MONGO DB COMMANDS

<div align="right">

RATNA JASHWANTH

</div>

test> **show dbs**
admin   40.00 KiB
config 72.00 KiB
local   64.00 KiB

Creation of Insurance DB
test> use insdb
switched to db insdb

Creation of Collections
insdb> db.createCollection("customers")
{ ok: 1 }
insdb> db.createCollection("agents")
{ ok: 1 }
insdb> db.createCollection("policies")
{ ok: 1 }
insdb> db.createCollection("policyAssignments")
{ ok: 1 }
insdb> db.createCollection("claims")
{ ok: 1 }

To view existing collections
insdb> show collections
agents
claims
customers
policies
policyAssignments

Insertion of sample date into collections
insdb> **db.customers.insertMany**([
... {
...   _id: 1,
...   firstName: "Ravi",
...   lastName: "Kumar",
...   dateOfBirth: ISODate("1998-05-10"),
...   phone: "9876543210",
...   email: "ravi@gmail.com"
... },
... {
...   _id: 2,
...   firstName: "Anita",
...   lastName: "Sharma",

```
...    dateOfBirth: ISODate("1995-03-15"),
...    phone: "9123456780",
...    email: "anita@gmail.com"
...  }
... ])
{ acknowledged: true, insertedIds: { '0': 1, '1': 2 } }

insdb> db.agents.insertMany([
...  {
...    _id: 101,
...    agentName: "Suresh",
...    phone: "9000011111",
...    city: "Hyderabad"
...  },
...  {
...    _id: 102,
...    agentName: "Mahesh",
...    phone: "9000022222",
...    city: "Bangalore"
...  }
... ])
{ acknowledged: true, insertedIds: { '0': 101, '1': 102 } }

insdb> db.policies.insertMany([
...  {
...    _id: 201,
...    policyName: "Health Plus",
...    policyType: "Health",
...    premiumAmount: 12000,
...    durationYears: 5
...  },
...  {
...    _id: 202,
...    policyName: "Life Secure",
...    policyType: "Life",
...    premiumAmount: 15000,
...    durationYears: 10
...  }
... ])
{ acknowledged: true, insertedIds: { '0': 201, '1': 202 } }

insdb> db.policyAssignments.insertMany([
...  {
...    _id: 301,
...    customerId: 1,
...    policyId: 201,
...    agentId: 101,
...    startDate: ISODate("2023-01-01"),
```

```
...      endDate: ISODate("2028-01-01")
...    },
...    {
...      _id: 302,
...      customerId: 2,
...      policyId: 202,
...      agentId: 102,
...      startDate: ISODate("2022-06-01"),
...      endDate: ISODate("2032-06-01")
...    }
... ])
{ acknowledged: true, insertedIds: { '0': 301, '1': 302 } }

insdb> db.claims.insertMany([
...    {
...      _id: 401,
...      assignmentId: 301,
...      claimDate: ISODate("2024-05-20"),
...      claimAmount: 25000,
...      claimStatus: "approved"
...    },
...    {
...      _id: 402,
...      assignmentId: 302,
...      claimDate: ISODate("2025-01-15"),
...      claimAmount: 30000,
...      claimStatus: "pending"
...    }
... ])
{ acknowledged: true, insertedIds: { '0': 401, '1': 402 } }
```

View data in customers

```
insdb> db.customers.find()
[
  {
    _id: 1,
    firstName: 'Ravi',
    lastName: 'Kumar',
    dateOfBirth: ISODate('1998-05-10T00:00:00.000Z'),
    phone: '9876543210',
    email: 'ravi@gmail.com'
  },
  {
    _id: 2,
    firstName: 'Anita',
    lastName: 'Sharma',
    dateOfBirth: ISODate('1995-03-15T00:00:00.000Z'),
```

```
      phone: '9123456780',
      email: 'anita@gmail.com'
  }
]
```

View data in agents

```
insdb> db.agents.find()
[
  {
    _id: 101,
    agentName: 'Suresh',
    phone: '9000011111',
    city: 'Hyderabad'
  },
  {
    _id: 102,
    agentName: 'Mahesh',
    phone: '9000022222',
    city: 'Bangalore'
  }
]
```

View data in policies

```
insdb> db.policies.find()
[
  {
    _id: 201,
    policyName: 'Health Plus',
    policyType: 'Health',
    premiumAmount: 12000,
    durationYears: 5
  },
  {
    _id: 202,
    policyName: 'Life Secure',
    policyType: 'Life',
    premiumAmount: 15000,
    durationYears: 10
  }
]
```

View data in policyAssignments

```
insdb> db.policyAssignments.find()
[
  {
    _id: 301,
```

```
    customerId: 1,
    policyId: 201,
    agentId: 101,
    startDate: ISODate('2023-01-01T00:00:00.000Z'),
    endDate: ISODate('2028-01-01T00:00:00.000Z')
  },
  {
    _id: 302,
    customerId: 2,
    policyId: 202,
    agentId: 102,
    startDate: ISODate('2022-06-01T00:00:00.000Z'),
    endDate: ISODate('2032-06-01T00:00:00.000Z')
  }
]
```

View data in claims

```
insdb> db.claims.find()
[
  {
    _id: 401,
    assignmentId: 301,
    claimDate: ISODate('2024-05-20T00:00:00.000Z'),
    claimAmount: 25000,
    claimStatus: 'approved'
  },
  {
    _id: 402,
    assignmentId: 302,
    claimDate: ISODate('2025-01-15T00:00:00.000Z'),
    claimAmount: 30000,
    claimStatus: 'pending'
  }
]
```

BASIC COMMANDS

1. Find Customer by Name

```
insdb> db.customers.find({ firstName: "Ravi" })
[
  {
    _id: 1,
    firstName: 'Ravi',
    lastName: 'Kumar',
```

```
    dateOfBirth: ISODate('1998-05-10T00:00:00.000Z'),
    phone: '9876543210',
    email: 'ravi@gmail.com'
  }
]
```

2. Policies taken by Customer

```
insdb> db.policyAssignments.find({ customerId: 1 })
[
  {
    _id: 301,
    customerId: 1,
    policyId: 201,
    agentId: 101,
    startDate: ISODate('2023-01-01T00:00:00.000Z'),
    endDate: ISODate('2028-01-01T00:00:00.000Z')
  }
]
```

3. Claims of a Customer

```
insdb> db.claims.find({ assignmentId: 301 })
[
  {
    _id: 401,
    assignmentId: 301,
    claimDate: ISODate('2024-05-20T00:00:00.000Z'),
    claimAmount: 25000,
    claimStatus: 'approved'
  }
]
```

4. Display only Customers firstname and phone number

```
insdb> db.customers.find(
...   {},
...   { firstName: 1, phone: 1, _id: 0 }
... )
[
  { firstName: 'Ravi', phone: '9876543210' },
  { firstName: 'Anita', phone: '9123456780' }
]
```

5. Find policies with premium greater than 13000

```
insdb> db.policies.find({premiumAmount:{$gt:13000}});
[
  {
```

```
  _id: 202,
  policyName: 'Life Secure',
  policyType: 'Life',
  premiumAmount: 15000,
  durationYears: 10
 }
]
```

6. Find claims amount between 20000 and 30000

```
insdb> db.claims.find({claimAmount:{$gte:20000,$lte:30000}});
[
 {
  _id: 401,
  assignmentId: 301,
  claimDate: ISODate('2024-05-20T00:00:00.000Z'),
  claimAmount: 25000,
  claimStatus: 'approved'
 },
 {
  _id: 402,
  assignmentId: 302,
  claimDate: ISODate('2025-01-15T00:00:00.000Z'),
  claimAmount: 30000,
  claimStatus: 'pending'
 }
]
```

7. Find health policies with premium > 10000

```
insdb> db.policies.find({
...   $and: [
...     { policyType: "Health" },
...     { premiumAmount: { $gt: 10000 } }
...   ]
... })
[
 {
  _id: 201,
  policyName: 'Health Plus',
  policyType: 'Health',
  premiumAmount: 12000,
  durationYears: 5
 }
]
```

8. Find agents from Hyderabad or Bangalore

```
insdb> db.agents.find({
```

```
...    $or: [{ city: "Hyderabad" }, { city: "Bangalore" }]
... })
[
  {
    _id: 101,
    agentName: 'Suresh',
    phone: '9000011111',
    city: 'Hyderabad'
  },
  {
    _id: 102,
    agentName: 'Mahesh',
    phone: '9000022222',
    city: 'Bangalore'
  }
]
```

9. Find policies with IDs 201 or 202

```
insdb> db.policies.find({ _id: { $in: [201, 202] } })
[
  {
    _id: 201,
    policyName: 'Health Plus',
    policyType: 'Health',
    premiumAmount: 12000,
    durationYears: 5
  },
  {
    _id: 202,
    policyName: 'Life Secure',
    policyType: 'Life',
    premiumAmount: 15000,
    durationYears: 10
  }
]
```

10.   Show claims sorted by amount (descending)

```
insdb> db.claims.find().sort({ claimAmount: -1 })
[
  {
    _id: 402,
    assignmentId: 302,
    claimDate: ISODate('2025-01-15T00:00:00.000Z'),
    claimAmount: 30000,
    claimStatus: 'pending'
  },
  {
    _id: 401,
```

```
    assignmentId: 301,
    claimDate: ISODate('2024-05-20T00:00:00.000Z'),
    claimAmount: 25000,
    claimStatus: 'approved'
  }
]
```

11.   Display top 1 highest claim

```
insdb> db.claims.find().sort({ claimAmount: -1 }).limit(1)
[
  {
    _id: 402,
    assignmentId: 302,
    claimDate: ISODate('2025-01-15T00:00:00.000Z'),
    claimAmount: 30000,
    claimStatus: 'pending'
  }
]
```

12.   Update claim status to approved

```
insdb> db.claims.updateOne(
...   { _id: 402 },
...   { $set: { claimStatus: "approved" } }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

13.   Increase premium by 1000 for all health policies

```
insdb> db.policies.updateMany(
...   { policyType: "Health" },
...   { $inc: { premiumAmount: 1000 } }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

14.   Remove phone number from agent

```
insdb> db.agents.updateOne(
...   { _id: 101 },
...   { $unset: { phone: "" } }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

15.   Delete claim with ID 401

```
insdb> db.claims.deleteOne({ _id: 401 })
{ acknowledged: true, deletedCount: 1 }
```

16.   Delete all pending claims

```
insdb> db.claims.deleteMany({ claimStatus: "pending" })
{ acknowledged: true, deletedCount: 0 }
```

17.   Count number of policies

```
insdb> db.policies.countDocuments()
2
```

18.   List distinct policy types

```
insdb> db.policies.distinct("policyType")
[ 'Health', 'Life' ]
```

19.   Find customers whose name starts with "A"

```
insdb> db.customers.find({ firstName: { $regex: "^A" } })
[
  {
    _id: 2,
    firstName: 'Anita',
    lastName: 'Sharma',
    dateOfBirth: ISODate('1995-03-15T00:00:00.000Z'),
    phone: '9123456780',
    email: 'anita@gmail.com'
  }
]
```

20.   Find policies started after 2023

```
insdb> db.policyAssignments.find({
...   startDate: { $gte: ISODate("2023-01-01") }
... })
[
  {
    _id: 301,
    customerId: 1,
    policyId: 201,
    agentId: 101,
    startDate: ISODate('2023-01-01T00:00:00.000Z'),
    endDate: ISODate('2028-01-01T00:00:00.000Z')
  }
]
```