Scenario 1: Generate Monthly Statements

DECLARE

CURSOR cust\_cursor IS

SELECT Customers.CustomerID, Customers.Name

FROM Customers;

CURSOR trans\_cursor IS

SELECT TransactionID, AccountID, TransactionDate, Amount, TransactionType

FROM Transactions

WHERE TransactionDate BETWEEN TRUNC(ADD\_MONTHS(SYSDATE, -1), 'MM') AND LAST\_DAY(ADD\_MONTHS(SYSDATE, -1));

v\_customer\_id NUMBER;

v\_customer\_name VARCHAR2(100);

v\_transaction\_id NUMBER;

v\_account\_id NUMBER;

v\_transaction\_date DATE;

v\_amount NUMBER;

v\_transaction\_type VARCHAR2(10);

BEGIN

OPEN cust\_cursor;

LOOP

FETCH cust\_cursor INTO v\_customer\_id, v\_customer\_name;

EXIT WHEN cust\_cursor%NOTFOUND;

-- Logic to generate statement for the customer, e.g.,

DBMS\_OUTPUT.PUT\_LINE('Statement for Customer: ' || v\_customer\_name);

OPEN trans\_cursor;

LOOP

FETCH trans\_cursor INTO v\_transaction\_id, v\_account\_id, v\_transaction\_date, v\_amount, v\_transaction\_type;

EXIT WHEN trans\_cursor%NOTFOUND;

-- Logic to process transaction for the statement, e.g.,

DBMS\_OUTPUT.PUT\_LINE('Transaction ID: ' || v\_transaction\_id || ', Amount: ' || v\_amount || ', Type: ' || v\_transaction\_type);

END LOOP;

CLOSE trans\_cursor;

-- Logic to calculate totals, balances, etc. for the statement

END LOOP;

CLOSE cust\_cursor;

END;

/

Scenario 2: Apply Annual Fee

DECLARE

CURSOR account\_cursor IS

SELECT AccountID, Balance FROM Accounts;

v\_account\_id NUMBER;

v\_balance NUMBER;

v\_annual\_fee NUMBER := 10; -- Replace with actual annual fee

BEGIN

OPEN account\_cursor;

LOOP

FETCH account\_cursor INTO v\_account\_id, v\_balance;

EXIT WHEN account\_cursor%NOTFOUND;

IF v\_balance >= v\_annual\_fee THEN

UPDATE Accounts SET Balance = Balance - v\_annual\_fee WHERE AccountID = v\_account\_id;

ELSE

-- Handle insufficient balance, e.g., log an error

END IF;

END LOOP;

CLOSE account\_cursor;

END;

/

Scenario 3: Update Loan Interest Rates

DECLARE

CURSOR loan\_cursor IS

SELECT LoanID, InterestRate FROM Loans;

v\_loan\_id NUMBER;

v\_old\_interest\_rate NUMBER;

v\_new\_interest\_rate NUMBER := 5; -- Replace with new interest rate

BEGIN

OPEN loan\_cursor;

LOOP

FETCH loan\_cursor INTO v\_loan\_id, v\_old\_interest\_rate;

EXIT WHEN loan\_cursor%NOTFOUND;

UPDATE Loans SET InterestRate = v\_new\_interest\_rate WHERE LoanID = v\_loan\_id;

END LOOP;

CLOSE loan\_cursor;

END;

/