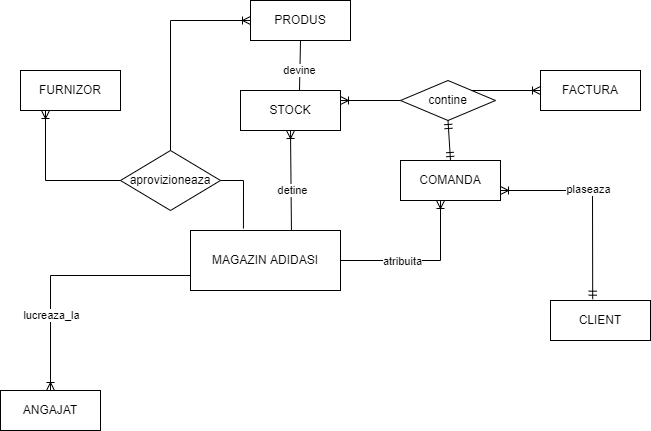
## **PROIECT SGBD**

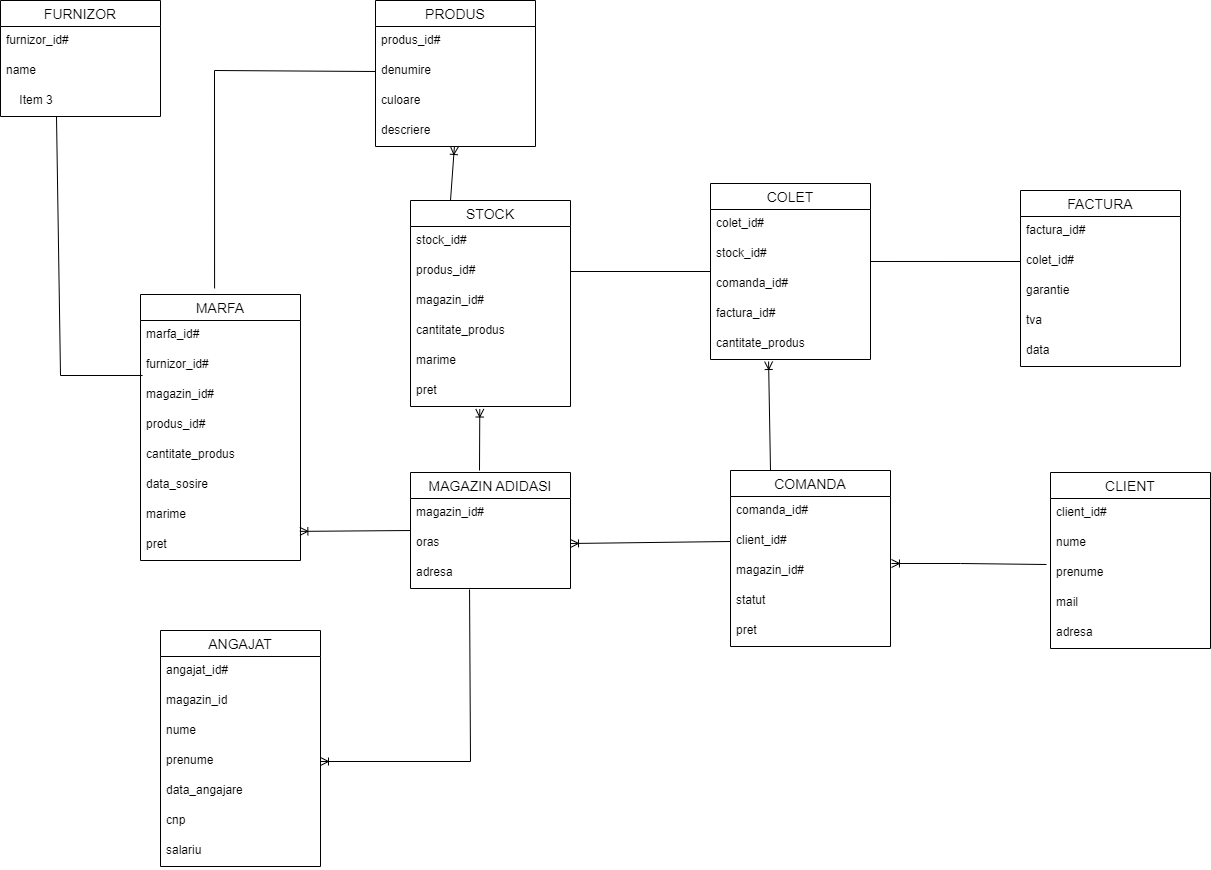
1. Prezentați pe scurt baza de date (utilitatea ei).

Baza de date pentru un lant de magazine fizice care se ocupa cu comercializarea produselor de incaltaminte. Aceste magazine se aprovizioneaza de la mai multi furnizori. Clientii avand optiunea de a plasa comenzi acestea urmad a fi directionate catre cel mai apropiat magazin.

2. Realizați diagrama entitate-relație (ERD).



3. Pornind de la diagrama entitate-relație realizați diagrama conceptuală a modelului propus, integrand toate atributele necesare.



4. Implementați în Oracle diagrama conceptuală realizată: definiți toate tabelele, implementând toate constrângerile de integritate necesare (chei primare, cheile externe etc).

DROP TABLE client;

CREATE TABLE client(

client\_id number(5) PRIMARY KEY,

nume varchar2(15) NOT NULL,

prenume varchar2(15) NOT NULL,

mail varchar2(35) NOT NULL,

adresa varchar2(40) NOT NULL

);

DROP TABLE magazin;

CREATE TABLE magazin(

magazin\_id number(5) PRIMARY KEY,

oras varchar2(15) NOT NULL,

adresa varchar2(40) NOT NULL

);

DROP TABLE produs;

CREATE TABLE produs(

produs\_is number(5) PRIMARY KEY,

denumire varchar2(15) NOT NULL,

culoare varchar2(15) NOT NULL,

descriere VARCHAR2(200)

);

DROP TABLE angajat;

CREATE TABLE angajat (

angajat\_id number(5) PRIMARY KEY,

magazin\_id number(5) NOT NULL,

nume varchar2(15) NOT NULL,

prenume varchar2(15) NOT NULL,

data\_angajare TIMESTAMP NOT NULL,

cnp varchar2(13) NOT NULL,

salariu number(5) NOT NULL,

CONSTRAINT fk\_magazin FOREIGN KEY (magazin\_id) REFERENCES magazin(magazin\_id)

);

DROP TABLE comanda;

CREATE TABLE comanda(

comanda\_id number(5) PRIMARY KEY,

magazin\_id number(5) NOT NULL,

client\_id number(5) NOT NULL,

statut varchar2(25) NOT NULL,

pret number(5) NOT NULL,

CONSTRAINT fk1\_magazin FOREIGN KEY (magazin\_id) REFERENCES magazin(magazin\_id),

CONSTRAINT fk1\_client FOREIGN KEY (client\_id) REFERENCES client(client\_id)

);

DROP TABLE stock;

CREATE TABLE stock(

stock\_id number(5) PRIMARY KEY,

magazin\_id number(5) NOT NULL,

produs\_id number(5) NOT NULL,

cantitate\_produs number(5) NOT NULL,

marime number(3) NOT NULL,

pret number(5) NOT NULL,

CONSTRAINT fk2\_magazin FOREIGN KEY (magazin\_id) REFERENCES magazin(magazin\_id),

CONSTRAINT fk2\_produs FOREIGN KEY (produs\_id) REFERENCES produs(produs\_id)

);

DROP TABLE marfa;

CREATE TABLE marfa(

marfa\_id number(5) PRIMARY KEY,

furnizor\_id number(5) NOT NULL,

magazin\_id number(5) NOT NULL,

produs\_id number(5) NOT NULL,

cantitate\_produs number(5) NOT NULL,

data\_sosire TIMESTAMP NOT NULL,

marime number(3) NOT NULL,

pret number(5) NOT NULL,

CONSTRAINT fk3\_magazin FOREIGN KEY (magazin\_id) REFERENCES magazin(magazin\_id),

CONSTRAINT fk3\_furnizor FOREIGN KEY (furnizor\_id) REFERENCES furnizor(furnizor\_id),

CONSTRAINT fk3\_produs FOREIGN KEY (produs\_id) REFERENCES produs(produs\_id)

);

DROP TABLE factura;

CREATE TABLE factura(

factura\_id number(5) PRIMARY KEY,

colet\_id number(5) NOT NULL,

data TIMESTAMP NOT NULL,

garantie TIMESTAMP NOT NULL,

tva number(5) NOT NULL,

CONSTRAINT fk5\_colet FOREIGN KEY (colet\_id) REFERENCES colet(colet\_id)

);

DROP TABLE colet;

CREATE TABLE colet(

colet\_id number(5) PRIMARY KEY,

comanda\_id number(5) NOT NULL,

stock\_id number(5) NOT NULL,

factura\_id number(5) NOT NULL,

cantitate\_stock number(5) NOT NULL,

CONSTRAINT fk4\_colet FOREIGN KEY (colet\_id) REFERENCES colet(colet\_id),

CONSTRAINT fk4\_comanda FOREIGN KEY (comanda\_id) REFERENCES comanda(comanda\_id),

CONSTRAINT fk4\_stock FOREIGN KEY (stock\_id) REFERENCES stock(stock\_id)

);

DROP TABLE furnizor;

CREATE TABLE furnizor(

furnizor\_id number(5) PRIMARY KEY,

nume varchar2(25) NOT NULL

);

5. Adăugați informații coerente în tabelele create (minim 5 înregistrări pentru fiecare entitate independentă; minim 10 înregistrări pentru tabela asociativă).

INSERT INTO furnizor VALUES(1, 'Nike Delivery');

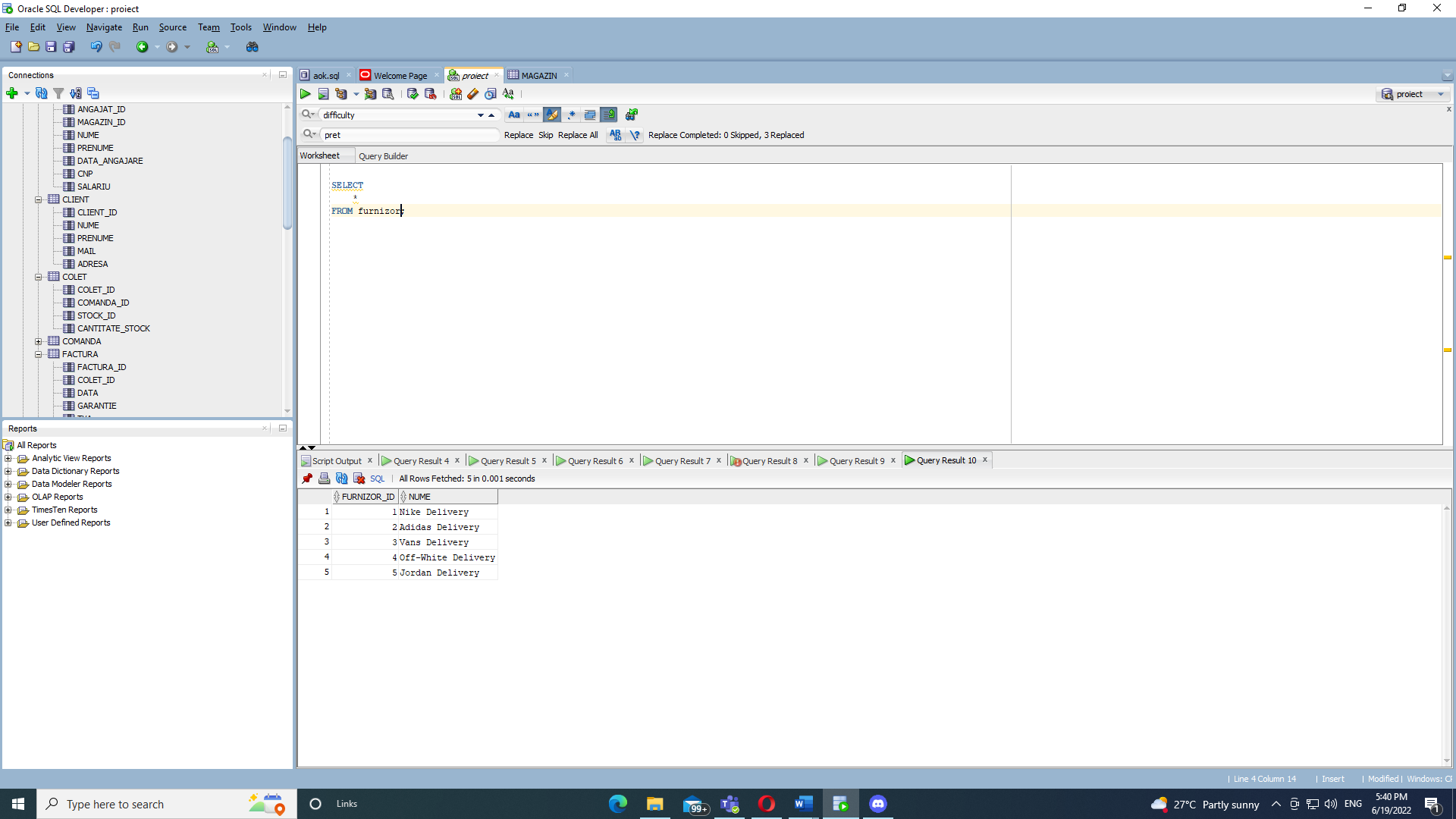
INSERT INTO furnizor VALUES(2, 'Adidas Delivery');

INSERT INTO furnizor VALUES(3, 'Vans Delivery');

INSERT INTO furnizor VALUES(4, 'Off-White Delivery');

INSERT INTO furnizor VALUES(5, 'Jordan Delivery');

SELECT \* FROM furnizor;



INSERT INTO magazin VALUES(1, 'Bucuresti', 'Bulevardul Ion C. Bratianu nr 44');

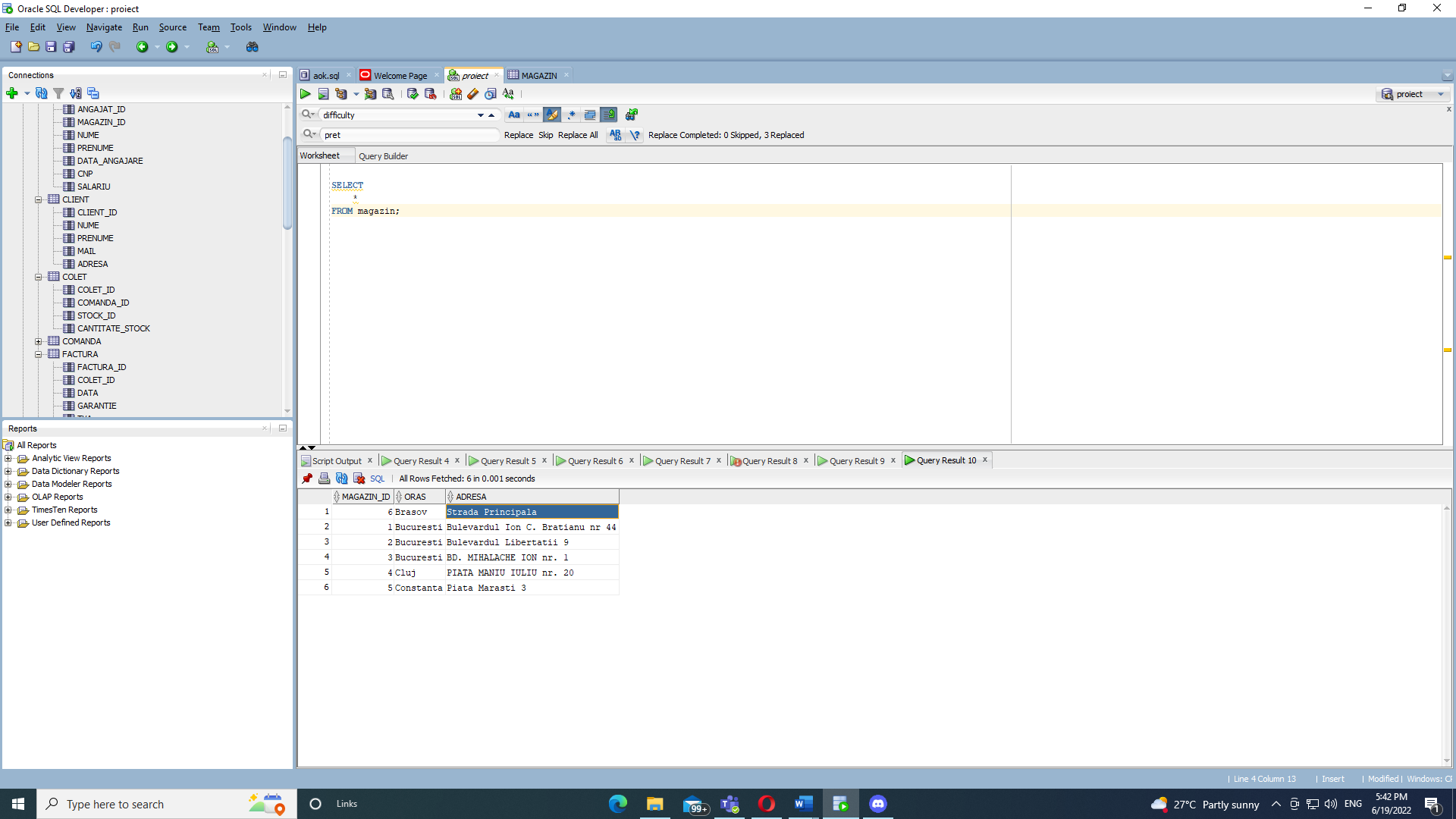
INSERT INTO magazin VALUES(2, 'Bucuresti', 'Bulevardul Libertatii 9' );

INSERT INTO magazin VALUES(3, 'Bucuresti', 'BD. MIHALACHE ION nr. 1');

INSERT INTO magazin VALUES(4, 'Cluj', 'PIATA MANIU IULIU nr. 20');

INSERT INTO magazin VALUES(5, 'Constanta', 'Piata Marasti 3');

SELECT \* FROM magazin;



INSERT INTO angajat VALUES(1, 1, 'Ionescu', 'Stefan', '02-JAN-2013', '2952632094297', 2500);

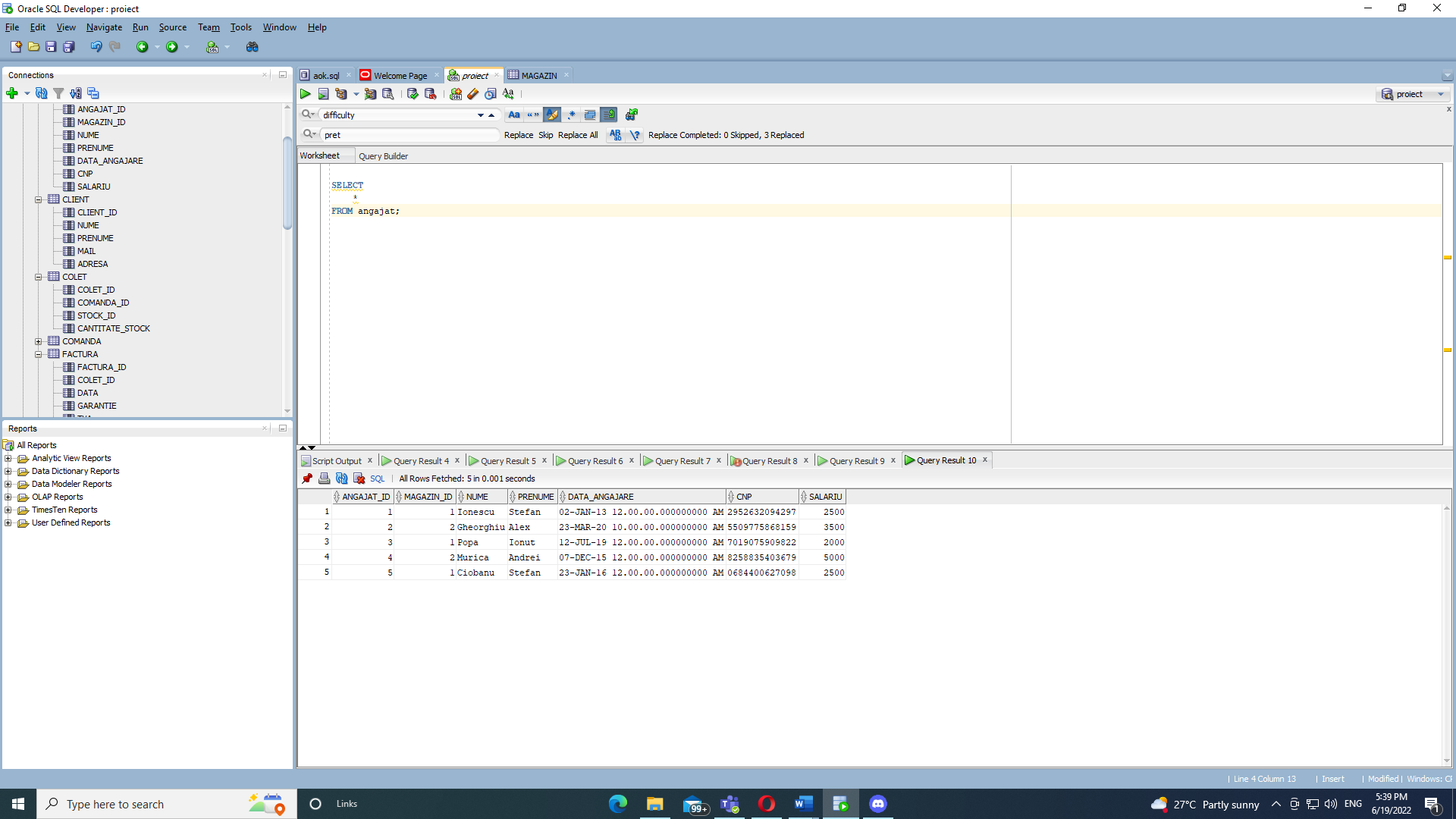
INSERT INTO angajat VALUES(2, 2, 'Gheorghiu', 'Alex', '23-MAR-2010', '5509775868159', 3500);

INSERT INTO angajat VALUES(3, 1, 'Popa', 'Ionut', '12-JUL-2019', '7019075909822', 2000);

INSERT INTO angajat VALUES(4, 2, 'Murica', 'Andrei', '07-DEC-2015', '8258835403679', 5000);

INSERT INTO angajat VALUES(5, 1, 'Ciobanu', 'Stefan', '23-JAN-2016', '0684400627098', 2500);

SELECT \* FROM angajat;



INSERT INTO client VALUES(1, 'Popescu', 'Stefan', 'popstef@gmail.com', 'Str Stefan cel mare 12');

INSERT INTO client VALUES(2, 'Stanciu', 'Mihai', 'mishu@gmail.com', 'Str Valea Argesului 18');

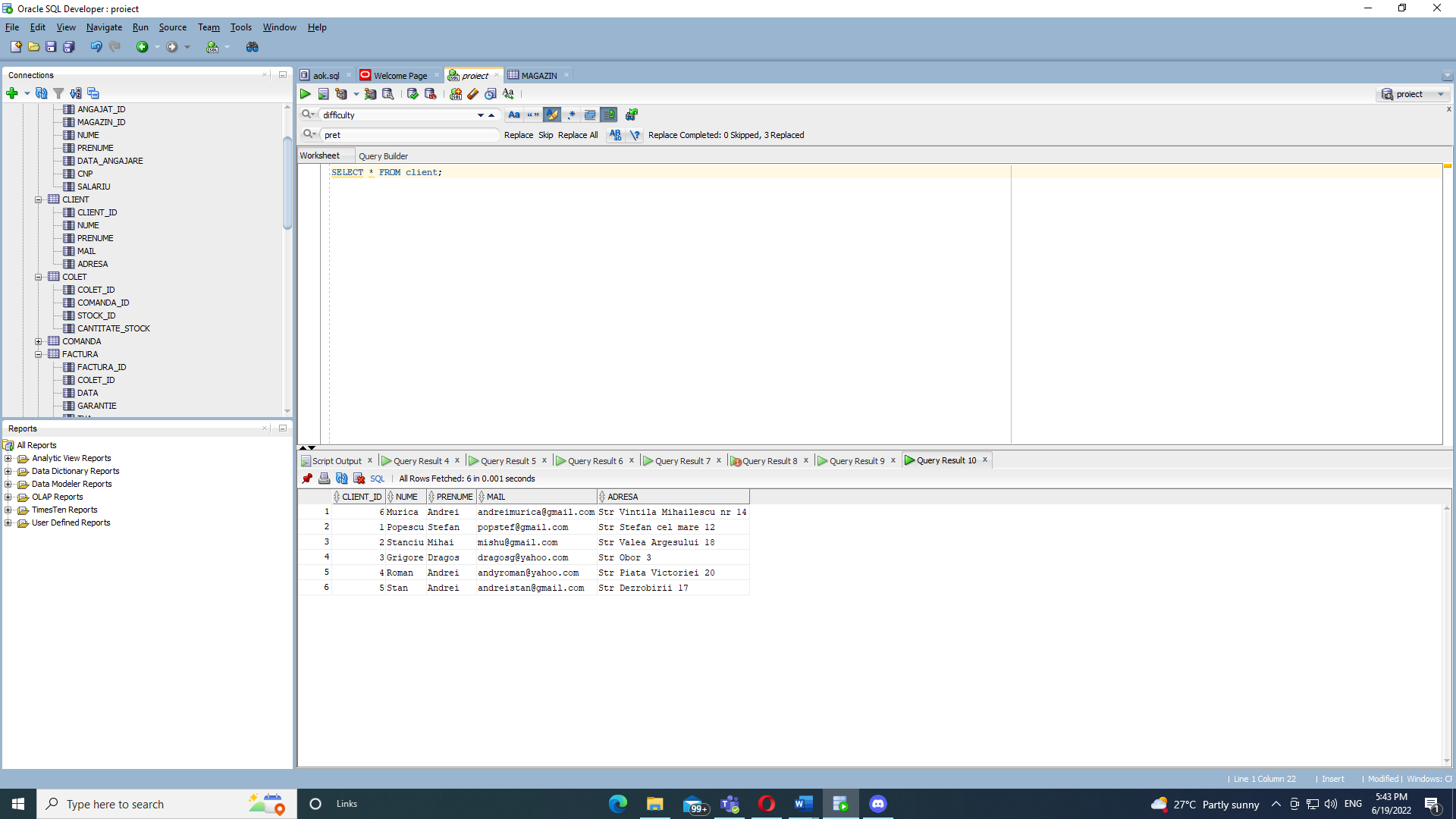
INSERT INTO client VALUES(3, 'Grigore', 'Dragos', 'dragosg@yahoo.com', 'Str Obor 3');

INSERT INTO client VALUES(4, 'Roman', 'Andrei', 'andyroman@yahoo.com', 'Str Piata Victoriei 20');

INSERT INTO client VALUES(5, 'Stan', 'Andrei', 'andreistan@gmail.com', 'Str Dezrobirii 17');

INSERT INTO client VALUES(6, 'Murica', 'Andrei', 'andreimurica@gmail.com', 'Str Vintila Mihailescu nr 14');

SELECT \* FROM client;



INSERT INTO comanda VALUES(1, 1, 1, 'Plasata', 850);

INSERT INTO comanda VALUES(2, 1, 1, 'Finalizata', 500);

INSERT INTO comanda VALUES(3, 2, 1, 'Pe drum', 1000);

INSERT INTO comanda VALUES(4, 1, 2, 'Pregatita', 650);

INSERT INTO comanda VALUES(5, 3, 2, 'Pregatita', 700);

INSERT INTO comanda VALUES(6, 4, 4, 'Pregatita', 850);

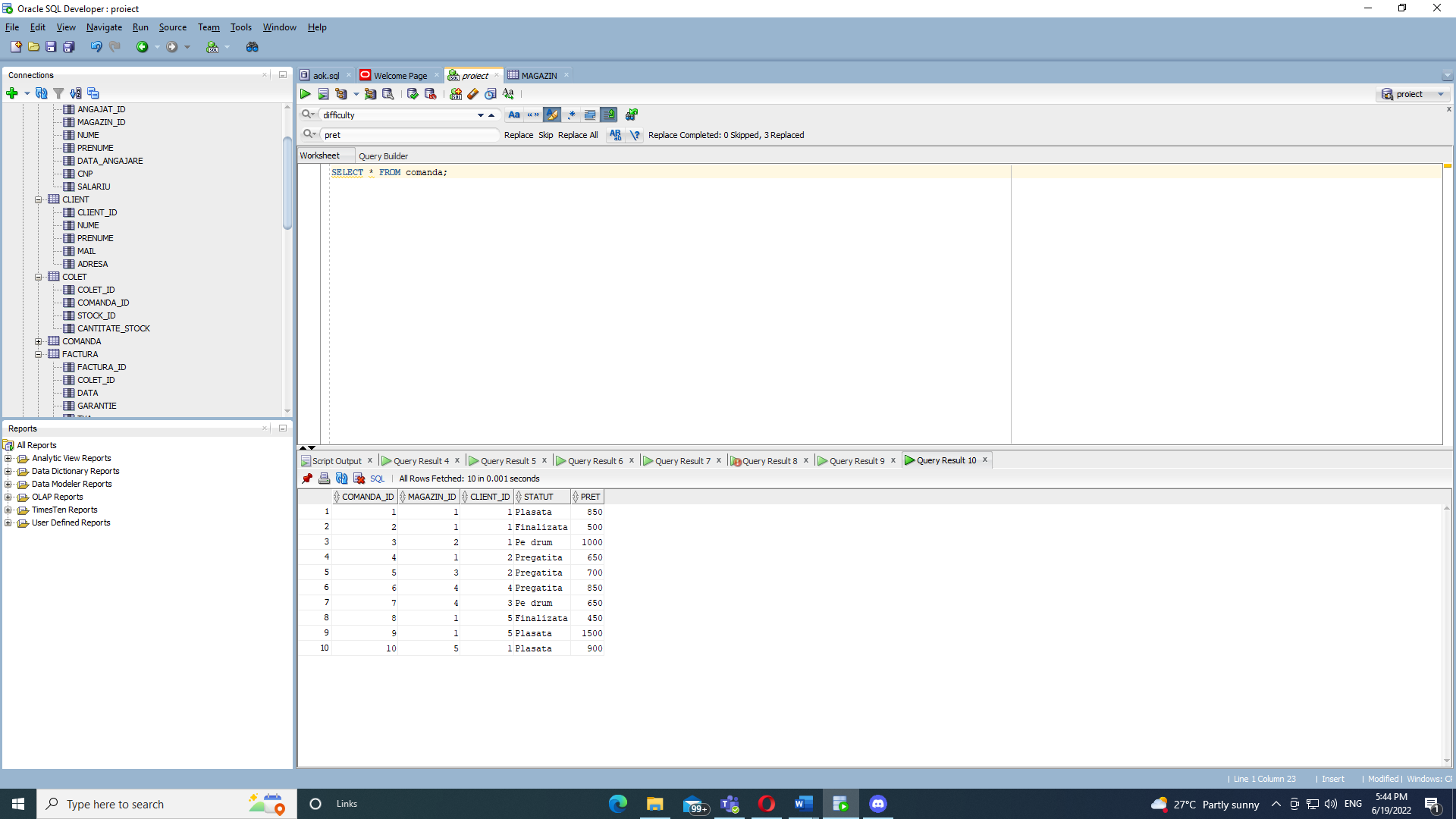
INSERT INTO comanda VALUES(7, 4, 3, 'Pe drum', 650);

INSERT INTO comanda VALUES(8, 1, 5, 'Finalizata', 450);

INSERT INTO comanda VALUES(9, 1, 5, 'Plasata', 1500);

INSERT INTO comanda VALUES(10, 5, 1, 'Plasata', 900);

SELECT \* FROM comanda;



INSERT INTO produs VALUES(1, 'Nike black', 'negru', 'Cei mai buni adidasi negri');

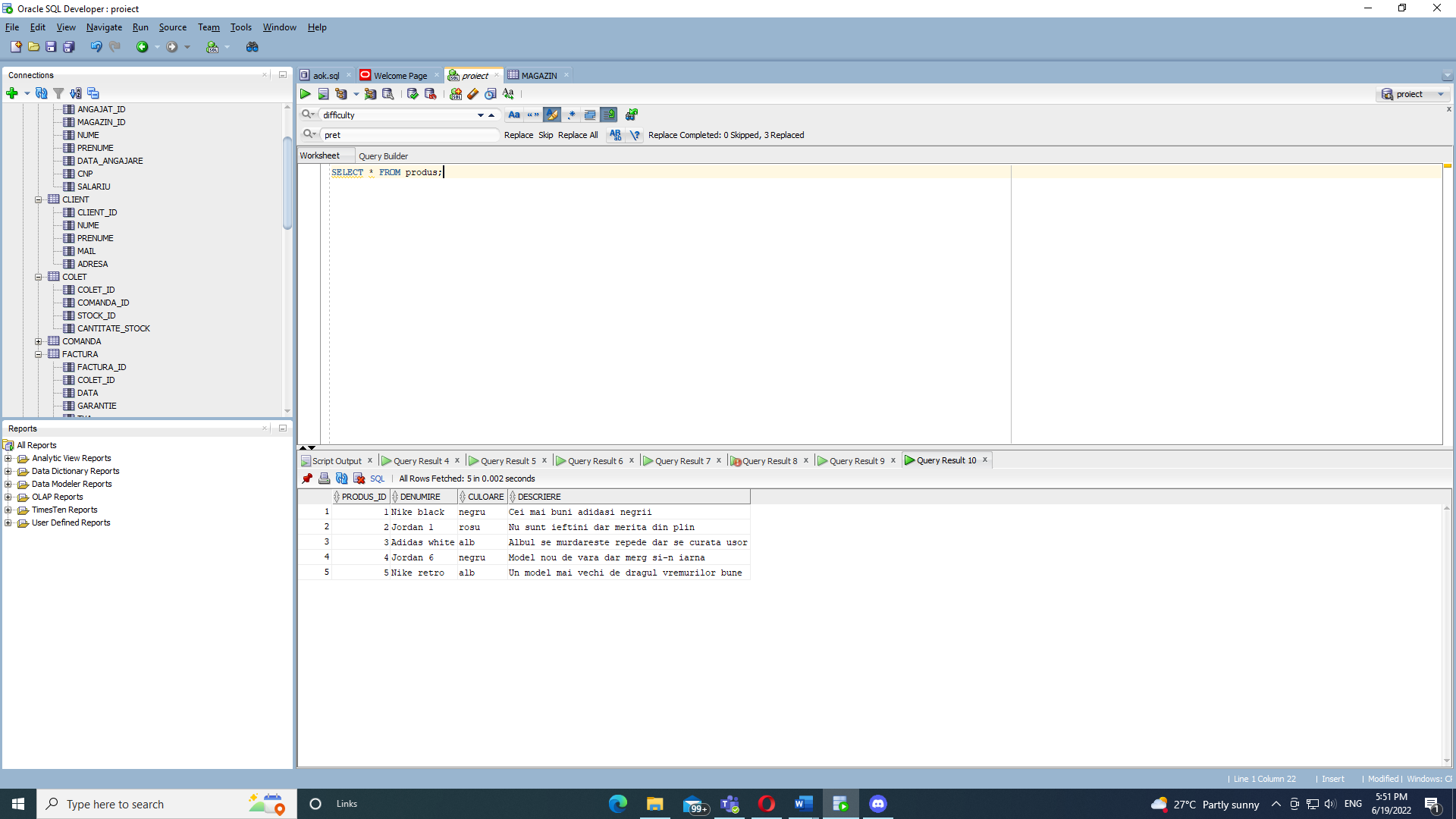
INSERT INTO produs VALUES(2, 'Jordan 1', 'rosu', 'Nu sunt ieftini dar merita din plin' );

INSERT INTO produs VALUES(3, 'Adidas white', 'alb', 'Albul se murdareste repede dar se curata usor');

INSERT INTO produs VALUES(4, 'Jordan 6', 'negru', 'Model nou de vara dar merg si-n iarna');

INSERT INTO produs VALUES(5, 'Nike retro', 'alb', 'Un model mai vechi de dragul vremurilor bune');

SELECT \* FROM produs;



INSERT INTO marfa VALUES(1, 1, 1, 1, 10, '12-MAY-2020', 41, 650);

INSERT INTO marfa VALUES(2, 1, 1, 1, 15, '15-MAY-2020', 42, 650);

INSERT INTO marfa VALUES(3, 2, 1, 2, 8, '12-JUL-2020', 41, 500);

INSERT INTO marfa VALUES(4, 1, 2, 2, 18, '02-MAY-2020', 40, 450);

INSERT INTO marfa VALUES(5, 3, 2, 2, 22, '25-JUL-2020', 43, 550);

INSERT INTO marfa VALUES(6, 4, 4, 3, 10, '12-MAY-2020', 44, 850);

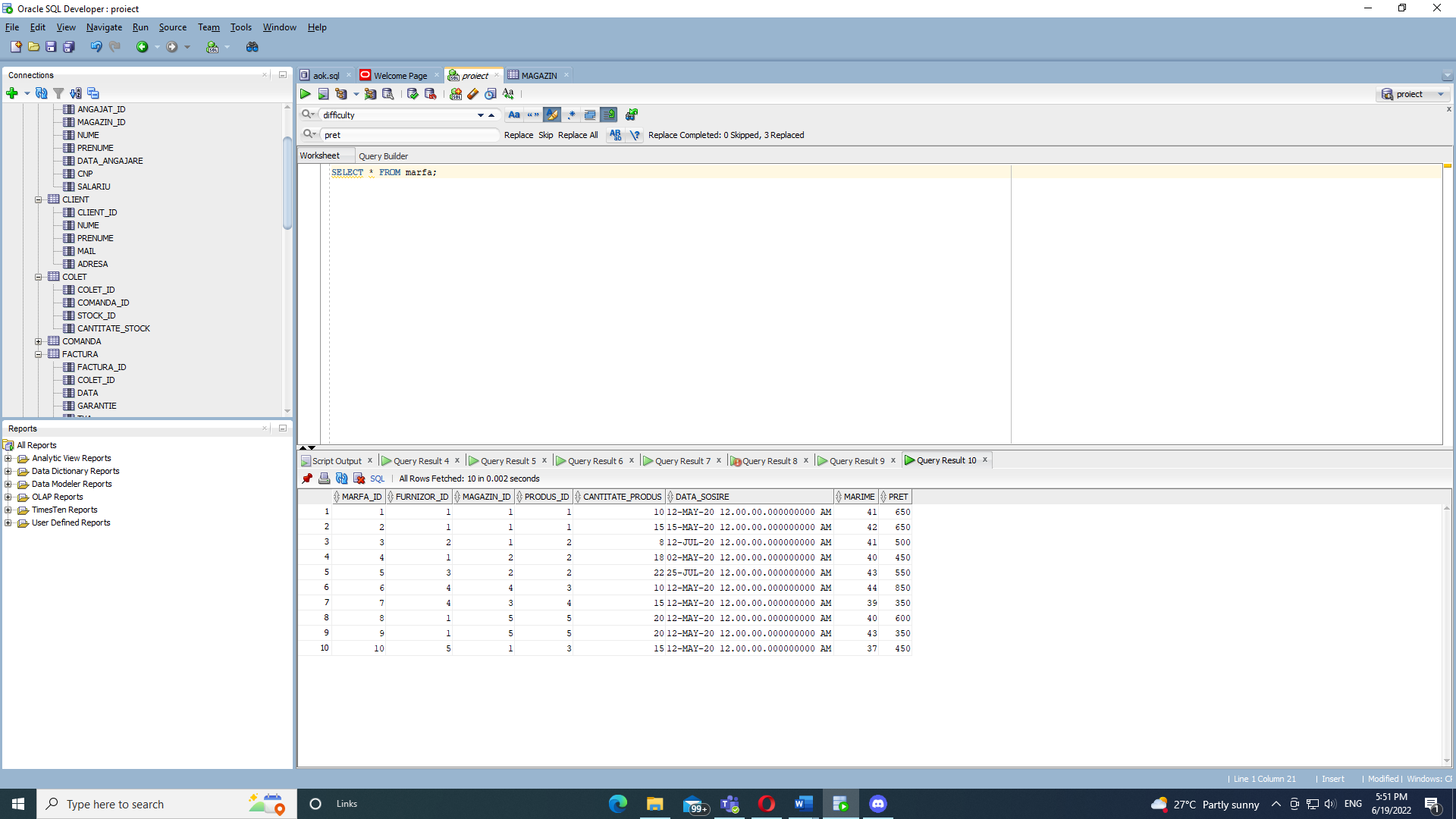
INSERT INTO marfa VALUES(7, 4, 3, 4, 15, '12-MAY-2020', 39, 350);

INSERT INTO marfa VALUES(8, 1, 5, 5, 20, '12-MAY-2020', 40, 600);

INSERT INTO marfa VALUES(9, 1, 5, 5, 20, '12-MAY-2020', 43, 350);

INSERT INTO marfa VALUES(10, 5, 1, 3, 15, '12-MAY-2020', 37, 450);

SELECT \* FROM marfa;



INSERT INTO stock VALUES(1, 1, 1, 10, 41, 750);

INSERT INTO stock VALUES(2, 1, 1, 15, 42, 750);

INSERT INTO stock VALUES(3, 1, 2, 8, 41, 600);

INSERT INTO stock VALUES(4, 2, 2, 18, 40, 550);

INSERT INTO stock VALUES(5, 2, 2, 22, 43, 650);

INSERT INTO stock VALUES(6, 4, 3, 10, 44, 1000);

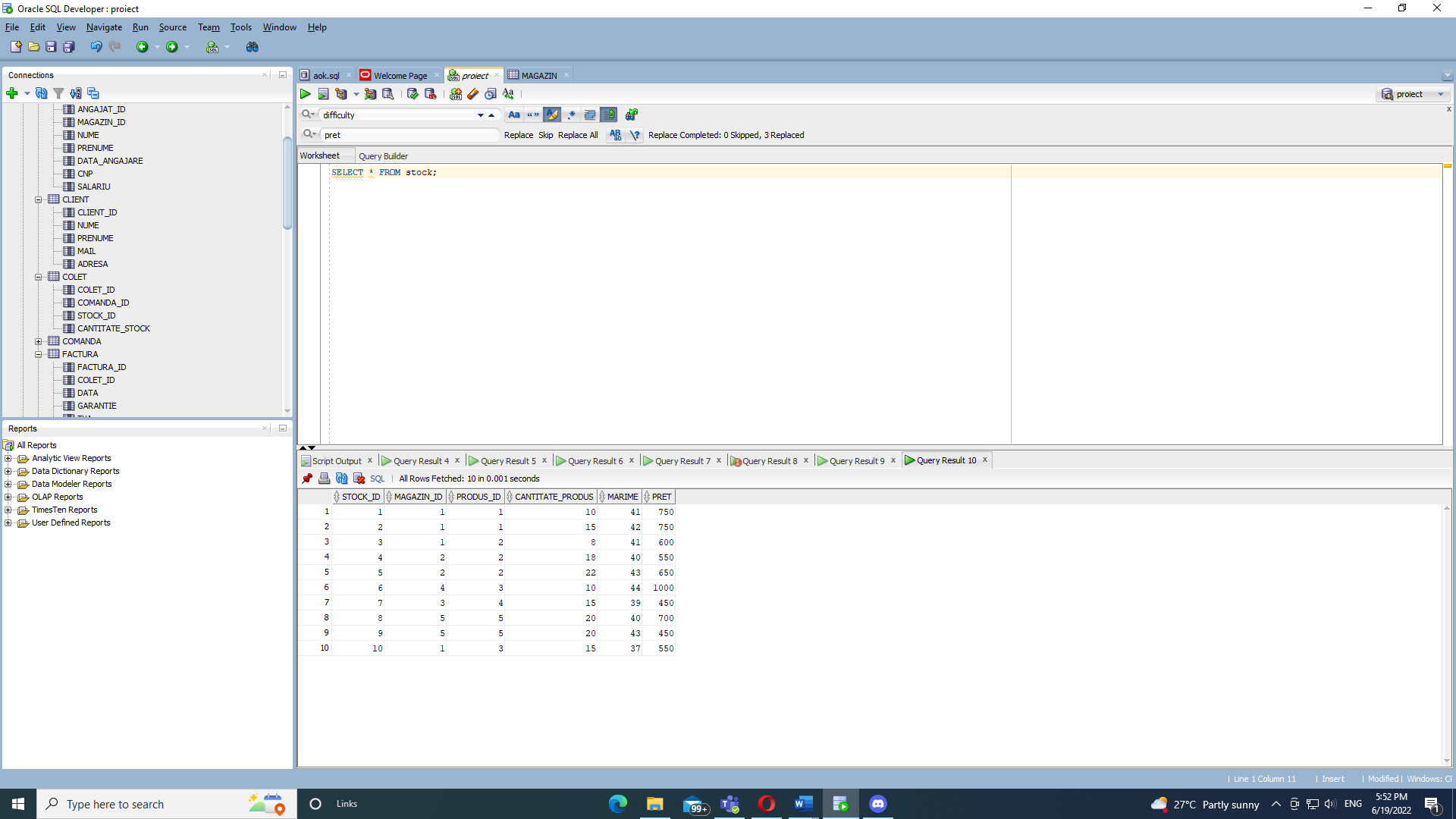
INSERT INTO stock VALUES(7, 3, 4, 15, 39, 450);

INSERT INTO stock VALUES(8, 5, 5, 20, 40, 700);

INSERT INTO stock VALUES(9, 5, 5, 20, 43, 450);

INSERT INTO stock VALUES(10, 1, 3, 15, 37, 550);

SELECT \* FROM stock;



INSERT INTO colet VALUES(1, 1, 1, 1);

INSERT INTO colet VALUES(2, 1, 2, 2);

INSERT INTO colet VALUES(3, 2, 3, 1);

INSERT INTO colet VALUES(4, 6, 4, 3);

INSERT INTO colet VALUES(5, 7, 7, 1);

INSERT INTO colet VALUES(6, 4, 3, 2);

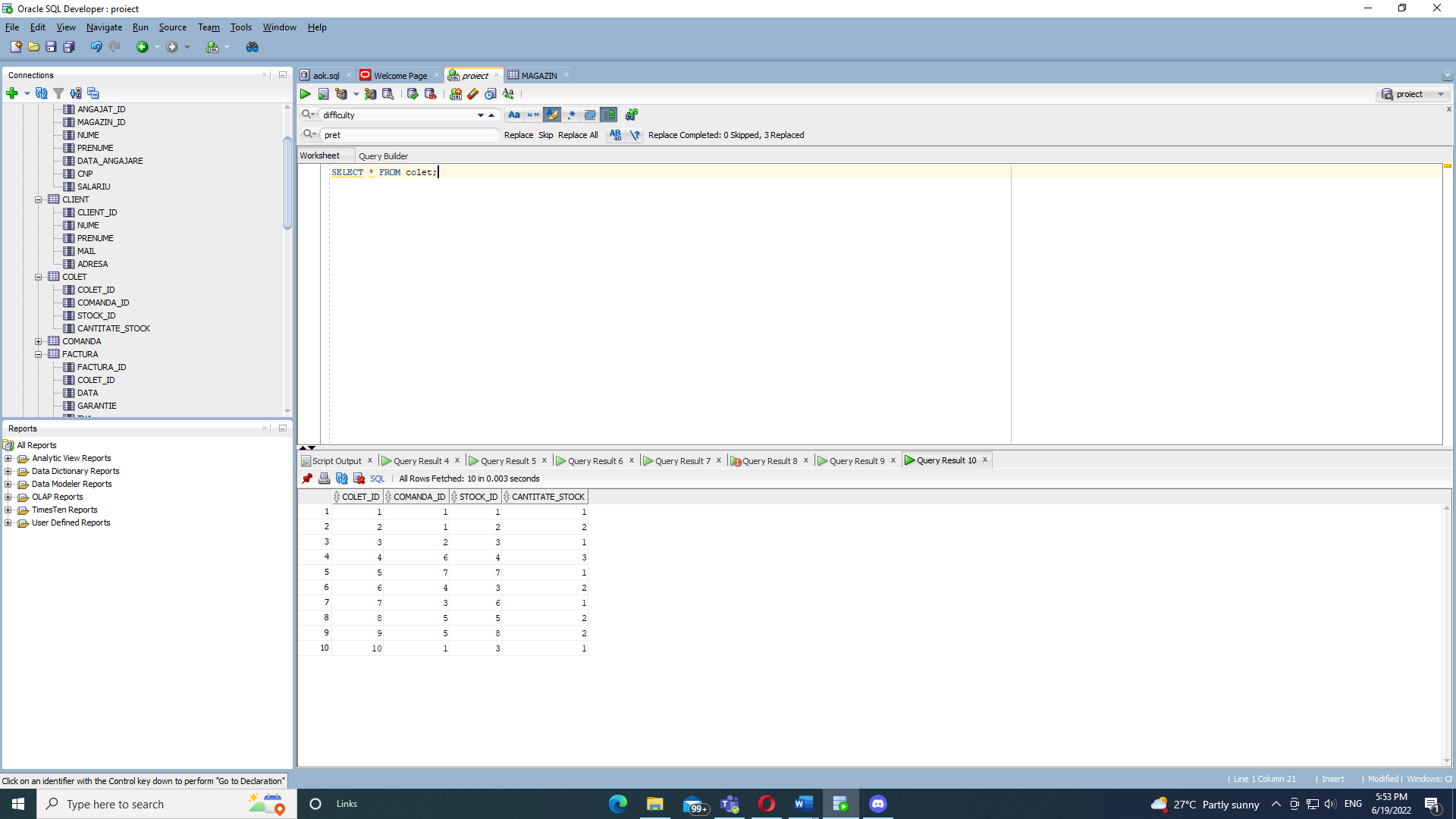
INSERT INTO colet VALUES(7, 3, 6, 1);

INSERT INTO colet VALUES(8, 5, 5, 2);

INSERT INTO colet VALUES(9, 5, 8, 2);

INSERT INTO colet VALUES(10, 1, 3, 1);

SELECT \* FROM colet;



INSERT INTO factura VALUES(1, 1, '12-MAY-19', '12-MAY-21', 100);

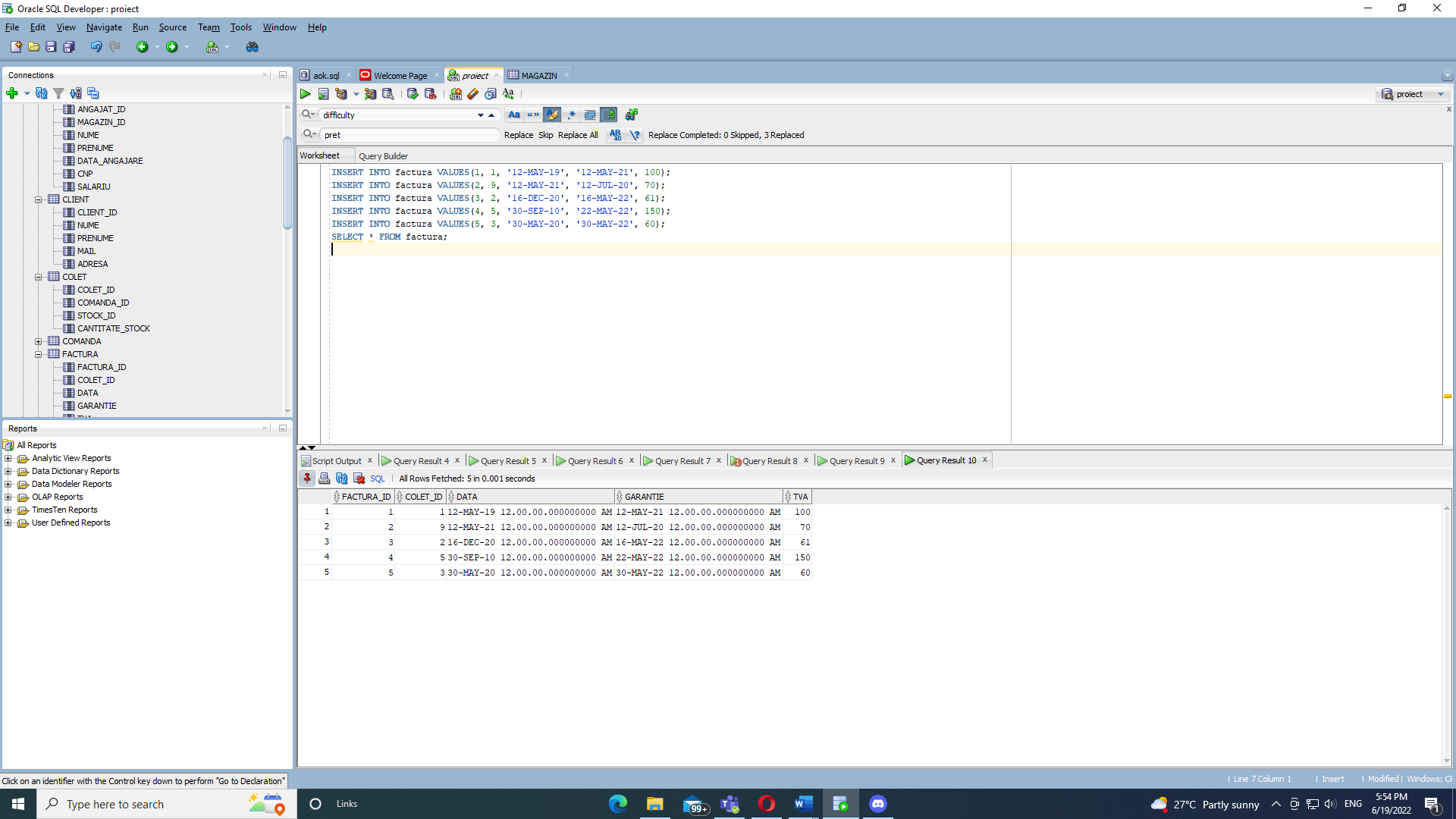
INSERT INTO factura VALUES(2, 9, '12-MAY-21', '12-JUL-20', 70);

INSERT INTO factura VALUES(3, 2, '16-DEC-20', '16-MAY-22', 61);

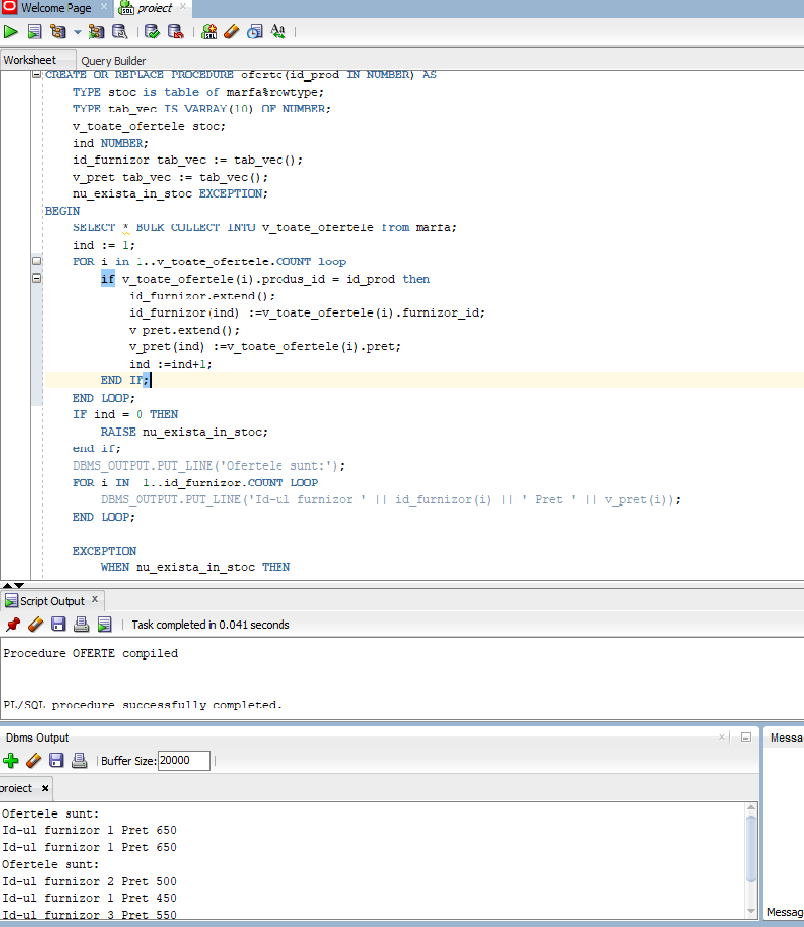
INSERT INTO factura VALUES(4, 5, '30-SEP-10', '22-MAY-22', 150);

INSERT INTO factura VALUES(5, 3, '30-MAY-20', '30-MAY-22', 60);

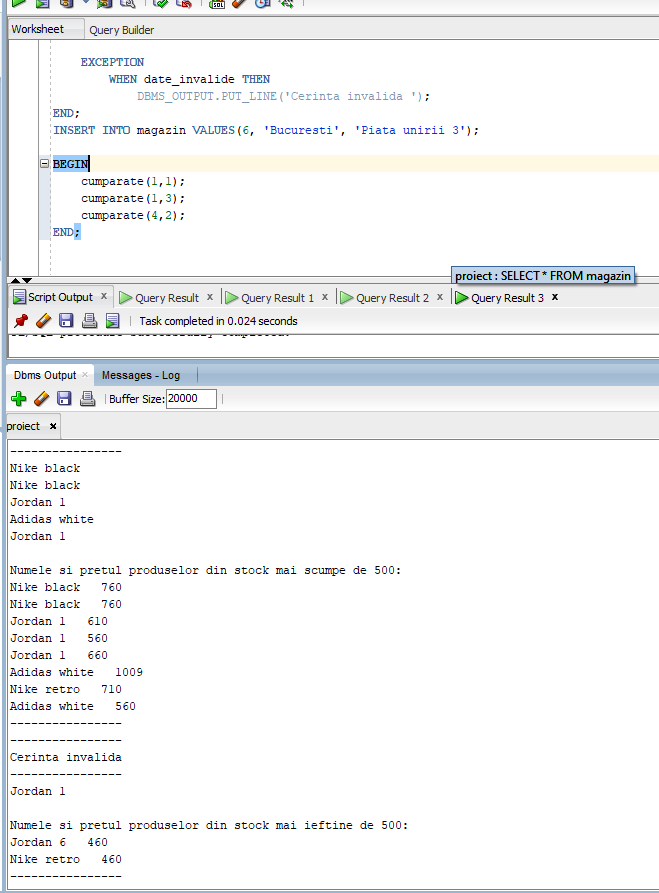
SELECT \* FROM factura;



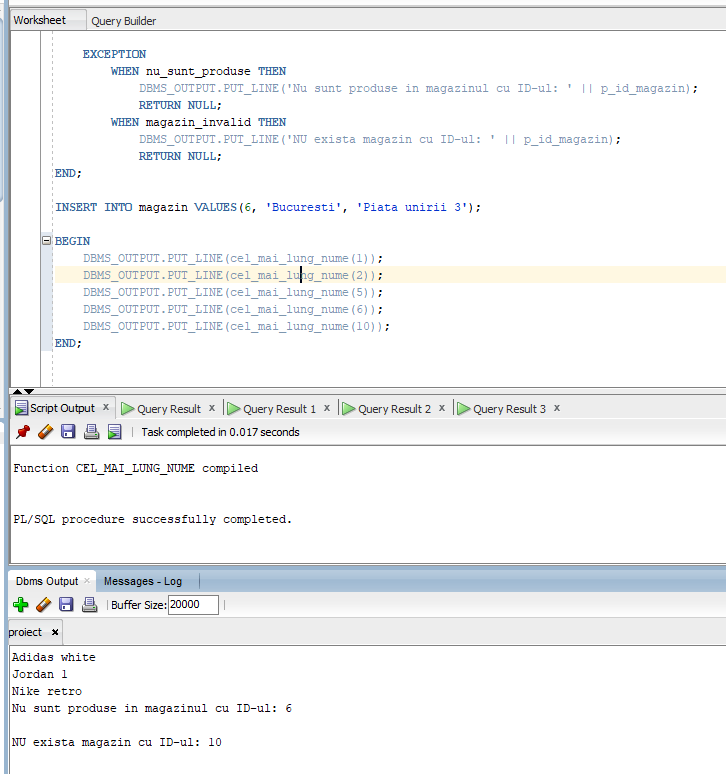
6. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze două tipuri de colecție studiate. Apelați subprogramul.



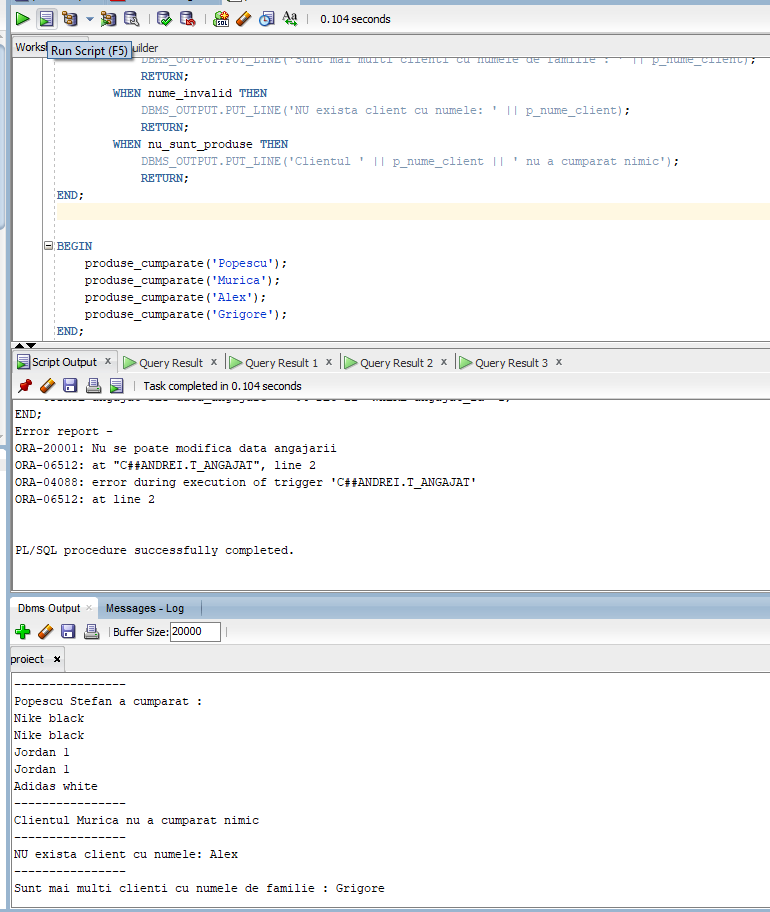
7. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze 2 tipuri de cursoare studiate, unul dintre acestea fiind cursor parametrizat. Apelați subprogramul.



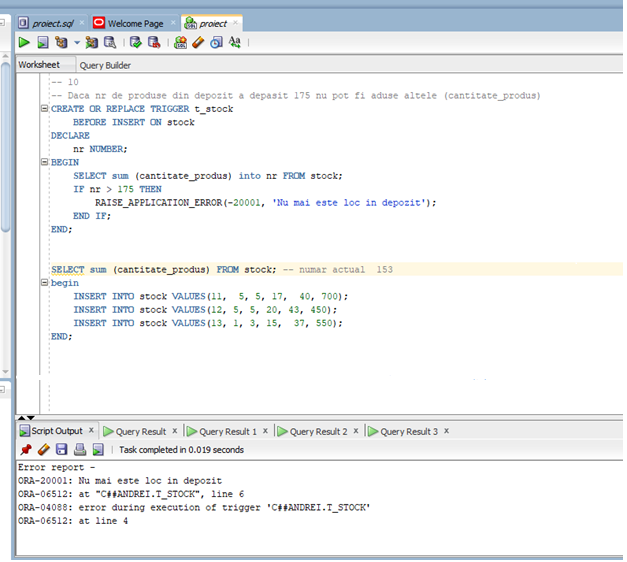
8. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip funcție care să utilizeze într-o singură comandă SQL 3 dintre tabelele definite.

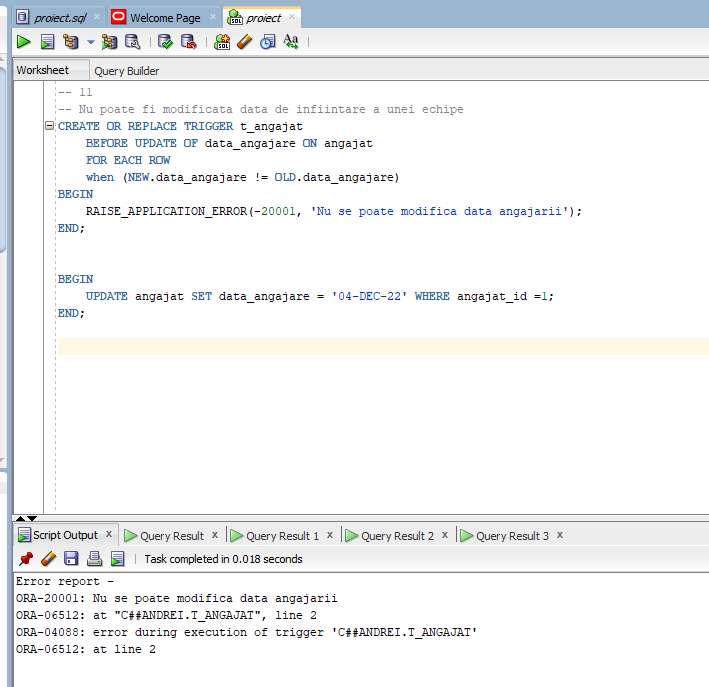


9. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip procedură care să utilizeze într-o singură comandă SQL 5 dintre tabelele definite.

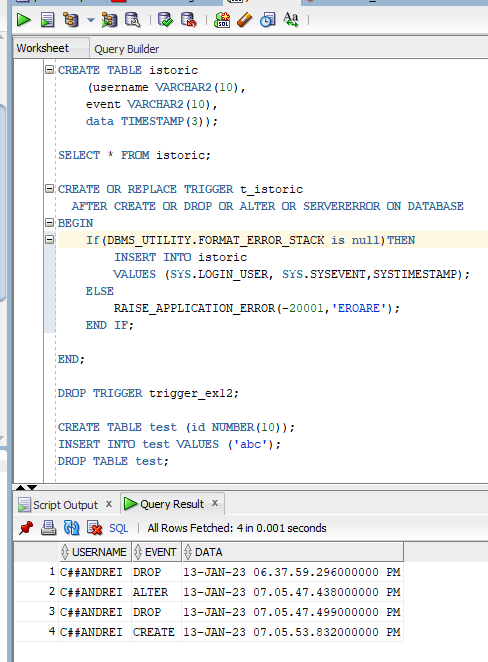


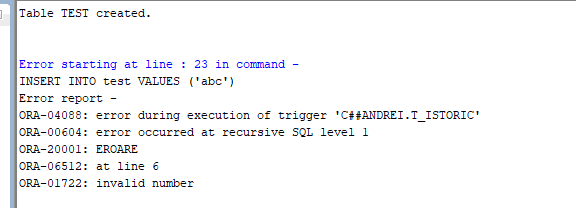
10. Definiți un trigger de tip LMD la nivel de comandă. Declanșați trigger-ul.



11. Definiți un trigger de tip LMD la nivel de linie. Declanșați trigger-ul.

12. Definiți un trigger de tip LDD. Declanșați trigger-ul.





13. Definiti un pachet care sa contina toate obiectele definite in cadrul proiectului.

