**Exercise 1: Control Structures**

**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.

**SOLUTION:**

DECLARE

CURSOR cur\_Customers IS

SELECT CustomerID,

TRUNC((MONTHS\_BETWEEN(SYSDATE, DOB) / 12)) AS Age

FROM Customers;

v\_CustomerID Loans.CustomerID%TYPE;

v\_Age NUMBER;

BEGIN

-- Open the cursor to loop through each customer

OPEN cur\_Customers;

LOOP

FETCH cur\_Customers INTO v\_CustomerID, v\_Age;

EXIT WHEN cur\_Customers%NOTFOUND;

IF v\_Age > 60 THEN

-- Apply a 1% discount to the interest rate for all loans for this customer

UPDATE Loans

SET InterestRate = InterestRate - 1

WHERE CustomerID = v\_CustomerID;

END IF;

END LOOP;

-- Close the cursor

CLOSE cur\_Customers;

**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.

ALTER TABLE Customers ADD IsVIP VARCHAR2(5);

DECLARE

CURSOR cur\_Customers IS

SELECT CustomerID, Balance

FROM Customers;

v\_CustomerID Customers.CustomerID%TYPE;

v\_Balance Customers.Balance%TYPE;

BEGIN

-- Open the cursor to loop through each customer

OPEN cur\_Customers;

LOOP

FETCH cur\_Customers INTO v\_CustomerID, v\_Balance;

EXIT WHEN cur\_Customers%NOTFOUND;

IF v\_Balance > 10000 THEN

-- Set IsVIP **flag** to TRUE for customers with a balance over $10,000

UPDATE Customers

SET IsVIP = 'TRUE'

WHERE CustomerID = v\_CustomerID;

ELSE

-- Optionally, reset IsVIP to 'FALSE' for balances below $10,000

UPDATE Customers

SET IsVIP = 'FALSE'

WHERE CustomerID = v\_CustomerID;

END IF;

END LOOP;

-- Close the cursor

CLOSE cur\_Customers;

-- Commit the changes

COMMIT;

END;

/

-- Commit the changes

COMMIT;

END;

/

**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.

DECLARE

CURSOR cur\_DueLoans IS

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

FROM Loans l

JOIN Customers c ON l.CustomerID = c.CustomerID

WHERE l.EndDate BETWEEN SYSDATE AND SYSDATE + 30;

v\_LoanID Loans.LoanID%TYPE;

v\_CustomerID Loans.CustomerID%TYPE;

v\_EndDate Loans.EndDate%TYPE;

v\_CustomerName Customers.Name%TYPE;

BEGIN

-- Open the cursor to fetch loans due in the next 30 days

OPEN cur\_DueLoans;

LOOP

FETCH cur\_DueLoans INTO v\_LoanID, v\_CustomerID, v\_EndDate, v\_CustomerName;

EXIT WHEN cur\_DueLoans%NOTFOUND;

-- Print the reminder message for the customer

DBMS\_OUTPUT.PUT\_LINE('Reminder: Dear ' || v\_CustomerName ||

', your loan (Loan ID: ' || v\_LoanID ||

') is due on ' || TO\_CHAR(v\_EndDate, 'YYYY-MM-DD') ||

'. Please ensure timely repayment.');

END LOOP;

-- Close the cursor

CLOSE cur\_DueLoans;

END;

/