**Exercise 1: Control Structures**

**Scenario 1:** The bank wants to apply a discount to loan interest rates for customers above 60 years old.

* + **Question:** Write a PL/SQL block that loops through all customers, checks their age, and if they are above 60, apply a 1% discount to their current loan interest rates.

**PROGRAM:**

SET SERVEROUTPUT ON;

DECLARE

CURSOR TEMP IS

SELECT c.CUSTOMERID,c.DOB,l.LOANID,l.INTERESTRATE

FROM CUSTOMERS c

JOIN LOANS l

ON c.CUSTOMERID=l.CUSTOMERID;

age NUMBER;

BEGIN

FOR rec IN TEMP LOOP

age :=FLOOR(MONTHS\_BETWEEN(SYSDATE,rec.DOB)/12);

DBMS\_OUTPUT.PUT\_LINE('CUSTOMERID :' || rec.CUSTOMERID || ' AGE :' || age);

IF age>60 THEN

UPDATE LOANS

SET INTERESTRATE=INTERESTRATE-1

WHERE LOANID = rec.LOANID;

DBMS\_OUTPUT.PUT\_LINE('DISCOUNT APPLIED TO LOADID ' || rec.LOANID ||' AGE IS '|| age);

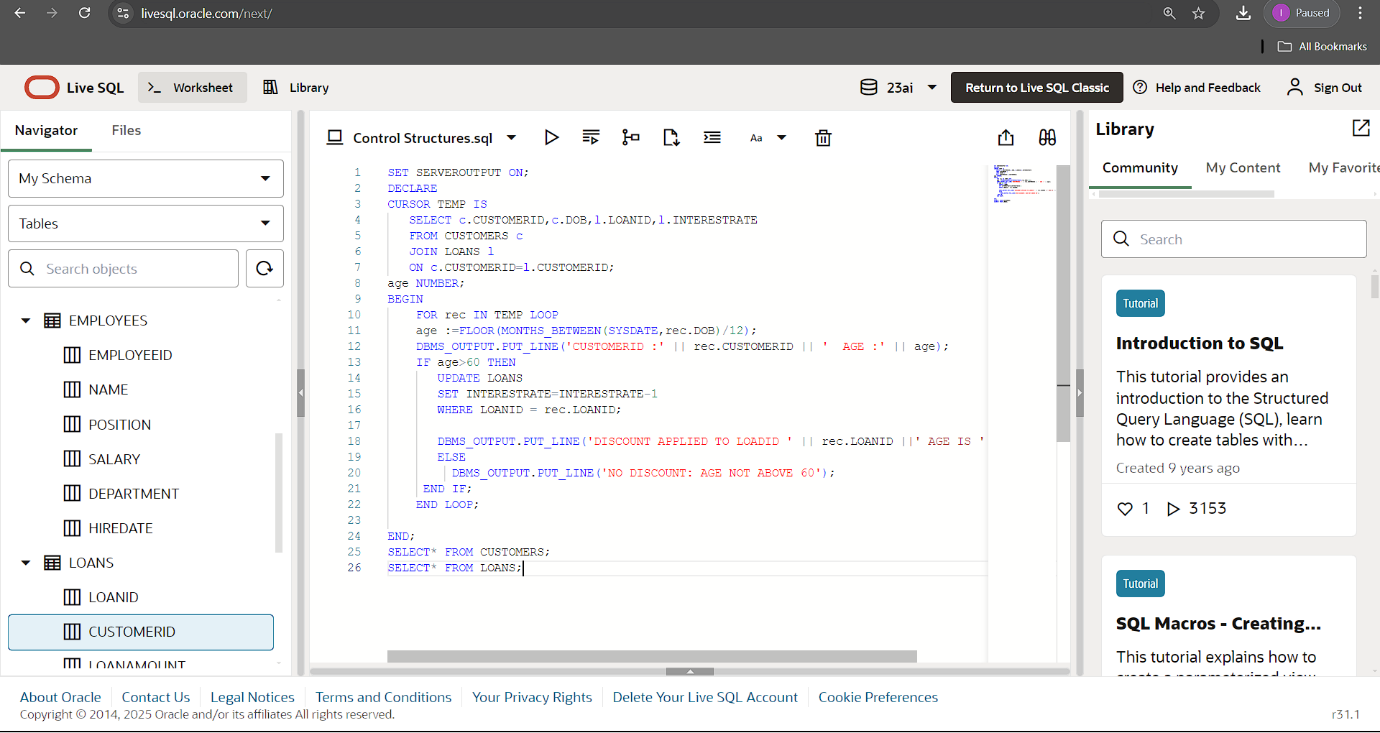
ELSE

DBMS\_OUTPUT.PUT\_LINE('NO DISCOUNT: AGE NOT ABOVE 60');

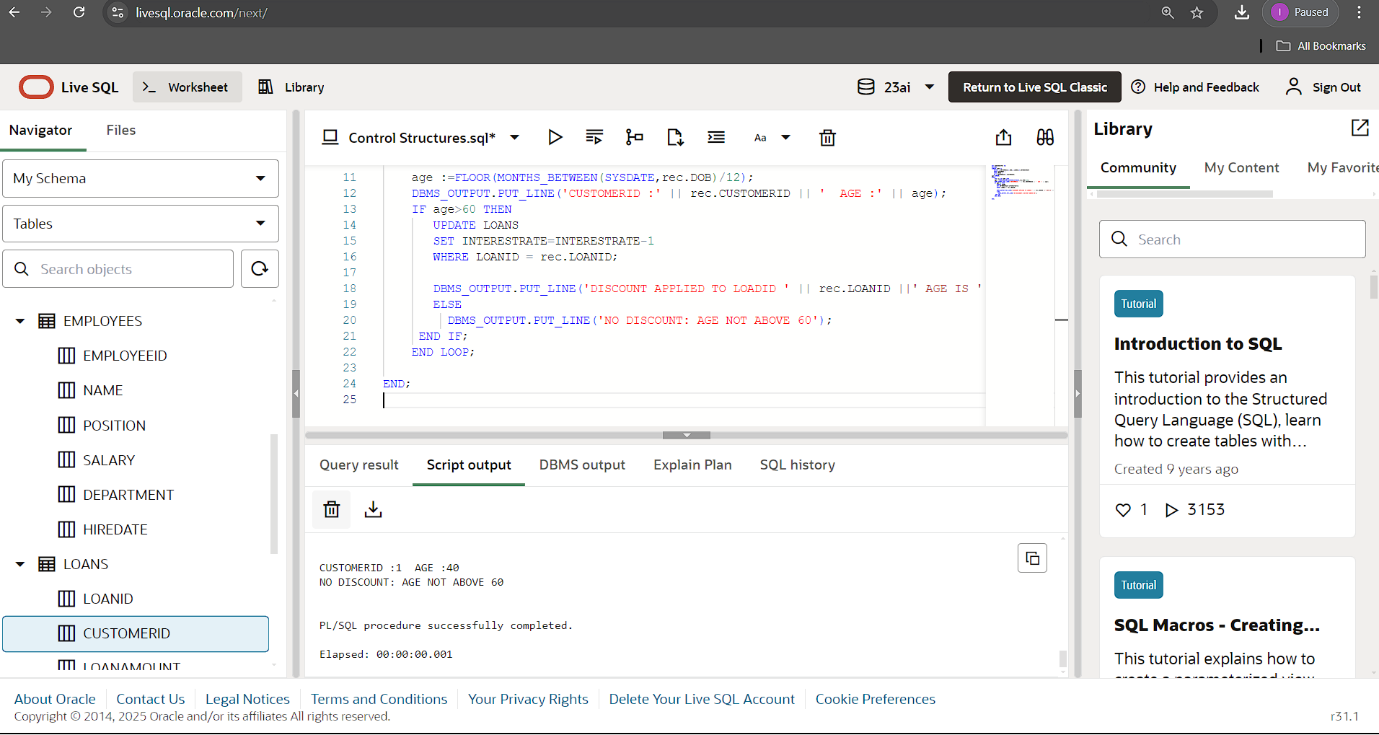
END IF;

END LOOP;

END;



**SCRIPT OUTPUT:**



**Scenario 2:** A customer can be promoted to VIP status based on their balance.

* + **Question:** Write a PL/SQL block that iterates through all customers and sets a flag IsVIP to TRUE for those with a balance over $10,000.

**PROGRAM:**

SET SERVEROUTPUT ON;

DECLARE

CURSOR cust\_cursor IS

SELECT CUSTOMERID, BALANCE

FROM CUSTOMERS;

BEGIN

FOR rec IN cust\_cursor LOOP

IF rec.BALANCE > 10000 THEN

UPDATE CUSTOMERS

SET ISVIP = 'TRUE'

WHERE CUSTOMERID = rec.CUSTOMERID;

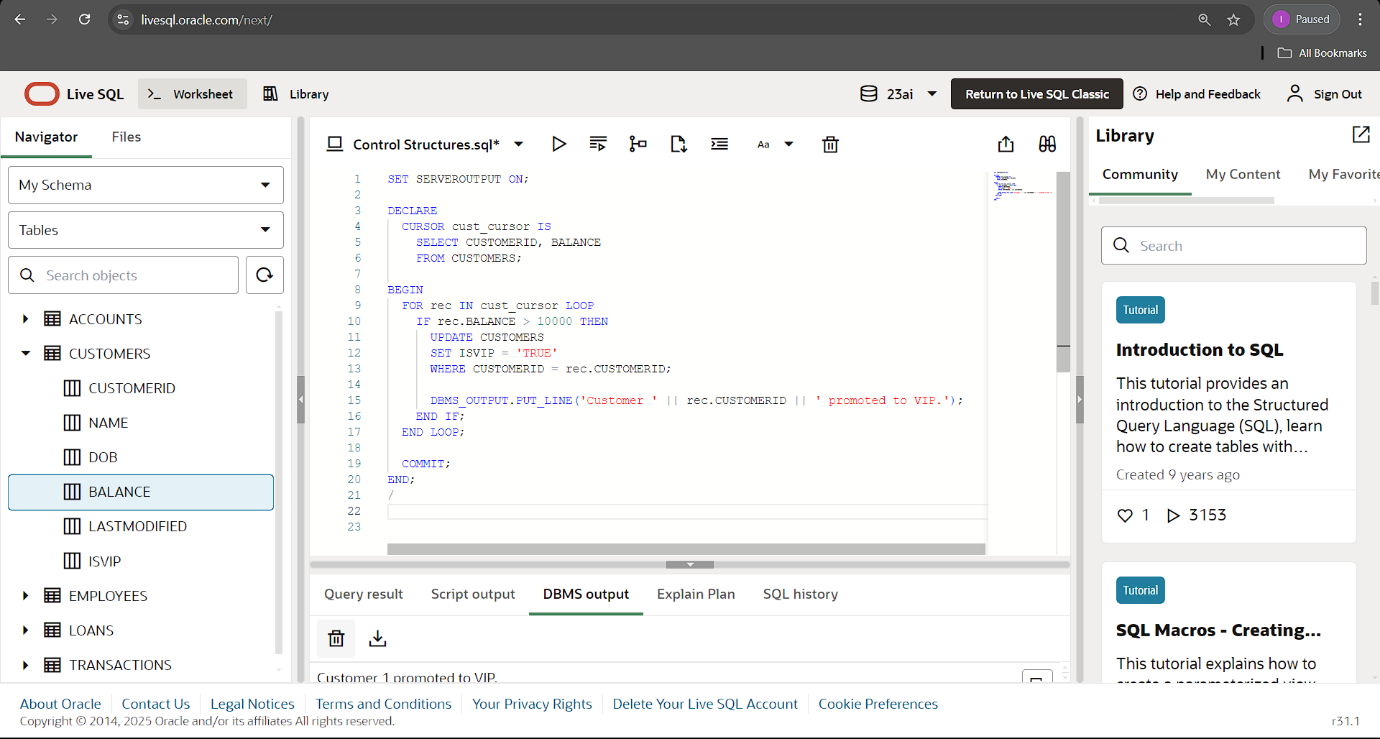
DBMS\_OUTPUT.PUT\_LINE('Customer ' || rec.CUSTOMERID || ' promoted to VIP.');

END IF;

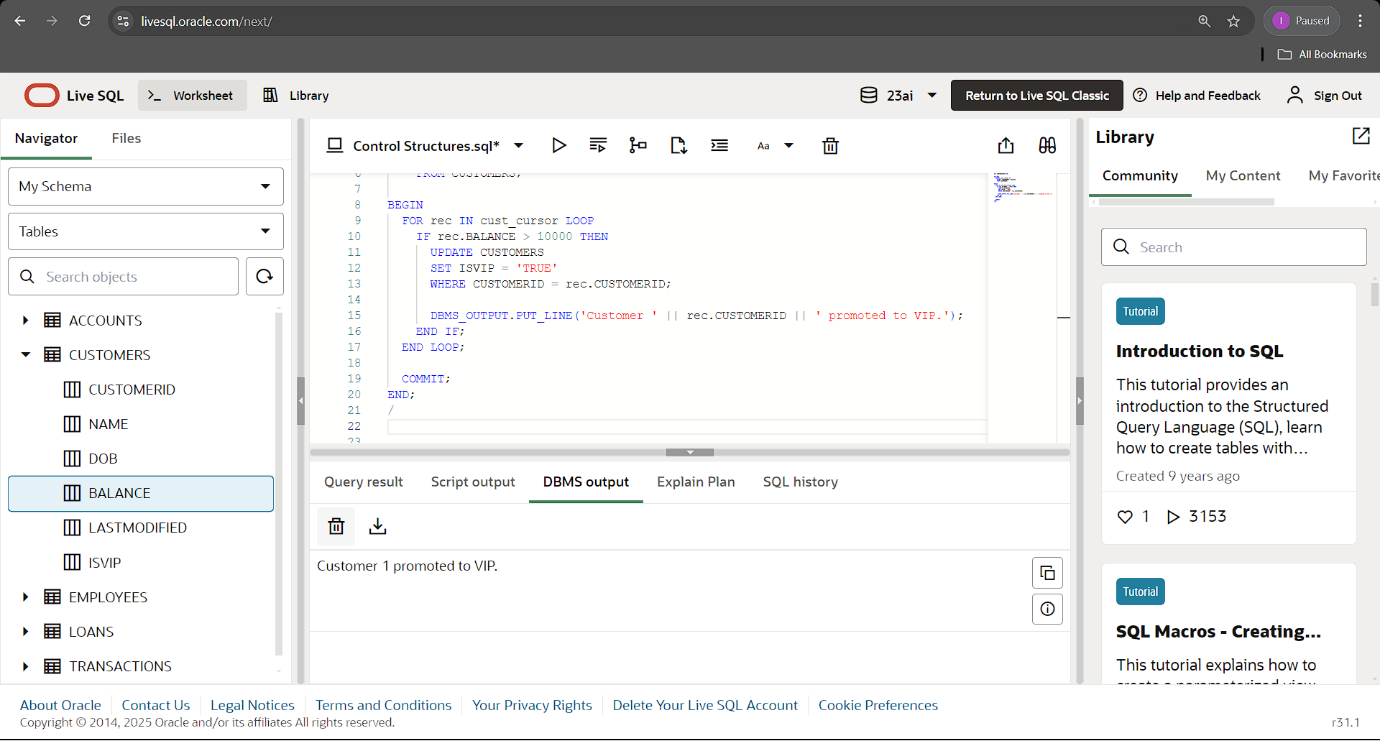
END LOOP;

COMMIT;

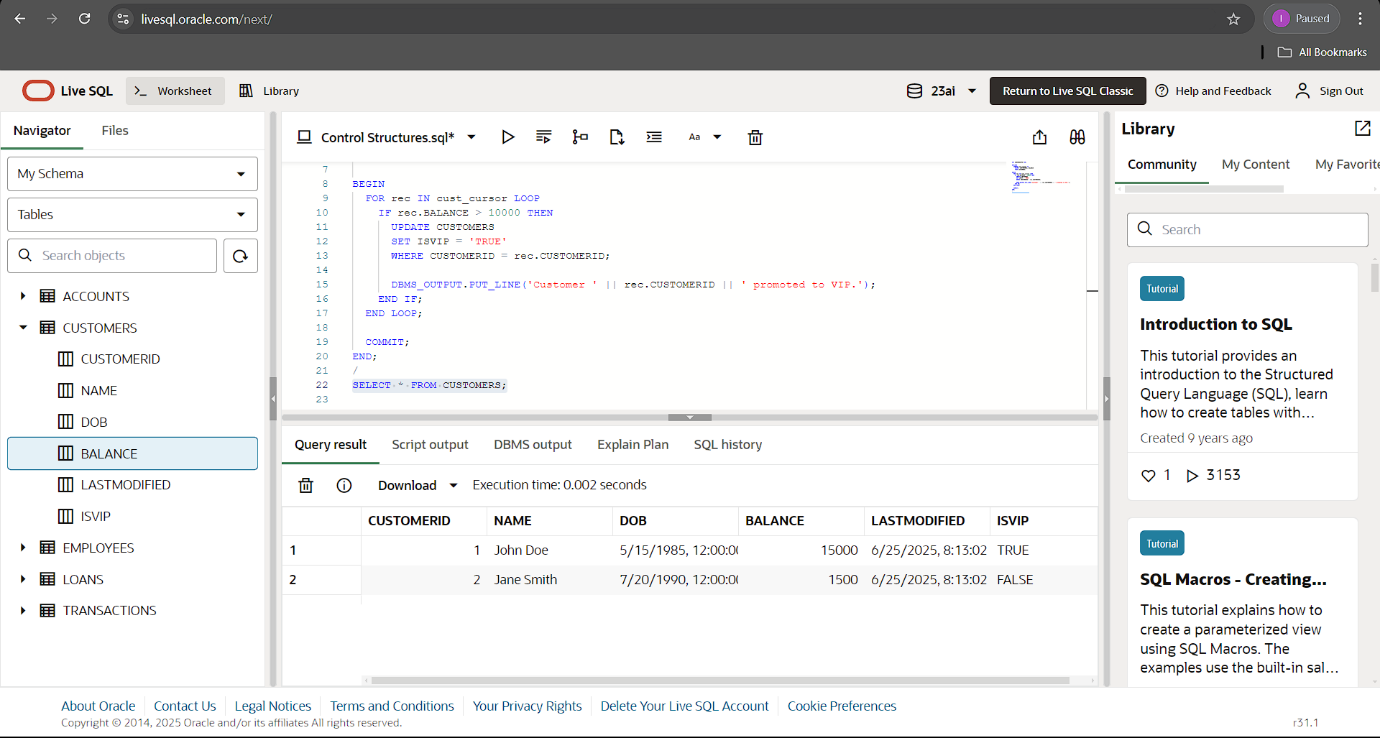
END;



**DBMS OUTPUT:**



**UPDATED CUTOMER TABLE:**



**Scenario 3:** The bank wants to send reminders to customers whose loans are due within the next 30 days.

* + **Question:** Write a PL/SQL block that fetches all loans due in the next 30 days and prints a reminder message for each customer.

**PROGRAM:**

SET SERVEROUTPUT ON;

DECLARE

CURSOR due\_loans IS

SELECT c.CustomerID, c.Name, l.LoanID, l.EndDate

FROM Loans l

JOIN Customers c ON l.CustomerID = c.CustomerID

WHERE l.EndDate BETWEEN SYSDATE AND SYSDATE + 30;

BEGIN

FOR rec IN due\_loans LOOP

DBMS\_OUTPUT.PUT\_LINE('Reminder: Customer ' || rec.Name ||

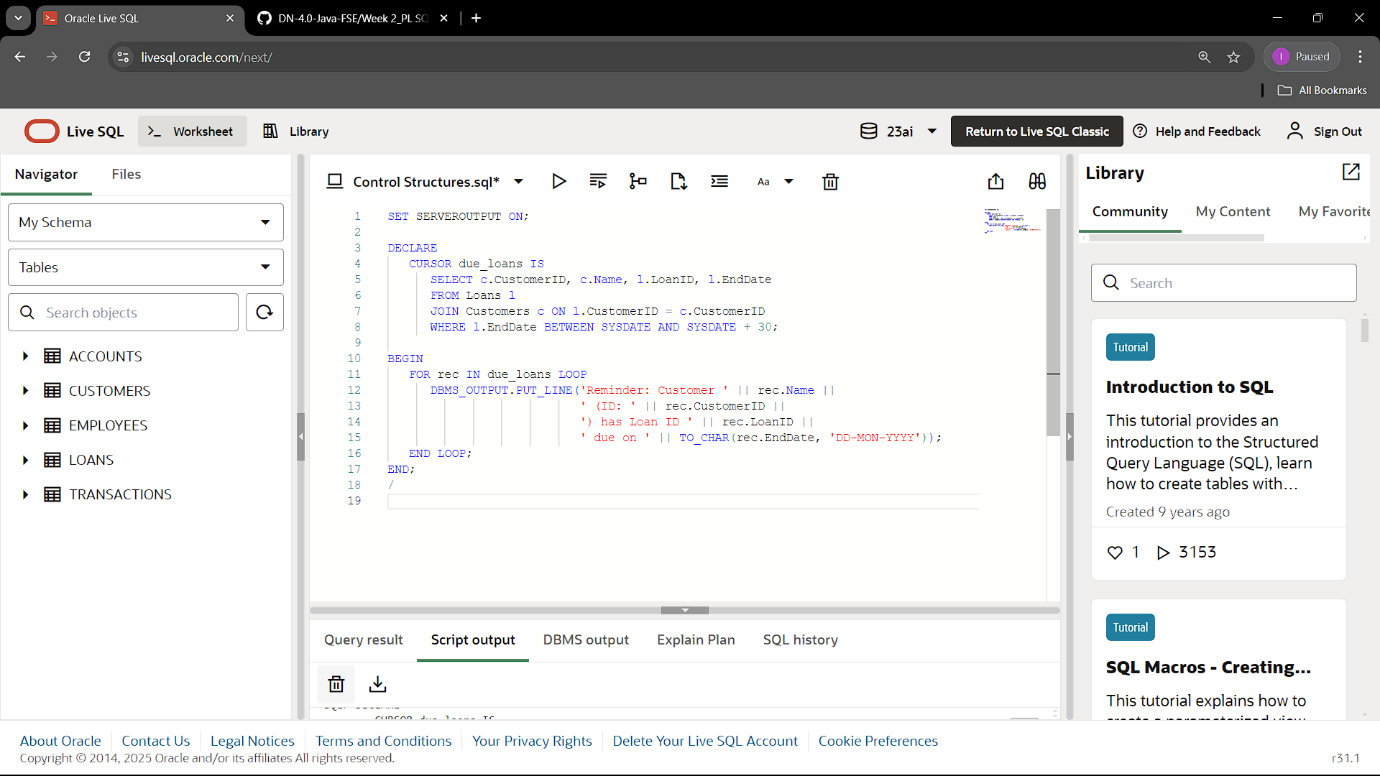
' (ID: ' || rec.CustomerID ||

') has Loan ID ' || rec.LoanID ||

' due on ' || TO\_CHAR(rec.EndDate, 'DD-MON-YYYY'));

END LOOP;

END;



**SCRIPT OUTPUT:**

