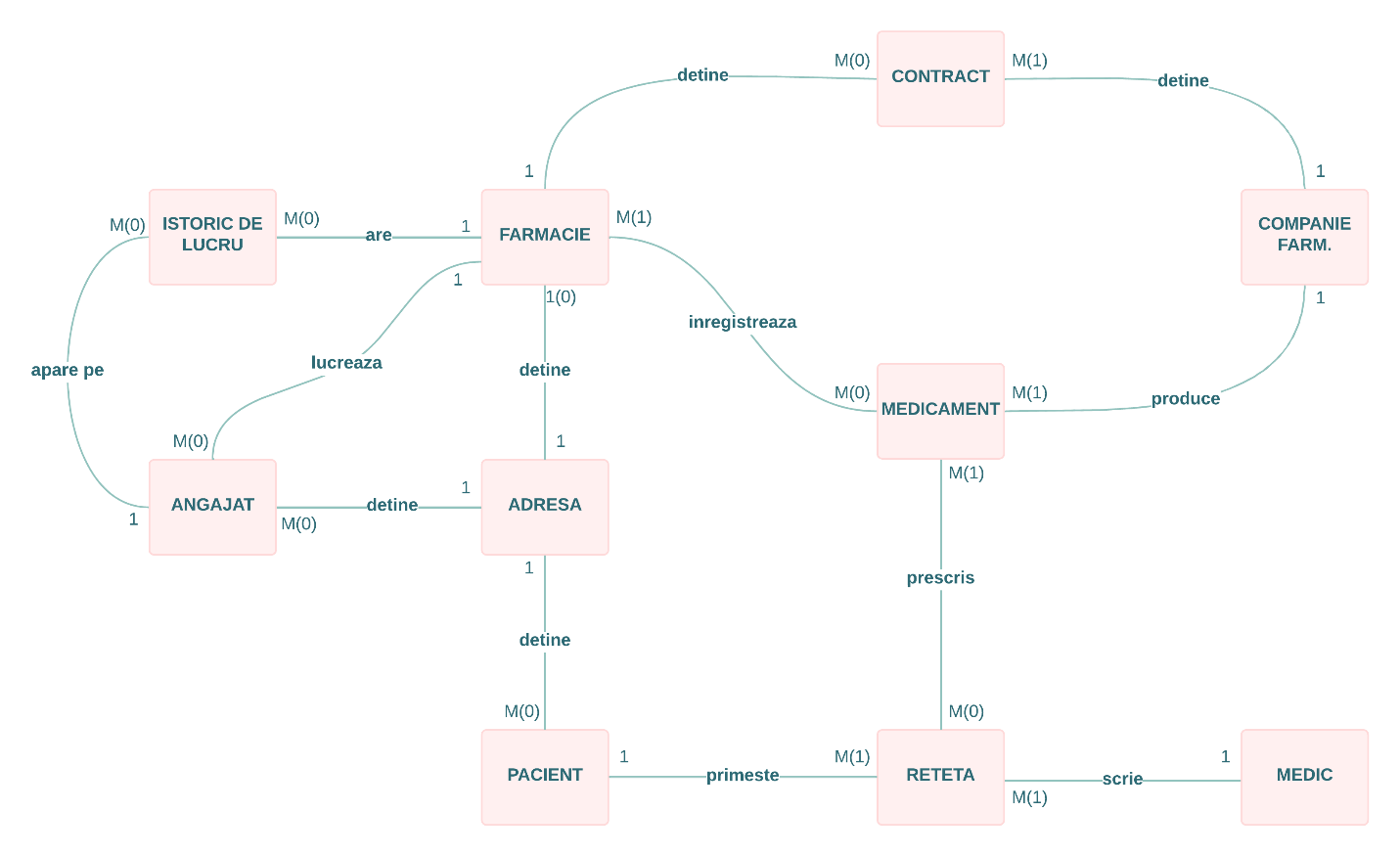
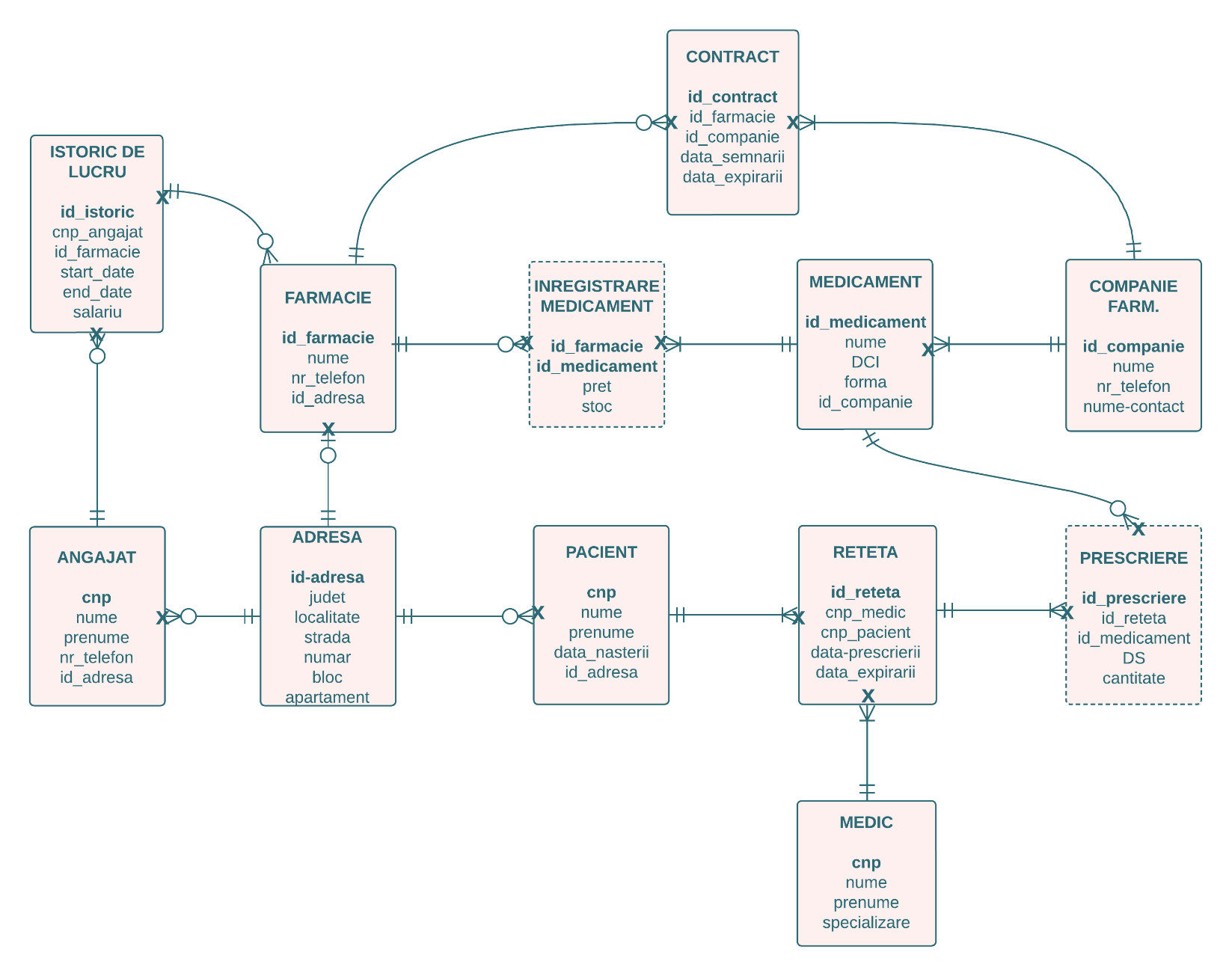
**PROIECT SISTEME DE GESTIUNE A BAZELOR DE DATE**

1. **Prezentare**

Baza de date conține informațiile necesare pentru gestionarea activității unui lanț de farmacii. **Farmaciile** sunt aprovizionate cu medicamente produse de către **companiile farmaceutice** cu care acestea au **contract**. **Medicamentele** din farmacii sunt vândute **pacienților** în urma unor **rețete** date de către **medici**. Nu în ultiuml rând, farmaciile au angajați **farmaciști** și se dorește păstrarea unui **istoric** al activității acestora în cadrul farmaciilor (data angajării, data intreruperii activitătii, farmacia la care acesta a lucrat etc.).

1. **Diagrama entitate-relatie**



**3. Diagrama conceptuală**

**4. Implementare diagramă conceptuală**

CREATE TABLE **ADRESA**(

id\_adresa NUMBER(6) primary key,

judet VARCHAR2(50) NOT NULL,

localitate VARCHAR2(50) NOT NULL,

strada VARCHAR2(50) NOT NULL,

numar NUMBER(4) NOT NULL,

bloc NUMBER(4),

apartament NUMBER(4));

CREATE TABLE **FARMACIE**(

id\_farmacie NUMBER(6) primary key,

nume VARCHAR2(50) NOT NULL,

nr\_telefon VARCHAR2(50) NOT NULL,

id\_adresa NUMBER(6),

FOREIGN KEY (id\_adresa) REFERENCES ADRESA(id\_adresa) ON DELETE SET NULL);

CREATE TABLE **PACIENT**(

cnp CHAR(13) primary key,

nume VARCHAR2(50) NOT NULL,

prenume VARCHAR2(50) NOT NULL,

data\_nasterii DATE NOT NULL,

id\_adresa NUMBER(6),

FOREIGN KEY (id\_adresa) REFERENCES ADRESA(id\_adresa) ON DELETE SET NULL);

CREATE TABLE **MEDIC**(

cnp CHAR(13) primary key,

nume VARCHAR2(50) NOT NULL,

prenume VARCHAR2(50) NOT NULL,

specializare VARCHAR2(50) NOT NULL);

CREATE TABLE **COMPANIE\_FARMACEUTICA**(

id\_companie NUMBER(6) primary key,

nume VARCHAR2(50) NOT NULL,

nr\_telefon VARCHAR2(50) NOT NULL,

nume\_contact VARCHAR2(50) NOT NULL);

CREATE TABLE **MEDICAMENT**(

id\_medicament NUMBER(6) primary key,

nume VARCHAR2(50) NOT NULL,

DCI VARCHAR2(50) NOT NULL,

forma VARCHAR2(50) NOT NULL,

id\_companie NUMBER(6) NOT NULL,

FOREIGN KEY (id\_companie) REFERENCES COMPANIE\_FARMACEUTICA(id\_companie) ON DELETE CASCADE);

CREATE TABLE **RETETA**(

id\_reteta NUMBER(6) primary key,

cnp\_medic CHAR(13) NOT NULL,

cnp\_pacient CHAR(13) NOT NULL,

data\_prescrierii DATE NOT NULL,

data\_expirarii DATE NOT NULL,

FOREIGN KEY (cnp\_medic) REFERENCES MEDIC(cnp) ON DELETE CASCADE,

FOREIGN KEY (cnp\_pacient) REFERENCES PACIENT(cnp) ON DELETE CASCADE);

CREATE TABLE **CONTRACT**(

id\_contract NUMBER(6) primary key,

id\_farmacie NUMBER(6) NOT NULL,

id\_companie NUMBER(6) NOT NULL,

data\_semnarii DATE NOT NULL,

data\_expirarii DATE NOT NULL,

FOREIGN KEY (id\_farmacie) REFERENCES FARMACIE(id\_farmacie) ON DELETE CASCADE,

FOREIGN KEY (id\_companie) REFERENCES COMPANIE\_FARMACEUTICA(id\_companie) ON DELETE CASCADE);

CREATE TABLE **INREGISTRARE\_MEDICAMENT**(

id\_farmacie NUMBER(6) NOT NULL,

id\_medicament NUMBER(6) NOT NULL,

pret NUMBER(6) NOT NULL,

stoc NUMBER(6) NOT NULL,

PRIMARY KEY (id\_farmacie, id\_medicament),

FOREIGN KEY (id\_farmacie) REFERENCES FARMACIE(id\_farmacie) ON DELETE CASCADE,

FOREIGN KEY (id\_medicament) REFERENCES MEDICAMENT(id\_medicament) ON DELETE CASCADE);

CREATE TABLE **ISTORIC\_DE\_LUCRU**(

id\_istoric NUMBER(6) primary key,

id\_farmacie NUMBER(6) NOT NULL,

cnp\_angajat CHAR(13) NOT NULL,

salariu NUMBER(6) NOT NULL,

end\_date DATE,

start\_date DATE NOT NULL,

FOREIGN KEY (id\_farmacie) REFERENCES FARMACIE(id\_farmacie) ON DELETE CASCADE,

FOREIGN KEY (cnp\_angajat) REFERENCES ANGAJAT(cnp) ON DELETE CASCADE);

CREATE TABLE **PRESCRIERE**(

id\_prescriere NUMBER(6) primary key,

id\_reteta NUMBER(6) NOT NULL,

id\_medicament NUMBER(6) NOT NULL,

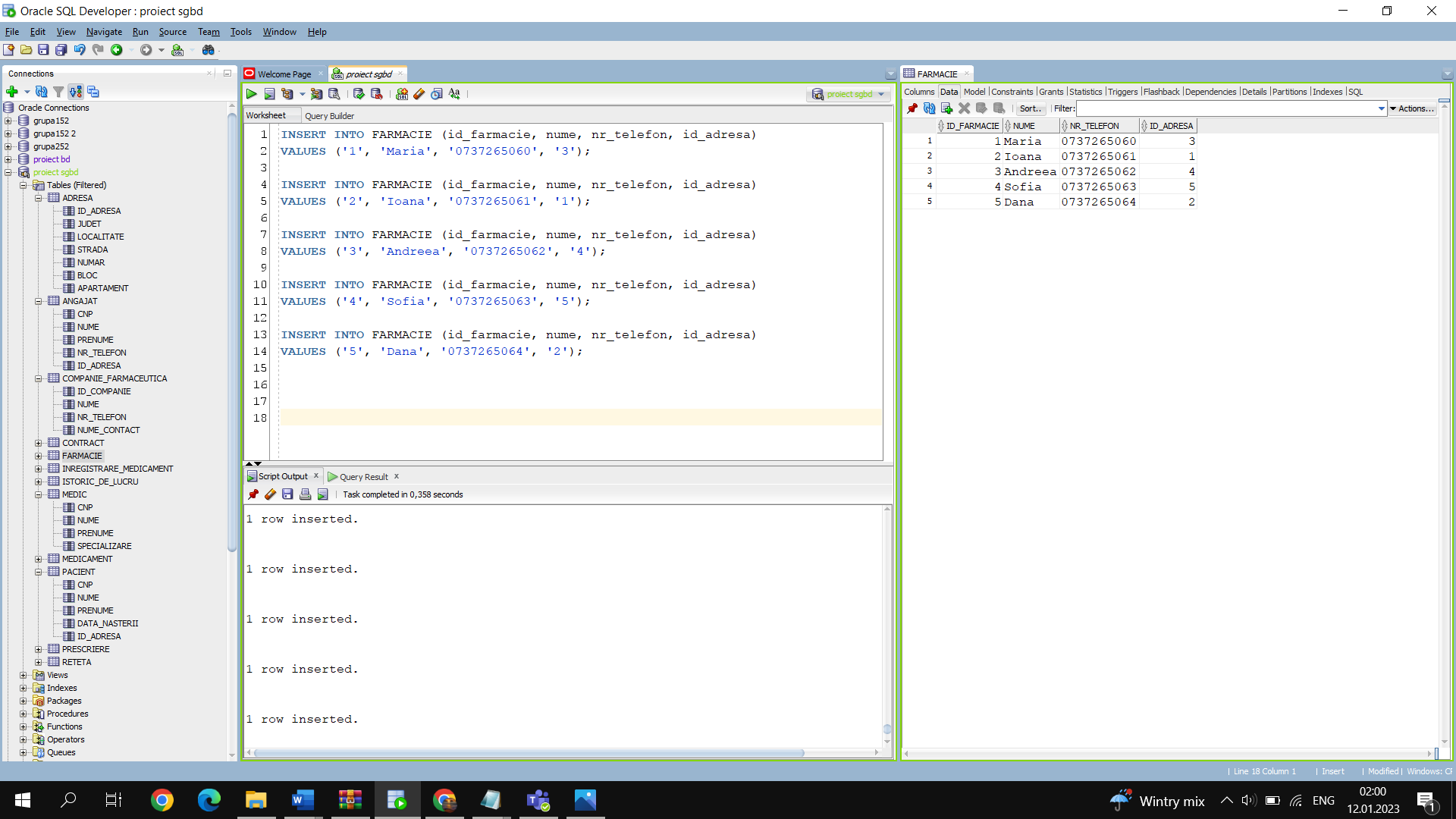
DS VARCHAR2(50),

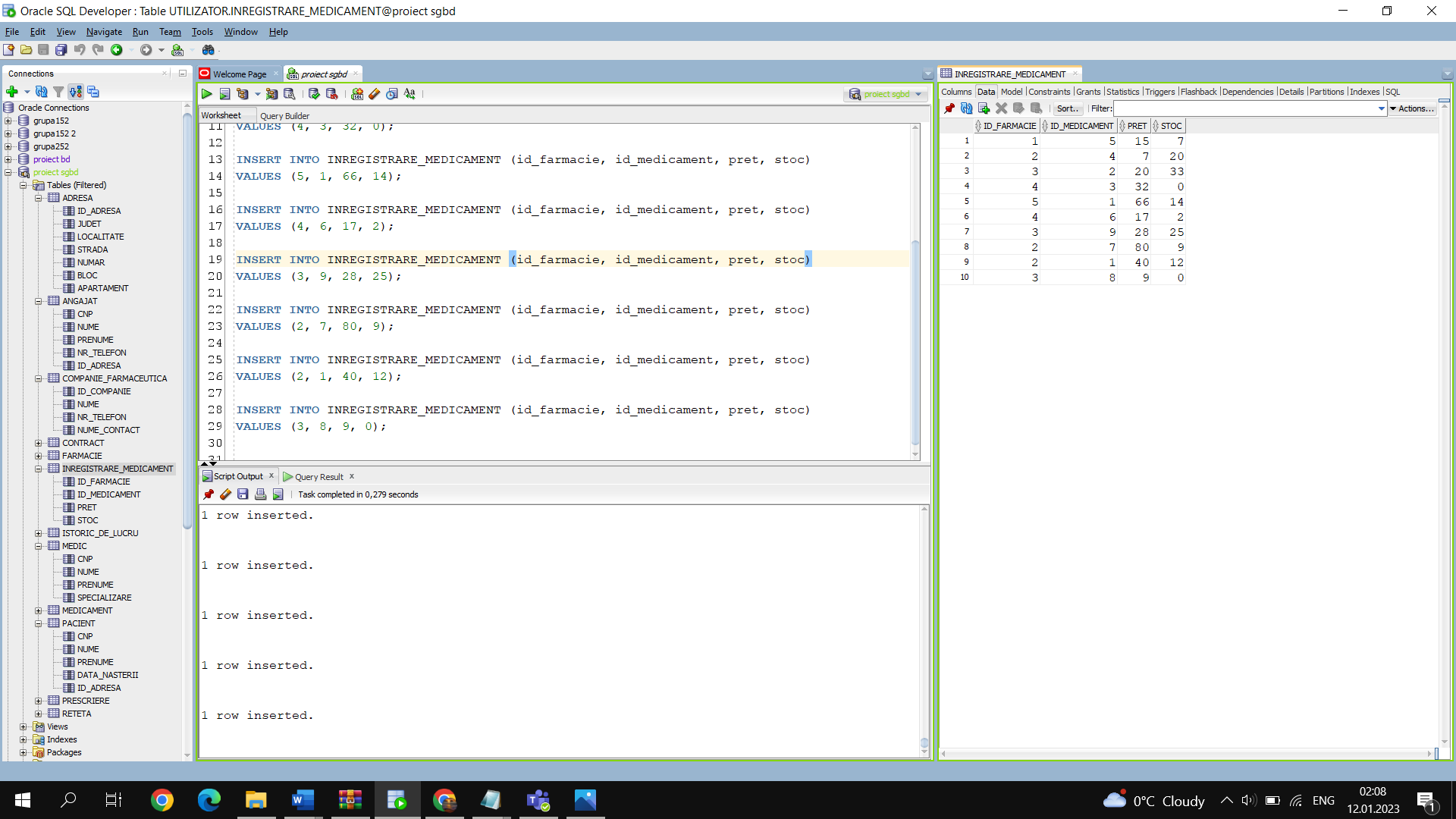
cantitate NUMBER(6) NOT NULL,

FOREIGN KEY (id\_reteta) REFERENCES RETETA(id\_reteta) ON DELETE CASCADE,

FOREIGN KEY (id\_medicament) REFERENCES MEDICAMENT(id\_medicament) ON DELETE CASCADE);

**5. Înregistrări**





INSERT INTO ADRESA (id\_adresa, judet, localitate, strada, numar)

VALUES ('1', 'Prahova', 'Ploiesti', 'Aleea Zanelor', '42');

INSERT INTO ADRESA (id\_adresa, judet, localitate, strada, numar)

VALUES ('2', 'Suceava', 'Falticeni', 'Margaretelor', '18');

INSERT INTO ADRESA (id\_adresa, judet, localitate, strada, numar)

VALUES ('3', 'Brasov', 'Codlea', 'Inchisorii', '11');

INSERT INTO ADRESA (id\_adresa, judet, localitate, strada, numar)

VALUES ('4', 'Braila', 'Ciresu', 'Panselutelor', '24');

INSERT INTO ADRESA (id\_adresa, judet, localitate, strada, numar)

VALUES ('5', 'Prahova', 'Valenii de Munte', 'Fat-Frumos', '16');

INSERT INTO FARMACIE (id\_farmacie, nume, nr\_telefon, id\_adresa)

VALUES ('1', 'Maria', '0737265060', '3');

INSERT INTO FARMACIE (id\_farmacie, nume, nr\_telefon, id\_adresa)

VALUES ('2', 'Ioana', '0737265061', '1');

INSERT INTO FARMACIE (id\_farmacie, nume, nr\_telefon, id\_adresa)

VALUES ('3', 'Andreea', '0737265062', '4');

INSERT INTO FARMACIE (id\_farmacie, nume, nr\_telefon, id\_adresa)

VALUES ('4', 'Sofia', '0737265063', '5');

INSERT INTO FARMACIE (id\_farmacie, nume, nr\_telefon, id\_adresa)

VALUES ('5', 'Dana', '0737265064', '2');

INSERT INTO PACIENT (cnp, nume, prenume, data\_nasterii)

VALUES ('1236878800020', 'Mirunescu', 'Miruna', '16-01-2002');

INSERT INTO PACIENT (cnp, nume, prenume, data\_nasterii)

VALUES ('1236878800021', 'Albu', 'Gabriela', '26-07-1976');

INSERT INTO PACIENT (cnp, nume, prenume, data\_nasterii)

VALUES ('1236878800022', 'Popescu', 'Florentina', '22-06-1953');

INSERT INTO PACIENT (cnp, nume, prenume, data\_nasterii)

VALUES ('1236878800023', 'Coada', 'Vlad', '07-11-1952');

INSERT INTO PACIENT (cnp, nume, prenume, data\_nasterii)

VALUES ('1236878800024', 'Dunareanu', 'Matei', '23-11-1981');

INSERT INTO MEDIC (cnp, nume, prenume, specializare)

VALUES ('3216168800021', 'Ionescu', 'Ion', 'cardiolog');

INSERT INTO MEDIC (cnp, nume, prenume, specializare)

VALUES ('3216168800022', 'Stefanescu', 'Stefania', 'stomaolog');

INSERT INTO MEDIC (cnp, nume, prenume, specializare)

VALUES ('3216168800023', 'Sorescu', 'Sorana', 'oftalmmolog');

INSERT INTO MEDIC (cnp, nume, prenume, specializare)

VALUES ('3216168800024', 'Cristescu', 'Cristian', 'dermatolog');

INSERT INTO MEDIC (cnp, nume, prenume, specializare)

VALUES ('3216168800025', 'Alexandrescu', 'Alexia', 'cardiolog');

INSERT INTO ANGAJAT (cnp, nume, prenume, nr\_telefon)

VALUES ('9236168847021', 'Deleanu', 'Elisabeta', '0720530902');

INSERT INTO ANGAJAT (cnp, nume, prenume, nr\_telefon)

VALUES ('9236168847022', 'Marinescu', 'Marina', '0720530903');

INSERT INTO ANGAJAT (cnp, nume, prenume, nr\_telefon)

VALUES ('9236168847023', 'Popescu', 'Andrei', '0720530904');

INSERT INTO ANGAJAT (cnp, nume, prenume, nr\_telefon)

VALUES ('9236168847024', 'Pop', 'Alin', '0720530905');

INSERT INTO ANGAJAT (cnp, nume, prenume, nr\_telefon)

VALUES ('9236168847025', 'Olteanu', 'Stefan', '0720530906');

INSERT INTO COMPANIE\_FARMACEUTICA (id\_companie, nume, nr\_telefon, nume\_contact)

VALUES ('1', 'LABORMED PHARMA S.A.', '0722171360', 'Andreescu Andrei');

INSERT INTO COMPANIE\_FARMACEUTICA (id\_companie, nume, nr\_telefon, nume\_contact)

VALUES ('2', 'TERAPIA S.A.', '0722171361', 'Mircescu Mircea');

INSERT INTO COMPANIE\_FARMACEUTICA (id\_companie, nume, nr\_telefon, nume\_contact)

VALUES ('3', 'GEDEON RICHTER ROMANIA S.A.', '0722171362', 'Paulescu Paul');

INSERT INTO COMPANIE\_FARMACEUTICA (id\_companie, nume, nr\_telefon, nume\_contact)

VALUES ('4', 'ZENTIVA S.A.', '0722171363', 'Stefanescu Stefan');

INSERT INTO COMPANIE\_FARMACEUTICA (id\_companie, nume, nr\_telefon, nume\_contact)

VALUES ('5', 'BIOFARM S.A.', '0722171364', 'Ionescu Ion');

INSERT INTO MEDICAMENT (id\_medicament, nume, DCI, forma, id\_companie)

VALUES ('1', 'TANTUM VERDE CU AROMA DE LAMAIE 3 mg', 'BENYDAMINUM', 'PASTILE', '3');

INSERT INTO MEDICAMENT (id\_medicament, nume, DCI, forma, id\_companie)

VALUES ('2', 'ZONTIVITY 2mg', 'VORAPAXAR', 'COMPR. FILM.', '2');

INSERT INTO MEDICAMENT (id\_medicament, nume, DCI, forma, id\_companie)

VALUES ('3', 'FAMODIN 40', 'FAMOTIDINUM', 'COMPR. FILM.', '5');

INSERT INTO MEDICAMENT (id\_medicament, nume, DCI, forma, id\_companie)

VALUES ('4', 'CIMETIDINA ARENA 200 mg', 'CIMETIDINUM', 'CAPS', '4');

INSERT INTO MEDICAMENT (id\_medicament, nume, DCI, forma, id\_companie)

VALUES ('5', 'ORTANOL 20 mg', 'OMEPRAZOLUM', 'CAPS. GASTROREZ.', '1');

INSERT INTO MEDICAMENT (id\_medicament, nume, DCI, forma, id\_companie)

VALUES ('6', 'ORTANOL 40 mg', 'OMEPRAZOLUM', 'CAPS. GASTROREZ.', '3');

INSERT INTO MEDICAMENT (id\_medicament, nume, DCI, forma, id\_companie)

VALUES ('7', 'VISIPAQUE 320 mg I/ml', 'IODIXANOLUM', 'SOL.INJ.', '2');

INSERT INTO MEDICAMENT (id\_medicament, nume, DCI, forma, id\_companie)

VALUES ('8', 'ANTISPASMIN FORTE 80 mg', 'DROTAVERINUM', 'COMPR.', '5');

INSERT INTO MEDICAMENT (id\_medicament, nume, DCI, forma, id\_companie)

VALUES ('9', 'PAPAVERINA ARENA 50 mg', 'PAPAVERINI HYDROCHLORIDUM', 'COMPR.', '3');

INSERT INTO INREGISTRARE\_MEDICAMENT (id\_farmacie, id\_medicament, pret, stoc)

VALUES (1, 5, 15, 7);

INSERT INTO INREGISTRARE\_MEDICAMENT (id\_farmacie, id\_medicament, pret, stoc)

VALUES (2, 4, 7, 20);

INSERT INTO INREGISTRARE\_MEDICAMENT (id\_farmacie, id\_medicament, pret, stoc)

VALUES (3, 2, 20, 33);

INSERT INTO INREGISTRARE\_MEDICAMENT (id\_farmacie, id\_medicament, pret, stoc)

VALUES (4, 3, 32, 0);

INSERT INTO INREGISTRARE\_MEDICAMENT (id\_farmacie, id\_medicament, pret, stoc)

VALUES (5, 1, 66, 14);

INSERT INTO INREGISTRARE\_MEDICAMENT (id\_farmacie, id\_medicament, pret, stoc)

VALUES (4, 6, 17, 2);

INSERT INTO INREGISTRARE\_MEDICAMENT (id\_farmacie, id\_medicament, pret, stoc)

VALUES (3, 9, 28, 25);

INSERT INTO INREGISTRARE\_MEDICAMENT (id\_farmacie, id\_medicament, pret, stoc)

VALUES (2, 7, 80, 9);

INSERT INTO INREGISTRARE\_MEDICAMENT (id\_farmacie, id\_medicament, pret, stoc)

VALUES (2, 1, 40, 12);

INSERT INTO INREGISTRARE\_MEDICAMENT (id\_farmacie, id\_medicament, pret, stoc)

VALUES (3, 8, 9, 0);

INSERT INTO RETETA (id\_reteta, cnp\_medic, cnp\_pacient, data\_prescrierii, data\_expirarii)

VALUES (1, '3216168800024', '1236878800022', '16-01-2022', '16-02-2022');

INSERT INTO RETETA (id\_reteta, cnp\_medic, cnp\_pacient, data\_prescrierii, data\_expirarii)

VALUES (2, '3216168800021', '1236878800024', '24-04-2022', '7-05-2022');

INSERT INTO RETETA (id\_reteta, cnp\_medic, cnp\_pacient, data\_prescrierii, data\_expirarii)

VALUES (3, '3216168800025', '1236878800023', '20-12-2022', '20-01-2023');

INSERT INTO RETETA (id\_reteta, cnp\_medic, cnp\_pacient, data\_prescrierii, data\_expirarii)

VALUES (4, '3216168800023', '1236878800022', '10-01-2023', '10-03-2023');

INSERT INTO RETETA (id\_reteta, cnp\_medic, cnp\_pacient, data\_prescrierii, data\_expirarii)

VALUES (5, '3216168800022', '1236878800020', '25-05-2022', '05-06-2022');

INSERT INTO RETETA (id\_reteta, cnp\_medic, cnp\_pacient, data\_prescrierii, data\_expirarii)

VALUES (6, '3216168800022', '1236878800020', '10-11-2022', '20-11-2022');

INSERT INTO PRESCRIERE (id\_prescriere, id\_reteta, id\_medicament, DS, cantitate)

VALUES (1, 1, 5, '1CP/ZI', 10);

INSERT INTO PRESCRIERE (id\_prescriere, id\_reteta, id\_medicament, DS, cantitate)

VALUES (2, 1, 2, '2CP/ZI', 15);

INSERT INTO PRESCRIERE (id\_prescriere, id\_reteta, id\_medicament, DS, cantitate)

VALUES (3, 2, 3, '2CP/ZI', 15);

INSERT INTO PRESCRIERE (id\_prescriere, id\_reteta, id\_medicament, cantitate)

VALUES (4, 2, 6, 1);

INSERT INTO PRESCRIERE (id\_prescriere, id\_reteta, id\_medicament, DS, cantitate)

VALUES (5, 2, 4, '3CP/ZI', 20);

INSERT INTO PRESCRIERE (id\_prescriere, id\_reteta, id\_medicament, DS, cantitate)

VALUES (6, 3, 1, '2CP/ZI', 35);

INSERT INTO PRESCRIERE (id\_prescriere, id\_reteta, id\_medicament, cantitate)

VALUES (7, 4, 6, 2);

INSERT INTO PRESCRIERE (id\_prescriere, id\_reteta, id\_medicament, DS, cantitate)

VALUES (8, 4, 5, '2CP/ZI', 18);

INSERT INTO PRESCRIERE (id\_prescriere, id\_reteta, id\_medicament, DS, cantitate)

VALUES (9, 4, 2, '2CP/ZI', 10);

INSERT INTO PRESCRIERE (id\_prescriere, id\_reteta, id\_medicament, DS, cantitate)

VALUES (10, 5, 3, '3CP/ZI', 18);

INSERT INTO CONTRACT (id\_contract, id\_farmacie, id\_companie, data\_semnarii, data\_expirarii) VALUES (1, 1, 4, '16-01-2022', '16-05-2023');

INSERT INTO CONTRACT (id\_contract, id\_farmacie, id\_companie, data\_semnarii, data\_expirarii) VALUES (2, 3, 3, '24-04-2022', '24-04-2025');

INSERT INTO CONTRACT (id\_contract, id\_farmacie, id\_companie, data\_semnarii, data\_expirarii) VALUES (3, 2, 1, '20-02-2022', '20-07-2022');

INSERT INTO CONTRACT (id\_contract, id\_farmacie, id\_companie, data\_semnarii, data\_expirarii) VALUES (4, 4, 2, '20-02-2008', '30-04-2012');

INSERT INTO CONTRACT (id\_contract, id\_farmacie, id\_companie, data\_semnarii, data\_expirarii) VALUES (5, 5, 5, '15-04-2022', '15-04-2023');

INSERT INTO CONTRACT (id\_contract, id\_farmacie, id\_companie, data\_semnarii, data\_expirarii) VALUES (6, 2, 4, '12-01-2023', '12-01-2027');

INSERT INTO ISTORIC\_DE\_LUCRU (id\_istoric, id\_farmacie, cnp\_angajat, salariu, start\_date)

VALUES (1,2,'9236168847023',3003,'10-06-2022');

INSERT INTO ISTORIC\_DE\_LUCRU (id\_istoric, id\_farmacie, cnp\_angajat, salariu, start\_date, end\_date)

VALUES (2,4,'9236168847023',3500,'10-08-2021', '20-09-2021');

INSERT INTO ISTORIC\_DE\_LUCRU (id\_istoric, id\_farmacie, cnp\_angajat, salariu, start\_date, end\_date)

VALUES (3,3,'9236168847022',3400,'10-03-2022', '20-07-2022');

INSERT INTO ISTORIC\_DE\_LUCRU (id\_istoric, id\_farmacie, cnp\_angajat, salariu, start\_date)

VALUES (4,1,'9236168847025',3050,'10-06-2020');

INSERT INTO ISTORIC\_DE\_LUCRU (id\_istoric, id\_farmacie, cnp\_angajat, salariu, start\_date, end\_date)

VALUES (5,5,'9236168847023',4000,'10-06-2018', '20-06-2020');

INSERT INTO ISTORIC\_DE\_LUCRU (id\_istoric, id\_farmacie, cnp\_angajat, salariu, start\_date)

VALUES (6,4,'9236168847024',3650,'5-01-2023');

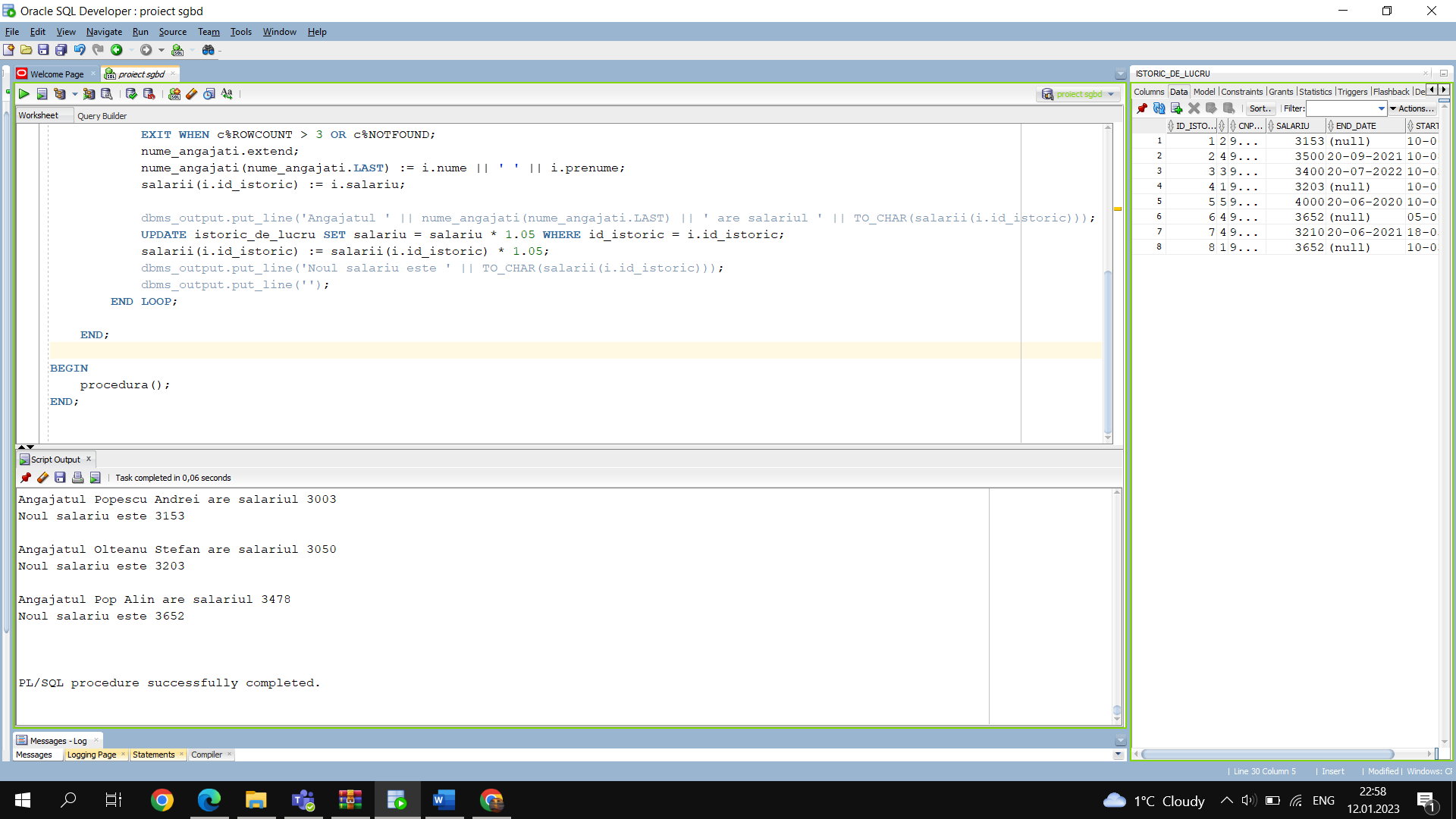
INSERT INTO ISTORIC\_DE\_LUCRU (id\_istoric, id\_farmacie, cnp\_angajat, salariu, start\_date, end\_date) VALUES (7,4,'9236168847024',3210, '18-05-2019', '20-06-2021');

INSERT INTO ISTORIC\_DE\_LUCRU (id\_istoric, id\_farmacie, cnp\_angajat, salariu, start\_date)

VALUES (8,1,'9236168847021',3478, '10-03-2022');

**6. Utilizarea a doua tipuri de colecție studiate**

Determin primii 3 angajați care au cel mai mic salariu și le măresc salariul cu 5%. Afișez numele lor, salariul vechi și salariul nou.



**CREATE OR REPLACE PROCEDURE** procedura

**AS**

TYPE tablou\_indexat IS TABLE OF istoric\_de\_lucru.salariu%type INDEX BY VARCHAR2(13);

salarii tablou\_indexat;

TYPE tablou\_imbricat IS TABLE OF angajat.nume%type;

nume\_angajati tablou\_imbricat := tablou\_imbricat();

CURSOR c IS

SELECT nume, prenume, id\_istoric, salariu

FROM istoric\_de\_lucru i, angajat a

WHERE i.cnp\_angajat = a.cnp AND i.end\_date IS NULL

ORDER BY salariu;

**BEGIN**

FOR i in c LOOP

EXIT WHEN c%ROWCOUNT > 3 OR c%NOTFOUND;

nume\_angajati.extend;

nume\_angajati(nume\_angajati.LAST) := i.nume || ' ' || i.prenume;

salarii(i.id\_istoric) := i.salariu;

dbms\_output.put\_line('Angajatul ' || nume\_angajati(nume\_angajati.LAST) || ' are salariul ' || TO\_CHAR(salarii(i.id\_istoric)));

UPDATE istoric\_de\_lucru SET salariu = salariu \* 1.05 WHERE id\_istoric = i.id\_istoric;

salarii(i.id\_istoric) := salarii(i.id\_istoric) \* 1.05;

dbms\_output.put\_line('Noul salariu este ' || TO\_CHAR(salarii(i.id\_istoric)));

dbms\_output.put\_line('');

END LOOP;

**END**;

/

**BEGIN**

procedura();

**END;**

**7. Utilizarea a doua tipuri de cursoare studiate, dintre care unul paramerizat**

Pentru fiecare farmacie din județul Suceava, afișez stocul medicamentelor produse de compania farmaceutica cu id-ul 3;

**CREATE OR REPLACE PROCEDURE** procedura2

**AS**

med medicament.nume%type;

stoc inregistrare\_medicament.stoc%type;

CURSOR c (parametru NUMBER) IS

SELECT nume, stoc

FROM medicament m

JOIN inregistrare\_medicament i ON m.id\_medicament = i.id\_medicament

WHERE i.id\_farmacie = parametru and m.id\_companie = 3;

**BEGIN**

FOR farmacie in (SELECT id\_farmacie id, nume

FROM farmacie f, adresa a

WHERE f.id\_adresa = a.id\_adresa and a.judet = 'Suceava')

**LOOP**

dbms\_output.put\_line('Farmacia ' || farmacie.nume || ':');

OPEN c(farmacie.id);

**LOOP**

FETCH c INTO med, stoc;

EXIT WHEN c%NOTFOUND;

CASE WHEN stoc = 0

THEN dbms\_output.put\_line(' ' || med || ' NU MAI ESTE IN STOC');

ELSE dbms\_output.put\_line(' ' || med || ' - ' || stoc || ' bucati');

END CASE;

**END LOOP**;

CLOSE c;

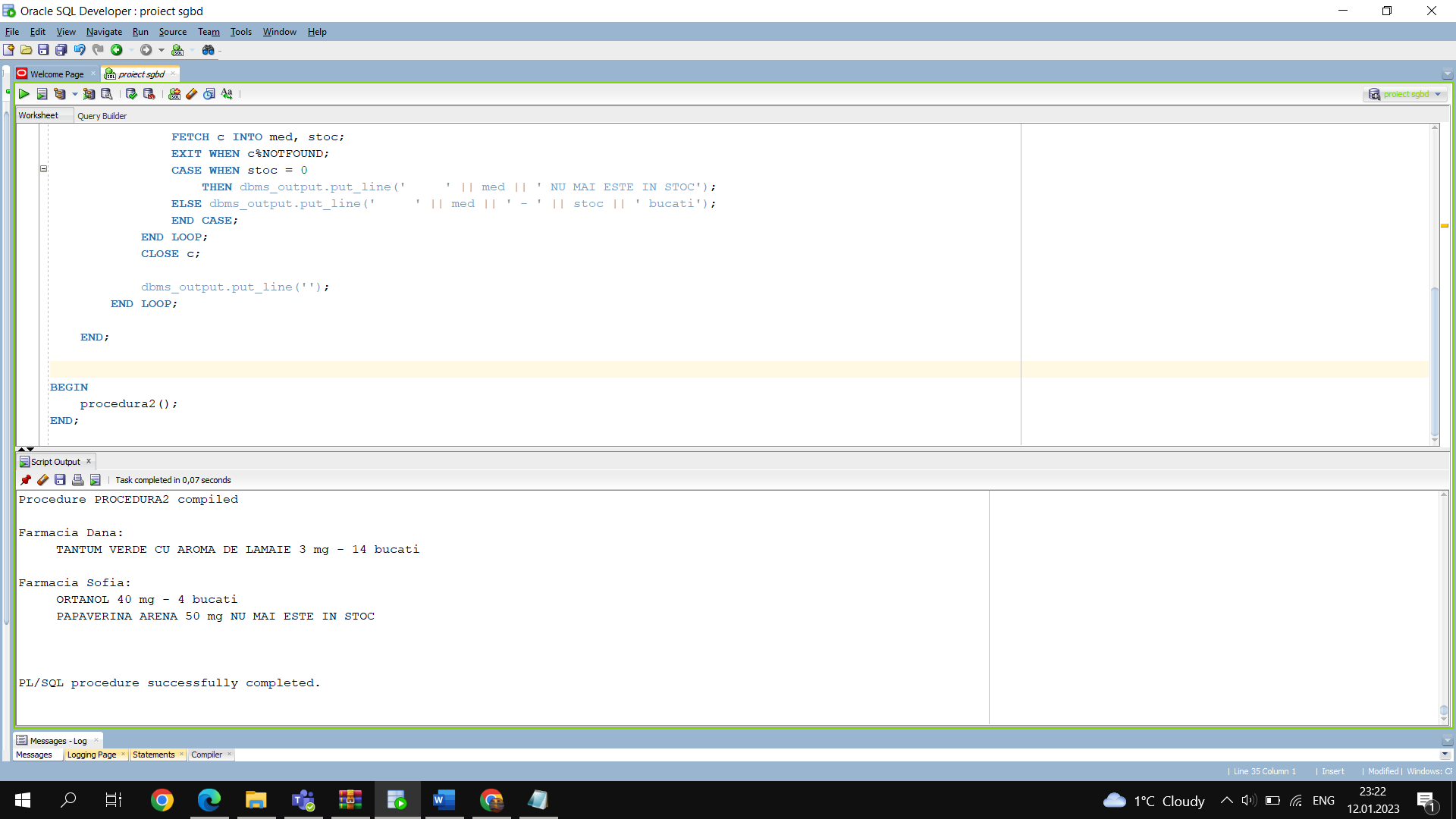
dbms\_output.put\_line('');

**END LOOP**;

**END**;

/

**BEGIN** procedura2(); **END;**



**8. Subprogram de tip funcție, utilizând 3 tabele într-o singură comandă SQL și având definite 3 excepții**

Funcția primește ca parametru id-ul unui medicament și numele unei localități și returneaza id-ul unei farmacii din acea localitate la care se poate găsi acel medicament.

**CREATE OR REPLACE FUNCTION** gaseste\_farmacia

(v\_loc adresa.localitate%type,

v\_id\_medicament medicament.id\_medicament%type)

**RETURN** farmacie.id\_farmacie%type

**IS**

localitate\_inexistenta EXCEPTION;

id\_inexistent EXCEPTION;

farmacie\_inexistenta EXCEPTION;

id\_farm farmacie.id\_farmacie%type := -1;

contor INT := 0;

no\_id INT;

no\_loc INT;

**BEGIN**

SELECT COUNT(\*) INTO no\_loc FROM adresa WHERE localitate = v\_loc;

IF no\_loc = 0 THEN RAISE localitate\_inexistenta;

END IF;

SELECT COUNT(\*) INTO no\_id FROM medicament WHERE id\_medicament = v\_id\_medicament;

IF no\_id = 0 THEN RAISE id\_inexistent;

END IF;

-- folosesc tabelele adresa, farmacie, inregistrare\_medicament

FOR i IN (SELECT f.id\_farmacie id

FROM farmacie f

JOIN inregistrare\_medicament inreg ON f.id\_farmacie = inreg.id\_farmacie

JOIN adresa a ON a.id\_adresa = f.id\_adresa

WHERE inreg.id\_medicament = v\_id\_medicament and a.localitate = v\_loc)

LOOP

contor := contor + 1;

id\_farm := i.id;

EXIT;

END LOOP;

IF contor = 0 THEN RAISE farmacie\_inexistenta;

END IF;

RETURN id\_farm;

**EXCEPTION**

WHEN localitate\_inexistenta THEN

dbms\_output.put\_line('Localiatea nu exista in baza de date');

RETURN null;

WHEN id\_inexistent THEN

dbms\_output.put\_line('Id-ul medicamentului nu exista in baza de date');

RETURN null;

WHEN farmacie\_inexistenta THEN

dbms\_output.put\_line('Medicamentul nu se gaseste la nicio farmacie din aceasta localitate');

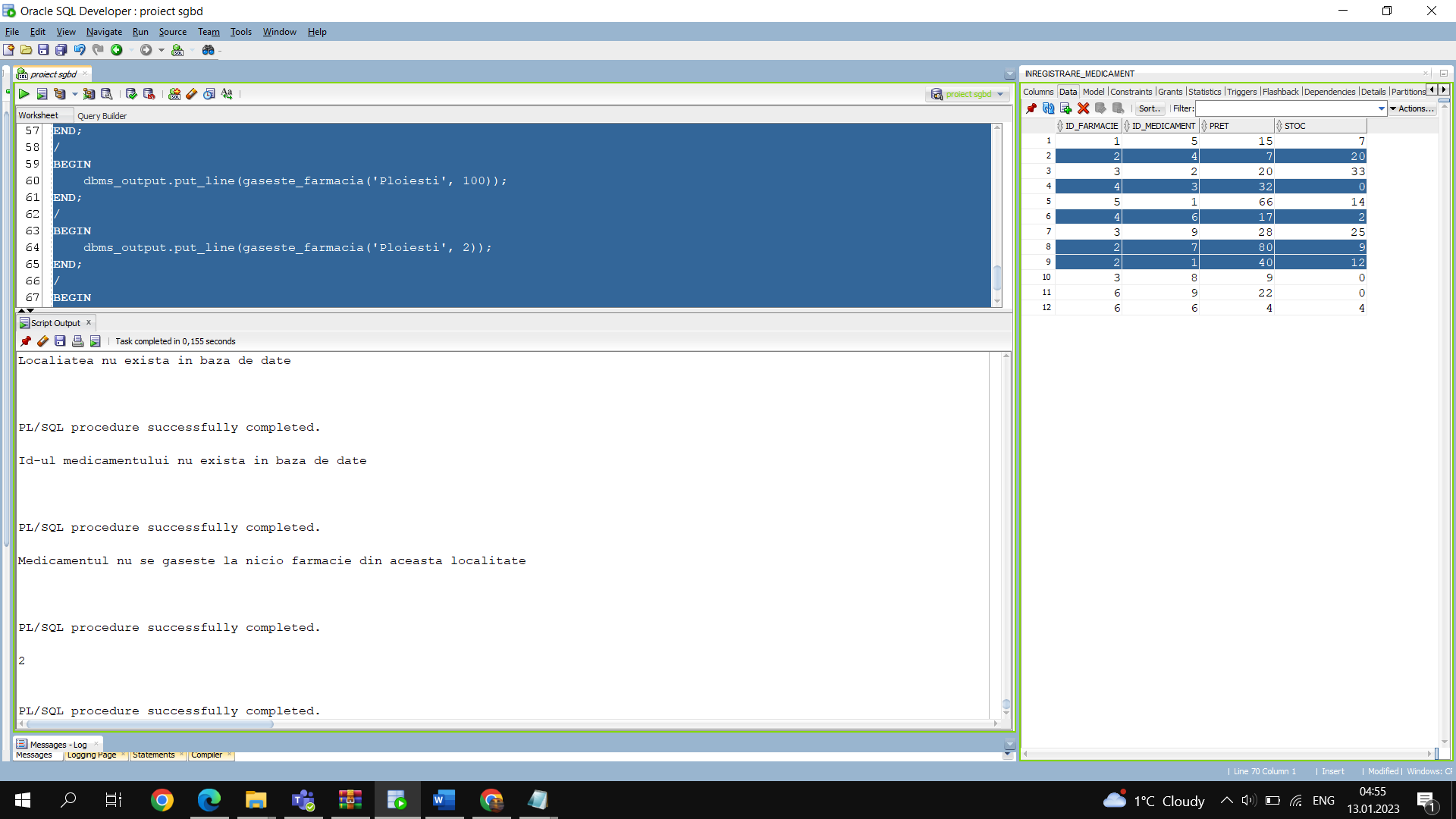
RETURN null;

WHEN OTHERS THEN

dbms\_output.put\_line('Eroare!');

RETURN null;

**END** gaseste\_farmacia;



**BEGIN**

dbms\_output.put\_line(gaseste\_farmacia('Localitate', 4));

**END**;

/

**BEGIN**

dbms\_output.put\_line(gaseste\_farmacia('Ploiesti', 100));

**END**;

/

**BEGIN**

dbms\_output.put\_line(gaseste\_farmacia('Ploiesti', 2));

**END**;

/

**BEGIN**

dbms\_output.put\_line(gaseste\_farmacia('Ploiesti', 4));

**END**;

**9. Subprogram de tip procedura, utilizând 5 tabele într-o singură comandă SQL și având trtatând posibilele exceptii.**

Afișez toate prescrierile de pe rețetele date de medicul cu numele X înainte de data Y, cât și numele pacienților corespunzători.

**CREATE OR REPLACE PROCEDURE** procedura3

(v\_nume medic.nume%type, v\_data reteta.data\_prescrierii%type)

**AS**

CURSOR c IS

SELECT m.nume as medicament, NVL(p.ds, 'neprecizat') as doza, CONCAT(p1.nume, CONCAT(' ', p1.prenume)) as pacient

FROM prescriere p

JOIN medicament m ON m.id\_medicament = p.id\_medicament

JOIN reteta r ON p.id\_reteta = r.id\_reteta

JOIN medic m1 ON m1.cnp = r.cnp\_medic

JOIN pacient p1 ON p1.cnp = r.cnp\_pacient

WHERE LOWER(m1.nume) = LOWER(v\_nume) AND r.data\_prescrierii <= v\_data;

medic\_inexistent EXCEPTION;

mai\_multi\_medici EXCEPTION;

prescrieri\_inexistente EXCEPTION;

nr\_medici INT;

nr\_prescrieri INT := 0;

**BEGIN**

SELECT COUNT(\*) INTO nr\_medici FROM medic WHERE LOWER(nume) = LOWER(v\_nume);

IF nr\_medici = 0 THEN RAISE medic\_inexistent;

ELSIF nr\_medici > 1 THEN RAISE mai\_multi\_medici;

END IF;

FOR i IN c LOOP

EXIT WHEN c%NOTFOUND;

dbms\_output.put\_line(i.pacient || ', ' || i.medicament || ', ' || i.doza);

nr\_prescrieri := nr\_prescrieri + 1;

END LOOP;

IF nr\_prescrieri = 0 THEN RAISE prescrieri\_inexistente;

END IF;

**EXCEPTION**

WHEN prescrieri\_inexistente THEN dbms\_output.put\_line('Nu exista nicio prescriere.');

WHEN medic\_inexistent THEN dbms\_output.put\_line('Nu exista un medic cu acel nume.');

WHEN mai\_multi\_medici THEN dbms\_output.put\_line('Exista mai multi medici cu acel nume.');

WHEN OTHERS THEN dbms\_output.put\_line('Eroare!');

**END** procedura3;

/

**BEGIN** procedura3('Ionescu', '30-04-2022'); **END**;

/

**BEGIN** procedura3('', '30-04-2022'); **END**;

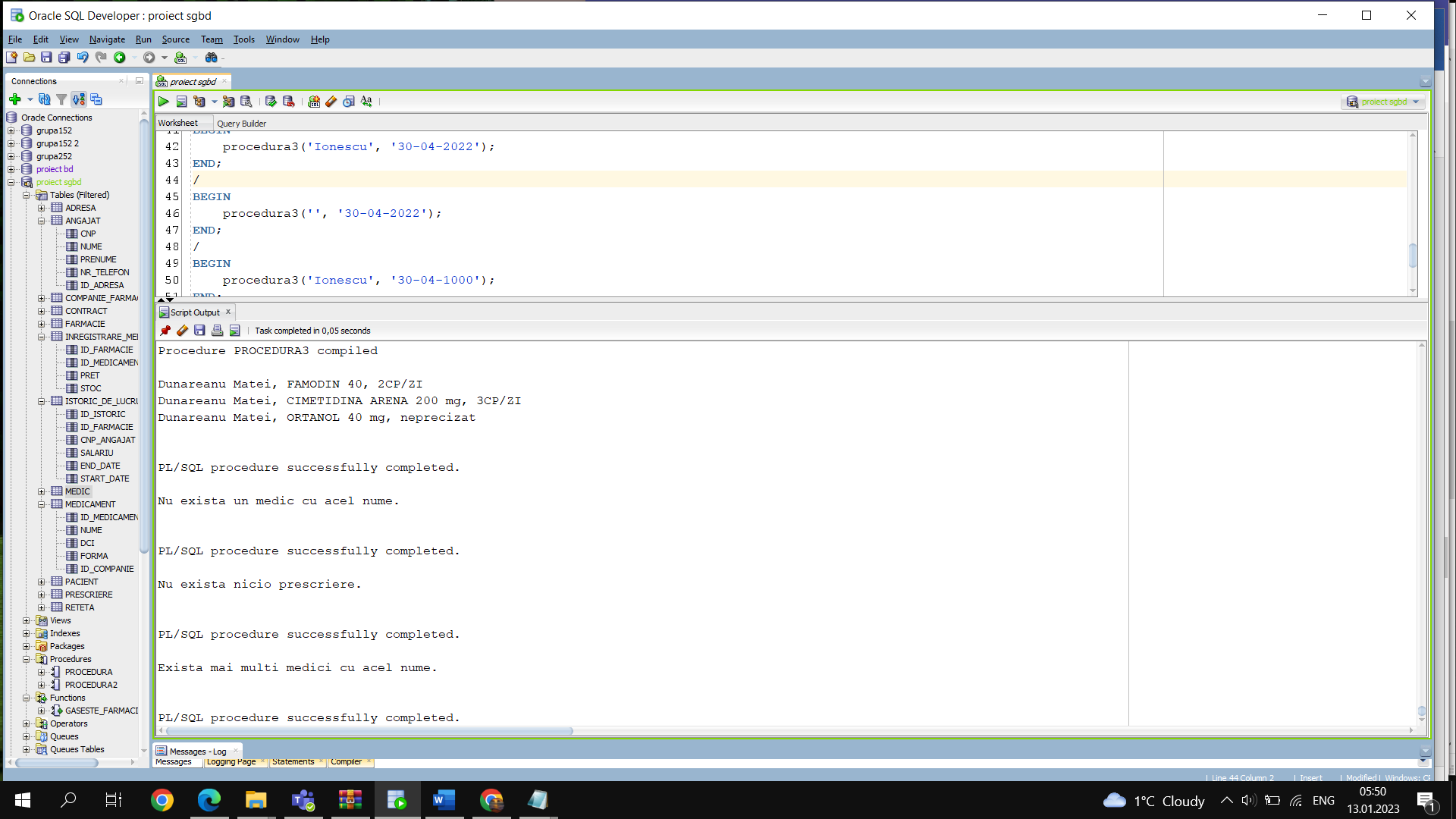
/

**BEGIN** procedura3('Ionescu', '30-04-1000'); **END**;

/

**BEGIN** procedura3('Alexandrescu', '30-04-2022'); **END**;

/



**9. Trigger de tip LMD la nivel de comanda**

Triggerul nu permite inserarea și modificarea unui contract într-o anumită zi.

**CREATE OR REPLACE TRIGGER** contract\_neprielnic

BEFORE INSERT OR UPDATE ON contract

**DECLARE**

zi\_saptamana CHAR(10);

zi\_luna CHAR(10);

**BEGIN**

SELECT TO\_CHAR(SYSDATE, 'Day') INTO zi\_saptamana FROM dual;

SELECT TO\_CHAR(SYSDATE, 'dd') INTO zi\_luna FROM dual;

IF zi\_saptamana = 'Vineri' AND zi\_luna = 13

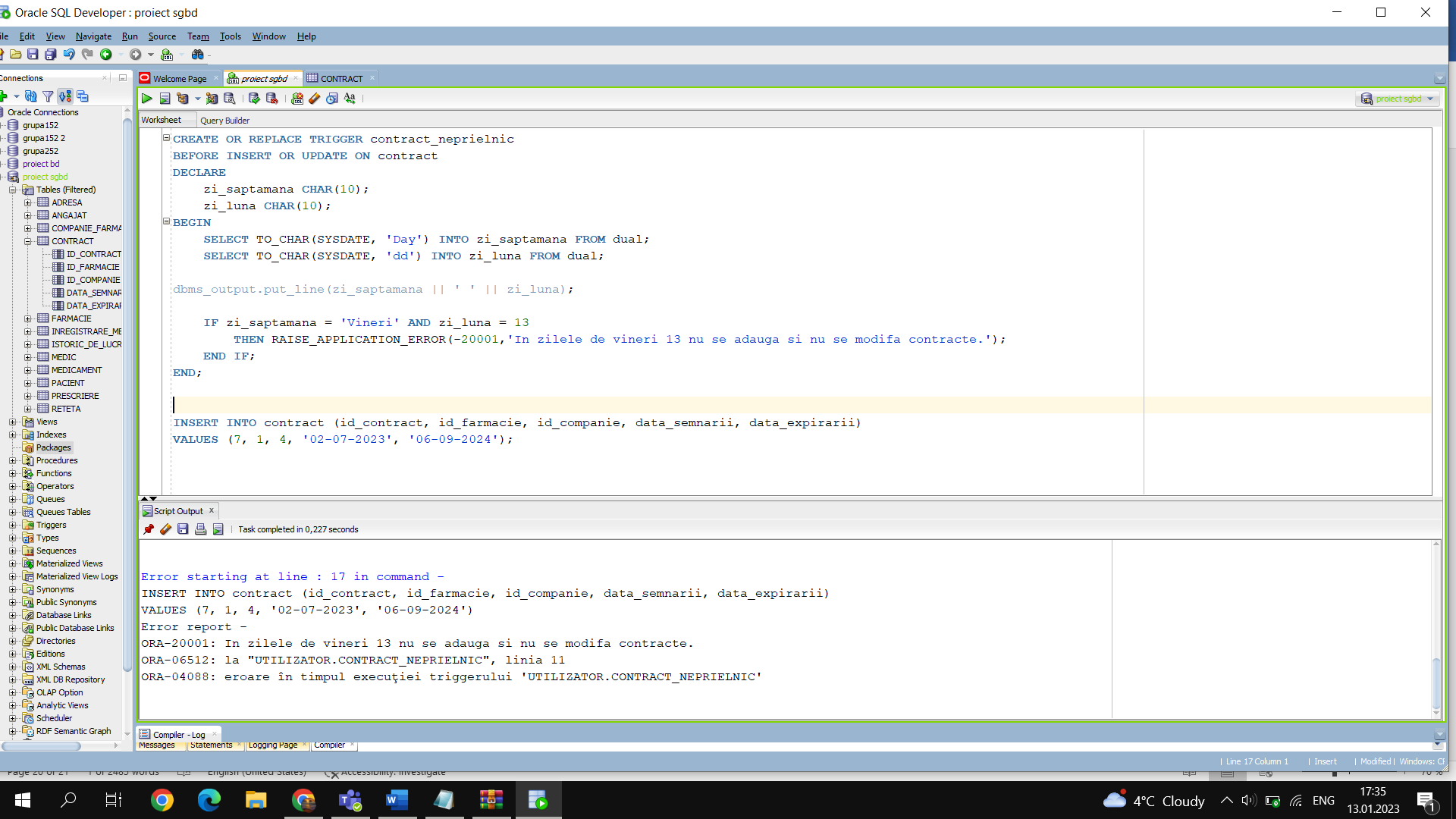
THEN RAISE\_APPLICATION\_ERROR(-20001,'In zilele de vineri 13 nu se adauga si nu se modifa contracte.');

END IF;

**END**;

INSERT INTO contract (id\_contract, id\_farmacie, id\_companie, data\_semnarii, data\_expirarii)

VALUES (7, 1, 4, '02-07-2023', '06-09-2024');



**10. Trigger de tip LMD la nivel de linie**

Triggerul anunță dacă un medicament a ieșit din stoc, a fost readăugat în stoc, sau daca se încearca depășirea stocului acelui medicament într-o farmacie.

**CREATE OR REPLACE TRIGGER** update\_stoc

BEFORE UPDATE ON inregistrare\_medicament

FOR EACH ROW

**BEGIN**

IF :OLD.stoc > 0 and :NEW.stoc = 0

THEN dbms\_output.put\_line('La farmacia cu id-ul ' || :OLD.id\_farmacie || ', stocul de medicamente cu id-ul ' || :OLD.id\_medicament || ' a fost epuizat');

ELSIF :NEW.stoc < 0

THEN RAISE\_APPLICATION\_ERROR(-20001,'Stoc depasit');

ELSIF :OLD.stoc = 0 and :NEW.stoc > 0

THEN dbms\_output.put\_line('La farmacia cu id-ul ' || :OLD.id\_farmacie || ', medicamentul cu id-ul ' || :OLD.id\_medicament || ' a reaparut in stoc');

END IF;

**END**;

**UPDATE** inregistrare\_medicament

SET stoc = 0

WHERE id\_farmacie = 4 and id\_medicament = 6;

**UPDATE** inregistrare\_medicament

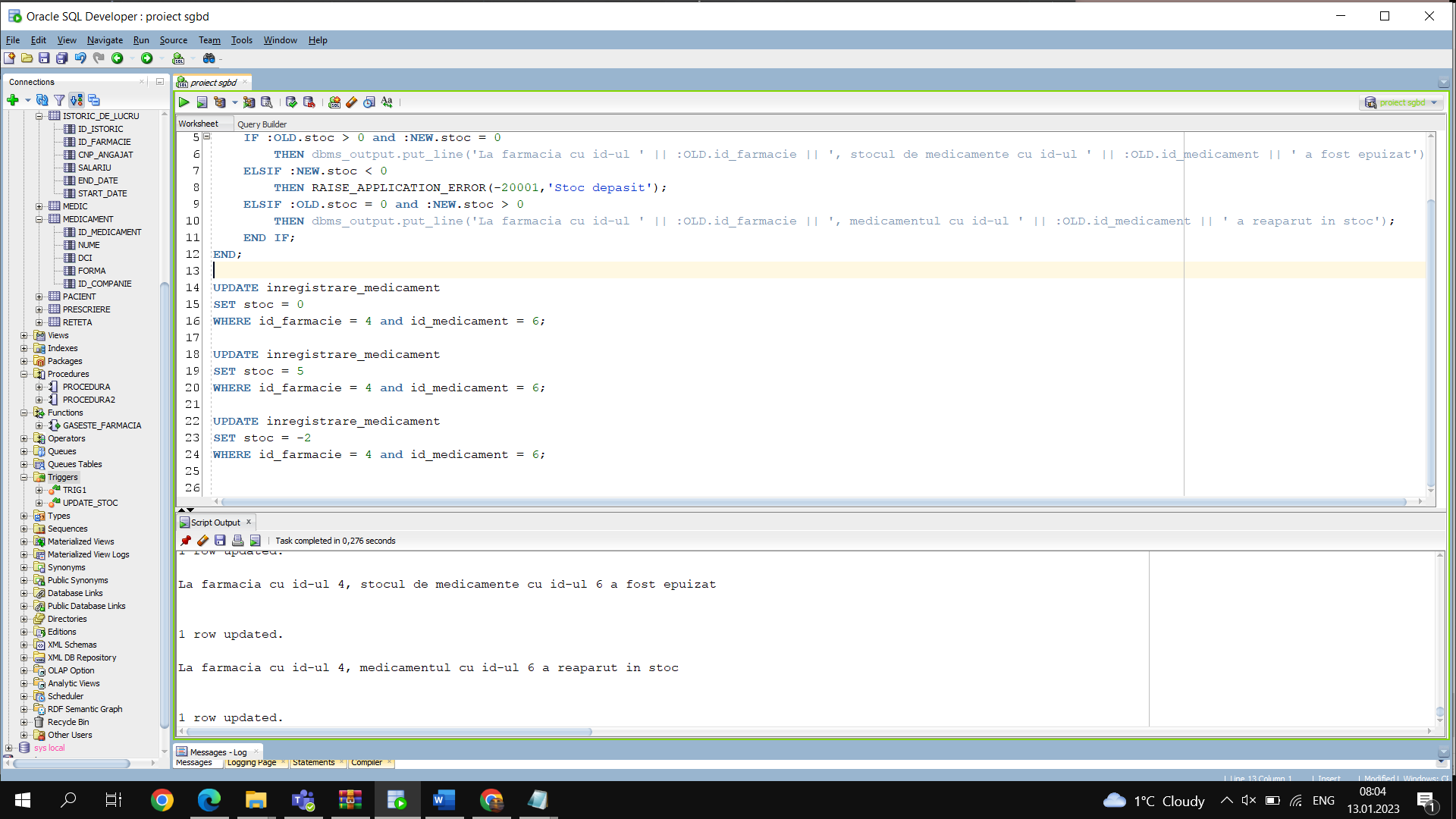
SET stoc = 5

WHERE id\_farmacie = 4 and id\_medicament = 6;

**UPDATE** inregistrare\_medicament

SET stoc = -2

WHERE id\_farmacie = 4 and id\_medicament = 6;



**12. Trigger de tip LDD**

De fiecare data cand se creează un tabel, triggerul adauga într-un tabel cu istoricul creărilor informații despre acea creare (user, data, numele tabelului creat).

**CREATE TABLE** istoric\_creare\_tabele

(

utilizator VARCHAR(30),

data TIMESTAMP(3),

obiect VARCHAR(30)

);

**CREATE OR REPLACE** TRIGGER created\_table

AFTER CREATE ON DATABASE

BEGIN

INSERT INTO istoric\_creare\_tabele

VALUES (SYS.LOGIN\_USER, SYSTIMESTAMP, SYS.DICTIONARY\_OBJ\_NAME);

END;

**CREATE TABLE** tabel ( atribut INT );

