

ractical -04

Aim: Querying MongoDB

Write and execute MongoDB queries to retrieve specific data from a collection.

Execution:

Q1. Create collection employee and department in mongodb using company database. Insert Sample Data (At least 10 rows in employee collection and 5 rows in department. Make sure you use the same dept_id mentioned by you in the department table). Make sure you insert data in order to get output for the queries below:

```
company> db.department.insertMany([
  "Operations", location: "Dallas" }
...   { _id: 1, name: "IT", location: "New York" },
...   { _id: 2, name: "HR", location: "Chicago" },
...   { _id: 3, name: "Finance", location: "Los Angeles" },
...   { _id: 4, name: "Marketing", location: "San Francisco" },
...   { _id: 5, name: "Operations", location: "Dallas" }
... ]);
{
  acknowledged: true,
  insertedIds: { '0': 1, '1': 2, '2': 3, '3': 4, '4': 5 }
}
company> db.createCollection("employee");
{ ok: 1 }
```

```
company> db.employee.insertMany([
...   { name: "Alice Johnson", age: 28, dept_id: 1 },
...   { name: "Bob Smith", age: 32, dept_id: 2 },
...   { name: "Charlie Brown", age: 29, dept_id: 1 },
...   { name: "David Miller", age: 35, dept_id: 3 },
...   { name: "Emma Davis", age: 31, dept_id: 4 },
...   { name: "Frank Wilson", age: 40, dept_id: 2 },
...   { name: "Grace Taylor", age: 27, dept_id: 5 },
...   { name: "Harry Johnson", age: 33, dept_id: 1 },
...   { name: "Ivy Martinez", age: 36, dept_id: 3 },
...   { name: "Jack Anderson", age: 30, dept_id: 4 },
... ]);
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('65a6173f0756174d3330a4f8'),
    '1': ObjectId('65a6173f0756174d3330a4f9'),
    '2': ObjectId('65a6173f0756174d3330a4fa'),
    '3': ObjectId('65a6173f0756174d3330a4fb'),
    '4': ObjectId('65a6173f0756174d3330a4fc'),
    '5': ObjectId('65a6173f0756174d3330a4fd'),
    '6': ObjectId('65a6173f0756174d3330a4fe'),
    '7': ObjectId('65a6173f0756174d3330a4ff'),
    '8': ObjectId('65a6173f0756174d3330a500'),
    '9': ObjectId('65a6173f0756174d3330a501')
  }
}
company>
```

1. Retrieve all employees

```
company> db.employee.find()
[
  {
    _id: ObjectId('65a6173f0756174d3330a4f8'),
    name: 'Alice Johnson',
    age: 28,
    dept_id: 1
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4f9'),
    name: 'Bob Smith',
    age: 32,
    dept_id: 2
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fa'),
    name: 'Charlie Brown',
    age: 29,
    dept_id: 1
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fb'),
    name: 'David Miller',
    age: 35,
    dept_id: 3
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fc'),
    name: 'Emma Davis',
    age: 31,
    dept_id: 4
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fd'),
    name: 'Frank Wilson',
    age: 40,
    dept_id: 2
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fe'),
    name: 'Grace Taylor',
    age: 27,
    dept_id: 5
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4ff'),
    name: 'Harry Johnson',
    age: 33,
    dept_id: 1
  },
  {
    _id: ObjectId('65a6173f0756174d3330a500'),
    name: 'Ivy Martinez',
    age: 36,
    dept_id: 3
  },
  {
    _id: ObjectId('65a6173f0756174d3330a501'),
    name: 'Jack Anderson',
    age: 30,
    dept_id: 4
  }
]
```

2. Retrieve employees in a specific department (e.g., IT department)

```
company> db.employee.find({ dept_id: 1 });
[
  {
    _id: ObjectId('65a6173f0756174d3330a4fa'),
    name: 'Charlie Brown',
    age: 30,
    dept_id: 1,
    id: 3
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4ff'),
    name: 'Harry Johnson',
    age: 34,
    dept_id: 1,
    id: 7
  }
]
```

3. Retrieve all departments

```
company> db.department.find()
[
  { _id: 1, name: 'IT', location: 'New York' },
  { _id: 2, name: 'HR', location: 'Chicago' },
  { _id: 3, name: 'Finance', location: 'Los Angeles' },
  { _id: 4, name: 'Marketing', location: 'San Francisco' },
  { _id: 5, name: 'Operations', location: 'Dallas' }
]
```

4. Retrieve employees older than 30 years

```
company> db.employee.find({ age: { $gt: 30 } })
[
  {
    _id: ObjectId('65a6173f0756174d3330a4fb'),
    name: 'David Miller',
    age: 36,
    dept_id: 3,
    id: 4
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fc'),
    name: 'Emma Davis',
    age: 32,
    dept_id: 4,
    id: 5
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4ff'),
    name: 'Harry Johnson',
    age: 34,
    dept_id: 1,
    id: 7
  }
]
```

5. Retrieve employees sorted by age in ascending order

```
company> db.employee.find().sort({ age: 1 });
[
  {
    _id: ObjectId('65a6173f0756174d3330a4fe'),
    name: 'Grace Taylor',
    age: 28,
    dept_id: 5,
    id: 6
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fa'),
    name: 'Charlie Brown',
    age: 30,
    dept_id: 1,
    id: 3
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fc'),
    name: 'Emma Davis',
    age: 32,
    dept_id: 4,
    id: 5
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4ff'),
    name: 'Harry Johnson',
    age: 34,
    dept_id: 1,
    id: 7
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fb'),
    name: 'David Miller',
    age: 36,
    dept_id: 3,
    id: 4
  }
]
```

6. Retrieve employees belonging to a specific department with their department details (using \$lookup)

```
company> db.employee.aggregate([{$lookup: {from: "department", localField: "dept_id", foreignField: "_id", as: "department" } }]);
[
  {
    _id: ObjectId('65a6173f0756174d3330a4fa'),
    name: 'Charlie Brown',
    age: 30,
    dept_id: 1,
    id: 3,
    department: [
      {
        _id: 1,
        name: 'Information Technology',
        location: 'San Francisco',
        id: 1
      }
    ]
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fb'),
    name: 'David Miller',
    age: 36,
    dept_id: 3,
    id: 4,
    department: [ { _id: 3, name: 'Finance', location: 'Los Angeles', id: 3 } ]
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fc'),
    name: 'Emma Davis',
    age: 32,
    dept_id: 4,
    id: 5,
    department: [
      {
        _id: 4,
        name: 'Marketing',
        location: 'San Francisco',
        id: null
      }
    ]
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fe'),
    name: 'Grace Taylor',
    age: 28,
    dept_id: 5,
    id: 6,
    department: [ { _id: 5, name: 'Operations', location: 'Dallas', id: 5 } ]
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4ff'),
    name: 'Harry Johnson',
    age: 34,
    dept_id: 1,
    id: 7,
    department: [
      {
        _id: 1,
        name: 'Information Technology',
        location: 'San Francisco',
        id: 1
      }
    ]
  }
]
```

7. Retrieve employees sorted by department and age

```
company> db.employee.find().sort({ dept_id: 1, age: 1 });
[
  {
    _id: ObjectId('65a6173f0756174d3330a4fa'),
    name: 'Charlie Brown',
    age: 30,
    dept_id: 1,
    id: 3
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4ff'),
    name: 'Harry Johnson',
    age: 34,
    dept_id: 1,
    id: 7
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fb'),
    name: 'David Miller',
    age: 36,
    dept_id: 3,
    id: 4
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fc'),
    name: 'Emma Davis',
    age: 32,
    dept_id: 4,
    id: 5
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fe'),
    name: 'Grace Taylor',
    age: 28,
    dept_id: 5,
    id: 6
  }
]
```

8. Increment the age of all employees by 1 year

```

company> db.employee.find()
[
  {
    _id: ObjectId('65a6173f0756174d3330a4fa'),
    name: 'Charlie Brown',
    age: 32,
    dept_id: 1,
    id: 3
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fb'),
    name: 'David Miller',
    age: 38,
    dept_id: 3,
    id: 4
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fc'),
    name: 'Emma Davis',
    age: 34,
    dept_id: 4,
    id: 5
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fe'),
    name: 'Grace Taylor',
    age: 30,
    dept_id: 5,
    id: 6
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4ff'),
    name: 'Harry Johnson',
    age: 36,
    dept_id: 1,
    id: 7
  }
]
company> db.employee.updateMany({}, { $inc: { age: 1 } });
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 5,
  modifiedCount: 5,
  upsertedCount: 0
}
company> db.employee.find()
[
  {
    _id: ObjectId('65a6173f0756174d3330a4fa'),
    name: 'Charlie Brown',
    age: 33,
    dept_id: 1,
    id: 3
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fb'),
    name: 'David Miller',
    age: 39,
    dept_id: 3,
    id: 4
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fc'),
    name: 'Emma Davis',
    age: 35,
    dept_id: 4,
    id: 5
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fe'),
    name: 'Grace Taylor',
    age: 31,
    dept_id: 5,
    id: 6
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4ff'),
    name: 'Harry Johnson',
    age: 37,
    dept_id: 1,
    id: 7
  }
]

```

9. Rename a department (e.g., rename "IT" department to "Information Technology")

```
company> db.department.update({ name: "IT" }, { $set: { name: "Information Technology" } });
DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
company> db.department.find()
[
  { _id: 1, name: 'Information Technology', location: 'New York' },
  { _id: 2, name: 'HR', location: 'Chicago' },
  { _id: 3, name: 'Finance', location: 'Los Angeles' },
  { _id: 4, name: 'Marketing', location: 'San Francisco' },
  { _id: 5, name: 'Operations', location: 'Dallas' }
]
```

10. Update an employee's information (e.g., update Bob Smith's age)

```
company> db.employee.update({ name: "Bob Smith" }, { $set: { age: 33 } });
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 0,
  upsertedCount: 0
}
company> db.employee.find({name:'Bob Smith'});
[
  {
    _id: ObjectId('65a6173f0756174d3330a4f9'),
    name: 'Bob Smith',
    age: 33,
    dept_id: 2
  }
]
```

11. Update a department's location (e.g., update IT department's location to "San Francisco")

```
company> db.department.find();
[
  { _id: 2, name: 'HR', location: 'Chicago' },
  { _id: 3, name: 'Finance', location: 'Los Angeles' },
  { _id: 4, name: 'Marketing', location: 'San Francisco', id: null },
  { _id: 5, name: 'Operations', location: 'Dallas' },
  { _id: 1, name: 'IT', location: 'New York' }
]
company> db.department.update({ name: "IT" }, { $set: { name: "Information Technology" } });
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
company> db.department.update({ name: "Information Technology" }, { $set: { location: "San Francisco" } });
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
company> db.department.find();
[
  { _id: 2, name: 'HR', location: 'Chicago' },
  { _id: 3, name: 'Finance', location: 'Los Angeles' },
  { _id: 4, name: 'Marketing', location: 'San Francisco', id: null },
  { _id: 5, name: 'Operations', location: 'Dallas' },
  { _id: 1, name: 'Information Technology', location: 'San Francisco' }
]
```

12. Delete a specific employee by their ID

```
company> db.employee.deleteOne({ _id: ObjectId('65a62699bb66225202cbbfcc') });
{ acknowledged: true, deletedCount: 1 }
```

13. Delete a specific department by its ID

```

company> db.department.find();
[
  { _id: 3, name: 'Finance', location: 'Los Angeles', id: 3 },
  { _id: 4, name: 'Marketing', location: 'San Francisco', id: null },
  { _id: 5, name: 'Operations', location: 'Dallas', id: 5 },
  {
    _id: 1,
    name: 'Information Technology',
    location: 'San Francisco',
    id: 1
  }
]
company> db.department.deleteOne({_id:1});
{ acknowledged: true, deletedCount: 1 }
company> db.department.find();
[
  { _id: 3, name: 'Finance', location: 'Los Angeles', id: 3 },
  { _id: 4, name: 'Marketing', location: 'San Francisco', id: null },
  { _id: 5, name: 'Operations', location: 'Dallas', id: 5 }
]

```

14. Delete employees older than 40 years

```

company> db.employee.find()
[
  {
    _id: ObjectId('65a6173f0756174d3330a4fa'),
    name: 'Charlie Brown',
    age: 37,
    dept_id: 1,
    id: 3
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fb'),
    name: 'David Miller',
    age: 43,
    dept_id: 3,
    id: 4
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fc'),
    name: 'Emma Davis',
    age: 39,
    dept_id: 4,
    id: 5
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fe'),
    name: 'Grace Taylor',
    age: 35,
    dept_id: 5,
    id: 6
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4ff'),
    name: 'Harry Johnson',
    age: 41,
    dept_id: 1,
    id: 7
  }
]
company> db.employee.deleteMany({ age: { $gt: 40 } });
{ acknowledged: true, deletedCount: 2 }
company> db.employee.find()
[
  {
    _id: ObjectId('65a6173f0756174d3330a4fa'),
    name: 'Charlie Brown',
    age: 37,
    dept_id: 1,
    id: 3
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fc'),
    name: 'Emma Davis',
    age: 39,
    dept_id: 4,
    id: 5
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fe'),
    name: 'Grace Taylor',
    age: 35,
    dept_id: 5,
    id: 6
  }
]

```

15. Delete employees not belonging to any department (where departmentId is null)

```

company> cursor.forEach(function(doc) {
...   var newValue = i;
...   db.employee.update(
...     { _id: doc._id },
...     { $set: { id: newValue } }
...   );
...   i++;});

```

```

company> db.employee.find()
[
  {
    _id: ObjectId('65a6173f0756174d3330a4f8'),
    name: 'Alice Johnson',
    age: 29,
    dept_id: 1,
    id: 1
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4f9'),
    name: 'Bob Smith',
    age: 33,
    dept_id: 2,
    id: 2
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fa'),
    name: 'Charlie Brown',
    age: 30,
    dept_id: 1,
    id: 3
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fb'),
    name: 'David Miller',
    age: 36,
    dept_id: 3,
    id: 4
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fc'),
    name: 'Emma Davis',
    age: 32,
    dept_id: 4,
    id: 5
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4fe'),
    name: 'Grace Taylor',
    age: 28,
    dept_id: 5,
    id: 6
  },
  {
    _id: ObjectId('65a6173f0756174d3330a4ff'),
    name: 'Harry Johnson',
    age: 34,
    dept_id: 1,
    id: 7
  },
  {
    _id: ObjectId('65a6173f0756174d3330a500'),
    name: 'Ivy Martinez',
    age: 37,
    dept_id: null,
    id: 8
  }
]

```

16. Delete an employee (e.g., delete Alice Johnson from the employees)

```

company> db.employee.find({ name: "Alice Johnson" });
[
  {
    _id: ObjectId('65a6173f0756174d3330a4f8'),
    name: 'Alice Johnson',
    age: 29,
    dept_id: 1,
    id: 1
  }
]
company> db.employee.deleteOne({ name: "Alice Johnson" });
{ acknowledged: true, deletedCount: 1 }
company> db.employee.find({ name: "Alice Johnson" });
[]
company> db.employee.find({ dept_id: 2 });
[
  {
    _id: ObjectId('65a6173f0756174d3330a4f9'),
    name: 'Bob Smith',
    age: 33,
    dept_id: 2,
    id: 2
  }
]

```


17. Delete employees in a specific department (e.g., delete all employees in HR department)

```
company> db.employee.find({ dept_id: 2 });
[
  {
    _id: ObjectId('65a6173f0756174d3338a4f9'),
    name: 'Bob Smith',
    age: 33,
    dept_id: 2,
    id: 2
  }
]
company> db.employee.deleteMany({ dept_id: 2 });
{ acknowledged: true, deletedCount: 1 }
company> db.employee.find({ dept_id: 2 });
```

18 .Delete a department (e.g., delete the HR department)

```
company> db.department.find({ name: "HR" });
[ { _id: 2, name: 'HR', location: 'Chicago', id: 2 } ]
company> db.department.deleteOne({ name: "HR" });
{ acknowledged: true, deletedCount: 1 }
company> db.department.find({ name: "HR" });
```