

Name: Aryan Sanjay Mangrule
Roll No.: B46
PRN: 2122000220
Subject: Advance Database System Lab
Experiment No.: 07

Install MongoDB Compass. Create and manage NoSQL Databases with MongoDB

Install MongoDB Compass

Problem Statement 1:

1. Create database: product
2. Create collection: inventory
3. Perform following operations on created collections:

Insert documents (one and many).

```
> db.inventory.insertOne({
  item: "apple",
  status: "A",
  qty: 20
})
< {
  acknowledged: true,
  insertedId: ObjectId('67342e2dcf347316856e6ba3')
}
```

```
> db.inventory.insertMany([
  { item: "banana", status: "A", qty: 15 },
  { item: "cherry", status: "B", qty: 25 },
  { item: "date", status: "D", qty: 10 }
])
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('67342d9fcf347316856e6ba0'),
    '1': ObjectId('67342d9fcf347316856e6ba1'),
    '2': ObjectId('67342d9fcf347316856e6ba2')
  }
}
```

Update documents (one and many).

```
> db.inventory.updateOne(
  { item: "apple" },
  { $set: { qty: 30 } }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}
```

```
> db.inventory.updateMany(
  { status: "B" },
  { $set: { status: "A" } }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Replace documents (one and many).

```
> db.inventory.replaceOne(
  { item: "apple" },
  { item: "apple", status: "A", qty: 50 }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

```
> db.inventory.updateMany(
  { status: "D" },
  { $set: { status: "A", qty: 40 } }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Delete documents (one and many).

```
> db.inventory.deleteOne({ item: "apple" })
< {
  acknowledged: true,
  deletedCount: 1
}
```

Find documents.

```
> db.inventory.find({ status: "A" })
< {
  _id: ObjectId('67342d9fcf347316856e6ba0'),
  item: 'banana',
  status: 'A',
  qty: 15
}
{
  _id: ObjectId('67342d9fcf347316856e6ba1'),
  item: 'cherry',
  status: 'A',
  qty: 25
}
{
  _id: ObjectId('67342d9fcf347316856e6ba2'),
  item: 'date',
  status: 'A',
  qty: 40
}
```

4. Use a filter to find documents in the database. Perform following queries in filter on inventory collection.

a. `SELECT * FROM inventory`

```
> db.inventory.find()
< {
  _id: ObjectId('67342d9fcf347316856e6ba0'),
  item: 'banana',
  status: 'A',
  qty: 15
}
{
  _id: ObjectId('67342d9fcf347316856e6ba1'),
  item: 'cherry',
  status: 'A',
  qty: 25
}
{
  _id: ObjectId('67342d9fcf347316856e6ba2'),
  item: 'date',
  status: 'A',
  qty: 40
}
```

b. `SELECT * FROM inventory WHERE status = "D"`

```
> db.inventory.find({ status: "D" })
<
```

c. `SELECT * FROM inventory WHERE status in ("A", "D")`

```
> db.inventory.find({ status: { $in: ["A", "D"] } })
< {
  _id: ObjectId('67342d9fcf347316856e6ba0'),
  item: 'banana',
  status: 'A',
  qty: 15
}
{
  _id: ObjectId('67342d9fcf347316856e6ba1'),
  item: 'cherry',
  status: 'A',
  qty: 25
}
{
  _id: ObjectId('67342d9fcf347316856e6ba2'),
  item: 'date',
  status: 'A',
  qty: 40
}
> db.inventory.find({
  $and: [
    { status: "A" },
    { qty: { $lt: 30 } }
  ]
})
```

d. `SELECT * FROM inventory WHERE status = "A" AND qty < 30`

```
> db.inventory.find({
  $and: [
    { status: "A" },
    { qty: { $lt: 30 } }
  ]
})
< {
  _id: ObjectId('67342d9fcf347316856e6ba0'),
  item: 'banana',
  status: 'A',
  qty: 15
}
{
  _id: ObjectId('67342d9fcf347316856e6ba1'),
  item: 'cherry',
  status: 'A',
  qty: 25
}
```

e. `SELECT * FROM inventory WHERE status = "A" OR qty < 30`

```
> db.inventory.find({
  $or: [
    { status: "A" },
    { qty: { $lt: 30 } }
  ]
})
< {
  _id: ObjectId('67342d9fcf347316856e6ba0'),
  item: 'banana',
  status: 'A',
  qty: 15
}
{
  _id: ObjectId('67342d9fcf347316856e6ba1'),
  item: 'cherry',
  status: 'A',
  qty: 25
}
{
  _id: ObjectId('67342d9fcf347316856e6ba2'),
  item: 'date',
  status: 'A',
  qty: 40
}
```

f. `SELECT * FROM inventory`
`WHERE status = "A" AND (qty < 30 OR item LIKE "p%")`

```
> db.inventory.find({
  $and: [
    { status: "A" },
    { $or: [
      { qty: { $lt: 30 } },
      { item: { $regex: "^p", $options: "i" } } // case-insensitive regex for "p%"
    ]}
  ]
})
< {
  _id: ObjectId('67342d9fcf347316856e6ba0'),
  item: 'banana',
  status: 'A',
  qty: 15
}
{
  _id: ObjectId('67342d9fcf347316856e6ba1'),
  item: 'cherry',
  status: 'A',
  qty: 25
}
```

Problem Statement 2:

1. Create collection: books under product database

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

2. Insert the following documents into a books collection:

```
{ "title": "1984", "author": "George Orwell", "year": 1949, "genre": "Dystopian" }
```

```
{ "title": "To Kill a Mockingbird", "author": "Harper Lee", "year": 1960, "genre": "Fiction" }
```

```
{ "title": "The Great Gatsby", "author": "F. Scott Fitzgerald", "year": 1925, "genre": "Fiction" }
```

```
{ "title": "Brave New World", "author": "Aldous Huxley", "year": 1932, "genre": "Dystopian" }
```

Add more such documents.

```
> db.books.insertMany([
  { "title": "1984", "author": "George Orwell", "year": 1949, "genre": "Dystopian" },
  { "title": "To Kill a Mockingbird", "author": "Harper Lee", "year": 1960, "genre": "Fiction" },
  { "title": "The Great Gatsby", "author": "F. Scott Fitzgerald", "year": 1925, "genre": "Fiction" },
  { "title": "Brave New World", "author": "Aldous Huxley", "year": 1932, "genre": "Dystopian" },
  { "title": "Fahrenheit 451", "author": "Ray Bradbury", "year": 1953, "genre": "Dystopian" },
  { "title": "Catch-22", "author": "Joseph Heller", "year": 1961, "genre": "Fiction" },
  { "title": "The Catcher in the Rye", "author": "J.D. Salinger", "year": 1951, "genre": "Fiction" }
])
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('67343367e581b0dcabc11be1'),
    '1': ObjectId('67343367e581b0dcabc11be2'),
    '2': ObjectId('67343367e581b0dcabc11be3'),
    '3': ObjectId('67343367e581b0dcabc11be4'),
    '4': ObjectId('67343367e581b0dcabc11be5'),
    '5': ObjectId('67343367e581b0dcabc11be6'),
    '6': ObjectId('67343367e581b0dcabc11be7')
  }
}
```


Find all books published after the year 1950.

```
> db.books.find({ year: { $gt: 1950 } })
< {
  _id: ObjectId('67343367e581b0dcabc11be2'),
  title: 'To Kill a Mockingbird',
  author: 'Harper Lee',
  year: 1960,
  genre: 'Fiction'
}
{
  _id: ObjectId('67343367e581b0dcabc11be5'),
  title: 'Fahrenheit 451',
  author: 'Ray Bradbury',
  year: 1953,
  genre: 'Dystopian'
}
{
  _id: ObjectId('67343367e581b0dcabc11be6'),
  title: 'Catch-22',
  author: 'Joseph Heller',
  year: 1961,
  genre: 'Fiction'
}
{
  _id: ObjectId('67343367e581b0dcabc11be7'),
  title: 'The Catcher in the Rye',
  author: 'J.D. Salinger',
  year: 1951,
  genre: 'Fiction'
}
```

Find all Dystopian books published before 1950.

```
> db.books.find({ genre: "Dystopian", year: { $lt: 1950 } })
< {
  _id: ObjectId('67343367e581b0dcabc11be1'),
  title: '1984',
  author: 'George Orwell',
  year: 1949,
  genre: 'Dystopian'
}
{
  _id: ObjectId('67343367e581b0dcabc11be4'),
  title: 'Brave New World',
  author: 'Aldous Huxley',
  year: 1932,
  genre: 'Dystopian'
}
```

Update the genre of "1984" to "Science Fiction".

```
> db.books.updateOne(
  { title: "1984" },
  { $set: { genre: "Science Fiction" } }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Delete all books in the "Fiction" genre.

```
> db.books.deleteMany({ genre: "Fiction" })
< {
  acknowledged: true,
  deletedCount: 4
}
```

Calculate the total number of books for each genre.

```
> db.books.aggregate([
  { $group: { _id: "$genre", count: { $sum: 1 } } }
])
< {
  _id: 'Dystopian',
  count: 2
}
{
  _id: 'Science Fiction',
  count: 1
}
```

Create an index on the author field to improve query performance.

```
> db.books.createIndex({ author: 1 })
< author_1
```


Retrieve all books sorted by year in ascending order.

```
> db.books.find().sort({ year: 1 })
< {
  _id: ObjectId('67343367e581b0dcabc11be4'),
  title: 'Brave New World',
  author: 'Aldous Huxley',
  year: 1932,
  genre: 'Dystopian'
}
{
  _id: ObjectId('67343367e581b0dcabc11be1'),
  title: '1984',
  author: 'George Orwell',
  year: 1949,
  genre: 'Science Fiction'
}
{
  _id: ObjectId('67343367e581b0dcabc11be5'),
  title: 'Fahrenheit 451',
  author: 'Ray Bradbury',
  year: 1953,
  genre: 'Dystopian'
}
```

Count the number of books written by "Harper Lee".

```
> db.books.countDocuments({ author: "Harper Lee" })
< 0
```

Retrieve only the titles and authors of all books.

```
> db.books.find({}, { title: 1, author: 1 })
< {
  _id: ObjectId('67343367e581b0dcabc11be1'),
  title: '1984',
  author: 'George Orwell'
}
{
  _id: ObjectId('67343367e581b0dcabc11be4'),
  title: 'Brave New World',
  author: 'Aldous Huxley'
}
{
  _id: ObjectId('67343367e581b0dcabc11be5'),
  title: 'Fahrenheit 451',
  author: 'Ray Bradbury'
}
```

Use filter to find documents in database. Perform following queries in filter on inventory collection.

- a. Find books published between 1930 and 1960.

```
> db.books.find({ year: { $gte: 1930, $lte: 1960 } })
< {
  _id: ObjectId('67343367e581b0dcabc11be1'),
  title: '1984',
  author: 'George Orwell',
  year: 1949,
  genre: 'Science Fiction'
}
{
  _id: ObjectId('67343367e581b0dcabc11be4'),
  title: 'Brave New World',
  author: 'Aldous Huxley',
  year: 1932,
  genre: 'Dystopian'
}
{
  _id: ObjectId('67343367e581b0dcabc11be5'),
  title: 'Fahrenheit 451',
  author: 'Ray Bradbury',
  year: 1953,
  genre: 'Dystopian'
}
```

Find books with titles containing the word "The".

```
> db.books.find({ title: { $regex: "The", $options: "i" } })  
<
```

Find all books published before 1950 and in the Fiction genre.

```
> db.books.find({ year: { $lt: 1950 }, genre: "Fiction" })  
<
```

Find all books not written by Aldous Huxley.

```
> db.books.find({ author: { $ne: "Aldous Huxley" } })  
< {  
  _id: ObjectId('67343367e581b0dcabc11be1'),  
  title: '1984',  
  author: 'George Orwell',  
  year: 1949,  
  genre: 'Science Fiction'  
}  
{  
  _id: ObjectId('67343367e581b0dcabc11be5'),  
  title: 'Fahrenheit 451',  
  author: 'Ray Bradbury',  
  year: 1953,  
  genre: 'Dystopian'  
}
```

