

PROIECT BAZE DE DATE

IULIE 2021

BOBOC ȘTEFAN

Grupa 141

EXERCITIUL 1: Descrierea modelului real, a utilității acestuia și a regulilor de funcționare.

Modelul de date realizat este aplicat pentru o platformă online care se ocupă de rezervare biletelor la teatru. Există spectacole care se desfășoară în orașe diferite, cu actori în roluri principale diferiți și cu echipe de producători diferite.

Baza de date memorează date despre clienți și ține evidența fiecărui client la ce spectacol merge. De asemenea se reține și fiecare spectacol ce echipă de producători are, la ce teatru se joacă și cine joacă în rol principal.

EXERCITIUL 2: Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului.

Fiecare client poate avea un singur card cu care poate face o rezervare la spectacole. Clienții își pot rezerva locul la mai multe spectacole, iar la un spectacol pot veni mai mulți clienți.

Fiecare spectacol are un singur gen (comedie, tragedie etc.) și o singură echipă de producători ce a realizat piesa de teatru.

Echipa de producători poate să realizeze mai multe piese de teatru aflate în baza de date.

Un spectacol se poate afla în repertoriul mai multor teatre din toată țara și pot avea în rol principal mereu alți actori.

Teatrele se pot afla pe aceeași stradă și în același oraș.

EXERCITIUL 3: Descrierea entităților, incluzând precizarea cheii primare.

Baza de date este realizată din următoarele entități:

CLIENT: persoană ce dorește să-și rezerve un loc la un spectacol din cele disponibile pe platforma online.

CARD: este deținut de către client și poate face o rezervare

SPECTACOL: un eveniment de teatru pentru care un client își rezervă un loc. Cheia primară a entității este atributul *id_spectacol*.

GEN: categoria din care face parte un spectacol. Cheia primară a entității este atributul *id_gen*.

PRODUCATORI: echipa de oameni ce realizează *un spectacol*. Cheia primară a entității este atributul *id_prod*.

ACTOR: persoana din rolul principal al unei piese de teatru. Cheia primară a entității este atributul *id_actor*.

TEATRU: locul unde se desfășoară o piesă de teatru. Cheia primară a entității este atributul *id_teatru*.

LOCATIE: este adresa unde se află teatrul. Cheia primară a entității este atributul *id_locație*.

EXERCITIUL 4: Descrierea relațiilor, incluzând precizarea cardinalității acestora.

CLIENT_are_CARD: leagă entitățile CLIENT și CARD, reflectând relația dintre acestea (un client are un card și numai unul). Ea are cardinalitate minimă 1:1 și cardinalitate maximă 1:1;

CLIENT_rezervă_SPECTACOL: leagă entitățile CLIENT și SPECTACOL, reflectând relația dintre acestea (clientul rezervă un loc la un spectacol). Ea are cardinalitate minimă 0:0 și cardinalitate maximă n:n;

SPECTACOL_este(de tipul)_GEN: leagă entitățile SPECTACOL și GEN, reflectând relația dintre acestea (un spectacol de teatru are un gen). Ea are cardinalitate minimă 0:1 și cardinalitate maximă n:1.

PRODUCATORII_realizează_SPECTACOL: leagă entitățile PRODUCATORI și SPECTACOL, reflectând relația dintre acestea (un spectacol de teatru are un gen). Ea are cardinalitate minimă 0:1 și cardinalitate maximă n:1.

PRODUCATORII_realizează_SPECTACOL: leagă entitățile PRODUCATORI și SPECTACOL, reflectând relația dintre acestea (un spectacol de teatru are un gen). Ea are cardinalitate minimă 0:1 și cardinalitate maximă n:1.

ACTORUL_colaborează_cu_TEATRUL_pentru_SPECTACOL: relație de tip 3 care leagă entitățile SPECTACOL, PRODUCĂTORI, ACTOR, reflectând relația dintre acestea (spectacolul are loc într-un teatru cu o anumită persoană în rol principal, dar același spectacol se poate realiza în colaborare și cu alte teatre sau alte persoane/actori).

TEATRUL_se_află_la_LOCATIA: leagă entitățile TEATRU și LOCATIE, reflectând relația dintre acestea (teatrul se află la o anumite locație). Ea are cardinalitate minimă 1:1 și cardinalitate maximă n:1.

EXERCITIUL 5: Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicite, valori posibile ale atributelor.

Entitatea CLIENT are atributele:

id_client = cheie primară, număr de maxim șase cifre;

nume = numele clientului, maxim 30 de caractere, nu poate fi NULL;

prenume = prenumele clientului, maxim 30 de caractere, nu poate fi NULL;

email = adresa de e-mail a clientului, maxim 25 de caractere, nu poate fi NULL, este unic;

data_nastere = data la care s-a născut clientul;

Entitatea CARD are atributele:

id_card = cheie primară, număr de maxim șase cifre;

lună_expirare = luna la care expiră cardul, număr de maxim două cifre, nu poate fi NULL;

an_expirare = anul la care expiră cardul, număr de maxim două cifre, nu poate fi NULL;

brand = numele brandului cardului, maxim 15 de caractere, nu poate fi NULL;

id_client = cheia externă ce referă cheia id_client din CLIENT, număr de maxim șase cifre, unic;

Entitatea SPECTACOL are attributele:

id_spectacol = cheie primară, număr de maxim șase cifre;
nume_spectacol = numele spectacolului, maxim 35 de caractere, nu poate fi NULL, unic;
durata = durata spectacolului în ore, o singură cifră, nu poate fi NULL, valori din [1,5];
rating = ratingul spectacolului, o singură cifră cu posibilitatea unei zecimale, valori din [1,5];
an_lansare = anul în care s-a lansat spectacolul, nu poate fi NULL;
pret = prețul pentru un bilet (o rezervare de o persoană) la un spectacol, nu poate fi NULL;
id_gen = cheia externă ce referă cheia id_gen din GEN, număr de maxim șase cifre;
id_prod = cheia externă ce referă cheia id_prod din PRODUCATORI, număr de maxim șase cifre;

Entitatea GEN are attributele:

id_gen = cheie primară, număr maxim de șase cifre;
nume_gen = denumirea genului, maxim 10 de caractere, nu poate fi NULL;

Entitatea PRODUCATORI are attributele:

id_prod = cheie primară, număr maxim de șase cifre;
regizor = numele regizorului ce face parte din echipa de producători, maxim 25 de caractere, nu poate fi NULL;
scenarist = numele scenaristului ce face parte din echipa de producători, maxim 25 de caractere, nu poate fi NULL;

Entitatea LOCATIE are attributele:

id_locatie = cheie primară, număr maxim de șase cifre;
strada = strada locației, maxim 20 de caractere, nu poate fi NULL;
oras = orașul în care se află locația, maxim 20 de caractere, nu poate fi NULL;
cod_postal = codul poștal al adresei (locației), număr maxim de șase cifre;

Entitatea TEATRU are attributele:

id_teatru = cheie primară, număr maxim de șase cifre;
nume_teatru = numele teatrului, maxim 25 de caractere, nu poate fi NULL;
director = numele directorului care conduce respectivul teatru, maxim 20 de caractere;
id_locatie = cheia externă ce referă cheia id_locatie din LOCATIE, număr de maxim șase cifre;

Entitatea ACTOR are attributele:

id_actor = cheie primară, număr maxim de șase cifre;
nume_actor = numele actorului, maxim 20 de caractere, nu poate fi NULL;
prenume_actor = prenumele actorului, maxim 20 de caractere, nu poate fi NULL;
nr_premii = numărul de premii pe care le are actorul respectiv;

Tabelul asociativ REZERVARE are attributele:

id_rezervare = cheie primară, număr maxim de șase cifre;

data_rezervare = data pentru care s-a realizat rezervarea, nu poate fi NULL;

id_spectacol = cheia externă ce referă cheia id_spectacol din SPECTACOL, număr de maxim șase cifre, nu poate fi NULL;

id_client = cheia externă ce referă cheia id_client spectacol din CLIENT, număr de maxim șase cifre, nu poate fi NULL;

Tabelul asociativ COLABORARE are attributele:

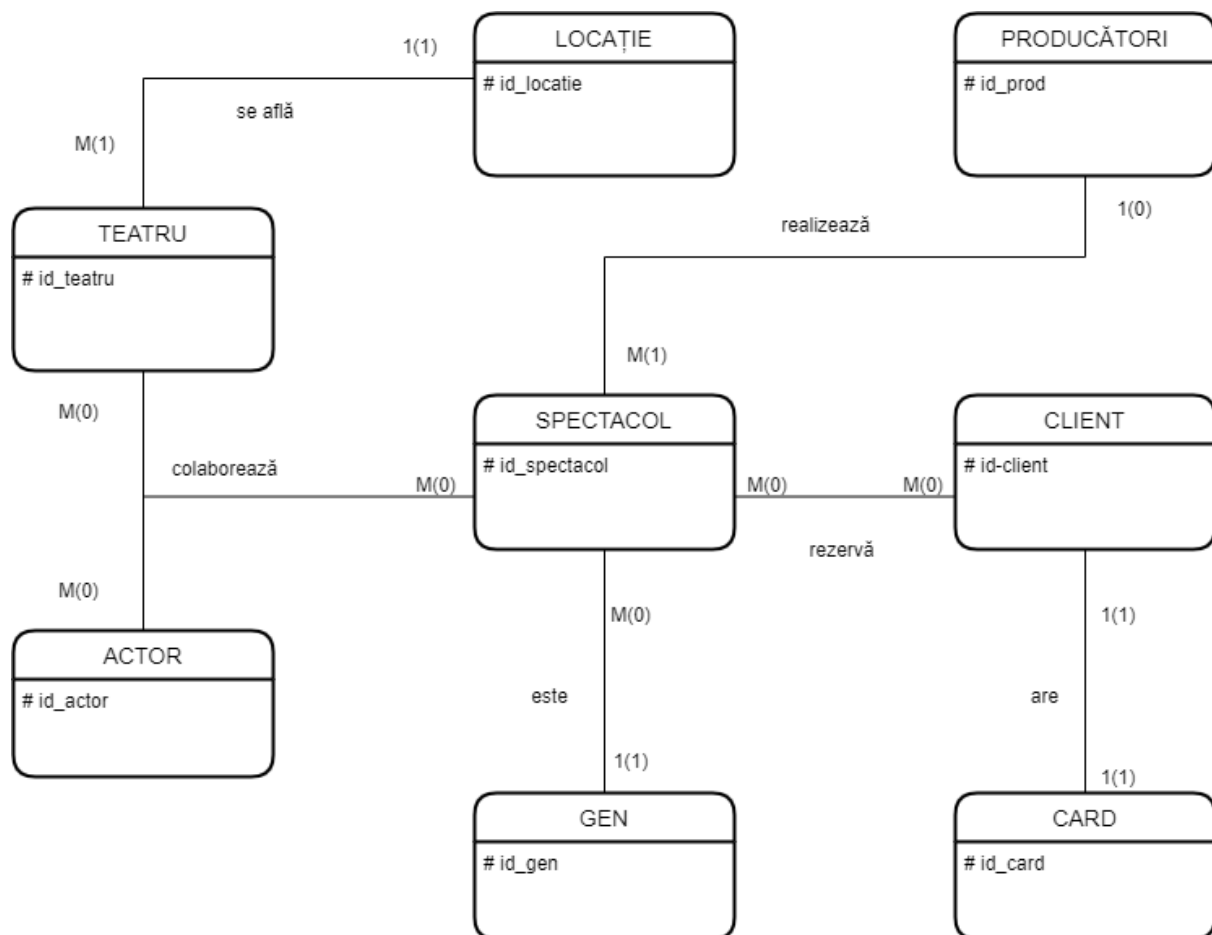
id_colaborare = cheie primară, număr maxim de șase cifre;

id_spectacol = cheia externă ce referă cheia id_spectacol spectacol din SPECTACOL, număr de maxim șase cifre, nu poate fi NULL;

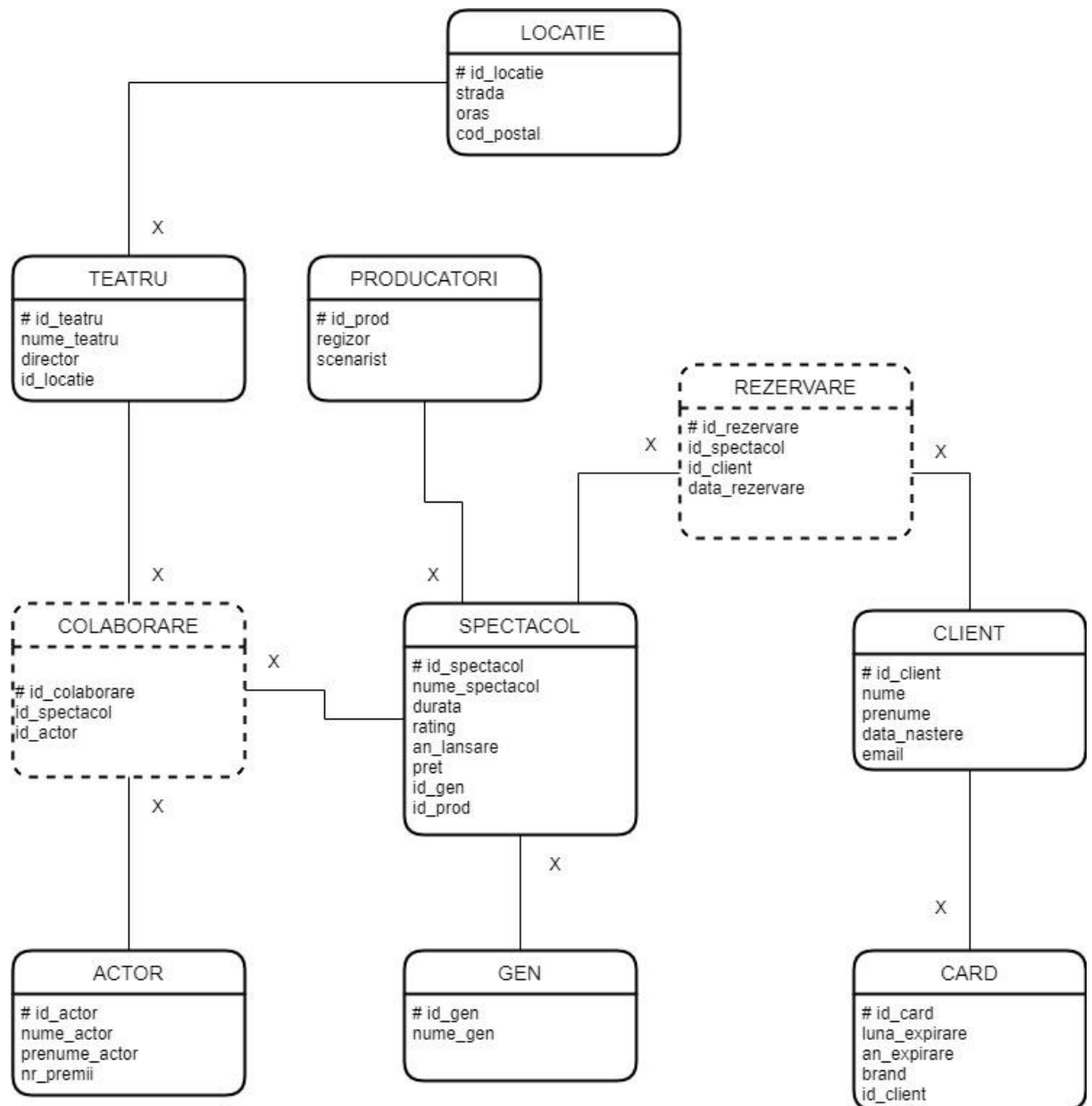
id_teatru = cheia externă ce referă cheia id_teatru spectacol din TEATRU, număr de maxim șase cifre, nu poate fi NULL;

id_actor = cheia externă ce referă cheia id_actor din ACTOR, număr de maxim șase cifre, nu poate fi NULL;

EXERCITIUL 6: Realizarea diagramei entitate-relație corespunzătoare descrierii de la punctele 3-5.



EXERCITIUL 7: Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectată la punctul 6. Diagrama conceptuală obținută trebuie să conțină minimum 6 tabele (fără considerarea subentităților), dintre care cel puțin un tabel asociativ.



EXERCITIUL 8: Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectată la punctul 7.

CLIENT(#id_client, nume, prenume, data_nastere, email),

CARD(#id_card, luna_expirare, an_expirare, brand),

SPECTACOL(#id_spectacol, nume_spectacol, durata, rating, an_lansare, pret),

GEN(#id_gen, nume_gen),

PRODUCATORI(#id_prod, regizor, scenarist),

LOCATIE(#id_locatie, strada, oras, cod_postal);

TEATRU(#id_teatru, nume_teatru, director),

ACTOR(#id_actor, nume_actor, prenume_actor, nr_premii),

REZERVARE(#id_rezervare, data_rezervare),

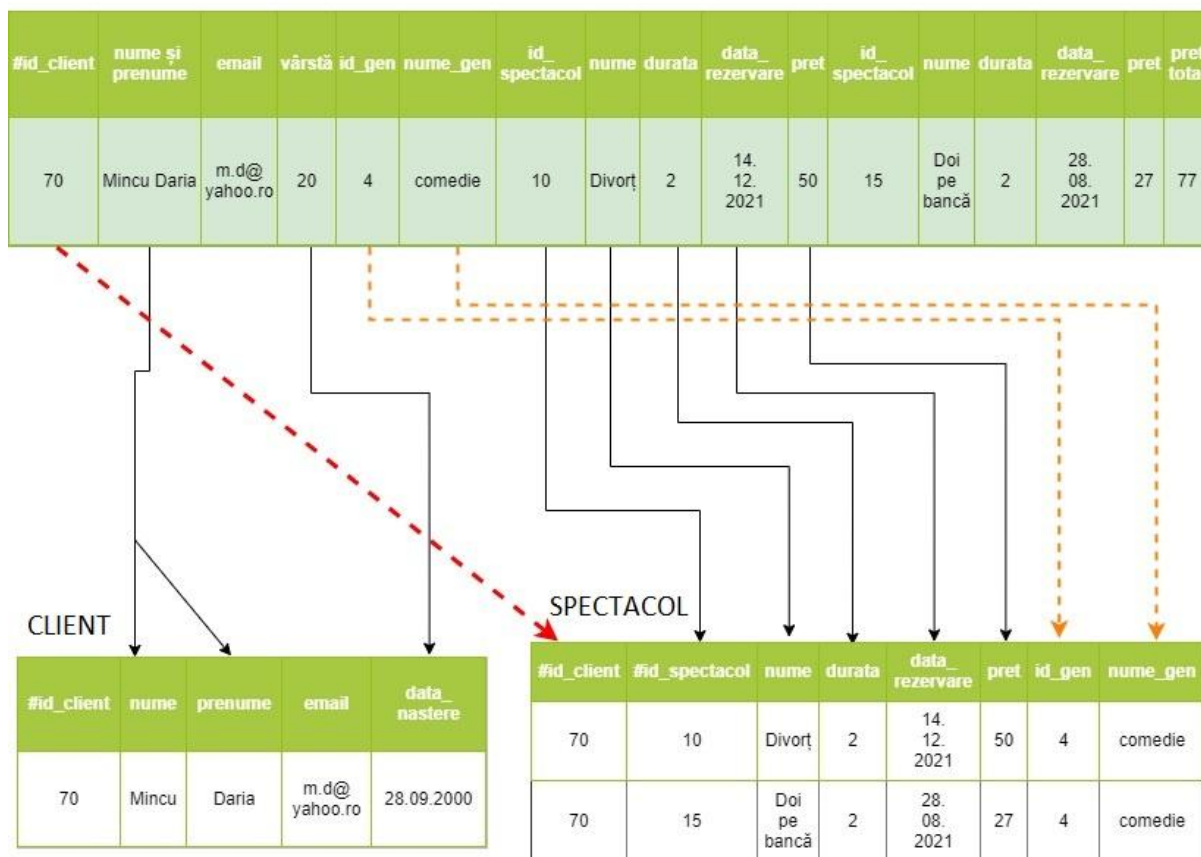
COLABORARE(#id_colaborare);

EXERCITIUL 9: Realizarea normalizării până la forma normală 3 (FN1-FN3).

Pentru acest exercițiu au fost eliminate câteva dintre atributele entității SPECTACOL pentru a nu încărca schemele cu etapele normalizării.

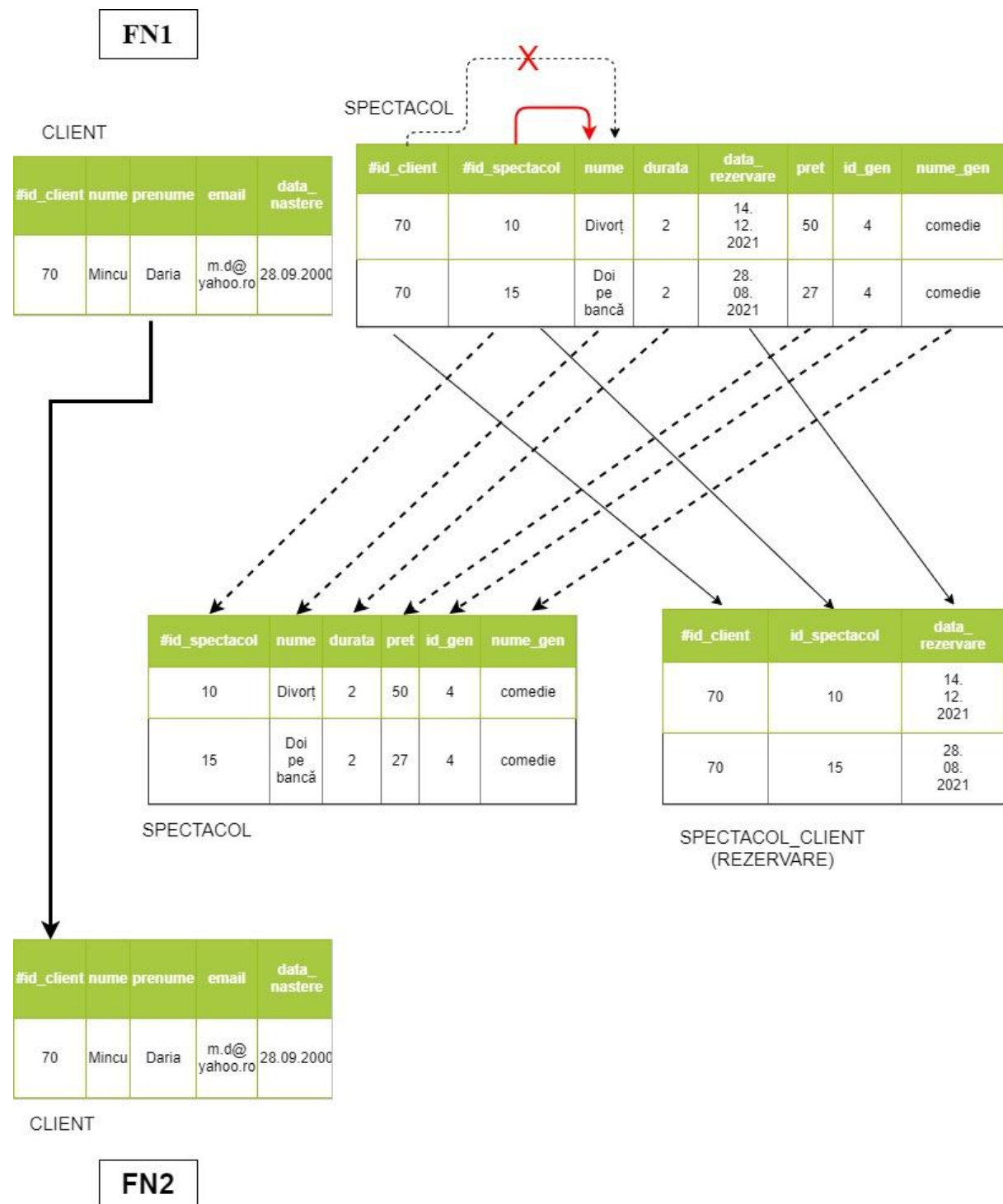
Din RU în FN1:

RELAȚIA UNIVERSALĂ

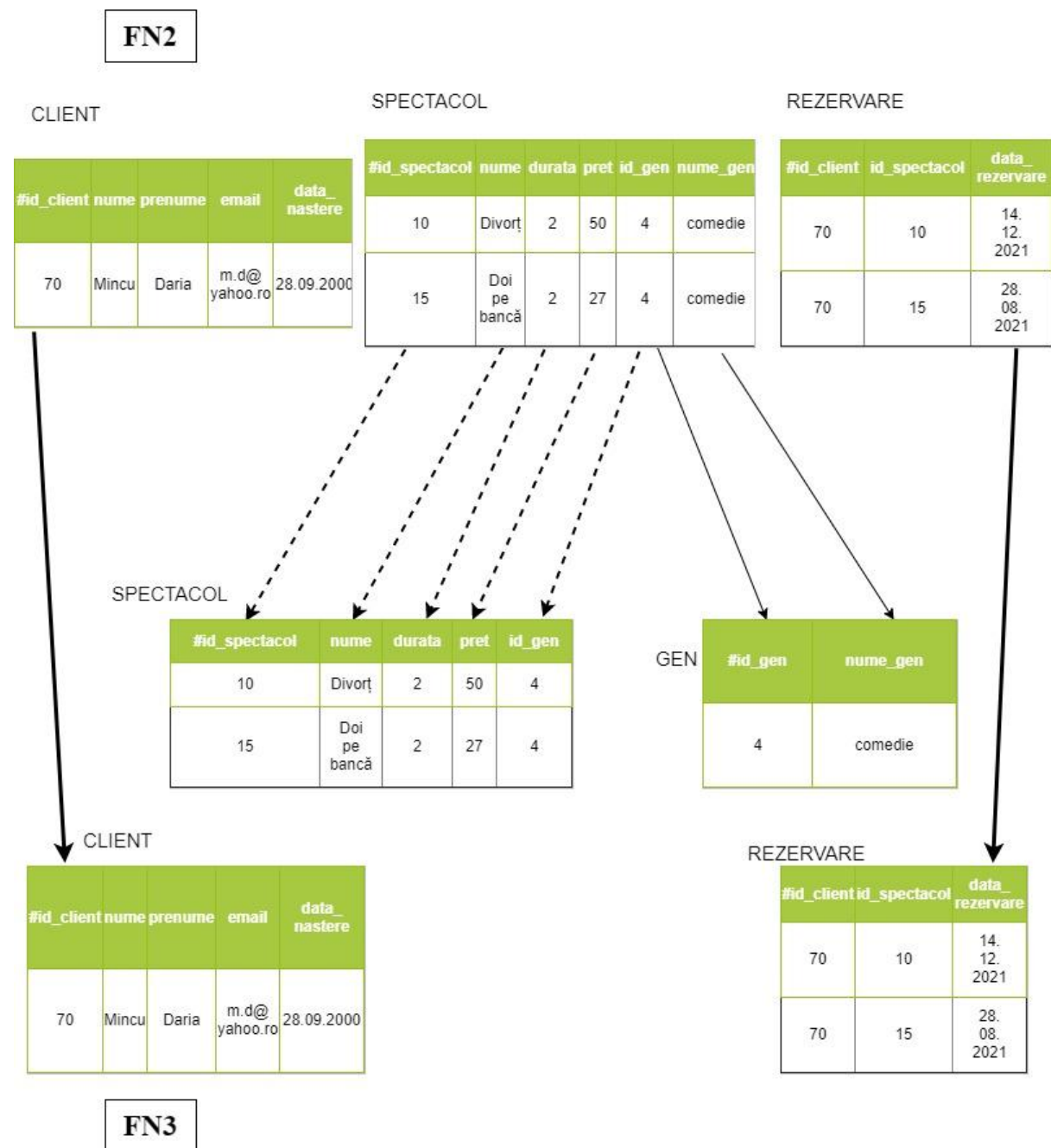


FN1

Din FN1 în FN2:



Din FN2 în FN3:



EXERCITIUL 10: Crearea tabelelor în SQL și inserarea de date coerente în fiecare dintre acestea (minimum 5 înregistrări în fiecare tabel neasociativ; minimum 10 înregistrări în tabelele asociative).

```
create table CLIENT(
    id_client number(6) primary key,
    nume varchar2(30) not null,
    prenume varchar2(30) not null,
    email varchar2(25) not null,
    data_nastere date,
    unique (email)
);

insert into CLIENT values(10,'Pop','Alex','popa@gmail.com',null);
insert into CLIENT
values(20,'Ionescu','Ana','ionescu.a@gmail.com',to_date('18-02-1999','DD-MM-YYYY'));
insert into CLIENT
values(30,'Gabor','Mara','gabor.m@gmail.com',to_date('06-12-1987','DD-MM-YYYY'));
insert into CLIENT
values(40,'Suciu','Matei','suciu.m@gmail.com',to_date('18-05-2001','DD-MM-YYYY'));
insert into CLIENT
values(50,'Avram','George','avram.g@gmail.com',null);
insert into CLIENT
values(60,'Pop','Alex','popa.a@gmail.com',null);
insert into CLIENT
values(70,'Mincu','Daria','mincu.d@yahoo.com',to_date('28-09-2000','DD-MM-YYYY'));
```

	ID_CLIENT	NUME	PRENUME	EMAIL	DATA_NASTERE
1	10	Pop	Alex	popa@gmail.com	(null)
2	20	Ionescu	Ana	ionescu.a@gmail.com	18-FEB-99
3	30	Gabor	Mara	gabor.m@gmail.com	06-DEC-87
4	40	Suciu	Matei	suciu.m@gmail.com	18-MAY-01
5	50	Avram	George	avram.g@gmail.com	(null)
6	60	Pop	Alex	popa.a@gmail.com	(null)
7	70	Mincu	Daria	mincu.d@yahoo.com	28-SEP-00

```

create table CARD(
    id_card number(6) primary key,
    luna_expirare number(2) not null,
    an_expirare number(2) not null,
    brand varchar2(15) not null,
    id_client number(6) unique,
    constraint FK_client_card foreign key(id_client) references
CLIENT(id_client)
);

```

```

insert into CARD values(1,'09','23','Visa',30);
insert into CARD values(2,'04','21','Mastercard',50);
insert into CARD values(3,'07','22','Mastercard',70);
insert into CARD values(4,'11','21','Citibank',60);
insert into CARD values(5,'04','23','Visa',40);
insert into CARD values(6,'01','24','AmericanExpress',20);

```

	ID_CARD	LUNA_EXPIRARE	AN_EXPIRARE	BRAND	ID_CLIENT
1	1	9	23	Visa	30
2	2	4	21	Mastercard	50
3	3	7	22	Mastercard	70
4	4	11	21	Citibank	60
5	5	4	23	Visa	40
6	6	1	24	AmericanExpress	20

```
create table GEN(
    id_gen number(6) primary key,
    nume_gen varchar2(10) not null
);
```

```
insert into GEN values(1,'absurd');
insert into GEN values(2,'grotesc');
insert into GEN values(3,'satira');
insert into GEN values(4,'comedie');
insert into GEN values(5,'tragic');
insert into GEN values(6,'drama');
```

ID_GEN	NUME_GEN
1	1absurd
2	2grotesc
3	3satira
4	4comedie
5	5tragic
6	6drama

```
create table PRODUCATORI(
    id_prod number(6) primary key,
    regizor varchar(25) not null,
    scenarist varchar(25) not null
);
```

```
insert into PRODUCATORI values(1,'Felix Alexa', 'Alex Ifrim');
insert into PRODUCATORI values(2,'Sica Alexandrescu', 'Vlad
Musteriu');
insert into PRODUCATORI values(3,'Mara Barbu', 'Mara Barbu');
insert into PRODUCATORI values(4,'Mihai Bendeac', 'Alexandra
Matei');
```

ID_PROD	REGIZOR	SCENARIST
1	1Felix Alexa	Alex Ifrim
2	2Sica Alexandrescu	Vlad Musteriu
3	3Mara Barbu	Mara Barbu
4	4Mihai Bendeac	Alexandra Matei

```

create table SPECTACOL(
    id_spectacol number(6) primary key,
    nume_spectacol varchar2(35) not null unique,
    durata number(1) not null,
    rating number(2,1),
    an_lansare varchar2(4) not null,
    pret number(5,1) not null,
    id_gen number(6),
    id_prod number(6),
    constraint FK_gen_spectacol foreign key(id_gen) references
GEN(id_gen),
    constraint FK_producatori_spectacol foreign key(id_prod)
references PRODUCATORI(id_prod),
    constraint CONS_durata check (durata <= 5 and durata >= 1),
    constraint CONS_rating check (rating <= 5 and rating >= 1)
);

insert into SPECTACOL values(10,'Divort in ziua nuntii', 2, 3,
2002, 50, 4, 3);
insert into SPECTACOL values(11,'Ursul', 3, 3.5, 2010, 20, 2, 3);
insert into SPECTACOL values(12,'Necasatoria', 2, 4, 2010, 15, 5,
2);
insert into SPECTACOL values(13,'Cursa de soareci', 2, 2, 1999,
33, 6, 3);
insert into SPECTACOL values(14,'O noapte furtunoasa', 3, 5, 1999,
45, 4, 1);
insert into SPECTACOL values(15,'Doi pe o banca', 2, null, 2019,
27, 4, 4);
insert into SPECTACOL values(16,'Colectionarul', 2, null, 2021,
16, 4, 2);
insert into SPECTACOL values(17,'Meciul de comedie', 2, 4, 2011,
17, 2, 2);
insert into SPECTACOL values(18,'Avarul', 2, 4, 2021, 28, 04, 04);

```

	ID_SPECTACOL	NUME_SPECTACOL	DURATA	RATING	AN_LANSARE	PRET	ID_GEN	ID_PROD
1	10	Divort in ziua nuntii	2	3	2002	50	4	3
2	11	Ursul	3	3.5	2010	20	2	3
3	12	Necasatoria	2	4	2010	15	5	2
4	13	Cursa de soareci	2	2	1999	33	6	3
5	14	O noapte furtunoasa	3	5	1999	45	4	1
6	15	Doi pe o banca	2	(null)	2019	27	4	4
7	16	Colectionarul	2	(null)	2021	16	4	2
8	17	Meciul de comedie	2	4	2011	17	2	2
9	18	Avarul	2	4	2021	28	4	4

```

create table LOCATIE(
    id_locatie number(6) primary key,
    strada varchar2(20) not null,
    oras varchar2(25) not null,
    cod_postal number(6)
);

insert into LOCATIE values(1,'Lalelelor','Bucuresti',null);
insert into LOCATIE values(2,'Florilor','Bucuresti',null);
insert into LOCATIE values(3,'Ghioceilor','Bucuresti',null);
insert into LOCATIE values(4,'Bujorilor','Cluj',null);
insert into LOCATIE values(5,'Frunzelor','Brasov',null);

```

	ID_LOCATIE	STRADA	ORAS	COD_POSTAL
1	1	Lalelelor	Bucuresti	(null)
2	2	Florilor	Bucuresti	(null)
3	3	Ghioceilor	Bucuresti	(null)
4	4	Bujorilor	Cluj	(null)
5	5	Frunzelor	Brasov	(null)

```

create table TEATRU(
    id_teatru number(6) primary key,
    nume_teatru varchar2(25) not null,
    director varchar2(20),
    id_locatie number(6),
    constraint FK_locatie_teatru foreign key(id_locatie)
references LOCATIE(id_locatie)
);

insert into TEATRU values(10,'Teatrul Judetean Brasov','Zaharia
Herdelea',5);
insert into TEATRU values(20,'Teatrul Judetean Cluj','Vasile
Baciu',4);
insert into TEATRU values(30,'Teatrul de Comedie Brasov','Stefan
Gheorghidui',5);
insert into TEATRU values(40,'Teatrul Artcub','Otilia
Marculescu',3);
insert into TEATRU values(50,'Teatrul Nottara','Nechifor
Lipan',1);
insert into TEATRU values(60,'Teatrul Guguta','Lica Samadai',2);

```


	ID_TEATRU	NUME_TEATRU	DIRECTOR	ID_LOCATIE
1	10	Teatrul Judetean Brasov	Zaharia Herdelea	5
2	20	Teatrul Judetean Cluj	Vasile Baci	4
3	30	Teatrul de Comedie Brasov	Stefan Gheorghid	5
4	40	Teatrul Artcub	Otilia Marculescu	3
5	50	Teatrul Nottara	Nechifor Lipan	1
6	60	Teatrul Guguta	Lica Samadau	2

```

create table ACTOR(
    id_actor number(6) primary key,
    nume_actor varchar(20) not null,
    prenume_actor varchar(20) not null,
    nr_premii number(2)
);

insert into ACTOR values(10,'Morgenstern','Maia',2);
insert into ACTOR values(20,'Malaele','Horatiu',3);
insert into ACTOR values(30,'Iures','Marcel',1);
insert into ACTOR values(40,'Alexandru','Alex', null);
insert into ACTOR values(50,'Arsinel','Alexandru',2);

```

	ID_ACTOR	NUME_ACTOR	PRENUME_ACTOR	NR_PREMII
1	10	Morgenstern	Maia	2
2	20	Malaele	Horatiu	3
3	30	Iures	Marcel	1
4	40	Alexandru	Alex	(null)
5	50	Arsinel	Alexandru	2

```

create table REZERVARE(
    id_rezervare number(6) primary key,
    id_spectacol number(6) not null,
    id_client number(6) not null,
    data_rezervare date not null,
    constraint FK_client_rezervare foreign key(id_client) references
CLIENT(id_client),
    constraint FK_spectacol_rezervare foreign key(id_spectacol) references
SPECTACOL(id_spectacol)
);

```

```

insert into REZERVARE values(101, 10, 10,
to_date('04-11-2021','DD-MM-YYYY'));
insert into REZERVARE values(102, 11, 10,
to_date('13-10-2021','DD-MM-YYYY'));
insert into REZERVARE values(103, 12, 10,
to_date('25-03-2021','DD-MM-YYYY'));
insert into REZERVARE values(104, 10, 70,
to_date('04-12-2021','DD-MM-YYYY'));
insert into REZERVARE values(105, 11, 70,
to_date('28-08-2021','DD-MM-YYYY'));
insert into REZERVARE values(106, 12, 70,
to_date('11-01-2021','DD-MM-YYYY'));
insert into REZERVARE values(107, 13, 70,
to_date('14-08-2021','DD-MM-YYYY'));
insert into REZERVARE values(108, 17, 20,
to_date('07-12-2021','DD-MM-YYYY'));
insert into REZERVARE values(109, 17, 20,
to_date('12-08-2021','DD-MM-YYYY'));
insert into REZERVARE values(110, 10, 40,
to_date('23-10-2021','DD-MM-YYYY'));
insert into REZERVARE values(111, 13, 20,
to_date('14-08-2021','DD-MM-YYYY'));
insert into REZERVARE values(112, 10, 10,
to_date('01-06-2021','DD-MM-YYYY'));

```

	ID_REZERVARE	ID_SPECTACOL	ID_CLIENT	DATA_REZERVARE
1	101	10	10	04-NOV-21
2	102	11	10	13-OCT-21
3	103	12	10	25-MAR-21
4	104	10	70	04-DEC-21
5	105	11	70	28-AUG-21
6	106	12	70	11-JAN-21
7	107	13	70	14-AUG-21
8	108	17	20	07-DEC-21
9	109	17	20	12-AUG-21
10	110	10	40	23-OCT-21
11	111	13	20	14-AUG-21
12	112	10	10	01-JUN-21

```

create table COLABORARE(
    id_colaborare number(6) primary key,
    id_spectacol number(6) not null,
    id_teatru number(6) not null,
    id_actor number(6) not null,
    constraint FK_spectacol_colaborare foreign key(id_spectacol)
references SPECTACOL(id_spectacol),
    constraint FK_actor_colaborare foreign key(id_actor)
references ACTOR(id_actor),
    constraint FK_teatru_colaborare foreign key(id_teatru)
references TEATRU(id_teatru)
);

```

```

insert into COLABORARE values(101,10,10,20);
insert into COLABORARE values(102,10,20,20);
insert into COLABORARE values(103,10,40,20);
insert into COLABORARE values(104,11,30,30);
insert into COLABORARE values(105,12,40,30);
insert into COLABORARE values(106,12,40,30);
insert into COLABORARE values(107,12,60,40);
insert into COLABORARE values(108,12,50,40);
insert into COLABORARE values(109,16,10,40);
insert into COLABORARE values(110,17,10,20);

```

	ID_COLABORARE	ID_SPECTACOL	ID_TEATRU	ID_ACTOR
1	101	10	10	20
2	102	10	20	20
3	103	10	40	20
4	104	11	30	30
5	105	12	40	30
6	106	12	40	30
7	107	12	60	40
8	108	12	50	40
9	109	16	10	40
10	110	17	10	20

EXERCITIUL 11: Formulați în limbaj natural și implementați 5 cereri SQL complexe ce vor utiliza, în ansamblul lor, următoarele elemente:

- operație join pe cel puțin 4 tabele
- filtrare la nivel de linii
- subcereri sincronizate în care intervin cel puțin 3 tabele
- subcereri nesincronizate în care intervin cel puțin 3 tabele
- grupări de date, funcții grup, filtrare la nivel de grupuri
- ordonări
- utilizarea a cel puțin 2 funcții pe șiruri de caractere, 2 funcții pe date calendaristice, a funcțiilor NVL și DECODE, a cel puțin unei expresii CASE
- utilizarea a cel puțin 1 bloc de cerere (clauza WITH)

a) Afișarea orașelor în care se joacă piese de teatru pe strazile cu numele „Frunzelor” și ratingul spectacolelor. Rezultatele sunt formate astfel: <Nume_spectacol> se joacă în <nume_orăș>; și sunt afișate în ordine descrescătoare după cheia primară a spectacolelor. Dacă spectacolele nu au rating se afișează „Fara rating”.

```
select concat(concat(s.num_e_spectacol,' se joaca in '),l.oras)
"Unde se joaca spectacolele", decode(s.rating, null,'Fara rating',
s.rating) "Rating"
from locatie l join teatru t on l.id_locatie = t.id_locatie
              join colaborare c on c.id_teatru = t.id_teatru
              join spectacol s on s.id_spectacol =
c.id_spectacol
where initcap(l.strada) = 'Frunzelor'
order by s.id_spectacol desc;
```

	Unde se joaca spectacolele	Rating
1	Meciul de comedie se joaca in Brasov	4
2	Colectionarul se joaca in Brasov	Fara rating
3	Ursul se joaca in Brasov	3.5
4	Divort in ziua nuntii se joaca in Brasov	3

b) Afișarea datei rezervării (ziua săptămânii, luna, numărul zilei din lună, anul), id-ul rezervării și câte zile mai sunt până când clientul merge la spectacol, pentru spectacolele cu genul „comedie”. Dacă data rezervării a trecut atunci se va afișa „Expirat”.

```
select to_char(data_rezervare, 'dl') "Data rezervare", id_rezervare,
case when to_char(round(data_rezervare - sysdate) + 1) <
to_char(0) then 'Expirat'
else
to_char(round(data_rezervare - sysdate) + 1) end "Numar zile
ramase"
from rezervare
where id_spectacol in(
select id_spectacol
from spectacol
where id_gen in(
select id_gen
from gen
where nume_gen = 'comedie'
)
);
```

	Data rezervare	ID_REZERVARE	Numar zile ramase
1	Thursday, November 04, 2021	101	132
2	Saturday, December 04, 2021	104	162
3	Saturday, October 23, 2021	110	120
4	Tuesday, June 01, 2021	112	Expirat

c) Afișarea ratingului mediu al spectacolelor pentru fiecare an mai mare decât anul 2000.

```
select an_lansare, nvl(avg(rating), 0) "Rating mediu"
from spectacol
group by an_lansare
having an_lansare > 2000;
```

	AN_LANSARE	Rating mediu
1	2010	3.75
2	2021	4
3	2019	0
4	2011	4
5	2002	3

d) Afișarea genurilor de spectacole și suma totală a prețurilor spectacolelor cu genul respectiv, al căror spectacole au valoarea totală a prețului mai mică decât media tuturor prețurilor spectacolelor.

```
with total_genuri as (  
    select nume_gen, sum(pret) Total  
    from gen g join spectacol s on (g.id_gen =  
s.id_gen)  
    group by nume_gen  
    ),  
fin_medie as (  
    select avg(Total) Medie  
    from total_genuri  
    )  
  
select *  
from total_genuri  
where Total < (  
    select Medie  
    from fin_medie  
    )  
order by nume_gen;
```

	NUME_GEN	TOTAL
1	drama	33
2	grotesc	37
3	tragic	15

e) Afișarea numelui spectacolului, a prețului, a regizorului și al anului lansării spectacolelor al căror preț este mai mare decât media prețurilor spectacolelor ale unui regizor. Dacă anul lansării este anul curent atunci se va afișa „*NOU*”.

```
select s.ume_spectacol, s.pret, p.regizor, case s.an_lansare when
to_char(sysdate,'yyyy') then '*NOU*' else s.an_lansare end
"Vechime"
from spectacol s, producatori p
where pret > (select avg(ss.pret)
              from spectacol ss
              where ss.id_prod = s.id_prod
) and s.id_prod = p.id_prod;
```

	NUME_SPECTACOL	PRET	REGIZOR	Vechime
1	Divort in ziua nuntii	50	Mara Barbu	2002
2	Meciul de comedie	17	Sica Alexandrescu	2011
3	Avarul	28	Mihai Bendeac	*NOU*

EXERCITIUL 12: Implementarea a trei operații de actualizare sau suprimare a datelor utilizând subcereri.

a) Actualizarea prețurilor spectacolelor care s-au lansat între anii [2000, 2012]. Noul preț va fi mai scump cu 30% decât prețul celui mai scump spectacol.

Înainte:

	ID_SPECTACOL	PRET
1	10	50
2	11	20
3	12	15
4	13	33
5	14	45
6	15	27
7	16	16
8	17	17
9	18	28

```
update spectacol
set pret = (
    select max(pret) * 1.3
    from spectacol
)
where 2000 <= an_lansare and an_lansare <= 2012;
```

După:

	ID_SPECTACOL	PRET
1	10	65
2	11	65
3	12	65
4	13	33
5	14	45
6	15	27
7	16	16
8	17	65
9	18	28

b) Ștergerea spectacolelor unde în echipă de producători regizor este „Mihai Bendeac”.

Înainte:

	ID_SPECTACOL	ID_PROD
1	10	3
2	11	3
3	12	2
4	13	3
5	14	1
6	15	4
7	16	2
8	17	2
9	18	4

```
delete
from spectacol
where id_prod = any(
    select id_prod
    from producatori
    where regizor = 'Mihai Bendeac'
);
```

După:

	ID_SPECTACOL	ID_PROD
1	10	3
2	11	3
3	12	2
4	13	3
5	14	1
6	16	2
7	17	2

c) Actualizarea prețurilor spectacolelor al căror rating este unic.

Înainte:

	ID_SPECTACOL	PRET
1	10	50
2	11	20
3	12	15
4	13	33
5	14	45
6	15	27
7	16	16
8	17	17
9	18	28

```
update spectacol
set pret = pret / 2
where rating in (
    select rating
    from spectacol
    group by rating
    having count(*) = 1
);
```

După:

	ID_SPECTACOL	PRET
1	10	25
2	11	10
3	12	15
4	13	16.5
5	14	22.5
6	15	27
7	16	16
8	17	17
9	18	28

EXERCITIUL 13: Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele (punctul 10).

Înainte:

	ID_CLIENT	NUME	PRENUME	EMAIL	DATA_NASTERE
1	10	Pop	Alex	popa@gmail.com	(null)
2	20	Ionescu	Ana	ionescu.a@gmail.com	18-FEB-99
3	30	Gabor	Mara	gabor.m@gmail.com	06-DEC-87
4	40	Suciu	Matei	suciu.m@gmail.com	18-MAY-01
5	50	Avram	George	avram.g@gmail.com	(null)
6	60	Pop	Alex	popa.a@gmail.com	(null)
7	70	Mincu	Daria	mincu.d@yahoo.com	28-SEP-00

```
create sequence secventa_client
increment by 10
start with 80
maxvalue 90000
nocycle;
```

```
insert into client
values (secventa_client.nextval, 'Stanciu', 'Miron',
'stanciu.mi@yahoo.com', null);
```

```
insert into client
values (secventa_client.nextval, 'Stamate', 'Dumitru',
'stamate.d@office.ro', to_date('18-02-1982','DD-MM-YYYY'));
```

După:

	ID_CLIENT	NUME	PRENUME	EMAIL	DATA_NASTERE
1	10	Pop	Alex	popa@gmail.com	(null)
2	20	Ionescu	Ana	ionescu.a@gmail.com	18-FEB-99
3	30	Gabor	Mara	gabor.m@gmail.com	06-DEC-87
4	40	Suciu	Matei	suciu.m@gmail.com	18-MAY-01
5	50	Avram	George	avram.g@gmail.com	(null)
6	60	Pop	Alex	popa.a@gmail.com	(null)
7	70	Mincu	Daria	mincu.d@yahoo.com	28-SEP-00
8	80	Stanciu	Miron	stanciu.mi@yahoo.com	(null)
9	90	Stamate	Dumitru	stamate.d@office.ro	18-FEB-82