**Exercise 5: Triggers**

**Scenario 1:** Automatically update the last modified date when a customer's record is updated.

**Question:** Write a trigger **UpdateCustomerLastModified** that updates the LastModified column of the Customers table to the current date whenever a customer's record is updated.

**Answer:**

DELIMITER //

CREATE TRIGGER UpdateCustomerLastModified

BEFORE UPDATE ON Customers

FOR EACH ROW

BEGIN

SET NEW.LastModified = CURRENT\_TIMESTAMP;

END //

DELIMITER ;

**Scenario 2:** Maintain an audit log for all transactions.

**Question:** Write a trigger **LogTransaction** that inserts a record into an AuditLog table whenever a transaction is inserted into the Transactions table.

**Answer:**

DELIMITER //

CREATE TABLE AuditLog (

AuditID INT AUTO\_INCREMENT PRIMARY KEY,

TransactionID INT,

LogDate DATETIME,

FOREIGN KEY (TransactionID) REFERENCES Transactions(TransactionID)

);

CREATE TRIGGER LogTransaction

AFTER INSERT ON Transactions

FOR EACH ROW

BEGIN

INSERT INTO AuditLog (TransactionID, LogDate) VALUES (NEW.TransactionID, CURRENT\_TIMESTAMP);

END //

DELIMITER ;

**Scenario 3:** Enforce business rules on deposits and withdrawals.

**Question:** Write a trigger **CheckTransactionRules** that ensures withdrawals do not exceed the balance and deposits are positive before inserting a record into the Transactions table.

**Answer:**

DELIMITER //

CREATE TRIGGER CheckTransactionRules

BEFORE INSERT ON Transactions

FOR EACH ROW

BEGIN

IF NEW.TransactionType = 'Withdrawal' THEN

DECLARE accountBalance DECIMAL(10,2);

SELECT Balance INTO accountBalance

FROM Accounts

WHERE AccountID = NEW.AccountID;

IF accountBalance < NEW.Amount THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Insufficient balance for withdrawal';

END IF;

END IF;

IF NEW.TransactionType = 'Deposit' AND NEW.Amount <= 0 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Deposit amount must be positive';

END IF;

END //

DELIMITER ;