WEEK 2

Mandatory Hands-On

**PL/ SQL Programming**

**EX:3 Stored Procedures**

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

**PL/ SQL QUERY:**

CREATE PROCEDURE ProcessMonthlyInterest

IS

BEGIN

UPDATE Accounts

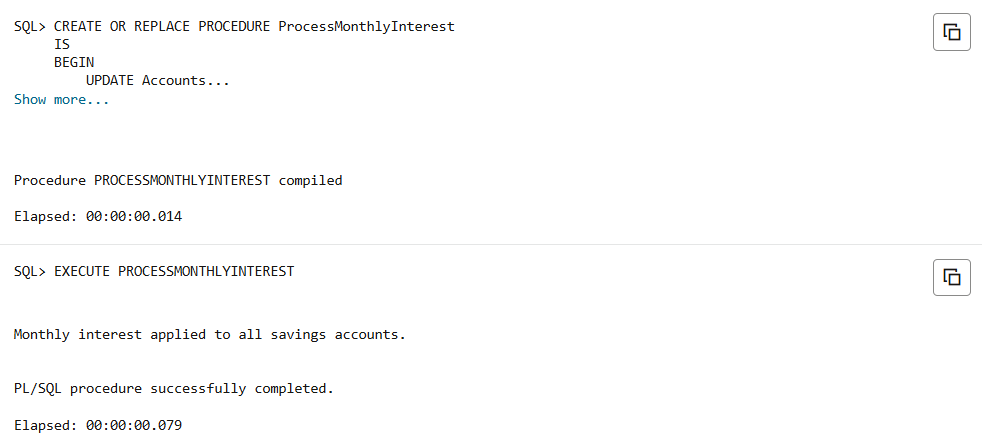
SET Balance = Balance + (Balance \* 0.01),

LastModified = SYSDATE

WHERE AccountType = 'Savings';

DBMS\_OUTPUT.PUT\_LINE('Monthly interest applied to all savings accounts.');

END;

**OUTPUT:**

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

**PL/ SQL QUERY:**

CREATE PROCEDURE UpdateEmployeeBonus (

p\_department IN VARCHAR2,

p\_bonus\_pct IN NUMBER

)

IS

BEGIN

UPDATE Employees

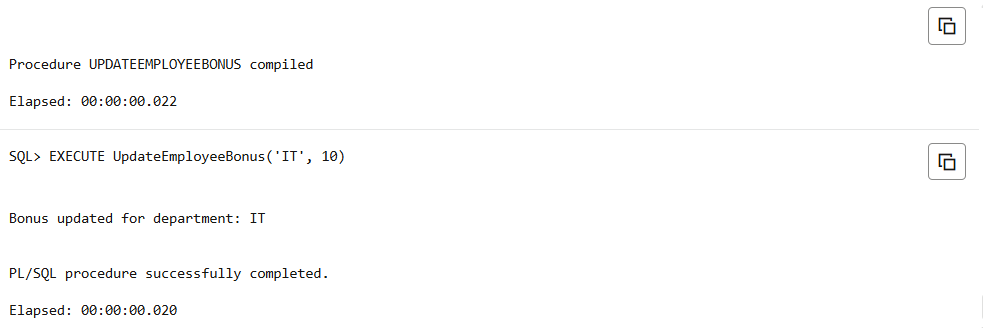
SET Salary = Salary + (Salary \* p\_bonus\_pct / 100)

WHERE Department = p\_department;

DBMS\_OUTPUT.PUT\_LINE('Bonus updated for department: ' || p\_department);

END;

**OUTPUT:**

****

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

**PL/ SQL QUERY:**

CREATE OR REPLACE PROCEDURE TransferFunds (

p\_from\_account IN NUMBER,

p\_to\_account IN NUMBER,

p\_amount IN NUMBER

)

IS

v\_from\_balance NUMBER;

BEGIN

SELECT Balance INTO v\_from\_balance

FROM Accounts

WHERE AccountID = p\_from\_account;

IF v\_from\_balance < p\_amount THEN

DBMS\_OUTPUT.PUT\_LINE('Transfer failed: Not enough balance.');

RETURN;

END IF;

UPDATE Accounts

SET Balance = Balance - p\_amount,

LastModified = SYSDATE

WHERE AccountID = p\_from\_account;

UPDATE Accounts

SET Balance = Balance + p\_amount,

LastModified = SYSDATE

WHERE AccountID = p\_to\_account;

DBMS\_OUTPUT.PUT\_LINE('Transfer of ' || p\_amount || ' completed from Account ' ||

p\_from\_account || ' to Account ' || p\_to\_account);

END;

**OUTPUT:**

