/\* E1. Adaptati cerinta exercitiului 4 din fisierul Laborator8\_PLSQL\_optional.pdf pe schema aleasa pentru proiect.\*/

/\*4. Sa se creeze un pachet care sa contina:

- o functie prin care se vor returna toti medicii care indeplinesc o anumita conditie, data ca

parametru;

- o functie prin care se vor returna toti medicii care lucreaza intr-un anumit spital al carui id e dat ca parametru

\*/

-- Define package declarations

CREATE OR REPLACE PACKAGE test\_package AS

TYPE r\_cursor IS REF CURSOR;

FUNCTION get\_by\_cond(cond VARCHAR2) RETURN r\_cursor;

FUNCTION get\_by\_h\_id(h\_id hospitals\_medics\_t.hospital\_id%TYPE) RETURN r\_cursor;

END test\_package;

/

-- Define package body

CREATE OR REPLACE PACKAGE BODY test\_package AS

FUNCTION get\_by\_cond(cond VARCHAR2) RETURN r\_cursor IS

by\_cond\_cursor r\_cursor ;

query\_stmt VARCHAR(500);

BEGIN

query\_stmt := 'SELECT \* FROM medics\_t ' || cond;

OPEN by\_cond\_cursor FOR query\_stmt;

RETURN by\_cond\_cursor;

END;

FUNCTION get\_by\_h\_id(h\_id hospitals\_medics\_t.hospital\_id%TYPE) RETURN r\_cursor IS

by\_h\_id\_cursor r\_cursor;

query\_stmt VARCHAR(500);

BEGIN

query\_stmt := 'SELECT m.medic\_id, m.medic\_name, m.years\_exp, m.specialization\_id

FROM hospitals\_medics\_t hm

JOIN medics\_t m ON m.medic\_id = hm.medic\_id

WHERE hospital\_id = :h\_id';

OPEN by\_h\_id\_cursor FOR query\_stmt USING h\_id;

RETURN by\_h\_id\_cursor;

END;

END test\_package;

/

-- Test package

DECLARE

test\_cursor test\_package.r\_cursor;

medic\_rec medics\_t%ROWTYPE;

BEGIN

DBMS\_OUTPUT.PUT\_LINE('By condition test');

test\_cursor := test\_package.get\_by\_cond('WHERE years\_exp > 5');

LOOP

FETCH test\_cursor INTO medic\_rec;

EXIT WHEN test\_cursor%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Medic with id ' || medic\_rec.medic\_id || ' is called ' || medic\_rec.medic\_name || ' has specialization ' ||

medic\_rec.specialization\_id || ' and has ' || medic\_rec.years\_exp || ' years experience');

END LOOP;

CLOSE test\_cursor;

DBMS\_OUTPUT.PUT\_LINE('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

DBMS\_OUTPUT.PUT\_LINE('By hospital\_id test');

test\_cursor := test\_package.get\_by\_h\_id(4);

LOOP

FETCH test\_cursor INTO medic\_rec;

EXIT WHEN test\_cursor%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Medic with id ' || medic\_rec.medic\_id || ' is called ' || medic\_rec.medic\_name || ' has specialization ' ||

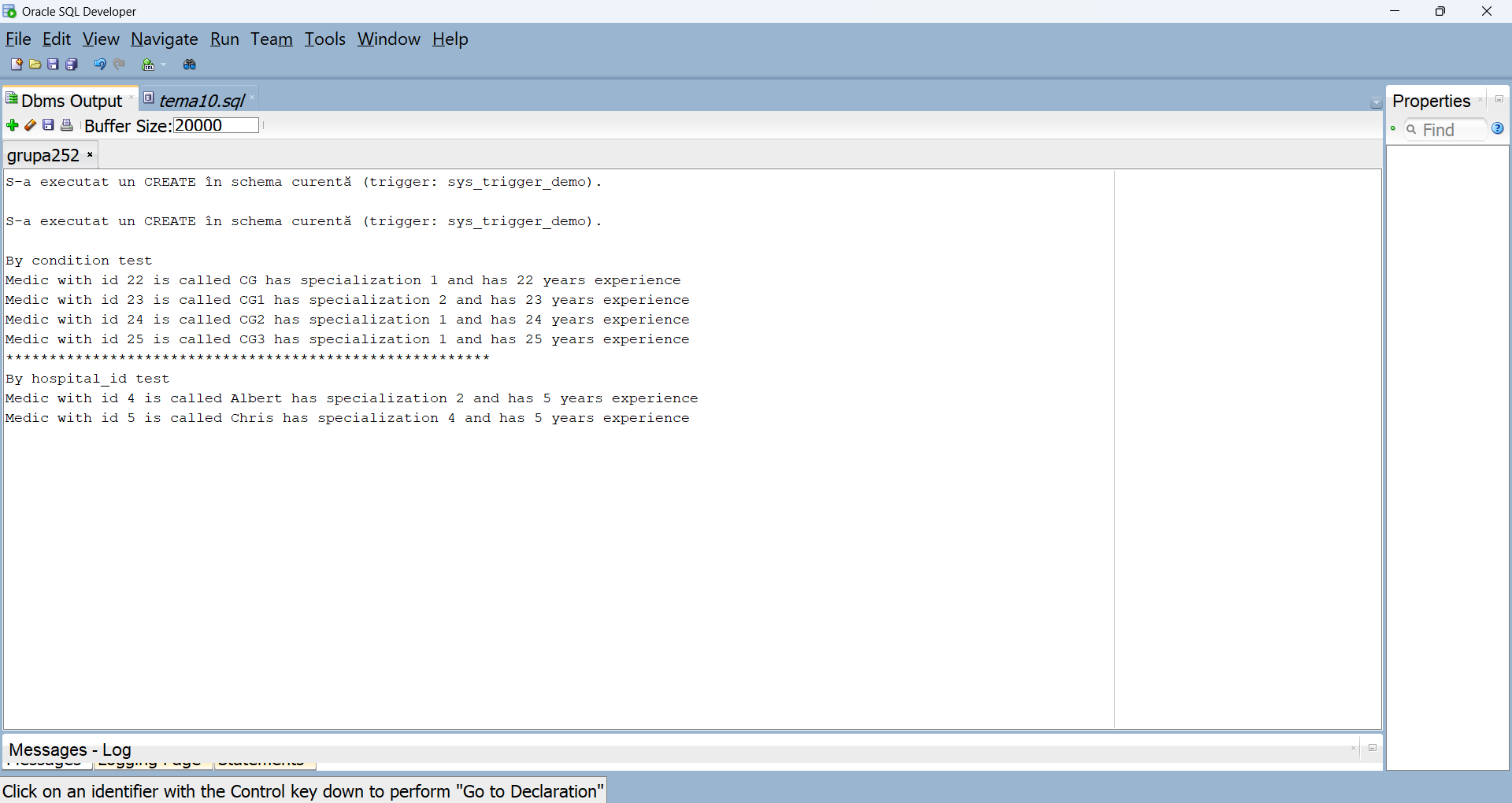
medic\_rec.specialization\_id || ' and has ' || medic\_rec.years\_exp || ' years experience');

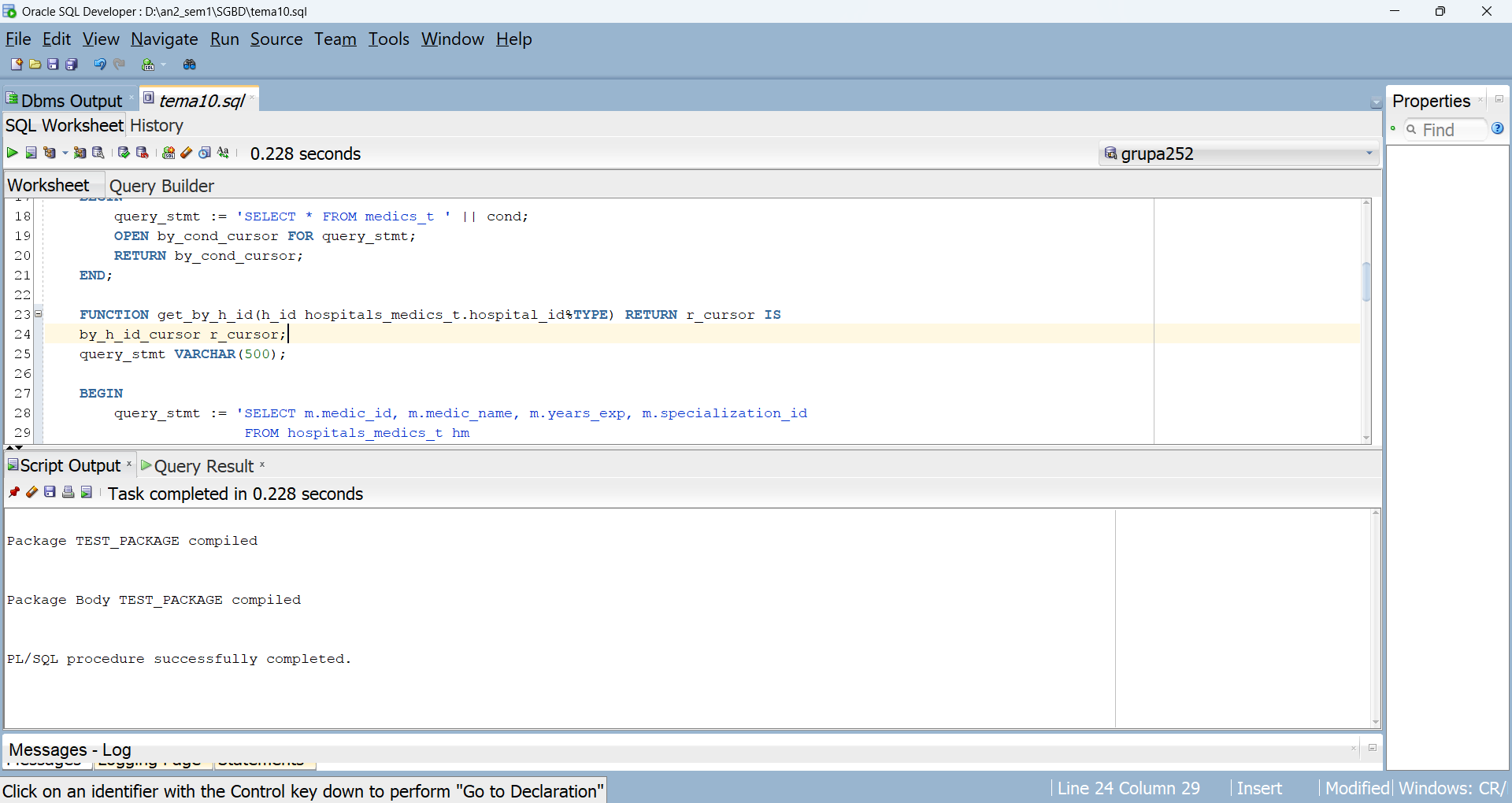
END LOOP;

CLOSE test\_cursor;

END;

/





/\*E2. Implementati o procedură care declanseaza trigger-ul sistem creat pentru proiect.\*/

CREATE OR REPLACE PROCEDURE test\_trigger\_proc IS

BEGIN

EXECUTE IMMEDIATE 'DROP TABLE programari';

END;

/

BEGIN

test\_trigger\_proc;

END;

/