**Stored Procedures**

**Scenario 1: Process Monthly Interest**

**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

FOR rec IN (SELECT AccountID, Balance FROM Accounts WHERE AccountType = 'Savings') LOOP

UPDATE Accounts

SET Balance = Balance + (Balance \* 0.01)

WHERE AccountID = rec.AccountID;

DBMS\_OUTPUT.PUT\_LINE('Interest applied to Account ID: ' || rec.AccountID);

END LOOP;

COMMIT;

END;  
  
**Scenario 2: Update Employee Bonus**

**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

BEGIN

UPDATE Employees

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

WHERE Department = p\_department;

DBMS\_OUTPUT.PUT\_LINE('Bonus applied to employees in ' || p\_department);

COMMIT;

END;  
  
**Scenario 3: Transfer Funds Between 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\_fromBalance Accounts.Balance%TYPE;

v\_toBalance Accounts.Balance%TYPE;

INSUFFICIENT\_FUNDS EXCEPTION;

BEGIN

SELECT Balance INTO v\_fromBalance FROM Accounts WHERE AccountID = p\_fromAccountID FOR UPDATE;

SELECT Balance INTO v\_toBalance FROM Accounts WHERE AccountID = p\_toAccountID FOR UPDATE;

IF v\_fromBalance < p\_amount THEN

RAISE INSUFFICIENT\_FUNDS;

END IF;

UPDATE Accounts SET Balance = Balance - p\_amount WHERE AccountID = p\_fromAccountID;

UPDATE Accounts SET Balance = Balance + p\_amount WHERE AccountID = p\_toAccountID;

COMMIT;

DBMS\_OUTPUT.PUT\_LINE('Transfer successful from Account ' || p\_fromAccountID || ' to ' || p\_toAccountID);

EXCEPTION

WHEN INSUFFICIENT\_FUNDS THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Error: Insufficient funds in account ' || p\_fromAccountID);

WHEN OTHERS THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Error: ' || SQLERRM);

END;