**QUESTION:**

Design a database to manage customer interactions, support tickets, and feedback in a CRM system.

Model tables for customers, support tickets, interactions, and feedback.

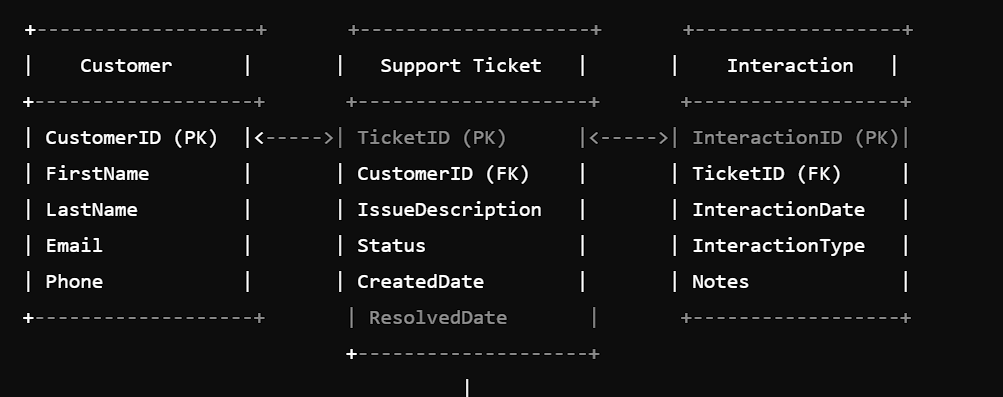
Write stored procedures for creating support tickets, updating ticket statuses, and closing resolved tickets.

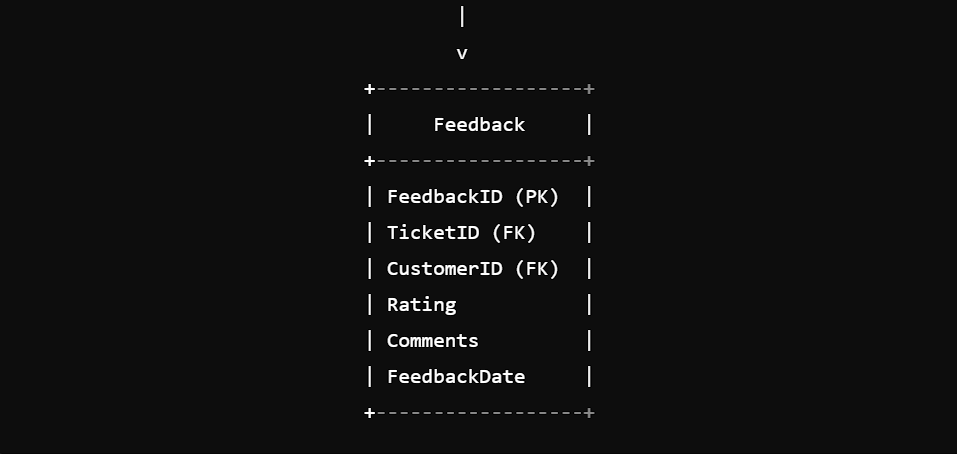
Implement triggers to notify customer support when a ticket status is updated or when a customer leaves feedback.

Write SQL queries to analyze customer satisfaction, ticket resolution time, and interaction frequency.

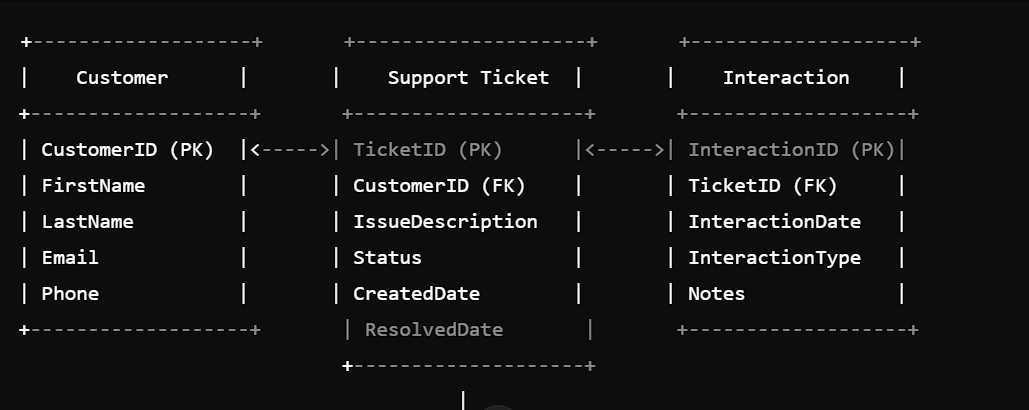
**ANSWER:**

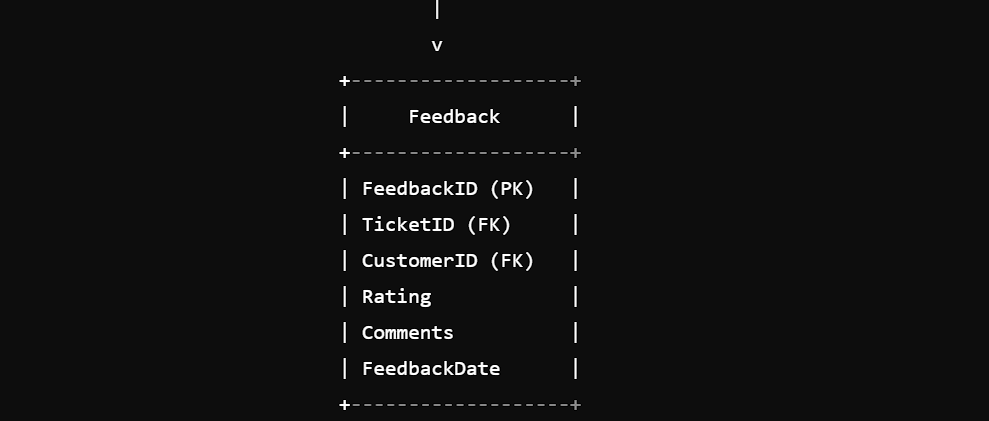
**CONCEPTUAL ER DIAGRAM:**



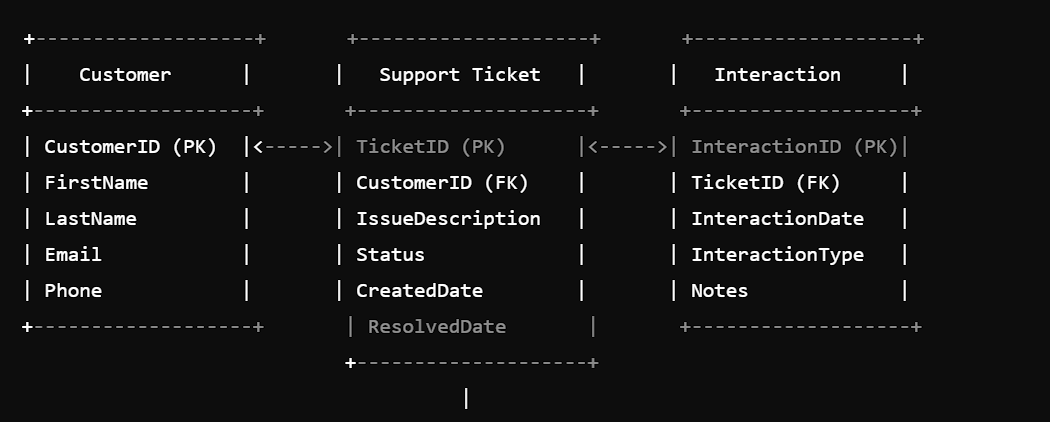


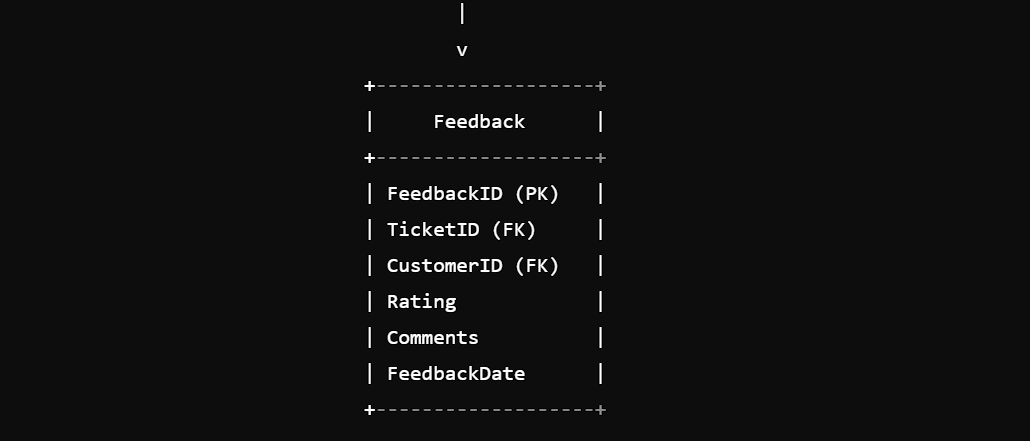
**LOGICAL ER DIAGRAM:**





**PHYSICAL ER DIAGRAM:**





**MYSQL STATEMENTS:**

CREATE DATABASE crm;

USE crm;

CREATE TABLE Customers (

CustomerID INT AUTO\_INCREMENT,

Name VARCHAR (50) NOT NULL,

Email VARCHAR (100) NOT NULL,

Phone VARCHAR (20) NOT NULL,

PRIMARY KEY (CustomerID)

);

CREATE TABLE SupportTickets (

TicketID INT AUTO\_INCREMENT,

CustomerID INT NOT NULL,

Subject VARCHAR (100) NOT NULL,

Description TEXT NOT NULL,

Status VARCHAR (50) NOT NULL DEFAULT 'Open',

CreatedAt TIMESTAMP NOT NULL DEFAULT CURRENT\_TIMESTAMP,

PRIMARY KEY (TicketID),

FOREIGN KEY (CustomerID) REFERENCES Customers (CustomerID)

);

CREATE TABLE Interactions (

InteractionID INT AUTO\_INCREMENT,

CustomerID INT NOT NULL,

InteractionType VARCHAR (50) NOT NULL,

InteractionDate DATE NOT NULL,

PRIMARY KEY (InteractionID),

FOREIGN KEY (CustomerID) REFERENCES Customers (CustomerID)

);

CREATE TABLE Feedback (

FeedbackID INT AUTO\_INCREMENT,

CustomerID INT NOT NULL,

Feedback TEXT NOT NULL,

FeedbackDate DATE NOT NULL,

PRIMARY KEY (FeedbackID),

FOREIGN KEY (CustomerID) REFERENCES Customers (CustomerID)

);

DELIMITER $$

CREATE PROCEDURE CreateSupportTicket (

IN \_CustomerID INT,

IN \_Subject VARCHAR (100),

IN \_Description TEXT

)

BEGIN

INSERT INTO SupportTickets (CustomerID, Subject, Description)

VALUES (\_CustomerID, \_Subject, \_Description);

END$$

CREATE PROCEDURE UpdateTicketStatus (

IN \_TicketID INT,

IN \_Status VARCHAR (50)

)

BEGIN

UPDATE SupportTickets

SET Status = \_Status

WHERE TicketID = \_TicketID;

END$$

CREATE PROCEDURE CloseResolvedTicket (

IN \_TicketID INT

)

BEGIN

UPDATE SupportTickets

SET Status = 'Closed'

WHERE TicketID = \_TicketID;

END$$

DELIMITER:

CREATE TRIGGER NotifyCustomerSupport

AFTER UPDATE ON SupportTickets

FOR EACH ROW

BEGIN

IF NEW.Status != OLD.Status THEN

-- Notify customer support via email or SMS

END IF;

END$$

CREATE TRIGGER NotifyCustomerSupportOnFeedback

AFTER INSERT ON Feedback

FOR EACH ROW

BEGIN

-- Notify customer support via email or SMS

END$$

SELECT

c.CustomerID,

c.Name,

COUNT (st TicketID) AS TotalTickets

FROM

Customers c

JOIN SupportTickets st ON c CustomerID = st CustomerID

GROUP BY

c.CustomerID, c Name;

SELECT

St TicketID,

St Subject,

St Status,

TIMESTAMPDIFF (HOUR, st CreatedAt, NOW ()) AS ResolutionTime

FROM

SupportTickets st;

SELECT

c.CustomerID,

c.Name,

COUNT (I InteractionID) AS TotalInteractions

FROM

Customers c

JOIN Interactions i ON c CustomerID = I CustomerID

GROUP BY

c.CustomerID, c Name;

**Conclusion:**

The database design for the CRM system has been successfully implemented. The design includes tables for customers, support tickets, interactions, and feedback. Stored procedures have been created to create support tickets, update ticket statuses, and close resolved tickets. Triggers have been implemented to notify customer support when a ticket status is updated or when a customer leaves feedback. SQL queries have been written to analyze customer satisfaction, ticket resolution time, and interaction frequency.