**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.**

**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.**

**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.**

i)Scenario 1-

DECLARE

CURSOR cur\_transactions IS

SELECT customer\_id, transaction\_id, amount, transaction\_date

FROM transactions

WHERE transaction\_date BETWEEN TRUNC(SYSDATE, 'MM') AND LAST\_DAY(SYSDATE);

v\_customer\_id transactions.customer\_id%TYPE;

v\_transaction\_id transactions.transaction\_id%TYPE;

v\_amount transactions.amount%TYPE;

v\_transaction\_date transactions.transaction\_date%TYPE;

BEGIN

OPEN cur\_transactions;

LOOP

FETCH cur\_transactions INTO v\_customer\_id, v\_transaction\_id, v\_amount, v\_transaction\_date;

EXIT WHEN cur\_transactions%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Customer ID: ' || v\_customer\_id ||

', Transaction ID: ' || v\_transaction\_id ||

', Amount: ' || v\_amount ||

', Date: ' || TO\_CHAR(v\_transaction\_date, 'YYYY-MM-DD'));

END LOOP;

CLOSE cur\_transactions;

END;

/

ii)Scenario 2-

DECLARE

CURSOR cur\_accounts IS

SELECT account\_id, balance

FROM accounts;

v\_account\_id accounts.account\_id%TYPE;

v\_balance accounts.balance%TYPE;

v\_annual\_fee CONSTANT NUMBER := 50; -- Set the annual fee amount

BEGIN

OPEN cur\_accounts;

LOOP

FETCH cur\_accounts INTO v\_account\_id, v\_balance;

EXIT WHEN cur\_accounts%NOTFOUND;

UPDATE accounts

SET balance = balance - v\_annual\_fee

WHERE account\_id = v\_account\_id;

END LOOP;

CLOSE cur\_accounts;

COMMIT;

END;

/

iii)Scenario 3-

DECLARE

CURSOR cur\_loans IS

SELECT loan\_id, interest\_rate

FROM loans;

v\_loan\_id loans.loan\_id%TYPE;

v\_interest\_rate loans.interest\_rate%TYPE;

v\_new\_interest\_rate NUMBER;

BEGIN

OPEN cur\_loans;

LOOP

FETCH cur\_loans INTO v\_loan\_id, v\_interest\_rate;

EXIT WHEN cur\_loans%NOTFOUND;

-- Define the new interest rate policy here, e.g., increase by 0.5%

v\_new\_interest\_rate := v\_interest\_rate + 0.5;

UPDATE loans

SET interest\_rate = v\_new\_interest\_rate

WHERE loan\_id = v\_loan\_id;

END LOOP;

CLOSE cur\_loans;

COMMIT;

END;

/