**Tema laborator 3 SGBD**

**E2 A.**

-- E2

-- a.

WITH OCTOMBRIE AS (

SELECT

TO\_DATE('2024-10-01', 'YYYY-MM-DD') + LEVEL - 1 AS ZI

FROM DUAL

CONNECT BY LEVEL <= 31

)

SELECT

TO\_DATE(OCT.ZI),

COUNT(R.BOOK\_DATE)

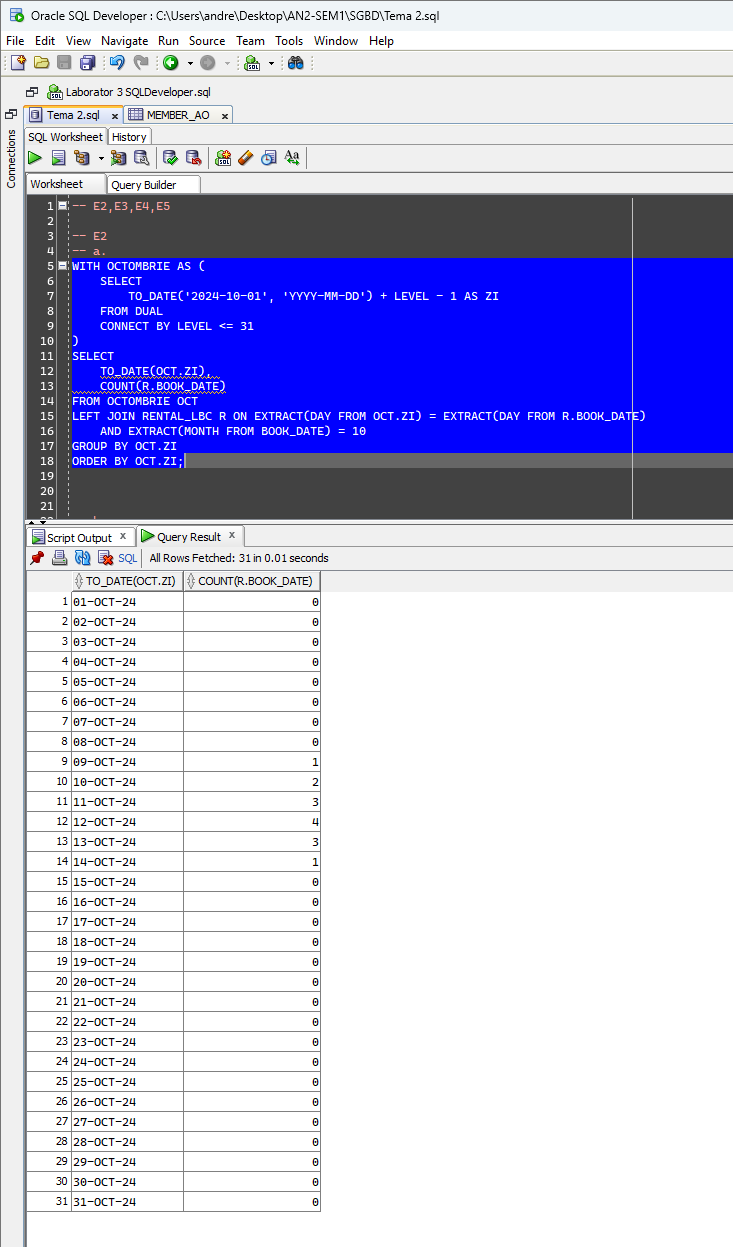
FROM OCTOMBRIE OCT

LEFT JOIN RENTAL\_LBC R ON EXTRACT(DAY FROM OCT.ZI) = EXTRACT(DAY FROM R.BOOK\_DATE)

AND EXTRACT(MONTH FROM BOOK\_DATE) = 10

GROUP BY OCT.ZI

ORDER BY OCT.ZI;

****

**E2 B.**

-- b.

CREATE TABLE OCTOMBRIE\_AO(

ID INT PRIMARY KEY,

DATA DATE NOT NULL

);

-- POPULARE UTILIZAND PL/SQL

DECLARE

AUX DATE;

BEGIN

FOR CONT IN 1..31 LOOP

AUX := TO\_DATE('2024-10-' || TO\_CHAR(CONT), 'YYYY-MM-DD');

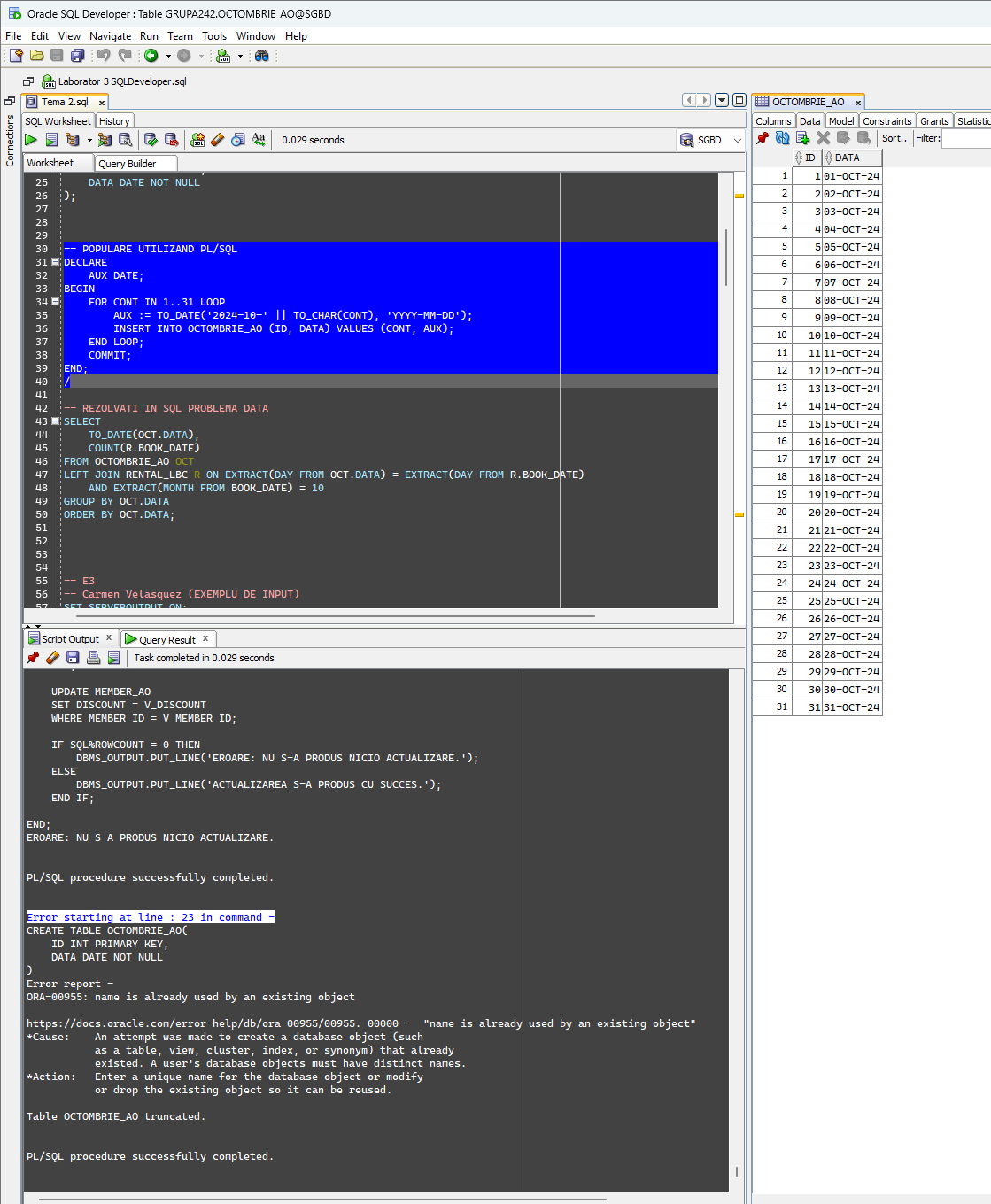
INSERT INTO OCTOMBRIE\_AO (ID, DATA) VALUES (CONT, AUX);

END LOOP;

COMMIT;

END;

/



-- REZOLVATI IN SQL PROBLEMA DATA

SELECT

TO\_DATE(OCT.DATA),

COUNT(R.BOOK\_DATE)

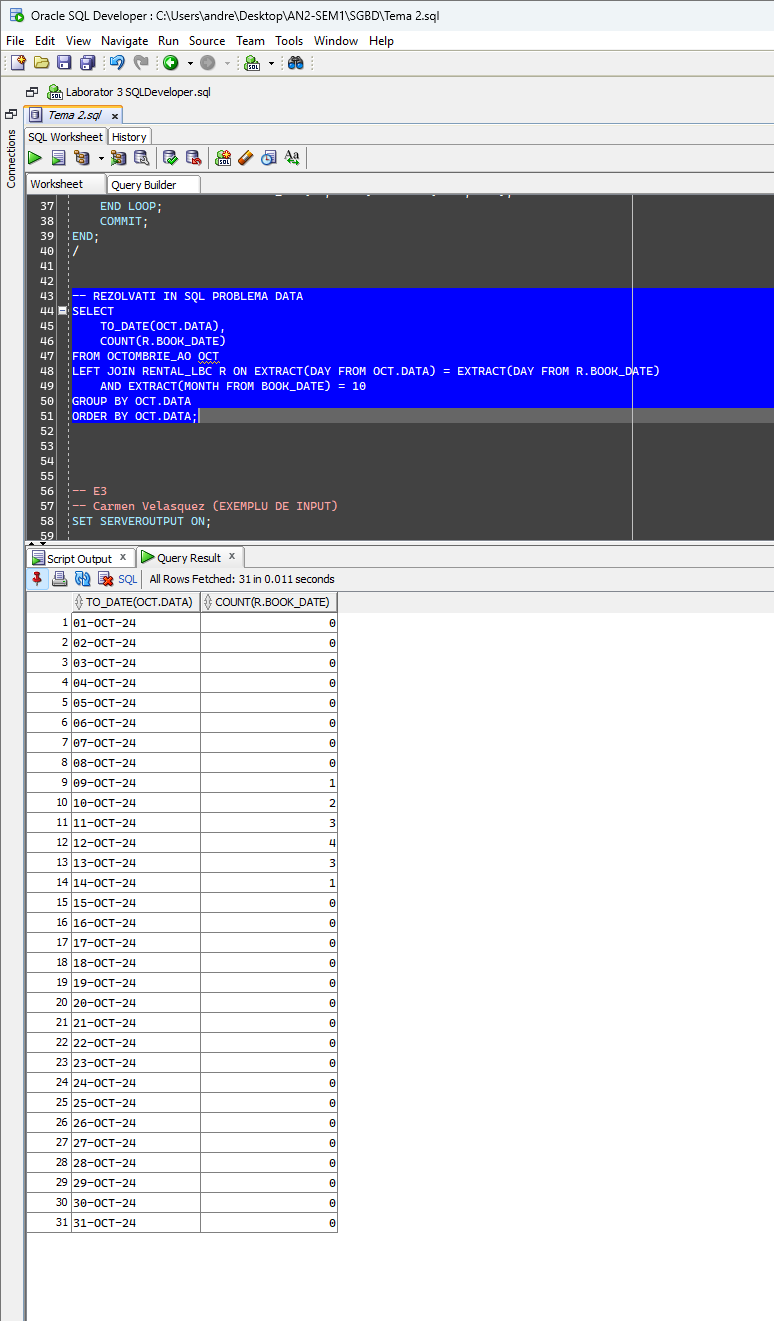
FROM OCTOMBRIE\_AO OCT

LEFT JOIN RENTAL\_LBC R ON EXTRACT(DAY FROM OCT.DATA) = EXTRACT(DAY FROM R.BOOK\_DATE)

AND EXTRACT(MONTH FROM BOOK\_DATE) = 10

GROUP BY OCT.DATA

ORDER BY OCT.DATA;



**E3**

-- E3

-- Carmen Velasquez (EXEMPLU DE INPUT)

SET SERVEROUTPUT ON;

DECLARE

V\_FIRST\_NAME VARCHAR2(100) := '&V\_FIRST\_NAME';

V\_LAST\_NAME VARCHAR2(100) := '&V\_LAST\_NAME';

REZ NUMBER(6);

V\_CONT\_FIRST\_NAME NUMBER(6);

V\_CONT\_LAST\_NAME NUMBER(6);

BEGIN

SELECT

COUNT(\*)

INTO REZ

FROM RENTAL\_LBC R

JOIN MEMBER\_LBC M ON M.MEMBER\_ID = R.MEMBER\_ID

WHERE UPPER(M.FIRST\_NAME) = UPPER(V\_FIRST\_NAME) AND UPPER(M.LAST\_NAME) = UPPER(V\_LAST\_NAME);

-- VERIFICARE CATI MEMBRI AU PRENUMELE CITIT DE LA TASTATURA

SELECT

COUNT(\*)

INTO V\_CONT\_FIRST\_NAME

FROM MEMBER\_LBC M

WHERE UPPER(M.FIRST\_NAME) = UPPER(V\_FIRST\_NAME);

-- VERIFICARE CATI MEMBRI AU NUMELE CITIT DE LA TASTATURA

SELECT

COUNT(\*)

INTO V\_CONT\_LAST\_NAME

FROM MEMBER\_LBC M

WHERE UPPER(M.LAST\_NAME) = UPPER(V\_LAST\_NAME);

IF V\_CONT\_FIRST\_NAME = 0 OR V\_CONT\_LAST\_NAME = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('NU EXISTA MEMBRI CU NUMELE DAT.');

ELSIF V\_CONT\_FIRST\_NAME != 1 AND V\_CONT\_LAST\_NAME != 1 THEN

DBMS\_OUTPUT.PUT\_LINE('NU SE POATE AFISA NUMARUL DE IMPRUMUTURI PENTRU CA EXISTA MAI MULTI MEMBRI CU ACELASI NUME.');

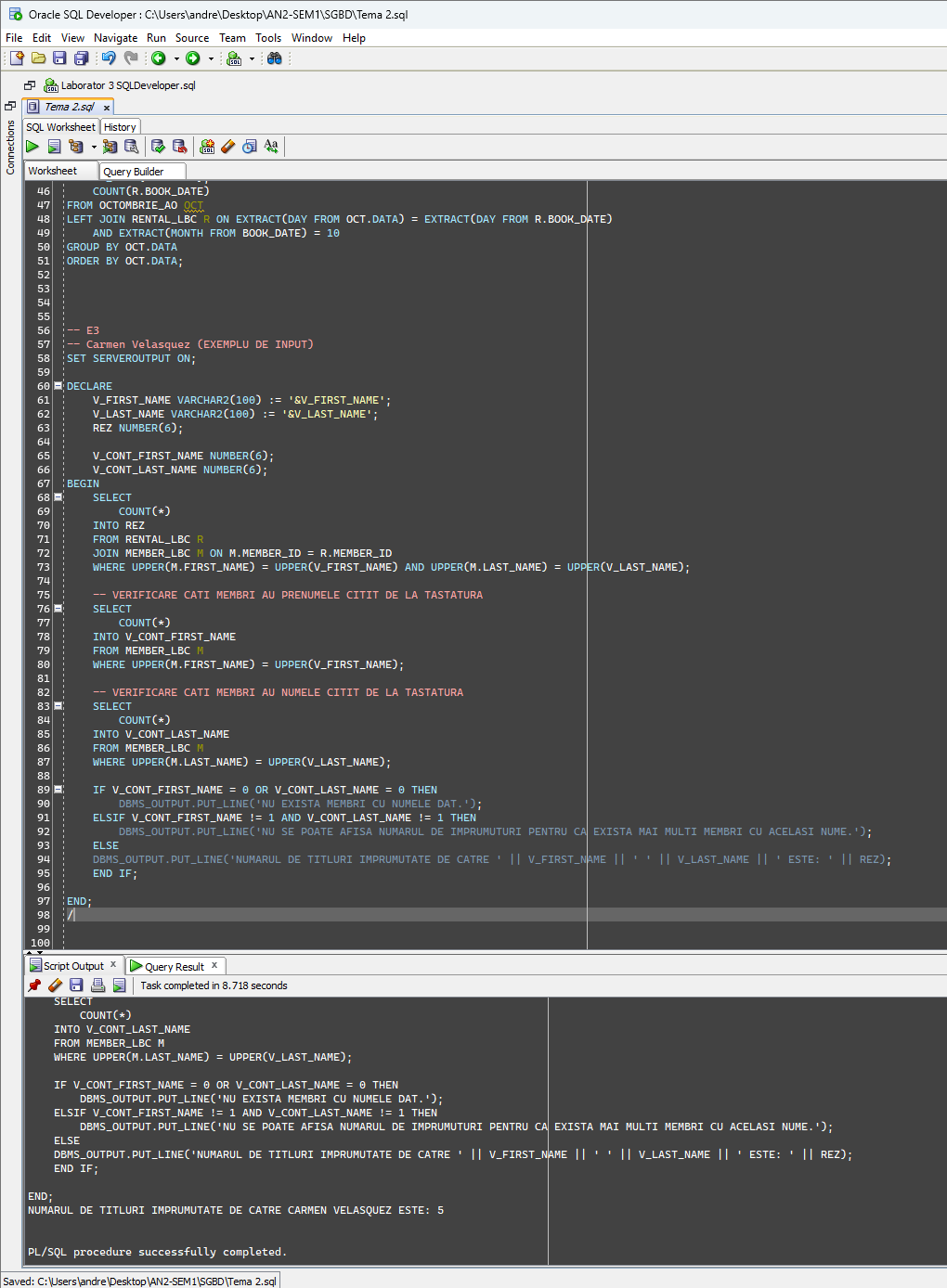
ELSE

DBMS\_OUTPUT.PUT\_LINE('NUMARUL DE TITLURI IMPRUMUTATE DE CATRE ' || V\_FIRST\_NAME || ' ' || V\_LAST\_NAME || ' ESTE: ' || REZ);

END IF;

END;

/



**E4**

-- E4

-- Carmen Velasquez (EXEMPLU DE INPUT)

SET SERVEROUTPUT ON;

DECLARE

V\_FIRST\_NAME VARCHAR2(100) := '&V\_FIRST\_NAME';

V\_LAST\_NAME VARCHAR2(100) := '&V\_LAST\_NAME';

REZ NUMBER(6);

V\_CONT\_FIRST\_NAME NUMBER(6);

V\_CONT\_LAST\_NAME NUMBER(6);

V\_TITLURI NUMBER(6);

V\_RAPORT NUMBER;

V\_CATEGORIE VARCHAR(5);

BEGIN

-- EXTRAGERE NUMAR DE IMPRUMUTURI

SELECT

COUNT(\*)

INTO REZ

FROM RENTAL\_LBC R

JOIN MEMBER\_LBC M ON M.MEMBER\_ID = R.MEMBER\_ID

WHERE UPPER(M.FIRST\_NAME) = UPPER(V\_FIRST\_NAME) AND UPPER(M.LAST\_NAME) = UPPER(V\_LAST\_NAME);

-- VERIFICARE CATI MEMBRI AU PRENUMELE CITIT DE LA TASTATURA

SELECT

COUNT(\*)

INTO V\_CONT\_FIRST\_NAME

FROM MEMBER\_LBC M

WHERE UPPER(M.FIRST\_NAME) = UPPER(V\_FIRST\_NAME);

-- VERIFICARE CATI MEMBRI AU NUMELE CITIT DE LA TASTATURA

SELECT

COUNT(\*)

INTO V\_CONT\_LAST\_NAME

FROM MEMBER\_LBC M

WHERE UPPER(M.LAST\_NAME) = UPPER(V\_LAST\_NAME);

-- EXTRAGERE NUMAR TITLURI

SELECT

COUNT(T.TITLE\_ID)

INTO V\_TITLURI

FROM TITLE\_LBC T;

IF V\_CONT\_FIRST\_NAME = 0 OR V\_CONT\_LAST\_NAME = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('NU EXISTA MEMBRI CU NUMELE DAT.');

ELSIF V\_CONT\_FIRST\_NAME != 1 AND V\_CONT\_LAST\_NAME != 1 THEN

DBMS\_OUTPUT.PUT\_LINE('NU SE POATE AFISA NUMARUL DE IMPRUMUTURI PENTRU CA EXISTA MAI MULTI MEMBRI CU ACELASI NUME.');

ELSE

DBMS\_OUTPUT.PUT\_LINE('NUMARUL DE TITLURI IMPRUMUTATE DE CATRE ' || V\_FIRST\_NAME || ' ' || V\_LAST\_NAME || ' ESTE: ' || REZ);

-- CALCULARE CATEGORIE

V\_RAPORT := REZ / V\_TITLURI;

V\_CATEGORIE :=

CASE

WHEN V\_RAPORT > 0.75 THEN '1'

WHEN V\_RAPORT > 0.50 THEN '2'

WHEN V\_RAPORT > 0.25 THEN '3'

ELSE '4'

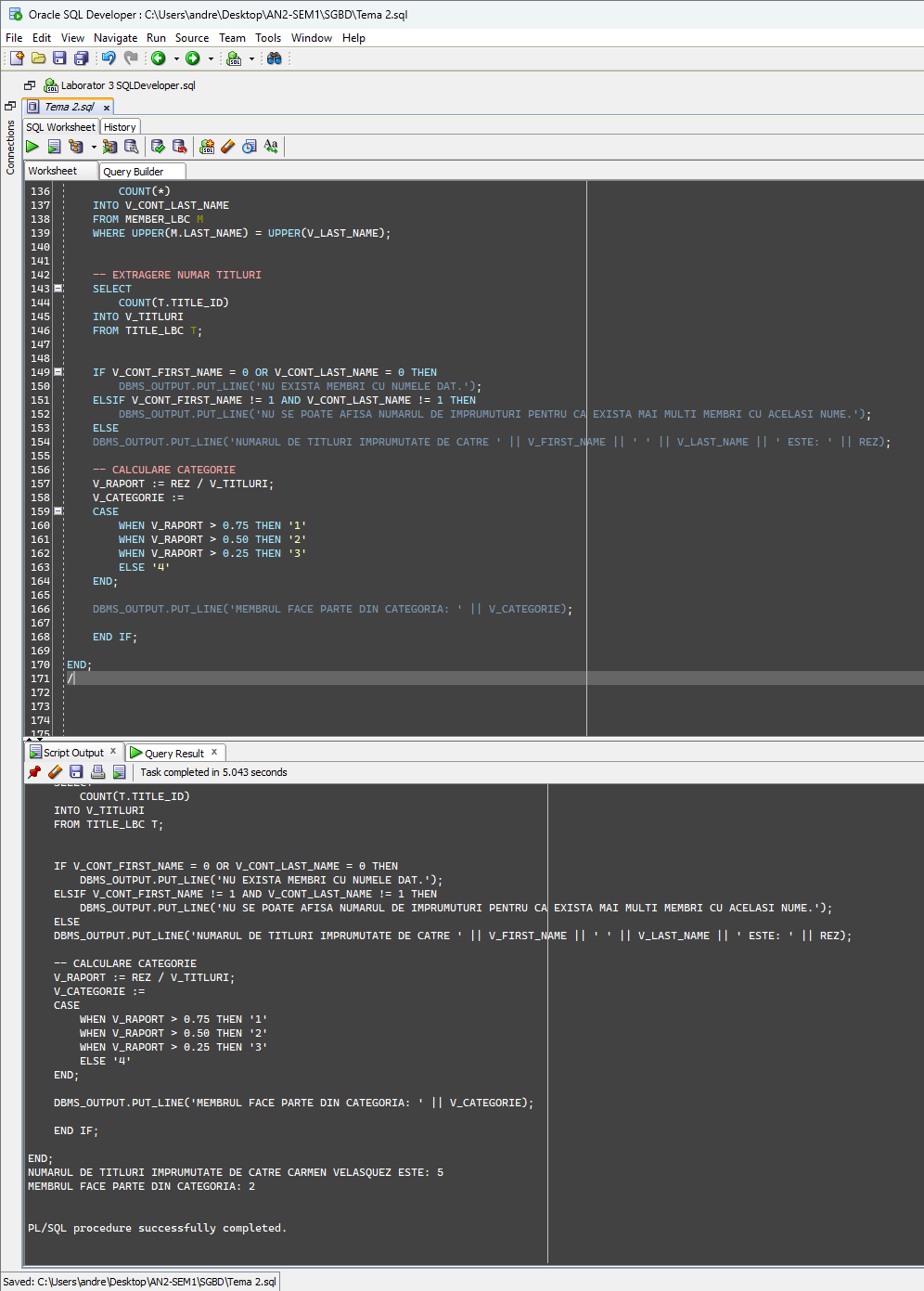
END;

DBMS\_OUTPUT.PUT\_LINE('MEMBRUL FACE PARTE DIN CATEGORIA: ' || V\_CATEGORIE);

END IF;

END;

/

****

**E5 – PENTRU MEMBER\_ID = 102**

-- E5

CREATE TABLE MEMBER\_AO AS

SELECT \*

FROM MEMBER\_LBC;

ALTER TABLE MEMBER\_AO

ADD DISCOUNT NUMBER(3, 2);

-- Actualizati coloana discount pentru un membru al carui cod este dat de la tastatura.

DECLARE

V\_MEMBER\_ID MEMBER\_AO.MEMBER\_ID%TYPE := &V\_MEMBER\_ID;

V\_DISCOUNT MEMBER\_AO.DISCOUNT%TYPE;

V\_TITLURI NUMBER(6);

V\_NUMAR\_IMPRUMUTURI NUMBER(6);

V\_RAPORT NUMBER;

BEGIN

-- EXTRAGERE NUMAR TITLURI

SELECT

COUNT(T.TITLE\_ID)

INTO V\_TITLURI

FROM TITLE\_LBC T;

-- EXTRAGERE NUMAR DE IMPRUMUTURI ALE MEMBRULUI

SELECT

COUNT(\*)

INTO V\_NUMAR\_IMPRUMUTURI

FROM RENTAL\_LBC R

JOIN MEMBER\_AO M ON M.MEMBER\_ID = R.MEMBER\_ID

WHERE M.MEMBER\_ID = V\_MEMBER\_ID;

V\_RAPORT := V\_NUMAR\_IMPRUMUTURI / V\_TITLURI;

V\_DISCOUNT :=

CASE

WHEN V\_RAPORT > 0.75 THEN 0.1

WHEN V\_RAPORT > 0.50 THEN 0.05

WHEN V\_RAPORT > 0.25 THEN 0.03

ELSE NULL

END;

UPDATE MEMBER\_AO

SET DISCOUNT = V\_DISCOUNT

WHERE MEMBER\_ID = V\_MEMBER\_ID;

IF SQL%ROWCOUNT = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('EROARE: NU S-A PRODUS NICIO ACTUALIZARE.');

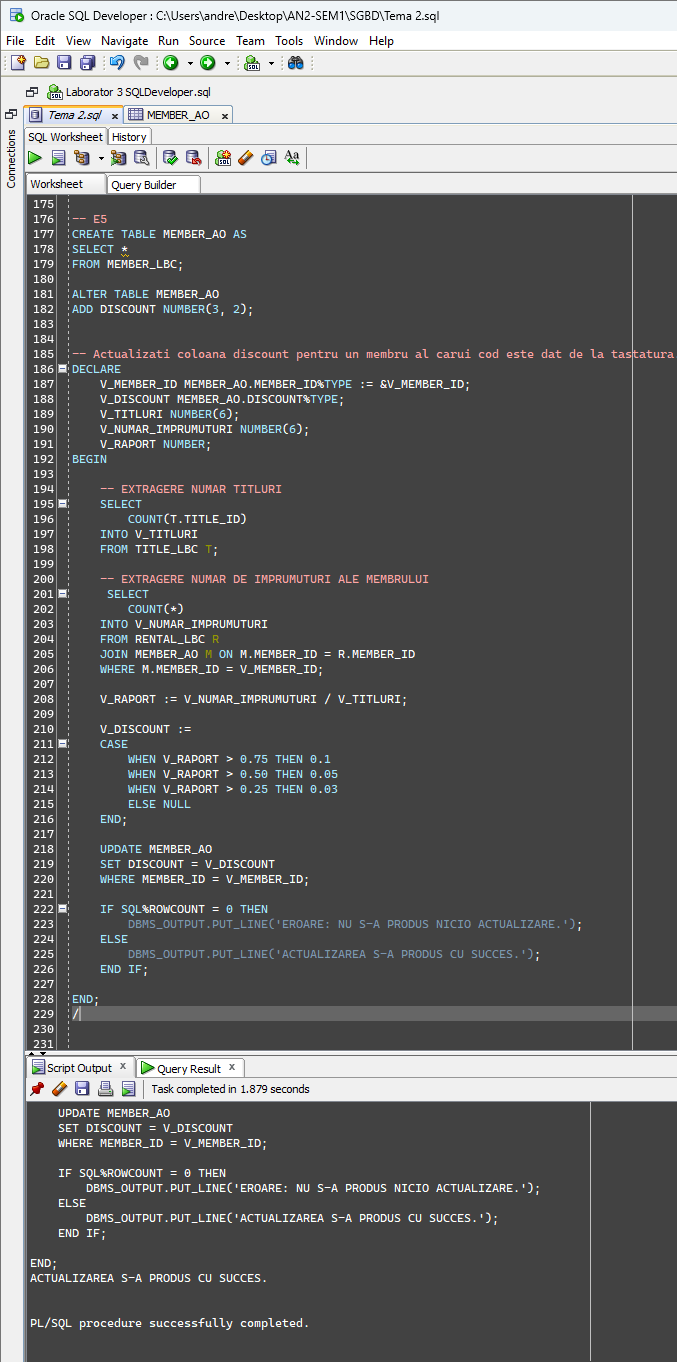
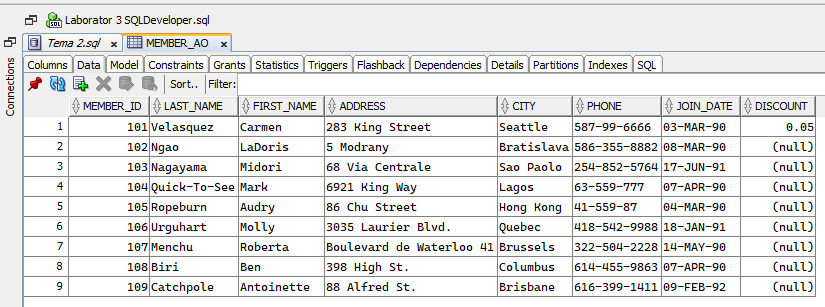
ELSE

DBMS\_OUTPUT.PUT\_LINE('ACTUALIZAREA S-A PRODUS CU SUCCES.');

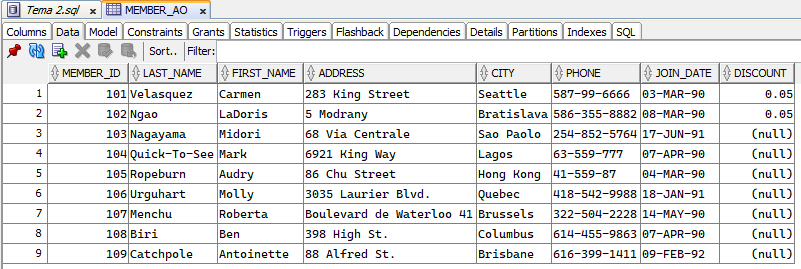
END IF;

END;

/

**Inainte de rulare:**

**Dupa rulare:**

****