Day 1 Lab Assignments

Use MongoDB to do the following assignments:

1. Create database named: FacultySystemDB.

```
Atlas atlas-enstam-shard-0 [primary] test> use FacultySystemDB
switched to db FacultySystemDB
```

2. Create collection (student) that has:
   - FirstName: string
   - LastName: string
   - Age: Number
   - Faculty: An object that has Name and Address
   - Grades: An array of objects, each object has: CourseName, Grade, Pass (Boolean).
   - IsFired: Boolean

```
db.createCollection("student")
db.student.insertOne(
{
    "FirstName" : "shaymaa",
    "LastName" : "Mohamed",
    "Age" : 21,
    "Faculty" : {
            "Name" : "Faculty of computer science and information",
            "Address" : "Menofia",
        },
    "Grades" : [ { "CourseName" : "softWare" ,
                "Grade" : 95 ,
                "Pass" : true
            }],
    "IsFired" : "false" })
```

3. Insert 3 (at least) documents in Student collection with different values.
   - Try inserting one record each time
   - Try inserting collection using on insert statement.

insertOne

```
student
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.insertOne({ "FirstName":"Esraa", "LastName":"Ahmed", "Age":20, "Faculty":{ "Name":"Faculty Of
Science", "Address":"Menofia"}, "Grades":[{"CourseName":"English", "Grade":80, "Pass":"true"}], "IsFired":"false"}
... )
{
  acknowledged: true,
  insertedId: ObjectId("62e31c86006bf325adc8fcba")
}
```

insertMeny

```
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.insertMany([{ "FirstName": "Esraa", "LastName": "Ahmed", "Age": 20, "Faculty": { "Name": "Facu
lty Of Science", "Address": "Menofia" }, "Grades": [{ "CourseName": "English", "Grade": 80, "Pass": "true" }], "IsFired": "false" },{"FirstName": "Anas","LastN
ame": "Mohammed", "Age": 19, "Faculty": { "Name": "Faculty Of AI", "Address": "Menofia" }, "Grades": [{ "CourseName": "Arabic", "Grade": 70, "Pass": "true" }],
 "IsFired": "false" },{ "FirstName": "Mahmoud", "LastName": "Ahmed", "Age": 23, "Faculty": { "Name": "Faculty Of Science", "Address": "Menofia" }, "Grades": [{
 "CourseName": "Math", "Grade": 100, "Pass": "true" }], "IsFired": "false" }])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("62e31f4c006bf325adc8fcbb"),
    '1': ObjectId("62e31f4c006bf325adc8fcbc"),
    '2': ObjectId("62e31f4c006bf325adc8fcbd")
  }
}
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB>
```

4. Retrieve the following data:
    - All Students.
    db.student.find({}).pretty()

```
tlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.find( {} ).pretty()

{
  _id: ObjectId("62e31b51006bf325adc8fcb9"),
  FirstName: 'Shaymaa',
  LastName: 'Mohamed',
  Age: 21,
  Faculty: {
    Name: 'Faculty Of Computer Science And Information',
    Address: 'Menofia'
  },
  Grades: [ { CourseName: 'Software', Grade: 95, Pass: 'true' } ],
  IsFired: 'true'
},
{
  _id: ObjectId("62e31c86006bf325adc8fcba"),
  FirstName: 'Esraa',
  LastName: 'Ahmed',
  Age: 20,
  Faculty: { Name: 'Faculty Of Science', Address: 'Menofia' },
  Grades: [ { CourseName: 'English', Grade: 80, Pass: 'true' } ],
  IsFired: 'false'
},
```

    - Student with specific First Name.
    db.student.find({FirstName : "Shaymaa"}).pretty()

```
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.find( {"FirstName" : "Shaymaa"} ).pretty()
[
  {
    _id: ObjectId("62e31b51006bf325adc8fcb9"),
    FirstName: 'Shaymaa',
    LastName: 'Mohamed',
    Age: 21,
    Faculty: {
      Name: 'Faculty Of Computer Science And Information',
      Address: 'Menofia'
    },
    Grades: [ { CourseName: 'Software', Grade: 95, Pass: 'true' } ],
    IsFired: 'true'
  }
]
```

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

db.student.find({$or :[{"FirstName" : "Ahmed"},{ "LastName" : "Ahmed"}]}).pretty()

```
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.find( {$or :[{"FirstName" : "Ahmed"},{ "LastName" : "Ahmed"}]} ).pretty()
[
  {
    _id: ObjectId("62e31c86006bf325adc8fcba"),
    FirstName: 'Esraa',
    LastName: 'Ahmed',
    Age: 20,
    Faculty: { Name: 'Faculty Of Science', Address: 'Menofia' },
    Grades: [ { CourseName: 'English', Grade: 80, Pass: 'true' } ],
    IsFired: 'false'
  },
  {
    _id: ObjectId("62e31f4c006bf325adc8fcbb"),
    FirstName: 'Esraa',
    LastName: 'Ahmed',
    Age: 20,
    Faculty: { Name: 'Faculty Of Science', Address: 'Menofia' },
    Grades: [ { CourseName: 'English', Grade: 80, Pass: 'true' } ],
    IsFired: 'false'
  },
  {
    _id: ObjectId("62e31f4c006bf325adc8fcbd"),
    FirstName: 'Mahmoud',
    LastName: 'Ahmed',
    Age: 23,
    Faculty: { Name: 'Faculty Of Science', Address: 'Menofia' },
    Grades: [ { CourseName: 'Math', Grade: 100, Pass: 'true' } ],
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB>
```

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

db.student.find({FirstName : {$nin: ["Ahmed"]}}).pretty()

```
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.find( {FirstName : {$nin: ["Ahmed"]}}).pretty()
[
  {
    _id: ObjectId("62e31b51006bf325adc8fcb9"),
    FirstName: 'Shaymaa',
    LastName: 'Mohamed',
    Age: 21,
    Faculty: {
      Name: 'Faculty Of Computer Science And Information',
      Address: 'Menofia'
    },
    Grades: [ { CourseName: 'Software', Grade: 95, Pass: 'true' } ],
    IsFired: 'true'
  },
  {
    _id: ObjectId("62e31c86006bf325adc8fcba"),
    FirstName: 'Esraa',
    LastName: 'Ahmed',
    Age: 20,
    Faculty: { Name: 'Faculty Of Science', Address: 'Menofia' },
    Grades: [ { CourseName: 'English', Grade: 80, Pass: 'true' } ],
    IsFired: 'false'
  },
  {
    _id: ObjectId("62e31f4c006bf325adc8fcbb"),
    FirstName: 'Esraa',
    LastName: 'Ahmed',
    Age: 20,
    Faculty: { Name: 'Faculty Of Science', Address: 'Menofia' },
    Grades: [ { CourseName: 'English', Grade: 80, Pass: 'true' } ],
    IsFired: 'false'
  },
  {
```

- Students with Age less than 21.

db.student.find({ Age: {$lt: 21}})

```
]
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.find({ Age: {$lt: 21}})
[
  {
    _id: ObjectId("62e31c86006bf325adc8fcba"),
    FirstName: 'Esraa',
    LastName: 'Ahmed',
    Age: 20,
    Faculty: { Name: 'Faculty Of Science', Address: 'Menofia' },
    Grades: [ { CourseName: 'English', Grade: 80, Pass: 'true' } ],
    IsFired: 'false'
  },
  {
    _id: ObjectId("62e31f4c006bf325adc8fcbb"),
    FirstName: 'Esraa',
    LastName: 'Ahmed',
    Age: 20,
    Faculty: { Name: 'Faculty Of Science', Address: 'Menofia' },
    Grades: [ { CourseName: 'English', Grade: 80, Pass: 'true' } ],
    IsFired: 'false'
  },
  {
```

- All fired students.

db.student.find({IsFired: {$eq: "true"}}).pretty()

```
]
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.find( {IsFired: {$eq: "true"}} ).pretty()
[
  {
    _id: ObjectId("62e31b51006bf325adc8fcb9"),
    FirstName: 'Shaymaa',
    LastName: 'Mohamed',
    Age: 21,
    Faculty: {
      Name: 'Faculty Of Computer Science And Information',
      Address: 'Menofia'
    },
    Grades: [ { CourseName: 'Software', Grade: 95, Pass: 'true' } ],
    IsFired: 'true'
  }
]
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> _
```

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

db.student.find({$and:[{ Age: {$gte: 21}} , {Faculty: {$ne:null}}]}).pretty()

```
2 |
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.find({$and:[{ Age: {$gte: 21}} , {Faculty: {$ne:null}}]}).pretty(
[
  {
    _id: ObjectId("62e31b51006bf325adc8fcb9"),
    FirstName: 'Shaymaa',
    LastName: 'Mohamed',
    Age: 21,
    Faculty: {
      Name: 'Faculty Of Computer Science And Information',
      Address: 'Menofia'
    },
    Grades: [ { CourseName: 'Software', Grade: 95, Pass: 'true' } ],
    IsFired: 'true'
  },
  {
    _id: ObjectId("62e31f4c006bf325adc8fcbd"),
    FirstName: 'Mahmoud',
    LastName: 'Ahmed',
    Age: 23,
    Faculty: { Name: 'Faculty Of Science', Address: 'Menofia' },
    Grades: [ { CourseName: 'Math', Grade: 100, Pass: 'true' } ],
    IsFired: 'false'
  }
]
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB>
```

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

db.student.find({"FirstName":"Shaymaa"},{_id:0 , FirstName: 1 , LastName: 1 , IsFired:1}).pretty()

```
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.find( {"FirstName":"Shaymaa"},{_id:0 , FirstName: 1 , LastName: 1 , IsFired:1} ).pretty()
[ { FirstName: 'Shaymaa', LastName: 'Mohamed', IsFired: 'true' } ]
```

5. Update the student with specific FirstName, and change his LastName.
   - Try Update() statement.

db.student.update({FirstName: "Esraa"} , {$set : {LastName : "Ali"}})

```
  }
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.update({FirstName: "Esraa"} , {$set : {LastName : "Ali"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 0,
  upsertedCount: 0
}
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.find( {FirstName : "shaymaa"} ).pretty()

Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.find( {FirstName : "Esraa"} ).pretty()
[
  {
    _id: ObjectId("62e31c86006bf325adc8fcba"),
    FirstName: 'Esraa',
    LastName: 'Ali',
    Age: 20,
    Faculty: { Name: 'Faculty Of Science', Address: 'Menofia' },
    Grades: [ { CourseName: 'English', Grade: 80, Pass: 'true' } ],
    IsFired: 'false'
  },
```

- Try Update() with Mulit option.

db.student.updateMany({FirstName: "Esraa" }, {$set:{LastName : "ABDelwaahab" , Age : 22 }})

```
]
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.updateMany({FirstNamFirstName: "Esraa" }, {$set:{LastName : "ABDelwaahab" , Age : 22 }})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 2,
  modifiedCount: 2,
  upsertedCount: 0
}
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.find( {FirstName : "Esraa"} ).pretty()
[
  {
    _id: ObjectId("62e31c86006bf325adc8fcba"),
    FirstName: 'Esraa',
    LastName: 'ABDelwaahab',
    Age: 22,
    Faculty: { Name: 'Faculty Of Science', Address: 'Menofia' },
    Grades: [ { CourseName: 'English', Grade: 80, Pass: 'true' } ],
    IsFired: 'false'
  },
```

6. Delete Fired students.

db.student.deleteMany({IsFired : "true"})

```
  }
]
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.deleteMany( {IsFired : "true"} )
{ acknowledged: true, deletedCount: 1 }
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB>
```

7. Delete all students collection.

db.student.deleteMany({})

```
{ acknowledged: true, deletedCount: 1 }
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> db.student.deleteMany( {} )
{ acknowledged: true, deletedCount: 4 }
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB>
```

8. Delete the whole DB.
db.dropDatabase()

9. Create new database named: FacultySystemV2.
use FacultySystemV2

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

```
db.createCollection("student")
db.student.insertOne(
{
    "FirstName" : "shaymaa",
    "LastName" : "Mohamed",

    "FacultyID" : [{
        "CourseID" : 123 ,
        "Grade" : 80 ,
    }]})
```

```
Atlas atlas-enstam-shard-0 [primary] FacultySystemDB> use FacultySystemV2
switched to db FacultySystemV2
Atlas atlas-enstam-shard-0 [primary] FacultySystemV2> db.createCollection("student")
{ ok: 1 }
Atlas atlas-enstam-shard-0 [primary] FacultySystemV2> db.student.insertOne(
... {
.....        "FirstName" : "shaymaa",
.....        "LastName" : "Mohamed",
......
.....        "FacultyID" : [{
.......             "CourseID" : 123 ,
.......             "Grade" :  80 ,
.......          }]})
{
  acknowledged: true,
  insertedId: ObjectId("62e3b2d3006bf325adc8fcbe")
}
Atlas atlas-enstam-shard-0 [primary] FacultySystemV2>
```

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

```
db.createCollection("Faculty")
db.Faculty.insertOne({
    "FacultyName" : "Faculty of computer science and information",
```

```
        "Address" : "Menofia",})
}
Atlas atlas-enstam-shard-0 [primary] FacultySystemV2> db.createCollection("Faculty")
{ ok: 1 }
Atlas atlas-enstam-shard-0 [primary] FacultySystemV2> db.Faculty.insertOne({
...          "FacultyName" : "Faculty of computer science and information",
...       "Address" : "Menofia",})
{
  acknowledged: true,
  insertedId: ObjectId("62e3b317006bf325adc8fcbf")
}
Atlas atlas-enstam-shard-0 [primary] FacultySystemV2>
```

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

db.createCollection("Course ")
db.Faculty.insertOne({
    "CourseName" : "Cloud",
     "FinalMark" : "100",})

```
{ ok: 1 }
Atlas atlas-enstam-shard-0 [primary] FacultySystemV2> db.createCollection("Course")
{ ok: 1 }
Atlas atlas-enstam-shard-0 [primary] FacultySystemV2> db.Faculty.insertOne({
...          "CourseName" : "Cloud",
...       "FinalMark" : "100",})
{
  acknowledged: true,
  insertedId: ObjectId("62e3b377006bf325adc8fcc0")
}
Atlas atlas-enstam-shard-0 [primary] FacultySystemV2>
```

<Script>document.write("Thank YOU") </Script>