**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.

**Ans**:

DECLARE

cust\_id NUMBER;

dob DATE;

age NUMBER;

loan\_id NUMBER;

interest\_rate NUMBER;

BEGIN

FOR r IN (SELECT customerid, dob FROM customers) LOOP

cust\_id := r.customerid;

dob := r.dob;

age := TRUNC(MONTHS\_BETWEEN(SYSDATE, dob) / 12);

IF age > 60 THEN

FOR loan\_rec IN (SELECT loanid, interestrate FROM loans WHERE customerid = cust\_id) LOOP

loan\_id := loan\_rec.loanid;

interest\_rate := loan\_rec.interestrate;

UPDATE loans

SET interestrate = interest\_rate - 1

WHERE loanid = loan\_id;

END LOOP;

END IF;

END LOOP;

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.

**Ans:**

--First adding a column name as “**isvip**” in “**customers**” Table

BEGIN

   EXECUTE IMMEDIATE '

      ALTER TABLE customers

      ADD (isvip VARCHAR2(5))';

EXCEPTION

   WHEN OTHERS THEN

      DBMS\_OUTPUT.PUT\_LINE('An error occurred: ' || SQLERRM);

END;

/

DECLARE

cust\_id NUMBER;

balance NUMBER;

BEGIN

FOR rec IN (SELECT customerid, balance FROM customers) LOOP

cust\_id := rec.customerid;

balance := rec.balance;

IF balance > 10000 THEN

UPDATE customers

SET isvip = 'TRUE'

WHERE customerid = cust\_id;

DBMS\_OUTPUT.PUT\_LINE('Customer ' || cust\_id || ' is now a VIP');

ELSE

UPDATE customers

SET isvip = 'FALSE'

WHERE customerid = cust\_id;

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.

**Ans**:

DECLARE

loan\_id NUMBER;

cust\_id NUMBER;

due\_date DATE;

BEGIN

FOR rec IN (SELECT loanid, customerid, enddate FROM loans WHERE enddate <= SYSDATE + 30) LOOP

loan\_id := rec.loanid;

cust\_id := rec.customerid;

due\_date := rec.enddate;

DBMS\_OUTPUT.PUT\_LINE('Reminder: Loan ' || loan\_id || ' for customer ' || cust\_id || ' is due on ' || due\_date);

END LOOP;

END;

/