**PROIECT la disciplina BAZE DE DATE**

***Farmaciile dintr-un oras oarecare***



Coordonator,

Prof. Dr. Ing. Viorel Stoian

**Cuprins**

CAPITOLUL I. INTRODUCERE.........................................3

CAPITOLUL II. TEMA DE PROIECT.................................4

CAPITOLUL III. SHEMA CONCEPTUALA.......................5

**3.1. Noțiuni Teoretice.......................................................................................5**

**3.2. Schema Conceptuală................................................................................6**

CAPITOLUL IV. SCHEMA LOGICA..................................7

**4.1. Noțiuni Teoretice.................................................................................7**

**4.2. Schema Logică.....................................................................................8**

**Tabele..........................................................................................................9**

CAPITOLUL V. NORMALIZAREA BD............................10

**5.1. Noțiuni Teoretice..............................................................................10**

**5.2. Normalizarea tabeleleror bazei de date..........................................10**

CAPITOLUL VI. DENORMALIZAREA BD.....................11

**6.1. Noțiuni Teoretice..............................................................................11**

**6.2. Denormalizarea bazei de date.........................................................11**

CAPITOLUL VII. SGBD UTILIZAT..................................12

**7.1. Noțiuni Teoretice..............................................................................12**

**7.2. Aplicații.............................................................................................14**

CAPITOLUL VIII. CONCLUZII.........................................33

BIBLIOGRAFIE...................................................................34

**I.**

**Introducere**

Datele reprezintă informaţii fixate pe un anumit suport fizic în vederea utilizării şi

prelucrării într-un anumit scop.Baza de date (data base) este o colecţie de date organizate care serveşte unui anumit scop (nu conţine date care nu sunt relevante). Faptul că sunt organizate înseamnă că sunt stocate, reprezentate şi accesate într-o manieră consistentă.

Dezvoltarea bazelor de date s-a bazat pe 2 cerinţe:

- persistenţa datelor (datele trebuie să fie valide pentru mai multe rulări),

- simplitatea stocării şi manipulării datelor.

Sistemul bazelor de date are 4 niveluri: Nivelul conceptual, Nivelul extern, Nivelul logic, Nivelul intern.

**Sisteme de gestiune a bazelor de date (SGBD)**

Un SGBD (Sistem de Gestiune a Bazelor de Date) sau DBMS (DataBase Management System) este un sistem software care gestionează toate procesele dintr-o bază de date şi care permite utilizatorului să interacţioneze cu aceasta.

Principalele funcţiuni ale unui SGBD sunt:stocarea datelor,definirea structurilor de date,,manipularea datelor,,interogarea (extragerea şi prelucrarea) datelor,,asigurarea securităţii datelor,, asigurarea integrităţii datelor,accesul concurent la date cu păstrarea consistenţei acestora, asigurarea unui mecanism de recuperare a datelor,, asigurarea unui mechanism de indexare care să permită accesul rapid la date.

**După modelul folosit există mai multe tipuri de SGBD:** SGBD ierarhic, SGBD reţea, SGBD relational, SGBD orientat pe obiect.

**Comunicarea cu baza de date**

În ORACLE comunicarea cu baze de date se face prin intermediul limbajului SQL

(Structured Query Language) care este în prezent cel mai răspândit limbaj de interogare a bazelor de date relationale.

**Datele operaţionale**

*Datele operaţionale* sunt date din bazele de date, distincte de datele de intrare, ieşire sau alte tipuri de date. O bază de date este o colecţie de date operaţionale folosite de către aplicaţiile sistem ale unei instituţii (Muzeu, Bibliotecă, Întreprindere, etc.)

*Datele de intrare* sunt informaţii introduse în sistem din lumea exterioară, de obicei prin terminale.

*Datele de ieşire* se referă la mesajele şi rapoartele extrase din sistem (tipărite sau afişate pe ecran).

*Entităţile de bază* sunt elementele constitutive ale unei baze de date (ex. materialul arheologic, materialul bibliografic, materialul grafic, etc.). Între aceste entităţi există întotdeauna *asociaţii* sau *relaţii* ce le leagă într-o bază de date comună.

**II.**

**TEMA DE PROIECT**

**Farmaciile dintr-un oras oarecare**

Sa se proiecteze o BD care sa modeleze problematica legata de un lant de farmacii dintr-un oras.In oras exista un lant de mai multe farmacii despre care se inregistreaza date.

In fiecare farmacie exista mai multi angajati cu statuturi diferite (sef, angajati obisnuiti (absolventi ai unor institutii de specialitate, personal administrativ, personal de intretinere), studenti practicanti de la facultatile de Medicina, de Farmacie, cursanti practicanti de la scoli postliceale de asistenti medicali etc.

Farmaciiile vand produse diferite: medicamente, alte produse medicale (seringi, bandaje, vata etc.), produse de PLAFAR etc.

Livrarea produselor se face in localurile farmaciilor, direct clientilor, sau la domiciliul acestora, atunci cand sunt probleme cu deplasarea (in urma citirii retetei medicale la telefon).

Pentru acest serviciu exista un numar de masini si de soferi angajati ai lantului de farmacii.

Farmaciile se aprovizioneaza de la depozite specializate, amplasate in orase diferite si in locatii diferite (pot exista mai multe depozite intr-un oras).Farmaciile sunt deschise non-stop (3 ture).

Baza de date tine o evidenta a informatiilor necesare despre angajati (inclusiv a situatiilor in care se lucreaza in ture (vanzatori)), despre produse, despre interventii de intretinere, deplasari la domiciliu, despre clientii fideli pentru a li se trimite materiale promotionale, despre furnizori etc.

*OBS: Proiectantii bazei de date pot veni cu noi propuneri pentru marirea cantitatii de informatii.*

**III.**

**SCHEMA CONCEPTUALA**

**3.1 Notiuni teoretica**

Proiectarea BD cuprinde 3 etape principale:

- Realizarea schemei conceptuale a BD

- Realizarea proiectului logic al BD (schemei logice a BD)

- Realizarea proiectului fizic al BD (schemei fizice a BD)

Realizarea schemei conceptuale a BD (modelul entitate - legătură)

În prima fază, o echipă nominalizată colectează (achiziţionează) datele

corespunzatoare din sistem, apoi urmează faza de organizare a acestora utilizându-se modelul

entitate-legătură. Principalele concepte folosite în acest model sunt: entitatea, relaţia (legătura) şi atributul. Entitatea este un obiect de interes din sistem pentru care trebuie să existe date

înregistrate.

**Cardinalitatea unei relaţii** indică numărul de instanţe din fiecare entitate care poate participa la relaţie. Există 3 tipuri de cardinalitate:

- "**mulţi-la-unu" (many-to-one, M:1)**.

Relaţia dintre entităţile A şi B este de tipul "mulţi-la-unu" dacă fiecarei instanţe din A i se poate asocia cel mult o singură instanţă din B şi fiecărei instanţe din B i se pot asocia mai multe instanţe din A.

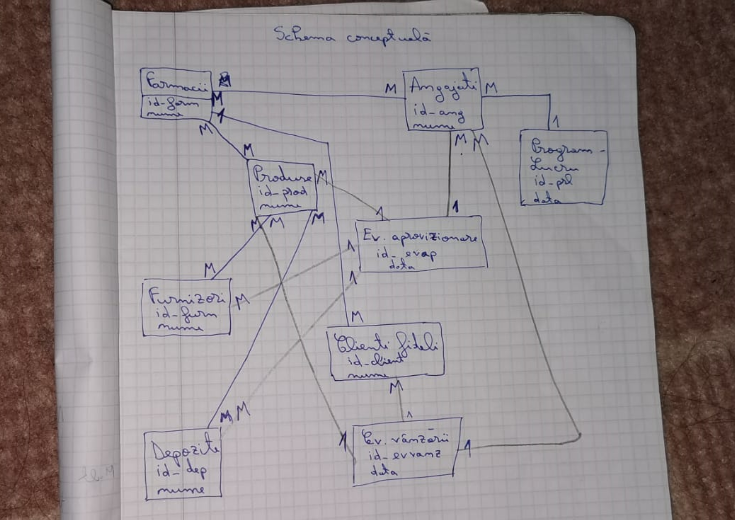
- **"unu-la-unu" (one-to-one, 1:1)**.

Relaţia dintre entităţile A şi B este de tipul "unu-la-unu" dacă fiecărei instanţe din A i se poate asocia cel mult o singură instanţă din B şi fiecărei instanţe din B i se poate asocia cel mult o singură instanţă din A.

- **"mulţi-la-mulţi"** (**many-to-many, M:M)**.

Relaţia dintre entităţile A şi B este de tipul "mulţi-la-unu" dacă fiecărei instanţe din A i se pot asocia mai multe instanţe din B şi fiecărei instanţe din B i se pot asocia mai multe instanţe din A.Valorile prezentate până acum (M:1, 1:1, M:M) reprezintă cardinalitatea maximă a unei relaţii. Pe de altă parte, o relaţie este caracterizată şi de o cardinalitate minimă ce indică obligativitatea participării entităţilor la relaţie. Cardinalitatea minimă a unei relaţii poate avea valorile: 0:0, 0:1, 1:1. Dacă avem cardinalitatea minimă a unei relaţii egală cu 1 înseamnă că există o participare totală a entităţii la relaţie (participare obligatorie). Dacă avem cardinalitatea minimă egală cu 0 înseamnă că există o participare parţială a entităţii la relaţie.

**3.2 Schema conceptuala a temei proiectului**



**IV. SCHEMA LOGICA**

**4.1. Noțiuni Teoretice**

Pentru realizarea schemei logice a unei baze de date se porneşte de la schema conceptuală (modelul entitate – legătură) urmărindu-se conversia entităţilor şi a legăturilor în tabele relaţionale.

Regulile de conversie ale entităţilor şi legăturilor sunt următoarele:

**Transformarea entităţilor**

**Regulă generală**: entităţile se transformă în tabele.

Subcazuri:

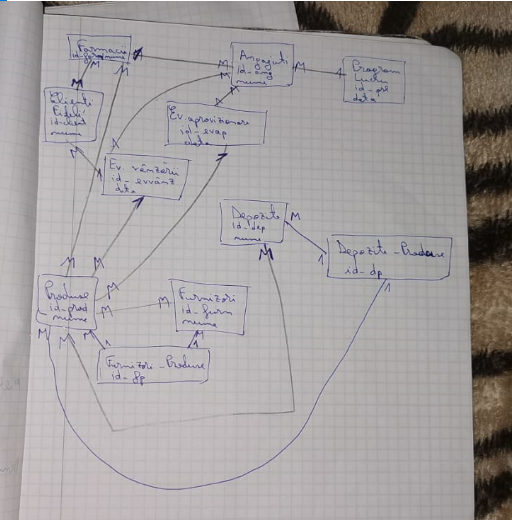
1. Entităţile independente devin tabele independente, adică tabele a căror cheie primară nu conţine chei străine.
2. Entităţile dependente devin tabele dependente (tabele detaliu) adică tabele a căror cheie primară conţine cheia străină ce face referinţă la cheia primară a entităţii de care depinde entitatea în cauză.
3. Subentităţile devin subtabele, adică tabele a căror cheie primară este cheia străină pentru tabelul superentitate.

**Transformarea relaţiilor (legăturilor)**

**Cazuri**:

1. Relaţiile 1:1 devin chei străine. Cheia străină este plasată în tabelul cu linii mai puţine.
2. Relaţiile M:1 devin chei străine plasate în tabelul care se află în partea de “mulţi” a relaţiei.
3. O relaţie M:M se transformă în 2 relaţii M:1. În acest caz se construieşte un tabel special numit *tabel asociativ* care are 2 chei străine care fac referinţă la cheile primare ale celor 2 tabele aflate în relaţia M:M. Cheia sa primară este formată din cele 2 chei străine plus (eventual) alte atribute suplimentare.
4. O relaţie de tip 3 se transformă într-un număr de relaţii de tip 2, egal cu numărul de tabele asociate. Aceste relaţii (legături) se stabilesc între un tabel asociativ si tabelele asociate. Tabelul asociativ are câte o cheie străină pentru fiecare tabel asociat, iar cheia sa primară este formată din toate aceste chei străine plus (eventual) alte atribute suplimentare.

**4.2 Schema logica a temei proiectului**



**Tabele:**

FARMACII (id\_farm, nume, adresa, tel, email, site, id\_manag)

(id\_manag face referire la id\_ang)

PRODUSE (id\_prod, nume, descriere)

FURNIZORI (id\_furn, nume, oras, adresa, tel, email, site)

DEPOZITE (id\_dep, nume, adresa, tel, email, site)

FURNIZORI-PRODUSE (id\_fp, id\_furn, id\_prod)

DEPOZITE-PRODUSE (id\_dp, id\_furn, id\_prod)

ANGAJATI (id\_ang, nume, prenume, data\_nast, functie, salariu, data\_angaj, id\_farm, adresa, tel, email)

(functie: manager, economist, vanzari, aprovizionare, IT-ist,sofer )

EVIDENTA\_ APROVIZIONARE (id\_evap, data, id\_prod, id\_furn,id\_depozit , pret\_cump, nr\_factura,cantitate, id\_ang\_resp)

CLIENTI\_FIDELI (id\_client, nume, prenume, adresa, tel, email)

EVIDENTA\_VANZARI (id\_evvanz, data, id\_client, nr\_factura, id\_prod, pret\_vanz, discount, cantitate,id\_ang\_resp)

PROGRAM\_LUCRU (id\_prl, data, tura, id\_ang)

**CAPITOLUL V. NORMALIZAREA BD**

**5.1. Noțiuni Teoretice**

Normalizarea reprezintă procesul de descompunere a unui tabel relaţional în mai multe tabele care satisfac anumite reguli şi care stochează aceleaşi date ca şi tabelul iniţial astfel încât să fie eliminate redundanţa în date şi anomaliile la actualizare.Obiectivele normalizării sunt :

1.Să permită reprezentarea oricărei relaţii în baza de date .

2.Să opţină algoritmii de căutare puternici bazaţi pe o colecţie de operatori relaţionali .

3.Să elimine din relaţii anomaliile de inserţie ,modificare şi ştergere.

4.Să reducă necesitatea restructurării relaţiilor atunci cănd se introduce un nou tip de date sau se modifică ipotezele de lucru.

5.Să facă colecţia de relaţii imună în raport cu cererile aleatoare a căror semantică se modifică în timp.

a) Caracterul reversibil al normalizării.

Prin caracter reversibil al normalizării se înţelege faptul că descompunerea se face fără pierdere de informaţie, adică tabelul iniţial poate fi reconstituit prin compunerea naturală, pe atribute comune, a tabelelor rezultate.

Pentru un tabel R care se descompune prin proiecţie în mai multe tabele: R1, R2, … Rn, condiţia de descompunere fără pierdere de informaţie presupune că în urma operaţiei de compunere naturală a tabelelor R1, R2, … Rn să se obţină tabelul R.

b) Dependenţa funcţională

Fie R un tabel relaţional si X si Y doua submulţimi de coloane ale lui R. Spunem ca X determină funcţional pe Y sau ca Y depinde funcţional de X dacă nu există două rânduri în tabelul R care să aibă aceleaşi valori pentru coloanele din X şi să aibă valori diferite pentru cel puţin o coloană din Y.Notaţie uzuala: X 🡪 Yunde X = determinantY = determina X 🡪 Y este triviala daca Y X.

c) Dependenta funcţională tranzitivă

Fie R un tabel relaţional, X o submulţime de coloane a lui R si A o coloană a lui R. Spunem că A este dependentă tranzitiv de X daca există o submulţime de coloane Y care nu include coloana A şi care nu determină funcţional pe X astfel încât X 🡪 Y si Y 🡪 A. Dacă în această definiţie se doreşte să se evidenţieze şi Y atunci se spune că A depinde funcţional de X prin intermediul lui Y şi se scrie: X 🡪 Y 🡪 A.

d) Descompunerea minimală

Prin descompunerea minimală a unui tabel se înţelege o descompunere astfel încât nici o coloană din tabelele rezultate nu poate fi eliminată fără a duce la pierderea de informaţii şi implicit la pierderea caracterului ireversibil al transformării. Aceasta înseamnă că nici unul dintre tabelele rezultate nu poate fi conţinut unul în altul.

5.2. Normalizarea tabelelor bazei de date

Deoarece în etapele elaborarii schemei conceptuale si logice s-au avut în vedere si caracteristicile de la etapa de normalizare (caracteristici desprinse din algoritmii de la fiecare etapa a procesului de normalizare) aceasta nu mai este necesară.

**CAPITOLUL VI. DENORMALIZAREA BD**

**6.1. Noțiuni Teoretice**

Denormalizarea unei BD reprezinta procesul invers operatiei de normalizare si duce la cresterea redundantei datelor.Prin aceasta se doreste, in principal, cresterea performantei si simplificarea programelor de interogare a datelor.

**Obs.:**Denormalizarea se face numai dupa o normalizare corecta.

- Denormalizarea se face printr-o selectare strategica a structurilor care aduc avantaje semnificative

Denormalizarea trebuie insotita de masuri suplimentare de asigurare a integritatii datelor.

**a) Cresterea performantei**

Un caz frecvent intalnit in interogarea BD este cazul unor operatii sau calcule foarte des utilizate.

**b) Simplificarea codului pentru manipularea datelor**

**Exemplu:** Fie tabelul STOCURI utilizat de o firma pentru inregistrarea cantitatilor de materiale existente in fiecare din depozitele sale.

STOCURI (cod\_depozit, cod\_material, nume\_material, cantitate)

Se cere aflarea tuturor depozitelor in care exista ciocolata.

Dupa normalizare avem:

STOCURI\_1 (cod\_depozit, cod\_material, cantitate)

MATERIAL (cod\_material, nume\_material)

Interogarea in SQL va fi:

- varianta nenormalizata:

SELECT cod\_depozit, cantitate

FROM stocuri

WHERE nume\_material = “ciocolata”;

- varianta normalizata:

SELECT cod\_depozit, cantitate

FROM stocuri\_1, material

WHERE stocuri\_1.cod\_material = material.cod\_material

AND nume\_material = “ciocolata”;

O solutie care rezolva problema conta in a crea vederi bazate pe tabele normalizate.

Ex: CREATE VIEW stocuri AS

SELECT cod\_depozit, stocuri\_1.cod\_material, nume\_material, cantitate

FROM stocuri\_1, material

WHERE stocuri\_1.cod\_material = material. cod\_material;

**6.2 Denormalizarea bazei de date**

Deoarece în etapele elaborarii schemei conceptuale si logice s-au avut în vedere si caracteristicile de la etapa de denormalizare aceasta nu mai este necesară.

**CAPITOLUL VII. SGBD UTILIZAT**

**7.1. Noțiuni Teoretice**

Un SGBD (Sistem de Gestiune a Bazelor de Date) sau DBMS (DataBase Management System) este un sistem software care gestionează toate procesele dintr-o bază de date şi care permite utilizatorului să interacţioneze cu aceasta. Principalele funcţiuni ale unui SGBD sunt:

- stocarea datelor,

- definirea structurilor de date,

- manipularea datelor,

- interogarea (extragerea şi prelucrarea) datelor,

- asigurarea securităţii datelor,

- asigurarea integrităţii datelor,

- accesul concurent la date cu păstrarea consistenţei acestora,

- asigurarea unui mecanism de recuperare a datelor,

- asigurarea unui mecanism de indexare care să permită accesul rapid la date.

**Modelul de date** reprezintă un tipar după care este organizată din punct de vedere logic baza de date. După modelul folosit există mai multe tipuri de SGBD.

**- SGBD ierarhic**

Modelul ierarhic stochează datele în structuri de tip arbore. Se consideră că între date există o relaţie de tip părinte-copil. Nivelul superior al arborelui (rădăcina) poate avea orice număr de descendenţi care şi ei, la rândul lor, au alţi descendenţi etc. În prezent, modelul ierarhic este depăşit şi nu se mai foloseşte aproape deloc.

**- SGBD reţea**

Datele sunt stocate sub formă de înregistrări cu legături multiple şi complexe între ele. Este o extindere a celui ierarhic. Aici un copil poate avea mai mulţi părinţi sau chiar niciunul. Caracteristicile principale ale SGBD reţea sunt: reprezentare date complexe , extrem de puţin flexibil, design extrem de complicat.În prezent este puţin folosit.

**- SGBD relaţional**

Reprezintă cea mai simplă structură pe care o poate avea o bază de date. Datele sunt organizate în tabele formate din înregistrări şi câmpuri. În acest caz bazele de date relaţionale sunt foarte flexibile şi uşor de mânuit. Cele mai populare baze de date relaţionale: Oracle, Acces, Informix şi Sybase. Altele : SQL server şi DB2.

**- SGBD orientat pe obiect**

Este tipul cel mai nou de SGBD fiind introdus conceptul de obiect. Integrează principiile programării orientate pe obiect (C++, Actor etc.) cu cele ale bazelor de date. Gestionează obiecte complexe (date neconvenţionale) (texte, grafice, hărţi imagini sunete); obiecte dinamice (programe, simulări). Tehnologia este la început (Oracle 8 şi 9).

Un SGBDR este un SGBD care utilizează modelul relaţional ca şi concepţie de organizare a datelor. În 1985 Codd a publicat un set de 13 reguli în raport cu care un SGBD poate fi considerat relaţional. În prezent niciun SGBD nu respectă întreg setul de reguli care are rolul de a stabili gradul în care unul sau altul dintre SGBD-uri este relaţional.

Regulile lui Codd Rg.

1: Regula reprezentării logice a datelor . Când este încălcată crează probleme de integritate a datelor, demonstrează o proiectare deficientă a BD, iar SGBD-ul îşi pierde calitatea de relaţional.

Rg. 2: Regula accesului la date.

Rg. 3: Regula reprezentării valorilor necunoscute .Un SGBDR trebuie să facă diferenţa între valoarea numerică 0 şi Null sau între şirul de caractere “spaţiu” şi valoarea Null. Valoarea Null trebuie să reprezinte absenţa informaţiei respective şi are un rol important în implementarea restricţiilor de integritate structurală (integritatea entităţii şi integritatea referirii).

Rg. 4: Regula dicţionarului de date. Descrierea bazei de date (dicţionarul de date) trebuie să fie reprezentată la nivel logic tot sub formă de tabele, astfel încât asupra acesteia să se poată aplica aceleaşi operaţii ca şi asupra datelor propriuzise.

Rg. 5: Regula limbajului de acces.Limbajul SQL permite: - definirea tabelelor (comenzile CREATE TABLE, ALTER TABLE, DROP TABLE); - definirea vederilor (comenzile CREATE VIEW, ALTER VIEW, DROP VIEW); - manipularea datelor (comenzile INSERT, UPDATE, DELETE); - interogarea datelor (comanda SELECT); - definirea restricţiilor de integritate (clauza CONSTRAINT folosită la definirea tabelelor) - autorizarea accesului la date (printr-un set de privilegii de sistem şi la nivel de obiect); - delimitarea tranzacţiilor (operaţiile COMMIT şi ROLLBACK).

Rg. 6: Regula de actualizare a tabelelor virtuale (vederilor) Un SGBD trebuie să poată determina dacă o vedere poate fi actualizată sau nu.

Rg. 7: Regula manipulării datelor Un sistem relaţional trebuie să ofere posibilitatea procesării tabelelor (de bază sau virtuale) nu numai în operaţiile de interogare a datelor cât şi în cele de inserare, actualizare şi ştergere.

Rg. 8: Regula independenţei fizice a datelor .Un SGBDR trebuie să separe complet aspectele de ordin fizic ale datelor (modul de stocare şi modul de acces la date) de cele de ordin logic.

Rg. 9: Regula independenţei logice a datelor .

Rg. 10: Regula independenţei datelor din punctul de vedere al integrităţii.

Rg. 11: Regula independenţei datelor din punctul de vedere al distribuirii .

Rg. 12: Regula privind prelucrarea datelor de către un limbaj de nivel inferior .

Rg. 0: Regula de bază Un SGBD Relaţional trebuie să fie capabil să gestioneze BD exclusiv pe baza caracteristicilor sale relaţionale. Această regulă are rolul de a rezuma concluziile desprinse din celelalte reguli.

Nici unul dintre SGBD-urile actuale nu satisface în totalitate toate cele 13 reguli ale lui Codd. De aceea, in practică, pentru a putea fi considerat relaţional, un SGBD trebuie să îndeplinească un set minimal de cerinţe. Un SGDB se numeşte minimal relaţional dacă satisface următoarele condiţii:

● Toate datele din cadrul bazei de date sunt reprezentate prin valori în tabele.

● Nu există pointeri observabili de către utilizator între tabele.

● Sistemul asigură operatorii relaţionali de proiecţie, selecţie şi compunere naturală, fără limitări impuse de considerente interne. Un SGDB se numeşte complet relaţional dacă este minimal relaţional şi satisface în plus următoarele condiţii:

● Sistemul asigură toate operaţiile de bază ale algebrei relaţionale, fără limitări impuse de considerente interne.

● Sistemul asigură restricţiile de integritate de bază ale modelului relaţional (integritatea entităţii şi integritatea referenţială). Un SGDB definit prin regulile lui Codd este un SGDB relaţional ideal.

**7.2. Aplicații:**

select \*from PRODUSE;

CREATE TABLE PRODUSE (

Id\_prod NUMBER(7) PRIMARY KEY,

Nume VARCHAR2(40) NOT NULL,

Descriere VARCHAR2(80));

CREATE TABLE FURNIZORI (

Id\_furn NUMBER(7) PRIMARY KEY,

Nume VARCHAR2(20),

Adresa VARCHAR2(60),

Tel NUMBER(11),

Email VARCHAR2(30),

SiTE VARCHAR2(30),

Oras VARCHAR2(20));

CREATE TABLE CLIENTI\_FIDELI (

Id\_client NUMBER(7) PRIMARY KEY,

Nume VARCHAR2(50),

Prenume VARCHAR2(50),

Adresa VARCHAR2(70),

Tel NUMBER(11),

Email VARCHAR2(30));

CREATE TABLE DEPOZITE (

Id\_dep NUMBER(7) PRIMARY KEY,

Nume VARCHAR2(30),

Adresa VARCHAR2(70),

Tel NUMBER(11),

Email VARCHAR2(30),

Sitee VARCHAR2(30));

CREATE TABLE DEPOZITE\_PRODUSE(

Id\_dp NUMBER (7) PRIMARY KEY,

Id\_dep NUMBER(7),

Id\_prod NUMBER(7),

FOREIGN KEY (Id\_dep) REFERENCES DEPOZITE (Id\_dep) on delete set null,

FOREIGN KEY (Id\_prod) REFERENCES PRODUSE (Id\_prod) on delete set null

);

CREATE TABLE FURNIZORI\_PRODUSE(

Id\_fp NUMBER (7) PRIMARY KEY,

Id\_furn NUMBER(7),

Id\_prod NUMBER(7),

FOREIGN KEY (Id\_furn) REFERENCES FURNIZORI (Id\_furn) on delete set null,

FOREIGN KEY (Id\_prod) REFERENCES PRODUSE (Id\_prod) on delete set null

);

create table FARMACI (

Id\_farm number(7) PRIMARY KEY,

Nume varchar2(40),

Adresa varchar2 (70),

Sitee varchar2(30),

Id\_manager number(7),

Telefon number(11),

Email varchar2 (30) );

CREATE TABLE ANGAJATI (

Id\_ang NUMBER(7) PRIMARY KEY,

Nume varchar2(30),

Prenume varchar2 (20),

Data\_nast date,

Functie varchar2(20),

Salariu number(7),

Data\_angaj date,

Id\_farm number(7),

Adresa varchar2(60),

Tel number(11),

Email varchar(50),

FOREIGN KEY (Id\_farm) REFERENCES FARMACI (Id\_farm) on delete set null

);

create table EVIDENTA\_APROVIZIONARE (

Id\_evap number(7) PRIMARY KEY,

Data\_evap date,

Id\_prod number(7),

Id\_furn number(7),

Id\_dep number(7),

Pret\_cump number(30),

Nr\_factura number(30),

Cantitate number(20),

Id\_ang\_resp number(7),

FOREIGN KEY (Id\_prod) REFERENCES PRODUSE (Id\_prod) on delete set null,

FOREIGN KEY (Id\_furn) REFERENCES FURNIZORI (Id\_furn) on delete set null,

FOREIGN KEY (Id\_dep) REFERENCES DEPOZITE (Id\_dep) on delete set null

);

create table EVIDENTA\_VANZARII (

Id\_evvanz number(7) PRIMARY KEY,

Data\_evvanz date,

Id\_client number(7),

Nr\_factura number(30),

Id\_prod number(7),

Pret\_vanz number(30),

Discont number(20),

Cantitate number(20),

Id\_ang\_resp number(7),

FOREIGN KEY (Id\_client) REFERENCES CLIENTI\_FIDELI (Id\_client) on delete set null,

FOREIGN KEY (Id\_prod) REFERENCES PRODUSE (Id\_prod) on delete set null

);

CREATE TABLE PROGRAM\_LUCRU (

Id\_prl number(7) PRIMARY KEY,

Data\_prl date,

Tura number(7),

Id\_ang number(7),

FOREIGN KEY (Id\_ang) REFERENCES ANGAJATI (Id\_ang) on delete set null

);

SELECT \*FROM FARMACI;

INSERT INTO FARMACI VALUES (1, 'FARMACIE1', 'STRADA NICOLAE IORGA ,SLATINA,OLT,BL 9','FARMACIE1.RO',1,0249556894,'FARMACIE1@GMAIL.COM');

INSERT INTO FARMACI VALUES (2, 'FARMACIE2', 'STRADA ALEEA ROLEZOR,SLATINA,OLT, BL 1','FARMACIE2.RO',2,0249556457,'FARMACIE2@GMAIL.COM');

INSERT INTO FARMACI VALUES (3, 'FARMACIE3', 'STRADA ALEEA LALELELOR,SLATINA,OLT,BL 7','FARMACIE3.RO',3,0349556894,'FARMACIE3@GMAIL.COM');

INSERT INTO FARMACI VALUES (4, 'FARMACIE4', 'STRADA AL.I.CUZA,SLATINA,OLT,BL 5','FARMACIE4.RO',4,0765556894,'FARMACIE4@GMAIL.COM');

INSERT INTO FARMACI VALUES (5, 'FARMACIE5', 'STRADA PRIMAVERRII,SLATINA,OLT BL 2','FARMACIE5.RO',5,0723495894,'FARMACIE5@GMAIL.COM');

INSERT INTO ANGAJATI VALUES (1, 'POPA', 'FLORIN',DATE '1995-01-01','MANAGER',5000,DATE '2018-01-01',1,'Strada Verde,bl9,ap21,Slatina,Olt',0723464385,'popa.flo@gmail.com');

INSERT INTO ANGAJATI VALUES (2, 'PANA', 'SORIN',DATE '1996-02-01','MANAGER',2000,DATE '2018-04-05',2,'Strada Lalele,b9,ap23,Slatina,Olt',0763468385,'pana.sorinel@gmail.com');

INSERT INTO ANGAJATI VALUES (3, 'PLOPEANU', 'BIANCA',DATE '1994-03-03','MANAGER',4000,DATE '2019-02-02',3,'Strada Fagului,bl3,ap22,Slatina,Olt',07637683845,'plopeanu.bibi@gmail.com');

INSERT INTO ANGAJATI VALUES (4, 'MARCEL', 'GEORGE',DATE '1992-07-06','MANAGER',5000,DATE '2020-03-03',4,'Strada Fantanii,bl4,ap19,Slatina,Olt',0763347385,'marcel@gmail.com');

INSERT INTO ANGAJATI VALUES (5, 'ANDREI', 'VIOREL',DATE '1990-05-08','MANAGER',3000,DATE '2020-04-05',5,'Strada Primaverii,bl5,ap15,Slatina,Olt',0765468381,'andrei.vio@gmail.com');

INSERT INTO ANGAJATI VALUES (6, 'ADAM', 'FLORIN',DATE '1992-01-01','ECONOMIST',3000,DATE '2018-01-01',1,'Strada Verde,bl9,ap21,Slatina,Olt',0723464385,'adam.flo@gmail.com');

INSERT INTO ANGAJATI VALUES (7, 'PANA', 'CLAUDIU',DATE '1995-02-01','ECONOMIST',3000,DATE '2018-04-05',2,'Strada Lalele,b9,ap23,Slatina,Olt',0763468385,'pana.claudiu@gmail.com');

INSERT INTO ANGAJATI VALUES (8, 'PLOPEAN', 'ANDRA',DATE '1992-03-03','ECONOMIST',3000,DATE '2019-02-02',3,'Strada Fagului,bl3,ap22,Slatina,Olt',07637683845,'plopean.andra@gmail.com');

INSERT INTO ANGAJATI VALUES (9, 'POPA', 'ANDREI',DATE '1993-07-06','ECONOMIST',3000,DATE '2020-03-03',4,'Strada Fantanii,bl4,ap19,Slatina,Olt',0763347385,'popa.andrei@gmail.com');

INSERT INTO ANGAJATI VALUES (10, 'DIACONU', 'CRISTIAN',DATE '1991-05-08','ECONOMIST',3000,DATE '2020-04-05',5,'Strada Primaverii,bl5,ap15,Slatina,Olt',0765468381,'doacpnu.cristi@gmail.com');

INSERT INTO ANGAJATI VALUES (11, 'DUMITRU', 'FLORIN',DATE '1992-03-03','APROVIZIONARE',3000,DATE '2018-01-01',1,'Strada Cnetrala,bl9,ap1,Slatina,Olt',0723460005,'dumi.flo@gmail.com');

INSERT INTO ANGAJATI VALUES (12, 'PUSCASU', 'CLAUDIU',DATE '1995-04-06','APROVIZIONARE',3000,DATE '2018-04-05',2,'Strada Primaverii,b9,ap2,Slatina,Olt',0767908385,'puscasu.claudiu@gmail.com');

INSERT INTO ANGAJATI VALUES (13, 'PLOP', 'ANDRA',DATE '1992-03-12','APROVIZIONARE',3000,DATE '2019-02-02',3,'Strada Verii,b3,ap4,Slatina,Olt',0763747685,'plop.andra@gmail.com');

INSERT INTO ANGAJATI VALUES (14, 'POPA', 'MONICA',DATE '1993-12-12','APROVIZIONARE',3000,DATE '2020-03-03',4,'Strada Ciorilor,b4,ap5,Slatina,Olt',0763015385,'popa.moni@gmail.com');

INSERT INTO ANGAJATI VALUES (15, 'DIACONU', 'LORENA',DATE '1991-06-18','APROVIZIONARE',3000,DATE '2020-04-05',5,'Strada Cepari,bl,ap15,Slatina,Olt',0760046831,'doaconu.lore@gmail.com');

INSERT INTO ANGAJATI VALUES (16, 'DIACONU', 'ILIE',DATE '1998-01-03','VANZATOR',2500,DATE '2018-01-01',1,'Strada Garii,bl9,ap1,Slatina,Olt',0723580005,'diaconu.ilie@gmail.com');

INSERT INTO ANGAJATI VALUES (17, 'PUSCASU', 'MARIANA',DATE '1998-04-15','VANZATOR',2500,DATE '2018-04-05',2,'Strada Crucii,bl 4,ap21,Slatina,Olt',0767000385,'mariana@gmail.com');

INSERT INTO ANGAJATI VALUES (18, 'POPA', 'IULIAN',DATE '1998-03-19','VANZATOR',2500,DATE '2019-02-02',3,'Strada Viorelelor,b1 2,ap1,Slatina,Olt',0763001685,'iuli.popa@gmail.com');

INSERT INTO ANGAJATI VALUES (19, 'ELENA', 'ANDREEA',DATE '1998-12-22','VANZATOR',2500,DATE '2020-03-03',4,'Strada Tiganilor,bl 1,ap4,Slatina,Olt',0760015385,'elena.andreea@gmail.com');

INSERT INTO ANGAJATI VALUES (20, 'DRAGAN', 'ANDREI',DATE '1998-06-27','VANZATOR',2500,DATE '2020-04-05',5,'Strada Italienilor,bl 2,ap15,Slatina,Olt',0760046831,'dragan.andrei@gmail.com');

INSERT INTO ANGAJATI VALUES (21, 'BURTA', 'MARI',DATE '1989-06-03','IT-IST',4500,DATE '2018-01-01',1,'Strada Farmaciei,bl9,ap1,Slatina,Olt',0765580005,'burta.mariandiaconu.ilie@gmail.com');

INSERT INTO ANGAJATI VALUES (22, 'MARIO', 'GEORGE',DATE '1997-06-15','IT-IST',4500,DATE '2018-04-05',2,'Strada Clopolnitei,bl 4,ap21,Slatina,Olt',0787007385,'mario.geo@gmail.com');

INSERT INTO ANGAJATI VALUES (23, 'GIURA', 'MARIUS',DATE '1998-02-19','IT-IST',4500,DATE '2019-02-02',3,'Strada Beicii,b1 2,ap1,Slatina,Olt',0723001685,'giura.marius@gmail.com');

INSERT INTO ANGAJATI VALUES (24, 'TEACA', 'OVIDIU',DATE '1993-5-22','IT-IST',4500,DATE '2020-03-03',4,'Strada Mugurilor,bl 1,ap4,Slatina,Olt',0760002385,'teaca.ovi@gmail.com');

INSERT INTO ANGAJATI VALUES (25, 'VUVA', 'COSMIN',DATE '1997-01-27','IT-IST',4500,DATE '2020-04-05',5,'Strada Nicolae Iorga,bl 2,ap15,Slatina,Olt',0760046691,'vuvix.cosmin@gmail.com');

INSERT INTO ANGAJATI VALUES (26, 'MARIUS', 'MARIAN',DATE '1989-06-03','SOFER',4500,DATE '2018-01-01',1,'Strada Farma,bl9,ap1,Slatina,Olt',0725580005,'marius@gmail.com');

INSERT INTO ANGAJATI VALUES (27, 'MARCEL', 'GEORGE',DATE '1997-06-15','SOFER',4500,DATE '2018-04-05',2,'Strada Clopot,bl 4,ap21,Slatina,Olt',0727007385,'marcel@gmail.com');

INSERT INTO ANGAJATI VALUES (28, 'OVIDIU', 'MARIUS',DATE '1998-02-19','SOFER',4500,DATE '2019-02-02',3,'Strada Bei,b1 2,ap1,Slatina,Olt',0783001685,'ovidiu.mari@gmail.com');

INSERT INTO ANGAJATI VALUES (29, 'SCORBUREANU', 'ALEX',DATE '1993-5-22','SOFER',4500,DATE '2020-03-03',4,'Strada Mugur,bl 1,ap4,Slatina,Olt',0720002385,'alex@gmail.com');

INSERT INTO ANGAJATI VALUES (30, 'ANA', 'MARIA',DATE '1997-01-27','SOFER',4500,DATE '2020-04-05',5,'Strada Cuza Voda,bl 2,ap15,Slatina,Olt',0720046691,'maria.cosmin@gmail.com');

SELECT \*FROM PRODUSE;

INSERT INTO PRODUSE VALUES (1, 'ANTIINFLAMATOARE','Reduc inflamația(roșeață,umflătura,durerea)');

INSERT INTO PRODUSE VALUES (2, 'ANTIBIOTICE', 'Implicate în tratamentul infecțiilor bacteriene');

INSERT INTO PRODUSE VALUES (3, 'SUPLIMENTE-ALIMENTARE', 'Au în compoziția lor micro-sau macronutrienți');

INSERT INTO PRODUSE VALUES (4, 'ANTIDEPRESIVE','Antidepresivele vizează simptomele depresiei');

INSERT INTO PRODUSE VALUES (5, 'ANTISEPTICE','Actioneaza pe suprafete in afara organismului');

INSERT INTO FURNIZORI VALUES (1, 'FURNIZOR-1','Strada Clopolnitei,bl 4,ap21,Olt',0723474186,'FURNIZOR1@GMAIL.COM','WWW.FURNIZOR1.RO','Slatina');

INSERT INTO FURNIZORI VALUES (2, 'FURNIZOR-2','Strada Bei,b1 2,ap1,Olt',0723400086,'FURNIZOR2@GMAIL.COM','WWW.FURNIZOR2.RO','Slatina');

INSERT INTO FURNIZORI VALUES (3, 'FURNIZOR-3','Strada Nicolae Iorga,bl 2,ap15,Olt',0763470186,'FURNIZOR3@GMAIL.COM','WWW.FURNIZOR3.RO','Slatina');

INSERT INTO FURNIZORI VALUES (4, 'FURNIZOR-4','Strada Cepari,bl,ap15,Olt',0723004186,'FURNIZOR4@GMAIL.COM','WWW.FURNIZOR4.RO','Slatina');

INSERT INTO FURNIZORI VALUES (5, 'FURNIZOR-5','Strada Viorelelor b1 2 ap1 Olt',0720004186,'FURNIZOR5@GMAIL.COM','WWW.FURNIZOR5.RO','Slatina');

INSERT INTO FURNIZORI\_PRODUSE VALUES (1,1,2);

INSERT INTO FURNIZORI\_PRODUSE VALUES (2,2,1);

INSERT INTO FURNIZORI\_PRODUSE VALUES (3,3,3);

INSERT INTO FURNIZORI\_PRODUSE VALUES (4,4,5);

INSERT INTO FURNIZORI\_PRODUSE VALUES (5,5,4);

INSERT INTO DEPOZITE VALUES (1,'DEPOZIT\_1','Strada Bisericii nr 1',0768412679,'Depozit1@email.ro','www.depozit1.ro');

INSERT INTO DEPOZITE VALUES (2,'DEPOZIT\_2','Strada Ioan Cuza nr 1',0768416509,'Depozit2@email.ro','www.depozit2.ro');

INSERT INTO DEPOZITE VALUES (3,'DEPOZIT\_3','Strada Minulescu nr 3',0760012679,'Depozit3@email.ro','www.depozit3.ro');

INSERT INTO DEPOZITE VALUES (4,'DEPOZIT\_4','Strada Verde nr 1',0768448279,'Depozit4@email.ro','www.depozit4.ro');

INSERT INTO DEPOZITE VALUES (5,'DEPOZIT\_5','Strada Centrala nr 4',0760002679,'Depozit5@email.ro','www.depozit5.ro');

select \*from DEPOZITE;

SELECT \*FROM FURNIZORI;

select \*from DEPOZITE\_PRODUSE;

INSERT INTO DEPOZITE\_PRODUSE VALUES (1,2,1);

INSERT INTO DEPOZITE\_PRODUSE VALUES (2,1,2);

INSERT INTO DEPOZITE\_PRODUSE VALUES (3,3,4);

INSERT INTO DEPOZITE\_PRODUSE VALUES (4,4,5);

INSERT INTO DEPOZITE\_PRODUSE VALUES (5,5,3);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (1,DATE'2018-01-01',1,2,1,70000,23600026791,500,11);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (2,DATE'2018-04-05',2,1,2,60000,34564782902,400,11);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (3,DATE' 2019-02-02',3,4,5,70000,23600026873,300,11);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (4,DATE'2020-03-03',4,3,3,60000,34564782914,600,11);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (5,DATE'2020-04-05',5,5,4,70000,23600026885,500,11);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (6,DATE'2018-01-01',1,2,1,70000,2360002679,500,12);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (7,DATE'2018-04-05',2,1,2,60000,3456478290,400,12);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (8,DATE' 2019-02-02',3,4,5,70000,2360002687,300,12);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (9,DATE'2020-03-03',4,3,3,60000,3456478291,600,12);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (10,DATE'2020-04-05',5,5,4,70000,2360002688,500,12);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (11,DATE'2018-01-01',1,2,1,70000,23600026796,500,13);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (12,DATE'2018-04-05',2,1,2,60000,34564782907,400,13);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (13,DATE' 2019-02-02',3,4,5,70000,23600026878,300,13);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (14,DATE'2020-03-03',4,3,3,60000,34564782919,600,13);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (15,DATE'2020-04-05',5,5,4,70000,23600026880,500,13);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (16,DATE'2018-01-01',1,2,1,70000,23602002679,500,14);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (17,DATE'2018-04-05',2,1,2,60000,34564478290,400,14);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (18,DATE' 2019-02-02',3,4,5,70000,23600502687,300,14);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (19,DATE'2020-03-03',4,3,3,60000,34567478291,600,14);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (20,DATE'2020-04-05',5,5,4,70000,23630002688,500,14);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (21,DATE'2018-01-01',1,2,1,70000,23610002679,500,15);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (22,DATE'2018-04-05',2,1,2,60000,34565478290,400,15);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (23,DATE' 2019-02-02',3,4,5,70000,23608002687,300,15);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (24,DATE'2020-03-03',4,3,3,60000,34564978291,600,15);

INSERT INTO EVIDENTA\_APROVIZIONARE VALUES (25,DATE'2020-04-05',5,5,4,70000,23600002688,500,15);

SELECT \*FROM EVIDENTA\_APROVIZIONARE;

INSERT INTO CLIENTI\_FIDELI VALUES (1,'POPA','DANUT','STR LIBERTATII NR1 BL8 AP12',0765486213,'POPA.DAN@GMAIL.COM');

INSERT INTO CLIENTI\_FIDELI VALUES (2,'ANDREI','MARIUS','STR VIORELELOR NR2 BL1 AP1',0249658925,'ANDREI@GMAL.COM');

INSERT INTO CLIENTI\_FIDELI VALUES (3,'MONICA','ELENA','STR NICOLAE IORGA NR7 BL7,AP1',0768413015,'ELENA@GMAIL.COM');

INSERT INTO CLIENTI\_FIDELI VALUES (4,'POPESCU','ION','STR VERDE NR2 BL2 AP2',0768543015,'POPESCU.ION@GMAIL.COM');

INSERT INTO CLIENTI\_FIDELI VALUES (5,'ION','ALEXANDRU','STR VESELIEI NR3 BL3 AP3',0765850015,'ION.ALEX@GMAIL.COM');

SELECT \*FROM CLIENTI\_FIDELI;

INSERT INTO EVIDENTA\_VANZARII VALUES (1,DATE'2019-01-01',1,2435,1,70,15,2,16);

INSERT INTO EVIDENTA\_VANZARII VALUES (2,DATE'2019-12-12',2,2437,2,20,20,3,16);

INSERT INTO EVIDENTA\_VANZARII VALUES (3,DATE'2019-02-03',3,2685,3,50,20,5,16);

INSERT INTO EVIDENTA\_VANZARII VALUES (4,DATE'2019-06-12',4,2987,4,60,15,1,16);

INSERT INTO EVIDENTA\_VANZARII VALUES (5,DATE'2019-06-04',5,2478,5,90,10,1,16);

INSERT INTO EVIDENTA\_VANZARII VALUES (6,DATE'2019-01-02',1,2435,1,70,15,2,17);

INSERT INTO EVIDENTA\_VANZARII VALUES (7,DATE'2019-12-13',2,2437,2,20,20,3,17);

INSERT INTO EVIDENTA\_VANZARII VALUES (8,DATE'2019-02-04',3,2685,3,50,20,5,17);

INSERT INTO EVIDENTA\_VANZARII VALUES (9,DATE'2019-07-12',4,2987,4,60,15,1,17);

INSERT INTO EVIDENTA\_VANZARII VALUES (10,DATE'2019-07-04',5,2478,5,90,10,1,17);

INSERT INTO EVIDENTA\_VANZARII VALUES (11,DATE'2019-02-02',1,2435,1,70,15,2,18);

INSERT INTO EVIDENTA\_VANZARII VALUES (12,DATE'2019-12-15',2,2437,2,10,20,3,18);

INSERT INTO EVIDENTA\_VANZARII VALUES (13,DATE'2019-02-06',3,2685,3,50,20,5,18);

INSERT INTO EVIDENTA\_VANZARII VALUES (14,DATE'2019-07-18',4,2987,4,60,15,1,18);

INSERT INTO EVIDENTA\_VANZARII VALUES (15,DATE'2019-07-08',5,2478,5,90,10,1,18);

INSERT INTO EVIDENTA\_VANZARII VALUES (16,DATE'2019-03-02',1,24535,1,50,15,2,19);

INSERT INTO EVIDENTA\_VANZARII VALUES (17,DATE'2019-12-18',2,24537,2,20,15,3,19);

INSERT INTO EVIDENTA\_VANZARII VALUES (18,DATE'2019-03-06',3,26585,3,50,10,4,19);

INSERT INTO EVIDENTA\_VANZARII VALUES (19,DATE'2019-08-18',4,29587,4,60,15,2,19);

INSERT INTO EVIDENTA\_VANZARII VALUES (20,DATE'2019-08-18',5,24758,5,60,10,1,19);

INSERT INTO EVIDENTA\_VANZARII VALUES (21,DATE'2019-04-02',1,245135,1,70,15,2,20);

INSERT INTO EVIDENTA\_VANZARII VALUES (22,DATE'2019-12-25',2,241537,2,20,15,3,20);

INSERT INTO EVIDENTA\_VANZARII VALUES (23,DATE'2019-06-16',3,261585,3,30,10,4,20);

INSERT INTO EVIDENTA\_VANZARII VALUES (24,DATE'2019-09-18',4,291587,4,60,15,2,20);

INSERT INTO EVIDENTA\_VANZARII VALUES (25,DATE'2019-09-20',5,247158,5,50,10,1,20);

select \*from EVIDENTA\_VANZARII;

INSERT INTO PROGRAM\_LUCRU VALUES (1,DATE'2019-04-02',1,1);

INSERT INTO PROGRAM\_LUCRU VALUES (2,DATE'2019-04-02',1,6);

INSERT INTO PROGRAM\_LUCRU VALUES (3,DATE'2019-04-02',1,11);

INSERT INTO PROGRAM\_LUCRU VALUES (4,DATE'2019-04-02',1,16);

INSERT INTO PROGRAM\_LUCRU VALUES (5,DATE'2019-04-02',1,21);

SELECT\*FROM PROGRAM\_LUCRU;

Tabel Farmacii:Angajatii de la farmacii sunt

-Farmacie1:id\_ang=1,6,11,16,21,26

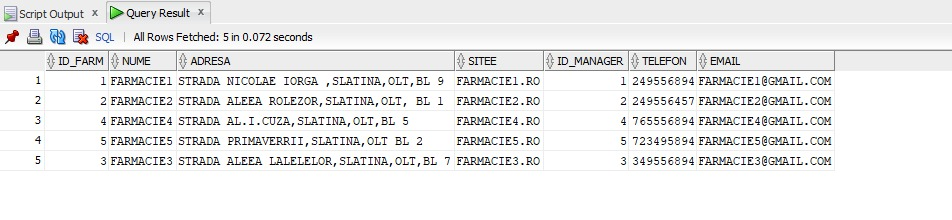
-Farmacie2:id\_ang=2,7,12,17,22,27

-Farmacie3:id\_ang=3,8,13,18,23,28

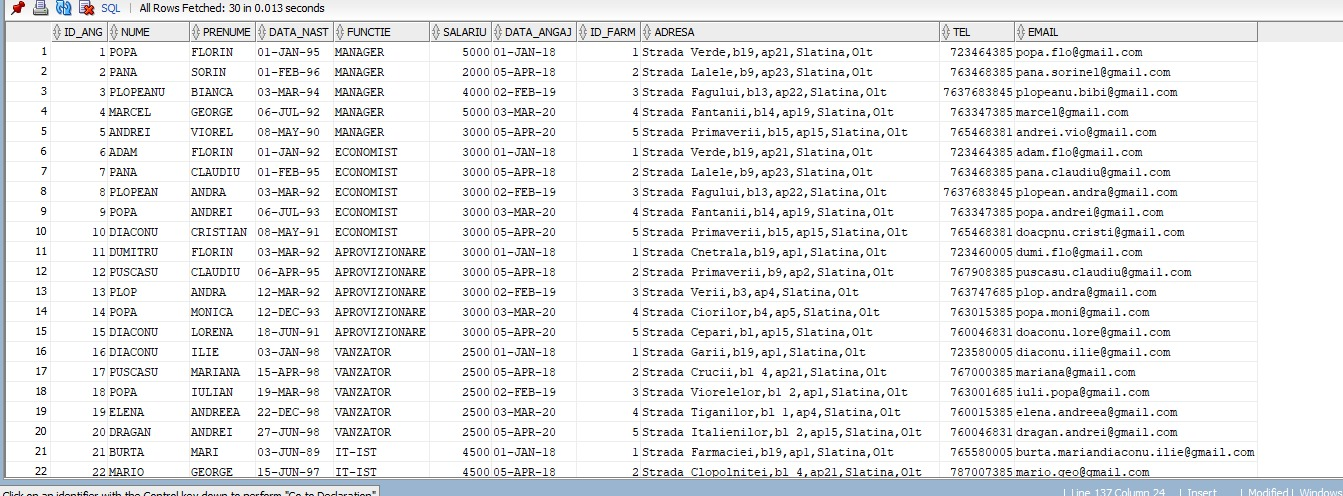
-Farmacie4:id\_ang=4,9,14,19.24,29

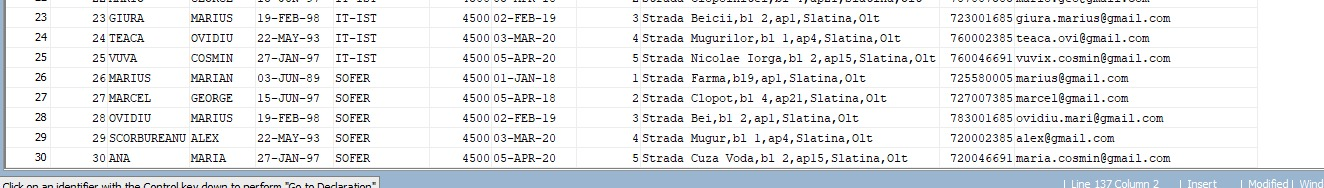
-Farmacie5:id\_ang=5,10,15,20,25,30

Aceste Farmacii fac parte din acelasi lant ,prin urmare,acestea se aprovizioneaza de la aceeasi furnizori cu aceleasi produse si din aceleasi depozite.

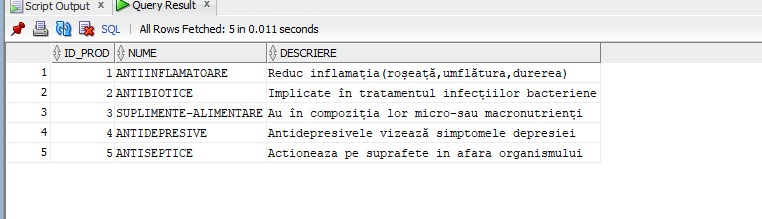


Tabel Angajati:

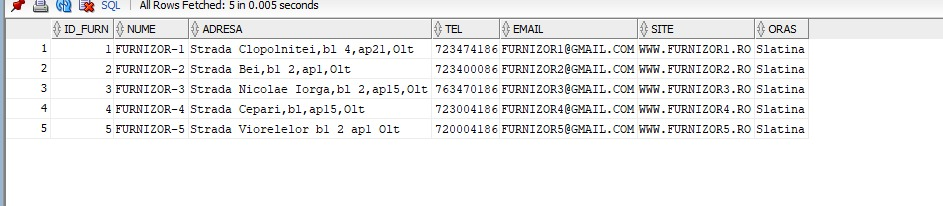




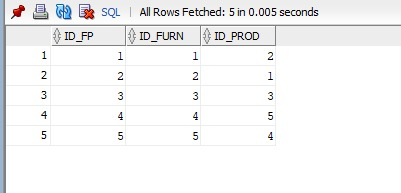
Tabel Produse:Fiecare farmacie are aceste produse.



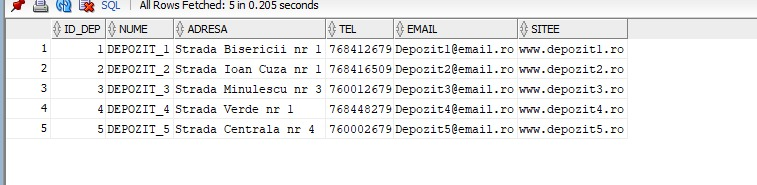
Tabel Furnizori



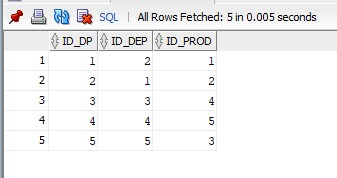
Tabel Furnizori\_Produse



Tabel Depozite

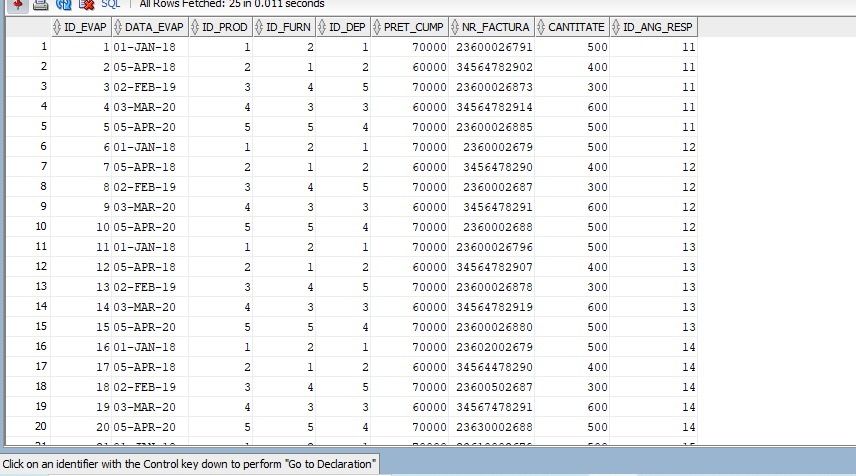


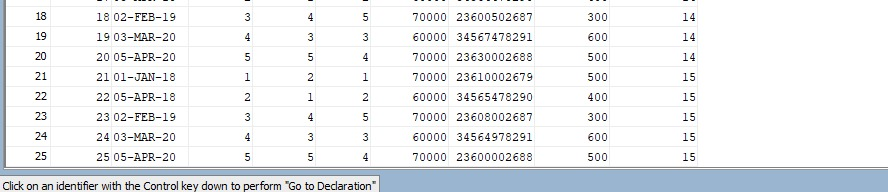
Tabel Depozite\_Produse



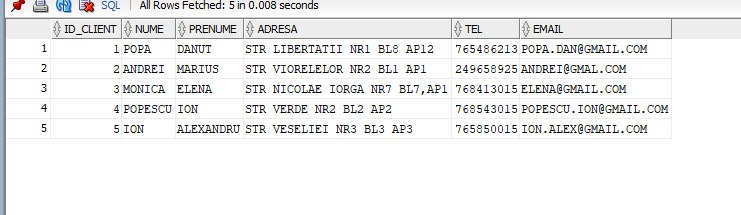
Tabel Evidenta\_Aprovizionare

Aprovizionarea este facuta la fiecare farmacie de catre angajatul cu acest post adica id.urile 11,12,13,14,15.Fiecare farmacie cumpara toate tipurile de produse.

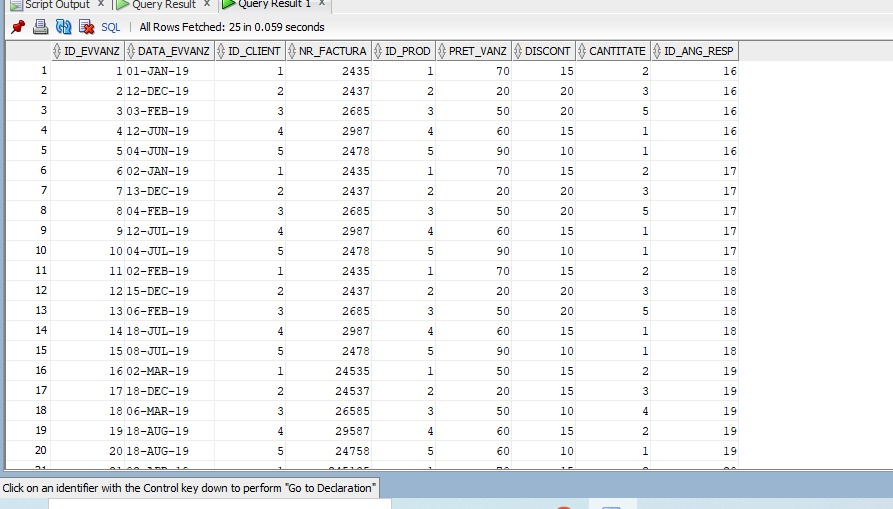


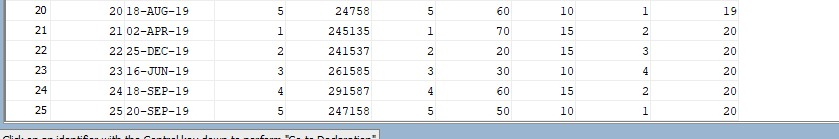


Tabelul Clienti\_Fideli:Am pus 5 clienti care frecventeaza toate cele 5 farmacii acestea aflandu-se in acelasi oras.Ei au carduri de fidelitate la toate 5.



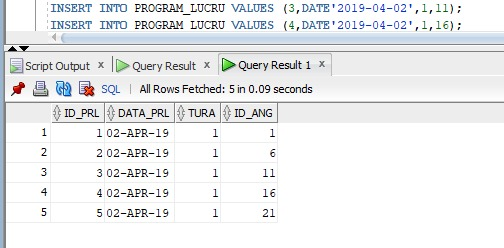
Tabelul Evidenta\_Vanzare





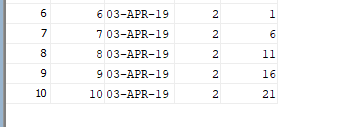
Tabelul Program\_Lucru:Acesta este programul de lucru de la Farmacia1 angajatii lucrand in 2 ture.Prima tura :De la 8 dimineata pana la ora 16

A doua tura:De la ora 16 pana la ora 00;



Angajatii 1,6,11,16,21 lucreaza in ziua de 2 aprilie in tura 1 iar in ziua de 3 aprilie in tura 2.La fiecare schimbare de tura sunt alti angajati care le ia locul.

A doua tura:



**CAPITOLUL VIII. CONCLUZII**

Considerăm că în acest proiect am parcurs toți pașii teoretici și practici ai creării unei baze de date plecând de la un scenariu și formândune atât ideile cât și acțiunile în conformitate cu cerințele impuse. Totodată s-au realizat și principalele obiective:

✓ Noțiuni generale

✓ Prezentarea aplicației

✓ Proiectarea structurii bazei de date

✓ Definirea relațiilor

✓ Operații cu tabele

✓ Operații cu datele tabelelor

✓ Crearea relațiilor între tabele

✓ Sortarea, filtrarea și indexarea datelor

**BIBLIOGRAFIE**

* **V. Stoian – “Note de curs”, 2020-2021**
* **MANCAŞ, C., Modelul relaţional al datelor. Editura Ovidius University Press, 2005.**
* **POPESCU, I., Modelarea bazelor de date. Editura Tehnicã, Bucureşti 2001.**
* **Jeff Perkins, Bryan Morgan – SQL fara profesor, in 14 zile, Editura Teora 1998.**