**Exercise 1: Control Structures**

* Creating schema and inserting values:

CREATE TABLE Customers (

CustomerID NUMBER PRIMARY KEY,

Name VARCHAR2(50),

Age NUMBER,

Balance NUMBER,

IsVIP VARCHAR2(5) DEFAULT 'FALSE'

);

CREATE TABLE Loans (

LoanID NUMBER PRIMARY KEY,

CustomerID NUMBER,

DueDate DATE,

InterestRate NUMBER,

FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)

);

INSERT INTO Customers VALUES (1, 'John Doe', 65, 15000, 'FALSE');

INSERT INTO Customers VALUES (2, 'Jane Smith', 45, 8000, 'FALSE');

INSERT INTO Customers VALUES (3, 'Bob Brown', 70, 12000, 'FALSE');

INSERT INTO Loans VALUES (101, 1, SYSDATE + 10, 7.5);

INSERT INTO Loans VALUES (102, 2, SYSDATE + 40, 6.8);

INSERT INTO Loans VALUES (103, 3, SYSDATE + 5, 8.2);

COMMIT;

* **Scenario-1(plsql-code):**

BEGIN

FOR cust IN (SELECT CustomerID FROM Customers WHERE Age > 60) LOOP

UPDATE Loans

SET InterestRate = InterestRate - 1

WHERE CustomerID = cust.CustomerID;

END LOOP;

COMMIT;

END;

* **Scenario-2:**

BEGIN

FOR cust IN (SELECT CustomerID FROM Customers WHERE Balance > 10000) LOOP

UPDATE Customers

SET IsVIP = 'TRUE'

WHERE CustomerID = cust.CustomerID;

END LOOP;

COMMIT;

END;

* **Scenario-3:**

BEGIN

FOR loan IN (

SELECT L.LoanID, C.Name, L.DueDate

FROM Loans L

JOIN Customers C ON L.CustomerID = C.CustomerID

WHERE L.DueDate <= SYSDATE + 30

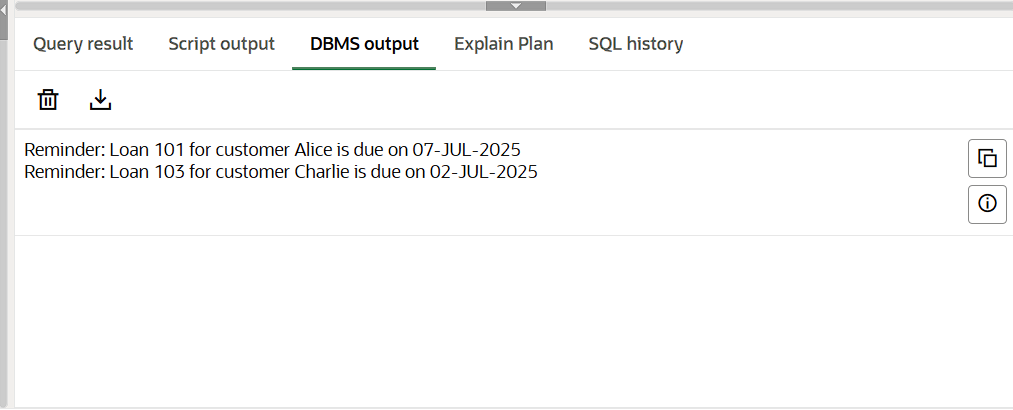
) LOOP

DBMS\_OUTPUT.PUT\_LINE('Reminder: Loan ' || loan.LoanID || ' for customer ' || loan.Name || ' is due on ' || TO\_CHAR(loan.DueDate, 'DD-MON-YYYY'));

END LOOP;

END;

* **Output:**



**Exercise 3: Stored Procedures**

* Creating schema and inserting values:

CREATE TABLE Customers (

CustomerID NUMBER PRIMARY KEY,

Name VARCHAR2(50),

Age NUMBER,

Balance NUMBER,

IsVIP VARCHAR2(5) DEFAULT 'FALSE'

);

CREATE TABLE Loans (

LoanID NUMBER PRIMARY KEY,

CustomerID NUMBER,

DueDate DATE,

InterestRate NUMBER,

FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)

);

INSERT INTO Customers VALUES (1, 'John Doe', 65, 15000, 'FALSE');

INSERT INTO Customers VALUES (2, 'Jane Smith', 45, 8000, 'FALSE');

INSERT INTO Customers VALUES (3, 'Bob Brown', 70, 12000, 'FALSE');

INSERT INTO Loans VALUES (101, 1, SYSDATE + 10, 7.5);

INSERT INTO Loans VALUES (102, 2, SYSDATE + 40, 6.8);

INSERT INTO Loans VALUES (103, 3, SYSDATE + 5, 8.2);

COMMIT;

* **Scenario-1:**

BEGIN

FOR cust IN (SELECT CustomerID FROM Customers WHERE Age > 60) LOOP

UPDATE Loans

SET InterestRate = InterestRate - 1

WHERE CustomerID = cust.CustomerID;

END LOOP;

COMMIT;

END;

* **Scenario-2:**

BEGIN

FOR cust IN (SELECT CustomerID FROM Customers WHERE Balance > 10000) LOOP

UPDATE Customers

SET IsVIP = 'TRUE'

WHERE CustomerID = cust.CustomerID;

END LOOP;

COMMIT;

END;

* **Scenario-3:**

BEGIN

FOR loan IN (

SELECT L.LoanID, C.Name, L.DueDate

FROM Loans L

JOIN Customers C ON L.CustomerID = C.CustomerID

WHERE L.DueDate <= SYSDATE + 30

) LOOP

DBMS\_OUTPUT.PUT\_LINE('Reminder: Loan ' || loan.LoanID || ' for customer ' || loan.Name || ' is due on ' || TO\_CHAR(loan.DueDate, 'DD-MON-YYYY'));

END LOOP;

END;

* **Output:**

Procedure PROCESSMONTHLYINTEREST compiled

Elapsed: 00:00:00.018

Procedure UPDATEEMPLOYEEBONUS compiled

Elapsed: 00:00:00.022

Procedure TRANSFERFUNDS compiled

Elapsed: 00:00:00.012

PL/SQL procedure successfully completed.

Elapsed: 00:00:00.085

PL/SQL procedure successfully completed.

Elapsed: 00:00:00.085

Transfer successful.

PL/SQL procedure successfully completed.

Elapsed: 00:00:00.021

ACCOUNTID CUSTOMERNAME ACCOUNTTYPE BALANCE

--------- ------------- ----------- -------

101 John Doe Savings 5050

102 Jane Smith Savings 7080

103 Bob Brown Current 4000

104 Alice Green Savings 12120

Elapsed: 00:00:00.010

4 rows selected.

EMPLOYEEID NAME DEPARTMENT SALARY

---------- -------- ---------- ------

1 Emily HR 30000

2 Robert IT 55000

3 Nina IT 49500

Elapsed: 00:00:00.006

3 rows selected.