**EXERCISE 5 : TRIGGERS**

-- ==========================================

-- Scenario 1: UpdateCustomerLastModified

-- Automatically update LastModified on update

-- ==========================================

DELIMITER //

CREATE TRIGGER UpdateCustomerLastModified

BEFORE UPDATE ON Customers

FOR EACH ROW

BEGIN

SET NEW.LastModified = NOW();

END;

//

-- ==========================================

-- Scenario 2: LogTransaction

-- Insert a row in AuditLog after transaction

-- ==========================================

CREATE TRIGGER LogTransaction

AFTER INSERT ON Transactions

FOR EACH ROW

BEGIN

INSERT INTO AuditLog (AccountID, TransactionDate, Amount, TransactionType)

VALUES (NEW.AccountID, NEW.TransactionDate, NEW.Amount, NEW.TransactionType);

END;

//

-- ==========================================

-- Scenario 3: CheckTransactionRules

-- Enforce rules on deposits/withdrawals

-- ==========================================

CREATE TRIGGER CheckTransactionRules

BEFORE INSERT ON Transactions

FOR EACH ROW

BEGIN

DECLARE acc\_balance DECIMAL(10,2);

-- Get current balance

SELECT Balance INTO acc\_balance

FROM Accounts

WHERE AccountID = NEW.AccountID;

IF NEW.TransactionType = 'Withdrawal' AND NEW.Amount > acc\_balance THEN

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'Withdrawal exceeds available balance';

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 ;

**OUTPUT:**

USE bankdb;

UPDATE Customers SET Balance = Balance + 100 WHERE CustomerID = 1;

SELECT \* FROM Customers;

INSERT INTO Transactions VALUES (3, 1, NOW(), 500, 'Deposit');

SELECT \* FROM AuditLog;

-- Try inserting invalid transaction and see error

INSERT INTO Transactions VALUES (4, 2, NOW(), -100, 'Deposit');



