**Cursors**

**Scenario 1: Generate Monthly Statements**

**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.

DECLARE

CURSOR transaction\_cursor 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 TRUNC(t.TransactionDate, 'MM') = TRUNC(SYSDATE, 'MM');

v\_customerID Customers.CustomerID%TYPE;

v\_customerName Customers.Name%TYPE;

v\_transactionDate Transactions.TransactionDate%TYPE;

v\_amount Transactions.Amount%TYPE;

v\_transactionType Transactions.TransactionType%TYPE;

BEGIN

OPEN transaction\_cursor;

LOOP

FETCH transaction\_cursor INTO v\_customerID, v\_customerName, v\_transactionDate, v\_amount, v\_transactionType;

EXIT WHEN transaction\_cursor%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Customer: ' || v\_customerName || ' | Date: ' || TO\_CHAR(v\_transactionDate, 'YYYY-MM-DD') || ' | Amount: ' || v\_amount || ' | Type: ' || v\_transactionType);

END LOOP;

CLOSE transaction\_cursor;

END;

**Scenario 2: Apply Annual Fee to Accounts**

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

CURSOR account\_cursor IS

SELECT AccountID, Balance

FROM Accounts;

v\_accountID Accounts.AccountID%TYPE;

v\_balance Accounts.Balance%TYPE;

v\_annualFee CONSTANT NUMBER := 50; -- Example annual fee amount

BEGIN

OPEN account\_cursor;

LOOP

FETCH account\_cursor INTO v\_accountID, v\_balance;

EXIT WHEN account\_cursor%NOTFOUND;

UPDATE Accounts

SET Balance = Balance - v\_annualFee

WHERE AccountID = v\_accountID;

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

END LOOP;

CLOSE account\_cursor;

COMMIT;

END;  
  
**Scenario 3: Update Loan Interest Rates**

**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.

DECLARE

CURSOR loan\_cursor IS

SELECT LoanID, InterestRate

FROM Loans;

v\_loanID Loans.LoanID%TYPE;

v\_interestRate Loans.InterestRate%TYPE;

v\_newInterestRate CONSTANT NUMBER := 4.5; -- New interest rate as per policy

BEGIN

OPEN loan\_cursor;

LOOP

FETCH loan\_cursor INTO v\_loanID, v\_interestRate;

EXIT WHEN loan\_cursor%NOTFOUND;

UPDATE Loans

SET InterestRate = v\_newInterestRate

WHERE LoanID = v\_loanID;

DBMS\_OUTPUT.PUT\_LINE('Interest rate updated for Loan ID: ' || v\_loanID);

END LOOP;

CLOSE loan\_cursor;

COMMIT;

END;