**Exercise 1:Control Statements**

**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.

**Procedure:**

set serveroutput on;

DECLARE

CURSOR customer\_cursor IS

SELECT CustomerID, DOB

FROM Customers;

v\_customer\_id Customers.CustomerID%TYPE;

v\_dob Customers.DOB%TYPE;

v\_age NUMBER;

v\_current\_rate Loans.InterestRate%TYPE;

BEGIN

FOR customer\_record IN customer\_cursor LOOP

v\_customer\_id := customer\_record.CustomerID;

v\_dob := customer\_record.DOB;

v\_age := FLOOR(MONTHS\_BETWEEN(SYSDATE, v\_dob) / 12);

IF v\_age > 60 THEN

FOR loan\_record IN (SELECT LoanID, InterestRate

FROM Loans

WHERE CustomerID = v\_customer\_id) LOOP

v\_current\_rate := loan\_record.InterestRate;

UPDATE Loans

SET InterestRate = GREATEST(v\_current\_rate - 1, 0) WHERE LoanID = loan\_record.LoanID;

END LOOP;

END IF;

END LOOP;

COMMIT;

END;/

**Before executing procedure:**

| LoanID | CustomerID | LoanAmount | InterestRate | StartDate | EndDate |

| 1 | 1 | 5000 | 5.0 | 06-AUG-24 | 06-AUG-29 |

| 2 | 2 | 6000 | 4.5 | 06-AUG-24 | 06-AUG-28 |

| 3 | 3 | 7000 | 5.2 | 06-AUG-24 | 06-AUG-27 |

| 4 | 4 | 8000 | 4.8 | 06-AUG-24 | 06-AUG-29 |

| 5 | 5 | 9000 | 5.0 | 06-AUG-24 | 06-AUG-28 |

| 6 | 6 | 10000 | 5.5 | 06-AUG-24 | 06-AUG-27 |

| 7 | 7 | 11000 | 4.7 | 06-AUG-24 | 06-AUG-29 |

| 8 | 8 | 12000 | 5.1 | 06-AUG-24 | 06-AUG-28 |

| 9 | 9 | 13000 | 4.9 | 06-AUG-24 | 06-AUG-27 |

| 10 | 10 | 14000 | 5.3 | 06-AUG-24 | 06-AUG-29 |

**After executing procedure:**

| LoanID | CustomerID | LoanAmount | InterestRate | StartDate | EndDate |

1 1 5000 4 06-AUG-24 06-AUG-29

2 2 6000 3.5 06-AUG-24 06-AUG-28

3 3 7000 4.2 06-AUG-24 06-AUG-27

4 4 8000 4.8 06-AUG-24 06-AUG-29

5 5 9000 4 06-AUG-24 06-AUG-28

6 6 10000 4.5 06-AUG-24 06-AUG-27

7 7 11000 3.7 06-AUG-24 06-AUG-29

8 8 12000 5.1 06-AUG-24 06-AUG-28

9 9 13000 4.9 06-AUG-24 06-AUG-27

10 10 14000 5.3 06-AUG-24 06-AUG-29

**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.

**PROCEDURE:**

DECLARE

CURSOR customer\_cursor IS

SELECT CustomerID, Balance

FROM Customers

v\_customer\_id Customers.CustomerID%TYPE;

v\_balance Customers.Balance%TYPE;

BEGIN

FOR customer\_record IN customer\_cursor LOOP

v\_customer\_id := customer\_record.CustomerID;

v\_balance := customer\_record.Balance;

IF v\_balance > 10000 THEN

UPDATE Customers

SET IsVIP = 'Y'

WHERE CustomerID = v\_customer\_id;

ELSE

UPDATE Customers

SET IsVIP = 'N'

WHERE CustomerID = v\_customer\_id;

END IF;

END LOOP;

COMMIT;

END;

/

**Before Procedure:**

| CustomerId | Name | DOB | Balance | LastModified | IsVip |

1 John Doe 15-MAY-20 1000 06-AUG-24

2 Jane Smith 20-JUL-35 1500 06-AUG-24

3 Michael Johnson 30-JAN-62 2000 06-AUG-24

4 Emily Davis 25-MAR-95 25000 06-AUG-24

5 William Brown 10-SEP-55 3000 06-AUG-24

6 Sophia Wilson 05-DEC-50 35000 06-AUG-24

7 James Miller 14-APR-42 400089 06-AUG-24

8 Olivia Taylor 22-JUN-80 4500 06-AUG-24

9 Liam Anderson 30-AUG-69 50000 06-AUG-24

10 Ava Thomas 11-NOV-91 5500 06-AUG-24

**After Procedure:**

| CustomerId | Name | DOB | Balance | LastModified | IsVip |

1 John Doe 15-MAY-20 1000 06-AUG-24 N

2 Jane Smith 20-JUL-35 1500 06-AUG-24 N

3 Michael Johnson 30-JAN-62 2000 06-AUG-24 N

4 Emily Davis 25-MAR-95 25000 06-AUG-24 Y

5 William Brown 10-SEP-55 3000 06-AUG-24 N

6 Sophia Wilson 05-DEC-50 35000 06-AUG-24 Y

7 James Miller 14-APR-42 400089 06-AUG-24 Y

8 Olivia Taylor 22-JUN-80 4500 06-AUG-24 N

9 Liam Anderson 30-AUG-69 50000 06-AUG-24 Y

10 Ava Thomas 11-NOV-91 5500 06-AUG-24 N

**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.

**PROCEDURE:**

DECLARE

CURSOR loan\_cursor IS

SELECT l.LoanID, l.CustomerID, l.LoanAmount, l.InterestRate, l.StartDate, l.EndDate, c.Name

FROM Loans l

JOIN Customers c ON l.CustomerID = c.CustomerID

WHERE l.EndDate BETWEEN SYSDATE AND SYSDATE + 30;

v\_loan\_id Loans.LoanID%TYPE;

v\_customer\_id Loans.CustomerID%TYPE;

v\_loan\_amount Loans.LoanAmount%TYPE;

v\_interest\_rate Loans.InterestRate%TYPE;

v\_end\_date Loans.EndDate%TYPE;

v\_customer\_name Customers.Name%TYPE;

BEGIN

FOR loan\_record IN loan\_cursor LOOP

-- Fetch details

v\_loan\_id := loan\_record.LoanID;

v\_customer\_id := loan\_record.CustomerID;

v\_loan\_amount := loan\_record.LoanAmount;

v\_interest\_rate := loan\_record.InterestRate;

v\_end\_date := loan\_record.EndDate;

v\_customer\_name := loan\_record.Name;

-- Print reminder message

DBMS\_OUTPUT.PUT\_LINE('Reminder: Loan with ID ' || v\_loan\_id ||

' for customer ' || v\_customer\_name ||

' (CustomerID: ' || v\_customer\_id ||

') is due on ' || TO\_CHAR(v\_end\_date, 'DD-MON-YYYY') ||

'. The loan amount is ' || v\_loan\_amount ||

' with an interest rate of ' || v\_interest\_rate || '.');

END LOOP;

END;

/

**OUTPUT:**

PL/SQL procedure successfully completed.

Reminder: Loan with ID 6 for customer Sophia Wilson (CustomerID: 6) is due on 01-SEP-2024. The loan amount is 10000 with an interest rate of 5.5.

Reminder: Loan with ID 7 for customer James Miller (CustomerID: 7) is due on 01-SEP-2024. The loan amount is 11000 with an interest rate of 4.7.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |