**Exercise 5: Triggers**

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

CREATE TRIGGER UpdateCustomerLastModified

BEFORE UPDATE ON customers

FOR EACH ROW

BEGIN

:NEW.last\_modified := SYSDATE;

END UpdateCustomerLastModified;

**Scenario 2: Maintain an audit log for all transactions**CREATE TRIGGER LogTransaction

AFTER INSERT ON transactions

FOR EACH ROW

BEGIN

INSERT INTO audit\_log (transaction\_id, account\_id, transaction\_date, amount, transaction\_type)

VALUES (:NEW.transaction\_id, :NEW.account\_id, :NEW.transaction\_date, :NEW.amount, :NEW.transaction\_type);

END LogTransaction;

**Scenario 3: Enforce business rules on deposits and withdrawals**CREATE TRIGGER CheckTransactionRules

BEFORE INSERT ON transactions

FOR EACH ROW

BEGIN

IF :NEW.transaction\_type = 'Withdrawal' THEN

IF :NEW.amount > (SELECT balance FROM accounts WHERE account\_id = :NEW.account\_id) THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Insufficient balance');

END IF;

ELSIF :NEW.transaction\_type = 'Deposit' THEN

IF :NEW.amount <= 0 THEN

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

END IF;

END IF;

END CheckTransactionRules;