**Superset ID: 6394198**

**Name: Priyadharshini S**

**EXERCISE 1: CONTROL STRUCTURES**

**Scenario 1:**

BEGIN

FOR rec IN (

SELECT c.CustomerID, l.LoanID, l.InterestRate, c.Age

FROM Customers c

JOIN Loans l ON c.CustomerID = l.CustomerID

)

LOOP

IF rec.Age > 60 THEN

UPDATE Loans

SET InterestRate = rec.InterestRate - 1

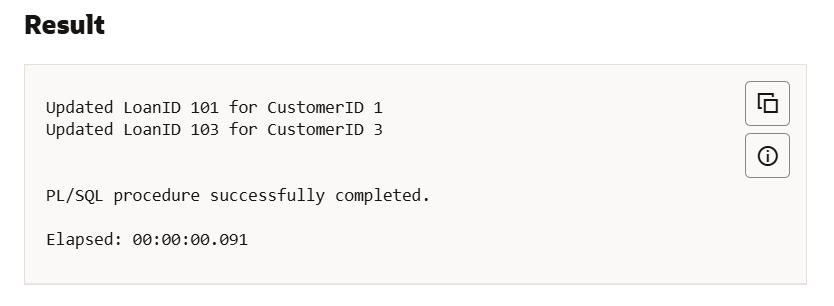
WHERE LoanID = rec.LoanID;

DBMS\_OUTPUT.PUT\_LINE('Updated LoanID ' || rec.LoanID || ' for CustomerID ' || rec.CustomerID);

END IF;

END LOOP;

END;

****

**Scenario 2:**

BEGIN

FOR rec IN (SELECT CustomerID, Balance

FROM Customers)

LOOP

IF rec.Balance > 10000 THEN

UPDATE Customers

SET IsVIP = 'TRUE'

WHERE CustomerID = rec.CustomerID;

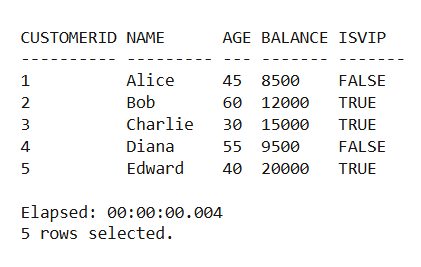
END IF;

END LOOP;

COMMIT;

END;

**Output:**



**Scenario 3:**

BEGIN

FOR rec IN (

SELECT l.LoanID, l.DueDate, c.CustomerID, c.Name

FROM Loans l

JOIN Customers c ON l.CustomerID = c.CustomerID

WHERE l.DueDate BETWEEN SYSDATE AND SYSDATE + 30

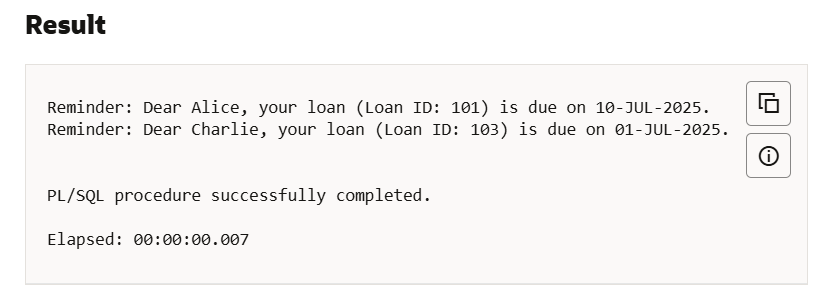
) LOOP

DBMS\_OUTPUT.PUT\_LINE('Reminder: Dear ' || rec.Name ||

', your loan (Loan ID: ' || rec.LoanID ||

') is due on ' || TO\_CHAR(rec.DueDate, 'DD-MON-YYYY') || '.');

END LOOP;

END;

**EXERCISE 3: STORED PROCEDURES**

**Scenario 1:**

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

FOR acc\_rec IN (

SELECT AccountID, Balance

FROM Accounts

WHERE AccountType = 'SAVINGS'

) LOOP

UPDATE Accounts

SET Balance = acc\_rec.Balance + (acc\_rec.Balance \* 0.01)

WHERE AccountID = acc\_rec.AccountID;

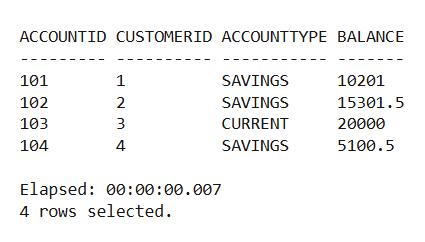
END LOOP;

COMMIT;

END;

/

**Output:**

****

**Scenario 2:**

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus(

dept\_id IN NUMBER,

bonus\_percent IN NUMBER

) IS

BEGIN

UPDATE Employees

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

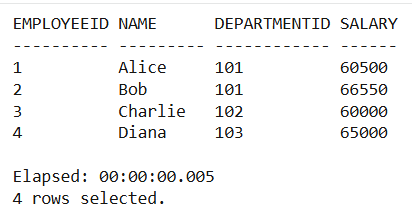
WHERE DepartmentID = dept\_id;

COMMIT;

END;

/

**Output:**



**Scenario 3:**

CREATE OR REPLACE PROCEDURE TransferFunds(

source\_acc IN NUMBER,

target\_acc IN NUMBER,

amount IN NUMBER

) IS

source\_balance NUMBER;

BEGIN

SELECT Balance INTO source\_balance

FROM Accounts

WHERE AccountID = source\_acc

FOR UPDATE;

IF source\_balance < amount THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Not enough balance to transfer.');

END IF;

UPDATE Accounts

SET Balance = Balance - amount

WHERE AccountID = source\_acc;

UPDATE Accounts

SET Balance = Balance + amount

WHERE AccountID = target\_acc;

COMMIT;

END;

/

**Output:**

