PROIECT CLINICĂ VETERINARĂ

Student: Staicu Octavian-Florin

Grupa 506 Master Inginerie Software
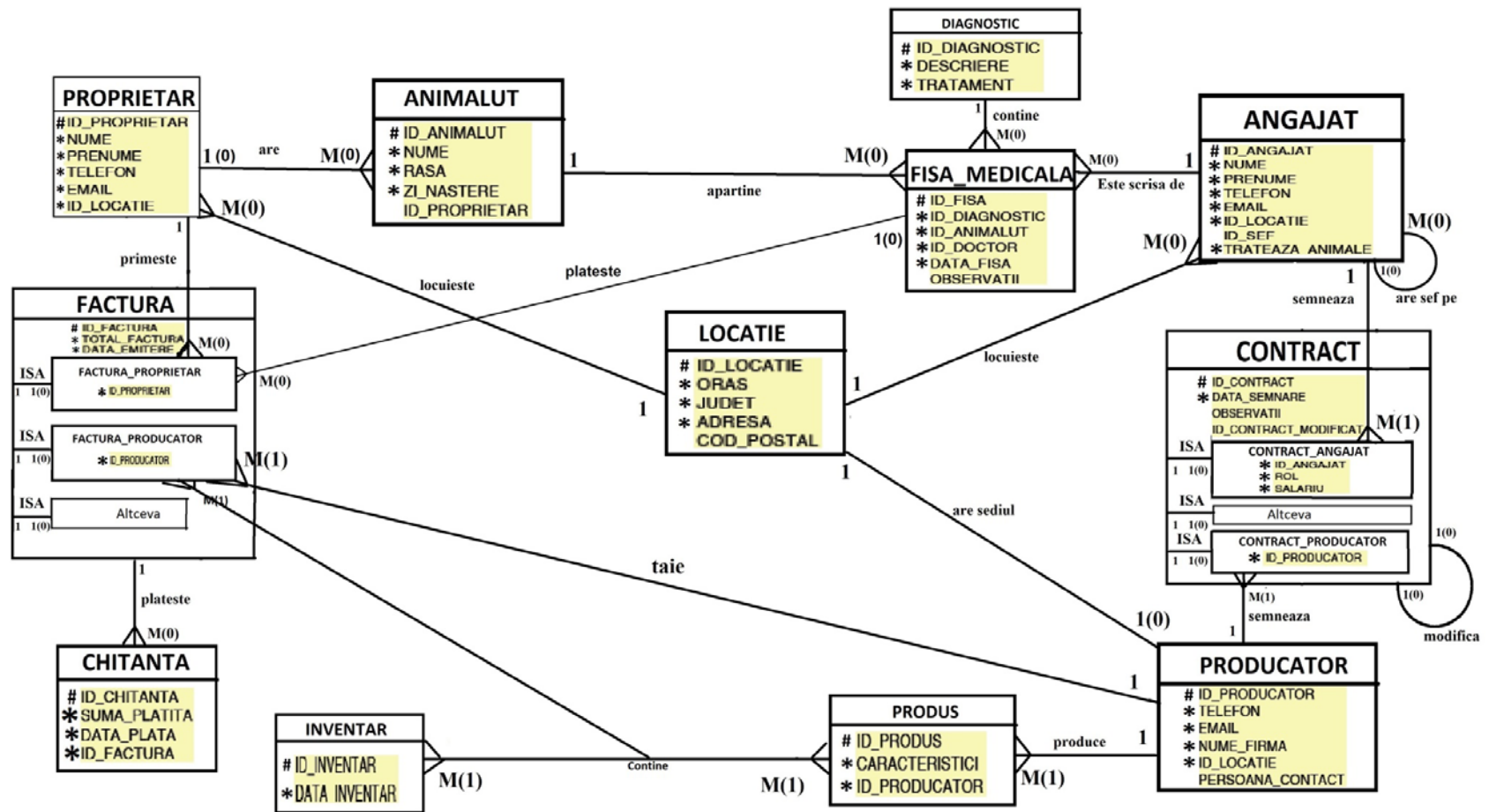
Clinică veterinară
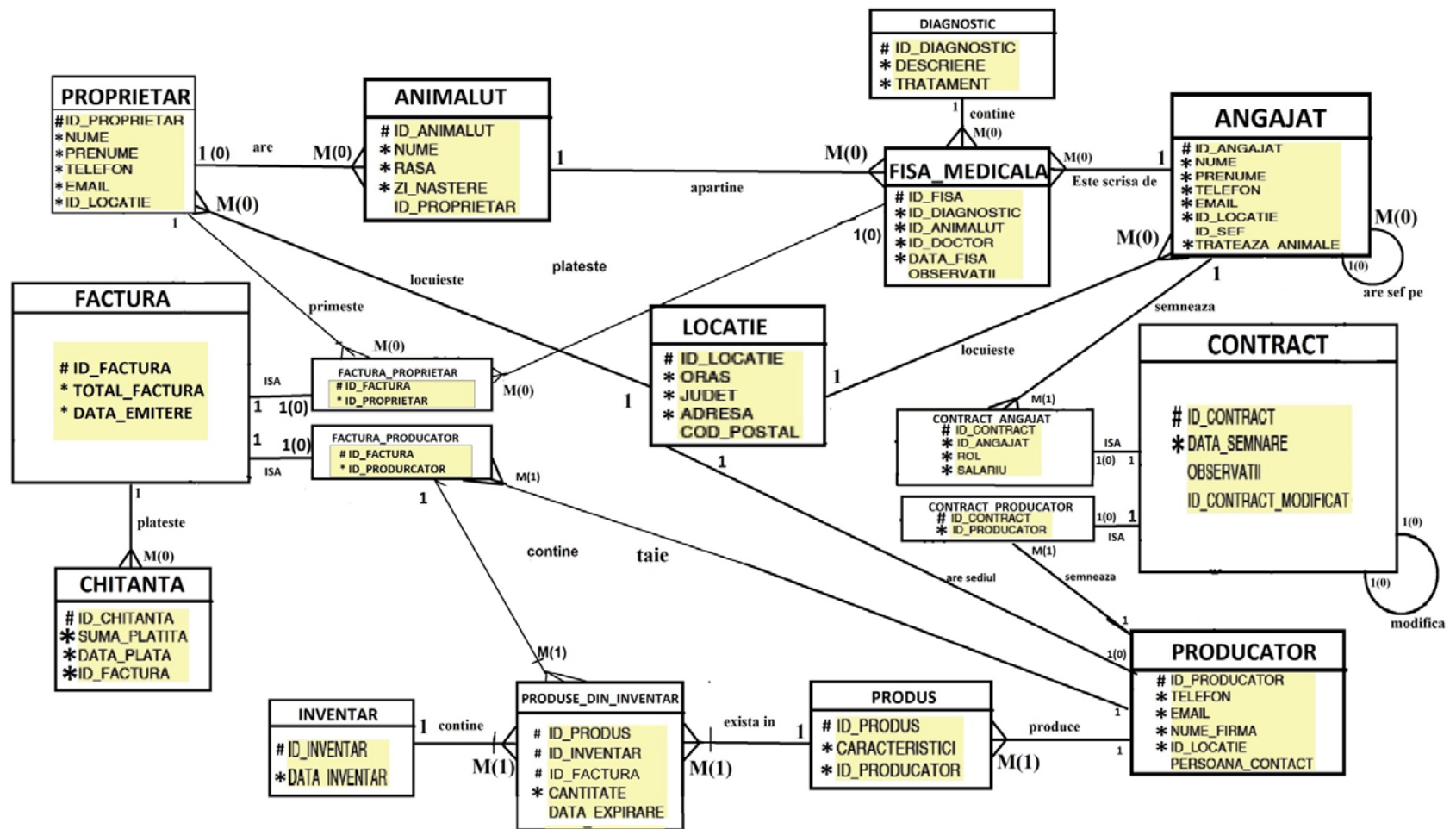
Acest proiect se adresează unei clinici veterinare

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.

PROPRIETAR
- #ID_PROPRIETAR
- *NUME
- *PRENUME
- *TELEFON
- *EMAIL
- *ID_LOCATIE

ANIMALUT
- # ID_ANIMALUT
- *NUME
- *RASA
- *ZI_NASTERE
- ID_PROPRIETAR

DIAGNOSTIC
- # ID_DIAGNOSTIC
- *DESCRIERE
- *TRATAMENT

ANGAJAT
- #ID_ANGAJAT
- *NUME
- *PRENUME
- *TELEFON
- *EMAIL
- *ID_LOCATIE
- ID_SEF
- *TRATEAZA_ANIMALE

FISA_MEDICALA
- # ID_FISA
- *ID_DIAGNOSTIC
- *ID_ANIMALUT
- *ID_DOCTOR
- *DATA_FISA
- OBSERVATII

FACTURA
- # ID_FACTURA
- * TOTAL_FACTURA
- * DATA_EMITERE

LOCATIE
- # ID_LOCATIE
- * ORAS
- * JUDET
- * ADRESA
- COD_POSTAL

CONTRACT
- # ID_CONTRACT
- *DATA_SEMNARE
- OBSERVATII
- ID_CONTRACT_MODIFICAT

FACTURA_PROPRIETAR
- # ID_FACTURA
- * ID_PROPRIETAR

FACTURA_PRODUCATOR
- # ID_FACTURA
- * ID_PRODURCATOR

CONTRACT_ANGAJAT
- # ID_CONTRACT
- *ID_ANGAJAT
- * ROL
- * SALARIU

CONTRACT_PRODUCATOR
- # ID_CONTRACT
- * ID_PRODUCATOR

CHITANTA
- # ID_CHITANTA
- *SUMA_PLATITA
- *DATA_PLATA
- *ID_FACTURA

PRODUCATOR
- # ID_PRODUCATOR
- * TELEFON
- * EMAIL
- * NUME_FIRMA
- * ID_LOCATIE
- PERSOANA_CONTACT

INVENTAR
- # ID_INVENTAR
- *DATA_INVENTAR

PRODUSE_DIN_INVENTAR
- # ID_PRODUS
- # ID_INVENTAR
- # ID_FACTURA
- * CANTITATE
- DATA_EXPIRARE

PRODUS
- # ID_PRODUS
- * CARACTERISTICI
- * ID_PRODUCATOR

Relationships / cardinalities:
- PROPRIETAR — are — ANIMALUT: 1 (0) / M(0)
- ANIMALUT — apartine — FISA_MEDICALA: 1 / M(0)
- DIAGNOSTIC — contine — FISA_MEDICALA: 1 / M(0)
- FISA_MEDICALA — Este scrisa de — ANGAJAT: M(0) / 1
- ANGAJAT — are sef pe: M(0) / 1(0)
- PROPRIETAR — locuieste: M(0)
- PROPRIETAR — primeste: M(0)
- PROPRIETAR — plateste
- FACTURA — ISA — FACTURA_PROPRIETAR: 1 / 1(0)
- FACTURA — ISA — FACTURA_PRODUCATOR: 1 / 1(0)
- FACTURA — plateste — CHITANTA: 1 / M(0)
- ANGAJAT — semneaza — CONTRACT_ANGAJAT
- CONTRACT_ANGAJAT — ISA — CONTRACT: 1(0) / 1
- CONTRACT_PRODUCATOR — ISA — CONTRACT: 1(0) / 1
- CONTRACT — modifica: 1(0) / 1(0)
- ANGAJAT — locuieste — LOCATIE: 1
- LOCATIE — are sediul
- PRODUCATOR — semneaza — CONTRACT_PRODUCATOR: M(1)
- LOCATIE — taie
- FACTURA_PRODUCATOR — contine — PRODUSE_DIN_INVENTAR: M(1) / 1
- INVENTAR — contine — PRODUSE_DIN_INVENTAR: 1 / M(1)
- PRODUSE_DIN_INVENTAR — exista in — PRODUS: M(1) / 1
- PRODUS — produce — PRODUCATOR: M(1) / 1

C##REKSIO.DIAGNOSTIC

P	* ID_DIAGNOSTIC	NUMBER (10)
	* DESCRIERE	VARCHAR2 (4000 BYTE)
	* TRATAMENT	VARCHAR2 (4000 BYTE)
PK_DIAGNOSTIC (ID_DIAGNOSTIC)		
PK_DIAGNOSTIC (ID_DIAGNOSTIC)		

C##REKSIO.ANGAJAT

P	* ID_ANGAJAT	NUMBER (10)
	* NUME	VARCHAR2 (32 BYTE)
	* PRENUME	VARCHAR2 (32 BYTE)
	* TELEFON	NUMBER (10)
U	* EMAIL	VARCHAR2 (64 BYTE)
F	* ID_LOCATIE	NUMBER (10)
F	ID_SEF	NUMBER (10)
	* TRATEAZA_ANIMALE	NUMBER (1)
PK_ANGAJAT (ID_ANGAJAT)		
UNIQUE_EMAIL_ANGAJAT (EMAIL)		
UNIQUE_TEL_ANGAJAT (TELEFON)		
FK_ANGAJAT (ID_SEF)		
FK_ANGAJAT_LOCATIE (ID_LOCATIE)		
PK_ANGAJAT (ID_ANGAJAT)		
UNIQUE_EMAIL_ANGAJAT (EMAIL)		
UNIQUE_TEL_ANGAJAT (TELEFON)		

C##REKSIO.PROPRIETAR

P	* ID_PROPRIETAR	NUMBER (10)
	* NUME	VARCHAR2 (32 BYTE)
	* PRENUME	VARCHAR2 (32 BYTE)
	* TELEFON	NUMBER (10)
U	* EMAIL	VARCHAR2 (64 BYTE)
F	* ID_LOCATIE	NUMBER (10)
PK_PROPRIETAR (ID_PROPRIETAR)		
UNIQUE_EMAIL_PROPRIETAR (EMAIL)		
UNIQUE_TEL_PROPRIETAR (TELEFON)		
FK_PROPRIETAR_LOCATIE (ID_LOCATIE)		
PK_PROPRIETAR (ID_PROPRIETAR)		
UNIQUE_EMAIL_PROPRIETAR (EMAIL)		
UNIQUE_TEL_PROPRIETAR (TELEFON)		

C##REKSIO.ANIMALUT

P	* ID_ANIMALUT	NUMBER (10)
	* NUME	VARCHAR2 (32 BYTE)
	* RASA	VARCHAR2 (32 BYTE)
	ZI_NASTERE	DATE
F	* ID_PROPRIETAR	NUMBER (10)
PK_ANIMALUT (ID_ANIMALUT)		
FK_ANIMALUT_PROPRIETAR (ID_PROPRIETAR)		
PK_ANIMALUT (ID_ANIMALUT)		

C##REKSIO.FISA_MEDICALA

P	* ID_FISA	NUMBER (10)
F	* ID_DIAGNOSTIC	NUMBER (10)
F	* ID_ANIMALUT	NUMBER (10)
F	* ID_DOCTOR	NUMBER (10)
	* DATA_FISA	DATE
	OBSERVATII	VARCHAR2 (4000 BYTE)
PK_FISA_MEDICALA (ID_FISA)		
FK_FISA_ANIMALUT (ID_ANIMALUT)		
FK_FISA_DIAGNOSTIC (ID_DIAGNOSTIC)		
FK_FISA_DOCTOR (ID_DOCTOR)		
PK_FISA_MEDICALA (ID_FISA)		

C##REKSIO.FACTURA_PROPRIETAR

PF	* ID_FACTURA	NUMBER (10)
F	* ID_PROPRIETAR	NUMBER (10)
F	ID_FISA_MEDICALA	NUMBER (10)
PK_FACTURA_PROPRIETAR (ID_FACTURA)		
FK_FACTURA_PROPRIETAR (ID_PROPRIETAR)		
FK_FACTURA_PROPRIETAR_FACT (ID_FACTURA)		
FK_FACTURA_PROPRIETAR_FISA_MEDICALA (ID_FISA_MEDICALA)		
PK_FACTURA_PROPRIETAR (ID_FACTURA)		

C##REKSIO.LOCATIE

P	* ID_LOCATIE	NUMBER (10)
	* ORAS	VARCHAR2 (32 BYTE)
	* JUDET	VARCHAR2 (32 BYTE)
	* ADRESA	VARCHAR2 (256 BYTE)
	COD_POSTAL	NUMBER (10)
PK_LOCATIE (ID_LOCATIE)		
PK_LOCATIE (ID_LOCATIE)		

C##REKSIO.CONTRACT_ANGAJAT

PF	* ID_CONTRACT	NUMBER (10)
F	* ID_ANGAJAT	NUMBER (10)
	* ROL	VARCHAR2 (32 BYTE)
	* SALARIU	NUMBER (5)
PK_CONTRACT_ANGAJAT (ID_CONTRACT)		
FK_CONTRACT_ANGAJAT (ID_ANGAJAT)		
FK_CONTRACT_ANGAJAT_CONTRACT (ID_CONTRACT)		
PK_CONTRACT_ANGAJAT (ID_CONTRACT)		

C##REKSIO.CONTRACT

P	* ID_CONTRACT	NUMBER (10)
	* DATA_SEMNARE	DATE
	OBSERVATII	VARCHAR2 (4000 BYTE)
UF	ID_CONTRACT_MODIFICAT	NUMBER (10)
PK_CONTRACT (ID_CONTRACT)		
UNIQUE_CONTRACT (ID_CONTRACT_MODIFICAT)		
FK_CONTRACT (ID_CONTRACT_MODIFICAT)		
PK_CONTRACT (ID_CONTRACT)		
UNIQUE_CONTRACT (ID_CONTRACT_MODIFICAT)		

C##REKSIO.FACTURA

P	* ID_FACTURA	NUMBER (10)
	* TOTAL_FACTURA	NUMBER (7,2)
	* DATA_EMITERE	DATE
PK_FACTURA (ID_FACTURA)		
PK_FACTURA (ID_FACTURA)		

C##REKSIO.CONTRACT_PRODUCATOR

PF	* ID_CONTRACT	NUMBER (10)
F	* ID_PRODUCATOR	NUMBER (10)
PK_CONTRACT_PRODUCATOR (ID_CONTRACT)		
FK_CONTRACT_PRODUCATOR (ID_PRODUCATOR)		
FK_CONTRACT_PROD_CONTRACT (ID_CONTRACT)		
PK_CONTRACT_PRODUCATOR (ID_CONTRACT)		

C##REKSIO.FACTURA_PRODUCATOR

PF	* ID_FACTURA	NUMBER (10)
F	* ID_PRODUCATOR	NUMBER (10)
PK_FACTURA_PRODUCATOR (ID_FACTURA)		
FK_FACTURA_PRODUCATOR (ID_PRODUCATOR)		
FK_FACTURA_PRODUCATOR_FACT (ID_FACTURA)		
PK_FACTURA_PRODUCATOR (ID_FACTURA)		

C##REKSIO.PRODUCATOR

P	* ID_PRODUCATOR	NUMBER (10)
U	* TELEFON	NUMBER (10)
U	* EMAIL	VARCHAR2 (64 BYTE)
	* NUME_FIRMA	VARCHAR2 (64 BYTE)
UF	* ID_LOCATIE	NUMBER (10)
	PERSOANA_CONTACT	VARCHAR2 (64 BYTE)
PK_PRODUCATOR (ID_PRODUCATOR)		
PRODUCATOR_UNIQUE_LOCATION (ID_LOCATIE)		
UNIQUE_EMAIL_PRODUCATOR (EMAIL)		
UNIQUE_TEL_PRODUCATOR (TELEFON)		
FK_PRODUCATOR_LOCATIE (ID_LOCATIE)		
PK_PRODUCATOR (ID_PRODUCATOR)		
PRODUCATOR_UNIQUE_LOCATION (ID_LOCATIE)		
UNIQUE_EMAIL_PRODUCATOR (EMAIL)		
UNIQUE_TEL_PRODUCATOR (TELEFON)		

C##REKSIO.CHITANTA

P	* ID_CHITANTA	NUMBER (10)
	* SUMA_PLATITA	NUMBER (7,2)
	* DATA_PLATA	DATE
F	* ID_FACTURA	NUMBER (10)
PK_CHITANTA (ID_CHITANTA)		
FK_CHITANTA_FACTURA (ID_FACTURA)		
PK_CHITANTA (ID_CHITANTA)		

C##REKSIO.PRODUSE_DIN_INVENTAR

PF	* ID_PRODUS	NUMBER (10)
PF	* ID_INVENTAR	NUMBER (10)
PF	* ID_FACTURA	NUMBER (10)
	* CANTITATE	NUMBER (5)
	DATA_EXPIRARE	DATE
PK_PRODUSE_INVENTAR (ID_PRODUS, ID_INVENTAR, ID_FACTURA)		
FK_PROD_FACTURA (ID_FACTURA)		
FK_PROD_INVENTAR (ID_INVENTAR)		
FK_PROD_PRODUS (ID_PRODUS)		
PK_PRODUSE_INVENTAR (ID_PRODUS, ID_INVENTAR, ID_FACTURA)		

C##REKSIO.PRODUS

P	* ID_PRODUS	NUMBER (10)
	* CARACTERISTICI	VARCHAR2 (256 BYTE)
F	* ID_PRODUCATOR	NUMBER (10)
PK_PRODUS (ID_PRODUS)		
FK_PRODUS (ID_PRODUCATOR)		
PK_PRODUS (ID_PRODUS)		

C##REKSIO.INVENTAR

P	* ID_INVENTAR	NUMBER (10)
	* DATA_INVENTAR	DATE
PK_INVENTAR (ID_INVENTAR)		
PK_INVENTAR (ID_INVENTAR)		

```sql
--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
```

```sql
(
    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
```

```sql
(
    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),
```

```sql
    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,
```

```sql
    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;
```

```
C##REKSIO> 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)
        )
[2023-02-06 05:36:21] completed in 26 ms
C##REKSIO> 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)
        )
[2023-02-06 05:36:27] completed in 31 ms
C##REKSIO> 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)
        )
[2023-02-06 05:36:33] completed in 25 ms
```

```sql
--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);
```

```sql
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'));
```

```sql
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)
```

```sql
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);
```

```sql
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);
```

```sql
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);
```

```sql
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)
```

```sql
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);
```

```sql
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)
```

```sql
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;
```

```
[2023-08-28 22:47:47] 1 row affected in 5 ms
C##REKSIO> 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')
[2023-08-28 22:47:48] 1 row affected in 5 ms
C##REKSIO> 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'))
[2023-08-28 22:47:48] 1 row affected in 5 ms
C##REKSIO> 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'))
[2023-08-28 22:47:48] 1 row affected in 8 ms
C##REKSIO> 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'))
[2023-08-28 22:47:48] 1 row affected in 4 ms
C##REKSIO> 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')
[2023-08-28 22:47:48] 1 row affected in 8 ms
C##REKSIO> 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')
[2023-08-28 22:47:48] 1 row affected in 5 ms
```

```sql
-- 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 ;
```

```sql
BEGIN
    Encrypt_Proprietar();
end;

SELECT *
FROM ENCKeys;
```

Output   C##REKSIO.ENCKEYS ×   ✓ ↺   ⊞

|◁ < 6 rows ∨ > ▷| ↻ ⏱ ■   + − ↶ ↻ ⇧   Tx: Manual ∨

	ID_KEY ⇅	ENC_KEY	⇕
1	1	B6B357C38370D8A396886C27A6C0A2DC	
2	2	EAB568D7E0CB24331A4237E55F851B69	
3	3	2CBE5DB6027446A7C7C3BCCC8C634DC5	
4	4	E2D961C81204780FB07C72457A1A0F09	
5	5	FBD2BC29FF37FB3E8AEA3A853D4764B3	
6	6	5C371FFC6614F7AA6955889759188D4B	

```sql
SELECT *
FROM PROPRIETAR_ENCRYPTED;
```

Output   C##REKSIO.PROPRIETAR_ENCRYPTED ×   ✓ ↺   ⊞

|◁ < 6 rows ∨ > ▷| ↻ ⏱ ■ + − ↶ ⇧   Tx: Manual ∨ ✓ ↺   DDL ⬚   CSV ∨ ⬇ ⬆ ⤴ ⊙

	ID_PROPRIETAR	NUME	PRENUME	TELEFON	EMAIL	ID_LOCATIE
1	1	BC7002B91B057ACEA6747C17FF44E4BC	F5AA6E8356303B805B60514CC31DB407	1E8E9EA162C213F6C9486029897A8102	0x19182C06D67ECC624D92F541A256A14643426BF042B47948A9AA215…	7C6DEC331737E689486B68CA86A89E56
2	2	E2922D7097D5182F17D0AFCF87D199B0	7675CA2D5BA116AE152A89AD954BABAD	08101F5D06BA6D49A85438D390EB7A59	0xD9C4AEFBC0A1E9BDDA94493875D079A0A2B483B6221D4B17C35D162…	E95950217366546DC14F3D5826B55F35
3	3	E5100ED41D002D1F015D9E49A9997FD6	6BC9C3EBE1E110C1582DA258CFEFCB26	1111422654A4ADA3CEAE54281C88B33B	4FD2A5D4524DCF18095335641B86C4FD	53BE4439FD3713BDE7B258BD8F00D0F2
4	4	88567C7E1BD307CD903A8405AAF5FF6B	AB83FD75C5179BFA280145605D794CE38	D1C4E56A5342265048E3C62018586A11	0xDC94EE94F1E2EFBA2CE87BEC7FC86BE3771DF74E68124DB54C249D4…	761487D1295440E559CB8E4A93A97FEF
5	5	A4C177D54DFC82581D22C6994E4FBF46	7327EC733B6D317EE2C551362BBF2EEA	B9708A5775F9E571E703258CB5859658	0xCD5486014CEA986A5AD44AA0B66205F6F3A02805BEC9D1AB8C12FBE…	188CF08687EED4BA59A0A37087861F00
6	6	9A14956FD5988527DA082293D3D32FD5	4A6F3A0B7214C4247861C2CBEC8FFC75	0553F815E2214571D52FA3AD6E7EA348	0xA0CDC5611B67F06745B3F8CCEFDCCD91F3F0C1801ACEB33AB3C4636…	B5ECE7123DEB28C101A653CDB7234CD4

```sql
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
```

```sql
    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;
```

⊢ |◁ 6 rows ∨ 〉 〉| ↻ ⊙ ■ │ + − ↺ ◉ ⇧ Tx: Manual ∨ ✓ ↺ │ DDL 📌

	ID_PROPRIETAR	NUME	PRENUME	TELEFON	EMAIL	ID_LOCATIE
1	1	Staicu	Octavian-Florin	736659442	octavian.staicu@s.unibuc.ro	1
2	2	Bodea	George	737392518	george@yahoo.com	2
3	3	Staicu	Adrian	736659441	adi@gmail.com	1
4	4	Popescu	Ion	771654789	popescu.ion@gmail.com	3
5	5	Cineva	Interesant	777111999	jmekerie@my.fmi.unibuc.ro	4
6	6	Fara	Animalut	729123123	cumparator@my.fmi.unibuc.ro	2

```
-- 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;
```

	OBJ$NAME	SQLTEXT	USERID
1	FISA_MEDICALA	<null>	C##REKSIO
2	FISA_MEDICALA	<null>	C##REKSIO
3	FISA_MEDICALA	<null>	C##REKSIO
4	FISA_MEDICALA	<null>	C##REKSIO
5	FISA_MEDICALA	<null>	C##REKSIO

```
-- 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;
```

SEC_ID	USER_NAME	SESSION_NAME	DATA_MODIFICARE	OBSERVATII_VECHI	OBSERVATII_NOI
22 C##REKSIO	DELL-E5570	2024-01-24 18:40:49	obs audit	obs audit 2

```sql
-- 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(
```

```sql
                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;
    C##REKSIO> UPDATE FISA_MEDICALA
               set observatii = 'obs audit 3'
               where id_fisa = 1113
    [2024-01-24 22:34:03] completed in 23 ms
    Audit Policy: [object_schema = C##REKSIO]; [object_name = FISA_MEDICALA]; [policy_name = POLICY_FISA_MEDICALA]


SELECT *
FROM dba_fga_audit_trail
ORDER BY TIMESTAMP DESC;
```

	SESSION_ID	TIMESTAMP	DB_USER	OS_USER	USERHOST	CLIENT_ID	ECONTEXT_ID	EXT_NAME	OBJECT_SCHEMA	OBJECT_NAME	POLICY_NAME	SCN	SQL_TEXT
1	280056	2024-01-24 17:21:42	C##REKSIO	User	DELL-E5570	<null>	<null>	<null>	C##REKSIO	FISA_MEDICALA	POLICY_FISA_MEDICALA	23553258	UPDATE FISA_MEDICAL
2	280056	2024-01-24 17:20:04	C##REKSIO	User	DELL-E5570	<null>	<null>	<null>	C##REKSIO	FISA_MEDICALA	POLICY_FISA_MEDICALA	23553160	UPDATE FISA_MEDICAL

CSV

SQL_BIND	COMMENT$TEXT	STATEMENT_TYPE	EXTENDED_TIMESTAMP	PROXY_SESSIONID	GLOBAL_UID	INSTANCE_NUMBER	OS_PROCESS	TRANSACTIONID	STATEMENTID	ENTRYID
1 ll>	DB_UNIQUE_NAME="XE"	UPDATE	2024-01-24 17:21:42.756000 +00:00	<null>	<null>	0	17256:6172	0x0000000000000000	1810	35 <
2 ll>	DB_UNIQUE_NAME="XE"	UPDATE	2024-01-24 17:20:04.839000 +00:00	<null>	<null>	0	17256:6172	0x0000000000000000	1719	32 <

ENTRYID	OBJ_EDITION_NAME	DBID	RLS_INFO	CURRENT_USER
35	<null>	3027188878	<null>	C##REKSIO
32	<null>	3027188878	<null>	C##REKSIO

```sql
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					

2. 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

3. 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

```sql
-- 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';
```

	TABLESPACE_NAME	USERNAME	BYTES	MAX_BYTES	BLOCKS	MAX_BLOCKS	DROPPED
1	USERS	C##ADMINISTRATOR	0	-1	0	-1	NO
2	USERS	C##GUEST_DOCTOR	0	1048576	0	128	NO
3	USERS	C##CONTABIL	0	-1	0	-1	NO
4	USERS	C##GUEST_FINANTE	0	1048576	0	128	NO
5	USERS	C##DOCTOR	0	-1	0	-1	NO

```sql
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##%';
```

	USERNAME	USER_ID	PASSWORD	ACCOUNT_STATUS	LOCK_DATE	EXPIRY_DATE	DEFAULT_...	TEMPORAR...	LOCAL_T...	CREATED	PROFILE	
1	C##REKSIO	109	<null>	OPEN	<null>	2024-02-24 16:29:56	USERS	TEMP	TEMP	2023-08-28 16:29:56	DEFAULT	DE
2	C##CONTABIL	112	<null>	EXPIRED	<null>	2024-01-24 19:55:12	USERS	TEMP	TEMP	2024-01-24 19:55:12	C##EDIT_PROFILE	DE
3	C##DOCTOR	111	<null>	EXPIRED	<null>	2024-01-24 19:55:06	USERS	TEMP	TEMP	2024-01-24 19:55:06	C##EDIT_PROFILE	DE
4	C##GUEST_DOCTOR	118	<null>	EXPIRED	<null>	2024-01-24 22:07:22	USERS	TEMP	TEMP	2024-01-24 22:07:22	C##GUEST_PROFILE	DE
5	C##GUEST_FINANTE	119	<null>	EXPIRED	<null>	2024-01-24 22:07:29	USERS	TEMP	TEMP	2024-01-24 22:07:29	C##GUEST_PROFILE	DE
6	C##ADMINISTRATOR	110	<null>	EXPIRED	<null>	2024-01-24 19:54:56	USERS	TEMP	TEMP	2024-01-24 19:54:56	C##FULL_PROFILE	DE

```sql
-- 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;
```

```sql
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;
```

```sql
GRANT C##GUEST_DOCTOR_ROLE to C##GUEST_DOCTOR;
GRANT C##GUEST_FINANCE_ROLE to C##GUEST_FINANTE;
```

```sql
-- 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;
```

```
C##REKSIO> 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;
[2024-01-24 21:31:59] completed in 5 ms
Night Shift Value: YES
```