

Exercise 6: Cursors

Scenario 1: (Ex6-Scenario1.sql)

@InitializeData.sql

SET ECHO ON

SET SERVEROUTPUT ON SIZE UNLIMITED

SPOOL output-Ex6-Scenario1.txt

VARIABLE input VARCHAR2(30)

-- PL/SQL block to generate monthly statements for all customers

DECLARE

CURSOR c_monthly_transactions IS

SELECT

t.TRANSACTIONID,
t.ACCOUNTID,
t.TRANSACTIONDATE,
t.AMOUNT,
t.TRANSACTIONTYPE,
a.CUSTOMERID,
c.NAME

FROM

TRANSACTIONS t

JOIN ACCOUNTS a ON t.ACCOUNTID = a.ACCOUNTID

JOIN CUSTOMERS c ON a.CUSTOMERID = c.CUSTOMERID

WHERE

EXTRACT(MONTH FROM t.TRANSACTIONDATE) = EXTRACT(MONTH FROM SYSDATE)

```

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

v_customer_id CUSTOMERS.CUSTOMERID%TYPE;
v_customer_name CUSTOMERS.NAME%TYPE;
v_transaction_details VARCHAR2(1000);
BEGIN
    FOR r IN c_monthly_transactions LOOP
        v_customer_id := r.CUSTOMERID;
        v_customer_name := r.NAME;
        v_transaction_details := 'Transaction ID: ' || r.TRANSACTIONID ||
            ', Date: ' || TO_CHAR(r.TRANSACTIONDATE, 'YYYY-MM-DD') ||
            ', Amount: ' || r.AMOUNT ||
            ', Type: ' || r.TRANSACTIONTYPE;

        DBMS_OUTPUT.PUT_LINE('Customer: ' || v_customer_name || ' (ID: ' ||
v_customer_id || ')');
        DBMS_OUTPUT.PUT_LINE(v_transaction_details);
        DBMS_OUTPUT.PUT_LINE('-----');
    END LOOP;
END;

/

SPOOL OFF

@DropData.sql

```

Scenario 2: (Ex6-Scenario2.sql)

```
@InitializeData.sql
```

```
SET ECHO ON
```

```
SET SERVEROUTPUT ON SIZE UNLIMITED
```

```
SPOOL output-Ex6-Scenario2.txt
```

```
VARIABLE input VARCHAR2(30)
```

```
-- PL/SQL block to apply an annual fee to all accounts
```

```
DECLARE
```

```
    CURSOR c_accounts IS
```

```
        SELECT ACCOUNTID, BALANCE
```

```
        FROM ACCOUNTS;
```

```
    v_fee NUMBER := 50; -- Example annual fee amount
```

```
BEGIN
```

```
    FOR r IN c_accounts LOOP
```

```
        UPDATE ACCOUNTS
```

```
        SET BALANCE = BALANCE - v_fee
```

```
        WHERE ACCOUNTID = r.ACCOUNTID;
```

```
        DBMS_OUTPUT.PUT_LINE('Account ID: ' || r.ACCOUNTID || ' - Fee applied. New  
balance: ' || (r.BALANCE - v_fee));
```

```
    END LOOP;
```

```
END;
```

```
/
```

```
SPOOL OFF
```

```
@DropData.sql
```

Scenario 3: (Ex6-Scenario3.sql)

@InitializeData.sql

SET ECHO ON

SET SERVEROUTPUT ON SIZE UNLIMITED

SPOOL output-Ex6-Scenario3.txt

VARIABLE input VARCHAR2(30)

-- PL/SQL block to update interest rates for all loans based on a new policy

DECLARE

CURSOR c_loans IS

SELECT LOANID, INTERESTRATE

FROM LOANS;

v_new_interest_rate NUMBER := 3; *-- Example new interest rate*

BEGIN

FOR r IN c_loans LOOP

UPDATE LOANS

SET INTERESTRATE = v_new_interest_rate

WHERE LOANID = r.LOANID;

DBMS_OUTPUT.PUT_LINE('Loan ID: ' || r.LOANID || ' - Interest rate updated to: ' ||
v_new_interest_rate);

END LOOP;

END;

/

SPOOL OFF

@DropData.sql