```
1.create  database insurancedb;
2.use insurancedb;

3.
create table Customers(
CustomerID int primary key identity,
FirstName nchar(50),
LastName nchar(50),
DateOfBirth date,
Phone nchar(50),
Email nchar(50)
);
create table Policies(
PolicyID int primary key identity,
PolicyName nchar(50),
PolicyType nchar(50),
PremiumAmount int,
DurationYears int
);

create table Agents(
AgentID int primary key identity,
AgentName nchar(50),
Phone nchar(10),
City nchar(30)
);

create table PolicyAssignments(
AssignmentID int primary key identity,
CustomerID int,
PolicyID int,
AgentID int,
StartDate date,
EndDate date
constraint customer_fk foreign key(CustomerID) references
Customers(CustomerID),
constraint policy_fk foreign key(PolicyID) references Policies(PolicyID),
constraint agent_fk foreign key(AgentID) references Agents(AgentID));

CREATE TABLE Claims(
ClaimID INT PRIMARY KEY,
AssignmentID INT,
ClaimDate DATE,
ClaimAmount DECIMAL(10,2),
ClaimStatus VARCHAR(10),
Constraint fk_pa_Claims
FOREIGN KEY (AssignmentID) REFERENCES PolicyAssignments(AssignmentID));
```

```
INSERT INTO Customers (FirstName, LastName, DateOfBirth, Phone, Email)
VALUES
('Hari', 'Charan', '2002-05-12', '9876543210', 'hari@gmail.com'),
('Anil', 'Kumar', '1998-08-25', '9123456780', 'anil@gmail.com'),
('Sita', 'Reddy', '2000-01-15', '9012345678', 'sita@gmail.com');

INSERT INTO Policies (PolicyName, PolicyType, PremiumAmount,
DurationYears)
VALUES
('Life Secure', 'Life', 15000, 20),
('Health Plus', 'Health', 12000, 10),
('Car Protect', 'Vehicle', 8000, 5);

INSERT INTO Agents (AgentName, Phone, City)
VALUES
('Ramesh', '9988776655', 'Hyderabad'),
('Suresh', '8877665544', 'Bangalore'),
('Mahesh', '7766554433', 'Chennai');

INSERT INTO PolicyAssignments (CustomerID, PolicyID,
AgentID,StartDate,EndDate)VALUES
(1, 1, 1, '2023-01-01', '2043-01-01'),
(2, 2, 2, '2022-06-01', '2032-06-01'),
(3, 3, 3, '2024-03-15', '2029-03-15');


INSERT INTO Claims (ClaimID, AssignmentID, ClaimDate, ClaimAmount,
ClaimStatus) VALUES
(101, 1, '2024-02-10', 50000.00, 'Approved'),
(102, 2, '2024-05-20', 25000.00, 'Pending'),
(103, 3, '2024-07-05', 18000.00, 'Rejected');

INSERT INTO Customers (FirstName, LastName, DateOfBirth, Phone,
Email)VALUES
('Ravi', 'Teja', '2001-03-10', '9345678123', 'ravi@gmail.com'),
('Neha', 'Sharma', '1999-11-20', '9988776655', 'neha@gmail.com');

INSERT INTO Policies (PolicyName, PolicyType, PremiumAmount,
DurationYears) VALUES
('Health Gold', 'Health', 20000, 1),
('Bike Protect', 'Motor', 6000, 1);

INSERT INTO Agents (AgentName, Phone, City)VALUES
('Arjun', '8899776655', 'Kakinada'),
('Vijay', '7788665544', 'Warangal');
```

```
INSERT INTO PolicyAssignments (CustomerID, PolicyID, AgentID, StartDate,
EndDate) VALUES
(4, 4, 4, '2021-01-01', '2022-01-01'),
(5, 2, 1, '2023-05-01', '2033-05-01'),
(5, 5, 5, '2024-01-01', '2025-01-01');


INSERT INTO Claims (ClaimID, AssignmentID, ClaimDate, ClaimAmount,
ClaimStatus) VALUES
(104, 5, '2024-09-15', 40000.00, 'Approved'),
(105, 6, '2024-10-10', 30000.00, 'Rejected');
```

1. View all records Customers table.

```
 select * from Customers;
```

Results | Messages

	CustomerID	FirstName	LastName	DateOfBirth	Phone	Email
1	1	Hari	Charan	2002-05-12	9876543210	hari@gmail.com
2	2	Anil	Kumar	1998-08-25	9123456780	anil@gmail.com
3	3	Sita	Reddy	2000-01-15	9012345678	sita@gmail.com
4	4	Ravi	Teja	2001-03-10	9345678123	ravi@gmail.com
5	5	Neha	Sharma	1999-11-20	9988776655	neha@gmail.com

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

```
select CustomerId, PolicyId, StartDate,EndDate from policyassignments;
```

	CustomerId	PolicyId	StartDate	EndDate
1	1	1	2023-01-01	2043-01-01
2	2	2	2022-06-01	2032-06-01
3	3	3	2024-03-15	2029-03-15
4	4	4	2021-01-01	2022-01-01
5	5	2	2023-05-01	2033-05-01
6	5	5	2024-01-01	2025-01-01

3. Display all policies of Health type.

```
  select * from Policies where PolicyType='Health';
```

Results | Messages

	PolicyID	PolicyName	PolicyType	PremiumAmount	DurationYears
1	2	Health Plus	Health	13200	10
2	4	Health Gold	Health	20000	1

4. Display policies having premium amount more than 10000 and DurationYears is 1.

```
select * from Policies where PremiumAmount>10000 and DurationYears=1;
```

Results Messages

	PolicyID	PolicyName	PolicyType	PremiumAmount	DurationYears
1	4	Health Gold	Health	20000	1

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

```
select distinct City from Agents;
```

Results Messages

	City
1	Bangalore
2	Chennai
3	Hyderabad
4	Kakinada
5	Warangal

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

```
select * from Policies where PolicyType='Life' or PolicyType='Health' or PolicyType='Motor';
```

Results Messages

	PolicyID	PolicyName	PolicyType	PremiumAmount	DurationYears
1	1	Life Secure	Life	15000	20
2	2	Health Plus	Health	13200	10
3	4	Health Gold	Health	20000	1
4	5	Bike Protect	Motor	6000	1

7. List policies of type Life, Health, Motor use IN operator.

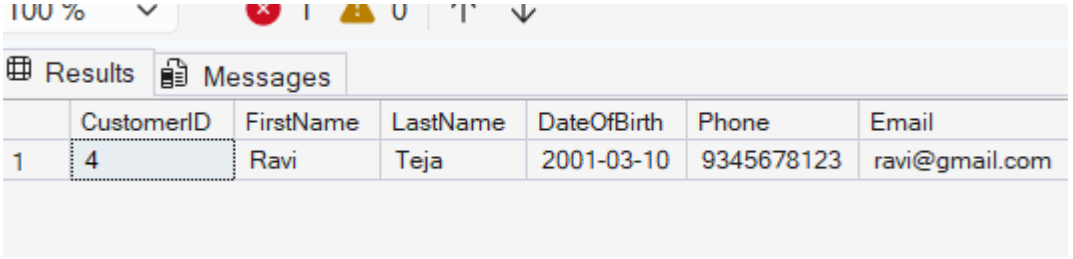
```
select * from Policies Where PolicyType IN ('Life', 'Health', 'Motor');
```

Results Messages

	PolicyID	PolicyName	PolicyType	PremiumAmount	DurationYears
1	1	Life Secure	Life	15000	20
2	2	Health Plus	Health	13200	10
3	4	Health Gold	Health	20000	1
4	5	Bike Protect	Motor	6000	1

8. Display list of customers born after January 1 st , 2001 and before December 31 st , 2020 using >= and <= operators.

```
select * from customers where DateOfBirth>'2001-01-01' and DateOfBirth <
'2001-12-31';
```



	CustomerID	FirstName	LastName	DateOfBirth	Phone	Email
1	4	Ravi	Teja	2001-03-10	9345678123	ravi@gmail.com

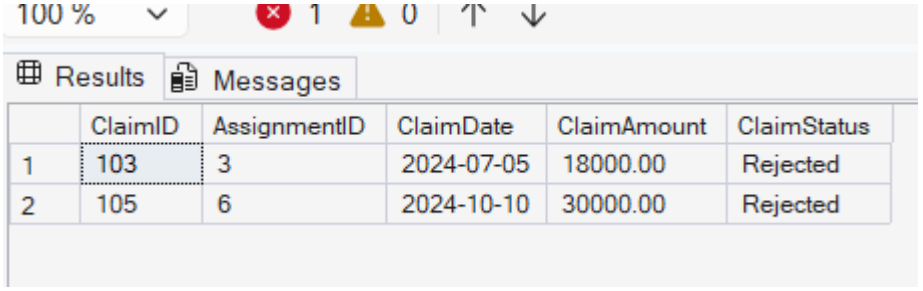
9. Display list of customers born after January 1 st , 2001 and before December 31 st , 2020 using between operator.

```
select * from customers where DateOfBirth between '2001-01-01' and
'2001-12-31';
```

	CustomerID	FirstName	LastName	DateOfBirth	Phone	Email
1	4	Ravi	Teja	2001-03-10	9345678123	ravi@gmail.com

10. Display claims data where claim status is Rejected.

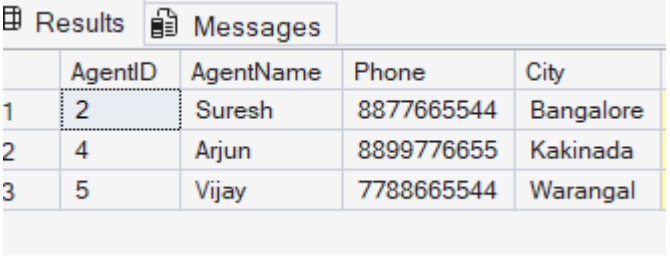
```
select * from Claims where ClaimStatus='Rejected';
```



	ClaimID	AssignmentID	ClaimDate	ClaimAmount	ClaimStatus
1	103	3	2024-07-05	18000.00	Rejected
2	105	6	2024-10-10	30000.00	Rejected

11. Display records of Agents who stay in a city whose second letter is 'a'.

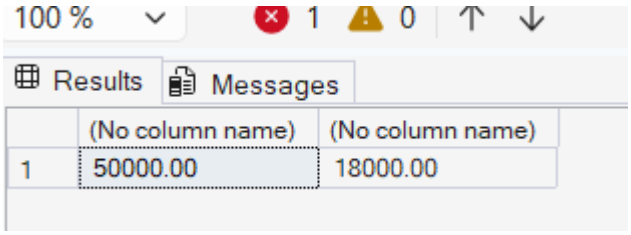
```
select * from Agents where City like '_a%';
```



	AgentID	AgentName	Phone	City
1	2	Suresh	8877665544	Bangalore
2	4	Arjun	8899776655	Kakinada
3	5	Vijay	7788665544	Warangal

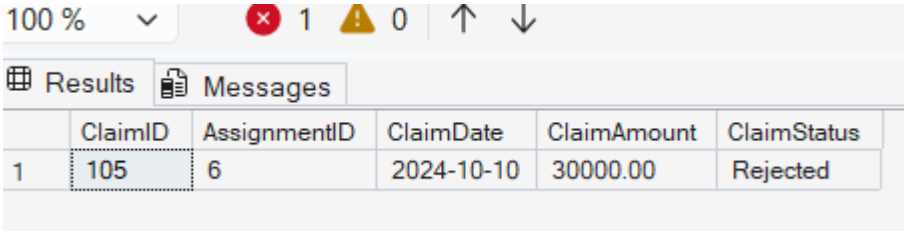
12. Display highest and lowest claimAmount from Claims table.

```
select Max(ClaimAmount),MIN(ClaimAmount) from Claims;
```



13. Display latest claim record.

```
select top 1 * from Claims order by ClaimDate desc;
```



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

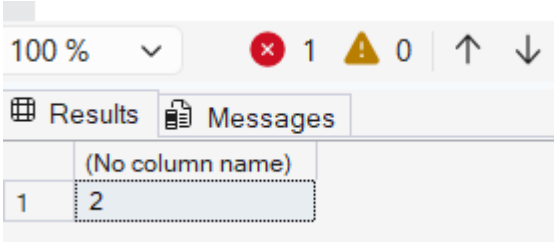
```
update Policies set PremiumAmount=PremiumAmount*1.1 where
PolicyType='Health';
```

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

```
delete from PolicyAssignments where EndDate < cast(getdate()as date);
```

16. Display no of claims rejected.

```
select count(*) from Claims where ClaimStatus='Rejected';
```



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

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

100 % 1 0

Results Messages

	PolicyID	PolicyName	PremiumAmount	LocalTaxes	PremiumAmountWithTax	MonthlyPremiumAmount
1	1	Life Secure	15000	900.00	15900.00	1250.000000
2	2	Health Plus	14520	871.20	15391.20	1210.000000
3	3	Car Protect	8000	480.00	8480.00	666.666666
4	4	Health Gold	22000	1320.00	23320.00	1833.333333
5	5	Bike Protect	6000	360.00	6360.00	500.000000

18. Write a command to add Address and City Columns in the Customers table.

```
alter table customers add  Address varchar(50),City varchar(50);
```

19. Write a command to add a new column named DevOfId (DevelopmentOfficerId) in an existing Agents table

```
alter table Agents add DevOfId int;
```

20. Write command to make the above DevOfId as a recursive foreign key to AgentId as Parent.

```
alter table Agents add constraint Rec_fk foreign key(DevOfId) references Agents(AgentID);
```

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

1. List all Policies for a CustomerId 5.

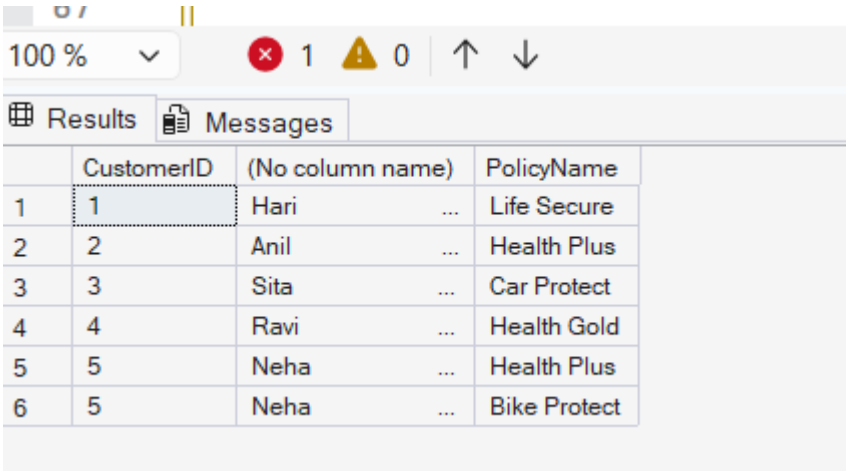
```
SELECT p.PolicyName
FROM Policies p
JOIN PolicyAssignments pa
ON p.PolicyID = pa.PolicyID
WHERE pa.CustomerID = 5;
```

Results Messages

	PolicyName
1	Health Plus
2	Bike Protect

2. View all customers with their policies.

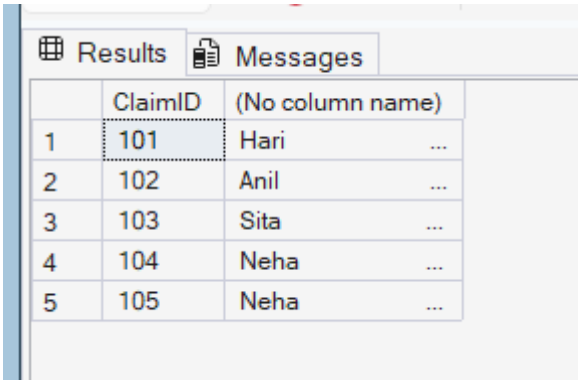
```
SELECT c.CustomerID, c.FirstName + c.LastName,p.PolicyName
FROM Customers c
JOIN PolicyAssignments pa ON c.CustomerID = pa.CustomerID
JOIN Policies p ON p.PolicyID = pa.PolicyID;
```



	CustomerID	(No column name)	PolicyName
1	1	Hari ...	Life Secure
2	2	Anil ...	Health Plus
3	3	Sita ...	Car Protect
4	4	Ravi ...	Health Gold
5	5	Neha ...	Health Plus
6	5	Neha ...	Bike Protect

3. View claims with customer name.

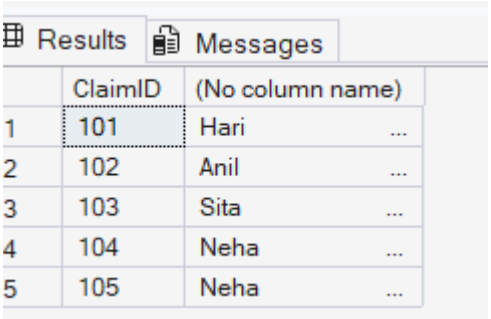
```
 select c.ClaimID,cu.FirstName+cu.Lastname from Customers cu join
PolicyAssignments pa on cu.Customerid=pa.CustomerID join Claims c on
c.AssignmentID=pa.AssignmentID;
```



	ClaimID	(No column name)
1	101	Hari ...
2	102	Anil ...
3	103	Sita ...
4	104	Neha ...
5	105	Neha ...

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

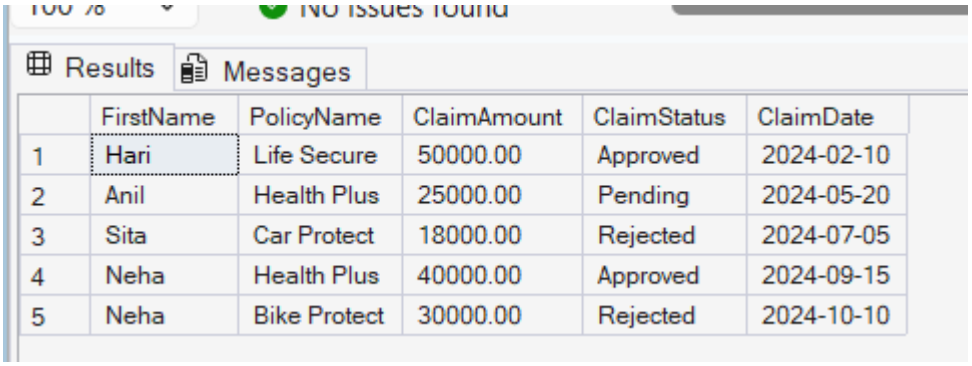
```
 select c.ClaimID,cu.FirstName+cu.Lastname, from Customers cu join
PolicyAssignments pa on cu.Customerid=pa.CustomerID join Claims c on
c.AssignmentID=pa.AssignmentID;
```



	ClaimID	(No column name)
1	101	Hari ...
2	102	Anil ...
3	103	Sita ...
4	104	Neha ...
5	105	Neha ...

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

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

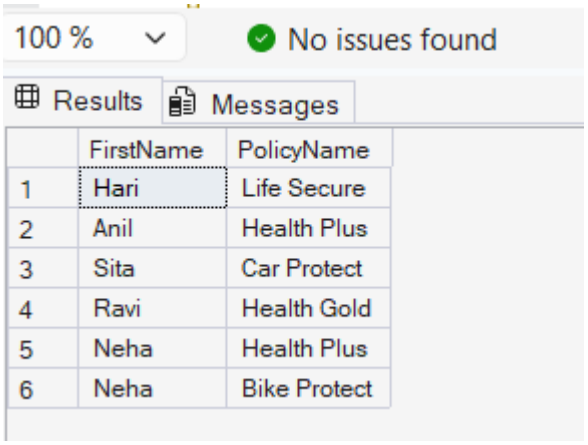


Results | Messages

	FirstName	PolicyName	ClaimAmount	ClaimStatus	ClaimDate
1	Hari	Life Secure	50000.00	Approved	2024-02-10
2	Anil	Health Plus	25000.00	Pending	2024-05-20
3	Sita	Car Protect	18000.00	Rejected	2024-07-05
4	Neha	Health Plus	40000.00	Approved	2024-09-15
5	Neha	Bike Protect	30000.00	Rejected	2024-10-10

6. Display records of Customers with or without Policies.

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



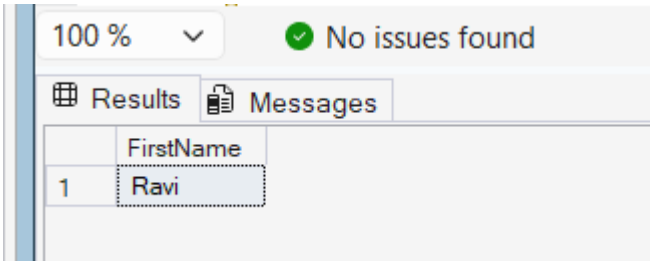
100 % No issues found

Results | Messages

	FirstName	PolicyName
1	Hari	Life Secure
2	Anil	Health Plus
3	Sita	Car Protect
4	Ravi	Health Gold
5	Neha	Health Plus
6	Neha	Bike Protect

7. Display all Customers with NO Claims.

```
SELECT DISTINCT c.FirstName
FROM Customers c
JOIN PolicyAssignments pa ON c.CustomerID=pa.CustomerID
LEFT JOIN Claims cl ON cl.AssignmentID=pa.AssignmentID
WHERE cl.ClaimID IS NULL;
```



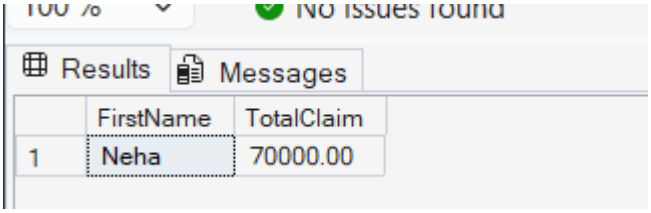
8. Show CustomerName with Total Claim Amount per Customer.

```
SELECT c.FirstName, SUM(cl.ClaimAmount) AS TotalClaim
FROM Customers c
JOIN PolicyAssignments pa ON c.CustomerID=pa.CustomerID
JOIN Claims cl ON cl.AssignmentID=pa.AssignmentID
GROUP BY c.FirstName;
```

	FirstName	TotalClaim
1	Anil	25000.00
2	Hari	50000.00
3	Neha	70000.00
4	Sita	18000.00

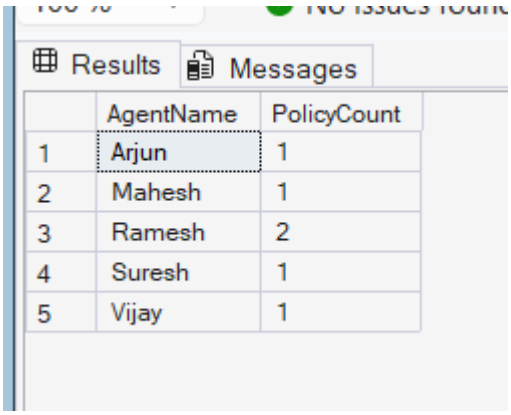
9. Show names and total claim amount of Customers With Claim Amount > 50000 (Use HAVING Clause).

```
SELECT c.FirstName, SUM(cl.ClaimAmount) AS TotalClaim
FROM Customers c
JOIN PolicyAssignments pa ON c.CustomerID=pa.CustomerID
JOIN Claims cl ON cl.AssignmentID=pa.AssignmentID
GROUP BY c.FirstName
HAVING SUM(cl.ClaimAmount) > 50000;
```



10. Display list with Agent Wise Policy Count.

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



	AgentName	PolicyCount
1	Arjun	1
2	Mahesh	1
3	Ramesh	2
4	Suresh	1
5	Vijay	1