EAS 550 Project: Music Inventory System
3NF Justification Report
Team: Halle Bryant, Arun Ghontale, Phalguni Raparla
Free Music Archive Dataset: https://github.com/mdeff/fma

Third Normal Form (3NF) requires all database table rows to meet the criteria of the First and Second Normal Forms while also eliminating transitive partial dependencies. First Normal Form (1NF) requires all entity rows to be unique and all columns to be atomic, meaning there are no columns with multiple values. Second Normal Form (2NF) eliminates all partial dependencies, meaning every non-key attribute is dependent on the primary key for each entity. Third Normal Form (3NF) eliminates all transitive partial dependencies, meaning there are no non-key attributes that are dependent on another non-key attribute.

Below is the list of entities and entity relations for the Music Inventory System. (Note that primary keys for each entity are underlined.) Parent entities Albums, Artists, Composers, Engineers, Genres, Labels, Licenses, Lyricists, Producers, and Tracks are constructed so each row is uniquely identified by its ID (e.g. "album_id", "artist_id", "composer_id", etc). Each ID determines every attribute included in its respective table. For example, two albums with the same name will have unique IDs, which correctly identify their respective artists, types (album vs. live performance vs. radio program), number of tracks, release dates, etc. This same logic applies to the Artists and Tracks entities, which make up the largest and most complex in the schema attributes-wise. Parent entities Composers, Engineers, Genres, Labels, Licenses, Lyricists, and Producers each consist of 2-3 attributes in the form (ID → Attribute 1, Attribute 2). For example, "genres_id" in the Genres table determines "parent_genre_id" (for entries considered to be subgenres) and "genre_name." In the Licenses entity, "license_id" determines "license_title" and "license_url", which links to a webpage describing the legal terms of a specific license type.

Weak entity sets include Audio and Social, whose entries are uniquely identified by the foreign key "track_id" from the Tracks entity. The Audio table contains information about the musical features of a track, with each non-key attribute clearly depending on "track_id". The Social table has information about the popularity of a track and associated artist, which is determined by "track_id" as well. Based on further analysis to determine whether attributes such as "artist_familiarity" and "artist_discovery" are consistent across all of an artist's tracks, it may be necessary to create distinct TrackSocial and ArtistSocial entities detailing the social features of tracks and artists separately. For now, we house all of this information in one table.

Six bridge tables have been included to account for the six many-to-many relations in the schema. These relations include Tracks to Lyricists, Tracks to Composers, Tracks to Genres, Albums to Producers, Albums to Engineers, and Labels to Artists. For each of the bridge entities, every entry consists only of the primary key, which consists of the two primary keys (or IDs) from the linked tables. Since there are no non-key attributes in these tables, there are no partial or transitive partial dependencies.

Entities:

- **Albums:** <u>album_id</u>, album_title, artist_id, album_type, album_favorites, album_tracks, album_date_released, album_listens
- **Artists:** <u>artist_id</u>, artist_name, artist_active_year_begin, artists_favorites, artist_handle, artist_website
- **Composers**: <u>composer_id</u>, composer_name
- **Engineers:** <u>engineer_id</u>, engineer_name
- **Genres:** <u>genre_id</u>, parent_genre_id, genre_name
- **Labels:** <u>label_id</u>, label_name
- **Licenses:** <u>license_id</u>, license_title, license_url
- **Lyricists:** <u>lyricist_id</u>, lyricist_name
- **Producers:** <u>producer_id</u>, producer_name
- **Tracks**: <u>track_id</u>, album_id, track_title, track_language_code, track_listens, track_url, license_id, track_date_recorded, track_bit_rate, track_duration, track_explicit, track_favorites
  - **Audio:** <u>track_id</u>, acousticness, danceability, energy, instrumentalness, liveness, speechiness, tempo, valence
  - **Social:** <u>track_id</u>, artist_discovery, artist_familiarity, artist_hotttnesss, song_currency, song_hotttnesss
- **AlbumProducers**: <u>album_id, producer_id</u>
- **AlbumEngineers:** <u>album_id, engineer_id</u>
- **TrackComposers:** <u>composer_id, track_id</u>
- **ArtistLabels**: <u>artist_id, label_id</u>
- **TrackGenres:** <u>track_id, genre_id</u>
- **TrackLyricists:** <u>track_id, lyricist_id</u>

Entity Relations

- **Albums** to **Artists** (Many to one)
- **Tracks** to **Albums** (Many to one)
- **Tracks** to **Licenses** (Many to one)
- **Tracks** to **Audio** (One to One)
- **Tracks** to **Social** (One to One)
- **Tracks** to **Lyricists** (Many to many linked by **TrackLyricists** table)
- **Tracks** to **Composers** (Many to Many, **TrackComposers**)
- **Tracks** to **Genres** (Many to many, total participation, **TrackGenres**)
- **Albums** to **Producers** (Many to Many, **AlbumProducers**)
- **Albums** to **Engineers** (Many to many linked by **AlbumEngineers** associative entity)
- **Labels** to **Artists** (Many to many linked by **ArtistLabels** table)