**WEEK-2**

**MODULE -3**

**PL/SQL PROGRAMMING**

**SUPERSET ID:6407550**

**Exercise:6-Cursors**

-- Insert some transactions for this month

INSERT INTO Transactions (TransactionID, AccountID, TransactionDate, Amount, TransactionType)

VALUES (20, 1, SYSDATE, 200, 'Deposit');

INSERT INTO Transactions (TransactionID, AccountID, TransactionDate, Amount, TransactionType)

VALUES (21, 2, SYSDATE, 150, 'Withdrawal');

COMMIT;

-- Add new loans to see change

INSERT INTO Loans (LoanID, CustomerID, LoanAmount, InterestRate, StartDate, EndDate)

VALUES (5, 1, 60000, 6.0, SYSDATE, ADD\_MONTHS(SYSDATE, 60));

INSERT INTO Loans (LoanID, CustomerID, LoanAmount, InterestRate, StartDate, EndDate)

VALUES (6, 2, 30000, 5.5, SYSDATE, ADD\_MONTHS(SYSDATE, 36));

COMMIT;

**Scenario:1**

SET SERVEROUTPUT ON;

DECLARE

CURSOR trans\_cursor IS

SELECT t.TransactionID, c.Name, t.TransactionDate, t.Amount, t.TransactionType

FROM Transactions t

JOIN Accounts a ON t.AccountID = a.AccountID

JOIN Customers c ON a.CustomerID = c.CustomerID

WHERE EXTRACT(MONTH FROM t.TransactionDate) = EXTRACT(MONTH FROM SYSDATE)

AND EXTRACT(YEAR FROM t.TransactionDate) = EXTRACT(YEAR FROM SYSDATE);

rec trans\_cursor%ROWTYPE;

BEGIN

OPEN trans\_cursor;

LOOP

FETCH trans\_cursor INTO rec;

EXIT WHEN trans\_cursor%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('🧾 Customer: ' || rec.Name ||

' | Transaction ID: ' || rec.TransactionID ||

' | Type: ' || rec.TransactionType ||

' | Amount: ₹' || rec.Amount ||

' | Date: ' || TO\_CHAR(rec.TransactionDate, 'YYYY-MM-DD'));

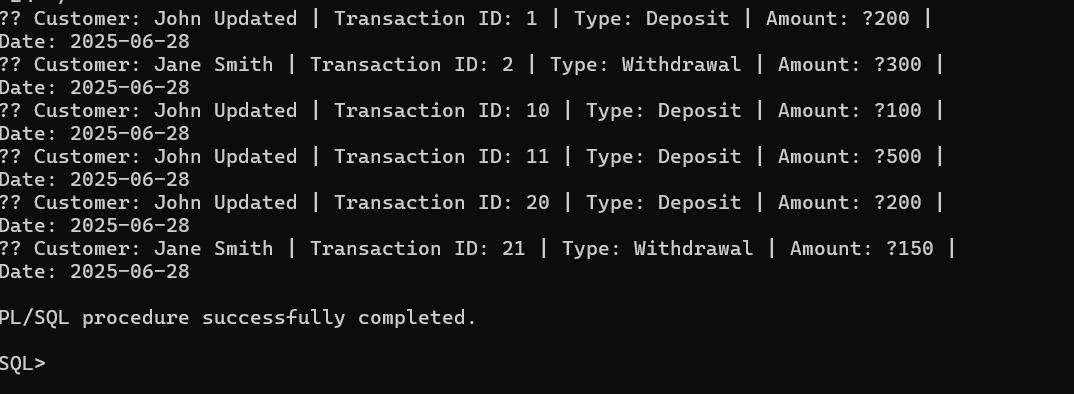
END LOOP;

CLOSE trans\_cursor;

END;

/

**OUTPUT:**



**Scenario-2**

SET SERVEROUTPUT ON;

DECLARE

CURSOR acc\_cursor IS

SELECT AccountID, Balance FROM Accounts;

rec acc\_cursor%ROWTYPE;

v\_fee CONSTANT NUMBER := 100;

BEGIN

OPEN acc\_cursor;

LOOP

FETCH acc\_cursor INTO rec;

EXIT WHEN acc\_cursor%NOTFOUND;

IF rec.Balance >= v\_fee THEN

UPDATE Accounts

SET Balance = Balance - v\_fee,

LastModified = SYSDATE

WHERE AccountID = rec.AccountID;

DBMS\_OUTPUT.PUT\_LINE('✅ Annual fee of ₹' || v\_fee || ' deducted from Account ID: ' || rec.AccountID);

ELSE

DBMS\_OUTPUT.PUT\_LINE('⚠️ Insufficient balance for Account ID: ' || rec.AccountID);

END IF;

END LOOP;

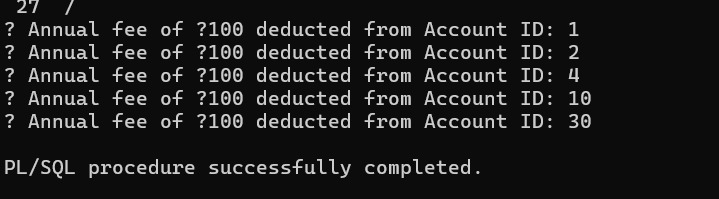
CLOSE acc\_cursor;

COMMIT;

END;

/

**OUTPUT:**

****

**Scenario:3**

SET SERVEROUTPUT ON;

DECLARE

CURSOR loan\_cursor IS

SELECT LoanID, LoanAmount, InterestRate FROM Loans;

rec loan\_cursor%ROWTYPE;

v\_new\_rate NUMBER;

BEGIN

OPEN loan\_cursor;

LOOP

FETCH loan\_cursor INTO rec;

EXIT WHEN loan\_cursor%NOTFOUND;

IF rec.LoanAmount > 50000 THEN

v\_new\_rate := rec.InterestRate + 0.5;

ELSE

v\_new\_rate := rec.InterestRate + 0.2;

END IF;

UPDATE Loans

SET InterestRate = v\_new\_rate

WHERE LoanID = rec.LoanID;

DBMS\_OUTPUT.PUT\_LINE('🔄 Loan ID: ' || rec.LoanID ||

' | New Interest Rate: ' || v\_new\_rate || '%');

END LOOP;

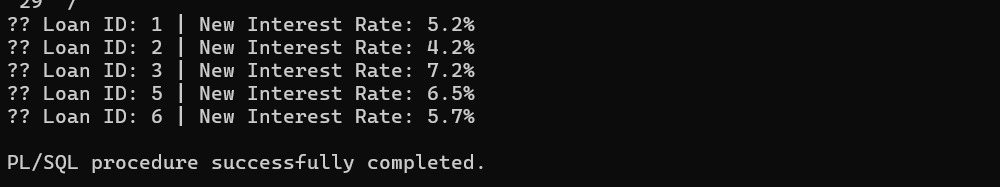
CLOSE loan\_cursor;

COMMIT;

END;

/

**OUTPUT:**

****