**Module 3 - PL/SQL Programming**

**Exercise 1: Control Structures**

**Question:**

**Scenario 1: The bank wants to apply a discount to loan interest rates for customers above 60 years old.**

**Question: Write a PL/SQL block that loops through all customers, checks their age, and if they are above 60, apply a 1% discount to their current loan interest rates.**

**CODE:**

DECLARE

CURSOR cust\_cursor IS

SELECT CustomerID, Age, InterestRate

FROM Customers

WHERE Age > 60;

BEGIN

FOR cust\_rec IN cust\_cursor LOOP

UPDATE Customers

SET InterestRate = InterestRate - 1

WHERE CustomerID = cust\_rec.CustomerID;

END LOOP;

COMMIT;

END;

**OUTPUT:**

**Before:**

| **CustomerID** | **Age** | **InterestRate** |
| --- | --- | --- |
| 101 | 65 | 9.5% |
| 102 | 45 | 10.0% |
| 103 | 70 | 8.0% |

**After**:

| **CustomerID** | **Age** | **InterestRate** |
| --- | --- | --- |
| 101 | 65 | 8.5% |
| 102 | 45 | 10.0% |
| 103 | 70 | 7.0% |

**Customers 101 and 103 got a 1% reduction.**

**Scenario 2: A customer can be promoted to VIP status based on their balance.**

**Question: Write a PL/SQL block that iterates through all customers and sets a flag IsVIP to TRUE for those with a balance over $10,000.**

**CODE:**

DECLARE

CURSOR vip\_cursor IS

SELECT CustomerID, Balance

FROM Customers

WHERE Balance > 10000;

BEGIN

FOR vip\_rec IN vip\_cursor LOOP

UPDATE Customers

SET IsVIP = 'TRUE'

WHERE CustomerID = vip\_rec.CustomerID;

END LOOP;

COMMIT;

END;

**OUTPUT:**

**Before:**

| **CustomerID** | **Balance** | **IsVIP** |
| --- | --- | --- |
| 201 | 9500 | FALSE |
| 202 | 12000 | FALSE |
| 203 | 15000 | FALSE |

**After:**

| **CustomerID** | **Balance** | **IsVIP** | **CustomerID** |
| --- | --- | --- | --- |
| 201 | 9500 | FALSE | 201 |
| 202 | 12000 | TRUE | 202 |
| 203 | 15000 | TRUE | 203 |

**Customers 202 and 203 are now VIPs.**

**Scenario 3: The bank wants to send reminders to customers whose loans are due within the next 30 days.**

**Question:Write a PL/SQL block that fetches all loans due in the next 30 days and prints a reminder message for each customer.**

**CODE:**

DECLARE

CURSOR loan\_cursor IS

SELECT LoanID, CustomerID, DueDate

FROM Loans

WHERE DueDate BETWEEN SYSDATE AND SYSDATE + 30;

BEGIN

FOR loan\_rec IN loan\_cursor LOOP

DBMS\_OUTPUT.PUT\_LINE('Reminder: Customer ' || loan\_rec.CustomerID ||

' has a loan (Loan ID: ' || loan\_rec.LoanID ||

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

END LOOP;

END;

**OUTPUT:**

**Before:**

| **LoanID** | **CustomerID** | **DueDate** |
| --- | --- | --- |
| 301 | 401 | 15-JUL-25 |
| 302 | 402 | 29-JUL-25 |
| 303 | 403 | 10-AUG-25 |

**System date: 29-JUN-25**

**After:**

Reminder: Customer 401 has a loan (Loan ID: 301) due on 15-JUL-2025

Reminder: Customer 402 has a loan (Loan ID: 302) due on 29-JUL-2025

**Only loans due before 29-JUL-25 + 30 days are included.**