**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.

**Ans:**

DECLARE

CURSOR c\_tran IS

SELECT t.TransactionID, t.AccountID, t.TransactionDate, t.Amount, t.TransactionType, a.CustomerID, c.Name

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);

tID Transactions.TransactionID%TYPE;

accID Transactions.AccountID%TYPE;

tDate Transactions.TransactionDate%TYPE;

amt Transactions.Amount%TYPE;

tType Transactions.TransactionType%TYPE;

custID Accounts.CustomerID%TYPE;

custname Customers.Name%TYPE;

BEGIN

FOR r\_transaction IN c\_tran LOOP

tID := r\_transaction.TransactionID;

accID := r\_transaction.AccountID;

tDate := r\_transaction.TransactionDate;

amt := r\_transaction.Amount;

tType := r\_transaction.TransactionType;

custID := r\_transaction.CustomerID;

custname := r\_transaction.Name;

DBMS\_OUTPUT.PUT\_LINE('Statement for Customer: ' || custname);

DBMS\_OUTPUT.PUT\_LINE('Account ID: ' || accID);

DBMS\_OUTPUT.PUT\_LINE('Transaction ID: ' || tID);

DBMS\_OUTPUT.PUT\_LINE('Date: ' || tDate);

DBMS\_OUTPUT.PUT\_LINE('Amount: ' || amt);

DBMS\_OUTPUT.PUT\_LINE('Type: ' || tType);

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

END LOOP;

END;

/

**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.

**Ans:**

DECLARE

CURSOR c\_acc IS

SELECT AccountID, Balance

FROM Accounts;

accID Accounts.AccountID%TYPE;

bal Accounts.Balance%TYPE;

fee NUMBER := 42;

BEGIN

FOR ac IN c\_acc LOOP

accID := ac.AccountID;

bal := ac.Balance;

UPDATE Accounts

SET Balance = bal - fee

WHERE AccountID = accID;

DBMS\_OUTPUT.PUT\_LINE('Annual fee applied to Account ID: ' || accID);

END LOOP;

COMMIT;

END;

/

**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.

**Ans:**

DECLARE

CURSOR c\_loans IS

SELECT LoanID, InterestRate

FROM Loans;

cust\_loanID Loans.LoanID%TYPE;

currentInterestRate Loans.InterestRate%TYPE;

newInterestRate Loans.InterestRate%TYPE;

BEGIN

newInterestRate := 10;

FOR r\_loan IN c\_loans LOOP

cust\_loanID := r\_loan.LoanID;

currentInterestRate := r\_loan.InterestRate;

UPDATE Loans

SET InterestRate = newInterestRate

WHERE LoanID = cust\_loanID;

DBMS\_OUTPUT.PUT\_LINE('Updated Interest Rate for Loan ID: ' || cust\_loanID || ' to ' || newInterestRate || '%');

END LOOP;

COMMIT;

END;

/