**Exercise 6: Cursors**

**Scenario 1: Generate monthly statements for all customers.**

Question: Write a PL/SQL block using an explicit cursor GenerateMonthlyStatements that retrieves all transactions for the current month and prints a statement for each customer**.**

**Scenario 2: Apply annual fee to all accounts.**

Question: Write a PL/SQL block using an explicit cursor ApplyAnnualFee that deducts an annual maintenance fee from the balance of all accounts

**Scenario 3: Update the interest rate for all loans based on a new policy.**

Question: Write a PL/SQL block using an explicit cursor UpdateLoanInterestRates that fetches all loans and updates their interest rates based on the new policy.

**Scenario 1:**

DECLARE

CURSOR cur\_monthly\_statements IS

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

FROM Customers c

JOIN Accounts a ON c.CustomerID = a.CustomerID

JOIN Transactions t ON a.AccountID = t.AccountID

WHERE t.TransactionDate BETWEEN TRUNC(CURRENT\_DATE, 'MM') AND LAST\_DAY(CURRENT\_DATE);

v\_customerID Customers.CustomerID%TYPE;

v\_name Customers.Name%TYPE;

v\_transactionDate Transactions.TransactionDate%TYPE;

v\_amount Transactions.Amount%TYPE;

v\_transactionType Transactions.TransactionType%TYPE;

BEGIN

OPEN cur\_monthly\_statements;

LOOP

FETCH cur\_monthly\_statements INTO v\_customerID, v\_name, v\_transactionDate, v\_amount, v\_transactionType;

EXIT WHEN cur\_monthly\_statements%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Customer ID: ' || v\_customerID || ' - ' || v\_name);

DBMS\_OUTPUT.PUT\_LINE('Transaction Date: ' || v\_transactionDate || ', Amount: ' || v\_amount || ', Type: ' || v\_transactionType);

DBMS\_OUTPUT.PUT\_LINE('-----------------------------------');

END LOOP;

CLOSE cur\_monthly\_statements;

END;

/

**Scenario 2:**

DECLARE

CURSOR cur\_annual\_fees IS

SELECT AccountID, Balance

FROM Accounts;

v\_accountID Accounts.AccountID%TYPE;

v\_balance Accounts.Balance%TYPE;

v\_annualFee CONSTANT NUMBER := 100.00;

BEGIN

OPEN cur\_annual\_fees;

LOOP

FETCH cur\_annual\_fees INTO v\_accountID, v\_balance;

EXIT WHEN cur\_annual\_fees%NOTFOUND;

UPDATE Accounts

SET Balance = v\_balance - v\_annualFee

WHERE AccountID = v\_accountID;

DBMS\_OUTPUT.PUT\_LINE('Account ID: ' || v\_accountID || ' - New Balance: ' || (v\_balance - v\_annualFee));

END LOOP;

CLOSE cur\_annual\_fees;

END;

/

**Scenario 3:**

DECLARE

CURSOR cur\_interest\_update IS

SELECT LoanID, InterestRate

FROM Loans;

v\_loanID Loans.LoanID%TYPE;

v\_interestRate Loans.InterestRate%TYPE;

v\_newInterestRate CONSTANT NUMBER := 5.00;

BEGIN

OPEN cur\_interest\_update;

LOOP

FETCH cur\_interest\_update INTO v\_loanID, v\_interestRate;

EXIT WHEN cur\_interest\_update%NOTFOUND;

UPDATE Loans

SET InterestRate = v\_newInterestRate

WHERE LoanID = v\_loanID;

DBMS\_OUTPUT.PUT\_LINE('Loan ID: ' || v\_loanID || ' - New Interest Rate: ' || v\_newInterestRate);

END LOOP;

CLOSE cur\_interest\_update;

END;

/