Day5 MongoDB Assignment

Shetty Meghana
shettymeghana273@gmail.com

Database Creation:

Database Name: Insurance

Created using MongoDB Compass by creating the first collection.

Collections:

1. **Customers:**

   ```
   {
    "customerId": 1,
    "firstName": "Ravi",
    "lastName": "Kumar",
    "dateOfBirth": "1995-06-12",
    "phone": "9876543210",
    "email": "ravi@gmail.com"
   }
   {
    "customerId": 2,
    "firstName": "Anita",
    "lastName": "Sharma",
    "dateOfBirth": "1998-03-22",
    "phone": "9123456789",
    "email": "anita@gmail.com"
   }
   ```

2. **Agents:**

   ```
   {
    "agentId": 101,
    "agentName": "Suresh Rao",
    "phone": "9988776655",
   ```

```
  "city": "Hyderabad"
}
{
  "agentId": 102,
  "agentName": "Meena Joshi",
  "phone": "9877665544",
  "city": "Bangalore"
}
```

3. **Policies:**

```
{
  "policyId": 1001,
  "policyName": "Health Secure",
  "policyType": "Health",
  "premiumAmount": 12000,
  "durationYears": 10
}
{
  "policyId": 1002,
  "policyName": "Life Shield",
  "policyType": "Life",
  "premiumAmount": 15000,
  "durationYears": 20
}
```

4. **Claims:**

```
{
  "claimId": 5001,
  "customerId": 1,
  "policyId": 1001,
  "agentId": 101,
  "claimDate": "2024-01-10",
  "claimMoney": 45000,
  "claimStatus": "Approved"
}
{
  "claimId": 5002,
  "customerId": 2,
  "policyId": 1002,
  "agentId": 102,
  "claimDate": "2023-11-05",
  "claimMoney": 25000,
  "claimStatus": "Rejected"
}
```

In the MongoDB design, the policyassignments and claims tables are combined into a single claims collection.

Basic Queries:

1. Display all customers:
   ```
   db.customers.find()
   ```

2. Display all agents:
   ```
   db.agents.find()
   ```

Conditional Queries

1. Find claims that are Approved:
   ```
   db.claims.find({ claimStatus: "Approved" })
   ```

2. Find claims with amount greater than 30,000:
   ```
   db.claims.find({ claimMoney: { $gt: 30000 } })
   ```

3. Find customers from customers collection using email:
   ```
   db.customers.find({ email: "ravi@gmail.com" })
   ```

CRUD Operations:

Insert:

```
db.customers.insertOne({

 customerId: 3,

 firstName: "Rahul",

 lastName: "Verma",

 dateOfBirth: "1997-09-15",
```

```
 phone: "9012345678",
 email: "rahul@gmail.com"
})
```

Update:

```
db.customers.updateOne(
 { customerId: 3 },
 { $set: { phone: "8888888888" } }
)
```

Delete:

```
db.claims.deleteOne({ claimId: 5003 })
```

Filtering Queries:

1. **Find customers born after 1996:**
   ```
   db.customers.find({ dateOfBirth: { $gt: "1996-01-01" } })
   ```

2. **Find agents from Hyderabad:**
   ```
   db.agents.find({ city: "Hyderabad" })
   ```

3. **Find claims between 20,000 and 50,000:**
   ```
   db.claims.find({
     claimMoney: { $gte: 20000, $lte: 50000 }
   })
   ```

Aggregation Queries:

1. **Total number of claims per status:**
   ```
   db.claims.aggregate([
    {
     $group: {
      _id: "$claimStatus",
      totalClaims: { $sum: 1 }
   ```

```
    }
  }
])
```