Date: 29-12-2025 à Day 2 – Querying & Modifying Data

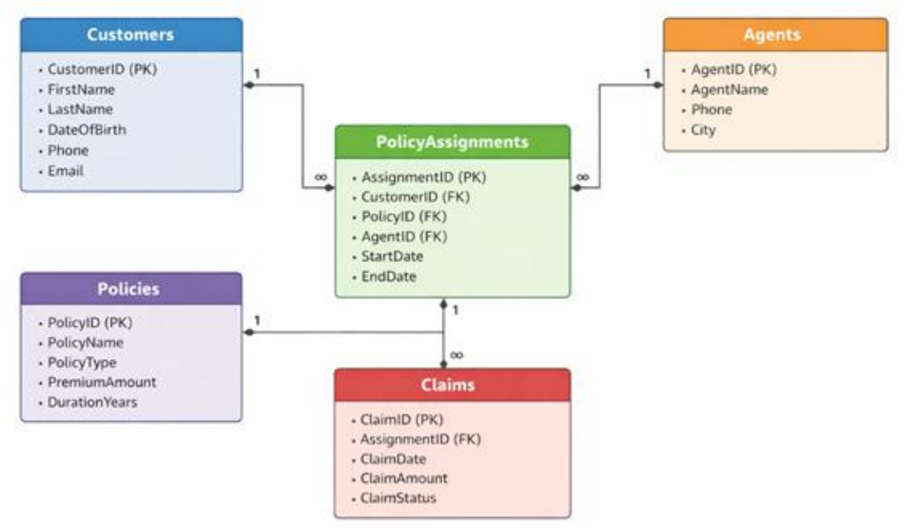
Create Sample Insurance database named **InsuranceDB** with tables, constraints, and initial sample data. Based on following er diagram and descriptions.

Entities we'll model:

- **Customers** – people who buy insurance
- **Policies** – insurance products (Health, Motor, Life)
- **Agents** – insurance agents
- **PolicyAssignments** – which customer bought which policy
- **Claims** – claims raised against policies

Relationship Explanation

- **One Customer → Many PolicyAssignments**
- **One Policy → Many PolicyAssignments**
- **One Agent → Many PolicyAssignments**
- **One PolicyAssignment → Many Claims**



Submit following:

1. Create Database command.
   Query: create database insuranceDB;

2. Create table commands for all the tables with constraints, relationships etc.
   Query:

```sql
CREATE TABLE Customers (
   CustomerID   INT IDENTITY(1,1) PRIMARY KEY,
   FirstName    VARCHAR(50) NOT NULL,
   LastName     VARCHAR(50) NOT NULL,
   DateOfBirth  DATE NULL,
   Phone        VARCHAR(20) NULL,
   Email        VARCHAR(100) NULL
);


CREATE TABLE Policies (
   PolicyID       INT IDENTITY(1,1) PRIMARY KEY,
   PolicyName     VARCHAR(100) NOT NULL,
   PolicyType     VARCHAR(50) NOT NULL,
   PremiumAmount  DECIMAL(10,2) NOT NULL,
   DurationYears  INT NOT NULL
);


CREATE TABLE Agents (
   AgentID    INT IDENTITY(1,1) PRIMARY KEY,
   AgentName  VARCHAR(100) NOT NULL,
   Phone      VARCHAR(20) NULL,
   City       VARCHAR(50) NULL
);

CREATE TABLE PolicyAssignments (
   AssignmentID INT IDENTITY(1,1) PRIMARY KEY,
   CustomerID   INT NOT NULL,
   PolicyID     INT NOT NULL,
   AgentID      INT NOT NULL,
   StartDate    DATE NOT NULL,
   EndDate      DATE NULL,

   CONSTRAINT FK_PolicyAssignments_Customers
      FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),

   CONSTRAINT FK_PolicyAssignments_Policies
      FOREIGN KEY (PolicyID) REFERENCES Policies(PolicyID),

   CONSTRAINT FK_PolicyAssignments_Agents
      FOREIGN KEY (AgentID) REFERENCES Agents(AgentID)
);
```

```
CREATE TABLE Claims (
   ClaimID      INT IDENTITY(1,1) PRIMARY KEY,
   AssignmentID INT NOT NULL,
   ClaimDate    DATE NOT NULL,
   ClaimAmount  DECIMAL(10,2) NOT NULL,
   ClaimStatus  VARCHAR(50) NOT NULL,

   CONSTRAINT FK_Claims_PolicyAssignments
      FOREIGN KEY (AssignmentID) REFERENCES PolicyAssignments(AssignmentID)
);
```

3. Insert commands for all tables.

```
INSERT INTO Customers (FirstName, LastName, DateOfBirth, Phone, Email) VALUES
('Koushik',     'Reddy',     '2004-03-12', '9876500011', 'koushik.reddy@gmail.com'),
('Rithvik',     'Sharma',    '2004-07-25', '9876500012', 'rithvik.sharma@gmail.com'),
('Ganesh',      'Kumar',     '2003-11-14', '9876500013', 'ganesh.kumar@gmail.com'),
('Mani',        'Teja',      '2004-01-09', '9876500014', 'mani.teja@gmail.com'),
('Spoorthy',    'Priya',     '2004-05-18', '9876500015', 'spoorthy.priya@gmail.com'),
('Sri',         'Nayani',    '2004-09-02', '9876500016', 'sri.nayani@gmail.com'),
('Nanda         Kishor','Gudala',                     '2004-08-20',        '8688580131',
'nandakishorgudala@gmail.com'),
('Varshith',    'Rao',       '2004-12-07', '9876500017', 'varshith.rao@gmail.com');
```

```
INSERT  INTO  Policies  (PolicyName,  PolicyType,  PremiumAmount,  DurationYears)
VALUES
('Health Secure Basic',      'Health',  8000.00,  1),
('Health Secure Plus',       'Health', 15000.00,  1),
('Health Family Floater',    'Health', 22000.00,  2),

('Life Shield Term',         'Life',   12000.00,  1),
('Life Wealth Plan',         'Life',   30000.00,  5),

('Motor Protect TwoWheeler', 'Motor',   6000.00,  1),
```

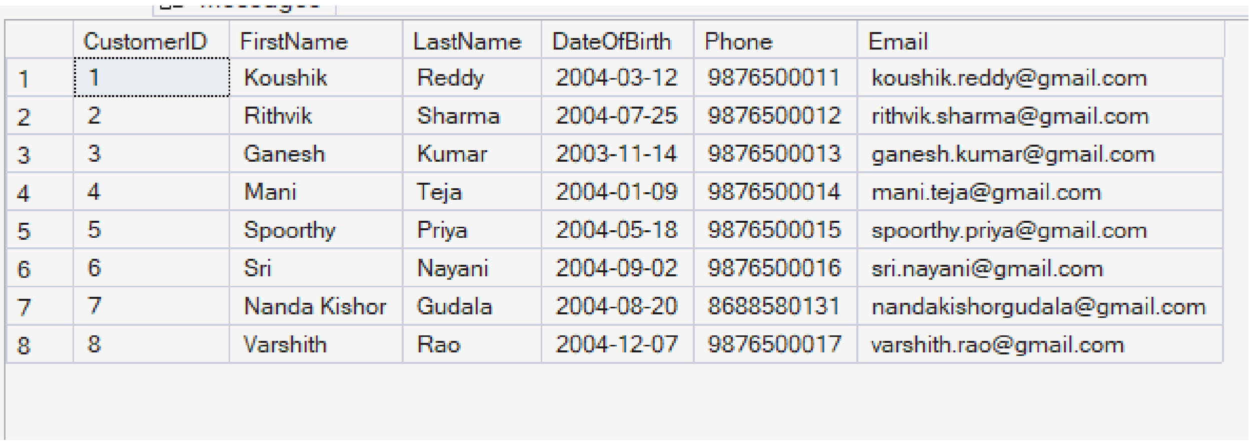
```sql
('Motor Protect Car',        'Motor',  14000.00,  1),
('Motor Comprehensive',      'Motor',  25000.00,  3),

('Travel Safe',              'Travel',  5000.00,  1),
('Home Protect',             'Home',   18000.00,  2);


INSERT INTO Agents (AgentName, Phone, City) VALUES
('Ramesh',      '9876500101', 'Hyderabad'),
('Bhoopendhar', '9876500102', 'Warangal'),
('Raju',        '9876500103', 'Karimnagar'),
('Raghu',       '9876500104', 'Vijayawada'),
('Ravi',        '9876500105', 'Vizag'),
('Rajeev',      '9876500106', 'Bangalore'),
('Rahul',       '9876500107', 'Chennai'),
('Rakesh',      '9876500108', 'Mumbai');



INSERT INTO PolicyAssignments (CustomerID, PolicyID, AgentID, StartDate, EndDate)
VALUES
(1, 1, 1, '2024-01-10', '2025-01-09'),
(2, 2, 2, '2024-03-15', '2025-03-14'),
(3, 4, 3, '2023-07-01', '2024-06-30'),
(4, 6, 4, '2024-05-20', '2025-05-19'),
(5, 3, 5, '2024-09-05', NULL),
(6, 5, 6, '2022-12-12', '2027-12-11'),
(7, 2, 7, '2024-02-01', '2025-01-31'),
(8, 7, 8, '2024-08-18', '2025-08-17'),
(1, 8, 2, '2023-11-25', '2026-11-24'),
(4,1,1,'2024-12-01','2025-11-30');


INSERT INTO Claims (AssignmentID, ClaimDate, ClaimAmount, ClaimStatus) VALUES
(3,  '2023-08-21', 18500.00, 'Approved'),
(1,  '2024-04-14',  3200.00, 'Rejected'),
(6,  '2024-09-03', 74250.00, 'Approved'),
(2,  '2024-12-11',  9900.00, 'Pending'),
(5,  '2024-10-26', 26800.00, 'Approved'),
(4,  '2024-06-07',  4100.00, 'Approved'),
(7,  '2024-02-19', 12500.00, 'Rejected'),
(2,  '2024-07-29', 15600.00, 'Approved'),
(8,  '2024-11-05',  5400.00, 'Rejected'),
(1,  '2024-09-18',  8700.00, 'Approved'),
(6,  '2023-12-22', 30000.00, 'Rejected'),
(5,  '2025-01-12', 19500.00, 'Pending'),
(4,  '2024-12-28', 10200.00, 'Rejected'),
(3,  '2024-03-09',  6600.00, 'Pending');
```

4. Select commands

1. View all records Customers table.

   Query:

   ```
   select * from customers;
   ```



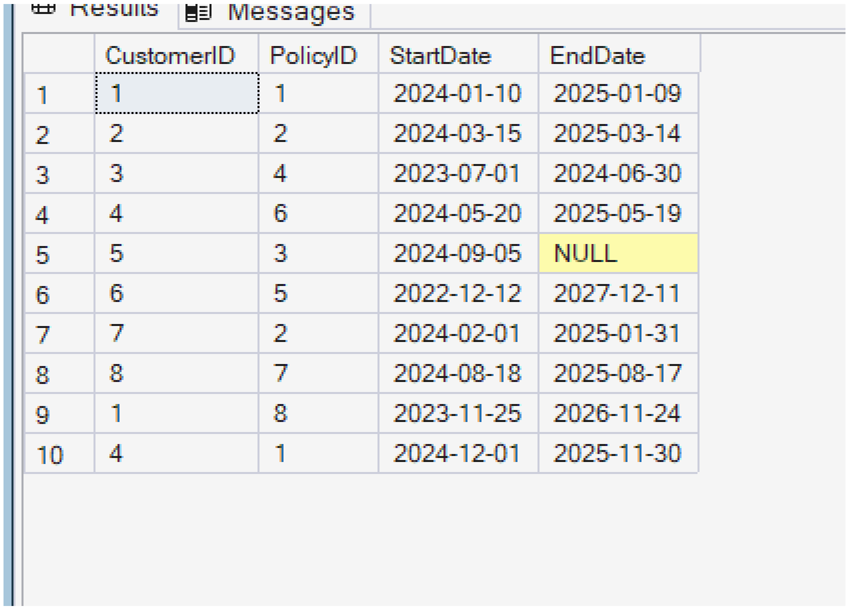
	CustomerID	FirstName	LastName	DateOfBirth	Phone	Email
1	1	Koushik	Reddy	2004-03-12	9876500011	koushik.reddy@gmail.com
2	2	Rithvik	Sharma	2004-07-25	9876500012	rithvik.sharma@gmail.com
3	3	Ganesh	Kumar	2003-11-14	9876500013	ganesh.kumar@gmail.com
4	4	Mani	Teja	2004-01-09	9876500014	mani.teja@gmail.com
5	5	Spoorthy	Priya	2004-05-18	9876500015	spoorthy.priya@gmail.com
6	6	Sri	Nayani	2004-09-02	9876500016	sri.nayani@gmail.com
7	7	Nanda Kishor	Gudala	2004-08-20	8688580131	nandakishorgudala@gmail.com
8	8	Varshith	Rao	2004-12-07	9876500017	varshith.rao@gmail.com

2. View all records of PolicyAssignment table with CustomerId, PolicyId, StartDate and EndDate columns only.

   Query:

   ```
   SELECT CustomerID, PolicyID, StartDate, EndDate
   FROM PolicyAssignments;
   ```

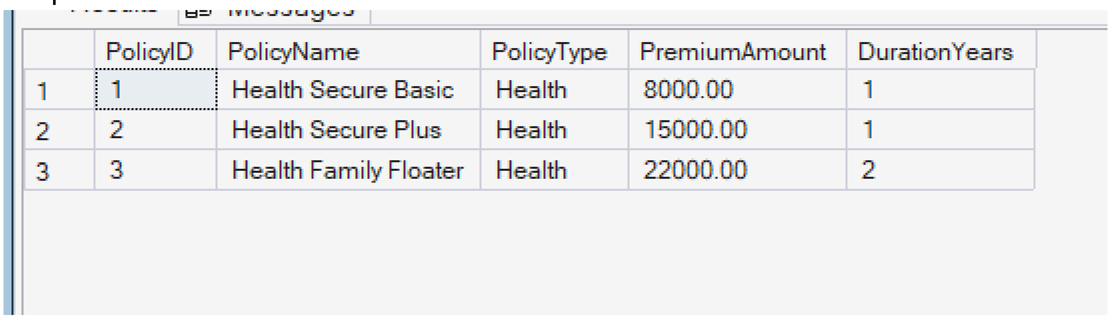
   Output:



	CustomerID	PolicyID	StartDate	EndDate
1	1	1	2024-01-10	2025-01-09
2	2	2	2024-03-15	2025-03-14
3	3	4	2023-07-01	2024-06-30
4	4	6	2024-05-20	2025-05-19
5	5	3	2024-09-05	NULL
6	6	5	2022-12-12	2027-12-11
7	7	2	2024-02-01	2025-01-31
8	8	7	2024-08-18	2025-08-17
9	1	8	2023-11-25	2026-11-24
10	4	1	2024-12-01	2025-11-30

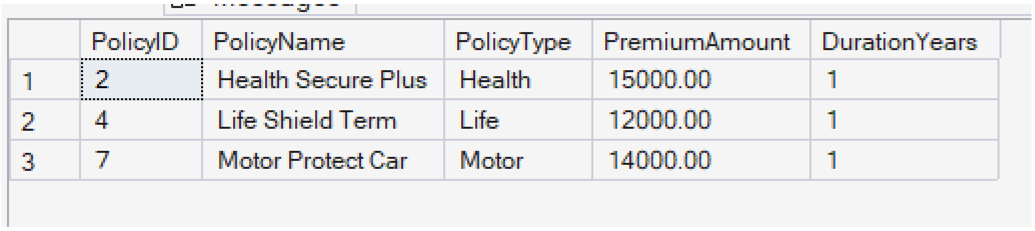
3. Display all policies of Health type.
   Query:
   SELECT *
   FROM Policies
   WHERE PolicyType = 'Health';

   Output:



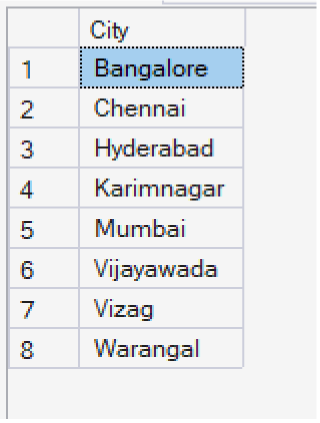
	PolicyID	PolicyName	PolicyType	PremiumAmount	DurationYears
1	1	Health Secure Basic	Health	8000.00	1
2	2	Health Secure Plus	Health	15000.00	1
3	3	Health Family Floater	Health	22000.00	2

4. Display policies having premium amount more than 10000 and DurationYears is 1.
   Query:
   SELECT * FROM Policies WHERE PremiumAmount > 10000 AND DurationYears = 1;
   output:



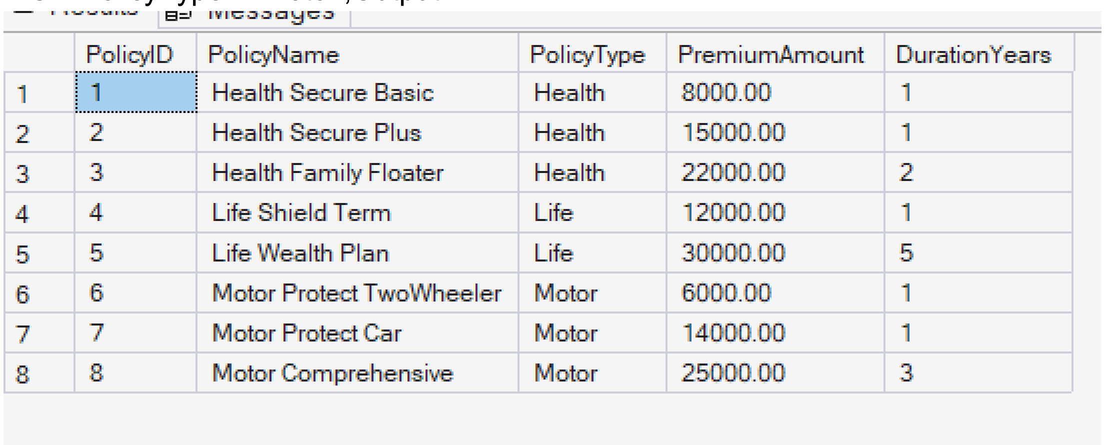
	PolicyID	PolicyName	PolicyType	PremiumAmount	DurationYears
1	2	Health Secure Plus	Health	15000.00	1
2	4	Life Shield Term	Life	12000.00	1
3	7	Motor Protect Car	Motor	14000.00	1

5. Display unique city names from where agents belong to.
   Query:
   SELECT DISTINCT City FROM Agents;



	City
1	Bangalore
2	Chennai
3	Hyderabad
4	Karimnagar
5	Mumbai
6	Vijayawada
7	Vizag
8	Warangal

6. List policies of type Life, Health, Motor use OR clause.
   Query:
   SELECT *
   FROM Policies
   WHERE PolicyType = 'Life'
   OR PolicyType = 'Health'
   OR PolicyType = 'Motor';Output:

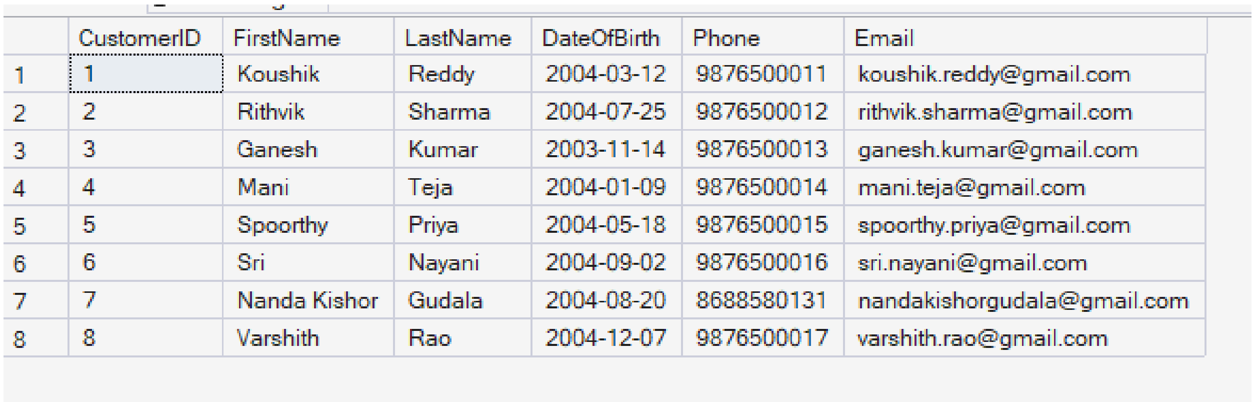


	PolicyID	PolicyName	PolicyType	PremiumAmount	DurationYears
1	1	Health Secure Basic	Health	8000.00	1
2	2	Health Secure Plus	Health	15000.00	1
3	3	Health Family Floater	Health	22000.00	2
4	4	Life Shield Term	Life	12000.00	1
5	5	Life Wealth Plan	Life	30000.00	5
6	6	Motor Protect TwoWheeler	Motor	6000.00	1
7	7	Motor Protect Car	Motor	14000.00	1
8	8	Motor Comprehensive	Motor	25000.00	3

7. List policies of type Life, Health, Motor use IN operator.
   Query:
   Query:
   SELECT *
   FROM Policies
   WHERE PolicyType in ('Life','Health','Motor');

8. Display list of customers born after January 1st, 2001 and before December 31st, 2020 using >= and <= operators.
   Query:
   SELECT *
   FROM Customers
   WHERE DateOfBirth >= '2001-01-01'
     AND DateOfBirth <= '2020-12-31';

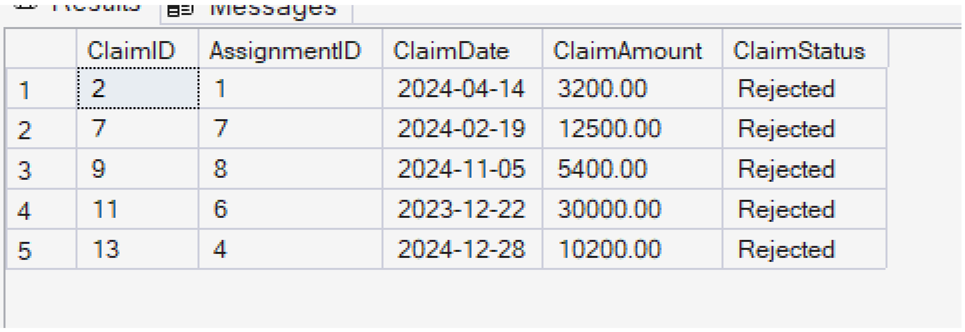
   Output:



	CustomerID	FirstName	LastName	DateOfBirth	Phone	Email
1	1	Koushik	Reddy	2004-03-12	9876500011	koushik.reddy@gmail.com
2	2	Rithvik	Sharma	2004-07-25	9876500012	rithvik.sharma@gmail.com
3	3	Ganesh	Kumar	2003-11-14	9876500013	ganesh.kumar@gmail.com
4	4	Mani	Teja	2004-01-09	9876500014	mani.teja@gmail.com
5	5	Spoorthy	Priya	2004-05-18	9876500015	spoorthy.priya@gmail.com
6	6	Sri	Nayani	2004-09-02	9876500016	sri.nayani@gmail.com
7	7	Nanda Kishor	Gudala	2004-08-20	8688580131	nandakishorgudala@gmail.com
8	8	Varshith	Rao	2004-12-07	9876500017	varshith.rao@gmail.com

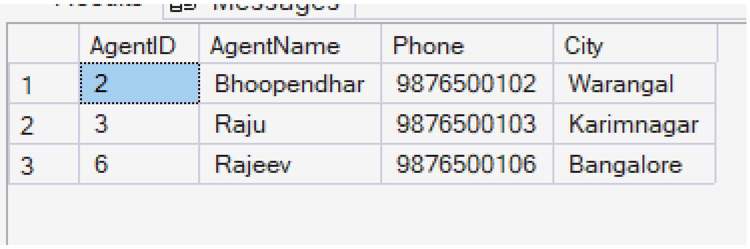
9. Display list of customers born after January 1st, 2001 and before December 31st, 2020 using between operator.
   Query:
   SELECT *
   FROM Customers
   WHERE DateOfBirth BETWEEN '2001-01-01' AND '2020-12-31';

10. Display claims data where claim status is Rejected.
   Query:
   SELECT *
   FROM Claims
   WHERE ClaimStatus = 'Rejected';
   Output:



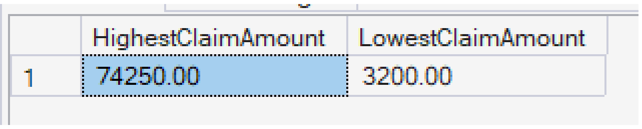
	ClaimID	AssignmentID	ClaimDate	ClaimAmount	ClaimStatus
1	2	1	2024-04-14	3200.00	Rejected
2	7	7	2024-02-19	12500.00	Rejected
3	9	8	2024-11-05	5400.00	Rejected
4	11	6	2023-12-22	30000.00	Rejected
5	13	4	2024-12-28	10200.00	Rejected

11. Display records of Agents who stay in a city whose second letter is 'a'.
    Query:
    SELECT *
    FROM Agents
    WHERE City LIKE '_a%';
    Output:



	AgentID	AgentName	Phone	City
1	2	Bhoopendhar	9876500102	Warangal
2	3	Raju	9876500103	Karimnagar
3	6	Rajeev	9876500106	Bangalore

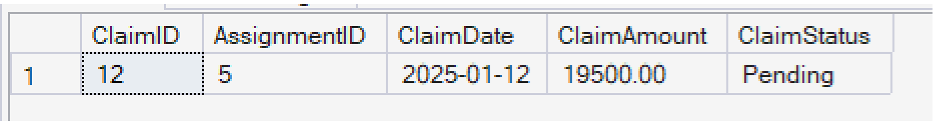
12. Display highest and lowest claimAmount from Claims table.
    Query:
    SELECT
        MAX(ClaimAmount) AS HighestClaimAmount,
        MIN(ClaimAmount) AS LowestClaimAmount
    FROM Claims;
    Output:



	HighestClaimAmount	LowestClaimAmount
1	74250.00	3200.00

13. Display latest claim record.
    Query:
    SELECT *
    FROM Claims
    ORDER BY ClaimDate DESC
    OFFSET 0 ROWS FETCH NEXT 1 ROW ONLY;

    Output:



	ClaimID	AssignmentID	ClaimDate	ClaimAmount	ClaimStatus
1	12	5	2025-01-12	19500.00	Pending

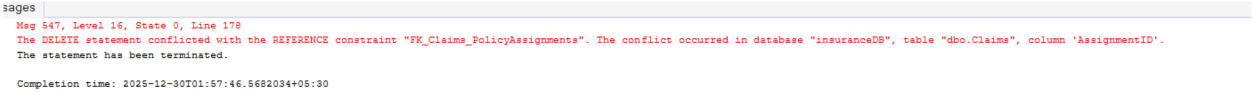
14. Increase premium amount to 10% for all health insurance policies.
    Query:

    UPDATE Policies
    SET PremiumAmount = PremiumAmount * 1.10
    WHERE PolicyType = 'Health';

15. Delete the record of PolicyAssignments whose EndDate is before today's date.

    Query: DELETE FROM PolicyAssignments
    WHERE EndDate < CAST(GETDATE() AS DATE);

    Output:



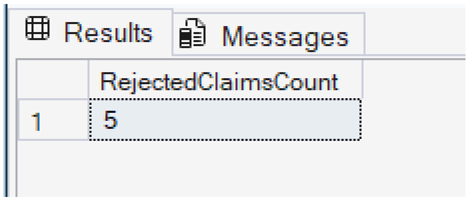
```
Msg 547, Level 16, State 0, Line 178
The DELETE statement conflicted with the REFERENCE constraint "FK_Claims_PolicyAssignments". The conflict occurred in database "insuranceDB", table "dbo.Claims", column 'AssignmentID'.
The statement has been terminated.

Completion time: 2025-12-30T01:57:46.5682034+05:30
```

16. Display no of claims rejected.

    Query:

```
SELECT COUNT(*) AS RejectedClaimsCount
FROM Claims
WHERE ClaimStatus = 'Rejected';
```

    Output:



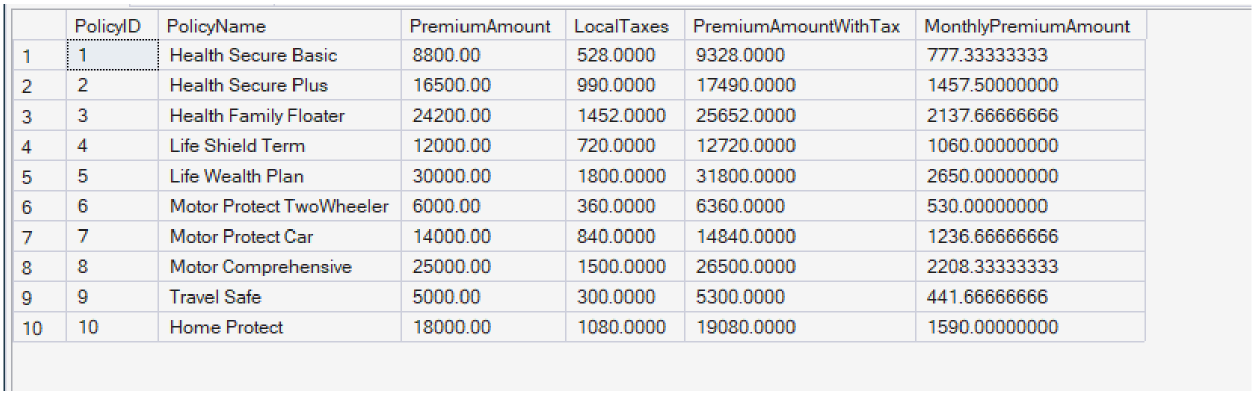
	RejectedClaimsCount
1	5

17. Display PolicyId, PolicyName, PremiumAmount along with computed fields not in table à 6% LocalTaxes, PremiumAmountWithTax and MonthlyPremiumAmount considering PremiumAmount is Annual.

    Query:

```
SELECT
    PolicyID,
    PolicyName,
    PremiumAmount,
    PremiumAmount * 0.06 AS LocalTaxes,
    PremiumAmount + (PremiumAmount * 0.06) AS PremiumAmountWithTax,
    (PremiumAmount + (PremiumAmount * 0.06)) / 12.0 AS MonthlyPremiumAmount
FROM Policies;
```

    Output:



	PolicyID	PolicyName	PremiumAmount	LocalTaxes	PremiumAmountWithTax	MonthlyPremiumAmount
1	1	Health Secure Basic	8800.00	528.0000	9328.0000	777.33333333
2	2	Health Secure Plus	16500.00	990.0000	17490.0000	1457.50000000
3	3	Health Family Floater	24200.00	1452.0000	25652.0000	2137.66666666
4	4	Life Shield Term	12000.00	720.0000	12720.0000	1060.00000000
5	5	Life Wealth Plan	30000.00	1800.0000	31800.0000	2650.00000000
6	6	Motor Protect TwoWheeler	6000.00	360.0000	6360.0000	530.00000000
7	7	Motor Protect Car	14000.00	840.0000	14840.0000	1236.66666666
8	8	Motor Comprehensive	25000.00	1500.0000	26500.0000	2208.33333333
9	9	Travel Safe	5000.00	300.0000	5300.0000	441.66666666
10	10	Home Protect	18000.00	1080.0000	19080.0000	1590.00000000

18. Write a command to add Address and City Columns in the Customers table.
    Query: ALTER TABLE Customers
    ADD Address VARCHAR(100) NULL,
       City   VARCHAR(50) NULL;

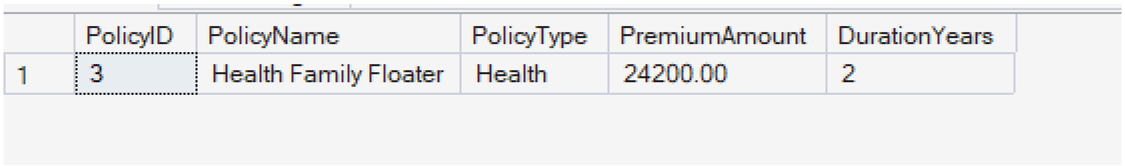
19. Write a command to add a new column named DevOfId (DevelopmentOfficerId) in an existing Agents table.
    Query: ALTER TABLE Agents
    ADD DevOfId INT NULL;

20. Write command to make the above DevOfId as a recursive foreign key to AgentId as Parent.
    Query:
    ALTER TABLE Agents
    ADD CONSTRAINT FK_Agents_DevOfId
    FOREIGN KEY (DevOfId) REFERENCES Agents(AgentID);

5. Queries using Joins, Group By, Having etc.

1. List all Policies for a CustomerId 5.
   Query:
   SELECT p.*
   FROM PolicyAssignments pa
   JOIN Policies p ON p.PolicyID = pa.PolicyID
   WHERE pa.CustomerID = 5;

   Output:



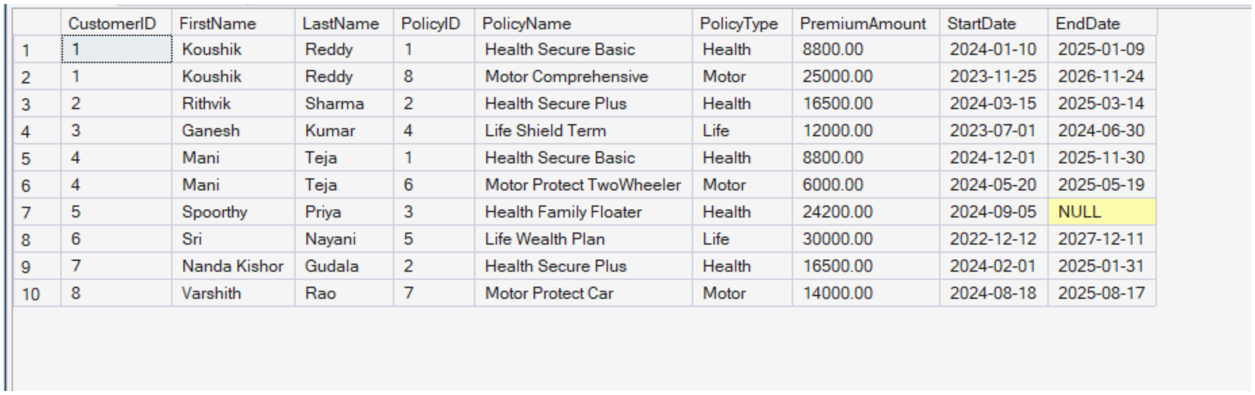
	PolicyID	PolicyName	PolicyType	PremiumAmount	DurationYears
1	3	Health Family Floater	Health	24200.00	2

2. View all customers with their policies.

Query:

```
SELECT
    c.CustomerID, c.FirstName, c.LastName,
    p.PolicyID,p.PolicyName,p.PolicyType,    p.PremiumAmount,
    pa.StartDate,pa.EndDate
FROM Customers c
JOIN PolicyAssignments pa ON pa.CustomerID = c.CustomerID
JOIN Policies p ON p.PolicyID = pa.PolicyID
ORDER BY c.CustomerID, p.PolicyID;
```

Output:



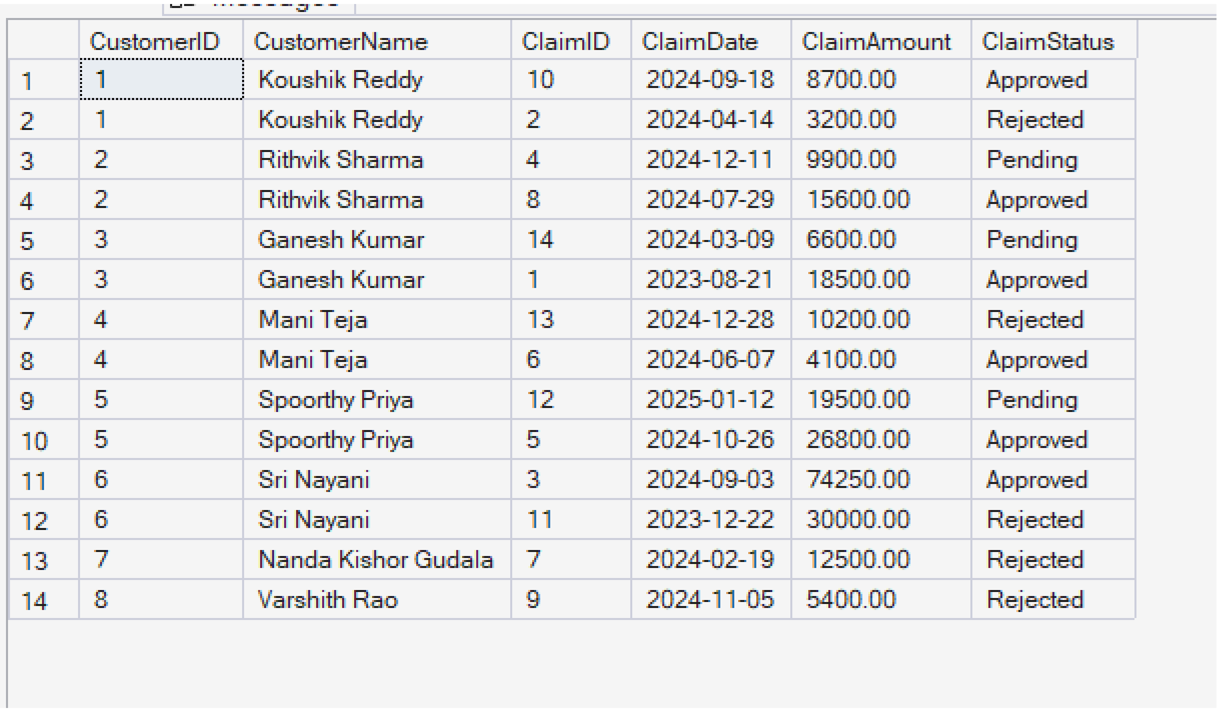
	CustomerID	FirstName	LastName	PolicyID	PolicyName	PolicyType	PremiumAmount	StartDate	EndDate
1	1	Koushik	Reddy	1	Health Secure Basic	Health	8800.00	2024-01-10	2025-01-09
2	1	Koushik	Reddy	8	Motor Comprehensive	Motor	25000.00	2023-11-25	2026-11-24
3	2	Rithvik	Sharma	2	Health Secure Plus	Health	16500.00	2024-03-15	2025-03-14
4	3	Ganesh	Kumar	4	Life Shield Term	Life	12000.00	2023-07-01	2024-06-30
5	4	Mani	Teja	1	Health Secure Basic	Health	8800.00	2024-12-01	2025-11-30
6	4	Mani	Teja	6	Motor Protect TwoWheeler	Motor	6000.00	2024-05-20	2025-05-19
7	5	Spoorthy	Priya	3	Health Family Floater	Health	24200.00	2024-09-05	NULL
8	6	Sri	Nayani	5	Life Wealth Plan	Life	30000.00	2022-12-12	2027-12-11
9	7	Nanda Kishor	Gudala	2	Health Secure Plus	Health	16500.00	2024-02-01	2025-01-31
10	8	Varshith	Rao	7	Motor Protect Car	Motor	14000.00	2024-08-18	2025-08-17

3. View claims with customer name.

Query:

```
SELECT
    c.CustomerID,
    c.FirstName + ' ' + c.LastName AS CustomerName,
    cl.ClaimID,
    cl.ClaimDate,
    cl.ClaimAmount,
    cl.ClaimStatus
FROM Claims cl
JOIN PolicyAssignments pa ON pa.AssignmentID = cl.AssignmentID
JOIN Customers c ON c.CustomerID = pa.CustomerID
ORDER BY c.CustomerID, cl.ClaimDate DESC;
```

Ouput:

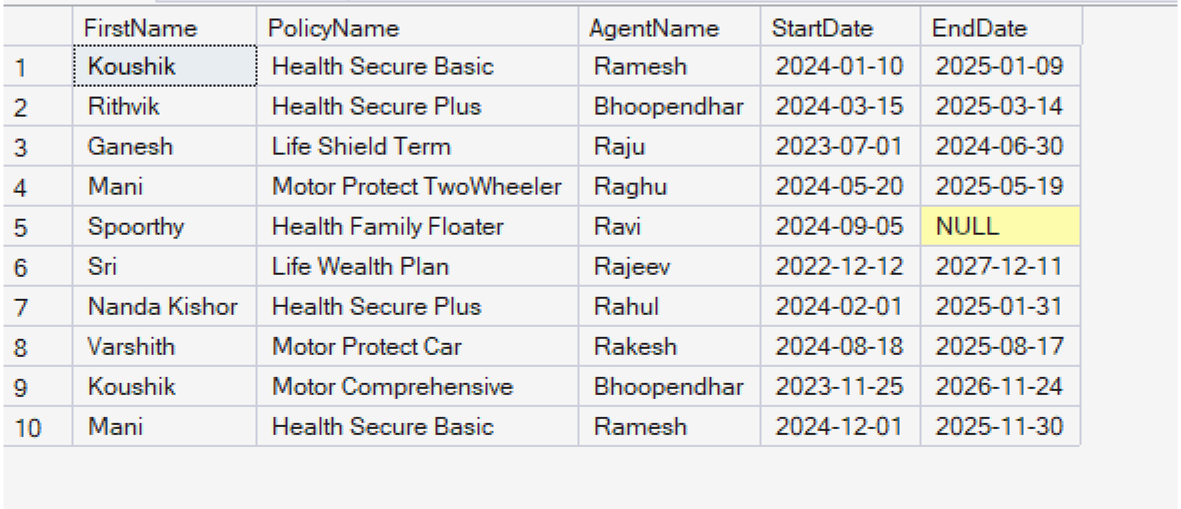


	CustomerID	CustomerName	ClaimID	ClaimDate	ClaimAmount	ClaimStatus
1	1	Koushik Reddy	10	2024-09-18	8700.00	Approved
2	1	Koushik Reddy	2	2024-04-14	3200.00	Rejected
3	2	Rithvik Sharma	4	2024-12-11	9900.00	Pending
4	2	Rithvik Sharma	8	2024-07-29	15600.00	Approved
5	3	Ganesh Kumar	14	2024-03-09	6600.00	Pending
6	3	Ganesh Kumar	1	2023-08-21	18500.00	Approved
7	4	Mani Teja	13	2024-12-28	10200.00	Rejected
8	4	Mani Teja	6	2024-06-07	4100.00	Approved
9	5	Spoorthy Priya	12	2025-01-12	19500.00	Pending
10	5	Spoorthy Priya	5	2024-10-26	26800.00	Approved
11	6	Sri Nayani	3	2024-09-03	74250.00	Approved
12	6	Sri Nayani	11	2023-12-22	30000.00	Rejected
13	7	Nanda Kishor Gudala	7	2024-02-19	12500.00	Rejected
14	8	Varshith Rao	9	2024-11-05	5400.00	Rejected

4. Display FirstName, PolicyName, AgentName, StartDate and EndDate from their respective tables.
   Query:

```
SELECT
    c.FirstName,
    p.PolicyName,
    a.AgentName,
    pa.StartDate,
    pa.EndDate
FROM PolicyAssignments pa
JOIN Customers c ON c.CustomerID = pa.CustomerID
JOIN Policies p   ON p.PolicyID   = pa.PolicyID
JOIN Agents a     ON a.AgentID     = pa.AgentID;
```

   Output:

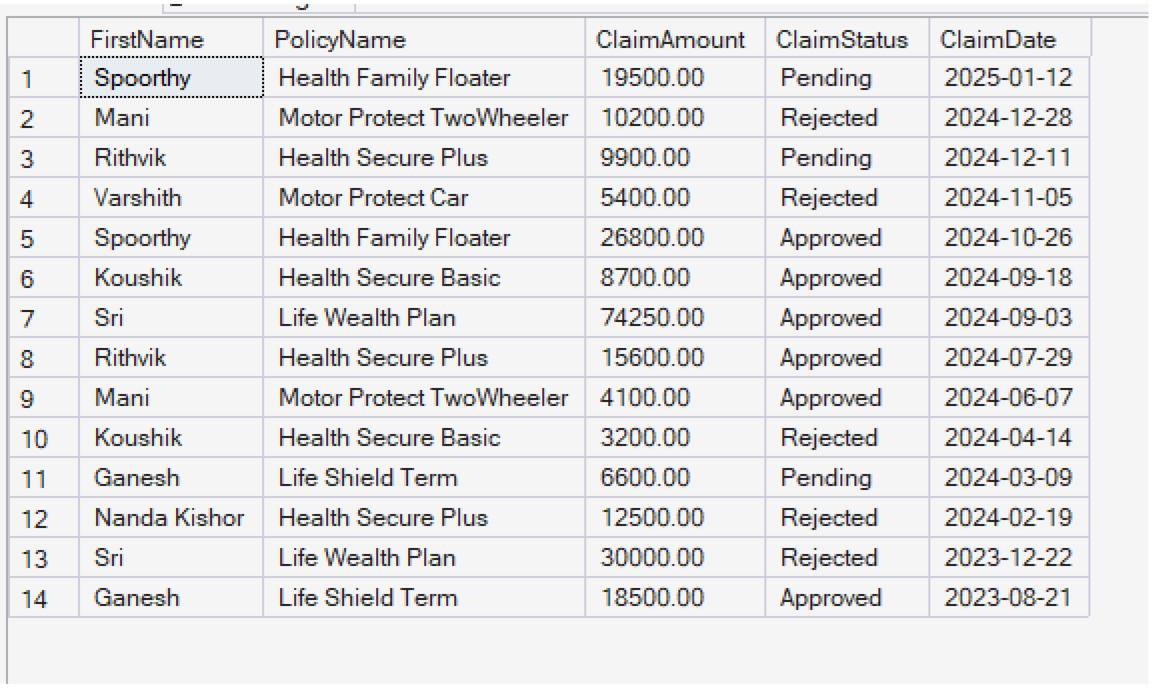


	FirstName	PolicyName	AgentName	StartDate	EndDate
1	Koushik	Health Secure Basic	Ramesh	2024-01-10	2025-01-09
2	Rithvik	Health Secure Plus	Bhoopendhar	2024-03-15	2025-03-14
3	Ganesh	Life Shield Term	Raju	2023-07-01	2024-06-30
4	Mani	Motor Protect TwoWheeler	Raghu	2024-05-20	2025-05-19
5	Spoorthy	Health Family Floater	Ravi	2024-09-05	NULL
6	Sri	Life Wealth Plan	Rajeev	2022-12-12	2027-12-11
7	Nanda Kishor	Health Secure Plus	Rahul	2024-02-01	2025-01-31
8	Varshith	Motor Protect Car	Rakesh	2024-08-18	2025-08-17
9	Koushik	Motor Comprehensive	Bhoopendhar	2023-11-25	2026-11-24
10	Mani	Health Secure Basic	Ramesh	2024-12-01	2025-11-30

5. Display claims report with FirstName, PolicyName, ClaimAmount, ClaimStatus, and ClaimDate from their respective tables.
   Query:

```
SELECT
    c.FirstName,
    p.PolicyName,
    cl.ClaimAmount,
    cl.ClaimStatus,
    cl.ClaimDate
FROM Claims cl
JOIN PolicyAssignments pa ON pa.AssignmentID = cl.AssignmentID
JOIN Customers c          ON c.CustomerID = pa.CustomerID
JOIN Policies p           ON p.PolicyID = pa.PolicyID
ORDER BY cl.ClaimDate DESC;
```

   Output:



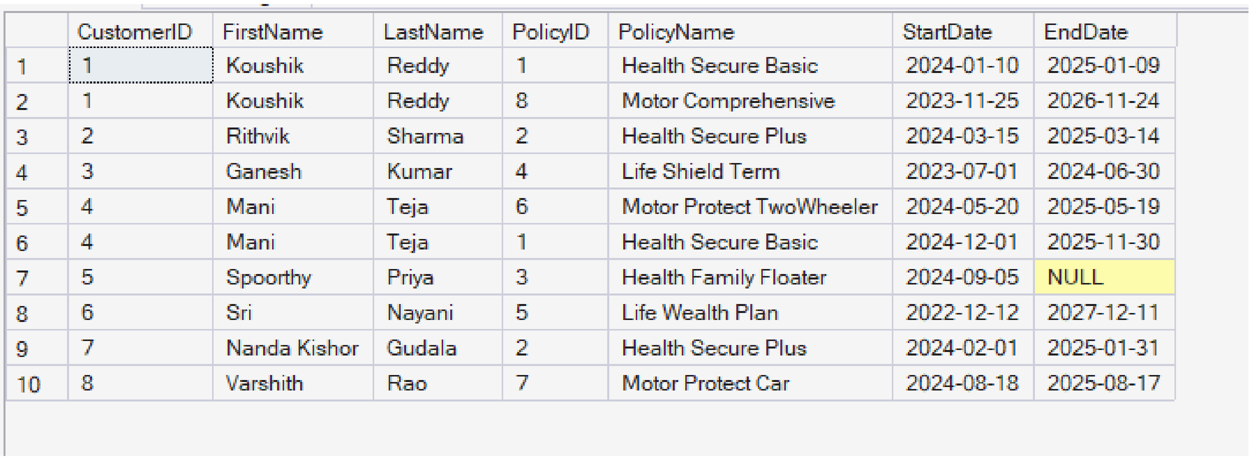
	FirstName	PolicyName	ClaimAmount	ClaimStatus	ClaimDate
1	Spoorthy	Health Family Floater	19500.00	Pending	2025-01-12
2	Mani	Motor Protect TwoWheeler	10200.00	Rejected	2024-12-28
3	Rithvik	Health Secure Plus	9900.00	Pending	2024-12-11
4	Varshith	Motor Protect Car	5400.00	Rejected	2024-11-05
5	Spoorthy	Health Family Floater	26800.00	Approved	2024-10-26
6	Koushik	Health Secure Basic	8700.00	Approved	2024-09-18
7	Sri	Life Wealth Plan	74250.00	Approved	2024-09-03
8	Rithvik	Health Secure Plus	15600.00	Approved	2024-07-29
9	Mani	Motor Protect TwoWheeler	4100.00	Approved	2024-06-07
10	Koushik	Health Secure Basic	3200.00	Rejected	2024-04-14
11	Ganesh	Life Shield Term	6600.00	Pending	2024-03-09
12	Nanda Kishor	Health Secure Plus	12500.00	Rejected	2024-02-19
13	Sri	Life Wealth Plan	30000.00	Rejected	2023-12-22
14	Ganesh	Life Shield Term	18500.00	Approved	2023-08-21

6. Display records of Customers with or without Policies.

Query:

```
SELECT
    c.CustomerID,
    c.FirstName,
    c.LastName,
    p.PolicyID,
    p.PolicyName,
    pa.StartDate,
    pa.EndDate
FROM Customers c
LEFT JOIN PolicyAssignments pa ON pa.CustomerID = c.CustomerID
LEFT JOIN Policies p ON p.PolicyID = pa.PolicyID
```

Output:



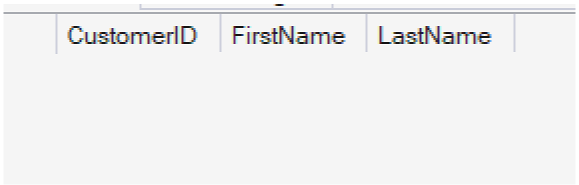
	CustomerID	FirstName	LastName	PolicyID	PolicyName	StartDate	EndDate
1	1	Koushik	Reddy	1	Health Secure Basic	2024-01-10	2025-01-09
2	1	Koushik	Reddy	8	Motor Comprehensive	2023-11-25	2026-11-24
3	2	Rithvik	Sharma	2	Health Secure Plus	2024-03-15	2025-03-14
4	3	Ganesh	Kumar	4	Life Shield Term	2023-07-01	2024-06-30
5	4	Mani	Teja	6	Motor Protect TwoWheeler	2024-05-20	2025-05-19
6	4	Mani	Teja	1	Health Secure Basic	2024-12-01	2025-11-30
7	5	Spoorthy	Priya	3	Health Family Floater	2024-09-05	NULL
8	6	Sri	Nayani	5	Life Wealth Plan	2022-12-12	2027-12-11
9	7	Nanda Kishor	Gudala	2	Health Secure Plus	2024-02-01	2025-01-31
10	8	Varshith	Rao	7	Motor Protect Car	2024-08-18	2025-08-17

7. Display all Customers with NO Claims.

Query:

```
SELECT c.CustomerID, c.FirstName, c.LastName
FROM Customers c
WHERE NOT EXISTS (
   SELECT 1
   FROM PolicyAssignments pa
   JOIN Claims cl ON cl.AssignmentID = pa.AssignmentID
   WHERE pa.CustomerID = c.CustomerID
);
```

Output:



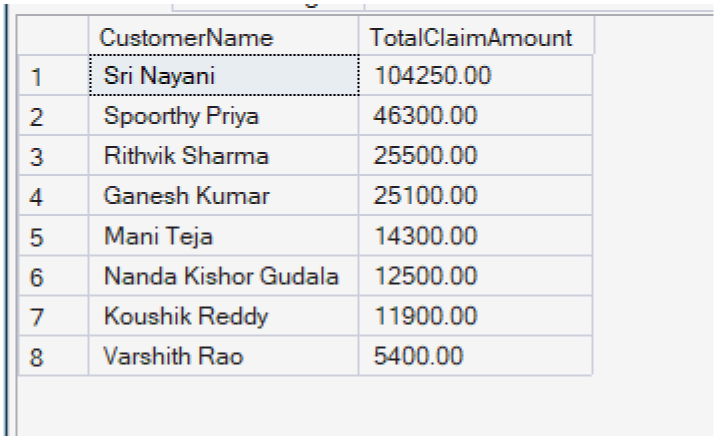
CustomerID	FirstName	LastName

8. Show CustomerName with Total Claim Amount per Customer.

Query:

```
SELECT
   c.FirstName + ' ' + c.LastName AS CustomerName,
   SUM(cl.ClaimAmount) AS TotalClaimAmount
FROM Customers c
JOIN PolicyAssignments pa ON pa.CustomerID = c.CustomerID
JOIN Claims cl ON cl.AssignmentID = pa.AssignmentID
GROUP BY c.FirstName, c.LastName
ORDER BY TotalClaimAmount DESC;
```

Output:



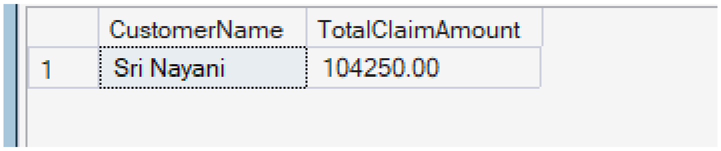
	CustomerName	TotalClaimAmount
1	Sri Nayani	104250.00
2	Spoorthy Priya	46300.00
3	Rithvik Sharma	25500.00
4	Ganesh Kumar	25100.00
5	Mani Teja	14300.00
6	Nanda Kishor Gudala	12500.00
7	Koushik Reddy	11900.00
8	Varshith Rao	5400.00

9. Show names and total claim amount of Customers With Claim Amount > 50000 (Use HAVING Clause).

   Query:

```
SELECT
    c.FirstName + ' ' + c.LastName AS CustomerName,
    SUM(cl.ClaimAmount) AS TotalClaimAmount
FROM Customers c
JOIN PolicyAssignments pa ON pa.CustomerID = c.CustomerID
JOIN Claims cl ON cl.AssignmentID = pa.AssignmentID
GROUP BY c.FirstName, c.LastName
HAVING SUM(cl.ClaimAmount) > 50000
ORDER BY TotalClaimAmount DESC;
```

   Output:



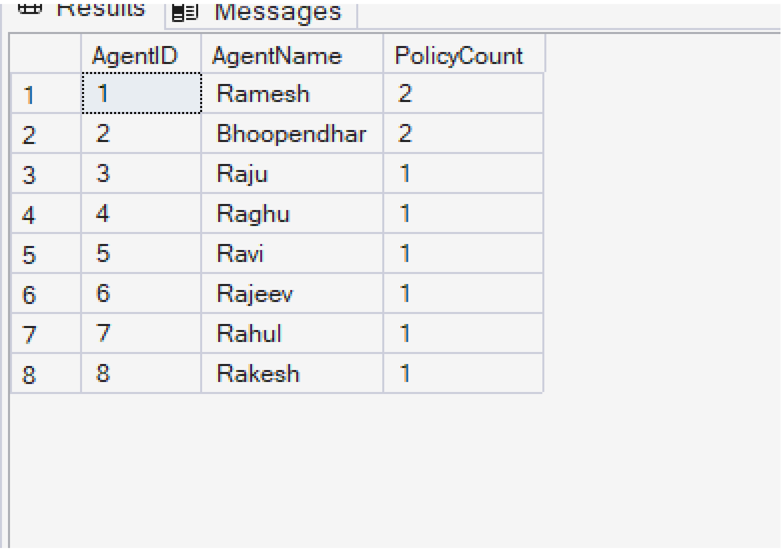
	CustomerName	TotalClaimAmount
1	Sri Nayani	104250.00

10. Display list with Agent Wise Policy Count.

   Query:

```
SELECT
    a.AgentID,
    a.AgentName,
    COUNT(pa.AssignmentID) AS PolicyCount
FROM Agents a
LEFT JOIN PolicyAssignments pa ON pa.AgentID = a.AgentID
GROUP BY a.AgentID, a.AgentName
ORDER BY PolicyCount DESC;
```

   Ouput:



	AgentID	AgentName	PolicyCount
1	1	Ramesh	2
2	2	Bhoopendhar	2
3	3	Raju	1
4	4	Raghu	1
5	5	Ravi	1
6	6	Rajeev	1
7	7	Rahul	1
8	8	Rakesh	1