Anuvind M P (AM.EN.U4AIE22010)

MongoDB Lab Sheet 7

1. Insertion

Insert the following documents into a collection named students.

Sample Documents

```
db.students.insertMany([
  {
    name: "Alice",
    age: 23,
    regno: "CSE1001",
    department: "CSE",
    mark: 78
  },
  {
    name: "Bob",
    age: 21,
    regno: "ECE1002",
    department: "ECE",
    mark: 55
  },
  {
    name: "Arun",
    age: 24,
    regno: "CSE1003",
    department: "CSE",
    mark: 65
  },
  {
    name: "Divya",
    age: 22,
    regno: "EEE1004",
    department: "EEE",
    mark: 88
  },
  {
    name: "Amit",
    age: 23,
    regno: "CSE1005",
    department: "CSE",
    mark: 34
  }
])
```

---

2. Querying (Read Operations)

a. Retrieve all documents

```
db.students.find()
```

Output:

```
[
  { "name": "Alice", "age": 23, "regno": "CSE1001", "department": "CSE", "mark": 78 },
  { "name": "Bob", "age": 21, "regno": "ECE1002", "department": "ECE", "mark": 55 },
  { "name": "Arun", "age": 24, "regno": "CSE1003", "department": "CSE", "mark": 65 },
  { "name": "Divya", "age": 22, "regno": "EEE1004", "department": "EEE", "mark": 88 },
  { "name": "Amit", "age": 23, "regno": "CSE1005", "department": "CSE", "mark": 34 }
]
```

b. Students in CSE department

```
db.students.find({ department: "CSE" })
```

c. Students older than 22

```
db.students.find({ age: { $gt: 22 } })
```

---

3. Projection

a. Retrieve only name and department

```
db.students.find({}, { name: 1, department: 1, _id: 0 })
```

Output:

```
[
  { "name": "Alice", "department": "CSE" },
  { "name": "Bob", "department": "ECE" },
  { "name": "Arun", "department": "CSE" },
  { "name": "Divya", "department": "EEE" },
  { "name": "Amit", "department": "CSE" }
]
```

---

4. Sorting

a. Sort by age (ascending)

```
db.students.find().sort({ age: 1 })
```

b. Sort by mark (descending)

```
db.students.find().sort({ mark: -1 })
```

---

5. Conditional Operators

a. Names where mark > 40

```
db.students.find({ mark: { $gt: 40 } }, { name: 1, _id: 0 })
```

b. Names starting with "A"

```
db.students.find({ name: /^A/ })
```

Output:

```
[
  { "name": "Alice", "age": 23, "regno": "CSE1001", "department":
"CSE", "mark": 78 },
  { "name": "Arun", "age": 24, "regno": "CSE1003", "department":
"CSE", "mark": 65 },
  { "name": "Amit", "age": 23, "regno": "CSE1005", "department":
"CSE", "mark": 34 }
]
```