**PL/SQL programming (Mandatory hands-on)**

**Exercise 1: Control Structures**

**Scenario 1: Apply discount for customers above 60**

BEGIN

FOR cust IN (SELECT CustomerID FROM Customers WHERE MONTHS\_BETWEEN(SYSDATE, DOB)/12 > 60) LOOP

UPDATE Loans

SET InterestRate = InterestRate - 1

WHERE CustomerID = cust.CustomerID;

END LOOP;

END;

/

**Scenario 2: Set VIP flag**

**Assuming we add IsVIP column in Customers:**

**ALTER TABLE Customers ADD IsVIP CHAR(1);**

BEGIN

FOR cust IN (SELECT CustomerID FROM Customers WHERE Balance > 10000) LOOP

UPDATE Customers

SET IsVIP = 'Y'

WHERE CustomerID = cust.CustomerID;

END LOOP;

END;

/

**Scenario 3: Send loan due reminders**

BEGIN

FOR loan\_rec IN (

SELECT L.LoanID, C.Name

FROM Loans L

JOIN Customers C ON L.CustomerID = C.CustomerID

WHERE EndDate BETWEEN SYSDATE AND SYSDATE + 30

) LOOP

DBMS\_OUTPUT.PUT\_LINE('Reminder: Loan ID ' || loan\_rec.LoanID || ' for customer ' || loan\_rec.Name || ' is due soon.');

END LOOP;

END;

/

**RESULT TABLE:**



**Exercise 3: Stored Procedures**

**Scenario 1: Process monthly interest**

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

UPDATE Accounts

SET Balance = Balance + (Balance \* 0.01)

WHERE AccountType = 'Savings';

COMMIT;

END;

/

**Scenario 2: Update employee bonus**

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus(p\_dept VARCHAR2, p\_bonus\_pct NUMBER) IS

BEGIN

UPDATE Employees

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

WHERE Department = p\_dept;

COMMIT;

END;

/

**Scenario 3: Transfer funds**

CREATE OR REPLACE PROCEDURE TransferFunds(p\_from NUMBER, p\_to NUMBER, p\_amt NUMBER) IS

bal NUMBER;

BEGIN

SELECT Balance INTO bal FROM Accounts WHERE AccountID = p\_from FOR UPDATE;

IF bal >= p\_amt THEN

UPDATE Accounts SET Balance = Balance - p\_amt WHERE AccountID = p\_from;

UPDATE Accounts SET Balance = Balance + p\_amt WHERE AccountID = p\_to;

COMMIT;

ELSE

RAISE\_APPLICATION\_ERROR(-20002, 'Insufficient balance.');

END IF;

EXCEPTION

WHEN OTHERS THEN

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

ROLLBACK;

END;

/

**RESULT TABLE:**

