**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.

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.

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.

Here are the PL/SQL blocks for the given scenarios:

**Scenario 1: Apply a Discount to Loan Interest Rates for Customers Above 60 Years Old**

DECLARE

CURSOR c\_customers IS

SELECT customer\_id, age, loan\_interest\_rate

FROM customers

WHERE age > 60;

v\_customer\_id customers.customer\_id%TYPE;

v\_loan\_interest\_rate customers.loan\_interest\_rate%TYPE;

BEGIN

OPEN c\_customers;

LOOP

FETCH c\_customers INTO v\_customer\_id, v\_loan\_interest\_rate;

EXIT WHEN c\_customers%NOTFOUND;

-- Apply a 1% discount to the loan interest rate

v\_loan\_interest\_rate := v\_loan\_interest\_rate - 1;

UPDATE customers

SET loan\_interest\_rate = v\_loan\_interest\_rate

WHERE customer\_id = v\_customer\_id;

END LOOP;

CLOSE c\_customers;

DBMS\_OUTPUT.PUT\_LINE('Interest rates updated for customers above 60 years old.');

END;

/

**Scenario 2: Set VIP Status for Customers with a Balance Over $10,000**

DECLARE

CURSOR c\_customers IS

SELECT customer\_id, balance

FROM customers

WHERE balance > 10000;

v\_customer\_id customers.customer\_id%TYPE;

BEGIN

OPEN c\_customers;

LOOP

FETCH c\_customers INTO v\_customer\_id;

EXIT WHEN c\_customers%NOTFOUND;

-- Set IsVIP flag to TRUE

UPDATE customers

SET IsVIP = TRUE

WHERE customer\_id = v\_customer\_id;

END LOOP;

CLOSE c\_customers;

DBMS\_OUTPUT.PUT\_LINE('VIP status updated for customers with balance over $10,000.');

END;

/

**Scenario 3: Send Reminders to Customers Whose Loans are Due Within the Next 30 Days**

DECLARE

CURSOR c\_loans IS

SELECT customer\_id, loan\_due\_date

FROM loans

WHERE loan\_due\_date BETWEEN SYSDATE AND SYSDATE + 30;

v\_customer\_id loans.customer\_id%TYPE;

v\_loan\_due\_date loans.loan\_due\_date%TYPE;

BEGIN

OPEN c\_loans;

LOOP

FETCH c\_loans INTO v\_customer\_id, v\_loan\_due\_date;

EXIT WHEN c\_loans%NOTFOUND;

-- Print a reminder message

DBMS\_OUTPUT.PUT\_LINE('Reminder: Customer ' || v\_customer\_id || ', your loan is due on ' || TO\_CHAR(v\_loan\_due\_date, 'DD-MON-YYYY'));

END LOOP;

CLOSE c\_loans;

DBMS\_OUTPUT.PUT\_LINE('Reminders sent for loans due within the next 30 days.');

END;

/

Each of these PL/SQL blocks accomplishes the task described in the respective scenario by iterating through the relevant data, applying conditions, and making the necessary updates or prints.