MongoDB Assignment 1

1. Create a new database called student_management.

```
> use student_management;
< switched to db student_management
student_management>
```

2. Create a collection called students in the student_management database.

```
> db.createCollection("students");
< { ok: 1 }
```

3. Insert at least five student records into the students collection. Each record should have the following fields:

   - student_id (integer)
   - name (string)
   - age (integer)
   - department (string)
   - courses (array of strings)
   - grade (string)

```
> db.students.insertMany([
    {
      student_id: 101,
      name: "John Doe",
      age: 20,
      department: "Computer Science",
      courses: ["Database Systems", "Algorithms"],
      grade: "A"
    },
    {
      student_id: 102,
      name: "Jane Smith",
      age: 22,
      department: "Mathematics",
      courses: ["Calculus", "Statistics"],
      grade: "B"
    },
    {
      student_id: 103,
      name: "Emily Johnson",
      age: 23,
      department: "Computer Science",
      courses: ["Database Systems", "Data Structures"],
      grade: "A"
    },
    {
      student_id: 104,
      name: "Michael Brown",
      age: 21,
      department: "Physics",
      courses: ["Quantum Mechanics", "Thermodynamics"],
      grade: "C"
    }]);
```

```
< {
    acknowledged: true,
    insertedIds: {
      '0': ObjectId('6738d993ef2d8263e0b162aa'),
      '1': ObjectId('6738d993ef2d8263e0b162ab'),
      '2': ObjectId('6738d993ef2d8263e0b162ac'),
      '3': ObjectId('6738d993ef2d8263e0b162ad')
    }
  }
```

4. Query the Collection:

   Write queries to perform the following tasks:

- Retrieve all students who are in the "Computer Science" department.

```
> db.students.find({ department: "Computer Science" });
< {
    _id: ObjectId('6738d993ef2d8263e0b162aa'),
    student_id: 101,
    name: 'John Doe',
    age: 20,
    department: 'Computer Science',
    courses: [
      'Database Systems',
      'Algorithms'
    ],
    grade: 'A'
  }
  {
    _id: ObjectId('6738d993ef2d8263e0b162ac'),
    student_id: 103,
    name: 'Emily Johnson',
    age: 23,
    department: 'Computer Science',
    courses: [
      'Database Systems',
      'Data Structures'
    ],
    grade: 'A'
  }
```

- Retrieve students who have an age greater than 21.

```
> db.students.find({ age: { $gt: 21 } });
< {
    _id: ObjectId('6738d993ef2d8263e0b162ab'),
    student_id: 102,
    name: 'Jane Smith',
    age: 22,
    department: 'Mathematics',
    courses: [
      'Calculus',
      'Statistics'
    ],
    grade: 'B'
  }
  {
    _id: ObjectId('6738d993ef2d8263e0b162ac'),
    student_id: 103,
    name: 'Emily Johnson',
    age: 23,
    department: 'Computer Science',
    courses: [
      'Database Systems',
      'Data Structures'
    ],
    grade: 'A'
  }
```

- Retrieve students who are taking the "Database Systems" course.

```
> db.students.find({ courses: "Database Systems" });
< {
    _id: ObjectId('6738d993ef2d8263e0b162aa'),
    student_id: 101,
    name: 'John Doe',
    age: 20,
    department: 'Computer Science',
    courses: [
      'Database Systems',
      'Algorithms'
    ],
    grade: 'A'
  }
  {
    _id: ObjectId('6738d993ef2d8263e0b162ac'),
    student_id: 103,
    name: 'Emily Johnson',
    age: 23,
    department: 'Computer Science',
    courses: [
      'Database Systems',
      'Data Structures'
    ],
    grade: 'A'
  }
```

- Retrieve students with a grade of "A".

```
> db.students.find({ grade: "A" });
< {
    _id: ObjectId('6738d993ef2d8263e0b162aa'),
    student_id: 101,
    name: 'John Doe',
    age: 20,
    department: 'Computer Science',
    courses: [
      'Database Systems',
      'Algorithms'
    ],
    grade: 'A'
  }
  {
    _id: ObjectId('6738d993ef2d8263e0b162ac'),
    student_id: 103,
    name: 'Emily Johnson',
    age: 23,
    department: 'Computer Science',
    courses: [
      'Database Systems',
      'Data Structures'
    ],
    grade: 'A'
  }
```

5. **Update Documents:**

- Update the `age` of a student with `student_id` 101 to 21.

```
> db.students.updateOne({student_id: 101},{$set:{age: 21}});
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 1,
    modifiedCount: 1,
    upsertedCount: 0
  }
> db.students.find()
< {
    _id: ObjectId('6738d993ef2d8263e0b162aa'),
    student_id: 101,
    name: 'John Doe',
    age: 21,
    department: 'Computer Science',
    courses: [
      'Database Systems',
      'Algorithms'
    ],
    grade: 'A'
  }
```

- Add a new course, "Machine Learning", to the `courses` array for students in the "Computer Science" department.

```
> db.students.updateMany(
    { department: "Computer Science" },
    { $addToSet: { courses: "Machine Learning" } }
  );
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 2,
    modifiedCount: 0,
    upsertedCount: 0
  }
```

```
> db.students.find()
< {
    _id: ObjectId('6738d993ef2d8263e0b162aa'),
    student_id: 101,
    name: 'John Doe',
    age: 21,
    department: 'Computer Science',
    courses: [
      'Database Systems',
      'Algorithms',
      'Machine Learning'
    ],
    grade: 'A'
  }
  {
    _id: ObjectId('6738d993ef2d8263e0b162ab'),
    student_id: 102,
    name: 'Jane Smith',
    age: 22,
    department: 'Mathematics',
    courses: [
      'Calculus',
      'Statistics'
    ],
    grade: 'B'
  }
```

```
{
  _id: ObjectId('6738d993ef2d8263e0b162ac'),
  student_id: 103,
  name: 'Emily Johnson',
  age: 23,
  department: 'Computer Science',
  courses: [
    'Database Systems',
    'Data Structures',
    'Machine Learning'
  ],
  grade: 'A'
}
{
  _id: ObjectId('6738d993ef2d8263e0b162ad'),
  student_id: 104,
  name: 'Michael Brown',
  age: 21,
  department: 'Physics',
  courses: [
    'Quantum Mechanics',
    'Thermodynamics'
  ],
  grade: 'C'
}
```

6. **Delete Documents:**

- Delete a student record with `student_id` 104.

```
> db.students.deleteOne({ student_id: 104 });
< {
    acknowledged: true,
    deletedCount: 1
  }
```

```
> db.students.find()
< {
    _id: ObjectId('6738d993ef2d8263e0b162aa'),
    student_id: 101,
    name: 'John Doe',
    age: 21,
    department: 'Computer Science',
    courses: [
      'Database Systems',
      'Algorithms',
      'Machine Learning'
    ],
    grade: 'A'
  }
  {
    _id: ObjectId('6738d993ef2d8263e0b162ab'),
    student_id: 102,
    name: 'Jane Smith',
    age: 22,
    department: 'Mathematics',
    courses: [
      'Calculus',
      'Statistics'
    ],
    grade: 'B'
  }
```

```
{
  _id: ObjectId('6738d993ef2d8263e0b162ac'),
  student_id: 103,
  name: 'Emily Johnson',
  age: 23,
  department: 'Computer Science',
  courses: [
    'Database Systems',
    'Data Structures',
    'Machine Learning'
  ],
  grade: 'A'
}
```

- Delete all students who have a grade lower than "C".

```
> db.students.deleteMany({ grade: { $lt: "C" } });
< {
    acknowledged: true,
    deletedCount: 3
  }
> db.students.find()
<
```