**Exercise 3: Stored Procedures**

**Scenario 1:** The bank needs to process monthly interest for all savings accounts.

* + **Question:** Write a stored procedure **ProcessMonthlyInterest** that calculates and updates the balance of all savings accounts by applying an interest rate of 1% to the current balance.

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

    UPDATE Accounts

    SET Balance = Balance \* 1.01,LastModified = SYSDATE

    WHERE AccountType = 'Savings';

END ProcessMonthlyInterest;

/

**Scenario 2:** The bank wants to implement a bonus scheme for employees based on their performance.

* + **Question:** Write a stored procedure **UpdateEmployeeBonus** that updates the salary of employees in a given department by adding a bonus percentage passed as a parameter.

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus (p\_department IN VARCHAR2, p\_bonusPercentage IN NUMBER)

IS

    v\_newSalary NUMBER;

BEGIN

    FOR emp\_rec IN (SELECT EmployeeID, Salary FROM Employees WHERE Department = p\_department) LOOP

        v\_newSalary := emp\_rec.Salary \* (1 + p\_bonusPercentage / 100);

        UPDATE Employees

        SET Salary = v\_newSalary

        WHERE EmployeeID = emp\_rec.EmployeeID;

    END LOOP;

END UpdateEmployeeBonus;

/

**Scenario 3:** Customers should be able to transfer funds between their accounts.

* + **Question:** Write a stored procedure **TransferFunds** that transfers a specified amount from one account to another, checking that the source account has sufficient balance before making the transfer.

CREATE OR REPLACE PROCEDURE TransferFunds

    (p\_fromAccountID IN NUMBER,p\_toAccountID IN NUMBER,p\_amount IN NUMBER)

IS

    v\_balance NUMBER;

BEGIN

    SELECT Balance INTO v\_balance

    FROM Accounts

    WHERE AccountID = p\_fromAccountID;

    IF v\_balance < p\_amount THEN

        RAISE\_APPLICATION\_ERROR(-20001, 'Insufficient funds in the source account.');

    END IF;

    UPDATE Accounts

    SET Balance = Balance - p\_amount,LastModified = SYSDATE WHERE AccountID = p\_fromAccountID;

    UPDATE Accounts

    SET Balance = Balance + p\_amount,LastModified = SYSDATE WHERE AccountID = p\_toAccountID;

EXCEPTION

    WHEN OTHERS THEN

        ROLLBACK;

        RAISE;

END TransferFunds;

/