Project

**Tables:**

Album (Album\_ID, SongCount, name, name, age,)

Artist(Age,StageName,RecordLabel,ArtistID,Description, OriginCountry)

Genre (GenreID,Name)

Playlist (PlaylistID,Name,SongCount,Description,GenreID)

PlayList\_song (SongID, PlaylistID)

Podcast (PodcastID,Name,EpisodeNO,Description,YearReleased,Length,ArtistID)

Song(SongID,Name,YearReleased,RecordLabel,Language,Length,ArtistID,AlbumID)

User (UserID,Email,Name,MembershipPlan)

User\_artist (ArtistID,UserID)

User\_podcast (PodcastID,UserID)

User\_song (SondID,UserID)

**SQL Commands for Table Creation:**

CREATE DATABASE music\_libb;

USE music\_libb;

CREATE TABLE `album` (

`AlbumID` int(11) NOT NULL CHECK (`AlbumID` >= 10000 AND `AlbumID` <= 99999),

`SongCount` int(11) NOT NULL CHECK (`SongCount` > 0),

`name` varchar(255) DEFAULT NULL,

`ArtistID` int(11) NOT NULL CHECK (`ArtistID` >= 1000 AND `ArtistID` <= 9999),

PRIMARY KEY (`AlbumID`),

KEY `ArtistID` (`ArtistID`)

);

CREATE TABLE `artist` (

`Age` int(11) NOT NULL,

`StageName` varchar(255) NOT NULL,

`RecordLabel` varchar(255) DEFAULT NULL,

`ArtistID` int(11) NOT NULL CHECK (`ArtistID` >= 1000 AND `ArtistID` <= 9999),

`Description` text DEFAULT NULL,

`OriginCountry` varchar(255) NOT NULL,

PRIMARY KEY (`ArtistID`)

);

CREATE TABLE `genre` (

`GenreID` int(11) NOT NULL CHECK (`GenreID` > 0 AND `GenreID` < 1000),

`Name` varchar(255) NOT NULL,

PRIMARY KEY (`GenreID`),

UNIQUE KEY `Name` (`Name`)

);

CREATE TABLE `playlist` (

`PlaylistID` int(11) NOT NULL CHECK (`PlaylistID` >= 10000 AND `PlaylistID` <= 99999),

`Name` varchar(255) NOT NULL,

`SongCount` int(11) NOT NULL CHECK (`SongCount` > 0),

`Description` text DEFAULT NULL,

`GenreID` int(11) NOT NULL CHECK (`GenreID` > 0 AND `GenreID` < 1000),

PRIMARY KEY (`PlaylistID`),

KEY `GenreID` (`GenreID`)

);

CREATE TABLE `playlist\_song` (

`SongID` int(11) NOT NULL CHECK (`SongID` >= 100000 AND `SongID` <= 999999),

`PlaylistID` int(11) NOT NULL CHECK (`PlaylistID` >= 10000 AND `PlaylistID` <= 99999),

PRIMARY KEY (`SongID`,`PlaylistID`),

KEY `PlaylistID` (`PlaylistID`),

CONSTRAINT `FK\_playlist\_song\_playlist` FOREIGN KEY (`PlaylistID`) REFERENCES `playlist` (`PlaylistID`),

CONSTRAINT `FK\_playlist\_song\_song` FOREIGN KEY (`SongID`) REFERENCES `song` (`SongID`)

);

CREATE TABLE `podcast` (

`PodcastID` int(11) NOT NULL CHECK (`PodcastID` >= 1000 AND `PodcastID` <= 9999),

`Name` varchar(255) NOT NULL,

`EpisodeNo` int(11) NOT NULL CHECK (`EpisodeNo` > 0),

`Description` text DEFAULT NULL,

`YearReleased` int(11) NOT NULL CHECK (`YearReleased` > 1600 AND `YearReleased` <= 2021),

`Length` time NOT NULL,

`ArtistID` int(11) NOT NULL CHECK (`ArtistID` >= 1000 AND `ArtistID` <= 9999),

PRIMARY KEY (`PodcastID`),

KEY `ArtistID` (`ArtistID`),

CONSTRAINT `FK\_podcast\_artist` FOREIGN KEY (`ArtistID`) REFERENCES `artist` (`ArtistID`)

);

CREATE TABLE `song` (

`SongID` int(11) NOT NULL CHECK (`SongID` >= 100000 AND `SongID` <= 999999),

`Name` varchar(255) NOT NULL,

`YearReleased` int(11) NOT NULL CHECK (`YearReleased` >= 1600 AND `YearReleased` <= 2021),

`RecordLabel` varchar(255) DEFAULT NULL,

`Language` varchar(255) NOT NULL,

`Length` time NOT NULL,

`ArtistID` int(11) NOT NULL CHECK (`ArtistID` >= 1000 AND `ArtistID` <= 9999),

`AlbumID` int(11) NOT NULL CHECK (`AlbumID` >= 10000 AND `AlbumID` <= 99999),

PRIMARY KEY (`SongID`),

KEY `ArtistID` (`ArtistID`),

KEY `AlbumID` (`AlbumID`),

CONSTRAINT `FK\_song\_artist` FOREIGN KEY (`ArtistID`) REFERENCES `artist` (`ArtistID`),

CONSTRAINT `FK\_song\_album` FOREIGN KEY (`AlbumID`) REFERENCES `album` (`AlbumID`)

);

CREATE TABLE `users` (

`UserID` int(11) NOT NULL CHECK (`UserID` > 10000 AND `UserID` <= 99999),

`Email` varchar(255) NOT NULL CHECK (`Email` LIKE '%@%'),

`Name` varchar(255) NOT NULL,

`MembershipPlan` varchar(255) NOT NULL CHECK (`MembershipPlan` IN ('Standard','Premium')),

PRIMARY KEY (`UserID`)

);

CREATE TABLE `user\_artist` (

`ArtistID` int(11) NOT NULL CHECK (`ArtistID` >= 1000 and `ArtistID` <= 9999),

`UserID` int(11) NOT NULL CHECK (`UserID` > 10000 and `UserID` <= 99999),

PRIMARY KEY (`ArtistID`,`UserID`),

KEY `UserID` (`UserID`)

);

CREATE TABLE `user\_friend` (

`UserID` int(11) NOT NULL CHECK (`UserID` > 10000 and `UserID` <= 99999),

`FriendUserID` int(11) NOT NULL CHECK (`FriendUserID` > 10000 and `FriendUserID` <= 99999),

PRIMARY KEY (`UserID`,`FriendUserID`),

KEY `FriendUserID` (`FriendUserID`)

);

CREATE TABLE `user\_playlist` (

`PlaylistID` int(11) NOT NULL CHECK (`PlaylistID` >= 10000 and `PlaylistID` <= 99999),

`UserID` int(11) NOT NULL CHECK (`UserID` > 10000 and `UserID` <= 99999),

PRIMARY KEY (`PlaylistID`,`UserID`),

KEY `UserID` (`UserID`)

);

CREATE TABLE `user\_podcast` (

`PodcastID` int(11) NOT NULL CHECK (`PodcastID` >= 1000 and `PodcastID` <= 9999),

`UserID` int(11) NOT NULL CHECK (`UserID` > 10000 and `UserID` <= 99999),

PRIMARY KEY (`PodcastID`,`UserID`),

KEY `UserID` (`UserID`)

);

CREATE TABLE `user\_song` (

`SongID` int(11) NOT NULL CHECK (`SongID` >= 100000 and `SongID` <= 999999),

`UserID` int(11) NOT NULL CHECK (`UserID` > 10000 and `UserID` <= 99999),

PRIMARY KEY (`SongID`,`UserID`),

KEY `UserID` (`UserID`)

);

***SQL Queries(40 questions)***

1.Retrieve all user informations.

SELECT \* FROM user;

A screen shot of a computer

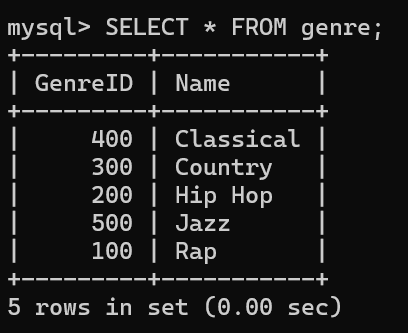
Description automatically generated

A screen shot of a computer

Description automatically generated2.Retrieve Artist information

3.Display all genre names

SELECT \* FROM genre;



4.Display all playlists

SELECT \* FROM playlist;

A screen shot of a black screen

Description automatically generated

5. Display all playlists

A black screen with white text

Description automatically generated

A black screen with white text

Description automatically generated6. DISPLAY ALL PODCAST INFORMATION

7.Display all song details

A screen shot of a computer

Description automatically generated

8.Display the number of albums

A black screen with white text

Description automatically generated

9.Display the number of artist

A black screen with white text

Description automatically generated

10.Display the number of genres

A black screen with white text

Description automatically generated

11.Display the number of playlist

A black screen with white text

Description automatically generated

12. Display the number of songs

A black screen with white text

Description automatically generated

A screen shot of a computer

Description automatically generated13. Display the artist name who’s artistID 7777

A black screen with white text

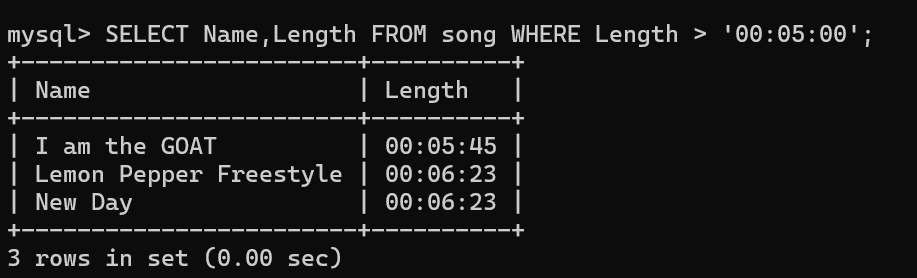
Description automatically generated14.Display genre details whos genre id is 400

15.Display users who have premium membership

A black screen with white text

Description automatically generated

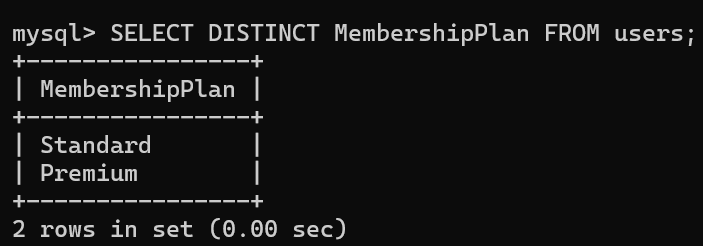
16 Retrieve names of songs longer than 5 mins



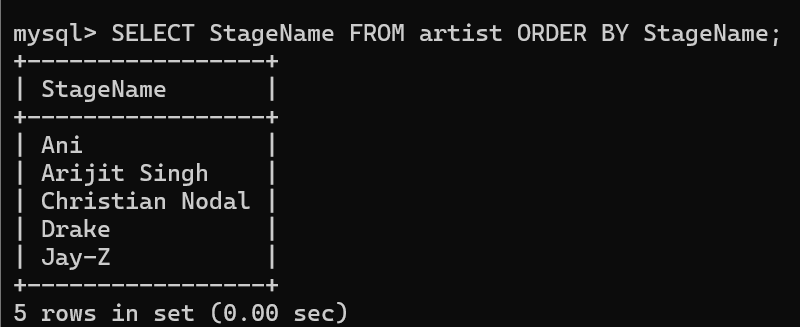
17.Retrieve the names of playlist categorized by genre



18.List the membership of all users



19. Get name of all artist in alphabetical order



20.Get names of all playlist along with their genres

A screen shot of a computer

Description automatically generated

21.Get names of all songs along with their artists.

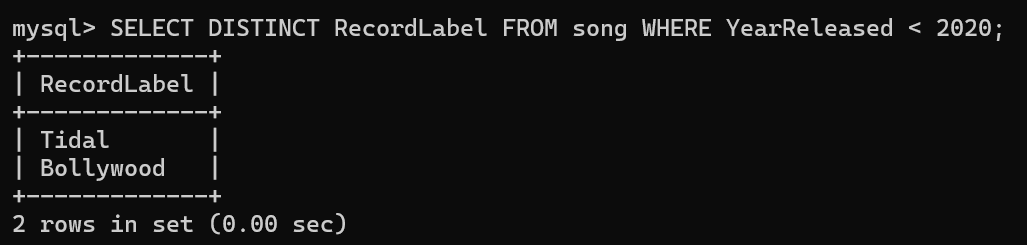


22. Get names of all songs along with their artists

A screen shot of a computer

Description automatically generated

23. Show record label of songs released before year 2020



24. List all users along with their email address

A screen shot of a computer

Description automatically generated

25.Show names of artist who have released podcasts

A black screen with white text

Description automatically generated

26.Show user names along with their friends

A screenshot of a computer

Description automatically generated

27.Retrieve the name of playlist along with their genre

A black screen with white text

Description automatically generated

28. List names of artist along with their their albums

A screen shot of a computer

Description automatically generated

29.Display the max release year of songs each artist

A computer screen shot of a computer program

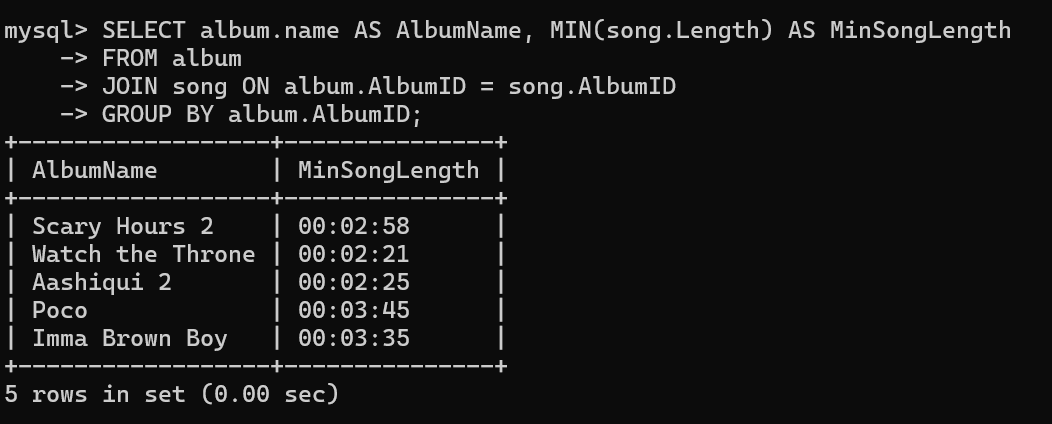
Description automatically generated

30.Max episode number of podcast for each artist

A screen shot of a computer program

Description automatically generated

31.Display the min song length for each album



32.Display max and min eps of podcast for each artist

A screen shot of a computer program

Description automatically generated

33.Calculate avg length of each song for each artist

A computer screen shot of a black screen

Description automatically generated

34.Count number of songs for each language

A screen shot of a computer program

Description automatically generated

35.Calculate total length of songs for each artist

A screenshot of a computer

Description automatically generated

36. Find total songs in each genre

A screen shot of a computer

Description automatically generated

37. Display users with an email containing “gmail.com”

A screen shot of a computer

Description automatically generated

38.Display all songs in English

A screen shot of a computer screen

Description automatically generated

39.List artist whose desc is longer than 30 words

A screenshot of a computer

Description automatically generated

40. Max and Min ages of artist along with their avg ages

A black screen with white text

Description automatically generated