**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.**

**CODE:**

BEGIN

FOR rec IN (

SELECT loanid, interestrate, customerid

FROM loans

JOIN customers ON loans.customerid = customers.customer\_id

WHERE MONTHS\_BETWEEN(SYSDATE, dob) / 12 > 60

) LOOP

UPDATE loans

SET interestrate = interestrate - 1

WHERE loanid = rec.loanid;

DBMS\_OUTPUT.PUT\_LINE('Discount applied for Customer ID: ' || rec.customerid);

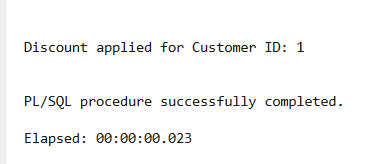
END LOOP;

COMMIT;

END;

/

**OUTPUT:**

****

**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.**

**CODE:**

BEGIN

FOR acc IN (SELECT accountid, balance FROM accounts) LOOP

IF acc.balance >= 1000 THEN

DBMS\_OUTPUT.PUT\_LINE('Account ' || acc.accountid || ' is Premium');

ELSIF acc.balance >= 500 THEN

DBMS\_OUTPUT.PUT\_LINE('Account ' || acc.accountid || ' is Standard');

ELSE

DBMS\_OUTPUT.PUT\_LINE('Account ' || acc.accountid || ' is Basic');

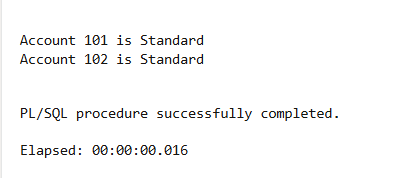
END IF;

END LOOP;

END;

/

**OUTPUT:**

****

**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.**

**CODE:**

BEGIN

FOR loan\_rec IN (

SELECT loanid, amount, duedate

FROM loans

WHERE duedate < SYSDATE

) LOOP

UPDATE loans

SET amount = loan\_rec.amount + 500

WHERE loanid = loan\_rec.loanid;

DBMS\_OUTPUT.PUT\_LINE('Late fee applied to Loan ID: ' || loan\_rec.loanid);

END LOOP;

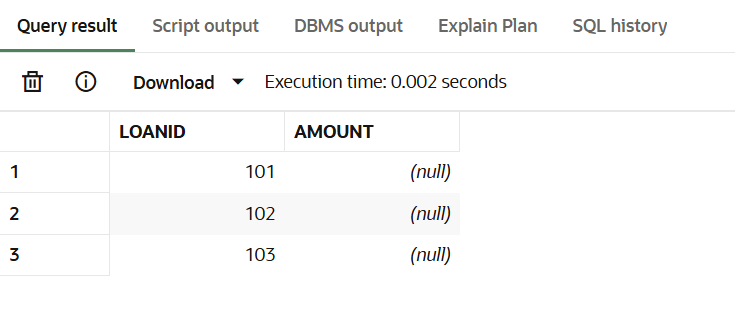
COMMIT;

END;

/

**OUTPUT:**

**Late fee applied to Loan ID: 1**

****