

1 – open mongo shell and view the help

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()

-----
test> help

Shell Help:

use          Set current database
show         'show databases': Print a list of all available databases.
             'show collections': Print a list of all collections for current database.
             'show profile': Prints system,profile information.
             'show users': Print a list of all users for current database.
             'show roles': Print a list of all roles for current database.
             'show log <type>': log for current connection, if type is not set uses 'global'
             'show logs': Print all logs.

exit         Quit the MongoDB shell with exit/exit()/exit
quit         Quit the MongoDB shell with quit/quit()
Mongo        Create a new connection and return the Mongo object. Usage: new Mongo(URI, options [optional])
connect      Create a new connection and return the Database object. Usage: connect(URI, username [optional], password [optional])
it           result of the last line evaluated; use to further iterate
version      Shell version
load         Loads and runs a JavaScript file into the current shell environment
enableTelemetry Enables collection of anonymous usage data to improve the mongosh CLI
disableTelemetry Disables collection of anonymous usage data to improve the mongosh CLI
passwordPrompt Prompts the user for a password
sleep        Sleep for the specified number of milliseconds
print        Prints the contents of an object to the output
printjson    Alias for print()
convertShardKeyToHashed Returns the hashed value for the input using the same hashing function as a hashed index.
cls          Clears the screen like console.clear()
isInteractive Returns whether the shell will enter or has entered interactive mode

For more information on usage: https://docs.mongodb.com/manual/reference/method
test>
```

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

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

exit         Quit the MongoDB shell with exit/exit()/exit
quit         Quit the MongoDB shell with quit/quit()
Mongo        Create a new connection and return the Mongo object. Usage: new Mongo(URI, options [optional])
connect      Create a new connection and return the Database object. Usage: connect(URI, username [optional], password [optional])
it           result of the last line evaluated; use to further iterate
version      Shell version
load         Loads and runs a JavaScript file into the current shell environment
enableTelemetry Enables collection of anonymous usage data to improve the mongosh CLI
disableTelemetry Disables collection of anonymous usage data to improve the mongosh CLI
passwordPrompt Prompts the user for a password
sleep        Sleep for the specified number of milliseconds
print        Prints the contents of an object to the output
printjson    Alias for print()
convertShardKeyToHashed Returns the hashed value for the input using the same hashing function as a hashed index
cls          Clears the screen like console.clear()
isInteractive Returns whether the shell will enter or has entered interactive mode

For more information on usage: https://docs.mongodb.com/manual/reference/method
test> db
test
test> show databases
admin      40.00 KiB
bookstore  72.00 KiB
config     108.00 KiB
local      40.00 KiB
test>
```

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).

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
2023-02-21T15:30:08.699+02:00: Access control is not enabled for the database. Read and write access to data and configuration is un...

-----

Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()

-----

test> use iti
switched to db iti
iti> db.createCollection("students")
{ ok: 1 }
iti> db.students.insertOne({name:"Ahmed",age:23})
{
  acknowledged: true,
  insertedId: ObjectId("63f50856e34e5cc67ce074c3")
}
iti>
```

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

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
{
  acknowledged: true,
  insertedId: ObjectId("63f50856e34e5cc67ce074c3")
}
iti> show collections
students
iti> db.students.find()
[
  { _id: ObjectId("63f50856e34e5cc67ce074c3"), name: 'Ahmed', age: 23 }
]
iti> show dbs
admin      40.00 KiB
bookstore  72.00 KiB
config     108.00 KiB
iti        40.00 KiB
local      40.00 KiB
iti>
```

The iti database appears

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).

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
iti> db.students.insertMany([({name: "Wafaa", age: 22, gender: "Female"}, {name: "Orabi", age: 21, hobbies: ["rub", "read"]}))
{
  acknowledged: true,
  insertedIds: {
    0: ObjectId("63f50a5ce34e5cc67ce074c4"),
    1: ObjectId("63f50a5ce34e5cc67ce074c5")
  }
}
iti> db.students.insertMany([({name: "Esraa", age: 20, gender: "Female"}, {name: "Bakry", age: 19, movies: ["101", "102"]}))
{
  acknowledged: true,
  insertedIds: {
    0: ObjectId("63f50a9fe34e5cc67ce074c6"),
    1: ObjectId("63f50a9fe34e5cc67ce074c7")
  }
}
iti> db.students.insertMany([({name: "Sara", age: 25, gender: "Female"}, {name: "Abdelrehim", age: 26, movies: ["103", "104"]}))
{
  acknowledged: true,
  insertedIds: {
    0: ObjectId("63f50adfe34e5cc67ce074c8"),
    1: ObjectId("63f50adfe34e5cc67ce074c9")
  }
}
iti> db.students.insertMany([({name: "Radwa", age: 27, gender: "Female"}, {name: "Malak", age: 28, movies: ["105", "106"]}))
{
  acknowledged: true,
  insertedIds: {
    0: ObjectId("63f50b2de34e5cc67ce074ca"),
    1: ObjectId("63f50b2de34e5cc67ce074cb")
  }
}
iti> db.students.insertMany([({name: "Karima", age: 17, gender: "Female"}, {name: "Salim", age: 16, movies: ["107", "108"]}))
{
  acknowledged: true,
  insertedIds: {
    0: ObjectId("63f50b57e34e5cc67ce074cc"),
    1: ObjectId("63f50b57e34e5cc67ce074cd")
  }
}
}
```

6 – Search for your object by name.

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
]
iti> db.students.findOne({name: "Ahmed"})
{ _id: ObjectId("63f50856e34e5cc67ce074c3"), name: 'Ahmed', age: 23 }
iti>
```

7– Search for your friend(s) by age.

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
> movies: [ 'les misérables', 'omg' ]
}
]
iti> db.students.findOne({name:"Ahmed"})
{ _id: ObjectId("63f50856e34e5cc67ce074c3"), name: 'Ahmed', age: 23 }
iti> db.students.find({age:27})
[
  {
    _id: ObjectId("63f50b2de34e5cc67ce074ca"),
    name: 'Radwa',
    age: 27,
    gender: 'Female'
  }
]
iti>
```

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

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
iti> db.students.find({age: {$gt:23}})
[
  {
    _id: ObjectId("63f50adfe34e5cc67ce074c8"),
    name: 'Sara',
    age: 25,
    gender: 'Female'
  },
  {
    _id: ObjectId("63f50adfe34e5cc67ce074c9"),
    name: 'Abdeirehin',
    age: 26,
    movies: [ 'lof', 'yes' ]
  },
  {
    _id: ObjectId("63f50b2de34e5cc67ce074ca"),
    name: 'Radwa',
    age: 27,
    gender: 'Female'
  },
  {
    _id: ObjectId("63f50b2de34e5cc67ce074cb"),
    name: 'Malak',
    age: 28,
    movies: [ 'La mante', 'Got' ]
  }
]
iti>
```

9 – delete any of your friends by id.

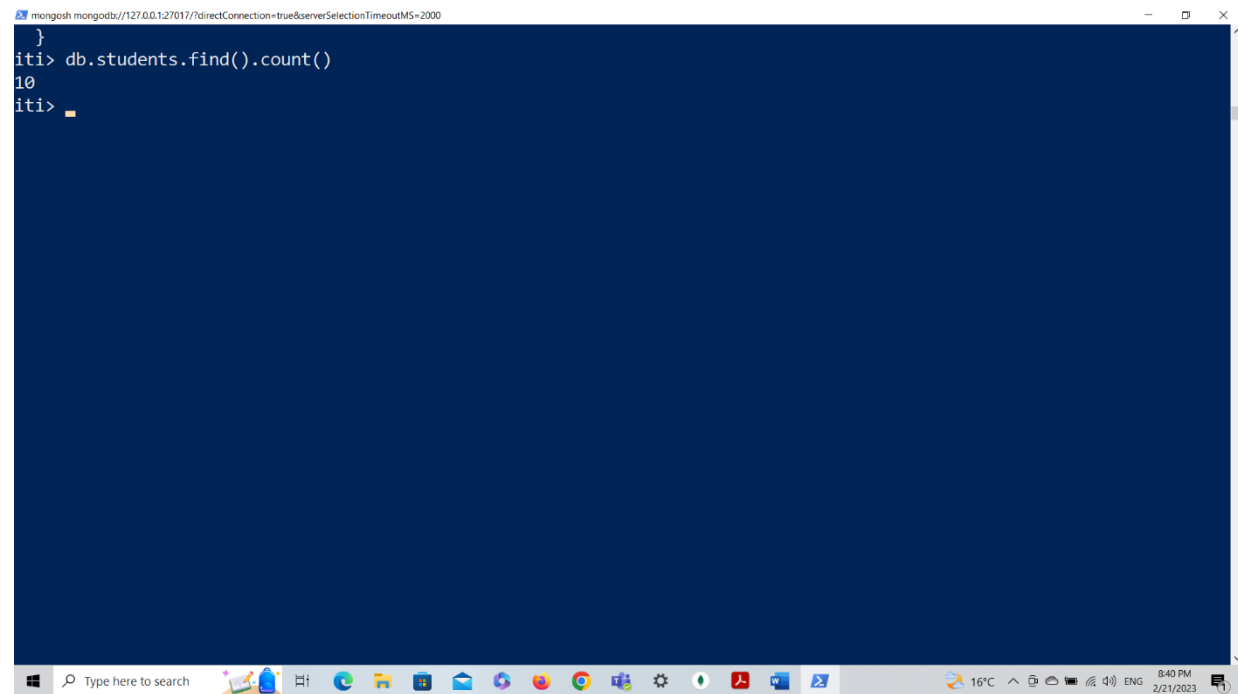
```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
}
]
iti> db.students.deleteOne({ _id: ObjectId("63f50adfe34e5cc67ce074c8")})
{ acknowledged: true, deletedCount: 1 }
iti> db.students.find({ _id: ObjectId("63f50adfe34e5cc67ce074c8")})

iti> _
```

10 – view all documents in students collection in a prettified format.

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
]
iti> db.students.find().pretty()
[
  {
    _id: ObjectId("63f50856e34e5cc67ce074c3"), name: 'Ahmad', age: 23 },
  {
    _id: ObjectId("63f50a5ce34e5cc67ce074c4"),
    name: 'Wafaa',
    age: 22,
    gender: 'Female'
  },
  {
    _id: ObjectId("63f50a5ce34e5cc67ce074c5"),
    name: 'Amal',
    age: 21,
    hobbies: [ 'run', 'read' ]
  },
  {
    _id: ObjectId("63f50a9fe34e5cc67ce074c6"),
    name: 'Faras',
    age: 20,
    gender: 'Female'
  },
  {
    _id: ObjectId("63f50a9fe34e5cc67ce074c7"),
    name: 'Nakry',
    age: 19,
    movies: [ 'lol', 'luc' ]
  },
  {
    _id: ObjectId("63f50adfe34e5cc67ce074c9"),
    name: 'Hadirowid',
    age: 26,
    movies: [ 'lcf', 'yes' ]
  },
  {
    _id: ObjectId("63f50b2de34e5cc67ce074ca"),
    name: 'Nadwa',
    age: 27,
    gender: 'Female'
  },
]
```

11 – count all documents in students collection.



The screenshot shows a MongoDB shell window with a dark blue background. The title bar at the top reads "mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000". The command prompt shows the following interaction:

```
}  
iti> db.students.find().count()  
10  
iti> █
```

The Windows taskbar is visible at the bottom, showing the search bar, task view button, and several application icons including Edge, File Explorer, and various utilities. The system tray on the right shows the temperature as 16°C, the time as 8:40 PM, and the date as 2/21/2023.

part 2

1- Create database with name ems
use ems

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"}
```

db.createCollection("faculty")

```

mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
ems> db.faculty.insertMany(
... [
...   { "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" }
... ]
... )
{
  acknowledged: true,
  insertedIds: {
    0: ObjectId("63f512984ced7ae12a60df2f"),
    1: ObjectId("63f512984ced7ae12a60df30"),
    2: ObjectId("63f512984ced7ae12a60df31"),
    3: ObjectId("63f512984ced7ae12a60df32"),
    4: ObjectId("63f512984ced7ae12a60df33"),
    5: ObjectId("63f512984ced7ae12a60df34"),
    6: ObjectId("63f512984ced7ae12a60df35"),
    7: ObjectId("63f512984ced7ae12a60df36"),
    8: ObjectId("63f512984ced7ae12a60df37"),
    9: ObjectId("63f512984ced7ae12a60df38")
  }
}
ems>

```

1. Get the details of all the faculty.

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
qualification: 'M.Tech'
},
{
ems> db.faculty.find()
[
  { name: 'Suresh',
    { age: 40,
      _id: ObjectId("63f512984ced7ae12a60df2f"),
      name: 'Suresh',
      age: 35, : [ 'JAVA', 'DBMS', 'NETWORKING' ],
      gender: 'M', Time: 'C', 'OS' ],
      exp: 10, ation: 'Ph.D'
      subjects: [ 'OS', 'C', 'OS' ], ETWORKING' ],
      type: 'Full Time',
      qualification: 'M.Tech'4ced7ae12a60df33"),
    }, name: 'Rajesh', 63f512984ced7ae12a60df34"),
    { age: 35, ani',
      _id: ObjectId("63f512984ced7ae12a60df30"),
      name: 'Rajesh',
      age: 38, : [ 'OS', 'C', 'OS' ],
      gender: 'M', Time: 'DBMS', 'OS' ],
      exp: 12, ation: 'M.Tech'
      subjects: [ 'JAVA', 'DBMS' ],
      type: 'Full Time',
      qualification: 'Ph.D'984ced7ae12a60df34"),
    }, name: 'Mani', ("63f512984ced7ae12a60df35"),
    { age: 38, ivani',
      _id: ObjectId("63f512984ced7ae12a60df31"),
      name: 'Mani',
      age: 32, : [ 'JAVA', 'DBMS', 'OS' ],
      gender: 'F', Time: 'CPP', 'MATHS' ],
      exp: 8, cation: 'Ph.D'
      subjects: [ 'C', 'CPP' ], 'OS' ],
      type: 'Part Time',
      qualification: 'M.Tech'4ced7ae12a60df35"),
    }, name: 'Sivani', 63f512984ced7ae12a60df36"),
    { age: 32, aresh',
      _id: ObjectId("63f512984ced7ae12a60df37"),
      name: 'Sivani',
      age: 30, : [ 'JAVA', 'DBMS', 'OS' ],
      gender: 'F', Time: 'C', 'OS' ],
      exp: 5, cation: 'Ph.D'
      subjects: [ 'C', 'OS' ],
      type: 'Part Time',
      qualification: 'M.Tech'4ced7ae12a60df36")
  ]
}
```

2. Get the count of all faculty members.

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
ems> db.faculty.find().count()
10
ems>
```


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

```
Select mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
age: 38,
ems> db.faculty.find({qualification: 'Ph.D'})
[
  {
    _id: ObjectId("63f512984ced7ae12a60df30"),
    name: 'Ranjit',
    age: 38,
    gender: 'M',
    exp: 12,
    subjects: [ 'JAVA', 'DBMS' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df32"),
    name: 'Suresh',
    age: 40,
    gender: 'M',
    exp: 9,
    subjects: [ 'JAVA', 'DBMS', 'NETWORKING' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df34"),
    name: 'Rani',
    age: 38,
    gender: 'F',
    exp: 10,
    subjects: [ 'JAVA', 'DBMS', 'OS' ],
    type: 'Part Time',
    qualification: 'Ph.D'
  }
]
```

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

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
ems> db.faculty.find({$and:[{exp:{$gt:8}}, {exp:{$lt:12}}]})
[
  {
    _id: ObjectId("63f512984ced7ae12a60df2f"),
    name: 'Rani',
    age: 35,
    gender: 'M',
    exp: 10,
    subjects: [ 'OS', 'C', 'DB' ],
    type: 'Part Time',
    qualification: 'M.Tech'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df32"),
    name: 'Suresh',
    age: 40,
    exp: 9,
    subjects: [ 'JAVA', 'DBMS', 'NETWORKING' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  }
]
ems> db.faculty.find({$and:[{exp:{$gt:8}}, {exp:{$lt:12}}]})
[
  {
    _id: ObjectId("63f512984ced7ae12a60df2f"),
    name: 'Rani',
    age: 35,
    gender: 'M',
    exp: 10,
    subjects: [ 'OS', 'C', 'DB' ],
    type: 'Part Time',
    qualification: 'M.Tech'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df32"),
    name: 'Suresh',
    age: 40,
    exp: 9,
    subjects: [ 'JAVA', 'DBMS', 'NETWORKING' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  }
]
```

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

mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

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

```
[
  {
    _id: ObjectId("63f512984ced7ae12a60df32"),
    name: 'Suresh',
    age: 40,
    gender: 'M',
    exp: 9,
    subjects: [ 'JAVA', 'DBMS', 'NETWORKING' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df35"),
    name: 'Sivani',
    age: 32,
    gender: 'F',
    exp: 8,
    subjects: [ 'C', 'CPP', 'MATHS' ],
    type: 'Part Time',
    qualification: 'M.Tech'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df36"),
    name: 'Ragesh',
    age: 39,
    gender: 'M',
    exp: 11,
    subjects: [ 'JAVA', 'DBMS', 'NETWORKING' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  }
]
```

”

.

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

mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

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

```
[
  {
    _id: ObjectId("63f512984ced7ae12a60df38"),
    name: 'Latha',
    age: 40,
    gender: 'F',
    exp: 13,
    subjects: [ 'MATHS' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  }
]
ems> Ph.D
exp: 13,
subjects: [ 'MATHS' ],
type: 'Full Time',
qualification: 'Ph.D'
]
ems>
```

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

```
Select mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
] age: 40,
ems> db.faculty.find({$or:[{subjects: 'JAVA'},{type: 'Part Time'}]})
[
  {
    _id: ObjectId("63f512984ced7ae12a60df30"),
    name: 'Manoj',
    age: 38,
    gender: 'M',
    exp: 12,
    subjects: [ 'JAVA', 'DBMS' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df31"),
    name: 'Anush',
    age: 32,
    gender: 'F',
    exp: 8,
    subjects: [ 'C', 'CPP' ],
    type: 'Part Time',
    qualification: 'M.Tech'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df32"),
    name: 'Suresh',
    age: 40,
    gender: 'M',
    exp: 9,
    subjects: [ 'MATHS', 'DE' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  }
]
```

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" }
```

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
ems> db.faculty.insertOne(
... {  "name": "Suresh Babu", "age": 55, "gender": "M", "exp": 25, subjects:  [ "MATHS", "DE" ], "type": "Full Time", "qualificat
... )
{
  acknowledged: true,
  insertedId: ObjectId("63f51e544ced7ae12a60df39")
}
ems>
```

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

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
ems> db.faculty.updateMany({}, { $inc: { age: 1, exp: 1 } })
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 11,
  modifiedCount: 11,
  upsertedCount: 0
}
ems>
```

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



```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
ems> db.faculty.updateOne({name: "Sivani"}, {$set:{ qualification: "Ph.D", type: "Full Time" }})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
ems>
```

The screenshot shows a MongoDB command prompt window. The title bar indicates the connection string: `mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000`. The prompt shows the command `db.faculty.updateOne({name: "Sivani"}, {$set:{ qualification: "Ph.D", type: "Full Time" }})` being executed. The response is a JSON object: `{ acknowledged: true, insertedId: null, matchedCount: 1, modifiedCount: 1, upsertedCount: 0 }`. The prompt is currently at `ems>`. The window is running on a Windows desktop, as evidenced by the taskbar at the bottom showing various application icons and system information like temperature (16°C) and time (10:08 PM, 2/21/2023).

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

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
ems> db.faculty.updateOne({subjects:"Silvani"}, {$push:{subjects:"PSK"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}
ems>
```

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

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
2 |
ems> db.faculty.deleteMany({age:{>55}})
{ acknowledged: true, deletedCount: 1 }
ems>
```

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

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
ReferenceError: name is not defined
ems> db.faculty.find({}, {name:1, qualification:1})
[
  {
    _id: ObjectId("63f512984ced7ae12a60df2f"),
    name: 'Anush',
    qualification: 'M.Tech'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df30"),
    name: 'Ramesh',
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df31"),
    name: 'Anush',
    qualification: 'M.Tech'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df32"),
    name: 'Sumesh',
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df33"),
    name: 'Rajesh',
    qualification: 'M.Tech'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df34"),
    name: 'Ramesh',
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df35"),
    name: 'Anush',
    qualification: 'M.Tech'
  }
]
```

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

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
ems> db.faculty.find({}, {name:1, qualification:1, exp:1}).sort({exp:1})
[
  {
    _id: ObjectId("63f512984ced7ae12a60df37"),
    _id: ObjectId("63f512984ced7ae12a60df33"),
    name: 'Rajesh', 'Ph.D'
    exp: 8,
    qualification: 'M.Tech'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df38"),
    name: 'Latha',
    _id: ObjectId("63f512984ced7ae12a60df31"),
    name: 'Anush',
    exp: 9,
    qualification: 'M.Tech'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df35"),
    name: 'Sivani',
    exp: 9,
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df32"),
    name: 'Sumesh',
    exp: 10,
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df37"),
    name: 'Rajesh',
    exp: 10,
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df2f"),
    name: 'Anush',
    exp: 10,
    qualification: 'M.Tech'
  }
]
```

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

```
Select mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
ems> db.faculty.find().sort({age:-1}).limit(5)
[
  {
    _id: ObjectId("63f512984ced7ae12a60df38"),
    name: 'Latha',
    age: 41,
    gender: 'F',
    exp: 14,
    subjects: [ 'MATHS' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df32"),
    name: 'Suresh',
    age: 41,
    gender: 'M',
    exp: 10,
    subjects: [ 'JAVA', 'DBMS', 'NETWORKING' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df36"),
    name: 'Nagash',
    age: 40,
    gender: 'M',
    exp: 12,
    subjects: [ 'JAVA', 'DBMS', 'NETWORKING' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df34"),
    name: 'Ravi',
    age: 38,
    gender: 'M',
    exp: 8,
    subjects: [ 'JAVA', 'DBMS', 'NETWORKING' ],
    type: 'Full Time',
    qualification: 'Ph.D'
  },
  {
    _id: ObjectId("63f512984ced7ae12a60df30"),
    name: 'Anitha',
    age: 35,
    gender: 'F',
    exp: 6,
    subjects: [ 'MATHS', 'PHYSICS' ],
    type: 'Part Time',
    qualification: 'M.Sc'
  }
]
```