Universitatea din București

Facultatea de Matematică și Informatică

Specializarea Informatică

**Gestionarea datelor pentru o aplicație de tip configurator auto**

Ianuarie 2023

Proiect realizat de:

Panait Ana-Maria

Cuprins

[1. Prezentarea utilității bazei de date 3](#_Toc124543833)

[Regulile de funcționare ale modelului: 3](#_Toc124543834)

[Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului. 3](#_Toc124543835)

[2. Diagrama entitate-relație (ERD) 5](#_Toc124543836)

[3. Diagrama conceptuală 6](#_Toc124543837)

[4. Crearea tabelelor împreună cu constrângerile de integritate necesare 7](#_Toc124543838)

[5. Inserarea datelor in tabele 19](#_Toc124543839)

[6. Formulați in limbaj natural o problema pe care sa o rezolvați folosind un subprogram stocat independent care sa utilizeze două tipuri deferite de colecții studiate. Apelați subprogramul. 42](#_Toc124543840)

[7. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze 2 tipuri diferite de cursoare studiate, unul dintre acestea fiind cursor parametrizat. Apelați subprogramul. 46](#_Toc124543841)

[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 singura comandă SQL 3 dintre tabelele definite. Definiți minim 2 excepții. Apelați subprogramul astfel încât sa evidențiați toate cazurile tratate. 49](#_Toc124543842)

[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 comanda SQL 5 dintre tabelele definite. Tratați toate excepțiile care pot apărea, incluzând excepțiile NO\_DATA\_FOUND și TOO\_MANY\_ROWS. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate. 52](#_Toc124543843)

[10. Definiți un trigger de tip LMD la nivel de comandă. Declanșați trigger-ul. 56](#_Toc124543844)

[11. Definiți un trigger de tip LMD la nivel de linie. Declanșați trigger-ul. 59](#_Toc124543845)

[12. Definiți un trigger de tip LDD. Declanșați trigger-ul. 63](#_Toc124543846)

[13. Definiți un pachet care să conțină toate obiectele definite în cadrul proiectului. 67](#_Toc124543847)

[14. Definiți un pachet care să includă tipuri de date complexe și obiecte necesare unui flux de acțiuni integrate, specifice bazei de date definite (minim 2 tipuri de date, minim 2 funcții, minim 2 proceduri). 75](#_Toc124543848)

## Prezentarea utilității bazei de date

Proiectul are ca scop gestionarea datelor pentru un serviciu de configurare a autovehiculelor pentru mai multe firme auto. Acesta se adresează atât firmelor de reprezentanță auto, cât și potențialilor clienți. Astfel, firmele auto pot ține evidența configurațiilor încercate de clienți (informație din care se pot face statistici utile la conceperea viitoarelor produse), dar și a ofertelor făcute fiecărui client (pentru aceeași configurație clienții pot primi oferte diferite de preț în funcție de țara în care se vinde mașina respectivă, spre exemplu).

Clienții beneficiază de informația necesară pentru a plasa o comanda cu autoturismul configurat rapid și eficient. După ce un client finalizează o configurație aceasta este salvată și clientul primește un cod al acesteia pentru a merge la reprezentanță să comande direct acel autoturism prezentând codul primit, care este chiar id\_configuratie detaliat mai încolo. Acest procedeu salvează mult timp de așteptare la coada de la reprezentanța auto. Clienții mai au avantajul vizualizării diferitelor combinații de echipări pentru mașini și prețul final pentru acestea, putând verifica dacă se încadrează în bugetul disponibil pentru achiziționarea produsului.

## 

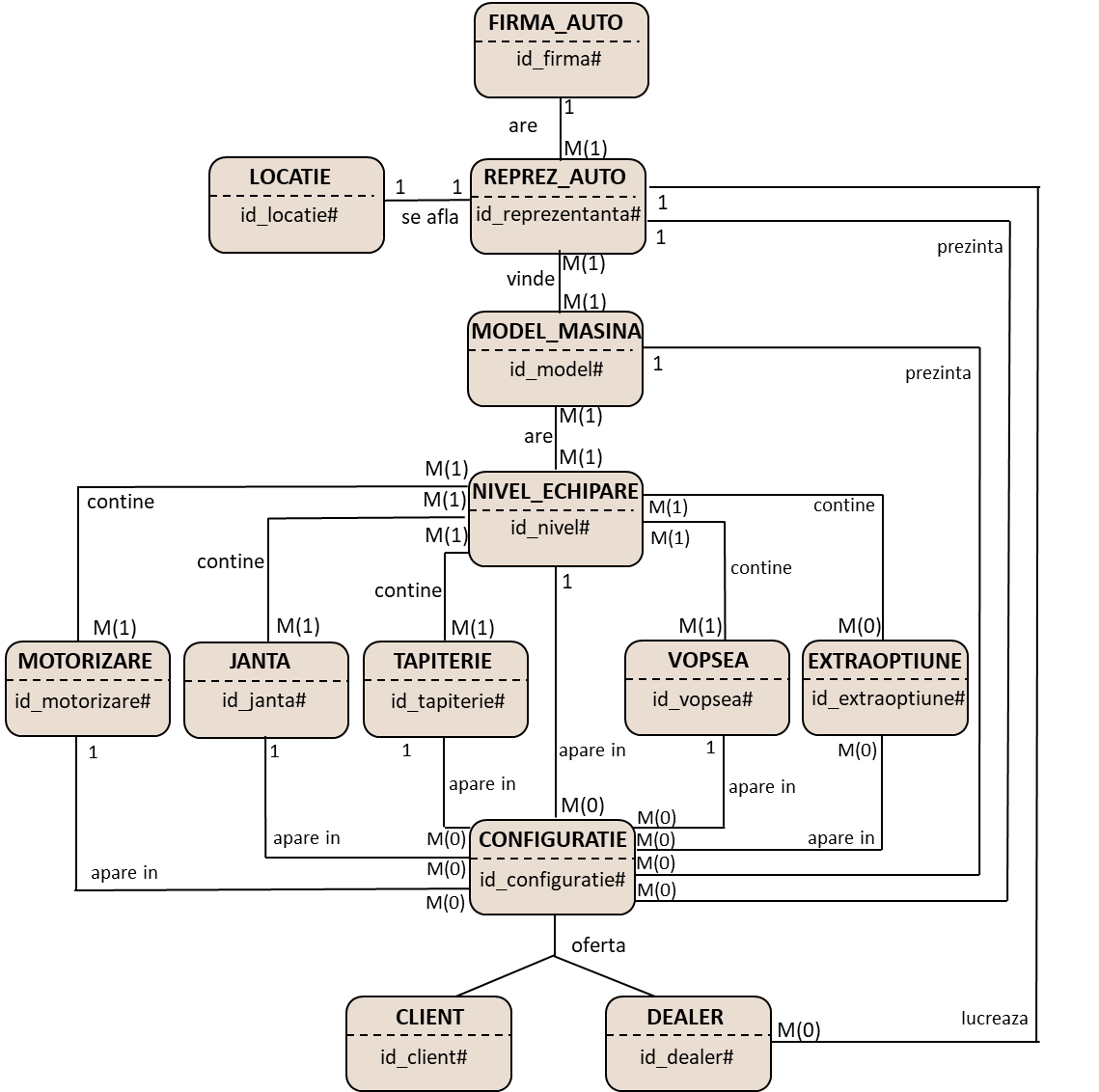
## Regulile de funcționare ale modelului:

* Un model de autovehicul are unul sau mai multe nivele de echipare disponibile, rolul lor fiind limitarea opțiunilor disponibile pentru selecție. Un nivel de echipare poate să apară în cadrul mai multor modele.
* În funcție de nivelul de echipare selectat sunt disponibile anumite motorizări, jante, tapițerii, extraopțiuni și vopseluri. Fiecare nivel de echipare pune la dispoziție alegerea dintre mai multe astfel de opțiuni, dar o opțiune poate să fie disponibilă în cadrul mai multor nivele de echipare.
* Fiecare configurație realizată de utilizatori reține neapărat reprezentanța de unde se poate cumpăra mașina, modelul, nivelul de echipare și câte o motorizare, un set de jante, o tapițerie, o vopsea și zero sau mai multe extraopțiuni, toate din același nivel de echipare. Oricare din aceste opțiuni selectate poate să apară în cadrul mai multor configurații.
* Nu toate reprezentanțele de mașini au posibilitatea de vânzare a tuturor modelelor de mașini produse de o firmă. Dacă acestea nu sunt disponibile în reprezentanța selectată atunci nici nu va putea fi configurată o mașină cu modelul respectiv.
* Un dealer poată să facă oferte unui client pentru una sau mai multe configurații. Aceeași ofertă se poate face mai multor clienți pentru o configurație sau poate să difere în funcție de țara în care se află reprezentanța auto, spre exemplu.

# Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului.

* O reprezentanță auto trebuie să aparțină unei singure firme auto.
* O firma auto poate avea una sau mai multe reprezentanțe. O firma auto trebuie să aibă cel puțin o reprezentanță auto.
* O reprezentanță auto trebuie să se afle într-o singură locație. La o locație se poate afla o singură reprezentanță auto.
* O reprezentanță auto are cel puțin un model de mașină disponibil.
* Fiecare model de mașină are cel puțin un nivel de echipare disponibil.
* Fiecare nivel de echipare al unui model de mașină trebuie să aibă disponibilă cel puțin o motorizare, un set de jante, o tapițerie și o culoare.
* O reprezentanță auto trebuie să aibă măcar un dealer angajat care se ocupă cu crearea și prezentarea ofertei de preț clienților.

## Diagrama entitate-relație (ERD)



## Diagrama conceptuală

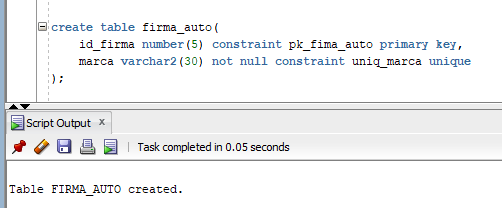
## Crearea tabelelor împreună cu constrângerile de integritate necesare

* FIRMA\_AUTO

create table firma\_auto(

id\_firma number(5) constraint pk\_fima\_auto primary key,

marca varchar2(30) not null constraint uniq\_marca unique

);

* LOCATIE

create table locatie(

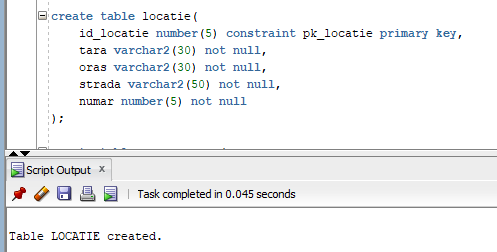
id\_locatie number(5) constraint pk\_locatie primary key,

tara varchar2(30) not null,

oras varchar2(30) not null,

strada varchar2(50) not null,

numar number(5) not null

);

* REPREZ\_AUTO

create table reprez\_auto(

id\_reprez number(5) constraint pk\_reprez\_auto primary key,

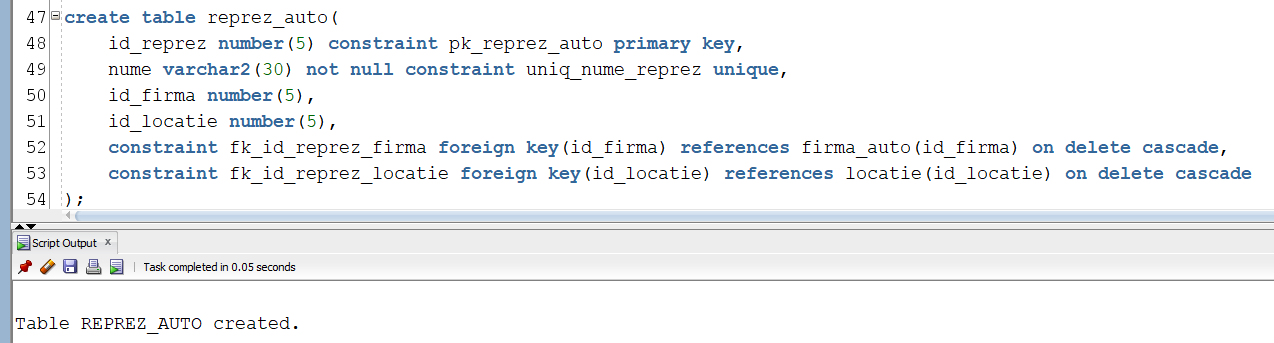
nume varchar2(30) not null constraint uniq\_nume\_reprez unique,

id\_firma number(5),

id\_locatie number(5),

constraint fk\_id\_reprez\_firma foreign key(id\_firma) references firma\_auto(id\_firma) on delete cascade,

constraint fk\_id\_reprez\_locatie foreign key(id\_locatie) references locatie(id\_locatie) on delete cascade

);

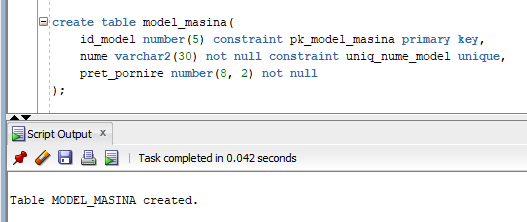
* MODEL\_MASINA

create table model\_masina(

id\_model number(5) constraint pk\_model\_masina primary key,

nume varchar2(30) not null constraint uniq\_nume\_model unique,

pret\_pornire number(8, 2) not null

);

* MODELE\_DISP

create table modele\_disp(

id\_reprez number(5),

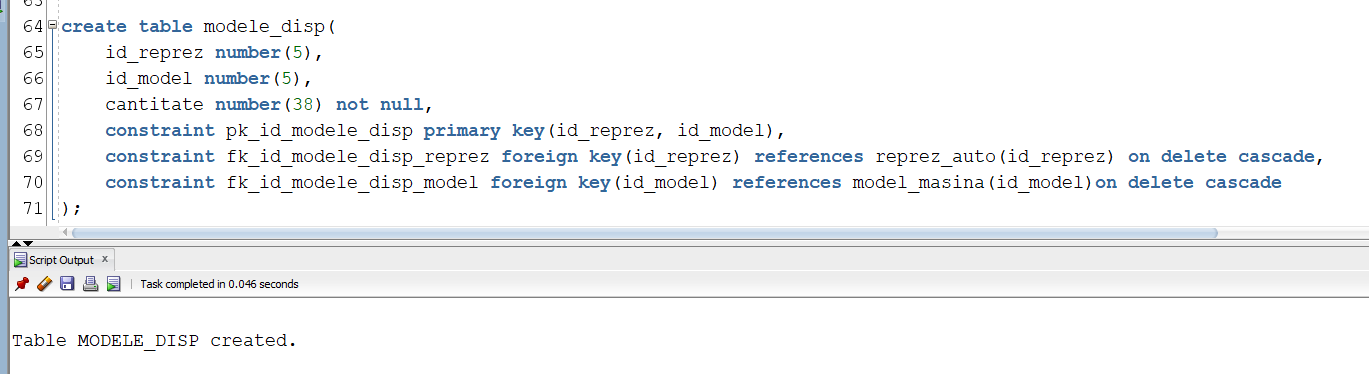
id\_model number(5),

cantitate number(38) not null,

constraint pk\_id\_modele\_disp primary key(id\_reprez, id\_model),

constraint fk\_id\_modele\_disp\_reprez foreign key(id\_reprez) references reprez\_auto(id\_reprez) on delete cascade,

constraint fk\_id\_modele\_disp\_model foreign key(id\_model) references model\_masina(id\_model)on delete cascade

);

* NIVEL\_ECHIPARE

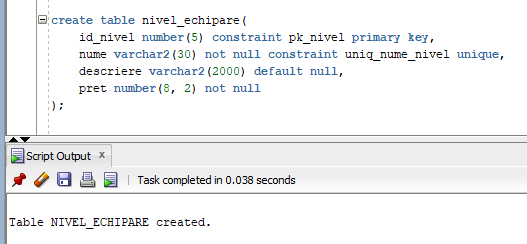
create table nivel\_echipare(

id\_nivel number(5) constraint pk\_nivel primary key,

nume varchar2(30) not null constraint uniq\_nume\_nivel unique,

descriere varchar2(2000) default null,

pret number(8, 2) not null

);

* NIV\_ECH\_DISP

create table niv\_ech\_disp(

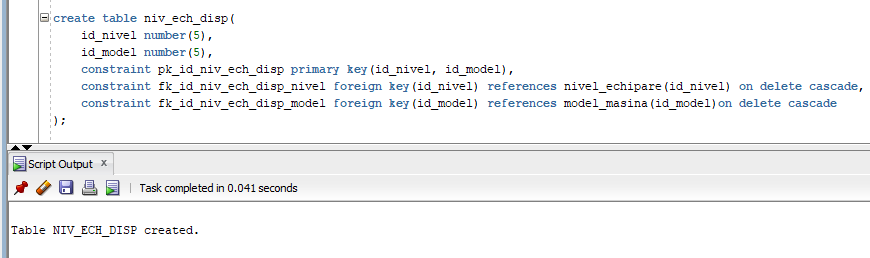
id\_nivel number(5),

id\_model number(5),

constraint pk\_id\_niv\_ech\_disp primary key(id\_nivel, id\_model),

constraint fk\_id\_niv\_ech\_disp\_nivel foreign key(id\_nivel) references nivel\_echipare(id\_nivel) on delete cascade,

constraint fk\_id\_niv\_ech\_disp\_model foreign key(id\_model) references model\_masina(id\_model)on delete cascade

);

* MOTORIZARE

create table motorizare(

id\_motorizare number(5) constraint pk\_motorizare primary key,

combustibil varchar2(20) not null check(lower(combustibil) in ('diesel', 'benzina', 'electric', 'hybrid', 'plug-in hybrid')),

putere number(3) not null,

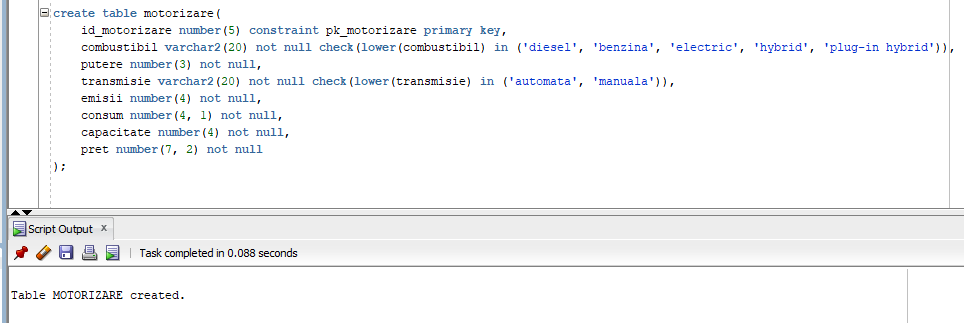
transmisie varchar2(20) not null check(lower(transmisie) in ('automata', 'manuala')),

emisii number(4) not null,

consum number(4, 1) not null,

capacitate number(4) not null,

pret number(7, 2) not null

);

* JANTA

create table janta(

id\_janta number(5) constraint pk\_janta primary key,

nume varchar2(30) not null constraint uniq\_nume\_janta unique,

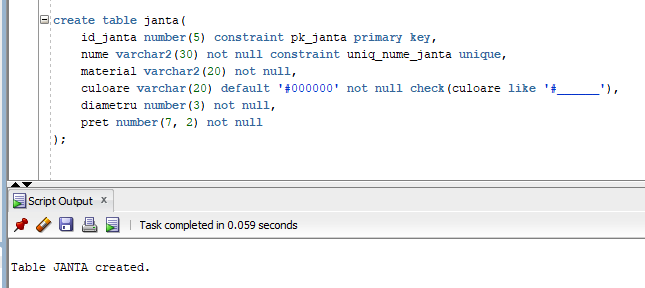
material varchar2(20) not null,

culoare varchar(20) default '#000000' not null check(culoare like '#\_\_\_\_\_\_'),

diametru number(3) not null,

pret number(7, 2) not null

);



* TAPITERIE

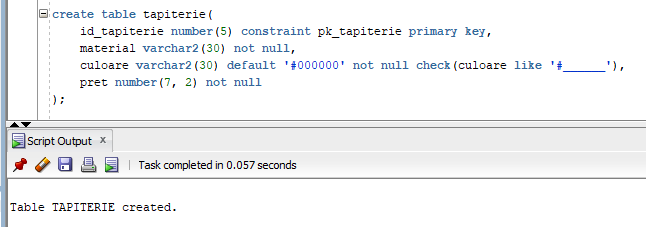
create table tapiterie(

id\_tapiterie number(5) constraint pk\_tapiterie primary key,

material varchar2(30) not null,

culoare varchar2(30) default '#000000' not null check(culoare like '#\_\_\_\_\_\_'),

pret number(7, 2) not null

);

* VOPSEA

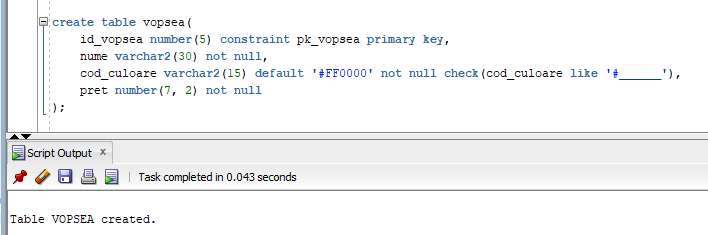
create table vopsea(

id\_vopsea number(5) constraint pk\_vopsea primary key,

nume varchar2(30) not null,

cod\_culoare varchar2(15) default '#FF0000' not null check(cod\_culoare like '#\_\_\_\_\_\_'),

pret number(7, 2) not null

);

* EXTRAOPTIUNE

create table extraoptiune(

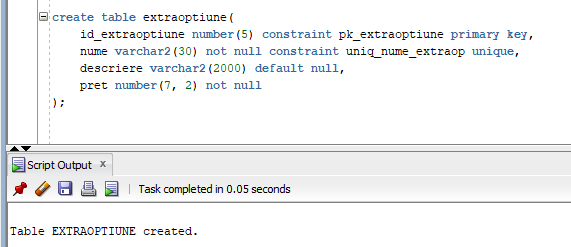
id\_extraoptiune number(5) constraint pk\_extraoptiune primary key,

nume varchar2(30) not null constraint uniq\_nume\_extraop unique,

descriere varchar2(2000) default null,

pret number(7, 2) not null

);



* MOTORIZARE\_NIV

create table motorizare\_niv(

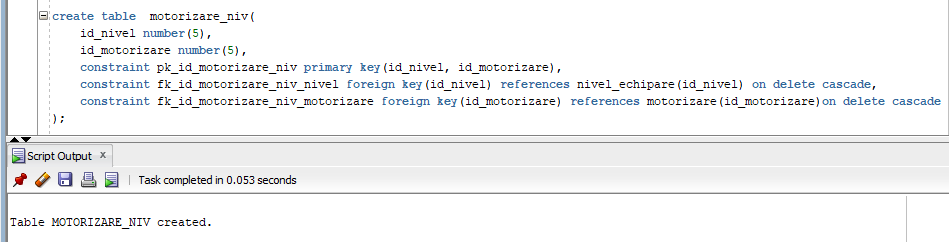
id\_nivel number(5),

id\_motorizare number(5),

constraint pk\_id\_motorizare\_niv primary key(id\_nivel, id\_motorizare),

constraint fk\_id\_motorizare\_niv\_nivel foreign key(id\_nivel) references nivel\_echipare(id\_nivel) on delete cascade,

constraint fk\_id\_motorizare\_niv\_motorizare foreign key(id\_motorizare) references motorizare(id\_motorizare)on delete cascade

);

* JANTA\_NIV

create table janta\_niv(

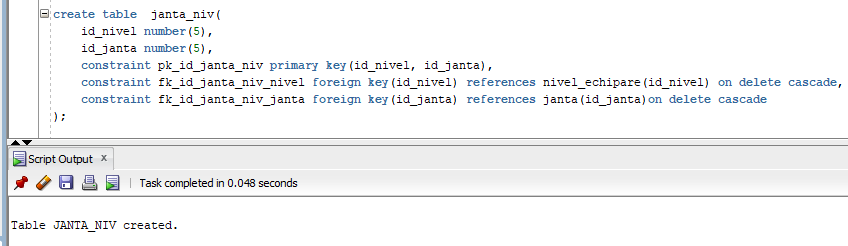
id\_nivel number(5),

id\_janta number(5),

constraint pk\_id\_janta\_niv primary key(id\_nivel, id\_janta),

constraint fk\_id\_janta\_niv\_nivel foreign key(id\_nivel) references nivel\_echipare(id\_nivel) on delete cascade,

constraint fk\_id\_janta\_niv\_janta foreign key(id\_janta) references janta(id\_janta)on delete cascade

);

* TAPIT\_NIV

create table tapit\_niv(

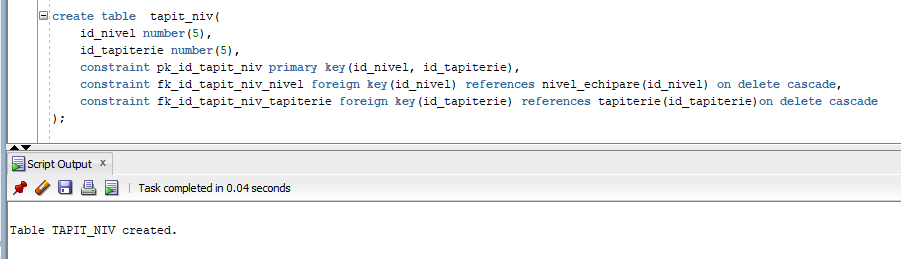
id\_nivel number(5),

id\_tapiterie number(5),

constraint pk\_id\_tapit\_niv primary key(id\_nivel, id\_tapiterie),

constraint fk\_id\_tapit\_niv\_nivel foreign key(id\_nivel) references nivel\_echipare(id\_nivel) on delete cascade,

constraint fk\_id\_tapit\_niv\_tapiterie foreign key(id\_tapiterie) references tapiterie(id\_tapiterie)on delete cascade

);

* VOPSEA\_NIV

create table vopsea\_niv(

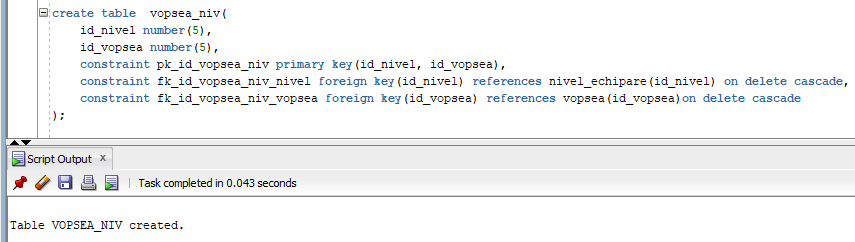
id\_nivel number(5),

id\_vopsea number(5),

constraint pk\_id\_vopsea\_niv primary key(id\_nivel, id\_vopsea),

constraint fk\_id\_vopsea\_niv\_nivel foreign key(id\_nivel) references nivel\_echipare(id\_nivel) on delete cascade,

constraint fk\_id\_vopsea\_niv\_vopsea foreign key(id\_vopsea) references vopsea(id\_vopsea)on delete cascade

);

* EXTRAOP\_NIV

create table extraop\_niv(

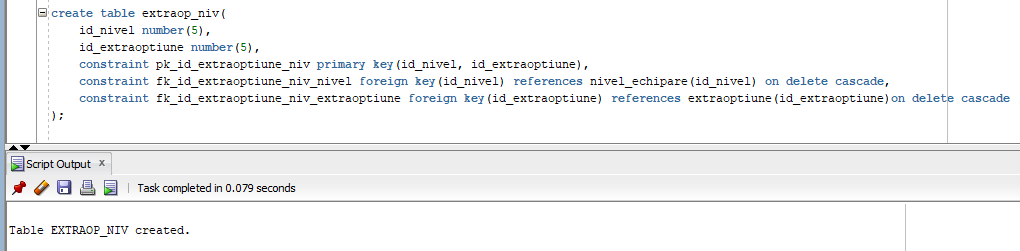
id\_nivel number(5),

id\_extraoptiune number(5),

constraint pk\_id\_extraoptiune\_niv primary key(id\_nivel, id\_extraoptiune),

constraint fk\_id\_extraoptiune\_niv\_nivel foreign key(id\_nivel) references nivel\_echipare(id\_nivel) on delete cascade,

constraint fk\_id\_extraoptiune\_niv\_extraoptiune foreign key(id\_extraoptiune) references extraoptiune(id\_extraoptiune)on delete cascade

);

* CONFIGURATIE

create table configuratie(

id\_configuratie number(5) constraint pk\_configuratie primary key,

id\_motorizare number(5) not null,

id\_janta number(5) not null,

id\_tapiterie number(5) not null,

id\_vopsea number(5) not null,

id\_nivel number(5) not null,

id\_reprez number(5) not null,

id\_model number(5) not null,

pret\_total number(9, 2),

constraint fk\_configuratie\_id\_motorizari foreign key(id\_motorizare) references motorizare(id\_motorizare) on delete cascade,

constraint fk\_configuratie\_id\_janta foreign key(id\_janta) references janta(id\_janta) on delete cascade,

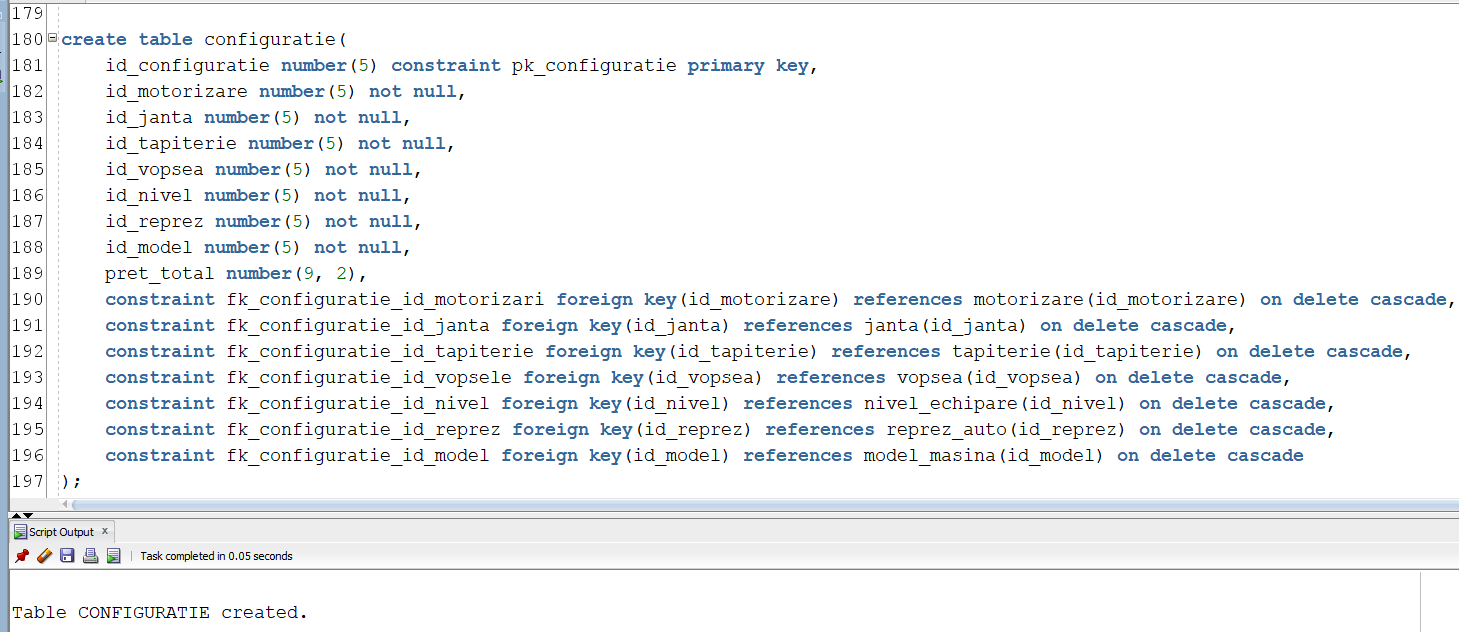
constraint fk\_configuratie\_id\_tapiterie foreign key(id\_tapiterie) references tapiterie(id\_tapiterie) on delete cascade,

constraint fk\_configuratie\_id\_vopsele foreign key(id\_vopsea) references vopsea(id\_vopsea) on delete cascade,

constraint fk\_configuratie\_id\_nivel foreign key(id\_nivel) references nivel\_echipare(id\_nivel) on delete cascade,

constraint fk\_configuratie\_id\_reprez foreign key(id\_reprez) references reprez\_auto(id\_reprez) on delete cascade,

constraint fk\_configuratie\_id\_model foreign key(id\_model) references model\_masina(id\_model) on delete cascade

);

* CONF\_EXTRAOP

create table conf\_extraop(

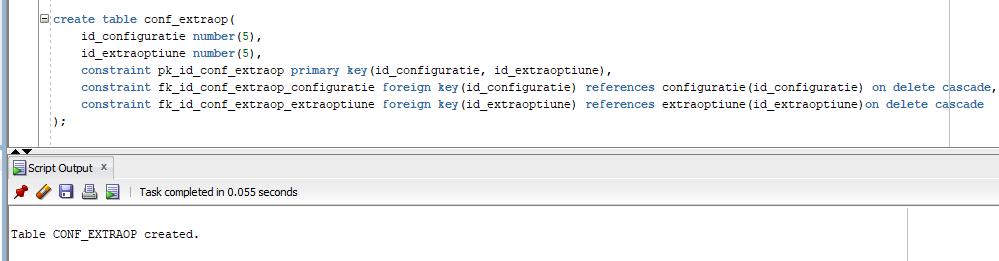
id\_configuratie number(5),

id\_extraoptiune number(5),

constraint pk\_id\_conf\_extraop primary key(id\_configuratie, id\_extraoptiune),

constraint fk\_id\_conf\_extraop\_configuratie foreign key(id\_configuratie) references configuratie(id\_configuratie) on delete cascade,

constraint fk\_id\_conf\_extraop\_extraoptiune foreign key(id\_extraoptiune) references extraoptiune(id\_extraoptiune)on delete cascade

);

* CLIENT

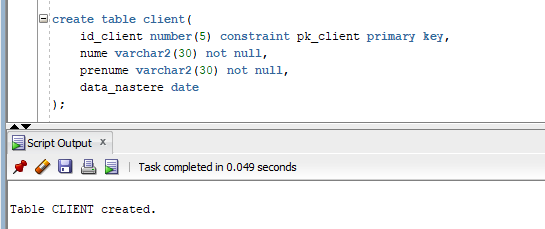
create table client(

id\_client number(5) constraint pk\_client primary key,

nume varchar2(30) not null,

prenume varchar2(30) not null,

data\_nastere date

);

* DEALER

create table dealer(

id\_dealer number(5) constraint pk\_dealer primary key,

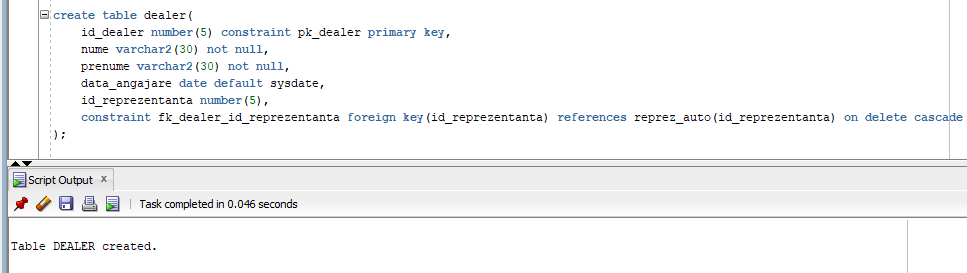
nume varchar2(30) not null,

prenume varchar2(30) not null,

data\_angajare date default sysdate,

id\_reprezentanta number(5),

constraint fk\_dealer\_id\_reprezentanta foreign key(id\_reprezentanta) references reprez\_auto(id\_reprezentanta) on delete cascade

);

* OFERTA

create table oferta(

id\_configuratie number(5),

id\_client number(5),

id\_dealer number(5),

pret\_oferta number(9, 2),

data\_inceput date,

data\_expirare date,

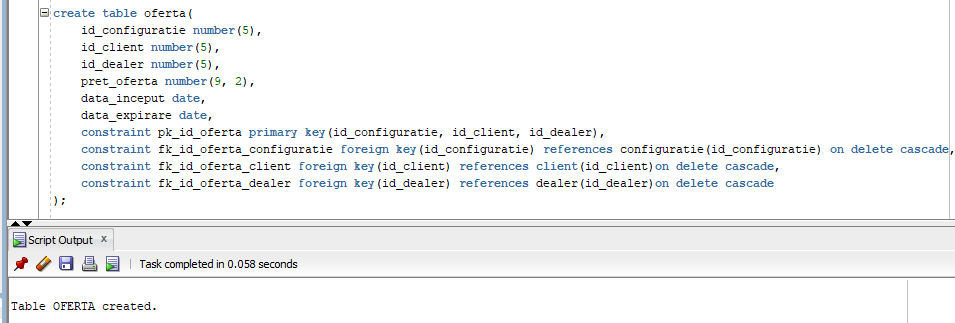
constraint pk\_id\_oferta primary key(id\_configuratie, id\_client, id\_dealer),

constraint fk\_id\_oferta\_configuratie foreign key(id\_configuratie) references configuratie(id\_configuratie) on delete cascade,

constraint fk\_id\_oferta\_client foreign key(id\_client) references client(id\_client)on delete cascade,

constraint fk\_id\_oferta\_dealer foreign key(id\_dealer) references dealer(id\_dealer)on delete cascade

);



## Inserarea datelor in tabele

Pentru inserarea datelor in tabele am folosit de o secventa:

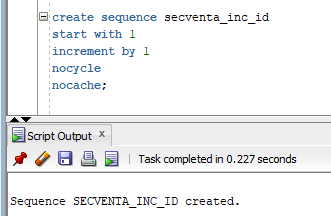
create sequence secventa\_inc\_id

start with 1

increment by 1

nocycle

nocache;



* FIRMA\_AUTO

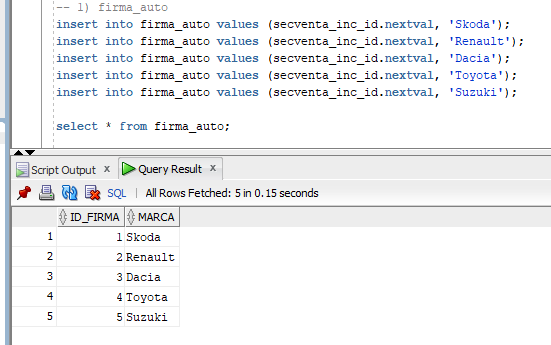
insert into firma\_auto values (secventa\_inc\_id.nextval, 'Skoda');

insert into firma\_auto values (secventa\_inc\_id.nextval, 'Renault');

insert into firma\_auto values (secventa\_inc\_id.nextval, 'Dacia');

insert into firma\_auto values (secventa\_inc\_id.nextval, 'Toyota');

insert into firma\_auto values (secventa\_inc\_id.nextval, 'Suzuki');



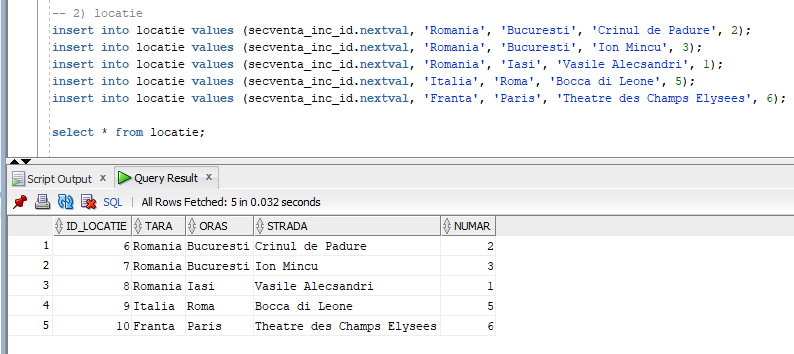
* LOCATIE

insert into locatie values (secventa\_inc\_id.nextval, 'Romania', 'Bucuresti', 'Crinul de Padure', 2);

insert into locatie values (secventa\_inc\_id.nextval, 'Romania', 'Bucuresti', 'Ion Mincu', 3);

insert into locatie values (secventa\_inc\_id.nextval, 'Romania', 'Iasi', 'Vasile Alecsandri', 1);

insert into locatie values (secventa\_inc\_id.nextval, 'Italia', 'Roma', 'Bocca di Leone', 5);

insert into locatie values (secventa\_inc\_id.nextval, 'Franta', 'Paris', 'Theatre des Champs Elysees', 6);

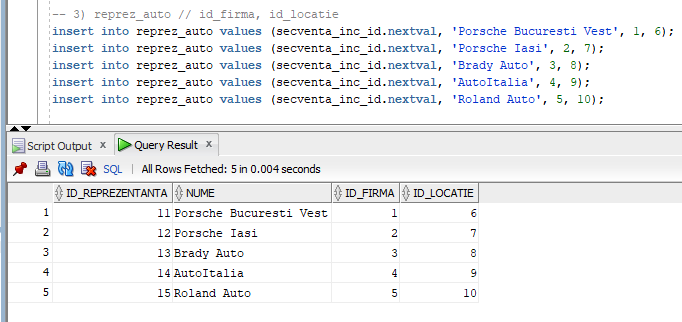
* REPREZ\_AUTO

insert into reprez\_auto values (secventa\_inc\_id.nextval, 'Porsche Bucuresti Vest', 1, 6);

insert into reprez\_auto values (secventa\_inc\_id.nextval, 'Porsche Iasi', 2, 7);

insert into reprez\_auto values (secventa\_inc\_id.nextval, 'Brady Auto', 3, 8);

insert into reprez\_auto values (secventa\_inc\_id.nextval, 'AutoItalia', 4, 9);

insert into reprez\_auto values (secventa\_inc\_id.nextval, 'Roland Auto', 5, 10);

* MODEL\_MASINA

insert into model\_masina values (secventa\_inc\_id.nextval, 'Octavia', 23000); -- skoda

insert into model\_masina values (secventa\_inc\_id.nextval, 'Fabia', 13000);

insert into model\_masina values (secventa\_inc\_id.nextval, 'Swift', 12000); --suzuki

insert into model\_masina values (secventa\_inc\_id.nextval, 'Vitara', 18000);

insert into model\_masina values (secventa\_inc\_id.nextval, 'Ignis', 14000);

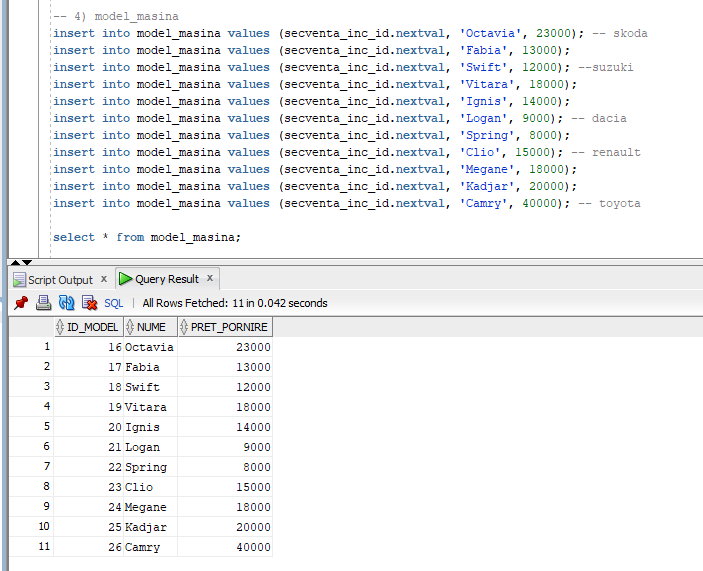
insert into model\_masina values (secventa\_inc\_id.nextval, 'Logan', 9000); -- dacia

insert into model\_masina values (secventa\_inc\_id.nextval, 'Spring', 8000);

insert into model\_masina values (secventa\_inc\_id.nextval, 'Clio', 15000); -- renault

insert into model\_masina values (secventa\_inc\_id.nextval, 'Megane', 18000);

insert into model\_masina values (secventa\_inc\_id.nextval, 'Kadjar', 20000);

insert into model\_masina values (secventa\_inc\_id.nextval, 'Camry', 40000); -- toyota

* MODELE\_DISP

insert into modele\_disp values(11, 16, 10);

insert into modele\_disp values(11, 17, 20);

insert into modele\_disp values(12, 18, 60);

insert into modele\_disp values(12, 19, 100);

insert into modele\_disp values(13, 20, 500);

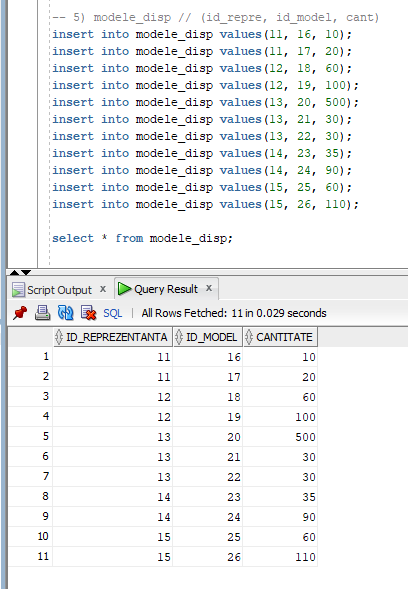
insert into modele\_disp values(13, 21, 30);

insert into modele\_disp values(13, 22, 30);

insert into modele\_disp values(14, 23, 35);

insert into modele\_disp values(14, 24, 90);

insert into modele\_disp values(15, 25, 60);

insert into modele\_disp values(15, 26, 110);

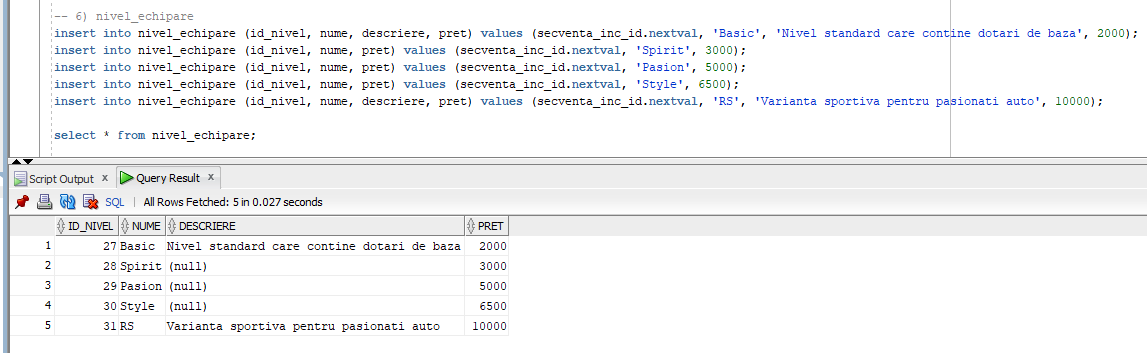
* NIVEL\_ECHIPARE

insert into nivel\_echipare (id\_nivel, nume, descriere, pret) values (secventa\_inc\_id.nextval, 'Basic', 'Nivel standard care contine dotari de baza', 2000);

insert into nivel\_echipare (id\_nivel, nume, pret) values (secventa\_inc\_id.nextval, 'Spirit', 3000);

insert into nivel\_echipare (id\_nivel, nume, pret) values (secventa\_inc\_id.nextval, 'Pasion', 5000);

insert into nivel\_echipare (id\_nivel, nume, pret) values (secventa\_inc\_id.nextval, 'Style', 6500);

insert into nivel\_echipare (id\_nivel, nume, descriere, pret) values (secventa\_inc\_id.nextval, 'RS', 'Varianta sportiva pentru pasionati auto', 10000);

* NIV\_ECH\_DISP

insert into niv\_ech\_disp values(27, 17);

insert into niv\_ech\_disp values(27, 18);

insert into niv\_ech\_disp values(27, 21);

insert into niv\_ech\_disp values(27, 22);

insert into niv\_ech\_disp values(27, 23);

insert into niv\_ech\_disp values(27, 24);

insert into niv\_ech\_disp values(27, 25);

insert into niv\_ech\_disp values(28, 18);

insert into niv\_ech\_disp values(28, 19);

insert into niv\_ech\_disp values(28, 20);

insert into niv\_ech\_disp values(28, 23);

insert into niv\_ech\_disp values(28, 24);

insert into niv\_ech\_disp values(28, 25);

insert into niv\_ech\_disp values(28, 26);

insert into niv\_ech\_disp values(29, 18);

insert into niv\_ech\_disp values(29, 19);

insert into niv\_ech\_disp values(29, 20);

insert into niv\_ech\_disp values(29, 23);

insert into niv\_ech\_disp values(29, 24);

insert into niv\_ech\_disp values(29, 25);

insert into niv\_ech\_disp values(30, 16);

insert into niv\_ech\_disp values(30, 17);

insert into niv\_ech\_disp values(30, 21);

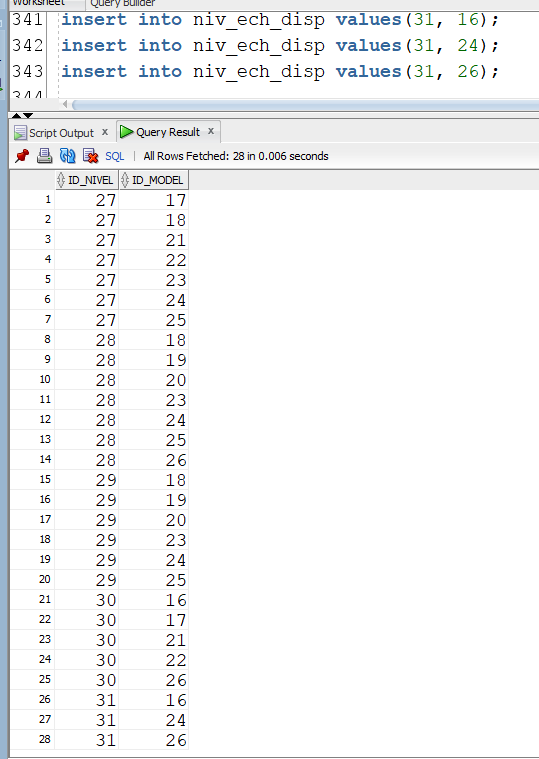
insert into niv\_ech\_disp values(30, 22);

insert into niv\_ech\_disp values(30, 26);

insert into niv\_ech\_disp values(31, 16);

insert into niv\_ech\_disp values(31, 24);

insert into niv\_ech\_disp values(31, 26);



* MOTORIZARE

insert into motorizare(id\_motorizare, combustibil, putere, capacitate, transmisie, consum, emisii, pret) values (secventa\_inc\_id.nextval, 'diesel', 170, 2000, 'Automata', 7, 130, 2500);

insert into motorizare(id\_motorizare, combustibil, putere, capacitate, transmisie, consum, emisii, pret) values (secventa\_inc\_id.nextval, 'diesel', 150, 2200, 'Manuala', 8, 135, 2000);

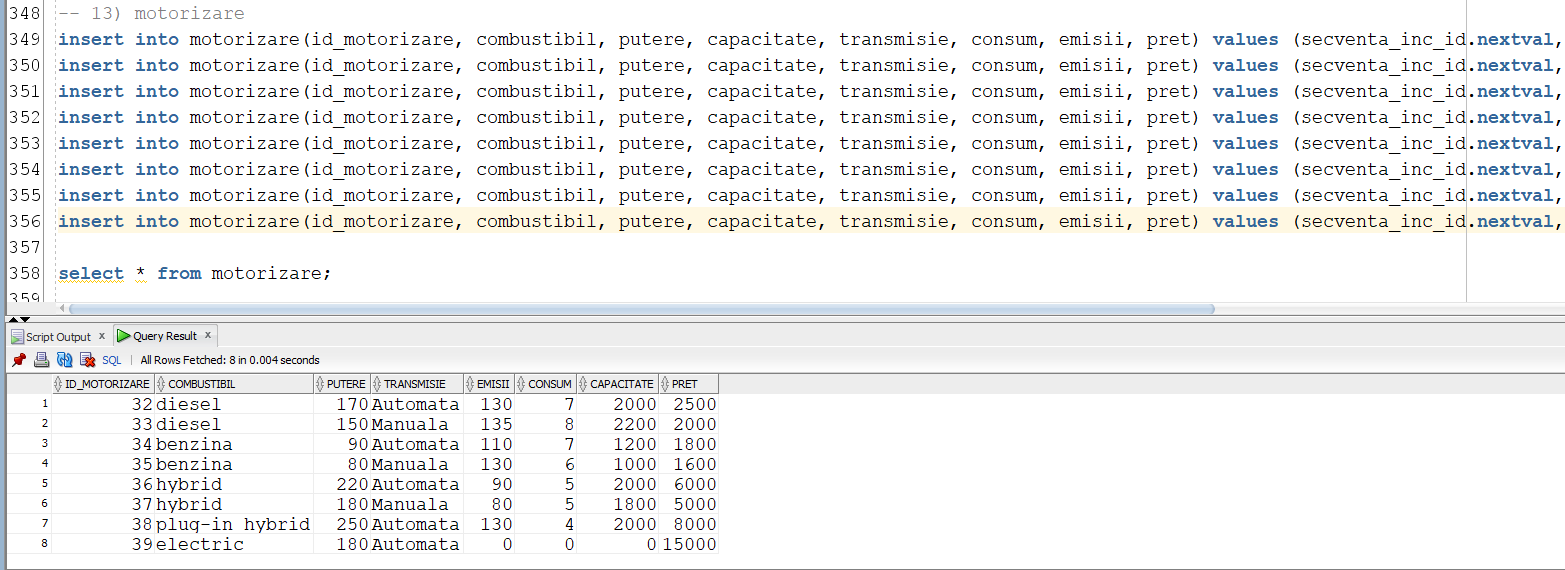
insert into motorizare(id\_motorizare, combustibil, putere, capacitate, transmisie, consum, emisii, pret) values (secventa\_inc\_id.nextval, 'benzina', 90, 1200, 'Automata', 7, 110, 1800);

insert into motorizare(id\_motorizare, combustibil, putere, capacitate, transmisie, consum, emisii, pret) values (secventa\_inc\_id.nextval, 'benzina', 80, 1000, 'Manuala', 6, 130, 1600);

insert into motorizare(id\_motorizare, combustibil, putere, capacitate, transmisie, consum, emisii, pret) values (secventa\_inc\_id.nextval, 'hybrid', 220, 2000, 'Automata', 5, 90, 6000);

insert into motorizare(id\_motorizare, combustibil, putere, capacitate, transmisie, consum, emisii, pret) values (secventa\_inc\_id.nextval, 'hybrid', 180, 1800, 'Manuala', 5, 80, 5000);

insert into motorizare(id\_motorizare, combustibil, putere, capacitate, transmisie, consum, emisii, pret) values (secventa\_inc\_id.nextval, 'plug-in hybrid', 250, 2000, 'Automata', 4, 130, 8000);

insert into motorizare(id\_motorizare, combustibil, putere, capacitate, transmisie, consum, emisii, pret) values (secventa\_inc\_id.nextval, 'electric', 180, 0, 'Automata', 0, 0, 15000);

* JANTA

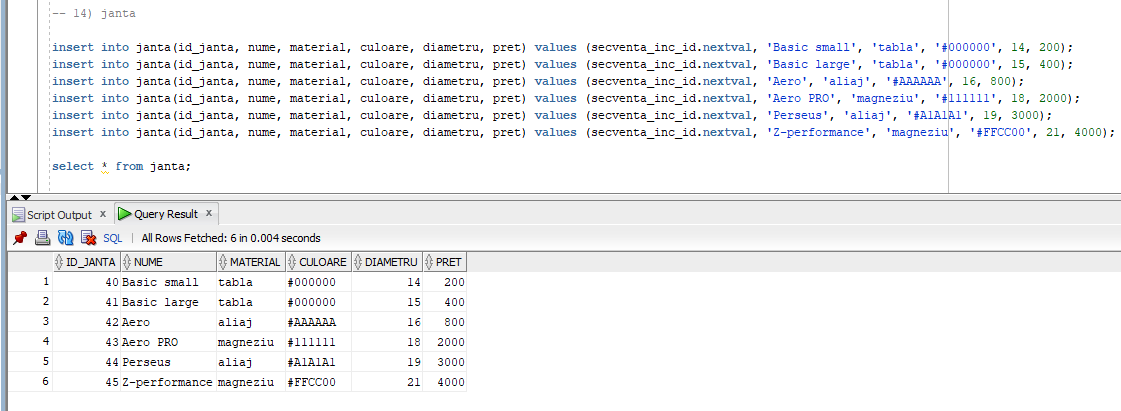
insert into janta(id\_janta, nume, material, culoare, diametru, pret) values (secventa\_inc\_id.nextval, 'Basic small', 'tabla', '#000000', 14, 200);

insert into janta(id\_janta, nume, material, culoare, diametru, pret) values (secventa\_inc\_id.nextval, 'Basic large', 'tabla', '#000000', 15, 400);

insert into janta(id\_janta, nume, material, culoare, diametru, pret) values (secventa\_inc\_id.nextval, 'Aero', 'aliaj', '#AAAAAA', 16, 800);

insert into janta(id\_janta, nume, material, culoare, diametru, pret) values (secventa\_inc\_id.nextval, 'Aero PRO', 'magneziu', '#111111', 18, 2000);

insert into janta(id\_janta, nume, material, culoare, diametru, pret) values (secventa\_inc\_id.nextval, 'Perseus', 'aliaj', '#A1A1A1', 19, 3000);

insert into janta(id\_janta, nume, material, culoare, diametru, pret) values (secventa\_inc\_id.nextval, 'Z-performance', 'magneziu', '#FFCC00', 21, 4000);

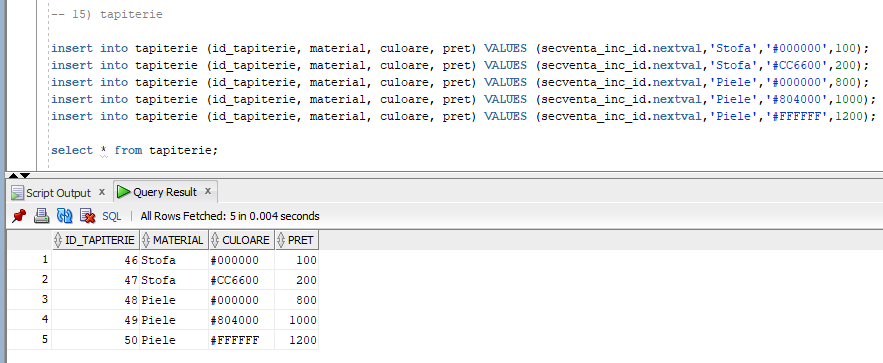
* TAPITERIE

insert into tapiterie (id\_tapiterie, material, culoare, pret) VALUES (secventa\_inc\_id.nextval,'Stofa','#000000',100);

insert into tapiterie (id\_tapiterie, material, culoare, pret) VALUES (secventa\_inc\_id.nextval,'Stofa','#CC6600',200);

insert into tapiterie (id\_tapiterie, material, culoare, pret) VALUES (secventa\_inc\_id.nextval,'Piele','#000000',800);

insert into tapiterie (id\_tapiterie, material, culoare, pret) VALUES (secventa\_inc\_id.nextval,'Piele','#804000',1000);

insert into tapiterie (id\_tapiterie, material, culoare, pret) VALUES (secventa\_inc\_id.nextval,'Piele','#FFFFFF',1200);

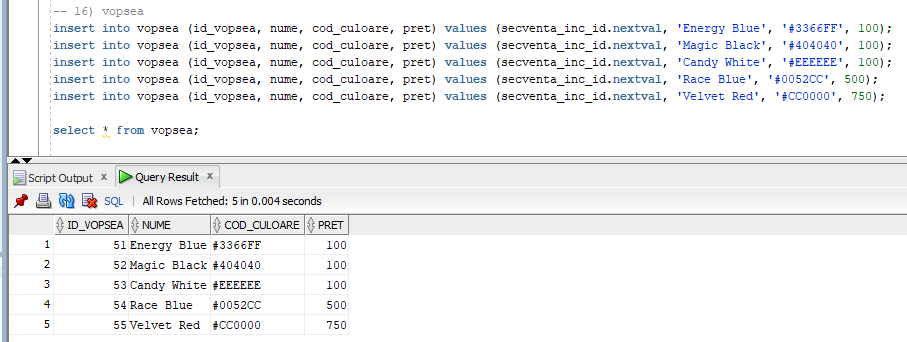
* VOPSEA

insert into vopsea (id\_vopsea, nume, cod\_culoare, pret) values (secventa\_inc\_id.nextval, 'Energy Blue', '#3366FF', 100);

insert into vopsea (id\_vopsea, nume, cod\_culoare, pret) values (secventa\_inc\_id.nextval, 'Magic Black', '#404040', 100);

insert into vopsea (id\_vopsea, nume, cod\_culoare, pret) values (secventa\_inc\_id.nextval, 'Candy White', '#EEEEEE', 100);

insert into vopsea (id\_vopsea, nume, cod\_culoare, pret) values (secventa\_inc\_id.nextval, 'Race Blue', '#0052CC', 500);

insert into vopsea (id\_vopsea, nume, cod\_culoare, pret) values (secventa\_inc\_id.nextval, 'Velvet Red', '#CC0000', 750);

* EXTRAOPTIUNE

insert into extraoptiune(id\_extraoptiune, nume, descriere, pret) values(secventa\_inc\_id.nextval, 'Oglinzi incalzite', 'Incalzire', 200);

insert into extraoptiune(id\_extraoptiune, nume, descriere, pret) values(secventa\_inc\_id.nextval, 'Scaune incalzite', '', 200);

insert into extraoptiune(id\_extraoptiune, nume, descriere, pret) values(secventa\_inc\_id.nextval, 'Scaune ventilate', 'Scaune care impiedica transpiratia', 500);

insert into extraoptiune(id\_extraoptiune, nume, descriere, pret) values(secventa\_inc\_id.nextval, 'Parbriz cu degivrare rapida', 'Incalzire in parbriz', 300);

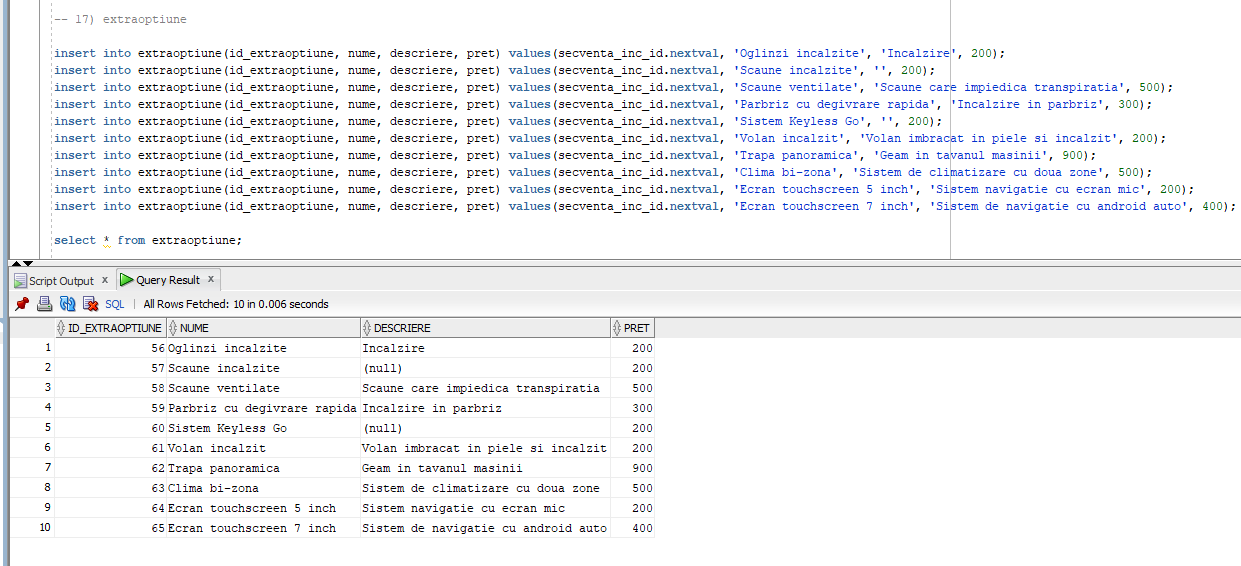
insert into extraoptiune(id\_extraoptiune, nume, descriere, pret) values(secventa\_inc\_id.nextval, 'Sistem Keyless Go', '', 200);

insert into extraoptiune(id\_extraoptiune, nume, descriere, pret) values(secventa\_inc\_id.nextval, 'Volan incalzit', 'Volan imbracat in piele si incalzit', 200);

insert into extraoptiune(id\_extraoptiune, nume, descriere, pret) values(secventa\_inc\_id.nextval, 'Trapa panoramica', 'Geam in tavanul masinii', 900);

insert into extraoptiune(id\_extraoptiune, nume, descriere, pret) values(secventa\_inc\_id.nextval, 'Clima bi-zona', 'Sistem de climatizare cu doua zone', 500);

insert into extraoptiune(id\_extraoptiune, nume, descriere, pret) values(secventa\_inc\_id.nextval, 'Ecran touchscreen 5 inch', 'Sistem navigatie cu ecran mic', 200);

insert into extraoptiune(id\_extraoptiune, nume, descriere, pret) values(secventa\_inc\_id.nextval, 'Ecran touchscreen 7 inch', 'Sistem de navigatie cu android auto', 400);

* MOTORIZARE\_NIV

insert into motorizare\_niv values(27, 35);

insert into motorizare\_niv values(27, 33);

insert into motorizare\_niv values(27, 37);

insert into motorizare\_niv values(28, 32);

insert into motorizare\_niv values(28, 33);

insert into motorizare\_niv values(28, 34);

insert into motorizare\_niv values(28, 35);

insert into motorizare\_niv values(28, 36);

insert into motorizare\_niv values(28, 37);

insert into motorizare\_niv values(29, 33);

insert into motorizare\_niv values(29, 34);

insert into motorizare\_niv values(29, 35);

insert into motorizare\_niv values(30, 32);

insert into motorizare\_niv values(30, 33);

insert into motorizare\_niv values(30, 34);

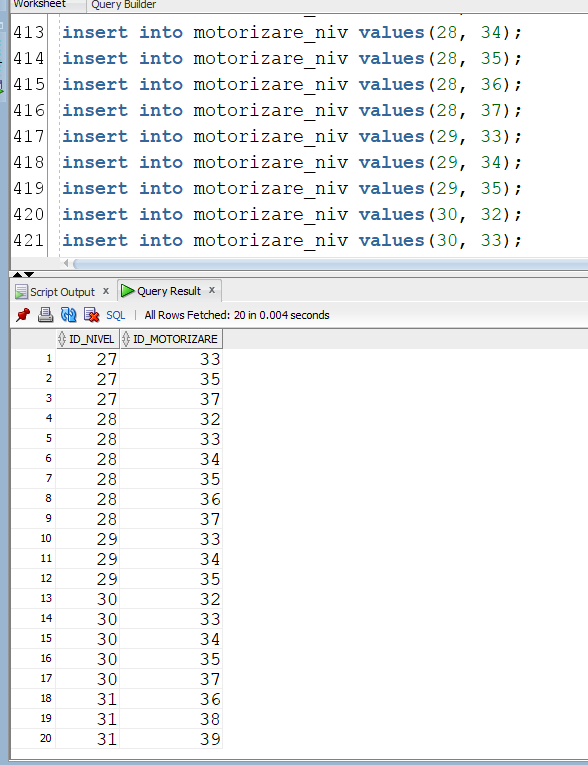
insert into motorizare\_niv values(30, 35);

insert into motorizare\_niv values(30, 37);

insert into motorizare\_niv values(31, 36);

insert into motorizare\_niv values(31, 38);

insert into motorizare\_niv values(31, 39);



* JANTA\_NIV

insert into janta\_niv values(27, 40);

insert into janta\_niv values(27, 41);

insert into janta\_niv values(28, 41);

insert into janta\_niv values(28, 42);

insert into janta\_niv values(29, 41);

insert into janta\_niv values(29, 42);

insert into janta\_niv values(29, 44);

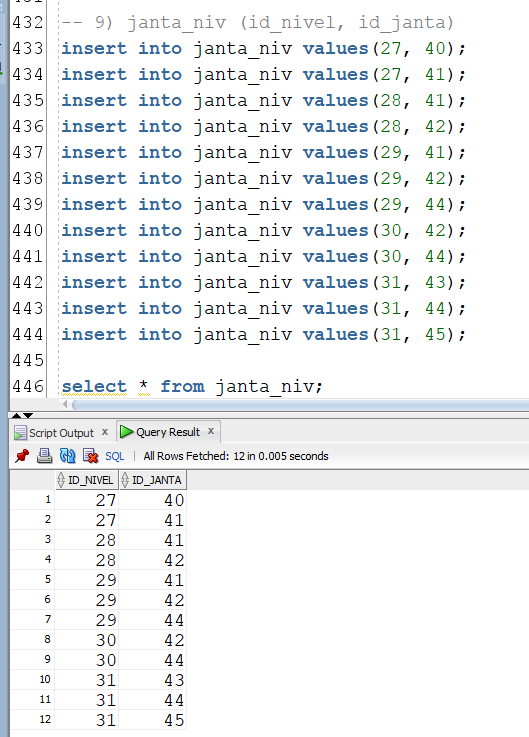
insert into janta\_niv values(30, 42);

insert into janta\_niv values(30, 44);

insert into janta\_niv values(31, 43);

insert into janta\_niv values(31, 44);

insert into janta\_niv values(31, 45);



* TAPIT\_NIV

insert into tapit\_niv values(27, 46);

insert into tapit\_niv values(27, 47);

insert into tapit\_niv values(28, 46);

insert into tapit\_niv values(28, 47);

insert into tapit\_niv values(29, 46);

insert into tapit\_niv values(29, 47);

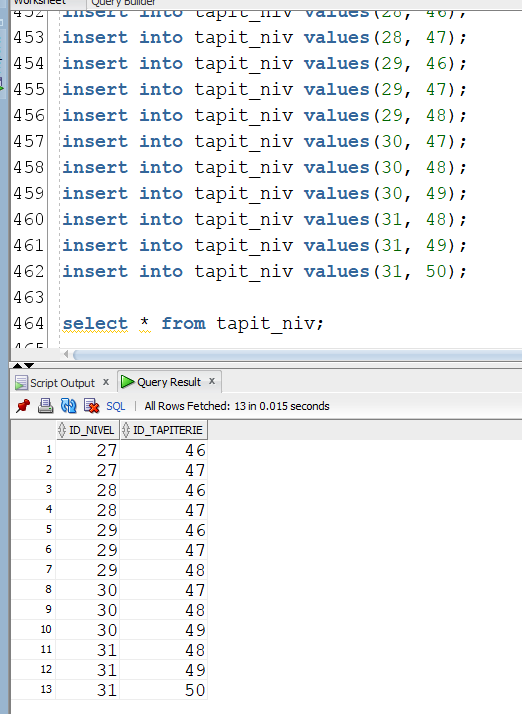
insert into tapit\_niv values(29, 48);

insert into tapit\_niv values(30, 47);

insert into tapit\_niv values(30, 48);

insert into tapit\_niv values(30, 49);

insert into tapit\_niv values(31, 48);

insert into tapit\_niv values(31, 49);insert into tapit\_niv values(31, 50);

* VOPSEA\_NIV

insert into vopsea\_niv values(27, 51);

insert into vopsea\_niv values(27, 52);

insert into vopsea\_niv values(27, 53);

insert into vopsea\_niv values(28, 52);

insert into vopsea\_niv values(28, 53);

insert into vopsea\_niv values(29, 51);

insert into vopsea\_niv values(29, 52);

insert into vopsea\_niv values(30, 51);

insert into vopsea\_niv values(30, 54);

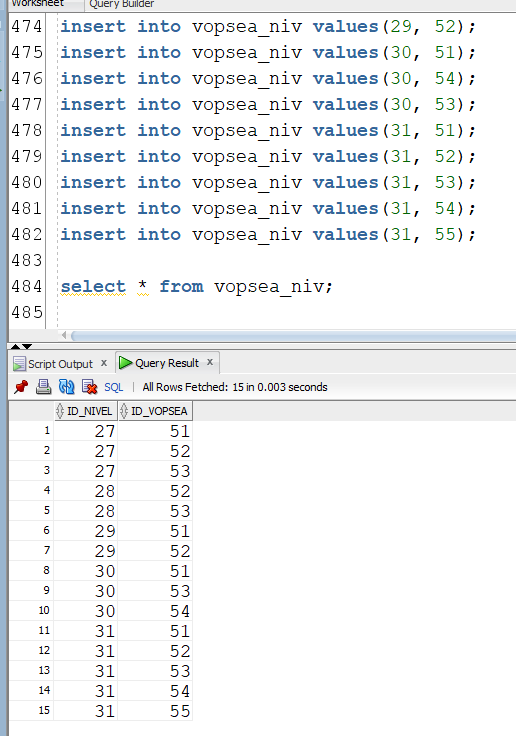
insert into vopsea\_niv values(30, 53);

insert into vopsea\_niv values(31, 51);

insert into vopsea\_niv values(31, 52);

insert into vopsea\_niv values(31, 53);

insert into vopsea\_niv values(31, 54);

insert into vopsea\_niv values(31, 55);

* EXTRAOP\_NIV

insert into extraop\_niv values(27, 56);

insert into extraop\_niv values(27, 57);

insert into extraop\_niv values(27, 64);

insert into extraop\_niv values(28, 56);

insert into extraop\_niv values(28, 57);

insert into extraop\_niv values(28, 64);

insert into extraop\_niv values(28, 65);

insert into extraop\_niv values(29, 56);

insert into extraop\_niv values(29, 57);

insert into extraop\_niv values(29, 63);

insert into extraop\_niv values(29, 65);

insert into extraop\_niv values(27, 65);

insert into extraop\_niv values(30, 56);

insert into extraop\_niv values(30, 57);

insert into extraop\_niv values(30, 59);

insert into extraop\_niv values(30, 60);

insert into extraop\_niv values(30, 62);

insert into extraop\_niv values(30, 63);

insert into extraop\_niv values(30, 65);

insert into extraop\_niv values(31, 56);

insert into extraop\_niv values(31, 57);

insert into extraop\_niv values(31, 58);

insert into extraop\_niv values(31, 59);

insert into extraop\_niv values(31, 60);

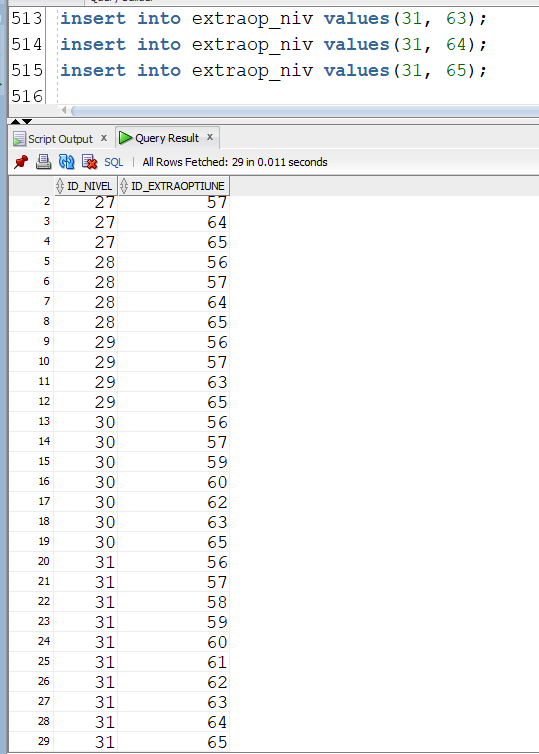
insert into extraop\_niv values(31, 61);

insert into extraop\_niv values(31, 62);

insert into extraop\_niv values(31, 63);

insert into extraop\_niv values(31, 64);

insert into extraop\_niv values(31, 65);

* CONFIGURATIE

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 32, 42, 47, 51, 30, 11, 16);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 32, 44, 49, 53, 31, 11, 16);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 33, 40, 46, 52, 27, 11, 17);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 35, 41, 47, 51, 27, 11, 17);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 37, 44, 48, 54, 30, 11, 17);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 35, 40, 46, 51, 27, 12, 18);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 37, 41, 47, 52, 28, 12, 18);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 32, 42, 46, 53, 28, 12, 19);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 34, 44, 48, 52, 29, 12, 19);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 33, 41, 46, 53, 28, 13, 20);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 35, 42, 47, 52, 29, 13, 20);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 37, 40, 46, 51, 27, 13, 21);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 32, 44, 49, 54, 30, 13, 21);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 34, 42, 48, 51, 30, 13, 22);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 37, 40, 46, 53, 27, 14, 23);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 35, 41, 47, 52, 28, 14, 23);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 33, 42, 48, 51, 29, 14, 24);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 35, 41, 46, 52, 28, 14, 24);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 37, 43, 50, 55, 31, 14, 24);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 33, 40, 46, 53, 27, 15, 25);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 35, 41, 47, 52, 28, 15, 25);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

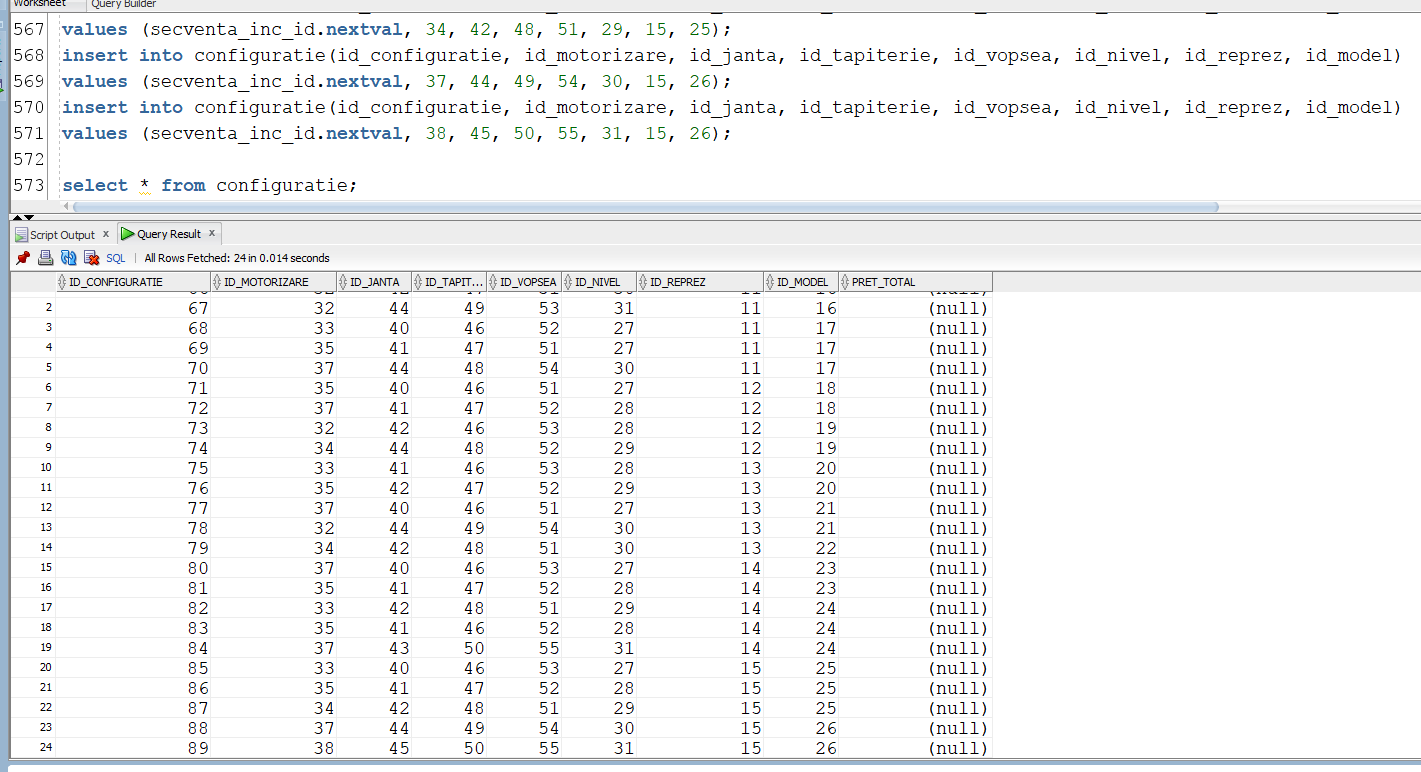
values (secventa\_inc\_id.nextval, 34, 42, 48, 51, 29, 15, 25);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 37, 44, 49, 54, 30, 15, 26);

insert into configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

values (secventa\_inc\_id.nextval, 38, 45, 50, 55, 31, 15, 26);



* CONF\_EXTRAOP

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (66, 56);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (66, 65);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (67, 57);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (67, 58);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (68, 56);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (68, 57);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (68, 64);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (69, 56);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (70, 62);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (70, 60);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (71, 56);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (71, 57);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (71, 64);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (72, 65);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (73, 57);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (74, 63);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (75, 56);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (76, 63);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (77, 64);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (78, 59);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (79, 62);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (79, 60);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (81, 57);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (81, 64);

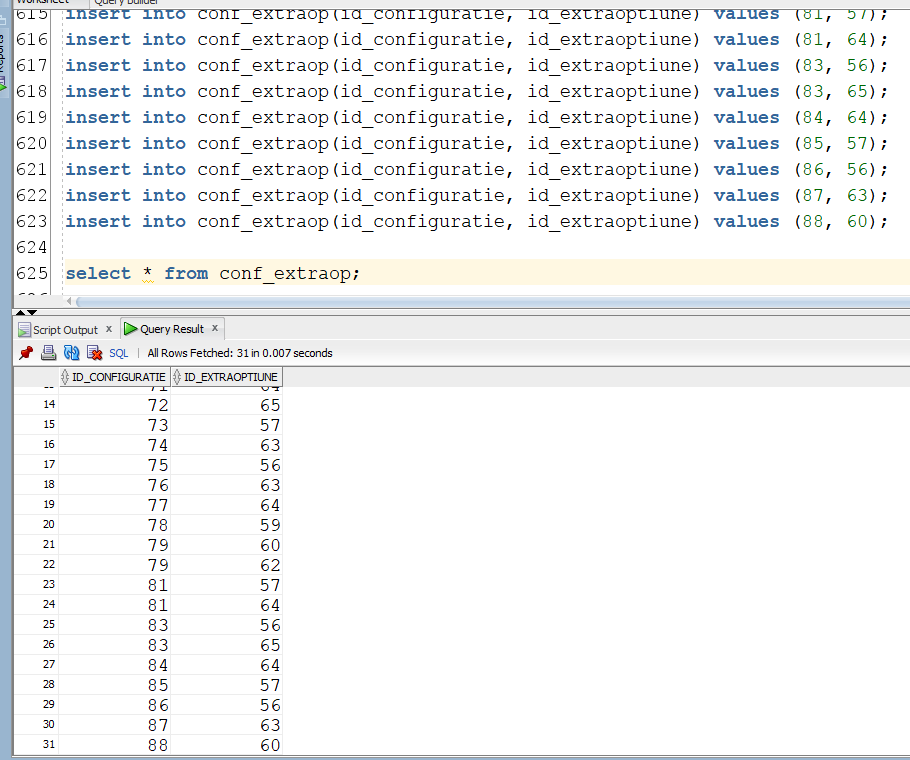
insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (83, 56);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (83, 65);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (84, 64);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (85, 57);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (86, 56);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (87, 63);

insert into conf\_extraop(id\_configuratie, id\_extraoptiune) values (88, 60);

* CLIENT

insert into client (id\_client, nume, prenume, data\_nastere) values (secventa\_inc\_id.nextval, 'Ion', 'Ionut', '28-FEB-1975');

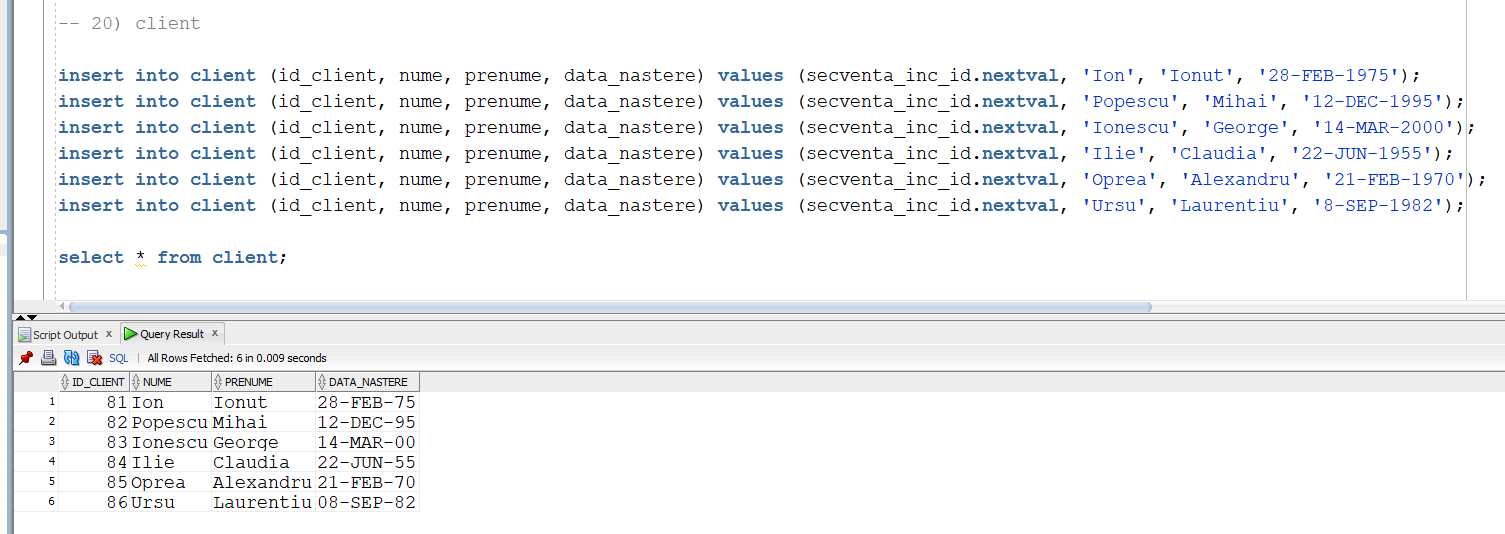
insert into client (id\_client, nume, prenume, data\_nastere) values (secventa\_inc\_id.nextval, 'Popescu', 'Mihai', '12-DEC-1995');

insert into client (id\_client, nume, prenume, data\_nastere) values (secventa\_inc\_id.nextval, 'Ionescu', 'George', '14-MAR-2000');

insert into client (id\_client, nume, prenume, data\_nastere) values (secventa\_inc\_id.nextval, 'Ilie', 'Claudia', '22-JUN-1955');

insert into client (id\_client, nume, prenume, data\_nastere) values (secventa\_inc\_id.nextval, 'Oprea', 'Alexandru', '21-FEB-1970');

insert into client (id\_client, nume, prenume, data\_nastere) values (secventa\_inc\_id.nextval, 'Ursu', 'Laurentiu', '8-SEP-1982');



* DEALER

insert into dealer (id\_dealer, nume, prenume, data\_angajare, id\_reprez) values (secventa\_inc\_id.nextval, 'Moraru', 'Stefan', '6-SEP-2021', 11);

insert into dealer (id\_dealer, nume, prenume, data\_angajare, id\_reprez) values (secventa\_inc\_id.nextval, 'Varzan', 'Gheorghe', '12-DEC-2021', 12);

insert into dealer (id\_dealer, nume, prenume, data\_angajare, id\_reprez) values (secventa\_inc\_id.nextval, 'Surdu', 'Mihai', '2-NOV-2019', 13);

insert into dealer (id\_dealer, nume, prenume, data\_angajare, id\_reprez) values (secventa\_inc\_id.nextval, 'Georgescu', 'Adrian', '26-FEB-2021', 14);

insert into dealer (id\_dealer, nume, prenume, data\_angajare, id\_reprez) values (secventa\_inc\_id.nextval, 'Goga', 'Albert', '11-MAY-2021', 15);

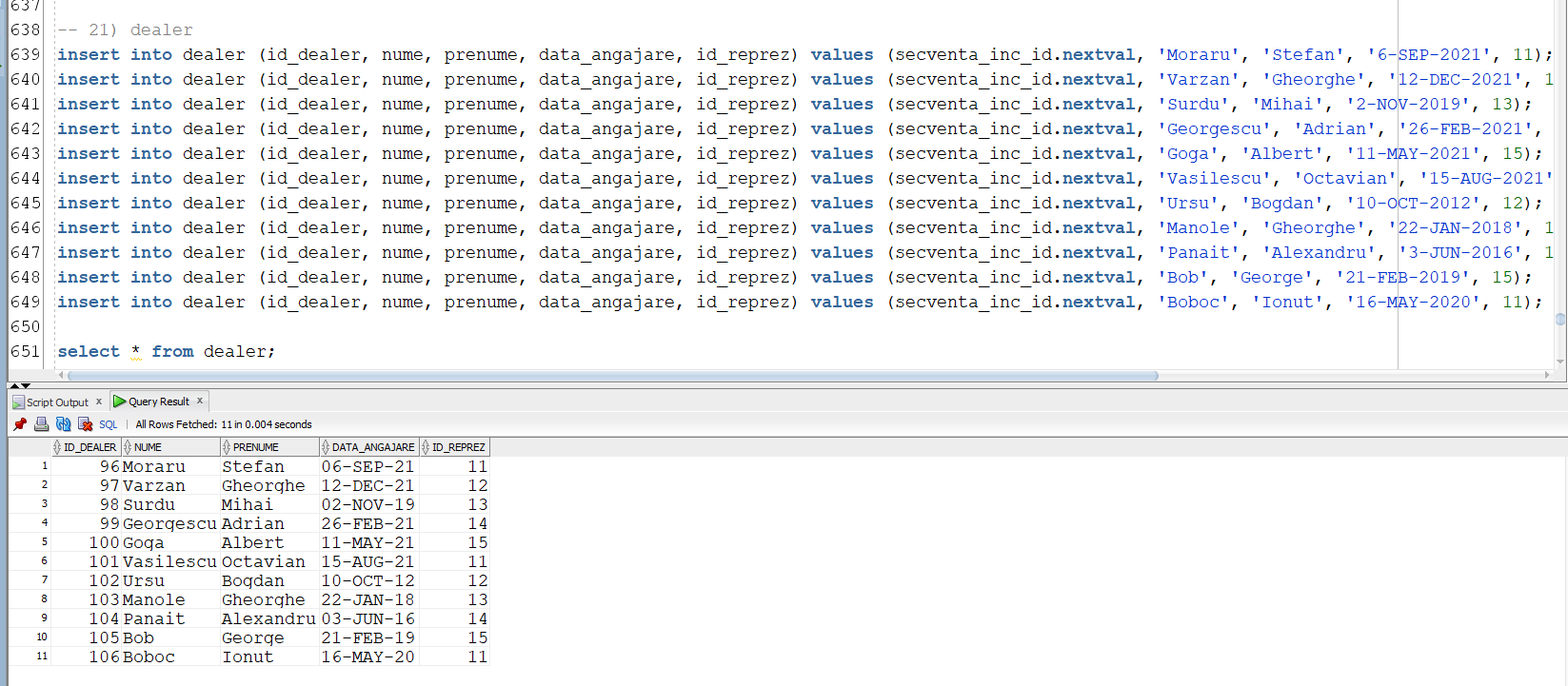
insert into dealer (id\_dealer, nume, prenume, data\_angajare, id\_reprez) values (secventa\_inc\_id.nextval, 'Vasilescu', 'Octavian', '15-AUG-2021', 11);

insert into dealer (id\_dealer, nume, prenume, data\_angajare, id\_reprez) values (secventa\_inc\_id.nextval, 'Ursu', 'Bogdan', '10-OCT-2012', 12);

insert into dealer (id\_dealer, nume, prenume, data\_angajare, id\_reprez) values (secventa\_inc\_id.nextval, 'Manole', 'Gheorghe', '22-JAN-2018', 13);

insert into dealer (id\_dealer, nume, prenume, data\_angajare, id\_reprez) values (secventa\_inc\_id.nextval, 'Panait', 'Alexandru', '3-JUN-2016', 14);

insert into dealer (id\_dealer, nume, prenume, data\_angajare, id\_reprez) values (secventa\_inc\_id.nextval, 'Bob', 'George', '21-FEB-2019', 15);

insert into dealer (id\_dealer, nume, prenume, data\_angajare, id\_reprez) values (secventa\_inc\_id.nextval, 'Boboc', 'Ionut', '16-MAY-2020', 11);

* OFERTA

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (66, 90, 96, 29002, '15-AUG-2020', '15-AUG-2021');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (66, 91, 101, 12320, '23-JUL-2021', '23-SEP-2021');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (66, 92, 101, 29002, '8-OCT-2019', '8-NOV-2020');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (66, 93, 96, 12320, '22-JUL-2018', '22-AUG-2019');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (66, 94, 96, 29002, '9-MAR-2023', '9-APR-2024');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (67, 95, 106, 12071, '14-JAN-2021', '14-JAN-2023');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (67, 90, 101, 23683, '1-NOV-2021', '1-NOV-2022');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (68, 91, 96, 18657, '4-FEB-2019', '4-MAR-2019');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (68, 92, 101, 26654, '10-APR-2022', '10-APR-2023');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (69, 93, 106, 17463, '11-DEC-2024', '11-DEC-2026');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (69, 94, 96, 8919, '8-APR-2020', '20-APR-2020');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (70, 95, 96, 28684, '20-JUN-2022', '20-JUL-2023');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (71, 90, 102, 16856, '1-JUL-2020', '30-JUL-2020');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (72, 91, 97, 28684, '18-JAN-2021', '21-FEB-2021');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (72, 92, 97, 16856, '2-APR-2022', '9-APR-2022');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (73, 93, 102, 28684, '13-SEP-2022', '13-OCT-2023');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (74, 94, 97, 16856, '22-MAY-2022', '22-MAY-2023');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (75, 95, 103, 28684, '17-OCT-2020', '17-OCT-2024');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (76, 90, 98, 16856, '14-NOV-2022', '14-NOV-2024');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (77, 91, 103, 28684, '24-SEP-2021', '24-DEC-2021');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (77, 92, 103, 16856, '4-JAN-2020', '4-JAN-2021');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (78, 93, 103, 28684, '24-SEP-2021', '24-DEC-2021');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (78, 94, 98, 16856, '4-FEB-2020', '4-FEB-2021');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (78, 95, 98, 16856, '13-JAN-2022', '13-JAN-2023');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (79, 90, 98, 16856, '25-APR-2016', '25-MAY-2017');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (80, 91, 104, 16856, '21-DEC-2018', '21-JAN-2020');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (81, 92, 104, 16856, '13-JAN-2023', '13-JAN-2024');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (81, 93, 99, 16856, '4-JAN-2020', '4-JAN-2021');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (82, 94, 99, 16856, '24-SEP-2021', '24-DEC-2021');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (83, 95, 104, 16856, '13-JAN-2022', '13-JAN-2023');

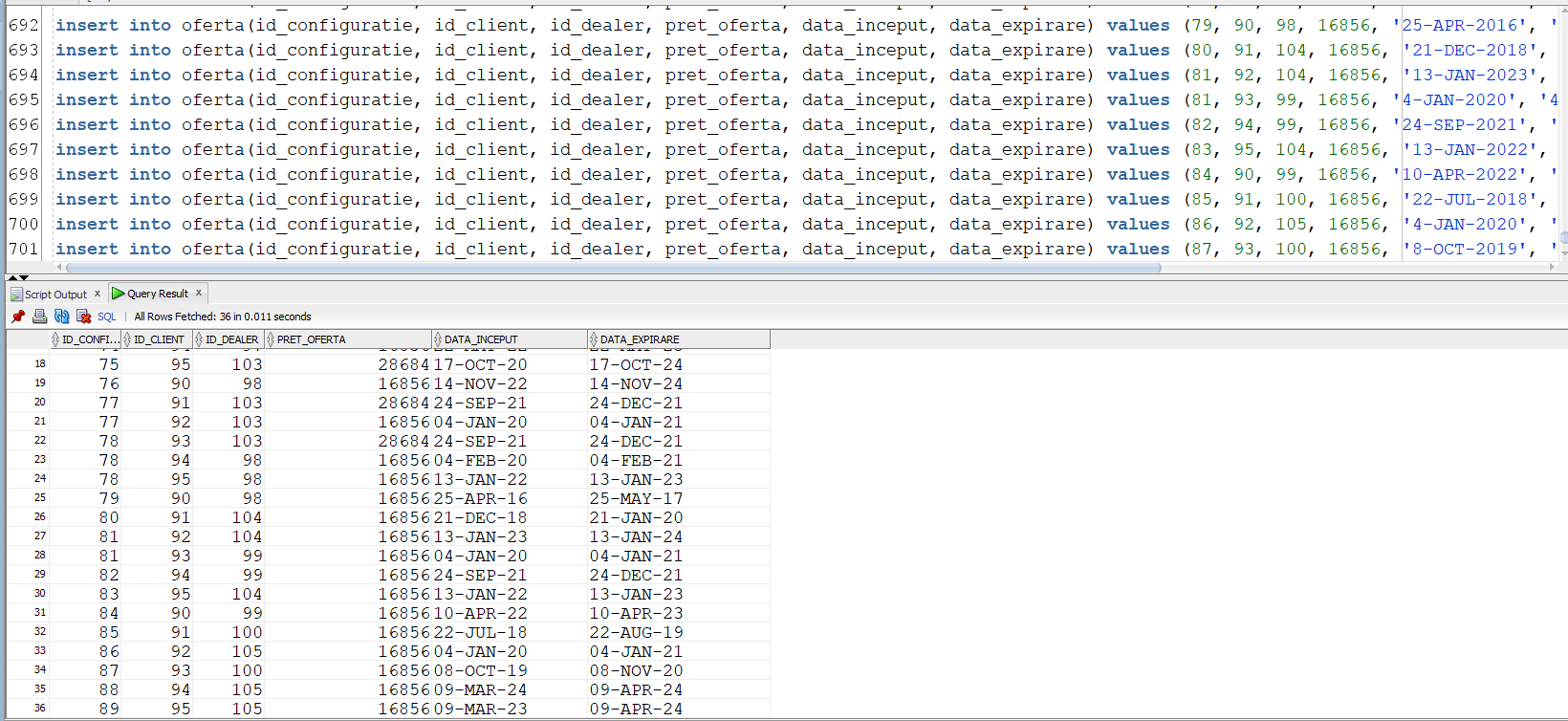
insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (84, 90, 99, 16856, '10-APR-2022', '10-APR-2023');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (85, 91, 100, 16856, '22-JUL-2018', '22-AUG-2019');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (86, 92, 105, 16856, '4-JAN-2020', '4-JAN-2021');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (87, 93, 100, 16856, '8-OCT-2019', '8-NOV-2020');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (88, 94, 105, 16856, '9-MAR-2024', '9-APR-2024');

insert into oferta(id\_configuratie, id\_client, id\_dealer, pret\_oferta, data\_inceput, data\_expirare) values (89, 95, 105, 16856, '9-MAR-2023', '9-APR-2024');

## Formulați in limbaj natural o problema pe care sa o rezolvați folosind un subprogram stocat independent care sa utilizeze două tipuri deferite de colecții studiate. Apelați subprogramul.

**Enunț problemă**: Definiți un subprogram prin care să se afișeze toate ofertele care au expirat (se va afișa doar id\_configuratie, dealer, client, pret) și să se actualizeze perioadele de valabilitate a ofertelor ca începând de azi și cu durata de 2 ani.

Pentru rezolvarea cerinței a fost folosit un subprogram stocat independent de tip **procedură** și următoarele două **colecții** studiate:

* tablou imbricat
* tablou indexat

CREATE OR REPLACE PROCEDURE actualiz\_oferte\_exp IS

TYPE oferta\_record IS RECORD

(cod\_config oferta.id\_configuratie%TYPE,

pret oferta.pret\_oferta%TYPE,

cod\_client oferta.id\_client%TYPE,

cod\_dealer oferta.id\_dealer%TYPE);

TYPE t\_tbl\_imbricat IS TABLE OF oferta\_record;

t\_oferta t\_tbl\_imbricat;

TYPE dealer\_record IS RECORD

(cod\_dealer dealer.id\_dealer%TYPE,

nume dealer.nume%TYPE,

prenume dealer.prenume%TYPE);

TYPE t\_tbl\_indx\_dealer IS TABLE OF dealer\_record INDEX BY PLS\_INTEGER;

t\_dealer t\_tbl\_indx\_dealer := t\_tbl\_indx\_dealer();

TYPE client\_record IS RECORD

(cod\_client client.id\_client%TYPE,

nume client.nume%TYPE,

prenume client.prenume%TYPE);

TYPE t\_tbl\_indx\_client IS TABLE OF client\_record INDEX BY PLS\_INTEGER;

t\_client t\_tbl\_indx\_client := t\_tbl\_indx\_client();

are\_oferte\_expirate BOOLEAN;

BEGIN

SELECT id\_dealer, nume, prenume BULK COLLECT INTO t\_dealer

FROM dealer

ORDER BY nume;

SELECT id\_client, nume, prenume BULK COLLECT INTO t\_client

FROM client;

SELECT id\_configuratie, pret\_oferta, id\_client, id\_dealer BULK COLLECT INTO t\_oferta

FROM oferta

WHERE data\_expirare < sysdate;

IF t\_oferta.COUNT = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista oferte expirate');

RETURN;

END IF;

DBMS\_OUTPUT.PUT\_LINE('Ofertele expirate sunt:');

FOR i IN t\_dealer.FIRST..t\_dealer.LAST LOOP

DBMS\_OUTPUT.PUT\_LINE('- Dealerul ' || t\_dealer(i).nume ||' '||t\_dealer(i).prenume||': ');

are\_oferte\_expirate := false;

FOR j IN t\_oferta.FIRST..t\_oferta.LAST LOOP

IF t\_oferta(j).cod\_dealer = t\_dealer(i).cod\_dealer THEN

are\_oferte\_expirate := true;

DBMS\_OUTPUT.PUT(' -> oferta pentru configuratia ' || t\_oferta(j).cod\_config || ' facuta pentru clientul ');

FOR k IN t\_client.FIRST..t\_client.LAST LOOP

IF t\_oferta(j).cod\_client = t\_client(k).cod\_client THEN

DBMS\_OUTPUT.PUT(t\_client(k).nume || ' ' || t\_client(k).prenume);

END IF;

END LOOP;

DBMS\_OUTPUT.PUT\_LINE(' cu pretul de ' || t\_oferta(j).pret);

END IF;

END LOOP;

IF are\_oferte\_expirate = false THEN

DBMS\_OUTPUT.PUT\_LINE(' Nu are nicio oferta expirata');

END IF;

END LOOP;

FOR o IN t\_oferta.FIRST..t\_oferta.LAST LOOP

UPDATE OFERTA SET data\_inceput = sysdate WHERE id\_configuratie=t\_oferta(o).cod\_config and id\_dealer=t\_oferta(o).cod\_dealer and id\_client=t\_oferta(o).cod\_client;

UPDATE OFERTA SET data\_expirare = add\_months(sysdate, 24) WHERE id\_configuratie=t\_oferta(o).cod\_config and id\_dealer=t\_oferta(o).cod\_dealer and id\_client=t\_oferta(o).cod\_client;

END LOOP;

END actualiz\_oferte\_exp;

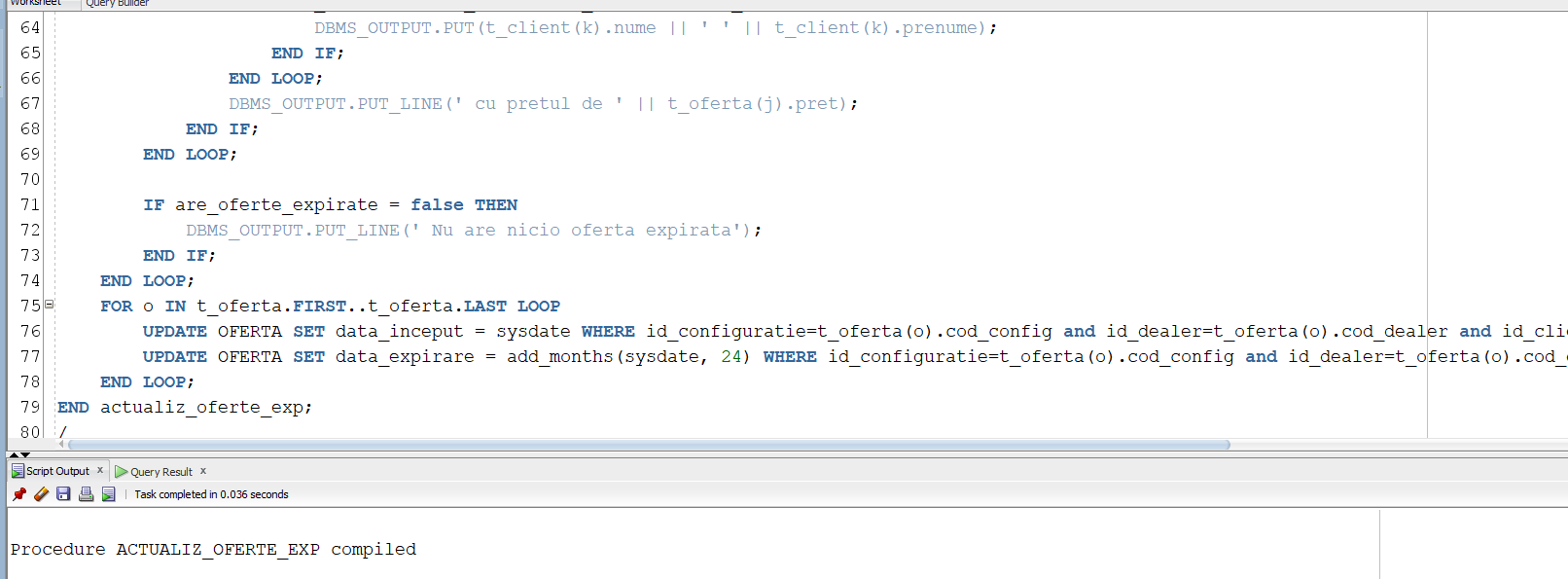
/

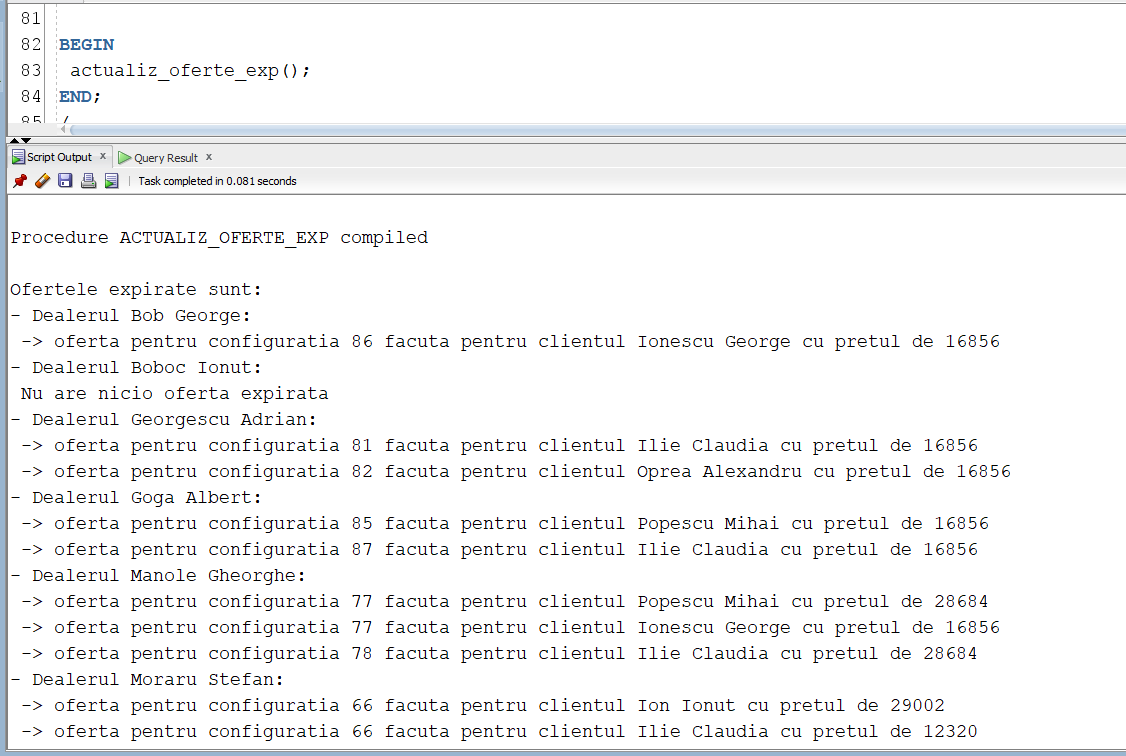
-- Cod pentru apelul procedurii

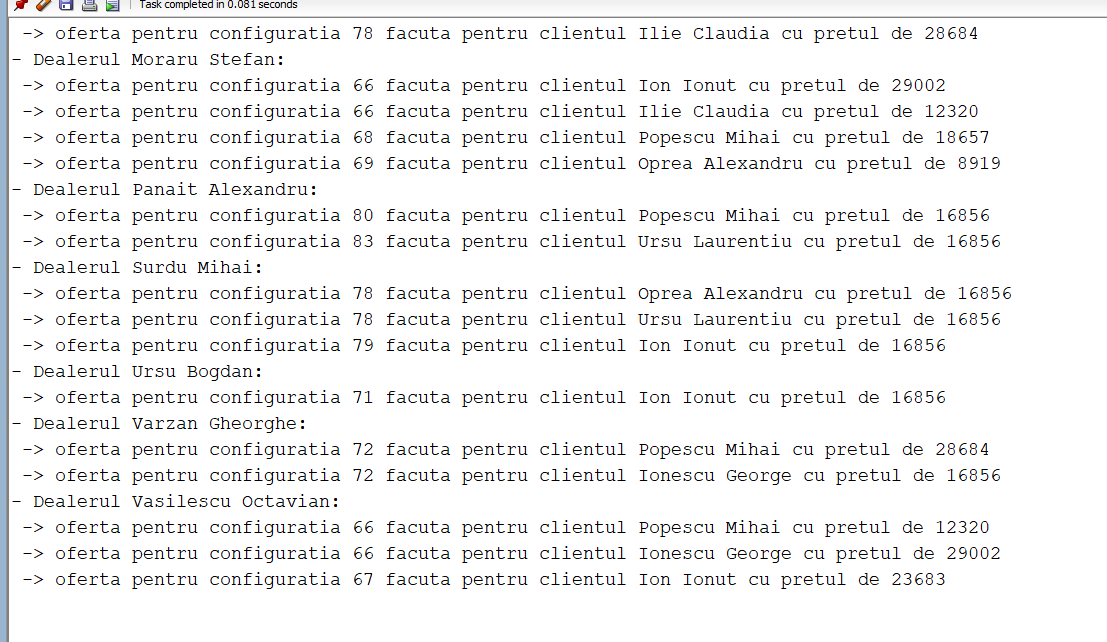
BEGIN

actualiz\_oferte\_exp();

END;

/





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

**Enunț problemă**: Definiți un subprogram prin care să se mărească cu 500 prețul motorizărilor, care au emisiile mai mari de 120 grame de CO2/km și care au capacitatea cilindrica mai mare decât media, pentru fiecare apariție în ofertele făcute de un dealer dat ca parametru.

Pentru rezolvarea cerinței a fost folosit un subprogram stocat independent de tip **procedură** și două cursoareunul **explicit, parametrizat de tip ciclu cursor** și altul **implicit cu subcereri**.

CREATE OR REPLACE PROCEDURE actualiz\_pret\_mot(d dealer.nume%TYPE) IS

-- cursor cu toate configuratiile care apar in ofertele

-- dealerului cu numele dat ca parametru

CURSOR c7 (d1 dealer.nume%TYPE) IS

SELECT c.id\_configuratie, c.id\_motorizare

FROM oferta o, configuratie c, dealer de

WHERE o.id\_configuratie = c.id\_configuratie and o.id\_dealer = de.id\_dealer

and lower(de.nume) = lower(d1);

v\_pret\_vechi NUMBER;

v\_pret\_nou NUMBER;

v\_macar\_o\_actualizare BOOLEAN := FALSE;

v\_contor NUMBER;

BEGIN

SELECT COUNT(\*) INTO v\_contor

FROM dealer de

WHERE lower(de.nume)=lower(d);

IF v\_contor = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista niciun dealer cu acest nume');

ELSE

FOR i in c7(d) LOOP

FOR j IN (SELECT \*

FROM motorizare

WHERE emisii > 120 and capacitate > (SELECT AVG(capacitate)

FROM motorizare)) LOOP

IF i.id\_motorizare = j.id\_motorizare THEN

SELECT pret INTO v\_pret\_vechi

FROM motorizare

WHERE id\_motorizare=i.id\_motorizare;

UPDATE motorizare

SET pret = pret + 500

WHERE id\_motorizare=i.id\_motorizare;

SELECT pret INTO v\_pret\_nou

FROM motorizare

WHERE id\_motorizare=i.id\_motorizare;

DBMS\_OUTPUT.PUT\_LINE('Pretul motorizarii cu id-ul ' || i.id\_motorizare || ' era ' || v\_pret\_vechi || ' si acum e ' || v\_pret\_nou);

v\_macar\_o\_actualizare := TRUE;

END IF;

END LOOP;

END LOOP;

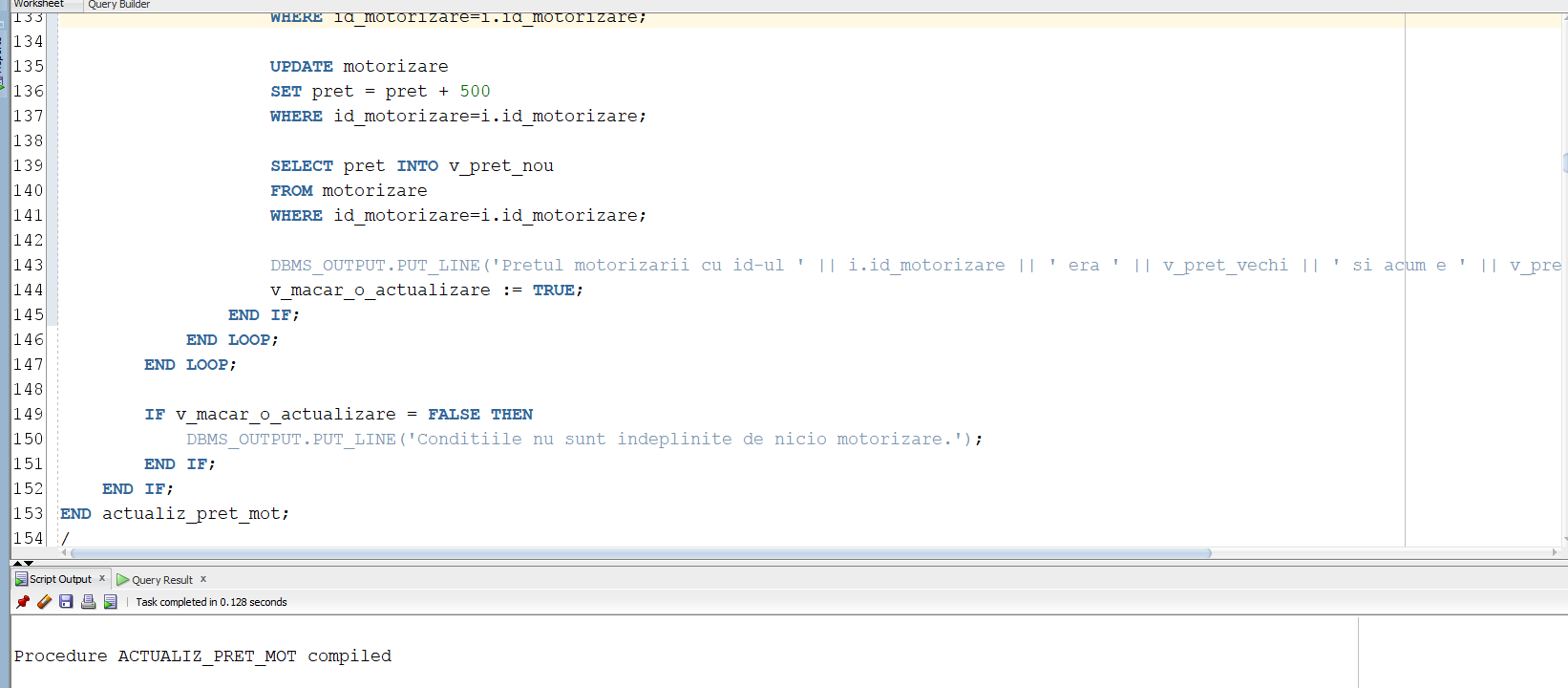
IF v\_macar\_o\_actualizare = FALSE THEN

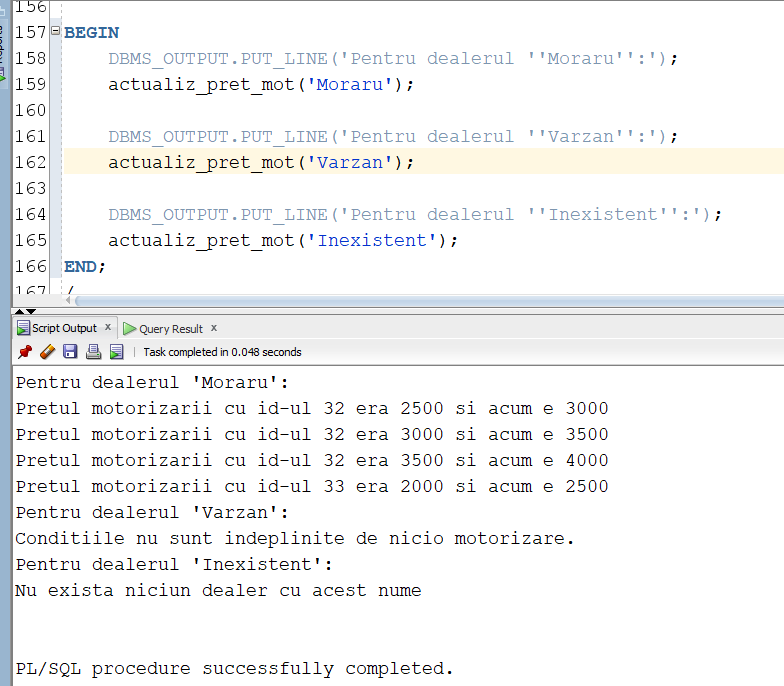
DBMS\_OUTPUT.PUT\_LINE('Conditiile nu sunt indeplinite de nicio motorizare.');

END IF;

END IF;

END actualiz\_pret\_mot;

/



## Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip funcție care să utilizeze într-o singura comandă SQL 3 dintre tabelele definite. Definiți minim 2 excepții. Apelați subprogramul astfel încât sa evidențiați toate cazurile tratate.

**Enunț problemă**: Definiți un subprogram de tip funcție care primește ca parametru numele unei reprezentante auto și returnează numele modelului de mașină care apare cel mai des in configurațiile încercate de clienți.

Pentru rezolvarea cerinței au fost folosite următoarele tabele într-o singură comandă SQL: REPREZ\_AUTO, MODELE\_DISP, MODEL\_MASINA, CONFIGURATIE, iar excepțiile definite sunt FARA\_REPREZ, FARA\_CONFIG, PREA\_MULTE\_MODELE.

CREATE OR REPLACE FUNCTION cel\_mai\_popular\_model\_din\_reprez(v\_nume\_reprez\_auto REPREZ\_AUTO.nume%TYPE)

RETURN VARCHAR2 IS

v\_id\_reprez REPREZ\_AUTO.id\_reprez%TYPE;

TYPE vect IS VARRAY(1000) OF MODEL\_MASINA.nume%TYPE;

modele vect := vect();

v\_contor NUMBER;

FARA\_REPREZ EXCEPTION;

FARA\_CONFIG EXCEPTION;

PREA\_MULTE\_MODELE EXCEPTION;

BEGIN

SELECT COUNT(\*) INTO v\_contor

FROM REPREZ\_AUTO

WHERE lower(nume) = lower(v\_nume\_reprez\_auto);

IF v\_contor = 0 THEN

RAISE FARA\_REPREZ;

END IF; -- nu avem cum sa avem contorul > 1 pt ca numele din tabelul reprez\_auto este unique

SELECT id\_reprez INTO v\_id\_reprez

FROM REPREZ\_AUTO

WHERE lower(nume) = lower(v\_nume\_reprez\_auto);

-- datorita verificarii de mai sus stim ca exista o reprezentanta si numai una cu acest nume deci nu se vor genera erori

-- luam toate modelele care au numarul de aparitii din configuratiile reprezentantei maxim

SELECT mm.nume BULK COLLECT INTO modele

FROM REPREZ\_AUTO r JOIN MODELE\_DISP md ON r.id\_reprez=md.id\_reprez

JOIN MODEL\_MASINA mm ON md.id\_model=mm.id\_model

JOIN CONFIGURATIE c ON mm.id\_model=c.id\_model

WHERE r.id\_reprez = v\_id\_reprez

GROUP BY c.id\_model, mm.nume

HAVING COUNT(\*) = (

SELECT MAX(COUNT(cc.id\_model))

FROM CONFIGURATIE cc

WHERE cc.id\_reprez = v\_id\_reprez

GROUP BY cc.id\_model

);

IF modele.count = 0 THEN

RAISE FARA\_CONFIG;

ELSIF modele.count > 1 THEN

RAISE PREA\_MULTE\_MODELE;

END IF;

RETURN 'Modelul care apare cel mai des in configuratii in reprezentanta ' || v\_nume\_reprez\_auto || ' este ' || modele(1) ||'.';

EXCEPTION

WHEN FARA\_REPREZ THEN

RETURN 'Nu exista nicio reprezentanta auto cu acest nume.';

WHEN FARA\_CONFIG THEN

RETURN 'Nu exista configuratii incercate pentru acesta reprezentanta.';

WHEN PREA\_MULTE\_MODELE THEN

RETURN 'Exista mai multe modele cu numarul de aparitii maxim in configuratii pentru acesta reprezentanta.';

END cel\_mai\_popular\_model\_din\_reprez;

/

-- Apel subprogram

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Reprezentanta Porsche Bucuresti Vest:');

DBMS\_OUTPUT.PUT\_LINE('-> ' || cel\_mai\_popular\_model\_din\_reprez('Porsche Bucuresti Vest'));

-- stergem temporar configuratiile din reprezentanta AutoItalia

-- pentru a testa generarea de eroare corespunzatoare

DELETE

FROM CONFIGURATIE c

WHERE c.id\_reprez = (SELECT id\_reprez

FROM REPREZ\_AUTO r

WHERE lower(r.nume)=lower('AutoItalia'));

DBMS\_OUTPUT.PUT\_LINE('Reprezentanta AutoItalia:');

DBMS\_OUTPUT.PUT\_LINE('-> ' || cel\_mai\_popular\_model\_din\_reprez('AutoItalia'));

ROLLBACK;

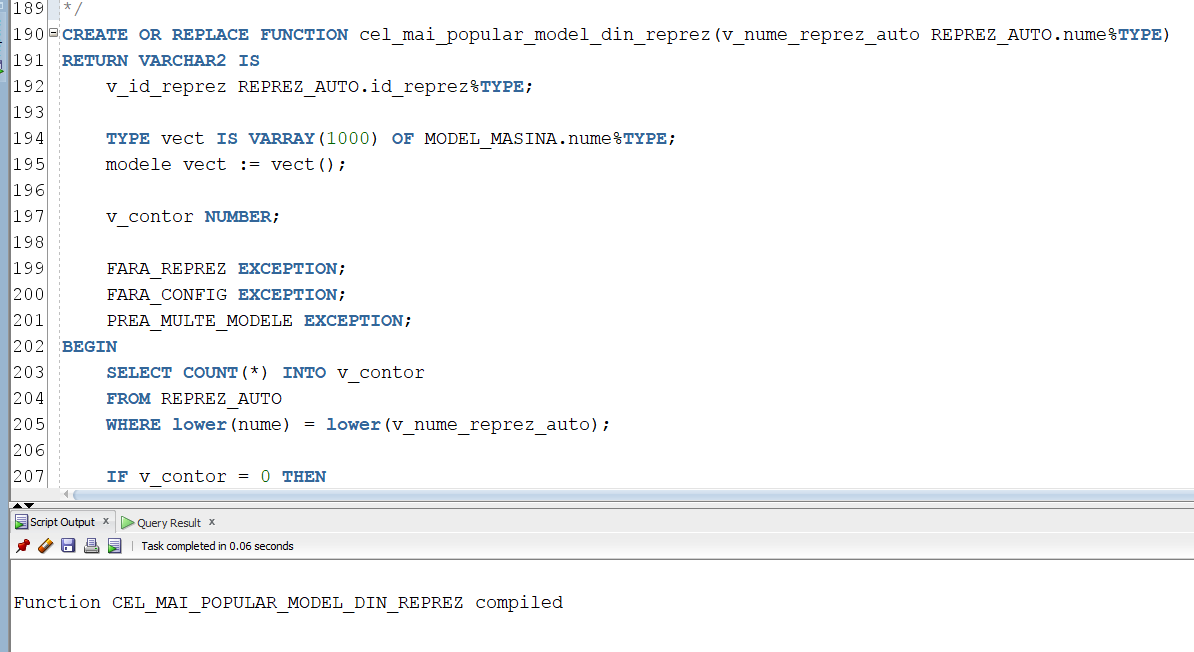
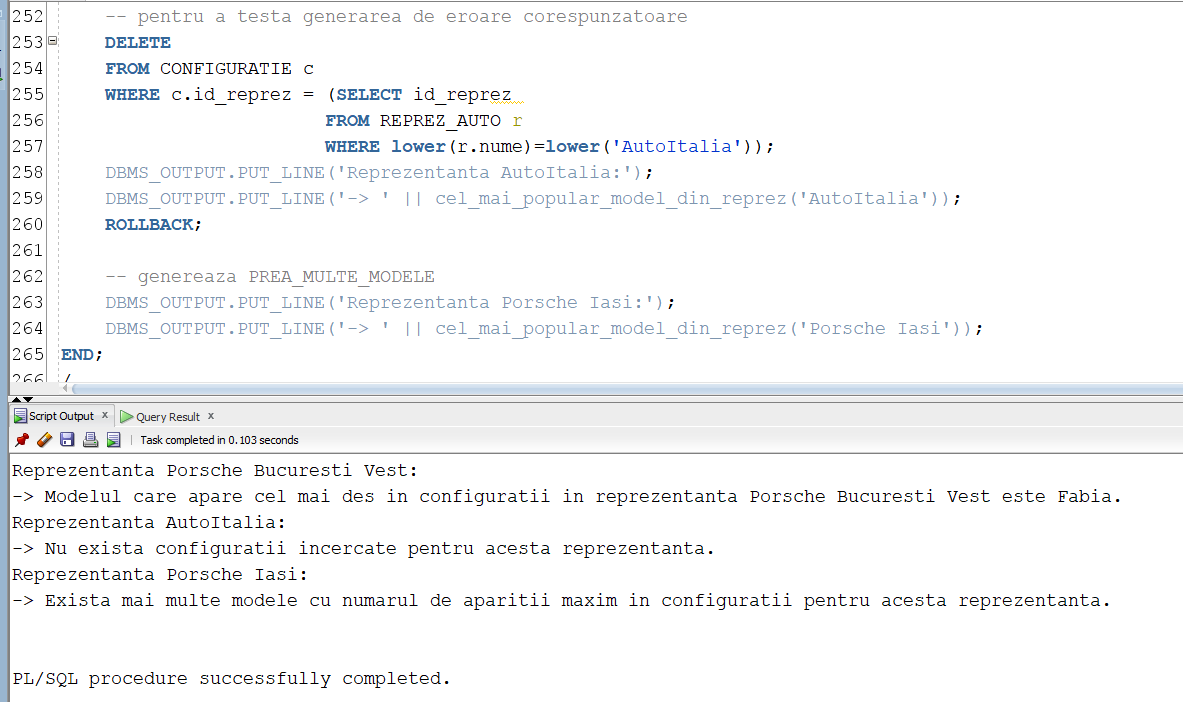
-- genereaza PREA\_MULTE\_MODELE

DBMS\_OUTPUT.PUT\_LINE('Reprezentanta Porsche Iasi:');

DBMS\_OUTPUT.PUT\_LINE('-> ' || cel\_mai\_popular\_model\_din\_reprez('Porsche Iasi'));

END;

/



## 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 comanda SQL 5 dintre tabelele definite. Tratați toate excepțiile care pot apărea, incluzând excepțiile NO\_DATA\_FOUND și TOO\_MANY\_ROWS. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

**Enunț problemă**: Definiți un subprogram de tip procedură care mărește cu 5% prețul unei oferte

care este făcută pentru configurația care conține un anumit model de mașina dat ca parametru și care

este făcută de un dealer dat ca parametru unui client care are vârsta mai mica de 45 de ani.

Sa se trateze prin afișarea unui mesaj de eroare cazul în care nu exista nicio astfel de ofertă sau există mai multe.

CREATE OR REPLACE PROCEDURE mareste\_pret\_oferta(v\_nume\_model MODEL\_MASINA.nume%TYPE, v\_nume\_dealer DEALER.nume%TYPE) IS

v\_id\_model MODEL\_MASINA.id\_model%TYPE;

v\_contor NUMBER;

v\_nr\_select NUMBER := 0;

v\_id\_dealer DEALER.id\_dealer%TYPE;

v\_oferta\_id\_conf OFERTA.id\_configuratie%TYPE;

v\_oferta\_id\_dealer OFERTA.id\_dealer%TYPE;

v\_oferta\_id\_client OFERTA.id\_client%TYPE;

BEGIN

SELECT id\_model INTO v\_id\_model

FROM MODEL\_MASINA

WHERE lower(nume) = lower(v\_nume\_model);

v\_nr\_select := 1;

SELECT id\_dealer INTO v\_id\_dealer

FROM DEALER

WHERE lower(nume) = lower(v\_nume\_dealer);

v\_nr\_select := 2;

SELECT o.id\_configuratie, o.id\_dealer, o.id\_client INTO v\_oferta\_id\_conf, v\_oferta\_id\_dealer, v\_oferta\_id\_client

FROM MODEL\_MASINA mm JOIN CONFIGURATIE c ON mm.id\_model=c.id\_model

JOIN OFERTA o ON c.id\_configuratie=o.id\_configuratie

JOIN DEALER d ON d.id\_dealer=o.id\_dealer

JOIN CLIENT cl ON cl.id\_client=o.id\_client

WHERE mm.id\_model= v\_id\_model and d.id\_dealer=v\_id\_dealer and trunc(months\_between(sysdate, cl.data\_nastere) / 12) < 45;

UPDATE OFERTA o

SET pret\_oferta = pret\_oferta + pret\_oferta \* 0.05

WHERE o.id\_configuratie=v\_oferta\_id\_conf and o.id\_dealer=v\_oferta\_id\_dealer and o.id\_client=v\_oferta\_id\_client;

DBMS\_OUTPUT.PUT\_LINE('Update pret oferta realizat cu succes');

EXCEPTION

WHEN TOO\_MANY\_ROWS THEN

DBMS\_OUTPUT.PUT\_LINE('Exista mai multe oferte cu modelul dat, dealerul dat si facute pentru un client mai tanar de 45 de ani.');

WHEN NO\_DATA\_FOUND THEN

IF v\_nr\_select = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista niciun model de masina cu numele dat.');

ELSIF v\_nr\_select = 1 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista niciun dealer cu numele dat.');

ELSE

DBMS\_OUTPUT.PUT\_LINE('Nu exista nicio oferta cu modelul dat, dealerul dat si facuta pentru un client mai tanar de 45 de ani.');

END IF;

END mareste\_pret\_oferta;

/

-- Apel subprogram

BEGIN

-- Exista o singura oferta care indeplineste conditiile deci nu se va generea nicio eroare, iar pretul va fi acualizat

DBMS\_OUTPUT.PUT\_LINE('Modelul ''Octavia'', dealerul ''Boboc'': ');

DBMS\_OUTPUT.PUT('-> ');

mareste\_pret\_oferta('Octavia', 'Boboc');

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

-- Nu exista acest model de masina => se va genera NO\_DATA\_FOUND cu mesajul corespunzator

DBMS\_OUTPUT.PUT\_LINE('Modelul ''ModelInexistent'', dealerul ''Vasilescu'': ');

DBMS\_OUTPUT.PUT('-> ');

mareste\_pret\_oferta('ModelInexistent', 'Vasilescu');

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

-- Nu exista acest dealer => se va genera NO\_DATA\_FOUND cu mesajul corespunzator

DBMS\_OUTPUT.PUT\_LINE('Modelul ''Octavia'', dealerul ''DealerInexistent'': ');

DBMS\_OUTPUT.PUT('-> ');

mareste\_pret\_oferta('Octavia', 'DealerInexistent');

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

-- Exista oferte facute de acest dealer pentru acest model dar clientii carora le-a fost facuta oferta au mai mult de 45 de ani

-- => se va genera NO\_DATA\_FOUND si se va afisa mesajul corespunzator

DBMS\_OUTPUT.PUT\_LINE('Modelul ''Octavia'', dealerul ''Moraru'': ');

DBMS\_OUTPUT.PUT('-> ');

mareste\_pret\_oferta('Octavia', 'Moraru');

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

-- Exista acest model si acest dealer dar nu exista o oferta facuta de dealer pentru acest model

-- => se va genereza NO\_DATA\_FOUND si se va afisa mesajul corespunzator

DBMS\_OUTPUT.PUT\_LINE('Modelul ''Octavia'', dealerul ''Bob'': ');

DBMS\_OUTPUT.PUT('-> ');

mareste\_pret\_oferta('Octavia', 'Bob');

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

-- Exista doua oferte care indeplinesc conditiile din cerinta

-- => se va genera eroarea TOO\_MANY\_ROWS si se va afisa mesajul corespunzator

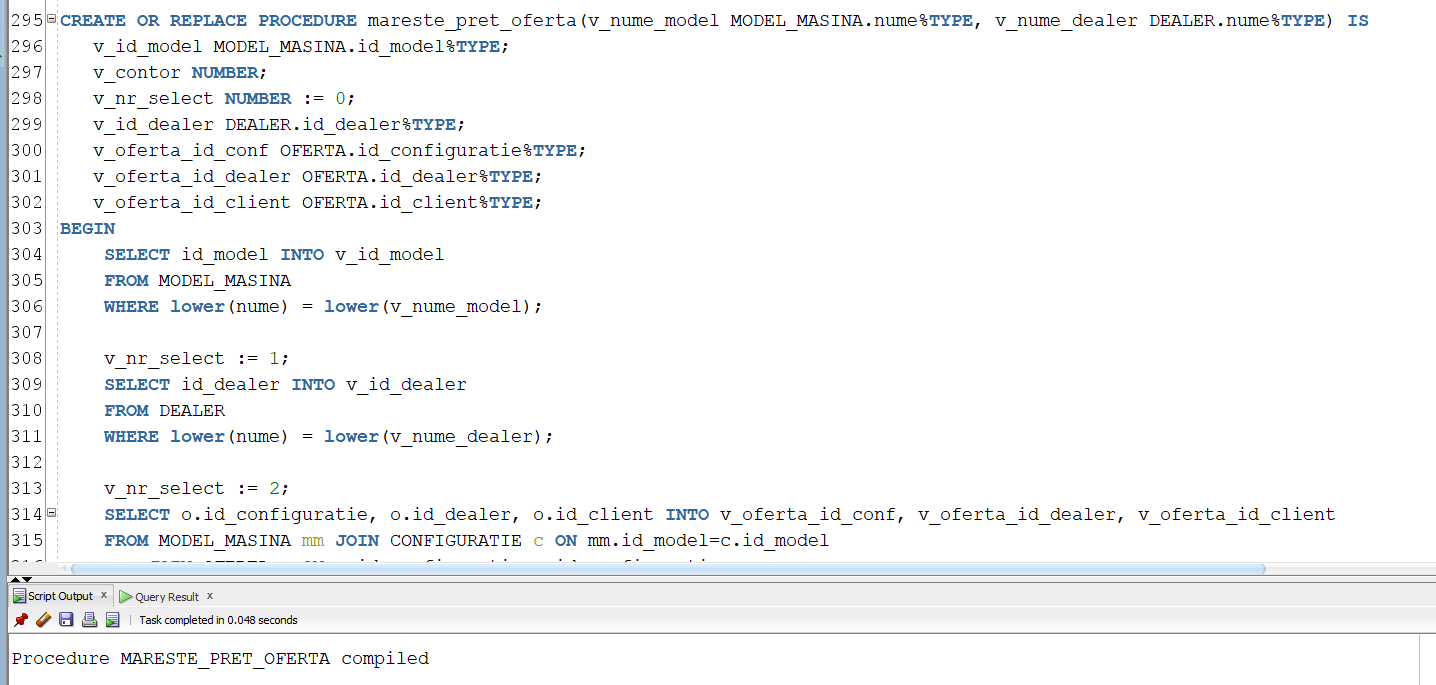
DBMS\_OUTPUT.PUT\_LINE('Modelul ''Octavia'', dealerul ''Vasilescu'': ');

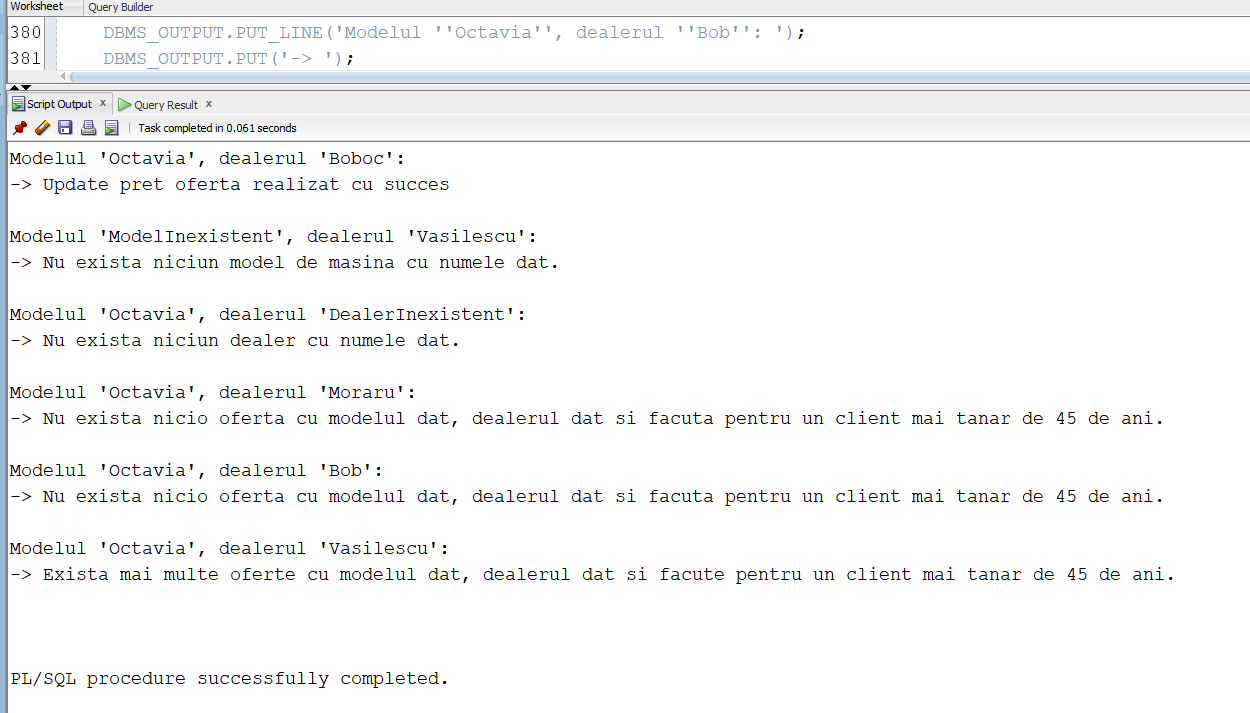
DBMS\_OUTPUT.PUT('-> ');

mareste\_pret\_oferta('Octavia', 'Vasilescu');

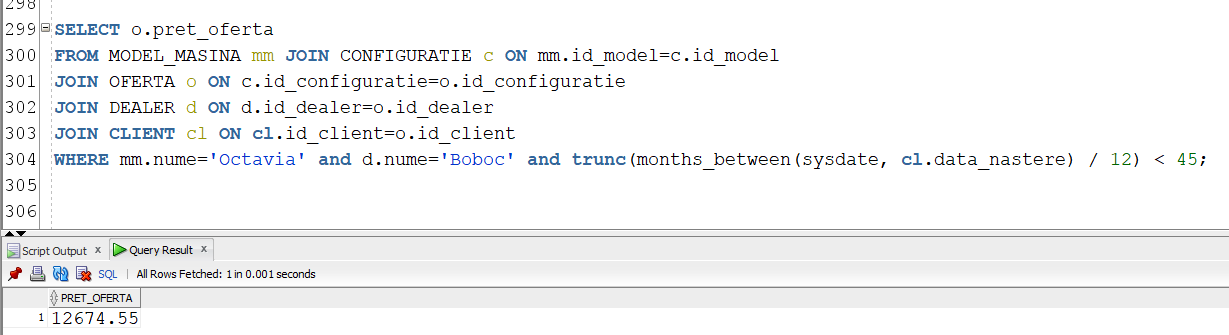
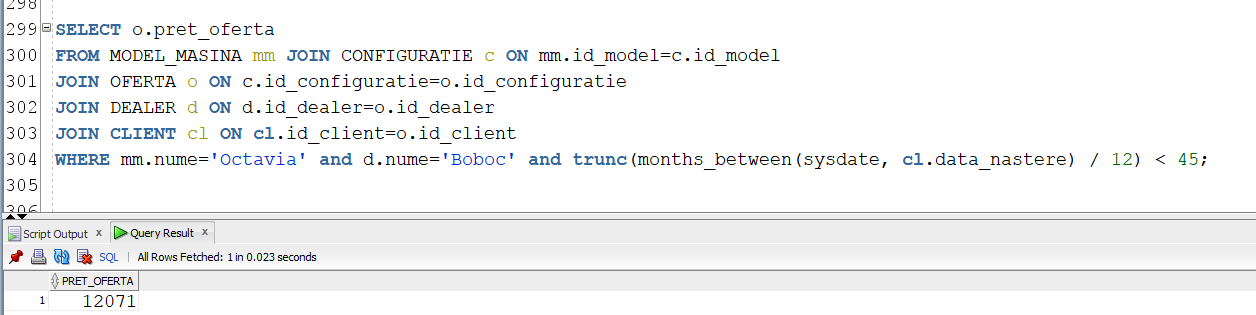
DBMS\_OUTPUT.PUT\_LINE('');

END;

/



Pentru a verifica ca s-a produs update-ul prețului pentru dealerul Boboc am selectat prețul ofertei înainte și după apelul procedurii:



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

**Enunț problemă**: Definiți un trigger care să nu permită ștergerea unei motorizări dacă în urma ștergerii

nu ar mai rămâne nicio motorizare. De asemenea, intervalul in care se pot face ștergerile este 8:00-20:00 de luni pana vineri.

CREATE OR REPLACE TRIGGER stergere\_motoriz

BEFORE DELETE ON motorizare

DECLARE

v\_nr\_motoriz NUMBER;

BEGIN

IF (TO\_CHAR(sysdate, 'D') = 1 OR TO\_CHAR(sysdate, 'D') = 2 OR TO\_CHAR(sysdate, 'HH24') NOT BETWEEN 8 AND 20) THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Nu se poate sterge o motorizare inafara programului!');

ELSE

SELECT count(\*) INTO v\_nr\_motoriz

FROM motorizare m;

IF v\_nr\_motoriz = 1 THEN

RAISE\_APPLICATION\_ERROR(-20002, 'Nu se poate sterge ultima motorizare disponibila!');

END IF;

END IF;

END;

/

-- Declansare trigger

BEGIN

-- Stergem toate motorizarile mai putin una

DELETE

FROM motorizare

WHERE id\_motorizare!=32;

DBMS\_OUTPUT.PUT\_LINE('A mai ramas o singura motorizare');

-- Incercam stergerea ultimei => genereaza eroare

DELETE

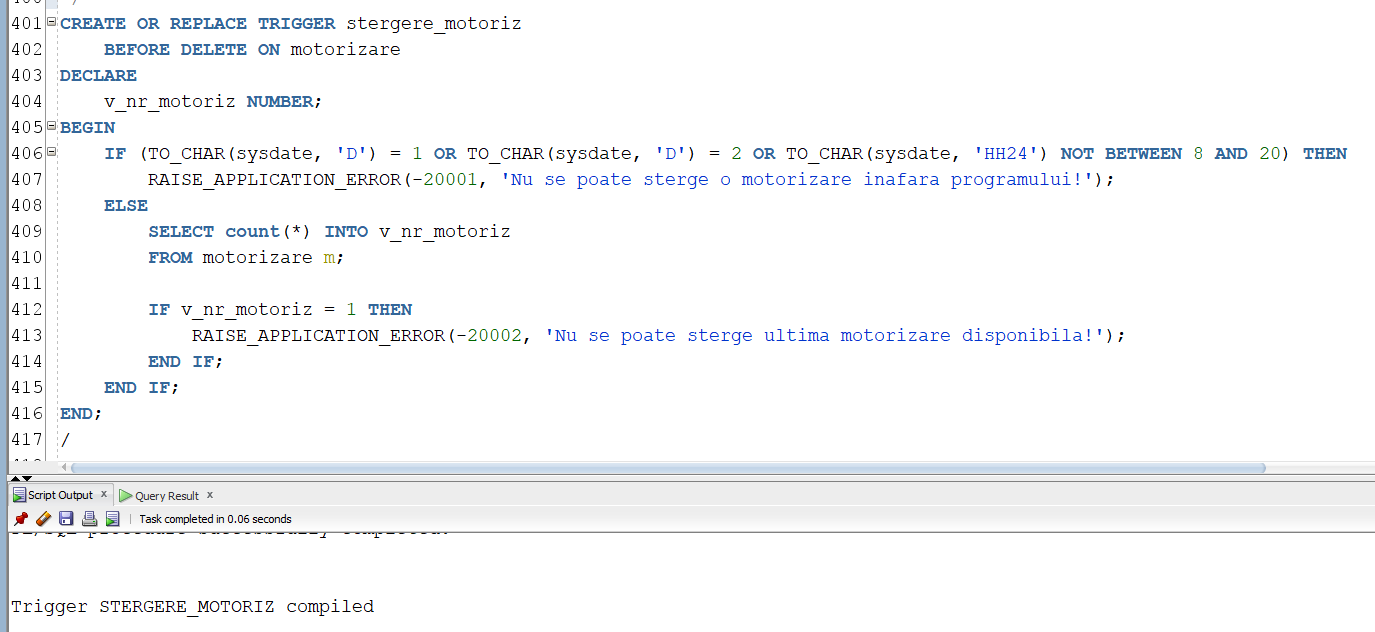
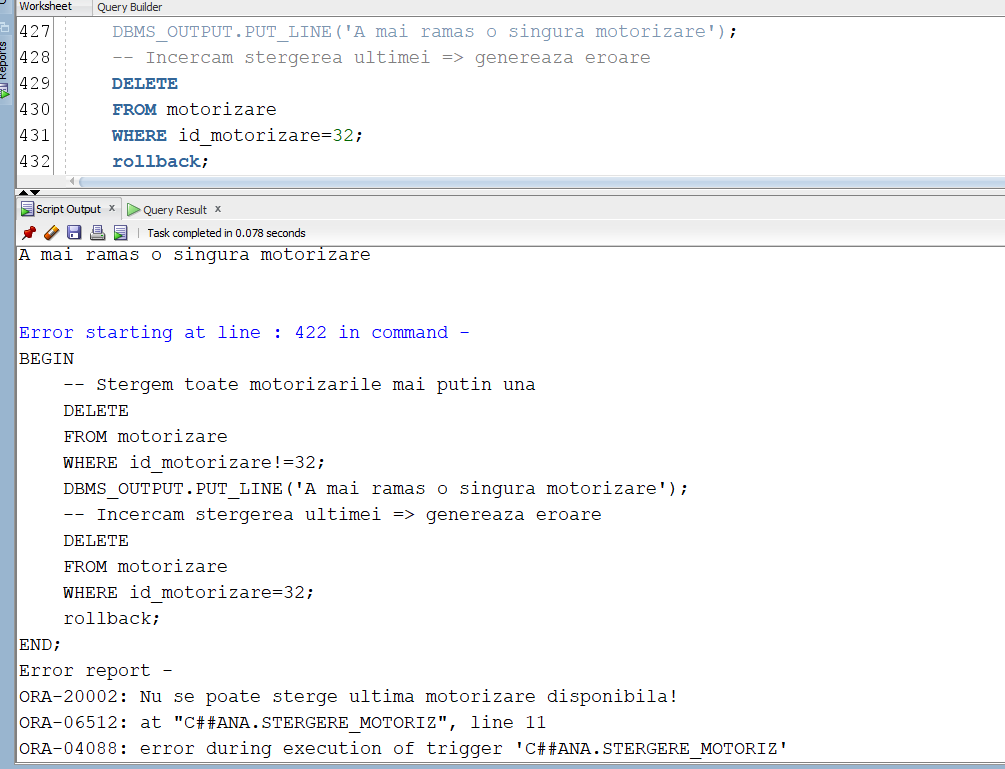
FROM motorizare

WHERE id\_motorizare=32;

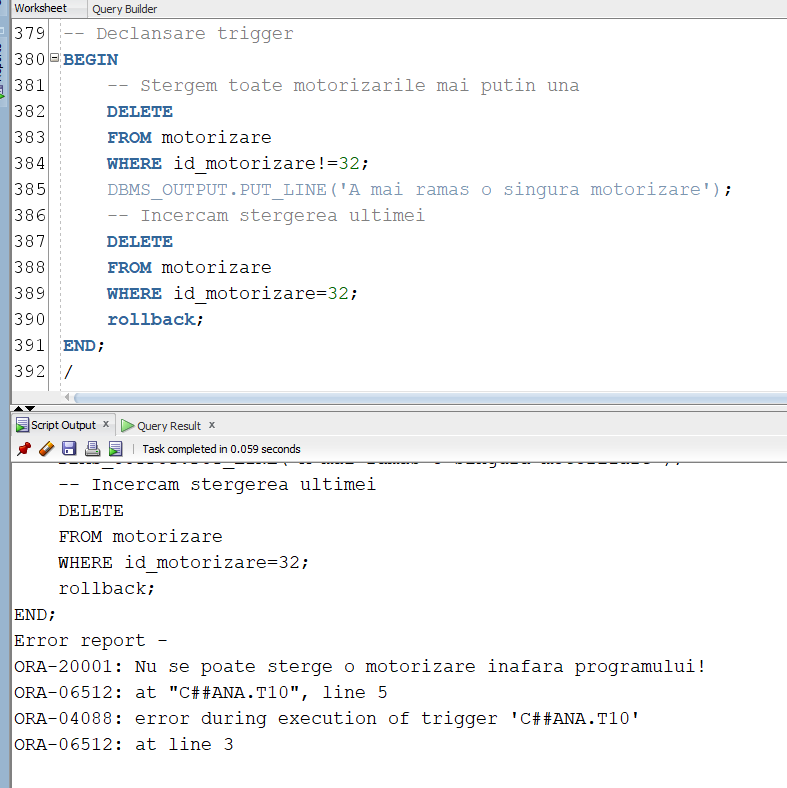
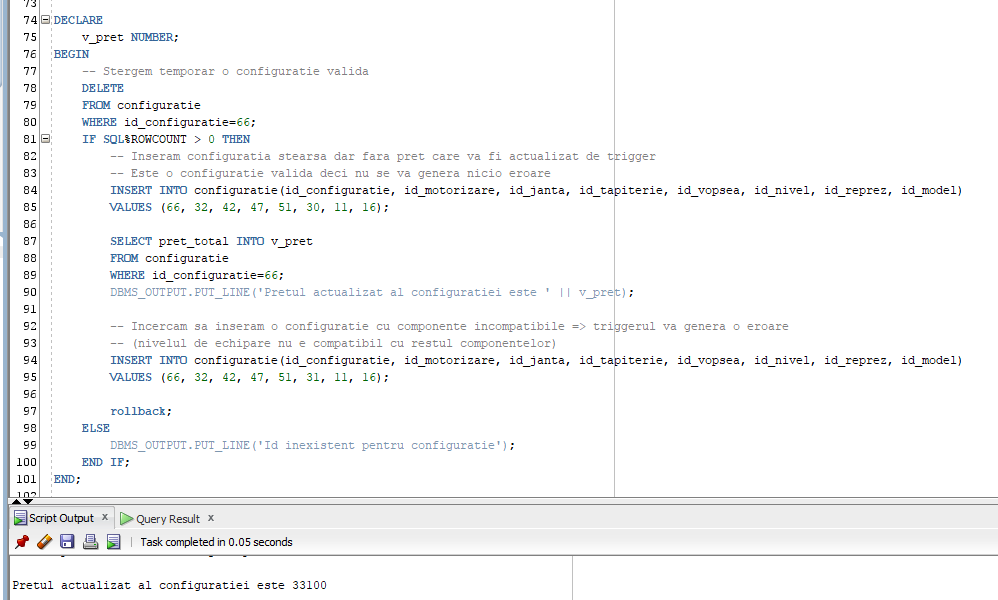
rollback;

END;

/

 Dacă se rulează blocul care declanșează triggerul în timpul programului se vor putea șterge motorizările mai puțin ultima la care se generează o eroare:

Dacă se rulează blocul care declanșează triggerul înafara programului se genera eroarea corespunzătoare:



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

**Enunț problemă**: Definiți un trigger care la inserarea in tabelul configuratie să

verifice daca datele sunt valide (modelul selectat sa fie compatibil cu nivelul de echipare selectat dar si componentele să facă parte din nivelul de echipare selectat) și setează prețul ca suma prețurilor

modelului, nivelului de echipare, motorizării, jantei, tapițeriei si vopselei folosite.

CREATE OR REPLACE TRIGGER validare\_config

BEFORE INSERT OR UPDATE ON configuratie

FOR EACH ROW

DECLARE

v\_pret\_model NUMBER;

v\_pret\_niv\_ech NUMBER;

v\_pret\_mot NUMBER;

v\_pret\_jante NUMBER;

v\_pret\_tapit NUMBER;

v\_pret\_vopsea NUMBER;

v\_contor NUMBER;

BEGIN

-- Verificam compatibilitatea id-urilor date

SELECT count(\*) INTO v\_contor

FROM reprez\_auto r, modele\_disp md, model\_masina mm, niv\_ech\_disp ned, nivel\_echipare ne,

configuratie c, motorizare\_niv mn, motorizare mot, janta\_niv jn, janta j, tapit\_niv tn,

tapiterie t, vopsea\_niv vn, vopsea v

WHERE r.id\_reprez=md.id\_reprez and md.id\_model=mm.id\_model and mm.id\_model=ned.id\_model and ned.id\_nivel=ne.id\_nivel

and ne.id\_nivel=mn.id\_nivel and ne.id\_nivel=jn.id\_nivel and ne.id\_nivel=tn.id\_nivel

and ne.id\_nivel=vn.id\_nivel and mn.id\_motorizare=mot.id\_motorizare and jn.id\_janta=j.id\_janta

and tn.id\_tapiterie=t.id\_tapiterie and vn.id\_vopsea=v.id\_vopsea and r.id\_reprez=:NEW.id\_reprez

and mm.id\_model=:NEW.id\_model and ne.id\_nivel=:NEW.id\_nivel and v.id\_vopsea=:NEW.id\_vopsea

and t.id\_tapiterie=:NEW.id\_tapiterie and j.id\_janta=:NEW.id\_janta and mot.id\_motorizare=:NEW.id\_motorizare;

IF v\_contor > 0 THEN

-- Daca am ajuns pana aici inseamna ca datele sunt valide si putem sa ne

-- calculam pretul ca suma de componente din configuratie

SELECT pret\_pornire INTO v\_pret\_model

FROM MODEL\_MASINA mm

WHERE mm.id\_model=:NEW.id\_model;

SELECT pret INTO v\_pret\_niv\_ech

FROM NIVEL\_ECHIPARE ne

WHERE ne.id\_nivel=:NEW.id\_nivel;

SELECT pret INTO v\_pret\_mot

FROM MOTORIZARE mot

WHERE mot.id\_motorizare=:NEW.id\_motorizare;

SELECT pret INTO v\_pret\_jante

FROM JANTA j

WHERE j.id\_janta=:NEW.id\_janta;

SELECT pret INTO v\_pret\_tapit

FROM TAPITERIE t

WHERE t.id\_tapiterie=:NEW.id\_tapiterie;

SELECT pret INTO v\_pret\_vopsea

FROM VOPSEA v

WHERE v.id\_vopsea=:NEW.id\_vopsea;

:NEW.pret\_total := v\_pret\_model + v\_pret\_niv\_ech + v\_pret\_mot + v\_pret\_jante + v\_pret\_tapit + v\_pret\_vopsea;

ELSE

RAISE\_APPLICATION\_ERROR ( -20001, 'Incompatibilitate intre componentele configuratiei!');

END IF;

END;

/

-- Declansare trigger

DECLARE

v\_pret NUMBER;

BEGIN

-- Stergem temporar o configuratie valida

DELETE

FROM configuratie

WHERE id\_configuratie=66;

IF SQL%ROWCOUNT > 0 THEN

-- Inseram configuratia stearsa dar fara pret care va fi actualizat de trigger

-- Este o configuratie valida deci nu se va genera nicio eroare

INSERT INTO configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

VALUES (66, 32, 42, 47, 51, 30, 11, 16);

SELECT pret\_total INTO v\_pret

FROM configuratie

WHERE id\_configuratie=66;

DBMS\_OUTPUT.PUT\_LINE('Pretul actualizat al configuratiei este ' || v\_pret);

-- Incercam sa inseram o configuratie cu componente incompatibile => triggerul va genera o eroare

-- (nivelul de echipare nu e compatibil cu restul componentelor)

INSERT INTO configuratie(id\_configuratie, id\_motorizare, id\_janta, id\_tapiterie, id\_vopsea, id\_nivel, id\_reprez, id\_model)

VALUES (66, 32, 42, 47, 51, 31, 11, 16);

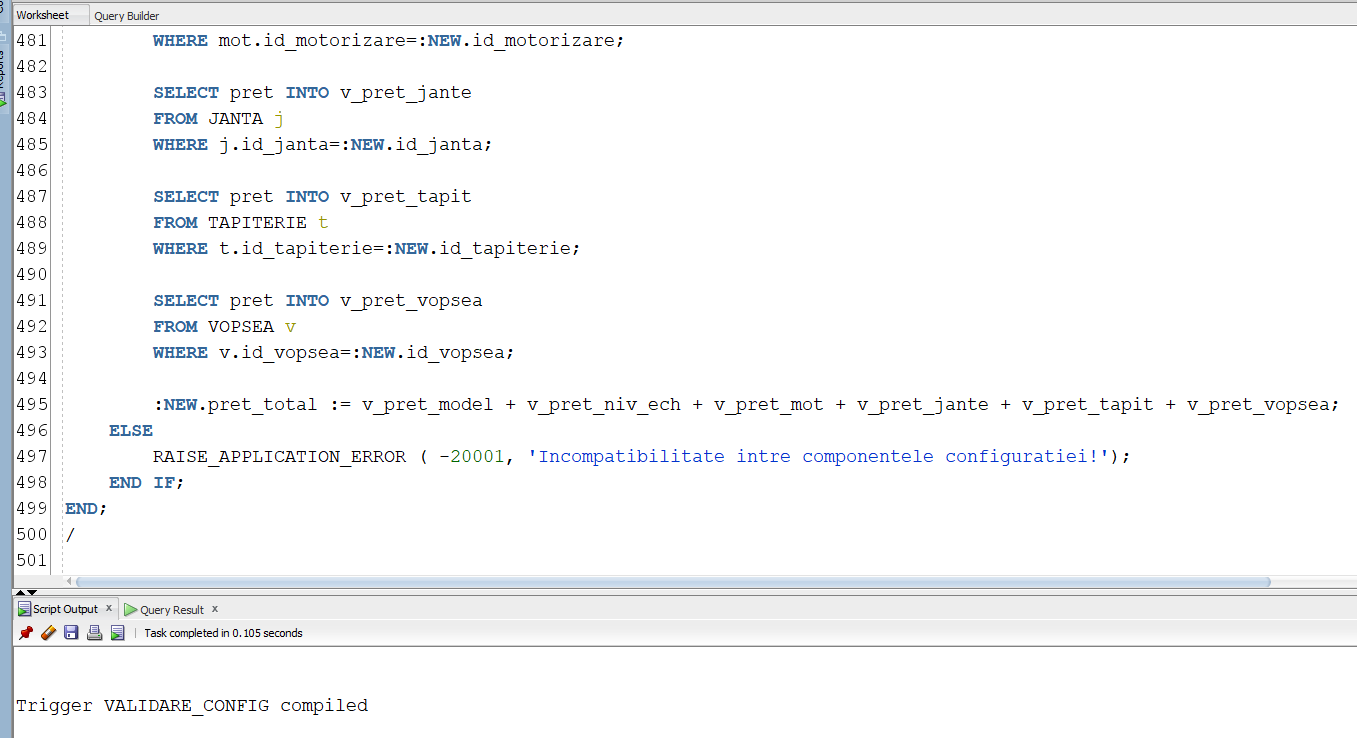
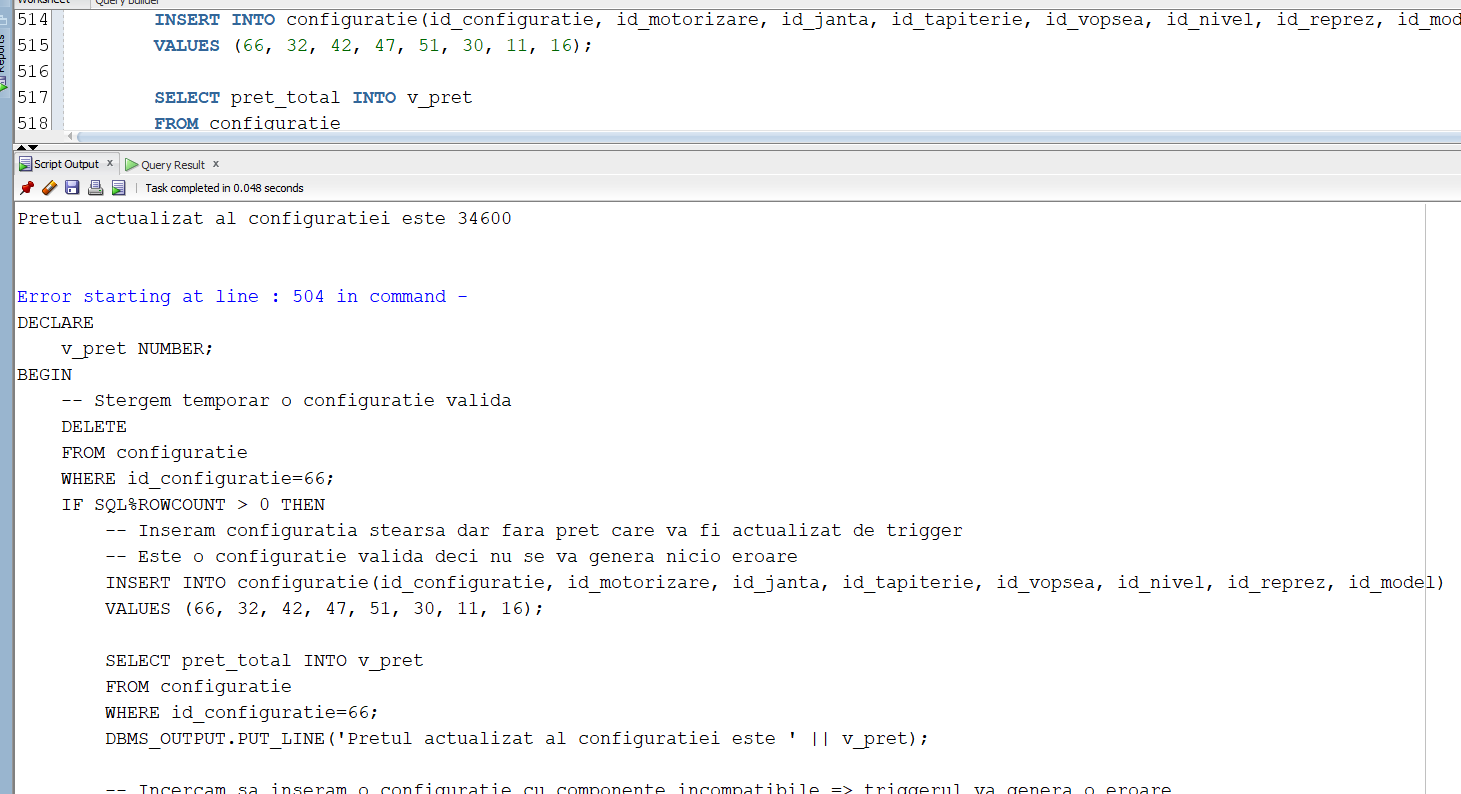
rollback;

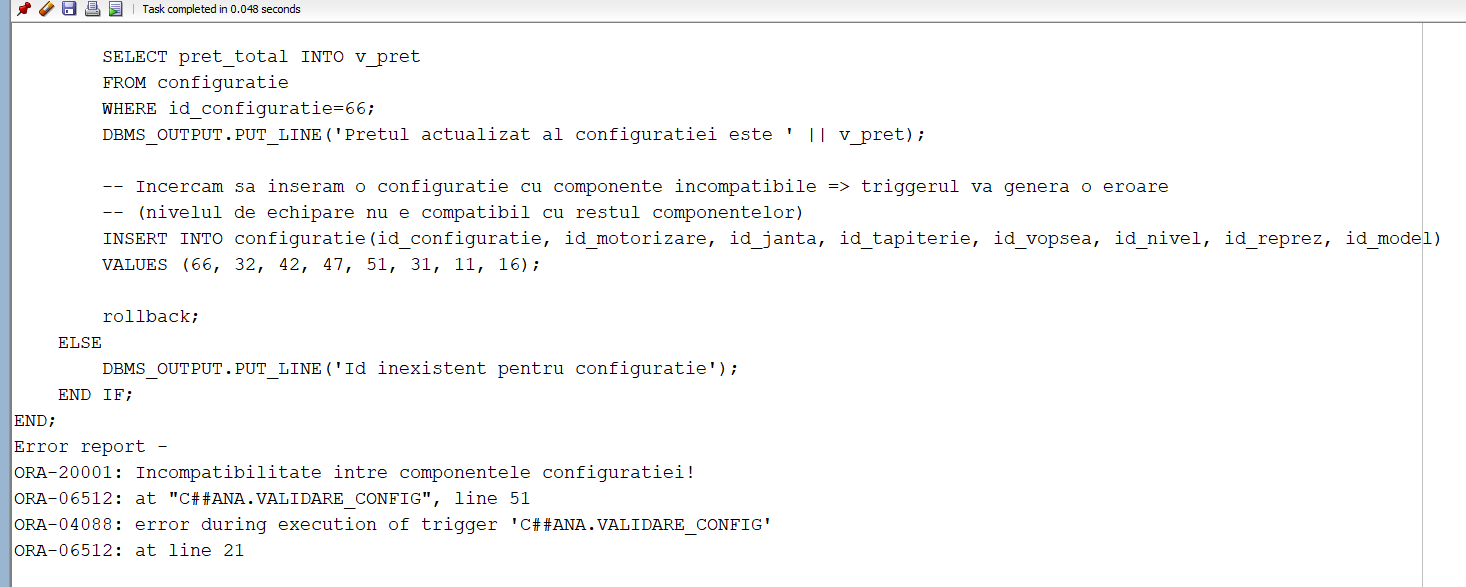
ELSE

DBMS\_OUTPUT.PUT\_LINE('Id inexistent pentru configuratie');

END IF;

END;

/



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

**Enunț problemă**: Definiți un trigger care va pune in tabelul istoric\_comenzi (care urmeaza sa fie definit)

instrucțiunile care au avut loc fără erori. In cazul comenzilor care dau eroare, se va scrie într-un fișier text ultima eroare apărută.

CREATE TABLE istoric\_comenzi(

utilizator VARCHAR2(30),

comanda VARCHAR(100),

nume\_bd VARCHAR(20),

data date

);

CREATE OR REPLACE DIRECTORY CTEST AS 'C:\test';

GRANT READ ON DIRECTORY CTEST TO PUBLIC;

CREATE OR REPLACE TRIGGER t12

AFTER CREATE OR DROP OR ALTER OR SERVERERROR ON DATABASE

DECLARE

out\_File UTL\_FILE.FILE\_TYPE;

v\_count NUMBER;

BEGIN

IF DBMS\_UTILITY.FORMAT\_ERROR\_STACK IS NULL THEN

INSERT INTO istoric\_comenzi VALUES

(SYS.LOGIN\_USER, SYS.SYSEVENT, SYS.DICTIONARY\_OBJ\_NAME, SYSDATE);

ELSE

out\_File := UTL\_FILE.FOPEN('CTEST', 'trigger\_test.txt' , 'W');

UTL\_FILE.PUT\_LINE(out\_file , DBMS\_UTILITY.FORMAT\_ERROR\_STACK);

UTL\_FILE.FCLOSE(out\_file);

END IF;

END;

/

-- Instructiune care nu da eroare

CREATE TABLE test(id NUMBER);

-- Instructiune care da eroare

CREATE TABLE test\_tbl (

id NUMBER PRIMARY KEY,

name VARCHAR2

);

-- Instructiune care nu da eroare

DROP TABLE test;

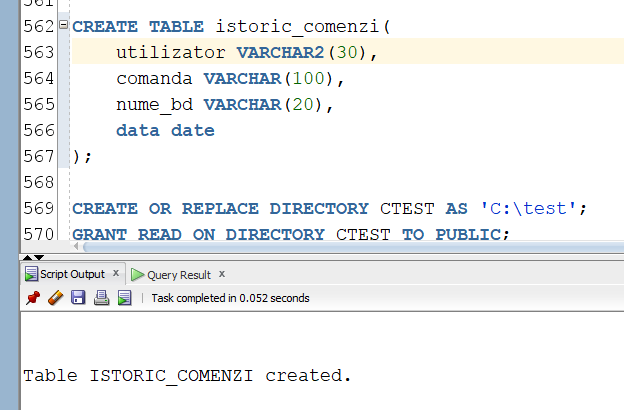
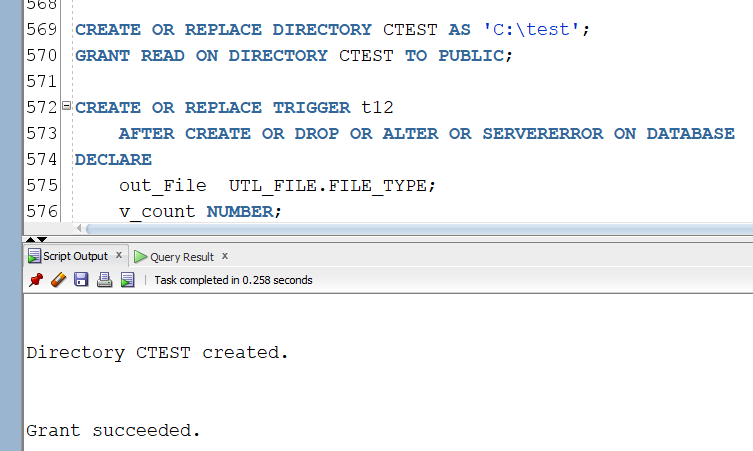
-- Instructiune care da eroare

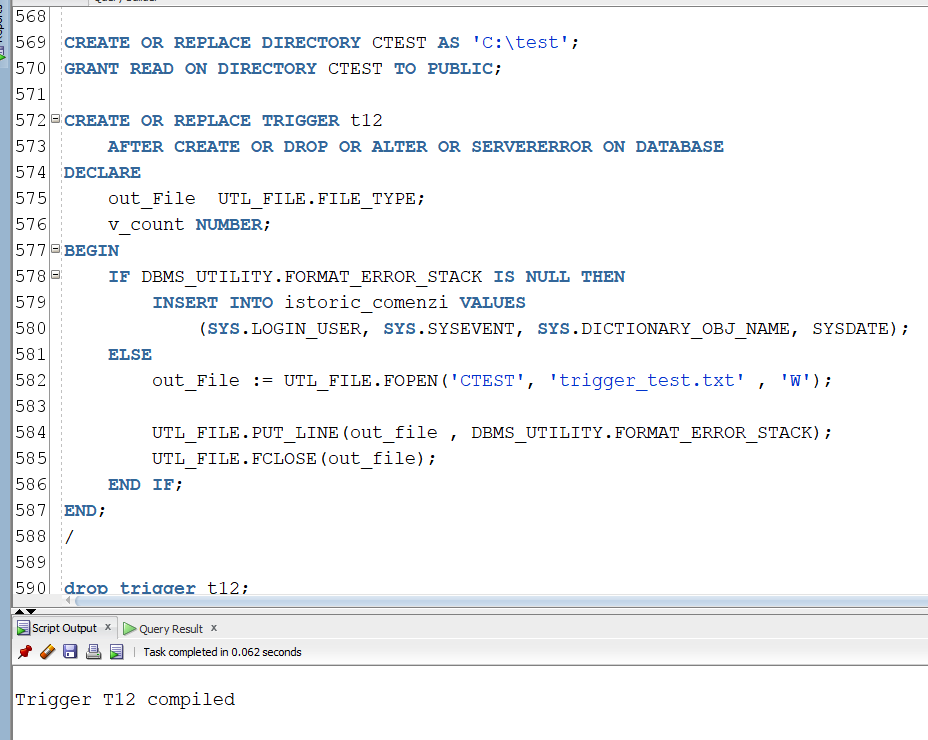
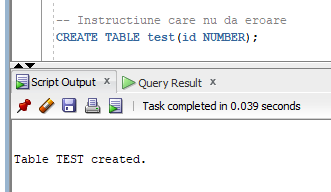
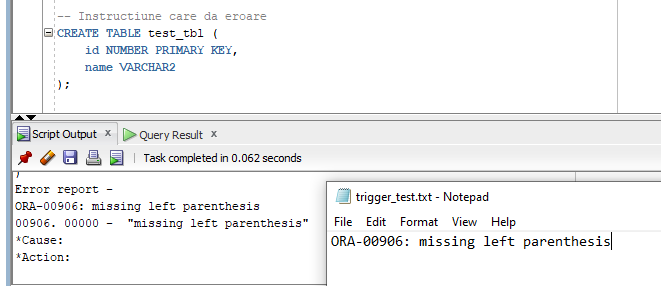
DROP TABLE test\_tbl;

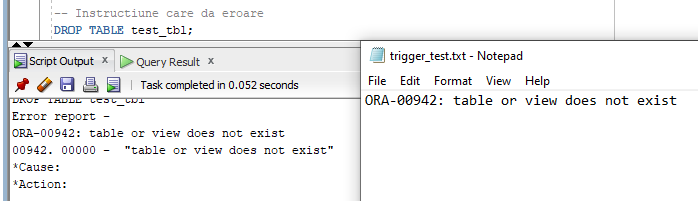
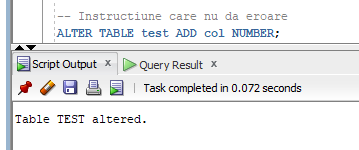
-- Instructiune care da eroare

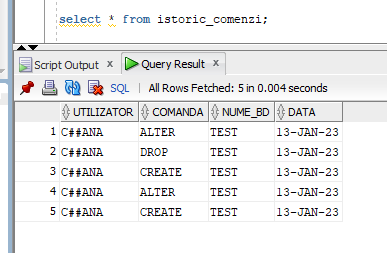
ALTER TABLE test\_tbl ADD col VARCHAR2;

-- Instructiune care nu da eroare

ALTER TABLE test ADD col NUMBER;







## Definiți un pachet care să conțină toate obiectele definite în cadrul proiectului.

CREATE OR REPLACE PACKAGE pachet13 AS

-- 6

PROCEDURE actualiz\_oferte\_exp;

-- 7

PROCEDURE actualiz\_pret\_mot(d dealer.nume%TYPE);

-- 8

FUNCTION cel\_mai\_popular\_model\_din\_reprez(v\_nume\_reprez\_auto REPREZ\_AUTO.nume%TYPE)

RETURN VARCHAR2;

-- 9

PROCEDURE mareste\_pret\_oferta(v\_nume\_model MODEL\_MASINA.nume%TYPE, v\_nume\_dealer DEALER.nume%TYPE);

END pachet13;

/

CREATE OR REPLACE PACKAGE BODY pachet13 AS

--6. Afisare si actualizare de oferte expirate

PROCEDURE actualiz\_oferte\_exp IS

TYPE oferta\_record IS RECORD

(cod\_config oferta.id\_configuratie%TYPE,

pret oferta.pret\_oferta%TYPE,

cod\_client oferta.id\_client%TYPE,

cod\_dealer oferta.id\_dealer%TYPE);

TYPE t\_tbl\_imbricat IS TABLE OF oferta\_record;

t\_oferta t\_tbl\_imbricat;

TYPE dealer\_record IS RECORD

(cod\_dealer dealer.id\_dealer%TYPE,

nume dealer.nume%TYPE,

prenume dealer.prenume%TYPE);

TYPE t\_tbl\_indx\_dealer IS TABLE OF dealer\_record INDEX BY PLS\_INTEGER;

t\_dealer t\_tbl\_indx\_dealer := t\_tbl\_indx\_dealer();

TYPE client\_record IS RECORD

(cod\_client client.id\_client%TYPE,

nume client.nume%TYPE,

prenume client.prenume%TYPE);

TYPE t\_tbl\_indx\_client IS TABLE OF client\_record INDEX BY PLS\_INTEGER;

t\_client t\_tbl\_indx\_client := t\_tbl\_indx\_client();

are\_oferte\_expirate BOOLEAN;

BEGIN

SELECT id\_dealer, nume, prenume BULK COLLECT INTO t\_dealer

FROM dealer

ORDER BY nume;

SELECT id\_client, nume, prenume BULK COLLECT INTO t\_client

FROM client;

SELECT id\_configuratie, pret\_oferta, id\_client, id\_dealer BULK COLLECT INTO t\_oferta

FROM oferta

WHERE data\_expirare < sysdate;

IF t\_oferta.COUNT = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista oferte expirate');

RETURN;

END IF;

DBMS\_OUTPUT.PUT\_LINE('Ofertele expirate sunt:');

FOR i IN t\_dealer.FIRST..t\_dealer.LAST LOOP

DBMS\_OUTPUT.PUT\_LINE('- Dealerul ' || t\_dealer(i).nume ||' '||t\_dealer(i).prenume||': ');

are\_oferte\_expirate := false;

FOR j IN t\_oferta.FIRST..t\_oferta.LAST LOOP

IF t\_oferta(j).cod\_dealer = t\_dealer(i).cod\_dealer THEN

are\_oferte\_expirate := true;

DBMS\_OUTPUT.PUT(' -> oferta pentru configuratia ' || t\_oferta(j).cod\_config || ' facuta pentru clientul ');

FOR k IN t\_client.FIRST..t\_client.LAST LOOP

IF t\_oferta(j).cod\_client = t\_client(k).cod\_client THEN

DBMS\_OUTPUT.PUT(t\_client(k).nume || ' ' || t\_client(k).prenume);

END IF;

END LOOP;

DBMS\_OUTPUT.PUT\_LINE(' cu pretul de ' || t\_oferta(j).pret);

END IF;

END LOOP;

IF are\_oferte\_expirate = false THEN

DBMS\_OUTPUT.PUT\_LINE(' Nu are nicio oferta expirata');

END IF;

END LOOP;

FOR o IN t\_oferta.FIRST..t\_oferta.LAST LOOP

UPDATE OFERTA SET data\_inceput = sysdate WHERE id\_configuratie=t\_oferta(o).cod\_config and id\_dealer=t\_oferta(o).cod\_dealer and id\_client=t\_oferta(o).cod\_client;

UPDATE OFERTA SET data\_expirare = add\_months(sysdate, 24) WHERE id\_configuratie=t\_oferta(o).cod\_config and id\_dealer=t\_oferta(o).cod\_dealer and id\_client=t\_oferta(o).cod\_client;

END LOOP;

END actualiz\_oferte\_exp;

--7. Scumpire motorizari care au emisii si capaciatea cilindirca mare, proportional cu nr de oferte in care apar

PROCEDURE actualiz\_pret\_mot(d dealer.nume%TYPE) IS

-- cursor cu toate configuratiile care apar in ofertele

-- dealerului cu numele dat ca parametru

CURSOR c7 (d1 dealer.nume%TYPE) IS

SELECT c.id\_configuratie, c.id\_motorizare

FROM oferta o, configuratie c, dealer de

WHERE o.id\_configuratie = c.id\_configuratie and o.id\_dealer = de.id\_dealer

and lower(de.nume) = lower(d1);

v\_pret\_vechi NUMBER;

v\_pret\_nou NUMBER;

v\_macar\_o\_actualizare BOOLEAN := FALSE;

v\_contor NUMBER;

BEGIN

SELECT COUNT(\*) INTO v\_contor

FROM dealer de

WHERE lower(de.nume)=lower(d);

IF v\_contor = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista niciun dealer cu acest nume');

ELSE

FOR i in c7(d) LOOP

FOR j IN (SELECT \*

FROM motorizare

WHERE emisii > 120 and capacitate > (SELECT AVG(capacitate)

FROM motorizare)) LOOP

IF i.id\_motorizare = j.id\_motorizare THEN

SELECT pret INTO v\_pret\_vechi

FROM motorizare

WHERE id\_motorizare=i.id\_motorizare;

UPDATE motorizare

SET pret = pret + 500

WHERE id\_motorizare=i.id\_motorizare;

SELECT pret INTO v\_pret\_nou

FROM motorizare

WHERE id\_motorizare=i.id\_motorizare;

DBMS\_OUTPUT.PUT\_LINE('Pretul motorizarii cu id-ul ' || i.id\_motorizare || ' era ' || v\_pret\_vechi || ' si acum e ' || v\_pret\_nou);

v\_macar\_o\_actualizare := TRUE;

END IF;

END LOOP;

END LOOP;

IF v\_macar\_o\_actualizare = FALSE THEN

DBMS\_OUTPUT.PUT\_LINE('Conditiile nu sunt indeplinite de nicio motorizare.');

END IF;

END IF;

END actualiz\_pret\_mot;

--8. Cel mai popular model din configuratii dintr-o anumita reprezentanta

FUNCTION cel\_mai\_popular\_model\_din\_reprez(v\_nume\_reprez\_auto REPREZ\_AUTO.nume%TYPE)

RETURN VARCHAR2 IS

v\_id\_reprez REPREZ\_AUTO.id\_reprez%TYPE;

TYPE vect IS VARRAY(1000) OF MODEL\_MASINA.nume%TYPE;

modele vect := vect();

v\_contor NUMBER;

FARA\_REPREZ EXCEPTION;

FARA\_CONFIG EXCEPTION;

PREA\_MULTE\_MODELE EXCEPTION;

BEGIN

SELECT COUNT(\*) INTO v\_contor

FROM REPREZ\_AUTO

WHERE lower(nume) = lower(v\_nume\_reprez\_auto);

IF v\_contor = 0 THEN

RAISE FARA\_REPREZ;

END IF; -- nu avem cum sa avem contorul > 1 pt ca numele din tabelul reprez\_auto este unique

SELECT id\_reprez INTO v\_id\_reprez

FROM REPREZ\_AUTO

WHERE lower(nume) = lower(v\_nume\_reprez\_auto);

-- datorita verificarii de mai sus stim ca exista o reprezentanta si numai una cu acest nume deci nu se vor genera erori

-- luam toate modelele care au numarul de aparitii din configuratiile reprezentantei maxim

SELECT mm.nume BULK COLLECT INTO modele

FROM REPREZ\_AUTO r JOIN MODELE\_DISP md ON r.id\_reprez=md.id\_reprez

JOIN MODEL\_MASINA mm ON md.id\_model=mm.id\_model

JOIN CONFIGURATIE c ON mm.id\_model=c.id\_model

WHERE r.id\_reprez = v\_id\_reprez

GROUP BY c.id\_model, mm.nume

HAVING COUNT(\*) = (

SELECT MAX(COUNT(cc.id\_model))

FROM CONFIGURATIE cc

WHERE cc.id\_reprez = v\_id\_reprez

GROUP BY cc.id\_model

);

IF modele.count = 0 THEN

RAISE FARA\_CONFIG;

ELSIF modele.count > 1 THEN

RAISE PREA\_MULTE\_MODELE;

END IF;

RETURN 'Modelul care apare cel mai des in configuratii in reprezentanta ' || v\_nume\_reprez\_auto || ' este ' || modele(1) ||'.';

EXCEPTION

WHEN FARA\_REPREZ THEN

RETURN 'Nu exista nicio reprezentanta auto cu acest nume.';

WHEN FARA\_CONFIG THEN

RETURN 'Nu exista configuratii incercate pentru acesta reprezentanta.';

WHEN PREA\_MULTE\_MODELE THEN

RETURN 'Exista mai multe modele cu numarul de aparitii maxim in configuratii pentru acesta reprezentanta.';

END cel\_mai\_popular\_model\_din\_reprez;

--9. Scumpire oferta pentru un model de masina si un dealer dat

PROCEDURE mareste\_pret\_oferta(v\_nume\_model MODEL\_MASINA.nume%TYPE, v\_nume\_dealer DEALER.nume%TYPE) IS

v\_id\_model MODEL\_MASINA.id\_model%TYPE;

v\_contor NUMBER;

v\_nr\_select NUMBER := 0;

v\_id\_dealer DEALER.id\_dealer%TYPE;

v\_oferta\_id\_conf OFERTA.id\_configuratie%TYPE;

v\_oferta\_id\_dealer OFERTA.id\_dealer%TYPE;

v\_oferta\_id\_client OFERTA.id\_client%TYPE;

BEGIN

SELECT id\_model INTO v\_id\_model

FROM MODEL\_MASINA

WHERE lower(nume) = lower(v\_nume\_model);

v\_nr\_select := 1;

SELECT id\_dealer INTO v\_id\_dealer

FROM DEALER

WHERE lower(nume) = lower(v\_nume\_dealer);

v\_nr\_select := 2;

SELECT o.id\_configuratie, o.id\_dealer, o.id\_client INTO v\_oferta\_id\_conf, v\_oferta\_id\_dealer, v\_oferta\_id\_client

FROM MODEL\_MASINA mm JOIN CONFIGURATIE c ON mm.id\_model=c.id\_model

JOIN OFERTA o ON c.id\_configuratie=o.id\_configuratie

JOIN DEALER d ON d.id\_dealer=o.id\_dealer

JOIN CLIENT cl ON cl.id\_client=o.id\_client

WHERE mm.id\_model= v\_id\_model and d.id\_dealer=v\_id\_dealer and trunc(months\_between(sysdate, cl.data\_nastere) / 12) < 45;

UPDATE OFERTA o

SET pret\_oferta = pret\_oferta + pret\_oferta \* 0.05

WHERE o.id\_configuratie=v\_oferta\_id\_conf and o.id\_dealer=v\_oferta\_id\_dealer and o.id\_client=v\_oferta\_id\_client;

DBMS\_OUTPUT.PUT\_LINE('Update pret oferta realizat cu succes');

EXCEPTION

WHEN TOO\_MANY\_ROWS THEN

DBMS\_OUTPUT.PUT\_LINE('Exista mai multe oferte cu modelul dat, dealerul dat si facute pentru un client mai tanar de 45 de ani.');

WHEN NO\_DATA\_FOUND THEN

IF v\_nr\_select = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista niciun model de masina cu numele dat.');

ELSIF v\_nr\_select = 1 THEN

DBMS\_OUTPUT.PUT\_LINE('Nu exista niciun dealer cu numele dat.');

ELSE

DBMS\_OUTPUT.PUT\_LINE('Nu exista nicio oferta cu modelul dat, dealerul dat si facuta pentru un client mai tanar de 45 de ani.');

END IF;

END mareste\_pret\_oferta;

END pachet13;

/

-- Testare pachet

BEGIN

pachet13.actualiz\_oferte\_exp();

pachet13.actualiz\_pret\_mot('Moraru');

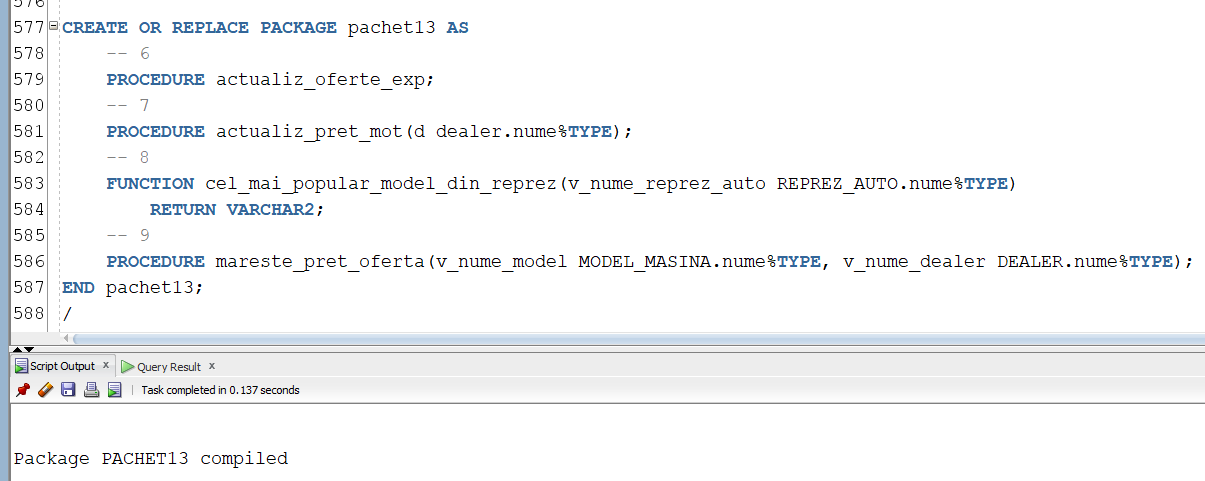
DBMS\_OUTPUT.PUT\_LINE(pachet13.cel\_mai\_popular\_model\_din\_reprez('Porsche Bucuresti Vest'));

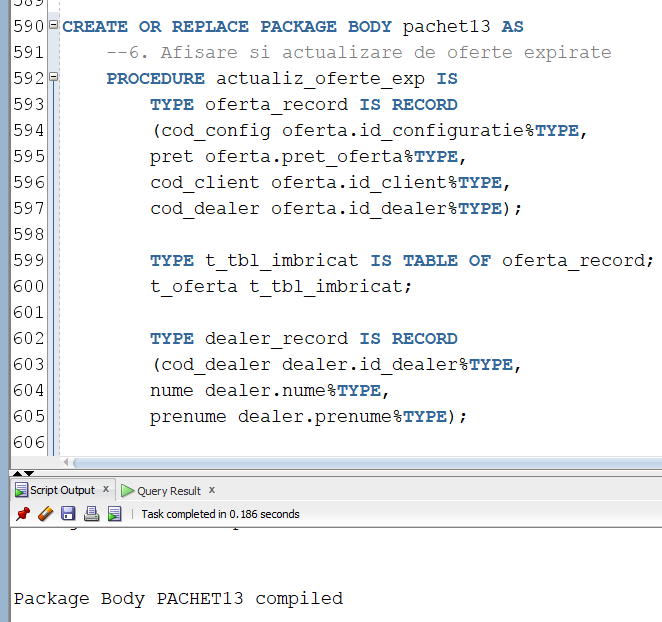
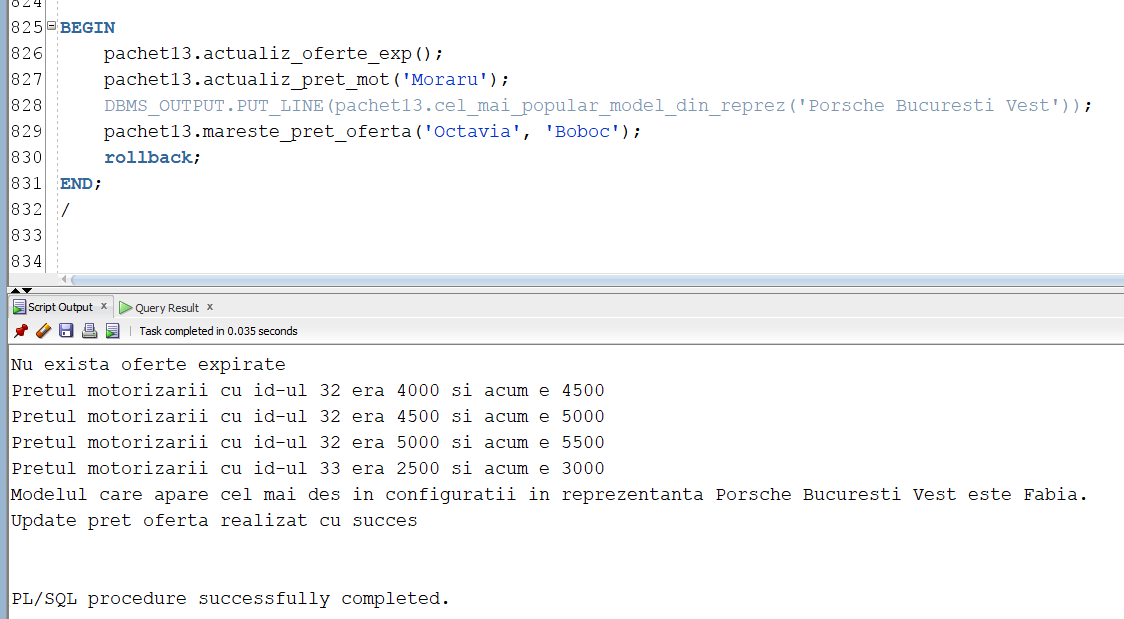
pachet13.mareste\_pret\_oferta('Octavia', 'Boboc');

rollback;

END;

/





## Definiți un pachet care să includă tipuri de date complexe și obiecte necesare unui flux de acțiuni integrate, specifice bazei de date definite (minim 2 tipuri de date, minim 2 funcții, minim 2 proceduri).

**Enunț:** Definiți un pachet pentru gestionarea de oferte și configurații care să conțină:

* Procedura populeaza\_cu\_extraoptiuni
* Procedura afisare\_extraoptiuni
* Procedura sterge\_extraoptiuni
* Procedura populeaza\_cu\_oferte
* Procedura afisare\_oferte
* Procedura sterge\_oferte
* Functia nr\_clienti\_oferta\_reducere (procent)
* Functia config\_cu\_cele\_mai\_multe\_extraop

CREATE OR REPLACE PACKAGE pachet14 AS

TYPE oferta\_record IS RECORD

(cod\_config oferta.id\_configuratie%TYPE,

pret oferta.pret\_oferta%TYPE,

cod\_client oferta.id\_client%TYPE,

cod\_dealer oferta.id\_dealer%TYPE);

TYPE t\_tbl\_imbricat IS TABLE OF oferta\_record;

t\_oferta t\_tbl\_imbricat := t\_tbl\_imbricat();

TYPE extraoptiune\_record IS RECORD

(cod\_config configuratie.id\_configuratie%TYPE,

cod\_extraop extraoptiune.id\_extraoptiune%TYPE,

nume extraoptiune.nume%TYPE);

TYPE vector IS VARRAY (100) OF extraoptiune\_record;

TYPE matrice IS VARRAY (100) OF vector;

--matrice in care retin extraoptiunile tuturor configuratilor

mat matrice := matrice();

PROCEDURE populeaza\_cu\_extraoptiuni;

PROCEDURE afisare\_extraoptiuni;

PROCEDURE sterge\_extraoptiuni;

PROCEDURE populeaza\_cu\_oferte;

PROCEDURE afisare\_oferte;

PROCEDURE sterge\_oferte;

FUNCTION nr\_clienti\_oferta\_reducere (procent IN NUMBER)

RETURN NUMBER;

FUNCTION config\_cu\_cele\_mai\_multe\_extraop

RETURN configuratie.id\_configuratie%TYPE;

END pachet14;

/

CREATE OR REPLACE PACKAGE BODY pachet14 AS

PROCEDURE populeaza\_cu\_extraoptiuni IS

r extraoptiune\_record;

CURSOR c IS

SELECT c.id\_configuratie

FROM configuratie c;

CURSOR c2(conf configuratie.id\_configuratie%TYPE) IS

SELECT c.id\_configuratie, e.id\_extraoptiune, e.nume

FROM configuratie c, conf\_extraop ce, extraoptiune e

WHERE e.id\_extraoptiune=ce.id\_extraoptiune and ce.id\_configuratie=c.id\_configuratie

and conf=c.id\_configuratie;

BEGIN

FOR i IN c LOOP

mat.EXTEND();

mat(mat.LAST) := vector();

FOR j in c2(i.id\_configuratie) LOOP

r.cod\_config := j.id\_configuratie;

r.cod\_extraop := j.id\_extraoptiune;

r.nume := j.nume;

mat(mat.LAST).EXTEND();

mat(mat.LAST)(mat(mat.LAST).LAST) := r;

END LOOP;

END LOOP;

END populeaza\_cu\_extraoptiuni;

PROCEDURE afisare\_extraoptiuni IS

BEGIN

FOR i IN 1..mat.COUNT LOOP

IF mat(i).COUNT > 0 THEN

DBMS\_OUTPUT.PUT('Configuratia cu id-ul ' || mat(i)(1).cod\_config || ' are urmatoarele extraoptiuni: ');

FOR j IN 1..mat(i).COUNT LOOP

DBMS\_OUTPUT.PUT(mat(i)(j).nume || ', ');

END LOOP;

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

END IF;

END LOOP;

END afisare\_extraoptiuni;

PROCEDURE sterge\_extraoptiuni IS

BEGIN

FOR i IN 1..mat.COUNT LOOP

mat(i) := vector();

END LOOP;

mat := matrice();

END sterge\_extraoptiuni;

PROCEDURE populeaza\_cu\_oferte IS

r oferta\_record;

CURSOR c IS

SELECT id\_configuratie, pret\_oferta, id\_client, id\_dealer

FROM oferta;

BEGIN

OPEN c;

LOOP

FETCH c INTO r;

EXIT WHEN c%NOTFOUND;

t\_oferta.EXTEND;

t\_oferta(t\_oferta.LAST) := r;

END LOOP;

CLOSE c;

END populeaza\_cu\_oferte;

PROCEDURE afisare\_oferte IS

BEGIN

FOR i IN 1..t\_oferta.COUNT LOOP

DBMS\_OUTPUT.PUT(i || '. Oferta in valoare de ' || t\_oferta(i).pret || ' facuta pentru configuratie cu id-ul ');

DBMS\_OUTPUT.PUT\_LINE(t\_oferta(i).cod\_config || ', id-ul dealer-ului ' || t\_oferta(i).cod\_dealer || ' si id-ul clientului ' || t\_oferta(i).cod\_client);

END LOOP;

END afisare\_oferte;

PROCEDURE sterge\_oferte IS

BEGIN

t\_oferta := t\_tbl\_imbricat();

END sterge\_oferte;

-- numarul de clienti care au primit reducere la oferta mai mare decat un procent dat ca parametru

FUNCTION nr\_clienti\_oferta\_reducere (procent IN NUMBER)

RETURN NUMBER

IS

v\_pret\_conf configuratie.pret\_total%TYPE;

v\_cnt NUMBER := 0;

BEGIN

FOR i IN t\_oferta.FIRST..t\_oferta.LAST LOOP

SELECT pret\_total INTO v\_pret\_conf

FROM configuratie

WHERE t\_oferta(i).cod\_config=id\_configuratie;

IF v\_pret\_conf - t\_oferta(i).pret > procent \* v\_pret\_conf / 100 THEN

v\_cnt := v\_cnt + 1;

END IF;

END LOOP;

return v\_cnt;

END nr\_clienti\_oferta\_reducere;

-- returneaza id-ul configuratiei cu cele mai multe extraoptiuni

FUNCTION config\_cu\_cele\_mai\_multe\_extraop

RETURN configuratie.id\_configuratie%TYPE

IS

v\_id\_conf configuratie.id\_configuratie%TYPE;

v\_max NUMBER := 0;

BEGIN

FOR i IN 1..mat.COUNT LOOP

IF mat(i).COUNT > v\_max THEN

v\_id\_conf := mat(i)(1).cod\_config;

v\_max := mat(i).COUNT;

END IF;

END LOOP;

return v\_id\_conf;

END config\_cu\_cele\_mai\_multe\_extraop;

END pachet14;

/

-- Apel functii/proceduri

BEGIN

pachet14.sterge\_extraoptiuni();

pachet14.populeaza\_cu\_extraoptiuni();

pachet14.afisare\_extraoptiuni();

pachet14.sterge\_oferte();

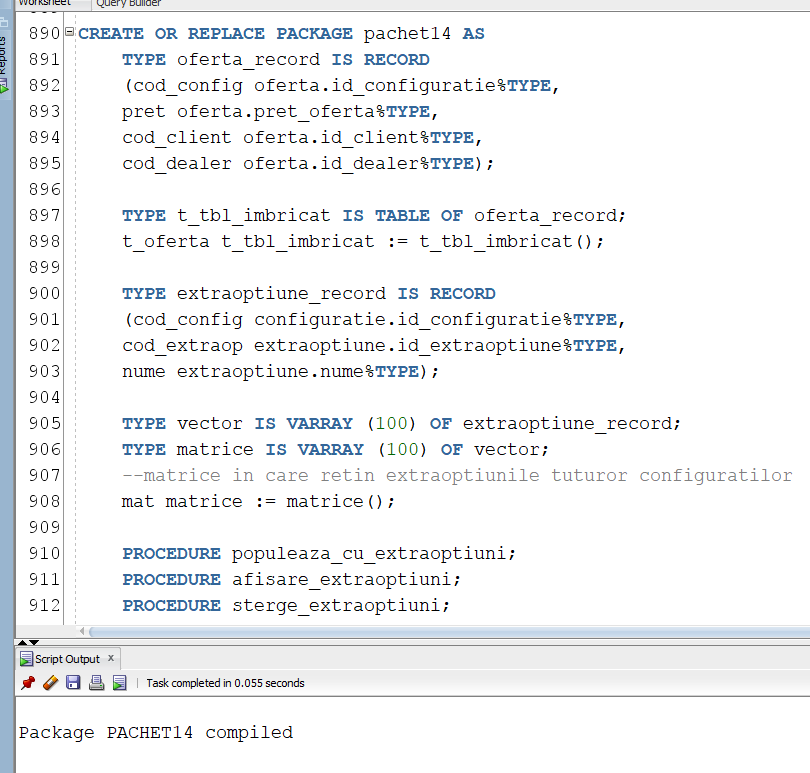
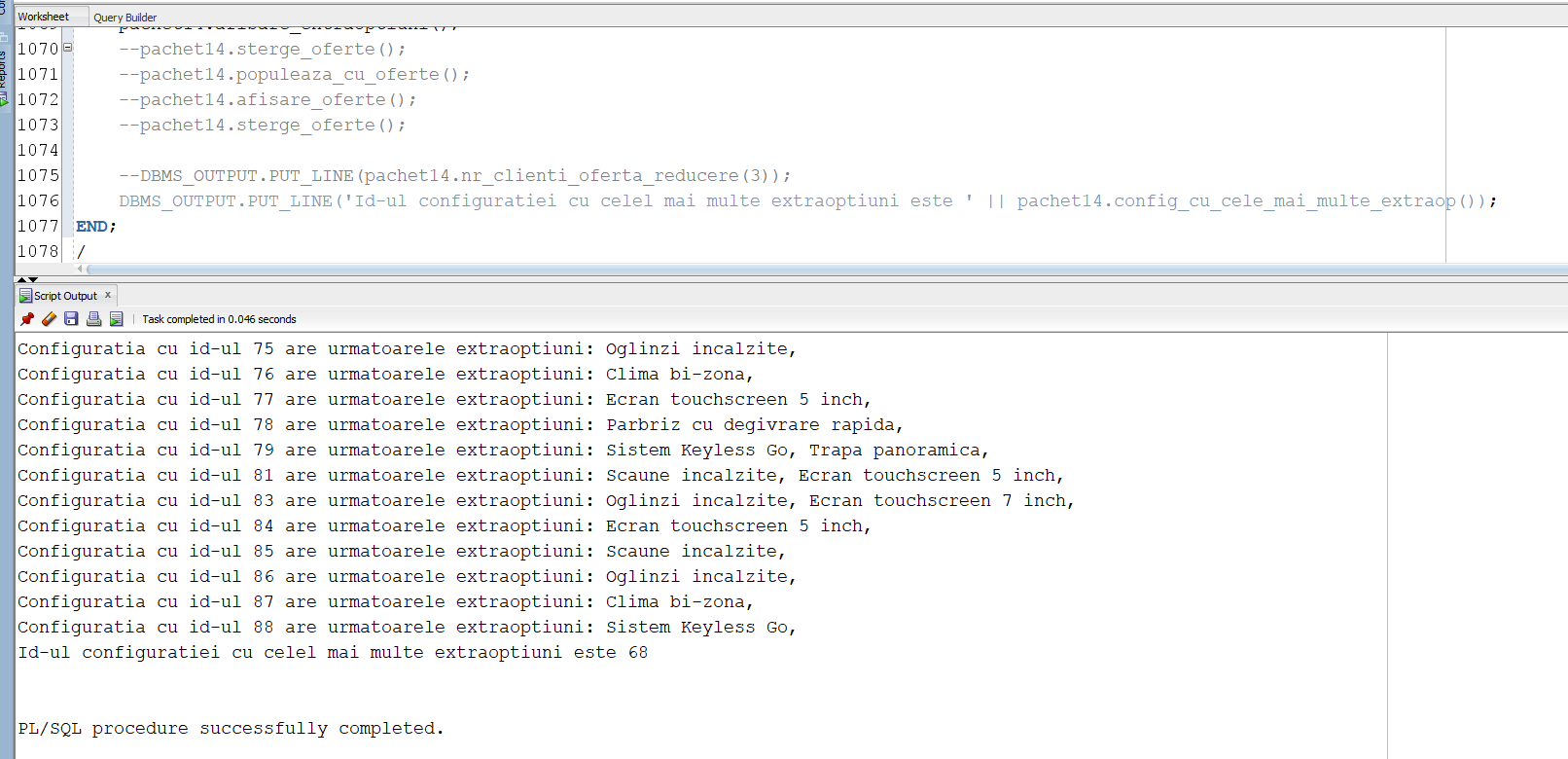
pachet14.populeaza\_cu\_oferte();

pachet14.afisare\_oferte();

DBMS\_OUTPUT.PUT\_LINE(pachet14.nr\_clienti\_oferta\_reducere(3));

DBMS\_OUTPUT.PUT\_LINE('Id-ul configuratiei cu celel mai multe extraoptiuni este ' || pachet14.config\_cu\_cele\_mai\_multe\_extraop());

END;

/

