

Gestiunea unui spital

Pasare Roxana-Francisca

Grupa 143

Studiu de caz (Exercitiul 1)

Tema aleasa la cursul “Baze de date” este Gestiunea unui spital. Intr-un spital, evidenta pacientilor este foarte importanta, lucru ce face ca baza de date a unei astfel de institutii sa conteze extrem de mult.

Erorile dintr-o baza de date pot duce la pierderea informatilor despre pacienti, lucru ce poate sa fie fatal in cazul unor accidentari grave sau afectiuni rare.

Baza de date contine informatii cu privire la spital, departamentele si saloanele acestuia, cat si la doctorii care lucreaza in cadrul spitalului, ce urmaresc evolutia pacientilor , dar si starea pacientului rezultata din evaluarea realizata de catre medic intr-un salon.

Fiecare pacient este repartizat la un department, intr-o sectie si un salon, participand la anumite studii realizate de doctori care au posibilitatea de a lucra la mai multe spitale. Pacientii se pot inscrie la centrele de donare in cadru unui spital.

Evolutia pacientilor este in functie de etiologie, fiind urmarita in incinta diferitor saloane in cadrul spitalului.

Un doctor are contracte cu spitalele in care lucreaza, acesta putand sa aiba una sau mai multe specializari in cadrul unui spital.

O analiza se poate realiza in mai multe moduri: prin sange, prin urina si prin saliva. Rezultatul acesteia difera in functie de afectiunile pacientului. Analizele unui pacient sunt considerate bune daca peste 95% din rezultate au valori normale.

Reguli de functionare (exercitiul 2)

Baza de date respecta urmatoarele reguli:

- Un doctor este angajat la cel putin un spital, iar un spital angajeaza cel putin un doctor.
- Un spital are cel putin un departament.
- Un departament are cel putin o sectie.
- O sectie are in componenta cel putin o etiologie.
- Intr-o etiologie se pot incadra mai multi pacienti, iar pacienti pot avea mai multe patologii.
- Un pacient poate sa fie membru in cadrul unui centru de donare.
- Un centru de donare apartine doar unui spital.
- Un pacient efectueaza una sau mai multe analize, o analiza este facuta unuia sau de mai multor pacienti.
- O analiza este efectuata in functie de evolutia pacientului, evolutia pacientului presupune efectuarea a mai multor analize.
- O analiza este efectuata de unul sau mai multi doctori. O analiza apartine uneia sau mai multor etiologii. O analiza se poate realiza in unul sau mai multe saloane.
- Evolutia pacientului poate sa fie mai rea, la fel sau mai buna.
- O analiza poate sa fie efectuata prin sange, urina sau saliva.
- Un centru de donare are cel putin un membru
- O etiologie este reprezentata din cel putin un pacient.

Entitati (exercitiul 3)

Toate entitatile de mai jos sunt independente, cu exceptia entitatilor FISA MEDICALA, DESFASURATOR, CONTRACT, DIAGNOSTIC care sunt dependente.

SPITAL -> institutia in care se realizeaza procedurile si analizele unui pacient. Cheia primara este id_spital.

PACIENT->persoana internata intr-un spital. Un pacient poate sa fie membru in cadrul centrului de donare al spitalului in care este internat. Cheie primara id_pacient.

CENTRU DE DONARE->organizatie in cadrul careia pacientii pot sa fie membrii. Cheie primara este id_donare.

ANALIZE->forma de evaluare a pacientului realizata prin sange, urina sau saliva prin care pacientul primeste un feedback despre starea sa de sanatate. Analizele unui pacient sunt considerate bune daca peste 95% din rezultate au valori normale. Cheie primara este id_analize.

EVOLUTIE PACIENT->stare de sanatate a unui pacient. Evolutia pacientului poate sa fie mai rea, la fel sau mai buna. Cheie primara id_evolutie_pacient.

SALON->locul de stabilire a evolutiei pacientului. Intr-o sala se pot realize mai multe evaluari, la intervale orare diferite, Saloanele pot sa fie de doi, de patru sau de sase pacienti. Cheie primara id_salon.

DEPARTAMENT->modalitate de distribuire a pacientilor in cadrul spitalului. In functie de acesta, intre doi pacienti apartinand a doua sectii diferite pot sa difere doctorii, saloanele. Cheie primara id_departament.

SECTIE->modalitate de distribuire a pacientilor in cadrul unui departament. Pacientii aflati in aceeasi sectie sunt ingrijiti de aceeasi doctori. Cheie primara id_sectie.

ETIOLOGIE->modalitate de distribuire a pacientilor. Pacientii aflati la aceeasi etiologie au aceeasi doctori. Cheie primara id_etiologie.

DOCTOR->persoana angajata intr-un spital, care opereaza si evalueaza pacientii. Doctorul poate avea specializari diferite si sa fie angajat la mai multe spitale. Doctorul poate sa recolteze atat analize de sange cat si de urina si saliva. Cheie primara id_doctor.

FISA_MEDICALA->cuprinde informatii despre etiologiile din care face parte pacientul. Cheia primara compusa este formata din id_etiologie si id_pacient.

DESFASURATOR->programul unui spital, cuprinzand intervalele de stabilire a etiologiilor pacientilor realizate de catre doctori in saloane cat si evaluarea evolutiei pacientului. Cheia compusa este formata din id_etiologie, id_doctor, id_salon, si id_evolutie_pacient.

Baze de date
Pasare Roxana-Francisca
Grupa 143

CONTRACT->act care cuprinde informatii despre durata angajarii si salariul unui doctor in cadrul unui spital. Cum un doctor poate lucra la mai multe spitale, salariul poate sa fie diferit in functie de spital, Cheia primara compusa este alcatuita din id_doctor si id_spital.

DIAGNOSTIC->reprezinta starea de sanatate a pacientului rezultata in urma analizelor. Cheia primara este alcatuita din id_analiza si id_pacient.

Relatii (exercitiul 4)

DOCTOR este angajat la SPITAL->relatie de tipul many-to-many care leaga entitatile DOCTOR si SPITAL, evidentiind legatura dintre acestea (la ce spital este angajat un anumit doctor). Aceasta relatie are cardinalitate minima 1:1 (un spital trebuie sa aiba cel putin un doctor angajat, iar un doctor poate sa fie angajat la cel putin un spital) si cardinalitate maxima m:n (un spital are mai multi doctori angajati, iar un doctor poate sa lucreze la mai multe spitale).

SPITALUL are CENTRU DE DONARE-> relatie de tipul one-to-one care leaga entitatile SPITAL si CENTRU DE DONARE, evidentiind legatura dintre acestea (spitalul are un centru de donare). Relatia are cardinalitate minima si maxima 1:1.

PACIENTUL este membru al CENTRULUI de DONARE->relatie de tipul one-to-many care leaga entitatile PACIENT si CENTRU DE DONARE evidentiind legatura dintre acestea (ce pacienti sunt inscrisi in programul centrului de donare). Ea are cardinalitate minima 1:1 si cardinalitate maxima 1:n.

SPITAL are DEPARTAMENT->relatie de tipul one-to-many care leaga entitatile SPITAL si DEPARTAMENT, evidentiind legatura dintre acestea (un spital are unul sau mai multe departamente). Relatia are cardinalitatea minima 1:1 si cardinalitatea maxima 1:n.

DEPARTAMENT are SECTIE-> relatie de tipul one-to-many care leaga entitatile DEPARTAMENT si SECTIE, evidentiind legatura dintre acestea (sectiile din spital care fac parte din anumite departamente). Relatia are cardinalitate minima 1:1 si cardinalitate maxima 1:n.

SECTIA este alcatuita ETIOLOGII-> relatie de tipul one-to-many care leaga entitatile SECTIA si ETIOLOGII, evidentiind legatura dintre acestea (etiologiile care alcatuiesc anumite sectii din cadrul unui departament). Ea are cardinalitate minima 1:1 si cardinalitate maxima 1:n.

PACIENT efectueaza ANALIZE-> relatie de tip many-to-many dintre entitatile PACIENT si ANALIZE (ce pacient face ce analize). Relatia are cardinalitatea minima 1:1 si cardinalitate m:n.

DOCTOR analizeaza EVOLUTIE PACIENT repartizati pe ETIOLOGIE in SALON->relatie de tip trei ce leaga entitatile DOCTOR, EVOLUTIE PACIENT, ETIOLOGIE, SALON, evidentiind cine a analizat evolutia unui pacient, in cadrul carei etiologii si in ce salon. Denumirea acestei relatii va fi DESFASURATOR.

Atribute(exercitiul 5)

Entitatea **SPITAL** este o entitate independent care are ca atribute:

cod_spital-variabila de tip intreg, de lungime maxima 5, reprezentand codul spitalului

denumire-variabila de tip character, de lungime maxim 100, reprezentand denumire
spitalului

adresa-variabila de tip character, de lungime maxim 100, reprezentand adresa spitalului

telefon-variabila de tip character, lungime maxima 20, reprezentand numarul de telefon
al spitalului

mail-variabila de tip character, lungime maxima 50, reprezentand mail-ul spitalului

cod_postal-variabila de tip character, lungime maxima 6, reprezentand codul postal
al spitalului

cod_spital	adresa	denumire	telefon	mail	cod_postal
PK	NOT NULL	NOT NULL	NOT NULL	UNIQUE	-

Spital
Cod_spital#
Denumire
Adresa
Telefon
Mail
Cod_postal

Entitatea **DEPARTAMENT** are ca atribute:

cod_departament-variabila de tip intreg, lungime maxima 5, reprezentand
codul departamentului

cod_spital- variabila de tip intreg, de lungime maxima 5, reprezentand codul spitalului
din care face parte departamentul. Atribut trebuie sa corespunda la o valoare
a cheii primare din tabelul facultate

denumire-variabila de tip caracter, de lungime maxim 100, reprezentand denumire
departamentului

cod_departament	cod_spital	denumire
PK	FK	NOT NULL

DEPARTAMENT
Cod_departament#
Denumire
Cod_spital

Entitatea **SECTIE** are ca attribute:

cod_sectie-variabila de tip intreg, lungime maxima 5, reprezentand

codul sectiei

cod_departament- variabila de tip intreg, de lungime maxima 5, reprezentand codul

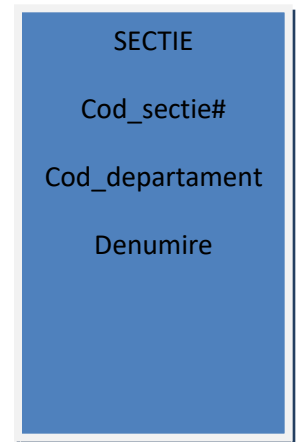
departamentului din care face parte sectia. Atribut trebuie sa

corespunda la o valoare a cheii primare din tabelul

DEPARTAMENT

denumire-variabila de tip caracter, de lungime maxim 100, reprezentand denumire

sectiei



cod sectie	cod departament	denumire
PK	FK	NOT NULL

Entitatea **ETIOLOGIE** are ca attribute:

cod_etiologie-variabila de tip caracter, lungime maxima 5, reprezentand

codul etiologiei.

cod_sectie- variabila de tip intreg, de lungime maxima 5, reprezentand codul

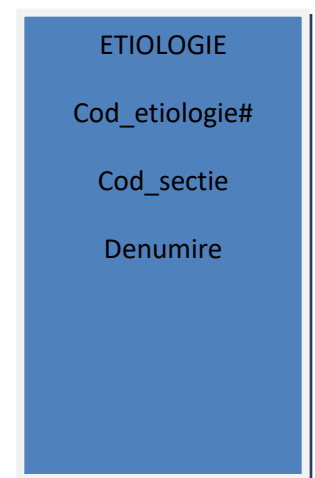
sectiei din care face parte etiologia. Atribut trebuie sa

corespunda la o valoare a cheii primare din tabelul

SECTIE.

denumire-variabila de tip caracter, de lungime maxim 100, reprezentand denumirea

etiologiei



cod etiologie	cod sectie	denumire
PK	FK	Not NULL

Entitate PACIENT are ca atribut:

cod_pacient-variabila de tip intreg, de lungime maxima 5, reprezentand codul unui pacient

nume-variabila de tip character, de lungime maxim 50, reprezentand numele de familie al pacientului

prenume-variabila de tip character, de lungime maxim 50, reprezentand prenumele pacientului

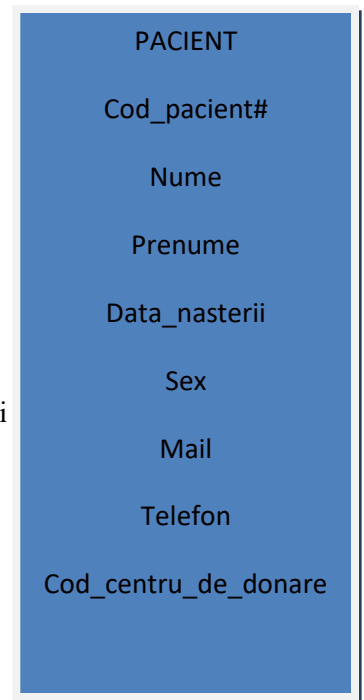
data_nasterii-variabila de tip data calanderistica, reprezentand data nasterii a unui pacient

sex-variabila de tip character, de lungime maxima 10, reprezentand sexul pacientului

mail-variabila de tip character, lungime maxima 50, reprezentand mail-ul pacientului

telefon-variabila de tip character, lungime maxima 50, reprezentand numarul de telefon al pacientului

cod_centru_de_donare-variabila de tip intreg, de lungime maxima 5. care reprezinta codul centrului de donare de care apartine pacientul. Atributul trebuie sa corespunda la o valoare a cheii primare din tabelul CENTU DE DONARE



Cod_student	Nume	Prenume	Data_nasterii	mail	Sex	Cod_centru_de_donare
PK	NOT NULL	NOT NULL	Not NULL	UNIQUE	NOT NULL	FK

Entitatea **CENTRU DE DONARE** are ca atribute:

cod_centru_de_donare- variabila de tip intreg, de lungime maxima 5, reprezentand

codul unui centru de donare

denumire- variabila de tip character, de lungime maxim 100, reprezentand

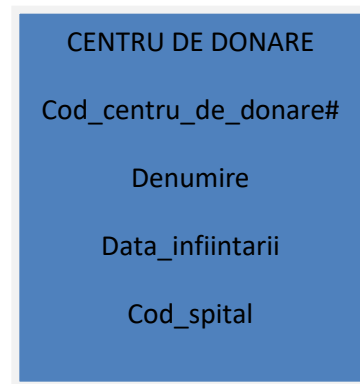
denumirea centrului de donare

data_infiintarii- variabila de tip data calanderistica, reprezentand data infiintarii

centrului de donare

cod_spital- variabila de tip intreg, de lungime maxima 5, care reprezinta codul spitalului de care apartine centrului de donare. Atributul trebuie sa corespunda la o valoare a cheii primare din tabelul SPITAL.

Cod centru de donare	denumire	Data infintarii	Cod spital
PK	NOT NULL	NOT NULL	FK



Entitatea **ANALIZA** are ca atribut:

cod_analiza-variabila de tip intreg, lungime maxima 5, reprezentand codul

unei analize

forma-variabila de tip character, luand valorile “Sange”, ”Urina” sau “Saliva”,

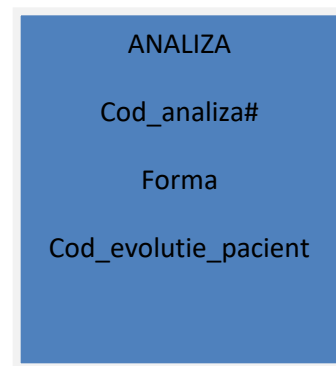
reprezentand modalitatea de evaluare a pacientului

cod_evolutie_pacient-variabila de tip intreg, lungime maxima 5, reprezentand

codul evolutiei pacientul la care este efectuata analiza.

Atributul trebuie sa corespunda la o valoare a cheii primare din tabelul EVOLUTIE_PACIENT.

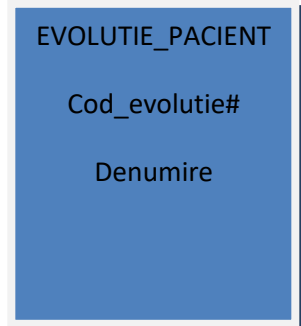
Cod analiza	Cod evolutie pacient	Forma
PK	FK	Not NULL



Entitatea **EVOLUTIE PACIENT** are ca atribute:

cod_evolutie-variabila de tip intreg, lungime maxim 10, reprezentand
codul evolutiei pacientului

denumire-variabila de tip caracter, lungime maxima 50, reprezentand
evolutia pacientului(rea/la fel/mai buna)



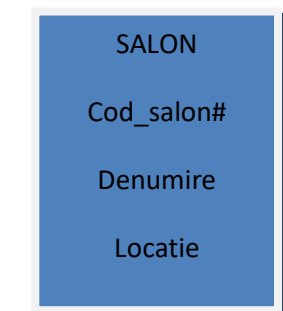
Cod_evolutie	Denumire
PK	NOT NULL

Entitatea **SALON** are ca atribute:

cod_salon-variabila de tip intreg, lungime maxim 10, reprezentand codul
salonului

denumire-variabila de tip caracter, lungime maxima 50, reprezentand
denumirea salonului

locatie-variabila de tip caracter, lungime maxima 50, reprezentand locatia
salonului



Cod_salon#	Locatie	Denumire
PK	NOT NULL	NOT NULL

Entitatea **DOCTOR** are ca attribute:

cod_doctor-variabila de tip intreg, lungime maxima 10, reprezentand

codul profesorului

nume-variabila de tip caracter, lungime maxima 50, reprezentand

numele doctorului

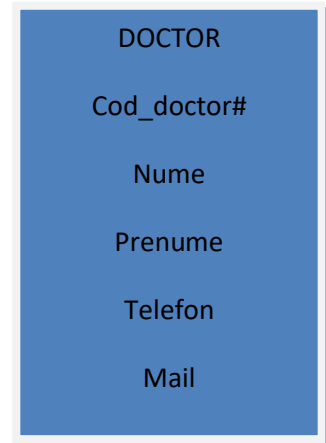
prenume-variabila de tip caracter, lungime maxima 50, reprezentand

prenumele doctorului

telefon-variabila de tip caracter, lungime maxima 20, reprezentand

numarul de telefon al doctorului

mail-variabila de tip caracter, lungime maxima 50, reprezentand mail-ul



Cod_doctor#	Nume	Prenume	Telefon	Mail
PK	NOT NULL	NOT NULL	NOT NULL	UNIQUE

Relatia **PACIENT** face **ANALIZE**(Entitatea DIAGNOSTIC) are ca attribute:

rezultat-variabila de tip real, de lungime maxim 3, care reprezinta rezultatul analizei

cod_pacient-variabila de tip intreg, de lungime maxima 5, care reprezinta codul pacientului caruia ii apartine analiza. Atributul este corespondent la o valoare a cheii primare din tabelul PACIENT.

cod_analiza-variabila de tip intreg, de lungime maxim 5, care reprezinta codul analizei la care s-a acordat rezultatul. Atributul este corespondent la o valoare a cheii primare din tabelul ANALIZA.



Cod_pacient	Cod_spital	Rezultat
PK	PK	NOT NULL

Relatia **DOCTOR_este_angajat_la_SPITAL**(Entitatea **CONTRACT**) are ca atribute:

salariu-variabila de tip intreg, reprezentand salariul

cod_doctor-variabila de tip intreg, lungime maxima 5, reprezentand
codul doctorului care lucreaza la spitalul respectiv. Atributul
este corespunda la o valoare a cheii primare din tabelul
DOCTOR.

data_inceput-variabila de tip data calendarisica, reprezentand data angajarii
unui doctor la spital.

cod_spital-variabila de tip intreg, lungime maxima 5, care reprezinta spitalul
la care lucreaza doctorul. Atributul este corespulent la o valoare
a cheii primare din tabelul SPITAL.

CONTRACT

Cod_doctor#

Cod_spital

Salariu

Data_inceput

Cod_doctor#	Cod_spital#	Spital	Data_inceput
PK	PK	NOT NULL	NOT NULL

Relatia **DOCTOR_analizeaza_EVOLUTIA_PACIENTULUI_grupati_pe_ETIOLOGIE_in_SALON**(Entitatea **DESFASURATOR**) are ca atribute:

cod_doctor-variabila de tip intreg, lungime maxima 5, care reprezinta
codul doctorul care analizeaza evolutie, Atributul are
corespulent o valoare a cheii primare din tabelul DOCTOR.

cod_evolutie-variabila de tip intreg, lungime maxima 5, care reprezinta
codul evolutiei, Atributul are corespulent o valoare a

DESFASURATOR

Cod_doctor#

Cod_evolutie#

Cod_salon#

Cod_etiologie#

cheii primare din tabelul EVOLUTIE_PACIENT.

cod_salon-variabila de tip intreg, lungime maxima 5, care reprezinta

codul salonului unde se realizeaza analiza, Atributul are corespondent o valoare a

cheii primare din tabelul SALON.

cod_etiologie-variabila de tip intreg, lungime maxima 5, care reprezinta

codul etiologiei supuse la analiza, Atributul are corespondent o valoare a

cheii primare din tabelul ETIOLOGIE.

Cod_doctor#	Cod_evolutie#	Cod_salon#	Cod_etiologie#
PK	PK	PK	PK

Relatia ETIOLOGIA_ are _PACIENTI(Entitatea FISA_MEDICALA) are ca atribute:

data_intemeierii-variabila de tip data calendaristica, care reprezinta data

intemeierii fisei medicale a unui pacient in cadrul unui etiologii.

cod_etiologie-variabila de tip intreg, lungime maxima 5, care reprezinta

codul etiologiei din care face parte pacientul. Atributul are

corespondent o valoare a cheii primare din grupa ETIOLOGIE.

cod_pacient-variabila de tip intreg, lungime maxima 5, care reprezinta codul

studentului, Atributul are corespondent o valoare a cheii primare din tabelul PACIENT

FISA_MEDICALA
Cod_etiologie#
Cod_pacient#
Data_intemeierii

Cod_grupa#	Cod_student#	Data_intemierii
PK	PK	NOT NULL

Diagrama E/R(exercitiul 6)

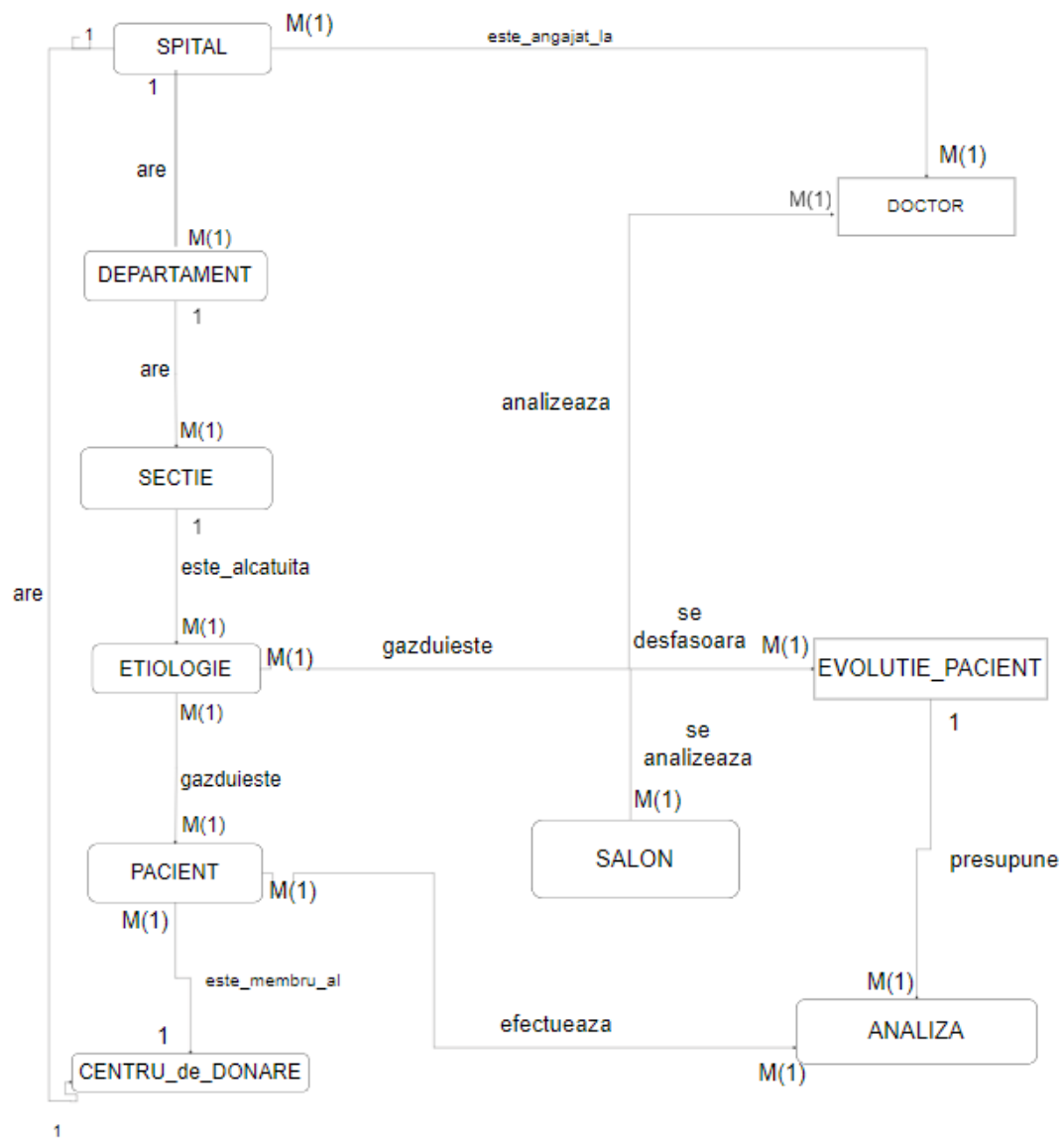
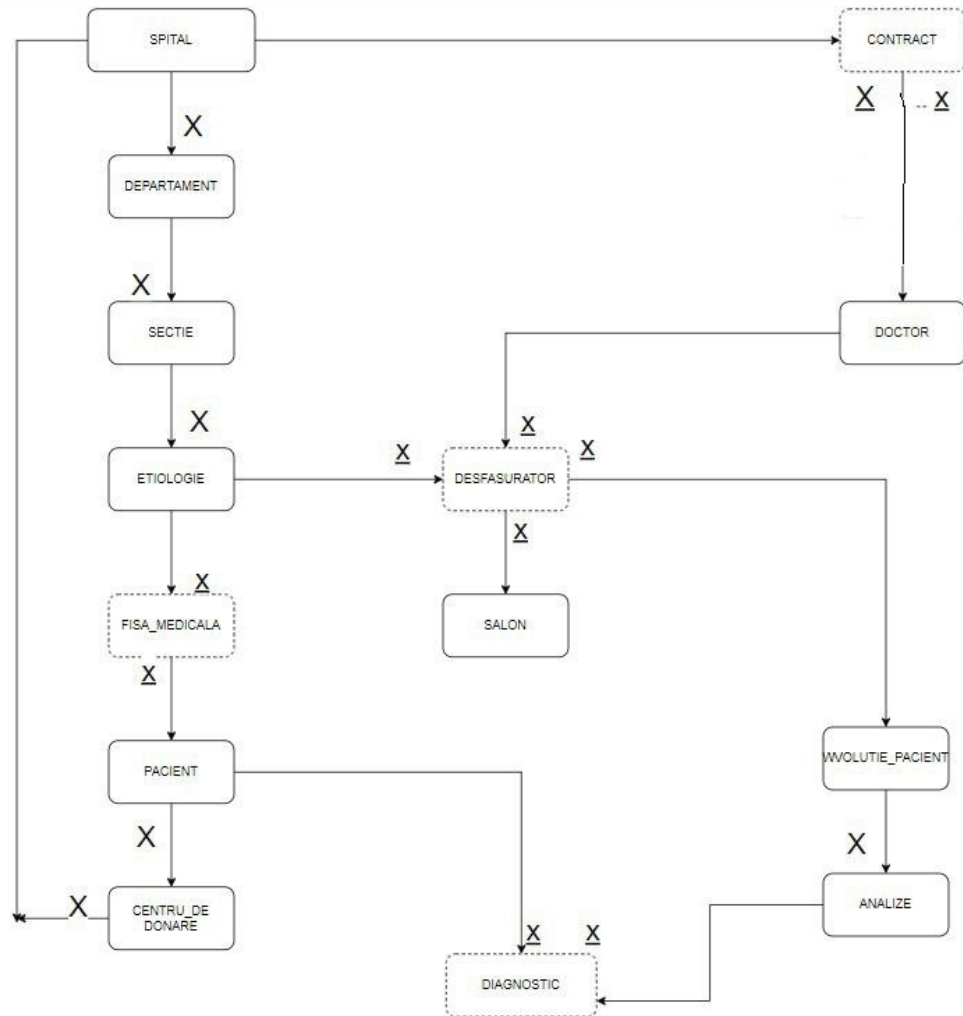


Diagrama conceptuala(exercitiul 7)



Scheme relationale(exercitiul 8)

CENTRU_DE_DONARE(cod_centru#, denumire, data_infintarii, cod_spital#)

CONTRACT(cod_doctor#, cod_spital#, salariu)

EVOLUTIE_PACIENT(cod_analize#, forma, cod_evolutie#)

SPITAL(cod_spital#, denumire, adresa, telefon, mail, cod_postal)

ETIOLOGIE(cod_etiologie#, denumire, cod_sectie#)

FISA_MEDICALA(data_intemeierii, cod_pacient#, cod_etiologie#)

DESFASURATOR(cod_etiologie#, cod_evolutie#, cod_doctor#, cod_salon#)

DOCTOR(cod_doctor#, nume, prenume, data_nasterii, telefon, mail)

DIAGNOSTIC(procent, cod_pacient#, cod_analize#)

SALON(cod_salon#, denumire, locatie)

DEPARTAMENT(cod_departament#, denumire, cod_spital#)

SECTIE(cod_sectie#, denumire, cod_departament#)

PACIENT(cod_pacient#, nume, prenume, data_nasterii, sex, telefon, mail, cod_centru)

Normalizare si Denormalizare(exercitiul 9&18)

1.Forma normala 1(FN1)

O relatie se afla in FN1 daca fiecarui atribut care are o compunere ii corespunde o valoare indivizibila.

Forma normala 1 este si cea care impune faptul ca fiecare inregistrare sa fie definita incat sa fie identificata unic prin intermediul unei chei primare.

Luam entitatile SPITAL si DEPARTAMENT.In cadrul unui spital, denumirea departamentelor nu se repeta, dar in cadrul unui lant de spitale acest lucru nu este garantat.

1.Mai multe valori semnifica acelasi camp

SPITAL	DEPARTAMENT
Spital de adulti	Geriatric, Chirurgie vasculara
Spital de copii	Neonatalogie, Chirurgie vasculara



SPITAL	DEPARTAMENT
Spital de adulti	Geriatric
Spital de adulti	Chirurgie vasculara
Spital de copii	Neonatalogie
Spital de copii	Chirurgie vasculara



ID	SPITAL	DEPARTAMENT
1	Spital de adulti	Geriatric
2	Spital de adulti	Chirurgie vasculara

3	Spital de copii	Neonatologie
4	Spital de copii	Chirurgie vasculara

2.Mai multe coloane reprezinta acelasi tip de date

SPITAL	DEPARTAMENT	DEPARTAMENT(1)
Spital de adulti	Chirurgie vasculara	Geriatric
Spital de copii	Chirurgie vasculara	Neonatologie



SPITAL	DEPARTAMENT
Spital de adulti	Chirurgie vasculara
Spital de copii	Chirurgie vasculara
Spital de adulti	Geriatric
Spital de copii	Neonatologie



ID	SPITAL	DEPARTAMENT
1	Spital de adulti	Chirurgie vasculara
2	Spital de adulti	Geriatric
3	Spital de copii	Chirurgie vasculara
4	Spital de copii	Neonatologie

Pentru asigurarea unicitatii unei inregistrari, se va utiliza cheia primara a acesteia. In exemplul de mai sus introducerea coloanei aditionale(ID) de tip intreg putem asigura unicitatea fiecarei inregistrari.

2. Forma normala 2(FN2)

O relatie se afla in a doua forma normala daca si numai daca aceasta relatie este deja in FN1 si fiecare atribut care nu este cheie primara este dependent de intreaga cheie primara.

FN2 interzice existenta dependentelor dunctionale partiale in cadrul relatiei.

Daca unul sau mai multe elemente sunt dependete functional numai de o parte a cheii primare, atunci ele trebuie sa fie separate in tabele diferite. Daca tabela are o cheie primara formata de numai un atribut, atunci ea este automat in FN2.

Exemplificare pentru diagrama CONTACT:

Cod_doctor#	Nume_doctor	Cod_spital#	Data_inceput	Salariu
D1	N1	S1	12-09-2009	5300
D2	N2	S2	17-02-2001	6500
D2	N2	S3	28-10-2013	7800

Un doctor poate sa lucreze la mai multe spitale, iar un spital are angajati mai mult doctori.

Relatia noastra este in FN1, deoarece avem identificator unic pentru toate intrarile in tabel.

Fiecare atribut care nu este cheie primara(nu participa la cheia primara) este dependent de intreaga cheie primara. Cheia primara a acestui tabel este (Cod_doctor si Cod_spital). Se observa dependenta directa dintre Nume_doctor si cod_doctor insemnand ca Nume_doctor depinde direct doar de o parte a cheii primare, si anume, doar de cod_doctor->relatia nu se afla in FN2.

Asta inseamna ca:

cod_doctor#->nume->cod_doctor determina nume_doctor

(cod_doctor#, cod_spital#)->(data_inceput, salariu)

Pentru a avea relatia in FN2, numele doctorului trebuie sa fie doar in entitatea doctor (se aplica regula Casey Delobel).

Cod_doctor#	Nume	Cod_spital	Data_inceput	Salariu
D1	N1	S1	12-09-2009	5300
D2	N2	S2	17-02-2001	6500
D2	N2	S3	28-10-2013	7800



Cod_doctor#	Cod_spital#	Data_inceput	Salariu
D1	N1	12-09-2009	5300
D2	N2	17-02-2001	6500
D2	N2	28-10-2013	7800

3. Forma Normala 3(FN3)

O relatie este in a treia forma normala daca si numai daca este in FN2 si fiecare atribut care nu este cheie depinde direct de cheia primara.

Initial tabelul PACIENT ar avea urmatoarea forma:

Cod_pacient#	Nume	Prenume	Data_nasterii	Sex	Telefon	Mail	Denumire_centru	Data_infiintarii
P1	N1	Prn1	04-05-2001	m	T1	M1	G1	D1
P2	N2	Prn2	29-01-2001	f	T2	M2	G2	D2
P3	N3	Prn3	18-07-2000	f	T3	M3	G3	D3

Se poate observa ca atribut Data_infiintarii depinde de atributul Denumire_centru care depinde la randul sau de cheia primara Cod_pacient, altfel nefiind in FN3.

Pentru a aduce in FN3, separ attributele din tabelul PACIENT, aparand astfel tabela Centru_donare.

Astfel, inlocuiesc aceste attribute cu cheia straina cod_centru (aparand astfel o relatie one-to-many intre CENTR_DONARE si PACIENT) pentru a determina mai usor din ce centru face parte un pacient.

Cod_pacient#	Nume	Prenume	Data_nasterii	Sex	Telefon	Cod_centru	Mail
P1	N1	Prn1	04-05-2001	m	T1	NULL	M1
P2	N2	Prn2	29-01-2001	f	T2	C1	M2
P3	N3	Prn3	18-07-2000	f	T3	C2	M3

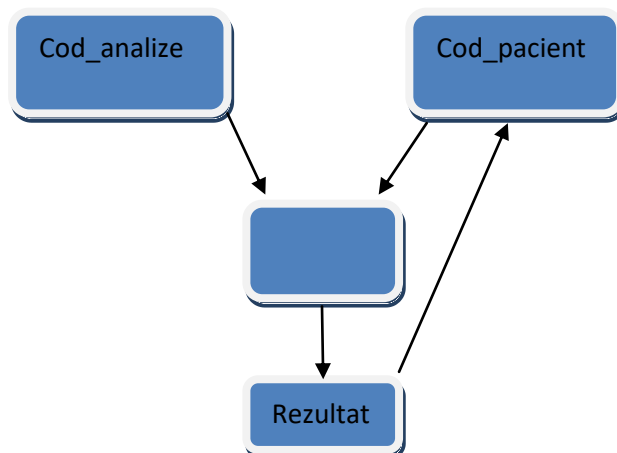
Cod_centru	Denumire_centru	Data_infiintarii	Cod_spital
1	Cntr1	D1	S1
2	Cntr2	D2	S2
3	Cntr3	D3	S3

4. Forma normala Boyce-Codd(BCNF)

Forma normala Boyce-Codd se bazeaza pe dependentele functionale care iau in considerare toate cheile candidat dintr-o relatie.



Pentru relatiile cu o singura cheie candidat, formale FN3 si BCNF sunt echivalente (Spital, Evolutie_pacient etc.)

Vom lua relatia relatie DIAGNOSTIC(cod_analize#, cod_pacient#, rezultat)



Regula Casey Delobel pentru DIAGNOSTIC(Cod_pacient#, Cod_analize#, Rezultate) ne arata ca Rezultat determina functional pe Cod_pacient.

Rezultat->Cod_pacient

 DIAGNOSTIC_1(cod_analiza, rezultat)
 DIAGNOSTIC_2(rezultat, cod_pacient)

5. Forma normala 4(FN4)

FN4 elimina redundantele datorate relatiilor m:n, adica datorate dependentei multiple.

O relatie este in a patra forma normala daca si numai daca este in BCNF si nu contine relatii m:n independente.

Se ia entitatea SALON(cod_salon#, denumire, locatie) si presupun ca un salon poate avea mai multe denumiri si mai multe locatii. De exemplu, o aripa in care se poate intra prin mai multe locuri si fiecare intrare e numele salonului, plus intrarea (A, B, C etc) si locatiile sunt date de holurile de unde se poate intra.

Cod_salon#-> Denumire

Cod_salon#-> Locatie

Relatia SALON este in BCNF. Pentru a aduce relatia in FN4 o vom descompune pe proiectie in doua relatii:

SALON1(Cod_salon#, Denumire)

SALON2(Cod_salon#, locatie)

->SALON=JOIN(SALON1, SALON2)

6. Forma normala 5(FN5)

O relatie R este in FN5 (numita si forma normala proiectie-uniune) daca si numai daca orice dependenta de uniune a lui R este o consecinta a unui chei candidat a lui R.

Orice relatie care este in FN5 este si in FN4, deoarece fiecare dependenta multivaloare poate fi privita ca un caz particular de dependenta de uniune. Orice relatie poate fi descompusa fara pierderi la uniune intr-o multime de relatii care sunt in FN5.

Pentru a preciza daca o relatie este in FN5, este suficient sa cunoastem cheile candidate si toate dependentele de uniune din R.

Aducerea in FN5 presupune eliminarea join dependentelor.

Voi lua relatia DOCTOR si presupun ca am o join dependenta in aceasta. Voi lua multimea (Mail, Numar_telefon, Adresa) si presupun ca exista multiple dependente intre fiecare

dintre perechile din multimi (Mail, Numar_telefon), (Adresa, Mail), (Adresa, Numar_telefon). Se omite astfel faptul ca Mail se duce in Numar_telefon.

7. Denormalizarea

Denormalizarea este procesul invers al procesului de normalizare. Denormalizarea functioneaza adaugand date redundante sau grupand date pentru a optimiza performanta.

Denormalizare are rolul de a realiza executarea mai rapidă a interogărilor prin introducerea redundanței, punând accentul pe rapiditatea analizei și scăzând numărul de tabele.

Motivul efectuării denormalizării este dat de costul produs în procesor de interogare printr-o structură supra-normalizată.

Denormalizarea poate fi definită și ca metoda de stocare a îmbinării relațiilor de formă normală superioară ca relație de bază, care se află într-o formă normală inferioară. Reduce numărul de tabele și îmbinările de tabel complicate, deoarece un număr mai mare de îmbinări poate încetini procesul.

Aici abordarea denormalizării, subliniază conceptul că, plasând toate datele într-un singur loc, ar putea elimina necesitatea căutării acelor fișiere multiple pentru a colecta aceste date.

În cadrul bazei mele de date, luând spre exemplu, relația DIAGNOSTIC și descompunerea ei de la FN4, este inutil și mult mai costisitor din punct de vedere al timpului de executare să parcurgem datele și din DIAGNOSTIC_1 și din DIAGNOSTIC_2.

Tabele(exercitiul 10)

-----SPITAL-----

```
CREATE TABLE SPITAL(COD_SPITAL INT IDENTITY(1,1) CONSTRAINT PKEY_SPITAL PRIMARY KEY,  
                     DENUMIRE VARCHAR(100) CONSTRAINT DENUMIRE_SPITAL NOT NULL,  
                     ADRESA VARCHAR(100) CONSTRAINT ADRESA_SPITAL NOT NULL,  
                     TELEFON VARCHAR(20) CONSTRAINT TELEFON_SPITAL NOT NULL,  
                     MAIL VARCHAR(50) UNIQUE,  
                     COD_POSTAL VARCHAR(6)  
);
```

INSERT INTO SPITAL

Baze de date
Pasare Roxana-Francisca
Grupa 143

```
VALUES('Spitalul Universitar de Urgente Bucuresti', 'St. Splaiul independentei nr.169', '021-318-0523',  
'universitarurgente@gamil.com', '210728')
```

```
INSERT INTO SPITAL
```

```
VALUES('Spitalul Clinic CF2 Bucuresti', 'Bd. Marasesti nr. 63', '037-229-8711', 'clinicurgentecf@gamil.com', '284928')
```

```
INSERT INTO SPITAL
```

```
VALUES('Spitalul CF Witting', 'St. Calea Pleviei nr. 142-144', '021-317-0068', 'secretariat@spitalcfwiting.com', '284263')
```

```
INSERT INTO SPITAL
```

```
VALUES('Spitalul Universitar de Urgenta Elias', 'Bd. Marasti nr. 17', '021-312-0465', 'spital@elias.com', '837239')
```

```
INSERT INTO SPITAL
```

```
VALUES('Spitalul Clinic Coltea', 'Bd. Ion C. Bratianu nr. 17', '031-333-2873', 'urgente@coltea.com', '906232')
```

100 %						
Results Messages						
	COD_SPITAL	DENUMIRE	ADRESA	TELEFON	MAIL	COD_POSTAL
1	1	Spitalul Universitar de Urgente Bucuresti	St. Splaiul independentei nr.169	021-318-0523	universitarurgente@gamil.com	210728
2	2	Spitalul Clinic CF2 Bucuresti	Bd. Marasesti nr. 63	037-229-8711	clinicurgentecf@gamil.com	284928
3	3	Spitalul CF Witting	St. Calea Pleviei nr. 142-144	021-317-0068	secretariat@spitalcfwiting.com	284263
4	4	Spitalul Universitar de Urgenta Elias	Bd. Marasti nr. 17	021-312-0465	spital@elias.com	837239
5	5	Spitalul Clinic Coltea	Bd. Ion C. Bratianu nr. 17	031-333-2873	urgente@coltea.com	906232

-----CENTRU DE DONARE-----

```
CREATE TABLE CENTRU_DONARE(COD_CENTRU INT IDENTITY(1,1) CONSTRAINT PKEY_CENTRU PRIMARY KEY,  
DENUMIRE VARCHAR(100) CONSTRAINT DENUMIRE_CENTRU NOT NULL,  
DATA_INSCRIERII DATE CONSTRAINT DATA_INSCRIERII NOT NULL,  
COD_SPITAL INT, CONSTRAINT FK_CENTRU FOREIGN KEY(COD_SPITAL)  
REFERENCES SPITAL(COD_SPITAL));
```

```
INSERT INTO CENTRU_DONARE
```

```
VALUES('CDSUUB','2010-02-07',1)
```

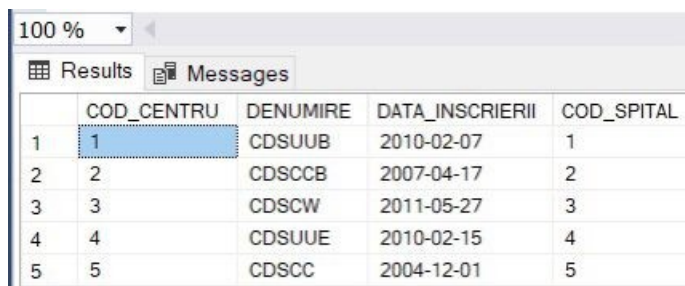
Baze de date
Pasare Roxana-Francisca
Grupa 143

```
INSERT INTO CENTRU_DONARE  
VALUES('CDSCCB','2007-04-17',2)
```

```
INSERT INTO CENTRU_DONARE  
VALUES('CDSCW','2011-05-27',3)
```

```
INSERT INTO CENTRU_DONARE  
VALUES('CDSUUE','2010-02-15',4)
```

```
INSERT INTO CENTRU_DONARE  
VALUES('CDSCC','2004-12-01',5)
```



	COD_CENTRU	DENUMIRE	DATA_INSCRIERII	COD_SPITAL
1	1	CDSUUB	2010-02-07	1
2	2	CDSCCB	2007-04-17	2
3	3	CDSCW	2011-05-27	3
4	4	CDSUUE	2010-02-15	4
5	5	CDSCC	2004-12-01	5

-----DEPARTAMENT-----

```
CREATE TABLE DEPARTAMENT(COD_DEPARTAMENT INT IDENTITY(1,1) CONSTRAINT PKEY_DEPARTAMENT PRIMARY  
KEY,  
  
DENUMIRE VARCHAR(100) CONSTRAINT DENUMIRE_DEPARTAMENT NOT NULL,  
  
COD_SPITAL INT, CONSTRAINT FK_DEPARTAMENT FOREIGN KEY(COD_SPITAL)  
REFERENCES SPITAL(COD_SPITAL));
```

```
INSERT INTO DEPARTAMENT  
VALUES('URGENTE',1)
```

```
INSERT INTO DEPARTAMENT
```

Baze de date
Pasare Roxana-Francisca
Grupa 143

```
VALUES('ATI',1)
```

```
INSERT INTO DEPARTAMENT
```

```
VALUES('CHIRURGIE GENERALA',1)
```

```
INSERT INTO DEPARTAMENT
```

```
VALUES('MEDICINA INTERNA',2)
```

```
INSERT INTO DEPARTAMENT
```

```
VALUES('MEDICINA LEGALA',2)
```

```
INSERT INTO DEPARTAMENT
```

```
VALUES('MEDICINA PRIMARA',2)
```

100 %

Results Messages			
	COD_DEPARTAMENT	DENUMIRE	COD_SPITAL
1	1	URGENTE	1
2	2	ATI	1
3	3	CHIRURGIE GENERALA	1
4	4	MEDICINA INTERNA	2
5	5	MEDICINA LEGALA	2
6	6	MEDICINA PRIMARA	2

----SECTIE----

```
CREATE TABLE SECTIE(COD_SECTIE INT IDENTITY(1,1) CONSTRAINT PKEY_SECTIE PRIMARY KEY,
```

```
DENUMIRE VARCHAR(100) CONSTRAINT DENUMIRE_SECTIE NOT NULL,
```

```
COD_DEPARTAMENT INT, CONSTRAINT FK_SECTIE FOREIGN  
KEY(COD_DEPARTAMENT) REFERENCES DEPARTAMENT(COD_DEPARTAMENT));
```

```
INSERT INTO SECTIE
```

```
VALUES('GASTROLOGIE',1)
```

```
INSERT INTO SECTIE
```

Baze de date
Pasare Roxana-Francisca
Grupa 143

```
VALUES('UROLOGIE',1)
```

```
INSERT INTO SECTIE
```

```
VALUES('CARDIOLOGIE',1)
```

```
INSERT INTO SECTIE
```

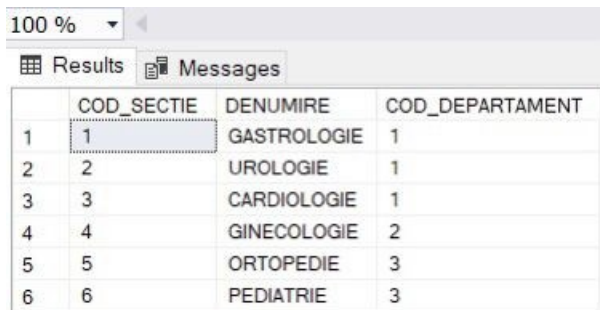
```
VALUES('GINECOLOGIE',2)
```

```
INSERT INTO SECTIE
```

```
VALUES('ORTOPEDIE',3)
```

```
INSERT INTO SECTIE
```

```
VALUES('PEDIATRIE',3)
```



	COD_SECTIE	DENUMIRE	COD_DEPARTAMENT
1	1	GASTROLOGIE	1
2	2	UROLOGIE	1
3	3	CARDIOLOGIE	1
4	4	GINECOLOGIE	2
5	5	ORTOPEDIE	3
6	6	PEDIATRIE	3

-----ETIOLOGIE-----

```
CREATE TABLE ETIOLOGIE(COD_ETIOLOGIE INT IDENTITY(1,1) CONSTRAINT PKEY_ETIOLOGIE PRIMARY KEY,  
                        DENUMIRE VARCHAR(100) CONSTRAINT DENUMIRE_ETIOLOGIE NOT  
NULL,  
                        COD_SECTIE INT, CONSTRAINT FK_ETIOLOGIE FOREIGN  
KEY(COD_SECTIE) REFERENCES SECTIE(COD_SECTIE));
```

```
INSERT INTO ETIOLOGIE
```

```
VALUES('TOXICA',1)
```

```
INSERT INTO ETIOLOGIE
```

Baze de date
Pasare Roxana-Francisca
Grupa 143

```
VALUES('CONGENITALA',1)
```

```
INSERT INTO ETIOLOGIE
```

```
VALUES('ISCHEMICA',1)
```

```
INSERT INTO ETIOLOGIE
```

```
VALUES('INFECTIOASA',2)
```

```
INSERT INTO ETIOLOGIE
```

```
VALUES('RENALA',2)
```

```
INSERT INTO ETIOLOGIE
```

```
VALUES('ETIOLOGIE GENETICA',2)
```

```
INSERT INTO ETIOLOGIE
```

```
VALUES('IDIOPATICA',3)
```

100 %

Results Messages			
	COD_ETIOLOGIE	DENUMIRE	COD_SECTIE
1	1	TOXICA	1
2	2	CONGENITALA	1
3	3	ISCHEMICA	1
4	4	INFECTIOASA	2
5	5	RENALA	2
6	6	ETIOLOGIE GENETICA	2
7	7	IDIOPATICA	3

----PACIENT----

```
CREATE TABLE PACIENT(COD_PACIENT INT IDENTITY(1,1) CONSTRAINT PKEY_PACIENT PRIMARY KEY,  
NUME VARCHAR(100) CONSTRAINT NUME_PACIENT NOT NULL,  
PRENUME VARCHAR(100) CONSTRAINT PRENUME_PACIENT NOT NULL,
```

Baze de date
Pasare Roxana-Francisca
Grupa 143

```
DATA_NASTERII DATE CONSTRAINT DATA_NASTERII NOT NULL,  
SEX VARCHAR(10) CONSTRAINT SEX_PACIENT NOT NULL,  
TELEFON VARCHAR(20) CONSTRAINT TELEFON_PACIENT NOT NULL,  
MAIL VARCHAR(50) UNIQUE,  
COD_CENTRU INT, CONSTRAINT FK_PACIENT FOREIGN  
KEY(COD_CENTRU) REFERENCES CENTRU_DONARE(COD_CENTRU));
```

INSERT INTO PACIENT

VALUES('POPESCU', 'IRINA', '1993-04-24','F','0745-263-972','popescuirina@yahoo.com',1)

INSERT INTO PACIENT

VALUES('IONESCU', 'MIHAI', '1989-11-25','M','0765-433-823','ionescumihai25@gmail.com',1)

INSERT INTO PACIENT

VALUES('DUMITRESCU', 'DAN', '1997-11-08','M','0734-435-893','dandumitrescu8@yahoo.com',null)

INSERT INTO PACIENT

VALUES('CONSTANTINESCU', 'LUMINITA', '2000-12-24','F','0775-678-543','cntlumy@yahoo.com',1)

INSERT INTO PACIENT

VALUES('IROD', 'ANDREEA', '1987-09-21','F','0744-098-322','andreeairod@gmail.com',1)

INSERT INTO PACIENT

VALUES('MARCU', 'ELY', '1984-09-21','F','0789-536-645','marcuely@gmail.com',1)

INSERT INTO PACIENT

VALUES('PATRASCU', 'CLAUDIU', '1967-02-25','M','0795-543-657','claupatrascu@gmail.com',null)

INSERT INTO PACIENT

VALUES('RUS', 'ALIN', '1945-03-08','M','0754-523-535','alinrus8@yahoo.com',null)

Baze de date
Pasare Roxana-Francisca
Grupa 143

INSERT INTO PACIENT

VALUES('EFTENIE', 'ROBERTA', '1987-11-23','F','0744-235-345','robyeftenoIU@yahoo.com',null)

INSERT INTO PACIENT

VALUES('IFTIM', 'ANDREEA', '1998-04-21','F','0735-454-322','iftimandreea21@gmail.com',1)

100 % Results Messages								
	COD_PACIENT	NUME	PRENUME	DATA_NASTERII	SEX	TELEFON	MAIL	COD_CENTRU
1	1	POPESCU	IRINA	1993-04-24	F	0745-263-972	popescuirina@yahoo.com	1
2	2	IONESCU	MIHAI	1989-11-25	M	0765-433-823	ionescumihai25@gmail.com	1
3	3	DUMITRESCU	DAN	1997-11-08	M	0734-435-893	dandumitrescu8@yahoo.com	NULL
4	4	CONSTANTINESCU	LUMINITA	2000-12-24	F	0775-678-543	cntlummy@yahoo.com	1
5	5	IROD	ANDREEA	1987-09-21	F	0744-098-322	andreeairod@gmail.com	1
6	6	MARCU	ELY	1984-09-21	F	0789-536-645	marcuely@gmail.com	1
7	7	PATRASCU	CLAUDIU	1967-02-25	M	0795-543-657	claupatrascu@gmail.com	NULL
8	8	RUS	ALIN	1945-03-08	M	0754-523-535	alinrus8@yahoo.com	NULL
9	9	EFTENIE	ROBERTA	1987-11-23	F	0744-235-345	robyeftenoIU@yahoo.com	NULL
10	10	IFTIM	ANDREEA	1998-04-21	F	0735-454-322	iftimandreea21@gmail.com	1

-----FISA MEDICALA-----

CREATE TABLE FISA_MEDICALA(COD_PACIENT INT CONSTRAINT PK_REPARTIZARE_PACIENT REFERENCES
PACIENT(COD_PACIENT),

DATA_INTEMEIERII DATE CONSTRAINT DATA_INTEMEIRII NOT NULL,

COD_ETIOLOGIE INT CONSTRAINT PK_REPARTIZARE_ETIOLOGIE
REFERENCES ETIOLOGIE(COD_ETIOLOGIE),

CONSTRAINT PK_REPARTIZARE PRIMARY KEY(COD_PACIENT,
COD_ETIOLOGIE));

INSERT INTO FISA_MEDICALA

VALUES(1,'2007-02-07',1)

INSERT INTO FISA_MEDICALA

VALUES(2,'2009-10-10',2)

INSERT INTO FISA_MEDICALA

Baze de date
Pasare Roxana-Francisca
Grupa 143

```
VALUES(3,'2015-06-01',1)
```

```
INSERT INTO FISA_MEDICALA
```

```
VALUES(4,'2011-11-11',4)
```

```
INSERT INTO FISA_MEDICALA
```

```
VALUES(5,'2013-01-12',4)
```

```
INSERT INTO FISA_MEDICALA
```

```
VALUES(6,'2010-11-17',5)
```

```
INSERT INTO FISA_MEDICALA
```

```
VALUES(7,'2006-04-19',4)
```

```
INSERT INTO FISA_MEDICALA
```

```
VALUES(8,'2002-08-22',3)
```

```
INSERT INTO FISA_MEDICALA
```

```
VALUES(9,'2009-09-16',4)
```

```
INSERT INTO FISA_MEDICALA
```

```
VALUES(10,'2013-12-07',5)
```

100 %			
Results Messages			
	COD_PACIENT	DATA_INTEMEIERII	COD_ETIOLOGIE
1	1	2007-02-07	1
2	2	2009-10-10	2
3	3	2015-06-01	1
4	4	2011-11-11	4
5	5	2013-01-12	4
6	6	2010-11-17	5
7	7	2006-04-19	4
8	8	2002-08-22	3
9	9	2009-09-16	4
10	10	2013-12-07	5

Baze de date
Pasare Roxana-Francisca
Grupa 143

-----DOCTOR-----

```
CREATE TABLE DOCTOR(COD_DOCTOR INT IDENTITY(1,1) CONSTRAINT PKEY_DOCTOR PRIMARY KEY,  
                     NUME VARCHAR(100) CONSTRAINT NUME_DOCTOR NOT NULL,  
                     PRENUME VARCHAR(100) CONSTRAINT PRENUME_DOCTOR NOT NULL,  
                     DATA_NASTERII DATE CONSTRAINT DATA_NASTERII NOT NULL,  
                     TELEFON VARCHAR(20) CONSTRAINT TELEFON_DOCTOR NOT NULL,  
                     MAIL VARCHAR(50) UNIQUE);
```

INSERT INTO DOCTOR

VALUES('MARTIN', 'ILEANA', '1997-07-09', '0767-263-342', 'martinileana@yahoo.com')

INSERT INTO DOCTOR

VALUES('DIACONESCU', 'MIHAI', '1992-06-20', '0732-453-535', 'diaconescumihai@gmail.com')

INSERT INTO DOCTOR

VALUES('GHEORGHE', 'DAN', '1980-12-08', '0738-435-345', 'dangh1980@yahoo.com')

INSERT INTO DOCTOR

VALUES('IROD', 'ANA', '1991-07-14', '0723-345-435', 'annairod@yahoo.com')

INSERT INTO DOCTOR

VALUES('TELEMAN', 'DENIS', '1978-11-01', '0790-453-425', 'denisstl@gmail.com')

INSERT INTO DOCTOR

VALUES('ISTRATE', 'MARIA', '1990-03-11', '0744-567-423', 'mariaistrate@gmail.com')

100 %						
Results Messages						
	COD_DOCTOR	NUME	PRENUME	DATA_NASTERII	TELEFON	MAIL
1	1	MARTIN	ILEANA	1997-07-09	0767-263-342	martinileana@yahoo.com
2	2	DIACONESCU	MIHAI	1992-06-20	0732-453-535	diaconescumihai@gmail.com
3	3	GHEORGHE	DAN	1980-12-08	0738-435-345	dangh1980@yahoo.com
4	4	IROD	ANA	1991-07-14	0723-345-435	annairod@yahoo.com
5	5	TELEMAN	DENIS	1978-11-01	0790-453-425	denisstl@gmail.com
6	6	ISTRATE	MARIA	1990-03-11	0744-567-423	mariaistrate@gmail.com

-----CONTRACT-----

```
CREATE TABLE CONTRACT(COD_SPITAL INT CONSTRAINT PK_CONTRACT_FACULTATE REFERENCES
SPITAL(COD_SPITAL),
DATA_INCEPUT DATE CONSTRAINT DATA_INCEPUT NOT NULL,
COD_DOCTOR INT CONSTRAINT PK_CONTRACT_DOCTOR REFERENCES
DOCTOR(COD_DOCTOR),
SALARIU INT CONSTRAINT SALARIU_CONTRACT NOT NULL,
CONSTRAINT PK_CONTRACT PRIMARY KEY(COD_SPITAL,
COD_DOCTOR));
```

```
INSERT INTO CONTRACT
VALUES (1,'1990-12-12',1,5000)
```

```
INSERT INTO CONTRACT
VALUES (2,'1989-02-22',2,6453)
```

```
INSERT INTO CONTRACT
VALUES (3,'1978-01-29',1,7387)
```

```
INSERT INTO CONTRACT
VALUES (1,'1970-03-31',3,5624)
```

```
INSERT INTO CONTRACT
```

Baze de date
Pasare Roxana-Francisca
Grupa 143

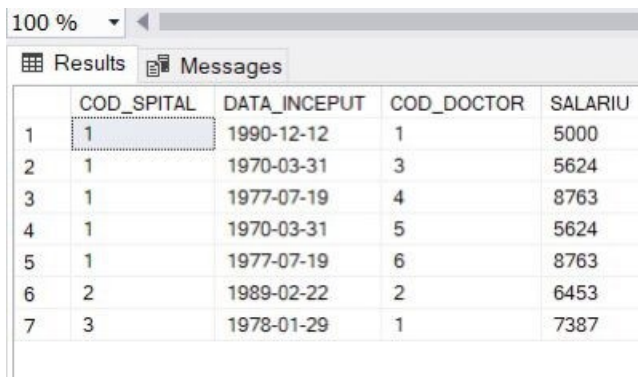
```
VALUES (1,'1977-07-19',4,8763)
```

```
INSERT INTO CONTRACT
```

```
VALUES (1,'1970-03-31',5,5624)
```

```
INSERT INTO CONTRACT
```

```
VALUES (1,'1977-07-19',6,8763)
```



	COD_SPITAL	DATA_INCEPUT	COD_DOCTOR	SALARIU
1	1	1990-12-12	1	5000
2	1	1970-03-31	3	5624
3	1	1977-07-19	4	8763
4	1	1970-03-31	5	5624
5	1	1977-07-19	6	8763
6	2	1989-02-22	2	6453
7	3	1978-01-29	1	7387

-----SALON-----

```
CREATE TABLE SALON(COD_SALON INT IDENTITY(1,1) CONSTRAINT PKEY_SALON PRIMARY KEY,  
DENUMIRE VARCHAR(100) CONSTRAINT DENUMIRE_SALON NOT NULL,  
LOCATIE VARCHAR(100) CONSTRAINT LOCATIE_SALON NOT NULL);
```

```
INSERT INTO SALON
```

```
VALUES ('212','ETAJ 2')
```

```
INSERT INTO SALON
```

```
VALUES ('172','ETAJ 1')
```

```
INSERT INTO SALON
```

```
VALUES ('021','PARTER')
```

```
INSERT INTO SALON
```

Baze de date
Pasare Roxana-Francisca
Grupa 143

```
VALUES ('567','ETAJ 5')
```

```
INSERT INTO SALON
```

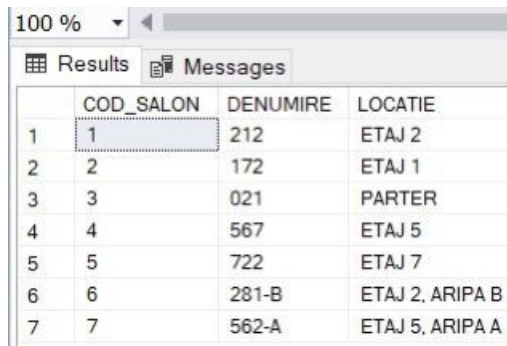
```
VALUES ('722','ETAJ 7')
```

```
INSERT INTO SALON
```

```
VALUES ('281-B','ETAJ 2, ARIPA B')
```

```
INSERT INTO SALON
```

```
VALUES ('562-A','ETAJ 5, ARIPA A')
```



	COD_SALON	DENUMIRE	LOCATIE
1	1	212	ETAJ 2
2	2	172	ETAJ 1
3	3	021	PARTER
4	4	567	ETAJ 5
5	5	722	ETAJ 7
6	6	281-B	ETAJ 2, ARIPA B
7	7	562-A	ETAJ 5, ARIPA A

-----EVOLUTIE_PACIENT-----

```
CREATE TABLE EVOLUTIE_PACIENT(COD_EVOLUTIE INT IDENTITY(1,1) CONSTRAINT PKEY_EVOLUTIE PRIMARY KEY,  
                                DENUMIRE VARCHAR(100) CONSTRAINT DENUMIRE_STARE NOT NULL);
```

```
SELECT * FROM EVOLUTIE_PACIENT
```

```
INSERT INTO EVOLUTIE_PACIENT
```

```
VALUES('BUNA');
```

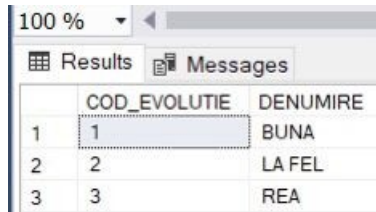
```
INSERT INTO EVOLUTIE_PACIENT
```

Baze de date
Pasare Roxana-Francisca
Grupa 143

```
VALUES('LA FEL');
```

```
INSERT INTO EVOLUTIE_PACIENT
```

```
VALUES('REA');
```



	COD_EVOLUTIE	DENUMIRE
1	1	BUNA
2	2	LA FEL
3	3	REA

-----ANALIZE-----

```
CREATE TABLE ANALIZE(COD_ANALIZE INT IDENTITY(1,1) CONSTRAINT PKEY_ANALIZE PRIMARY KEY,  
                      FORMA VARCHAR(100) CONSTRAINT FORMA_ANALIZE NOT NULL,  
                      COD_EVOLUTIE INT, CONSTRAINT FK_ANALIZE FOREIGN  
KEY(COD_EVOLUTIE) REFERENCES EVOLUTIE_PACIENT(COD_EVOLUTIE));
```

```
SELECT * FROM ANALIZE
```

```
INSERT INTO ANALIZE
```

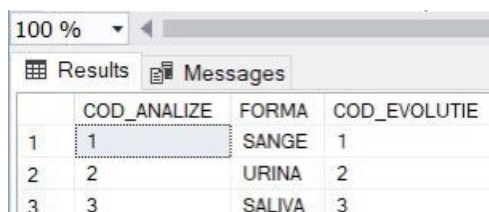
```
VALUES('SANGE',1);
```

```
INSERT INTO ANALIZE
```

```
VALUES('URINA',2);
```

```
INSERT INTO ANALIZE
```

```
VALUES('SALIVA',3);
```



	COD_ANALIZE	FORMA	COD_EVOLUTIE
1	1	SANGE	1
2	2	URINA	2
3	3	SALIVA	3

Baze de date
Pasare Roxana-Francisca
Grupa 143

-----DIAGNOSTIC-----

```
CREATE TABLE DIAGNOSTIC(PROCENT INT CONSTRAINT PROCENT_ANALIZA NOT NULL,  
                           COD_PACIENT INT CONSTRAINT PK_PACIENT REFERENCES  
PACIENT(COD_PACIENT),  
                           COD_ANALIZE INT CONSTRAINT PK_ANALIZE REFERENCES  
ANALIZE(COD_ANALIZE),  
                           CONSTRAINT PK_DIAGNOSTIC PRIMARY KEY(COD_PACIENT,  
COD_ANALIZE));
```

```
INSERT INTO DIAGNOSTIC  
VALUES(90,1,1);
```

```
INSERT INTO DIAGNOSTIC  
VALUES(80,1,2);
```

```
INSERT INTO DIAGNOSTIC  
VALUES(60,1,3);
```

```
INSERT INTO DIAGNOSTIC  
VALUES(45,2,1);
```

```
INSERT INTO DIAGNOSTIC  
VALUES(70,2,2);
```

```
INSERT INTO DIAGNOSTIC  
VALUES(82,3,1);
```

```
INSERT INTO DIAGNOSTIC  
VALUES(93,3,3);
```

```
INSERT INTO DIAGNOSTIC
```

Baze de date
Pasare Roxana-Francisca
Grupa 143

```
VALUES(54,4,1);
```

```
INSERT INTO DIAGNOSTIC
```

```
VALUES(88,5,2);
```

```
INSERT INTO DIAGNOSTIC
```

```
VALUES(39,6,2);
```

```
INSERT INTO DIAGNOSTIC
```

```
VALUES(32,7,1);
```

```
INSERT INTO DIAGNOSTIC
```

```
VALUES(59,8,1);
```

```
INSERT INTO DIAGNOSTIC
```

```
VALUES(88,9,1);
```

```
INSERT INTO DIAGNOSTIC
```

```
VALUES(89,10,1);
```

```
INSERT INTO DIAGNOSTIC
```

```
VALUES(49,10,3);
```

```
INSERT INTO DIAGNOSTIC
```

```
VALUES(78,8,2);
```

```
INSERT INTO DIAGNOSTIC
```

```
VALUES(90,5,1);
```

100 %			
Results Messages			
	PROCENT	COD_PACIENT	COD_ANALIZE
1	90	1	1
2	80	1	2
3	60	1	3
4	45	2	1
5	70	2	2
6	82	3	1
7	93	3	3
8	54	4	1
9	90	5	1
10	88	5	2
11	39	6	2
12	32	7	1
13	59	8	1
14	78	8	2
15	88	9	1
16	89	10	1
17	49	10	3

-----DESFASURATOR-----

```
CREATE TABLE DESFASURATOR(COD_ETIOLOGIE INT CONSTRAINT PK_ETI_DESFASURATOR REFERENCES
ETIOLOGIE(COD_ETIOLOGIE),
                                COD_EVOLUTIE INT CONSTRAINT PK_EV_DESFASURATOR
REFERENCES EVOLUTIE_PACIENT(COD_EVOLUTIE),
                                COD_DOCTOR INT CONSTRAINT PK_DOC_DESFASURATOR
REFERENCES DOCTOR(COD_DOCTOR),
                                COD_SALON INT CONSTRAINT PK_SALON_DESFASURATOR
REFERENCES SALON(COD_SALON),
                                CONSTRAINT PK_DESFASURATOR PRIMARY
KEY(COD_ETIOLOGIE, COD_EVOLUTIE, COD_DOCTOR, COD_SALON));
```

INSERT INTO DESFASURATOR

VALUES(1,1,1,1)

INSERT INTO DESFASURATOR

VALUES(2,2,2,2)

INSERT INTO DESFASURATOR

Baze de date
Pasare Roxana-Francisca
Grupa 143

```
VALUES(3,3,3,3)
```

```
INSERT INTO DESFASURATOR
```

```
VALUES(4,2,4,4)
```

```
INSERT INTO DESFASURATOR
```

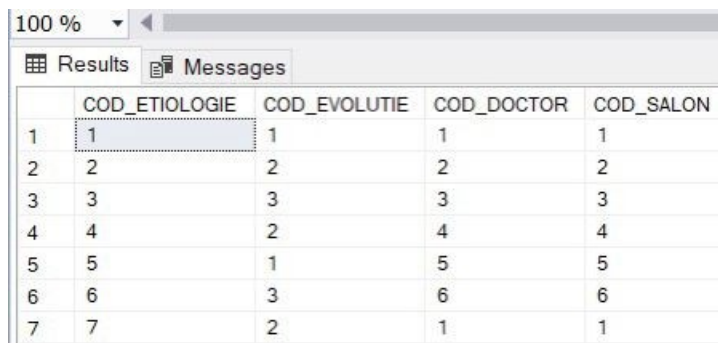
```
VALUES(5,1,5,5)
```

```
INSERT INTO DESFASURATOR
```

```
VALUES(6,3,6,6)
```

```
INSERT INTO DESFASURATOR
```

```
VALUES(7,2,1,1)
```



	COD_ETIOLOGIE	COD_EVOLUTIE	COD_DOCTOR	COD_SALON
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	2	4	4
5	5	1	5	5
6	6	3	6	6
7	7	2	1	1

Secvente SQL(exercitiul 11)

1.Afisarea denumirilor spitalelor care au etiologii in cadrul careia se analizeaza evolutia pacientului in salonul 212.

```
SELECT DISTINCT CONCAT(CONCAT('Spitalul',s.denumire),' are si etiologii secundare in salonul 212') as 'Rezultat'
FROM SPITAL s, DEPARTAMENT d, SECTIE sec, ETIOLOGIE e, DESFASURATOR desf, SALON sal
WHERE s.cod_spital=d.cod_spital and d.cod_departament=sec.cod_departament and
sec.cod_sectie=e.cod_sectie and e.cod_etiologie=desf.cod_etiologie and
desf.cod_salon=sal.cod_salon and UPPER(sal.denumire) like UPPER('212');
```

Results Messages	
	Rezultat
1	Spitalul Universitar de Urgente Bucuresti are si ...

2. Afisare nume, prenume, mail si daca salariul este peste media salariilor al doctorilor care lucreaza la spitalul al caror cod postal se termina cu 28.

```
SELECT d.NUME+' '+d.PRENUME as 'Nume', d.mail as 'Mail'
from DOCTOR d
where d.cod_doctor in(select cod_doctor
                        from CONTRACT
                        WHERE SALARIU>(SELECT AVG(SALARIU)
                        FROM CONTRACT))
and d.cod_doctor in(select cod_doctor
                    from CONTRACT
                    WHERE cod_spital in(select cod_spital
                    from SPITAL
                    where cod_postal%100=28));
```

Results		Messages
	Nume	Mail
1	MARTIN ILEANA	martinileana@yahoo.com
2	DIACONESCU MIHAI	diaconescumihai@gmail.com
3	GHEORGHE DAN	dangh1980@yahoo.com
4	IROD ANA	annairod@yahoo.com
5	TELEMAN DENIS	denisatl@gmail.com
6	ISTRATE MARIA	mariaistrate@gmail.com

3.Ordonare pacienti in functie de starea lor de sanatate afisand numele, prenumele si forma de analiza realizata.

```
SELECT p.num, p.prenume, a.forma
FROM pacient p, diagnostic d , analize a
WHERE p.cod_pacient = d.cod_pacient and a.cod_analize= a.cod_analize
ORDER By d.procent;
```

	nume	prenume	forma
1	PATRASCU	CLAUDIU	SANGE
2	PATRASCU	CLAUDIU	URINA
3	PATRASCU	CLAUDIU	SALIVA
4	MARCU	ELY	SANGE
5	MARCU	ELY	URINA
6	MARCU	ELY	SALIVA
7	IONESCU	MIHAI	SANGE
8	IONESCU	MIHAI	URINA
9	IONESCU	MIHAI	SALIVA
10	IFTIM	ANDREEA	SANGE
11	IFTIM	ANDREEA	URINA
12	IFTIM	ANDREEA	SALIVA
13	CONSTANTINESCU	LUMINITA	SANGE
14	CONSTANTINESCU	LUMINITA	URINA
15	CONSTANTINESCU	LUMINITA	SALIVA
16	RUS	ALIN	SANGE
17	RUS	ALIN	URINA
18	RUS	ALIN	SALIVA
19	POPESCU	IRINA	SANGE
20	POPESCU	IRINA	URINA
21	POPESCU	IRINA	SALIVA
22	IONESCU	MIHAI	SANGE
23	IONESCU	MIHAI	URINA
24	IONESCU	MIHAI	SALIVA
25	RUS	ALIN	SANGE
26	RUS	ALIN	URINA
27	RUS	ALIN	SALIVA
28	DUMITRESCU	DAN	SANGE
29	DUMITRESCU	DAN	URINA
30	DUMITRESCU	DAN	SALIVA
31	POPESCU	IRINA	SANGE
32	POPESCU	IRINA	URINA

4. Ordonarea crescatoare a salarilor.

```
select *
from contract
order by salariu
```

	COD_SPITAL	DATA_INCEPUT	COD_DOCTOR	SALARIU
1	2	1989-02-22	4	2850
2	3	1954-06-27	2	2900
3	3	1967-03-13	4	2950
4	3	1989-12-22	3	3050
5	1	1990-12-12	1	5000
6	1	1970-03-31	3	5624
7	1	1970-03-31	5	5624
8	2	1989-02-22	2	6453
9	3	1978-01-29	1	7387
10	1	1977-07-19	4	8763
11	1	1977-07-19	6	9639

5.Sortare salarii (mic/mediu/mare)

```
SELECT SALARIU,
CASE
WHEN SALARIU<3000 THEN 'Salariul este mic'
```

```
WHEN SALARIU BETWEEN 2000 AND 5000 THEN 'Salariul este mediu'
else 'Salariu este mare'
end as Salarii
from Contract
```

	SALARIU	Salarii
1	5000	Salariul este mediu
2	5624	Salariu este mare
3	8763	Salariu este mare
4	5624	Salariu este mare
5	9639	Salariu este mare
6	6453	Salariu este mare
7	2850	Salariul este mic
8	7387	Salariu este mare
9	2900	Salariul este mic
10	3050	Salariul este mediu
11	2950	Salariul este mic

5. Sa se afiseze codul doctorilor, cu noul lor salariu, care a scazut cu 20% din cadrul spitalelor care au centrul de donare infiintat la data de 07-02-2010. In aceasta cerere am folosit NVL, filtrare la nivel de linii, subcereri nesincronizate, functii care lucreaza cu date.

```
SELECT COD_DOCTOR, COALESCE(SALARIU-0,2*SALARIU) "Noul salariu al doctorului"
FROM CONTRACT
WHERE COD_SPITAL IN (SELECT COD_SPITAL
                        FROM CENTRU_DONARE
                        WHERE COD_CENTRU=(SELECT COD_CENTRU
                                           FROM CENTRU_DONARE
                                           WHERE DATA_INSCRIERII='2010-02-07'));
```

	COD_DOCTOR	Noul salariu al doctorului
1	1	5000
2	3	5624
3	4	8763
4	5	5624
5	6	9639

6.Vrem sa vedem spitalele care au codul mai mare decat 2 si adresa diferita de null. Pentru a afla acest lucru, vom folosi in query o cerere sincronizata pe tabele, functia concat si functia care transforma un intreg in string.

```
SELECT CONCAT(CONCAT('-',>', ' '),s.DENUMIRE)+STR(s.COD_SPITAL) as "Denumire+cod"
FROM SPITAL s
where s.COD_SPITAL IN (SELECT COD_SPITAL
                        FROM SPITAL
```

```
WHERE COD_SPITAL>2)
AND s.COD_SPITAL in (SELECT s.COD_SPITAL
FROM SPITAL
WHERE (ADRESA IS NOT
NULL));
```

Results		Messages
Denumire+cod		
1	-> Spitalul CF Witting 3	
2	-> Spitalul Universitar de Urgenta Elias 4	
3	-> Spitalul Clinic Coltea 5	

7. Vrem sa aflam denumire spitalelor care au contracte in cadrul centrelor de donare cu denumirea 'CDSUUB' sau 'CDSCCB'.

```
select DENUMIRE
from spital s
where s.cod_spital in (select cod_spital
FROM CONTRACT
WHERE COD_SPITAL
IN ( SELECT COD_SPITAL
FROM CENTRU_DONARE
WHERE DENUMIRE='CDSUUB' OR DENUMIRE='CDSCCB' )
);
```

Results		Messages
DENUMIRE		
1	Spitalul Universitar de Urgente Bucuresti	
2	Spitalul Clinic CF2 Bucuresti	

```
SELECT * FROM CONTRACT
WHERE SALARIU IN (3050, 5000);
```

8. Sa se afiseze codul doctorilor care au salariul peste medie di lucreaza in Spitalul Cf Witting.

```
with salary(avgsalary) as (select avg(salariu) from contract )
select cod_doctor
from contract, salary
where contract.salariu>salary.avgsalary and contract.cod_doctor in (select cod_doctor
from desfasurator
where cod_spital=(select cod_spital from spital where denumire='Spitalul CF Witting'))
```

9. Sa se afiseze numarul de verdicte legate de evolutia pacientului realizate intr-un salon.

```
select s.cod_salon, isnull(count(d.cod_salon),0) as 'Numar verdicte'
from salon s left outer join desfasurator d on (s.cod_salon=d.cod_salon)
group by s.cod_salon
```

Update-uri(exercitiul 12)

```
UPDATE CONTRACT
SET SALARIU=SALARIU+0.1*SALARIU
WHERE COD_DOCTOR IN(SELECT COD_DOCTOR
                     FROM DOCTOR
                     WHERE TELEFON LIKE '074%');
```

```
UPDATE DIAGNOSTIC
SET PROCENT=PROCENT+2
WHERE procent < 70 and cod_pacient in (select cod_pacient
                                       from pacient
                                       where sex LIKE 'F');
```

```
UPDATE DIAGNOSTIC
SET PROCENT=PROCENT+2
WHERE procent > 50 and cod_pacient in (select cod_pacient
                                       from pacient
                                       WHERE TELEFON LIKE '074%');
```

- 1.Sa se mareasca salariu cu 10% pentru doctorii care au numarul de telefon incepand cu "074".
2. Sa se mareasca procentul cu 2 pentru pacientii care au un procent mai mic decat 70 si sunt fete – marire de precautie din cauza schimbarilor hormonale.
3. Sa se scada procentul pacientilor al caror numar incepe cu '074'.

New_insert(exercitiul 13)

```
CREATE SEQUENCE salary_increment
INCREMENT BY 50
START WITH 2800
MAXVALUE 9700
CYCLE
```

```
INSERT INTO CONTRACT
VALUES (2, '1989-02-22', 4, NEXT VALUE FOR salary_increment)
```

```
INSERT INTO CONTRACT
VALUES (3, '1954-06-27', 2, NEXT VALUE FOR salary_increment)
```

```
INSERT INTO CONTRACT
VALUES (3, '1967-03-13', 4, NEXT VALUE FOR salary_increment)
```

```
INSERT INTO CONTRACT
VALUES (1, '1990-09-18', 4, NEXT VALUE FOR salary_increment)
```

```
INSERT INTO CONTRACT
VALUES (3, '1989-12-22', 3, NEXT VALUE FOR salary_increment)
```

	COD_SPITAL	DATA_INCEPUT	COD_DOCTOR	SALARIU
1	1	1990-12-12	1	5000
2	1	1970-03-31	3	5624
3	1	1977-07-19	4	8763
4	1	1970-03-31	5	5624
5	1	1977-07-19	6	9639
6	2	1989-02-22	2	6453
7	2	1989-02-22	4	2850
8	3	1978-01-29	1	7387
9	3	1954-06-27	2	2900
10	3	1989-12-22	3	3050
11	3	1967-03-13	4	2950

LMD(exercitiul 14)

```
CREATE VIEW [Doctori care au salarii mari] AS
SELECT COUNT(*) AS 'Doctori'
FROM CONTRACT
WHERE CONTRACT.COD_DOCTOR IN (SELECT COD_DOCTOR FROM DOCTOR)
GROUP BY SALARIU/1000
```

```
SELECT * FROM [Doctori care au salarii mari]
```

--Operatie LMD care nu merge

```
SELECT * FROM [Doctori care au salarii mari]
WHERE [DOCTORI]< 1
```

--Operatie LMD care merge

```
SELECT * FROM [Doctori care au salarii mari]
WHERE [DOCTORI]> 1
```

	Doctori
1	3
2	3

Operatii permise:

Vizualizarile pot fi folosite pentru operatii DML (Delete, Insert, Update).

Operatii nepermise:

Nu este permis sa se creze index-uri pe vizualizari.

Index(exercitiul 15)

```
create index index_doc
on contract(cod_doctor, cod_spital)
select
salariu
from contract
where cod_spital=1
```

	salariu
1	5000
2	5624
3	8763
4	5624
5	9639

Operatii division si outer-join(exercitiul 16)

1.Sa se afiseze doctorii care au de consultat pacienti la etiologii din sectia 1 (utilizarea opeatiei division).

```
select * from doctor d
where not exists(
select cod_etiologie
from desfasurator
where d.cod_doctor=cod_doctor
except
select cod_etiologie
from etiologie
where cod_sectie like '1%');
```

	COD_DOCTOR	NUME	PRENUME	DATA_NASTERII	TELEFON	MAIL
1	2	DIACONESCU	MIHAI	1992-06-20	0732-453-535	diaconescumihai@gmail.com
2	3	GHEORGHE	DAN	1980-12-08	0738-435-345	dangh1980@yahoo.com

2.Sa se afiseze toti pacientii care au efectuat analize pentru a fi observata evolutia cu codul 1.

```
select * from pacient p
where not exists(
(select cod_analize
from DIAGNOSTIC
where p.cod_pacient=cod_pacient)
except
(select cod_analize
from analize
```


where cod_evolutie=1));

	COD_PACIENT	NUME	PRENUME	DATA_NASTERII	SEX	TELEFON	MAIL	COD_CENTRU
1	4	CONSTANTINESCU	LUMINITA	2000-12-24	F	0775-678-543	cntlummy@yahoo.com	1
2	7	PATRASCU	CLAUDIU	1967-02-25	M	0795-543-657	claupatrascu@gmail.com	NULL
3	9	EFTENIE	ROBERTA	1987-11-23	F	0744-235-345	robeyftenoiu@yahoo.com	NULL

3. Sa se afiseze pentru fiecare pacient numele si prenumele, forma analizei si evolutia sa.(utilizarea operatiei outer-join).

```
select p.num, p.prenume, a.forma as 'Forma analiza', ev.denumire as 'Evolutie'
from pacient p right outer join diagnostic d on (d.cod_pacient=p.cod_pacient) left outer
join analize a
on (d.cod_analiza=a.cod_analiza) full outer join evolutie_pacient ev
on(ev.cod_evolutie=a.cod_evolutie)
```

	nume	prenume	Forma analiza	Evolutie
1	POPESCU	IRINA	SANGE	BUNA
2	POPESCU	IRINA	URINA	LA FEL
3	POPESCU	IRINA	SALIVA	REA
4	IONESCU	MIHAI	SANGE	BUNA
5	IONESCU	MIHAI	URINA	LA FEL
6	DUMITRESCU	DAN	SANGE	BUNA
7	DUMITRESCU	DAN	SALIVA	REA
8	CONSTANTINESCU	LUMINITA	SANGE	BUNA
9	IROD	ANDREEA	SANGE	BUNA
10	IROD	ANDREEA	URINA	LA FEL
11	MARCU	ELY	URINA	LA FEL
12	PATRASCU	CLAUDIU	SANGE	BUNA
13	RUS	ALIN	SANGE	BUNA
14	RUS	ALIN	URINA	LA FEL
15	EFTENIE	ROBERTA	SANGE	BUNA
16	IFTIM	ANDREEA	SANGE	BUNA
17	IFTIM	ANDREEA	SALIVA	REA

4. Sa se afiseze numarul de verdicte legate de evolutia pacientului realizate intr-un salon.

```
select s.cod_salon, isnull(count(d.cod_salon),0) as 'Numar verdicte'
from salon s left outer join desfasurator d on (s.cod_salon=d.cod_salon)
group by s.cod_salon
```

	cod_salon	Numar verdicte
1	1	2
2	2	1
3	3	1
4	4	1
5	5	1
6	6	1
7	7	0