**Experiment – 6: MongoDB**

| | |
|---|---|
| **Name of student** | **Arnav Santosh Sawant** |
| **Class Roll no** | **D15A - 52** |
| **D.O.P** | **06/03/25** |
| **D.O.S** | |
| **Sign and Grade** | |

1) **Aim:** To study CRUD operations in MongoDB
2) **Problem Statement:**
   A) Create a database, create a collection, insert data, query and manipulate data using various MongoDB operations.

      1. Create a database named "inventory".
      2. Create a collection named "products" with the fields: (ProductID, ProductName, Category, Price, Stock).
      3. Insert 10 documents into the "products" collection.
      4. Display all the documents in the "products" collection.
      5. Display all the products in the "Electronics" category.
      6. Display all the products in ascending order of their names.
      7. Display the details of the first 5 products.
      8. Display the categories of products with a specific name.
      9. Display the number of products in the "Electronics" category.
      10. Display all the products without showing the "_id" field.
      11. Display all the distinct categories of products.
      12. Display products in the "Electronics" category with prices greater than 50 but less than 100.
      13. Change the price of a product.
      14. Delete a particular product entry.

3) **Theory:**
   A. Describe some of the features of MongoDB?
      - NoSQL, document-based
      - Schema-less (flexible)
      - Scalable (horizontal scaling)
      - High performance
      - Built-in replication & sharding
        Powerful querying with indexing

   B. What are Documents and Collections in MongoDB?
      - **Document**: A JSON-like data record (BSON)
      - **Collection**: A group of related documents (like a table)

C. When to use MongoDB?
- When schema is dynamic
- For big, unstructured, or semi-structured data
- Real-time analytics, IoT, content management, mobile apps

D. What is Sharding in MongoDB?
- Splits large data across multiple servers
- Increases scalability and performance

4) **Output:**

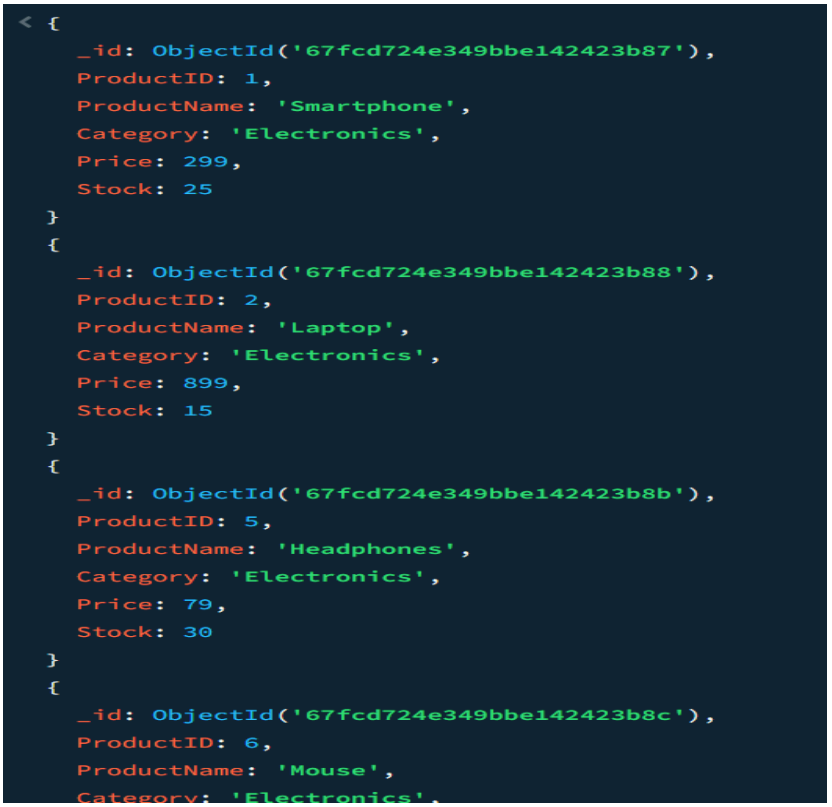
```
db.products.insertMany([

  { ProductID: 1, ProductName: "Smartphone", Category: "Electronics", Price: 299, Stock: 25 },

  { ProductID: 2, ProductName: "Laptop", Category: "Electronics", Price: 899, Stock: 15 },

  { ProductID: 3, ProductName: "Book", Category: "Stationery", Price: 15, Stock: 100 },

  { ProductID: 4, ProductName: "Pen", Category: "Stationery", Price: 2, Stock: 200 },

  { ProductID: 5, ProductName: "Headphones", Category: "Electronics", Price: 79, Stock: 30 },

  { ProductID: 6, ProductName: "Mouse", Category: "Electronics", Price: 25, Stock: 50 },

  { ProductID: 7, ProductName: "Chair", Category: "Furniture", Price: 49, Stock: 20 },

  { ProductID: 8, ProductName: "Table", Category: "Furniture", Price: 120, Stock: 10 },

  { ProductID: 9, ProductName: "Notebook", Category: "Stationery", Price: 5, Stock: 75 },

  { ProductID: 10, ProductName: "Keyboard", Category: "Electronics", Price: 45, Stock: 40 }

])
```

```
< {
    acknowledged: true,
    insertedIds: {
      '0': ObjectId('67fcd724e349bbe142423b87'),
      '1': ObjectId('67fcd724e349bbe142423b88'),
      '2': ObjectId('67fcd724e349bbe142423b89'),
      '3': ObjectId('67fcd724e349bbe142423b8a'),
      '4': ObjectId('67fcd724e349bbe142423b8b'),
      '5': ObjectId('67fcd724e349bbe142423b8c'),
      '6': ObjectId('67fcd724e349bbe142423b8d'),
      '7': ObjectId('67fcd724e349bbe142423b8e'),
      '8': ObjectId('67fcd724e349bbe142423b8f'),
      '9': ObjectId('67fcd724e349bbe142423b90')
    }
  }
```

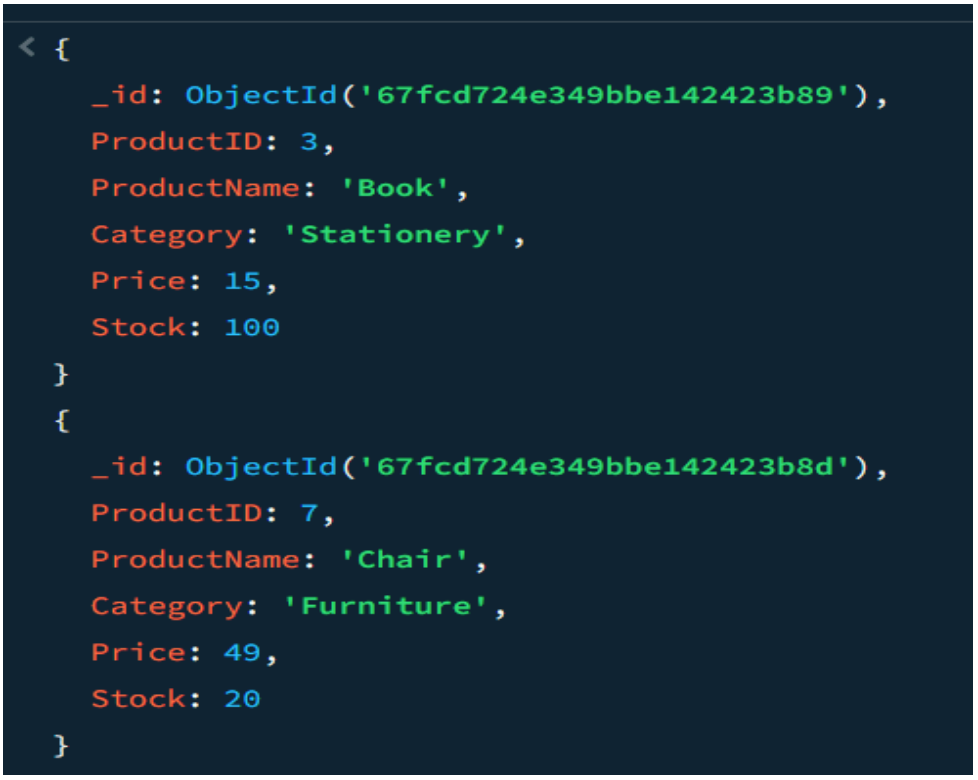
db.products.find().pretty()

```
< {
    _id: ObjectId('67fcd724e349bbe142423b87'),
    ProductID: 1,
    ProductName: 'Smartphone',
    Category: 'Electronics',
    Price: 299,
    Stock: 25
  }
  {
    _id: ObjectId('67fcd724e349bbe142423b88'),
    ProductID: 2,
    ProductName: 'Laptop',
    Category: 'Electronics',
    Price: 899,
    Stock: 15
  }
  {
    _id: ObjectId('67fcd724e349bbe142423b89'),
    ProductID: 3,
    ProductName: 'Book',
    Category: 'Stationery',
    Price: 15,
    Stock: 100
  }
  {
    _id: ObjectId('67fcd724e349bbe142423b8a'),
    ProductID: 4,
    ProductName: 'Pen',
```

db.products.find({ Category: "Electronics" }).pretty()




db.products.find().sort({ ProductName: 1 }).pretty()




db.products.find().limit(5).pretty()

```
< {
    _id: ObjectId('67fcd724e349bbe142423b87'),
    ProductID: 1,
    ProductName: 'Smartphone',
    Category: 'Electronics',
    Price: 299,
    Stock: 25
  }
  {
    _id: ObjectId('67fcd724e349bbe142423b88'),
    ProductID: 2,
    ProductName: 'Laptop',
    Category: 'Electronics',
    Price: 899,
    Stock: 15
  }
```

db.products.find({ ProductName: "Pen" }, { Category: 1, _id: 0 })
db.products.countDocuments({ Category: "Electronics" })
db.products.find({}, { _id: 0 }).pretty()

```
> db.products.find({ ProductName: "Pen" }, { Category: 1, _id: 0 })
< {
    Category: 'Stationery'
  }
> db.products.countDocuments({ Category: "Electronics" })
< 5
> db.products.find({}, { _id: 0 }).pretty()
< {
    ProductID: 1,
    ProductName: 'Smartphone',
    Category: 'Electronics',
    Price: 299,
    Stock: 25
  }
  {
    ProductID: 2,
    ProductName: 'Laptop',
    Category: 'Electronics',
    Price: 899,
    Stock: 15
  }
  {
```

db.products.distinct("Category")
db.products.find({ Category: "Electronics", Price: { $gt: 50, $lt: 100 } }).pretty()

db.products.updateOne({ ProductName: "Mouse" }, { $set: { Price: 30 } })
db.products.deleteOne({ ProductName: "Book" })