Create Database:

test> use insuranceDB;

Create Collections:

insuranceDB> db.createCollection('Customers');

{ ok: 1 }

insuranceDB> db.createCollection('Policies');

{ ok: 1 }

insuranceDB> db.createCollection('PolicyAssignments');

{ ok: 1 }

insuranceDB> db.createCollection('Agents');

{ ok: 1 }

insuranceDB> db.createCollection('Claims');

{ ok: 1 }

insuranceDB> db.Customers.createIndex({ email: 1 }, { unique: true });

email_1

Insertion:

db.customers.insertMany([

 { customerID: 1, firstName: "poojitha", lastName: "bejgum", dateOfBirth: ISODate("2004-07-13"), phone: "9347758510", email: "poojitha@gmail.com" },

 { customerID: 2, firstName: "ruchitha", lastName: "bejgum", dateOfBirth: ISODate("2001-12-11"), phone: "9390423662", email: "ruchitha@gmail.com" },

 { customerID: 3, firstName: "Aditi", lastName: "Sharma", dateOfBirth: ISODate("1998-03-21"), phone: "8123456789", email: "aditi@gmail.com" },

 { customerID: 4, firstName: "Meera", lastName: "Rao", dateOfBirth: ISODate("1985-11-30"), phone: "9988776655", email: "meera@gmail.com" }

]);

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6957577ccac4658d402710bc'),
    '1': ObjectId('6957577ccac4658d402710bd'),
    '2': ObjectId('6957577ccac4658d402710be'),
    '3': ObjectId('6957577ccac4658d402710bf'),
    '4': ObjectId('6957577ccac4658d402710c0')
  }
}
```

```
db.policies.insertMany([

 { policyID: 101, policyName: "HealthPlus", policyType: "Health", premiumAmount:
8000, durationYears: 50 },

 { policyID: 102, policyName: "Care Plus", policyType: "Health", premiumAmount:4000,
durationYears: 50 },

 { policyID: 103, policyName: "Drive Safe Plus", policyType: "Vehicle",
premiumAmount:8000, durationYears: 40 },

 { policyID: 104, policyName: "Safe Nest", policyType: "Property",
premiumAmount:1000, durationYears: 100 },

 { policyID: 105, policyName: "LifeSecure", policyType: "Life",
premiumAmount:1000,durationYears: 40 }

]);
```

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('695774b5cac4658d402710de'),
    '1': ObjectId('695774b5cac4658d402710df'),
    '2': ObjectId('695774b5cac4658d402710e0'),
    '3': ObjectId('695774b5cac4658d402710e1'),
    '4': ObjectId('695774b5cac4658d402710e2')
  }
}
```

```
db.PolicyAssignments.insertMany([

 { assignmentID: 1, customerID: 1, policyID: 101, agentID: 1, startDate: ISODate("2025-12-28"), endDate: ISODate("2075-12-28") },

 { assignmentID: 2, customerID: 2, policyID: 102, agentID: 2, startDate: ISODate("2025-12-28"), endDate: ISODate("2075-12-28") },

 { assignmentID: 3, customerID: 3, policyID: 105, agentID: 3, startDate: ISODate("2025-01-10"), endDate: ISODate("2030-01-10") },

 { assignmentID: 4, customerID: 4, policyID: 104, agentID: 4, startDate: ISODate("2024-05-20"), endDate: ISODate("2049-05-20") },

 { assignmentID: 5, customerID: 4, policyID: 101, agentID: 1, startDate: ISODate("2023-08-15"), endDate: ISODate("2043-08-15") }

]);
```

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('695773e8cac4658d402710d9'),
    '1': ObjectId('695773e8cac4658d402710da'),
    '2': ObjectId('695773e8cac4658d402710db'),
    '3': ObjectId('695773e8cac4658d402710dc'),
    '4': ObjectId('695773e8cac4658d402710dd')
  }
}
```

```
agents.insertMany([

 { agentID: 1, agentName: "Rohith", phone: "6309874810", city: "Hyderabad" },

 { agentID: 2, agentName: "Swetha", phone: "9489928999", city: "Nizamabad" },

 { agentID: 3, agentName: "Arjun", phone: "8899001122", city: "Hyderabad" },

 { agentID: 4, agentName: "Farah", phone: "9900112233", city: "Secunderabad" }

]);
```

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('69576003cac4658d402710c6'),
    '1': ObjectId('69576003cac4658d402710c7'),
    '2': ObjectId('69576003cac4658d402710c8'),
    '3': ObjectId('69576003cac4658d402710c9')
  }
}
```

```
db.claims.insertMany([

 { claimID: 1, assignmentID: 1, claimDate: ISODate("2026-02-02"), claimAmount:
50000, claimStatus: "PENDING", claimState: "Active" },

 { claimID: 2, assignmentID: 2, claimDate: ISODate("2026-02-12"), claimAmount:
100000, claimStatus: "PENDING", claimState: "Active" },

 { claimID: 3, assignmentID: 3, claimDate: ISODate("2025-03-01"), claimAmount:
12000, claimStatus: "PENDING" },

 { claimID: 4, assignmentID: 4, claimDate: ISODate("2025-06-18"), claimAmount:
55000, claimStatus: "APPROVED" },

 { claimID: 5, assignmentID: 5, claimDate: ISODate("2025-09-30"), claimAmount:
25000, claimStatus: "REJECTED" }

]);
```

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6957601ccac4658d402710cf'),
    '1': ObjectId('6957601ccac4658d402710d0'),
    '2': ObjectId('6957601ccac4658d402710d1'),
    '3': ObjectId('6957601ccac4658d402710d2'),
    '4': ObjectId('6957601ccac4658d402710d3')
  }
}
```

Querying:

1.View all records of PolicyAssignment table with CustomerId, PolicyId, AgentID columns only

insuranceDB>
db.PolicyAssignments.find({},{customerID:1,policyID:1,agentID:1,_id:0});

```
[
  { customerID: 1, policyID: 101, agentID: 1 },
  { customerID: 2, policyID: 102, agentID: 2 },
  { customerID: 3, policyID: 105, agentID: 3 },
  { customerID: 4, policyID: 104, agentID: 4 },
  { customerID: 4, policyID: 101, agentID: 1 }
]
```

2.Display all policies of Health type.

insuranceDB> db.Policies.find({ policyType: "Health" }).pretty();

```
[
  {
    _id: ObjectId('695785cacac4658d402710e8'),
    policyID: 101,
    policyName: 'HealthPlus',
    policyType: 'Health',
    premiumAmount: 8000,
    durationYears: 50
  },
  {
    _id: ObjectId('695785cacac4658d402710e9'),
    policyID: 102,
    policyName: 'Care Plus',
    policyType: 'Health',
    premiumAmount: 4000,
    durationYears: 50
  }
]
```

3.Display unique city names from where agents belong to.

db.Agents.distinct("city");

```
[ 'Hyderabad', 'Nizamabad', 'Secunderabad' ]
```

4.List policies of type Life, Health, Motor use OR clause.

db.Policies.find({ $or: [{ policyType: "Life" }, { policyType: "Health" }, { policyType: "Motor" }] }, { _id: 0 })

```
[
  {
    policyID: 101,
    policyName: 'HealthPlus',
    policyType: 'Health',
    premiumAmount: 8000,
    durationYears: 50
  },
  {
    policyID: 102,
    policyName: 'Care Plus',
    policyType: 'Health',
    premiumAmount: 4000,
    durationYears: 50
  },
  {
    policyID: 105,
    policyName: 'LifeSecure',
    policyType: 'Life',
    premiumAmount: 1000,
    durationYears: 40
  }
]
```

5.Premium greater than 1000

insuranceDB> db.Policies.find({ premiumAmount: { $gt: 1000 } })

```
[
  {
    _id: ObjectId('695785cacac4658d402710e8'),
    policyID: 101,
    policyName: 'HealthPlus',
    policyType: 'Health',
    premiumAmount: 8000,
    durationYears: 50
  },
  {
    _id: ObjectId('695785cacac4658d402710e9'),
    policyID: 102,
    policyName: 'Care Plus',
    policyType: 'Health',
    premiumAmount: 4000,
    durationYears: 50
  },
  {
    _id: ObjectId('695785cacac4658d402710ea'),
    policyID: 103,
    policyName: 'Drive Safe Plus',
    policyType: 'Vehicle',
    premiumAmount: 8000,
    durationYears: 40
  }
]
```

6.Update premium of policyNo 101

db.Policies.updateOne({policyID:101},{$set:{premiumAmount:1000}})

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

7.Display claims data where claim status is Rejected.

db.Claims.find({claimStatus:'REJECTED'})

```
[
  {
    _id: ObjectId('6957601ccac4658d402710d3'),
    claimID: 5,
    assignmentID: 5,
    claimDate: ISODate('2025-09-30T00:00:00.000Z'),
    claimAmount: 25000,
    claimStatus: 'REJECTED'
  }
]
```

8.Delete policy with policyNo 103

insuranceDB> db.Policies.deleteOne({ policyID: 103 })

```
{ acknowledged: true, deletedCount: 1 }
```

9.Premium between 1000 and 8000

db.Policies.find({ premiumAmount: { $gt: 1000, $lt: 8000 } })

```
[
  {
    _id: ObjectId('695785cacac4658d402710e9'),
    policyID: 102,
    policyName: 'Care Plus',
    policyType: 'Health',
    premiumAmount: 4000,
    durationYears: 50
  }
]
```

10. Top 2 highest premium policies

```
[
  {
    _id: ObjectId('695785cacac4658d402710e9'),
    policyID: 102,
    policyName: 'Care Plus',
    policyType: 'Health',
    premiumAmount: 4000,
    durationYears: 50
  },
  {
    _id: ObjectId('695785cacac4658d402710e8'),
    policyID: 101,
    policyName: 'HealthPlus',
    policyType: 'Health',
    premiumAmount: 1000,
    durationYears: 50
  }
]
```

11. Policies sorted by premium (descending)

```
[
  {
    _id: ObjectId('695785cacac4658d402710e9'),
    policyID: 102,
    policyName: 'Care Plus',
    policyType: 'Health',
    premiumAmount: 4000,
    durationYears: 50
  },
  {
    _id: ObjectId('695785cacac4658d402710e8'),
    policyID: 101,
    policyName: 'HealthPlus',
    policyType: 'Health',
    premiumAmount: 1000,
    durationYears: 50
  },
  {
    _id: ObjectId('695785cacac4658d402710eb'),
    policyID: 104,
    policyName: 'Safe Nest',
    policyType: 'Property',
    premiumAmount: 1000,
    durationYears: 100
  },
  {
    _id: ObjectId('695785cacac4658d402710ec'),
    policyID: 105,
    policyName: 'LifeSecure',
    policyType: 'Life',
    premiumAmount: 1000,
    durationYears: 40
  }
]
```

12. Skip first 2 records from Customers.

```
[
  {
    _id: ObjectId('6957577ccac4658d402710be'),
    customerID: 3,
    firstName: 'Aditi',
    lastName: 'Sharma',
    dateOfBirth: ISODate('1998-03-21T00:00:00.000Z'),
    phone: '8123456789',
    email: 'aditi@gmail.com'
  },
  {
    _id: ObjectId('6957577ccac4658d402710bf'),
    customerID: 4,
    firstName: 'Meera',
    lastName: 'Rao',
    dateOfBirth: ISODate('1985-11-30T00:00:00.000Z'),
    phone: '9988776655',
    email: 'meera@gmail.com'
  },
  {
    _id: ObjectId('6957577ccac4658d402710c0'),
    customerID: 8,
    firstName: 'Kabir',
    lastName: 'Khan',
    phone: '7766554433',
    email: 'kab@gmail.com'
  }
]
```

13.Policies where premium is NOT between 10k–20k

insuranceDB> db.Policies.find({ premiumAmount:{$not: { $gt: 1000, $lt: 8000}}})

```
[
  {
    _id: ObjectId('695785cacac4658d402710e8'),
    policyID: 101,
    policyName: 'HealthPlus',
    policyType: 'Health',
    premiumAmount: 1000,
    durationYears: 50
  },
  {
    _id: ObjectId('695785cacac4658d402710eb'),
    policyID: 104,
    policyName: 'Safe Nest',
    policyType: 'Property',
    premiumAmount: 1000,
    durationYears: 100
  },
  {
    _id: ObjectId('695785cacac4658d402710ec'),
    policyID: 105,
    policyName: 'LifeSecure',
    policyType: 'Life',
    premiumAmount: 1000,
    durationYears: 40
  }
]
```

14. Show policyType in uppercase

db.Policies.aggregate([{$project: {policyType: {$toUpper:"$policyType" }, _id: 0}}]);

```
[
  { policyType: 'HEALTH' },
  { policyType: 'HEALTH' },
  { policyType: 'PROPERTY' },
  { policyType: 'LIFE' }
]
```

15. Add GST (18%) to premium

db.Policies.aggregate([{$project:{customerName: 1, totalPremium: {$multiply: ["$premiumAmount", 1.18]}}}])

```
[
  { _id: ObjectId('695785cacac4658d402710e8'), totalPremium: 1180 },
  { _id: ObjectId('695785cacac4658d402710e9'), totalPremium: 4720 },
  { _id: ObjectId('695785cacac4658d402710eb'), totalPremium: 1180 },
  { _id: ObjectId('695785cacac4658d402710ec'), totalPremium: 1180 }
]
```

16. Increase all Health premiums by 10%

db.Policies.updateMany({policyType:'Health'},{$mul:{premiumAmount:1.1}})

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 2,
  modifiedCount: 2,
  upsertedCount: 0
}
```

17. Find a customer who have policies

db.Customers.aggregate([{ $lookup: { from: "Policies", localField: "customerId", foreignField: "customerId", as: "policyDetails" } }, { $unwind: "$policyDetails"} ,{$limit:1}])

```
[
  {
    _id: ObjectId('6957577ccac4658d402710bc'),
    customerID: 1,
    firstName: 'poojitha',
    lastName: 'bejgum',
    dateOfBirth: ISODate('2004-07-13T00:00:00.000Z'),
    phone: '9347758510',
    email: 'poojitha@gmail.com',
    policyDetails: {
      _id: ObjectId('695773e8cac4658d402710d9'),
      assignmentID: 1,
      customerID: 1,
      policyID: 101,
      agentID: 1,
      startDate: ISODate('2025-12-28T00:00:00.000Z'),
      endDate: ISODate('2075-12-28T00:00:00.000Z')
    }
  }
]
```

18. Delete policies with premium > 25000

db.Policies.deleteMany({ premiumAmount: { $gt: 4000 } })

```
{ acknowledged: true, deletedCount: 1 }
```

19. Add lastUpdated field to all documents

insuranceDB> db.Policies.updateMany({}, { $set: { lastUpdated: new Date() } })

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 3,
  modifiedCount: 3,
  upsertedCount: 0
}
```

20. Increase all Health premiums by 10%

insuranceDB> db.Policies.updateMany({ policyType: "Health" }, { $mul: { premiumAmount: 1.1 } })

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

21. Insert one customer

db.Customers.insertOne({customerID:7,firstName:"riya",email:"riya@mail.com",policies:[105]});

```
{
  acknowledged: true,
  insertedId: ObjectId('6957b877cac4658d402710ed')
}
```

22. Update PENDING claims to APPROVED

db.Claims.updateMany({claimStatus:"PENDING"},{$set:{claimStatus:"APPROVED"}});

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 3,
  modifiedCount: 3,
  upsertedCount: 0
}
```

23. Update one policy name

db.Policies.updateOne({policyID:104},{$set:{policyName:"secure drive"}});

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

24. Update many agents by OR condition

db.Agents.find({ $or:[{ city:"Hyderabad" },{ city:"Secunderabad" }] }, { _id:0 }).pretty()

```
[
  {
    agentID: 1,
    agentName: 'Rohith',
    phone: '6309874810',
    city: 'Hyderabad'
  },
  {
    agentID: 3,
    agentName: 'Arjun',
    phone: '8899001122',
    city: 'Hyderabad'
  },
  {
    agentID: 4,
    agentName: 'Farah',
    phone: '9900112233',
    city: 'Secunderabad'
  }
]
```

25. increment premiumAmount for policy 101

db.Policies.updateOne({policyID:101},{$inc:{premiumAmount:5000}});

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```