**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.

**Query:**

DECLARE

CURSOR transaction\_cursor IS

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

FROM Transactions t

JOIN Accounts a ON t.AccountID = a.AccountID

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

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

transaction\_record transaction\_cursor%ROWTYPE;

current\_customer\_id Accounts.CustomerID%TYPE := NULL;

BEGIN

OPEN transaction\_cursor;

FETCH transaction\_cursor INTO transaction\_record;

WHILE transaction\_cursor%FOUND LOOP

IF current\_customer\_id IS NULL OR current\_customer\_id != transaction\_record.CustomerID THEN

IF current\_customer\_id IS NOT NULL THEN

DBMS\_OUTPUT.PUT\_LINE('End of statement for Customer ID: ' || current\_customer\_id);

END IF;

DBMS\_OUTPUT.PUT\_LINE('Statement for Customer ID: ' || transaction\_record.CustomerID);

current\_customer\_id := transaction\_record.CustomerID;

END IF;

DBMS\_OUTPUT.PUT\_LINE('Transaction ID: ' || transaction\_record.TransactionID);

DBMS\_OUTPUT.PUT\_LINE('Account ID: ' || transaction\_record.AccountID);

DBMS\_OUTPUT.PUT\_LINE('Date: ' || TO\_CHAR(transaction\_record.TransactionDate, 'DD-MON-YYYY'));

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

DBMS\_OUTPUT.PUT\_LINE('Type: ' || transaction\_record.TransactionType);

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

FETCH transaction\_cursor INTO transaction\_record;

END LOOP;

CLOSE transaction\_cursor;

DBMS\_OUTPUT.PUT\_LINE('End of statement for Customer ID: ' || current\_customer\_id);

EXCEPTION

WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE('An error occurred: ' || SQLERRM);

IF transaction\_cursor%ISOPEN THEN

CLOSE transaction\_cursor;

END IF;

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.

**Query:**

DECLARE

CURSOR account\_cursor IS

SELECT AccountID, Balance

FROM Accounts;

account\_record account\_cursor%ROWTYPE;

annual\_fee NUMBER := 50;

BEGIN

OPEN account\_cursor;

FETCH account\_cursor INTO account\_record;

WHILE account\_cursor%FOUND LOOP

]

UPDATE Accounts

SET Balance = Balance - annual\_fee

WHERE AccountID = account\_record.AccountID;

DBMS\_OUTPUT.PUT\_LINE('Applied annual fee to Account ID: ' || account\_record.AccountID);

DBMS\_OUTPUT.PUT\_LINE('New Balance: ' || (account\_record.Balance - annual\_fee));

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

FETCH account\_cursor INTO account\_record;

END LOOP;

CLOSE account\_cursor;

EXCEPTION

WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE('An error occurred: ' || SQLERRM);

IF account\_cursor%ISOPEN THEN

CLOSE account\_cursor;

END IF;

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.

**Query:**

DECLARE

CURSOR loan\_cursor IS

SELECT LoanID, InterestRate

FROM Loans;

loan\_record loan\_cursor%ROWTYPE;

new\_interest\_rate NUMBER := 6;

BEGIN

OPEN loan\_cursor;

FETCH loan\_cursor INTO loan\_record;

WHILE loan\_cursor%FOUND LOOP

UPDATE Loans

SET InterestRate = new\_interest\_rate

WHERE LoanID = loan\_record.LoanID;

DBMS\_OUTPUT.PUT\_LINE('Updated interest rate for Loan ID: ' || loan\_record.LoanID);

DBMS\_OUTPUT.PUT\_LINE('Old Interest Rate: ' || loan\_record.InterestRate);

DBMS\_OUTPUT.PUT\_LINE('New Interest Rate: ' || new\_interest\_rate);

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

FETCH loan\_cursor INTO loan\_record;

END LOOP;

CLOSE loan\_cursor;

EXCEPTION

WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE('An error occurred: ' || SQLERRM);

IF loan\_cursor%ISOPEN THEN

CLOSE loan\_cursor;

END IF;

**END;**