

## **Day - 5 MongoDB Assignment:**

### **Listing databases and Creating InsuranceDB**

```
insuranceDB> show dbs
```

- admin 40.00 KiB
- college 80.00 KiB
- config 108.00 KiB
- insuracneDB 8.00 KiB
- local 64.00 KiB

```
insuranceDB> use insuranceDB
```

### **Creating InsuranceDB collections:**

```
db.customers.insertMany([  
  {  
    customerID: 1,  
    firstName: "Virat",  
    lastName: "Kohli",  
    DOB: ISODate("2000-11-05"),  
    phone: "9876543210",  
    email: "virat.kohli@gmail.com"  
  },  
  {  
    customerID: 2,  
    firstName: "Rohit",  
    lastName: "Sharma",  
    DOB: ISODate("2003-03-15"),  
    phone: "9876543211",  
    email: "rohit.sharma@gmail.com"  
  },  
])
```

```
{  
    customerID: 3,  
    firstName: "Sachin",  
    lastName: "Tendulkar",  
    DOB: ISODate("2010-07-20"),  
    phone: "9876543212",  
    email: "sachin.t@gmail.com"  
},  
{  
    customerID: 4,  
    firstName: "MS",  
    lastName: "Dhoni",  
    DOB: ISODate("1995-11-25"),  
    phone: "9876543213",  
    email: "ms.dhoni@gmail.com"  
},  
{  
    customerID: 5,  
    firstName: "Hardik",  
    lastName: "Pandya",  
    DOB: ISODate("2005-01-05"),  
    phone: "9876543214",  
    email: "hardik.pandya@gmail.com"  
}  
])  
{  
    acknowledged: true,  
    insertedIds: {
```

```
'0': ObjectId('695763a2c61f3782be1e2631'),  
'1': ObjectId('695763a2c61f3782be1e2632'),  
'2': ObjectId('695763a2c61f3782be1e2633'),  
'3': ObjectId('695763a2c61f3782be1e2634'),  
'4': ObjectId('695763a2c61f3782be1e2635')  
}  
}
```

#### **Agents Collections:**

```
db.agents.insertMany([  
{  
    agentID: 1,  
    agentName: "Rahul Dravid",  
    phone: "9123456780",  
    city: "Hyderabad"  
},  
{  
    agentID: 2,  
    agentName: "Anil Kumble",  
    phone: "9123456781",  
    city: "Nagpur"  
},  
{  
    agentID: 3,  
    agentName: "Kapil Dev",  
    phone: "9123456782",  
    city: "Jaipur"  
}  
])
```

**Policies Collections:**

```
db.policies.insertMany([
  {
    policyID: 1,
    policyName: "Kohli Life Secure",
    policyType: "Life",
    premiumAmount: 15000,
    durationYears: 1
  },
  {
    policyID: 2,
    policyName: "Rohit Health Plus",
    policyType: "Health",
    premiumAmount: 12000,
    durationYears: 1
  },
  {
    policyID: 3,
    policyName: "Bumrah Motor Shield",
    policyType: "Motor",
    premiumAmount: 8000,
    durationYears: 1
  }
])
```

```
{  
    policyID: 4,  
    policyName: "Dhoni Health Gold",  
    policyType: "Health",  
    premiumAmount: 20000,  
    durationYears: 2  
}  
])
```

**Output:**

```
{  
    acknowledged: true,  
    insertedIds: {  
        '0': ObjectId('695763c9c61f3782be1e2636'),  
        '1': ObjectId('695763c9c61f3782be1e2637'),  
        '2': ObjectId('695763c9c61f3782be1e2638')  
    }  
}
```

**Policies Collection:**

```
db.policies.insertMany([  
    {  
        policyID: 1,  
        policyName: "Kohli Life Secure",  
        policyType: "Life",  
        premiumAmount: 15000,  
        durationYears: 1  
},
```

```
{  
    policyID: 2,  
    policyName: "Rohit Health Plus",  
    policyType: "Health",  
    premiumAmount: 12000,  
    durationYears: 1  
},  
{  
    policyID: 3,  
    policyName: "Bumrah Motor Shield",  
    policyType: "Motor",  
    premiumAmount: 8000,  
    durationYears: 1  
},  
{  
    policyID: 4,  
    policyName: "Dhoni Health Gold",  
    policyType: "Health",  
    premiumAmount: 20000,  
    durationYears: 2  
}  
])
```

**Output:**

```
{  
    acknowledged: true,  
    insertedIds: {  
        '0': ObjectId('69576414c61f3782be1e2639'),  
        '1': ObjectId('69576414c61f3782be1e263a'),  
    }  
}
```

```
'2': ObjectId('69576414c61f3782be1e263b'),  
'3': ObjectId('69576414c61f3782be1e263c')  
}  
}
```

### **Policy Assignments Collection:**

```
db.policyassignments.insertMany([  
{  
    assignmentID: 1,  
    customerID: 5,  
    policyID: 1,  
    agentID: 1,  
    startDate: ISODate("2023-01-01"),  
    endDate: ISODate("2024-01-01")  
},  
{  
    assignmentID: 2,  
    customerID: 2,  
    policyID: 2,  
    agentID: 2,  
    startDate: ISODate("2024-02-01"),  
    endDate: ISODate("2025-02-01")  
},  
{  
    assignmentID: 3,  
    customerID: 1,  
    policyID: 3,  
    agentID: 3,  
    startDate: ISODate("2023-06-01"),
```

```
        endDate: ISODate("2024-06-01")  
    },  
    {  
        assignmentID: 4,  
        customerID: 4,  
        policyID: 4,  
        agentID: 1,  
        startDate: ISODate("2024-01-01"),  
        endDate: ISODate("2026-01-01")  
    }  
])
```

**Output:**

```
{  
    acknowledged: true,  
    insertedIds: {  
        '0': ObjectId('695764cbc61f3782be1e263d'),  
        '1': ObjectId('695764cbc61f3782be1e263e'),  
        '2': ObjectId('695764cbc61f3782be1e263f'),  
        '3': ObjectId('695764cbc61f3782be1e2640')  
    }  
}
```

**Claims Collection:**

```
db.claims.insertMany([
  {
    claimID: 1,
    assignmentID: 1,
    claimDate: ISODate("2023-05-10"),
    claimAmount: 30000,
    claimStatus: "Approved"
  },
  {
    claimID: 2,
    assignmentID: 2,
    claimDate: ISODate("2024-06-15"),
    claimAmount: 20000,
    claimStatus: "Rejected"
  },
  {
    claimID: 3,
    assignmentID: 4,
    claimDate: ISODate("2024-08-01"),
    claimAmount: 60000,
    claimStatus: "Approved"
  }
])
```

**Output:**

```
{  
    acknowledged: true,  
    insertedIds: {  
        '0': ObjectId('695764d3c61f3782be1e2641'),  
        '1': ObjectId('695764d3c61f3782be1e2642'),  
        '2': ObjectId('695764d3c61f3782be1e2643')  
    }  
}
```

**CRUD Operations:****BASIC READ QUERIES:****1. Display all customers**

```
insuranceDB> db.customers.find()  
[  
    {  
        _id: ObjectId('695763a2c61f3782be1e2631'),  
        customerID: 1,  
        firstName: 'Virat',  
        lastName: 'Kohli',  
        DOB: ISODate('1998-05-10T00:00:00.000Z'),  
        phone: '9876543210',  
        email: 'virat.kohli@gmail.com'  
    },  
    {  
        _id: ObjectId('695763a2c61f3782be1e2632'),  
        customerID: 2,  
        firstName: 'Rohit',  
        lastName: 'Sharma',  
        DOB: ISODate('2003-03-15T00:00:00.000Z'),  
        phone: '9876543211',  
        email: 'rohit.sharma@gmail.com'  
    },  
]
```

```
{
  _id: ObjectId('695763a2c61f3782be1e2633'),
  customerID: 3,
  firstName: 'Sachin',
  lastName: 'Tendulkar',
  DOB: ISODate('2010-07-20T00:00:00.000Z'),
  phone: '9876543212',
  email: 'sachin.t@gmail.com'
},
{
  _id: ObjectId('695763a2c61f3782be1e2634'),
  customerID: 4,
  firstName: 'MS',
  lastName: 'Dhoni',
  DOB: ISODate('1995-11-25T00:00:00.000Z'),
  phone: '9876543213',
  email: 'ms.dhoni@gmail.com'
},
{
  _id: ObjectId('695763a2c61f3782be1e2635'),
  customerID: 5,
  firstName: 'Hardik',
  lastName: 'Pandya',
  DOB: ISODate('2005-01-05T00:00:00.000Z'),
  phone: '9876543214',
  email: 'hardik.pandya@gmail.com'
}
]
```

## 2. Display all agents from Hyderabad

```
insuranceDB> db.agents.find({ city: "Hyderabad" })
[
  {
    _id: ObjectId('6957636ec61f3782be1e262e'),
    agentID: 1,
    agentName: 'Rahul Dravid',
```

```
    phone: '9123456780',  
    city: 'Hyderabad'  
  }  
]  
]
```

### 3. Display customers born after 2000

```
insuranceDB> db.customers.find({ DOB: { $gt: ISODate("2000-01-01") } })
```

```
[  
{  
  _id: ObjectId('695763a2c61f3782be1e2632'),  
  customerID: 2,  
  firstName: 'Rohit',  
  lastName: 'Sharma',  
  DOB: ISODate('2003-03-15T00:00:00.000Z'),  
  phone: '9876543211',  
  email: 'rohit.sharma@gmail.com'  
},  
{  
  _id: ObjectId('695763a2c61f3782be1e2633'),  
  customerID: 3,  
  firstName: 'Sachin',  

```

## **PROJECTIONS:**

### **1. Show only customer firstName and email**

```
insuranceDB> db.customers.find({}, { _id: 0, firstName: 1, email: 1 })  
[  
  { firstName: 'Virat', email: 'virat.kohli@gmail.com' },  
  { firstName: 'Rohit', email: 'rohit.sharma@gmail.com' },  
  { firstName: 'Sachin', email: 'sachin.t@gmail.com' },  
  { firstName: 'MS', email: 'ms.dhoni@gmail.com' },  
  { firstName: 'Hardik', email: 'hardik.pandya@gmail.com' }  
]
```

### **2. Show all fields except phone from agents**

```
insuranceDB> db.agents.find({}, { phone: 0 })  
[  
  {  
    _id: ObjectId('6957636ec61f3782be1e262e'),  
    agentID: 1,  
    agentName: 'Rahul Dravid',  
    city: 'Hyderabad'  
  },  
  {  
    _id: ObjectId('6957636ec61f3782be1e262f'),  
    agentID: 2,  
    agentName: 'Anil Kumble',  
    city: 'Nagpur'  
  },  
  {  
    _id: ObjectId('6957636ec61f3782be1e2630'),  
    agentID: 3,  
    agentName: 'Kapil Dev',  
    city: 'Jaipur'  
  }  
]
```

## **CONDITIONAL QUERIES:**

### **1.Policies with premiumAmount greater than 15000**

```
insuranceDB> db.policies.find({ premiumAmount: { $gt: 15000 } })  
[  
  {  
    _id: ObjectId('69576414c61f3782be1e263c'),  
    policyID: 4,  
    policyName: 'Health Gold',  
    policyType: 'Health',  
    premiumAmount: 20000,  
    durationYears: 2  
  }  
]
```

### **2.Claims with status Approved**

```
insuranceDB> db.claims.find({ claimStatus: "Approved" })  
[  
  {  
    _id: ObjectId('695764d3c61f3782be1e2641'),  
    claimID: 1,  
    assignmentID: 1,  
    claimDate: ISODate('2023-05-10T00:00:00.000Z'),  
    claimAmount: 30000,  
    claimStatus: 'Approved'  
  },  
  {  
    _id: ObjectId('695764d3c61f3782be1e2643'),  
    claimID: 3,  
    assignmentID: 4,  
    claimDate: ISODate('2024-08-01T00:00:00.000Z'),  
    claimAmount: 60000,  
    claimStatus: 'Approved'  
  }  
]
```

## **UPDATE AND DELETE:**

### **1. Update premium of policyID = 2**

```
insuranceDB> db.policies.updateOne( { policyID: 2 }, { $set: { premiumAmount: 14000 } })  
{  
  acknowledged: true,  
  insertedId: null,  
  matchedCount: 1,  
  modifiedCount: 1,  
  upsertedCount: 0  
}
```

### **2. Increase all Health policy premiums by 10%**

```
insuranceDB> db.policies.updateMany({ policyType: "Health" },  
{ $mul: { premiumAmount:1.1 } })  
{  
  acknowledged: true,  
  insertedId: null,  
  matchedCount: 2,  
  modifiedCount: 2,  
  upsertedCount: 0  
}
```

### **3. Delete claims with claimAmount less than 25000**

```
insuranceDB> db.claims.deleteMany({ claimAmount: { $lt: 25000 } })  
{ acknowledged: true, deletedCount: 1 }
```

## **SORT, LIMIT, COUNT:**

### **1 .Sort policies by premiumAmount (descending)**

```
insuranceDB> db.policies.find().sort({ premiumAmount: -1 })  
[  
  {  
    _id: ObjectId('69576414c61f3782be1e263c'),  
    policyID: 4,  
    policyName: 'Health Gold',
```

```
policyType: 'Health',
premiumAmount: 22000,
durationYears: 2
},
{
_id: ObjectId('69576414c61f3782be1e263a'),
policyID: 2,
policyName: 'Health Plus',
policyType: 'Health',
premiumAmount: 15400.000000000002,
durationYears: 1
},
{
_id: ObjectId('69576414c61f3782be1e2639'),
policyID: 1,
policyName: 'Life Secure',
policyType: 'Life',
premiumAmount: 15000,
durationYears: 1
},
{
_id: ObjectId('69576414c61f3782be1e263b'),
policyID: 3,
policyName: 'Motor Shield',
policyType: 'Motor',
premiumAmount: 8000,
durationYears: 1
}
]
```

## 2.Top 2 highest premium policies:

```
insuranceDB> db.policies.find().sort({ premiumAmount: -1 }).limit(2)
```

**Output:**

```
[  
  {  
    _id: ObjectId('69576414c61f3782be1e263c'),  
    policyID: 4,  
    policyName: 'Health Gold',  
    policyType: 'Health',  
    premiumAmount: 22000,  
    durationYears: 2  
  },  
  {  
    _id: ObjectId('69576414c61f3782be1e263a'),  
    policyID: 2,  
    policyName: 'Health Plus',  
    policyType: 'Health',  
    premiumAmount: 15400.00000000002,  
    durationYears: 1  
  }  
]
```

**3. Count total number of agents**

```
insuranceDB> db.agents.countDocuments()
```

**Output:**

3

## Aggregation & Grouping

### 1. Total premium amount of all policies

```
insuranceDB> db.policies.aggregate([
  {
    $group: {
      _id: null, totalPremium: { $sum: "$premiumAmount" }
    }
  }
])
```

#### Output:

```
[ { _id: null, totalPremium: 60400 } ]
```

### 2. Average premium amount per policy type

```
insuranceDB> db.policies.aggregate([
  {
    $group: {
      _id: "$policyType",
      avgPremium: { $avg: "$premiumAmount" }
    }
  }
])
```

#### Output:

```
[
  { _id: 'Health', avgPremium: 18700 },
  { _id: 'Life', avgPremium: 15000 },
  { _id: 'Motor', avgPremium: 8000 }
]
```

### **3. Number of policies per policy type**

```
db.policies.aggregate([
  {
    $group: {
      _id: "$policyType",
      policyCount: { $sum: 1 }
    }
  }
])
```

#### **Output:**

```
[
  { _id: 'Health', policyCount: 2 },
  { _id: 'Life', policyCount: 1 },
  { _id: 'Motor', policyCount: 1 }
]
```