**Module 3 - PL/SQL Programming**

**Exercise 3: Stored Procedures**

**Question:**

**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.**

**CODE:**

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

UPDATE Accounts

SET Balance = Balance + (Balance \* 0.01)

WHERE AccountType = 'SAVINGS';

COMMIT;

END;

**OUTPUT:**

**Before:**

| **AccountID** | **AccountType** | **Balance** |
| --- | --- | --- |
| 101 | SAVINGS | 10,000 |
| 102 | SAVINGS | 5,000 |
| 103 | CURRENT | 8,000 |

**After running:**

BEGIN

ProcessMonthlyInterest;

END;

**After**:

| **AccountID** | **AccountType** | **Balance** |
| --- | --- | --- |
| 101 | SAVINGS | 10,100.00 |
| 102 | SAVINGS | 5,050.00 |
| 103 | CURRENT | 8,000.00 |

**Only SAVINGS accounts received a 1% interest update.**

**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.**

**CODE:**

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus(

dept\_id IN NUMBER,

bonus\_percent IN NUMBER

) IS

BEGIN

UPDATE Employees

SET Salary = Salary + (Salary \* bonus\_percent / 100)

WHERE DepartmentID = dept\_id;

COMMIT;

END;

**OUTPUT:**

**Before:**

| EmployeeID | Name | DepartmentID | Salary |
| --- | --- | --- | --- |
| 201 | Alice | 10 | 50,000 |
| 202 | Bob | 10 | 60,000 |
| 203 | Carol | 20 | 55,000 |

**After executing :**

BEGIN

UpdateEmployeeBonus(10, 10);

END;

**After:**

| EmployeeID | Name | DepartmentID | Salary |
| --- | --- | --- | --- |
| 201 | Alice | 10 | 55,000 |
| 202 | Bob | 10 | 66,000 |
| 203 | Carol | 20 | 55,000 |

**Only employees in department 10 received a 10% bonus.**

**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.**

**CODE:**

CREATE OR REPLACE PROCEDURE TransferFunds(

from\_acc IN NUMBER,

to\_acc IN NUMBER,

amount IN NUMBER

) IS

from\_balance NUMBER;

BEGIN

-- Get current balance of source account

SELECT Balance INTO from\_balance

FROM Accounts

WHERE AccountID = from\_acc;

IF from\_balance < amount THEN

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

END IF;

-- Deduct from source

UPDATE Accounts

SET Balance = Balance - amount

WHERE AccountID = from\_acc;

-- Add to destination

UPDATE Accounts

SET Balance = Balance + amount

WHERE AccountID = to\_acc;

COMMIT;

END;

**OUTPUT:**

**Before:**

| **AccountID** | **Balance** | **AccountID** |
| --- | --- | --- |
| 301 | 5,000 | 301 |
| 302 | 2,000 | 302 |
| **AccountID** | **Balance** | **AccountID** |

**After executing:**

BEGIN

TransferFunds(301, 302, 1,500);

END;

**Output:**

| **AccountID** | **Balance** |
| --- | --- |
| 301 | 3,500 |
| 302 | 3,500 |

**$1,500 was transferred from account 301 to 302.**