

# Lab2

1. Create database named: **FacultySystemDB**.

```
FacultySystemDB> use FacultySystemDB
already on db FacultySystemDB
FacultySystemDB>
```

2. Create collection (student)

```
FacultySystemDB> db.createCollection("student")
{ ok: 1 }
FacultySystemDB> show collection
MongoshInvalidInputError: [COMMON-10001] 'collect
FacultySystemDB> show collections
student
FacultySystemDB>
```

3. Insert 3 (at least) documents in Student collection with different values.

```
FacultySystemDB> db.student.insertOne({FirstName: "Ali", LastName: "Ahmed", Age: 20, Faculty: {Name: "Art", Address: "Cairo"}, Grades: [{CourseName: "first", Grade: "A", Pass: true}, {CourseName: "second", Grade: "B", Pass: false}, {CourseName: "third", Grade: "C", Pass: false}], IsFired: false})
{ acknowledged: true, insertedId: ObjectId("64bfebb9b1ecb9c249dca4a9") }
FacultySystemDB> db.student.find()
[
  {
    _id: ObjectId("64bfebb9b1ecb9c249dca4a9"),
    FirstName: 'Ali',
    LastName: 'Ahmed',
    Age: 20,
    Faculty: { Name: 'Art', Address: 'Cairo' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ]
  }
]
FacultySystemDB> db.student.find()
[
  { IsFired: false
    {
      _id: ObjectId("64bfebb9b1ecb9c249dca4a9"),
      FirstName: 'Ali',
      LastName: 'Ahmed',
      Age: 20,
      Faculty: { Name: 'Art', Address: 'Cairo' },
      Grades: [
        { CourseName: 'first', Grade: 'A', Pass: true },
        { CourseName: 'second', Grade: 'B', Pass: false },
        { CourseName: 'third', Grade: 'C', Pass: false }
      ],
      IsFired: false
    }
  }
]
FacultySystemDB>
```

```

FacultySystemDB> db.student.insertOne({FirstName: "Nada", LastName: "Ahmed", Age: 18, Faculty: {Name: "Engineering", Address: "Alex"}, Grades: [{CourseName: "first", Grade: "A", Pass: true}, {CourseName: "second", Grade: "B", Pass: false}, {CourseName: "third", Grade: "C", Pass: false}], IsFired: false})
{
  acknowledged: true,
  insertedId: ObjectId("64bfed4db1ecb9c249dca4aa")
}
FacultySystemDB> db.student.find()
[
  {
    _id: ObjectId("64bfed4db1ecb9c249dca4aa"),
    FirstName: 'Ali',
    LastName: 'Ahmed',
    Age: 20,
    Faculty: { Name: 'Art', Address: 'Cairo' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ],
    IsFired: false
  },
  {
    _id: ObjectId("64bfed4db1ecb9c249dca4aa"),
    FirstName: 'Nada',
    LastName: 'Ahmed',
    Age: 18,
    Faculty: { Name: 'Engineering', Address: 'Alex' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ],
    IsFired: false
  }
]
FacultySystemDB>

```

4.Retrieve the following data:

- -All Students.

```
FacultySystemDB> db.student.find()
[
  {
    _id: ObjectId("64bfebb9b1ecb9c249dca4a9"),
    FirstName: 'Ali',
    LastName: 'Ahmed',
    Age: 20,
    Faculty: { Name: 'Art', Address: 'Cairo' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ],
    IsFired: false
  },
  {
    _id: ObjectId("64bfed4db1ecb9c249dca4aa"),
    FirstName: 'Nada',
    LastName: 'Ahmed',
    Age: 18,
    Faculty: { Name: 'Engineering', Address: 'Alex' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ],
    IsFired: false
  },
  {
    _id: ObjectId("64bfec5fb1ecb9c249dca4ab"),
    FirstName: 'Noha',
    LastName: 'Ahmed',
    Age: 22,
    Faculty: { Name: 'Engineering', Address: 'Qena' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: true }
    ],
    IsFired: true
  }
]
```

-Student with specific First Name.

```
]
FacultySystemDB> db.student.findOne({FirstName: "Nada"})
{
  _id: ObjectId("64bfed4db1ecb9c249dca4aa"),
  FirstName: 'Nada',
  LastName: 'Ahmed',
  Age: 18,
  Faculty: { Name: 'Engineering', Address: 'Alex' },
  Grades: [
    { CourseName: 'first', Grade: 'A', Pass: true },
    { CourseName: 'second', Grade: 'B', Pass: false },
    { CourseName: 'third', Grade: 'C', Pass: false }
  ],
  IsFired: false
}
FacultySystemDB>
```

-Students who his First Name=Ahmed, or Last Name= Ahmed.

```
]
FacultySystemDB> db.student.find({$or:[{FirstName: "Ali"}, {FirstName: "Noha"}]})
[
  {
    _id: ObjectId("64bfebb9b1ecb9c249dca4a9"),
    FirstName: 'Ali',
    LastName: 'Ahmed',
    Age: 20,
    Faculty: { Name: 'Art', Address: 'Cairo' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ],
    IsFired: false
  },
  {
    _id: ObjectId("64bfef5fb1ecb9c249dca4ab"),
    FirstName: 'Noha',
    LastName: 'Ahmed',
    Age: 22,
    Faculty: { Name: 'Engineering', Address: 'Qena' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: true }
    ],
    IsFired: true
  }
]
]
```

-Students that their First name isn't "Ahmed".

```
FacultySystemDB> db.student.find({FirstName: {$ne: "Ali"}})
[
  {
    _id: ObjectId("64bfed4db1ecb9c249dca4aa"),
    FirstName: 'Nada',
    LastName: 'Ahmed',
    Age: 18,
    Faculty: { Name: 'Engineering', Address: 'Alex' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ],
    IsFired: false
  },
  {
    _id: ObjectId("64bfef5fb1ecb9c249dca4ab"),
    FirstName: 'Noha',
    LastName: 'Ahmed',
    Age: 22,
    Faculty: { Name: 'Engineering', Address: 'Qena' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: true }
    ],
    IsFired: true
  }
]
```

-Students with Age less than 21.

```
FacultySystemDB> db.student.find({Age: {$lt: 21}})
[
  {
    _id: ObjectId("64bfefbb9b1ecb9c249dca4a9"),
    FirstName: 'Ali',
    LastName: 'Ahmed',
    Age: 20,
    Faculty: { Name: 'Art', Address: 'Cairo' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ],
    IsFired: false
  },
  {
    _id: ObjectId("64bfed4db1ecb9c249dca4aa"),
    FirstName: 'Nada',
    LastName: 'Ahmed',
    Age: 18,
    Faculty: { Name: 'Engineering', Address: 'Alex' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ],
    IsFired: false
  }
]
```



-All fired students.

```
FacultySystemDB> db.student.find({IsFired: true})
[
  {
    _id: ObjectId("64bfee5fb1ecb9c249dca4ab"),
    FirstName: 'Noha',
    LastName: 'Ahmed',
    Age: 22,
    Faculty: { Name: 'Engineering', Address: 'Qena' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: true }
    ],
    IsFired: true
  }
]
```

-Students with Age more than or equal to 21, and their faculty isn't NULL.

```
FacultySystemDB> db.student.find({Age: {$gt: 21}})
[
  {
    _id: ObjectId("64bfee5fb1ecb9c249dca4ab"),
    FirstName: 'Noha',
    LastName: 'Ahmed',
    Age: 22,
    Faculty: { Name: 'Engineering', Address: 'Qena' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: true }
    ],
    IsFired: true
  }
]
```

-Display student with specific First Name, and display his First Name, Last name, IsFired fields only.

```
FacultySystemDB> db.student.find({}, {FirstName: 1, LastName: 1, IsFired: 1, _id: 0})
[
  { FirstName: 'Ali', LastName: 'Ahmed', IsFired: false },
  { FirstName: 'Nada', LastName: 'Ahmed', IsFired: false },
  { FirstName: 'Noha', LastName: 'Ahmed', IsFired: true }
]
FacultySystemDB>
```

5.Update the student with specific FirstName, and change his LastName.

```
FacultySystemDB> db.student.updateOne({FirstName: "Ali"}, {$set: {LastName: "Mohammed"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
FacultySystemDB> db.student.find({FirstName: "Ali"})
[
  {
    _id: ObjectId("64bfebb9b1ecb9c249dca4a9"),
    FirstName: 'Ali',
    LastName: 'Mohammed',
    Age: 20,
    Faculty: { Name: 'Art', Address: 'Cairo' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ],
    IsFired: false
  }
]
```

## 6.Delete Fired students.

```
FacultySystemDB> db.student.deleteOne({IsFired: true})
{ acknowledged: true, deletedCount: 1 }
FacultySystemDB> db.student.find()
[
  {
    _id: ObjectId("64bfebb9b1ecb9c249dca4a9"),
    FirstName: 'Ali',
    LastName: 'Mohammed',
    Age: 20,
    Faculty: { Name: 'Art', Address: 'Cairo' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ],
    IsFired: false
  },
  {
    _id: ObjectId("64bfed4db1ecb9c249dca4aa"),
    FirstName: 'Nada',
    LastName: 'Ahmed',
    Age: 18,
    Faculty: { Name: 'Engineering', Address: 'Alex' },
    Grades: [
      { CourseName: 'first', Grade: 'A', Pass: true },
      { CourseName: 'second', Grade: 'B', Pass: false },
      { CourseName: 'third', Grade: 'C', Pass: false }
    ],
    IsFired: false
  }
]
```

## 7.Delete all students collection.

db.student.drop()



9. Create new database named: FacultySystemV2.

```
]
FacultySystemDB> use FacultySystemV2
switched to db FacultySystemV2
FacultySystemV2>
```

-Create student collection that has (FirstName, lastName, IsFired, FacultyID, courses :>>array of objects, each object has CourseID, grade).

```
FacultySystemV2> db.createCollection("student")
{ ok: 1 }
FacultySystemV2> show collections
student
FacultySystemV2> db.student.insertOne({FirstName: "Ali", LastName: "Omer", IsFired: false, FacultyID: 11, courses: [{CourseID: 1, Grade: "A"}, {CourseID: 2, Grade: "B"}]})
{
  acknowledged: true,
  insertedId: ObjectId("64bffd08b1ecb9c249dca4ac")
}
FacultySystemV2> db.student.find()
[
  {
    _id: ObjectId("64bffd08b1ecb9c249dca4ac"),
    FirstName: 'Ali',
    LastName: 'Omer',
    IsFired: false,
    FacultyID: 11,
    courses: [ { CourseID: 1, Grade: 'A' }, { CourseID: 2, Grade: 'B' } ]
  }
]
```

-Create Faculty collection that has (Faculty Name, Address).

```
]
FacultySystemV2> db.createCollection("faculty")
{ ok: 1 }
FacultySystemV2> show collections
faculty
student
FacultySystemV2> db.faculty.insertOne({FacultyName: "Art", Address: "Cairo"})
{
  acknowledged: true,
  insertedId: ObjectId("64bffe51b1ecb9c249dca4ad")
}
FacultySystemV2> db.faculty.find()
[
  {
    _id: ObjectId("64bffe51b1ecb9c249dca4ad"),
    FacultyName: 'Art',
    Address: 'Cairo'
  }
]
```

-Create Course collection, which has (Course Name, Final Mark).

```
FacultySystemV2> db.createCollection("course")
{ ok: 1 }
FacultySystemV2> show collections
course
faculty
student
FacultySystemV2> db.course.insertOne({CourseName: "First", FinalMark: 100})
{
  acknowledged: true,
  insertedId: ObjectId("64bffff18b1ecb9c249dca4ae")
}
FacultySystemV2> db.course.find()
[
  {
    _id: ObjectId("64bffff18b1ecb9c249dca4ae"),
    CourseName: 'First',
    FinalMark: 100
  }
]
```