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.

CREATE OR REPLACE TRIGGER UpdateCustomerLastModified

BEFORE UPDATE ON Customers

FOR EACH ROW

BEGIN

:NEW.LastModified := SYSDATE;

END;

/

# 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.

**AuditLog Table:**

CREATE TABLE AuditLog (

TransactionID NUMBER,

AccountID NUMBER,

TransactionDate DATE,

Amount NUMBER,

TransactionType VARCHAR2(10),

LogTimestamp DATE

);

**LogTransaction Trigger:**

CREATE OR REPLACE TRIGGER LogTransaction

AFTER INSERT ON Transactions

FOR EACH ROW

BEGIN

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

VALUES (

:NEW.TransactionID,

:NEW.AccountID,

:NEW.TransactionDate,

:NEW.Amount,

:NEW.TransactionType,

SYSDATE

);

END;

/

# 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.

CREATE OR REPLACE TRIGGER CheckTransactionRules

BEFORE INSERT ON Transactions

FOR EACH ROW

DECLARE

v\_balance Accounts.Balance%TYPE;

BEGIN

SELECT Balance INTO v\_balance

FROM Accounts

WHERE AccountID = :NEW.AccountID

FOR UPDATE;

IF :NEW.TransactionType = 'Withdrawal' THEN

IF :NEW.Amount > v\_balance THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Withdrawal amount exceeds current balance.');

ELSE

UPDATE Accounts

SET Balance = Balance - :NEW.Amount,

LastModified = SYSDATE

WHERE AccountID = :NEW.AccountID;

END IF;

ELSIF :NEW.TransactionType = 'Deposit' THEN

IF :NEW.Amount <= 0 THEN

RAISE\_APPLICATION\_ERROR(-20002, 'Deposit amount must be positive.');

ELSE

UPDATE Accounts

SET Balance = Balance + :NEW.Amount,

LastModified = SYSDATE

WHERE AccountID = :NEW.AccountID;

END IF;

ELSE

RAISE\_APPLICATION\_ERROR(-20003, 'Invalid transaction type. Must be "Deposit" or "Withdrawal".');

END IF;

END;

/