

Create database Claims\_Insurance;

#---Customers Table-----

CREATE TABLE Customers (

CustomerID INT auto\_increment PRIMARY KEY,

FirstName VARCHAR(50),

LastName VARCHAR(50),

DateOfBirth DATE,

Gender CHAR(1),

Address VARCHAR(100),

City VARCHAR(50),

State VARCHAR(50),

ZipCode VARCHAR(10)

);

	CustomerID	FirstName	LastName	DateOfBirth	Gender	Address	City	State	ZipCode
▶	1	John	Doe	1980-04-12	M	123 Elm St	Springfield	IL	62704
	2	Jane	Smith	1975-09-23	F	456 Maple Ave	Greenville	TX	75402
	3	Emily	Johnson	1990-01-17	F	789 Oak Dr	Phoenix	AZ	85001
	4	Michael	Brown	1985-07-30	M	321 Pine St	Riverside	CA	92501
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Select \* from customers

#---2. PolicyTypes Table

CREATE TABLE PolicyTypes (

PolicyTypeID INT auto\_increment  
PRIMARY KEY,

PolicyTypeName VARCHAR(50),

Description TEXT

);

	PolicyTypeID	PolicyTypeName	Description
▶	1	Auto	Insurance coverage for automobiles
	2	Home	Insurance coverage for residential homes
	3	Life	Long-term insurance coverage upon the policyh...
	4	Health	Insurance coverage for medical and surgical ex...
*	NULL	NULL	NULL

Select \* from PolicyTypes;

### #---3. Policies Table

CREATE TABLE Policies (

PolicyID INT auto\_increment PRIMARY KEY,

CustomerID INT REFERENCES Customers(CustomerID),

PolicyTypeID INT REFERENCES

PolicyTypes(PolicyTypeID),

PolicyStartDate DATE,

PolicyEndDate DATE,

Premium DECIMAL(10,2)

);

Select \* from Policies;

	PolicyID	CustomerID	PolicyTypeID	PolicyStartDate	PolicyEndDate	Premium
▶	1	1	1	2021-01-01	2022-01-01	120.00
	2	2	2	2021-02-01	2022-02-01	150.00
	3	1	3	2021-03-01	2024-03-01	300.00
	4	3	4	2021-04-01	2022-04-01	200.00
	5	4	1	2021-05-01	2022-05-01	100.00
*	NULL	NULL	NULL	NULL	NULL	NULL

### #---4. Claims Table

CREATE TABLE Claims (

ClaimID INT auto\_increment PRIMARY KEY,

PolicyID INT REFERENCES Policies(PolicyID),

ClaimDate DATE,

ClaimAmount DECIMAL(10,2),

ClaimDescription TEXT,

ClaimStatus VARCHAR(50)

);

Select \* from Claims

	ClaimID	PolicyID	ClaimDate	ClaimAmount	ClaimDescription	ClaimStatus
▶	1	1	2021-06-15	500.00	Car accident	Approved
	2	2	2021-07-20	1000.00	House fire	Pending
	3	3	2021-08-05	20000.00	Life insurance claim	Approved
	4	4	2021-09-10	150.00	Doctor visit	Denied
	5	5	2021-10-22	300.00	Car theft	Approved
*	NULL	NULL	NULL	NULL	NULL	NULL

#####

## -- Task 2: Data Population

- Insert realistic sample data into each table, ensuring a variety of scenarios are represented,
- such as different policy types, claim amounts, and customer profiles.

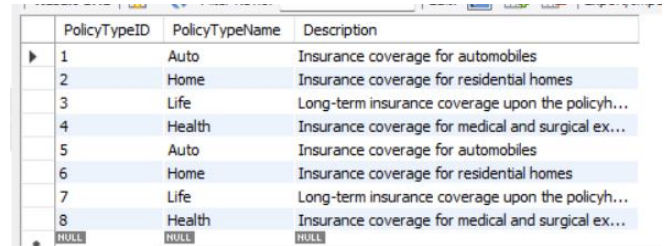
INSERT INTO PolicyTypes (PolicyTypeName, Description) VALUES

('Auto', 'Insurance coverage for automobiles'),

('Home', 'Insurance coverage for residential homes'),

('Life', 'Long-term insurance coverage upon the  
policyholder's death'),

('Health', 'Insurance coverage for medical and surgical  
expenses');



PolicyTypeID	PolicyTypeName	Description
1	Auto	Insurance coverage for automobiles
2	Home	Insurance coverage for residential homes
3	Life	Long-term insurance coverage upon the policyh...
4	Health	Insurance coverage for medical and surgical ex...
5	Auto	Insurance coverage for automobiles
6	Home	Insurance coverage for residential homes
7	Life	Long-term insurance coverage upon the policyh...
8	Health	Insurance coverage for medical and surgical ex...

Select \* from PolicyTypes

-----

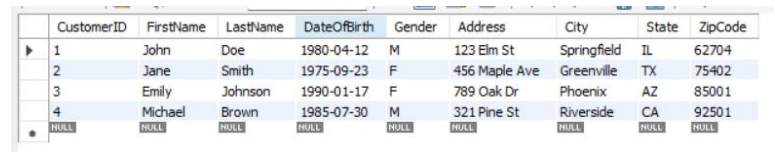
INSERT INTO Customers (FirstName, LastName, DateOfBirth, Gender, Address, City, State, ZipCode)  
VALUES

('John', 'Doe', '1980-04-12', 'M', '123 Elm St', 'Springfield', 'IL', '62704'),

('Jane', 'Smith', '1975-09-23', 'F', '456 Maple Ave',  
'Greenville', 'TX', '75402'),

('Emily', 'Johnson', '1990-01-17', 'F', '789 Oak Dr',  
'Phoenix', 'AZ', '85001'),

('Michael', 'Brown', '1985-07-30', 'M', '321 Pine St', 'Riverside', 'CA', '92501');



CustomerID	FirstName	LastName	DateOfBirth	Gender	Address	City	State	ZipCode
1	John	Doe	1980-04-12	M	123 Elm St	Springfield	IL	62704
2	Jane	Smith	1975-09-23	F	456 Maple Ave	Greenville	TX	75402
3	Emily	Johnson	1990-01-17	F	789 Oak Dr	Phoenix	AZ	85001
4	Michael	Brown	1985-07-30	M	321 Pine St	Riverside	CA	92501

Select \* from Customers;

-----

INSERT INTO Policies (CustomerID, PolicyTypeID, PolicyStartDate, PolicyEndDate, Premium) VALUES

(1, 1, '2021-01-01', '2022-01-01', 120.00),

(2, 2, '2021-02-01', '2022-02-01', 150.00),

(1, 3, '2021-03-01', '2024-03-01', 300.00),

(3, 4, '2021-04-01', '2022-04-01', 200.00),

(4, 1, '2021-05-01', '2022-05-01', 100.00);

	PolicyID	CustomerID	PolicyTypeID	PolicyStartDate	PolicyEndDate	Premium
▶	1	1	1	2021-01-01	2022-01-01	120.00
	2	2	2	2021-02-01	2022-02-01	150.00
	3	1	3	2021-03-01	2024-03-01	300.00
	4	3	4	2021-04-01	2022-04-01	200.00
	5	4	1	2021-05-01	2022-05-01	100.00
•	NULL	NULL	NULL	NULL	NULL	NULL

Select \* from Policies;

-----  
INSERT INTO Claims (PolicyID, ClaimDate, ClaimAmount, ClaimDescription, ClaimStatus) VALUES

(1, '2021-06-15', 500.00, 'Car accident', 'Approved'),

(2, '2021-07-20', 1000.00, 'House fire', 'Pending'),

(3, '2021-08-05', 20000.00, 'Life insurance claim',  
'Approved'),

(4, '2021-09-10', 150.00, 'Doctor visit', 'Denied'),

(5, '2021-10-22', 300.00, 'Car theft', 'Approved');

	ClaimID	PolicyID	ClaimDate	ClaimAmount	ClaimDescription	ClaimStatus
▶	1	1	2021-06-15	500.00	Car accident	Approved
	2	2	2021-07-20	1000.00	House fire	Pending
	3	3	2021-08-05	20000.00	Life insurance claim	Approved
	4	4	2021-09-10	150.00	Doctor visit	Denied
	5	5	2021-10-22	300.00	Car theft	Approved
•	NULL	NULL	NULL	NULL	NULL	NULL

Select \* from Claims;

#####

-- Task 3: Analytical Queries

-- Write a query to calculate the total number of claims per policy type.

-- Use analytical functions to determine the monthly claim frequency and average claim amount.

SELECT

pt.PolicyTypeName,

COUNT(c.ClaimID) AS TotalClaims

FROM

Claims c

JOIN

Policies p ON c.PolicyID = p.PolicyID

JOIN

PolicyTypes pt ON p.PolicyTypeID = pt.PolicyTypeID

GROUP BY

pt.PolicyTypeName

ORDER BY

TotalClaims DESC;

	PolicyTypeName	TotalClaims
▶	Auto	2
	Home	1
	Life	1
	Health	1

-----

## # --- Query 2: Monthly Claim Frequency and Average Claim Amount

SELECT

month(ClaimDate) AS ClaimMonth,

COUNT(\*) AS ClaimFrequency,

AVG(ClaimAmount) AS AverageClaimAmount

FROM

Claims

GROUP BY

ClaimMonth

ORDER BY

ClaimMonth;

	ClaimMonth	ClaimFrequency	AverageClaimAmount
▶	6	1	500.000000
	7	1	1000.000000
	8	1	20000.000000
	9	1	150.000000
	10	1	300.000000

#####

-- Task 4: 4. Optimization with

-- Discuss the creation of indexes on any columns used frequently in WHERE clauses or as join keys to improve performance.

CREATE INDEX idx\_claims\_claimdate ON Claims(ClaimDate);

#####

## # --Task 5: Roles and Permissions

-- Create roles: ClaimsAnalyst and ClaimsManager.

-- 'ClaimsAnalyst' role should have read-only access to claims and policies data.

-- 'ClaimsManager' role should have full access to claims data and the ability to update policy information.

```
CREATE USER 'ClaimsAnalyst'@'%' identified by 'password1';
```

-- Create ClaimsManager Role

```
CREATE USER 'ClaimsManager'@'%' identified by 'password2';
```

-- Grant select on necessary tables

```
GRANT SELECT, INSERT, UPDATE, DELETE ON Claims TO ClaimsAnalyst;
```

```
GRANT SELECT, INSERT, UPDATE, DELETE ON Policies TO ClaimsAnalyst;
```

```
GRANT SELECT, INSERT, UPDATE, DELETE ON PolicyTypes TO ClaimsAnalyst;
```

```
GRANT SELECT, INSERT, UPDATE, DELETE ON Claims TO ClaimsManager;
```

```
GRANT SELECT, INSERT, UPDATE, DELETE ON Policies TO ClaimsManager;
```

```
GRANT SELECT, INSERT, UPDATE, DELETE ON PolicyTypes TO ClaimsManager;
```