

## 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.

```
DECLARE
    CURSOR cur_customers IS
        SELECT CustomerID, TRUNC(MONTHS_BETWEEN(SYSDATE, DOB) / 12) AS Age
        FROM Customers;

    v_customerID Customers.CustomerID%TYPE;
    v_age NUMBER;
BEGIN
    FOR customer_rec IN cur_customers LOOP
        v_customerID := customer_rec.CustomerID;
        v_age := customer_rec.Age;

        IF v_age > 60 THEN
            UPDATE Loans
            SET InterestRate = InterestRate - 1
            WHERE CustomerID = v_customerID;
        END IF;
    END LOOP;

    COMMIT;
END;
```

**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 CHAR(1));
DECLARE
    CURSOR customer_cursor IS
        SELECT CustomerID, Balance
        FROM Customers;

BEGIN
    FOR customer_record IN customer_cursor LOOP
        IF customer_record.Balance > 10000 THEN
            UPDATE Customers
            SET IsVIP = 'Y'
            WHERE CustomerID = customer_record.CustomerID;

            DBMS_OUTPUT.PUT_LINE('Promoted to VIP status for customer ID: ' ||
customer_record.CustomerID);
        ELSE
            UPDATE Customers
            SET IsVIP = 'N'
            WHERE CustomerID = customer_record.CustomerID;
        END IF;
    END LOOP;
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 loan_cursor 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;

BEGIN
    FOR loan_record IN loan_cursor LOOP
        DBMS_OUTPUT.PUT_LINE('Reminder: Loan ID ' || loan_record.LoanID ||
                               ' for customer ' || loan_record.Name ||
                               ' is due on ' || loan_record.EndDate);
    END LOOP;
END;
```