Entities and Attributes for Music Streaming App

1. User: Stores information about registered users.

- **UserID (Primary Key):** It is a unique Identifier for each user.
- **Username:** Username of the user.
- **Email:** Email of the user.
- **Password:** Password of the user.

2. Artist: Contains details about music artists.

- **ArtistID (Primary Key):** Unique Identifier for each artist.
- **Name:** Name of the artist.
- **Genre:** Genre of the artist.

3. Album: Represents music albums.

- **AlbumID (Primary Key):** Unique Identifier for each album.
- **Title:** Title of the album.
- **ArtistID (Foreign Key):** Reference to the artist.
- **Genre:** Genre of the album.
- **Release Date:** Release date of the album.

4. Track: Stores details about individual songs.

- **TrackID (Primary Key):** Unique Identifier for each track. track
- **Title:** Title of the track.
- **ArtistID (Foreign Key):** Reference to the artist.
- **AlbumID (Foreign Key):** Reference to the album.
- **Duration:** Duration of the track.
- **Release Date:** Release date of the track.

5. Playlist: Represents a collection of tracks curated by a user.

- **PlaylistID (Primary Key):** Unique Identifier for each playlist.
- **UserID (Foreign Key):** Reference to the user.
- **Title:** Title of the playlist.
- **Creation Date:** Creation date of the playlist.

6. Like: Tracks the likes or favorites of users on tracks.

- **LikeID (Primary Key):** Unique Identifier for each like.
- **UserID (Foreign Key):** Reference to the user.
- **TrackID (Foreign Key):** Reference to the track.

7. Premium Feature: Stores information about premium features available to users.

- **Premium_Feature_ID (Primary Key):** Unique Identifier for each premium feature.
- **Name:** Name of the premium feature.

8. Subscription Plan: Contains details about subscription plans offered by the platform.

- **Subscription_Plan_ID (Primary Key):** Unique Identifier for each subscription plan.
- **Name:** Name of the subscription plan.
- **Price:** Price of the subscription plan.
- **Description:** Description of the subscription plan.

9. Payment: Stores information about payments made by users.

- **Payment_ID (Primary Key):** Unique Identifier for each payment.
- **User_ID (Foreign Key):** Reference to the user.
- **Amount:** Amount of the payment.
- **Date:** Date of the payment
- **Method:** Method by which the payment is done.

10. Track Playlists: To store which track belongs to which playlist

- **PlaylistID (Primary Key):** Unique identifier of each playlist
- **TrackID (Foreign Key):** Reference to track in a playlist
- **Order:** The order in which tracks are added into a playlist

Relationships between These Entities

1. User-Playlist Relationship

- Each user can create multiple playlists (One-to-Many).
- A user is associated with playlists through the UserID foreign key in the Playlist table.

2. Artist-Album Relationship

- An artist can have multiple albums (*One-to-Many*).
- Each album is associated with an artist through the ArtistID foreign key in the Album table.

3. Album-Track Relationship

- An album can contain multiple tracks (One-to-Many).
- Each track belongs to an album through the AlbumID foreign key in the Track table.

4. Track-Playlist Relationship

- Each track can be part of multiple playlist and a playlist can contain multiple tracks which signifies the Many-to-Many relationship.
- This relationship is realized through a junction table (Track-Playlist) that stores associations between tracks and playlists.

5. User-Premium Feature Relationship

- Each user can have access to multiple premium features, and each premium feature can be associated with multiple users (Many-to-Many).

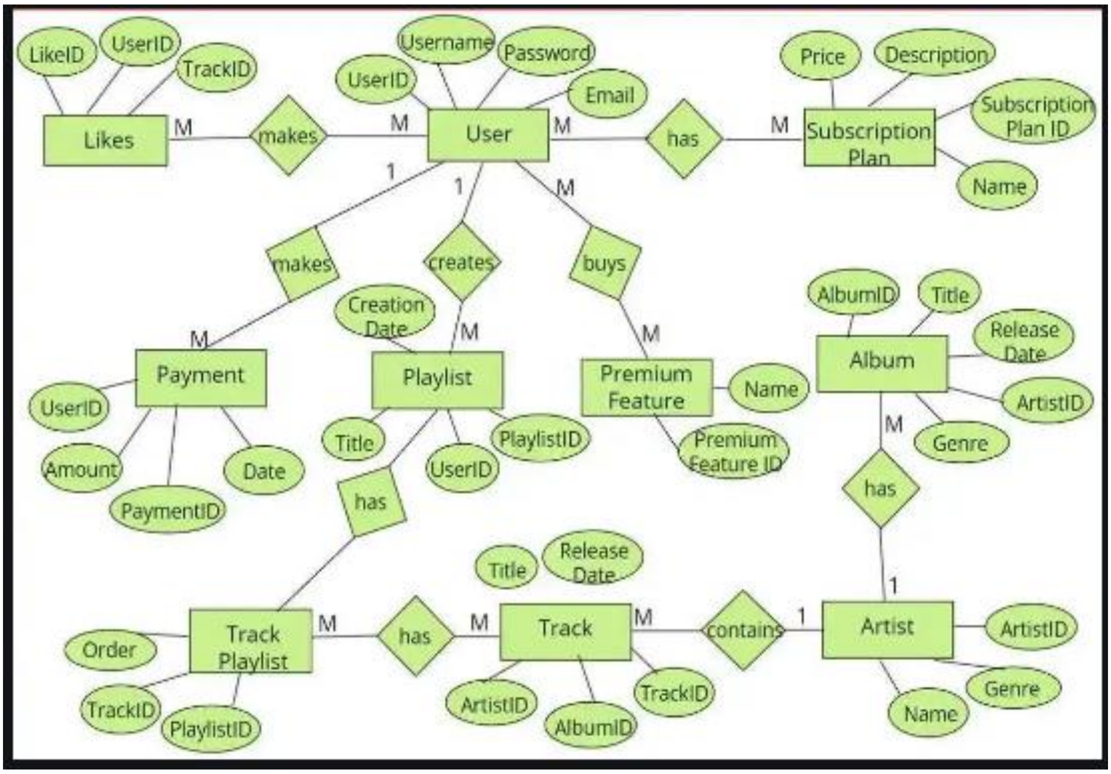
6. User – Subscription Plan Relationship

- Each user can subscribe to multiple subscription plans, and each subscription plan can be subscribed to by multiple users (Many-to-Many).

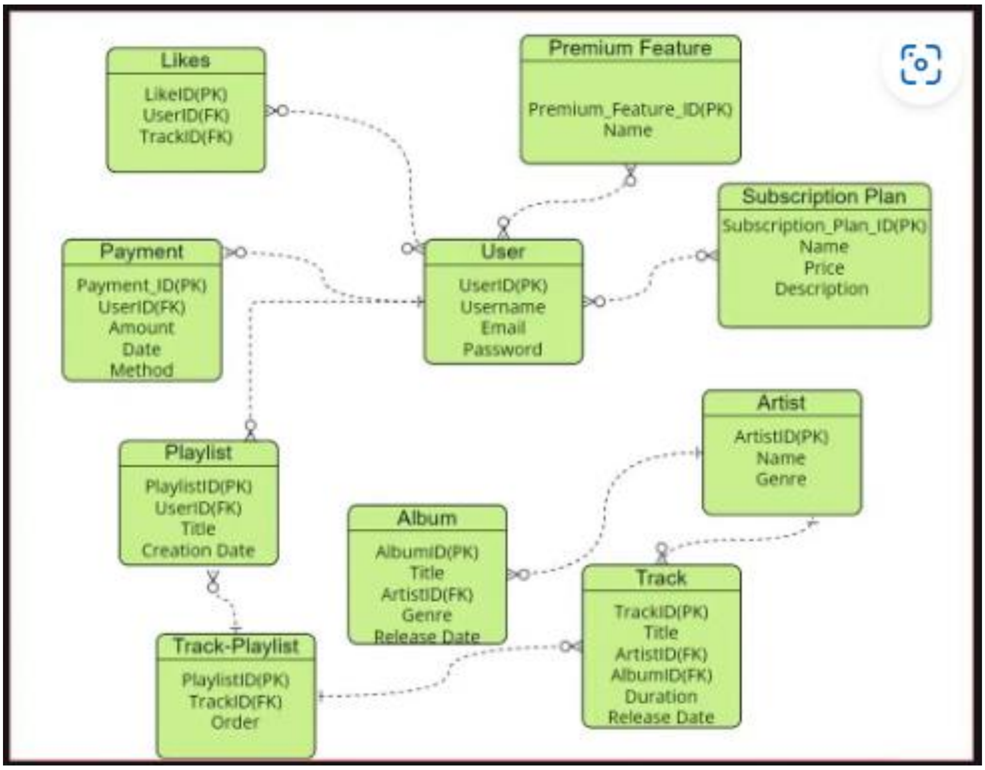
7. User – Payment Relationship

- Each user can make multiple payments, and each payment is made by one user (One-to-Many).
- UserID in the User table relates to UserID in the Payment table.

ER Diagram for Music Streaming App



Database Model for Music Streaming App



Entities Structures in SQL Format

```
CREATE TABLE User (
    UserID INT PRIMARY KEY,
    Username VARCHAR(255),
    Email VARCHAR(255),
    Password VARCHAR(255)
);

CREATE TABLE Artist (
    ArtistID INT PRIMARY KEY,
    Name VARCHAR(255),
    Genre VARCHAR(255)
);

CREATE TABLE Album (
    AlbumID INT PRIMARY KEY,
    Title VARCHAR(255),
    ArtistID INT,
    Genre VARCHAR(255),
    ReleaseDate DATE,
    FOREIGN KEY (ArtistID) REFERENCES Artist(ArtistID)
);

CREATE TABLE Track (
    TrackID INT PRIMARY KEY,
    Title VARCHAR(255),
    ArtistID INT,
    AlbumID INT,
    Duration TIME,
    ReleaseDate DATE,
    FOREIGN KEY (ArtistID) REFERENCES Artist(ArtistID),
    FOREIGN KEY (AlbumID) REFERENCES Album(AlbumID)
);

CREATE TABLE Playlist (
    PlaylistID INT PRIMARY KEY,
    UserID INT,
    Title VARCHAR(255),
    CreationDate DATE,
    FOREIGN KEY (UserID) REFERENCES User(UserID)
);

CREATE TABLE Like (
    LikeID INT PRIMARY KEY,
    UserID INT,
    TrackID INT,
    FOREIGN KEY (UserID) REFERENCES User(UserID),
  FOREIGN KEY (TrackID) REFERENCES Track(TrackID)
);

CREATE TABLE PremiumFeature (
  Premium_Feature_ID INT PRIMARY KEY,
  Name VARCHAR(255)
);

CREATE TABLE SubscriptionPlan (
  Subscription_Plan_ID INT PRIMARY KEY,
  Name VARCHAR(255),
  Price DECIMAL(10,2),
  Description TEXT
);

CREATE TABLE Payment (
  Payment_ID INT PRIMARY KEY,
  User_ID INT,
  Amount DECIMAL(10,2),
  Date DATE,
  Method VARCHAR(50),
  FOREIGN KEY (User_ID) REFERENCES User(UserID)
);
```