

Centrul de adopție pentru animalele de companie "Happy Puppy"

Tache Daria Elena
Grupa 234



~ Cuprins ~

1. [Descrierea modelului real, a utilității acestuia și a regulilor de funcționare](#)
2. [Prezentarea constrângerilor \(restricții, reguli\) impuse asupra modelului](#)
3. [Descrierea entităților, incluzând precizarea cheii primare.](#)
4. [Descrierea relațiilor, incluzând precizarea cardinalității acestora](#)
5. [Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicite, valori posibile ale tuturor atributelor](#)
6. [Diagrama entitate-relație](#)
7. [Diagrama conceptuală](#)
8. [Enumerarea schemelor relaționale corespunzătoare diagramei entitate-relație proiectate la punctul 7](#)
9. [Normalizarea până la forma normală 3 \(FN1- FN3\)](#)
10. [Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele \(punctul 11\).](#)
11. [Crearea tabelelor în SQL și inserarea de date coerente în fiecare dintre acestea \(minimum 5 înregistrări în fiecare tabel neasociativ; minimum 10 înregistrări în tabelele asociative\).](#)
12. Formulați în limbaj natural și [implementați 5 cereri SQL complexe](#) ce vor utiliza, în ansamblul lor, următoarele elemente:
 - subcereri sincronizate în care intervin cel puțin 3 tabele
 - subcereri nesincronizate în clauza FROM
 - grupări de date, funcții grup, filtrare la nivel de grupuri cu subcereri nesincronizate (în clauza de HAVING) în care intervin cel puțin 3 tabele (în cadrul aceleiași cereri)
 - ordonări și utilizarea funcțiilor NVL și DECODE (în cadrul aceleiași cereri)
 - utilizarea a cel puțin 2 funcții pe șiruri de caractere, 2 funcții pe date calendaristice, a cel puțin unei expresii CASE
 - utilizarea a cel puțin 1 bloc de cerere (clauza WITH)

[13. Implementarea a 3 operații de actualizare și de ștergere a datelor utilizând subcereri.](#)

~ Modelul real ~

Proiectul are ca scop crearea unei baze de date pentru fundația de adopție al animăluțelor de companie “Happy Puppy”, cu cabinete în întreaga țară. În acest fel, suntem mai aproape de persoanele ce vor să ofere micilor prieteni șansa la o viață alături de o familie iubitoare.

Centrele noastre vă oferă mai multe opțiuni: puteți adopta un animăluț, puteți dona pentru micii amici pufoși sau puteți veni cu copiii în baza unei programări la joacă!

Procesul de adopție este mai ușor ca niciodată prin contractul de adopție, iar colegii noștri vă vor fi alături la fiecare pas. La final veți primi carnetul medical al noului vostru animăluț de companie, cu vaccinurile la zi!

~ Prezentarea constrângerilor ~

Fiecare animal de companie pus spre adopție are o fișă unică asociată, care conține date despre acesta, și carnetul medical corespunzător.

Carnetul medical conține date despre starea de sănătate a animalului, și mereu va exista o fișă la care să fie anexat.

Prin contractul de adopție se face legătura dintre un animal pus spre adopție și un client dispus să adopte.

La fiecare centru sunt înscrise un număr maxim de 30 de animale de companie.

Considerăm client o persoană care face cel puțin una dintre acțiunile următoare: adoptă un animaluț, donează către oricare din centrele noastre, sau face o programare pentru o vizită una unul din centre.

Donățiile sunt anonime și sunt făcute în contul unui singur centru. În cazul în care se dorește a se dona mai multor centre, se fac mai multe donații, fiecare în contul centrului dorit.

Înainte de normalizare:

Vizitele se pot face la oricare dintre centrele noastre de Luni până Vineri. Cu o programare puteți alege să vizitați unul sau mai multe centre pe care le specificați când faceți programarea, pe tot parcursul zilei. Programarea se putea face pe unul sau mai multe centre pentru a favoriza adoptarea animaluțelor (Dacă un client nu adoptă din primul centru, va avea posibilitatea să meargă și la restul centrelor pentru care a făcut programarea și să adopte de acolo).

După normalizare:

Vizitele se pot face la oricare dintre centrele noastre de Luni până Vineri fie în cadrul unui TUR care are următoarele intervale orare: 1(08:00-13:00) sau 2(17:00- 20:00). Programarea poate fi făcută la centru, prin platforma online, sau la telefon.

Un centru va avea cel puțin doi angajați.

~ Descrierea entităților ~

ANIMAL_COMPANIE are cheie primară *#petID*, cheile externe *#id_fișă* (care conține informații despre fiecare animal de companie în parte, alături de fișa medicală corespunzătoare), *#id_centru* (care reține datele centrului în care a fost repartizat animaluțului) și atributul *dată_preluare* unde este specificată data la care acesta a fost luat în centru.

FIȘA_AC are drept cheie primară *#id_fișă*, attributele *specie*, *rasă*, *sex* și este superentitate pentru *CARNET_MEDICAL*.

CARNET_MEDICAL are cheia primară moștenită de la *FIȘA_AC* (*#id_fișă*) întrucât este în strânsă relație cu aceasta, și nu este folosită în mod individual în baza noastră de date. Moștenește attributele *specie*, *rasă* și *sex*, însă are și attributele sale proprii cum ar fi: *vârsta*, *vaccinuri*, *sterilizat*, *deparazitat* (iar pentru ultimele două se poate alege doar 'DA' sau 'NU') și *mențiuni*.

CLIENT este tabelul ce stochează date despre clienții noștri. Acesta are cheia primară *id_client*, și attributele *nume*, *prenume*, *telefon*, *adresa* și *email*.

CENTRU este tabelul care ține evidența locațiilor noastre. Are cheia primară *#id_centru*, și reține date precum: *adresa*, *nr_tel*, *program* și *nr_animale_companie*.

CONTRACT_ADOPTIE face legătura dintre un client doritor să adopte și un animaluț dintr-un centru. Are drept cheie primară *#id_contract*, și chei externe precum *#id_pet* și *#id_client*. Singurul său atribut este *data_adoptie*.

DONAȚIE reține date despre donațiile făcute de către clienți. Are cheia primară *#id_donație*, face legătura cu centrul care va beneficia de donație prin cheia externă *#id_centru* și clientul care a făcut donația prin cheia externă *#id_client*. Are și attributele *data_donație* și *suma*.

PROGRAMARE_VIZITA are drept cheie primară *#id_prog*. Cheile externe fac legătura cu clientul (*#id_client*) care a inițiat programarea și turul (*#id_tur*) la care s-a înscris, iar atributele de sine stătătoare sunt *nr_pers*, *data_prog*, și *interval_orar*.

TUR este entitatea care face legătura între o programare făcută de un client și centrul la care se ține acel tur. Are ca scop să evite conflictele în baza de date. Cheia primară este *#id_tur*, iar cea externă este *#id_centru*. Mai ține minte și atributul *detalii*.

ANGAJAT ține evidența angajaților din fiecare centru. Are drept cheie primară *#id_angajat*, iar cheia externă provine de la centrul în care lucrează fiecare angajat (*#id_centru*). Atributele sale sunt: *nume*, *prenume*, *telefon*, *data_angajare*, *ore_săptămână* și *salariu*.

~ Descrierea relațiilor ~

CARNET_MEDICAL se află în relație de 1 la 1 cu *FIȘA_AC*, și depinde de aceasta. O *fișă* trebuie să aibă un *carnet medical* asociat, iar un *carnet medical* trebuie să aibă o singură *fișă* căreia să i se poată atașa. (1 – 1)

FIȘA_AC este dependentă de *ANIMAL_COMPANIE* având în vedere că o *fișă* trebuie să fie atașată unui singur animal de companie și nu poate exista altfel, iar pentru fiecare *animal de companie* trebuie făcută o *fișă*. (1 – 1)

ANIMAL_COMPANIE și *CONTRACT_ADOPTIE* sunt legate printr-o relație de one-to-one deoarece pentru un *animal de companie* poate fi creat un *contract de adopție*, iar un *contract de adopție* trebuie să fie creat în cazul unei adopții. (1(0) - 1).

Un *CLIENT* poate să facă unul sau mai multe *contracte de adopție*, în timp ce un *CONTRACT_ADOPTIE* trebuie să fie făcut de un singur *client*. (1 – M(0))

Un *CLIENT* poate să facă unul sau mai multe *donații*, în timp ce o *DONAȚIE* trebuie să fie făcut de un singur *client*. (1 – M(0)).

Un *CLIENT* poate face una sau mai multe *PROGRAMARI_VIZITA*, dar o singură *PROGRAMARE_VIZITA* poate fi făcută de un singur *client*. A nu se confunda cu faptul ca toți clienții au acest drept. (M(0) – 1)).

Se poate face o *PROGRAMARE_VIZITA* la unul sau mai multe *CENTRE* iar un *CENTRU* poate găzdui una sau mai multe *PROGRAMARI_VIZITA*. (M(0) - M(0)).

Acest lucru se va rezolva prin adăugarea entității *TUR*. Așadar, un *client* va face o *programare_vizita* pentru un anumit *tur* organizat de la un anume *centru*.

Așadar, relațiile devin:

O *programare_vizita* este destinată unui singur TUR dar pentru un TUR se pot face una sau mai multe PROGRAMARI_VIZITA. $(1 - M(0))$.

Un *TUR* se va ține la un singur *CENTRU*, iar un *centru* poate avea unul sau mai multe tururi. $(1 - M(0))$.

Un *ANGAJAT* va lucra la un singur *CENTRU*, iar un *centru* va avea mai mulți angajați. $(1 - M(1))$.

~ Descrierea atributelor ~

FISA_AC

- id_fisa: INT, PRIMARY KEY - identificator unic pentru fișa animalului.
- specie: VARCHAR2(50) - specia animalului.
- rasa: VARCHAR2(100) - rasa animalului.
- sex: CHAR(1), CHECK (sex IN ('M', 'F')) - sexul animalului ('M' sau 'F').

CARNET_MEDICAL

- id_fisa: INT, PRIMARY KEY - referință către FISA_AC(id_fisa) - CARNET MEDICAL este subentitate pentru FISA_AC și pentru că sunt în relație de 1-1 nu au nevoie de două chei diferite.
- varsta: INT - vârsta animalului în ani.
- vaccin: VARCHAR2(500) - lista de vaccinuri administrate.
- sterilizat: CHAR(2), CHECK (sterilizat IN ('DA', 'NU')) - dacă animalul este sterilizat ('DA' sau 'NU').
- deparazitat: CHAR(2), CHECK (deparazitat IN ('DA', 'NU')) - dacă animalul este deparazitat ('DA' sau 'NU').
- mentiuni: VARCHAR2(500) - alte mențiuni medicale, un șir de caractere de lungime maxima 500.
- FOREIGN KEY(id_fisa) REFERENCES FISA_AC(id_fisa) ON DELETE CASCADE – pentru că nu poate exista un CARNET_MEDICAL fără o FISA_MEDICALA, în cazul în care ștergem o intrare a unui carnet, vrem să ștergem și fișa medicală asociată.

CENTRU

- id_centru: INT, PRIMARY KEY - identificator unic pentru centrul de adopție.
- adresa: VARCHAR2(200) - adresa centrului.
- nr_tel: VARCHAR2(20) - numărul de telefon al centrului.
- program: VARCHAR2(200) - programul de lucru al centrului.
- nr_animale_companie: INT - numărul de animale de companie în centru.

ANIMAL_COMPANIE

- id_pet: INT, PRIMARY KEY - identificator unic pentru animalul de companie.
- id_centru: INT, FOREIGN KEY - referință către CENTRU(id_centru).
- id_fisa: INT, FOREIGN KEY - referință către FISA_AC(id_fisa), ON DELETE CASCADE.
- data_preluare: DATE - data la care animalul a fost preluat în centru.

CLIENT

- id_client: INT, PRIMARY KEY, AUTO_INCREMENT - identificator unic pentru client.
- nume: VARCHAR2(20) - numele clientului.
- prenume: VARCHAR2(50) - prenumele clientului.
- telefon: VARCHAR2(20) - numărul de telefon al clientului.
- adresa: VARCHAR2(200) - adresa clientului.
- email: VARCHAR2(50) - adresa de email a clientului.

TUR

- id_tur: INT, PRIMARY KEY
- id_centru: INT, FOREIGN KEY - referință către CENTRU(id_centru).
- detalii: VARCHAR2(50) - detalii despre turul prestabilit.

PROGRAMARE_VIZITA

- id_prog: INT, PRIMARY KEY - identificator unic pentru programarea vizitei.
- id_client: INT, FOREIGN KEY - referință către CLIENT(id_client).
- id_tur: INT, FOREIGN KEY - referință către TUR(id_tur).
- nr_pers: INT - numărul de persoane care participă la vizită.
- data_prog: DATE - data programării.
- interval_orar: CHAR(1) - intervalul pentru turul la care a fost făcută programarea.

DONATIE

- id_donatie: INT, PRIMARY KEY - identificator unic pentru donație.
- id_centru: INT, FOREIGN KEY - referință către CENTRU(id_centru).

- id_client: INT, FOREIGN KEY - referință către CLIENT(id_client).
- data_donatie: DATE - data donației.
- suma: INT - suma donată.

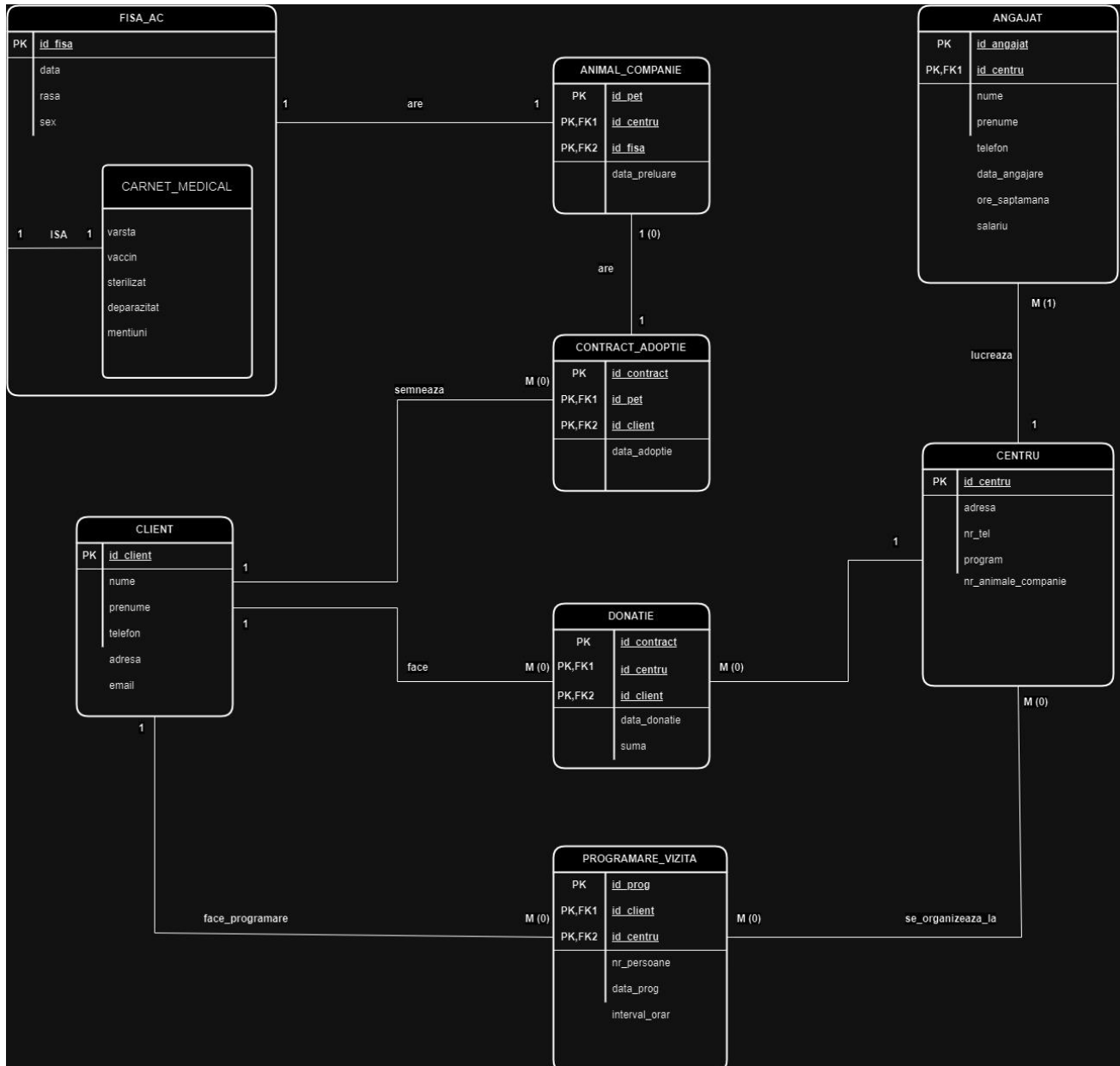
CONTRACT_ADOPTIE

- id_contract: INT, PRIMARY KEY - identificator unic pentru contractul de adopție.
- id_pet: INT, FOREIGN KEY - referință către ANIMAL_COMPANIE(id_pet).
- id_client: INT, FOREIGN KEY - referință către CLIENT(id_client).
- data_adopție: DATE - data adopției.

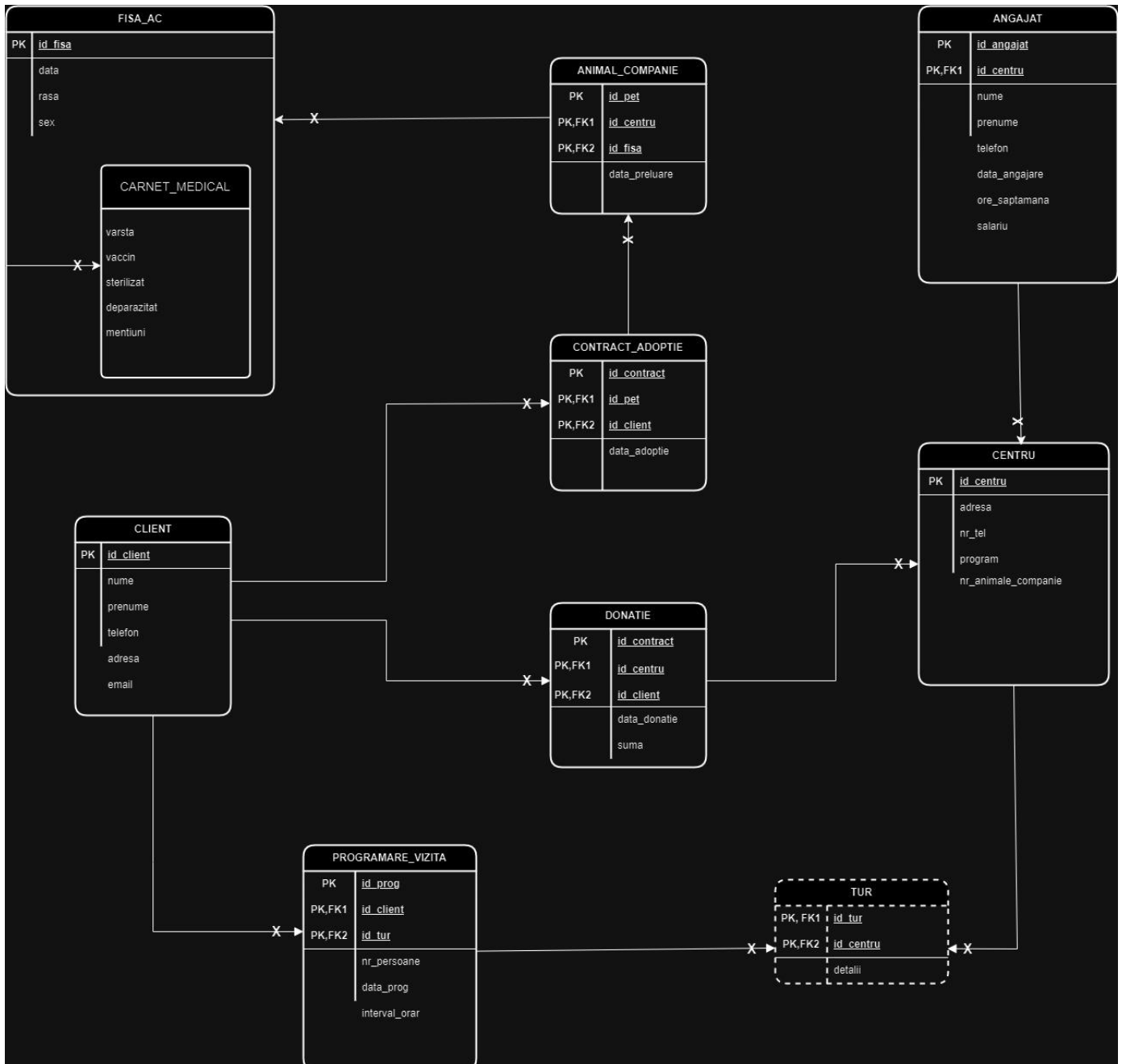
ANGAJAT

- id_angajat: INT, PRIMARY KEY - identificator unic pentru angajat.
- id_centru: INT, FOREIGN KEY - referință către CENTRU(id_centru).
- nume: VARCHAR2(20) - numele angajatului.
- prenume: VARCHAR2(50) - prenumele angajatului.
- telefon: VARCHAR2(20) - numărul de telefon al angajatului.
- data_angajare: DATE - data angajării.
- ore_saptamana: INT - numărul de ore lucrate pe săptămână.
- salariu: INT - salariul angajatului.

~Diagrama Entitate – Relație~



~Diagrama Conceptuală~



~Enumerarea schemelor relaționale~

FISA_AC(id_fisa#, specie, rasa, sex)

CARNET_MEDICAL(id_fisa#, varsta, vaccin, sterilizat, deparazitat, mentiuni)

CENTRU(id_centru#, adresa, nr_tel, program, nr_animale_companie)

ANIMAL_COMPANIE(id_pet#, id_centru#, id_fisa#, data_preluare)

CLIENT(id_client#, nume, prenume, telefon, adresa, email)

TUR(id_tur#, id_centru#, detalii)

PROGRAMARE_VIZITA(id_prog#, id_client#, id_tur#, nr_pers, data_prog, interval_orar)

DONATIE(id_donatie#, id_centru#, id_client#, data_donatie, suma)

CONTRACT_ADOPTIE(id_contract#, id_pet#, id_client#, data_adoptie)

ANGAJAT(id_angajat#, id_centru#, nume, prenume, telefon, data_angajare, ore_saptamana, salariu)

~ Normalizarea FN1- FN3 ~

Schema inițială îndeplinește FN1 prin faptul că toate coloanele au valori atomice și fiecare tabel are o cheie primară unică.

Forma normala 2 este îndeplinită datorită faptului că toate coloanele non-cheie depind de întreaga cheie primară.

Pentru a fi în FN3, tabelele nu trebuie să aibă dependențe tranzitive. În această schemă toate atributele non-cheie sunt dependente direct de cheia primară, așadar respectă FN3.

Exemplu non-FN1:

Presupunem că am fi stocat în tabela CLIENT un singur string care să rețină toate informațiile pentru un client (nume, prenume, telefon, adresa, email). Acest exemplu încalcă FN1 deoarece valorile nu ar mai fi atomice

Exemplu non-FN2:

În cazul în care am fi avut o cheie primară compusă pentru tabelul DONATIE formată din ('id_donatie', 'id_centru') și am fi reținut și 'nume_client' în acest tabel, ar exista o dependență parțială de 'id_client', dar nu de întreaga cheie primară compusă, încalcând FN2.

Exemplu non-FN3:

Un exemplu bun de încălcare a integrității formei normale 3 ar putea fi următorul: adăugăm 'adresa_centru' în tabelul ANGAJAT. În acest caz, 'adresa_centru' depinde de 'id_centru', care la randul său depinde de cheia primară 'id_angajat'.

~ Crearea unei secvențe utilizată la inserări în tabele ~

```
CREATE SEQUENCE seq_fisa_ac START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
```

~ Crearea tabelelor in SQL și inserarea de date coerente ~

```
CREATE TABLE FISA_AC(  
    id_fisa INT PRIMARY KEY,  
    specie VARCHAR2(50),  
    rasa VARCHAR2(100),  
    sex CHAR(1)  
);  
  
CREATE TABLE CARNET_MEDICAL(  
    id_fisa INT PRIMARY KEY,  
    varsta INT,  
    vaccin VARCHAR2(500),  
    sterilizat CHAR(2) CHECK (sterilizat IN ('DA', 'NU')),  
    deparazitat CHAR(2) CHECK (deparazitat IN ('DA', 'NU')),  
    mentiuni VARCHAR2(500),  
    FOREIGN KEY(id_fisa) REFERENCES FISA_AC(id_fisa) ON DELETE CASCADE  
);  
  
CREATE TABLE CENTRU(  
    id_centru INT PRIMARY KEY,  
    adresa VARCHAR2(200),  
    nr_tel VARCHAR2(20),  
    program VARCHAR2(200),  
    nr_animale_companie INT  
);  
  
CREATE TABLE ANIMAL_COMPANIE(  
    id_pet INT PRIMARY KEY,  
    id_centru INT,  
    id_fisa INT,  
    data_preluare DATE,  
    FOREIGN KEY (id_centru) REFERENCES CENTRU(id_centru),  
    FOREIGN KEY (id_fisa) REFERENCES FISA_AC(id_fisa) ON DELETE CASCADE
```

```
);

CREATE TABLE CLIENT(
    id_client INT PRIMARY KEY,
    nume VARCHAR2(20),
    prenume VARCHAR2(50),
    telefon VARCHAR2(20),
    adresa VARCHAR2(200),
    email VARCHAR2(50)
);

CREATE TABLE TUR(
    id_tur INT PRIMARY KEY,
    id_centru INT,
    detalii VARCHAR2(50),
    FOREIGN KEY (id_centru) REFERENCES CENTRU(id_centru)
);

CREATE TABLE PROGRAMARE_VIZITA(
    id_prog INT PRIMARY KEY,
    id_client INT,
    id_tur INT,
    nr_pers INT,
    data_prog DATE,
    interval_orar CHAR(1) CHECK (interval_orar IN ('1', '2')),
    FOREIGN KEY (id_client) REFERENCES CLIENT(id_client),
    FOREIGN KEY (id_tur) REFERENCES TUR(id_tur)
);

CREATE TABLE DONATIE(
    id_donatie INT PRIMARY KEY,
    id_centru INT,
    id_client INT,
    data_donatie DATE,
    suma INT,
    FOREIGN KEY(id_centru) REFERENCES CENTRU(id_centru),
    FOREIGN KEY(id_client) REFERENCES CLIENT(id_client)
);

CREATE TABLE CONTRACT_ADOPTIE(
    id_contract INT PRIMARY KEY,
    id_pet INT,
    id_client INT,
    data_adoptie DATE,
    FOREIGN KEY (id_pet) REFERENCES ANIMAL_COMPANIE(id_pet),
```

```
FOREIGN KEY (id_client) REFERENCES CLIENT(id_client)
);

CREATE TABLE ANGAJAT(
    id_angajat INT PRIMARY KEY,
    id_centru INT,
    nume VARCHAR2(20),
    prenume VARCHAR2(50),
    telefon VARCHAR2(20),
    data_angajare DATE,
    ore_saptamana INT,
    salariu INT,
    FOREIGN KEY (id_centru) REFERENCES CENTRU(id_centru)
);
```

----- SEQUENCE SI INSERARI CENTRU -----

```
CREATE SEQUENCE seq_centru START WITH 100 INCREMENT BY 1 NOCACHE NOCYCLE;
```

```
INSERT INTO CENTRU (id_centru, adresa, nr_tel, program, nr_animale_companie) VALUES
(seq_centru.NEXTVAL, 'Strada Florilor, Nr. 5', '0712345678', 'Luni-Vineri 9-17', 0);
INSERT INTO CENTRU (id_centru, adresa, nr_tel, program, nr_animale_companie) VALUES
(seq_centru.NEXTVAL, 'Strada Marilor, Nr. 8', '0722345678', 'Luni-Vineri 9-18', 0);
INSERT INTO CENTRU (id_centru, adresa, nr_tel, program, nr_animale_companie) VALUES
(seq_centru.NEXTVAL, 'Strada Trandafirilor, Nr. 10', '0732345678', 'Luni-Vineri 8-
16', 0);
INSERT INTO CENTRU (id_centru, adresa, nr_tel, program, nr_animale_companie) VALUES
(seq_centru.NEXTVAL, 'Strada Lalelelor, Nr. 12', '0742345678', 'Luni-Vineri 10-18',
0);
INSERT INTO CENTRU (id_centru, adresa, nr_tel, program, nr_animale_companie) VALUES
(seq_centru.NEXTVAL, 'Strada Violetei, Nr. 15', '0752345678', 'Luni-Vineri 9-17',
0);
INSERT INTO CENTRU (id_centru, adresa, nr_tel, program, nr_animale_companie) VALUES
(seq_centru.NEXTVAL, 'Strada Zambilei, Nr. 20', '0762345678', 'Luni-Vineri 8-15',
0);
```

----- SEQUENCE SI INSERARI FISA SI CARNET_AC -----

```
CREATE SEQUENCE seq_fisa_ac START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;

INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Caine', 'Labrador', 'M');
INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Pisica', 'Siameza', 'F');
INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Caine', 'Beagle', 'M');
INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Pisica', 'Persana', 'F');
INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Caine', 'Bulldog', 'M');
INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Pisica', 'Maine Coon', 'F');
INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Caine', 'Golden Retriever', 'M');
INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Pisica', 'Bengaleza', 'F');
INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Caine', 'Micut', 'F');
INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Pisica', 'Siameza', 'M');
INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Caine', 'Labrador', 'F');
INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL,
'Pisica', 'Persana', 'M');

INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazit,
mentiuni) VALUES (1 , 2, 'Rabie', 'DA', 'NU', 'None');
INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazit,
mentiuni) VALUES (2 , 3, 'Parvo', 'NU', 'DA', 'Checkup needed');
INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazit,
mentiuni) VALUES (3 , 1, 'Distemper', 'DA', 'DA', 'Healthy');
INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazit,
mentiuni) VALUES (4 , 4, 'Leptospirosis', 'NU', 'NU', 'Needs vaccination');
INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazit,
mentiuni) VALUES (5 , 2, 'Bordetella', 'DA', 'DA', 'None');
INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazit,
mentiuni) VALUES (6 , 3, 'Rabie', 'NU', 'NU', 'Slightly underweight');
INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazit,
mentiuni) VALUES (7 , 1, 'Parvo', 'DA', 'DA', 'Recovered from illness');
INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazit,
mentiuni) VALUES (8 , 4, 'Distemper', 'DA', 'NU', 'None');
```

```
INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat,
mentiuni) VALUES (9 , 1, 'Distemper', 'DA', 'NU', 'Healthy');
INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat,
mentiuni) VALUES (10 , 2, 'Leptospirosis', 'NU', 'NU', 'Needs vaccination');
INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat,
mentiuni) VALUES (11 , 2, 'Bordetella', 'NU', 'DA', 'None');
INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat,
mentiuni) VALUES (12 , 5, 'Rabie', 'NU', 'NU', 'None');
```

-----SEQUENCE SI INSERARI ANIMAL_COMPANIE-----

```
CREATE SEQUENCE seq_animal_companie START WITH 1400 INCREMENT BY 1 NOCACHE NOCYCLE;
```

```
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 100, 1, DATE '2023-01-01');
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 101, 2, DATE '2023-02-15');
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 100, 3, DATE '2023-03-10');
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 103, 4, DATE '2023-04-20');
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 104, 5, DATE '2023-05-05');
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 100, 6, DATE '2023-06-18');
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 105, 7, DATE '2023-07-22');
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 100, 8, DATE '2023-08-30');
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 102, 9, DATE '2023-05-05');
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 103, 10, DATE '2023-06-18');
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 104, 11, DATE '2023-07-22');
INSERT INTO ANIMAL_COMPANIE (id_pet, id_centru, id_fisa, data_preluare) VALUES
(seq_animal_companie.NEXTVAL, 101, 12, DATE '2023-08-30');
```

-----SEQUENCE SI INSERARI CLIENT-----

```
CREATE SEQUENCE seq_client START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
```

```
INSERT INTO CLIENT (id_client, nume, prenume, telefon, adresa, email) VALUES
(seq_client.NEXTVAL, 'Popescu', 'Ion', '0799123456', 'Strada Verde, Nr. 10',
'ion.popescu@example.com');
INSERT INTO CLIENT (id_client, nume, prenume, telefon, adresa, email) VALUES
(seq_client.NEXTVAL, 'Ionescu', 'Maria', '0799234567', 'Strada Alba, Nr. 12',
'maria.ionescu@example.com');
INSERT INTO CLIENT (id_client, nume, prenume, telefon, adresa, email) VALUES
(seq_client.NEXTVAL, 'Georgescu', 'Ana', '0799345678', 'Strada Rosu, Nr. 15',
'ana.georgescu@example.com');
INSERT INTO CLIENT (id_client, nume, prenume, telefon, adresa, email) VALUES
(seq_client.NEXTVAL, 'Vasilescu', 'Andrei', '0799456789', 'Strada Galbenă, Nr. 18',
'andrei.vasilescu@example.com');
INSERT INTO CLIENT (id_client, nume, prenume, telefon, adresa, email) VALUES
(seq_client.NEXTVAL, 'Dumitrescu', 'Elena', '0799567890', 'Strada Alba, Nr. 4',
'elena.dumitrescu@example.com');
INSERT INTO CLIENT (id_client, nume, prenume, telefon, adresa, email) VALUES
(seq_client.NEXTVAL, 'Marinescu', 'Cristina', '0799678901', 'Strada Albastra, Nr.
25', 'cristina.marinescu@example.com');
INSERT INTO CLIENT (id_client, nume, prenume, telefon, adresa, email) VALUES
(seq_client.NEXTVAL, 'Stoicescu', 'Dan', '0799789012', 'Strada Mov, Nr. 30',
'dan.stoicescu@example.com');
INSERT INTO CLIENT (id_client, nume, prenume, telefon, adresa, email) VALUES
(seq_client.NEXTVAL, 'Nica', 'Gabriela', '0799890123', 'Strada Roz, Nr. 35',
'gabriela.nica@example.com');
```

----- SEQUENCE SI INSERARI TUR -----

```
CREATE SEQUENCE seq_tur START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
```

```
INSERT INTO TUR (id_tur, id_centru) VALUES (seq_tur.NEXTVAL, 100);
INSERT INTO TUR (id_tur, id_centru, detalii) VALUES (seq_tur.NEXTVAL, 101,
'Detalii');
INSERT INTO TUR (id_tur, id_centru) VALUES (seq_tur.NEXTVAL, 102);
INSERT INTO TUR (id_tur, id_centru, detalii) VALUES (seq_tur.NEXTVAL, 103, 'Joaca');
INSERT INTO TUR (id_tur, id_centru) VALUES (seq_tur.NEXTVAL, 104);
INSERT INTO TUR (id_tur, id_centru) VALUES (seq_tur.NEXTVAL, 105);
INSERT INTO TUR (id_tur, id_centru, detalii) VALUES (seq_tur.NEXTVAL, 102,
'Vizualizare');
INSERT INTO TUR (id_tur, id_centru) VALUES (seq_tur.NEXTVAL, 100);
```

----- SEQUENCE SI INSERARI PROGRAMARE_VIZITA -----

```
CREATE SEQUENCE seq_programare_vizita START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
```

```
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 1, 1, 2, DATE '2023-01-05', '1');
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 2, 1, 3, DATE '2023-02-20', '2');
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 3, 4, 1, DATE '2023-03-15', '1');
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 4, 5, 4, DATE '2023-04-25', '2');
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 5, 6, 2, DATE '2023-05-10', '1');
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 6, 7, 1, DATE '2023-06-20', '2');
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 7, 2, 3, DATE '2023-07-25', '1');
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 8, 1, 2, DATE '2022-02-15', '1');
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 4, 1, 3, DATE '2023-08-05', '2');
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 3, 8, 2, DATE '2023-02-21', '2');
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 2, 3, 5, DATE '2023-03-11', '1');
INSERT INTO PROGRAMARE_VIZITA (id_prog, id_client, id_tur, nr_pers, data_prog, interval_orar) VALUES (seq_programare_vizita.NEXTVAL, 6, 2, 1, DATE '2023-11-05', '2');
```

----- SEQUENCE SI INSERARI DONATIE -----

```
CREATE SEQUENCE seq_donatie START WITH 400 INCREMENT BY 1 NOCACHE NOCYCLE;
```

```
INSERT INTO DONATIE (id_donatie, id_centru, id_client, data_donatie, suma) VALUES
(seq_donatie.NEXTVAL, 100, 1, DATE '2023-01-10', 500);
INSERT INTO DONATIE (id_donatie, id_centru, id_client, data_donatie, suma) VALUES
(seq_donatie.NEXTVAL, 101, 2, DATE '2023-02-25', 300);
INSERT INTO DONATIE (id_donatie, id_centru, id_client, data_donatie, suma) VALUES
(seq_donatie.NEXTVAL, 102, 3, DATE '2023-03-20', 200);
INSERT INTO DONATIE (id_donatie, id_centru, id_client, data_donatie, suma) VALUES
(seq_donatie.NEXTVAL, 100, 4, DATE '2023-04-30', 400);
INSERT INTO DONATIE (id_donatie, id_centru, id_client, data_donatie, suma) VALUES
(seq_donatie.NEXTVAL, 104, 5, DATE '2023-05-15', 150);
INSERT INTO DONATIE (id_donatie, id_centru, id_client, data_donatie, suma) VALUES
(seq_donatie.NEXTVAL, 100, 6, DATE '2023-06-25', 350);
INSERT INTO DONATIE (id_donatie, id_centru, id_client, data_donatie, suma) VALUES
(seq_donatie.NEXTVAL, 104, 5, DATE '2022-03-08', 100);
```

----- SEQUENCE SI INSERARI CONTRACT_ADOPTIE -----

```
CREATE SEQUENCE seq_contract_adoptie START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
```

```
INSERT INTO CONTRACT_ADOPTIE (id_contract, id_pet, id_client, data_adoptie) VALUES
(seq_contract_adoptie.NEXTVAL, 1400, 1, DATE '2023-01-15');
INSERT INTO CONTRACT_ADOPTIE (id_contract, id_pet, id_client, data_adoptie) VALUES
(seq_contract_adoptie.NEXTVAL, 1402, 2, DATE '2023-02-28');
INSERT INTO CONTRACT_ADOPTIE (id_contract, id_pet, id_client, data_adoptie) VALUES
(seq_contract_adoptie.NEXTVAL, 1403, 3, DATE '2023-03-25');
INSERT INTO CONTRACT_ADOPTIE (id_contract, id_pet, id_client, data_adoptie) VALUES
(seq_contract_adoptie.NEXTVAL, 1404, 4, DATE '2023-04-30');
INSERT INTO CONTRACT_ADOPTIE (id_contract, id_pet, id_client, data_adoptie) VALUES
(seq_contract_adoptie.NEXTVAL, 1405, 5, DATE '2023-05-20');
```

----- SEQUENCE SI INSERARI ANGAJAT -----

```
CREATE SEQUENCE seq_angajat START WITH 250 INCREMENT BY 1 NOCACHE NOCYCLE;
```



```
INSERT INTO ANGAJAT (id_angajat, id_centru, nume, prenume, telefon, data_angajare,
ore_saptamana, salariu) VALUES (seq_angajat.NEXTVAL, 100, 'Popa', 'Cristian',
'0711111111', DATE '2022-01-01', 40, 3000);
INSERT INTO ANGAJAT (id_angajat, id_centru, nume, prenume, telefon, data_angajare,
ore_saptamana, salariu) VALUES (seq_angajat.NEXTVAL, 101, 'Ionescu', 'Simona',
'0722222222', DATE '2022-02-01', 35, 2800);
INSERT INTO ANGAJAT (id_angajat, id_centru, nume, prenume, telefon, data_angajare,
ore_saptamana, salariu) VALUES (seq_angajat.NEXTVAL, 102, 'Vasilescu', 'Alexandru',
'0733333333', DATE '2022-03-01', 40, 3100);
INSERT INTO ANGAJAT (id_angajat, id_centru, nume, prenume, telefon, data_angajare,
ore_saptamana, salariu) VALUES (seq_angajat.NEXTVAL, 103, 'Marin', 'Adriana',
'0744444444', DATE '2022-04-01', 30, 2700);
INSERT INTO ANGAJAT (id_angajat, id_centru, nume, prenume, telefon, data_angajare,
ore_saptamana, salariu) VALUES (seq_angajat.NEXTVAL, 104, 'Dumitru', 'Stefan',
'0755555555', DATE '2022-05-01', 40, 3200);
INSERT INTO ANGAJAT (id_angajat, id_centru, nume, prenume, telefon, data_angajare,
ore_saptamana, salariu) VALUES (seq_angajat.NEXTVAL, 105, 'Nistor', 'Larisa',
'0766666666', DATE '2022-06-01', 35, 2900);
INSERT INTO ANGAJAT (id_angajat, id_centru, nume, prenume, telefon, data_angajare,
ore_saptamana, salariu) VALUES (seq_angajat.NEXTVAL, 100, 'Toma', 'Bogdan',
'0777777777', DATE '2022-07-01', 40, 3300);
INSERT INTO ANGAJAT (id_angajat, id_centru, nume, prenume, telefon, data_angajare,
ore_saptamana, salariu) VALUES (seq_angajat.NEXTVAL, 102, 'Radu', 'Elena',
'0788888888', DATE '2022-08-01', 30, 2600);
INSERT INTO ANGAJAT (id_angajat, id_centru, nume, prenume, telefon, data_angajare,
ore_saptamana, salariu) VALUES (seq_angajat.NEXTVAL, 104, 'Petrescu', 'Andreea',
'0799999999', DATE '2022-09-01', 38, 2950);
INSERT INTO ANGAJAT (id_angajat, id_centru, nume, prenume, telefon, data_angajare,
ore_saptamana, salariu) VALUES (seq_angajat.NEXTVAL, 103, 'Luca', 'George',
'0700000000', DATE '2022-10-01', 37, 2850);
```

~ Cerința 12 ~

```
-- Selectează numele și prenumele clienților care au făcut donații  
-- mai mari de 300 și care au adoptat animale din centrul cu adresa 'Strada  
Florilor, Nr. 5'  
-- (subcereri sincronizate cu cel puțin 3 tabele)
```

```
SELECT c.id_client, c.nume, c.prenume  
FROM CLIENT c  
JOIN DONATIE d ON c.id_client = d.id_client  
WHERE d.suma >= 300 AND EXISTS (  
    SELECT 1  
    FROM CONTRACT_ADOPTIE ca  
    JOIN ANIMAL_COMPANIE ac on ca.id_pet = ac.id_pet  
    JOIN CENTRU ce on ac.id_centru = ce.id_centru  
    WHERE ce.adresa = 'Strada Florilor, Nr. 5'  
    AND ca.id_client = c.id_client  
);
```

```
216  
217 -- Selectează numele și prenumele clienților care au făcut donații  
218 -- mai mari de 300 și care au adoptat animale din centrul cu adresa 'Strada Florilor, Nr. 5'  
219 -- (subcereri sincronizate cu cel puțin 3 tabele)  
220  
221 ✓ SELECT c.id_client, c.nume, c.prenume  
222 FROM CLIENT c  
223 JOIN DONATIE d ON c.id_client = d.id_client  
224 WHERE d.suma >= 300 AND EXISTS (  
225     SELECT 1  
226     FROM CONTRACT_ADOPTIE ca  
227     JOIN ANIMAL_COMPANIE ac on ca.id_pet = ac.id_pet  
228     JOIN CENTRU ce on ac.id_centru = ce.id_centru  
229     WHERE ce.adresa = 'Strada Florilor, Nr. 5'  
230     AND ca.id_client = c.id_client  
231 );  
232
```

ID_CLIENT	NUME	PRENUME
1	Popescu	Ion
2	Ionescu	Maria

Download CSV

2 rows selected.

Rezultate ok. Doar Popescu Ion și Ionescu Maria îndeplinesc criteriile

```
-- Selecteaza numele, prenumele, numarul de telefon si numarul
-- de ore/saptamana impreuna cu numarul total de donatii pentru fiecare
-- angajat care lucreaza in centrele care au cel putin 2 donatii
-- (subcereri nesincronizate in clauza FROM)

SELECT a.nume, a.prenume, a.telefon, a.ore_saptamana, COUNT(d.id_donatie) AS
numar_donatii
FROM (
    SELECT c.id_centru
    FROM CENTRU c
    JOIN DONATIE d ON d.id_centru = c.id_centru
    GROUP BY c.id_centru
    HAVING COUNT(d.id_donatie) >= 2
) ce
JOIN ANGAJAT a ON a.id_centru = ce.id_centru
LEFT JOIN DONATIE d ON a.id_centru = d.id_centru
GROUP BY a.nume, a.prenume, a.telefon, a.ore_saptamana;
```

SQL Worksheet

```
233
234 -- Selecteaza numele, prenumele, numarul de telefon si numarul
235 -- de ore/saptamana impreuna cu numarul total de donatii pentru fiecare
236 -- angajat care lucreaza in centrele care au cel putin 2 donatii
237 -- (subcereri nesincronizate in clauza FROM)
238
239 v SELECT a.num, a.prenume, a.telefon, a.ore_saptamana, COUNT(d.id_donatie) AS numar_donatii
240 FROM (
241     SELECT c.id_centru
242     FROM CENTRU c
243     JOIN DONATIE d ON d.id_centru = c.id_centru
244     GROUP BY c.id_centru
245     HAVING COUNT(d.id_donatie) >= 2
246 ) ce
247 JOIN ANGAJAT a ON a.id_centru = ce.id_centru
248 LEFT JOIN DONATIE d ON a.id_centru = d.id_centru
249 GROUP BY a.num, a.prenume, a.telefon, a.ore_saptamana;
250
```

NUME	PRENUME	TELEFON	ORE_SAPTAMANA	NUMAR_DONATII
Toma	Bogdan	0777777777	40	3
Petrescu	Andreea	0799999999	38	2
Popa	Cristian	0711111111	40	3
Dumitru	Stefan	0755555555	40	2

Download CSV

4 rows selected.

Rezultate ok.

```
-- Selectează numele, prenumele, adresa și numărul de telefon al clienților
-- împreună cu suma persoanelor din toate programările făcute de aceștia,
-- grupati in functie de datele clientilor si verificati
-- pentru fiecare client care daca făcut cel puțin o programare în intervalul
-- orar '2' și cel puțin o programare pentru centrul cu id_centru = 100.
-- (Grupari de date cu subcereri nesincronizate în clauza
-- HAVING cu cel puțin 3 tabele)

SELECT c.nume, c.prenume, c.adresa, c.telefon, SUM(pv.nr_pers) AS suma_persoanelor
FROM CLIENT c
JOIN PROGRAMARE_VIZITA pv ON c.id_client = pv.id_client
JOIN TUR t ON pv.id_tur = t.id_tur
JOIN CENTRU ce ON t.id_centru = ce.id_centru
WHERE c.id_client IN (
    SELECT DISTINCT pv1.id_client
    FROM PROGRAMARE_VIZITA pv1
    JOIN TUR t1 ON pv1.id_tur = t1.id_tur
    WHERE pv1.interval_orar = '2'
)
AND c.id_client IN (
    SELECT DISTINCT pv2.id_client
    FROM PROGRAMARE_VIZITA pv2
    JOIN TUR t2 ON pv2.id_tur = t2.id_tur
    WHERE t2.id_centru = 100
)
GROUP BY c.nume, c.prenume, c.adresa, c.telefon
HAVING SUM(pv.nr_pers) >= (
    SELECT AVG(pv3.nr_pers)
    FROM PROGRAMARE_VIZITA pv3);
```

```
252 -- Selectează numele, prenumele, adresa și numărul de telefon al clienților
253 -- împreună cu suma persoanelor din toate programările făcute de aceștia,
254 -- grupati în funcție de datele clienților și verificați
255 -- pentru fiecare client care dacă a făcut cel puțin o programare în intervalul
256 -- orar '2' și cel puțin o programare pentru centrul cu id_centru = 100.
257 -- (Grupări de date cu subcereri nesincronizate în clauza
258 -- HAVING cu cel puțin 3 tabele)
259 ✓ SELECT c.nume, c.prenume, c.adresa, c.telefon, SUM(pv.nr_pers) AS suma_persoanelor
260 FROM CLIENT c
261 JOIN PROGRAMARE_VIZITA pv ON c.id_client = pv.id_client
262 JOIN TUR t ON pv.id_tur = t.id_tur
263 JOIN CENTRU ce ON t.id_centru = ce.id_centru
264 WHERE c.id_client IN (
265     SELECT DISTINCT pv1.id_client
266     FROM PROGRAMARE_VIZITA pv1
267     JOIN TUR t1 ON pv1.id_tur = t1.id_tur
268     WHERE pv1.interval_orar = '2'
269 )
270 AND c.id_client IN (
271     SELECT DISTINCT pv2.id_client
272     FROM PROGRAMARE_VIZITA pv2
273     JOIN TUR t2 ON pv2.id_tur = t2.id_tur
274     WHERE t2.id_centru = 100
275 )
276 GROUP BY c.nume, c.prenume, c.adresa, c.telefon
277 HAVING SUM(pv.nr_pers) >= (
278     SELECT AVG(pv3.nr_pers)
279     FROM PROGRAMARE_VIZITA pv3);
```

NUME	PRENUME	ADRESA	TELEFON	SUMA_PERSOANELOR
Ionescu	Maria	Strada Alba, Nr. 12	0799234567	8
Vasilescu	Andrei	Strada Galbena, Nr. 18	0799456789	7
Georgescu	Ana	Strada Rosu, Nr. 15	0799345678	3

```
-- Selectează numele clienților, suma totală donată și un status al donației  
-- ('High' dacă suma totală este mai mare de 500, altfel 'Low'), ordonat  
descrescător  
-- după suma totală donată.  
-- ( ordonare, folosirea NVL si DECODE )
```

```
SELECT c.num, c.prenume, NVL(SUM(d.suma), 0) AS suma_totala,  
DECODE(SIGN(SUM(d.suma) - 500), 1, 'High', 'Low') AS status_donatie  
FROM CLIENT c  
LEFT JOIN DONATIE d ON c.id_client = d.id_client  
GROUP BY c.num, c.prenume  
ORDER BY suma_totala DESC;
```

SQL Worksheet

```
278  
279  
280 -- Selectează numele clienților, suma totală donată și un status al donației  
281 -- ('High' dacă suma totală este mai mare de 350, altfel 'Low'), ordonat descrescător  
282 -- după suma totală donată.  
283 -- ( ordonare, folosirea NVL si DECODE )  
284  
285 v SELECT c.num, c.prenume, NVL(SUM(d.suma), 0) AS suma_totala,  
286 DECODE(SIGN(SUM(d.suma) - 350), 1, 'High', 'Low') AS status_donatie  
287 FROM CLIENT c  
288 LEFT JOIN DONATIE d ON c.id_client = d.id_client  
289 GROUP BY c.num, c.prenume  
290 ORDER BY suma_totala DESC;  
291 |
```

NUME	PRENUME	SUMA_TOTALA	STATUS_DONATIE
Popescu	Ion	500	High
Vasilescu	Andrei	400	High
Marinescu	Cristina	350	Low
Ionescu	Maria	300	Low
Dumitrescu	Elena	250	Low
Georgescu	Ana	200	Low
Stoicescu	Dan	0	Low
Nica	Gabriela	0	Low




```
-- Selectează numele centrelor, data ultimei donații, numărul total de animale și
-- primele 10 caractere ale adresei centrului. Utilizează clauza WITH
-- pentru a obține datele despre donații și animalele din centru.
-- (functii pe siruri de caractere, date calendaristice, CASE si WITH)
WITH CentruInfo AS (
    SELECT c.id_centru, COUNT(ac.id_pet) AS nr_animale, MAX(d.data_donatie) AS
ultima_donatie
    FROM CENTRU c
    LEFT JOIN ANIMAL_COMPANIE ac ON c.id_centru = ac.id_centru
    LEFT JOIN DONATIE d ON c.id_centru = d.id_centru
    GROUP BY c.id_centru
)
SELECT c.adresa, SUBSTR(c.adresa, 1, 10) AS adresa_truncata, ci.nr_animale,
TO_CHAR(ci.ultima_donatie, 'DD-MON-YYYY') AS data_ultima_donatie,
CASE
    WHEN ci.nr_animale > 5 THEN 'Multe'
    WHEN ci.nr_animale > 2 THEN 'Mediu'
    ELSE 'Putine'
END AS status_nr_animale
FROM CENTRU c
JOIN CentruInfo ci ON c.id_centru = ci.id_centru;
```

SQL Worksheet

```
292 -- Selectează numele centrelor, data ultimei donații, statusul numărului total de animale și
293 -- primele 10 caractere ale adresei centrului. Utilizează clauza WITH
294 -- pentru a obține datele despre donații și animalele din centru.
295 -- (functii pe siruri de caractere, date calendaristice, CASE si WITH)
296
297 ✓ WITH CentruInfo AS (
298     SELECT c.id_centru, COUNT(ac.id_pet) AS nr_animale, MAX(d.data_donatie) AS ultima_donatie
299     FROM CENTRU c
300     LEFT JOIN ANIMAL_COMPANIE ac ON c.id_centru = ac.id_centru
301     LEFT JOIN DONATIE d ON c.id_centru = d.id_centru
302     GROUP BY c.id_centru
303 )
304 SELECT c.adresa, SUBSTR(c.adresa, 1, 10) AS adresa_truncata, ci.nr_animale,
305 TO_CHAR(ci.ultima_donatie, 'DD-MON-YYYY') AS data_ultima_donatie,
306 CASE
307     WHEN ci.nr_animale > 5 THEN 'Multe'
308     WHEN ci.nr_animale > 2 THEN 'Mediu'
309     ELSE 'Putine'
310 END AS status_nr_animale
311 FROM CENTRU c
312 JOIN CentruInfo ci ON c.id_centru = ci.id_centru;
```

ADRESA	ADRESA_TRUNCATA	NR_ANIMALE	DATA_ULTIMA_DONATIE	STATUS_NR_ANIMALE
Strada Marilor, Nr. 8	Strada Mar	2	25-FEB-2023	Putine
Strada Florilor, Nr. 5	Strada Flo	12	25-JUN-2023	Multe
Strada Zambilei, Nr. 20	Strada Zam	1	-	Putine
Strada Lalelelor, Nr. 12	Strada Lal	2	-	Putine
Strada Trandafirilor, Nr. 10	Strada Tra	1	20-MAR-2023	Putine
Strada Violetei, Nr. 15	Strada Vio	4	15-MAY-2023	Mediu

~ Cerința 13 ~

```
--      1.Setam pentru fiecare centru numarul de animale de companie  
-- care au fost inserate in tabelul ANIMAL_COMPANIE
```

```
UPDATE CENTRU c  
SET c.nr_animale_companie = (  
    SELECT COUNT(*)  
    FROM ANIMAL_COMPANIE ac  
    WHERE ac.id_centru = c.id_centru  
);
```

ID_CENTRU	ADRESA	NR_TEL	PROGRAM	NR_ANIMALE_COMPANIE
100	Strada Florilor, Nr. 5	0712345678	Luni-Vineri 9-17	0
101	Strada Marilor, Nr. 8	0722345678	Luni-Vineri 9-18	0
102	Strada Trandafirilor, Nr. 10	0732345678	Luni-Vineri 8-16	0
103	Strada Lalelelor, Nr. 12	0742345678	Luni-Vineri 10-18	0
104	Strada Violetei, Nr. 15	0752345678	Luni-Vineri 9-17	0
105	Strada Zambilei, Nr. 20	0762345678	Luni-Vineri 8-15	0

Download CSV

6 rows selected.

6 row(s) updated.

ID_CENTRU	ADRESA	NR_TEL	PROGRAM	NR_ANIMALE_COMPANIE
100	Strada Florilor, Nr. 5	0712345678	Luni-Vineri 9-17	4
101	Strada Marilor, Nr. 8	0722345678	Luni-Vineri 9-18	2
102	Strada Trandafirilor, Nr. 10	0732345678	Luni-Vineri 8-16	1
103	Strada Lalelelor, Nr. 12	0742345678	Luni-Vineri 10-18	2
104	Strada Violetei, Nr. 15	0752345678	Luni-Vineri 9-17	2
105	Strada Zambilei, Nr. 20	0762345678	Luni-Vineri 8-15	1

```
--      2. Adaug o noua coloana in tabelul CLIENT care va tine evidenta  
-- numarului total de programari pe care le-a facut fiecare client
```

```
ALTER TABLE CLIENT ADD total_programari INT;
```

```
UPDATE CLIENT c  
SET c.total_programari = (  
    SELECT COUNT(*)  
    FROM PROGRAMARE_VIZITA pv  
    WHERE pv.id_client = c.id_client  
);
```

ID_CLIENT	NUME	PRENUME	TELEFON	ADRESA	EMAIL
1	Popescu	Ion	0799123456	Strada Verde, Nr. 10	ion.popescu@example.com
2	Ionescu	Maria	0799234567	Strada Alba, Nr. 12	maria.ionescu@example.com
3	Georgescu	Ana	0799345678	Strada Rosu, Nr. 15	ana.georgescu@example.com
4	Vasilescu	Andrei	0799456789	Strada Galbena, Nr. 18	andrei.vasilescu@example.com
5	Dumitrescu	Elena	0799567890	Strada Alba, Nr. 4	elena.dumitrescu@example.com
6	Marinescu	Cristina	0799678901	Strada Albastra, Nr. 25	cristina.marinescu@example.com
7	Stoicescu	Dan	0799789012	Strada Mov, Nr. 30	dan.stoicescu@example.com
8	Nica	Gabriela	0799890123	Strada Roz, Nr. 35	gabriela.nica@example.com

Download CSV

8 rows selected.

Table altered.

8 row(s) updated.

ID_CLIENT	NUME	PRENUME	TELEFON	ADRESA	EMAIL	TOTAL_PROGRAMARI
1	Popescu	Ion	0799123456	Strada Verde, Nr. 10	ion.popescu@example.com	1
2	Ionescu	Maria	0799234567	Strada Alba, Nr. 12	maria.ionescu@example.com	2

ID_CLIENT	NUME	PRENUME	TELEFON	ADRESA	EMAIL	TOTAL_PROGRAMARI
1	Popescu	Ion	0799123456	Strada Verde, Nr. 10	ion.popescu@example.com	1
2	Ionescu	Maria	0799234567	Strada Alba, Nr. 12	maria.ionescu@example.com	2
3	Georgescu	Ana	0799345678	Strada Rosu, Nr. 15	ana.georgescu@example.com	2
4	Vasilescu	Andrei	0799456789	Strada Galbena, Nr. 18	andrei.vasilescu@example.com	2
5	Dumitrescu	Elena	0799567890	Strada Alba, Nr. 4	elena.dumitrescu@example.com	1
6	Marinescu	Cristina	0799678901	Strada Albastra, Nr. 25	cristina.marinescu@example.com	2
7	Stoicescu	Dan	0799789012	Strada Mov, Nr. 30	dan.stoicescu@example.com	1
8	Nica	Gabriela	0799890123	Strada Roz, Nr. 35	gabriela.nica@example.com	1

Download CSV

8 rows selected.

```
--      3. Sterg toate programarile care au fost facute inainte de
-- o anumita data. Exemplu: 2024-01-01

DELETE FROM PROGRAMARE_VIZITA
WHERE id_tur IN (
    SELECT id_tur
    FROM TUR
    WHERE data_tur < TO_DATE('2024-01-01', 'YYYY-MM-DD')
);
```

ID_PROG	ID_CLIENT	ID_TUR	NR_PERS	DATA_PROG	INTERVAL_ORAR
1	1	1	2	05-JAN-23	1
2	2	1	3	20-FEB-23	2
3	3	4	1	15-MAR-23	1
4	4	5	4	25-APR-23	2
5	5	6	2	10-MAY-23	1
6	6	7	1	20-JUN-23	2
7	7	2	3	25-JUL-23	1
8	8	1	2	15-FEB-22	1
9	4	1	3	05-AUG-23	2
10	3	8	2	21-FEB-23	2
11	2	3	5	11-MAR-23	1
12	6	2	1	05-NOV-23	2

Download CSV

12 rows selected.

5 row(s) deleted.

12 rows selected.

5 row(s) deleted.

ID_PROG	ID_CLIENT	ID_TUR	NR_PERS	DATA_PROG	INTERVAL_ORAR
3	3	4	1	15-MAR-23	1
4	4	5	4	25-APR-23	2
5	5	6	2	10-MAY-23	1
6	6	7	1	20-JUN-23	2
7	7	2	3	25-JUL-23	1
11	2	3	5	11-MAR-23	1
12	6	2	1	05-NOV-23	2

[Download CSV](#)

7 rows selected.

~ Printscreen-uri tabele și cod ~

```
290
291 select * from FISA_AC;
292
293
```

ID_FISA	SPECIE	RASA	SEX
1	Caine	Labrador	M
2	Pisica	Siameza	F
3	Caine	Beagle	M
4	Pisica	Persana	F
5	Caine	Bulldog	M
6	Pisica	Maine Coon	F
7	Caine	Golden Retriever	M
8	Pisica	Bengaleza	F
9	Caine	Micut	F
10	Pisica	Siameza	M
11	Caine	Labrador	F
12	Pisica	Persana	M

[Download CSV](#)

12 rows selected.


```
298 select * from CARNET_MEDICAL;
```

```
299
```

ID_FISA	VARSTA	VACCIN	STERILIZAT	DEPARAZITAT	MENTIUNI
1	2	Rabie	DA	NU	None
2	3	Parvo	NU	DA	Checkup needed
3	1	Distemper	DA	DA	Healthy
4	4	Leptospirosis	NU	NU	Needs vaccination
5	2	Bordetella	DA	DA	None
6	3	Rabie	NU	NU	Slightly underweight
7	1	Parvo	DA	DA	Recovered from illness
8	4	Distemper	DA	NU	None
9	1	Distemper	DA	NU	Healthy
10	2	Leptospirosis	NU	NU	Needs vaccination
11	2	Bordetella	NU	DA	None
12	5	Rabie	NU	NU	None

Download CSV

12 rows selected.

300

301 `select * from ANIMAL_COMPANIE;`

302

303

ID_PET	ID_CENTRU	ID_FISA	DATA_PRELUARE
1400	100	1	01-JAN-23
1401	101	2	15-FEB-23
1402	100	3	10-MAR-23
1403	103	4	20-APR-23
1404	104	5	05-MAY-23
1405	100	6	18-JUN-23
1406	105	7	22-JUL-23
1407	100	8	30-AUG-23
1408	102	9	05-MAY-23
1409	103	10	18-JUN-23
1410	104	11	22-JUL-23
1411	101	12	30-AUG-23

Download CSV

12 rows selected.

```
303
304 select * from CONTRACT_ADOPTIE;
305
306
```

ID_CONTRACT	ID_PET	ID_CLIENT	DATA_ADOPTIE
1	1400	1	15-JAN-23
2	1402	2	28-FEB-23
3	1403	3	25-MAR-23
4	1404	4	30-APR-23
5	1405	5	20-MAY-23

[Download CSV](#)

5 rows selected.

```
307
308 select * from CLIENT;
309
310
```

ID_CLIENT	NUME	PRENUME	TELEFON	ADRESA	EMAIL
1	Popescu	Ion	0799123456	Strada Verde, Nr. 10	ion.popescu@example.com
2	Ionescu	Maria	0799234567	Strada Alba, Nr. 12	maria.ionescu@example.com
3	Georgescu	Ana	0799345678	Strada Rosu, Nr. 15	ana.georgescu@example.com
4	Vasilescu	Andrei	0799456789	Strada Galbena, Nr. 18	andrei.vasilescu@example.com
5	Dumitrescu	Elena	0799567890	Strada Alba, Nr. 4	elena.dumitrescu@example.com
6	Marinescu	Cristina	0799678901	Strada Albastra, Nr. 25	cristina.marinescu@example.com
7	Stoicescu	Dan	0799789012	Strada Mov, Nr. 30	dan.stoicescu@example.com
8	Nica	Gabriela	0799890123	Strada Roz, Nr. 35	gabriela.nica@example.com

[Download CSV](#)

8 rows selected.

```
310
311 select * from DONATIE;
312
313
```

ID_DONATIE	ID_CENTRU	ID_CLIENT	DATA_DONATIE	SUMA
400	100	1	10-JAN-23	500
401	101	2	25-FEB-23	300
402	102	3	20-MAR-23	200
403	100	4	30-APR-23	400
404	104	5	15-MAY-23	150
405	100	6	25-JUN-23	350
406	104	5	08-MAR-22	100

Download CSV

7 rows selected.

```
313
314 select * from PROGRAMARE_VIZITA;
315
316
```

ID_PROG	ID_CLIENT	ID_TUR	NR_PERS	DATA_PROG	INTERVAL_ORAR
1	1	1	2	05-JAN-23	1
2	2	1	3	20-FEB-23	2
3	3	4	1	15-MAR-23	1
4	4	5	4	25-APR-23	2
5	5	6	2	10-MAY-23	1
6	6	7	1	20-JUN-23	2
7	7	2	3	25-JUL-23	1
8	8	1	2	15-FEB-22	1
9	4	1	3	05-AUG-23	2
10	3	8	2	21-FEB-23	2
11	2	3	5	11-MAR-23	1
12	6	2	1	05-NOV-23	2

Download CSV

12 rows selected.

```
310
317 select * from TUR;
318
319
```

ID_TUR	ID_CENTRU	DETALII
1	100	-
2	101	Detalii
3	102	-
4	103	Joaca
5	104	-
6	105	-
7	102	Vizualizare
8	100	-

Download CSV

8 rows selected.

319
320 `select * from CENTRU;`
321
322

ID_CENTRU	ADRESA	NR_TEL	PROGRAM	NR_ANIMALE_COMPANIE
100	Strada Florilor, Nr. 5	0712345678	Luni-Vineri 9-17	4
101	Strada Marilor, Nr. 8	0722345678	Luni-Vineri 9-18	2
102	Strada Trandafirilor, Nr. 10	0732345678	Luni-Vineri 8-16	1
103	Strada Lalelelor, Nr. 12	0742345678	Luni-Vineri 10-18	2
104	Strada Violetei, Nr. 15	0752345678	Luni-Vineri 9-17	2
105	Strada Zambilei, Nr. 20	0762345678	Luni-Vineri 8-15	1

Download CSV

6 rows selected.

322

323 `select * from ANGAJAT;`

324

325

ID_ANGAJAT	ID_CENTRU	NUME	PRENUME	TELEFON	DATA_ANGAJARE	ORE_SAPTAMANA	SALARIU
250	100	Popa	Cristian	0711111111	01-JAN-22	40	3000
251	101	Ionescu	Simona	0722222222	01-FEB-22	35	2800
252	102	Vasilescu	Alexandru	0733333333	01-MAR-22	40	3100
253	103	Marin	Adriana	0744444444	01-APR-22	30	2700
254	104	Dumitru	Stefan	0755555555	01-MAY-22	40	3200
255	105	Nistor	Larisa	0766666666	01-JUN-22	35	2900
256	100	Toma	Bogdan	0777777777	01-JUL-22	40	3300
257	102	Radu	Elena	0788888888	01-AUG-22	30	2600
258	104	Petrescu	Andreea	0799999999	01-SEP-22	38	2950
259	103	Luca	George	0700000000	01-OCT-22	37	2850

Download CSV

10 rows selected.

SQL Worksheet

```
1 CREATE SEQUENCE seq_fisa_ac START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;  
2 CREATE SEQUENCE seq_centru START WITH 100 INCREMENT BY 1 NOCACHE NOCYCLE;  
> 3 CREATE SEQUENCE seq_animal_companie START WITH 1400 INCREMENT BY 1 NOCACHE NOCYCLE;  
4 CREATE SEQUENCE seq_client START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;  
5 CREATE SEQUENCE seq_programare_vizita START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;  
6 CREATE SEQUENCE seq_tur START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;  
7 CREATE SEQUENCE seq_donatie START WITH 400 INCREMENT BY 1 NOCACHE NOCYCLE;  
> 8 CREATE SEQUENCE seq_contract_adoptie START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;  
9 CREATE SEQUENCE seq_angajat START WITH 250 INCREMENT BY 1 NOCACHE NOCYCLE;  
10  
11
```

Sequence created.

Sequence created.

Sequence created.

Sequence created.

Sequence created.

Sequence created.

Sequence created.

Sequence created.

Sequence created.

SQL Worksheet

```
12 ✓ CREATE TABLE FISA_AC(  
13     id_fisa INT PRIMARY KEY,  
14     specie VARCHAR2(50),  
15     rasa VARCHAR2(100),  
16     sex CHAR(1)  
17 );  
18  
19 ✓ CREATE TABLE CARNET_MEDICAL(  
20     id_fisa INT PRIMARY KEY,  
21     varsta INT,  
22     vaccin VARCHAR2(500),  
23     sterilizat CHAR(2) CHECK (sterilizat IN ('DA', 'NU')),  
24     deparazitat CHAR(2) CHECK (deparazitat IN ('DA', 'NU')),  
25     mentiuni VARCHAR2(500),  
26     FOREIGN KEY(id_fisa) REFERENCES FISA_AC(id_fisa) ON DELETE CASCADE  
27 );
```

Table created.

Table created.

Table created.

Table created.

Table created.

Table created.

Table created.

Table created.

Table created.

Table created.

SQL Worksheet

```
106
107 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Caine', 'Labrador', 'M');
108 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Pisica', 'Siameza', 'F');
109 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Caine', 'Beagle', 'M');
110 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Pisica', 'Persana', 'F');
111 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Caine', 'Bulldog', 'M');
112 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Pisica', 'Maine Coon', 'F');
113 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Caine', 'Golden Retriever', 'M');
114 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Pisica', 'Bengaleza', 'F');
115 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Caine', 'Micut', 'F');
116 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Pisica', 'Siameza', 'M');
117 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Caine', 'Labrador', 'F');
118 INSERT INTO FISA_AC (id_fisa, specie, rasa, sex) VALUES (seq_fisa_ac.NEXTVAL, 'Pisica', 'Persana', 'M');
```

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

SQL Worksheet

Clear

```
119
120 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (1, 2, 'Rabie', 'DA', 'NU', 'None');
121 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (2, 3, 'Parvo', 'NU', 'DA', 'Checkup needed');
122 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (3, 1, 'Distemper', 'DA', 'DA', 'Healthy');
123 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (4, 4, 'Leptospirosis', 'NU', 'NU', 'Needs vaccination');
124 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (5, 2, 'Bordetella', 'DA', 'DA', 'None');
125 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (6, 3, 'Rabie', 'NU', 'NU', 'Slightly underweight');
126 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (7, 1, 'Parvo', 'DA', 'DA', 'Recovered from illness');
127 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (8, 4, 'Distemper', 'DA', 'NU', 'None');
128 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (9, 1, 'Distemper', 'DA', 'NU', 'Healthy');
129 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (10, 2, 'Leptospirosis', 'NU', 'NU', 'Needs vaccination');
130 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (11, 2, 'Bordetella', 'NU', 'DA', 'None');
131 INSERT INTO CARNET_MEDICAL (id_fisa, varsta, vaccin, sterilizat, deparazitat, mentiuni) VALUES (12, 5, 'Rabie', 'NU', 'NU', 'None');
```

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

[illegible][illegible]

[illegible][illegible]

[illegible][illegible]