

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 AS
BEGIN
    UPDATE Accounts
    SET Balance = Balance * 1.01
    WHERE AccountType = 'Savings';

    COMMIT;

    DBMS_OUTPUT.PUT_LINE('Monthly interest of 1% has been successfully applied.');
```

EXCEPTION

```
    WHEN OTHERS THEN

        ROLLBACK;

        DBMS_OUTPUT.PUT_LINE('Error occurred: ' || SQLERRM);
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_bonus_percentage IN NUMBER  
) AS  
BEGIN  
    UPDATE Employees  
    SET Salary = Salary * (1 + p_bonus_percentage / 100)  
    WHERE Department = p_department;  
  
    IF SQL%ROWCOUNT = 0 THEN  
        DBMS_OUTPUT.PUT_LINE('No employees found in department ' || p_department || '.');  
        ROLLBACK;  
        RETURN;  
    END IF;  
  
    COMMIT;  
  
    DBMS_OUTPUT.PUT_LINE('Bonus of ' || p_bonus_percentage || '% applied to all employees in department: '  
    || p_department);  
  
EXCEPTION  
    WHEN OTHERS THEN  
        ROLLBACK;  
        DBMS_OUTPUT.PUT_LINE('Error occurred while updating employee bonuses: ' || SQLERRM);  
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_from_account_id IN NUMBER,  
    p_to_account_id IN NUMBER,  
    p_amount IN NUMBER  
) AS  
    v_from_balance NUMBER;  
BEGIN  
    SELECT COUNT(*) INTO v_from_balance  
    FROM Accounts  
    WHERE AccountID IN (p_from_account_id, p_to_account_id);  
  
    IF v_from_balance < 2 THEN  
        RAISE_APPLICATION_ERROR(-20002, 'One or both accounts do not exist.');    END IF;  
  
    SELECT Balance INTO v_from_balance  
    FROM Accounts  
    WHERE AccountID = p_from_account_id  
    FOR UPDATE;  
  
    IF v_from_balance < p_amount THEN  
        RAISE_APPLICATION_ERROR(-20001, 'Insufficient funds in the source account.');    END IF;  
  
    UPDATE Accounts  
    SET Balance = Balance - p_amount
```

```
WHERE AccountID = p_from_account_id;
```

```
UPDATE Accounts
```

```
SET Balance = Balance + p_amount
```

```
WHERE AccountID = p_to_account_id;
```

```
COMMIT;
```

```
DBMS_OUTPUT.PUT_LINE('Transfer of ' || p_amount || ' from account ' ||  
    p_from_account_id || ' to account ' ||  
    p_to_account_id || ' was successful.');
```

```
EXCEPTION
```

```
WHEN OTHERS THEN
```

```
    ROLLBACK;
```

```
    DBMS_OUTPUT.PUT_LINE('Error occurred during transfer: ' || SQLERRM);
```

```
END TransferFunds;
```