**PROIECT**

**CLINICĂ VETERINARĂ**

Student: Staicu Octavian-Florin

Grupa 506 Master Inginerie Software

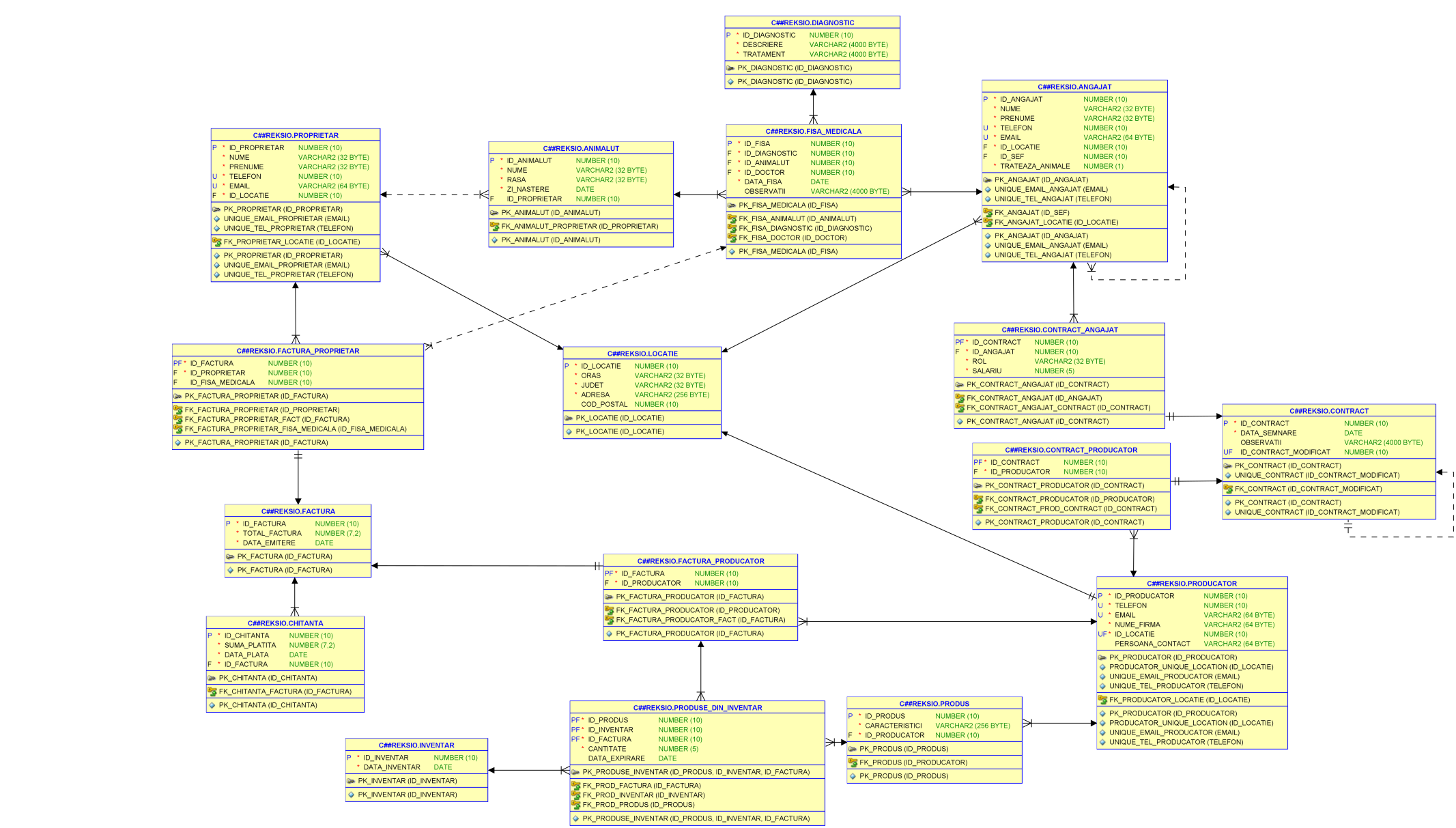
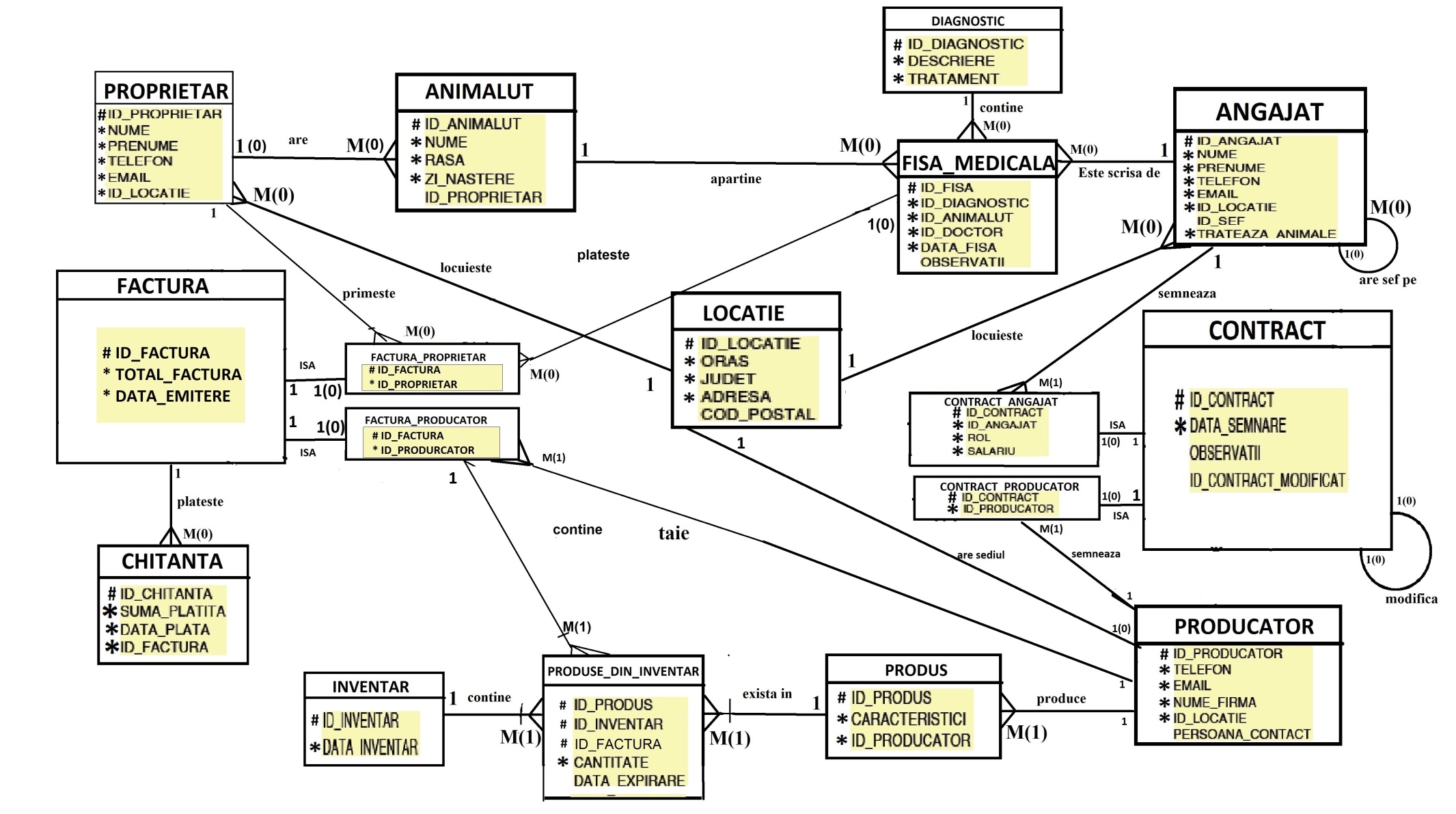
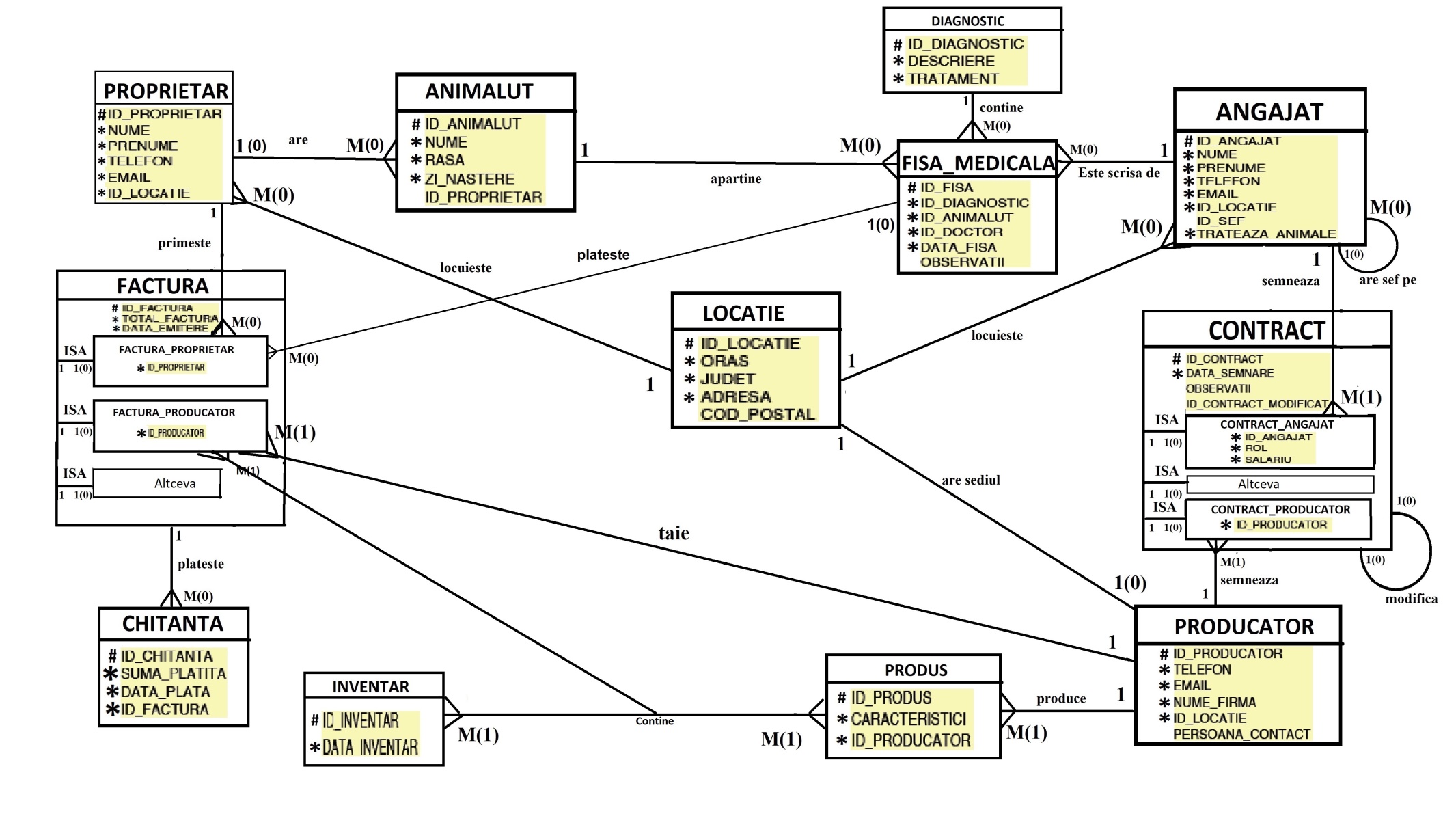
***Clinică veterinară***

Acest proiect se adresează unei clinici veterinare

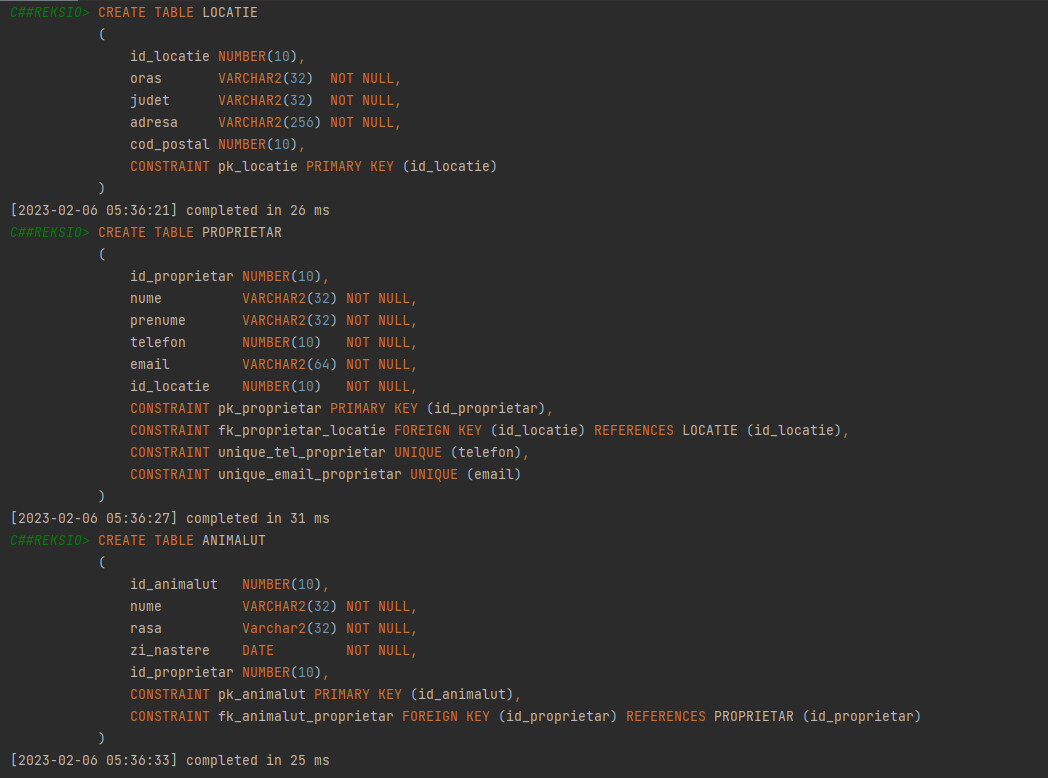
# Descrierea Regulilor de Bussines

* Un proprietar are mai multe animale. Când un animăluț este tratat la clinică, i se face o nouă fișă medicală (relație de tip 3 între animal, diagnostic și un angajat care are voie să trateze animale).  Pe fișa medicală apare un diagnostic, pentru care se știe tratamentul (sau se adaugă în tabela diagnostic). Un animăluț se poate găsi în baza de date fără să aibă o fișă medicală (spre exemplu, dacă doar s-a microcipat)
* Angajații au unul sau mai multe contracte de muncă (în caz că un angajat vrea să negocieze ulterior ore suplimentare sau măriri de salariu etc., se face un nou contract, care, eventual, poate modifica un contract anterior). Doar unii angajați au voie să trateze animale (precizat in atributul din tabela angajat – ex. doar doctorii și asistentele etc.; întrucât au mai multe contracte, am ales să includ acest atribut pentru a evita o căutare prin toate contractele semnate pentru a verifica dacă există unul care să îi permită să trateze animăluțe ).
* Clinica are mai multe produse: vaccinuri, medicamente, aparatură medicală, mâncare de animale, dezinfectanți etc. Produsele sunt produse de producători, care încheie contracte cu clinica. Produsele se vor găsi în inventarele care se fac o dată la o perioadă de timp îndelungată.
* Clinica emite facturi către proprietarii de animăluțe. Aceștia le pot achita eșalonat, deci există mai multe chitanțe pentru o singură factură. Pentru o fisa medicala se pot emite mai multe facturi.
* Producătorii de produse emit și ei facturi, pe care clinica trebuie să le plătească (la fel, eșalonat).

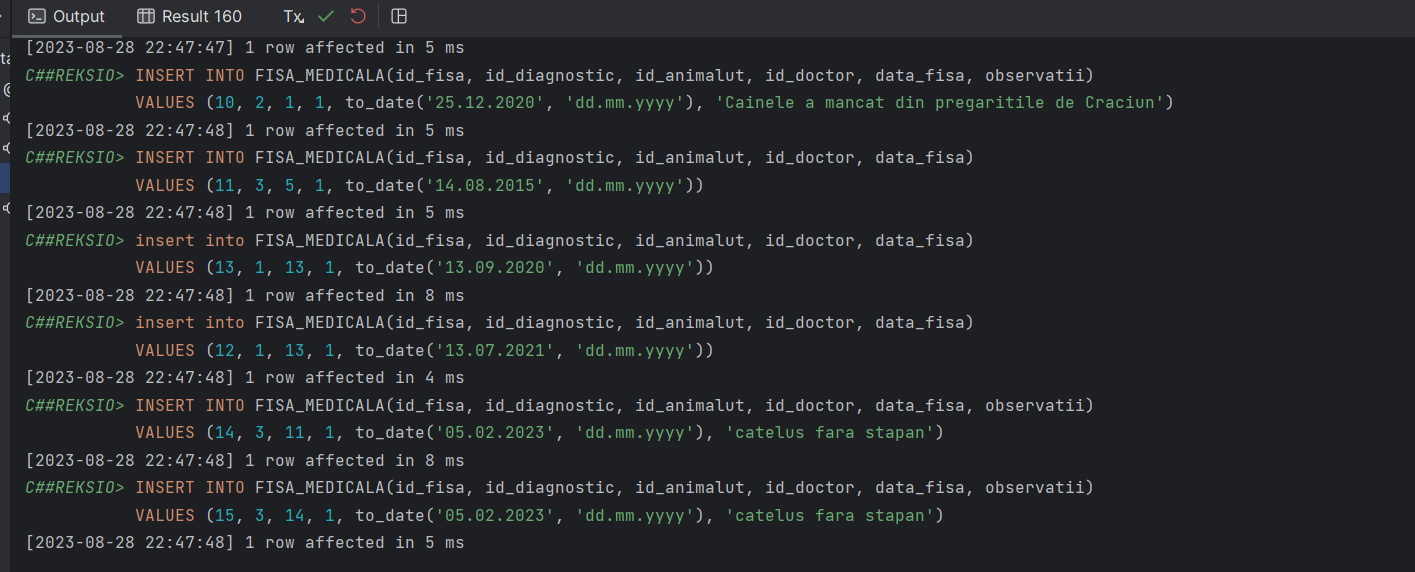
***Pe următoarele două pagini se află diagrama entitate relație și diagrama conceptuală.***

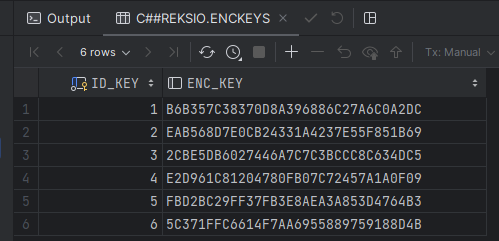
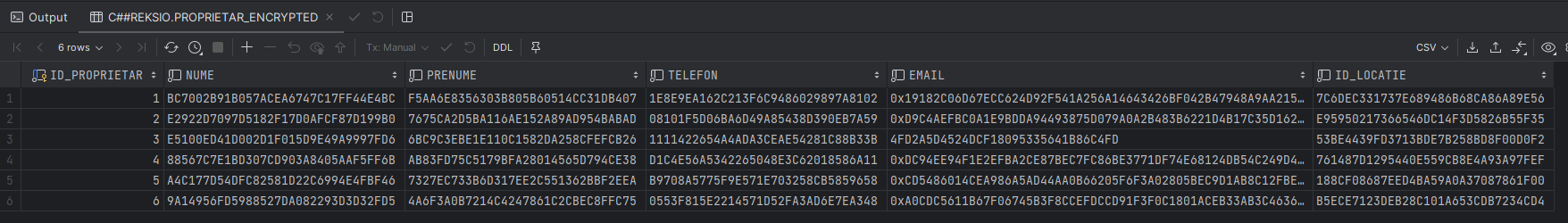
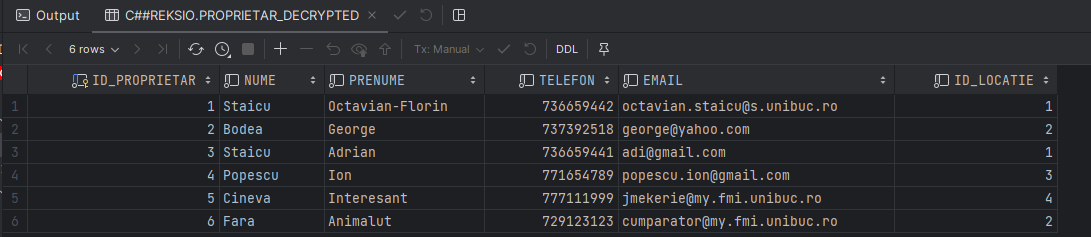
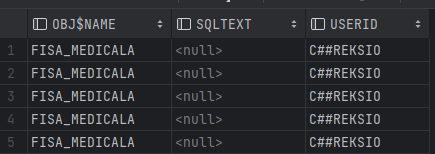
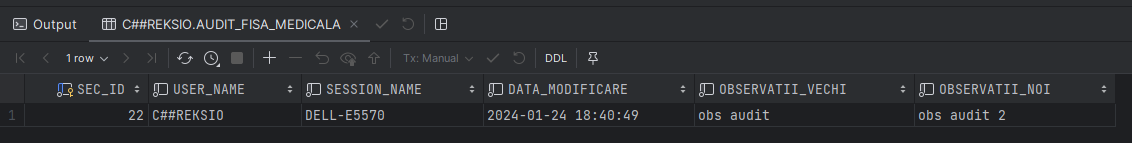


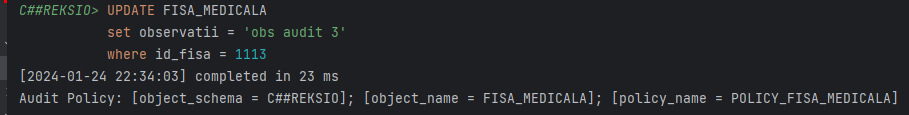
--6  
CREATE TABLE LOCATIE  
(  
 id\_locatie NUMBER(10),  
 oras VARCHAR2(32) NOT NULL,  
 judet VARCHAR2(32) NOT NULL,  
 adresa VARCHAR2(256) NOT NULL,  
 cod\_postal NUMBER(10),  
 CONSTRAINT pk\_locatie PRIMARY KEY (id\_locatie)  
);  
  
CREATE TABLE PROPRIETAR  
(  
 id\_proprietar NUMBER(10),  
 nume VARCHAR2(32) NOT NULL,  
 prenume VARCHAR2(32) NOT NULL,  
 telefon NUMBER(10) NOT NULL,  
 email VARCHAR2(64) NOT NULL,  
 id\_locatie NUMBER(10) NOT NULL,  
 CONSTRAINT pk\_proprietar PRIMARY KEY (id\_proprietar),  
 CONSTRAINT fk\_proprietar\_locatie FOREIGN KEY (id\_locatie) REFERENCES LOCATIE (id\_locatie),  
 CONSTRAINT unique\_tel\_proprietar UNIQUE (telefon),  
 CONSTRAINT unique\_email\_proprietar UNIQUE (email)  
);  
  
CREATE TABLE ANIMALUT  
(  
 id\_animalut NUMBER(10),  
 nume VARCHAR2(32) NOT NULL,  
 rasa Varchar2(32) NOT NULL,  
 zi\_nastere DATE NOT NULL,  
 id\_proprietar NUMBER(10),  
 CONSTRAINT pk\_animalut PRIMARY KEY (id\_animalut),  
 CONSTRAINT fk\_animalut\_proprietar FOREIGN KEY (id\_proprietar) REFERENCES PROPRIETAR (id\_proprietar)  
);  
  
CREATE TABLE DIAGNOSTIC  
(  
 id\_diagnostic NUMBER(10),  
 descriere VARCHAR2(4000) NOT NULL,  
 tratament VARCHAR2(4000) NOT NULL,  
 CONSTRAINT pk\_diagnostic PRIMARY KEY (id\_diagnostic)  
);  
  
CREATE TABLE ANGAJAT  
(  
 id\_angajat NUMBER(10),  
 nume VARCHAR2(32) NOT NULL,  
 prenume VARCHAR2(32) NOT NULL,  
 telefon NUMBER(10) NOT NULL,  
 email VARCHAR2(64) NOT NULL,  
 id\_locatie NUMBER(10) NOT NULL,  
 id\_sef NUMBER(10),  
 trateaza\_animale NUMBER(1) NOT NULL,  
 CONSTRAINT pk\_angajat PRIMARY KEY (id\_angajat),  
 CONSTRAINT fk\_angajat\_locatie FOREIGN KEY (id\_locatie) REFERENCES LOCATIE (id\_locatie),  
 CONSTRAINT fk\_angajat FOREIGN KEY (id\_sef) REFERENCES ANGAJAT (id\_angajat),  
 CONSTRAINT unique\_tel\_angajat UNIQUE (telefon),  
 CONSTRAINT unique\_email\_angajat UNIQUE (email)  
);  
  
CREATE TABLE FISA\_MEDICALA  
(  
 id\_fisa NUMBER(10),  
 id\_diagnostic NUMBER(10) NOT NULL,  
 id\_animalut NUMBER(10) NOT NULL,  
 id\_doctor NUMBER(10) NOT NULL,  
 data\_fisa DATE NOT NULL,  
 observatii VARCHAR2(4000),  
 CONSTRAINT pk\_fisa\_medicala PRIMARY KEY (id\_fisa),  
 CONSTRAINT fk\_fisa\_diagnostic FOREIGN KEY (id\_diagnostic) REFERENCES DIAGNOSTIC (id\_diagnostic),  
 CONSTRAINT fk\_fisa\_animalut FOREIGN KEY (id\_animalut) REFERENCES ANIMALUT (id\_animalut),  
 CONSTRAINT fk\_fisa\_doctor FOREIGN KEY (id\_doctor) REFERENCES ANGAJAT (id\_angajat)  
);  
  
CREATE TABLE PRODUCATOR  
(  
 id\_producator NUMBER(10),  
 telefon NUMBER(10) NOT NULL,  
 email VARCHAR2(64) NOT NULL,  
 nume\_firma VARCHAR2(64) NOT NULL,  
 id\_locatie NUMBER(10) NOT NULL,  
 persoana\_contact VARCHAR2(64),  
 CONSTRAINT pk\_producator PRIMARY KEY (id\_producator),  
 CONSTRAINT fk\_producator\_locatie FOREIGN KEY (id\_locatie) REFERENCES LOCATIE (id\_locatie),  
 CONSTRAINT producator\_unique\_location UNIQUE (id\_locatie),  
 CONSTRAINT unique\_tel\_producator UNIQUE (telefon),  
 CONSTRAINT unique\_email\_producator UNIQUE (email)  
);  
  
CREATE TABLE CONTRACT  
(  
 id\_contract NUMBER(10),  
 data\_semnare DATE NOT NULL,  
 observatii VARCHAR2(4000),  
 id\_contract\_modificat NUMBER(10),  
 CONSTRAINT pk\_contract PRIMARY KEY (id\_contract),  
 CONSTRAINT fk\_contract FOREIGN KEY (id\_contract\_modificat) REFERENCES CONTRACT (id\_contract),  
 CONSTRAINT unique\_contract UNIQUE (id\_contract\_modificat)  
);  
  
CREATE TABLE CONTRACT\_ANGAJAT  
(  
 id\_contract NUMBER(10),  
 id\_angajat NUMBER(10) NOT NULL,  
 rol VARCHAR2(32) NOT NULL,  
 salariu NUMBER(5) NOT NULL,  
 CONSTRAINT pk\_contract\_angajat PRIMARY KEY (id\_contract),  
 CONSTRAINT fk\_contract\_angajat\_contract FOREIGN KEY (id\_contract) REFERENCES CONTRACT (id\_contract),  
 CONSTRAINT fk\_contract\_angajat FOREIGN KEY (id\_angajat) REFERENCES ANGAJAT (id\_angajat)  
);  
  
CREATE TABLE CONTRACT\_PRODUCATOR  
(  
 id\_contract NUMBER(10),  
 id\_producator NUMBER(10) NOT NULL,  
 CONSTRAINT pk\_contract\_producator PRIMARY KEY (id\_contract),  
 CONSTRAINT fk\_contract\_prod\_contract FOREIGN KEY (id\_contract) REFERENCES CONTRACT (id\_contract),  
 CONSTRAINT fk\_contract\_producator FOREIGN KEY (id\_producator) REFERENCES PRODUCATOR (id\_producator)  
);  
  
CREATE TABLE FACTURA  
(  
 id\_factura NUMBER(10),  
 total\_factura NUMBER(7, 2) NOT NULL,  
 data\_emitere DATE NOT NULL,  
 CONSTRAINT pk\_factura PRIMARY KEY (id\_factura)  
);  
  
CREATE TABLE FACTURA\_PROPRIETAR  
(  
 id\_factura NUMBER(10),  
 id\_proprietar NUMBER(10) NOT NULL,  
 id\_fisa\_medicala NUMBER(10),  
 CONSTRAINT pk\_factura\_proprietar PRIMARY KEY (id\_factura),  
 CONSTRAINT fk\_factura\_proprietar FOREIGN KEY (id\_proprietar) REFERENCES PROPRIETAR (id\_proprietar),  
 CONSTRAINT fk\_factura\_proprietar\_fact FOREIGN KEY (id\_factura) REFERENCES FACTURA (id\_factura),  
 CONSTRAINT fk\_factura\_proprietar\_fisa\_medicala FOREIGN KEY (id\_fisa\_medicala) REFERENCES FISA\_MEDICALA (id\_fisa)  
);  
  
CREATE TABLE FACTURA\_PRODUCATOR  
(  
 id\_factura NUMBER(10),  
 id\_producator NUMBER(10) NOT NULL,  
 CONSTRAINT pk\_factura\_producator PRIMARY KEY (id\_factura),  
 CONSTRAINT fk\_factura\_producator FOREIGN KEY (id\_producator) REFERENCES PRODUCATOR (id\_producator),  
 CONSTRAINT fk\_factura\_producator\_fact FOREIGN KEY (id\_factura) REFERENCES FACTURA (id\_factura)  
);  
  
CREATE TABLE CHITANTA  
(  
 id\_chitanta NUMBER(10),  
 suma\_platita NUMBER(7, 2) NOT NULL,  
 data\_plata DATE NOT NULL,  
 id\_factura NUMBER(10) NOT NULL,  
 CONSTRAINT pk\_chitanta PRIMARY KEY (id\_chitanta),  
 CONSTRAINT fk\_chitanta\_factura FOREIGN KEY (id\_factura) references FACTURA (id\_factura)  
);  
  
  
CREATE TABLE PRODUS  
(  
 id\_produs NUMBER(10),  
 caracteristici VARCHAR2(256) NOT NULL,  
 id\_producator NUMBER(10) NOT NULL,  
 CONSTRAINT pk\_produs PRIMARY KEY (id\_produs),  
 CONSTRAINT fk\_produs FOREIGN KEY (id\_producator) REFERENCES PRODUCATOR (id\_producator)  
);  
  
CREATE TABLE INVENTAR  
(  
 id\_inventar NUMBER(10),  
 data\_inventar DATE NOT NULL,  
 CONSTRAINT pk\_inventar PRIMARY KEY (id\_inventar)  
);  
  
CREATE TABLE PRODUSE\_DIN\_INVENTAR  
(  
 id\_produs NUMBER(10) NOT NULL,  
 id\_inventar NUMBER(10) NOT NULL,  
 id\_factura NUMBER(10) NOT NULL,  
 cantitate NUMBER(5) NOT NULL,  
 data\_expirare DATE,  
 CONSTRAINT pk\_produse\_inventar PRIMARY KEY (id\_produs, id\_inventar, id\_factura),  
 CONSTRAINT fk\_prod\_inventar FOREIGN KEY (id\_inventar) REFERENCES INVENTAR (id\_inventar),  
 CONSTRAINT fk\_prod\_produs FOREIGN KEY (id\_produs) REFERENCES PRODUS (id\_produs),  
 CONSTRAINT fk\_prod\_factura FOREIGN KEY (id\_factura) REFERENCES FACTURA\_PRODUCATOR (id\_factura)  
);  
  
commit;



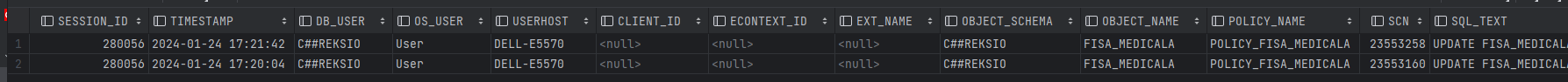
--5  
INSERT INTO LOCATIE(id\_locatie, oras, judet, adresa)  
VALUES (1, 'Bucuresti', 'Sector 2', 'soseaua colentina nr 2');  
INSERT INTO LOCATIE(id\_locatie, oras, judet, adresa, cod\_postal)  
VALUES (2, 'Bucuresti', 'Sector 1', 'soseaua Aviatorilor nr 1', 12345);  
INSERT INTO LOCATIE(id\_locatie, oras, judet, adresa)  
VALUES (3, 'Pantelimon', 'Ilfov', 'soseaua dn2');  
INSERT INTO LOCATIE(id\_locatie, oras, judet, adresa)  
VALUES (4, 'Cluj-Napoca', 'Cluj', 'soseaua jmekera');  
INSERT INTO LOCATIE(id\_locatie, oras, judet, adresa)  
VALUES (5, 'Iasi', 'Iasi', 'soseaua industriei');  
INSERT INTO LOCATIE(id\_locatie, oras, judet, adresa)  
VALUES (6, 'Timisoara', 'Timis', 'soseaua intreprinderii');  
INSERT INTO LOCATIE(id\_locatie, oras, judet, adresa)  
VALUES (7, 'Constanta', 'Constanta', 'soseaua antreprenorilor');  
INSERT INTO LOCATIE(id\_locatie, oras, judet, adresa)  
VALUES (8, 'Giurgiu', 'Giurgiu', 'soseaua giurgiului');  
INSERT INTO LOCATIE(id\_locatie, oras, judet, adresa)  
VALUES (9, 'Severin', 'Mehedinti', 'traian');  
  
INSERT INTO PROPRIETAR(id\_proprietar, nume, prenume, email, telefon, id\_locatie)  
VALUES (1, 'Staicu', 'Octavian-Florin', 'octavian.staicu@s.unibuc.ro', 0736659442, 1);  
INSERT INTO PROPRIETAR(id\_proprietar, nume, prenume, email, telefon, id\_locatie)  
VALUES (2, 'Bodea', 'George', 'george@yahoo.com', 0737392518, 2);  
INSERT INTO PROPRIETAR(id\_proprietar, nume, prenume, email, telefon, id\_locatie)  
VALUES (3, 'Staicu', 'Adrian', 'adi@gmail.com', 0736659441, 1);  
INSERT INTO PROPRIETAR(id\_proprietar, nume, prenume, email, telefon, id\_locatie)  
VALUES (4, 'Popescu', 'Ion', 'popescu.ion@gmail.com', 0771654789, 3);  
INSERT INTO PROPRIETAR(id\_proprietar, nume, prenume, email, telefon, id\_locatie)  
VALUES (5, 'Cineva', 'Interesant', 'jmekerie@my.fmi.unibuc.ro', 0777111999, 4);  
INSERT INTO PROPRIETAR(id\_proprietar, nume, prenume, email, telefon, id\_locatie)  
VALUES (6, 'Fara', 'Animalut', 'cumparator@my.fmi.unibuc.ro', 0729123123, 2);  
  
INSERT INTO ANIMALUT(id\_animalut, nume, rasa, zi\_nastere, id\_proprietar)  
VALUES (1, 'Bit', 'Boxer', *to\_date*('05/05/2017', 'DD/MM/YYYY'), 1);  
INSERT INTO ANIMALUT(id\_animalut, nume, rasa, zi\_nastere, id\_proprietar)  
VALUES (2, 'Kid', 'Ciobanesc German', *to\_date*('13/07/2010', 'DD/MM/YYYY'), 3);  
INSERT INTO ANIMALUT(id\_animalut, nume, rasa, zi\_nastere, id\_proprietar)  
VALUES (3, 'Kitty', 'Birmaneza', *to\_date*('25/12/2015', 'DD/MM/YYYY'), 2);  
INSERT INTO ANIMALUT(id\_animalut, nume, rasa, zi\_nastere, id\_proprietar)  
VALUES (4, 'Angelina', 'Pisica Tigrata de Hymalaia', *to\_date*('06/06/2016', 'DD/MM/YYYY'), 5);  
INSERT INTO ANIMALUT(id\_animalut, nume, rasa, zi\_nastere, id\_proprietar)  
VALUES (5, 'John', 'Metis', *to\_date*('05/12/2008', 'DD/MM/YYYY'), 4);  
insert into ANIMALUT(id\_animalut, nume, rasa, zi\_nastere, id\_proprietar)  
VALUES (6, 'dog', 'Boxer', *to\_date*('09.01.2021', 'dd.mm.yyyy'), 5);  
insert into ANIMALUT(id\_animalut, nume, rasa, zi\_nastere, id\_proprietar)  
VALUES (13, 'Thor', 'Golden Retriever', *to\_date*('13.07.2020', 'dd.mm.yyyy'), 1);  
insert into ANIMALUT(id\_animalut, nume, rasa, zi\_nastere)  
VALUES (11, 'Idk', 'Comuna', *to\_date*('13.01.2023', 'dd.mm.yyyy'));  
insert into ANIMALUT(id\_animalut, nume, rasa, zi\_nastere)  
VALUES (14, 'Idk2', 'Comuna', *to\_date*('01.02.2023', 'dd.mm.yyyy'));  
  
INSERT INTO DIAGNOSTIC(id\_diagnostic, descriere, tratament)  
VALUES (1, 'Vaccinare Leptospiroza', 'Vaccin Leptospiroza');  
INSERT INTO DIAGNOSTIC(id\_diagnostic, descriere, tratament)  
VALUES (2, 'Enterocolita', 'Enteroguarg');  
INSERT INTO DIAGNOSTIC(id\_diagnostic, descriere, tratament)  
VALUES (3, 'Rana deschisa', 'Spray Cicatrizant');  
INSERT INTO DIAGNOSTIC(id\_diagnostic, descriere, tratament)  
VALUES (4, 'durere abdominala', 'ecografie');  
INSERT INTO DIAGNOSTIC(id\_diagnostic, descriere, tratament)  
VALUES (5, 'capusa', 'eliminare capusa');  
  
INSERT INTO ANGAJAT(id\_angajat, nume, prenume, telefon, email, id\_locatie, trateaza\_animale)  
VALUES (1, 'Staicu', 'Clara', 0722012983, 'dr.staicu-clara@clinicatareanimale.ro', 1, 1);  
INSERT INTO ANGAJAT(id\_angajat, nume, prenume, telefon, email, id\_sef, id\_locatie, trateaza\_animale)  
VALUES (2, 'Popa', 'Adi', 0723145867, 'asistent.popa-adi@clinicatareanimale.ro', 1, 2, 1);  
INSERT INTO ANGAJAT(id\_angajat, nume, prenume, telefon, email, id\_locatie, trateaza\_animale)  
VALUES (3, 'Spalatorul', 'Pop', 0711012070, 'spalatorul.pop@gmail.com', 3, 0);  
INSERT INTO ANGAJAT(id\_angajat, nume, prenume, telefon, email, id\_locatie, trateaza\_animale)  
VALUES (4, 'Ana', 'Maria', 0723012982, 'dr.ana-maria@clinicatareanimale.ro', 1, 1);  
INSERT INTO ANGAJAT(id\_angajat, nume, prenume, telefon, email, id\_sef, id\_locatie, trateaza\_animale)  
VALUES (5, 'Numinescu', 'George', 0723145868, 'asistent.numinescu-george@clinicatareanimale.ro', 4, 2, 1);  
  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa, observatii)  
VALUES (1, 2, 1, 1, *to\_date*('13.07.2020', 'dd.mm.yyyy'), 'Stapanul a dat caineul sa manance tort');  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa)  
VALUES (2, 1, 2, 1, *to\_date*('13.07.2011', 'dd.mm.yyyy'));  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa)  
VALUES (3, 1, 3, 2, *to\_date*('13.07.2016', 'dd.mm.yyyy'));  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa, observatii)  
VALUES (4, 3, 1, 2, *to\_date*('13.09.2020', 'dd.mm.yyyy'), 'Rana deschisa in zona botului');  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa, observatii)  
VALUES (5, 1, 4, 1, *to\_date*('13.07.2017', 'dd.mm.yyyy'), 'Una din cele mai frumoase specimene');  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa)  
VALUES (6, 2, 5, 1, *to\_date*('04.05.2009', 'dd.mm.yyyy'));  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa, observatii)  
VALUES (7, 1, 5, 2, *to\_date*('14.06.2010', 'dd.mm.yyyy'), 'Vaccinare anuala');  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa)  
VALUES (8, 1, 4, 2, *to\_date*('13.07.2020', 'dd.mm.yyyy'));  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa, observatii)  
VALUES (9, 3, 3, 1, *to\_date*('13.12.2016', 'dd.mm.yyyy'), 'S-a taiat intr-o conserva de peste');  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa, observatii)  
VALUES (10, 2, 1, 1, *to\_date*('25.12.2020', 'dd.mm.yyyy'), 'Cainele a mancat din pregaritile de Craciun');  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa)  
VALUES (11, 3, 5, 1, *to\_date*('14.08.2015', 'dd.mm.yyyy'));  
insert into FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa)  
VALUES (13, 1, 13, 1, *to\_date*('13.09.2020', 'dd.mm.yyyy'));  
insert into FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa)  
VALUES (12, 1, 13, 1, *to\_date*('13.07.2021', 'dd.mm.yyyy'));  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa, observatii)  
VALUES (14, 3, 11, 1, *to\_date*('05.02.2023', 'dd.mm.yyyy'), 'catelus fara stapan');  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa, observatii)  
VALUES (15, 3, 14, 1, *to\_date*('05.02.2023', 'dd.mm.yyyy'), 'catelus fara stapan');  
  
INSERT INTO PRODUCATOR(id\_producator, nume\_firma, telefon, email, id\_locatie)  
VALUES (1, 'Mancare Catei SA', 0767582563, 'comenzi@mancarecatei.ro', 5);  
INSERT INTO PRODUCATOR(id\_producator, nume\_firma, telefon, email, persoana\_contact, id\_locatie)  
VALUES (2, 'Aparatura Animale SRL', 0798712365, 'aparatus.ion@aparaturaanimale.ro', 'Aparatus Ion', 6);  
INSERT INTO PRODUCATOR(id\_producator, nume\_firma, telefon, email, persoana\_contact, id\_locatie)  
VALUES (3, 'Vaccinam cu Iubire', 0712345678, 'bussines@vacciniubire.ro', 'Ana Vacix', 7);  
INSERT INTO PRODUCATOR(id\_producator, nume\_firma, telefon, email, persoana\_contact, id\_locatie)  
VALUES (4, 'Producator 4', 0700000000, 'producator4@bussines.ro', 'Producator 4 contact', 8);  
INSERT INTO PRODUCATOR(id\_producator, nume\_firma, telefon, email, persoana\_contact, id\_locatie)  
VALUES (5, 'Producator 5', 0711111111, 'bussines@producator5.ro', 'contact prod 5', 9);  
  
INSERT INTO CONTRACT(id\_contract, data\_semnare)  
VALUES (1, *to\_date*('02.01.2005', 'dd.mm.yyyy'));  
INSERT INTO CONTRACT(id\_contract, data\_semnare)  
VALUES (2, *to\_date*('13.07.2004', 'dd.mm.yyyy'));  
INSERT INTO CONTRACT(id\_contract, data\_semnare)  
VALUES (3, *to\_date*('09.09.2005', 'dd.mm.yyyy'));  
INSERT INTO CONTRACT(id\_contract, data\_semnare)  
VALUES (4, *to\_date*('01.05.2007', 'dd.mm.yyyy'));  
INSERT INTO CONTRACT(id\_contract, data\_semnare)  
VALUES (5, *to\_date*('02.06.2008', 'dd.mm.yyyy'));  
INSERT INTO CONTRACT(id\_contract, data\_semnare)  
VALUES (6, *to\_date*('02.01.2005', 'dd.mm.yyyy'));  
INSERT INTO CONTRACT(id\_contract, data\_semnare, observatii, id\_contract\_modificat)  
VALUES (7, *to\_date*('28.12.2020', 'dd.mm.yyyy'), 'Promovare', 2);  
INSERT INTO CONTRACT(id\_contract, data\_semnare)  
VALUES (8, *to\_date*('02.01.2020', 'dd.mm.yyyy'));  
INSERT INTO CONTRACT(id\_contract, data\_semnare)  
VALUES (9, *to\_date*('02.01.2021', 'dd.mm.yyyy'));  
INSERT INTO CONTRACT(id\_contract, data\_semnare)  
VALUES (10, *to\_date*('02.07.2022', 'dd.mm.yyyy'));  
INSERT INTO CONTRACT(id\_contract, data\_semnare)  
VALUES (11, *to\_date*('13.01.2023', 'dd.mm.yyyy'));  
  
INSERT INTO CONTRACT\_ANGAJAT(id\_contract, id\_angajat, rol, salariu)  
VALUES (1, 1, 'doctor veterinar', 6000);  
INSERT INTO CONTRACT\_ANGAJAT(id\_contract, id\_angajat, rol, salariu)  
VALUES (2, 2, 'asistent veterinar', 3000);  
INSERT INTO CONTRACT\_ANGAJAT(id\_contract, id\_angajat, rol, salariu)  
VALUES (3, 3, 'paznic', 1500);  
INSERT INTO CONTRACT\_ANGAJAT(id\_contract, id\_angajat, rol, salariu)  
VALUES (7, 2, 'doctor veterinar', 6000);  
INSERT INTO CONTRACT\_ANGAJAT(id\_contract, id\_angajat, rol, salariu)  
VALUES (8, 4, 'doctor veterinar', 6000);  
INSERT INTO CONTRACT\_ANGAJAT(id\_contract, id\_angajat, rol, salariu)  
VALUES (9, 5, 'asistent veterinar', 3000);  
  
INSERT INTO CONTRACT\_PRODUCATOR(id\_contract, id\_producator)  
VALUES (4, 1);  
INSERT INTO CONTRACT\_PRODUCATOR(id\_contract, id\_producator)  
VALUES (5, 2);  
INSERT INTO CONTRACT\_PRODUCATOR(id\_contract, id\_producator)  
VALUES (6, 3);  
INSERT INTO CONTRACT\_PRODUCATOR(id\_contract, id\_producator)  
VALUES (10, 4);  
INSERT INTO CONTRACT\_PRODUCATOR(id\_contract, id\_producator)  
VALUES (11, 5);  
  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (1, 300, *to\_date*('13.07.2020', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (2, 300, *to\_date*('13.07.2011', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (3, 300, *to\_date*('13.07.2015', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (4, 300, *to\_date*('13.09.2020', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (5, 300, *to\_date*('13.07.2017', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (6, 300, *to\_date*('04.05.2009', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (7, 300, *to\_date*('14.06.2010', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (8, 300, *to\_date*('13.07.2020', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (9, 300, *to\_date*('13.12.2016', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (10, 300, *to\_date*('25.12.2020', 'dd.mm.yyyy'));  
insert into FACTURA(id\_factura, total\_factura, data\_emitere)  
VALUES (104, 300, *to\_date*('13.07.2021', 'dd.mm.yyyy'));  
insert into FACTURA(id\_factura, total\_factura, data\_emitere)  
VALUES (105, 3000, *to\_date*('13.07.2021', 'dd.mm.yyyy'));  
insert into FACTURA(id\_factura, total\_factura, data\_emitere)  
VALUES (106, 3000, *to\_date*('12.05.2021', 'dd.mm.yyyy'));  
insert into FACTURA(id\_factura, total\_factura, data\_emitere)  
VALUES (107, 3000, *to\_date*('12.05.2021', 'dd.mm.yyyy'));  
Insert into FACTURA(id\_factura, total\_factura, data\_emitere)  
VALUES (1000, 100, *sysdate*);  
Insert into FACTURA(id\_factura, total\_factura, data\_emitere)  
VALUES (1001, 100, *sysdate*);  
Insert into FACTURA(id\_factura, total\_factura, data\_emitere)  
VALUES (1002, 499, *to\_date*('06.02.2023', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (11, 1500, *to\_date*('25.10.2009', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (12, 4325, *to\_date*('13.11.2009', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (13, 6000, *to\_date*('25.07.2009', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (14, 1400, *to\_date*('25.10.2014', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (15, 4725, *to\_date*('13.04.2013', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (16, 6025, *to\_date*('25.09.2014', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (17, 5000, *to\_date*('13.11.2019', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (18, 7500, *to\_date*('25.09.2019', 'dd.mm.yyyy'));  
INSERT INTO FACTURA(ID\_FACTURA, TOTAL\_FACTURA, DATA\_EMITERE)  
VALUES (19, 9256.35, *to\_date*('12.04.2019', 'dd.mm.yyyy'));  
  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (1, 1, 1);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (2, 3, 2);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (3, 2, 3);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (4, 1, 4);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (5, 5, 5);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (6, 4, 6);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (7, 4, 7);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (8, 5, 8);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (9, 2, 9);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (10, 1, 10);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (104, 1, 11);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (105, 1, 12);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (106, 2, 13);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (107, 2, 13);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar)  
VALUES (1000, 5);  
INSERT INTO FACTURA\_PROPRIETAR(id\_factura, id\_proprietar, id\_fisa\_medicala)  
VALUES (1002, 5, 15);  
  
INSERT INTO FACTURA\_PRODUCATOR(id\_factura, id\_producator)  
VALUES (11, 1);  
INSERT INTO FACTURA\_PRODUCATOR(id\_factura, id\_producator)  
VALUES (12, 2);  
INSERT INTO FACTURA\_PRODUCATOR(id\_factura, id\_producator)  
VALUES (13, 3);  
INSERT INTO FACTURA\_PRODUCATOR(id\_factura, id\_producator)  
VALUES (14, 1);  
INSERT INTO FACTURA\_PRODUCATOR(id\_factura, id\_producator)  
VALUES (15, 2);  
INSERT INTO FACTURA\_PRODUCATOR(id\_factura, id\_producator)  
VALUES (16, 3);  
INSERT INTO FACTURA\_PRODUCATOR(id\_factura, id\_producator)  
VALUES (17, 1);  
INSERT INTO FACTURA\_PRODUCATOR(id\_factura, id\_producator)  
VALUES (18, 2);  
INSERT INTO FACTURA\_PRODUCATOR(id\_factura, id\_producator)  
VALUES (19, 3);  
  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (1, 150, *to\_date*('13.07.2020', 'dd.mm.yyyy'), 1);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (2, 300, *to\_date*('13.07.2011', 'dd.mm.yyyy'), 2);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (3, 300, *to\_date*('13.08.2015', 'dd.mm.yyyy'), 3);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (4, 300, *to\_date*('03.01.2021', 'dd.mm.yyyy'), 4);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (5, 150, *to\_date*('13.07.2017', 'dd.mm.yyyy'), 5);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (6, 75.25, *to\_date*('13.08.2017', 'dd.mm.yyyy'), 5);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (7, 75.75, *to\_date*('13.09.2017', 'dd.mm.yyyy'), 5);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (8, 100, *to\_date*('13.07.2009', 'dd.mm.yyyy'), 6);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (9, 200, *to\_date*('17.08.2009', 'dd.mm.yyyy'), 6);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (10, 300, *to\_date*('13.07.2010', 'dd.mm.yyyy'), 7);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (11, 300, *to\_date*('13.09.2020', 'dd.mm.yyyy'), 8);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (12, 300, *to\_date*('13.12.2016', 'dd.mm.yyyy'), 9);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (13, 300, *to\_date*('15.01.2021', 'dd.mm.yyyy'), 10);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (14, 1500, *to\_date*('25.10.2009', 'dd.mm.yyyy'), 11);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (15, 4325, *to\_date*('25.10.2009', 'dd.mm.yyyy'), 12);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (16, 3000, *to\_date*('25.07.2009', 'dd.mm.yyyy'), 13);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (17, 3000, *to\_date*('25.09.2009', 'dd.mm.yyyy'), 13);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (18, 1400, *to\_date*('25.10.2014', 'dd.mm.yyyy'), 14);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (19, 4725, *to\_date*('13.05.2013', 'dd.mm.yyyy'), 15);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (20, 3025, *to\_date*('25.09.2014', 'dd.mm.yyyy'), 16);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (21, 3000, *to\_date*('01.11.2014', 'dd.mm.yyyy'), 16);  
INSERT INTO CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (22, 4500, *to\_date*('25.10.2019', 'dd.mm.yyyy'), 18);  
insert into CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (104, 300, *to\_date*('13.07.2021', 'dd.mm.yyyy'), 104);  
insert into CHITANTA(id\_chitanta, suma\_platita, data\_plata, id\_factura)  
VALUES (106, 600, *to\_date*('13.07.2021', 'dd.mm.yyyy'), 106);  
  
  
INSERT INTO PRODUS(id\_produs, caracteristici, id\_producator)  
VALUES (1, 'boabe de caini cu carne de vita si somon', 1);  
INSERT INTO PRODUS(id\_produs, caracteristici, id\_producator)  
VALUES (2, 'boabe de pisici cu ulei de peste', 1);  
INSERT INTO PRODUS(id\_produs, caracteristici, id\_producator)  
VALUES (3, 'aparat radiografie', 2);  
INSERT INTO PRODUS(id\_produs, caracteristici, id\_producator)  
VALUES (4, 'stetoscop', 2);  
INSERT INTO PRODUS(id\_produs, caracteristici, id\_producator)  
VALUES (5, 'masa operatii', 2);  
INSERT INTO PRODUS(id\_produs, caracteristici, id\_producator)  
VALUES (6, 'vaccin leptospiroza', 3);  
INSERT INTO PRODUS(id\_produs, caracteristici, id\_producator)  
VALUES (7, 'vaccin hexavalent', 3);  
  
INSERT INTO INVENTAR(id\_inventar, data\_inventar)  
VALUES (1, *to\_date*('05.01.2010', 'dd.mm.yyyy'));  
INSERT INTO INVENTAR(id\_inventar, data\_inventar)  
VALUES (2, *to\_date*('05.01.2015', 'dd.mm.yyyy'));  
INSERT INTO INVENTAR(id\_inventar, data\_inventar)  
VALUES (3, *to\_date*('05.01.2020', 'dd.mm.yyyy'));  
INSERT INTO INVENTAR(id\_inventar, data\_inventar)  
VALUES (4, *to\_date*('05.01.2021', 'dd.mm.yyyy'));  
INSERT INTO INVENTAR(id\_inventar, data\_inventar)  
VALUES (5, *to\_date*('05.01.2022', 'dd.mm.yyyy'));  
  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (1, 1, 11, 10, *to\_date*('05.01.2011', 'dd.mm.yyyy'));  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (2, 1, 12, 15, *to\_date*('05.01.2011', 'dd.mm.yyyy'));  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (1, 2, 13, 20, *to\_date*('05.01.2016', 'dd.mm.yyyy'));  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (2, 2, 14, 25, *to\_date*('05.01.2016', 'dd.mm.yyyy'));  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (1, 3, 15, 35, *to\_date*('05.03.2023', 'dd.mm.yyyy'));  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (2, 3, 16, 30, *to\_date*('05.01.2021', 'dd.mm.yyyy'));  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate)  
VALUES (3, 1, 17, 1);  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate)  
VALUES (3, 2, 18, 1);  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate)  
VALUES (3, 3, 19, 2);  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate)  
VALUES (4, 1, 11, 5);  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate)  
VALUES (4, 2, 12, 10);  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate)  
VALUES (4, 3, 13, 15);  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate)  
VALUES (5, 1, 14, 1);  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate)  
VALUES (5, 2, 15, 1);  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate)  
VALUES (5, 3, 16, 2);  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (6, 1, 17, 100, *to\_date*('13.07.2011', 'dd.mm.yyyy'));  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (7, 1, 18, 150, *to\_date*('13.07.2011', 'dd.mm.yyyy'));  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (6, 2, 19, 100, *to\_date*('13.07.2011', 'dd.mm.yyyy'));  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (7, 2, 11, 150, *to\_date*('13.07.2011', 'dd.mm.yyyy'));  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (6, 3, 12, 234, *to\_date*('13.07.2011', 'dd.mm.yyyy'));  
INSERT INTO PRODUSE\_DIN\_INVENTAR(id\_produs, id\_inventar, id\_factura, cantitate, data\_expirare)  
VALUES (7, 3, 13, 512, *to\_date*('13.07.2011', 'dd.mm.yyyy'));  
  
  
commit;

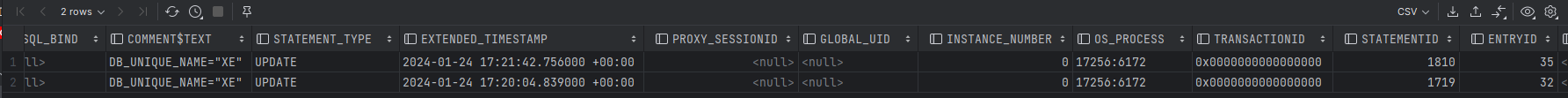
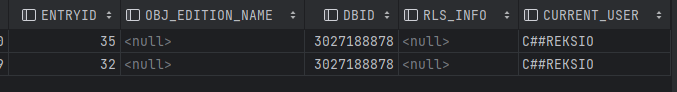


-- 2. criptarea datelor  
drop table ENCKEYS cascade constraints;  
drop table PROPRIETAR\_DECRYPTED cascade constraints;  
drop table PROPRIETAR\_ENCRYPTED cascade constraints;  
  
  
CREATE TABLE ENCKeys  
(  
 id\_key NUMBER(7),  
 enc\_key RAW(16),  
 CONSTRAINT pk\_ENCKeys PRIMARY KEY (id\_key)  
);  
  
CREATE TABLE Proprietar\_Encrypted  
(  
 id\_proprietar NUMBER(10),  
 nume RAW(100) NOT NULL,  
 prenume RAW(100) NOT NULL,  
 telefon RAW(100) NOT NULL,  
 email RAW(100) NOT NULL,  
 id\_locatie RAW(100) NOT NULL,  
 CONSTRAINT pk\_proprietar\_enc PRIMARY KEY (id\_proprietar)  
);  
  
CREATE TABLE Proprietar\_Decrypted  
(  
 id\_proprietar NUMBER(10),  
 nume VARCHAR2(32) NOT NULL,  
 prenume VARCHAR2(32) NOT NULL,  
 telefon NUMBER(10) NOT NULL,  
 email VARCHAR2(64) NOT NULL,  
 id\_locatie NUMBER(10) NOT NULL,  
 CONSTRAINT pk\_proprietar\_dec PRIMARY KEY (id\_proprietar)  
);  
  
CREATE OR REPLACE PROCEDURE *Encrypt\_Proprietar* AS  
 encryption\_key RAW(16);  
 encryption\_binary\_location RAW(100);  
 encryption\_binary\_last\_name RAW(100);  
 encryption\_binary\_first\_name RAW(100);  
 encryption\_binary\_email RAW(100);  
 encryption\_binary\_phone RAW(100);  
 encryption\_type PLS\_INTEGER;  
BEGIN  
 FOR c IN (SELECT \* FROM PROPRIETAR)  
 LOOP  
 encryption\_key := DBMS\_CRYPTO.*randombytes*(16);  
  
 INSERT INTO ENCKeys (id\_key, enc\_key) VALUES (c.id\_proprietar, encryption\_key);  
  
 encryption\_type := DBMS\_CRYPTO.ENCRYPT\_AES128 + DBMS\_CRYPTO.PAD\_PKCS5 + DBMS\_CRYPTO.CHAIN\_CBC;  
  
 encryption\_binary\_location :=  
 DBMS\_CRYPTO.*encrypt*(UTL\_I18N.*STRING\_TO\_RAW*(c.id\_locatie, 'AL32UTF8'), encryption\_type,  
 encryption\_key);  
 encryption\_binary\_last\_name :=  
 DBMS\_CRYPTO.*encrypt*(UTL\_I18N.*STRING\_TO\_RAW*(c.prenume, 'AL32UTF8'), encryption\_type, encryption\_key);  
 encryption\_binary\_first\_name :=  
 DBMS\_CRYPTO.*encrypt*(UTL\_I18N.*STRING\_TO\_RAW*(c.nume, 'AL32UTF8'), encryption\_type, encryption\_key);  
 encryption\_binary\_email :=  
 DBMS\_CRYPTO.*encrypt*(UTL\_I18N.*STRING\_TO\_RAW*(c.email, 'AL32UTF8'), encryption\_type, encryption\_key);  
 encryption\_binary\_phone :=  
 DBMS\_CRYPTO.*encrypt*(UTL\_I18N.*STRING\_TO\_RAW*(c.telefon, 'AL32UTF8'), encryption\_type, encryption\_key);  
  
 INSERT INTO PROPRIETAR\_Encrypted (id\_proprietar,  
 nume, prenume, telefon,  
 email, id\_locatie)  
 VALUES (c.id\_proprietar, encryption\_binary\_last\_name, encryption\_binary\_first\_name, encryption\_binary\_phone,  
 encryption\_binary\_email, encryption\_binary\_location);  
 END LOOP;  
 COMMIT;  
END ;  
  
BEGIN  
 *Encrypt\_Proprietar*();  
end;  
  
SELECT \*  
FROM ENCKeys;  
  
SELECT \*  
FROM PROPRIETAR\_ENCRYPTED;  
  
CREATE OR REPLACE PROCEDURE *Decrypt\_Proprietar* AS  
 encryption\_type PLS\_INTEGER;  
 decryption\_key RAW(16);  
 decrypted\_location VARCHAR2(100);  
 decrypted\_last\_name VARCHAR2(100);  
 decrypted\_first\_name VARCHAR2(100);  
 decrypted\_email VARCHAR2(100);  
 decrypted\_phone VARCHAR2(100);  
BEGIN  
 FOR c\_enc IN (SELECT \* FROM PROPRIETAR\_ENCRYPTED)  
 LOOP  
  
 SELECT enc\_key  
 INTO decryption\_key  
 FROM ENCKeys  
 WHERE id\_key = *TO\_NUMBER*(c\_enc.id\_proprietar);  
  
 encryption\_type := DBMS\_CRYPTO.ENCRYPT\_AES128 + DBMS\_CRYPTO.PAD\_PKCS5 + DBMS\_CRYPTO.CHAIN\_CBC;  
  
 decrypted\_location :=  
 UTL\_I18N.*RAW\_TO\_CHAR*(DBMS\_CRYPTO.*DECRYPT*(c\_enc.id\_locatie, encryption\_type, decryption\_key));  
 decrypted\_last\_name :=  
 UTL\_I18N.*RAW\_TO\_CHAR*(DBMS\_CRYPTO.*DECRYPT*(c\_enc.nume, encryption\_type, decryption\_key));  
 decrypted\_first\_name :=  
 UTL\_I18N.*RAW\_TO\_CHAR*(DBMS\_CRYPTO.*DECRYPT*(c\_enc.prenume, encryption\_type, decryption\_key));  
 decrypted\_email := UTL\_I18N.*RAW\_TO\_CHAR*(DBMS\_CRYPTO.*DECRYPT*(c\_enc.email, encryption\_type, decryption\_key));  
 decrypted\_phone :=  
 UTL\_I18N.*RAW\_TO\_CHAR*(DBMS\_CRYPTO.*DECRYPT*(c\_enc.telefon, encryption\_type, decryption\_key));  
  
 INSERT INTO PROPRIETAR\_DECRYPTED (id\_proprietar, nume, prenume, telefon, email, id\_locatie)  
 VALUES (c\_enc.id\_proprietar,  
 decrypted\_first\_name,  
 decrypted\_last\_name,  
 decrypted\_phone, decrypted\_email,  
 decrypted\_location);  
 END LOOP;  
 COMMIT;  
END;  
  
BEGIN  
 *DECRYPT\_PROPRIETAR*();  
end;  
  
SELECT \*  
FROM PROPRIETAR\_DECRYPTED;  
  
-- 3. Auditarea  
  
ALTER SYSTEM SET audit\_trail = db, extended SCOPE = spfile;  
  
AUDIT SELECT, UPDATE, INSERT, DELETE ON FISA\_MEDICALA;  
  
CREATE SYNONYM AUD$ FOR SYS.AUD$;  
  
SELECT \*  
FROM FISA\_MEDICALA;  
  
INSERT INTO FISA\_MEDICALA(id\_fisa, id\_diagnostic, id\_animalut, id\_doctor, data\_fisa, observatii)  
VALUES (1113, 1, 1, 1, *sysdate*, 'obs audit');  
  
SELECT obj$name, sqltext, userid  
from AUD$  
ORDER BY ntimestamp# DESC;  
  
-- b)  
  
CREATE SEQUENCE secv\_audit START WITH 1 INCREMENT BY 1;  
  
CREATE TABLE audit\_fisa\_medicala  
(  
 sec\_id NUMBER(4) PRIMARY KEY,  
 user\_name VARCHAR2(20) NOT NULL,  
 session\_name VARCHAR2(20) NOT NULL,  
 data\_modificare DATE NOT NULL,  
 observatii\_vechi VARCHAR2(4000) NOT NULL,  
 observatii\_noi VARCHAR2(4000) NOT NULL  
);  
  
CREATE OR REPLACE TRIGGER trigger\_audit\_fisa\_medicala  
 AFTER INSERT OR UPDATE OF observatii  
 ON FISA\_MEDICALA  
 FOR EACH ROW  
 WHEN (NEW.observatii IS NOT NULL and OLD.observatii IS NOT NULL )  
BEGIN  
 DECLARE  
 v\_old\_observatii FISA\_MEDICALA.observatii%TYPE := :OLD.observatii;  
 v\_new\_observatii FISA\_MEDICALA.observatii%TYPE := :NEW.observatii;  
 BEGIN  
 INSERT INTO audit\_fisa\_medicala(sec\_id, user\_name, session\_name, data\_modificare, observatii\_vechi,  
 observatii\_noi)  
 VALUES (secv\_audit.NEXTVAL, *sys\_context*('userenv', 'session\_user'), *sys\_context*('userenv', 'host'), *sysdate*,  
 v\_old\_observatii, v\_new\_observatii);  
 END;  
END;  
/  
  
UPDATE FISA\_MEDICALA  
set observatii = 'obs audit 2'  
where id\_fisa = 1113;  
  
SELECT \*  
FROM audit\_fisa\_medicala;   


-- c)  
-- GRANT EXECUTE ON DBMS\_FGA TO C##REKSIO;  
  
  
CREATE OR REPLACE PROCEDURE *POLICY\_HANDLER\_FISA\_MEDICALA*(object\_schema VARCHAR2, object\_name VARCHAR2, policy\_name VARCHAR2)  
AS  
BEGIN  
 DBMS\_OUTPUT.*PUT\_LINE*('Audit Policy: [object\_schema = ' || object\_schema || ']; [object\_name = ' || object\_name ||  
 ']; [policy\_name = ' || policy\_name || ']');  
END;  
/  
  
-- GRANT SELECT ON DBA\_FGA\_AUDIT\_TRAIL TO c##reksio;  
CREATE OR REPLACE PROCEDURE *CREATE\_AUDIT\_POLICY\_FOR\_FISA\_MEDICALA* as  
BEGIN  
  
 BEGIN  
 dbms\_fga.*drop\_policy*(  
 object\_schema => 'C##REKSIO',  
 object\_name => 'FISA\_MEDICALA',  
 policy\_name => 'POLICY\_FISA\_MEDICALA'  
 );  
 EXCEPTION  
 WHEN OTHERS then  
 NULL;  
 END;  
 dbms\_fga.*add\_policy*(  
 object\_schema => 'C##REKSIO',  
 object\_name => 'FISA\_MEDICALA',  
 policy\_name => 'POLICY\_FISA\_MEDICALA',  
 enable => false,  
 statement\_types => 'UPDATE',  
 handler\_module => 'POLICY\_HANDLER\_FISA\_MEDICALA');  
END;  
/  
  
  
BEGIN  
 *CREATE\_AUDIT\_POLICY\_FOR\_FISA\_MEDICALA*();  
 dbms\_fga.*enable\_policy*(object\_schema => 'C##REKSIO',  
 object\_name => 'FISA\_MEDICALA',  
 policy\_name => 'POLICY\_FISA\_MEDICALA');  
END;  
  
UPDATE FISA\_MEDICALA  
set observatii = 'obs audit 3'  
where id\_fisa = 1113;  


SELECT \*  
FROM dba\_fga\_audit\_trail  
ORDER BY TIMESTAMP DESC;



DELETE  
FROM FISA\_MEDICALA  
where id\_fisa = 1113;  
commit;

Utilizatorii:

* Administrator
* Doctor
* Contabil
* Guest

Procesele:

* P1: Diagnosticare animalut
* P2: Inregistrare factura/plata
* P3: Adaugare/alterare utilizator
* P4: Inventar
* P5: editare/stergere date (gresite sau la cererea clientilor etc)

Cele 3 matrici:

1. Proces-utilizator

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **P1** | **P2** | **P3** | **P4** | **P5** |
| **Administrator** |  |  | X |  | X |
| **Doctor** | X |  |  |  |  |
| **Contabil** |  | X |  | X |  |
| **Guest** |  |  |  |  |  |

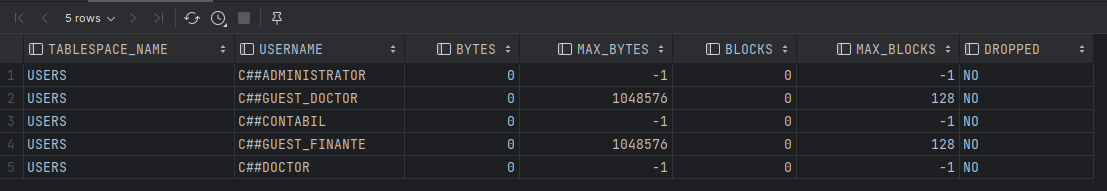
1. Entitate-proces

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | P1 | P2 | P3 | P4 | P5 |
| PROPRIETAR |  | S,U,I,D |  |  | U,D |
| ANIMALUT | S,U,I,D |  |  |  | U,D |
| DIAGNOSTIC | S,U,I,D |  |  |  | U,D |
| FACTURA |  | S,U,I,D |  |  | U,D |
| CHITANTA |  | S,U,I,D |  |  | U,D |
| FACTURA\_PROPRIETAR |  | S,U,I,D |  |  | U,D |
| FACTURA\_PRODUCATOR |  | S,U,I,D |  |  | U,D |
| INVENTAR |  |  |  | S,U,I | U,D |
| PRODUSE\_DIN\_INVENTAR |  |  |  | S,U,I | U,D |
| PRODUS |  | S,U,I,D |  | S,U,I | U,D |
| FISA\_MEDICALA | S,U,I,D | S |  |  | U,D |
| ANGAJAT | S |  | S,U,I,D |  | U,D |
| CONTRACT |  |  | S,U,I,D | S | U,D |
| CONTRACT\_PRODUCATOR |  |  |  | S | U,D |
| CONTRACT\_ANGAJAT |  |  | S,U,I,D |  | U,D |
| PRODUCATOR |  | S,U,I,D |  | S | U,D |
| LOCATIE |  | S,U,I,D |  | S | U,D |

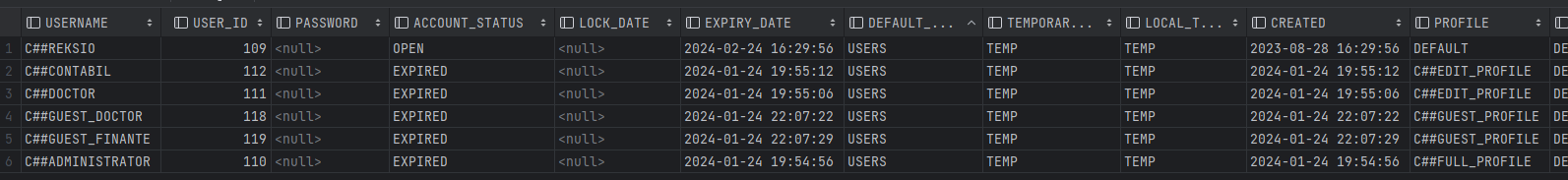
1. Entitate-utilizator

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Administrator | Doctor | Contabil | Guest\_Doctor | Guest\_Finante |
| PROPRIETAR | S,U,D | S,U,I | S,U,I | S |  |
| ANIMALUT | S,U,D | S,U,I | S | S |  |
| DIAGNOSTIC | S,U,D | S,U,I,D |  | S |  |
| FACTURA | S,U,D | S | S,U,I | S | S |
| CHITANTA | S,U,D |  | S,U,I |  | S |
| FACTURA\_PROPRIETAR | S,U,D | S | S,U,I | S | S |
| FACTURA\_PRODUCATOR | S,U,D |  | S,U,I |  | S |
| INVENTAR | S,U,D | S | S,U,I |  | S |
| PRODUSE\_DIN\_INVENTAR | S,U,D | S | S,U,I |  | S |
| PRODUS | S,U,D | S | S,U,I |  | S |
| FISA\_MEDICALA | S,U,D | S,U,I | S | S |  |
| ANGAJAT | S,I,U,D | S | S | S | S |
| CONTRACT | S,I,U,D |  | S |  |  |
| CONTRACT\_PRODUCATOR | S,I,U,D |  | S |  |  |
| CONTRACT\_ANGAJAT | S,I,U,D |  | S |  |  |
| PRODUCATOR | S,U,D | S | S,U,I,D |  | S |
| LOCATIE | S,U,D | S,I,U | S,I,U | S | S |

-- 4. gestionarea utilizatorilor unei baze de date si a resurseloc computationale  
CREATE USER C##ADMINISTRATOR IDENTIFIED BY PASSWORD PASSWORD EXPIRE;  
GRANT CREATE SESSION TO C##ADMINISTRATOR;  
  
CREATE USER C##DOCTOR IDENTIFIED BY PASSWORD PASSWORD EXPIRE;  
GRANT CREATE SESSION TO C##DOCTOR;  
  
CREATE USER C##CONTABIL IDENTIFIED BY PASSWORD PASSWORD EXPIRE;  
GRANT CREATE SESSION TO C##CONTABIL;  
  
CREATE USER C##GUEST\_DOCTOR IDENTIFIED BY PASSWORD PASSWORD EXPIRE;  
GRANT CREATE SESSION TO C##GUEST\_DOCTOR;  
  
CREATE USER C##GUEST\_FINANTE IDENTIFIED BY PASSWORD PASSWORD EXPIRE;  
GRANT CREATE SESSION TO C##GUEST\_FINANTE;  
  
ALTER USER C##ADMINISTRATOR QUOTA UNLIMITED ON USERS;  
ALTER USER C##DOCTOR QUOTA UNLIMITED ON USERS;  
ALTER USER C##CONTABIL QUOTA UNLIMITED ON USERS;  
ALTER USER C##GUEST\_DOCTOR QUOTA 1M ON USERS;  
ALTER USER C##GUEST\_FINANTE QUOTA 1M ON USERS;  
  
-- GRANT SELECT ON DBA\_TS\_QUOTAS to c##reksio;  
SELECT \*  
FROM DBA\_TS\_QUOTAS  
where TABLESPACE\_NAME='USERS';



CREATE PROFILE C##FULL\_PROFILE LIMIT  
 SESSIONS\_PER\_USER 5  
 IDLE\_TIME 20  
 FAILED\_LOGIN\_ATTEMPTS 4  
 CONNECT\_TIME 120;  
  
CREATE PROFILE C##GUEST\_PROFILE LIMIT  
 SESSIONS\_PER\_USER 20  
 IDLE\_TIME 5  
 FAILED\_LOGIN\_ATTEMPTS 10  
 CONNECT\_TIME 10;  
  
ALTER USER C##ADMINISTRATOR PROFILE C##FULL\_PROFILE;  
ALTER USER C##DOCTOR PROFILE C##FULL\_PROFILE;  
ALTER USER C##CONTABIL PROFILE C##FULL\_PROFILE;  
ALTER USER C##GUEST\_DOCTOR PROFILE C##GUEST\_PROFILE;  
ALTER USER C##GUEST\_FINANTE PROFILE C##GUEST\_PROFILE;

SELECT \*  
FROM DBA\_USERS  
WHERE *UPPER*(USERNAME) LIKE 'C##%';  


-- 5. Privilegii si roluri  
CREATE ROLE C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON PROPRIETAR TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON ANIMALUT TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON DIAGNOSTIC TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON FACTURA TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON CHITANTA TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON FACTURA\_PROPRIETAR TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON FACTURA\_PRODUCATOR TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON INVENTAR TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON PRODUSE\_DIN\_INVENTAR TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON PRODUS TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON FISA\_MEDICALA TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON PRODUCATOR TO C##ADMIN\_ROLE;  
GRANT SELECT, UPDATE, DELETE ON LOCATIE TO C##ADMIN\_ROLE;  
GRANT SELECT, INSERT, UPDATE, DELETE ON ANGAJAT TO C##ADMIN\_ROLE;  
GRANT SELECT, INSERT, UPDATE, DELETE ON CONTRACT TO C##ADMIN\_ROLE;  
GRANT SELECT, INSERT, UPDATE, DELETE ON CONTRACT\_ANGAJAT TO C##ADMIN\_ROLE;  
GRANT SELECT, INSERT, UPDATE, DELETE ON CONTRACT\_PRODUCATOR TO C##ADMIN\_ROLE;  
  
CREATE ROLE C##DOCTOR\_ROLE;  
GRANT SELECT, INSERT, UPDATE on PROPRIETAR to C##DOCTOR\_ROLE;  
GRANT SELECT, INSERT, UPDATE on ANIMALUT to C##DOCTOR\_ROLE;  
GRANT SELECT, INSERT, UPDATE, DELETE on DIAGNOSTIC to C##DOCTOR\_ROLE;  
GRANT SELECT on INVENTAR to C##DOCTOR\_ROLE;  
GRANT SELECT on PRODUSE\_DIN\_INVENTAR to C##DOCTOR\_ROLE;  
GRANT SELECT on PRODUS to C##DOCTOR\_ROLE;  
GRANT SELECT, INSERT, UPDATE on FISA\_MEDICALA to C##DOCTOR\_ROLE;  
GRANT SELECT on ANGAJAT to C##DOCTOR\_ROLE;  
GRANT SELECT on PRODUCATOR to C##DOCTOR\_ROLE;  
GRANT SELECT, INSERT, UPDATE on LOCATIE to C##DOCTOR\_ROLE;  
GRANT SELECT on FACTURA to C##DOCTOR\_ROLE;  
GRANT SELECT on FACTURA\_PROPRIETAR to C##DOCTOR\_ROLE;  
  
CREATE ROLE C##FINANCE\_ROLE;  
GRANT SELECT, INSERT, UPDATE on PROPRIETAR to C##FINANCE\_ROLE;  
GRANT SELECT on ANIMALUT to C##FINANCE\_ROLE;  
GRANT SELECT, INSERT, UPDATE on FACTURA to C##FINANCE\_ROLE;  
GRANT SELECT, INSERT, UPDATE on CHITANTA to C##FINANCE\_ROLE;  
GRANT SELECT, INSERT, UPDATE on FACTURA\_PROPRIETAR to C##FINANCE\_ROLE;  
GRANT SELECT, INSERT, UPDATE on FACTURA\_PRODUCATOR to C##FINANCE\_ROLE;  
GRANT SELECT, INSERT, UPDATE on INVENTAR to C##FINANCE\_ROLE;  
GRANT SELECT, INSERT, UPDATE on PRODUSE\_DIN\_INVENTAR to C##FINANCE\_ROLE;  
GRANT SELECT, INSERT, UPDATE on PRODUS to C##FINANCE\_ROLE;  
GRANT SELECT on FISA\_MEDICALA to C##FINANCE\_ROLE;  
GRANT SELECT on ANGAJAT to C##FINANCE\_ROLE;  
GRANT SELECT on CONTRACT to C##FINANCE\_ROLE;  
GRANT SELECT on CONTRACT\_PRODUCATOR to C##FINANCE\_ROLE;  
GRANT SELECT on CONTRACT\_ANGAJAT to C##FINANCE\_ROLE;  
GRANT SELECT, INSERT, UPDATE,DELETE on PRODUCATOR to C##FINANCE\_ROLE;  
GRANT SELECT, INSERT, UPDATE on LOCATIE to C##FINANCE\_ROLE;  
  
CREATE ROLE C##GUEST\_DOCTOR\_ROLE;  
GRANT SELECT on PROPRIETAR to C##GUEST\_DOCTOR\_ROLE;  
GRANT SELECT on ANIMALUT to C##GUEST\_DOCTOR\_ROLE;  
GRANT SELECT on DIAGNOSTIC to C##GUEST\_DOCTOR\_ROLE;  
GRANT SELECT on FACTURA to C##GUEST\_DOCTOR\_ROLE;  
GRANT SELECT on FACTURA\_PROPRIETAR to C##GUEST\_DOCTOR\_ROLE;  
GRANT SELECT on FISA\_MEDICALA to C##GUEST\_DOCTOR\_ROLE;  
GRANT SELECT on ANGAJAT to C##GUEST\_DOCTOR\_ROLE;  
GRANT SELECT on LOCATIE to C##GUEST\_DOCTOR\_ROLE;  
  
CREATE ROLE C##GUEST\_FINANCE\_ROLE;  
GRANT SELECT on FACTURA to C##GUEST\_FINANCE\_ROLE;  
GRANT SELECT on CHITANTA to C##GUEST\_FINANCE\_ROLE;  
GRANT SELECT on FACTURA\_PROPRIETAR to C##GUEST\_FINANCE\_ROLE;  
GRANT SELECT on FACTURA\_PRODUCATOR to C##GUEST\_FINANCE\_ROLE;  
GRANT SELECT on INVENTAR to C##GUEST\_FINANCE\_ROLE;  
GRANT SELECT on PRODUSE\_DIN\_INVENTAR to C##GUEST\_FINANCE\_ROLE;  
GRANT SELECT on PRODUS to C##GUEST\_FINANCE\_ROLE;  
GRANT SELECT on ANGAJAT to C##GUEST\_FINANCE\_ROLE;  
GRANT SELECT on PRODUCATOR to C##GUEST\_FINANCE\_ROLE;  
GRANT SELECT on LOCATIE to C##GUEST\_FINANCE\_ROLE;  
  
GRANT C##ADMIN\_ROLE to C##ADMINISTRATOR;  
GRANT C##DOCTOR\_ROLE to C##DOCTOR;  
GRANT C##FINANCE\_ROLE to C##CONTABIL;  
GRANT C##GUEST\_DOCTOR\_ROLE to C##GUEST\_DOCTOR;  
GRANT C##GUEST\_FINANCE\_ROLE to C##GUEST\_FINANTE;

-- 6. contextul aplicatiei  
  
CREATE OR REPLACE PROCEDURE *CONTEXT\_TURA\_ZI\_NOAPTE* IS  
 curr\_hour NUMBER(3);  
BEGIN  
 SELECT *TO\_NUMBER*(*TO\_CHAR*(*sysdate*, 'hh24'))  
 INTO curr\_hour  
 FROM dual;  
  
 DBMS\_OUTPUT.*PUT\_LINE*('THE HOUR IS: '|| curr\_hour);  
  
 IF curr\_hour < 10 OR curr\_hour > 20 THEN  
 DBMS\_OUTPUT.*PUT\_LINE*('Se lucreaza in tura de noapte! Atentie la datele introduse!');  
 DBMS\_SESSION.*SET\_CONTEXT*('APP\_CONTEXT', 'ACC\_NIGHT\_SHIFT', 'YES');  
 ELSE  
 dbms\_session.*set\_context*('APP\_CONTEXT', 'ACC\_NIGHT\_SHIFT', 'NO');  
 END IF;  
END;  
/  
  
CREATE CONTEXT APP\_CONTEXT USING CONTEXT\_TURA\_ZI\_NOAPTE;  
  
BEGIN  
 *CONTEXT\_TURA\_ZI\_NOAPTE*();  
end;  
  
CREATE OR REPLACE TRIGGER TRIGGER\_ACCESS  
 AFTER LOGON ON DATABASE  
DECLARE  
BEGIN  
 *CONTEXT\_TURA\_ZI\_NOAPTE*();  
END;  
/  
  
DECLARE  
 night\_shift\_value VARCHAR2(3);  
BEGIN  
 SELECT *SYS\_CONTEXT*('APP\_CONTEXT','ACC\_NIGHT\_SHIFT') INTO night\_shift\_value  
 FROM dual;  
  
 DBMS\_OUTPUT.*PUT\_LINE*('Night Shift Value: ' || night\_shift\_value);  
  
end;

