

Proiect baze de date
~ aplicație pentru streaming muzică/podcast-uri~

Spataru Mara-Andreea, grupa 141

Cuprins

1. Descrierea modelului real, a utilității acestuia și a regulilor de funcționare.	3
2. Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului.	3
3. Descrierea entităților, incluzând precizarea cheii primare	3
4. Descrierea relațiilor, incluzând precizarea cardinalității acestora.	4
5. Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicite, valori posibile ale atributelor.	5
6. Realizarea diagramei entitate-relație corespunzătoare descrierii de la punctele 3-5.	10
7. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6.	11
8. Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7.	11
9. Realizarea normalizării până la forma normală 3 (FN1-FN3).	12
10. Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele.	15
11. Crearea tabelelor în SQL și inserarea de date coerente în fiecare dintre acestea.	16
12. 5 cereri SQL complexe	44
13. Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri.	51
14. Crearea unei vizualizări complexe. Operații LMD	56
15. Analiza top-n.	57
16. Optimizarea unei cereri.	60

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

Tema aleasă pentru modelul de date este o platforma de streaming pentru muzică și podcast-uri. Utilitatea modelului este dată de posibilitatea de a folosi datele în scopul sugestiilor personalizate pentru fiecare dintre utilizatori, îmbunătățindu-le experiența, așa cum se întâmplă cu orice aplicație în momentul actual. Datele pot fi folosite pentru a crea clasamente, statistici, etc.

Regulile de funcționare sunt bine definite și asigură integritatea datelor. Funcționalitățile bazei de date includ crearea, citirea, actualizarea și ștergerea înregistrărilor, precum și posibilitatea de a schimba parola utilizatorilor și de a adăuga melodii sau albume la playlist-urile acestora. Această bază de date oferă un cadru solid pentru dezvoltarea unei platforme de streaming pentru muzică.

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

Modelul respectă anumite restricții și reguli:

- Orice entitate de tip media (Melodie, Album, Playlist, Podcast, Episod) trebuie să aibă un titlu.
- Orice melodie trebuie asociată unui artist și trebuie să aibă o dată de lansare.
- Orice album trebuie să aibă măcar o melodie, să fie asociat unui artist, să aibă un an de lansare și un preț.
- Un utilizator trebuie să aibă un unic nume_utilizator care să aibă asociată o unică parolă.
- Un utilizator poate avea mai multe playlist-uri/albume asociate.
- Un episod de podcast poate aparține unui singur podcast.
- O melodie sau un episod de podcast pot fi ascultate de mai multe ori de un utilizator.
- Un utilizator nu poate avea de mai multe ori aceeași melodie în același playlist.

3. Descrierea entităților, incluzând precizarea cheii primare

Pentru modelul de date ales, entitățile sunt: ARTIST, ALBUM, MELODIE, PLAYLIST, UTILIZATOR, EPISOD, PODCAST. Voi prezenta entitățile, precizând cheia primară pentru fiecare dintre ele.

ARTIST = persoană care are posibilitatea de a încărca melodii și albume în aplicație pentru a fi ascultate de utilizatori. Această entitate are drept cheie primară artist_id.

ALBUM = colecție de melodii încărcată de un artist. Această entitate are cheia primară dată de album_id.

MELODIE = compoziție muzicală cântată de un artist înregistrat în aplicație. Cheia primară a acestei entități este melodie_id.

PLAYLIST = colecție de melodii de la diferiți artiști, creată de un utilizator. Cheia primară a acestei entități este playlist_id.

UTILIZATOR = persoană care utilizează această aplicație prin intermediul contului pe care îl creează. Cheia primară este utilizator_id.

EPISOD = secțiune a unui podcast. Cheia primară este episod_id.

PODCAST = emisiune care se desfășoară, de cele mai multe ori, doar în format audio, este formată din mai multe episoade. Cheia primară este podcast_id.

4. Descrierea relațiilor, incluzând precizarea cardinalității acestora.

Relațiile pentru modelul de date ales poartă nume sugestive pentru a fi mai ușor de înțeles conținutul pe care îl exprimă și entitățile pe care le unește.

MELODIE_in_PLAYLIST_UTILIZATOR = relație de tip 3 ce leagă entitățile MELODIE, PLAYLIST și UTILIZATOR, folosită pentru a verifica apartenența unei melodii la un playlist al unui utilizator, dar și pentru a găsi toate playlist-urile în care se află o anumită melodie.

ARTIST_cântă_MELODIE = relație de tip many-to-many care leagă entitățile ARTIST și MELODIE, și exprimă apartenența unei melodii la un artist. Relația are cardinalitate maximă m:n și cardinalitate minimă 1:0.

ARTIST_lansează_ALBUM = relație de tip many-to-many care leagă entitățile ARTIST și ALBUM și exprimă apartenența unui album la un artist. Relația are cardinalitate minimă 1:0 și cardinalitate maximă m:n.

UTILIZATOR_urmeaște_ARTIST = relație de tip many-to-many dintre entitățile UTILIZATOR și ARTIST, și exprimă legătura dintre cele două, dacă un utilizator urmărește un artist, va putea găsi mai ușor lansările acestuia. Relația are cardinalitate minimă 0:0 și cardinalitate maximă m:n.

UTILIZATOR_apreciază_ALBUM = relație de tip many-to-many care leagă entitățile UTILIZATOR și ALBUM, și exprimă legătura dintre acestea, dacă un utilizator apreciază un album, îl va putea accesa mai ușor. Relația are cardinalitate minimă 0:0 și cardinalitate maximă m:n.

UTILIZATOR_apreciază_PLAYLIST = relație de tip many-to-many care leagă entitățile UTILIZATOR și PLAYLIST și exprimă legătura dintre acestea, dacă un utilizator apreciază un playlist, îl va putea accesa mai ușor în biblioteca sa. Relația are cardinalitatea minimă 0:0 și cardinalitatea maximă m:n.

ISTORIC_MELODII = relație de tip many-to-many care leagă entitățile UTILIZATOR și MELODIE și exprimă legătura dintre acestea, dacă un utilizator ascultă o melodie, aceasta va apărea în istoricul său. Relația are cardinalitatea minimă 0:0 și cardinalitatea maximă m:n.

PODCAST_ISTORIC = relație de tip many-to-many care leagă entitățile UTILIZATOR și EPISOD și exprimă legătura dintre acestea (istoricul de ascultare al episoadelor de podcast). Relația are cardinalitatea minimă 0:0 și cardinalitatea maximă m:n.

MELODIE_apartine_ALBUM = relație de tip many-to-one care leagă entitățile MELODIE și ALBUM și reprezintă din ce album face parte o melodie. Relația are cardinalitatea minimă 1:0 și cardinalitatea maximă n:1.

EPISOD_apartine_PODCAST = relație de tip many-to-one care leagă entitățile EPISOD și PODCAST și reprezintă din ce podcast face parte respectivul episod. Relația are cardinalitatea minimă 0:1 și cardinalitatea maximă n:1.

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

Entitatea UTILIZATOR are ca atribute:

utilizator_id = variabilă de tip întreg, care reprezintă id-ul unui utilizator.

nume_utilizator = variabila de tip caracter, de lungime maximă 50, nenulă, unică, care reprezintă numele pe care ceilalți utilizatori îl văd în aplicație.

parola = variabilă de tip caracter, de lungime maximă 128, nenulă, care reprezintă parola cu care utilizatorul se autentifică în cont.

email = variabilă de tip caracter, de lungime maximă 150, care reprezintă adresa de e-mail a utilizatorului.

data_nastere = variabilă de tip dată calendaristică, nenulă, care reprezintă data nașterii utilizatorului.

tara = variabilă de tip caracter, de lungime maximă 2, nenulă, care reprezintă codul țării unui utilizator, conform Anexei A6 europene, "Codurile țărilor și teritoriilor".

Entitatea PLAYLIST are ca atribute:

playlist_id = variabilă de tip întreg, care reprezintă id-ul unei melodii.

titlu = variabilă de tip caracter, de lungime maximă 255, nenulă, care reprezintă titlul unui playlist.

data_creatie = variabilă de tip dată calendaristică, nenulă, care reprezintă data la care s-a creat playlist-ul.

Entitatea MELODIE are ca atribute:

melodie_id = variabilă de tip întreg, care reprezintă id-ul unei melodii.

titlu = variabilă de tip caracter, nenulă, de lungime maximă 255, care reprezintă titlul unei melodii.

descriere = variabilă de tip caracter, de lungime maximă 255, care reprezintă o scurtă descriere pe care melodia o poate avea.

durata = variabilă de tip număr, care are maxim 3 cifre, și acceptă o precizie de 0, deci este număr natural, nenul, care reprezintă durata unei melodii exprimată în secunde.

data_lansare = variabilă de tip dată calendaristică, nenulă, care reprezintă data lansării melodiei respective.

album_id = variabilă de tip întreg, reprezintă albumul căreia melodia îi aparține. Acesta poate fi NULL, deoarece nu este obligatoriu ca o melodie să facă parte dintr-un album.

artist_id = variabilă de tip întreg, care reprezintă artistul care a lansat melodia. Aceasta nu poate fi NULL.

Entitatea ARTIST are ca atribute:

artist_id = variabilă de tip întreg, care reprezintă id-ul unui artist.

nume = variabilă de tip caracter, nenulă, de lungime maximă 100, care reprezintă numele artistului/trupei.

varsta = variabilă de tip întreg, care poate fi NULL pentru cazul trupelor, și reprezintă vârsta artistului, acolo unde e cazul.

descriere = variabilă de tip caracter, de dimensiune maximă 255, reprezintă o mică descriere pe care artistul/trupa o poate avea.

Entitatea ALBUM are ca atribute:

album_id = variabilă de tip întreg, care reprezintă id-ul unui album.

gen = variabilă de tip caracter, nenulă, de lungime maximă 50, care reprezintă genul muzical al unui album.

pret = variabilă de tip zecimal, cu maxim 10 cifre și 2 zecimale, mai mare decât 0, care reprezintă prețul aferent albumului.

titlu = variabilă de tip caracter, nenulă, de lungime maximă 100, care reprezintă titlul unui album.

data_lansare = variabilă de tip dată calendaristică, nenulă, și reprezintă data lansării albumului.

artist_id = variabilă de tip întreg, care reprezintă artistul asociat albumului. Nu poate fi NULL.

Entitatea PODCAST are ca atribute:

podcast_id = variabilă de tip întreg care reprezintă id-ul unui podcast.

nume = variabilă de tip caracter, nenulă, de lungime maximă 255, care reprezintă numele podcastului.

nr_episoade = variabilă de tip întreg, mai mare decât 0, care reprezintă numărul de episoade pe care podcast-ul îl are.

nume_autor = variabilă de tip caracter, nenulă, de lungime maximă 100, care reprezintă numele autorului podcast-ului.

Entitatea EPISOD are ca atribute:

episod_id = variabilă de tip întreg, care reprezintă id-ul unui episod.

podcast_id = variabilă de tip întreg, care reprezintă podcast-ul din care episodul face parte. Nu poate fi NULL.

titlu = variabilă de tip caracter, nenulă, de lungime maximă 120, care reprezintă titlul episodului.

Relația PODCAST_ISTORIC are ca atribute:

podcast_istoric_id = variabilă de tip întreg, care reprezintă id-ul unei ascultări a unui episod de podcast de către un utilizator.

utilizator_id = variabilă de tip întreg, care reprezintă id-ul utilizatorului care a ascultat episodul. Atributul trebuie să corespundă unei chei primare din tabelul UTILIZATOR.

episod_id = variabilă de tip întreg, care reprezintă id-ul unui episod de podcast. Atributul trebuie să corespundă unei chei primare din tabelul EPISOD.

data_ascultare = variabilă de tip dată calendaristică, nenulă, care reprezintă data de ascultare a episodului de către utilizator.

Relația ISTORIC_MELODII are ca attribute:

istoric_melodii_id = variabilă de tip întreg, nenulă, care reprezintă id-ul unei ascultări a unei melodii de către un utilizator

utilizator_id = variabilă de tip întreg, care reprezintă id-ul unui utilizator. Atributul trebuie să corespundă unei chei primare din tabelul UTILIZATOR.

melodie_id = variabilă de tip întreg, care reprezintă id-ul unei melodii. Atributul trebuie să corespundă unei chei primare din tabelul MELODIE.

data_ascultare = variabilă de tip dată calendaristică, nenulă, care reprezintă data de ascultare a melodiei de către utilizator.

Relația UTILIZATOR_apreciază_PLAYLIST are ca attribute:

utilizator_id = variabilă de tip întreg, care reprezintă id-ul unui utilizator. Atributul trebuie să corespundă unei chei primare din tabelul UTILIZATOR.

playlist_id = variabilă de tip întreg, care reprezintă id-ul unui playlist. Atributul trebuie să corespundă unei chei primare din tabelul PLAYLIST.

Relația UTILIZATOR_apreciază_ALBUM are ca attribute:

utilizator_id = variabilă de tip întreg, care reprezintă id-ul unui utilizator. Atributul trebuie să corespundă unei chei primare din tabelul UTILIZATOR.

album_id = variabilă de tip întreg, care reprezintă id-ul unui album. Atributul trebuie să corespundă unei chei primare din tabelul ALBUM.

Relația UTILIZATOR_urmeaște_ARTIST are ca attribute:

utilizator_id = variabilă de tip întreg, care reprezintă id-ul unui utilizator, Atributul trebuie să corespundă unei chei primare din tabelul UTILIZATOR.

artist_id = variabilă de tip întreg, care reprezintă id-ul unui artist. Atributul trebuie să corespundă unei chei primare din tabelul ARTIST.

Relația ARTIST_lansează_ALBUM are ca attribute:

artist_id = variabilă de tip întreg, care reprezintă id-ul unui artist. Atributul trebuie să corespundă unei chei primare din tabelul ARTIST.

album_id = variabilă de tip întreg, care reprezintă id-ul unui album. Atributul trebuie să corespundă unei chei primare din tabelul ALBUM.

Relația ARTIST_cântă_MELODIE are ca attribute:

artist_id = variabilă de tip întreg, care reprezintă id-ul unui artist. Atributul trebuie să corespundă unei chei primare din tabelul ARTIST.

melodie_id = variabilă de tip întreg, care reprezintă id-ul unei melodii. Atributul trebuie să corespundă unei chei primare din tabelul MELODIE.

Relația MELODIE_in_PLAYLIST_UTILIZATOR are ca attribute:

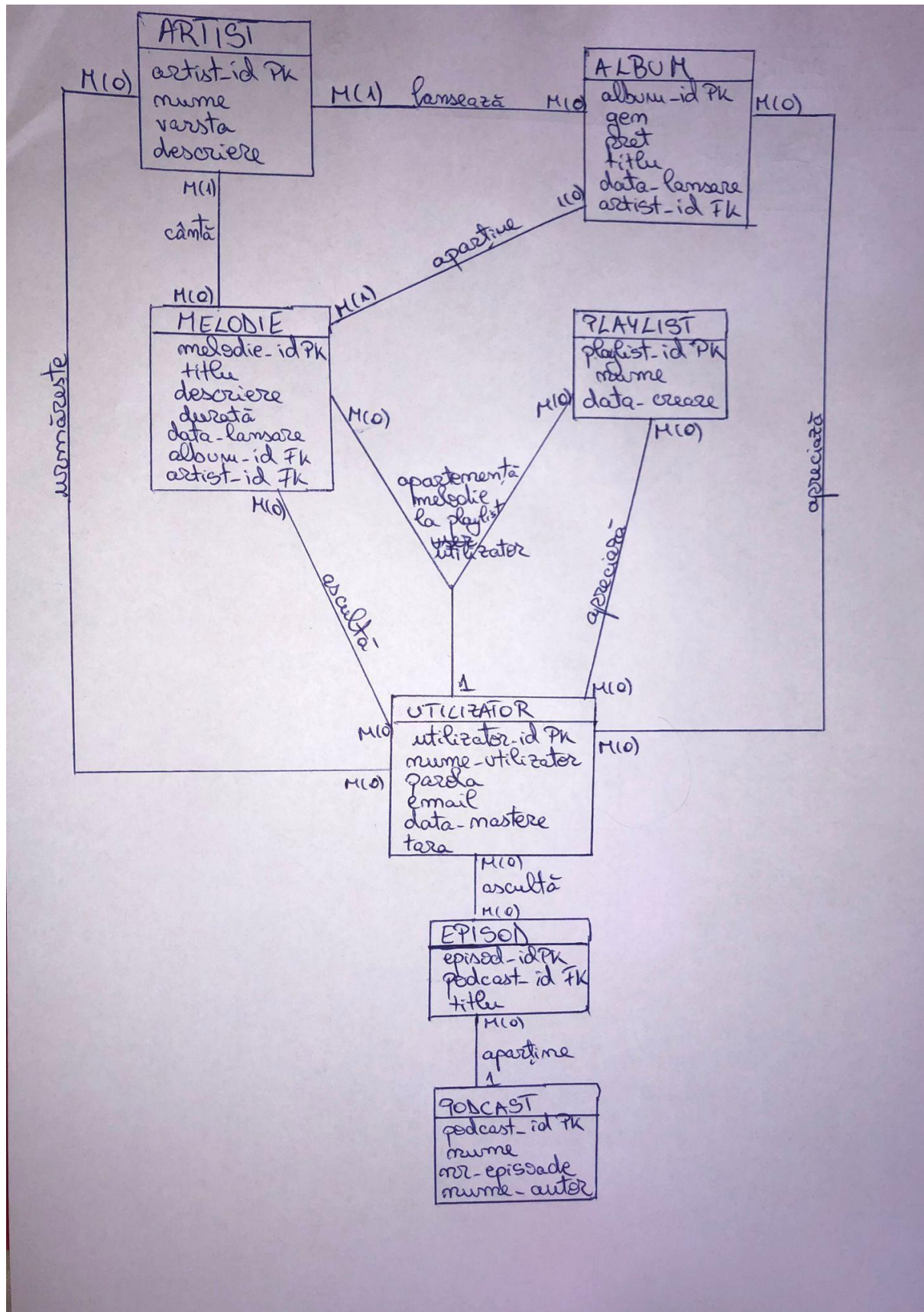
melodie_id = variabilă de tip întreg, care reprezintă id-ul unei melodii. Atributul trebuie să corespundă unei chei primare din tabelul MELODIE.

playlist_id = variabilă de tip întreg, care reprezintă id-ul unui playlist. Atributul trebuie să corespundă unei chei primare din tabelul PLAYLIST.

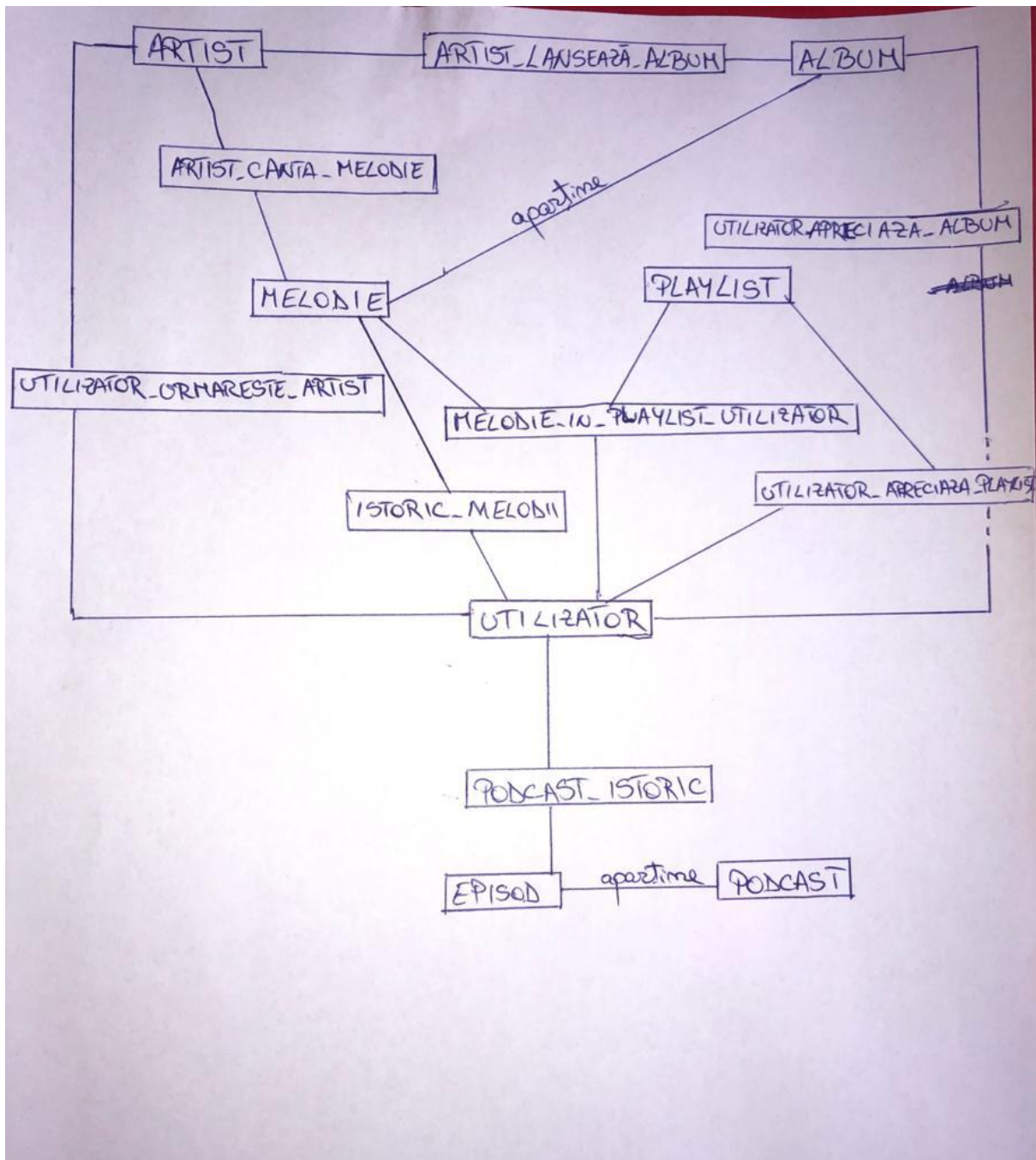
utilizator_id = variabilă de tip întreg, care reprezintă id-ul unui utilizator. Atributul trebuie să corespundă unei chei primare din tabelul UTILIZATOR.

data_adaugare = variabilă de tip dată calendaristică, nenulă, care reprezintă data de adaugare a melodiei în playlist de către utilizator.

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



7. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6.



8. Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7.

UTILIZATOR(utilizator_id(PK), nume_utilizator, parola, email, data_nastere, tara)

EPISOD(episod_id(PK), podcast_id(FK), titlu)

PODCAST(podcast_id(PK), nume, nr_episoade, nume_autor)

ARTIST(artist_id(PK), nume, varsta, descriere)

ALBUM(album_id(PK), gen, pret, titlu, data_lansare, artist_id(FK))

MELODIE(melodie_id(PK), titlu, descriere, durata, data_lansare, album_id(FK), artist_id(FK))

PLAYLIST(playlist_id(PK), titlu, data_creare)

ARTIST_LANSEAZA_ALBUM(artist_id(PK,FK), album_id(PK,FK))

ARTIST_CANTA_MELODIE(artist_id(PK,FK), melodie_id(PK,FK))

UTILIZATOR_URMARESTE_ARTIST(utilizator_id(PK,FK), artist_id(PK,FK))

UTILIZATOR_APRECIAZA_ALBUM(utilizator_id(PK,FK), album_id(PK,FK))

MELODIE_IN_PLAYLIST_UTILIZATOR(melodie_id(PK,FK), playlist_id(PK,FK), utilizator_id(PK,FK), data_adaugare)

UTILIZATOR_APRECIAZA_PLAYLIST(utilizator_id(PK,FK), playlist_id(PK,FK))

ISTORIC_MELODII(istoric_melodii_id(PK), utilizator_id(FK), melodie_id(FK), data_ascultare)

PODCAST_ISTORIC(podcast_istoric_id(PK), utilizator_id(FK), episod_id(FK), data_ascultare)

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

FN1 reprezintă obținerea atributelor atomice, adică attribute care nu mai pot fi divizate. Spre exemplu, dacă o piesă era cântată de doi artiști, iar cei doi apăreau scriși împreună pe coloana artist_id, atunci modelul nu ar mai fi fost în FN1. Cum pentru o melodie cântată de mai mulți artiști, se adaugă o nouă înregistrare pentru fiecare artist, acest atribut nu poate fi descompus mai mult. Iar cum, în general pentru a obține FN1, un atribut de tipul “nume_prenume_persoană” (numele întreg) se descompune în “nume_persoana” și “prenume_persoană”, dar în cazul de față numele artistului este nume de scenă (sau al unei trupe), deci descompunerea nu se poate realiza în acest fel. Astfel modelul de date propus se află deja în FN1.

MELODIE_ID	TITLU	DESCRIERE	DURATA	DATA_LANSARE	ALBUM_ID	ARTIST_ID
11	14Miss Independent	(null)	232	26-AUG-08	(null)	2
12	4My Boo	(null)	223	29-AUG-04	1	1
13	27My Boo	(null)	223	29-AUG-04	(null)	11
14	5No One	(null)	254	09-OCT-07	1	1
15	20One in a Million	(null)	244	14-SEP-10	(null)	2
16	22POWER	(null)	184	04-MAR-23	(null)	6
17	25Searchin	(null)	264	27-JAN-17	(null)	9

Astfel, chiar dacă apare redundanță la nivel de date, modelul se află în FN1, fiecare înregistrare cu numele aceleiași melodii se repetă pentru un număr diferit de artiști care o cântă.

Un tabel se află în FN2 dacă se află în FN1 și fiecare atribut care nu participă la cheia primară este dependent de întreaga cheie primară. Deoarece nu avem attribute care să nu depindă, de fapt, de cheia primară a entității, modelul de date se află deja în FN2.

tabelul ALBUM:

ALBUM_ID	GEN	PRET	DATA_LANSARE	ARTIST_ID	TITLU
1	1 R&B	69.99	04-NOV-16	1	HERE
2	2 R&B	59.99	28-FEB-06	2	In My Own Words
3	3 Pop	45.32	24-JAN-11	3	21
4	4 Indie	39.99	20-NOV-15	3	25
5	5 Pop	40.99	19-NOV-21	3	30
6	6 R&B	55.99	20-MAR-20	4	After Hours
7	7 R&B/Soul	63.99	07-JAN-22	4	Dawn FM
8	8 R&B/Soul	31.99	30-MAR-18	4	My dear Melancholy
9	9 R&B/Soul	99.99	25-NOV-16	4	Starboy
10	10 Pop	90.99	16-SEP-22	5	Girl Of My Dreams (Deluxe)

tabelul MELODIE:

ARTIST_ID	NUME	VA...	DESCRIERE
1	1Alicia Keys	42	American singer, songwriter, and pianist
2	2Ne-Yo	43	American singer, songwriter, actor, dancer, and record producer
3	6Dilja	21	Icelandic singer
4	7Brunette	21	Armenian singer-songwriter
5	8anees	30	American singer, rapper, and songwriter
6	9Sonder	...	Band created by the artist Brent Faiyaz
7	10Brent Fa...	27	American R&B singer
8	3Adele	35	English singer-songwriter
9	4The Weeknd	33	Canadian singer and songwriter
10	5Fletcher	29	American singer and songwriter
11	11Usher	44	American R&B singer

Un tabel se află în FN3 dacă acesta se află în FN2 și nu există dependențe tranzitive. Spre exemplu, dacă în tabelul MELODIE, după id-ul albumului ar fi apărut și numele acestuia, tabelul trebuia descompus, deoarece s-ar fi creat o dependență tranzitivă: **melodie_id** -> **album_id** -> **titlu_album**, însă în tabelul MELODIE salvăm doar id-ul acestuia, iar numele albumului apare doar în tabelul ALBUM.

tabelul MELODIE:

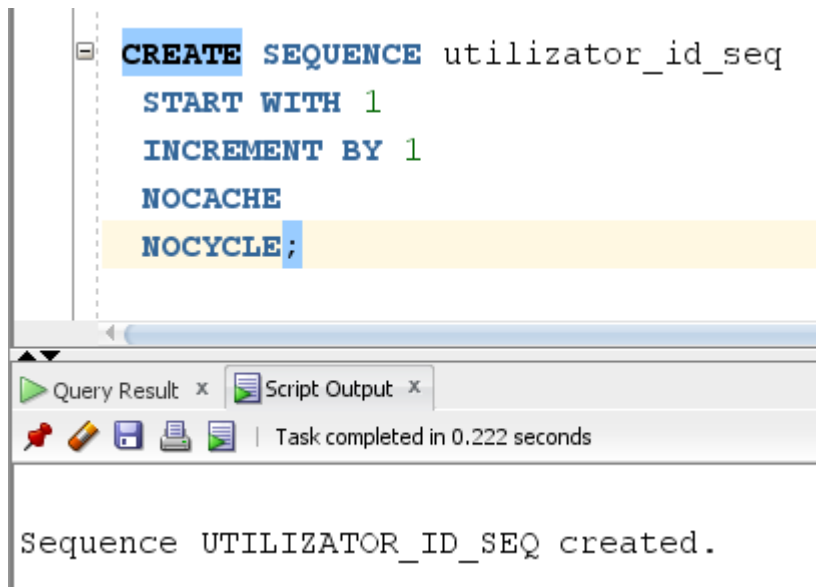
	MELODIE_ID	TITLU	DESCRIERE	DURATA	DATA_LANSARE	ALBUM_ID	ARTIST_ID
1	1	If I Ain't Got You	A s...	228	14-AUG-20	1	1
2	2	So Done	(null)	236	14-AUG-20	1	1
3	3	Girl on Fire	(null)	224	22-NOV-12	1	1
4	4	My Boo	(null)	223	29-AUG-04	1	1
5	5	No One	(null)	254	09-OCT-07	1	1
6	6	Fallin'	(null)	210	10-APR-01	1	1
7	7	Empire State of Mind	(null)	216	20-OCT-09	1	1
8	8	You don't know my name	(null)	366	10-NOV-03	1	1
9	9	Slow Down	(null)	258	02-DEC-03	1	1
10	10	Underdog	(null)	204	09-JAN-20	1	1
11	11	Like You'll Never See Me Again	(null)	315	25-OCT-07	1	1
12	12	Because of You	(null)	266	13-FEB-07	(null)	2
13	22	POWER	(null)	184	04-MAR-23	(null)	6
14	23	Future Lover	(null)	166	15-MAR-23	(null)	7
15	24	Sun and Moon	(null)	224	06-APR-22	(null)	8
16	25	Searchin	(null)	264	27-JAN-17	(null)	9
17	26	Fuck The World (Summer in London)	(null)	237	07-FEB-20	(null)	10

tabelul ALBUM:

	ALBUM_ID	GEN	PRET	DATA_LANSARE	ARTIST_ID	TITLU
1	1	R&B	69.99	04-NOV-16	1	HERE
2	2	R&B	59.99	28-FEB-06	2	In My Own Words
3	3	Pop	45.32	24-JAN-11	3	21
4	4	Indie	39.99	20-NOV-15	3	25
5	5	Pop	40.99	19-NOV-21	3	30
6	6	R&B	55.99	20-MAR-20	4	After Hours
7	7	R&B/Soul	63.99	07-JAN-22	4	Dawn FM
8	8	R&B/Soul	31.99	30-MAR-18	4	My dear Melancholy
9	9	R&B/Soul	99.99	25-NOV-16	4	Starboy
10	10	Pop	90.99	16-SEP-22	5	Girl Of My Dreams (Deluxe)
11	11	Pop	99.99	09-SEP-20	5	THE S(EX) TAPES

În acest fel, modelul de date propus se află deja în FN3.

10. Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele (punctul 11).



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

Exemplu inserare: INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nastere, tara)

VALUES(utilizator_id_seq.NEXTVAL, mara13, 'parola123', mara.spataru03@gmail.com', to_date('13-04-2003','dd-mm-yyyy'), 'RO');

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

```
CREATE SEQUENCE album_id_seq
```

```
START WITH 1  
INCREMENT BY 1  
NOCACHE  
NOCYCLE;
```

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

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

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

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

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

tabelul ARTIST:

	ARTIST_ID	NUME	VARSTA	DESCRIERE
1	1	Alicia Keys	42	American singer, songwriter, and...
2	2	Ne-Yo	43	American singer, songwriter, act...
3	6	Dilja	21	Icelandic singer
4	7	Brunette	21	Armenian singer-songwriter
5	8	anees	30	American singer, rapper, and son...
6	9	Sonder	(null)	Band created by the artist Brent...
7	10	Brent Fa...	27	American R&B singer
8	3	Adele	35	English singer-songwriter
9	4	The Weeknd	33	Canadian singer and songwriter
10	5	Fletcher	29	American singer and songwriter
11	11	Usher	44	American R&B singer

tabelul ALBUM:

	ALBUM_ID	GEN	PRET	DATA_LANSARE	ARTIST_ID	TITLU
1	1	R&B	69.99	04-NOV-16	1	HERE
2	2	R&B	59.99	28-FEB-06	2	In My Own Words
3	3	Pop	45.32	24-JAN-11	3	21
4	4	Indie	39.99	20-NOV-15	3	25
5	5	Pop	40.99	19-NOV-21	3	30
6	6	R&B	55.99	20-MAR-20	4	After Hours
7	7	R&B/Soul	63.99	07-JAN-22	4	Dawn FM
8	8	R&B/Soul	31.99	30-MAR-18	4	My dear Melancholy
9	9	R&B/Soul	99.99	25-NOV-16	4	Starboy
10	10	Pop	90.99	16-SEP-22	5	Girl Of My Dreams (Deluxe)
11	11	Pop	99.99	09-SEP-20	5	THE S(EX) TAPES

tabelul MELODIE:

MELODIE_ID	TITLU	DESCRIERE	DURATA	DATA Lansare	ALBUM_ID	ARTIST_ID
1	1 If I Ain't Got You	A s...	228	14-AUG-20	1	1
2	2 So Done	(null)	236	14-AUG-20	1	1
3	3 Girl on Fire	(null)	224	22-NOV-12	1	1
4	4 My Boo	(null)	223	29-AUG-04	1	1
5	5 No One	(null)	254	09-OCT-07	1	1
6	6 Fallin'	(null)	210	10-APR-01	1	1
7	7 Empire State of Mind	(null)	216	20-OCT-09	1	1
8	8 You don't know my name	(null)	366	10-NOV-03	1	1
9	9 Slow Down	(null)	258	02-DEC-03	1	1
10	10 Underdog	(null)	204	09-JAN-20	1	1
11	11 Like You'll Never See Me Again	(null)	315	25-OCT-07	1	1
12	12 Because of You	(null)	266	13-FEB-07	(null)	2
13	22 POWER	(null)	184	04-MAR-23	(null)	6
14	23 Future Lover	(null)	166	15-MAR-23	(null)	7
15	24 Sun and Moon	(null)	224	06-APR-22	(null)	8
16	25 Searchin	(null)	264	27-JAN-17	(null)	9
17	26 Fuck The World (Summer in London)	(null)	237	07-FEB-20	(null)	10
18	13 So Sick	(null)	207	21-NOV-05	(null)	2
19	14 Miss Independent	(null)	232	26-AUG-08	(null)	2
20	15 Sexy Love	(null)	220	06-JUN-06	(null)	2
21	16 She Knows	(null)	214	16-SEP-14	(null)	2
22	17 Mad	(null)	254	17-NOV-08	(null)	2
23	18 Time of Our Lives	(null)	229	17-NOV-14	(null)	2
24	19 U 2 Luv	(null)	209	29-MAY-20	(null)	2
25	20 One in a Million	(null)	244	14-SEP-10	(null)	2
26	21 Closer	(null)	235	15-APR-08	(null)	2
27	27 My Boo	(null)	223	29-AUG-04	(null)	11

tabelul PLAYLIST:

PLAYLIST_ID	TITLU	DATA CREARE
1	1 Alicia Keys songs	11-MAY-23
2	2 On Repeat	04-MAY-20
3	3 Disney	06-MAR-23
4	4 My Playlist #4	13-SEP-19
5	5 mare	04-DEC-21
6	6 r	27-AUG-17
7	7 rb	01-JAN-20
8	8 .	13-MAR-21
9	9 <bG	09-DEC-21
10	10 00	15-FEB-21
11	11 the weeknd	02-AUG-22
12	12 c	04-MAY-23

tabelul UTILIZATOR:

..	NUME_UTILIZATOR	PAROLA	EMAIL	DATA_NASTERE	TARA
1	1mara133	parola123	mara.s...	13-APR-03	RO
2	2thelibrasavannah	parola123	thelib...	08-SEP-03	US
3	3boatelcy	parola123	boat.c...	27-AUG-07	CA
4	4Finalka	parola123	fin.fi...	13-MAR-02	RO
5	5Johnadbo	parola123	john.b...	17-SEP-00	IT
6	6Allotikri	parola123	leon.s...	23-FEB-99	NL
7	7Myzgeo	parola123	mandyg...	08-SEP-03	RO
8	8Pltoex	parola123	alex.m...	14-MAY-05	DK
9	9Matweeknd	parola123	matthe...	04-DEC-03	FI
10	10Fletch04	parola123	fletch...	04-JUN-06	ES
11	11vestovlone	parola123	fourme...	12-DEC-12	NL

tabelul EPISOD:

EPISOD_ID	PODCAST_ID	TITLU
1	1	1The Killer Brothers Pt. 1
2	2	1The Killer Brothers Pt. 2
3	3	1"The Buried Bodies Killer" Robert Garrow Pt. 1
4	4	1"The Buried Bodies Killer" Robert Garrow Pt. 2
5	5	2Primul Episod : Misterul reginei frumusetii ai anilor '90 (
6	6	2Episodul 2: Disparitia lui Madeline McCann
7	7	2Episodul 3: Cultul lui Charles Manson - Cum arata "Procesul
8	8	2Episodul 4 : Este Ted Bundy cel mai malefic om a Americii a
9	9	3Sezonul 1, Episodul 1 - Introducere in Arhitectura Cerebral
10	10	3Sezonul 1, Episodul 2 - Calaretul si elefantul din capul fi
11	11	4"E gratis sa oferi un zambet!" - LA FILEU cu Levi Elekes
12	12	5Blues extraterestri si chemtrails
13	13	6BACKPACKYOURLIFE: CU UN GHIOZDAN IN JURUL LUMII
14	14	7How "second chance" laws could transform the US justice sys
15	15	8Jocurile Mintii- Ep.1-"Sa iti dai voie si tie sa existi"
16	16	9Iubire versus Atasament - 5 diferente
17	17	10Dr. Gheorghe Gica Recoltarea Celulelor Stem la nastere

tabelul PODCAST:

PODCAST_ID	NUME	NR_EPISOADE	NUME_AUTOR
1	1Serial Killers	4	Parcast
2	2Crime, Pisici si Cafea	4	Alerina
3	3Mind Architect	2	Mind Architect
4	4La Fileu	1	Speak
5	5MCN Podcast	1	MCN Podcast
6	6Vin de-o poveste	1	Radu Tibulca
7	7TED Talks Daily	1	TED
8	8Jocurile Mintii	1	Raluca Anton
9	9Secrete de la Psiholog	1	Psiholog Alexandru Plesea
10	10Podcastul medical	1	Cristian Badescu

tabelul ARTIST_lanseaza_ALBUM:

	ARTIST_ID	ALBUM_ID
1	1	1
2	2	2
3	3	3
4	3	4
5	3	5
6	4	6
7	4	7
8	4	8
9	4	9
10	5	10
11	5	11

tabelul ARTIST_canta_MELODIE:

	ARTIST_ID	MELODIE_ID
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	1	6
7	1	7
8	1	8
9	1	9
10	1	10
11	1	11
12	2	12
13	2	18
14	2	19
15	2	20
16	2	21
17	2	22
18	2	23
19	2	24
20	2	25
21	2	26
22	6	13
23	7	14
24	8	15
25	9	16
26	10	17

tabelul UTILIZATOR_apreciaza_ALBUM:

	UTILIZATOR_ID	ALBUM_ID
1	1	1
2	1	2
3	1	7
4	1	11
5	2	3
6	2	5
7	2	8
8	5	10
9	6	10
10	7	10
11	10	10
12	10	11

tabelul UTILIZATOR_urmareste_ARTIST:

	UTILIZATOR_ID	ARTIST_ID
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	1	6
7	1	7
8	1	8
9	1	9
10	1	10
11	5	2
12	7	3
13	7	10
14	8	4
15	9	10

tabelul MELODIE_in_PLAYLIST_UTILIZATOR:

	MELODIE_ID	PLAYLIST_ID	UTILIZATOR_ID	DATA_ADAUGARE
1	1	1	1	11-MAY-23
2	2	1	1	12-DEC-14
3	2	1	4	12-DEC-14
4	2	1	9	12-DEC-14
5	2	1	2	14-DEC-14
6	2	2	4	16-DEC-14
7	2	2	9	16-DEC-14
8	2	7	4	18-DEC-14
9	2	7	9	18-DEC-14
10	2	7	1	18-DEC-14
11	1	1	4	04-APR-20
12	1	1	10	04-APR-20
13	1	2	4	09-APR-20
14	1	2	9	09-APR-20
15	1	7	4	09-APR-20
16	1	7	10	09-APR-20
17	3	1	3	09-APR-20
18	3	1	7	09-APR-20
19	3	1	8	09-APR-20
20	3	1	2	09-APR-20
21	3	2	3	09-APR-20
22	3	2	7	09-APR-20
23	3	2	8	09-APR-20
24	3	2	2	09-APR-20
25	3	7	3	09-APR-20
26	3	7	7	09-APR-20
27	3	7	8	09-APR-20
28	3	7	2	10-APR-20
29	4	2	1	13-APR-21
30	4	2	2	13-APR-21
31	4	2	3	13-APR-21
32	4	7	1	13-APR-21
33	4	7	2	13-APR-21
34	4	7	3	13-APR-21

tabelul UTILIZATOR_apreciaza_PLAYLIST:

	UTILIZATOR_ID	PLAYLIST_ID
1	1	1
2	1	2
3	1	4
4	1	9
5	2	1
6	2	5
7	2	12
8	3	6
9	3	7
10	4	3
11	4	4
12	5	8
13	6	10
14	7	2
15	7	5
16	7	9
17	8	7
18	9	1
19	10	8
20	1	5
21	1	12

tabelul ISTORIC_MELODII:

	ISTORIC_MELODII_ID	UTILIZATOR_ID	MELODIE_ID	DATA_ASCULTARE
1	1	1	1	11-MAY-23
2	2	2	2	2 10-SEP-22
3	3	3	3	5 10-OCT-22
4	4	4	4	7 04-DEC-21
5	5	5	5	10 09-DEC-21
6	6	6	6	12 31-DEC-21
7	7	7	7	14 28-JAN-22
8	8	8	8	15 28-FEB-22
9	9	9	9	16 04-JUN-22
10	10	10	10	21 14-AUG-22
11	11	11	11	26 04-DEC-22
12	12	1	1	14 04-DEC-21
13	13	7	7	15 04-DEC-21

tabelul PODCAST_ISTORIC:

	PODCAST_ISTORIC_ID	UTILIZATOR_ID	EPISOD_ID	DATA_ASCULTARE
1	1	1	1	11-MAY-23
2	2	1	2	14-MAY-23
3	3	3	5	13-APR-23
4	4	4	4	04-DEC-21
5	5	5	8	04-APR-20
6	6	6	10	19-OCT-20
7	7	7	8	13-SEP-20
8	8	8	9	13-APR-23
9	9	9	9	20-APR-23
10	10	10	12	13-MAR-23
11	11	11	15	12-DEC-20

--tabel UTILIZATOR

```
CREATE TABLE UTILIZATOR(
    utilizator_id INT PRIMARY KEY,
    nume_utilizator VARCHAR(50) UNIQUE,
    parola VARCHAR(128) NOT NULL,
    email VARCHAR(150),
    data_nastere DATE NOT NULL,
    tara VARCHAR(2) NOT NULL
);
```

--table ARTIST

```
CREATE TABLE ARTIST(
    artist_id INT PRIMARY KEY,
    nume VARCHAR(100),
    varsta INT,
    descriere VARCHAR(255)
);
```

```

--tabel ALBUM
CREATE TABLE ALBUM (
    album_id INT PRIMARY KEY,
    gen VARCHAR(50) NOT NULL,
    pret DECIMAL(10,2) CHECK (pret>0),
    titlu VARCHAR(100) NOT NULL,
    data_lansare DATE NOT NULL,
    artist_id INT NOT NULL,
    CONSTRAINT album_artist_fk FOREIGN KEY (artist_id) REFERENCES ARTIST
(artist_id)
    --dam nume pt fk ca sa putem identifica erori
);

--tabel UTILIZATOR_apreciaza_ALBUM
CREATE TABLE UTILIZATOR_APRECIAZA_ALBUM(
    utilizator_id INT,
    album_id INT,
    PRIMARY KEY(utilizator_id, album_id),
    FOREIGN KEY(utilizator_id) REFERENCES UTILIZATOR(utilizator_id),
    FOREIGN KEY(album_id) REFERENCES ALBUM (album_id)
);

--tabel ARTIST_LANSEAZA_ALBUM
CREATE TABLE ARTIST_LANSEAZA_ALBUM(
    artist_id INT,
    album_id INT,
    PRIMARY KEY(artist_id, album_id),
    FOREIGN KEY(artist_id) REFERENCES ARTIST (artist_id),
    FOREIGN KEY(album_id) REFERENCES ALBUM(album_id)
);

```


--tabel UTILIZATOR_URMARESTE_ARTIST

```
CREATE TABLE UTILIZATOR_URMARESTE_ARTIST(  
    utilizator_id INT,  
    artist_id INT,  
    PRIMARY KEY (utilizator_id, artist_id),  
    FOREIGN KEY(utilizator_id) REFERENCES UTILIZATOR(utilizator_id),  
    FOREIGN KEY(artist_id) REFERENCES ARTIST(artist_id)  
);
```

--tabel MELODIE

```
CREATE TABLE MELODIE(  
    melodie_id INT PRIMARY KEY,  
    titlu VARCHAR(255) NOT NULL,  
    descriere VARCHAR(255),  
    durata NUMBER(3,0) NOT NULL,  
    data_lansare DATE NOT NULL,  
    album_id INT NOT NULL,  
    artist_id INT,  
    CONSTRAINT song_artist_fk FOREIGN KEY (artist_id) REFERENCES  
ARTIST(artist_id),  
    CONSTRAINT song_album_fk FOREIGN KEY (album_id) REFERENCES  
ALBUM(album_id)  
);
```

--tabel ARTIST_CANTA_MELODIE

```
CREATE TABLE ARTIST_CANTA_MELODIE (  
    artist_id INT,  
    melodie_id INT,  
    PRIMARY KEY(artist_id, melodie_id),
```

```

FOREIGN KEY (artist_id) REFERENCES ARTIST (artist_id),
FOREIGN KEY (melodie_id) REFERENCES MELODIE(melodie_id)
);

--tabel PLAYLIST
CREATE TABLE PLAYLIST (
    playlist_id INT PRIMARY KEY,
    titlu VARCHAR(255) NOT NULL,
    data_creatoare DATE NOT NULL
);

--TABEL MELODIE_IN_PLAYLIST_UTILIZATOR
CREATE TABLE MELODIE_IN_PLAYLIST_UTILIZATOR(
    melodie_id INT,
    playlist_id INT,
    utilizator_id INT,
    data_adaugare DATE NOT NULL, --data adaugarii melodiei in playlist-ul userului
    PRIMARY KEY (melodie_id, playlist_id, utilizator_id),
    FOREIGN KEY(melodie_id) REFERENCES MELODIE(melodie_id),
    FOREIGN KEY (playlist_id) REFERENCES PLAYLIST(playlist_id),
    FOREIGN KEY (utilizator_id) REFERENCES UTILIZATOR(utilizator_id)
);

--tabel UTILIZATOR_APRECIAZA_PLAYLIST
CREATE TABLE UTILIZATOR_APRECIAZA_PLAYLIST(
    utilizator_id INT,
    playlist_id INT,
    PRIMARY KEY (utilizator_id, playlist_id),
    FOREIGN KEY (utilizator_id) REFERENCES UTILIZATOR(utilizator_id),

```

```

FOREIGN KEY(playlist_id) REFERENCES PLAYLIST(playlist_id)
);

--tabel ISTORIC_MELODII
CREATE TABLE ISTORIC_MELODII(
    istoric_melodii_id INT PRIMARY KEY,
    utilizator_id INT,
    melodie_id INT,
    data_ascultare DATE NOT NULL,
    FOREIGN KEY (utilizator_id) REFERENCES UTILIZATOR(utilizator_id),
    FOREIGN KEY(melodie_id) REFERENCES MELODIE(melodie_id)
);

--table PODCAST
CREATE TABLE PODCAST (
    podcast_id INT PRIMARY KEY,
    nume VARCHAR(255) NOT NULL,
    nr_episoade INT CHECK (nr_episoade >0),
    nume_autor VARCHAR(100) NOT NULL
);

--tabel EPISOD
CREATE TABLE EPISOD(
    episod_id INT PRIMARY KEY,
    podcast_id INT,
    titlu VARCHAR(120) NOT NULL,
    FOREIGN KEY (podcast_id) REFERENCES PODCAST(podcast_id)
);

```

```
--tabel PODCAST_ISTORIC

CREATE TABLE PODCAST_ISTORIC(

    podcast_istoric_id INT PRIMARY KEY,

    utilizator_id INT,

    episod_id INT,

    data_ascultare DATE NOT NULL,

    FOREIGN KEY (utilizator_id) REFERENCES UTILIZATOR(utilizator_id),

    FOREIGN KEY (episod_id) REFERENCES EPISOD(episod_id)

);
```

Inserare date în tabel:

```
INSERT INTO ARTIST (artist_id, nume, varsta, descriere)
VALUES (1,'Alicia Keys', 42, 'American singer, songwriter, and pianist');

INSERT INTO ARTIST (artist_id, nume, varsta, descriere)
VALUES (2,'Ne-Yo', 43, 'American singer, songwriter, actor, dancer, and record producer.');
```

insert into artist(artist_id, nume, varsta, descriere)

values(3, 'Adele', 35, 'English singer-songwriter');


```
INSERT INTO ARTIST(artist_id, nume, varsta, descriere)
VALUES(4, 'The Weeknd', 33, 'Canadian singer and songwriter');
```

INSERT INTO ARTIST(artist_id, nume, varsta, descriere)

VALUES(5, 'Fletcher', 29, 'American singer and songwriter');

insert into artist(artist_id, nume, varsta, descriere)

values(6, 'Dilja', 21, 'Icelandic singer');

insert into artist(artist_id, nume, varsta, descriere)

values(7, 'Brunette', 21, 'Armenian singer-songwriter');

insert into artist(artist_id, nume, varsta, descriere)

values(8, 'anees', 30, 'American singer, rapper, and songwriter');

insert into artist(artist_id, nume, varsta, descriere)

```
values(9, 'Sonder', null, 'Band created by the artist Brent Faiyaz');
```

```
insert into artist(artist_id, nume, varsta, descriere)
```

```
values(10, 'Brent Faiyaz', 27, 'American R||&||'B singer');
```

```
insert into artist(artist_id, nume, varsta, descriere)
```

```
values(11, 'Usher', 44, 'American R||&||'B singer');
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere,durata, data_lansare, album_id,  
artist_id)
```

```
VALUES(1,'If I Ain't Got You', '', 228, to_date('17-02-2004','dd-mm-yyyy'),1,1);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere,durata, data_lansare, album_id,  
artist_id)
```

```
VALUES (2,'So Done','',236,TO_DATE('14-08-2020','dd-mm-yyyy'),1,1);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere,durata, data_lansare, album_id,  
artist_id)
```

```
VALUES (3,'Girl on Fire','', 224,TO_DATE('22-11-2012','DD-MM-YYYY'),1,1);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere,durata, data_lansare, album_id,  
artist_id)
```

```
VALUES (4,'My Boo','',223,to_date('29-08-2004','dd-mm-yyyy'),1,1);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere,durata, data_lansare, album_id,  
artist_id)
```

```
VALUES (5,'No One','', 254, to_date('09-10-2007','dd-mm-yyyy'),1,1);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere,durata, data_lansare, album_id,  
artist_id)
```

```
VALUES (6,'Fallin','',210, to_date('10-04-2001','dd-mm-yyyy'),1,1);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere,durata, data_lansare, album_id,  
artist_id)
```

```
VALUES(7,'Empire State of Mind','',216, to_date('20-10-2009','dd-mm-yyyy'),1,1);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere,durata, data_lansare, album_id,  
artist_id)
```

```
VALUES(8,'You don't know my name','',366, to_date('10-11-2003','dd-mm-yyyy'),1,1);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere,durata, data_lansare, album_id,  
artist_id)
```

```
VALUES(9,'Slow Down','', 258, to_date('02-12-2003','dd-mm-yyyy'),1,1);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere,durata, data_lansare, album_id,
artist_id)
```

```
VALUES(10,'Underdog',"204, to_date('09-01-2020','dd-mm-yyyy'),1,1);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere,durata, data_lansare, album_id,
artist_id)
```

```
VALUES(11,'Like You"ll Never See Me Again', ", 315,
to_date('25-10-2007','dd-mm-yyyy'),1,1);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere, durata, data_lansare, album_id,
artist_id)
```

```
VALUES(12, 'Because of You',"266, TO_DATE('13-02-2007','DD-MM-YYYY'),NULL,2);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere, durata, data_lansare, album_id,
artist_id)
```

```
VALUES(13,'So Sick', ",207, TO_DATE('21-11-2005','DD-MM-YYYY'), NULL, 2);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere, durata, data_lansare, album_id,
artist_id)
```

```
VALUES(14,'Miss Independent',"232,to_date('26-08-2008','dd-mm-yyyy'),null, 2);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere, durata, data_lansare, album_id,
artist_id)
```

```
VALUES(15,'Sexy Love',"220,to_date('06-06-2006','dd-mm-yyyy'), null, 2);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere, durata, data_lansare, album_id,
artist_id)
```

```
VALUES(16,'She Knows', ", 214, to_date('16-09-2014','dd-mm-yyyy'), null, 2);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere, durata, data_lansare, album_id,
artist_id)
```

```
VALUES(17, 'Mad', ", 254, to_date('17-11-2008','dd-mm-yyyy'), null, 2);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere, durata, data_lansare, album_id,
artist_id)
```

```
VALUES(18, 'Time of Our Lives', ", 229, to_date('17-11-2014','dd-mm-yyyy'), null, 2);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere, durata, data_lansare, album_id,
artist_id)
```

```
VALUES(19, 'U 2 Luv', '', 209, to_date('29-05-2020', 'dd-mm-yyyy'), null, 2);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere, durata, data_lansare, album_id,
artist_id)
```

```
VALUES(20, 'One in a Million', '', 244, to_date('14-09-2010', 'dd-mm-yyyy'), null, 2);
```

```
INSERT INTO MELODIE(melodie_id, titlu, descriere, durata, data_lansare, album_id,
artist_id)
```

```
VALUES(21, 'Closer', '', 235, to_date('15-04-2008', 'dd-mm-yyyy'), null, 2);
```

```
insert into melodie(melodie_id, titlu, descriere, durata, data_lansare, album_id, artist_id)
```

```
values(22, 'POWER', '', 184, to_date('04-03-2023', 'dd-mm-yyyy'), null, 6);
```

```
insert into melodie(melodie_id, titlu, descriere, durata, data_lansare, album_id, artist_id)
```

```
values(23, 'Future Lover', '', 166, to_date('15-03-2023', 'dd-mm-yyyy'), null, 7);
```

```
insert into melodie(melodie_id, titlu, descriere, durata, data_lansare, album_id, artist_id)
```

```
values(24, 'Sun and Moon', '', 224, to_date('06-04-2022', 'dd-mm-yyyy'), null, 8);
```

```
insert into melodie(melodie_id, titlu, descriere, durata, data_lansare, album_id, artist_id)
```

```
values(25, 'Searchin', '', 264, to_date('27-01-2017', 'dd-mm-yyyy'), null, 9);
```

```
insert into melodie(melodie_id, titlu, descriere, durata, data_lansare, album_id, artist_id)
```

```
values(26, 'Fuck The World (Summer in London)', '', 237,
to_date('07-02-2020', 'dd-mm-yyyy'), null, 10);
```

```
insert into melodie(melodie_id, titlu, descriere, durata, data_lansare, album_id, artist_id)
```

```
values(27, 'My Boo', '', 223, to_date('29-08-2004', 'dd-mm-yyyy'), null, 11);
```

```
INSERT INTO ALBUM(album_id, gen, pret, data_lansare, artist_id, titlu)
```

```
VALUES(1, 'R' || '&' || 'B', 69.99, to_date('04-11-2016', 'dd-mm-yyyy'), 1, 'HERE');
```

```
INSERT INTO ALBUM(album_id, gen, pret, data_lansare, artist_id, titlu)
```

```
VALUES(2, 'R' || '&' || 'B', 59.99, to_date('28-02-2006', 'dd-mm-yyyy'), 2, 'In My Own Words');
```

```
INSERT INTO ALBUM(album_id, gen, pret, data_lansare, artist_id, titlu)
```

```

VALUES(3, 'Pop', 54.99, to_date('24-01-2011','dd-mm-yyyy'), 3, '21');
INSERT INTO ALBUM(album_id, gen, pret, data_lansare, artist_id, titlu)
VALUES(4, 'Indie', 39.99, to_date('20-11-2015','dd-mm-yyyy'), 3, '25');
INSERT INTO ALBUM(album_id, gen, pret, data_lansare, artist_id, titlu)
VALUES(5, 'Pop', 40.99, to_date('19-11-2021','dd-mm-yyyy'), 3, '30');
INSERT INTO ALBUM(album_id, gen, pret, data_lansare, artist_id, titlu)
VALUES(6, 'R||&||B', 55.99, to_date('20-03-2020','dd-mm-yyyy'), 4, 'After Hours');
INSERT INTO ALBUM(album_id, gen, pret, data_lansare, artist_id, titlu)
VALUES(7, 'R||&||B', 63.99, to_date('07-01-2022','dd-mm-yyyy'), 4, 'Dawn FM');
INSERT INTO ALBUM(album_id, gen, pret, data_lansare, artist_id, titlu)
VALUES(8, 'R||&||B', 31.99, to_date('30-03-2018','dd-mm-yyyy'), 4, 'My dear Melancholy');
INSERT INTO ALBUM(album_id, gen, pret, data_lansare, artist_id, titlu)
VALUES(9, 'R||&||B', 99.99, to_date('25-11-2016','dd-mm-yyyy'), 4, 'Starboy');
INSERT INTO ALBUM(album_id, gen, pret, data_lansare, artist_id, titlu)
VALUES(10, 'Pop', 90.99, to_date('16-09-2022','dd-mm-yyyy'), 5, 'Girl Of My Dreams (Deluxe)');
INSERT INTO ALBUM(album_id, gen, pret, data_lansare, artist_id, titlu)
VALUES(11, 'Pop', 99.99, to_date('09-09-2020','dd-mm-yyyy'), 5, 'THE S(EX) TAPES');

INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(1, 'Alicia Keys songs', to_date('11-05-2023','dd-mm-yyyy'));
INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(2, 'On Repeat', to_date('04-05-2020','dd-mm-yyyy'));
INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(3, 'Disney', to_date('06-03-2023','dd-mm-yyyy'));
INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(4, 'My Playlist #4', to_date('13-09-2019','dd-mm-yyyy'));

```



```

INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(5, 'mare', to_date('04-12-2021','dd-mm-yyyy'));
INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(6, 'r', to_date('27-08-2017', 'dd-mm-yyyy'));
INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(7, 'rb', to_date('01-01-2020', 'dd-mm-yyyy'));
INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(8, '.', to_date('13-03-2021', 'dd-mm-yyyy'));
INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(9, '<bG', to_date('09-12-2021','dd-mm-yyyy'));
INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(10, '00', to_date('15-02-2021', 'dd-mm-yyyy'));
INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(11, 'the weeknd', to_date('02-08-2022', 'dd-mm-yyyy'));
INSERT INTO PLAYLIST(playlist_id, titlu, data_creat)
VALUES(12, 'c', to_date('04-05-2023','dd-mm-yyyy'));

```

```

INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nasterii,
tara)

```

```

VALUES(1, mara13, 'parola123', mara.spataru03@gmail.com',
to_date('13-04-2003','dd-mm-yyyy'), 'RO');

```

```

INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nasterii,
tara)

```

```

VALUES(2, 'thelibrasavannah', 'parola123', 'thelibra@yahoo.com',
to_date('08-09-2003','dd-mm-yyyy'), 'US');

```

```

INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nasterii,
tara)

```

```

values(3, 'boatelcy', 'parola123', 'boat.company@gmail.com', to_date('27-08-2007',
'dd-mm-yyyy'), 'CA');

```

```
INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nastere,
tara)
```

```
values(4,'Finalka', 'parola123', 'fin.finn@gmail.com', to_date('13-03-2002','dd-mm-yyyy'),
'RO');
```

```
INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nastere,
tara)
```

```
values(5,'Johnadbo','parola123', 'john.boa@gmail.com', to_date('17-09-2000','dd-mm-yyyy'),
'IT');
```

```
INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nastere,
tara)
```

```
VALUES(6,          'Allotikri',          'parola123',          'leon.sam@yahoo.com',
to_date('23-02-1999','dd-mm-yyyy'), 'NL');
```

```
INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nastere,
tara)
```

```
VALUES(7,          'Myzgeo',          'parola123',          'mandyg412@gmail.com',
to_date('08-09-2003','dd-mm-yyyy'), 'RO');
```

```
INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nastere,
tara)
```

```
values(8, 'Pltoex', 'parola123', 'alex.mex@yahoo.com', to_date('14-05-2005', 'dd-mm-yyyy'),
'DK');
```

```
INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nastere,
tara)
```

```
values(9,          'Matweeknd'          ,          'parola123',          'matthew.xo@gmail.com',
to_date('04-12-2003','dd-mm-yyyy'), 'FI');
```

```
INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nastere,
tara)
```

```
values(10,          'Fletch04',          'parola123',          'fletcher.a@yahoo.com',
to_date('04-06-2006','dd-mm-yyyy'), 'ES');
```

```
INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nastere,
tara)
```

```
VALUES(11,          'vestovlone',          'parola123',          'vincent23@gmail.com',
to_date('21-03-2004','dd-mm-yyyy'), 'RO');
```

```
INSERT INTO UTILIZATOR(utilizator_id, nume_utilizator, parola, email, data_nastere,
tara)
```

```
values(12, 'vlonefour', 'parola123', 'fourment.a23@gmail.com', to_date('12-12-2012',
'dd-mm-yyyy'), 'NL');
```

```

insert into EPISOD(episod_id, podcast_id, titlu)
VALUES(1, 1, 'The Killer Brothers Pt. 1');
insert into EPISOD(episod_id, podcast_id, titlu)
VALUES(2, 1, 'The Killer Brothers Pt. 2');
insert into EPISOD(episod_id, podcast_id, titlu)
values(3,1,"'The Buried Bodies Killer" Robert Garrow Pt. 1');
insert into EPISOD(episod_id, podcast_id, titlu)
values(4, 1, "'The Buried Bodies Killer" Robert Garrow Pt. 2');
insert into EPISOD(episod_id, podcast_id, titlu)
values(5, 2,'Primul Episod : Misterul reginei frumusetii ai anilor "90 (Cazul JonBenet Ramsey)');
insert into EPISOD(episod_id, podcast_id, titlu)
values(6, 2,'Episodul 2: Disparitia lui Madeline McCann');
insert into EPISOD(episod_id, podcast_id, titlu)
values(7,2,'Episodul 3: Cultul lui Charles Manson - Cum arata "Procesul Secolului"');
insert into EPISOD(episod_id, podcast_id, titlu)
values(8,2,'Episodul 4 : Este Ted Bundy cel mai malefic om a Americii anilor "70?');
insert into EPISOD(episod_id, podcast_id, titlu)
values(9,3, 'Sezonul 1, Episodul 1 - Introducere in Arhitectura Cerebrala');
insert into EPISOD(episod_id, podcast_id, titlu)
values(10,3, 'Sezonul 1, Episodul 2 - Calaretul si elefantul din capul fiecaruia dintre noi');
insert into EPISOD(episod_id, podcast_id, titlu)
values(11, 4, "'E gratis sa oferi un zambet!" - LA FILEU cu Levi Elekes');
insert into EPISOD(episod_id, podcast_id, titlu)
values(12,5, 'Blues extraterestri si chemtrails');
insert into EPISOD(episod_id, podcast_id, titlu)
values(13, 6, 'BACKPACKYOURLIFE: CU UN GHIOZDAN IN JURUL LUMII');
insert into EPISOD(episod_id, podcast_id, titlu)

```

```
values(14,7, 'How "second chance" laws could transform the US justice system | Sheena Meade');
```

```
insert into EPISOD(episod_id, podcast_id, titlu)
```

```
values(15, 8, 'Jocurile Mintii- Ep.1-"Sa iti dai voie si tie sa existi"');
```

```
insert into EPISOD(episod_id, podcast_id, titlu)
```

```
values(16, 9, 'Iubire versus Atasament - 5 diferite');
```

```
insert into EPISOD(episod_id, podcast_id, titlu)
```

```
values(17, 10, 'Dr. Gheorghe Gica | Recoltarea Celulelor Stem la nastere');
```

```
insert into podcast(podcast_id, nume, nr_episoade, nume_autor)
```

```
values(1,'Serial Killers', 4,'Parcast');
```

```
insert into podcast(podcast_id, nume, nr_episoade, nume_autor)
```

```
values(2, 'Crime, Pisici si Cafea', 4, 'Alerina');
```

```
insert into podcast(podcast_id, nume, nr_episoade, nume_autor)
```

```
values(3, 'Mind Architect', 2, 'Mind Architect');
```

```
INSERT INTO podcast(podcast_id, nume, nr_episoade, nume_autor)
```

```
VALUES (4, 'La Fileu', 1, 'Speak');
```

```
insert into podcast(podcast_id, nume, nr_episoade, nume_autor)
```

```
values(5, 'MCN Podcast', 1, 'MCN Podcast');
```

```
insert into podcast(podcast_id, nume, nr_episoade, nume_autor)
```

```
values(6, 'Vin de-o poveste', 1, 'Radu Tibulca');
```

```
insert into podcast(podcast_id, nume, nr_episoade, nume_autor)
```

```
values(7,'TED Talks Daily', 1, 'TED');
```

```
insert into podcast(podcast_id, nume, nr_episoade, nume_autor)
```

```
values(8, 'Jocurile Mintii', 1, 'Raluca Anton');
```

```
insert into podcast(podcast_id, nume, nr_episoade, nume_autor)
```

```
values(9, 'Secrete de la Psiholog', 1, 'Psiholog Alexandru Plesea');
```

```
insert into podcast(podcast_id, nume, nr_episoade, nume_autor)
```

```
values(10,'Podcastul medical', 1, 'Cristian Badescu');
```

```
insert into artist_lanseaza_album(artist_id, album_id)
```

```
values(1,1);
```

```
insert into artist_lanseaza_album(artist_id, album_id)
```

```
values(2,2);
```

```
insert into artist_lanseaza_album(artist_id, album_id)
```

```
values(3,3);
```

```
insert into artist_lanseaza_album(artist_id, album_id)
```

```
values(3,4);
```

```
insert into artist_lanseaza_album(artist_id, album_id)
```

```
values(3,5);
```

```
insert into artist_lanseaza_album(artist_id, album_id)
```

```
values(4,6);
```

```
insert into artist_lanseaza_album(artist_id, album_id)
```

```
values(4,7);
```

```
insert into artist_lanseaza_album(artist_id, album_id)
```

```
values(4,8);
```

```
insert into artist_lanseaza_album(artist_id, album_id)
```

```
values(4,9);
```

```
insert into artist_lanseaza_album(artist_id, album_id)
```

```
values(5,10);
```

```
insert into artist_lanseaza_album(artist_id, album_id)
```

```
values(5,11);
```

```
insert into artist_canta_melodie(artist_id, melodie_id)
```

```
values(1,1);
```

```
insert into artist_canta_melodie(artist_id, melodie_id)
```

```
values(1,2);
insert into artist_canta_melodie(artist_id, melodie_id)
values(1,3);
insert into artist_canta_melodie(artist_id, melodie_id)
values(1,4);
insert into artist_canta_melodie(artist_id, melodie_id)
values(1,5);
insert into artist_canta_melodie(artist_id, melodie_id)
values(1,6);
insert into artist_canta_melodie(artist_id, melodie_id)
values(1,7);
insert into artist_canta_melodie(artist_id, melodie_id)
values(1,8);
insert into artist_canta_melodie(artist_id, melodie_id)
values(1,9);
insert into artist_canta_melodie(artist_id, melodie_id)
values(1,10);
insert into artist_canta_melodie(artist_id, melodie_id)
values(1,11);
insert into artist_canta_melodie(artist_id, melodie_id)
values(2,12);
insert into artist_canta_melodie(artist_id, melodie_id)
values(2,18);
insert into artist_canta_melodie(artist_id, melodie_id)
values(2,19);
insert into artist_canta_melodie(artist_id, melodie_id)
values(2,20);
insert into artist_canta_melodie(artist_id, melodie_id)
values(2,21);
```

```
insert into artist_canta_melodie(artist_id, melodie_id)
values(2,22);
insert into artist_canta_melodie(artist_id, melodie_id)
values(2,23);
insert into artist_canta_melodie(artist_id, melodie_id)
values(2,24);
insert into artist_canta_melodie(artist_id, melodie_id)
values(2,25);
insert into artist_canta_melodie(artist_id, melodie_id)
values(2,26);
insert into artist_canta_melodie(artist_id, melodie_id)
values(6,13);
insert into artist_canta_melodie(artist_id, melodie_id)
values(7,14);
insert into artist_canta_melodie(artist_id, melodie_id)
values(8,15);
insert into artist_canta_melodie(artist_id, melodie_id)
values(9,16);
insert into artist_canta_melodie(artist_id, melodie_id)
values(10,17);
```

```
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
values(1,1);
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
values(1,2);
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
values(1,7);
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
```

```
values(1,11);
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
values(2, 5);
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
values(2,8);
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
values(2, 3);
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
values(5,10);
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
values(6,10);
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
values(7,10);
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
values(10,10);
insert into utilizator_apreciaza_album(utilizator_id, ALBUM_ID)
values(10,11);
```

```
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(1,1);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(1,2);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(1,3);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(1,4);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
```



```

values(1,5);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(1,6);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(1,7);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(1,8);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(1,9);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(1,10);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(7,3);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(5,2);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(8,4);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(9,10);
insert into utilizator_urmareste_artist(utilizator_id, artist_id)
values(7,10);

```

```

insert into melodie_in_playlist_utilizator(melodie_id, playlist_id, utilizator_id,
data_adaugare)
values(1,1,1,to_date('11-05-2023','dd-mm-yyyy'));

```

insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(2,1,1,to_date('12-12-2014','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(2,1,4,to_date('12-12-2014','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(2,1,9, to_date('12-12-2014','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(2,1,2, to_date('14-12-2014','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(2,2,4, to_date('16-12-2014','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(2,2,9, to_date('16-12-2014','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(2,7,4, to_date('18-12-2014','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(2,7,9, to_date('18-12-2014','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(2,7,1, to_date('18-12-2014','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(1,1,4, to_date('04-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(1,1,10, to_date('04-04-2020','dd-mm-yyyy'));		

insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(1,2,4, to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(1,2,9, to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(1,7,4, to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(1,7,10, to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,1,3, to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,1,7,to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,1,8, to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,1,2, to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,2,3, to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,2,7,to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,2,8,to_date('09-04-2020','dd-mm-yyyy'));		

insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,2,2,to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,7,3,to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,7,7,to_date('09-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,7,8, to_date('09-04-2020', 'dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(3,7,2,to_date('10-04-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(4,2,1,to_date('13-04-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(4,2,2,to_date('13-04-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(4,2,3, to_date('13-04-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(4,7,1,to_date('13-04-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(4,7,2,to_date('13-04-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(4,7,3,to_date('13-04-2021','dd-mm-yyyy'));		

insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(4,10,1,to_date('13-04-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(4,10,2,to_date('13-04-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(4,10,3,to_date('13-04-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,1,4,to_date('10-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,1,5,to_date('10-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,1,6,to_date('10-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,2,4,to_date('10-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,2,5,to_date('10-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,2,6,to_date('10-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,7,4,to_date('10-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,7,5,to_date('10-08-2022','dd-mm-yyyy'));		

insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,7,6,to_date('10-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,10,4,to_date('10-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,10,5,to_date('10-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(5,10,6,to_date('10-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(6,1,7,to_date('10-10-2010','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(6,2,7,to_date('10-10-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(6,7,7,to_date('10-10-2010','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(6,10,7,to_date('10-10-2010','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(7,1,8,to_date('24-04-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(7,1,10,to_date('24-04-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(7,7,8,to_date('24-04-2022','dd-mm-yyyy'));		

insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(7,7,10,to_date('24-04-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(8,1,2,to_date('13-09-2015','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(8,1,3,to_date('13-09-2015','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(9,1,4,to_date('30-05-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(9,1,5,to_date('30-05-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(10,1,6,to_date('02-02-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(10,1,7,to_date('02-02-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(11,1,8,to_date('10-10-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(11,1,10,to_date('10-10-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(12,2,2,to_date('27-08-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(12,7,2,to_date('27-08-2020','dd-mm-yyyy'));		

insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(12,10,2,to_date('27-08-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(12,2,3,to_date('27-08-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(12,7,3, to_date('27-08-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(12,10,3,to_date('27-08-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(13,2,4,to_date('03-03-2023','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(13,7,4,to_date('03-03-2023','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(13,10,4,to_date('03-03-2023','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(14,2,5,to_date('05-05-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(14,7,5,to_date('05-05-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(14,10,5,to_date('05-05-2020','dd-mm-yyyy'));		

insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(15,2,6,to_date('06-06-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(15,7,6,to_date('06-06-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(15,10,6,to_date('06-06-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(16,7,8,to_date('07-07-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(16,7,7,to_date('07-07-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(17,7,10,to_date('07-07-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(17,7,11, to_date('07-07-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(18,7,11,to_date('08-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(18,7,2,to_date('08-08-2022','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,

```

values(19,7,11,to_date('01-04-2020','dd-mm-yyyy'));

insert into melodie_in_playlist_utilizator(melodie_id, playlist_id, utilizator_id,
data_adaugare)

values(19,7,10,to_date('01-04-2020','dd-mm-yyyy'));


insert into melodie_in_playlist_utilizator(melodie_id, playlist_id, utilizator_id,
data_adaugare)

values(20,2,1,to_date('04-12-2021','dd-mm-yyyy'));

insert into melodie_in_playlist_utilizator(melodie_id, playlist_id, utilizator_id,
data_adaugare)

values(20,7,1,to_date('04-12-2021','dd-mm-yyyy'));

insert into melodie_in_playlist_utilizator(melodie_id, playlist_id, utilizator_id,
data_adaugare)

values(20,9,1,to_date('04-12-2021','dd-mm-yyyy'));

insert into melodie_in_playlist_utilizator(melodie_id, playlist_id, utilizator_id,
data_adaugare)

values(20,10,1,to_date('04-12-2021','dd-mm-yyyy'));

insert into melodie_in_playlist_utilizator(melodie_id, playlist_id, utilizator_id,
data_adaugare)

values(20,2,7,to_date('04-12-2021','dd-mm-yyyy'));

insert into melodie_in_playlist_utilizator(melodie_id, playlist_id, utilizator_id,
data_adaugare)

values(20,7,7,to_date('04-12-2021','dd-mm-yyyy'));

insert into melodie_in_playlist_utilizator(melodie_id, playlist_id, utilizator_id,
data_adaugare)

values(20,9,7,to_date('04-12-2021','dd-mm-yyyy'));

insert into melodie_in_playlist_utilizator(melodie_id, playlist_id, utilizator_id,
data_adaugare)

values(20,10,7,to_date('04-12-2021','dd-mm-yyyy'));


insert into melodie_in_playlist_utilizator(melodie_id, playlist_id, utilizator_id,
data_adaugare)

values(21,7,1,to_date('04-12-2021','dd-mm-yyyy'));

```

insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(21,9,1,to_date('04-12-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(21,7,7,to_date('04-12-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(21,9,7,to_date('04-12-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(22,2,3,to_date('09-06-2019','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(22,5,3,to_date('09-06-2019','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(22,12,3,to_date('09-06-2019','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(23,2,4,to_date('09-06-2019','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(23,5,4,to_date('09-06-2019','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(23,12,4,to_date('09-06-2019','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(24,2,6,to_date('10-08-2021','dd-mm-yyyy'));		

insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(24,5,6,to_date('10-08-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(24,9,6,to_date('10-08-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(24,12,6,to_date('10-08-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(25,2,8,to_date('10-08-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(25,5,8,to_date('10-08-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(25,7,8,to_date('10-08-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(25,12,8,to_date('10-08-2021','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(26,2,11,to_date('12-02-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(26,5,11,to_date('12-02-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,
values(26,7,11,to_date('12-02-2020','dd-mm-yyyy'));		
insert into melodie_in_playlist_utilizator(melodie_id, data_adaugare)	playlist_id,	utilizator_id,

```
values(26,12,11,to_date('12-02-2020','dd-mm-yyyy'));
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(1,1);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(1,2);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(1,4);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(1,9);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(2,1);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(2,5);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(2,12);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(3,6);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(3,7);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(4,3);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(4,4);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(5,8);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
```

```
values(6,10);
```

```
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
values(7,2);
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
values(7,5);
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
values(7,9);
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
values(8,7);
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
values(9,1);
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
values(10,8);
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
values(1,5);
insert into utilizator_apreciaza_playlist(utilizator_id,playlist_id)
values(1,12);
```

```
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(1,1,1,to_date('11-05-2023','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(2,2,2,to_date('10-09-2022','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(3,3,5,to_date('10-10-2022','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
```

```

values(4,4,7,to_date('04-12-2021','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(5,5,10,to_date('09-12-2021','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(6,6,12,to_date('31-12-2021','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(7,7,14,to_date('28-01-2022','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(8,8,15,to_date('28-02-2022','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(9,9,16,to_date('04-06-2022','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(10,10,21,to_date('14-08-2022','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(11,11,26,to_date('04-12-2022','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(12,1,14,to_date('04-12-2021','dd-mm-yyyy'));
insert into istoric_melodii(istoric_melodii_id, utilizator_id, melodie_id, data_ascultare)
values(13,7,15,to_date('04-12-2021','dd-mm-yyyy'));

```

```

insert into podcast_istoric(podcast_istoric_id, utilizator_id, episod_id, data_ascultare)
values(1,1,1,to_date('11-05-2023','dd-mm-yyyy'));
insert into podcast_istoric(podcast_istoric_id, utilizator_id, episod_id, data_ascultare)
values(2,1,2,to_date('14-05-2023','dd-mm-yyyy'));
insert into podcast_istoric(podcast_istoric_id, utilizator_id, episod_id, data_ascultare)
values(3,3,5,to_date('13-04-2023','dd-mm-yyyy'));

```

```

insert into podcast_istoric(podcast_istoric_id, utilizator_id, episod_id, data_ascultare)
values(4,4,4,to_date('04-12-2021','dd-mm-yyyy'));

insert into podcast_istoric(podcast_istoric_id, utilizator_id, episod_id, data_ascultare)
values(5,5,8,to_date('04-04-2020','dd-mm-yyyy'));

insert into podcast_istoric(podcast_istoric_id, utilizator_id, episod_id, data_ascultare)
values(6,6,10,to_date('19-10-2020','dd-mm-yyyy'));

insert into podcast_istoric(podcast_istoric_id, utilizator_id, episod_id, data_ascultare)
values(7,7,8,to_date('13-09-2020','dd-mm-yyyy'));

insert into podcast_istoric(podcast_istoric_id, utilizator_id, episod_id, data_ascultare)
values(8,8,9,to_date('13-04-2023','dd-mm-yyyy'));

insert into podcast_istoric(podcast_istoric_id, utilizator_id, episod_id, data_ascultare)
values(9,9,9,to_date('20-04-2023','dd-mm-yyyy'));

insert into podcast_istoric(podcast_istoric_id, utilizator_id, episod_id, data_ascultare)
values(10,10,12,to_date('13-03-2023','dd-mm-yyyy'));

insert into podcast_istoric(podcast_istoric_id, utilizator_id, episod_id, data_ascultare)
values(11, 11, 15,to_date('12-12-2020','dd-mm-yyyy'));

```

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

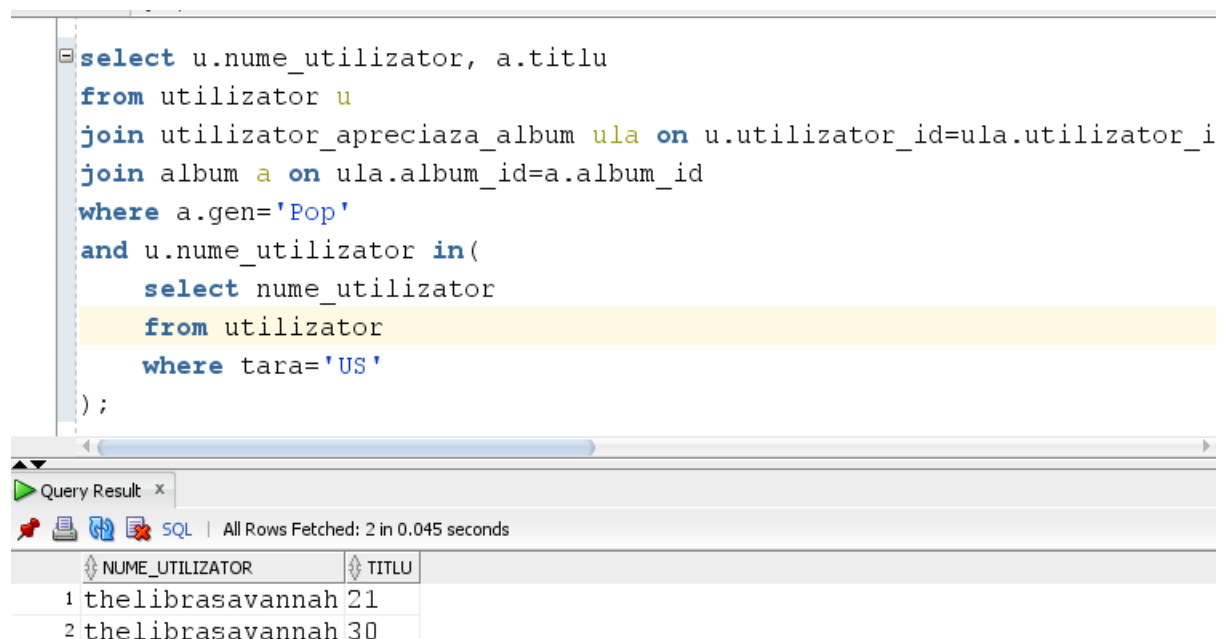
- subcereri sincronizate în care intervin cel puțin 3 tabele
- subcereri nesincronizate în clauza FROM
- grupări de date cu subcereri nesincronizate în care intervin cel puțin 3 tabele, funcții grup, filtrare la nivel de grupuri(în cadrul aceleiași cereri)
- ordonări și utilizarea funcțiilor NVL și DECODE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

a) subcerere sincronizată în care intervin 3 tabele

Limbaaj natural: Să se afișeze numele de utilizator și titlurile albumelor pentru utilizatorii care sunt din Statele Unite ale Americii (US) și care au apreciat albume care au genul muzical ‘Pop’.

Implementare SQL:

```
select u.ume_utilizator, a.titlu
from utilizator u
join utilizator_apreciaza_album ula on u.utilizator_id=ula.utilizator_id
join album a on ula.album_id=a.album_id
where a.gen='Pop'
and u.ume_utilizator in(
    select ume_utilizator
    from utilizator
    where tara='US'
);
```



Utilizatorul ‘thelibrasavannah’ a apreciat albumele ‘21’ și ‘30’.

Acestea au genul 'Pop':

	ALBUM_ID	GEN	PRET	DATA_LANSARE	ARTIST_ID	TITLU
1	1	R&B	69.99	04-NOV-16	1	HERE
2	2	R&B	59.99	28-FEB-06	2	In My Own Words
3	3	Pop	45.32	24-JAN-11	3	21
4	4	Indie	39.99	20-NOV-15	3	25
5	5	Pop	40.99	19-NOV-21	3	30

b) subcerere nesincronizată în clauza FROM:

Limbaaj natural: Să se afișeze numele utilizatorilor alături de numărul de playlist-uri pe care aceștia le-au apreciat.

Implementare SQL:

```
SELECT u.ume_utilizator, t.playlist_nr
FROM utilizator u
CROSS JOIN (
    SELECT utilizator_id, COUNT(*) as playlist_nr
    FROM utilizator_apreciaza_playlist
    GROUP BY utilizator_id
) t
WHERE u.utilizator_id = t.utilizator_id;
```

```

SELECT u.ume_utilizator, t.playlist_nr
FROM utilizator u
CROSS JOIN (
    SELECT utilizator_id, COUNT(*) as playlist_nr
    FROM utilizator_apreciaza_playlist
    GROUP BY utilizator_id
) t
WHERE u.utilizator_id = t.utilizator_id;

```

Query Result x

SQL | All Rows Fetched: 10 in 0.004 seconds

	NUME_UTILIZATOR	PLAYLIST_NR
1	maral33	6
2	thelibrasavannah	3
3	boatelcy	2
4	Finalka	2
5	Johnadbo	1
6	Allotikri	1
7	Myzgeo	3
8	Pltoex	1
9	Matweeknd	1
10	Fletch04	1

tabelul UTILIZATOR_APRECIAZA_PLAYLIST:

	UTILIZATOR_ID	PLAYLIST_ID
1	1	1
2	1	2
3	1	4
4	1	9
5	2	1
6	2	5
7	2	12
8	3	6
9	3	7
10	4	3
11	4	4
12	5	8
13	6	10
14	7	2
15	7	5
16	7	9
17	8	7
18	9	1
19	10	8
20	1	5
21	1	12

c) grupări de date cu subcereri nesincronizate și funcții grup.

Limbaaj natural: Să se afișeze id-ul artistului, numele acestuia și durata medie a melodiilor sale, pentru acei artiști care au lansat albume care sunt urmărite de același număr de utilizatori distincți precum numărul total de albume pe care le-au lansat.

Implementare SQL:

```
select a.artist_id, a.nume, avg(s.durata) as medie_durata
from artist a
join album al on a.artist_id=al.artist_id
join melodie s on al.album_id=s.album_id
group by a.artist_id, a.nume
having count(distinct al.album_id) = (select count(*) from utilizator_urmareste_artist ufa
where ufa.artist_id=a.artist_id
);
```

```
select a.artist_id, a.num, avg(s.durata) as medie_durata
from artist a
join album al on a.artist_id=al.artist_id
join melodie s on al.album_id=s.album_id
group by a.artist_id, a.num
having count(distinct al.album_id) = (select count(*) from utilizator_u
    where ufa.artist_id=a.artist_id
);
```

[illegible]

	ARTIST_ID	ALBUM_ID
1	1	1
2	2	2
3	3	3
4	3	4
5	3	5
6	4	6
7	4	7
8	4	8
9	4	9
10	5	10
11	5	11

Verificare:

deci artistul cu id-ul 1, are un singur album,

	UTILIZATOR_ID	ALBUM_ID
1	1	1
2	1	2
3	1	7
4	1	11
5	2	3
6	2	5
7	2	8
8	5	10
9	6	10
10	7	10
11	10	10
12	10	11

Şi

un singur utilizator a apreciat albumul respectiv.

d) Utilizarea funcțiilor NVL și DECODE

Limbaaj natural: Să se afișeze numele de utilizator, data de naștere și țara unui utilizator, dar înlocuindu-se codul țării cu Netherlands dacă are codul 'NL' sau cu 'Other' pentru celelalte țări.

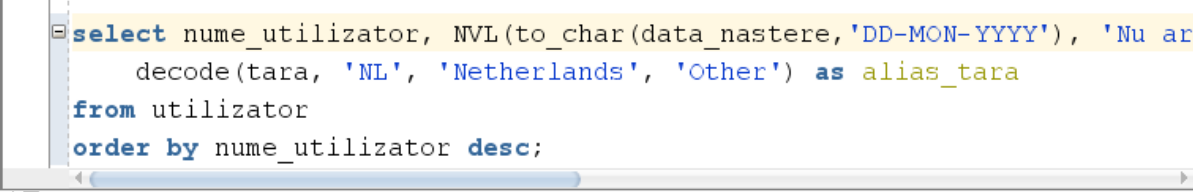
Implementare SQL:

```
select nume_utilizator, NVL(to_char(data_nastere,'DD-MON-YYYY'), 'Nu are data de  
nastere introdusa') as data_nastere,
```

```
decode(tara, 'NL', 'Netherlands', 'Other') as alias_tara
```

```
from utilizator
```

```
order by nume_utilizator desc;
```



The screenshot shows a SQL query editor with the following query:

```
select nume_utilizator, NVL(to_char(data_nastere,'DD-MON-YYYY'), 'Nu are  
data de nastere introdusa') as data_nastere,  
decode(tara, 'NL', 'Netherlands', 'Other') as alias_tara  
from utilizator  
order by nume_utilizator desc;
```

Below the query editor, the 'Query Result' window shows the results of the query. It indicates that all rows were fetched in 0.024 seconds. The results are displayed in a table with three columns: NUME_UTILIZATOR, DATA_NASTERE, and ALIAS_TARA.

	NUME_UTILIZATOR	DATA_NASTERE	ALIAS_TARA
1	vestovlone	12-DEC-2012	Netherlands
2	thelibrasavannah	08-SEP-2003	Other
3	maral33	13-APR-2003	Other
4	boatelcy	27-AUG-2007	Other
5	Pltoex	14-MAY-2005	Other
6	Myzgeo	08-SEP-2003	Other
7	Matweeknd	04-DEC-2003	Other
8	Johnadbo	17-SEP-2000	Other
9	Fletch04	04-JUN-2006	Other
10	Finalka	13-MAR-2002	Other
11	Allotikri	23-FEB-1999	Netherlands

	U...	NUME_UTILIZATOR	PAROLA	EMAIL	DATA_NASTERE	TARA
1		1maral333	parola123	mara.spataru03@gmail.com	13-APR-03	RO
2		2thelibrasavannah	parola123	thelibra@yahoo.com	08-SEP-03	US
3		3boatelcy	parola123	boat.company@gmail.com	27-AUG-07	CA
4		4Finalka	parola123	fin.finn@gmail.com	13-MAR-02	RO
5		5Johnadbo	parola123	john.boa@gmail.com	17-SEP-00	IT
6		6Allotikri	parola123	leon.sam@yahoo.com	23-FEB-99	NL
7		7Myzgeo	parola123	mandyg412@gmail.com	08-SEP-03	RO
8		8Pltoex	parola123	alex.mex@yahoo.com	14-MAY-05	DK
9		9Matweeknd	parola123	matthew.xo@gmail.com	04-DEC-03	FI
10		10Fletch04	parola123	fletcher.a@yahoo.com	04-JUN-06	ES
11		11vlonefour	parola123	fourment.a23@gmail.com	12-DEC-12	NL

-doar cei doi utilizatori au codul țării NL

- e) Utilizare funcții pe șiruri de caractere, pe date calendaristice, a unei expresii CASE, și utilizarea blocului de cerere (clauza WITH) (funcțiile pe șiruri de caractere: SUBSTR, UPPER; și cele pe date calendaristice TO_CHAR pentru formatare și EXTRACT)

Limbaaj natural: Să se afișeze primele 3 litere din titlul unei melodii, data de lansare formatată și statusul de lansare (adică dacă data de lansare a melodiei este mai mare decât data curentă, se afișează 'Upcoming', dacă este egală cu data curentă se afișează 'Released today', altfel se va afișa 'Released'), pentru melodiile care sunt lansate sau urmează să fie, în luna noiembrie și care conțin șirul de caractere 'YOU' în titlu.

Implementare SQL:

```
WITH song_data AS (  
    SELECT s.titlu, s.data_lansare,  
        CASE  
            WHEN s.data_lansare > SYSDATE THEN 'Upcoming'  
            WHEN s.data_lansare = TRUNC(SYSDATE) THEN 'Released today'  
            ELSE 'Released'  
        END AS release_status  
    FROM melodie s  
)  
SELECT SUBSTR(titlu, 1, 3) AS substring_titlu, TO_CHAR(data_lansare,  
'DD-MON-YYYY') AS data_formatata, release_status  
FROM song_data  
WHERE UPPER(titlu) LIKE '%YOU%'  
AND EXTRACT(MONTH FROM data_lansare) = 11;
```

```

WITH song_data AS (
    SELECT s.titlu, s.data_lansare,
    CASE
        WHEN s.data_lansare > SYSDATE THEN 'Upcoming'
        WHEN s.data_lansare = TRUNC(SYSDATE) THEN 'Released today'
        ELSE 'Released'
    END AS release_status
    FROM melodie s
)
SELECT SUBSTR(titlu, 1, 3) AS substring_titlu, TO_CHAR(data_lansare, 'D
FROM song_data
WHERE UPPER(titlu) LIKE '%YOU%'
    AND EXTRACT(MONTH FROM data_lansare) = 11;

```

Query Result x

SQL | All Rows Fetched: 1 in 0.002 seconds

	SUBSTRING_TITLU	DATA_FORMATATA	RELEASE_STATUS
1	You	10-NOV-2003	Released

Acestea sunt toate melodiile care conțin 'you' în titlu:

	MELODIE_ID	TITLU	DURATA	DATA_LANSARE	ALBUM_ID	ARTIST_ID
1	1	If I Ain't Got You	228	14-AUG-20	1	1
2	8	You don't know my name	366	10-NOV-03	1	1
3	11	Like You'll Never See Me Again	315	25-OCT-07	1	1
4	12	Because of You	266	13-FEB-07	(null)	2

13. Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri.

- Actualizarea prețului albumului cu id-ul 3 cu media prețurilor celorlalte albume lansate de același artist.

Implementare SQL:

UPDATE ALBUM

SET pret = (

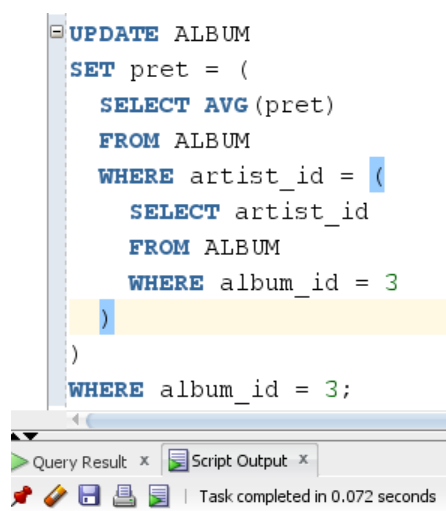
SELECT AVG(pret)

FROM ALBUM


```

WHERE artist_id = (
    SELECT artist_id
    FROM ALBUM
    WHERE album_id = 3
)
)
WHERE album_id = 3;

```



1 row updated.

-albumele 3, 4,5 aparțin aceluiași artist

	ALBUM_ID	GEN	PRET	DATA_LANSARE	ARTIST_ID	TITLU
1	1	R&B	69.99	04-NOV-16	1	HERE
2	2	R&B	59.99	28-FEB-06	2	In My Own Words
3	3	Pop	45.32	24-JAN-11	3	21
4	4	Indie	39.99	20-NOV-15	3	25
5	5	Pop	40.99	19-NOV-21	3	30

astfel prețul devine media celor 3 albume (se transformă în 45.32)

b) actualizarea datei de lansare a primei melodii cu ultima data de lansare dintre toate melodiile ale aceluiași artist

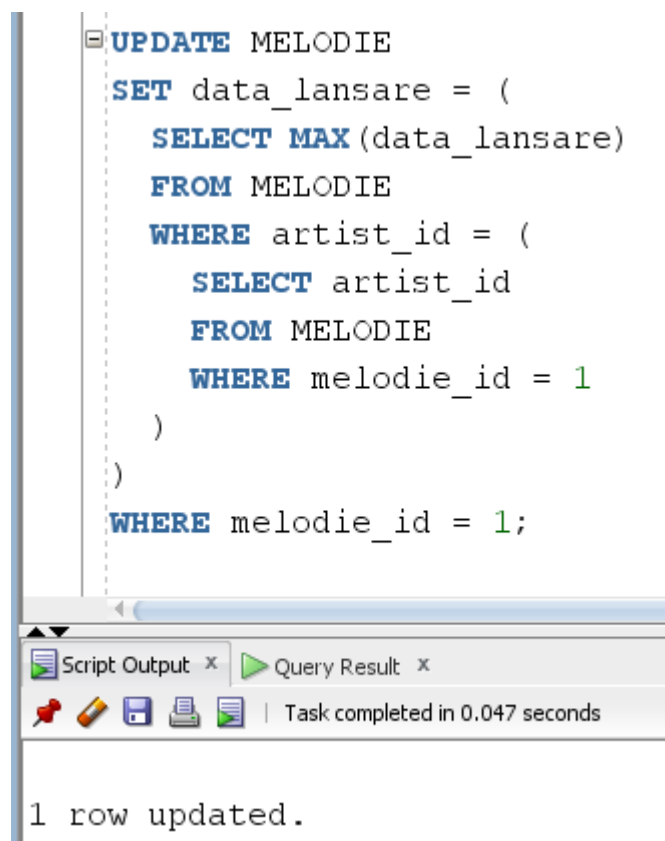
```
UPDATE MELODIE
```

```
SET data_lansare = (
```

```

SELECT MAX(data_lansare)
FROM MELODIE
WHERE artist_id = (
    SELECT artist_id
    FROM MELODIE
    WHERE melodie_id = 1
)
)
WHERE melodie_id = 1;

```



The screenshot shows a SQL IDE interface. The main editor displays an SQL UPDATE statement. Below the editor, there is a toolbar with icons for saving, running, and other functions. A status bar indicates that the task was completed in 0.047 seconds. At the bottom, the output shows that 1 row was updated.

```

UPDATE MELODIE
SET data_lansare = (
    SELECT MAX(data_lansare)
    FROM MELODIE
    WHERE artist_id = (
        SELECT artist_id
        FROM MELODIE
        WHERE melodie_id = 1
    )
)
WHERE melodie_id = 1;

```

Script Output x Query Result x

Task completed in 0.047 seconds

1 row updated.

-> data inițială de lansare a primei melodii a fost 17.02.2004, ultima dată de lansare a primului artist este 14.08.2020, deci după update, prima melodie ar trebui să aibă aceeași dată de lansare

	MELODIE_ID	TITLU		DURATA	DATA_LANSARE	ALBUM_ID	ARTIST_ID
1	1	If I Ain't Got You	.	228	14-AUG-20	1	1
2	2	So Done	.	236	14-AUG-20	1	1
3	3	Girl on Fire	.	224	22-NOV-12	1	1
4	4	My Boo	.	223	29-AUG-04	1	1
5	5	No One	.	254	09-OCT-07	1	1
6	6	Fallin'	.	210	10-APR-01	1	1
7	7	Empire State of Mind	.	216	20-OCT-09	1	1
8	8	You don't know my name	.	366	10-NOV-03	1	1
9	9	Slow Down	.	258	02-DEC-03	1	1

c) actualizarea numelui de utilizator a celui care a apreciat cele mai multe albume

Implementare SQL:

UPDATE UTILIZATOR

SET nume_utilizator= 'mara1333'

WHERE utilizator_id = (

SELECT utilizator_id

FROM (

SELECT utilizator_id

FROM UTILIZATOR_APRECIAZA_ALBUM

GROUP BY utilizator_id

ORDER BY COUNT(*) DESC

)

WHERE ROWNUM = 1

);

```

UPDATE UTILIZATOR
SET nume_utilizator= 'mara1333'
WHERE utilizator_id = (
    SELECT utilizator_id
    FROM (
        SELECT utilizator_id
        FROM UTILIZATOR_APRECIAZA_ALBUM
        GROUP BY utilizator_id
        ORDER BY COUNT(*) DESC
    )
    WHERE ROWNUM = 1
);

```

Script Output x

Task completed in 0.084 seconds

1 row updated.

	USER_ID	ALBUM_ID
1	1	1
2	1	2
3	1	7
4	1	11
5	2	3
6	2	5
7	2	8
8	5	10
9	6	10
10	7	10
11	10	10
12	10	11

->utilizatorul cu id-ul 1 este cel care a apreciat cele mai multe albume

	UTILIZATOR_ID	NUME_UTILIZATOR	PAROLA
1	1	mara1333	parola12
2	2	thelibrasavannah	parola12
3	3	boatelcy	parola12
4	4	Pinella	parola12

d) suprimarea datelor: ștergem melodiile din tabelul `melodie_in_playlist_utilizator`, acolo unde playlist-ul este unul din cel care are cele mai puține melodii.

Implementare SQL:

```
DELETE FROM MELODIE_IN_PLAYLIST_UTILIZATOR
WHERE playlist_id = (
  SELECT playlist_id
  FROM (
    SELECT playlist_id, COUNT(*) AS song_count
    FROM MELODIE_IN_PLAYLIST_UTILIZATOR
    GROUP BY playlist_id
    ORDER BY song_count
    FETCH FIRST 1 ROW ONLY
  ) subquery
);
```

```

DELETE FROM MELODIE_IN_PLAYLIST_UTILIZATOR
WHERE playlist_id = (
    SELECT playlist_id
    FROM (
        SELECT playlist_id, COUNT(*) AS song_count
        FROM MELODIE_IN_PLAYLIST_UTILIZATOR
        GROUP BY playlist_id
        ORDER BY song_count
        FETCH FIRST 1 ROW ONLY
    ) subquery
);

```

Script Output x Query Result x

Task completed in 0.051 seconds

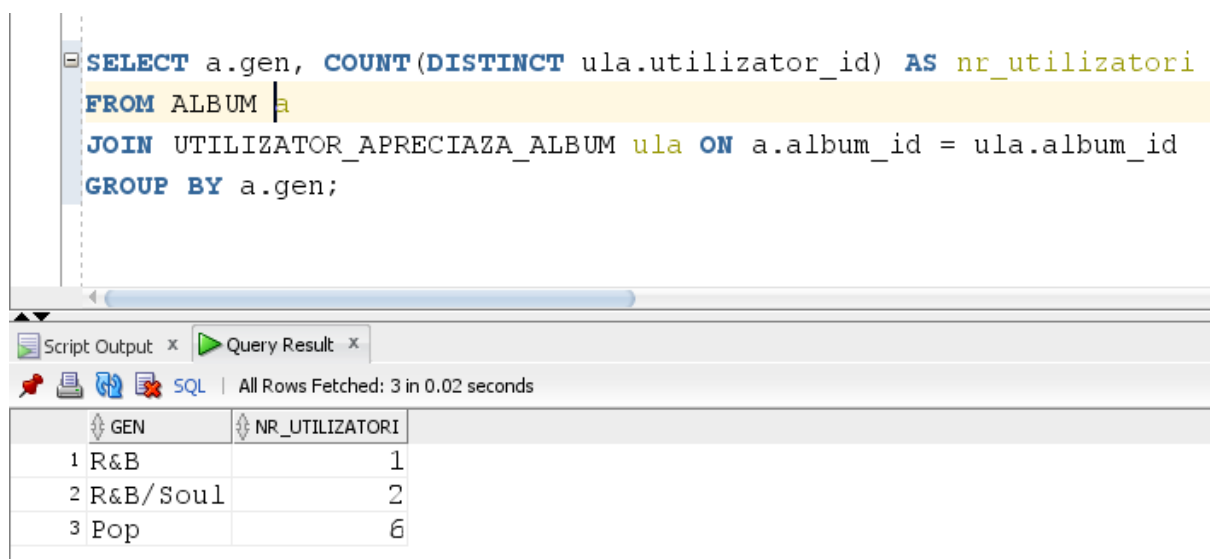
5 rows deleted.

->playlist-ul cu id-ul 9 avea cele mai puține melodii

	MELODIE_ID	PLAYLIST...	UTILIZATOR_ID	DATA_ADAUGARE
91	25	7	8	10-AUG-21
92	26	7	11	12-FEB-20
93	4	10	1	13-APR-21
94	4	10	2	13-APR-21
95	20	10	7	04-DEC-21
96	20	10	1	04-DEC-21
97	15	10	6	06-JUN-21
98	14	10	5	05-MAY-20
99	4	10	3	13-APR-21
100	5	10	4	10-AUG-20

14. Crearea unei vizualizări complexe. Dați un exemplu de operație LMD permisă pe vizualizarea respectivă și un exemplu de operație LMD nepermisă.

```
SELECT a.gen, COUNT(DISTINCT ula.utilizator_id) AS nr_utilizatori
FROM ALBUM a
JOIN UTILIZATOR_APRECIAZA_ALBUM ula ON a.album_id = ula.album_id
GROUP BY a.gen;
```



The screenshot shows a SQL query editor with the following query:

```
SELECT a.gen, COUNT(DISTINCT ula.utilizator_id) AS nr_utilizatori
FROM ALBUM a
JOIN UTILIZATOR_APRECIAZA_ALBUM ula ON a.album_id = ula.album_id
GROUP BY a.gen;
```

Below the query editor, the results are displayed in a table with two columns: GEN and NR_UTILIZATORI. The table shows three rows of data:

GEN	NR_UTILIZATORI
1 R&B	1
2 R&B/Soul	2
3 Pop	6

->această vizualizare permite afișarea numărului de utilizatori care au apreciat albume de diferite genuri muzicale

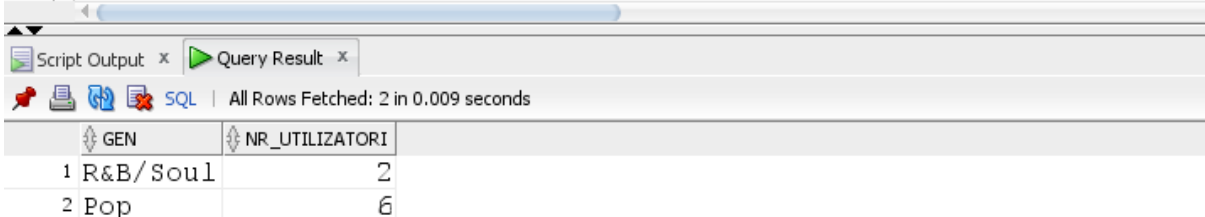
Operatii LMD permise: clauza HAVING. Putem modifica vizualizarea astfel încât să ne arate numărul de utilizatori care au apreciat albume de diferite genuri, acolo unde există măcar 2 aprecieri pentru genul respectiv.

Implementare SQL:

```
SELECT a.gen, COUNT(DISTINCT ula.utilizator_id) AS nr_utilizatori
FROM ALBUM a
JOIN UTILIZATOR_APRECIAZA_ALBUM ula ON a.album_id = ula.album_id
GROUP BY a.gen
```

HAVING COUNT(DISTINCT ula.utilizator_id) > 1;

```
SELECT a.gen, COUNT(DISTINCT ula.utilizator_id) AS nr_utilizatori
FROM ALBUM a
JOIN UTILIZATOR_APRECIAZA_ALBUM ula ON a.album_id = ula.album_id
GROUP BY a.gen
HAVING COUNT(DISTINCT ula.utilizator_id) > 1;
```



The screenshot shows a database query result in a window titled 'Query Result'. The query is a SQL statement that selects the genre (gen) and the count of distinct users (nr_utilizatori) who have appreciated albums of that genre. The result is displayed in a table with two columns: GEN and NR_UTILIZATORI. There are two rows: 1 R&B/Soul with 2 users, and 2 Pop with 6 users. The window also shows a 'Script Output' tab and a status bar indicating 'All Rows Fetched: 2 in 0.009 seconds'.

GEN	NR_UTILIZATORI
1 R&B/Soul	2
2 Pop	6

Operatii LMD nepermise: Nu putem folosi UPDATE, adică, nu putem actualiza numărul de utilizatori care au apreciat albume care au un anumit gen muzical în vizualizare. De exemplu: am putea vrea să mărim numărul de utilizatori care au apreciat albume care au genul setat ca 'Pop'. Aceasta operație nu este permisă în vizualizare.

15. Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația outer-join pe minim 4 tabele, o cerere ce utilizează operația division și o cerere care implementează analiza top-n.

a) outer-join

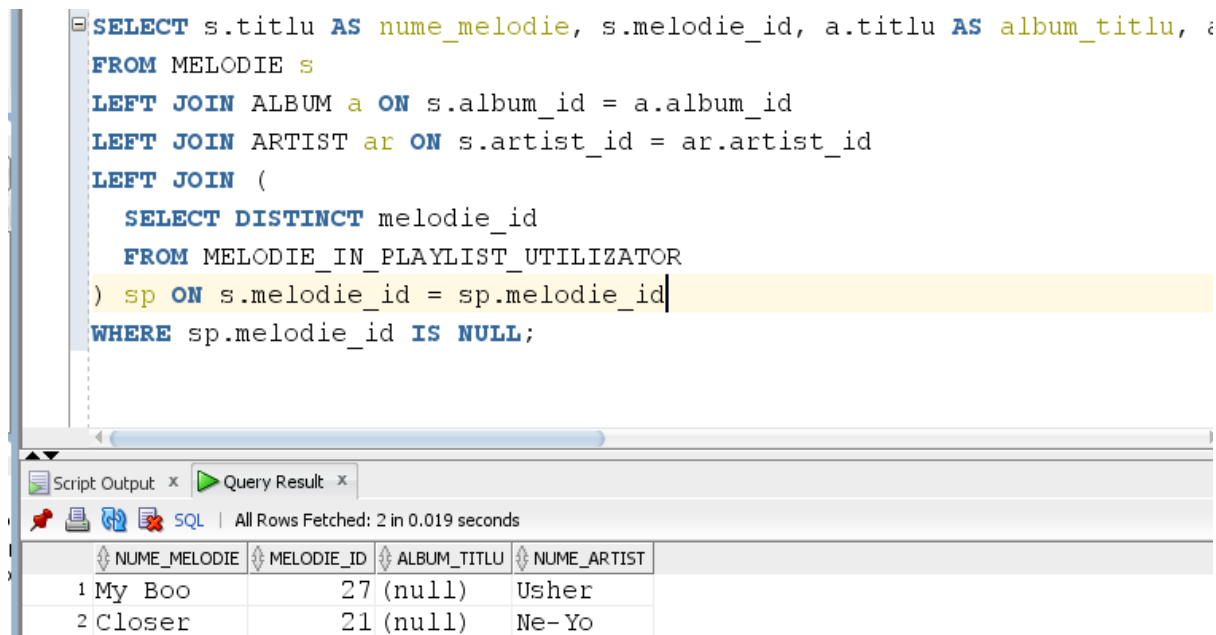
Pentru realiza această cerere și pentru a obține mai mult de un rezultat, am șters din tabelul melodii_in_playlist_utilizator 2 înregistrări (am observat că melodia cu id-ul 21 era doar de două ori în vreun playlist), iar cum înregistrarea cu numărul 27 din tabelul MELODII, a fost adăugată doar pentru a exemplifica un exercițiu anterior, aceasta nu a fost adăugată în niciun playlist.

```
delete from songs_in_playlist_user
where song_id =21;
```

Limbaj natural: Să se afișeze toate melodiile alături de albumul și artistul asociat, pentru melodiile care nu sunt prezente în niciun playlist.

Implementare SQL:

```
SELECT s.titlu AS nume_melodie, s.melodie_id, a.titlu AS album_titlu, ar.num AS
nume_artist
FROM MELODIE s
LEFT JOIN ALBUM a ON s.album_id = a.album_id
LEFT JOIN ARTIST ar ON s.artist_id = ar.artist_id
LEFT JOIN (
    SELECT DISTINCT melodie_id
    FROM MELODIE_IN_PLAYLIST_UTILIZATOR
) sp ON s.melodie_id = sp.melodie_id
WHERE sp.melodie_id IS NULL;
```



The screenshot shows a database query editor with the following SQL query:

```
SELECT s.titlu AS nume_melodie, s.melodie_id, a.titlu AS album_titlu, ar.num AS
nume_artist
FROM MELODIE s
LEFT JOIN ALBUM a ON s.album_id = a.album_id
LEFT JOIN ARTIST ar ON s.artist_id = ar.artist_id
LEFT JOIN (
    SELECT DISTINCT melodie_id
    FROM MELODIE_IN_PLAYLIST_UTILIZATOR
) sp ON s.melodie_id = sp.melodie_id
WHERE sp.melodie_id IS NULL;
```

Below the query editor, the 'Query Result' tab is active, showing the results of the query. The status bar indicates 'All Rows Fetched: 2 in 0.019 seconds'.

	NUME_MELODIE	MELODIE_ID	ALBUM_TITLU	NUME_ARTIST
1	My Boo	27	(null)	Usher
2	Closer	21	(null)	Ne-Yo

b) operația division

Limbaj natural: Să se afișeze toate playlist-urile care au mai mult de 10 melodii adăugate.

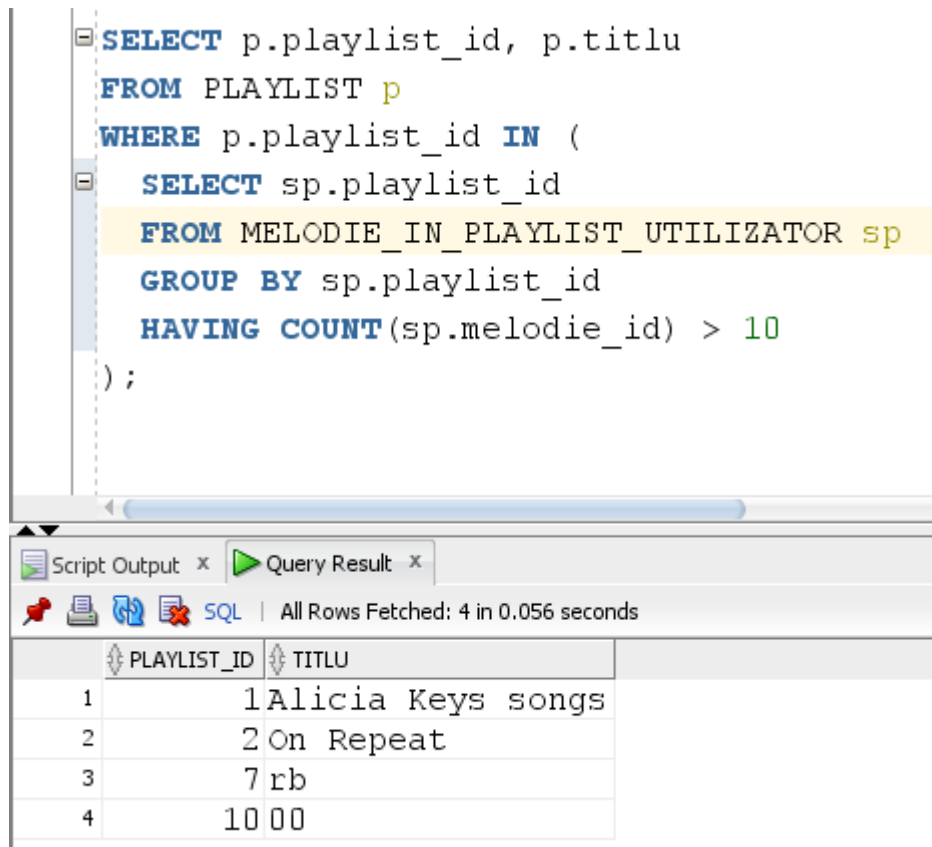
Implementare SQL:

```
SELECT p.playlist_id, p.titlu
FROM PLAYLIST p
WHERE p.playlist_id IN (
    SELECT sp.playlist_id
```

```

FROM MELODIE_IN_PLAYLIST_UTILIZATOR sp
GROUP BY sp.playlist_id
HAVING COUNT(sp.melodie_id) > 10
);

```



c) analiza top-n

Limbaaj natural: Să se afișeze primele 5 cele mai ascultate melodii și numărul de ascultări pentru fiecare.

Implementare SQL:

```

SELECT s.titlu AS nume_cantec, COUNT(sh.melodie_id) AS nr_ascultari
FROM melodie s
LEFT JOIN istoric_melodii sh ON s.melodie_id = sh.melodie_id
GROUP BY s.melodie_id, s.titlu
ORDER BY nr_ascultari DESC
FETCH FIRST 5 ROWS ONLY;

```

```

SELECT s.titlu AS nume_cantec, COUNT(sh.melodie_id) AS nr_ascultari
FROM melodie s
LEFT JOIN istoric_melodii sh ON s.melodie_id = sh.melodie_id
GROUP BY s.melodie_id, s.titlu
ORDER BY nr_ascultari DESC
FETCH FIRST 5 ROWS ONLY;

```

	NUME_CANTEC	NR_ASCULTARI
1	Sexy Love	2
2	Miss Independent	2
3	Empire State of Mind	1
4	Underdog	1
5	If I Ain't Got You	1

16. Optimizarea unei cereri, aplicând regulile de optimizare ce derivă din proprietățile operatorilor algebrei relaționale. Cererea va fi exprimată prin expresie algebrică, arbore algebric și limbaj (SQL), atât anterior cât și ulterior optimizării.

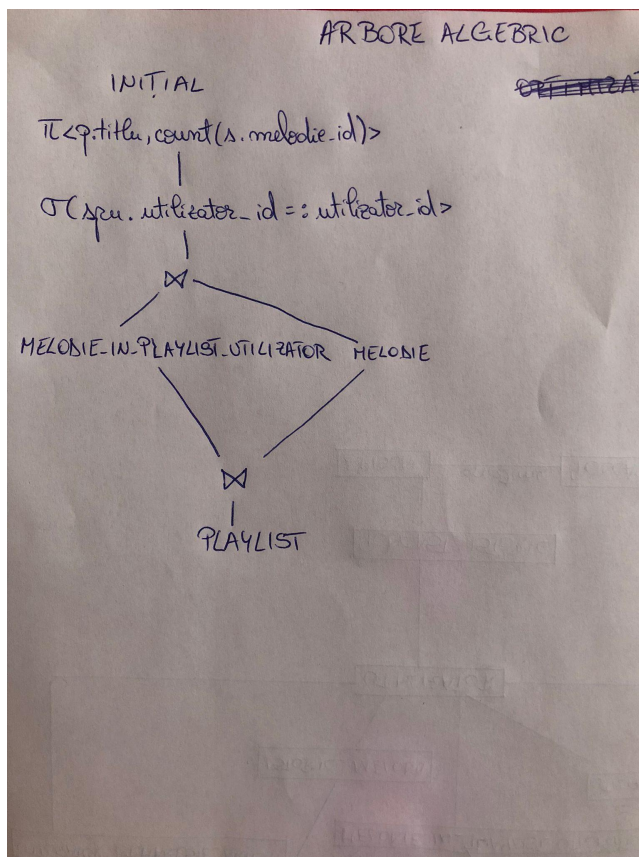
Limbaj natural: vom face operațiile pentru cererea care ar returna numele playlist-urilor și numărul de melodii din fiecare playlist pentru un utilizator anume.

Anterior optimizării:

Expresie algebrică:

$$\begin{aligned}
 &\pi_{\langle \text{p.titlu}, \text{COUNT}(s.melodie_id) \rangle} (\sigma_{\langle \text{spu.utilizator_id} = :utilizator_id \rangle} \\
 &(\text{PLAYLIST} \bowtie_{\langle \text{p.playlist_id} = \text{spu.playlist_id} \rangle} \\
 &(\text{MELODIE_IN_PLAYLIST_UTILIZATOR} \bowtie_{\langle \text{spu.melodie_id} = s.melodie_id} \\
 &\langle \text{MELODIE} \rangle))
 \end{aligned}$$

Arbore algebric:



Limbaaj SQL:

```
SELECT p.titlu AS playlist_nume, COUNT(s.melodie_id) AS nr_mel
FROM PLAYLIST p
JOIN MELODIE_IN_PLAYLIST_UTILIZATOR spu ON p.playlist_id = spu.playlist_id
JOIN MELODIE s ON spu.melodie_id = s.melodie_id
WHERE spu.utilizator_id = :utilizator_id
GROUP BY p.titlu;
```

De exemplu, pentru utilizatorul 1:

```
SELECT p.titlu AS playlist_nume, COUNT(s.melodie_id) AS nr_mel
FROM PLAYLIST p
JOIN MELODIE_IN_PLAYLIST_UTILIZATOR spu ON p.playlist_id = spu.playlist_id
JOIN MELODIE s ON spu.melodie_id = s.melodie_id
WHERE spu.utilizator_id = 1
GROUP BY p.titlu;
```

Script Output x Query Result x

SQL | All Rows Fetched: 4 in 0.002 seconds

PLAYLIST_NUME	NR_MEL
1 Alicia Keys songs	2
2 On Repeat	2
3 rb	3
4 00	2

Optimizare:

Expresie algebrică:

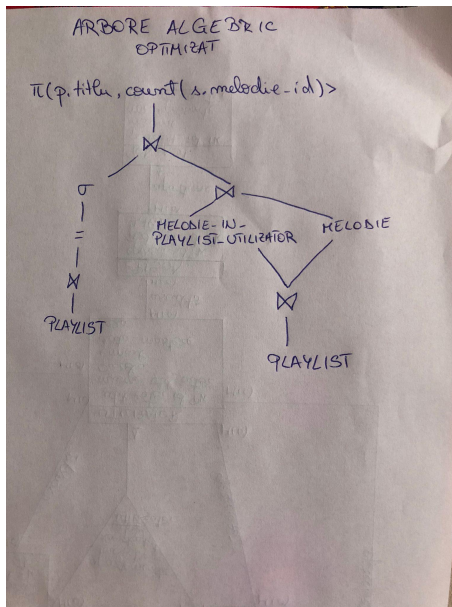
$\pi_{\langle p.name, COUNT(*) \rangle}$

$(\sigma_{\langle spu.user_id = :user_id \rangle}$

$(PLAYLIST \bowtie_{\langle p.playlist_id = spu.playlist_id \rangle}$

$(SONGS_IN_PLAYLIST_USER)))$

Arbore algebric:



Limbaaj SQL:

```
SELECT p.titlu AS playlist_nume, COUNT(*) AS nr_mel
FROM PLAYLIST p
JOIN MELODIE_IN_PLAYLIST_UTILIZATOR spu ON p.playlist_id = spu.playlist_id
WHERE spu.utilizator_id = :utilizator_id
GROUP BY p.titlu;
```

De exemplu, pentru utilizatorul 1

```
SELECT p.titlu AS playlist_nume, COUNT(*) AS nr_mel
FROM PLAYLIST p
JOIN MELODIE_IN_PLAYLIST_UTILIZATOR spu ON p.playlist_id = spu.playlist_id
WHERE spu.utilizator_id = :utilizator_id
GROUP BY p.titlu;
```

PLAYLIST_NUME	NR_MEL
1 Alicia Keys songs	2
2 On Repeat	2
3 rb	3
4 00	2

17. a) Realizarea normalizării BCNF, FN4, FN5.

Normalizare BCNF

Forma normală BCNF presupune ca pentru orice dependență funcțională non-trivială ($X \rightarrow Y$) într-un tabel, X trebuie să fie o supercheie.

Primul pas este identificarea dependențelor funcționale în fiecare tabel.

Tabel UTILIZATOR:

- utilizator_id \rightarrow nume_utilizator, parola, email, data_nastere, tara
- nume_utilizator \rightarrow utilizator_id
- email \rightarrow utilizator_id

Tabel ARTIST:

- artist_id \rightarrow nume, varsta, descriere

Tabel ALBUM:

- album_id \rightarrow gen, pret, titlu, data_lansare, artist_id

Tabel UTILIZATOR_APRECIAZA_ALBUM:

- (utilizator_id, album_id) \rightarrow utilizator_id, album_id

Tabel ARTIST_LANSEAZA_ALBUM:

- (artist_id, album_id) \rightarrow artist_id, album_id

Tabel UTILIZATOR_URMARESTE_ARTIST:

- (utilizator_id, artist_id) \rightarrow utilizator_id, artist_id

Tabel MELODIE:

- melodie_id -> nume, descriere, durata, data_lansare, album_id, artist_id

Tabel ARTIST_CANTA_MELODIE:

- (artist_id, melodie_id) -> artist_id, melodie_id

Tabel PLAYLIST:

- playlist_id -> nume, data_creare

Tabel MELODIE_IN_PLAYLIST_UTILIZATOR:

- (melodie_id, playlist_id, utilizator_id) -> melodie_id, playlist_id, utilizator_id

Tabel UTILIZATOR_APRECIAZA_PLAYLIST:

- (utilizator_id, playlist_id) -> utilizator_id, playlist_id

Tabel ISTORIC_MELODII:

- istoric_melodii_id -> utilizator_id, melodie_id, data_ascultare

Tabel PODCAST:

- podcast_id -> nume, nr_episoade, nume_autor

Tabel EPISOD:

- episod_id -> podcast_id, titlu

Tabel PODCAST_ISTORIC:

- podcast_istoric_id -> utilizator_id, episod_id, data_ascultare

Pasul 2: verificăm dacă fiecare dependență funcțională este o dependență pe o cheie candidat. În BCNF, partea din stânga dintr-o dependență funcțională trebuie să fie o supercheie. Analizăm dependențele funcționale pentru fiecare tabel și stabilim dacă sunt în BCNF.

Putem observa că toate tabelele respectă BCNF, mai puțin tabelul UTILIZATOR (dependența funcțională nume_utilizator -> utilizator_id)

Pasul 3

Descompunem tabelul pentru a șterge acea dependență funcțională. În acest caz, trebuie să facem un tabel separat pentru atributul “nume_utilizator” pentru a ne asigura că dependența nume_utilizator-> utilizator_id satisface BCNF.

Tabelul UTILIZATOR va pierde coloana cu nume_utilizator, și vom crea tabelul NUME_UTILIZATOR astfel, cheia primară va fi nume_utilizator, iar cheia străină va fi utilizator_id.

Descompunând astfel, tabelul UTILIZATOR, ne asigurăm că dependența funcțională nume_utilizator-> utilizator_id este validă acum în tabelul nou creat.

select * from nume_utilizatori nu

join utilizator u on u.utilizator_id=nu.utilizator_id;

NUME_UTILIZATOR	UTILIZATOR_ID	UTILIZATOR_ID_1	PAROLA	EMAIL	DATA_NASTERE	TARA
1 mara133	1	1	parola123	mara.spataru03@gmail.com	13-APR-03	RO
2 thelibrasavannah	2	2	parola123	thelibra@yahoo.com	08-SEP-03	US
3 boatelcy	3	3	parola123	boat.company@gmail.com	27-AUG-07	CA
4 Finalka	4	4	parola123	fin.finn@gmail.com	13-MAR-02	RO
5 Johnadbo	5	5	parola123	john.boa@gmail.com	17-SEP-00	IT
6 Allotikri	6	6	parola123	leon.sam@yahoo.com	23-FEB-99	NL
7 Myzgeo	7	7	parola123	mandyg412@gmail.com	08-SEP-03	RO
8 Pltoex	8	8	parola123	alex.mex@yahoo.com	14-MAY-05	DK
9 Matweeknd	9	9	parola123	matthew.xo@gmail.com	04-DEC-03	FI
10 Fletch04	10	10	parola123	fletcher.a@yahoo.com	04-JUN-06	ES
11 vestovlone	11	11	parola123	fourment.a23@gmail.com	12-DEC-12	NL

..	PAROLA	EMAIL	DATA_NASTERE	TARA
1	1 parola123	mara.s...	13-APR-03	RO
2	2 parola123	thelib...	08-SEP-03	US
3	3 parola123	boat.c...	27-AUG-07	CA
4	4 parola123	fin.finn@gmail.com	13-MAR-02	RO
5	5 parola123	john.b...	17-SEP-00	IT
6	6 parola123	leon.s...	23-FEB-99	NL
7	7 parola123	mandyg...	08-SEP-03	RO
8	8 parola123	alex.m...	14-MAY-05	DK
9	9 parola123	matthe...	04-DEC-03	FI
10	10 parola123	fletch...	04-JUN-06	ES
11	11 parola123	fourme...	12-DEC-12	NL

	NUME_UTILIZATOR	UTILIZATOR_ID
1	maral33	1
2	thelibrasavannah	2
3	boatelcy	3
4	Finalka	4
5	Johnadbo	5
6	Allotikri	6
7	Myzgeo	7
8	Pltoex	8
9	Matweeknd	9
10	Fletch04	10
11	vestovlone	11

Normalizare FN4:

Pentru a aduce tabelele la forma normală FN4, trebuie să identificăm dependențele parțiale și să facem tabele separate pentru acestea.

Primul pas este să identificăm dependențele parțiale, căutăm attribute care depind doar de o parte din cheia candidat. Apoi, determinăm dacă vreun atribut care nu face parte din cheie depinde de o parte din cheia candidat. De vreme ce în tabelele noastre toate attributele depind doar de cheia candidat, nu există dependențe parțiale.

Dacă, spre exemplu, atributul gen din tabelul ALBUM ar fi permis stocarea genurilor multiple, adică un album ar fi putut avea atribuite mai multe genuri, acestea ar fi trebuit separate adăugând un tabel separat pentru genurile unui album.

	ALBUM_ID	GEN	PRET	DATA_LANSARE	ARTIST_ID	TITLU
1	1	R&B	69.99	04-NOV-16	1	HERE
2	2	R&B	59.99	28-FEB-06	2	In My Own Words
3	3	Pop	45.32	24-JAN-11	3	21
4	4	Indie	39.99	20-NOV-15	3	25
5	5	Pop	40.99	19-NOV-21	3	30
6	6	R&B	55.99	20-MAR-20	4	After Hours
7	7	R&B	63.99	07-JAN-22	4	Dawn FM
8	8	R&B	31.99	30-MAR-18	4	My dear Melancholy
9	9	R&B	99.99	25-NOV-16	4	Starboy
10	10	Pop	90.99	16-SEP-22	5	Girl Of My Dreams (Deluxe)
11	11	Pop	99.99	09-SEP-20	5	THE S(EX) TAPES

Normalizare FN5

Pentru a normaliza tabelele până la forma normală FN5, trebuie să identificăm dependențele tranzitive și să facem tabele separate pentru ele.

Primul pas este identificarea lor, căutăm attribute care depind de alte attribute ce nu fac parte din cheie, apoi determinăm dacă vreun atribut care nu depinde de cheie depinde de un alt atribut care nu face parte din cheie.

Privind tabelele din modelul de date propus, putem observa că nu există dependențe tranzitive, deci modelul se află deja în forma normală FN5.

De exemplu, dacă am avea un tabel numit ALBUM_ARTIST_ROL în care am memora id-ul unui album, artistul respectiv acestuia, dar și rolul unui artist (vocalist, basist, pianist, etc.), cu cheia primară compusă din album_id și artist_id, acesta ar face ca modelul de date propus să nu mai respecte FN5. Asta se întâmplă deoarece este posibil, datorită acestui tabel, ca un artist să aibă mai multe roluri în cadrul aceluiași album.

	ARTIST_ID	NUME	VARSTA	DESCRIERE
1	1	Alicia Keys	42	American singer, songwriter, and...
2	2	Ne-Yo	43	American singer, songwriter, act...
3	6	Dilja	21	Icelandic singer
4	7	Brunette	21	Armenian singer-songwriter
5	8	anees	30	American singer, rapper, and son...
6	9	Sonder	(null)	Band created by the artist Brent...
7	10	Brent Fa...	27	American R&B singer
8	3	Adele	35	English singer-songwriter
9	4	The Weeknd	33	Canadian singer and songwriter
10	5	Fletcher	29	American singer and songwriter
11	11	Usher	44	American R&B singer

b) aplicarea denormalizării și justificarea acesteia

Denormalizarea este procesul de introducere intenționată a redundanței în designul unei baze de date pentru a îmbunătăți performanța sau a simplifica interogările. Aceasta implică combinarea tabelelor sau duplicarea datelor pentru a reduce necesitatea de a efectua un număr mare de join-uri complexe și pentru a îmbunătăți viteza de extragere a datelor.

Aceasta oferă performanță îmbunătățită a interogărilor: poate simplifica interogările complexe prin reducerea numărului de join-uri necesare. Aceasta poate duce la un timp de execuție mai rapid pentru cererile efectuate frecvent

Extragerea datelor simplificată: poate elimina necesitatea join-urilor și poate facilita extragerea datelor, făcând procesul mai ușor și mai direct. Acest lucru poate fi deosebit de benefic în scenariile în care accesul la date în timp real sau aproape în timp real este crucial.

Reducerea complexității: denormalizarea poate simplifica schema bazei de date prin reducerea numărului de tabele și relații. Aceasta poate face sistemul mai ușor de înțeles, întreținut și modificat.

Cu toate acestea, denormalizarea implică compromisuri. Aceasta introduce redundanță și poate duce la inconsistențe în date dacă nu este gestionată corespunzător.