit(5)
```