ERD paragraph:

In my Music Database, Jazz Entity, I have five primary entity sets: Artist, Album, Song, Instrument, and Record\_Label.

An **Artist** may OWN multiple **Albums**, and each **Album** must be owned by at most one **Artist**. Besides, an **Album** must CONTAIN at least one **Song**, and each **Song** may BELONG to multiple **Albums**. Moreover, an **Artist** may SING multiple **Songs**, and a **Song** must be SUNG by at most one **Artist**. Lastly, an **Artist** may be SIGNED by multiple **Record**\_**Labels**, and a **Record**\_**Label** may sign many Artists. (86 words)

Normalisation of the table:

1. Album: 2nd Normal Form
2. Artist: 2nd Normal Form
3. Song: 2nd Normal Form
4. Record\_Label: 3rd Normal Form

DDL Statement:

DROP TABLE IF EXISTS ALBUM\_SONGS;

DROP TABLE IF EXISTS ALBUM;

DROP TABLE IF EXISTS SONG;

DROP TABLE IF EXISTS CONTRACT;

DROP TABLE IF EXISTS ARTIST;

DROP TABLE IF EXISTS RECORD\_LABEL;

CREATE TABLE IF NOT EXISTS ARTIST (

ARTIST\_ID VARCHAR(50) NOT NULL,

ARTIST\_NAME VARCHAR(50),

ARTIST\_DOB DATE,

ARTIST\_DEBUT\_DATE DATE,

PRIMARY KEY (ARTIST\_ID)

);

CREATE TABLE IF NOT EXISTS SONG (

SONG\_ID INT NOT NULL AUTO\_INCREMENT,

ARTIST\_ID VARCHAR(50) NOT NULL,

SONG\_NAME VARCHAR(50),

SONG\_LENGTH TIME,

PRIMARY KEY (SONG\_ID, ARTIST\_ID),

FOREIGN KEY (ARTIST\_ID) REFERENCES ARTIST(ARTIST\_ID) ON DELETE RESTRICT

);

CREATE TABLE IF NOT EXISTS ALBUM (

ALBUM\_ID INT NOT NULL,

ARTIST\_ID VARCHAR(50) NOT NULL,

ALBUM\_NAME VARCHAR(50),

PUBLISH\_DATE DATE,

PRIMARY KEY (ALBUM\_ID, ARTIST\_ID),

FOREIGN KEY (ARTIST\_ID) REFERENCES ARTIST(ARTIST\_ID) ON DELETE RESTRICT

);

CREATE TABLE IF NOT EXISTS ALBUM\_SONGS (

ALBUM\_ID INT NOT NULL,

SONG\_ID INT NOT NULL AUTO\_INCREMENT,

PRIMARY KEY (ALBUM\_ID, SONG\_ID),

FOREIGN KEY (ALBUM\_ID) REFERENCES ALBUM(ALBUM\_ID) ON DELETE RESTRICT,

FOREIGN KEY (SONG\_ID) REFERENCES SONG(SONG\_ID) ON DELETE RESTRICT,

UNIQUE (ALBUM\_ID, SONG\_ID)

);

CREATE TABLE IF NOT EXISTS RECORD\_LABEL (

RECORD\_LABEL\_ID VARCHAR(10) NOT NULL,

RECORD\_LABEL\_NAME VARCHAR(50),

PRIMARY KEY (RECORD\_LABEL\_ID)

);

CREATE TABLE IF NOT EXISTS CONTRACT (

RECORD\_LABEL\_ID VARCHAR(10),

ARTIST\_ID VARCHAR(50),

PRIMARY KEY (RECORD\_LABEL\_ID, ARTIST\_ID),

FOREIGN KEY (ARTIST\_ID) REFERENCES ARTIST(ARTIST\_ID) ON DELETE RESTRICT,

FOREIGN KEY (RECORD\_LABEL\_ID) REFERENCES RECORD\_LABEL(RECORD\_LABEL\_ID) ON DELETE RESTRICT

);

Relational Algebra:

1. Find Artist(s) who have wrote songs with an average length greater than 3 minutes.

A close up of a sign

Description automatically generated

1. Find Artist(s) who own more than 6 albums.

A close up of some words

Description automatically generated

1. Find the number of Artist(s) that each record label has signed.

A close-up of some writing

Description automatically generated

1. Delete Artist(s) with less than 6 Albums.

A close-up of a whiteboard

Description automatically generated

1. Delete Album(s) with average Song Duration less than 3 minutes.

A close-up of some writing

Description automatically generated

DML statement:

1. Find Artist(s) who have wrote songs with an average length greater than 3 minutes.

SELECT ARTIST\_ID, ARTIST\_NAME FROM ARTIST

WHERE ARTIST\_ID IN (

SELECT ARTIST\_ID FROM SONG

GROUP BY ARTIST\_ID

HAVING AVG(TIME\_TO\_SEC(SONG\_LENGTH)) > 180

);

1. Find Artist(s) who own more than 6 albums.

SELECT ALBUM.ARTIST\_ID, ARTIST.ARTIST\_NAME, COUNT(\*) AS ALBUM\_NUMBER

FROM ALBUM, ARTIST

WHERE ALBUM.ARTIST\_ID = ARTIST.ARTIST\_ID

GROUP BY ALBUM.ARTIST\_ID

HAVING COUNT(\*) > 6;

1. Find the number of Artist(s) that each record label has signed.

SELECT C.RECORD\_LABEL\_ID, R.RECORD\_LABEL\_NAME, COUNT(C.ARTIST\_ID) AS TOTAL\_NUMBER\_OF\_ARTISTS

FROM CONTRACT C, RECORD\_LABEL R

WHERE C.RECORD\_LABEL\_ID = R.RECORD\_LABEL\_ID

GROUP BY C.RECORD\_LABEL\_ID, R.RECORD\_LABEL\_NAME;

1. Delete Artist(s) with less than 6 Albums.

DELETE FROM ARTIST A WHERE A.ARTIST\_ID IN (

SELECT ALBUM.ARTIST\_ID

FROM ALBUM

GROUP BY ALBUM.ARTIST\_ID

HAVING COUNT(\*) < 6

);

1. Delete Album(s) with average Song Duration less than 3 minutes.

DELETE FROM ALBUM WHERE ALBUM.ALBUM\_ID IN (

SELECT ALBUM\_SONGS.ALBUM\_ID FROM SONG S, ALBUM\_SONGS

WHERE ALBUM\_SONGS.SONG\_ID = S.SONG\_ID

GROUP BY ALBUM\_SONGS.ALBUM\_ID

HAVING AVG(TIME\_TO\_SEC(S.SONG\_LENGTH)) < 180

);