**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.

**PL/SQL Query:**

BEGIN

  FOR cust\_rec IN (

    SELECT CustomerID, Age, InterestRate

    FROM Customers

    WHERE Age > 60

  ) LOOP

    UPDATE Customers

    SET InterestRate = InterestRate - 1

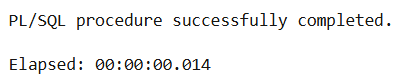
    WHERE CustomerID = cust\_rec.CustomerID;

  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.

**PL/SQL Query:**

BEGIN

FOR cust\_rec IN (

SELECT CustomerID, Balance

FROM Customers

WHERE Balance > 10000

) LOOP

UPDATE Customers

SET IsVIP = 'TRUE'

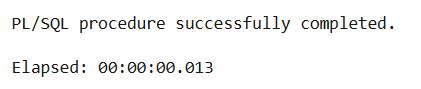
WHERE CustomerID = cust\_rec.CustomerID;

END LOOP;

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.

**PL/SQL Query:**

BEGIN

  FOR loan\_rec IN (

    SELECT L.LoanID, L.DueDate, C.Name

    FROM Loans L

    JOIN Customers C ON L.CustomerID = C.CustomerID

    WHERE L.DueDate <= 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.DueDate, 'DD-MON-YYYY'));

  END LOOP;

END;

/

