**Universitatea „Ovidius” din Constanţa**

**Facultatea de Matematică ṣi Informatică**

**Disciplina: Baze de date**

**Specializarea: Matematică-Informatică**

**Anul III**

**PROIECT DE CURS**

**Coordonator, Studenta,**

**Lect.dr. Băutu Elena Alecu Cristina**

**BAZĂ MUZICALĂ**

Entităţile modelează clase de obiecte concrete sau abstracte despre care se colectează informaţii, au existenţă independentă ṣi pot fi identificate în mod unic. Acestea se reprezintă grafic printr-un dreptunghi în interiorul căruia este scris numele entităţii.

Exemple de entităţi: ALBUME, ARTIȘTI, INSTRUMENTE etc.

Atributele modelează proprietăţi distincte ale entităţilor. De exemplu, entitatea MELODII are ca atribute *id\_melodie* (cheie primară), *nume* și *timp redare*. În procesul de modelare vor fi luate în considerare doar acele proprietăţi ale entităţilor care sunt semnificative pentru aplicaţia respectivă. În general, un atribut se reprezintă grafic printr-o elipsă, în interiorul căreia apare numele atributului, fiind legată printr-un segment de entitatea corespunzătoare. În reprezentarea mea, atributele sunt trecute sub entităţile corespunzătoare.

Asociaţiile reprezintă o conexiune între entităţi (chiar ṣi între asociaţii, sau entităţi de acelaṣi tip). De exemplu, între entităţile ALBUME și MELDII există o asociere one-to-many (1-la-n).Conține ce descrie corespondența dintre albume și melodii(unui album i se poate asocia mai multe melodii, dar unei melodii i se poate asocia un singur album). Asociaţiile se reprezintă grafic prin romburi, fiind conectate prin segmente la entităţile pe care le leagă, ṣi având scris în interior numele asocierii.

Pentru entităţile A ṣi B, putem avea una din următoarele tipuri de asocieri R:

* One-to-one (1-la-1): O entitate din A este asociată cel mult unei entităţi din B şi o entitate din B este asociată cel mult unei entităţi din A (de exemplu, între entităţile ADRESĂ și NUMĂR TELEFON).
* One-to-many (1-la-n): O entitate din A este asociată cu oricâte entităţi din B şi o entitate din B este asociată cel mult unei entităţi din A (de exemplu, între entităţile ALBUME și MELODII).
* Many-to-one (n-la-1): O entitate din A este asociată cel mult unei entităţi din B şi o entitate din B este asociată oricâtor (0 sau mai multe) entităţi din A (de exemplu, între entităţile ARTIST ṣi INSTRUMENTE).
* Many-to-many (n-la-n): O entitate din A este asociată cu oricâte entităţi din B şi o entitate din B este asociată cu oricâte entităţi din A (de exemplu, între entităţile MELODII ṣi HIT).

Baza de date muzicală stochează detalii ale unei biblioteci muzicale personale și poate fi utilizată pentru a vă gestiona colecția de MP3.

Deoarece această bază de date este pentru o colecție personală, stochează relațiile dintre artiști, albume, melodii,hituri, instrumente, fanii și adresele artiștilor.

Mai întâi întocmim o listă clară de cerințe pentru baza noastră de date:

Colecția este formată din albume.

Fiecare album care este înregistrat la casa de discuri are un titlu, un drept de autor

data, un format (de exemplu, CD sau MC) și un identificator de album.

Fiecare album are un număr de melodii, dar nicio melodie nu poate apărea pe mai multe

decât un album.

Un album este realizat de exact un artist.

Un artist realizează unul sau mai multe albume.

Fiecare artist care înregistrează o melodie are un id\_nume, un nume, o adresă și

un număr de telefon. Artiști au una sau mai multe adrese și

o adresă nu are mai mult de un telefon.

Fiecare instrument care este folosit în melodiile înregistrate are

un nume (de ex.chitară, sintetizator, flaut) și

o cheie muzicală (de exemplu, C,B‐plat, Eflat).

Un album conține una sau mai multe melodii.

Artiștii, albumele și melodiile au fiecare un nume.

Fiecare melodie este pe exact un album.

Fiecare melodie are o lungime de timp, măsurată în secunde.

Fiecare melodie are exact un artist.

Fiecare melodie este interpretată de unul sau mai mulți muzicieni și

un artist poate a executa o serie de cântece.

Fiecare artist poate cânta la mai multe instrumente și

un instrument dat poate fi interpretat de mai mulți muzicieni.

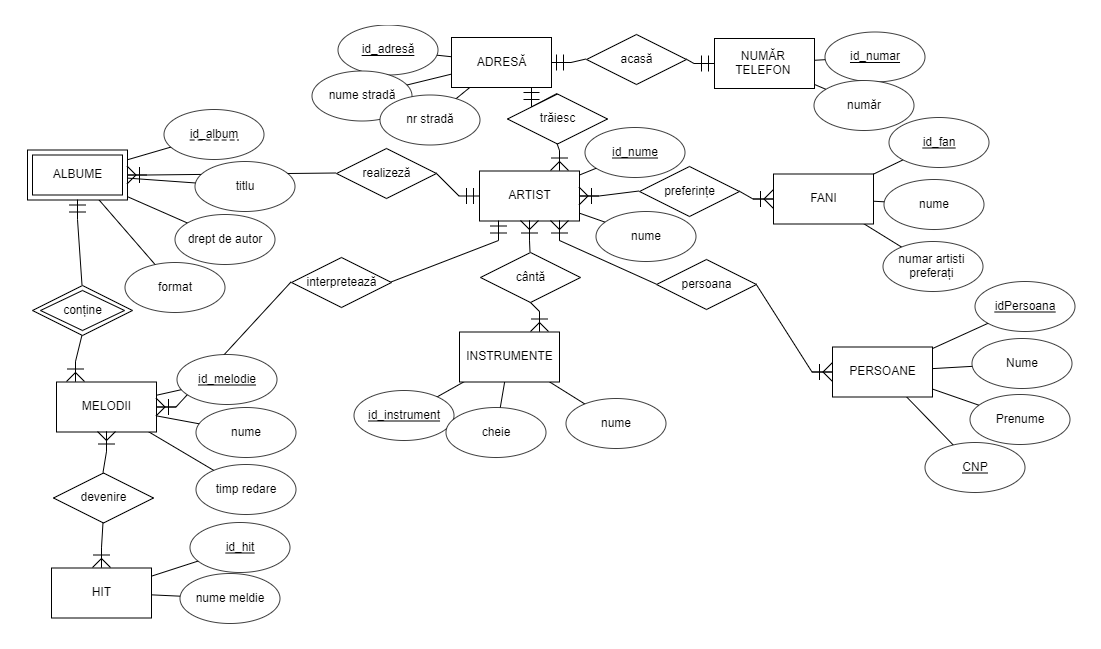
Fiecare artist are unul sa mai mulți fanii, iar fiecare fiecare poate fi fan pentru unul sau mai mulți artiștii.

Fiecare fan are un id\_fan, nume și un număr de artiști preferați.

Fiecare melodie poate deveni un hit, iar hitul poate fi orice melodie. Despre hit cunoaștem un id\_hit, nume.

Când este redată o melodie, data și ora la care a început redarea (până la cea mai apropiată secundă) trebuie înregistrate; aceasta este folosită pentru a raporta când a fost redată ultima melodie, precum și de câte ori a fost redată muzica unui artist, dintr-un album sau o piesă.

Structura bazei de date poate fi vizualizată prin intermediul următoarei diagrame Entităţi – Asociaţii (DEA):



Pentru proiectarea logică a bazei de date vom folosi Modelul Relaţional al Datelor (MRD). Modelul relațional reprezintă baza de date ca o colecție de relații, fiecăreia fiindu-i asociat un nume unic. O relație poate fi vazută ca un tabel de valori, fiecare linie din tabel reprezentând o colecție de valori interconectate.

Într-o bază de date relaţională, o relaţie este privită ca un tabel bidimensional cu toate valorile atomice. Pentru orice relaţie avem următoarele proprietăţi:

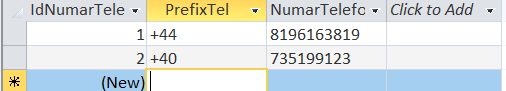
* Numele coloanei unei relaţii se numeşte atribut.
* Numărul de atribute ale unei relaţii reprezintă gradul sau aritatea relaţiei.
* O linie a tabelului se numeşte tuplu sau înregistrare.
* Numărul tuplurilor dintr-o relaţie reprezintă cardinalul relaţiei.

Pentru a face trecerea de la DEA la MRD trebuie să ţinem cont de următoarele reguli:

* Entitățile (dreptunghi) devin tabele (relații);
* Fiecărei relații i se adaugă cheia surogat de tip autonumber;
* Atributele entităților (conţinutul dreptunghiurilor) devin atribute ale relațiilor respective;
* Funcțiile (sgmente/săgeți) devin atribute în relația din care pleacă;
* Pentru fiecare funcție se adaugă o dependență de incluziune (cheie străină) corespunzătoare;
* Asociațiile devin relații, cu câte o cheie străină către relațiile suport.

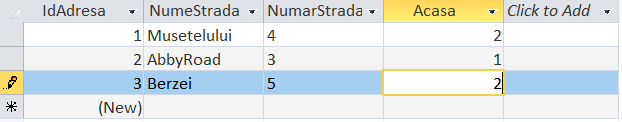
Pe baza regulilor descrise anterior, entitatea NUMĂR TELEFON devine un tabel cu trei atribute:

* idNumărTelefon: cheie-primară, de tip AutoNumber, long integer, cu funcţie de incrementare;
* PrefixTel: de tip long(4), UNIQUE not null.
* NumărTelefon :de tip long, UNIQUE not null



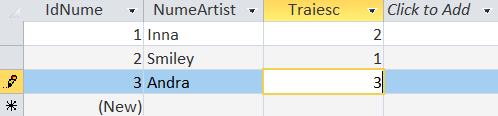
Entitatea ADRESĂ devine, de asemenea, un tabel cu următoarele atribute:

* idAdresă: cheie-primară, de tip AutoNumber, long integer, cu funcţie de incrementare;
* NumeStradă :de tip varchar(25) ,UNIQUE NOT NULL
* NumărStradă :de tip long ,UNIQUE NOT NULL
* Acasă : de tip long, fiind cheie străină pentru tabelul NUMĂR TELEFON cu legătură la câmpul idNumarTelefonA.



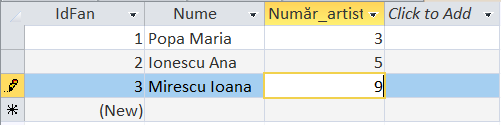
Entitatea ARTIST se transformă într-un tabel cu trei atribute:

* idNume: cheie-primară, de tip AutoNumber, long integer, cu funcţie de incrementare;
* NumeArtist :de tip varchar(25) ,UNIQUE NOT NULL
* Trăiesc : de tip long, fiind cheie străină pentru tabelul ADRESĂ cu legătură la câmpul id\_tip.



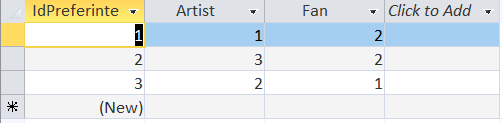
Entitatea FANI se transformă într-un tabel cu trei atribute:

* idFan: cheie-primară, de tip AutoNumber, long integer, cu funcţie de incrementare;
* Nume :de tip varchar(30) ,UNIQUE NOT NULL
* Număr\_artiști\_preferați :de tip long .



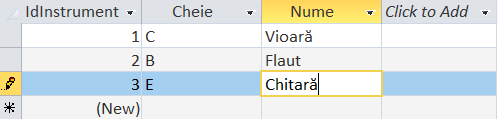
Entitatea PREFERINȚE se transformă într-un tabel cu trei atribute:

* idPreferințe: cheie-primară, de tip AutoNumber, long integer, cu funcţie de incrementare;
* Artist :de tip long, UNIQUE NOT NULL, fiind cheie străină pentru ARTIST cu legătură la câmpul idArtist.
* Fan:de tip long UNIQUE NOT NULL fiind cheie străină pentru tabelul FANI cu legătură la câmpul idFan.



Entitatea INSTRUMENTE se transformă într-un tabel cu trei atribute:

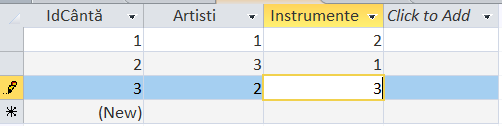
* idInstrument: cheie-primară, de tip AutoNumber, long integer, cu funcţie de incrementare;
* Cheie:de tip varchar(2), UNIQUE NOT NULL
* Nume:de tip varchar(25), UNIQUE NOT NULL.



Entitatea CÂNTĂ\_1 se transformă într-un tabel cu trei atribute:

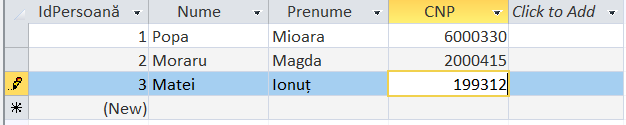
* idCântă: cheie-primară, de tip AutoNumber, long integer, cu funcţie de incrementare;
* Artist :de tip long, UNIQUE NOT NULL, fiind cheie străină pentru tabelul ARTIST cu legătură la câmpul idArtist.
* Instrumente:de tip long UNIQUE NOT NULL, fiind cheie străină pentru tabelul INSTRUMENTE cu legătură la câmpul idInstrument.

.



Entitatea PERSOANE se transformă într-un tabel cu patru atribute:

* idPersoană: cheie-primară, de tip AutoNumber, long integer, cu funcţie de incrementare;
* Nume:de tip varchar(20), UNIQUE NOT NULL
* Prenume:de tip varchar(20), UNIQUE NOT NULL.
* CNP : de tip long ,UNIQUE NOT NULL



Entitatea PERSOANĂ\_13 se transformă într-un tabel cu trei atribute:

* idPersoană: cheie-primară, de tip AutoNumber, long integer, cu funcţie de incrementare;
* Artist:de tip long, UNIQUE NOT NULL, fiind cheie străină pentru tabelul ARTIST cu legătură la câmpul idArtist.
* Persoane:de tip long, UNIQUE NOT NULL, fiind cheie străină pentru tabelul PERSOANE cu legătură la câmpul idPersoană.

