**Exercise 3: Stored Procedures**

**Step 1: Process Monthly Interest**

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

-- Update the balance of all savings accounts by applying a 1% interest rate

UPDATE savings\_accounts

SET balance = balance \* 1.01;

COMMIT;

END ProcessMonthlyInterest;

/

**Step 2: Update Employee Bonus**

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus (

p\_department\_id IN NUMBER,

p\_bonus\_percentage IN NUMBER

) IS

BEGIN

-- Update the salary of employees in the given department by adding a bonus percentage

UPDATE employees

SET salary = salary + (salary \* p\_bonus\_percentage / 100)

WHERE department\_id = p\_department\_id;

COMMIT;

END UpdateEmployeeBonus;

/

**Step 3: Transfer Funds**

CREATE OR REPLACE PROCEDURE TransferFunds (

p\_from\_account\_id IN NUMBER,

p\_to\_account\_id IN NUMBER,

p\_amount IN NUMBER

) IS

v\_from\_balance NUMBER;

BEGIN

-- Check if the source account has sufficient balance

SELECT balance INTO v\_from\_balance

FROM accounts

WHERE account\_id = p\_from\_account\_id

FOR UPDATE;

IF v\_from\_balance < p\_amount THEN

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

END IF;

-- Perform the transfer

UPDATE accounts

SET balance = balance - p\_amount

WHERE account\_id = p\_from\_account\_id;

UPDATE accounts

SET balance = balance + p\_amount

WHERE account\_id = p\_to\_account\_id;

COMMIT;

EXCEPTION

WHEN OTHERS THEN

ROLLBACK;

RAISE;

END TransferFunds;

/

**Example Data Insertion Scripts**

-- Creating tables

CREATE TABLE savings\_accounts (

account\_id NUMBER PRIMARY KEY,

balance NUMBER

);

CREATE TABLE employees (

employee\_id NUMBER PRIMARY KEY,

department\_id NUMBER,

salary NUMBER

);

CREATE TABLE accounts (

account\_id NUMBER PRIMARY KEY,

balance NUMBER

);

-- Inserting example data

INSERT INTO savings\_accounts (account\_id, balance) VALUES (1, 1000);

INSERT INTO savings\_accounts (account\_id, balance) VALUES (2, 2000);

INSERT INTO employees (employee\_id, department\_id, salary) VALUES (1, 10, 50000);

INSERT INTO employees (employee\_id, department\_id, salary) VALUES (2, 20, 60000);

INSERT INTO employees (employee\_id, department\_id, salary) VALUES (3, 10, 55000);

INSERT INTO accounts (account\_id, balance) VALUES (1, 1000);

INSERT INTO accounts (account\_id, balance) VALUES (2, 2000);

INSERT INTO accounts (account\_id, balance) VALUES (3, 1500);

COMMIT;