**Exercise 3: Stored Procedures**

Scenario 1: Process Monthly Interest

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest AS

BEGIN

UPDATE savings\_accounts

SET balance = balance \* 1.01; -- 1% interest

COMMIT;

DBMS\_OUTPUT.PUT\_LINE('Interest processed.');

END ProcessMonthlyInterest;

/

Scenario 2: Update Employee Bonus

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus(

p\_department\_id IN NUMBER,

p\_bonus\_percent IN NUMBER

) AS

BEGIN

UPDATE employees

SET salary = salary + salary \* p\_bonus\_percent / 100

WHERE department\_id = p\_department\_id;

COMMIT;

DBMS\_OUTPUT.PUT\_LINE('Bonus updated.');

END UpdateEmployeeBonus;

/

Scenario 3: Transfer Funds Between Accounts

CREATE OR REPLACE PROCEDURE TransferFunds(

p\_from\_account\_id IN NUMBER,

p\_to\_account\_id IN NUMBER,

p\_amount IN NUMBER

) AS

BEGIN

-- Transfer funds

UPDATE accounts

SET balance = CASE

WHEN account\_id = p\_from\_account\_id THEN balance - p\_amount

WHEN account\_id = p\_to\_account\_id THEN balance + p\_amount

ELSE balance

END

WHERE account\_id IN (p\_from\_account\_id, p\_to\_account\_id);

-- Validate update

IF SQL%ROWCOUNT < 2 THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Invalid accounts or insufficient funds.');

END IF;

COMMIT;

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

EXCEPTION

WHEN OTHERS THEN

ROLLBACK;

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

END TransferFunds;

/