PROIECT BAZE DE DATE

GESTIONAREA UNEI

GALERII DE OPERE DE ARTA

Hârnagea Andrei-Alexandru

Grupa 142

Exercitiul 1

Baza de date contine informatii cu privire la o galerie de arta, unde se desfasoara evenimentele, stafful care participa la evenimente si informatii legate despre operele de arta.

O galerie poate lua loc intr-o singura locatie, memorata ca un oras. O galerie are un sef.

Orice modificare a unei galerii duce catre un eveniment nou. Pentru fiecare eveniment se memoreaza numele si data desfasurarii. Un eveniment este sponzorizat de sponsori (in alte cuvinte, o firma) si are mai multi angajati staff.

Un eveniment se tine intr-o locatie. La o locatie se afla mai multe exponate, iar pentru fiecare exponat se mentine autorul.

Sunt doua tipuri de roluri: unul pentru staff si celalalt pentru sponsor. Un angajat are o singura sarcina si poate participa la un singur eveniment. Un sponsor poate sponsoriza mai multe evenimente.

Exercitiul 2

Modelul de date respecta anumite restrictii de functionare:

* Un exponat se poate afla la mai multe evenimente
* O galerie trebuie sa aiba loc intr-un oras
* Un eveniment trebuie sa aiba cel putin un staff si un sponsor
* La o locatie trebuie sa fie cel putin doua exponate
* Fiecare exponat are un autor
* Fiecare galerie are un nume
* Un sponsor poate sponsoriza mai multe evenimente si un eveniment poate avea mai multi sponsori
* Fiecare galerie are o echipa staff si pentru fiecare eveniment sunt mai multe sarcini cu angajati diferiti

Exercitiul 3

GALERIE = o galerie are loc intr-un singur oras, o singuta schimbare duce catre un eveniment nou. Cheia primara este id

EVENIMENT = aici se prezinta exponatele, fiecare eveniment are o locatie. Cheia primara este id.

STAFF = echipa care lucreaza la un eveniment. Cheia primara este id.

ANGAJAT = persoana care lucreaza la un eveniment, apartine unei echipe staff. Cheia primara este idt.

LOCATIE = unde se tine loc evenimentul. Cheia primara este id.

EXPONAT = ce se prezinta la un eveniment. Cheia primara este id.

AUTOR = cine a creat exponatul. Cheia primara este id.

COORDONATOR = o galerie are un „sef”. Cheia primara este id.

SPONSOR = sponsorizeaza cu bani o galerie . Cheia primara este id

Exercitiul 4

GALERIE\_are\_loc\_la\_Oras = relatie de tip one-to-many, o galerie are loc intr-un singur oras, dar un oras poate avea mai multe galerii

EXPONAT\_apartine\_de\_AUTOR = relatie one-to-many, un exponat are un autor.

EXPONAT\_se\_afla\_la\_LOCATIE = relatie one-to-many, o locatie aduna mai multe exponate, un exponat se afla intr-o singura locatie.

ANGAJAT\_apartine\_de\_STAFF = relatie one-to-many, o echipa este formata din mai multi angajati

STAFF\_participa\_la\_EVENIMET = relatie one-to-many, o echipa staff munceste pentru en eveniment.

GALERIE\_este\_sponsorizata\_de\_SPONSOR = relatie many to many, un sponsor poate sponsoriza mai multe galerii, iar o galerie poate avea mai multi sponsori

EVENIMENT\_apartine\_de\_GALERIE = relatie one to many, un eveniment face parte dintr-o galerie

Exercitiul 5

Entitatea GALERIE are ca atribute:

GALERIE

id#

nume

* id, variabila de tip intrg care reprezinta id-ul galeriei
* nume\_galerie, variabila de tip caracter, cu lungima maxima de 255, care reprezinta numele galeriei

|  |  |
| --- | --- |
| id | nume\_galerie |
| PK | NOT NULL |

EVENIMENT

id#

nume\_eveniment

data\_inceput

data\_final

idGalerie

Entitatea EVENIMENT are ca atribute:

* id, variabila de tip intreg, care reprezinta id-ul galeriei
* nume\_eveniment, variabila de tip caracter, cu lungima maxima de 255, care reprezinta numele evenimentului
* data\_inceput, variabila de tip data care marcheaza inceputul unui eveniment
* data\_final, variabila de tip data care marcheaza sfarsitul unui eveniment

|  |  |  |  |
| --- | --- | --- | --- |
| id | nume\_evenimet | data\_inceput | data\_final |
| PK | NOT NULL | NOT NULL | NOT NULL |

Enitatea LOCATIE are ca atribute:

LOCATIE

id#

ora\_inceput

ora\_final

nume\_loc

* id , variabila de tip intreg, care reprezinta id-ul locatiei
* ora\_inceput, variabila de tip intrg care spune la cat se deschide locatie
* ora\_final, variabila de tip intreg care spune la cat se inchide locatia
* nume\_loc, variabila de tip caracter, care reprezinta numele locatiei

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | nume\_loc | nume\_locatie | ora\_inceput | ora\_final |
| PK | NOT NULL | NOT NULL | NOT NULL | NOT NULL |

Entitatea EXPONAT are ca atribute:

EXPONAT

id#

nume\_loc

data\_publicare

descriere

* id, variabila de tip intreg, care reprezinta id-ul exponatului
* nume\_exponat, variabila de tip caracter, care reprezinta numele exponatului
* data\_publicare, data de tip intreg care reprezinta data la care a fost publicat exponatul
* descriere, data de tip intreg in care se descrie exponatul

|  |  |  |  |
| --- | --- | --- | --- |
| id | nume\_loc | data\_publicare | descriere |
| PK | NOT NULL | - | - |

Entitatea AUTOR are ca atribute:

AUTOR

id#

nume\_autor

* id, variabila de tip intreg, care reprezinta id-ul autorului
* nume\_autor, , variabila de tip caracter, care reprezinta numele autorului

|  |  |
| --- | --- |
| id | nume\_autor |
| PK | NOT NULL |

Entitatea SPONSOR are ca atribute:

SPONSOR

id#

nume\_sponsor

* id, variabila de tip intreg, care reprezinta id-ul sponsorului
* nume\_autor,, variabila de tip caracter, care reprezinta numele autorului

|  |  |
| --- | --- |
| id | nume\_sponsor |
| PK | NOT NULL |

Entitatea COORDONATOR are ca atribute:

* id, variabila de tip intreg, care reprezinta id-ul coordonatorului
* nume, variabila de tip caracter, care reprezinta numele coordonatorului

SPONSOR

id#

nume\_sponsor

* numar\_telefon, variabila de tip caracter, care reprezinta numarul de telefon al coordonatorului

|  |  |
| --- | --- |
| id | nume\_sponsor |
| PK | NOT NULL |

Entitatea ORAS, care are ca atribute:

* id, variabila de tip intreg, care reprezinta id-ul coordonatorului
* nume, variabila de tip caracter, care reprezinta numele orasului

ORAS

id#

nume\_oras

capitala

* capitala, variabila booleana care spune daca orasul este capitala sau nu

|  |  |  |
| --- | --- | --- |
| id | nume\_oras | capitala |
| PK | NOT NULL | NOT NULL |

Entitatea STAFF, care are ca atribute:

STAFF

id#

nume\_echipa

rol\_staff

* id, variabila de tip intreg, care reprezinta id-ul staff-ului
* nume\_echipa, variabila de tip caracter, care reprezinta numele echipei
* rol\_staff, variabila de tip caracter care reprezinta rolul echipei

Entitatea ANGAJAT, care are ca atribute:

ANGAJAT

id#

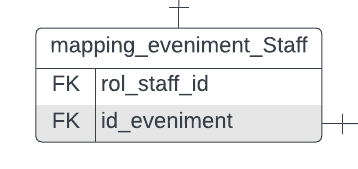
nume\_prenume

salariu

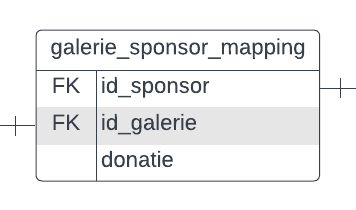
idEchipa

* id, variabila de tip intreg, care reprezinta id-ul unui angajat
* nume\_preniume, variabila de tip intreg care reprezinta numele intreg al unui angajat
* salariu, variabila de tip intreg care reprezinta salariul unui angajat
* idEchipa, variabila de tip intreg care reprezinta echipa din care face parte un angajat

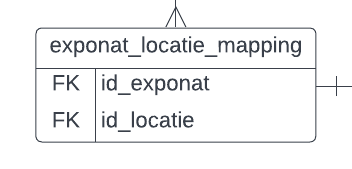
|  |  |  |
| --- | --- | --- |
| id | nume\_oras | capitala |
| PK | NOT NULL | NOT NULL |

Entitatea MAPPING\_EVENIMENT\_STAFF care are ca atribute:

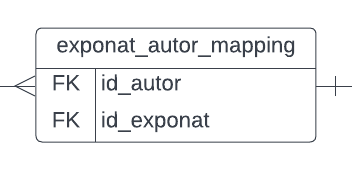
* rol\_staff\_id, variabila de tip intreg
* id\_eveniment, variabila de tip intreg
* leaga echipa stadd de evenimentul la care participa

GALERIE\_SPONSOR\_MAPPING care are atributele:

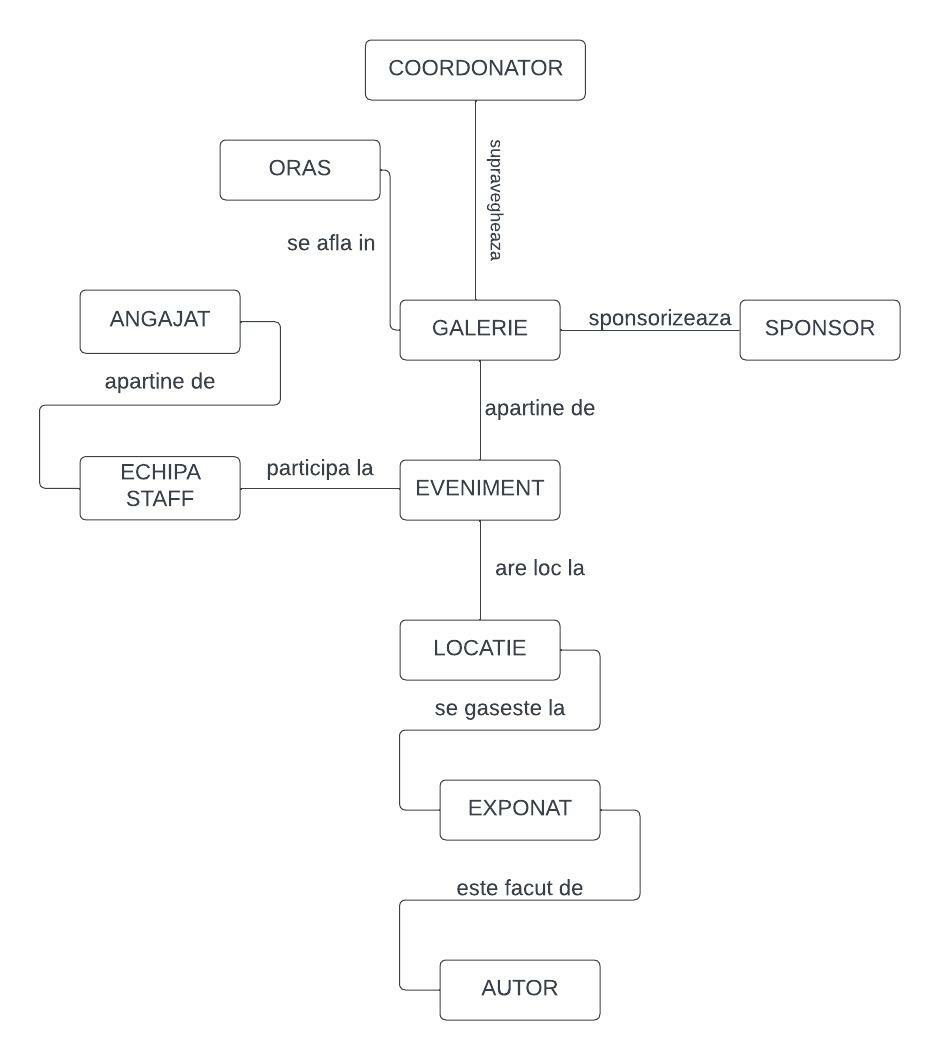
* id\_sponsor, variabila de tip intreg
* id\_galerie, variabila de tip intreg
* leaga galeria de sponsor
* donatie, variabila de tip intreg care reprezinta suma care este donata

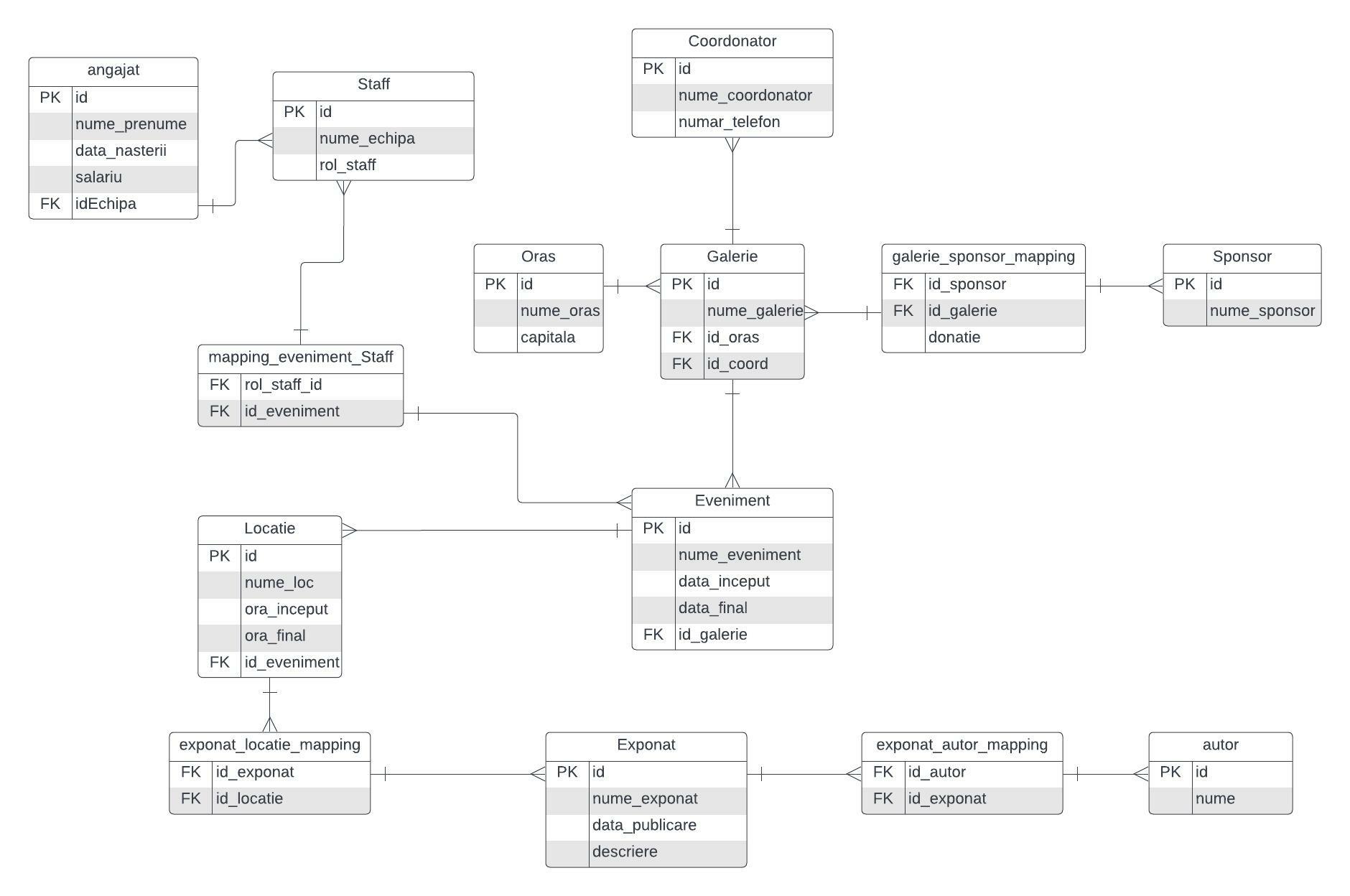
EXPONAT\_LOCATIE\_MAPPING care are atributele:

* id\_exponat, variabila de tip intreg
* id\_locatie, variabila de tip intreg
* leaga exponatul de locatie

EXPONAT\_AUTOR\_MAPPING care are atributele:

* id\_autor, variabila de tip intreg
* id\_exponat, variabila de tip intreg
* leaga exponatul de autor





Exercitiul 8

Galerie(#id, nume\_galerie, id\_oras, id\_eveniment, id\_locatie, id\_coord)

Oras(#id, nume\_oras, capitala)

Coordonator(#id, nume\_coordonator, numar telefon)

Galerie\_Sponsor\_Mapping(id\_sponsor, id\_galerie, donatie)

Sponsor(#id, nume\_sponsor)

Staff(#id, nume\_echipa, rol\_staff)

Angajat(#id, nume\_prenume, data\_nasterii, salariu, idEchipa)

Eveniment\_Staff\_Mapping(rol\_staff\_id, id\_eveniment)

Eveniment(#id, nume\_eveniment, data\_inceput, data\_final)

Locatie(#id, nume\_loc, ora\_inceput, ora\_final, id\_eveniment)

Exponat\_locatie\_mapping(id\_exponat, id\_locatie)

Exponat(#id, nume\_exponat, data\_publicare, descriere)

Exponat\_autor\_mapping(id\_autor, id\_exponat)

Autor(#id, nume)

FN 1

O relaţie este în FN1 dacă fiecărui atribut care o compune îi corespunde o valoare indivizibilă.

Inainte de modificare, tabela EXPOANAT avea atributul AUTOR. Cum un exponat poate avea mai multi autori, am creat tabela AUTOR unde fiecare autor are o cheie primara.

FN 2

O relaţie este în forma normală 2 daca se afla in FN1 si fiecare atribut care nu este cheie primară este dependent de întreaga cheie primară.

La inceput, tabela LOCATIE avea ca si PK id(Locatie) si idEveniment. Acum am facut idEveniment FK si o leaga cu Eveniment.

FN 3

O relație este în a treia formă normală dacă și numai dacă este in FN2 și fiecare atribut care nu este cheie depinde direct de cheia primară.

Initial, salariul unui anagajat tinea cont de Stafful din care facea parte, dar am modificat astfel incat salariul sa fie individual, sa nu depinda de echipa (mutand salariul din tabela STAFF in tabela ANGAJAT).

Exercitiul 10

CREAREA TABELELOR

CREATE TABLE AUTOR (

id int NOT NULL,

nume\_autor varchar(255) NOT NULL

PRIMARY KEY (id)

);

CREATE TABLE COORDONATOR (

id int NOT NULL,

nume\_coordonator varchar(255) NOT NULL,

numar\_telefon varchar(255)

PRIMARY KEY(id)

);

CREATE TABLE SPONSOR (

id int NOT NULL,

nume\_sponsor varchar(255) NOT NULL,

PRIMARY KEY(id)

);

CREATE TABLE EXPONAT (

id int NOT NULL,

nume\_exponat varchar(255) NOT NULL,

data\_publicare date,

descriere text

PRIMARY KEY(id)

);

CREATE TABLE EXPONAT\_AUTOR\_MAPPING (

id int NOT NULL,

id\_autor int FOREIGN KEY REFERENCES AUTOR(id),

id\_exponat int FOREIGN KEY REFERENCES EXPONAT(id),

PRIMARY KEY(id)

);

CREATE TABLE ORAS (

id int NOT NULL,

nume\_oras varchar(255) NOT NULL,

capitala binary NOT NULL

PRIMARY KEY(id)

);

CREATE TABLE STAFF (

id int NOT NULL,

nume\_echipa varchar(255) NOT NULL,

rol\_staff varchar(255) NOT NULL,

PRIMARY KEY(id)

);

CREATE TABLE ANGAJAT (

id int NOT NULL,

nume\_prenume varchar(255) NOT NULL,

data\_nasterii date NOT NULL,

salariu float NOT NULL,

id\_staff int FOREIGN KEY REFERENCES STAFF(id),

PRIMARY KEY(id)

);

CREATE TABLE GALERIE(

id int NOT NULL,

nume\_galerie varchar(255) NOT NULL,

id\_oras int FOREIGN KEY REFERENCES ORAS(id),

id\_coord int FOREIGN KEY REFERENCES COORDONATOR(id),

PRIMARY KEY(id)

);

ALTER TABLE EVENIMENT

ADD idGalerie int NOT NULL FOREIGN KEY REFERENCES GALERIE(id);

ALTER TABLE EVENIMENT

ADD PRIMARY KEY(id);

CREATE TABLE LOCATIE (

id int NOT NULL,

nume\_loc varchar(255) NOT NULL,

ora\_inceput int NOT NULL,

ora\_final int NOT NULL,

idEveniment int NOT NULL FOREIGN KEY REFERENCES EVENIMENT(id),

PRIMARY KEY(id)

);

CREATE TABLE EXPONAT\_LOCATIE\_MAPPING (

id int NOT NULL,

id\_exponat int FOREIGN KEY REFERENCES EXPONAT(id),

id\_locatie int FOREIGN KEY REFERENCES LOCATIE(id),

);

ALTER TABLE EXPONAT\_LOCATIE\_MAPPING

ADD PRIMARY KEY(id);

CREATE TABLE GALERIE\_SPONSOR\_MAPPING (

id int NOT NULL,

id\_galerie int FOREIGN KEY REFERENCES GALERIE(id),

id\_sponsor int FOREIGN KEY REFERENCES SPONSOR(id),

PRIMARY KEY(id)

);

CREATE TABLE EVENIMENT\_STAFF\_MAPPING(

id int NOT NULL,

id\_eveniment int FOREIGN KEY REFERENCES EVENIMENT(id),

id\_staff int FOREIGN KEY REFERENCES STAFF(id),

PRIMARY KEY(id)

);

ALTER TABLE GALERIE\_SPONSOR\_MAPPING

ADD donatie int NOT NULL;

POPULAREA TABELELOR

INSERT INTO AUTOR(id, nume\_autor)

VALUES (2,’PICASSO’),(3,’BRANCUSI’),(4,’Goya’);

INSERT INTO EXPONAT(id,nume\_exponat,data\_publicare,descriere)

VALUES (1,’Noapte Instelata’,’2004-01-22’,’este extrem de frumoasa’ );

INSERT INTO EXPONAT(id,nume\_exponat,data\_publicare,descriere)

VALUES (2,’Irisi’,’1960-04-02’,’’),(3,’Dormitor’,’’,’O opera controversata’),(4,’Cumintenia pamantului’,’’,’O sculptura renumita’),(5,’Coloana infinitului’,’1899-12-12’,’Cea mai renumita opera de arta romaneasca’),(6,’Guernica’,’’,’’),(7,’Autoportreet’,’1978-4-16’,’Portretul autorului’),(8,’Gigantul’,’1879-4-23’,’Opera horror’),(9,’Francisco’,’’,’Cutremurator’),(10,’Maja’,’1888-8-23’,’’);

INSERT INTO EXPONAT\_AUTOR\_MAPPING(id\_autor,id\_exponat,id)

VALUES (1,1,1);

INSERT INTO EXPONAT\_AUTOR\_MAPPING(id\_autor,id\_exponat,id)

VALUES (1,2,2),(1,3,3),(3,4,4),(3,5,5),(2,6,6),(2,7,7),(4,8,8),(4,9,9),(4,10,10);

INSERT INTO LOCATIE(id,nume\_loc,ora\_inceput,ora\_final,idEveniment)

VALUES (1,’MUZEU’,10,20,1),(2,’HALL’,12,23,2),(3,’PRIMARIE’,8,16,3),(4,’MUZEU’,10,20,4),(5,’MUZEU’,10,20,5),(6,’HALL’,10,20,6),(7,’PRIMARIE’,10,20,7),(8,’MUZEU’,10,20,8);

SELECT \* FROM LOCATIE

INSERT INTO EXPONAT\_LOCATIE\_MAPPING(id,id\_exponat,id\_locatie)

VALUES (1,1,1),(2,2,1),(3,1,2),(4,3,4),(5,10,1),(6,7,8),(7,4,5),(8,9,5),(9,8,6),(10,5,4)

SELECT \* FROM EXPONAT

INSERT INTO EVENIMENT(id,idGalerie,data\_final,data\_inceput,nume\_eveniment)

VALUES (1,1,’2012-12-14’,’2022-12-12’,’Partea 1’),(2,1,’2022-12-20’,’2022-12-25’,’Partea 2’),(3,2,’2002-3-12’,’2002-3-1’,’Can you believe?’),(4,3,’2012-6-14’,’2012-6-12’,’Wait’),(5,3,’2012-12-19’,’2012-12-15’,’Hold on’),(6,3,’2013-1-3’,’2013-12-6’,’You may go’),(7,4,’2020-4-7’,’6-4-2020’,’Flowers’),(8,4,’2022-5-7’,’2022-5-6’,’Trees’)

INSERT INTO ORAS(id,nume\_oras,capitala)

VALUES (1,’Bucuresti’,1),(2,’Iasi’,0),(3,’Paris’,1)

INSERT INTO ORAS(id,nume\_oras,capitala)

VALUES(4,’Vaslui’,0)

INSERT INTO COORDONATOR(id,nume\_coordonator,numar\_telefon)

VALUES (1,’Ionesco’,’0788888888’),(2,’Popesco’,’’),(3,’Boiesco’,’23984797132’)

INSERT INTO COORDONATOR(id,nume\_coordonator,numar\_telefon)

VALUES (4,’Crico’,’’)

INSERT INTO GALERIE(id,nume\_galerie,id\_coord,id\_oras)

VALUES (1,’O ventura in Bucuresti’,1,1),(2,’Cele mai frumoase arte’,2,2),(3,’De ce nu?’,3,3),(4,’Bonjour!’,3,4)

UPDATE GALERIE

SET id\_oras=3

WHERE id=4;

INSERT INTO STAFF (id,nume\_echipa,rol\_staff)

VALUES (1,’Cleansteam’,’curatare’),(2,’Ghidusii’,’guide’),(3,’Thunder’,’luminile’),(4,’Boddiies’,’securitate’)

Insert into ANGAJAT (id, nume\_prenume,salariu,data\_nasterii,id\_staff)

VALUES (1,’Ion Popa’,1000,’2000-2-23’,1),(2,’Vasile’,1340,’1999-4-2’,1),(3,’Cristi’,2300,’1990-10-25’,1),(4,’Ana’,3000,’2001-12-23’,2),(5,’Ionel’,1500,’1964-2-13’,2),(6,’Dolores’,2000,’2000-2-2’,3),(7,’Mitica’,1400,’1970-9-8’,4),(8,’Flaco’,2000,’2002-8-7’,4)

SELECT \* FROM SPONSOR;

INSERT INTO EVENIMENT\_STAFF\_MAPPING(id, id\_eveniment,id\_staff)

VALUES (1,1,4),(2,2,3),(3,3,3),(4,1,3),(5,3,1),(6,5,2),(7,4,3),(8,8,3),(9,4,2),(10,7,3)

INSERT INTO SPONSOR(id, nume\_sponsor)

VALUES (1,’Luca’),(2,’Microsoft’),(3,’Brita’)

INSERT INTO GALERIE\_SPONSOR\_MAPPING(id, id\_galerie,id\_sponsor,donatie)

VALUES (1,1,1,100),(2,1,2,322),(3,2,3,900),(4,2,1,89),(5,3,2,34),(6,3,1,200),(7,4,1,98),(8,1,3,324),(9,4,2,432),(10,4,3,234)

EXERCITIUL 11

SELECT nume\_exponat,descriere,nume\_autor,nume\_loc,ora\_inceput,ora\_final FROM EXPONAT\_AUTOR\_MAPPING

INNER JOIN EXPONAT ON EXPONAT\_AUTOR\_MAPPING.id\_exponat = EXPONAT.id

INNER JOIN AUTOR ON EXPONAT\_AUTOR\_MAPPING.id\_AUTOR = AUTOR.id

INNER JOIN EXPONAT\_LOCATIE\_MAPPING ON EXPONAT\_AUTOR\_MAPPING.id\_exponat = EXPONAT\_LOCATIE\_MAPPING.id

INNER JOIN LOCATIE ON EXPONAT\_LOCATIE\_MAPPING.id\_locatie= LOCATIE.ID

WHERE LOCATIE.ora\_final = (SELECT MAX(ora\_final) FROM LOCATIE)



Query-ul afiseaza exponatele care sunt afisate pentru cea mai lunga perioada de timp

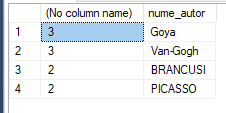
SELECT COUNT(AUTOR.nume\_autor), nume\_autor

FROM EXPONAT\_AUTOR\_MAPPING

INNER JOIN AUTOR ON EXPONAT\_AUTOR\_MAPPING.id\_AUTOR = AUTOR.id

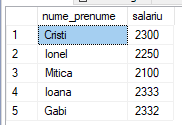
GROUP BY nume\_autor

ORDER BY COUNT(AUTOR.nume\_autor) DESC;



Query-ul afiseaza numarul cate exponate are fiecare autor

SELECT nume\_prenume,salariu FROM ANGAJAT WHERE (YEAR(GETDATE())-YEAR(data\_nasterii) >30)



Query-ul afiseaza persoanele care au varsta mai mica de 30 de ani.

SELECT donatie,

CASE

WHEN donatie > 150 THEN 'donatie mica'

ELSE 'donatie mare'

END as TXT

from GALERIE\_SPONSOR\_MAPPING

WITH aux(salmed) as

(SELECT avg(salariu) from ANGAJAT)

SELECT id,nume\_prenume, salariu

FROM ANGAJAT, aux

WHERE ANGAJAT.salariu > aux.salmed;

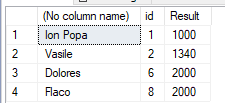
WITH aux(salmed) as

(SELECT avg(salariu) from ANGAJAT)

SELECT TRIM( '!,.-' FROM nume\_prenume), id, salariu AS Result

FROM ANGAJAT, aux

WHERE ANGAJAT.salariu < aux.salmed;



Query-ul afiseaza angajatii care au salariul mai mic decat media celorlalti angajati

SELECT COUNT(AUTOR.nume\_autor), LEFT(nume\_autor,1)

FROM EXPONAT\_AUTOR\_MAPPING

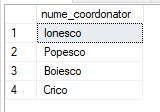
INNER JOIN AUTOR ON EXPONAT\_AUTOR\_MAPPING.id\_AUTOR = AUTOR.id

GROUP BY nume\_autor

ORDER BY COUNT(AUTOR.nume\_autor) DESC;

Query-ul afiseaza numarul de expoante a fiecarui autor si initiala acestora

SELECT nume\_coordonator FROM COORDONATOR WHERE numar\_telefon IS NOT NULL;



Query-ul afiseaza coordonatorii care au numar de telefon inregistrat.

Query-ul efectueaza mai multe joinuri.

SELECT nume\_galerie, nume\_sponsor,nume\_oras,nume\_coordonator,capitala from GALERIE\_SPONSOR\_MAPPING

INNER JOIN GALERIE ON GALERIE\_SPONSOR\_MAPPING.id\_galerie = GALERIE.id

INNER JOIN SPONSOR ON GALERIE\_SPONSOR\_MAPPING.id\_sponsor = SPONSOR.id

INNER JOIN ORAS ON GALERIE.id\_oras=ORAS.id

INNER JOIN COORDONATOR ON GALERIE.id\_coord=COORDONATOR.id

WHERE capitala = 1



Query-ul afiseaza angajatii care lucreaza la un eveniment la care se prezinta o opera care apartine lui Van-Gogh si se afla intr-o capitala

SELECT nume\_prenume FROM EVENIMENT\_STAFF\_MAPPING

INNER JOIN STAFF on EVENIMENT\_STAFF\_MAPPING.id\_staff= STAFF.id

INNER JOIN ANGAJAT ON STAFF.id=ANGAJAT.id\_staff

INNER JOIN EVENIMENT on EVENIMENT\_STAFF\_MAPPING.id\_eveniment = EVENIMENT.id

INNER JOIN LOCATIE ON LOCATIE.idEveniment=EVENIMENT.id

INNER JOIN EXPONAT\_LOCATIE\_MAPPING ON LOCATIE.id=EXPONAT\_LOCATIE\_MAPPING.id\_locatie

INNER JOIN EXPONAT ON EXPONAT.id=EXPONAT\_LOCATIE\_MAPPING.id\_exponat

INNER JOIN EXPONAT\_AUTOR\_MAPPING ON EXPONAT.id=EXPONAT\_AUTOR\_MAPPING.id\_exponat

INNER JOIN AUTOR ON EXPONAT\_AUTOR\_MAPPING.id\_autor=AUTOR.id

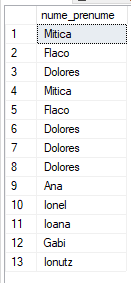
INNER JOIN GALERIE ON GALERIE.id=EVENIMENT.idGalerie

INNER JOIN ORAS ON GALERIE.id\_oras=ORAS.id

INNER JOIN COORDONATOR ON GALERIE.id\_coord=COORDONATOR.id

WHERE [ORAS].capitala = 1 AND [nume\_autor]='Van-Gogh'

ORDER BY [EVENIMENT].id



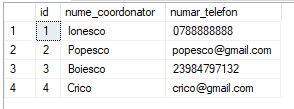
Query-ul inlocuieste numarul de telefon cu o adresa de email in caz ca numarul de telefon nu este disponibil.

UPDATE COORDONATOR

SET numar\_telefon=CONCAT(LOWER(nume\_coordonator),'@gmail.com')

WHERE numar\_telefon IS NULL

SELECT \* FROM COORDONATOR



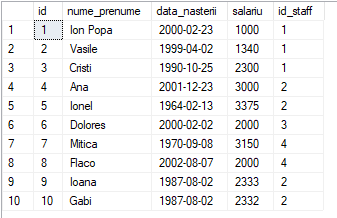
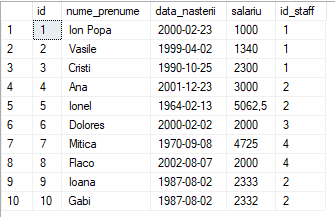
EXERCITIUL 12

Query-ul adauga cu 50% la salariul angajatilor care au maim ult de 50 de ani

UPDATE ANGAJAT

SET salariu=salariu+salariu\*0.5

WHERE (YEAR(GETDATE())-YEAR(data\_nasterii)>50)

Query-ul concediaza angajatii care nu au implinit inca 18 ani.

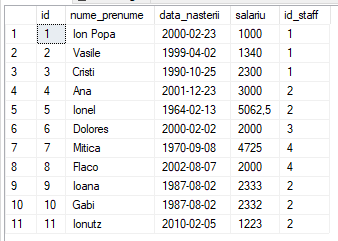
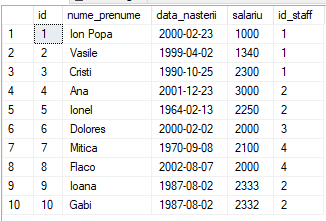
Insert into ANGAJAT (id, nume\_prenume,salariu,data\_nasterii,id\_staff)

VALUES (11,’Ionutz’,1223,’2010-2-5’,2)

SELECT \* FROM ANGAJAT

DELETE FROM ANGAJAT WHERE (YEAR(GETDATE())-YEAR(data\_nasterii) <18);

SELECT \* FROM ANGAJAT



EXERCITIUL 13

Query-ul face un sequence pentru a facilita introducerea de angajati.

CREATE SEQUENCE contor

start with 9

increment by 1

minvalue 0

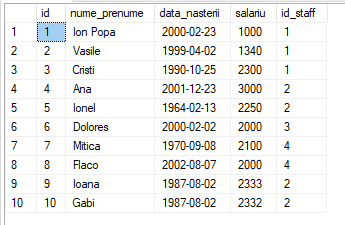
maxvalue 1000

cycle;

INSERT into ANGAJAT VALUES(NEXT VALUE FOR contor,'Ioana','1987-8-2',2333,2);

INSERT into ANGAJAT VALUES(NEXT VALUE FOR contor,'Gabi','1987-8-2',2332,2);

SELECT \* FROM ANGAJAT



EXERCITIUL 14

CREATE VIEW AUTORIxEXPONAT as

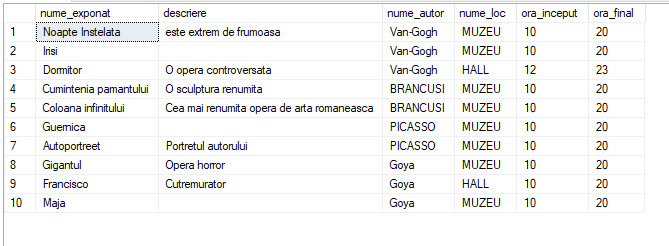
SELECT nume\_exponat,descriere,nume\_autor,nume\_loc,ora\_inceput,ora\_final FROM EXPONAT\_AUTOR\_MAPPING

INNER JOIN EXPONAT ON EXPONAT\_AUTOR\_MAPPING.id\_exponat = EXPONAT.id

INNER JOIN AUTOR ON EXPONAT\_AUTOR\_MAPPING.id\_AUTOR = AUTOR.id

INNER JOIN EXPONAT\_LOCATIE\_MAPPING ON EXPONAT\_AUTOR\_MAPPING.id\_exponat = EXPONAT\_LOCATIE\_MAPPING.id

INNER JOIN LOCATIE ON EXPONAT\_LOCATIE\_MAPPING.id\_locatie= LOCATIE.ID



SELECT nume\_exponat FROM AUTORIxEXPONAT where nume\_autor='Van-Gogh'

LMD este folosit, intr-un view, pentru adaugarea, modificarea sau stergea de linii. Nu putem modifica/sterge un view daca acesta contine primary keys.

EXERCITIUL 16

SELECT \* FROM EXPONAT\_AUTOR\_MAPPING

FULL OUTER JOIN EXPONAT ON EXPONAT\_AUTOR\_MAPPING.id = EXPONAT.id

FULL OUTER JOIN AUTOR ON EXPONAT\_AUTOR\_MAPPING.id\_autor=AUTOR.id

FULL OUTER JOIN EXPONAT\_LOCATIE\_MAPPING ON EXPONAT\_AUTOR\_MAPPING.id\_exponat=EXPONAT\_LOCATIE\_MAPPING.id\_exponat

FULL OUTER JOIN LOCATIE ON LOCATIE.id=EXPONAT\_LOCATIE\_MAPPING.id

FULL OUTER JOIN GALERIE ON LOCATIE.id=GALERIE.id

