Băeșu Șerban-Paul

Grupa 132

Proiect Baze de Date- Gestionarea unui spital

CUPRINS

Descriere…………………………………………………………………………………………………………..3

Constrangeri……………………………………………………………………………………………………..3

Descrierea entitatilor………………………………………………………………………………………..4

Descrierea relatiilor…………………………………………………………………………………………..5

Descrierea atributelor……………………………………………………………………………………….7

ERD…………………………………………………………………………………………………………………..10

Diagrama conceptuala………………………………………………………………………………………11

Scheme relationale……………………………………………………………………………………………13

Forme normale………………………………………………………………………………………………….14

Secventa……………………………………………………………………………………………………………14

Crearea tabelelor………………………………………………………………………………………………15

Cereri………………………………………………………………………………………………………………..42

Actualizare si suprimare…………………………………………………………………………………….50

1. **Descrierea modelului și a utilității acestuia**

Un spital are angajați de tip medic și paramedic. Medicii aparțin unei secții și pot consulta unul sau mai mulți pacienți și le pot administra unul sau mai multe tratamente. De asemenea, fiecare poate realiza mai multe operații. Fiecare tratament conține din unul sau mai multe medicamente, fiecare având cate un preț. Prețul total al tratamentului este suma prețurilor medicamentelor din acel tratament. Fiecare pacient se poate interna o data sau de mai multe ori.

Spitalul deține ambulanțe în care lucreaza paramedicii. Fiecare paramedic lucreaza într-o singura ambulanță și fiecare ambulanță poate avea mai mulți paramedici. O ambulanță poate aduce la spital mai mulți pacienți.

1. **Constrângeri impuse asupra modelului**

* Grupa de sânge poate fi O, A, B sau AB
* Durata tratamentului este măsurată în săptămâni
* Vârsta pacientului este un număr măsurat în ani
* Vârsta medicului este un număr măsurat în ani
* Vârsta medicului este mai mare de 25 ani
* Modul de administrare al unui medicament poate fi oral sau injectat
* Șansele de success ale unei operații reprezintă un număr real între 0 și 100
* Durata operației este măsurată în minute

1. **Descrierea entităților**

* PARAMEDIC
  + Informații despre paramedicii angajați la spital
  + Cheie primară – #paramedic\_id
* AMBULANTA
  + Mașini disponibile ale spitalului folosite la intervenții de urgență
  + Cheie primară – #ambulanta\_id
* MEDIC
  + Informații despre medicii angajați la spital
  + Cheie primară – #medic\_id
* OPERATIE
  + Intervenție chirurgicală realizată de un medic asupra unui pacient
  + Cheie primară – #operatie\_id
* PACIENT
  + Informații despre pacienții internați la spital
  + Cheie primară – #pacient\_id
* FIȘĂ INTERNARE
  + Fișă cu informații despre pacient și motivul internării sale
  + Cheie primară – #fisa\_Internare\_id
* TRATAMENT
  + Metodă de combatere a unei boli cu ajutorul medicamentelor
  + Cheie primară – #tratament\_id
* MEDICAMENT
  + Informații despre medicamentele disponibile la spital
  + Cheie primară – #medicament\_id
* SECTIE
  + Informații despre secțiile în care sunt împărțiți medicii
  + Cheie primară – #sectie\_id
* INFO\_OPERATIE
  + Informații generale despre un tip de operație
  + Cheie primară – #info\_operatie\_id

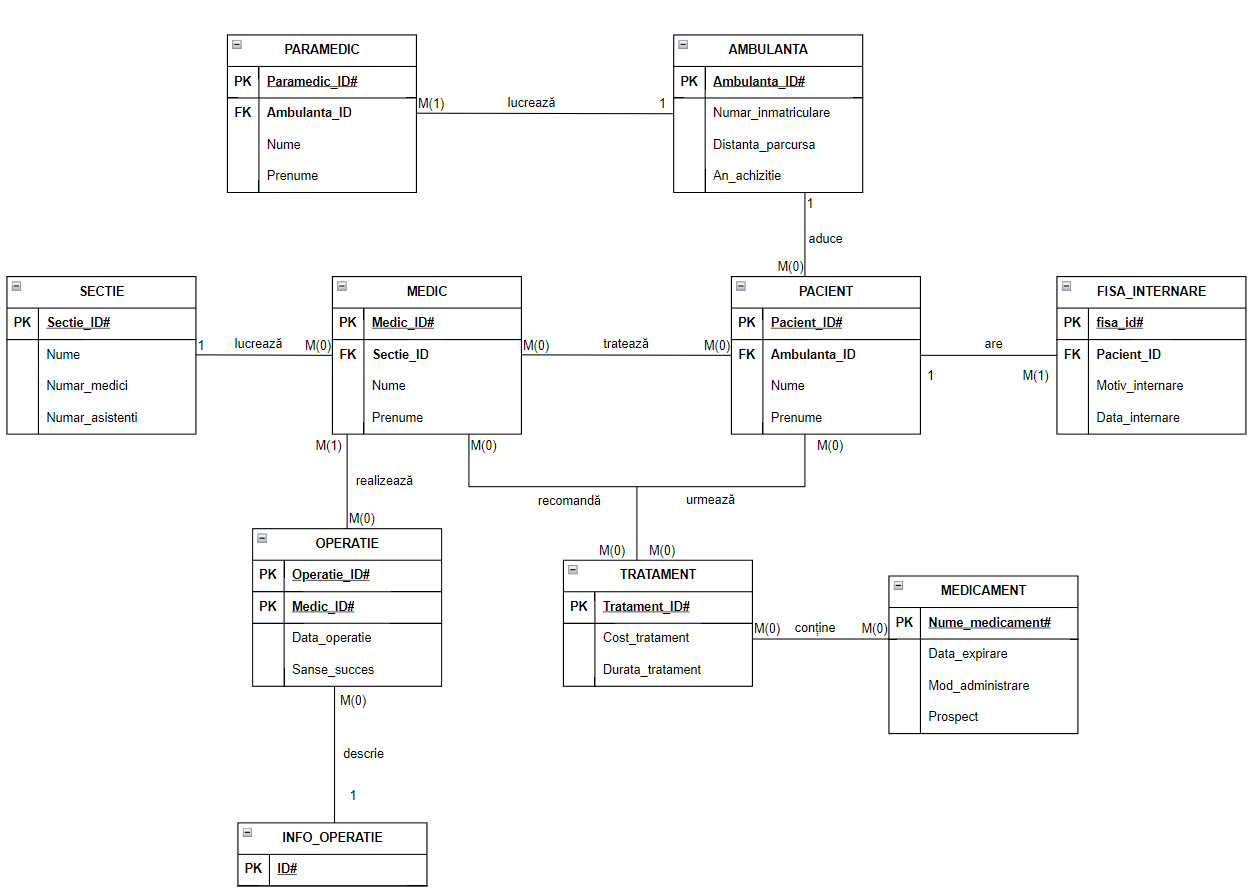
1. **Descrierea relațiilor**

* PARAMEDIC\_lucrează\_în\_AMBULANTA
  + - Fiecare PARAMEDIC trebuie să lucreze într-o singură AMBULANTA
    - În fiecare AMBULANTA pot lucra unul sau mai mulți PARAMEDICI
    - Cardinalitate maximă M:1
    - Cardinalitate minimă 1:1
* AMBULANȚA\_transportă\_PACIENT
  + - Fiecare AMBULANȚĂ poate să transporte mai mulți PACIENȚI
    - Fiecare PACIENT trebuie să fie transportat de o singură AMBULANȚĂ
    - Cardinalitate maximă M:1
    - Cardinalitate minimă 0:1
* MEDIC\_lucrează\_în\_SECȚIE
  + - Fiecare MEDIC trebuie să lucreze într-o singură SECȚIE
    - În fiecare SECTIE pot lucra mai mulți MEDICI
    - Cardinalitate maximă M:1
    - Cardinalitate minimă 0:1
* MEDIC\_realizeaza\_OPERATIE
  + - Fiecare MEDIC poate realiza mai multe OPERATII
    - Fiecare OPERATIE trebuie sa fie realizată de un singur MEDIC
    - Cardinalitate maximă 1:M
    - Cardinalitate minimă 1:0
* PACIENT\_are\_FIȘĂ
  + - Fiecare PACIENT poate avea mai multe FIȘE
    - Fiecare fișă trebuie să fie deținută de un singur PACIENT
    - Cardinalitate maximă 1:M
    - Cardinalitate minimă 1:0
* MEDIC\_TRATEAZA\_PACIENT
  + - Relație în 3
    - Fiecare MEDIC poate să trateze mai mulți PACIENȚI
    - Fiecare PACIENT poate fi tratat de mai mulți MEDICI
    - Fiecare PACIENT poate urma mai multe TRATAMENTE
    - Cardinalitate maximă M:M
    - Cardinalitate minimă 0:0
* TRATAMENT\_contine\_MEDICAMENT
  + - Fiecare TRATAMENT poate conține mai multe MEDICAMENTE
    - Fiecare MEDICAMENT poate fi conținut în mai multe TRATAMENTE
    - Cardinalitate maximă M:M
    - Cardinalitate minimă 0:0

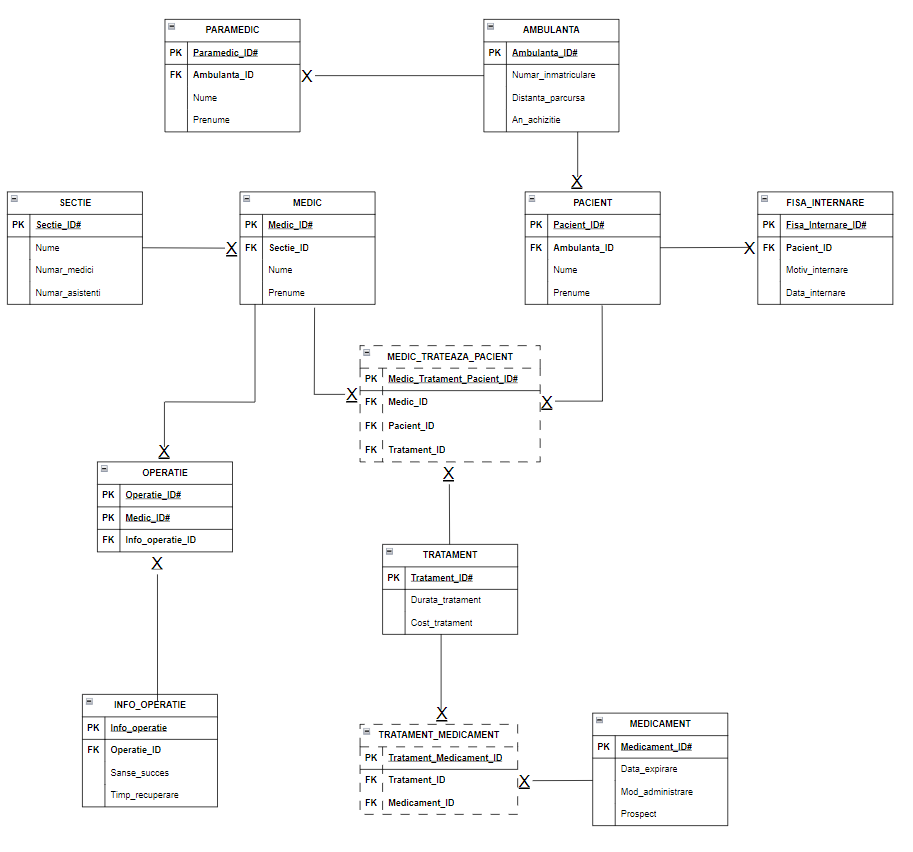
1. **Descrierea atributelor**

* PARAMEDIC
  + Paramedic\_id – number(4), primary key
  + Ambulanta\_id – number(4), foreign key, not null
  + Nume\_paramedic – varchar2(50), not null
  + Prenume\_paramedic – varchar2(50), not null
  + Data\_nastere – date, not null
  + CNP – number(15), not null
  + Salariu – number(10), not null
* AMBULANTA
  + Ambulanta\_id – number(4), primary key
  + Numar\_inmatriculare – varchar2(10), not null
  + Distanta\_parcursa – number(7), not null
  + An\_achizitie – number(4), not null
  + An\_revizie – number(4), not null
* PACIENT
  + Pacient\_id – number(4), primary key
  + Ambulanta\_id – number(4), foreign key, not null
  + Nume\_pacient – varchar2(50), not null
  + Prenume\_pacient – varchar2(50), not null
  + CNP – number(15), not null
  + Varsta – number(2), not null
  + Greutate – number(3, 2), not null
  + Grupa\_sange – varchar2(2), not null
* FISA\_INTERNARE
  + Id – number(4), primary key
  + Pacient\_id – number(4), foreign key, not null
  + Motiv\_internare – varchar2(50), not null
  + Data\_internare – date, not null
  + Data\_externare – date, not null
* MEDIC
  + Medic\_id – number(4), primary key
  + Sectie\_id – number(4), foreign key, not null
  + Nume\_medic – varchar2(50), not null
  + Prenume\_medic – varchar2(50), not null
  + Varsta – number(2), not null
  + Salariu – number(7), not null
  + CNP – number(15), not null
* OPERATIE
  + (Operatie\_id, medic\_id) – (number(4), number(4)), primary key,
  + Info\_id, number(4), foreign key, not null
  + Data\_operatie – date, not null
  + Durata\_operatie – number(4), not null
  + Timp\_recuperare – number(3), not null
* INFO\_OPERATIE
  + Id – number(4), primary key,
  + Tip\_operatie – varchar2(50), not null
  + Sanse\_succes – number(2), not null
  + Timp\_recuperare – number(3), not null
* TRATAMENT
  + Tratament\_id – number(4), primary key
  + Nume\_tratament – varchar(50), not null
  + Cost\_tratament – number(6, 2)
  + Durata\_tratament – number(4), not null
* MEDICAMENT
  + Medicament\_id – number(4), primary key
  + Nume – varchar2(50), not null
  + Data\_expirare – date, not null
  + Mod\_administrare – varchar2(20), not null
  + Prospect – varchar2(50)
* SECTIE
  + Sectie\_ID – number(4), primary key, not null
  + Nume – varchar2(50), not null
  + Numar\_medici – number(2), not null
  + Numar\_asistenti – number(2), not null
  + An\_infiintare – number(4), not null

1. **Realizarea diagramei entitate-relație**



1. **Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6**



1. **Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7**

* PARAMEDIC(paramedic\_id#, ambulanta\_id, nume\_paramedic, prenume\_paramedic, data\_nastere, CNP, salariu)
* AMBULANTA(ambulanta\_id#, numar\_inmatriculare,

distanta\_parcursa, an\_achizitie, an\_revizie)

* PACIENT(pacient\_id#, ambulanta\_id, nume\_pacient, prenume\_pacient, varsta, CNP, greutate, grupa\_sange)
* FISA\_INTERNARE(fisa\_id#, pacient\_id, motiv\_internare, data\_internare, data\_externare)
* SECTIE(sectie\_id#, nume\_sectie, numar\_medici, numar\_asistenti, an\_infiintare)
* MEDIC(medic\_id#, sectie\_id, nume\_medic, prenume\_medic, varsta, salariu, CNP)
* OPERATIE(operatie\_id#, medic\_id#, info\_id, data\_operatie, durata\_operatie)
* INFO\_OPERATIE(info\_id#, tip\_operatie, sanse\_succes, timp\_recuperare)
* TRATAMENT(tratament\_id#, nume\_tratament, cost\_tratament, durata\_tratament)
* MEDIC\_TRATEAZA\_PACIENT(id#, medic\_id, pacient\_id, tratament\_id)
* MEDICAMENT(medicament\_id#, nume, data\_expirare, mod\_administrare, prospect)
* TRATAMENT\_MEDICAMENT(id#, tratament\_id, medicament\_id)

1. **Realizarea normalizării până la forma normală 3(FN1-FN3)**

* FN1 – Tabela PACIENT nu respecta forma normală 1 deoarece conținea atributul detalii\_internare, care era format din atribute atomice. Astfel, am creat tabela FISA\_INTERNARE, unde am introdus toate atributele corespunzătoare și l-am legat de tabela PACIENT
* FN2 – În tabela OPERATIE, atributele sanse\_supravietuire și timp\_recuperare sunt doar niște estimări statistice care nu depind de medicul care realizează operația. Deoarece ele nu depind de întreaga cheie primară, ci doar de operatie\_ID, am creat tabela INFO\_OPERATIE, unde am introdus aceste informații și am pus-o în relație cu tabela inițială
* FN3 – Inițial, toate informațiile din tabela AMBULANTA se aflau în tabela PARAMEDIC, însă cum acestea depind de atributul numar\_inmatriculare care nu este cheie primară, se realizează o dependență tranzitivă. Astfel, am creat o nouă tabelă în care am pus toate informațiile legate de ambulanță

1. **Crearea unei secvențe ce va fi folosită la inserarea datelor în tabele**

create sequence secventa

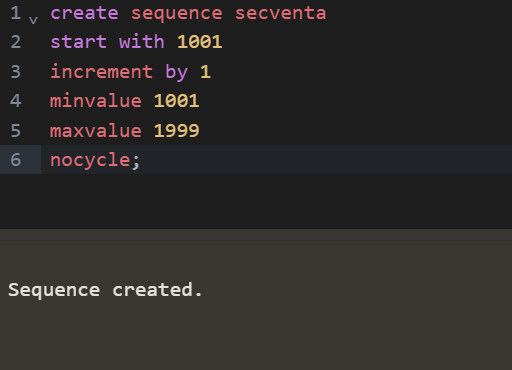
start with 1001

increment by 1

minvalue 1001

maxvalue 1999

nocycle;



1. **Crearea tabelelor SQL și inserarea datelor în acestea**

create table AMBULANTA(

ambulanta\_id number(4) constraint pk\_ambulanta primary key,

numar\_inmatriculare varchar(10) not null,

distanta\_parcursa number(7) not null,

an\_achizitie number(4) not null,

an\_revizie number(4) not null

);

create table PARAMEDIC(

paramedic\_id number(4) constraint pk\_paramedic primary key,

ambulanta\_id number(4) not null,

nume\_paramedic varchar2(50) not null,

prenume\_paramedic varchar2(50) not null,

data\_nasterii date not null,

cnp number(15) not null,

salariu number(10) not null,

foreign key(ambulanta\_id) references AMBULANTA(ambulanta\_id)

);

create table PACIENT(

pacient\_id number(4) constraint pk\_pacient primary key,

ambulanta\_id number(4) not null,

nume\_pacient varchar2(50) not null,

prenume\_pacient varchar2(50) not null,

cnp\_pacient number(15) not null,

varsta number(2) not null,

greutate number(5, 2) not null,

grupa\_sange varchar2(2) check(grupa\_sange = 'O' or grupa\_sange = 'A' or grupa\_sange = 'B' or grupa\_sange = 'AB') not null,

foreign key(ambulanta\_id) references AMBULANTA(ambulanta\_id)

);

create table SECTIE(

sectie\_id number(4) constraint pk\_sectie primary key,

nume\_sectie varchar2(50) not null,

numar\_medici number(2) not null,

numar\_asistenti number(2) not null,

an\_infiintare number(4) not null

);

create table MEDIC(

medic\_id number(4) constraint pk\_medic primary key,

sectie\_id number(4) not null,

nume\_medic varchar2(50) not null,

prenume\_medic varchar2(50) not null,

cnp varchar2(50) not null,

varsta number(2) not null,

salariu number(7) not null,

foreign key(sectie\_id) references SECTIE(sectie\_id)

);

create table FISA\_INTERNARE(

fisa\_id number(4) constraint pk\_fisa primary key,

pacient\_id number(4) not null,

motiv\_internare varchar2(50) not null,

data\_internare date not null,

data\_externare date not null,

foreign key(pacient\_id) references PACIENT(pacient\_id)

);

create table INFO\_OPERATIE(

id number(4) constraint pk\_info\_operatie\_id primary key,

tip\_operatie varchar(50) not null,

sanse\_succes number(2) not null,

timp\_recuperare number(4) not null

);

create table OPERATIE(

operatie\_id number(4),

medic\_id number(4),

info\_id number(4) not null,

data\_operatie date not null,

durata\_operatie number(4) not null,

primary key(operatie\_id, medic\_id),

foreign key(info\_id) references INFO\_OPERATIE(id)

);

create table TRATAMENT(

tratament\_id number(4) primary key,

nume\_tratament varchar(50) not null,

cost\_tratament number(8) not null,

durata\_tratament number(3) not null

);

create table MEDICAMENT(

medicament\_id number(4) primary key,

nume varchar(50) not null,

data\_expirare date not null,

mod\_administrare varchar2(10) check(mod\_administrare = 'capsule' or mod\_administrare = 'injectat') not null,

prospect varchar2(50) not null

);

create table TRATAMENT\_MEDICAMENT(

id number(4) primary key,

tratament\_id number(4) not null,

medicament\_id number(4) not null,

foreign key(tratament\_id) references TRATAMENT(tratament\_id),

foreign key(medicament\_id) references MEDICAMENT(medicament\_id)

);

create table MEDIC\_TRATEAZA\_PACIENT(

id number(4) primary key,

medic\_id number(4) not null,

pacient\_id number(4) not null,

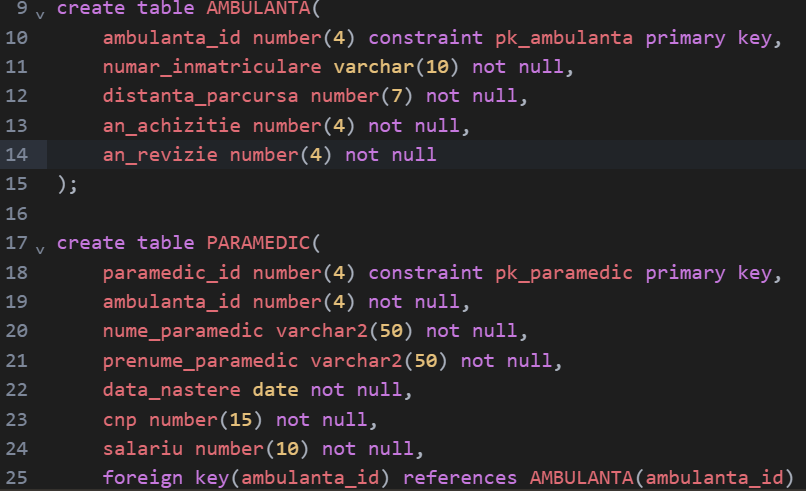
tratament\_id number(4) not null,

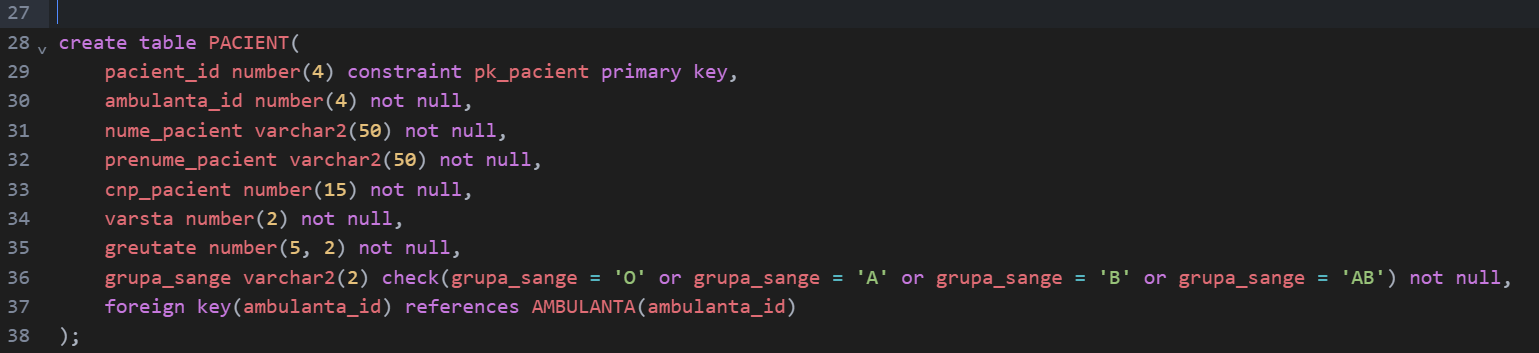
foreign key(medic\_id) references MEDIC(medic\_id),

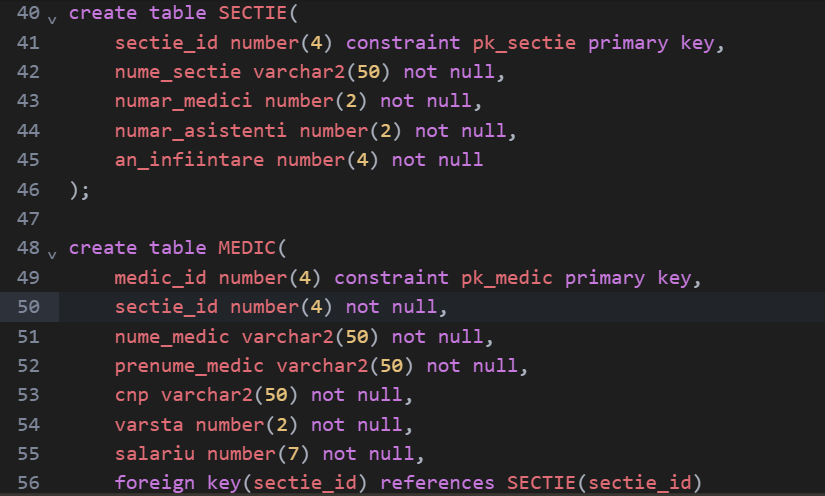
foreign key(pacient\_id) references PACIENT(pacient\_id),

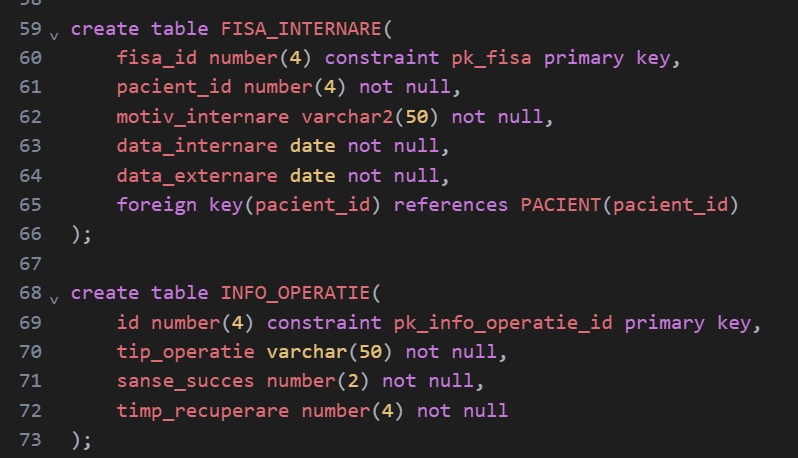
foreign key(tratament\_id) references TRATAMENT(tratament\_id)

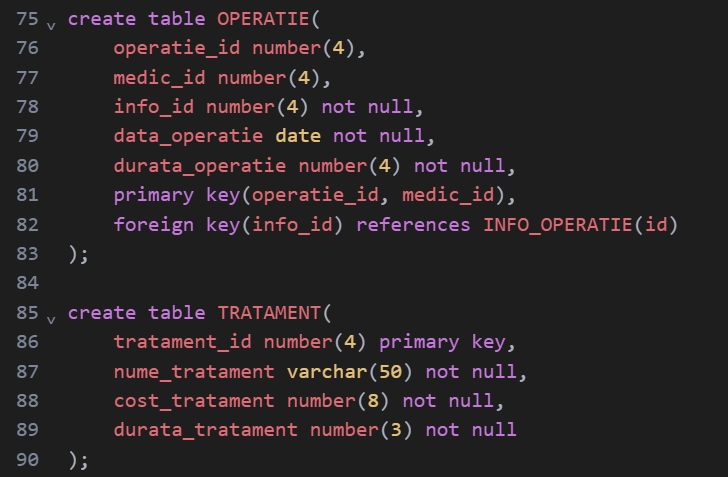
);

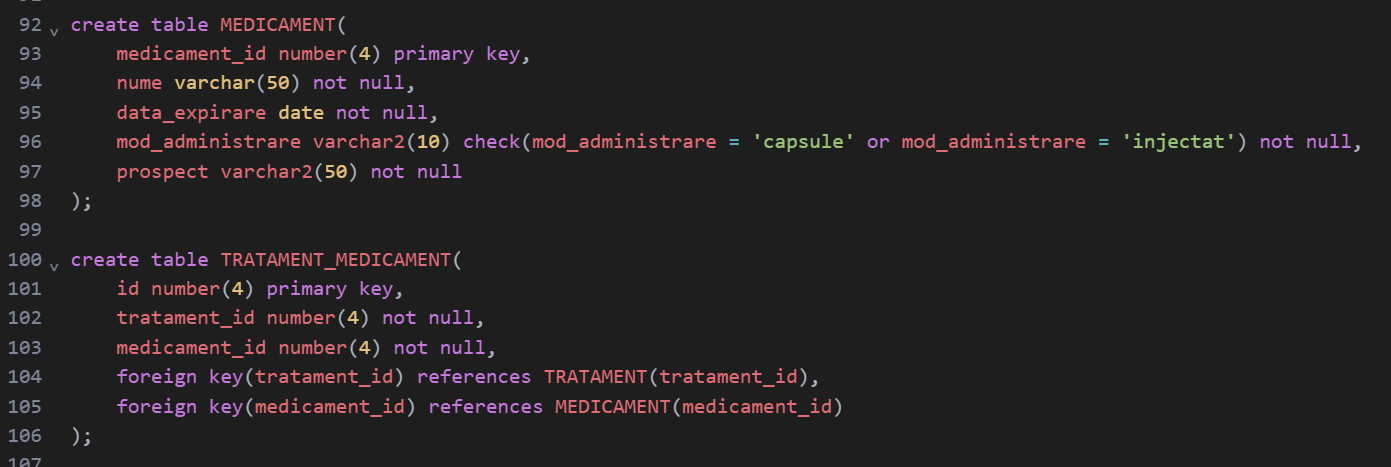


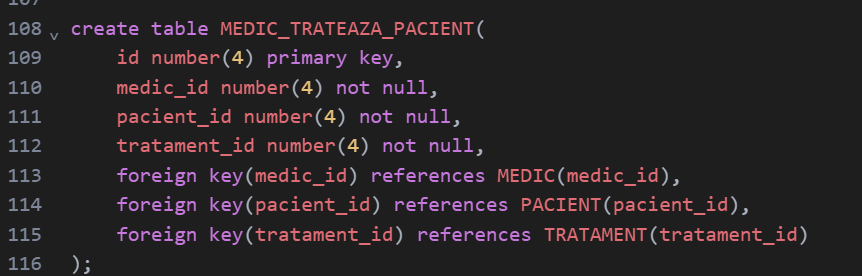












insert into AMBULANTA

values (secventa.nextval, 'B104HFG', 104924, 2011, 2022);

insert into AMBULANTA

values (secventa.nextval, 'B625JGF', 30432, 2018, 2023);

insert into AMBULANTA

values (secventa.nextval, 'B651JGK', 432563, 2008, 2020);

insert into AMBULANTA

values (secventa.nextval, 'B3245NMO', 97432, 2014, 2022);

insert into AMBULANTA

values (secventa.nextval, 'B151FSS', 201512, 2011, 2019);

select \* from ambulanta;

insert into PARAMEDIC

values (1025, 1002, 'Janet', 'Reynolds', '19-MAY-1996', 5015015015041, 7000);

insert into PARAMEDIC

values (1078, 1001, 'Nicolas', 'Faulkner','12-APR-1979', 5038125015041, 5500);

insert into PARAMEDIC

values (1151, 1002, 'Anne', 'Garcia', '09-JUN-2001', 6832583201851, 9200);

insert into PARAMEDIC

values (1301, 1002, 'Isaac', 'Jackson', '30-OCT-1991', 50183521532041, 7700);

insert into PARAMEDIC

values (1088, 1005, 'Nicole', 'Watts', '22-JAN-1989', 5029529350121, 4800);

insert into PARAMEDIC

values (1216, 1003, 'Andrew', 'Hansen', '10-MAR-1999', 5385718395912, 10000);

insert into PARAMEDIC

values (1087, 1001, 'Sarah', 'Johnson', '26-MAY-2000', 4029581259195, 6100);

insert into PARAMEDIC

values (1111, 1005, 'John', 'McDonald', '18-SEP-1982', 2921502594246, 4400);

insert into PARAMEDIC

values (1008, 1003, 'Alex', 'Wallace', '02-JUL-1998', 3952015939250, 8300);

insert into PARAMEDIC

values (1951, 1004, 'Alex', 'Johnson', '02-AUG-1995', 3416325939250, 8500);

select \* from paramedic;

insert into PACIENT

values (1682, 1002, 'Rafael', 'Lozano', 5001018294501, 25, 76.4, 'O');

insert into PACIENT

values (1245, 1005, 'Isaac', 'Price', 2960531398618, 27, 62, 'AB');

insert into PACIENT

values (1067, 1002, 'Christian', 'Davies', 1950624296122, 40, 94.4, 'O');

insert into PACIENT

values (1661, 1002, 'Brian', 'Archer', 2911203390945, 15, 55.8, 'B');

insert into PACIENT

values (1742, 1001, 'Emma', 'Bowen', 2900116523431, 31, 52, 'A');

insert into PACIENT

values (1582, 1005, 'Bonny', 'Krueger', 5010621073106, 44, 60.1, 'AB');

insert into PACIENT

values (1952, 1002, 'Barry', 'Nelson', 2990207122901, 55, 102.7, 'A');

insert into PACIENT

values (1142, 1003, 'Jeffrey', 'Carrey', 2921209458675, 31, 82.4, 'O');

select \* from pacient;

insert into sectie

values (1, 'Chirurgie', 12, 20, 1982);

insert into sectie

values (2, 'Oftalmologie', 6, 6, 1994);

insert into sectie

values (3, 'Pediatrie', 15, 18, 1968);

insert into sectie

values (4, 'Psihiatrie', 8, 10, 2002);

insert into sectie

values (5, 'Urgente', 25, 30, 1966);

select \* from sectie;

insert into MEDIC

values (1141, 1, 'Joe', 'Nixon', 1960215239651, 56, 13000);

insert into MEDIC

values (1935, 5, 'Rachel', 'Myers', 2910512215151, 42, 11500);

insert into MEDIC

values (1245, 5, 'Walter', 'White', 1890818463086, 34, 11000);

insert into MEDIC

values (1161, 2, 'Veronica', 'Barber', 2880521346647, 51, 15200);

insert into MEDIC

values (1052, 4, 'Frank', 'Conley', 2920704303097, 28, 8500);

insert into MEDIC

values (1852, 3, 'Thomas', 'Shelby', 1960127385451, 55, 12900);

insert into MEDIC

values (1526, 1, 'Patrick', 'Bateman', 2990907339842, 42, 12400);

insert into MEDIC

values (1294, 1, 'Jesse', 'Pinkman', 5030907363834, 38, 10200);

select \* from medic;

insert into FISA\_INTERNARE

values (1052, 1661, 'Fractura la mana', '02-MAR-2023', '05-MAR-2023');

insert into FISA\_INTERNARE

values (1755, 1742, 'Accident masina', '23-JUN-2021', '02-JUL-2021');

insert into FISA\_INTERNARE

values (1593, 1142, 'Transplant de rinichi', '17-FEB-2015', '12-MAR-2015');

insert into FISA\_INTERNARE

values (1531, 1067, 'Covid-19', '14-SEP-2020', '29-SEP-2020');

insert into FISA\_INTERNARE

values (1231, 1952, 'Accident de munca', '20-AUG-2022', '01-SEP-2022');

select \* from fisa\_internare;

insert into INFO\_OPERATIE

values (13, 'Apendicectomie clasica', 99, 4);

insert into INFO\_OPERATIE

values (21, 'Tumorectomie mamara', 82, 12);

insert into INFO\_OPERATIE

values (41, 'Tiroidectomie totala', 95, 6);

insert into INFO\_OPERATIE

values (37, 'Lobectomie tiroidiana', 97, 4);

insert into INFO\_OPERATIE

values (93, 'Nefrectomie clasica', 88, 9);

select \* from info\_operatie;

insert into OPERATIE

values (1241, 1245, 21, '15-DEC-2019', 124);

insert into OPERATIE

values (1742, 1161, 41, '09-JUN-2022', 97);

insert into OPERATIE

values (1676, 1052, 37, '25-JAN-2010', 221);

insert into OPERATIE

values (1444, 1526, 13, '02-NOV-2020', 146);

insert into OPERATIE

values (1420, 1294, 93, '19-SEP-2015', 52);

insert into OPERATIE

values (1051, 1052, 13, '31-MAY-2012', 184);

select \* from operatie;

insert into TRATAMENT

values (1, 'Tratament A', 2000, 6);

insert into TRATAMENT

values (2, 'Tratament B', 10000, 14);

insert into TRATAMENT

values (3, 'Tratament C', 400, 3);

insert into TRATAMENT

values (4, 'Tratament D', 40000, 30);

insert into TRATAMENT

values (5, 'Tratament E', 100000, 52);

insert into TRATAMENT

values (6, 'Tratament F', 5000, 4);

select \* from tratament;

insert into MEDICAMENT

values (41, 'Algocalmin', '15-MAR-2025', 'injectat', 'Prospect Algocalmin');

insert into MEDICAMENT

values (55, 'Adrenalina', '22-DEC-2027', 'injectat', 'Prospect Adrenalina');

insert into MEDICAMENT

values (12, 'Acid Acetilsalicilic', '02-APR-2026', 'capsule', 'Prospect Acid Acetilsalicilic');

insert into MEDICAMENT

values (91, 'Nitroglicerina', '15-SEP-2025', 'capsule', 'Prospect Nitroglicerina');

insert into MEDICAMENT

values (53, 'Hidrocortizon Zentiva', '28-JAN-2028', 'injectat', 'Prospect Hidrocortizon Zentiva');

insert into MEDICAMENT

values (88, 'Bromhexim', '09-JUL-2025', 'capsule', 'Prospect Bromhexim');

select \* from medicament;

insert into TRATAMENT\_MEDICAMENT

values (1, 1, 12);

insert into TRATAMENT\_MEDICAMENT

values (2, 5, 91);

insert into TRATAMENT\_MEDICAMENT

values (3, 4, 88);

insert into TRATAMENT\_MEDICAMENT

values (4, 1, 55);

insert into TRATAMENT\_MEDICAMENT

values (5, 3, 55);

insert into TRATAMENT\_MEDICAMENT

values (6, 5, 12);

insert into TRATAMENT\_MEDICAMENT

values (7, 6, 41);

insert into TRATAMENT\_MEDICAMENT

values (8, 5, 91);

insert into TRATAMENT\_MEDICAMENT

values (9, 2, 53);

insert into TRATAMENT\_MEDICAMENT

values (10, 3, 12);

select \* from tratament\_medicament;

insert into MEDIC\_TRATEAZA\_PACIENT

values(1, 1294, 1582, 5);

insert into MEDIC\_TRATEAZA\_PACIENT

values(2, 1052, 1142, 2);

insert into MEDIC\_TRATEAZA\_PACIENT

values(3, 1526, 1067, 6);

insert into MEDIC\_TRATEAZA\_PACIENT

values(4, 1161, 1682, 4);

insert into MEDIC\_TRATEAZA\_PACIENT

values(5, 1852, 1952, 3);

insert into MEDIC\_TRATEAZA\_PACIENT

values(6, 1245, 1742, 1);

insert into MEDIC\_TRATEAZA\_PACIENT

values(7, 1935, 1142, 4);

insert into MEDIC\_TRATEAZA\_PACIENT

values(8, 1245, 1582, 6);

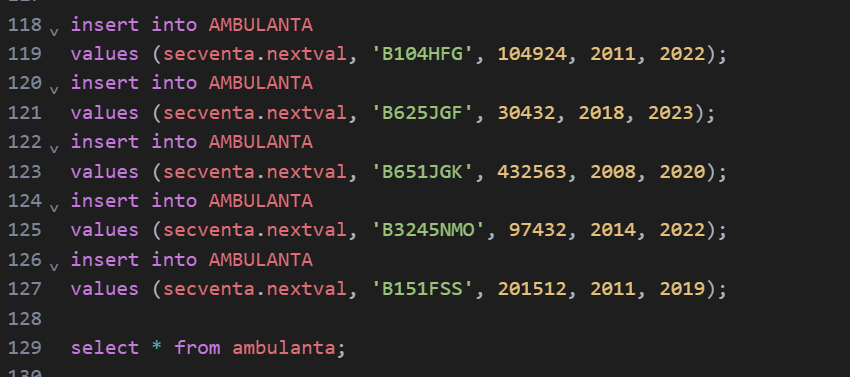
insert into MEDIC\_TRATEAZA\_PACIENT

values(9, 1526, 1067, 3);

insert into MEDIC\_TRATEAZA\_PACIENT

values(10, 1052, 1682, 5);

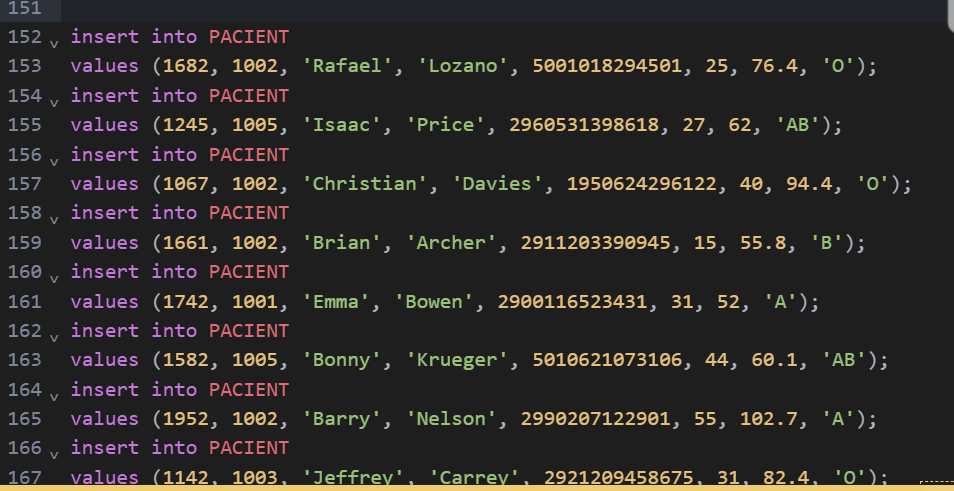
select \* from medic\_trateaza\_pacient;

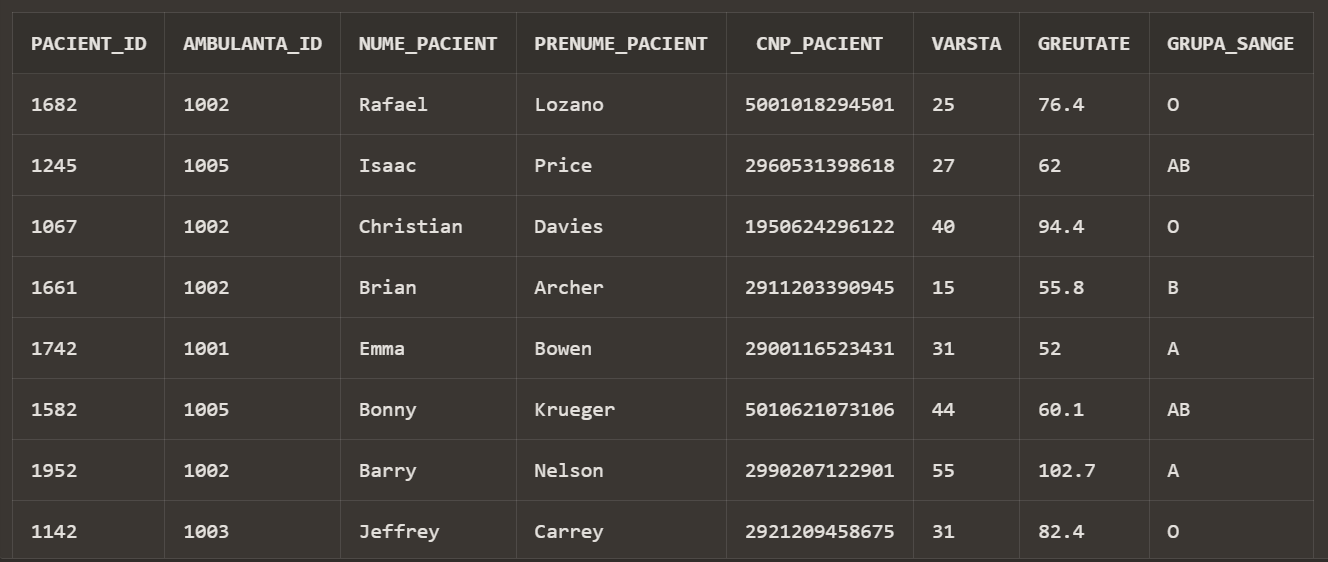


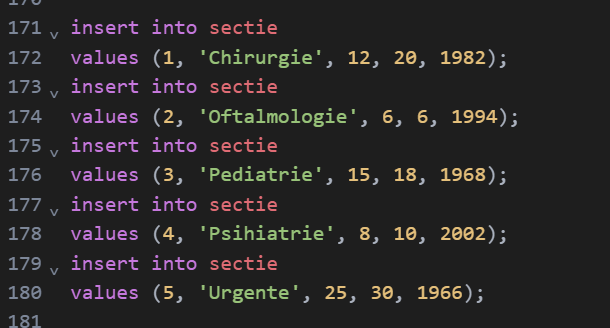


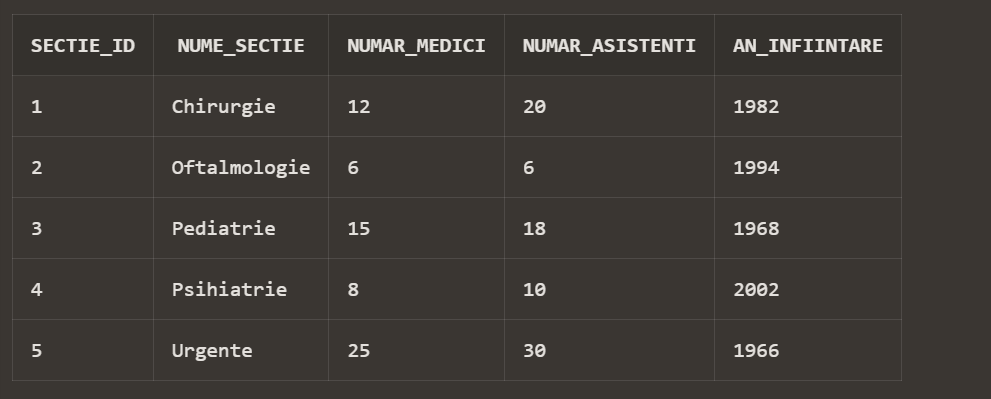




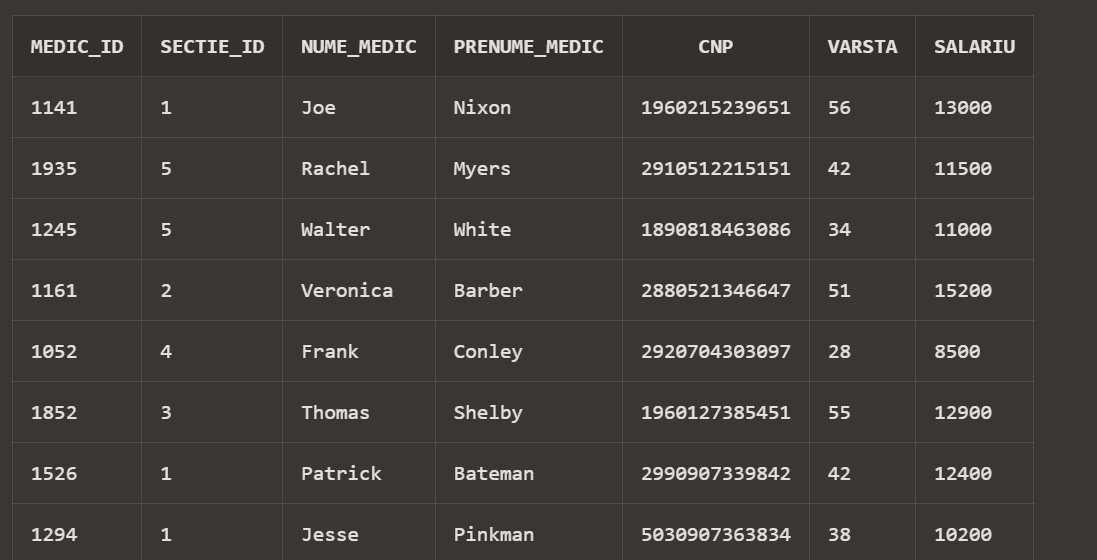


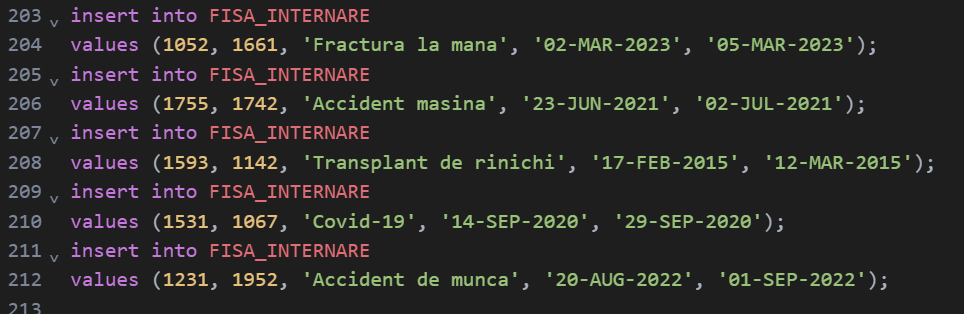




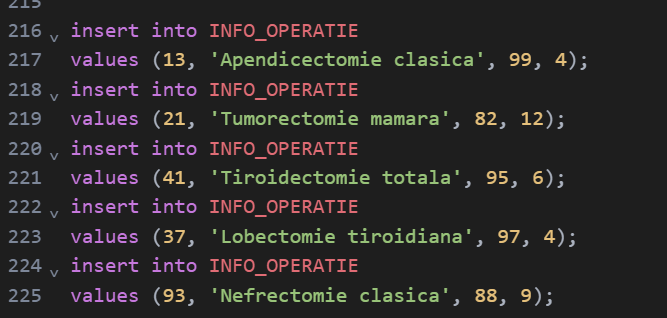




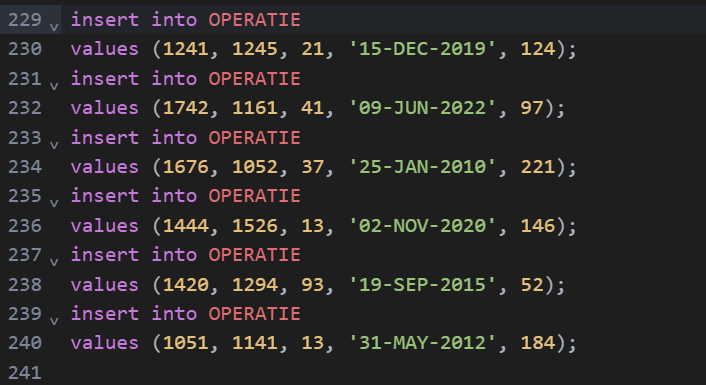


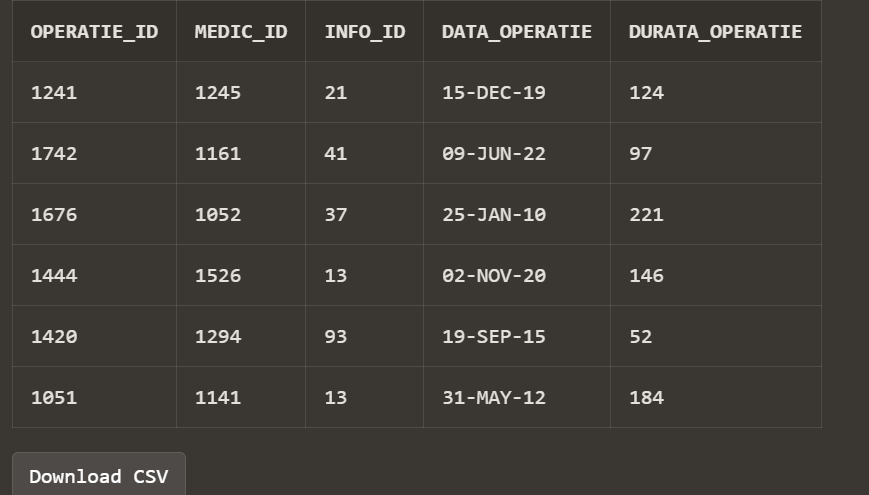


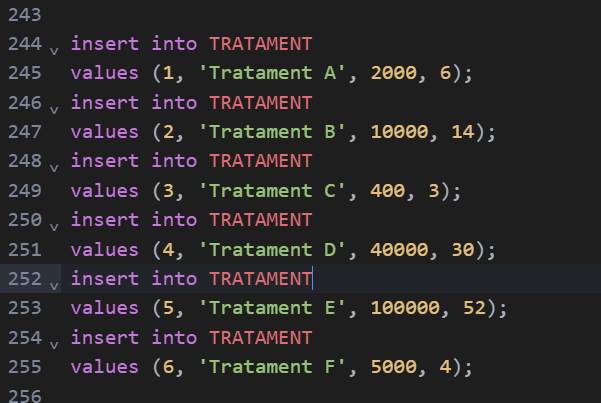




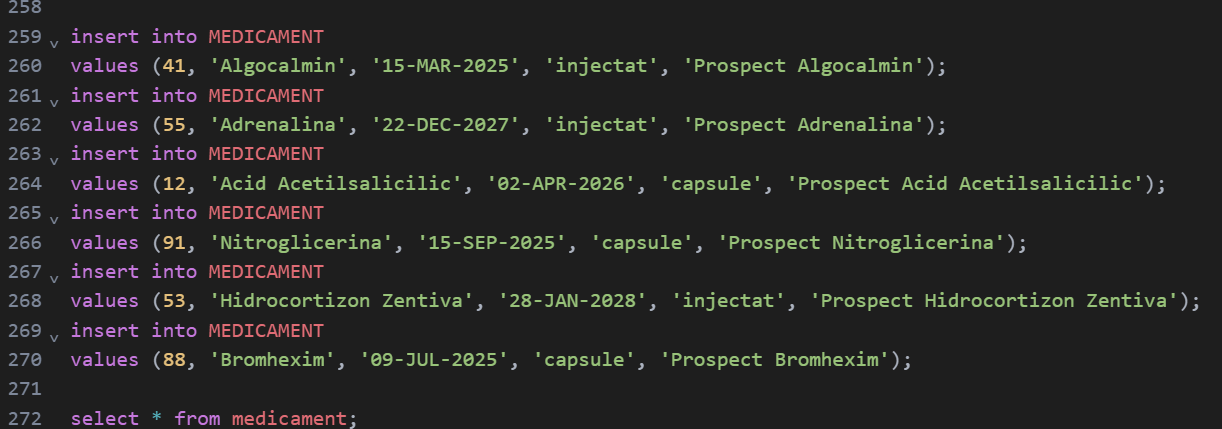


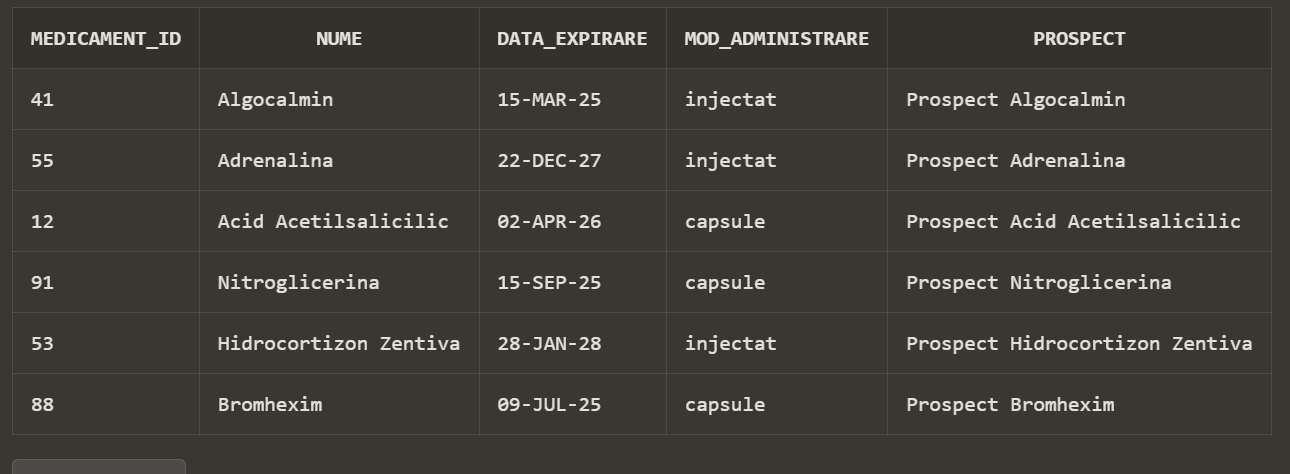


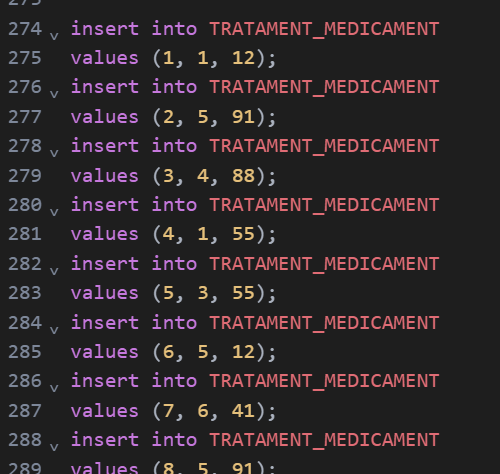


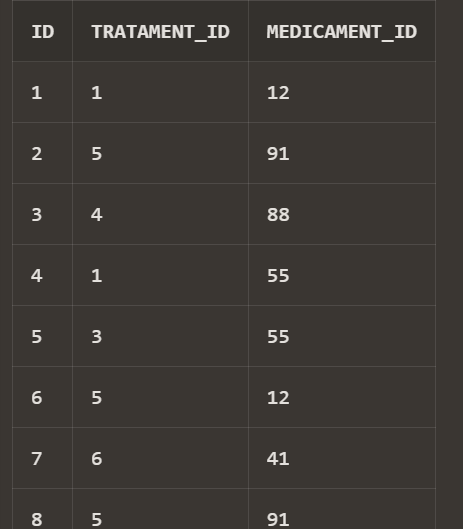


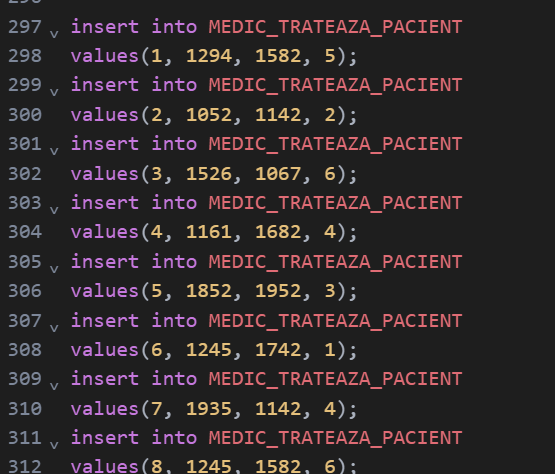


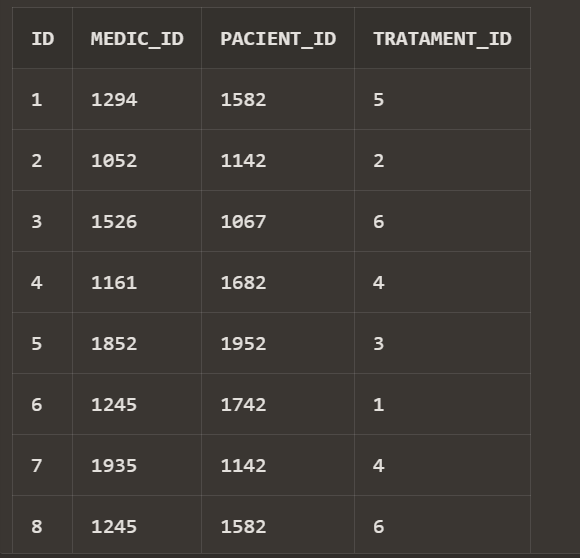












1. **Implementarea a 5 cereri SQL complexe**

--pacientii care urmeaza un tratament mai scump de 10000 care contine medicamente sub forma de capsule

--subcereri nesincronizate in clauza FROM

select distinct p.pacient\_id, p.nume\_pacient, p.prenume\_pacient, t.cost\_tratament

from pacient p, medic\_trateaza\_pacient m, (select t2.tratament\_id, t2.cost\_tratament

from tratament t2, tratament\_medicament tm, medicament med

where t2.tratament\_id = tm.tratament\_id

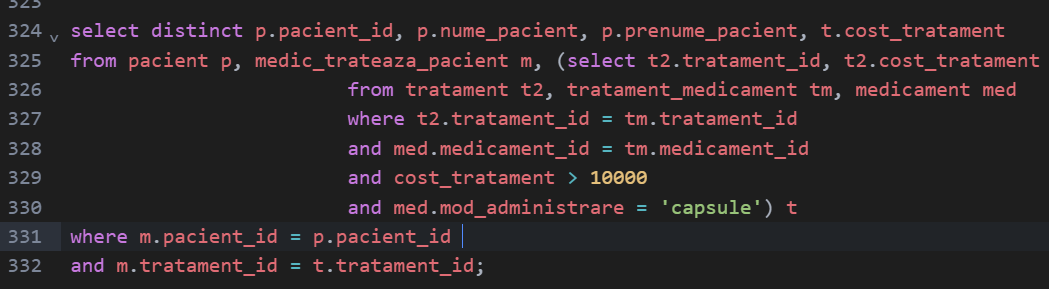
and med.medicament\_id = tm.medicament\_id

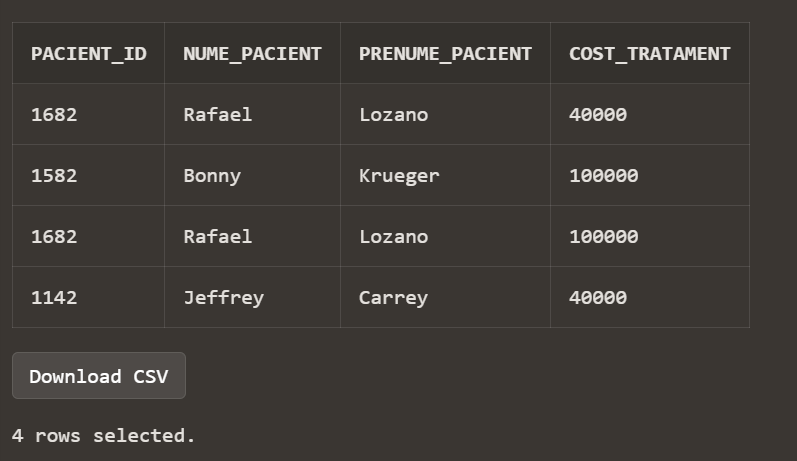
and cost\_tratament > 10000

and med.mod\_administrare = 'capsule') t

where m.pacient\_id = p.pacient\_id

and m.tratament\_id = t.tratament\_id;





--pacientii care au fost tratati de un medic mai tanar decat ei care lucreaza la sectiunea 'Chirurgie'

--subcereri sincronizate la care participa cel putin 3 tabele

select p.pacient\_id, p.nume\_pacient, p.prenume\_pacient

from pacient p, medic med, medic\_trateaza\_pacient mp

where p.pacient\_id = mp.pacient\_id

and med.medic\_id = mp.medic\_id

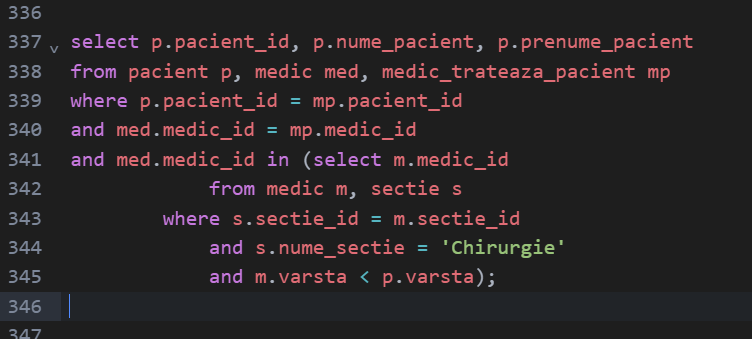
and med.medic\_id in (select m.medic\_id

from medic m, sectie s

where s.sectie\_id = m.sectie\_id

and s.nume\_sectie = 'Chirurgie'

and m.varsta < p.varsta);





--medicii care au tratat cel putin 2 pacienti si au realizat cel putin o operatie care a durat mai mult de 3 ore

--grupări de date cu subcereri nesincronizate in care intervin cel putin 3 tabele, funcții grup, filtrare la nivel de grupuri

select medic\_id, nume\_medic, prenume\_medic

from medic

where medic\_id in (select m.medic\_id

from medic m, medic\_trateaza\_pacient mp, operatie o

where m.medic\_id = mp.medic\_id

and m.medic\_id = o.medic\_id

group by m.medic\_id

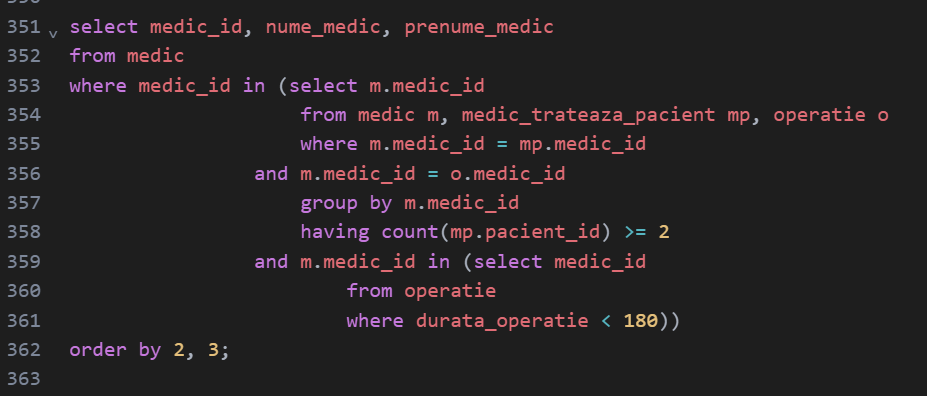
having count(mp.pacient\_id) >= 2

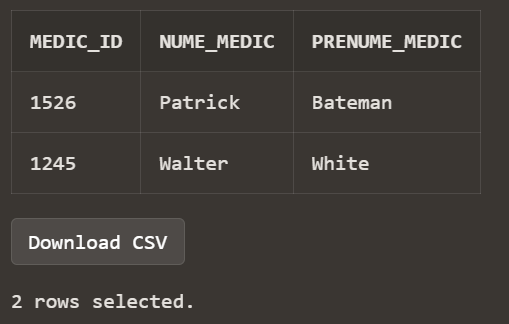
and m.medic\_id in (select medic\_id

from operatie

where durata\_operatie < 180))

order by 2, 3;





--sa se selecteze id-ul, numele concatenat cu prenumele, varsta si daca numele incepe cu A sau cu B din tabelul PARAMEDIC

--decode, nvl, functii cu date, functii cu siruri de caractere, case

select paramedic\_id, concat(nume\_paramedic,concat(' ', prenume\_paramedic)) nume, nvl(data\_nasterii, '01-JAN-1970') data\_nasterii,

case

when extract(month from data\_nasterii) < extract(month from sysdate) then to\_char(sysdate, 'yyyy') - to\_char(data\_nasterii, 'yyyy')

when extract(month from data\_nasterii) > extract(month from sysdate) then to\_char(sysdate, 'yyyy') - to\_char(data\_nasterii, 'yyyy') - 1

when extract(month from data\_nasterii) = extract(month from sysdate) then case

when extract(day from data\_nasterii) <= extract(day from sysdate) then to\_char(sysdate, 'yyyy') - to\_char(data\_nasterii, 'yyyy')

when extract(day from data\_nasterii) > extract(day from sysdate) then to\_char(sysdate, 'yyyy') - to\_char(data\_nasterii, 'yyyy') - 1

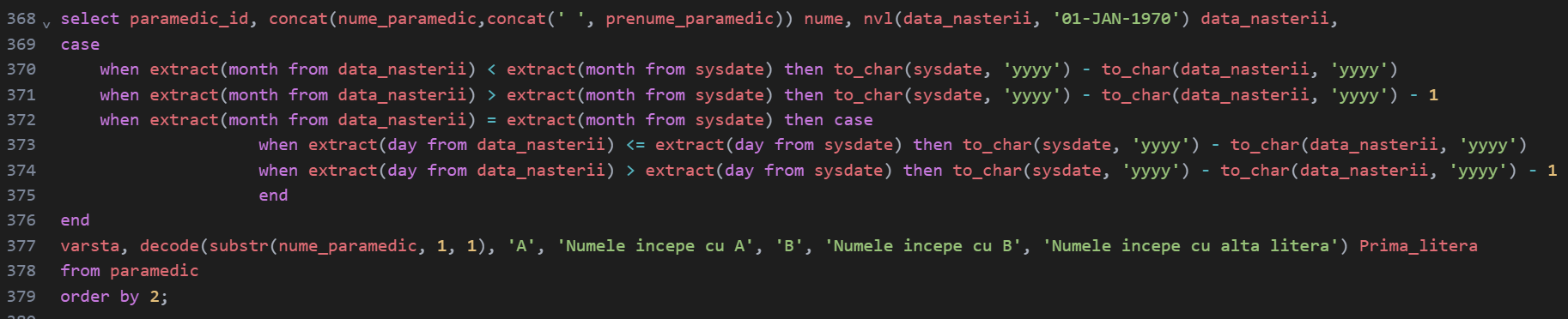
end

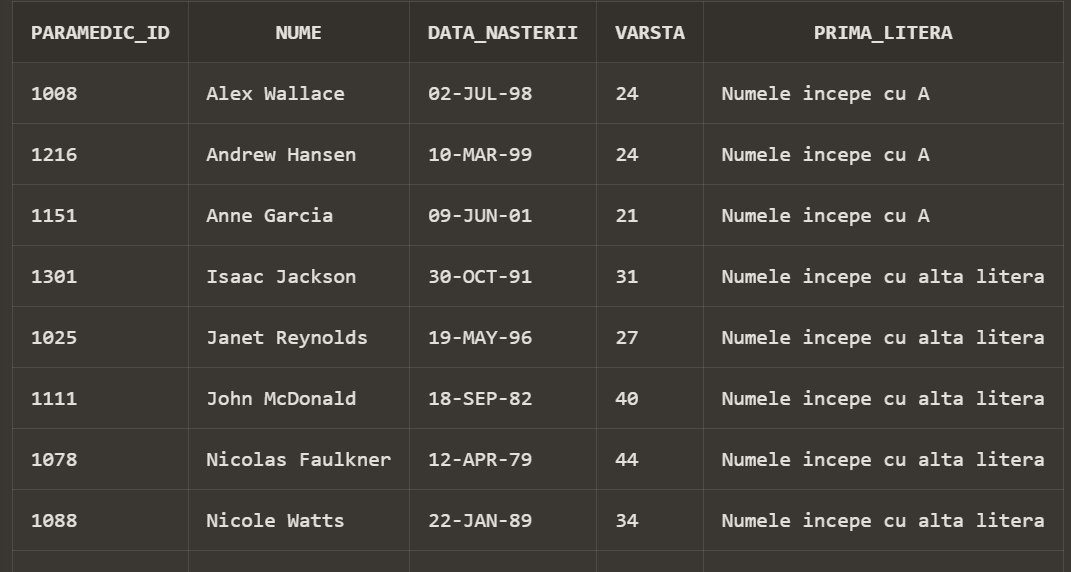
end

varsta, decode(substr(nume\_paramedic, 1, 1), 'A', 'Numele incepe cu A', 'B', 'Numele incepe cu B', 'Numele incepe cu alta litera') Prima\_litera

from paramedic

order by 2;





--sa se selecteze pacientii care au grupa de sange 'A', medicii care ii trateaza si motivul internarii

--clauza with

with aux as (select m.medic\_id

from pacient p, medic m, medic\_trateaza\_pacient mp

where p.pacient\_id = mp.pacient\_id

and m.medic\_id = mp.medic\_id

and p.grupa\_sange = 'A')

select m.medic\_id, m.nume\_medic, m.prenume\_medic, p.pacient\_id, p.nume\_pacient, p.prenume\_pacient, f.motiv\_internare

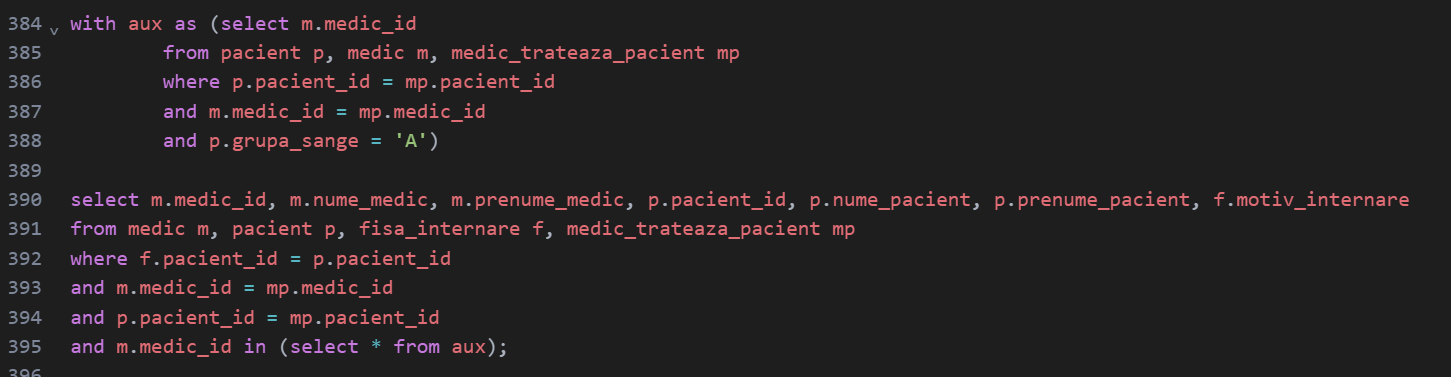
from medic m, pacient p, fisa\_internare f, medic\_trateaza\_pacient mp

where f.pacient\_id = p.pacient\_id

and m.medic\_id = mp.medic\_id

and p.pacient\_id = mp.pacient\_id

and m.medic\_id in (select \* from aux);





1. **Implementarea a 3 operații de actualizare și suprimare**

--sa se stearga paramedicii care nu au adus niciun pacient la spital

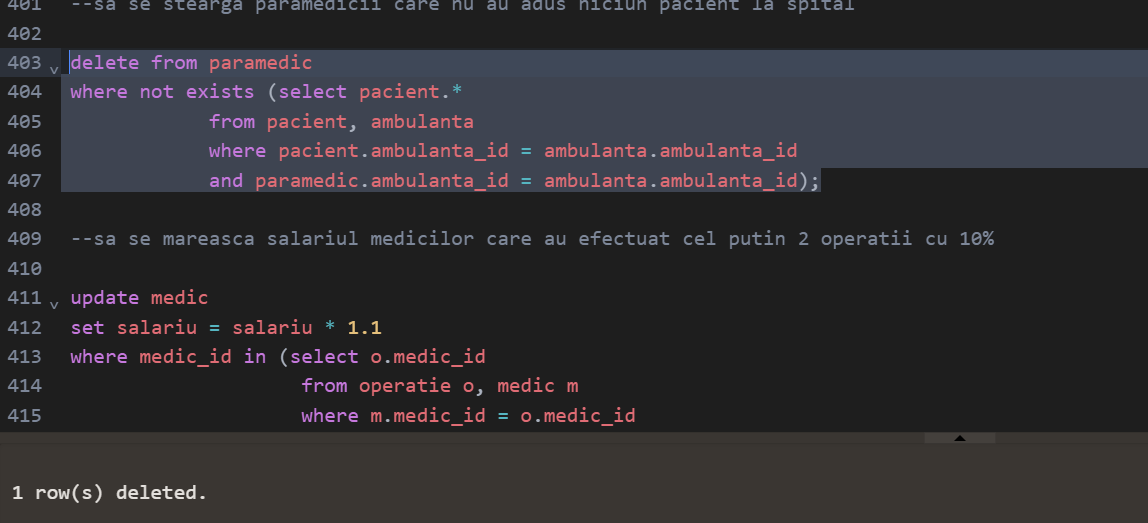
delete from paramedic

where not exists (select pacient.\*

from pacient, ambulanta

where pacient.ambulanta\_id = ambulanta.ambulanta\_id

and paramedic.ambulanta\_id = ambulanta.ambulanta\_id);



--sa se mareasca salariul medicilor care au efectuat cel putin 2 operatii cu 10%

update medic

set salariu = salariu \* 1.1

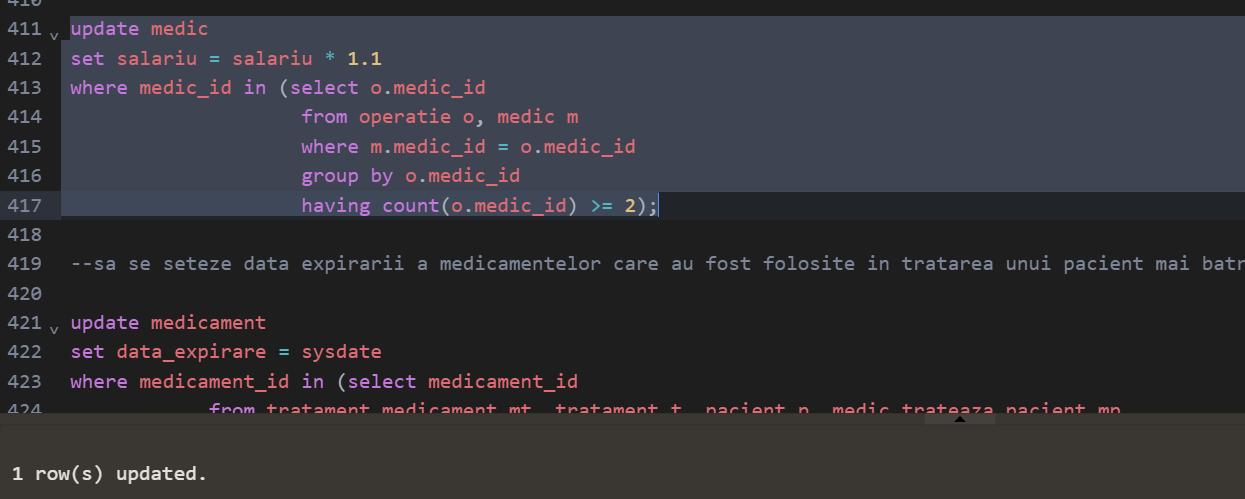
where medic\_id in (select o.medic\_id

from operatie o, medic m

where m.medic\_id = o.medic\_id

group by o.medic\_id

having count(o.medic\_id) >= 2);



--sa se seteze data expirarii a medicamentelor care au fost folosite in tratarea unui pacient mai batran de 40 ani la data curenta

update medicament

set data\_expirare = sysdate

where medicament\_id in (select medicament\_id

from tratament\_medicament mt, tratament t, pacient p, medic\_trateaza\_pacient mp

where t.tratament\_id = mt.tratament\_id

and mp.tratament\_id = t.tratament\_id

and mp.pacient\_id = p.pacient\_id

and p.varsta > 40);

