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:

-- Loop through customers and reduce loan interest rate by 1% if age > 60.
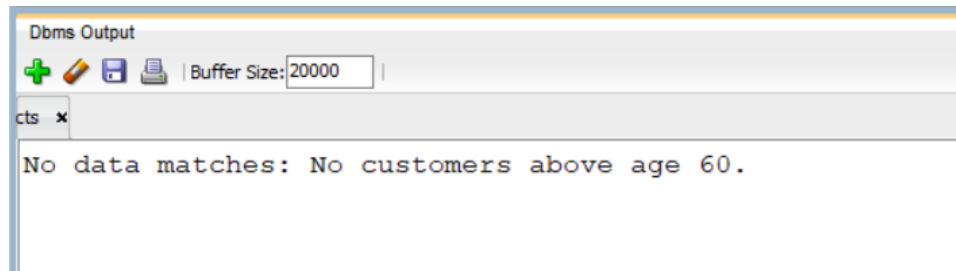
```
DECLARE
   v_match_found BOOLEAN := FALSE;
BEGIN
   FOR cust IN (
      SELECT c.CustomerID, c.Name, FLOOR(MONTHS_BETWEEN(SYSDATE, c.DOB)
/ 12) AS Age
      FROM Customers c
   )
   LOOP
      IF cust.Age > 60 THEN
         UPDATE Loans
         SET InterestRate = InterestRate - 1
         WHERE CustomerID = cust.CustomerID;

         DBMS_OUTPUT.PUT_LINE('Updated interest rate for: ' || cust.Name || ' (Age: ' ||
cust.Age || ')');
         v_match_found := TRUE;
      END IF;
   END LOOP;

   IF NOT v_match_found THEN
      DBMS_OUTPUT.PUT_LINE('No data matches: No customers above age 60.');
   END IF;

   COMMIT;
END;
/
```

```
Dbms Output
+  ✎  💾  🖨  | Buffer Size: 20000  |
cts  ✕
No data matches: No customers above age 60.
```

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.

Solution:

-- Promote customers to VIP if their balance is over $10,000

ALTER TABLE Customers ADD (IsVIP VARCHAR2(5));

```
DECLARE
   v_vip_found BOOLEAN := FALSE;
BEGIN
   FOR cust IN (
      SELECT CustomerID, Name, Balance
      FROM Customers
   )
   LOOP
      IF cust.Balance > 10000 THEN
         UPDATE Customers
         SET IsVIP = 'TRUE'
         WHERE CustomerID = cust.CustomerID;

         DBMS_OUTPUT.PUT_LINE('Promoted to VIP: ' || cust.Name || ' (Balance: ' ||
cust.Balance || ')');
         v_vip_found := TRUE;
      ELSE
         UPDATE Customers
         SET IsVIP = 'FALSE'
         WHERE CustomerID = cust.CustomerID;
      END IF;
   END LOOP;

   IF NOT v_vip_found THEN
      DBMS_OUTPUT.PUT_LINE('No customers qualified for VIP status.');
```
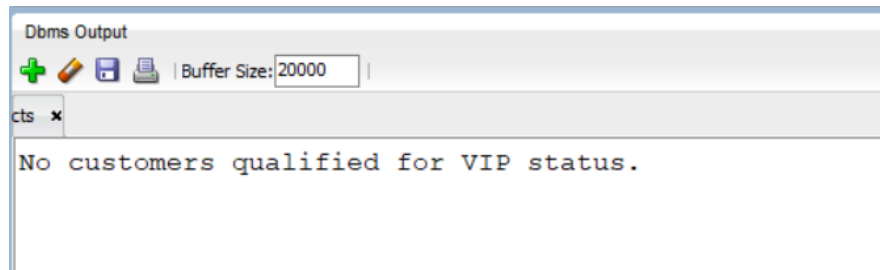
```
    END IF;

    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.

Solution:

-- Send reminders to customers whose loans are due within the next 30 days

```
DECLARE
    v_due_found BOOLEAN := FALSE;  -- Flag to check if any loan is due soon
BEGIN
    FOR loan_rec IN (
        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
    )
    LOOP
        DBMS_OUTPUT.PUT_LINE('Reminder: Loan ID ' || loan_rec.LoanID ||
                    ' for customer ' || loan_rec.Name ||
                    ' is due on ' || TO_CHAR(loan_rec.EndDate, 'DD-MON-YYYY'));
        v_due_found := TRUE;
    END LOOP;

    IF NOT v_due_found THEN
        DBMS_OUTPUT.PUT_LINE('No loans are due within the next 30 days.');
    END IF;
END;
```

/

Dbms Output

Buffer Size: 20000

cts ✕

No loans are due within the next 30 days.