**Exercise 3 - Stored Procedures**

**Scenario 1: Process Monthly Interest for Savings Accounts:**

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest

IS

CURSOR c\_SavingsAccounts IS

SELECT AccountID, Balance FROM Accounts WHERE AccountType = 'SAVINGS' FOR UPDATE;

BEGIN

FOR rec IN c\_SavingsAccounts

LOOP

-- Calculate the new balance by applying a 1% interest rate

UPDATE Accounts

SET Balance = Balance + (Balance \* 0.01)

WHERE AccountID = rec.AccountID;

END LOOP;

COMMIT;

END ProcessMonthlyInterest;

/

**Scenario 2: Implement Bonus Scheme for Employees:**

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus (

p\_DepartmentID IN NUMBER,

p\_BonusPercentage IN NUMBER

)

IS

BEGIN

-- Update the salary by adding the bonus percentage

UPDATE Employees

SET Salary = Salary + (Salary \* p\_BonusPercentage / 100)

WHERE DepartmentID = p\_DepartmentID;

COMMIT;

END UpdateEmployeeBonus;

/

**Scenario 3: Transfer Funds Between Accounts:**

-- File: TransferFunds.sql

CREATE OR REPLACE PROCEDURE TransferFunds (

p\_FromAccountID IN NUMBER,

p\_ToAccountID IN NUMBER,

p\_Amount IN NUMBER

)

IS

v\_FromBalance NUMBER;

BEGIN

-- Check the balance of the from account

SELECT Balance INTO v\_FromBalance FROM Accounts WHERE AccountID = p\_FromAccountID FOR UPDATE;

-- Ensure there are sufficient funds

IF v\_FromBalance < p\_Amount THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Error: Insufficient funds for transfer.');

END IF;

-- Perform the transfer

UPDATE Accounts SET Balance = Balance - p\_Amount WHERE AccountID = p\_FromAccountID;

UPDATE Accounts SET Balance = Balance + p\_Amount WHERE AccountID = p\_ToAccountID;

COMMIT;

EXCEPTION

WHEN OTHERS THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Error: An unexpected error occurred during the fund transfer.');

END TransferFunds;

/