

ACCA7002 Database Systems

Tutorial 16

Student database with MongoDB

Activity 1: Setup MongoDB (if you haven't installed)

Install MongoDB on your machine. You can follow the installation instructions provided on the official MongoDB website (<https://docs.mongodb.com/manual/installation/>).

Start the MongoDB server.

Alternatively, you can use <https://www.mycompiler.io/new/mongodb>

Activity 2: Create a Database and Collection

Open the MongoDB shell.

```
use student_database
```

```
// Sample Student Documents
```

```
db.students.insertMany([
  {
    studentID: "1001",
    firstName: "John",
    lastName: "Doe",
    address: {
      street: "123 Main St",
      suburb: "Cityville",
      state: "CA",
      postcode: "12345"
    },
    gender: "Male",
    course: "Computer Science",
    year: 2,
    offCampus: false,
    emails: ["john.doe@example.com",
"john.doe.school@gmail.com"]
  },
  {
    studentID: "1002",
    firstName: "Jane",
    lastName: "Smith",
    address: {
      street: "456 Oak Ave",
      suburb: "Townsville",
      state: "NY",
      postcode: "67890"
    },
    gender: "Female",
    course: "Biology",
    year: 3,
```

```

        offCampus: true,
        emails: ["jane.smith@example.com", "j.smith@yahoo.com"]
    },
    {
        studentID: "1003",
        firstName: "Michael",
        lastName: "Johnson",
        address: {
            street: "789 Pine Rd",
            suburb: "Villagetown",
            state: "TX",
            postcode: "54321"
        },
        gender: "Male",
        course: "Physics",
        year: 1,
        offCampus: false,
        emails: ["michael.j@example.com", "m.johnson@hotmail.com"]
    },
    {
        studentID: "1004",
        firstName: "Emily",
        lastName: "Williams",
        address: {
            street: "101 Cedar Blvd",
            suburb: "Hamletville",
            state: "FL",
            postcode: "98765"
        },
        gender: "Female",
        course: "Chemistry",
        year: 4,
        offCampus: true,
        emails: ["emily.w@example.com",
"emily.williams@gmail.com"]
    },
    {
        studentID: "1005",
        firstName: "Christopher",
        lastName: "Brown",
        address: {
            street: "202 Maple Ln",
            suburb: "Countryside",
            state: "AZ",
            postcode: "13579"
        },
        gender: "Male",
        course: "Mathematics",
        year: 2,
        offCampus: false,
        emails: ["chris.b@example.com", "chris.math@yahoo.com"]
    }
]
)

```

Activity 3: Create Index

Create an Index:

Create an index on the studentID field to optimize search operations:

```
db.students.createIndex({ studentID: 1 })
```

Activity 4: Search Operations

Basic Search:

Retrieve a student's information based on their ID:

```
db.students.find({ studentID: "1003" })
```

Search by Course:

Find all students enrolled in a specific course (e.g., "Computer Science"):

```
db.students.find({ course: "Computer Science" })
```

Search by Off-Campus Status:

Retrieve students who are living off-campus:

```
db.students.find({ offCampus: true })
```

Activity 5: Sorting

Sort by Last Name:

Retrieve all students and sort them alphabetically by last name:

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

Sort by Year and Course:

Sort students by their academic year in ascending order and within the same year, sort them by course alphabetically:

```
db.students.find().sort({ year: 1, course: 1 })
```

Activity 6: Combined Operations

Search and Sort:

Retrieve all female students who are living off-campus, sorted by last name:

```
db.students.find({ gender: "Female", offCampus: true }).sort({  
  lastName: 1 })
```

Search, Sort, and Limit:

Retrieve the first three students who are enrolled in the "Mathematics" course, sorted by their student IDs:

```
db.students.find({ course: "Mathematics" }).sort({ studentID: 1 }).limit(3)
```

Activity 7: Index Usage

Use the explain method to analyze the query execution plan and index usage:

```
db.students.find({ studentID: "1003"
}).explain("executionStats")
```

-----End-----