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 applied to all savings accounts.'); 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;    COMMIT;    DBMS\_OUTPUT.PUT\_LINE('Bonus applied to all employees in department: ' || p\_department); 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 IN NUMBER,  p\_to\_account IN NUMBER,  p\_amount IN NUMBER ) AS  v\_balance NUMBER; BEGIN  SELECT Balance INTO v\_balance  FROM Accounts  WHERE AccountID = p\_from\_account;    IF v\_balance < p\_amount THEN  RAISE\_APPLICATION\_ERROR(-20001, 'Insufficient funds in account ' || p\_from\_account);  END IF;    BEGIN  UPDATE Accounts  SET Balance = Balance - p\_amount  WHERE AccountID = p\_from\_account;    UPDATE Accounts  SET Balance = Balance + p\_amount  WHERE AccountID = p\_to\_account;    COMMIT;  DBMS\_OUTPUT.PUT\_LINE('Transfer of ' || p\_amount || ' from account ' || p\_from\_account || ' to account ' || p\_to\_account || ' completed successfully.');  EXCEPTION  WHEN OTHERS THEN  ROLLBACK;  DBMS\_OUTPUT.PUT\_LINE('Error: ' || SQLERRM);  END; END TransferFunds; |
| --- |