

Gestiunea Adăposturilor de animale din România

Ene Marius - Andrei

Grupa 242

1) Prezentări pe scurt baza de date (utilitatea ei).

Modelul de date va gestiona informații legate de gestionarea adăposturilor de animale din România. Astfel în diverse locații poate exista un adăpost.

În cadrul adăpostului lucrează angajați, care pot avea posturi diferite cum ar fi: contabil, veterinar, îngrijitor, administrator, hînghier, aprovizionar, asistent veterinar.

De asemenea, în cadrul bazei de date se țin în evidență și vizitatorii fiecărui adăpost, reținînd informații despre aceștia. Se rețin informații despre animalele din aceste adăposturi, despre sănătatea acestora, mîncarea lor preferată. Se rețin și informații despre asociațiile care ajută aceste adăposturi prin donații.

1.1) Modelul respecta anumite reguli de funcționare și constrîngerii:

- Un adăpost poate avea unul sau mai mulți angajați.
- Un angajat lucrează în cadrul unui singur adăpost.
- Unui angajat îi corespunde un singur post.
- Mai mulți angajați pot avea același post.
- Un adăpost poate avea unul sau, mai multe animale în grijă.
- Un animal este înregistrat într-un singur adăpost.
- Un adăpost poate fi vizitat de un vizitator.
- Mai mulți vizitatori pot vizita mai multe adăposturi.
- Un animal trebuie să consume un aliment.
- Mai multe animale pot consuma mai multe feluri de alimente.
- Un animal trebuie să aibă o fișă medicală.
- O fișă medicală îi corespunde unui singur animal.
- O asociație finanțează un adăpost.
- Mai multe adăposturi pot fi finanțate de mai multe asociații.
- Un adăpost se poate afla într-o singură locație.
- O locație poate găzdui un singur adăpost.
- Un adoptant poate adopta unul sau mai multe animale.
- Un animal poate fi adoptat de către un singur adoptant.

- Datele despre un animal vor rămâne în baza de date chiar dacă animalul a fost adoptat.

1.2) Entități

Pentru modelul de date referitor la adăpostul de animale, structurile **LOCATIE, ADAPOST, VIZITATOR, ASOCIATIE, ANGAJAT, POST, ANIMAL, ALIMENT, FISA MEDICALA** reprezintă entități.

Vom prezenta entitățile modelului de date, dând o descriere completă a fiecăreia. De asemenea, pentru fiecare entitate se va preciza cheia primară.

Adăpost = entitate ce oferă informații despre un adăpost. Cheia primară a entității este *#id_adăpost*.

Asociație = entitate care se referă la o afacere independentă ce finanțează un adăpost. Cheia primară a entității este *#id_asociație*.

Locație = o locație din România la care se afla un adăpost. Cheia primară a entității este *#id_locație*.

Vizitator = entitate care se referă la o persoană fizică ce vizitează adăpostul. Cheia primară a entității este *#id_vizitator*.

Angajat = entitate care referă un angajat al unui adăpost. Cheia primară a entității este *#id_angajat*.

Posturi = entitate care se referă la postul pe care-l are un angajat într-un adăpost. Cheia primară a entității este *#id_post*.

Animal = entitate care se referă la animalele (care pot fi câini sau pisici) aflate în grija adăpostului. Cheia primară a entității este *#id_animal*.

Fisa Medicală = entitate care oferă informații despre sănătatea animalelor. Cheia primară a entității este *#id_fisa*.

Aliment = entitate ce reprezintă tipurile de mâncare pe care o consumă animalele, pot fi diferite și de la producători diferiți. Cheia primară a entității este *#id_aliment*.

Adoptanți = entitate ce reprezintă persoanele care adoptă un animal. Cheia primară a entității este *#id_adoptant*.

1.3) Relații

ADAPOST_se_afla_intr-o_LOCATIE = relație care leagă entitățile ADAPOST și LOCATIE reflectând legătura dintre acestea (unde se afla adăpostul). Ea are cardinalitatea 1:1.

ASOCIATIE_sponsorizeaza_ADAPOST = relație care leagă entitățile ASOCIATIE și ADAPOST, reflectând legătura dintre acestea (afacerea independentă ce sponsorizează adăpostul). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă M:M.

VIZITATOR_viziteaza_ADAPOST = relație care leagă entitățile VIZITATOR și ADAPOST, reflectând legătura dintre acestea (o persoană care vizitează un adăpost). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă M:M.

ANGAJAT_lucraza_ADAPOST = relație care leagă entitățile ANGAJAT și ADAPOST, reflectând legătura dintre acestea (un adăpost are mai mulți angajați). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă 1:M.

ANGAJAT_are_un_POST = relație care leagă entitățile ANGAJAT și POST, reflectând legătura dintre acestea (un angajat are un post). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă M:1.

ANIMAL_are_FISA_MEDICALA = relație care leagă entitățile ANIMAL și FISA_MEDICALA, reflectând legătura dintre acestea (un animal are o fișă medicală cu informații despre el). Ea are cardinalitatea 1:1.

ANIMAL_consuma_ALIMENT = relație care leagă entitățile ANIMAL și MANCARE, reflectând legătura dintre acestea (animalele consumă mâncare). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă M:M.

ANIMAL_se_afla_intr-un_ADAPOST = relație care leagă entitățile ANIMAL și ADAPOST, reflectând legătura dintre acestea (animalele stau într-un adăpost). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă M:1.

ADOPTANT_adopta_ANIMAL = relație care leagă entitățile ADOPTANT și ADAPOST, reflectând legătura dintre acestea (un adoptant adopta un animal). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă 1:M.

1.4) Atribute

Entitatea **ANIMAL** are ca atribute:

id_animal = variabilă de tip întreg, care nu poate fi nulă și reprezintă id-ul unui animal.

data_inmatriculare = variabilă de tip dată calendaristica, care reprezintă data în care animalul a fost adus la adăpost.

specie_animal = variabilă de tip șir de caractere, de lungime maximă 10, care reprezintă tipul unui animal, nu poate fi nulă.

greutate = variabilă de tip real, care reprezintă greutatea unui animal.

inaltime = variabilă de tip real, care reprezintă înălțimea unui animal.

varsta = variabilă de tip real, care reprezintă vârsta unui animal.

sex = variabilă de tip șir de caractere, de lungime maximă 1, care reprezintă sex-ul unui animal.

id_adapost = variabila de tip întreg, care reprezintă id-ul adăpostului în care se află animalul, nu poate fi nulă. Atributul corespunde unei valori a cheii primare din tabela ADAPOST.

id_adoptant = variabila de tip întreg, care reprezintă id-ul persoanei care adopta un animal. Dacă animalul NU a fost adoptat, se inițializează cu NULL. Atributul corespunde unei valori a cheii primare din tabela ADOPTANTI.

Entitatea **ALIMENT** are ca attribute:

id_aliment = variabila de tip întreg, care nu poate fi nulă și reprezintă id-ul mâncării.

pret = variabilă de tip întreg care reprezintă prețul unui aliment.

brand = variabilă de tip șir de caractere, de lungime maximă 15, care reprezintă brand-ul unui aliment (cine l-a produs).

nume_aliment = variabila de tip șir de caractere, de lungime maximă 30, care reprezintă numele produsului cumpărat, nu poate fi nulă.

cantitate = variabila de tip întreg, ce reprezintă cantitatea alimentului.

data_expirare = variabila de tip calendaristica, care reprezintă data în care expira produsul alimentară, nu poate fi nulă.

Entitatea **FISA_MEDIACALA** are ca attribute:

id_fisa = variabilă de tip întreg, care nu poate fi nulă, și care reprezintă id-ul fișei unui animal.

id_animal = variabilă de tip întreg, care reprezintă id-ul animalului corespunzător fișei medicale, nu poate fi nulă. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul ANIMAL.

tratament = variabilă de tip șir de caractere, de lungime maximă 30 care reprezintă tratamentul prescris unui animal bolnav (daca nu este bolnav se inițializează cu NULL).

simptome = variabilă de tip șir de caractere, de lungime maximă 30 care reprezintă simptomele unui animal (daca nu sunt simptome se inițializează cu NULL).

detalii = variabilă de tip șir de caractere, de lungime maximă 50 care reprezintă detaliile despre animal (daca a fost bolnav înainte, rănit, este agresiv, cuminte etc dacă nu se trece nimic, se inițializează cu NULL).

Entitatea **ADAPOST** are ca attribute:

id_adapost = variabilă de tip întreg, care nu poate fi nulă, și care reprezintă id-ul adăpostului.

nume_adapost = variabila de tip șir de caractere, de lungime maximă 30, care reprezintă numele adăpostului, nu poate fi nulă.

telefon_adapost = variabilă de tip șir de caractere, de lungime maximă 10 care reprezintă numărul de telefon al adăpostului.

id_locatie = variabila de tip întreg, care reprezintă id-ul locației unde se află adăpostul, nu poate fi nulă. Atributul corespunde unei valori a cheii primare din tabela LOCATIE.

Entitatea **VIZITATOR** are ca attribute:

id_vizitator = variabilă de tip întreg, care nu poate fi nulă, și care reprezintă id-ul vizitatorului.

nume_vizitator = variabilă de tip șir de caractere, de lungime maximă 20, care reprezintă numele vizitatorului, nu poate fi nulă.

prenume_vizitator = variabilă de tip șir de caractere, de lungime maximă 20, care reprezintă prenumele vizitatorului, nu poate fi nulă.

Enitatea **LOCATIE** are ca attribute:

id_locatie = variabilă de tip întreg, nu poate fi nulă și care reprezintă id-ul locației unde se afla adăpostul.

judet = variabila de tip sir de caractere, de lungime maxima 20 care reprezintă județul in care se afla adăpostul, nu poate fi nulă.

strada = variabilă de tip șir de caractere, de lungime maximă 25 care reprezintă strada unde se afla adăpostul, nu poate fi nulă.

oras = variabilă de tip șir de caractere, de lungime maximă 20 care reprezintă orașul unde se află adăpostul, nu poate fi nulă.

numar = variabilă de tip întreg, care reprezintă numărul de pe stradă al adăpostului.

Entitatea **ASOCIATIE** are ca attribute:

id_asociatie = variabila de tip întreg, care nu poate fi nulă, și care reprezintă id-ul unei asociații.

nume_asociatie = variabila de tip șir de caractere, de lungime maximă 30, care reprezintă numele asociației, nu poate fi nulă.

adresa = variabila de tip șir de caractere, de lungime maximă 150, care reprezintă adresa unde se afla asociația, nu poate fi nulă.

telefon = variabilă de tip șir de caractere, de lungime maximă 10 care reprezintă numărul de telefon al asociației.

Entitatea **ANGAJAT** are ca attribute:

id_angajat = variabilă de tip întreg, care nu poate fi nulă, și care reprezintă id-ul unui angajat.

nume = variabilă de tip șir de caractere, de lungime maximă 20, care reprezintă numele angajatului, nu poate fi nulă.

prenume = variabilă de tip șir de caractere, de lungime maximă 20, care reprezintă prenumele angajatului, nu poate fi nulă.

data_angajarii = variabilă de tip dată calendaristică, care reprezintă data angajării unui angajat, nu poate fi nulă.

salariu = variabilă de tip real, care reprezintă salariul angajatului.

comision = variabila de tip real, care reprezintă comisionul pe care-l castiga un angajat.

id_adapost = variabilă de tip întreg, care reprezintă id-ul adăpostului la care lucrează angajatul, nu poate fi nulă. Atributul corespunde unei valori a cheii primare din tabela ADAPOST.

id_post = variabilă de tip întreg, care reprezintă id-ul postului pe care-l are un angajat, nu poate fi nulă. Atributul corespunde unei valori a cheii primare din tabela POST.

Entitatea **POSTURI** are ca attribute:

id_post = variabila de tip șir de caractere, de lungime maximă 15 care nu poate fi nula și care reprezintă id-ul postului.

denumire_post = variabila de tip șir de caractere, de lungime maximă 20, care reprezintă numele postului pe care-l ocupa un angajat într-un adăpost, nu poate fi nulă.

salariu_minim = variabila de tip întreg, care reprezintă salariul minim pe care-l poate câștiga un angajat.

salariu_maxim = variabila de tip întreg, care reprezintă salariul maxim pe care-l poate câștiga un angajat.

Entitatea **ADOPTANTI** are ca atribute:

id_adoptant = variabila de tip întreg, care nu poate fi nula și care reprezintă id-ul adoptantului.

nume = variabilă de tip șir de caractere, de lungime maximă 20, care reprezintă numele adoptantului, nu poate fi nulă.

prenume = variabilă de tip șir de caractere, de lungime maximă 20, care reprezintă prenumele adoptantului, nu poate fi nulă.

telefon = variabilă de tip șir de caractere, de lungime maximă 10 care reprezintă numărul de telefon al adoptantului.

Relația **ASOCIATIE_sponsorizeaza_ADAPOST** are ca atribute:

id_sponsorizare = variabilă de tip întreg, care reprezintă id-ul sponsorizării. Atributul nu poate fi null.

id_asociatie = variabilă de tip întreg, care reprezintă id-ul asociației care sponsorizează adăpostul, nu poate fi nulă. Atributul corespunde unei valori a cheii primare din tabela ASOCIATIE.

id_adapost = variabilă de tip întreg, care reprezintă id-ul adăpostului pe care-l sponsorizează asociația, nu poate fi nulă. Atributul corespunde unei valori a cheii primare din tabela ADAPOST.

suma_donata = variabila de tip real, care reprezintă suma donată unui adăpost de către o asociație.

Relația **ANIMAL_consuma_ALIMENT** are ca atribute:

id_consum = variabila de tip întreg, care reprezintă id-ul consumului unui produs de către un animal. Atributul nu poate fi null.

id_aliment = variabila de tip întreg, care reprezintă id-ul mâncării, nu poate fi nulă. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul ALIMENT.

id_animal = variabilă de tip întreg, reprezintă id-ul animalului care consumă un aliment, nu poate fi nulă. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul ANIMAL.

Relația **VIZITATOR_viziteaza_ADAPOST** are ca atribute:

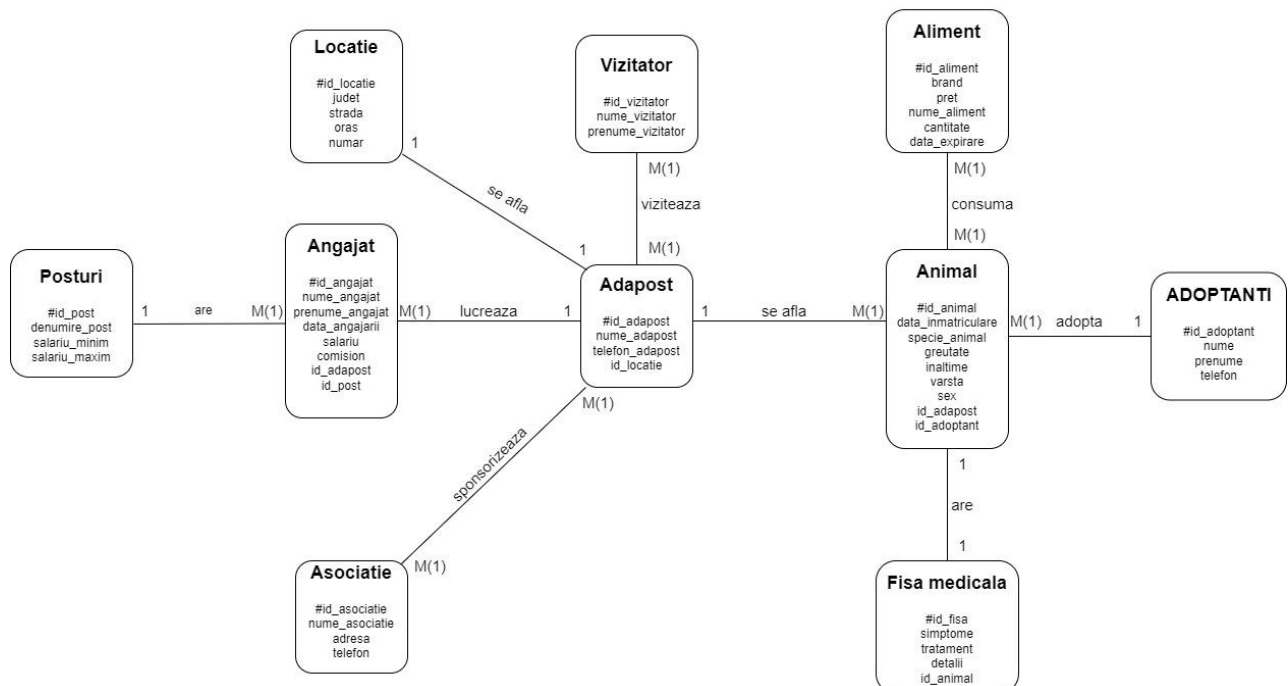
id_vizita = variabila de tip întreg, care reprezintă id-ul vizitei. Atributul nu poate fi null.

id_adapost = variabila de tip întreg, care reprezintă id-ul adăpostului pe care vizitatorul îl vizitează, nu poate fi nulă. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul ADAPOST.

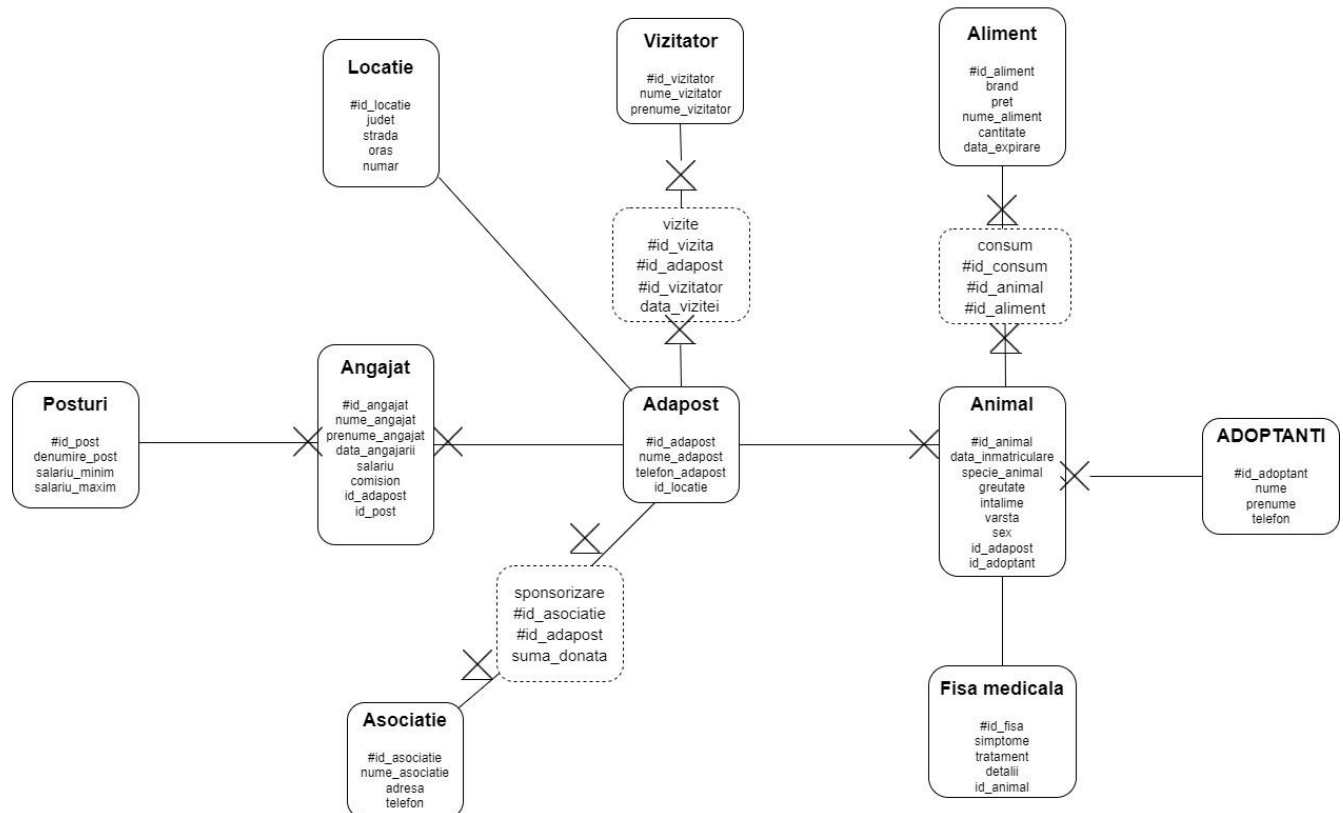
id_vizitator = variabila de tip întreg, care reprezintă id-ul vizitatorului care vizitează un adăpost, nu poate fi nulă. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul VIZITATOR.

data_vizitei = variabila de tip calendaristica, care reprezintă data vizitei unui adăpost de către un vizitator, nu poate fi nulă.

2) Realizați diagrama entitate-relație (ERD)



3) Pornind de la diagrama entitate-relație realizați diagrama conceptuală a modelului propus, integrând toate attributele necesare.



3.1) Scheme relationale

ADAPOST(#id_adapost, nume_adapost, telefon_adapost, id_locatie)

ASOCIATIE(#id_asociatie, nume_asociatie, adresa, telefon)

ANGAJAT(#id_angajat, nume_angajat, prenume_angajat, data_angajarii, salariu, comision, id_adapost, id_post)

POSTURI(#id_post, denumire_post, salariu_minim, salariu_maxim)

LOCATIE(#id_locatie, judet, oras, strada, numar)

VIZITATOR(#id_vizitator, nume, prenume)

ANIMAL(#id_animal, data_inmatriculare, specie_animal, greutate, inaltime, varsta, sex, id_adapost, id_adoptant)

ALIMENT(#id_aliment, brand, pret, nume_aliment, cantitate, data_expirare)

ADOPTANTI(#id_adoptant, nume, prenume, telefon)

FISA_MEDICALA(#id_fisa, simptome, tratament, detalii, id_animal)

SPONSORIZARE(#id_sponsorizare, #id_asociatie, #id_adapost, suma_donata)

VIZITE(#id_vizita, #id_vizitator, #id_adapost, data_vizitei)

CONSUM(#id_consum, #id_aliment, #id_animal)

4) Implementați în Oracle diagrama conceptuală realizată: definiți toate tabelele, implementând toate constrângerile de integritate necesare (chei primare, cheile externe etc).

- Crearea tabeli **ALIMENT**:

```
create table ALIMENT
```

```
(id_aliment number(5) constraint pk_aliment PRIMARY KEY,
```

```
brand varchar2(15),
```

```
nume_aliment varchar2(30),
```

```
pret number(5,2),
```

```
cantitate number(5),
```

```
data_expirare date default sysdate);
```

--Adăugarea constrângerilor:

```
ALTER TABLE ALIMENT MODIFY nume_aliment NOT NULL;
```

```
ALTER TABLE ALIMENT MODIFY data_expirare NOT NULL;
```

	❖ COLUMN_NAME	❖ DATA_TYPE	❖ NULLABLE	DATA_DEFAULT	❖ COLUMN_ID	❖ COMMENTS
1	ID_ALIMENT	NUMBER(5,0)	No	(null)	1	(null)
2	BRAND	VARCHAR2(15 BYTE)	Yes	(null)	2	(null)
3	NUME_ALIMENT	VARCHAR2(30 BYTE)	No	(null)	3	(null)
4	PRET	NUMBER(5,2)	Yes	(null)	4	(null)
5	CANTITATE	NUMBER(5,0)	Yes	(null)	5	(null)
6	DATA_EXPIRARE	DATE	No	sysdate	6	(null)

- Crearea tabeli **LOCATIE**:

```
create table LOCATIE(
```

```
id_locatie number(5) constraint pk_locatie PRIMARY KEY,
```

```
judet varchar2(20),
```

```
strada varchar2(25),
```

```
oras varchar2(20),
```

```
numar number(5));
```

--Adăugarea constrângerilor:

```
ALTER TABLE LOCATIE MODIFY judet NOT NULL;
```

```
ALTER TABLE LOCATIE MODIFY oras NOT NULL;
```

```
ALTER TABLE LOCATIE MODIFY strada NOT NULL;
```

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_LOCATIE	NUMBER (5,0)	No	(null)	1	(null)
2	JUDET	VARCHAR2 (20 BYTE)	No	(null)	2	(null)
3	STRADA	VARCHAR2 (25 BYTE)	No	(null)	3	(null)
4	ORAS	VARCHAR2 (20 BYTE)	No	(null)	4	(null)
5	NUMAR	NUMBER (5,0)	Yes	(null)	5	(null)

- Crearea tabelii **ADAPOST**:

```
create table ADAPOST(
```

```
id_adapost number(5) constraint pk_adapost PRIMARY KEY,
```

```
nume_adapost varchar2(30),
```

```
telefon varchar2(10),
```

```
id_locatie number(5), constraint fk_locatie_adapost foreign key(id_locatie)  
references LOCATIE(id_locatie));
```

--Adăugarea constrângerilor:

```
ALTER TABLE ADAPOST MODIFY nume_adapost NOT NULL;
```

```
ALTER TABLE ADAPOST MODIFY id_locatie NOT NULL;
```

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_ADAPOST	NUMBER (5,0)	No	(null)	1	(null)
2	NUME_ADAPOST	VARCHAR2 (30 BYTE)	No	(null)	2	(null)
3	TELEFON	VARCHAR2 (10 BYTE)	Yes	(null)	3	(null)
4	ID_LOCATIE	NUMBER (5,0)	No	(null)	4	(null)

- Crearea tabelii **ANIMAL**:

```
create table ANIMAL(
```

```
id_animal number(5) constraint pk_animal PRIMARY KEY,
```

```
data_inmatriculare date default sysdate,
```

```
specie_animal varchar2(50),
```

```
greutate float(4),
```

```
inaltime float(4),
```

```
varsta number(4),
```

```
sex varchar2(1),
```

id_adapost number(5),

id_adoptant number(5),

constraint fk_animal_adoptant foreign key (id_adoptant) references
ADOPTANTI(id_adoptant),

constraint fk_animal_adapost foreign key(id_adapost) references
ADAPOST(id_adapost));

--Adăugarea constrângerilor:

ALTER TABLE ANIMAL MODIFY data_inmatriculare NOT NULL;

ALTER TABLE ANIMAL MODIFY specie_animal NOT NULL;

ALTER TABLE ANIMAL MODIFY id_adapost NOT NULL;

❖ COLUMN_NAME	❖ DATA_TYPE	❖ NULLABLE	DATA_DEFAULT	❖ COLUMN_ID	❖ COMMENTS
1 ID_ANIMAL	NUMBER(5,0)	No	(null)	1 (null)	
2 DATA_INMATRICULARE	DATE	No	sysdate	2 (null)	
3 SPECIE_ANIMAL	VARCHAR2(50 BYTE)	No	(null)	3 (null)	
4 GREUTATE	FLOAT	Yes	(null)	4 (null)	
5 INALTIME	FLOAT	Yes	(null)	5 (null)	
6 VARSTA	NUMBER(4,0)	Yes	(null)	6 (null)	
7 SEX	VARCHAR2(1 BYTE)	Yes	(null)	7 (null)	
8 ID_ADAPOST	NUMBER(5,0)	No	(null)	8 (null)	
9 ID_ADOPTANT	NUMBER(5,0)	Yes	(null)	9 (null)	

- Crearea tabelii **FISA_MEDICALA**:

create table FISA_MEDICALA(

id_fisa number(5) constraint pk_fisa_medicala PRIMARY KEY,

simptome varchar2(100),

tratament varchar2(100),

detalii varchar2(200),

id_animal number(5),

constraint fk_animal_fisa foreign key(id_animal) references
ANIMAL(id_animal));

--Adăugarea constrângerilor:

ALTER TABLE FISA_MEDICALA MODIFY id_animal NOT NULL;

❖ COLUMN_NAME	❖ DATA_TYPE	❖ NULLABLE	DATA_DEFAULT	❖ COLUMN_ID	❖ COMMENTS
1 ID_FISA	NUMBER(5,0)	No	(null)	1 (null)	
2 SIMPTOME	VARCHAR2(100 BYTE)	Yes	(null)	2 (null)	
3 TRATAMENT	VARCHAR2(100 BYTE)	Yes	(null)	3 (null)	
4 DETALII	VARCHAR2(200 BYTE)	Yes	(null)	4 (null)	
5 ID_ANIMAL	NUMBER(5,0)	No	(null)	5 (null)	

- Crearea tabelii **VIZITATOR**:

```
create table VIZITATOR(  
id_vizitator number(5) constraint pk_vizitator PRIMARY KEY,  
nume varchar2(30),  
prenume varchar(30));
```

--Adăugarea constrângerilor:

```
ALTER TABLE VIZITATOR MODIFY nume NOT NULL;
```

```
ALTER TABLE VIZITATOR MODIFY prenume NOT NULL;
```

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_VIZITATOR	NUMBER (5,0)	No	(null)	1	(null)
2	NUME	VARCHAR2 (30 BYTE)	No	(null)	2	(null)
3	PRENUME	VARCHAR2 (30 BYTE)	No	(null)	3	(null)

- Crearea tabelii **ASOCIATIE**:

```
create table ASOCIATIE(  
id_asociatie number(5) constraint pk_asociatie PRIMARY KEY,  
nume_asociatie varchar2(50),  
adresa varchar2(150),  
telefon varchar2(10));
```

--Adăugarea constrângerilor:

```
ALTER TABLE ASOCIATIE MODIFY nume_asociatie NOT NULL;
```

```
ALTER TABLE ASOCIATIE MODIFY adresa NOT NULL;
```

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_ASOCIATIE	NUMBER (5,0)	No	(null)	1	(null)
2	NUME_ASOCIATIE	VARCHAR2 (50 BYTE)	No	(null)	2	(null)
3	ADRESA	VARCHAR2 (150 BYTE)	No	(null)	3	(null)
4	TELEFON	VARCHAR2 (10 BYTE)	Yes	(null)	4	(null)

- Crearea tabelii **POSTURI**:

```
create table POSTURI(  
id_post varchar2(15) constraint pk_posturi PRIMARY KEY,  
denumire_post varchar2(30),  
constraint check_post_angajat check (upper(denumire_post) in ('CONTABIL',  
'VETERINAR', 'INGRIJITOR', 'ADMINISTRATOR', 'HINGHIER',  
'APROVIZIONAR', 'ASISTENT VETERINAR'));
```

```
salariu_minim number(5),  
salariu_maxim number(5));
```

--Adăugarea constrângerilor:

```
ALTER TABLE POSTURI MODIFY denumire_post NOT NULL;
```

	❖ COLUMN_NAME	❖ DATA_TYPE	❖ NULLABLE	DATA_DEFAULT	❖ COLUMN_ID	❖ COMMENTS
1	ID_POST	VARCHAR2 (15 BYTE)	No	(null)	1 (null)	
2	DENUMIRE_POST	VARCHAR2 (30 BYTE)	No	(null)	2 (null)	
3	SALARIU_MINIM	NUMBER (5,0)	Yes	(null)	3 (null)	
4	SALARIU_MAXIM	NUMBER (5,0)	Yes	(null)	4 (null)	

- Crearea tabeli **ANGAJAT**:

```
create table ANGAJAT(  
id_angajat number(5) constraint pk_angajat PRIMARY KEY,  
nume_angajat varchar2(30),  
prenume_angajat varchar2(30),  
data_angajarii date default sysdate,  
salariu number(5),  
comision float(4),  
id_adapost number(5),  
constraint fk_angajat_adapost foreign key(id_adapost) references  
ADAPOST(id_adapost),  
id_post varchar(15),  
constraint fk_angajat_post foreign key(id_post) references  
POSTURI(id_post));
```

--Adăugarea constrângerilor:

```
ALTER TABLE ANGAJAT MODIFY nume_angajat NOT NULL;  
ALTER TABLE ANGAJAT MODIFY prenume_angajat NOT NULL;  
ALTER TABLE ANGAJAT MODIFY data_angajarii NOT NULL;  
ALTER TABLE ANGAJAT MODIFY id_adapost NOT NULL;  
ALTER TABLE ANGAJAT MODIFY id_post NOT NULL;
```

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_ANGAJAT	NUMBER(5,0)	No	(null)	1	(null)
2	NUME_ANGAJAT	VARCHAR2(30 BYTE)	No	(null)	2	(null)
3	PRENUME_ANGAJAT	VARCHAR2(30 BYTE)	No	(null)	3	(null)
4	DATA_ANGAJARII	DATE	No	sysdate	4	(null)
5	SALARIU	NUMBER(5,0)	Yes	(null)	5	(null)
6	COMISION	FLOAT	Yes	(null)	6	(null)
7	ID_ADAPOST	NUMBER(5,0)	No	(null)	7	(null)
8	ID_POST	VARCHAR2(15 BYTE)	No	(null)	8	(null)

- Crearea tabelii **ADOPTANTI**:

```
create table ADOPTANTI(
id_adoptant number(5) constraint pk_adoptanti PRIMARY KEY,
nume varchar2(30),
prenume varchar2(30),
telefon varchar2(10));
```

--Adăugarea constrângerilor:

```
ALTER TABLE ADOPTANTI MODIFY nume NOT NULL;
```

```
ALTER TABLE ADOPTANTI MODIFY prenume NOT NULL;
```

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_ADOPTANT	NUMBER(5,0)	No	(null)	1	(null)
2	NUME	VARCHAR2(30 BYTE)	No	(null)	2	(null)
3	PRENUME	VARCHAR2(30 BYTE)	No	(null)	3	(null)
4	TELEFON	VARCHAR2(10 BYTE)	Yes	(null)	4	(null)

- Crearea tabelii **SPONSORIZARE**:

```
create table SPONSORIZARE(
id_sponsorizare number(5) constraint pk_sponsorizare PRIMARY KEY,
id_asociatie number(5),
id_adapost number(5),
suma_donata number(10),
constraint fk_spon_asoc foreign key(id_asociatie) references
ASOCIATIE(id_asociatie),
constraint fk_spon_adap foreign key(id_adapost) references
ADAPOST(id_adapost));
```

--Adăugarea constrângerilor:

```
ALTER TABLE SPONSORIZARE MODIFY id_asociatie NOT NULL;
```

```
ALTER TABLE SPONSORIZARE MODIFY id_adapost NOT NULL;
```

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_SPONSORIZARE	NUMBER(5,0)	No	(null)	1	(null)
2	ID_ASOCIATIE	NUMBER(5,0)	No	(null)	2	(null)
3	ID_ADAPOST	NUMBER(5,0)	No	(null)	3	(null)
4	SUMA_DONATA	NUMBER(10,0)	Yes	(null)	4	(null)

- Crearea tabeli **VIZITE**:

```
create table VIZITE(
```

```
id_vizita number(5) constraint pk_vizite PRIMARY KEY,
```

```
data_vizitei date default sysdate,
```

```
id_vizitator number(5), constraint fk_vizit foreign key(id_vizitator) references  
VIZITATOR(id_vizitator),
```

```
id_adapost number(5), constraint fk_adapost foreign key(id_adapost)  
references ADAPOST(id_adapost));
```

--Adăugarea constrângerilor:

```
ALTER TABLE VIZITE MODIFY data_vizitei NOT NULL;
```

```
ALTER TABLE VIZITE MODIFY id_vizitator NOT NULL;
```

```
ALTER TABLE VIZITE MODIFY id_adapost NOT NULL;
```

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_VIZITA	NUMBER(5,0)	No	(null)	1	(null)
2	DATA_VIZITEI	DATE	No	sysdate	2	(null)
3	ID_VIZITATOR	NUMBER(5,0)	No	(null)	3	(null)
4	ID_ADAPOST	NUMBER(5,0)	No	(null)	4	(null)

- Crearea tabeli **CONSUM**:

```
create table CONSUM(
```

```
id_consum number(5) constraint pk_consum PRIMARY KEY,
```

```
id_animal number(5), constraint fk_animal foreign key(id_animal) references  
ANIMAL(id_animal),
```

```
id_aliment number(5), constraint fk_aliment foreign key(id_aliment) references  
ALIMENT(id_aliment));
```

--Adăugarea constrângerilor:

```
ALTER TABLE CONSUM MODIFY id_animal NOT NULL;
```

```
ALTER TABLE CONSUM MODIFY id_aliment NOT NULL;
```


COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_CONSUM	NUMBER(5,0)	No	(null)	1	(null)
2 ID_ANIMAL	NUMBER(5,0)	No	(null)	2	(null)
3 ID_ALIMENT	NUMBER(5,0)	No	(null)	3	(null)

5) Adăugați informații coerente în tabelele create (minim 5 înregistrări pentru fiecare entitate independentă; minim 10 înregistrări pentru tabela asociativă).

Inserare in tabelul **ALIMENT**:

```
create sequence seq_aliment
```

```
increment by 1
```

```
start with 10
```

```
maxvalue 100
```

```
nocycle;
```

```
INSERT INTO ALIMENT
```

```
VALUES(SEQ_ALIMENT.NEXTVAL, 'GRANULI', 'Anti-Hairball', 20.5, 10,  
,to_date('30/06/2022','dd/mm/yyyy'));
```

```
INSERT INTO ALIMENT
```

```
VALUES(SEQ_ALIMENT.NEXTVAL, 'WHISKAS', 'Multipack Pliculete', 17.5, 9,  
to_date('01/01/2022','dd/mm/yyyy'));
```

```
INSERT INTO ALIMENT
```

```
VALUES(SEQ_ALIMENT.NEXTVAL, 'HEALTHY-PET', 'Multipack Junior  
Pliculete', 40.99, 5, to_date('04/02/2021','dd/mm/yyyy'));
```

```
INSERT INTO ALIMENT
```

```
VALUES(SEQ_ALIMENT.NEXTVAL, 'YOUR-PET', 'Multipack Senior Pliculete',  
50.19, 4, to_date('13/06/2021','dd/mm/yyyy'));
```

```
INSERT INTO ALIMENT
```

```
VALUES(SEQ_ALIMENT.NEXTVAL, 'FORZZA', 'Pernute Crocante', 19.99, 21,  
to_date('30/07/2023','dd/mm/yyyy'));
```

```
INSERT INTO ALIMENT
```

```
VALUES(SEQ_ALIMENT.NEXTVAL, 'CATPET', 'Hrana umeda', 30,  
17,to_date('01/03/2022','dd/mm/yyyy'));
```

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'GRANULI', 'Conserva-Ton', 60.39,
11,to_date('30/06/2022','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'WHISKAS', 'Conserva-Pui',
17.49,69,to_date('12/02/2021','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'GRANULI', 'Conserva-Pui', 23.59, 42,
to_date('22/03/2022','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'GRANULI', 'Anti-Hairball', 24.99,
13,to_date('17/09/2023','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'CATPET', 'Conserva-Miel', 24.5, 15
,to_date('01/01/2023','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'CATPET', 'Pernute Moi', 22.99,
27,to_date('19/10/2022','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'FORZZA', 'Pernute Umede', 9.99, 32,
to_date('28/04/2022','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'YOUR-PET', 'Sterile Pui', 10.99,
10,to_date('31/12/2022','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'YOUR-PET', 'Indoor Pui', 20.5,
37,to_date('22/06/2023','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'HEALTHY-PET', 'Sterile Vita', 80.5,
7,to_date('31/07/2024','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'WHISKAS', 'Pachet-Economic', 99.99,
2,to_date('22/04/2022','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'WHISKAS', 'Dentabites', 15.5, 11, to_date('19/08/2021','dd/mm/yyyy'));

INSERT INTO ALIMENT

VALUES(SEQ_ALIMENT.NEXTVAL, 'WHISKAS', 'Vitamin E-Xtra', 79.5, 46, to_date('07/07/2027','dd/mm/yyyy'));

	ID_ALIMENT	BRAND	NUME_ALIMENT	PRET	CANTITATE	DATA_EXPIRARE
1	29	GRANULI	Anti-Hairball	20.5	10	30-JUN-22
2	30	WHISKAS	Multipack Pliculete	17.5	9	01-JAN-22
3	31	HEALTHY-PET	Multipack Junior Pliculete	40.99	5	04-FEB-21
4	32	YOUR-PET	Multipack Senior Pliculete	50.19	4	13-JUN-21
5	33	FORZZA	Pernute Crocante	19.99	21	30-JUL-23
6	34	CATPET	Hrana umeda	30	17	01-MAR-22
7	35	GRANULI	Conserva-Ton	60.39	11	30-JUN-22
8	36	WHISKAS	Conserva-Pui	17.49	69	12-FEB-21
9	37	GRANULI	Conserva-Pui	23.59	42	22-MAR-22
10	38	GRANULI	Anti-Hairball	24.99	13	17-SEP-23
11	39	CATPET	Conserva-Miel	24.5	15	01-JAN-23
12	40	CATPET	Pernute Moi	22.99	27	19-OCT-22
13	41	FORZZA	Pernute Umede	9.99	32	28-APR-22
14	42	YOUR-PET	Sterile Pui	10.99	10	31-DEC-22
15	43	YOUR-PET	Indoor Pui	20.5	37	22-JUN-23
16	44	HEALTHY-PET	Sterile Vita	80.5	7	31-JUL-24
17	45	WHISKAS	Pachet-Economic	99.99	2	22-APR-22
18	46	WHISKAS	Dentabites	15.5	11	19-AUG-21
19	47	WHISKAS	Vitamin E-Xtra	79.5	46	07-JUL-27

Inserare in tabelul **LOCATIE**:

create sequence seq_locatie

increment by 1

start with 1

maxvalue 100

nocycle;

INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'OLT', 'ALEEA MUNCII', 'SLATINA', 7);

INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'BRASOV', 'FLORILOR', 'BRASOV', 19);

```
INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'CONSTANTA',  
'VIILOR', 'MANGALIA', 96);
```

```
INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'DOLJ', 'UNIRII',  
'CALAFAT', 157);
```

```
INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'CLUJ',  
'TEILOR', 'TURDA', 9);
```

```
INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'BOTOSANI',  
'ION CREANGA', 'BOTOSANI', 28);
```

```
INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'BUCURESTI',  
'BISERICII', 'BUCURESTI', 77);
```

```
INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'SIBIU',  
'CRINULUI', 'SIBIU', 139);
```

```
INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'GALATI',  
'PLOPILOR', 'TECUCI', 73);
```

```
INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'MURES',  
'OITUZ', 'SIGHISOARA', 45);
```

```
INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'TIMIS', 'PACII',  
'LUGOJ', 232);
```

```
INSERT INTO LOCATIE VALUES (SEQ_LOCATIE.NEXTVAL,'NEAMT',  
'LUJERULUI', 'PIATRA NEAMT', 830);
```

	ID_LOCATIE	JUDET	STRADA	ORAS	NUMAR
1	2	BRASOV	FLORILOR	BRASOV	19
2	3	CONSTANTA	VIILOR	MANGALIA	96
3	4	DOLJ	UNIRII	CALAFAT	157
4	5	CLUJ	TEILOR	TURDA	9
5	6	BOTOSANI	ION CREANGA	BOTOSANI	28
6	7	BUCURESTI	BISERICII	BUCURESTI	77
7	8	SIBIU	CRINULUI	SIBIU	139
8	9	GALATI	PLOPILOR	TECUCI	73
9	10	MURES	OITUZ	SIGHISOARA	45
10	11	TIMIS	PACII	LUGOJ	232
11	12	NEAMT	LUJERULUI	PIATRA NEAMT	830
12	1	OLT	ALEEA MUNCII	SLATINA	7

Inserare in tabelul **ADAPOST**:

```
create sequence seq_adapost
```

```
increment by 1
```

start with 1001

maxvalue 2000

nocycle;

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL, 'RAZA DE  
SPERANTA', '0248294466', 1);
```

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL, 'DOGS FIRST',  
'0214904007', 6);
```

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL, 'ANIMAL  
LIFE', '0251370070', 3);
```

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL, 'ANIMAL  
RESCUE', '0734301394', 5);
```

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL, 'NEW LIFE',  
'0777000000', 2);
```

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL, 'OUR  
FRIENDS', '0786235333', 7);
```

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL,  
'ZBURATORII', '0763223453', 9);
```

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL, 'CAT LIFE',  
'0745375312', 11);
```

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL, 'RED PANDA',  
'0721234222', 8);
```

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL, 'CATEI ODAI',  
'0734231224', 12);
```

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL, 'ASPA  
BUCURESTI', '0756320488', 10);
```

```
INSERT INTO ADAPOST VALUES (SEQ_ADAPOST.NEXTVAL, 'STEAUA  
SPERANTEI', '0775200135', 4);
```

	ID_ADAPOST	NUME_ADAPOST	TELEFON	ID_LOCATIE
1	1001	RAZA DE SPERANTA	0248294466	1
2	1002	DOGS FIRST	0214904007	6
3	1003	ANIMAL LIFE	0251370070	3
4	1004	ANIMAL RESCUE	0734301394	5
5	1005	NEW LIFE	0777000000	2
6	1006	OUR FRIENDS	0786235333	7
7	1007	ZBURATORII	0763223453	9
8	1008	CAT LIFE	0745375312	11
9	1009	RED PANDA	0721234222	8
10	1010	CATEI ODAI	0734231224	12
11	1011	ASPA BUCURESTI	0756320488	10
12	1012	STEAUA SPERANTEI	0775200135	4

Inserare in tabelul **ANIMAL**:

create sequence seq_animal

increment by 1

start with 1

maxvalue 500

nocycle;

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('30/06/2015','dd/mm/yyyy'), 'CAINE', 5.8, 0.54, 2, 'M', 1001, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('12/11/2017','dd/mm/yyyy'), 'PISICA', 3.2, 0.44, 1, 'M', 1001, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('08/05/2019','dd/mm/yyyy'), 'CAINE', 9.3, 1.21, 3, 'M', 1001, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('23/01/2020','dd/mm/yyyy'), 'PISICA', 2.6, 0.31, 2, 'F',1001, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('15/08/2014','dd/mm/yyyy'), 'PISICA', 3.7, 0.48, 6, 'F', 1001, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('11/11/2011','dd/mm/yyyy'), 'CAINE', 4.9, 0.51, 8, 'F', 1002, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('01/06/2012','dd/mm/yyyy'), 'PISICA', 1.94, 0.25, 4, 'M',1002, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('01/09/2008','dd/mm/yyyy'), 'CAINE', 4.84, 0.43, 4, 'F',1002, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('14/05/2011','dd/mm/yyyy'), 'PISICA', 4.3, 0.39, 7, 'M', 1003, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('26/02/2012','dd/mm/yyyy'), 'CAINE', 6.91, 0.57, 1, 'M', 1003, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('23/06/2007','dd/mm/yyyy'), 'CAINE', 9.4, 0.98, 4, 'M', 1003, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('24/05/2015','dd/mm/yyyy'), 'CAINE', 10.1, 1.32, 3, 'F',1003, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('11/04/2021','dd/mm/yyyy'), 'PISICA', 9.42, 0.49, 3, 'F', 1004, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('04/02/2021','dd/mm/yyyy'), 'CAINE', 8.54, 0.83, 4, 'M', 1004, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('19/03/2021','dd/mm/yyyy'), 'CAINE', 5.65, 0.37, 5, 'F', 1004, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('18/01/2021','dd/mm/yyyy'), 'PISICA', 6.74, 0.37, 1, 'F', 1005, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('25/02/2021','dd/mm/yyyy'), 'CAINE', 6.66, 0.64, 1, 'M', 1005, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('06/06/2021','dd/mm/yyyy'), 'PISICA', 3.84, 0.24, 1, 'F',1006, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('30/05/2021','dd/mm/yyyy'), 'CAINE', 5.24, 0.34, 1, 'F', 1006, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('19/05/2021','dd/mm/yyyy'), 'PISICA', 5.81, 0.32, 1, 'F', 1006, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('01/11/2020','dd/mm/yyyy'), 'CAINE', 7.83, 0.71, 2, 'M',1007, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('18/03/2021','dd/mm/yyyy'), 'PISICA', 7.21, 0.28, 1, 'F', 1007, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('17/02/2021','dd/mm/yyyy'), 'CAINE', 5.19, 0.4, 1, 'F', 1007, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('09/03/2021','dd/mm/yyyy'), 'PISICA', 9.41, 0.45, 6, 'F', 1008, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('28/01/2021','dd/mm/yyyy'), 'CAINE', 7.4, 0.66, 4, 'M', 1008, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('13/06/2021','dd/mm/yyyy'), 'PISICA', 3.29, 0.27, 3, 'M', 1008, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('09/12/2020','dd/mm/yyyy'), 'CAINE', 11.2, 1.291, 3, 'M', 1008, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('11/04/2021','dd/mm/yyyy'), 'PISICA', 4.91, 0.39, 7, 'F',1009, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('06/02/2021','dd/mm/yyyy'), 'CAINE', 6.78, 0.627, 9, 'F',1009, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('27/12/2020','dd/mm/yyyy'), 'PISICA', 5.89, 0.40, 8, 'F', 1009, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('23/10/2019','dd/mm/yyyy'), 'CAINE', 4.53, 0.32, 3, 'M', 1010, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('15/03/2018','dd/mm/yyyy'), 'CAINE', 3.61, 0.29, 4, 'M', 1010, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('29/04/2020','dd/mm/yyyy'), 'CAINE', 2.37, 0.26, 2, 'F', 1010, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('21/09/2020','dd/mm/yyyy'), 'PISICA', 5.89, 0.34, 2, 'M', 1010, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('04/04/2021','dd/mm/yyyy'), 'PISICA', 3.76, 0.28, 1, 'F', 1011, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('11/06/2020','dd/mm/yyyy'), 'CAINE', 8.91, 0.52, 4, 'M', 1011, NULL);
```



```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('09/03/2017','dd/mm/yyyy'), 'PISICA', 4.23, 0.31, 6, 'F', 1011, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('29/06/2020','dd/mm/yyyy'), 'CAINE', 9.34, 0.89, 6, 'F', 1012, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('19/07/2020','dd/mm/yyyy'), 'PISICA', 5.62, 0.42, 4, 'F', 1012, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('23/11/2021','dd/mm/yyyy'), 'CAINE', 10.43, 1.08, 9, 'F', 1012, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('14/01/2021','dd/mm/yyyy'), 'CAINE', 4.53, 0.37, 2, 'F', 1012, NULL);
```

```
INSERT INTO ANIMAL VALUES (SEQ_ANIMAL.NEXTVAL,  
to_date('21/05/2018','dd/mm/yyyy'), 'PISICA', 3.61, 0.26, 1, 'F', 1012, NULL);
```

ID_ANIMAL	DATA_INMATRICULARE	SPECIE_ANIMAL	GREUTATE	INALTIME	VARSTA	SEX	ID_ADAPOST	ID_ADOPTANT
2	2 12-NOV-17	PISICA	3.2	0.44	1 M		1001	(null)
3	3 08-MAY-19	CAINE	9.3	1.2	3 M		1001	(null)
4	4 23-JAN-20	PISICA	2.6	0.31	2 F		1001	(null)
5	5 15-AUG-14	PISICA	3.7	0.48	6 F		1001	(null)
6	6 11-NOV-11	CAINE	4.9	0.51	8 F		1002	(null)
7	7 01-JUN-12	PISICA	1.9	0.25	4 M		1002	(null)
8	8 01-SEP-08	CAINE	4.8	0.43	4 F		1002	(null)
9	9 14-MAY-11	PISICA	4.3	0.39	7 M		1003	(null)
10	10 26-FEB-12	CAINE	6.9	0.57	1 M		1003	(null)
11	11 23-JUN-07	CAINE	9.4	0.98	4 M		1003	(null)
12	12 24-MAY-15	CAINE	10	1.3	3 F		1003	(null)
13	13 11-APR-21	PISICA	9.4	0.49	3 F		1004	(null)
14	14 04-FEB-21	CAINE	8.5	0.83	4 M		1004	(null)
15	15 19-MAR-21	CAINE	5.7	0.37	5 F		1004	(null)
16	16 18-JAN-21	PISICA	6.7	0.37	1 F		1005	(null)
17	17 25-FEB-21	CAINE	6.7	0.64	1 M		1005	(null)
18	18 06-JUN-21	PISICA	3.8	0.24	1 F		1006	(null)
19	19 30-MAY-21	CAINE	5.2	0.34	1 F		1006	(null)
20	20 19-MAY-21	PISICA	5.8	0.32	1 F		1006	(null)
21	21 01-NOV-20	CAINE	7.8	0.71	2 M		1007	(null)
22	22 18-MAR-21	PISICA	7.2	0.28	1 F		1007	(null)
23	23 17-FEB-21	CAINE	5.2	0.4	1 F		1007	(null)
24	24 09-MAR-21	PISICA	9.4	0.45	6 F		1008	(null)
25	25 28-JAN-21	CAINE	7.4	0.66	4 M		1008	(null)
26	26 13-JUN-21	PISICA	3.3	0.27	3 M		1008	(null)
27	27 09-DEC-20	CAINE	11	1.3	3 M		1008	(null)
28	28 11-APR-21	PISICA	4.9	0.39	7 F		1009	(null)
29	29 06-FEB-21	CAINE	6.8	0.63	9 F		1009	(null)
30	30 27-DEC-20	PISICA	5.9	0.4	8 F		1009	(null)
31	31 23-OCT-19	CAINE	4.5	0.32	3 M		1010	(null)
32	32 15-MAR-18	CAINE	3.6	0.29	4 M		1010	(null)
33	33 29-APR-20	CAINE	2.4	0.26	2 F		1010	(null)
34	34 21-SEP-20	PISICA	5.9	0.34	2 M		1010	(null)
35	35 04-APR-21	PISICA	3.8	0.28	1 F		1011	(null)
36	36 11-JUN-20	CAINE	8.9	0.52	4 M		1011	(null)
37	37 09-MAR-17	PISICA	4.2	0.31	6 F		1011	(null)
38	38 29-JUN-20	CAINE	9.3	0.89	6 F		1012	(null)
39	39 19-JUL-20	PISICA	5.6	0.42	4 F		1012	(null)
40	40 23-NOV-21	CAINE	10	1.1	9 F		1012	(null)
41	41 14-JAN-21	CAINE	4.5	0.37	2 F		1012	(null)
42	42 21-MAY-18	PISICA	3.6	0.26	1 F		1012	(null)

Inserare in tabelul **FISA_MEDICALA**:

```
create sequence seq_fisa
```

```
increment by 1
```

```
start with 50
```

```
maxvalue 200
```

```
nocycle;
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Foarte agresiv', 1);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, 'Leziune',  
'Bandaj ocular', 'Este somnoros', 7);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Mananca mult', 3);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Face multa galagie', 2);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Foarte cuminte', 4);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, 'Leziune  
picior stang', 'Bandajare', 'Schimbate odata la 2 zile', 5);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Prietenos', 9);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, 'Arsura',  
'Compresie sterila', NULL, 8);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 6);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Depresiv', 10);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Foarte agresiva', 16);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 13);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Are o culoare rara', 11);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Ti place sa se joace', 12);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Este ranit la piciorul stang', 14);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 15);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, 'Intoxicatie',  
'Regim alimentar corespunzator', 'O buna igiena alimentara', 19);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 17);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Probleme cu vederea', 18);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Este batrana', 20);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 27);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, 'Piciorul drept  
rupt', 'Gips', 'Pentru 4 saptamani', 29);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Este blanda', 28);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 30);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, 'Plaga  
deschisa', 'Operatie urgenta', NULL, 21);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, 'Oarba',  
NULL, NULL, 24);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, 'Obezitate',  
'Regim alimentar corespunzator', NULL, 25);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, 'Picior  
zdrobit', 'Amputare', NULL, 22);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, 'Surda',  
NULL, NULL, 23);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 26);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Foarte cuminte', 31);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 33);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Foarte jucaus', 34);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, 'Galagios', 32);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 37);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 35);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 36);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 39);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 38);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 40);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 42);
```

```
INSERT INTO FISA_MEDICALA VALUES(SEQ_FISA.NEXTVAL, NULL,  
NULL, NULL, 41);
```

2	51	Leziune	Bandaj ocular	Este somnoroș	7
3	52	(null)	(null)	Mananca mult	3
4	53	(null)	(null)	Face multa galagie	2
5	54	(null)	(null)	Foarte cuminte	4
6	55	Leziune picior stang	Bandajare	Schimbate odata la 2 zile	5
7	56	(null)	(null)	Prietenos	9
8	57	Arsura	Compresie sterila	(null)	8
9	58	(null)	(null)	(null)	6
10	59	(null)	(null)	Depresiv	10
11	60	(null)	(null)	Foarte agresiva	16
12	61	(null)	(null)	(null)	13
13	62	(null)	(null)	Are o culoare rara	11
14	63	(null)	(null)	Ii place sa se joace	12
15	64	(null)	(null)	Este ranit la piciorul stang	14
16	65	(null)	(null)	(null)	15
17	66	Intoxicatie	Regim alimentar corespunzator	O buna igiena alimentara	19
18	67	(null)	(null)	(null)	17
19	68	(null)	(null)	Probleme cu vederea	18
20	69	(null)	(null)	Este batrana	20
21	70	(null)	(null)	(null)	27
22	71	Piciorul drept rupt	Gips	Pentru 4 saptamani	29
23	72	(null)	(null)	Este blanda	28
24	73	(null)	(null)	(null)	30
25	74	Plaga deschisa	Operatie urgenta	(null)	21
26	75	Oarba	(null)	(null)	24
27	76	Obezitate	Regim alimentar corespunzator	(null)	25
28	77	Picior zdrobit	Amputare	(null)	22
29	78	Surda	(null)	(null)	23
30	79	(null)	(null)	(null)	26
31	80	(null)	(null)	Foarte cuminte	31
32	81	(null)	(null)	(null)	33
33	82	(null)	(null)	Foarte jucaus	34
34	83	(null)	(null)	Galagios	32
35	84	(null)	(null)	(null)	37
36	85	(null)	(null)	(null)	35
37	86	(null)	(null)	(null)	36
38	87	(null)	(null)	(null)	39
39	88	(null)	(null)	(null)	38
40	89	(null)	(null)	(null)	40
41	90	(null)	(null)	(null)	42
42	91	(null)	(null)	(null)	41

Inserare in tabelul **VIZITATOR**:

create sequence seq_viz

increment by 1

start with 200

maxvalue 500

nocycle;

INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Ene', 'Marius');

INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Doncea', 'Beatrice');

```
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Serban', 'Mihai');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Comanescu', 'Ionut');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Ivana', 'Florinel');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Ioana', 'Andreea');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Silviu', 'Bogdan');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Ionut', 'Iancu');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Petrolu', 'Tiriac');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Nastase', 'Aurel');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Alex', 'Nicoleta');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Maria', 'Predut');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Marinela', 'Stancu');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Stelea', 'Marian');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Sandu', 'Maria');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Morar', 'Florina');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Ramona', 'Carmen');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Dorin', 'Grigore');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Bratu', 'Sonia');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Vladau', 'Serban');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Toderoiu', 'Georgiana');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Alexe', 'Stefana');
INSERT INTO VIZITATOR VALUES(seq_viz.nextval, 'Daniela', 'Mariana');
```

	ID_VIZITATOR	NUME	PRENUME
1	200	Ene	Marius
2	201	Doncea	Beatrice
3	202	Serban	Mihai
4	203	Comanescu	Ionut
5	204	Ivana	Florinel
6	205	Ioana	Andreea
7	206	Silviu	Bogdan
8	207	Ionut	Iancu
9	208	Petrolu	Tiriac
10	209	Nastase	Aurel
11	210	Alex	Nicoleta
12	211	Maria	Predut
13	212	Marinela	Stancu
14	213	Stelea	Marian
15	214	Sandu	Maria
16	215	Morar	Florina
17	216	Ramona	Carmen
18	217	Dorin	Grigore
19	218	Bratu	Sonia
20	219	Vladau	Serban
21	220	Toderoiu	Georgiana
22	221	Alexe	Stefana
23	222	Daniela	Mariana

Inserare in tabelul **ASOCIATIE**:

```
create sequence seq_as
```

```
increment by 1
```

```
start with 100
```

```
maxvalue 150
```

```
nocycle;
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'Deja-Vu', 'Alexandru Ioan  
Cuza, Slatina, Nr 2', '0756190594');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'Gucci', 'Siretului, Sibiu, Nr  
6','0253216030');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'Enel', 'Gheorghe Doja,  
Galati, Nr 1024','0264425992');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'Romp petrol', 'Argesului, Cluj-  
Napoca, Nr 14', '0213524416');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'Versace', 'Caprelor, Ploiesti,  
Nr 8', '0264343735');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'Google Romania',  
'Albinelor, Corabia, Nr 41', '0244325903');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'Dristor', 'Mamaligilor,  
Bucuresti, Nr 254', '0264343735');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'Microsoft Romania', 'Regie,  
Bucuresti, Nr 76', '0722548789');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'CEZ', 'Margineni, Cluj, Nr  
194', '0773654651');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'Bit Defender', 'Independetei,  
Bucuresti, Nr 43', '0733624185');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'KRUK', 'Ciobanului, Slatina,  
Nr 95', '0724845132');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'Play Tika', 'Pacii, Neamt, Nr  
120', '0741256365');
```

```
INSERT INTO ASOCIATIE VALUES (seq_as.nextval, 'Metrorex', 'Miresmelor,  
Mures, Nr 321', '0741526376');
```

ID_ASOCIATIE	NUME_ASOCIATIE	ADRESA	TELEFON
1	100 Deja-Vu	Alexandru Ioan Cuza, Slatina, Nr 2	0756190594
2	101 Gucci	Siretului, Sibiu, Nr 6	0253216030
3	102 Enel	Gheorghe Doja, Galati, Nr 1024	0264425992
4	103 Rompetrol	Argesului, Cluj-Napoca, Nr 14	0213524416
5	104 Versace	Caprelor, Ploiesti, Nr 8	0264343735
6	105 Google Romania	Albinelor, Corabia, Nr 41	0244325903
7	106 Dristor	Mamaligilor, Bucuresti, Nr 254	0264343735
8	107 Microsoft Romania	Regie, Bucuresti, Nr 76	0722548789
9	108 CEZ	Margineni, Cluj, Nr 194	0773654651
10	109 Bit Defender	Independetei, Bucuresti, Nr 43	0733624185
11	110 KRUK	Ciobanului, Slatina, Nr 95	0724845132
12	111 Play Tika	Pacii, Neamt, Nr 120	0741256365
13	112 Metrorex	Miresmelor, Mures, Nr 321	0741526376

Inserare in tabelul **POSTURI**:

```
INSERT INTO POSTURI VALUES ('ING', 'INGRIJITOR', 2500, 3500);
```



```
INSERT INTO POSTURI VALUES ('VET', 'VETERINAR', 5500, 7900);
INSERT INTO POSTURI VALUES ('HIN', 'HINGHIER', 1500, 2750);
INSERT INTO POSTURI VALUES ('APR', 'APROVIZIONAR', 1900, 3000);
INSERT INTO POSTURI VALUES ('ASTVET', 'ASISTENT VETERINAR', 3000,
4200);
INSERT INTO POSTURI VALUES ('ADM', 'ADMINISTRATOR', 2600, 3420);
INSERT INTO POSTURI VALUES ('CON', 'CONTABIL', 2900, 4500);
```

	ID_POST	DENUMIRE_POST	SALARIU_MINIM	SALARIU_MAXIM
1	ING	INGRIJITOR	2500	3500
2	VET	VETERINAR	5500	7900
3	HIN	HINGHIER	1500	2750
4	APR	APROVIZIONAR	1900	3000
5	ASTVET	ASISTENT VETERINAR	3000	4200
6	ADM	ADMINISTRATOR	2600	3420
7	CON	CONTABIL	2900	4500

Inserare in tabelul **ANGAJAT**:

create sequence seq_ang

increment by 1

start with 1000

maxvalue 1200

nocycle;

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'CHIRITA',
'ANDREI', to_date('19-10-2008', 'dd-mm-yyyy'), 2900, 0.15, 1001, 'CON');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MATEESCU',
'VALENTIN', to_date('27-09-2013', 'dd-mm-yyyy') , 5700, 0.10, 1001, 'VET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'POPESCU',
'ANDREI', to_date('07-08-2010', 'dd-mm-yyyy') , 2000, 0.08, 1001, 'APR');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'POPESCU',
'ANDREEA', to_date('03-09-2015', 'dd-mm-yyyy') , 2800, 0.05, 1001, 'ADM');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'IONASCU',
'DANIEL', to_date('21-03-2011', 'dd-mm-yyyy') , 3100, NULL, 1001, 'ING');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'ILEA', 'ADELINA',  
to_date('26-02-2014', 'dd-mm-yyyy') , 3500, NULL, 1002, 'CON');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'CALIN', 'BOGDAN',  
to_date('14-11-2008', 'dd-mm-yyyy') , 5900, 0.15, 1002, 'VET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'PEICU', 'LIVIU',  
to_date('11-05-2011', 'dd-mm-yyyy') , 2600, 0.10, 1002, 'ADM');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'RUSEANU',  
'OVIDIU', to_date('17-04-2009', 'dd-mm-yyyy') , 2500, NULL, 1002, 'APR');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'CALIN',  
'SILVANA', to_date('29-03-2012', 'dd-mm-yyyy') , 3200, NULL, 1002, 'ING');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'SOREATA',  
'ADRIAN', to_date('20-08-2009', 'dd-mm-yyyy') , 3900, NULL, 1003, 'CON');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'DICU',  
'ALEXANDRU', to_date('11-01-2010', 'dd-mm-yyyy') , 6400, NULL, 1003, 'VET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MACOVEI',  
'RAZVAN', to_date('19-04-2012', 'dd-mm-yyyy') , 3300, 0.05, 1003, 'ADM');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'RADUCU',  
'MIHNEA', to_date('01-05-2009', 'dd-mm-yyyy') , 2200, NULL, 1003, 'APR');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'RIZESCU',  
'ADRIAN', to_date('30-07-2009', 'dd-mm-yyyy') , 2600, NULL, 1003, 'HIN');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MARINESCU',  
'COSMIN', to_date('18-10-2010', 'dd-mm-yyyy') , 3000, 0.05, 1004, 'CON');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'VOICULESCU',  
'CATALIN', to_date('16-08-2015', 'dd-mm-yyyy') , 7500, NULL, 1004, 'VET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'BARBU', 'COSMIN',  
to_date('15-01-2013', 'dd-mm-yyyy') , 2600, 0.15, 1004, 'ADM');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MACOVEI',  
'BOGDAN',to_date('08-09-2015', 'dd-mm-yyyy') , 2000, NULL, 1004, 'APR');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'VOICA',  
'GEORGIANA',to_date('29-07-2006', 'dd-mm-yyyy') , 2950, NULL, 1004, 'ING');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MONETE',  
'RAZVAN',to_date('02-01-2016', 'dd-mm-yyyy') , 4000, NULL, 1005, 'CON');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'BUDURE',  
'DANIEL', to_date('23-10-2014', 'dd-mm-yyyy') , 6500, NULL, 1005, 'VET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MAZILU',  
'VIOREL', to_date('27-06-2011', 'dd-mm-yyyy') , 2600, NULL, 1005, 'ADM');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'DONCEA',  
'BEATRICE', to_date('17-06-2013', 'dd-mm-yyyy') , 1900, 0.10, 1005, 'APR');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'ENE', 'ANDREI',  
to_date('10-09-2010', 'dd-mm-yyyy') , 5500, 0.05, 1006, 'VET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'IACOB', 'RAZVAN',  
to_date('02-09-2012', 'dd-mm-yyyy') , 2600, NULL, 1006, 'ADM');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'UNGUREANU',  
'MARIUS', to_date('17-04-2015', 'dd-mm-yyyy') , 2600, NULL, 1006, 'APR');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'OLTEANU',  
'STEFAN', to_date('15-07-2009', 'dd-mm-yyyy') , 3000, NULL, 1006, 'CON');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'UNGUREANU',  
'DAVID',to_date('08-03-2016', 'dd-mm-yyyy') , 6200, NULL, 1007, 'VET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'PANTUCU', 'DAN',  
to_date('03-05-2012', 'dd-mm-yyyy') , 2700, 0.06, 1007, 'APR');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'NICOLAE',  
'ALINA', to_date('09-07-2013', 'dd-mm-yyyy') , 2900, 0.02, 1007, 'ADM');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'STOICA', 'RADU',  
to_date('26-05-2012', 'dd-mm-yyyy') , 3900, NULL, 1007, 'CON');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'IVANA', 'FLORIN',  
to_date('20-07-2009', 'dd-mm-yyyy') , 2600, 0.1, 1007, 'ING');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'RATA',  
'ANDREEA', to_date('18-08-2008', 'dd-mm-yyyy') , 3000, NULL, 1008, 'ADM');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'GHEORGHE',  
'COSTEL', to_date('14-01-2011', 'dd-mm-yyyy') , 1900, NULL, 1008, 'APR');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'GRECU', 'DAVID',  
to_date('25-12-2009', 'dd-mm-yyyy') , 3400, 0.10, 1008, 'CON');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'SACEANU',  
'CLAUDIU',to_date('02-12-2010', 'dd-mm-yyyy') , 6000, NULL, 1008, 'VET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MIHAI',  
'CLAUDIU', to_date('14-09-2010', 'dd-mm-yyyy') , 4400, NULL, 1009, 'CON');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'ALEXE',  
'RAZVAN',to_date('18-10-2010', 'dd-mm-yyyy') , 2100, NULL, 1009, 'APR');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'BARBUIA', 'LUCA',  
to_date('15-07-2011', 'dd-mm-yyyy') , 2900, 0.05, 1009, 'ADM');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MATEI', 'SORIN',  
to_date('11-09-2012', 'dd-mm-yyyy') , 6600, NULL, 1009, 'VET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MIHALACHE',  
'DAN', to_date('04-07-2014', 'dd-mm-yyyy') , 1500, 0.2, 1009, 'HIN');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'POPESCU',  
'COSMIN', to_date('20-11-2013', 'dd-mm-yyyy') , 6600, NULL, 1010, 'VET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'SERBAN',  
'OVIDIU', to_date('07-04-2009', 'dd-mm-yyyy') , 2700, NULL, 1010, 'ADM');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MIREI',  
'ANDRADA',to_date('11-06-2009', 'dd-mm-yyyy') , 3300, NULL, 1010, 'CON');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'NETOIU',  
'ALEXANDRU', to_date('20-10-2008', 'dd-mm-yyyy') , 1500, 0.10, 1010, 'HIN');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'SACEANU',  
'ALEXANDRU', to_date('09-05-2009', 'dd-mm-yyyy') , 2600, NULL, 1011, 'ING');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MEREU',  
'ALEXANDRU',to_date('01-07-2013', 'dd-mm-yyyy') , 1900, 0.04, 1011, 'APR');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'MITRU',  
'ION',to_date('18-08-2010', 'dd-mm-yyyy') , 5500, NULL, 1011, 'VET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'GHEORGHE',  
'MARIUS', to_date('14-12-2011', 'dd-mm-yyyy') , 2650, NULL, 1011, 'ADM');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'GHERGHU', 'IOAN',  
to_date('25-05-2010', 'dd-mm-yyyy'), 3600, 0.02, 1012, 'ASTVET');
```

```
INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'STANCESCU',  
'TONUT', to_date('11-11-2013', 'dd-mm-yyyy') , 2650, 0.05, 1012, 'ING');
```

INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'JURUBITA',
'ROXANA',to_date('24-12-2010', 'dd-mm-yyyy') , 1700, 0.08, 1012, 'HIN');

INSERT INTO ANGAJAT VALUES (SEQ_ANG.NEXTVAL, 'ZIDARU',
'RAZVAN',to_date('17-06-2006', 'dd-mm-yyyy') , 3100, NULL, 1012, 'ADM');

	ID_ANGAJAT	NUME_ANGAJAT	PRENUME_ANGAJAT	DATA_ANGAJARII	SALARIU	COMISION	ID_ADAPOST	ID_PC
1	1000	CHIRITA	ANDREI	19-OCT-08	2900	0.15	1001	CON
2	1001	MATEESCU	VALENTIN	27-SEP-13	5700	0.1	1001	VET
3	1002	POPESCU	ANDREI	07-AUG-10	2000	0.08	1001	APR
4	1003	POPESCU	ANDREEA	03-SEP-15	2800	0.05	1001	ADM
5	1004	IONASCU	DANIEL	21-MAR-11	3100	(null)	1001	ING
6	1005	ILEA	ADELINA	26-FEB-14	3500	(null)	1002	CON
7	1006	CALIN	BOGDAN	14-NOV-08	5900	0.15	1002	VET
8	1007	PEICU	LIVIU	11-MAY-11	2600	0.1	1002	ADM
9	1008	RUSEANU	OVIDIU	17-APR-09	2500	(null)	1002	APR
10	1009	CALIN	SILVANA	29-MAR-12	3200	(null)	1002	ING
11	1010	SOREATA	ADRIAN	20-AUG-09	3900	(null)	1003	CON
12	1011	DICU	ALEXANDRU	11-JAN-10	6400	(null)	1003	VET
13	1012	MACOVEI	RAZVAN	19-APR-12	3300	0.05	1003	ADM
14	1013	RADUCU	MIHNEA	01-MAY-09	2200	(null)	1003	APR
15	1014	RIZESCU	ADRIAN	30-JUL-09	2600	(null)	1003	HIN
16	1015	MARINESCU	COSMIN	18-OCT-10	3000	0.05	1004	CON
17	1016	VOICULESCU	CATALIN	16-AUG-15	7500	(null)	1004	VET
18	1017	BARBU	COSMIN	15-JAN-13	2600	0.15	1004	ADM
19	1018	MACOVEI	BOGDAN	08-SEP-15	2000	(null)	1004	APR
20	1019	VOICA	GEORGIANA	29-JUL-06	2950	(null)	1004	ING
21	1020	MONETE	RAZVAN	02-JAN-16	4000	(null)	1005	CON
22	1021	BUDURE	DANIEL	23-OCT-14	6500	(null)	1005	VET
23	1022	MAZILU	VIOREL	27-JUN-11	2600	(null)	1005	ADM
24	1023	DONCEA	BEATRICE	17-JUN-13	1900	0.1	1005	APR
25	1024	ENE	ANDREI	10-SEP-10	5500	0.05	1006	VET
26	1025	IACOB	RAZVAN	02-SEP-12	2600	(null)	1006	ADM
27	1026	UNGUREANU	MARIUS	17-APR-15	2600	(null)	1006	APR
28	1027	OLTEANU	STEFAN	15-JUL-09	3000	(null)	1006	CON
29	1028	UNGUREANU	DAVID	08-MAR-16	6200	(null)	1007	VET
30	1029	PANTUCU	DAN	03-MAY-12	2700	0.06	1007	APR
31	1030	NICOLAE	ALINA	09-JUL-13	2900	0.02	1007	ADM
32	1031	STOICA	RADU	26-MAY-12	3900	(null)	1007	CON
33	1032	IVANA	FLORIN	20-JUL-09	2600	0.1	1007	ING
34	1033	RATA	ANDREEA	18-AUG-08	3000	(null)	1008	ADM
35	1034	GHEORGHE	COSTEL	14-JAN-11	1900	(null)	1008	APR
36	1035	GRECU	DAVID	25-DEC-09	3400	0.1	1008	CON
37	1036	SACEANU	CLAUDIU	02-DEC-10	6000	(null)	1008	VET
38	1037	MIHAI	CLAUDIU	14-SEP-10	4400	(null)	1009	CON
39	1038	ALEXE	RAZVAN	18-OCT-10	2100	(null)	1009	APR
40	1039	BARBUIA	LUCA	15-JUL-11	2900	0.05	1009	ADM
41	1040	MATEI	SORIN	11-SEP-12	6600	(null)	1009	VET

42	1041	MIHALACHE	DAN	04-JUL-14	1500	0.2	1009 HIN
43	1042	POPESCU	COSMIN	20-NOV-13	6600	(null)	1010 VET
44	1043	SERBAN	OVIDIU	07-APR-09	2700	(null)	1010 ADM
45	1044	MIREI	ANDRADA	11-JUN-09	3300	(null)	1010 CON
46	1045	NETOIU	ALEXANDRU	20-OCT-08	1500	0.1	1010 HIN
47	1046	SACEANU	ALEXANDRU	09-MAY-09	2600	(null)	1011 ING
48	1047	MEREU	ALEXANDRU	01-JUL-13	1900	0.04	1011 APR
49	1048	MITRU	ION	18-AUG-10	5500	(null)	1011 VET
50	1049	GHEORGHE	MARIUS	14-DEC-11	2650	(null)	1011 ADM
51	1050	GHERGHU	IOAN	25-MAY-10	3600	0.02	1012 ASTVET
52	1051	STANCESCU	IONUT	11-NOV-13	2650	0.05	1012 ING
53	1052	JURUBITA	ROXANA	24-DEC-10	1700	0.08	1012 HIN
54	1053	ZIDARU	RAZVAN	17-JUN-06	3100	(null)	1012 ADM

Inserare in tabelul **ADOPTANTI**:

create sequence seq_adoptanti

increment by 1

start with 300

maxvalue 380

nocycle;

INSERT INTO ADOPTANTI VALUES (seq_adoptanti.nextval,
'DRAGANESCU','STEFAN','0741526376');

INSERT INTO ADOPTANTI VALUES
(seq_adoptanti.nextval,'TEACA','MARIA','0748592625');

INSERT INTO ADOPTANTI VALUES
(seq_adoptanti.nextval,'STAN','STEFAN','0741256365');

INSERT INTO ADOPTANTI VALUES
(seq_adoptanti.nextval,'LAUFER','ANDREEA','0722548789');

INSERT INTO ADOPTANTI VALUES
(seq_adoptanti.nextval,'VLAD','FILIP','0789562314');

INSERT INTO ADOPTANTI VALUES
(seq_adoptanti.nextval,'CAZACU','IONUT','0752632415');

INSERT INTO ADOPTANTI VALUES
(seq_adoptanti.nextval,'ROSU','DAIANA','0756454533');

INSERT INTO ADOPTANTI VALUES
(seq_adoptanti.nextval,'IVAN','EMILIA','0773162241');

```
INSERT INTO ADOPTANTI VALUES  
(seq_adoptanti.nextval,'CRACIUNESCU','ANDREI','0773654651');  
  
INSERT INTO ADOPTANTI VALUES  
(seq_adoptanti.nextval,'FRUNZA','OVIDIU','0731411841');  
  
INSERT INTO ADOPTANTI VALUES  
(seq_adoptanti.nextval,'VARZARU','ALEXANDRU','0733624185');  
  
INSERT INTO ADOPTANTI VALUES  
(seq_adoptanti.nextval,'OLTEANU','RARES','0728422216');  
  
INSERT INTO ADOPTANTI VALUES  
(seq_adoptanti.nextval,'TAIFAS','MIHAI','0781531568');  
  
INSERT INTO ADOPTANTI VALUES  
(seq_adoptanti.nextval,'VOICU','DENISA','0757527852');  
  
INSERT INTO ADOPTANTI VALUES  
(seq_adoptanti.nextval,'ROTARU','ROBERT','0724845132');
```

	ID_ADOPTANT	NUME	PRENUME	TELEFON
1	300	DRAGANESCU	STEFAN	0741526376
2	301	TEACA	MARIA	0748592625
3	302	STAN	STEFAN	0741256365
4	303	LAUFER	ANDREEA	0722548789
5	304	VLAD	FILIP	0789562314
6	305	CAZACU	IONUT	0752632415
7	306	ROSU	DAIANA	0756454533
8	307	IVAN	EMILIA	0773162241
9	308	CRACIUNESCU	ANDREI	0773654651
10	309	FRUNZA	OVIDIU	0731411841
11	310	VARZARU	ALEXANDRU	0733624185
12	311	OLTEANU	RARES	0728422216
13	312	TAIFAS	MIHAI	0781531568
14	313	VOICU	DENISA	0757527852
15	314	ROTARU	ROBERT	0724845132

MODIFICAM IN TABELA ANIMAL ADOPTANTII:

```
UPDATE ANIMAL SET id_adoptant = 300 where id_animal = 1;  
UPDATE ANIMAL SET id_adoptant = 300 where id_animal = 3;  
UPDATE ANIMAL SET id_adoptant = 301 where id_animal = 5;  
UPDATE ANIMAL SET id_adoptant = 302 where id_animal = 9;
```

```
UPDATE ANIMAL SET id_adoptant = 303 where id_animal = 15;
UPDATE ANIMAL SET id_adoptant = 304 where id_animal = 21;
UPDATE ANIMAL SET id_adoptant = 305 where id_animal = 26;
UPDATE ANIMAL SET id_adoptant = 305 where id_animal = 27;
UPDATE ANIMAL SET id_adoptant = 306 where id_animal = 31;
UPDATE ANIMAL SET id_adoptant = 307 where id_animal = 34;
UPDATE ANIMAL SET id_adoptant = 308 where id_animal = 37;
UPDATE ANIMAL SET id_adoptant = 309 where id_animal = 41;
UPDATE ANIMAL SET id_adoptant = 310 where id_animal = 42;
UPDATE ANIMAL SET id_adoptant = 311 where id_animal = 18;
UPDATE ANIMAL SET id_adoptant = 312 where id_animal = 22;
UPDATE ANIMAL SET id_adoptant = 313 where id_animal = 11;
UPDATE ANIMAL SET id_adoptant = 314 where id_animal = 8;
UPDATE ANIMAL SET id_adoptant = 300 where id_animal = 6;
UPDATE ANIMAL SET id_adoptant = 305 where id_animal = 32;
UPDATE ANIMAL SET id_adoptant = 305 where id_animal = 13;
```


ID_ANIMAL	DATA_INMATRICULARE	SPECIE_ANIMAL	GREUTATE	INALTIME	VARSTA	SEX	ID_ADAPOST	ID_ADOPTANT
1	130-JUN-15	CAINE	5.8	0.54	2M		1001	300
2	212-NOV-17	PISICA	3.2	0.44	1M		1001	(null)
3	308-MAY-19	CAINE	9.3	1.2	3M		1001	300
4	423-JAN-20	PISICA	2.6	0.31	2F		1001	(null)
5	515-AUG-14	PISICA	3.7	0.48	6F		1001	301
6	611-NOV-11	CAINE	4.9	0.51	8F		1002	300
7	701-JUN-12	PISICA	1.9	0.25	4M		1002	(null)
8	801-SEP-08	CAINE	4.8	0.43	4F		1002	314
9	914-MAY-11	PISICA	4.3	0.39	7M		1003	302
10	1026-FEB-12	CAINE	6.9	0.57	1M		1003	(null)
11	1123-JUN-07	CAINE	9.4	0.98	4M		1003	313
12	1224-MAY-15	CAINE	10	1.3	3F		1003	(null)
13	1311-APR-21	PISICA	9.4	0.49	3F		1004	305
14	1404-FEB-21	CAINE	8.5	0.83	4M		1004	(null)
15	1519-MAR-21	CAINE	5.7	0.37	5F		1004	303
16	1618-JAN-21	PISICA	6.7	0.37	1F		1005	(null)
17	1725-FEB-21	CAINE	6.7	0.64	1M		1005	(null)
18	1806-JUN-21	PISICA	3.8	0.24	1F		1006	311
19	1930-MAY-21	CAINE	5.2	0.34	1F		1006	(null)
20	2019-MAY-21	PISICA	5.8	0.32	1F		1006	(null)
21	2101-NOV-20	CAINE	7.8	0.71	2M		1007	304
22	2218-MAR-21	PISICA	7.2	0.28	1F		1007	312
23	2317-FEB-21	CAINE	5.2	0.4	1F		1007	(null)
24	2409-MAR-21	PISICA	9.4	0.45	6F		1008	(null)
25	2528-JAN-21	CAINE	7.4	0.66	4M		1008	(null)
26	2613-JUN-21	PISICA	3.3	0.27	3M		1008	305
27	2709-DEC-20	CAINE	11	1.3	3M		1008	305
28	2811-APR-21	PISICA	4.9	0.39	7F		1009	(null)
29	2906-FEB-21	CAINE	6.8	0.63	9F		1009	(null)
30	3027-DEC-20	PISICA	5.9	0.4	8F		1009	(null)
31	3123-OCT-19	CAINE	4.5	0.32	3M		1010	306
32	3215-MAR-18	CAINE	3.6	0.29	4M		1010	305
33	3329-APR-20	CAINE	2.4	0.26	2F		1010	(null)
34	3421-SEP-20	PISICA	5.9	0.34	2M		1010	307
35	3504-APR-21	PISICA	3.8	0.28	1F		1011	(null)
36	3611-JUN-20	CAINE	8.9	0.52	4M		1011	(null)
37	3709-MAR-17	PISICA	4.2	0.31	6F		1011	308
38	3829-JUN-20	CAINE	9.3	0.89	6F		1012	(null)
39	3919-JUL-20	PISICA	5.6	0.42	4F		1012	(null)
40	4023-NOV-21	CAINE	10	1.1	9F		1012	(null)
41	4114-JAN-21	CAINE	4.5	0.37	2F		1012	309
42	4221-MAY-18	PISICA	3.6	0.26	1F		1012	310

Inserare in tabelul **SPONSORIZARE**:

INSERT INTO SPONSORIZARE VALUES (1, 100, 1001, 25500);

INSERT INTO SPONSORIZARE VALUES (2, 101, 1009, 44000);

INSERT INTO SPONSORIZARE VALUES (3, 102, 1007, 20000);

```
INSERT INTO SPONSORIZARE VALUES (4, 103, 1004, 71000);
INSERT INTO SPONSORIZARE VALUES (5, 110, 1001, 20000);
INSERT INTO SPONSORIZARE VALUES (6, 106, 1006, 54000);
INSERT INTO SPONSORIZARE VALUES (7, 107, 1006, 49000);
INSERT INTO SPONSORIZARE VALUES (8, 108, 1004, 12000);
INSERT INTO SPONSORIZARE VALUES (9, 109, 1006, 40000);
INSERT INTO SPONSORIZARE VALUES (10, 105, 1001, 13400);
INSERT INTO SPONSORIZARE VALUES (11, 111, 1010, 10000);
INSERT INTO SPONSORIZARE VALUES (12, 112, 1011, 50000);
```

	ID_SPONSORIZARE	ID_ASOCIATIE	ID_ADAPOST	SUMA_DONATA
1	1	100	1001	25500
2	2	101	1009	44000
3	3	102	1007	20000
4	4	103	1004	71000
5	5	110	1001	20000
6	6	106	1006	54000
7	7	107	1006	49000
8	8	108	1004	12000
9	9	109	1006	40000
10	10	105	1001	13400
11	11	111	1010	10000
12	12	112	1011	50000

Inserare in tabelul **VIZITE**:

```
create sequence seq_vizite
```

```
increment by 1
```

```
start with 800
```

```
maxvalue 1200
```

```
nocycle;
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('17/03/2020','dd/mm/yyyy'), 200, 1001);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('16/03/2021','dd/mm/yyyy'), 208, 1001);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('07/02/2021','dd/mm/yyyy'), 207, 1017);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('25/04/2021','dd/mm/yyyy'), 203, 1016);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('15/03/2021','dd/mm/yyyy'), 201, 1005);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('23/12/2020','dd/mm/yyyy'), 210, 1002);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('27/06/2019','dd/mm/yyyy'), 209, 1003);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('25/04/2018','dd/mm/yyyy'), 222, 1007);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('20/02/2020','dd/mm/yyyy'), 221, 1009);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('11/01/2021','dd/mm/yyyy'), 220, 1009);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('27/03/2019','dd/mm/yyyy'), 218, 1008);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL,  
to_date('31/01/2017','dd/mm/yyyy'), 219, 1011);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL, to_date('12-02-  
2021','dd-mm-yyyy'), 217, 1001);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL, to_date('17-05-  
2021','dd-mm-yyyy'), 214, 1002);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL, to_date('11-02-  
2018','dd-mm-yyyy'), 213, 1005);
```

```
INSERT INTO VIZITE VALUES(SEQ_VIZITE.NEXTVAL, to_date('15-02-  
2020','dd-mm-yyyy'), 203, 1002);
```

	ID_V...	DATA_VIZITEI	ID_VIZITATOR	ID_ADAPOST
1	800	17-MAR-20	200	1001
2	801	16-MAR-21	208	1001
3	804	15-MAR-21	201	1005
4	805	23-DEC-20	210	1002
5	806	27-JUN-19	209	1003
6	807	25-APR-18	222	1007
7	808	20-FEB-20	221	1009
8	809	11-JAN-21	220	1009
9	810	27-MAR-19	218	1008
10	811	31-JAN-17	219	1011
11	812	12-FEB-21	217	1001
12	813	17-MAY-21	214	1002
13	814	11-FEB-18	213	1005
14	815	15-FEB-20	203	1002

Inserare in tabelul **CONSUM**:

create sequence seq_consum

increment by 1

start with 300

maxvalue 400

nocycle;

INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 1, 29);

INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 1, 30);

INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 2, 31);

INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 3, 29);

INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 6, 32);

INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 5, 33);

INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 8, 34);

INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 9, 35);

INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 9, 37);

INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 8, 36);

INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 4, 41);

```
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 7, 42);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 10, 43);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 11, 43);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 12, 44);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 16, 45);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 14, 41);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 15, 29);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 13, 37);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 18, 38);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 17, 39);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 19, 36);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 20, 33);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 19, 31);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 23, 40);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 21, 45);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 22, 43);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 20, 40);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 22, 32);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 23, 47);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 25, 35);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 24, 44);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 29, 39);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 28, 32);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 26, 29);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 27, 35);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 30, 30);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 30, 40);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 31, 37);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 32, 38);
```

```
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 33, 47);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 34, 49);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 35, 47);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 35, 36);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 36, 36);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 37, 31);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 38, 29);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 39, 42);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 40, 33);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 40, 31);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 41, 32);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 42, 46);
INSERT INTO CONSUM VALUES(SEQ_CONSUM.NEXTVAL, 34, 43);
```

	ID_CONSUM	ID_ANIMAL	ID_ALIMENT				
1	300	1	29				
2	301	1	30				
3	302	2	31				
4	303	3	29				
5	304	6	32				
6	305	5	33				
7	306	8	34				
8	307	9	35				
9	308	9	37				
10	309	8	36				
11	310	4	41				
12	311	7	42				
13	312	10	43				
14	313	11	43				
15	314	12	44				
16	315	16	45				
17	316	14	41				
18	317	15	29				
19	318	13	37				
20	319	18	38				
21	320	17	39				
22	321	19	36				
23	322	20	33				
24	323	19	31				
25	324	23	40				
26	325	21	45				
27	326	22	43				
28	327	20	40	42	342	35	47
29	328	22	32	43	343	35	36
30	329	23	47	44	344	36	36
31	330	25	35	45	345	37	31
32	331	24	44	46	346	38	29
33	332	29	39	47	347	39	42
34	333	28	32	48	348	40	33
35	334	26	29	49	349	40	31
36	335	27	35	50	350	41	32
37	336	30	30	51	351	42	46
38	337	30	40	52	360	34	43
39	338	31	37				
40	339	32	38				
41	340	33	47				

6) Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat care să utilizeze două tipuri de colecție studiate. Apelați subprogramul.

ENUNT: Sa se afiseze in ordine crescatoare dupa data inmatricularii, data inmatricularii, si specia tuturor animalelor dintr-un adăpost care se afla intr-un oras introdus de la tastatura.

CREATE OR REPLACE PROCEDURE cerinta6_1

(pt_6_id in locatie.oras%type) IS


```
FOR i IN my_vector.first..my_vector.last LOOP
    select *
    into my_table(i)
    from animal
    where id_animal = my_vector(i)
    order by id_animal;
END LOOP;

DBMS_OUTPUT.PUT_LINE('In adapostul ' || v_numa || ' sunt ' ||
my_table.count || ' animale.');
```

```
IF my_table.count <> 0 THEN
    FOR i IN my_table.first..my_table.last LOOP
        dbms_output.put_line(my_table(i).id_animal || ' ' ||
my_table(i).data_inmatriculare || ' ' || my_table(i).specie_animal);
    END LOOP;
ELSE
    dbms_output.put_line('In adapostul ' || v_numa || ' nu se afla niciun
animal!');
END IF;

ELSE
    DBMS_OUTPUT.PUT_LINE('In orasul ' || pt_6_id || ' nu exista niciun
adapost!');
END IF;
```

```
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        dbms_output.put_line('Nu exista date!');
```

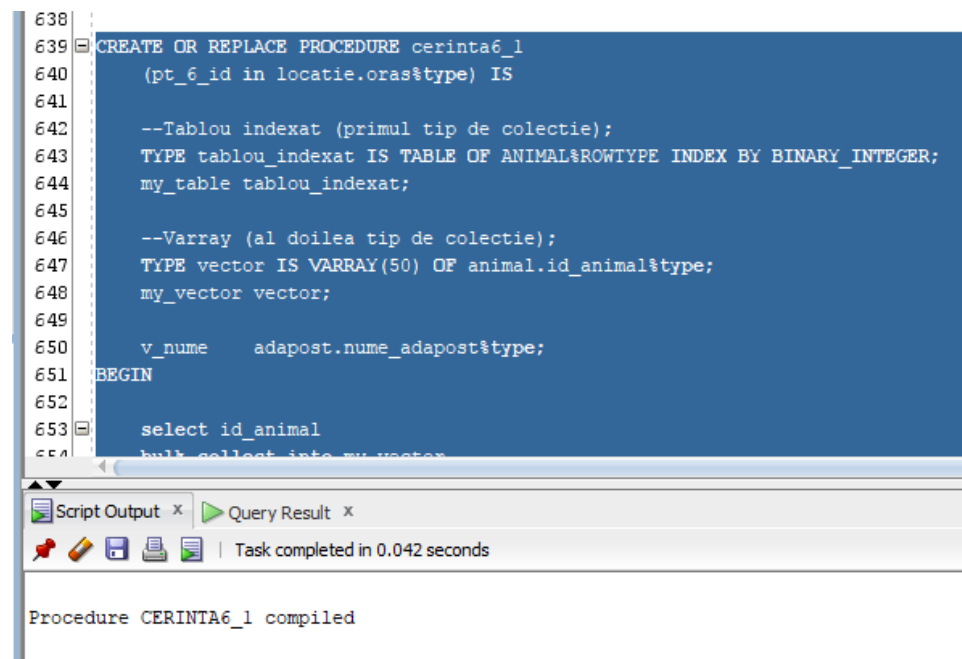
```
WHEN TOO_MANY_ROWS THEN

    dbms_output.put_line('Prea multe linii!');

WHEN OTHERS THEN

    dbms_output.put_line('Alta eroare!');

END;
```



The screenshot shows a SQL IDE window with a script editor on the left and a console on the right. The script editor contains the following SQL code:

```
638
639 CREATE OR REPLACE PROCEDURE cerinta6_1
640     (pt_6_id in locatie.oras%type) IS
641
642     --Tablou indexat (primul tip de colectie);
643     TYPE tablou_indexat IS TABLE OF ANIMAL%ROWTYPE INDEX BY BINARY_INTEGER;
644     my_table tablou_indexat;
645
646     --Varray (al doilea tip de colectie);
647     TYPE vector IS VARRAY(50) OF animal.id_animal%type;
648     my_vector vector;
649
650     v_nume     adapost.nume_adapost%type;
651 BEGIN
652
653     select id_animal
654     bulk collect into my_vector
```

The console on the right shows the following output:

```
Script Output x Query Result x
Task completed in 0.042 seconds

Procedure CERINTA6_1 compiled
```

--Apelarea procedurii:

accept oras prompt 'Introduceti numele orasului: ';

DECLARE

 v_id LOCATIE.oras%type := '&oras';

BEGIN

 cerinta6_1(v_id);

END;

Pentru orasul slatina avem:

```
In adapostul RAZA DE SPERANTA sunt 5 animale.  
5 15-AUG-14 PISICA  
1 30-JUN-15 CAINE  
2 12-NOV-17 PISICA  
3 08-MAY-19 CAINE  
4 23-JAN-20 PISICA
```

In caz ca nu exista niciun adapost se afiseaza:

```
In orasul asfaf nu exista niciun adapost!
```

7) Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat care să utilizeze un tip de cursor studiat. Apelați subprogramul.

ENUNT: Obțineti primii 3 cei mai bine platiti angajati (nume, prenume) din fiecare adapost (denumirea - se afiseaza o singura data), tratati exceptiile care pot aparea.

CREATE OR REPLACE PROCEDURE cerinta7 AS

CURSOR c IS

SELECT COUNT(aj.id_angajat) NR, a.nume_adapost ADAP

FROM angajat aj , adapost a

WHERE aj.id_adapost = a.id_adapost(+)

GROUP BY nume_adapost;

TYPE employees IS TABLE OF angajat%rowtype index by binary_integer;

t employees;

counter number := 0;

BEGIN

FOR i IN c LOOP

SELECT *

BULK COLLECT INTO t

FROM ANGAJAT

WHERE id_adapost = (SELECT id_adapost

```
        FROM adapost
        WHERE nume_adapost = i.ADAP)
ORDER BY SALARIU DESC;

IF i.nr = 0 THEN
    DBMS_OUTPUT.PUT_LINE('Adapostul ' || i.ADAP || ' are nu are niciun
angajat!');
ELSIF i.nr = 1 THEN
    DBMS_OUTPUT.PUT_LINE('Adapostul ' || i.ADAP || ' are ' || i.nr || '
angajat!');
ELSE
    DBMS_OUTPUT.PUT_LINE('Adapostul ' || i.ADAP || ' are ' || i.nr || ' angajati!'
);
END IF;

counter := 0;

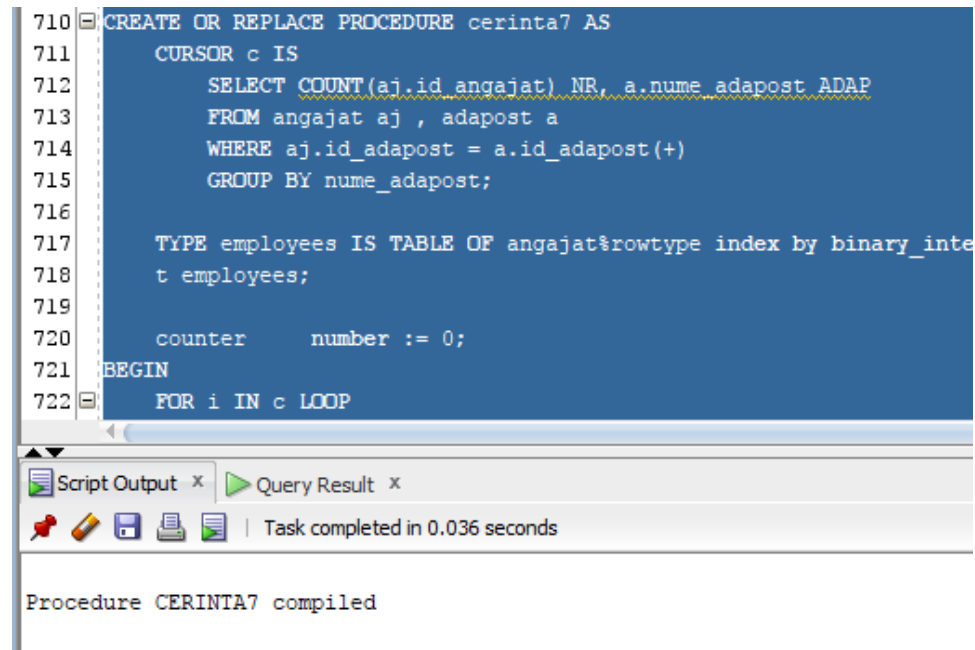
FOR x IN t.FIRST..t.LAST LOOP
    EXIT WHEN COUNTER > 2;

    DBMS_OUTPUT.PUT_LINE(t(x).nume_angajat || ' ' || t(x).prenume_angajat || '
are salariul de ' || t(x).salariu);

    counter := counter + 1;
END LOOP;

DBMS_OUTPUT.NEW_LINE();
END LOOP;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Nu exista date!');
    WHEN TOO_MANY_ROWS THEN
        DBMS_OUTPUT.PUT_LINE('Prea multe date!');
    WHEN OTHERS THEN
```

```
DBMS_OUTPUT.PUT_LINE('Alta eroare!');  
END;
```



```
710 CREATE OR REPLACE PROCEDURE cerinta7 AS  
711     CURSOR c IS  
712         SELECT COUNT(aj.id_angajat) NR, a.ume_adapost ADAP  
713         FROM angajat aj , adapost a  
714         WHERE aj.id_adapost = a.id_adapost(+)  
715         GROUP BY ume_adapost;  
716  
717     TYPE employees IS TABLE OF angajat%rowtype index by binary_int  
718     t employees;  
719  
720     counter    number := 0;  
721 BEGIN  
722     FOR i IN c LOOP
```

Script Output x Query Result x

Task completed in 0.036 seconds

Procedure CERINTA7 compiled

--Apelarea procedurii:

```
BEGIN  
    cerinta7;  
END;
```

Adapostul CAT LIFE are 4 angajati!
SACEANU CLAUDIU are salariul de 6000
GRECU DAVID are salariul de 3400
RATA ANDREEA are salariul de 3000

Adapostul CATEI ODAI are 4 angajati!
POPESCU COSMIN are salariul de 6600
MIREI ANDRADA are salariul de 3300
SERBAN OVIDIU are salariul de 2700

Adapostul DOGS FIRST are 5 angajati!
CALIN BOGDAN are salariul de 5900
ILEA ADELINA are salariul de 3500
CALIN SILVANA are salariul de 3200

Adapostul NEW LIFE are 4 angajati!
BUDURE DANIEL are salariul de 6500
MONETE RAZVAN are salariul de 4000
MAZILU VIOREL are salariul de 2600

Adapostul ANIMAL LIFE are 5 angajati!
DICU ALEXANDRU are salariul de 6400
SOREATA ADRIAN are salariul de 3900
MACOVEI RAZVAN are salariul de 3300

Adapostul STEAUA SPERANTEI are 4 angajati!
GHERGHU IOAN are salariul de 3600
ZIDARU RAZVAN are salariul de 3100
STANCESCU IONUT are salariul de 2650

Adapostul ANIMAL RESCUE are 5 angajati!
VOICULESCU CATALIN are salariul de 7500
MARINESCU COSMIN are salariul de 3000
VOICA GEORGIANA are salariul de 2950

Adapostul ASPA BUCURESTI are 4 angajati!
MITRU ION are salariul de 5500
GHEORGHE MARIUS are salariul de 2650
SACEANU ALEXANDRU are salariul de 2600

Adapostul RAZA DE SPERANTA are 5 angajati!
MATEESCU VALENTIN are salariul de 5700
IONASCU DANIEL are salariul de 3100
CHIRITA ANDREI are salariul de 2900

Adapostul OUR FRIENDS are 4 angajati!
ENE ANDREI are salariul de 5500
OLTEANU STEFAN are salariul de 3000
UNGUREANU MARIUS are salariul de 2600

Adapostul ZBURATORII are 5 angajati!
UNGUREANU DAVID are salariul de 6200
STOICA RADU are salariul de 3900
NICOLAE ALINA are salariul de 2900

Adapostul RED PANDA are 5 angajati!
MATEI SORIN are salariul de 6600
MIHAI CLAUDIU are salariul de 4400
BARBUIA LUCA are salariul de 2900

8) Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat de tip funcție care să utilizeze într-o singură comandă SQL 3 dintre tabelele definite. Tratați toate excepțiile care pot apărea. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

--8)

--Sa se afiseze numarul animalelor, cate animale din fiecare adapost si locatia adapostului de unde o persoana

--cu numele introdus de la tastatura a adoptat unul sau mai multe animale.

--Am creat un tip de colectie, si in ca o functie ajutatoare + metoda de apelare a
functiei CERINTA8.

```
CREATE OR REPLACE TYPE tablou_locatie IS TABLE OF VARCHAR2(200);
```

```
CREATE OR REPLACE FUNCTION ajutor_cerinta8
```

```
(v_nume adoptanti.nume%TYPE)
```

```
RETURN NUMBER IS
```

```
counter number;
```

```
BEGIN
```

```
select count(distinct id_adapost) --Functie care ma ajuta sa gasesc nr adaposturilor  
din care s-au adoptat animale,
```

```
into counter          --de catre o persoana.
```

```
from animal
```

```
where id_adoptant = (select id_adoptant
```

```
from adoptanti
```

```
where upper(nume) = upper(v_nume));
```

```
RETURN counter; --poate sa fie si 0, nu ne incurca.
```

```
END ajutor_cerinta8;
```

```
CREATE OR REPLACE FUNCTION cerinta8
```

```
(v_nume adoptanti.nume%TYPE DEFAULT 'Draganescu')
```

```
RETURN tablou_locatie IS
```

```
adresa tablou_locatie;
```

```
nr_adrese NUMBER := ajutor_cerinta8(v_nume);
```

```
BEGIN
```

```
IF nr_adrese = 0 THEN
```

```
DBMS_OUTPUT.PUT_LINE('Nu exista niciun adoptant cu numele: ' ||  
v_nume);
```

```
ELSE
```

```
SELECT ' din judetul ' || judet || ', orasul ' || oras
BULK COLLECT INTO adresa
FROM LOCATIE
WHERE ID_LOCATIE IN (SELECT locatie.ID_LOCATIE
FROM LOCATIE
INNER JOIN adapost on adapost.id_locatie = locatie.id_locatie
INNER JOIN animal on animal.id_adapost = adapost.id_adapost
INNER JOIN adoptanti on adoptanti.id_adoptant = animal.id_adoptant
WHERE animal.id_adoptant IS NOT NULL AND
UPPER(adoptanti.numa) = upper(v_numa))
ORDER BY id_locatie;
END IF;

RETURN adresa;
EXCEPTION
WHEN NO_DATA_FOUND THEN
RAISE_APPLICATION_ERROR(-20000, 'Nu exista un adoptant cu numele
specificat!');
WHEN OTHERS THEN
RAISE_APPLICATION_ERROR(-20001, 'Alta eroare!');
END cerinta8;

accept Nume prompt 'Introduceti numele adoptantului: ';
DECLARE
adresa tablou_locatie;
numa_ad varchar2(50) := '&Nume';
nr_adrese number;
nr_animale number;

cursor c is
```



```
select count(id_animal) nr
from animal
where id_adoptant = (select id_adoptant
                     from adoptanti
                     where upper(ume) = upper(ume_ad))
group by id_adapost
order by id_adapost;
```

BEGIN

```
select count(id_animal)
into nr_animale
from animal
where id_adoptant = (select id_adoptant
                     from adoptanti
                     where upper(ume) = upper(ume_ad));

adresa := cerinta8(ume_ad);
nr_adrese := ajutor_cerinta8(ume_ad);
DBMS_OUTPUT.PUT_LINE(ume_ad || ' a adoptat ' || nr_animale || ' animale. ');
--ma folosesc de nr animala ca un counter;
nr_animale := 1;
FOR i IN c LOOP
    DBMS_OUTPUT.PUT_LINE( i.nr || adresa(nr_animale));
    nr_animale := nr_animale+1;
END LOOP;
```

EXCEPTION

```
WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE('Code: ' || sqlcode || ' Error: ' || sqlerrm);
WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE('Code: ' || sqlcode || ' Error: ' || sqlerrm );
```

END;

Pentru Drăganescu:

```
drăganescu a adoptat 3 animale.  
2 din județul OLT, orașul SLATINA  
1 din județul BOTOSANI, orașul BOTOSANI
```

Pentru Cazacu:

```
Cazacu a adoptat 4 animale.  
1 din județul CLUJ, orașul TURDA  
2 din județul TIMIȘ, orașul LUGOJ  
1 din județul NEAMT, orașul PIATRA NEAMT
```

În caz de eroare:

```
Nu există niciun adoptant cu numele: sfasaf  
sfasaf a adoptat 0 animale.
```

9) Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat de tip procedură care să utilizeze într-o singură comandă SQL 5 dintre tabelele definite. Tratați toate excepțiile care pot apărea, incluzând excepțiile NO_DATA_FOUND și TOO_MANY_ROWS. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

--Cerinta 9

--Numele și prețul alimentelor pe care le consumă animalele dintr-un adăpost

--cu numele introdus de la tastatură și care este sponsorizat de minim o asociație.

CREATE OR REPLACE PROCEDURE cerinta_9

(v_nume_ad IN adăpost.nume_adăpost%TYPE DEFAULT 'RAZA DE SPERANTA')
IS

TYPE tablou_indexat IS TABLE OF consum%ROWTYPE INDEX BY
BINARY_INTEGER; --reținem detaliile despre o linie întreaga.

alimente tablou_indexat;

CURSOR c IS --selectez id-urile animalelor din adapostul sponsorizat de o asociatie

```
SELECT id_animal id_an  
FROM animal  
WHERE id_adapost = (SELECT id_adapost  
FROM adapost  
WHERE upper(ume_adapost) = upper(v_ume_ad));
```

TYPE tablou2 IS TABLE OF consum.id_consum%type INDEX BY
BINARY_INTEGER; --retin id-urile fiecarui consum facut de un animal
id_alimente tablou2;

counter number; --vefirc daca sunt animale intr-un adapost.

contor number := 1;

counter_eg_0 EXCEPTION; --ridic eroare pt adapostul gol

v_sponsorizare number := 0;

v_ume_alim aliment.ume_aliment%type;

v_pret aliment.pret%type;

BEGIN

```
SELECT COUNT(id_adapost)  
INTO counter  
FROM animal  
WHERE id_adapost = (SELECT id_adapost  
FROM adapost  
WHERE upper(ume_adapost) = upper(v_ume_ad));
```

SELECT count(id_asociatie)

INTO v_sponsorizare

```
FROM sponsorizare
WHERE id_adapost = (SELECT id_adapost
                    FROM adapost
                    WHERE upper(ume_adapost) = upper(v_ume_ad));

IF counter <> 0 THEN
    IF v_sponsorizare <> 0 THEN
        FOR i in c LOOP --pentru fiecare animal din adapost, retin cate feluri de
mancare consuma.
            SELECT id_consum
            BULK COLLECT INTO id_alimente
            FROM consum
            WHERE id_animal = i.id_an; --daca adapostul are animale, nu trebuie sa
verific daca id_alimente este gol pentru ca fiecare animal
                --consuma MINIM 1 aliment.

            FOR x IN id_alimente.first..id_alimente.last LOOP --apoi introduc in
alimente detaliile despre fiecare aliment consumat, la fel si aici
                SELECT *
                    -- nu trebuie sa verific.
                INTO alimente(contor)
                FROM consum
                WHERE id_consum in (select consum.id_consum
                                    FROM consum
                                    INNER JOIN aliment on consum.id_aliment =
aliment.id_aliment
                                    INNER JOIN animal on consum.id_animal =
animal.id_animal
                                    INNER JOIN adapost on adapost.id_adapost =
animal.id_adapost
                                    INNER JOIN sponsorizare on sponsorizare.id_adapost =
adapost.id_adapost
                                    INNER JOIN asociatie on asociatie.id_asociatie =
sponsorizare.id_asociatie
```

```
WHERE animal.id_animal = i.id_an and consum.id_consum  
= id_alimente(x));  
        contor := contor +1;  
    END LOOP;  
END LOOP;  
    DBMS_OUTPUT.PUT_LINE('Animalele din adapostul ' || v_numa_ad||  
consuma alimentele: ');  
    FOR i IN alimente.first..alimente.last LOOP  
        SELECT nume_aliment, pret  
        INTO v_numa_alim, v_pret  
        FROM aliment  
        WHERE id_aliment = alimente(i).id_aliment;  
  
        DBMS_OUTPUT.PUT_LINE('Animalul cu id-ul: ' || alimente(i).id_animal ||  
' consuma produsul: ' || v_numa_alim || ' care are pretul de: ' || v_pret);  
    END LOOP;  
ELSE  
    RAISE NO_DATA_FOUND;  
END IF;  
ELSIF counter = 0 THEN  
    RAISE counter_eg_0;  
END IF;  
EXCEPTION  
    WHEN counter_eg_0 THEN  
        DBMS_OUTPUT.PUT_LINE('Nu exista adapostul ' || v_numa_ad);  
    WHEN NO_DATA_FOUND THEN  
        DBMS_OUTPUT.PUT_LINE('Adapostul nu este sponsorizat!');  
    WHEN TOO_MANY_ROWS THEN  
        DBMS_OUTPUT.PUT_LINE('Too many rows!');  
END;
```

```
accept nume_ad prompt 'Nume adapost';
```

```
BEGIN
```

```
    cerinta_9('&nume_ad');
```

```
END;
```

```
Nu exista adapostul asfasf
```

Pentru DOGS FIRST

```
Adapostul nu este sponsorizat!
```

Pentru CATEI ODAI

```
Animalele din adapostul catei odai consuma alimentele:  
Animalul cu id-ul: 31 consuma produsul: Conserva-Pui care are pretul de: 23.59  
Animalul cu id-ul: 32 consuma produsul: Anti-Hairball care are pretul de: 24.99  
Animalul cu id-ul: 33 consuma produsul: Vitamin E-Xtra care are pretul de: 79.5  
Animalul cu id-ul: 34 consuma produsul: Indoor Pui care are pretul de: 20.5
```

10) Definiți un trigger de tip LMD la nivel de comandă. Declanșați trigger-ul.

--Cerinta 10.

--Sa se creeze un trigger care permite vizitarea unui adapost in intervalul 07:00-22:00
--(adica utilizarea comenzii insert in tabelul vizite).

```
CREATE OR REPLACE TRIGGER cerinta10
```

```
    BEFORE INSERT OR DELETE OR UPDATE ON vizite
```

```
BEGIN
```

```
    IF (TO_CHAR(SYSDATE, 'HH24') NOT BETWEEN 7 AND 22) THEN
```

```
        IF INSERTING THEN
```

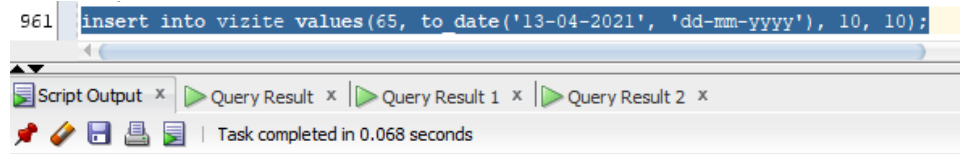
```
            RAISE_APPLICATION_ERROR(-20001, 'Vizitarea adapostului se realizeaza  
in intervalul 7-22');
```

```
        END IF;
```

```
    END IF;
```

```
END;
```

Am apelat la ora 02:43.



```
961 insert into vizite values(65, to_date('13-04-2021', 'dd-mm-yyyy'), 10, 10);
```

Script Output x Query Result x Query Result 1 x Query Result 2 x

Task completed in 0.068 seconds

Error starting at line : 961 in command -
insert into vizite values(65, to_date('13-04-2021', 'dd-mm-yyyy'), 10, 10)
Error report -
ORA-20001: Vizitarea adapostului se realizeaza in intervalul 7-22
ORA-06512: at "ENE.CERINTA10", line 4
ORA-04088: error during execution of trigger 'ENE.CERINTA10'

11) Definiți un trigger de tip LMD la nivel de linie. Declanșați trigger-ul.

--Cerinta 11;

--Creati un trigger care verifica daca in momentul introducerii sau modificarea unei linii din

--tabelul angajat ca data_angajarii sa fie mai mica decat data curenta si ca salariul sa fie in cerintele postului.

CREATE OR REPLACE TRIGGER cerinta11

BEFORE INSERT OR UPDATE ON ANGAJAT FOR EACH ROW

DECLARE

eroare1 EXCEPTION;

eroare2 EXCEPTION;

salariu_max posturi.salariu_maxim%type;

salariu_min posturi.salariu_minim%type;

BEGIN

select salariu_minim, salariu_maxim

into salariu_min, salariu_max

from posturi

where :NEW.id_post = id_post;

IF :NEW.data_angajarii > SYSDATE THEN

RAISE eroare1;

ELSIF :NEW.salariu > salariu_max OR :NEW.salariu < salariu_min THEN

```
        RAISE eroare2;

    END IF;

    IF inserting THEN

        DBMS_OUTPUT.PUT_LINE('Inserare reusita!');

    ELSIF updating THEN

        DBMS_OUTPUT.PUT_LINE('Modificare reusita!');

        DBMS_OUTPUT.PUT_LINE('Data veche: ' || :OLD.data_angajarii || 'Salariul
vechi: ' || :OLD.salariu);

    END IF;

EXCEPTION

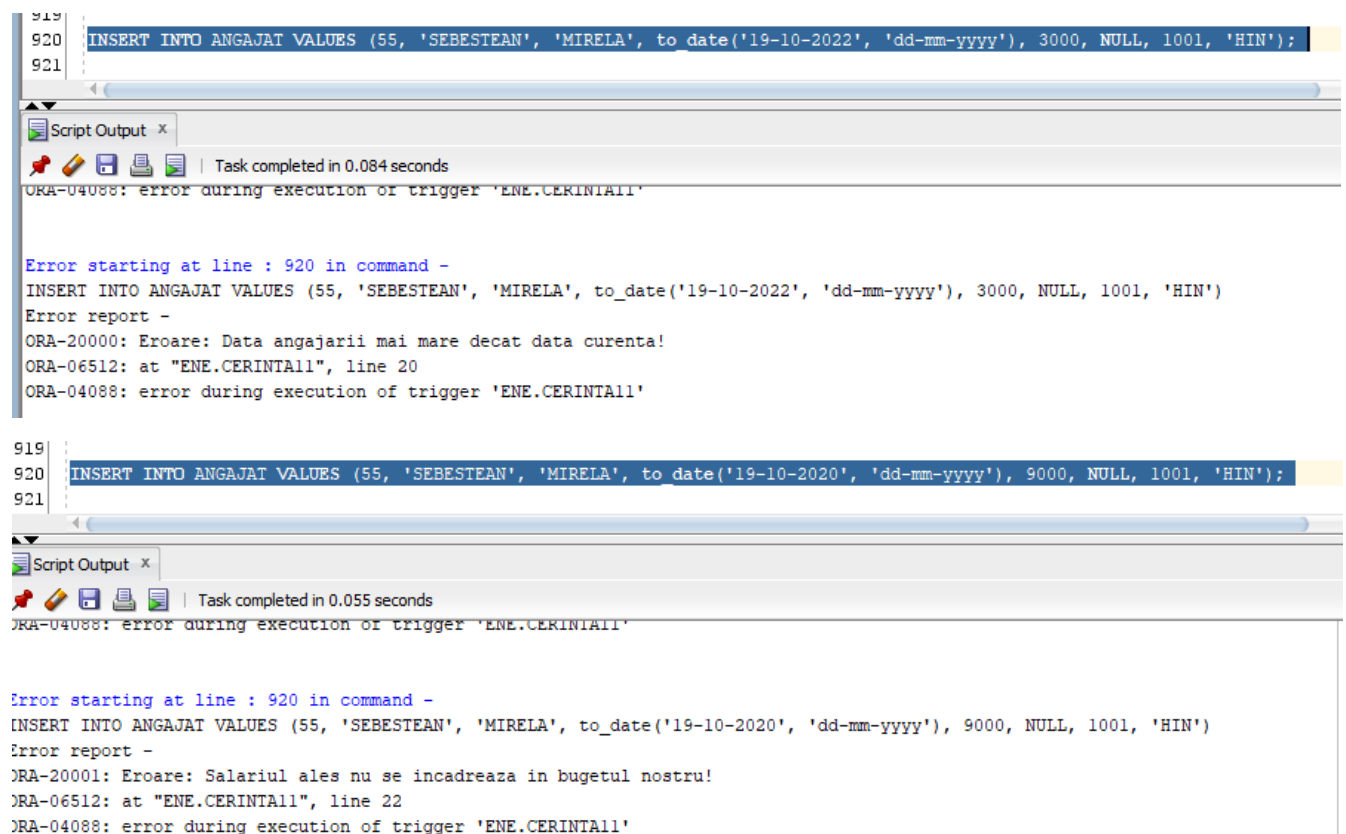
    WHEN eroare1 THEN

        RAISE_APPLICATION_ERROR(-20000, 'Eroare: Data angajarii mai mare
decat data curenta!');

    WHEN eroare2 THEN

        RAISE_APPLICATION_ERROR(-20001, 'Eroare: Salariul ales nu se incadreaza
in bugetul nostru!');

END;
```



```
919
920 INSERT INTO ANGAJAT VALUES (55, 'SEBESTEAN', 'MIRELA', to_date('19-10-2022', 'dd-mm-yyyy'), 3000, NULL, 1001, 'HIN');
921

Script Output x
Task completed in 0.084 seconds
ORA-04088: error during execution of trigger 'ENE.CERINTAI1'

Error starting at line : 920 in command -
INSERT INTO ANGAJAT VALUES (55, 'SEBESTEAN', 'MIRELA', to_date('19-10-2022', 'dd-mm-yyyy'), 3000, NULL, 1001, 'HIN')
Error report -
ORA-20000: Eroare: Data angajarii mai mare decat data curenta!
ORA-06512: at "ENE.CERINTAI1", line 20
ORA-04088: error during execution of trigger 'ENE.CERINTAI1'

919
920 INSERT INTO ANGAJAT VALUES (55, 'SEBESTEAN', 'MIRELA', to_date('19-10-2020', 'dd-mm-yyyy'), 9000, NULL, 1001, 'HIN');
921

Script Output x
Task completed in 0.055 seconds
ORA-04088: error during execution of trigger 'ENE.CERINTAI1'

Error starting at line : 920 in command -
INSERT INTO ANGAJAT VALUES (55, 'SEBESTEAN', 'MIRELA', to_date('19-10-2020', 'dd-mm-yyyy'), 9000, NULL, 1001, 'HIN')
Error report -
ORA-20001: Eroare: Salariul ales nu se incadreaza in bugetul nostru!
ORA-06512: at "ENE.CERINTAI1", line 22
ORA-04088: error during execution of trigger 'ENE.CERINTAI1'
```



```
925 INSERT INTO ANGAJAT VALUES (55, 'SEBESTEAN', 'MIRELA', to_date('19-10-2020', 'dd-mm-yyyy'), 1900, NULL, 1001, 'HIN');
926 rollback;
927
```

Script Output x

Task completed in 0.099 seconds

Dbms Output x

Buffer Size: 20000

ProiectSGBD x

Inserare reusita!
Inserare reusita!

12) Definiți un trigger de tip LDD. Declanșați trigger-ul.

--Cerinta 12

--Sa se creeze un trigger LDD care restrictioneaza stergerea unei tabele din schema noastra.

CREATE OR REPLACE TRIGGER cerinta12

BEFORE DROP ON SCHEMA

BEGIN

DBMS_OUTPUT.PUT_LINE('Nu aveti permisiunea de a sterge tabele!');

ROLLBACK;

END;

```
938 create table test(id number(2));
939 drop table test;
940
```

Script Output x

Task completed in 0.107 seconds

*ACTION: If the situation described in the next e
can be corrected, do so; otherwise conta

Dbms Output x

Buffer Size: 20000

ProiectSGBD x

Nu aveti permisiunea de a sterge tabele!

13) Definiți un pachet care să conțină toate obiectele definite în cadrul proiectului.

--Cerinta 13

CREATE OR REPLACE PACKAGE pack_ex13

IS

 PROCEDURE cerinta6_1 (pt_6_id in locatie.oras%type);

 PROCEDURE cerinta7;

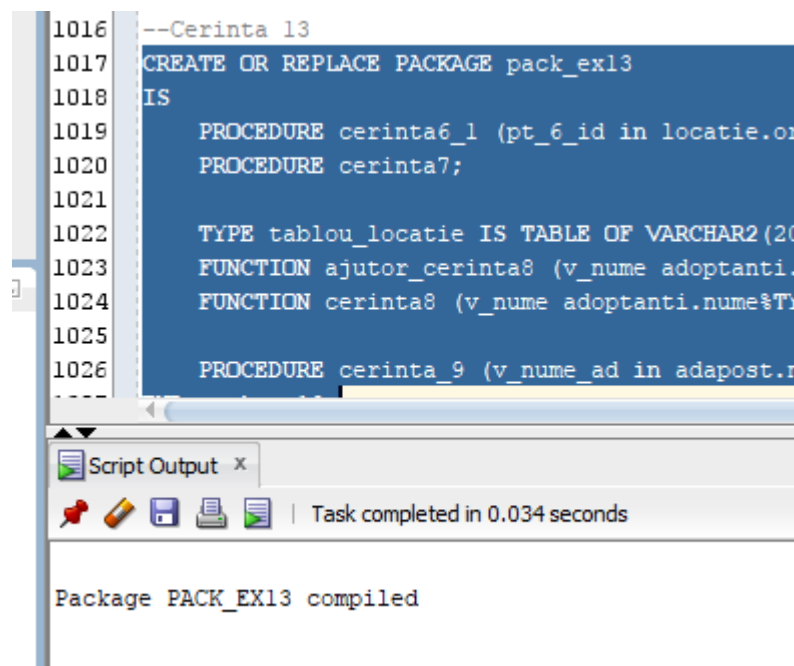
 TYPE tablou_locatie IS TABLE OF VARCHAR2(200);

 FUNCTION ajutor_cerinta8 (v_nume adoptanti.nume%TYPE) RETURN
NUMBER;

 FUNCTION cerinta8 (v_nume adoptanti.nume%TYPE DEFAULT 'Draganescu')
RETURN tablou_locatie;

 PROCEDURE cerinta_9 (v_nume_ad in adapost.nume_adapost%type DEFAULT
'RAZA DE SPERANTA');

END pack_ex13;



The screenshot shows a SQL IDE window with a script editor on the left and a 'Script Output' window on the right. The script editor contains the following SQL code:

```
1016 --Cerinta 13
1017 CREATE OR REPLACE PACKAGE pack_ex13
1018 IS
1019     PROCEDURE cerinta6_1 (pt_6_id in locatie.ora
1020     PROCEDURE cerinta7;
1021
1022     TYPE tablou_locatie IS TABLE OF VARCHAR2(20
1023     FUNCTION ajutor_cerinta8 (v_nume adoptanti.
1024     FUNCTION cerinta8 (v_nume adoptanti.nume%Ty
1025
1026     PROCEDURE cerinta_9 (v_nume_ad in adapost.n
```

The 'Script Output' window shows the message: 'Task completed in 0.034 seconds'. Below this, a status bar indicates 'Package PACK_EX13 compiled'.

CREATE OR REPLACE PACKAGE BODY pack_ex13

IS

 --Cerinta 6;

```
PROCEDURE cerinta6_1 (pt_6_id in locatie.oras%type) IS
    --Tablou indexat (primul tip de colectie);
    TYPE tablou_indexat IS TABLE OF ANIMAL%ROWTYPE INDEX BY
    BINARY_INTEGER;
    my_table tablou_indexat;

    --Varray (al doilea tip de colectie);
    TYPE vector IS VARRAY(50) OF animal.id_animal%type;
    my_vector vector;

    v_nume adapost.nume_adapost%type;
BEGIN

    select id_animal
    bulk collect into my_vector
    from ANIMAL
    join adapost on adapost.id_adapost = animal.id_adapost
    join locatie on locatie.id_locatie = adapost.id_locatie
    where upper(locatie.oras) = upper(pt_6_id)
    order by data_inmatriculare;

    IF my_vector.count <> 0 THEN
        select nume_adapost
        into v_nume
        from adapost
        where id_locatie = (select id_locatie
                           from locatie
                           where upper(oras) = upper(pt_6_id));

        FOR i IN my_vector.first..my_vector.last LOOP
```

```
        select *
        into my_table(i)
        from animal
        where id_animal = my_vector(i)
        order by id_animal;
    END LOOP;

    DBMS_OUTPUT.PUT_LINE('In adapostul ' || v_nume || ' sunt ' ||
my_table.count || ' animale.');
```

```
    IF my_table.count <> 0 THEN
        FOR i IN my_table.first..my_table.last LOOP
            dbms_output.put_line(my_table(i).id_animal || ' ' ||
my_table(i).data_inmatriculare || ' ' || my_table(i).specie_animal);
        END LOOP;
    ELSE
        dbms_output.put_line('In adapostul ' || v_nume || ' nu se afla niciun
animal!');
    END IF;
ELSE
    DBMS_OUTPUT.PUT_LINE('In orasul ' || pt_6_id || ' nu exista niciun
adapost!');
END IF;
```

```
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        dbms_output.put_line('Nu exista date!');
    WHEN TOO_MANY_ROWS THEN
        dbms_output.put_line('Prea multe linii!');
    WHEN OTHERS THEN
        dbms_output.put_line('Alta eroare!');
```

```
END cerinta6_1;
```

```
--Cerinta 7
```

```
PROCEDURE cerinta7 AS
```

```
    CURSOR c IS
```

```
        SELECT COUNT(aj.id_angajat) NR, a.nume_adapost ADAP
```

```
        FROM angajat aj , adapost a
```

```
        WHERE aj.id_adapost = a.id_adapost(+)
```

```
        GROUP BY nume_adapost;
```

```
TYPE employees IS TABLE OF angajat%rowtype index by binary_integer;
```

```
t employees;
```

```
counter    number := 0;
```

```
BEGIN
```

```
    FOR i IN c LOOP
```

```
        SELECT *
```

```
        BULK COLLECT INTO t
```

```
        FROM ANGAJAT
```

```
        WHERE id_adapost = (SELECT id_adapost
```

```
                            FROM adapost
```

```
                            WHERE nume_adapost = i.ADAP)
```

```
        ORDER BY SALARIU DESC;
```

```
    IF i.nr = 0 THEN
```

```
        DBMS_OUTPUT.PUT_LINE('Adapostul ' || i.ADAP || ' are nu are niciun  
angajat!');
```

```
    ELSIF i.nr = 1 THEN
```

```
        DBMS_OUTPUT.PUT_LINE('Adapostul ' || i.ADAP || ' are ' || i.nr || '  
angajat!');
```

```
ELSE
    DBMS_OUTPUT.PUT_LINE('Adapostul ' || i.ADAP || ' are ' || i.nr || '
angajati!');
END IF;

counter := 0;

FOR x IN t.FIRST..t.LAST LOOP
    EXIT WHEN COUNTER > 2;
    DBMS_OUTPUT.PUT_LINE(t(x).nume_angajat || ' ' ||
t(x).prenume_angajat || ' are salariul de ' || t(x).salariu);
    counter := counter + 1;
END LOOP;
DBMS_OUTPUT.NEW_LINE();
END LOOP;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Nu exista date!');
    WHEN TOO_MANY_ROWS THEN
        DBMS_OUTPUT.PUT_LINE('Prea multe date!');
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Alta eroare!');
END;

--Cerinta 8
FUNCTION ajutor_cerinta8
(v_nume adoptanti.nume%TYPE)
RETURN NUMBER IS
    counter number;
BEGIN
```

```
select count(distinct id_adapost) --Functie care ma ajuta sa gasesc nr  
adaposturilor din care s-au adoptat animale,
```

```
into counter          --de catre o persoana.
```

```
from animal
```

```
where id_adoptant = (select id_adoptant
```

```
                    from adoptanti
```

```
                    where upper(ume) = upper(v_ume));
```

```
RETURN counter; --poate sa fie si 0, nu ne incurca.
```

```
END ajutor_cerinta8;
```

```
FUNCTION cerinta8
```

```
(v_ume adoptanti.ume%TYPE DEFAULT 'Draganescu')
```

```
RETURN tablou_locatie IS
```

```
adresa tablou_locatie;
```

```
nr_adrese NUMBER := ajutor_cerinta8(v_ume);
```

```
BEGIN
```

```
IF nr_adrese = 0 THEN
```

```
    DBMS_OUTPUT.PUT_LINE('Nu exista niciun adoptant cu numele: ' ||  
v_ume);
```

```
ELSE
```

```
    SELECT ' din judetul ' || judet || ', orasul ' || oras
```

```
    BULK COLLECT INTO adresa
```

```
    FROM LOCATIE
```

```
    WHERE ID_LOCATIE IN (SELECT locatie.ID_LOCATIE
```

```
                        FROM LOCATIE
```

```
                        INNER JOIN adapost on adapost.id_locatie = locatie.id_locatie
```

```
                        INNER JOIN animal on animal.id_adapost = adapost.id_adapost
```

```
                        INNER JOIN adoptanti on adoptanti.id_adoptant = animal.id_adoptant
```

```
WHERE animal.id_adoptant IS NOT NULL AND  
UPPER(adoptanti.numa) = upper(v_nume))
```

```
ORDER BY id_locatie;
```

```
END IF;
```

```
RETURN adresa;
```

```
EXCEPTION
```

```
WHEN NO_DATA_FOUND THEN
```

```
RAISE_APPLICATION_ERROR(-20000, 'Nu exista un adoptant cu numele  
specificat!');
```

```
WHEN OTHERS THEN
```

```
RAISE_APPLICATION_ERROR(-20001, 'Alta eroare!');
```

```
END cerinta8;
```

```
--Cerinta 9
```

```
PROCEDURE cerinta_9
```

```
(v_nume_ad in adapost.nume_adapost%type DEFAULT 'RAZA DE  
SPERANTA') IS
```

```
TYPE tablou_indexat IS TABLE OF consum%rowtype INDEX BY  
BINARY_INTEGER; --retinem detaliile despre o linie intreaga.
```

```
alimente tablou_indexat;
```

```
CURSOR c IS --selectez id-urile animalelor din adapostul sponsorizat de o  
asociatie
```

```
SELECT id_animal id_an
```

```
FROM animal
```

```
WHERE id_adapost = (SELECT id_adapost
```

```
FROM adapost
```

```
WHERE upper(nume_adapost) = upper(v_nume_ad));
```



```
TYPE tablou2 IS TABLE OF consum.id_consum%type INDEX BY  
BINARY_INTEGER; --retin id-urile fiecarui consum facut de un animal
```

```
id_alimente tablou2;
```

```
counter number; --vefirc daca sunt animale intr-un adapost.
```

```
contor number := 1;
```

```
counter_eg_0 EXCEPTION; --ridic eroare pt adapostul gol
```

```
v_sponsorizare number := 0;
```

```
v_nume_alim aliment.nume_aliment%type;
```

```
v_pret aliment.pret%type;
```

```
BEGIN
```

```
SELECT COUNT(id_adapost)
```

```
INTO counter
```

```
FROM animal
```

```
WHERE id_adapost = (SELECT id_adapost
```

```
FROM adapost
```

```
WHERE upper(nume_adapost) = upper(v_nume_ad));
```

```
SELECT count(id_asociatie)
```

```
INTO v_sponsorizare
```

```
FROM sponsorizare
```

```
WHERE id_adapost = (SELECT id_adapost
```

```
FROM adapost
```

```
WHERE upper(nume_adapost) = upper(v_nume_ad));
```

```
IF counter <> 0 THEN
```

```
IF v_sponsorizare <> 0 THEN
```

FOR i in c LOOP --pentru fiecare animal din adapost, retin cate feluri de
mancare consuma.

SELECT id_consum

BULK COLLECT INTO id_alimente

FROM consum

WHERE id_animal = i.id_an; --daca adapostul are animale, nu trebuie sa
verific daca id_alimente este gol pentru ca fiecare animal

--consuma MINIM 1 aliment.

FOR x IN id_alimente.first..id_alimente.last LOOP --apoi introduc in
alimente detaliile despre fiecare aliment consumat, la fel si aici

SELECT * -- nu trebuie sa verific.

INTO alimente(contor)

FROM consum

WHERE id_consum in (select consum.id_consum

FROM consum

INNER JOIN aliment on consum.id_aliment =
aliment.id_aliment

INNER JOIN animal on consum.id_animal =
animal.id_animal

INNER JOIN adapost on adapost.id_adapost =
animal.id_adapost

INNER JOIN sponsorizare on sponsorizare.id_adapost =
adapost.id_adapost

INNER JOIN asociatie on asociatie.id_asociatie =
sponsorizare.id_asociatie

WHERE animal.id_animal = i.id_an and
consum.id_consum = id_alimente(x));

contor := contor +1;

END LOOP;

END LOOP;

DBMS_OUTPUT.PUT_LINE('Animalele din adapostul ' || v_nume_ad||'
consuma alimentele: ');

FOR i IN alimente.first..alimente.last LOOP

```
        SELECT nume_aliment, pret
        INTO v_numa_alim, v_pret
        FROM aliment
        WHERE id_aliment = alimente(i).id_aliment;

        DBMS_OUTPUT.PUT_LINE('Animalul cu id-ul: ' ||
alimente(i).id_animal || ' consuma produsul: ' || v_numa_alim || ' care are pretul de: ' ||
v_pret);

        END LOOP;

    ELSE

        RAISE NO_DATA_FOUND;

    END IF;

    ELSIF counter = 0 THEN

        RAISE counter_eg_0;

    END IF;

EXCEPTION

    WHEN counter_eg_0 THEN

        DBMS_OUTPUT.PUT_LINE('Nu exista adapostul ' || v_numa_ad);

    WHEN NO_DATA_FOUND THEN

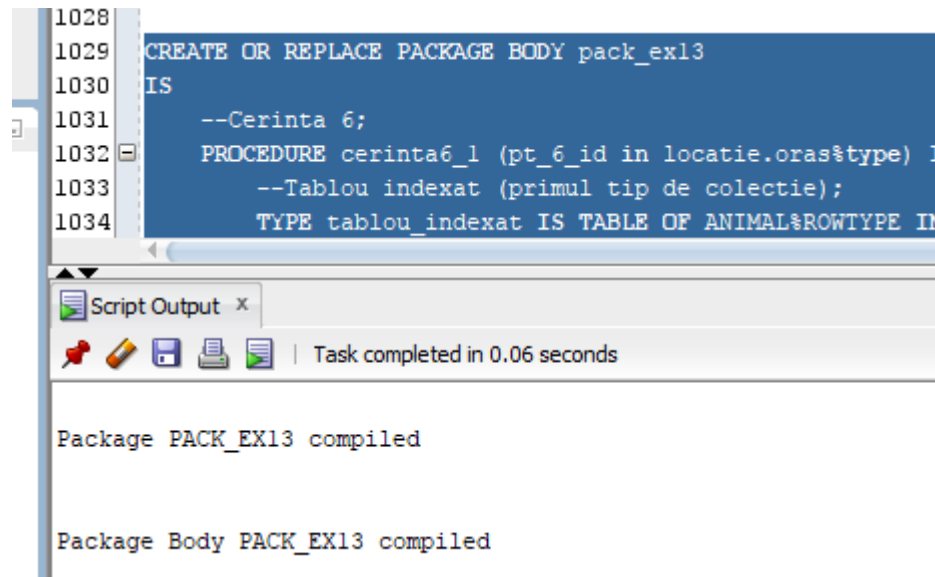
        DBMS_OUTPUT.PUT_LINE('Adapostul nu este sponsorizat!');

    WHEN TOO_MANY_ROWS THEN

        DBMS_OUTPUT.PUT_LINE('Too many rows!');

    END;

END pack_ex13;
```



```
1028
1029 CREATE OR REPLACE PACKAGE BODY pack_ex13
1030 IS
1031     --Cerinta 6;
1032     PROCEDURE cerinta6_1 (pt_6_id in locatie.oras%type) I
1033         --Tablou indexat (primul tip de colectie);
1034         TYPE tablou_indexat IS TABLE OF ANIMAL%ROWTYPE IN
```

Script Output x

Task completed in 0.06 seconds

Package PACK_EX13 compiled

Package Body PACK_EX13 compiled

--Apelarea cerintei 6 din pachet.

accept oras prompt 'Introduceti numele orasului: ';

DECLARE

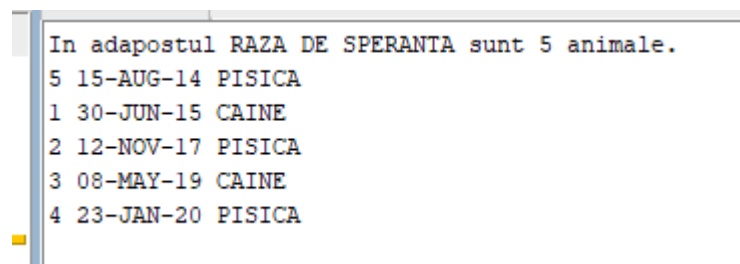
 v_id LOCATIE.oras%type := '&oras';

BEGIN

 pack_ex13.cerinta6_1(v_id);

END;

Pentru slatina:



```
In adăpostul RAZA DE SPERANTIA sunt 5 animale.
5 15-AUG-14 PISICA
1 30-JUN-15 CAINE
2 12-NOV-17 PISICA
3 08-MAY-19 CAINE
4 23-JAN-20 PISICA
```

--Apelarea cerintei 7 din pachet.

BEGIN

 pack_ex13.cerinta7;

END;

```
Adapostul CAT LIFE are 4 angajati!  
SACEANU CLAUDIU are salariul de 6000  
GRECU DAVID are salariul de 3400  
RATA ANDREEA are salariul de 3000  
  
Adapostul CATEI ODAI are 4 angajati!  
POPESCU COSMIN are salariul de 6600  
MIREI ANDRADA are salariul de 3300  
SERBAN OVIDIU are salariul de 2700  
  
Adapostul DOGS FIRST are 5 angajati!  
CALIN BOGDAN are salariul de 5900  
ILEA ADELINA are salariul de 3500  
CALIN SILVANA are salariul de 3200  
  
Adapostul NEW LIFE are 4 angajati!  
BUDURE DANIEL are salariul de 6500  
MONETE RAZVAN are salariul de 4000  
MAZILU VIOREL are salariul de 2600
```

Sunt destule date de afisat, nu le-am pus pe toate.

--Apelarea cerintei 8 din pachet.

accept Nume prompt 'Introduceti numele adoptantului: ';

DECLARE

adresa pack_ex13.tablou_locatie;

nume_ad varchar2(50) := '&Nume';

nr_adrese number;

nr_animale number;

cursor c is

select count(id_animal) nr

from animal

where id_adoptant = (select id_adoptant

from adoptanti

where upper(nume) = upper(nume_ad))

group by id_adapost

order by id_adapost;

BEGIN

select count(id_animal)

into nr_animale

from animal

where id_adoptant = (select id_adoptant

from adoptanti

where upper(ume) = upper(ume_ad));

adresa := pack_ex13.cerinta8(ume_ad);

nr_adrese := pack_ex13.ajutor_cerinta8(ume_ad);

DBMS_OUTPUT.PUT_LINE(ume_ad || ' a adoptat ' || nr_animale || ' animale.');

--ma folosesc de nr animale ca un counter;

nr_animale := 1;

FOR i IN c LOOP

DBMS_OUTPUT.PUT_LINE(i.nr || adresa(nr_animale));

nr_animale := nr_animale+1;

END LOOP;

EXCEPTION

WHEN NO_DATA_FOUND THEN

DBMS_OUTPUT.PUT_LINE('Code: ' || sqlcode || ' Error: ' || sqlerrm);

WHEN OTHERS THEN

DBMS_OUTPUT.PUT_LINE('Code: ' || sqlcode || ' Error: ' || sqlerrm);

END;

```
Draganescu a adoptat 3 animale.  
2 din judetul OLT, orasul SLATINA  
1 din judetul BOTOSANI, orasul BOTOSANI
```

--Apelarea cerintei 9 din pachet.

accept ume_ad prompt 'Nume adapost';

BEGIN

pack_ex13.cerinta_9('&nume_ad');

END;

Animalele din adapostul CATEI ODAI consuma alimentele:

Animalul cu id-ul: 31 consuma produsul: Conserva-Pui care are pretul de: 23.59

Animalul cu id-ul: 32 consuma produsul: Anti-Hairball care are pretul de: 24.99

Animalul cu id-ul: 33 consuma produsul: Vitamin E-Xtra care are pretul de: 79.5

Animalul cu id-ul: 34 consuma produsul: Indoor Pui care are pretul de: 20.5