**Exercise 3: Stored Procedures**

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

**o 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 + (Balance \* 0.01),

LastModified = SYSDATE

WHERE UPPER(AccountType) = 'SAVINGS';

COMMIT;

END;

/

BEGIN

ProcessMonthlyInterest;

END;

**OUTPUT:**

A white paper with black text

AI-generated content may be incorrect.

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

**o 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 (

dpt IN VARCHAR2,

bonus IN NUMBER

) IS

BEGIN

UPDATE Employees

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

WHERE UPPER(Department) = UPPER(dpt);

COMMIT;

END;

/

BEGIN

UpdateEmployeeBonus('IT', 10);

END;

/

**OUTPUT:**

A white background with black text

AI-generated content may be incorrect.

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

**o 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 (

from1 IN NUMBER,

to1 IN NUMBER,

amount IN NUMBER

) IS

v\_balance NUMBER;

BEGIN

SELECT Balance INTO v\_balance

FROM Accounts

WHERE AccountID = from1;

IF v\_balance < amount THEN

DBMS\_OUTPUT.PUT\_LINE('Insufficient balance.');

RETURN;

END IF;

UPDATE Accounts

SET Balance = Balance - amount

WHERE AccountID = from1;

UPDATE Accounts

SET Balance = Balance + amount

WHERE AccountID = to1;

COMMIT;

DBMS\_OUTPUT.PUT\_LINE('Transfer successful.');

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('One of the accounts does not exist.');

WHEN OTHERS THEN

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

END;

/

BEGIN

TransferFunds(1, 2, 300);

END;

/

A white paper with black text

AI-generated content may be incorrect.