**Exercise 6: Cursors**

**Scenario 1:**

**CODE:**

***-- Creating Customers Table***

CREATE TABLE Customers (

CustomerID NUMBER PRIMARY KEY,

Name VARCHAR2(100),

DOB DATE,

Balance NUMBER,

LastModified DATE

);

***-- Inserting Values into Customers Table***

BEGIN

INSERT INTO Customers VALUES (1, 'Alice Reddy', TO\_DATE('1950-05-10', 'YYYY-MM-DD'), 12000, SYSDATE);

INSERT INTO Customers VALUES (2, 'Bob Singh', TO\_DATE('1990-08-21', 'YYYY-MM-DD'), 8000, SYSDATE);

INSERT INTO Customers VALUES (3, 'Carol Mehta', TO\_DATE('1962-01-15', 'YYYY-MM-DD'), 15000, SYSDATE );

INSERT INTO Customers VALUES (4, 'David Rao', TO\_DATE('1985-12-01', 'YYYY-MM-DD'), 9500, SYSDATE);

INSERT INTO Customers VALUES (5, 'Esha Iyer', TO\_DATE('1945-03-25', 'YYYY-MM-DD'), 20000, SYSDATE);

END;

***-- Creating Accounts table***

CREATE TABLE Accounts (

AccountID NUMBER PRIMARY KEY,

CustomerID NUMBER,

AccountType VARCHAR2(20),

Balance NUMBER,

LastModified DATE,

FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)

);

***-- Inserting Values into Accounts table***

BEGIN

INSERT INTO Accounts (AccountID, CustomerID, AccountType, Balance, LastModified) VALUES (1, 1, 'Savings', 1200, SYSDATE);

INSERT INTO Accounts (AccountID, CustomerID, AccountType, Balance, LastModified) VALUES (2, 2, 'Checking', 800, SYSDATE);

INSERT INTO Accounts (AccountID, CustomerID, AccountType, Balance, LastModified) VALUES (3, 3, 'Savings', 15000, SYSDATE);

INSERT INTO Accounts (AccountID, CustomerID, AccountType, Balance, LastModified) VALUES (4, 4, 'Checking', 9500, SYSDATE);

INSERT INTO Accounts (AccountID, CustomerID, AccountType, Balance, LastModified) VALUES (5, 5, 'Savings', 20000, SYSDATE);

END;

***-- Creating Transactions table***

CREATE TABLE Transactions (

TransactionID NUMBER PRIMARY KEY,

AccountID NUMBER,

TransactionDate DATE,

Amount NUMBER,

TransactionType VARCHAR2(10),

FOREIGN KEY (AccountID) REFERENCES Accounts(AccountID)

);

***-- Inserting Values into Transactions table***

BEGIN

INSERT INTO Transactions (TransactionID, AccountID, TransactionDate, Amount, TransactionType) VALUES (6, 1, SYSDATE, 200, 'Deposit');

INSERT INTO Transactions (TransactionID, AccountID, TransactionDate, Amount, TransactionType) VALUES (7, 2, SYSDATE, 300, 'Withdrawal');

INSERT INTO Transactions (TransactionID, AccountID, TransactionDate, Amount, TransactionType) VALUES (8, 3, SYSDATE, 500, 'Deposit');

INSERT INTO Transactions (TransactionID, AccountID, TransactionDate, Amount, TransactionType) VALUES (9, 4, SYSDATE, 1000, 'Withdrawal');

INSERT INTO Transactions (TransactionID, AccountID, TransactionDate, Amount, TransactionType) VALUES (10, 5, SYSDATE, 1500, 'Deposit');

END;

/

DECLARE

***-- Cursor to retrieve all transactions for the current month***

CURSOR GenerateMonthlyStatements IS

SELECT

C.CustomerID,

C.Name,

T.TransactionID,

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

EXTRACT(MONTH FROM T.TransactionDate) = EXTRACT(MONTH FROM SYSDATE)

AND EXTRACT(YEAR FROM T.TransactionDate) = EXTRACT(YEAR FROM SYSDATE)

ORDER BY

C.CustomerID, T.TransactionDate;

***-- Record type for cursor***

txn\_rec GenerateMonthlyStatements%ROWTYPE;

***-- Variable to track current customer for header printing***

current\_customer\_id NUMBER := NULL;

BEGIN

DBMS\_OUTPUT.PUT\_LINE('--- Monthly Transaction Statements ---');

OPEN GenerateMonthlyStatements;

LOOP

FETCH GenerateMonthlyStatements INTO txn\_rec;

EXIT WHEN GenerateMonthlyStatements %NOTFOUND;

***-- Print header only when customer changes***

IF current\_customer\_id IS NULL OR txn\_rec.CustomerID != current\_customer\_id THEN

DBMS\_OUTPUT.PUT\_LINE(CHR(10) || 'Customer ID: ' || txn\_rec.CustomerID);

DBMS\_OUTPUT.PUT\_LINE('Customer Name: ' || txn\_rec.Name);

DBMS\_OUTPUT.PUT\_LINE('TransactionID | Date | Type | Amount');

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

current\_customer\_id:=txn\_rec.CustomerID;

END IF;

***-- Print transaction details***

DBMS\_OUTPUT.PUT\_LINE(

RPAD(txn\_rec.TransactionID, 14) || '|' ||

TO\_CHAR(txn\_rec.TransactionDate, 'YYYY-MM-DD') || ' |' ||

RPAD(txn\_rec.TransactionType, 11) || '|' ||

txn\_rec.Amount

);

END LOOP;

CLOSE GenerateMonthlyStatements;

END;

/

**OUTPUT:**



