Day-5 MongoDB Assignment

Submitted by Eada Jafrash

1. Starting MongoDB Shell

mongosh

2. Display Existing Databases

show dbs

3. Create and Switch to Database

use insDB

4. Create Collections

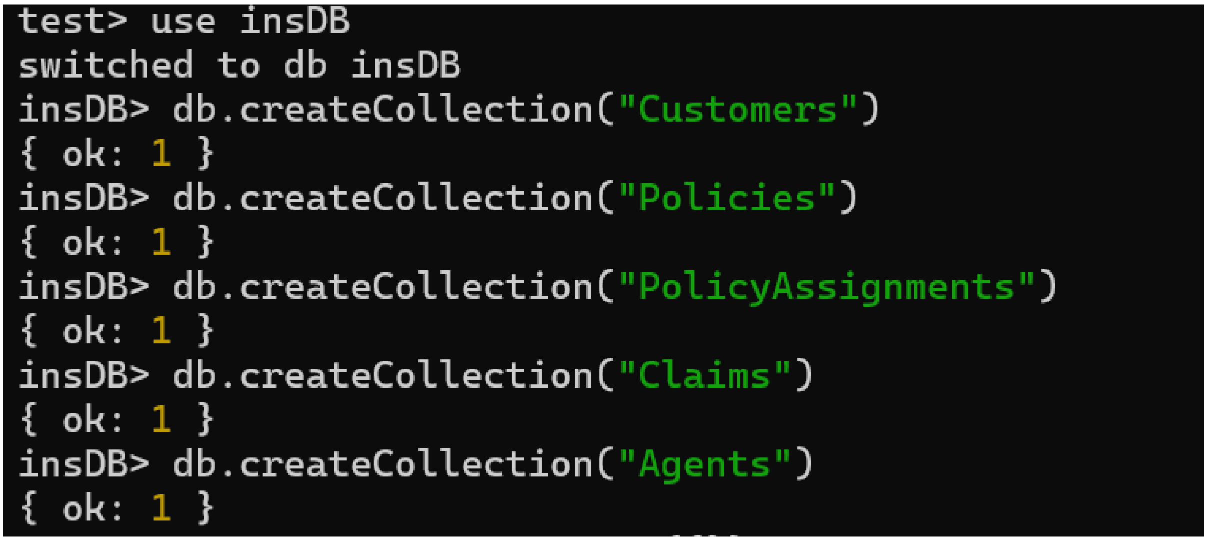
db.createCollection("Customers")

db.createCollection("Policies")

db.createCollection("Agents")

db.createCollection("PolicyAssignments")

db.createCollection("Claims")



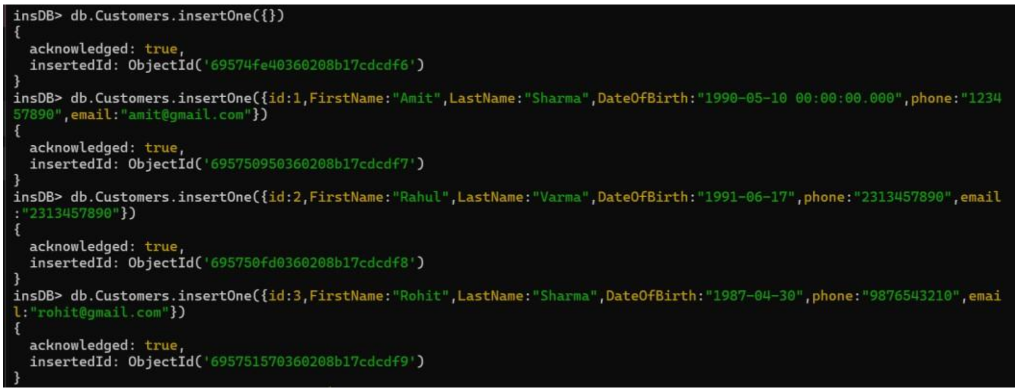
```
test> use insDB
switched to db insDB
insDB> db.createCollection("Customers")
{ ok: 1 }
insDB> db.createCollection("Policies")
{ ok: 1 }
insDB> db.createCollection("PolicyAssignments")
{ ok: 1 }
insDB> db.createCollection("Claims")
{ ok: 1 }
insDB> db.createCollection("Agents")
{ ok: 1 }
```

5. Insert Documents into Customers Collection

```
db.Customers.insertOne({
 id: 1,
 FirstName: "Amit",
 LastName: "Sharma",
 DateOfBirth: "1990-05-10",
 phone: "123457890",
 email: "amit@gmail.com"
})



db.Customers.insertOne({
 id: 2,
 FirstName: "Rahul",
 LastName: "Varma",
 DateOfBirth: "1991-06-17",
 phone: "2313457890",
 email: "rahul@gmail.com"
})

db.Customers.insertOne({
 id: 3,
 FirstName: "Rohit",
 LastName: "Sharma",
 DateOfBirth: "1987-04-30",
 phone: "9876543210",
 email: "rohit@gmail.com"
})
```

```
db.Customers.insertOne({
  id: 4,
  FirstName: "Virat",
  LastName: "Kohli",
  DateOfBirth: "1988-03-11",
  phone: "1029384756",
  email: "virat@gmail.com"
})
```

```
db.Customers.insertOne({
  id: 5,
  FirstName: "MS",
  LastName: "Dhoni",
  DateOfBirth: "1983-07-11",
  phone: "0192837465",
  email: "dhoni@gmail.com"
})
```

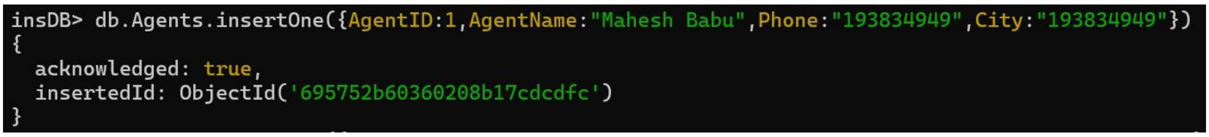


```
insDB> db.Customers.insertOne({})
{
  acknowledged: true,
  insertedId: ObjectId('69574fe40360208b17cdcdf6')
}
insDB> db.Customers.insertOne({id:1,FirstName:"Amit",LastName:"Sharma",DateOfBirth:"1990-05-10 00:00:00.000",phone:"1234
57890",email:"amit@gmail.com"})
{
  acknowledged: true,
  insertedId: ObjectId('695750950360208b17cdcdf7')
}
insDB> db.Customers.insertOne({id:2,FirstName:"Rahul",LastName:"Varma",DateOfBirth:"1991-06-17",phone:"2313457890",email
:"2313457890"})
{
  acknowledged: true,
  insertedId: ObjectId('695750fd0360208b17cdcdf8')
}
insDB> db.Customers.insertOne({id:3,FirstName:"Rohit",LastName:"Sharma",DateOfBirth:"1987-04-30",phone:"9876543210",emai
l:"rohit@gmail.com"})
{
  acknowledged: true,
  insertedId: ObjectId('695751570360208b17cdcdf9')
}
```

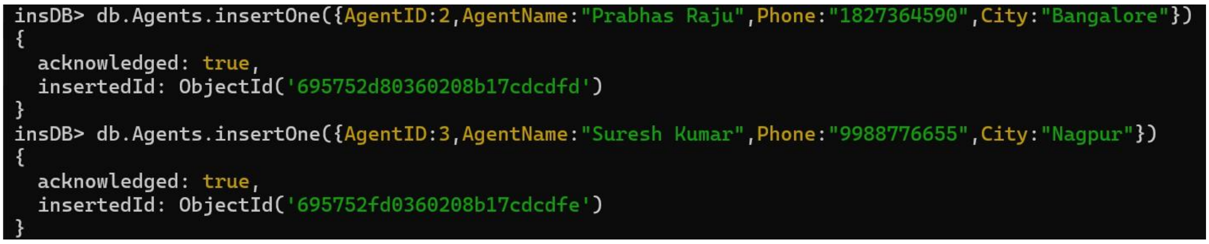
7. Insert Documents into Agents Collection

```
db.Agents.insertOne({
  AgentID: 101,
  AgentName: "Mahesh Babu",
  Phone: "193834949",
  City: "Hyderabad"
})
```

```
db.Agents.insertOne({
  AgentID: 102,
  AgentName: "Prabhas Raju",
  Phone: "1827364590",
  City: "Bangalore"
})

db.Agents.insertOne({
  AgentID: 103,
  AgentName: "Suresh Kumar",
  Phone: "9988776655",
  City: "Nagpur"
})
```



```
insDB> db.Agents.insertOne({AgentID:1,AgentName:"Mahesh Babu",Phone:"193834949",City:"193834949"})
{
  acknowledged: true,
  insertedId: ObjectId('695752b60360208b17cdcdfc')
}
```



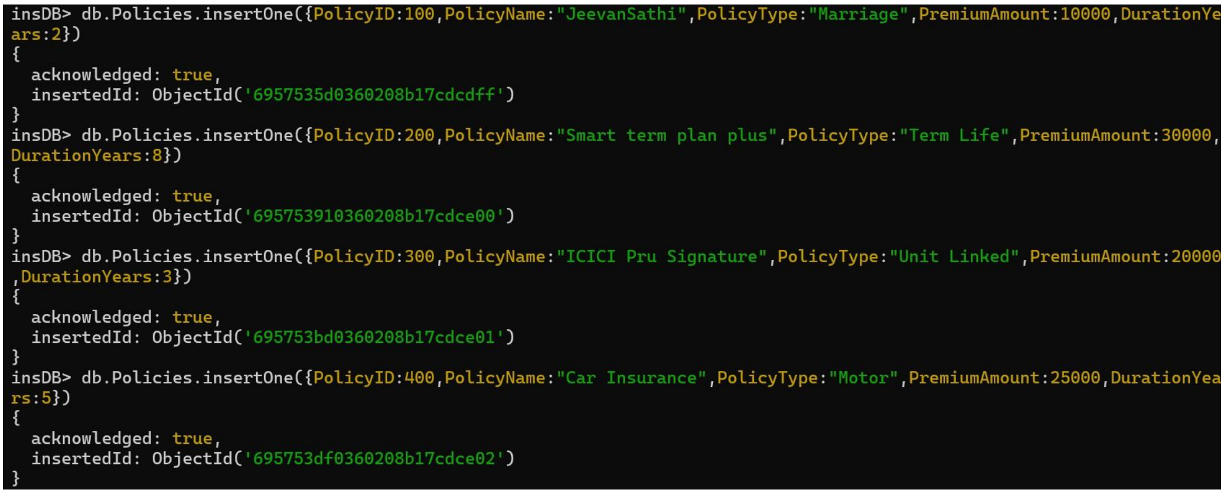
```
insDB> db.Agents.insertOne({AgentID:2,AgentName:"Prabhas Raju",Phone:"1827364590",City:"Bangalore"})
{
  acknowledged: true,
  insertedId: ObjectId('695752d80360208b17cdcdfd')
}
insDB> db.Agents.insertOne({AgentID:3,AgentName:"Suresh Kumar",Phone:"9988776655",City:"Nagpur"})
{
  acknowledged: true,
  insertedId: ObjectId('695752fd0360208b17cdcdfe')
}
```

8. Insert Documents into Policies Collection

```
db.Policies.insertOne({
  PolicyID: 100,
  PolicyName: "JeevanSathi",
  PolicyType: "Marriage",
  PremiumAmount: 10000,
  DurationYears: 2
})
```

```
db.Policies.insertOne({
  PolicyID: 200,
  PolicyName: "Smart Term Plan Plus",
  PolicyType: "Term Life",
  PremiumAmount: 30000,
  DurationYears: 8
})

db.Policies.insertOne({
  PolicyID: 300,
  PolicyName: "ICICI Pru Signature",
  PolicyType: "Unit Linked",
  PremiumAmount: 20000,
  DurationYears: 3
})

db.Policies.insertOne({
  PolicyID: 400,
  PolicyName: "Car Insurance",
  PolicyType: "Motor",
  PremiumAmount: 25000,
  DurationYears: 5
})
```

```
db.Policies.insertOne({
  PolicyID: 500,
  PolicyName: "Bike Insurance",
  PolicyType: "Motor",
  PremiumAmount: 9000,
  DurationYears: 3
})
```



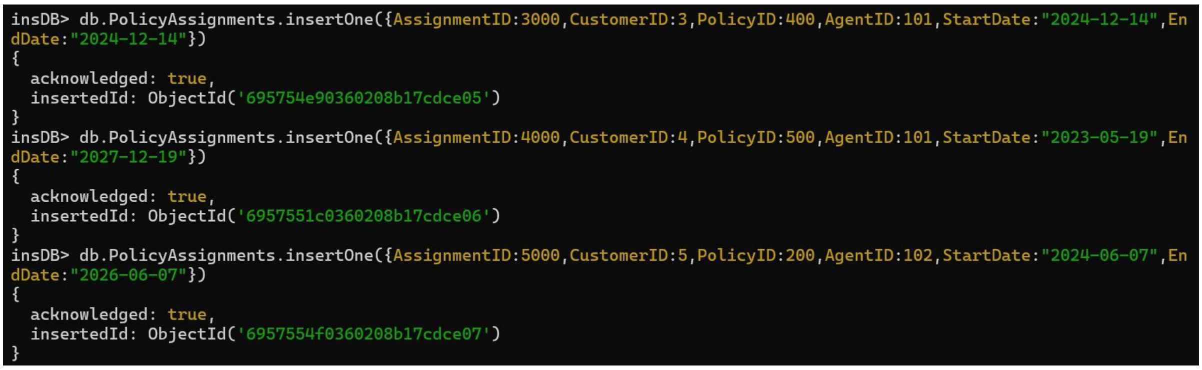
```
insDB> db.Policies.insertOne({PolicyID:100,PolicyName:"JeevanSathi",PolicyType:"Marriage",PremiumAmount:10000,DurationYe
ars:2})
{
  acknowledged: true,
  insertedId: ObjectId('6957535d0360208b17cdcdff')
}
insDB> db.Policies.insertOne({PolicyID:200,PolicyName:"Smart term plan plus",PolicyType:"Term Life",PremiumAmount:30000,
DurationYears:8})
{
  acknowledged: true,
  insertedId: ObjectId('695753910360208b17cdce00')
}
insDB> db.Policies.insertOne({PolicyID:300,PolicyName:"ICICI Pru Signature",PolicyType:"Unit Linked",PremiumAmount:20000
,DurationYears:3})
{
  acknowledged: true,
  insertedId: ObjectId('695753bd0360208b17cdce01')
}
insDB> db.Policies.insertOne({PolicyID:400,PolicyName:"Car Insurance",PolicyType:"Motor",PremiumAmount:25000,DurationYea
rs:5})
{
  acknowledged: true,
  insertedId: ObjectId('695753df0360208b17cdce02')
}
```

9. Insert Documents into PolicyAssignments

```
db.PolicyAssignments.insertOne({
  AssignmentID: 3000,
  CustomerID: 3,
  PolicyID: 400,
  AgentID: 101,
  StartDate: "2024-12-14",
  EndDate: "2029-12-14"
})
```

```
db.PolicyAssignments.insertOne({
AssignmentID: 4000,
  CustomerID: 4,
  PolicyID: 500,
  AgentID: 101,
  StartDate: "2023-05-19",
  EndDate: "2027-12-19"
})
```

```
db.PolicyAssignments.insertOne({
  AssignmentID: 5000,
  CustomerID: 5,
  PolicyID: 200,
  AgentID: 102,
  StartDate: "2024-06-07",
  EndDate: "2026-06-07"
```

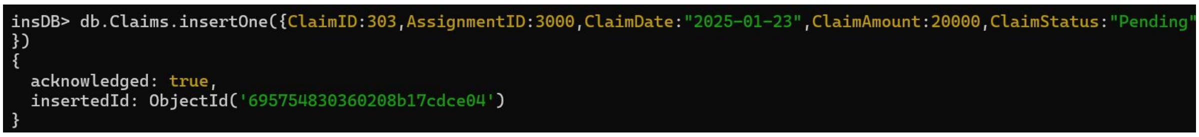


```
insDB> db.PolicyAssignments.insertOne({AssignmentID:3000,CustomerID:3,PolicyID:400,AgentID:101,StartDate:"2024-12-14",En
dDate:"2024-12-14"})
{
  acknowledged: true,
  insertedId: ObjectId('695754e90360208b17cdce05')
}
insDB> db.PolicyAssignments.insertOne({AssignmentID:4000,CustomerID:4,PolicyID:500,AgentID:101,StartDate:"2023-05-19",En
dDate:"2027-12-19"})
{
  acknowledged: true,
  insertedId: ObjectId('6957551c0360208b17cdce06')
}
insDB> db.PolicyAssignments.insertOne({AssignmentID:5000,CustomerID:5,PolicyID:200,AgentID:102,StartDate:"2024-06-07",En
dDate:"2026-06-07"})
{
  acknowledged: true,
  insertedId: ObjectId('6957554f0360208b17cdce07')
}
```

```
})
```

10. Insert Documents into Claims

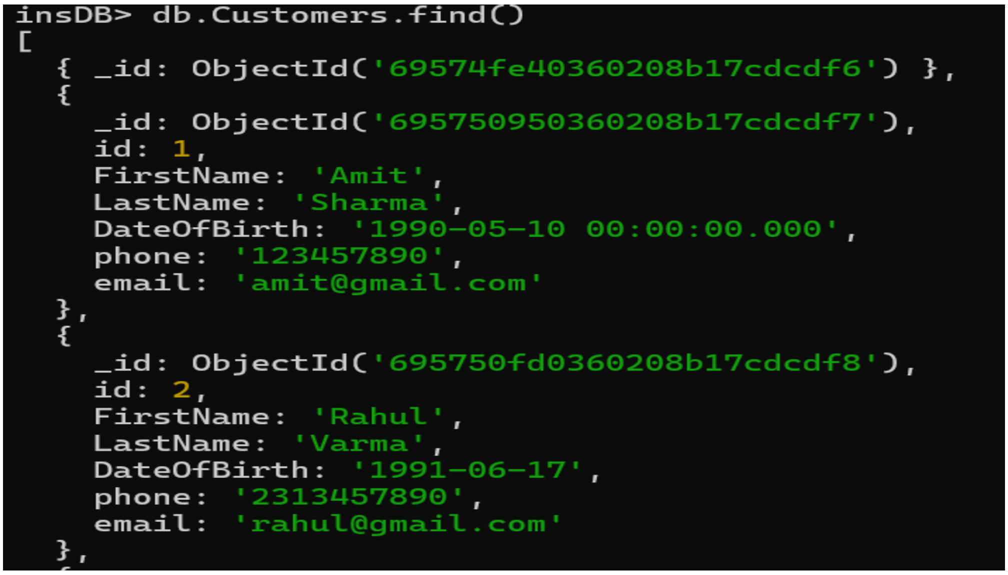
```
db.Claims.insertOne({
  ClaimID: 303,
  AssignmentID: 3000,
  ClaimDate: "2025-01-23",
  ClaimAmount: 20000,
  ClaimStatus: "Pending"
})
```



```
insDB> db.Claims.insertOne({ClaimID:303,AssignmentID:3000,ClaimDate:"2025-01-23",ClaimAmount:20000,ClaimStatus:"Pending"
})
{
  acknowledged: true,
  insertedId: ObjectId('695754830360208b17cdce04')
}
```

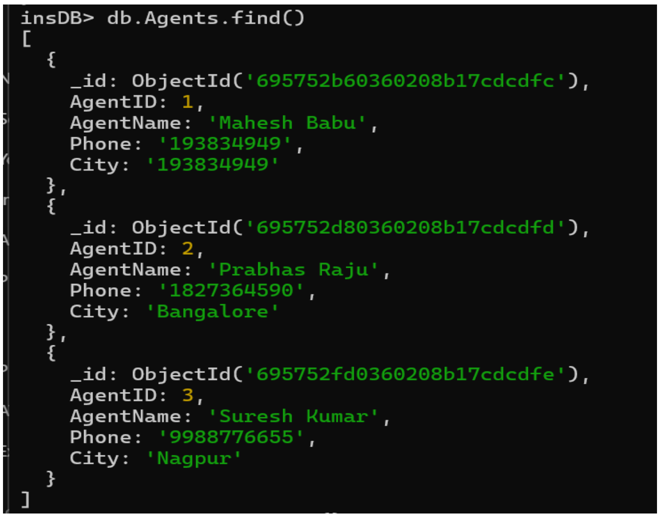
11. Retrieve Documents from Collections

db.Customers.find()



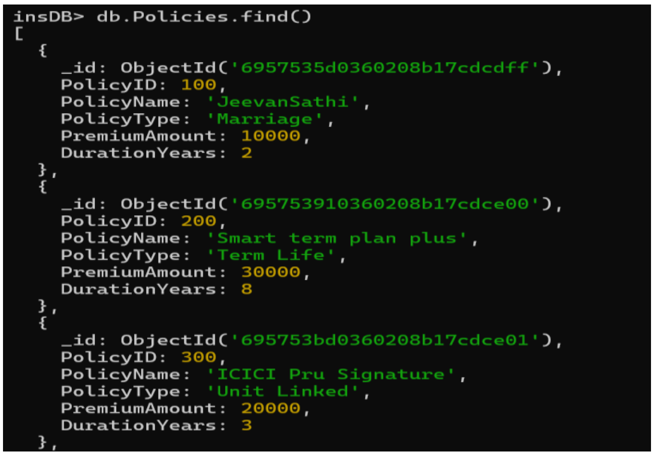
```
insDB> db.Customers.find()
[
  { _id: ObjectId('69574fe40360208b17cdcdf6') },
  {
    _id: ObjectId('695750950360208b17cdcdf7'),
    id: 1,
    FirstName: 'Amit',
    LastName: 'Sharma',
    DateOfBirth: '1990-05-10 00:00:00.000',
    phone: '123457890',
    email: 'amit@gmail.com'
  },
  {
    _id: ObjectId('695750fd0360208b17cdcdf8'),
    id: 2,
    FirstName: 'Rahul',
    LastName: 'Varma',
    DateOfBirth: '1991-06-17',
    phone: '2313457890',
    email: 'rahul@gmail.com'
  },
```

db.Agents.find()



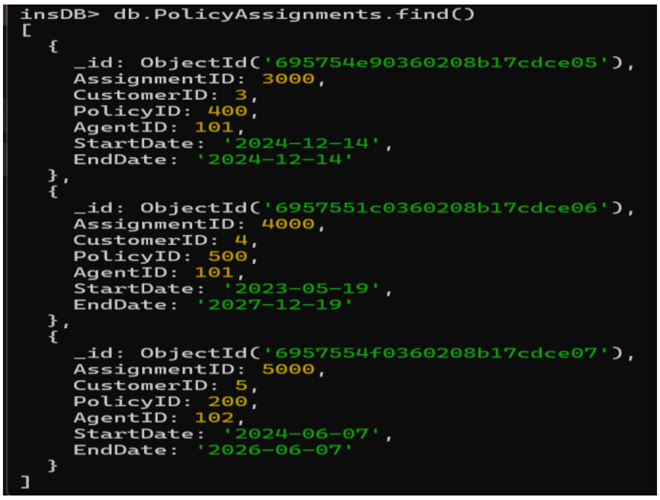
```
insDB> db.Agents.find()
[
  {
    _id: ObjectId('695752b60360208b17cdcdfc'),
    AgentID: 1,
    AgentName: 'Mahesh Babu',
    Phone: '193834949',
    City: '193834949'
  },
  {
    _id: ObjectId('695752d80360208b17cdcdfd'),
    AgentID: 2,
    AgentName: 'Prabhas Raju',
    Phone: '1827364590',
    City: 'Bangalore'
  },
  {
    _id: ObjectId('695752fd0360208b17cdcdfe'),
    AgentID: 3,
    AgentName: 'Suresh Kumar',
    Phone: '9988776655',
    City: 'Nagpur'
  }
]
```

db.Policies.find()



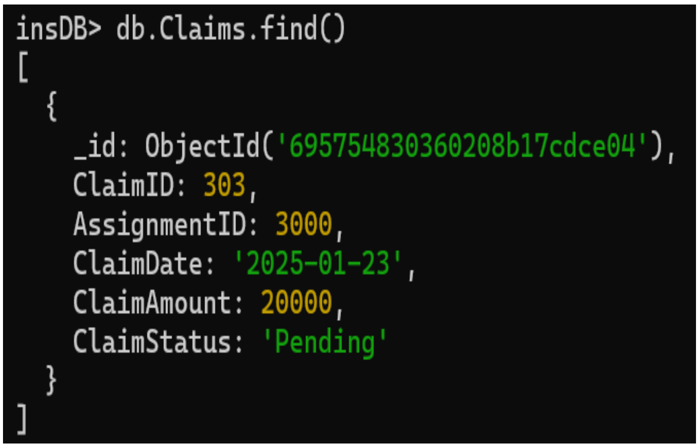
```
insDB> db.Policies.find()
[
  {
    _id: ObjectId('6957535d0360208b17cdcdff'),
    PolicyID: 100,
    PolicyName: 'JeevanSathi',
    PolicyType: 'Marriage',
    PremiumAmount: 10000,
    DurationYears: 2
  },
  {
    _id: ObjectId('695753910360208b17cdce00'),
    PolicyID: 200,
    PolicyName: 'Smart term plan plus',
    PolicyType: 'Term Life',
    PremiumAmount: 30000,
    DurationYears: 8
  },
  {
    _id: ObjectId('695753bd0360208b17cdce01'),
    PolicyID: 300,
    PolicyName: 'ICICI Pru Signature',
    PolicyType: 'Unit Linked',
    PremiumAmount: 20000,
    DurationYears: 3
  },
```

db.PolicyAssignments.find()



```
insDB> db.PolicyAssignments.find()
[
  {
    _id: ObjectId('695754e90360208b17cdce05'),
    AssignmentID: 3000,
    CustomerID: 3,
    PolicyID: 400,
    AgentID: 101,
    StartDate: '2024-12-14',
    EndDate: '2024-12-14'
  },
  {
    _id: ObjectId('6957551c0360208b17cdce06'),
    AssignmentID: 4000,
    CustomerID: 4,
    PolicyID: 500,
    AgentID: 101,
    StartDate: '2023-05-19',
    EndDate: '2027-12-19'
  },
  {
    _id: ObjectId('6957554f0360208b17cdce07'),
    AssignmentID: 5000,
    CustomerID: 5,
    PolicyID: 200,
    AgentID: 102,
    StartDate: '2024-06-07',
    EndDate: '2026-06-07'
  }
]
```

db.Claims.find()

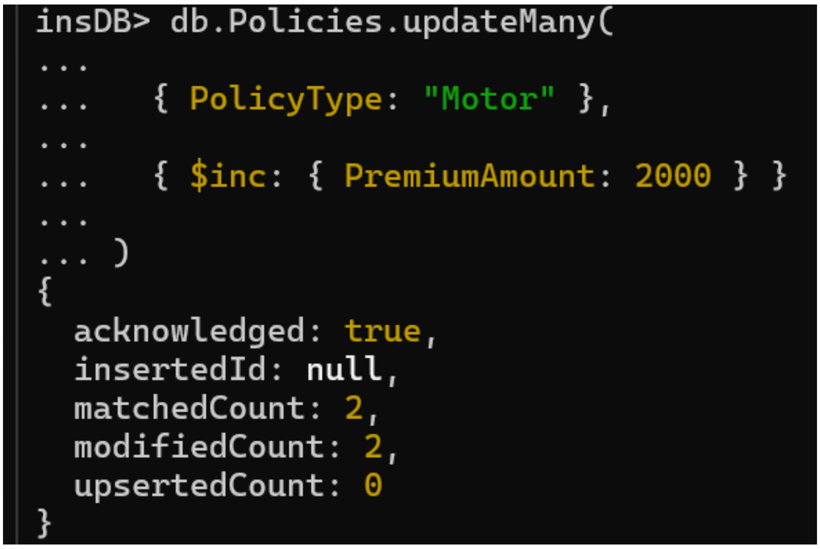


```
insDB> db.Claims.find()
[
  {
    _id: ObjectId('695754830360208b17cdce04'),
    ClaimID: 303,
    AssignmentID: 3000,
    ClaimDate: '2025-01-23',
    ClaimAmount: 20000,
    ClaimStatus: 'Pending'
  }
]
```

12. Update Multiple Documents

a) Increase premium by 2000 for all Motor policies

```
db.Policies.updateMany(
  { PolicyType: "Motor" },
  { $inc: { PremiumAmount: 2000 } }
)
```

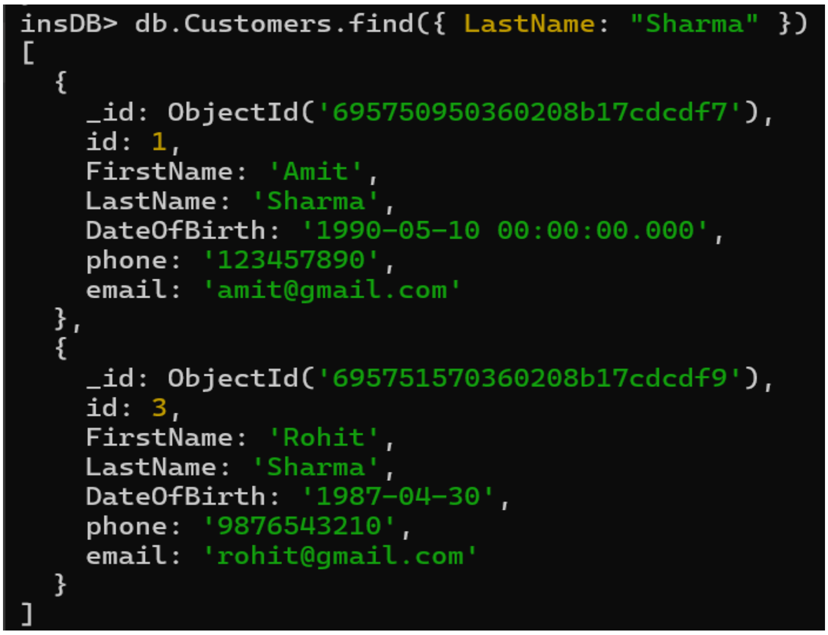


```
insDB> db.Policies.updateMany(
...
...   { PolicyType: "Motor" },
...
...   { $inc: { PremiumAmount: 2000 } }
...
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 2,
  modifiedCount: 2,
  upsertedCount: 0
}
```

13. Find with Condition (Filtering)

a) Find customers with last name "Sharma"

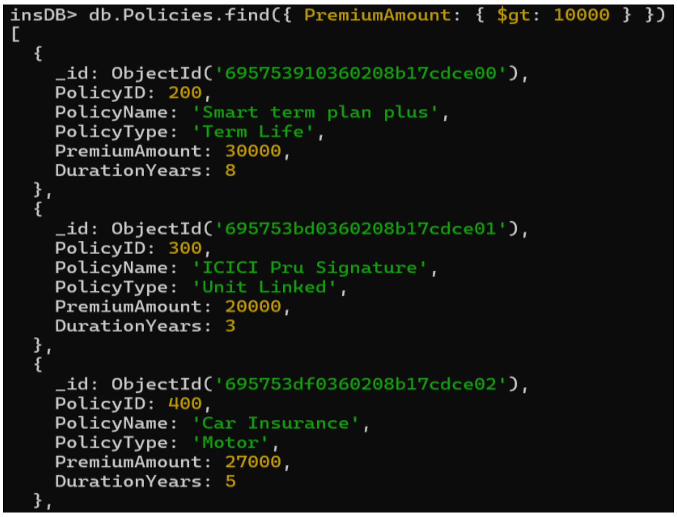
db.Customers.find({ LastName: "Sharma" })



```
insDB> db.Customers.find({ LastName: "Sharma" })
[
  {
    _id: ObjectId('695750950360208b17cdcdf7'),
    id: 1,
    FirstName: 'Amit',
    LastName: 'Sharma',
    DateOfBirth: '1990-05-10 00:00:00.000',
    phone: '123457890',
    email: 'amit@gmail.com'
  },
  {
    _id: ObjectId('695751570360208b17cdcdf9'),
    id: 3,
    FirstName: 'Rohit',
    LastName: 'Sharma',
    DateOfBirth: '1987-04-30',
    phone: '9876543210',
    email: 'rohit@gmail.com'
  }
]
```

b) Find policies with premium greater than 10,000

db.Policies.find({ PremiumAmount: { $gt: 10000 } })

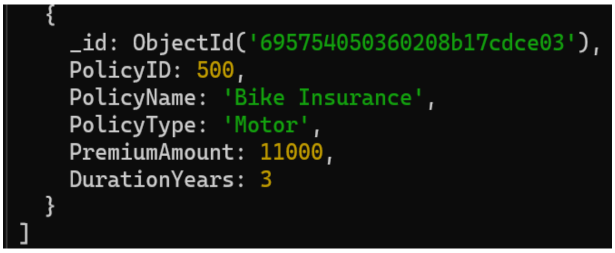


```
insDB> db.Policies.find({ PremiumAmount: { $gt: 10000 } })
[
  {
    _id: ObjectId('695753910360208b17cdce00'),
    PolicyID: 200,
    PolicyName: 'Smart term plan plus',
    PolicyType: 'Term Life',
    PremiumAmount: 30000,
    DurationYears: 8
  },
  {
    _id: ObjectId('695753bd0360208b17cdce01'),
    PolicyID: 300,
    PolicyName: 'ICICI Pru Signature',
    PolicyType: 'Unit Linked',
    PremiumAmount: 20000,
    DurationYears: 3
  },
  {
    _id: ObjectId('695753df0360208b17cdce02'),
    PolicyID: 400,
    PolicyName: 'Car Insurance',
    PolicyType: 'Motor',
    PremiumAmount: 27000,
    DurationYears: 5
  },
```

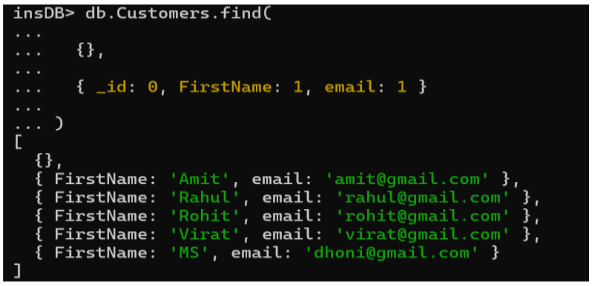
14. Projection (Display Selected Fields)

a) Display only FirstName and Email of customers

```
db.Customers.find(
  {},
  { _id: 0, FirstName: 1, email: 1 }
)
```



```
  {
    _id: ObjectId('695754050360208b17cdce03'),
    PolicyID: 500,
    PolicyName: 'Bike Insurance',
    PolicyType: 'Motor',
    PremiumAmount: 11000,
    DurationYears: 3
  }
]
```



```
insDB> db.Customers.find(
...
...   {},
...
...   { _id: 0, FirstName: 1, email: 1 }
...
... )
[
  {},
  { FirstName: 'Amit', email: 'amit@gmail.com' },
  { FirstName: 'Rahul', email: 'rahul@gmail.com' },
  { FirstName: 'Rohit', email: 'rohit@gmail.com' },
  { FirstName: 'Virat', email: 'virat@gmail.com' },
  { FirstName: 'MS', email: 'dhoni@gmail.com' }
]
```

15. Sorting Documents

a) Sort customers by DateOfBirth (ascending)

```
db.Customers.find().sort({ DateOfBirth: 1 })
```

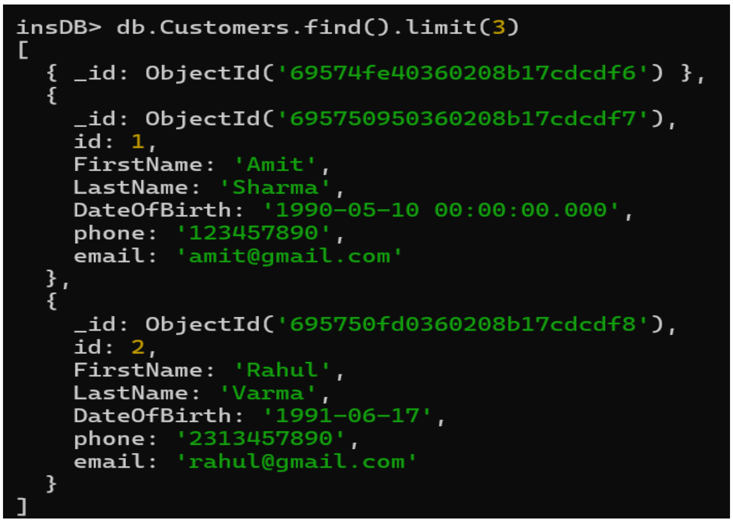
b) Sort policies by PremiumAmount (descending)

```
db.Policies.find().sort({ PremiumAmount: -1 })
```

16. Limit and Skip

a) Display first 3 customers

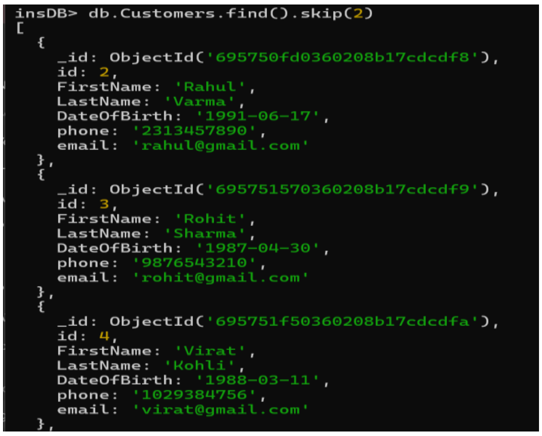
db.Customers.find().limit(3)



```
insDB> db.Customers.find().limit(3)
[
  { _id: ObjectId('69574fe40360208b17cdcdf6') },
  {
    _id: ObjectId('695750950360208b17cdcdf7'),
    id: 1,
    FirstName: 'Amit',
    LastName: 'Sharma',
    DateOfBirth: '1990-05-10 00:00:00.000',
    phone: '123457890',
    email: 'amit@gmail.com'
  },
  {
    _id: ObjectId('695750fd0360208b17cdcdf8'),
    id: 2,
    FirstName: 'Rahul',
    LastName: 'Varma',
    DateOfBirth: '1991-06-17',
    phone: '2313457890',
    email: 'rahul@gmail.com'
  }
]
```

a) Skip first 2 customers and display next records

db.Customers.find().skip(2)

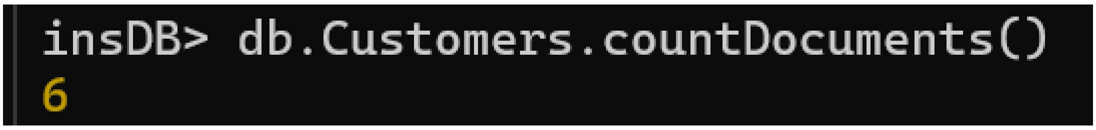


```
insDB> db.Customers.find().skip(2)
[
  {
    _id: ObjectId('695750fd0360208b17cdcdf8'),
    id: 2,
    FirstName: 'Rahul',
    LastName: 'Varma',
    DateOfBirth: '1991-06-17',
    phone: '2313457890',
    email: 'rahul@gmail.com'
  },
  {
    _id: ObjectId('695751570360208b17cdcdf9'),
    id: 3,
    FirstName: 'Rohit',
    LastName: 'Sharma',
    DateOfBirth: '1987-04-30',
    phone: '9876543210',
    email: 'rohit@gmail.com'
  },
  {
    _id: ObjectId('695751f50360208b17cdcdfa'),
    id: 4,
    FirstName: 'Virat',
    LastName: 'Kohli',
    DateOfBirth: '1988-03-11',
    phone: '1029384756',
    email: 'virat@gmail.com'
  },
```

17. Count Documents

a) Count total customers

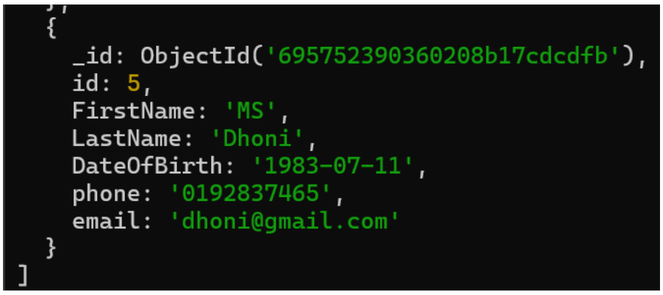
db.Customers.countDocuments()



```
insDB> db.Customers.countDocuments()
6
```

b) Count number of Motor insurance policies

db.Policies.countDocuments({ PolicyType: "Motor" })

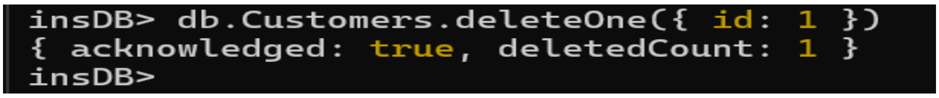


```
  },
  {
    _id: ObjectId('695752390360208b17cdcdfb'),
    id: 5,
    FirstName: 'MS',
    LastName: 'Dhoni',
    DateOfBirth: '1983-07-11',
    phone: '0192837465',
    email: 'dhoni@gmail.com'
  }
]
```

18. Delete Operations

a) Delete a customer with id = 1

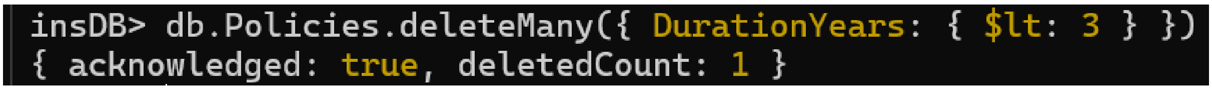
db.Customers.deleteOne({ id: 1 })



```
insDB> db.Customers.deleteOne({ id: 1 })
{ acknowledged: true, deletedCount: 1 }
insDB>
```

b) Delete all policies with duration less than 3 years

db.Policies.deleteMany({ DurationYears: { $lt: 3 } })

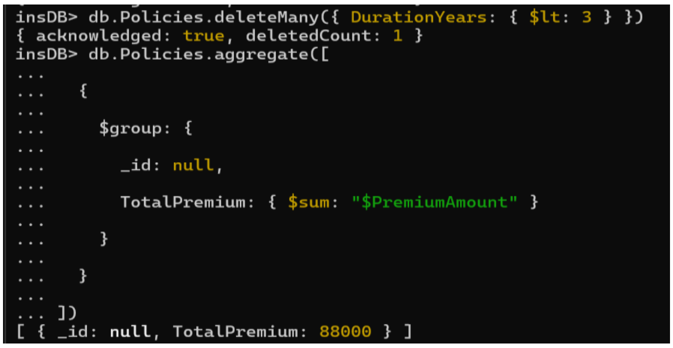


```
insDB> db.Policies.deleteMany({ DurationYears: { $lt: 3 } })
{ acknowledged: true, deletedCount: 1 }
```

19. Aggregate Operations

a) Find total premium amount of all policies

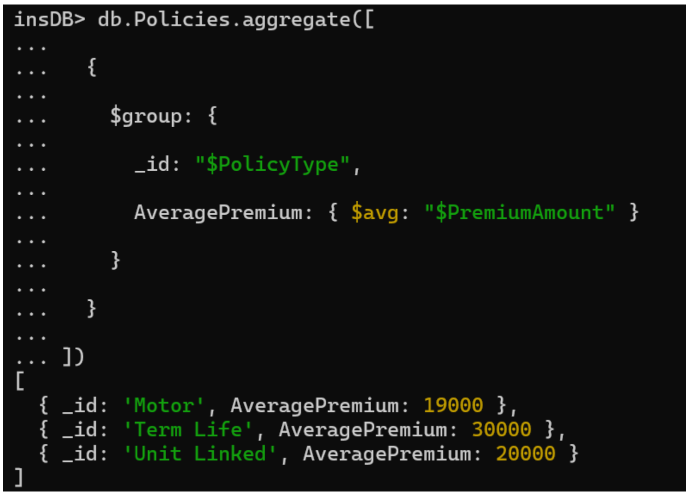
```
db.Policies.aggregate([
 {
  $group: {
   _id: null,
   TotalPremium: { $sum: "$PremiumAmount" }
  }
 }
])
```



```
insDB> db.Policies.deleteMany({ DurationYears: { $lt: 3 } })
{ acknowledged: true, deletedCount: 1 }
insDB> db.Policies.aggregate([
...
...   {
...
...     $group: {
...
...       _id: null,
...
...       TotalPremium: { $sum: "$PremiumAmount" }
...
...     }
...
...   }
...
... ])
[ { _id: null, TotalPremium: 88000 } ]
```

b) Find average premium amount by policy type

```
db.Policies.aggregate([

  {

    $group: {

      _id: "$PolicyType",

      AveragePremium: { $avg: "$PremiumAmount" }

    }

  }

])
```



```
insDB> db.Policies.aggregate([
...
...   {
...
...     $group: {
...
...       _id: "$PolicyType",
...
...       AveragePremium: { $avg: "$PremiumAmount" }
...
...     }
...
...   }
...
... ])
[
  { _id: 'Motor', AveragePremium: 19000 },
  { _id: 'Term Life', AveragePremium: 30000 },
  { _id: 'Unit Linked', AveragePremium: 20000 }
]
```