



User(userID: INT [PK], userName: VARCHAR(20), userEmail: VARCHAR(20), artistID: INT [FK to Artist.artistID], userProfile: BLOB, searchFor : VARCHAR(40))

Artist(artistID: INT [PK], artistFirstName: VARCHAR(15), artistLastName: VARCHAR(15), artistProfile: BLOB, artistIntroduction: VARCHAR(500))

Song(songID: INT [PK], artistID: INT [FK to Artist.artistID], songName: VARCHAR(40), likeCounter: INT, albumID: INT [FK to Album.albumID])

Review(firstReviewID: INT [PK], songID: INT [FK to Song.songID], userID: INT [FK to User.userID], time: DATETIME, reviewContent: VARCHAR(500), likeCounter: INT)

SubReview(time: DATETIME [PK], firstReviewID: INT [FK to Review.firstReviewID], userID: INT [FK to User.userID], reviewContent: VARCHAR(500), likeCounter: INT)

Album(albumID: INT [PK], albumName: VARCHAR(50), artistID: INT [FK to Artist.artistID], likeCounter: INT, albumProfile: BLOB)

Assumptions:

We think:

- There must be only one artist for each song/album.
- An artist must have at least one song, or album.
- An album will have at least one song.
- One user will search for one artist/album/song at a time.
- There will be many songs for one album.
- There will be one song for many reviews.
- A song can have 0 reviews and a user can write 0 reviews.
- A review can have 0 subreviews.
- Only songs will have reviews.
- One user can apply to be one artist.

Relationships and Cardinality:

Search (one to one) - A user will search for one artist or album or song, our application will show the one that shows up from our database.

Comment (one to many) - A user can comment many reviews to different songs and there can also be many reviews commented on one song. These reviews will show up based on the song selected. Only songs will have reviews, not artists or albums. There will also be many sub reviews commented on one review.

Create (one to many) - An artist can create many albums or songs. These albums and songs will be displayed until the artists name and stored in the database under the artists ID

Belongs to (one to many) - One album will contain many songs. These songs will be displayed when an album is selected.

Apply (one to one) - One user can apply to be an artist in the database. As an artist, they can create songs or albums.