PROJECT REPORT - 2

WolfMedia Streaming Service

CSC 540 Database Management concepts and Systems (001)

Created By:-

Team M

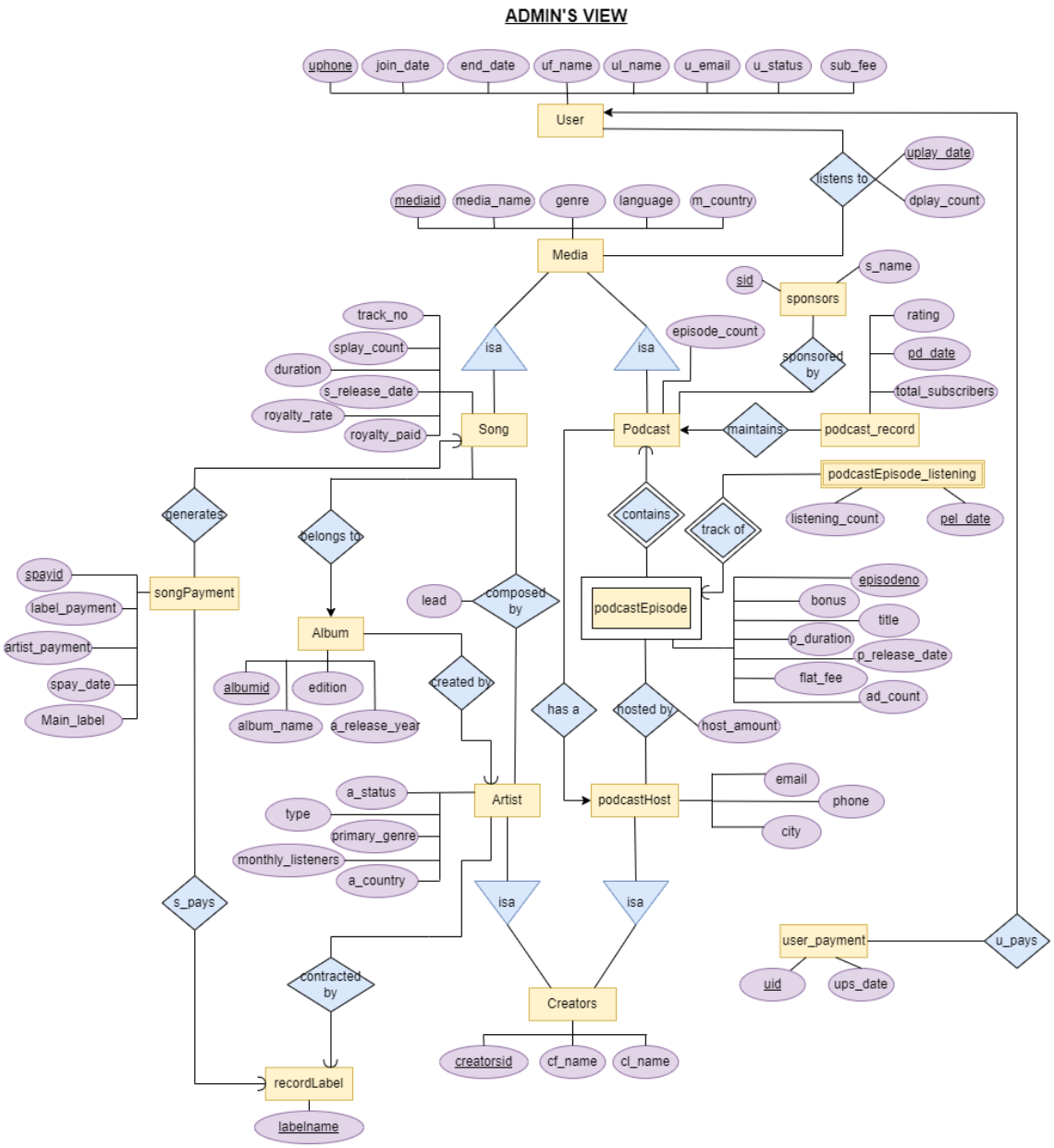
Vishwa Hiren Gandhi (vgandhi)
Dhruv Mukesh Patel (dpatel49)
Harshil Tushar Sanghavi (hsangha)
Mitul Patel (mpatel27)

ASSUMPTIONS:-

- Record Label name is unique
- Phone is unique for every user
- Artists can be under contract with only one record label.
- Every album belongs to only one artist but artist can collaborate with other artists in the songs for that album.
- Podcast hosts may or may not give their associated phone number and email.
- Play counts are updated every month and are checked on a monthly basis only.
- Every song belongs to at most one album.
- Song payment made to artist/ label once a month
- For every advertisement in a podcast episode, 2$ bonus is added to the associated episode and podcast.
- There can be multiple paymentid associated with one user but every paymentid will only be associated with exactly one user.
- Podcasts are associated with at most one host. But different podcast hosts can collaborate for various episodes in the podcasts.
- Podcast hosts are paid as soon as the podcast episode is released.
- Collaborators are the various different artists collaborating in the song.
- There can be more than one sponsors for the podcast.
- Media is considered to have one main genre.

UPDATED ADMIN E/R DIAGRAM

The updated E/R diagram for the Admin's view is shown below. All the relational schemas and tasks and operations have been performed with respect to this E/R diagram.



1. Global Relational Database Schema:

Media(<u>mediaid</u>, media_name, genre, language, m_country)

Mediaid → mediaid, media_name, genre, language, m_country holds because each mediaid is unique, and it identifies a media that has a name, genre, language, and country. If we were to try and take any combination of other attributes, it would severely limit the possibilities our database could have. For example, several media can have the same name and genre, or two media could be released in the same country and can have the same language. Because the left hand side is a superkey, this functional dependency is in BCNF (and thereby 3NF).

Song(<u>mediaid</u>, albumid, track_no, splay_count, duration, s_release_date, royalty_rate, royalty_paid)

mediaid → mediaid , duration , release_date, royalty_rate , royalty_paid holds true because more than one song can have the same duration, release date and play count hence can be differentiated by the id. There can even be the possibility of royalty rate and royalty paid being equal hence it cannot be used as a key. Hence mediaid is a superkey as it contains all the attributes so it is in BCNF and hence in 3NF.

Podcast(<u>mediaid</u>, episode_count,hostid)

mediaid → mediaid, episode_count, hostid holds because each podcast with all the attributes are unique for a mediaid. No other functional dependencies can be formed since other attributes can appear multiple times for different podcasts. Because the left hand side is a superkey, this functional dependency is in BCNF and thus 3NF.

podcast_record(<u>podcastid</u>, <u>pd_date</u>, rating, total_subscribers)

podcastid, pd_date → podcastid, pd_date, rating, total_subscribers holds because each podcast with all the attributes are unique for a podcastid and pd_date. Podcastid and pd_date together are a key and are able to define other attributes and pd_date cannot single-handedly be a key since it can be repeated for different podcasts . Also, any other attribute cannot be used to form a key such as rating being as it can be repeated multiple times. There is no other functional dependency able to do the same. Thus, the relation is in BCNF and hence in 3NF.

Sponsors(<u>sid</u>, s_name)

sid → sid, s_name holds since this relation only has two attributes, it is in BCNF and therefore 3NF. Name cannot be used as a key because multiple sponsors can have the same names.

sponsoredby(sid,podcastid)

sid , podcastid → sid, podcastid holds because the relationship sponsoredby is connected to two entities. This connection necessitates the keys for all entities to which it is linked, which are the two we have provided. We cannot adequately explain the link if any of the two are not mentioned in the key; for example, if the sid is not listed, we cannot determine who the podcastid is associated with. It is in BCNF since the left hand side has all characteristics (and thereby 3NF).

Album(albumid, album_name, edition, artistid, a_release_year)

albumid → albumid, album_name, edition, artistid, a_release_year holds true as albumid on the left hand side gives all the attributes of the respective table, hence, it is the super key. BCNF holds true as the LHS is the super key, so it is also in 3NF. Also, we cannot select any other attributes other than albumid for the key because no other attribute uniquely identifies the Album.

composedby(songid,artistid,lead)

songid , artistid → songid , artistid, lead holds because each lead singer with all the attributes are unique for a songid and artistid. songid and artistid together are a key and are able to define other attributes and artistid cannot single-handedly be a key since it can be repeated for different songid. Also, any other attribute cannot be used to form a key. There is no other functional dependency able to do the same. Thus, the relation is in BCNF and hence in 3NF.

podcastEpisode(podcastid,episodeno,title,p_duration,p_release_date,flat_fee,bonus,ad_count)

podcastid , episodeno → podcastid , episodeno, title, p_duration, p_release_date, flat_fee, bonus,ad_count holds because each podcast episodes with all the attributes are unique for a podcastid and episodeno. podcastid and episodeno together are a key and are able to define other attributes and episodeno cannot single-handedly be a key since it can be repeated for different podcasts. Also, any other attribute cannot be used to form a key such as title as it can be repeated in multiple podcast episodes. There is no other functional dependency able to do the same. Thus, the relation is in BCNF and hence in 3NF.

podcastEpisode_listening(podcastid, episodeno, pel_date, listening_count, ad_count)

podcastid , episodeno, pel_date → podcastid, episodeno, pel_date, listening_count, ad_count holds because the podcastEpisode_listening is a weak entity and inherited its key from another weak entity podcastEpisode. This connection necessitates all the three keys we have provided. We cannot adequately explain the connection if any of the keys are missing. It is in BCNF since the left hand side has all characteristics (and thereby 3NF).

user(<u>uphone</u>, join_date, end_date, uf_name, ul_name, uemail, u_status, sub_fee)

uphone → uphone, join_date, end_date, uf_name, ul_name, email, u_status, sub_fee holds because each user with all the attributes are unique for a uphone. No other functional dependencies can be formed since other attributes can appear multiple times for different users. Because the left hand side is a superkey, this functional dependency is in BCNF and thus 3NF.

songPayment(<u>spayid</u>,songid,artistid,labelname,label_payment,artist_payment,spay_date,main_label)

spayid → spayid, songid, artistid, labelname, label_payment, artist_payment, spay_date, main_label holds because each songPayment with all the attributes are unique for a spayid. No other functional dependencies can be formed since other attributes can appear multiple times for different payments. Because the left hand side is a superkey, this functional dependency is in BCNF and thus 3NF.

userpayment(<u>uid</u>,uphone,ups_date)

uid → uid, uphone, ups_date holds as uid is unique, and identifies a userpayment that has payment date and phone number. User can subscribe for two or more months so will have two payments associated with the same phone number and hence uid is the unique identifier. Because the left hand side is a superkey, this functional dependency is in BCNF and thus 3NF.

recordLabel(<u>labelname</u>)

labelname → labelname holds since this relation has only one attribute, it is in BCNF and therefore in 3NF.

podcastHost(<u>creatorsid</u>, email, phone, city)

creatorsid → creatorsid , email, phone, city holds because each podcastHost with all the attributes are unique for a creatorsid. No other functional dependencies can be formed since other attributes can appear multiple times for different podcast hosts. Because the left hand side is a superkey, this functional dependency is in BCNF and thus 3NF.

listensto(<u>uphone</u> , <u>mediaid</u>, <u>uplay_date</u>, dplay_count)

uphone , mediaid , uplay_date → uphone, mediaid, uplay_date, dplay_count holds because the relationship listensto is connected to two entities.. This connection necessitates the keys for all entities to which it is linked, which are the three we have provided. We cannot adequately explain the link if any of the three are not mentioned in the key; for example, if the uphone is not listed we cannot determine which media is associated with which user. It is in BCNF since the left hand side has all characteristics (and thereby 3NF).

hostedby(<u>podcastid</u>,<u>episodeno</u>,<u>hostid</u>,host_amount)

podcastid , episodeno , hostid → podcastid , episodeno, hostid, host_amount holds because the relationship hostedby is connected to two entities in which one of them is a weak entity. This connection necessitates the keys for all entities to which it is linked, which are the three we have provided. We cannot adequately explain the link if any of the three are not mentioned in the key. It is in BCNF since the left hand side has all characteristics (and thereby 3NF)

Creators(<u>creatorsid</u>, cf_name, cl_name)

Creatorsid → creatorsid , cf_name, cl_name holds because each creators with all the attributes are unique for a creatorsid. No other functional dependencies can be formed since other attributes can appear multiple times for different creators. Because the left hand side is a superkey, this functional dependency is in BCNF and thus 3NF.

artist(<u>creatorsid</u>, labelname, a_status, type, primary_genre, monthly_listeners, a_country)

creatorsid → creatorsid, labelname, a_status, type, primary_genre, monthly_listeners, a_country holds because each artist with all the attributes are unique for a creatorsid. No other functional dependencies can be formed since other attributes can appear multiple times for different artists. Because the left hand side is a superkey, this functional dependency is in BCNF and thus 3NF.

2. Design for Global Schema:

Design decision for Global Schema:-

- The entity sets USER, MEDIA, SPONSORS, SONG, ALBUM, PODCAST, PODCASTHOST, ARTIST, CREATORS and RECORDLABEL are transformed to relations with the same attributes as shown in the E/R diagram.
- USERPAYMENT has uphone as a foreign key from the user relation since we have assumed that the payementid will be unique for payment. There can be multiple paymentid associated with one user but every paymentid will only be associated with one user.
- Podcast episode and podcastepisode_listening are weak entities and hence they will inherit the key from the strong entity in order to describe the rows uniquely.
- SONGPAYMENT has paymentid which is unique and hence it is associated with only one song and label. One song can have multiple paymentid but every paymentid is associated with only one entity.
- All the relationships are converted into relations with the underlined attributes from the connecting entities.
- Only the relationships having many to one relations are not converted into the relational schema such as maintains, has a, belongsto, generates, s_pays, contractedby, u_pays, createdby. The key of the one side entity is added to the many side of the remaining entity.
- The relationship composedby, hostedby and listensto have their own attributes so while converting it into the relational schema along with the underlined attributes from the connecting entities their own attributes are maintained.
- Weak relationships contains and track of are not converted into the relational schema.
- Album has artistid as the foreign key because we have assumed that album has its main artist. The main artist can collaborate with different artists in the different songs for the album.
- Artist has the labelname as the foreign key as the artist can only be in contract with the one record label.
- Podcast_record and podcastepisode_listening have been made a separate entity in order to maintain the monthly records efficiently.

INTEGRITY CONSTRAINTS:-

media(<u>mediaid</u>, media_name, genre, language, m_country)

- Here mediaid is a unique identifier and is the PRIMARY KEY as it uniquely identifies the record and is NOT NULL.
- media_name is not allowed to be null. Also multiple media can have the same name as well. Some artists choose to give their songs numbers or use a sequence of symbols to represent the song instead of a traditional name.
- Genre can not be null as It is unlikely for a song or podcast to have no genre at all, as genre is used to categorize and describe different styles of music. Even songs or podcasts that don't fit neatly into a particular genre can still be categorized in some way. For our report we have assumed that songs or podcasts which are difficult to categorize are classified as “multigenre”.
- The inclusion of language and country columns can help provide useful information about the media item, such as the language in which it is spoken or sung and the country of origin or cultural context. However, it is possible that some media items may not have a language or country associated with them, particularly if they have a more global or universal appeal. In such cases, it may be appropriate to keep the language and country columns as null constraints, indicating that there is no value associated with these fields for that particular media item.

Song(<u>mediaid</u>, albumid, track_no, splay_count, duration, s_release_date, royalty_rate, royalty_paid)

- mediaid is the foreign key and the primary key as it is inherited from isa and cannot be null.
- albumid is the foreign key and cannot be null.
- duration and s_release_date are set to NOT NULL because they provide a critical piece of information that should always be included in the database to accurately search and categorize the song or podcast.
- royalty_rate are set to NOT NULL because they are critical for tracking the financial aspects of a song and are associated with the payments.
- royalty_paid is set be boolean value 0 or 1 which indicates whether the payment is received or not. 0 means royalty is not paid and 1 means royalty is being paid.
- track_no can be NULL since there can exist an independent song which is not in any album.
- s_play_count cannot be NOT NULL as it is the monthly play count for that song. If a song is released then its default value is 0.
- Referential Integrity: SONG is-a MEDIA with mediaid as Foreign Key, any update and deletion on a particular mediaid in Media table will have a cascading action on SONG table.
- Referential Integrity: SONG is dependent on Album with albumid as Foreign Key, any update and deletion on a particular albumid in Album table will have a cascading action on SONG table.

Podcast(<u>mediaid</u>, episode_count,hostid)

- mediaid is the Primary Key which makes it UNIQUE and NOT NULL.
- hostid is the foreign Key which makes it UNIQUE and NOT NULL.
- episode_count is NOT NULL since it defines the number of episodes podcasts have.
- Referential Integrity: Podcast is-a MEDIA with mediaid as Foreign Key, any update and deletion on a particular mediaid in Media table will have a cascading action on SONG table.
- Referential Integrity: Podcast is dependent on PodcastHost with hostid as Foreign Key, any update and deletion on a particular hostid in podcastHost table will have a cascading action on podcast table.

podcast_record(<u>podcastid, pd_date</u>, rating, total_subscribers)

- podcastid is the Foreign Key and part of the Composite Primary Key which makes it NOT NULL.
- pd_date is the second primary key and is NOT NULL.
- Rating and total_subscribers is NOT NULL since it defines monthly subscribers for the given podcasts and is used to generate the reports.
- Referential Integrity: podcast_record is dependent on Podcast with podcastid as Foreign Key, any update and deletion on a particular podcastid in Podcast table will have a cascading action on podcast_record table.

Sponsors(<u>sid</u>, s_name)

- sid is the PRIMARY KEY and is NOT NULL.
- s_name is NOT NULL as we want to maintain the data for the sponsors in our project.

sponsoredby(<u>sid,podcastid</u>)

- sid is one of the PRIMARY KEYS and is NOT NULL.
- podcastid is the second of two PRIMARY KEYS and is NOT NULL.
- Referential Integrity: sponsoredby is dependent on Podcast with podcastid as Foreign Key, any update and deletion on a particular podcastid in Podcast table will have a cascading action on sponsoredby table.
- Referential Integrity: sponsoredby is dependent on Sponsors with sid as Foreign Key, any update and deletion on a particular sid in Sponsors table will have a cascading action on sponsoredby table.

Album(<u>albumid</u>, album_name, edition, artistid, a_release_year)

- Albumid is the PRIMARY KEY and is NOT NULL.
- Album_name, and a_release_year are NOT NULL in order to better classify the record and maintain the integrity of the database.

- Edition has CHECK CONSTRAINTS set for this table as edition can be of three types- special, limited and collector's edition.
- artistid is the FOREIGN KEY and is NOT NULL.
- Referential Integrity: Album is dependent on Artist with artistid as Foreign Key, any update and deletion on a particular artistid in Artist table will have a cascading action on Album table.

composedby(songid,artistid,lead)

- songid is the FOREIGN KEY which is the part of the composite primary key for this table which makes it NOT NULL.
- artistid is the FOREIGN KEY which is the part of the composite primary key for this table which makes it NOT NULL
- Lead is the BOOLEAN attribute which is set to Yes or No. No means the artist is not the lead singer for that song. Yes means the artist is the lead singer. It is used to identify the recordlabel for that song since the song is owned by the recordlabel of its main artist.
- Referential Integrity: composedby is dependent on Artist with artistid as Foreign Key, any update and deletion on a particular artistid in Artist table will have a cascading action on composedby table.
- Referential Integrity: composedby is dependent on Song with songid as Foreign Key, any update and deletion on a particular songid in Song table will have a cascading action on composedby table.

podcastEpisode(podcastid,episodeno,title,p_duration,p_release_date,flat_fee,bonus,ad_count)

- podcastid is the FOREIGN KEY which is the part of the composite primary key for this table which makes it NOT NULL.
- episodeno is the FOREIGN KEY which is the part of the composite primary key for this table which makes it NOT NULL.
- title is allowed to be null. Its default value is Untitled.
- p_duration , p_release_date , flat_fee, bonus, ad_count are NOT NULL since they are important to define the podcastEpisode and to calculate payment for hosts.
- Referential Integrity: podcastEpisode is dependent on Podcast with podcastid as Foreign Key, any update and deletion on a particular podcastid in Podcast table will have a cascading action on podcastEpisode table.

podcastEpisode_listening(podcastid, episodeno, pel_date, listening_count, ad_count)

- podcastid is the FOREIGN KEY which is the part of the composite primary key for this table which makes it NOT NULL.
- episodeno is the FOREIGN KEY which is the part of the composite primary key for this table which makes it NOT NULL.
- pel_date is the third of the PRIMARY KEY for this table.

- listening_count is not allowed to be null and its default value is 0.
- ad_count is not allowed to be null and its default value is 0.
- Referential Integrity: podcastEpisode_listening is dependent on podcastEpisode with podcastid and episodeno as Foreign Key, any update and deletion on a particular podcastid or episodeno in podcastEpisode table will have a cascading action on podcastEpisode_listening table.

songPayment(<u>spayid</u>,songid,artistid,labelname,label_payment,artist_payment,spay_date,main_label)

- spayid is the PRIMARY KEY which makes it NOT NULL.
- songid is the FOREIGN KEY which makes it NOT NULL.
- artistid is the FOREIGN KEY which makes it NOT NULL.
- labelname is the FOREIGN KEY which makes it NOT NULL.
- label_payment , artist_payment , spay_date and main_label is NOT NULL in order to maintain the payment information.
- Referential Integrity: songPayment is dependent on Song with songid as Foreign Key, any update and deletion on a particular songid in Song table will have a cascading action on songPayment table.
- Referential Integrity: songPayment is dependent on Artist with artistid as Foreign Key, any update and deletion on a particular artistid in Artist table will have a cascading action on songPayment table.
- Referential Integrity: songPayment is dependent on Recordlabel with labelname as Foreign Key, any update and deletion on a particular labelname in Recordlabel table will have a cascading action on songPayment table.

user(<u>uphone</u>, join_date, end_date, uf_name, ul_name, uemail, u_status, sub_fee)

- spayid is the PRIMARY KEY which makes it NOT NULL.
- join_date , uf_name , ul_name are NOT NULL as the data is required for maintaining the database.
- end_date , uemail , u_status , sub_fee are allowed to be NULL as it is upto the user to subscribe to the podcast or not as well as give their email or not.

userpayment(<u>uid</u>,uphone,ups_date)

- uid is the PRIMARY KEY which makes it NOT NULL.
- uphone is the FOREIGN KEY which makes it NOT NULL.
- ups_date is NOT NULL as the user has some payment date. For users who do not have subscribed their entry will not be shown in the userpayment.
- Referential Integrity: userpayment is dependent on User with uphone as Foreign Key, any update and deletion on a particular uphone in User table will have a cascading action on userpayment table.

recordLabel(labelname)

- labelname is the PRIMARY KEY which makes it NOT NULL.

podcastHost(creatorsid, email, phone, city)

- creatorsid is the PRIMARY KEY which makes it NOT NULL.
- email , phone and city can be NULL as it is up to the podcast host to give their information or not.
- Referential Integrity: podcastHost is-a creator with creatorsid as Foreign Key, any update and deletion on a particular creatorsid in Creators table will have a cascading action on podcastHost table.

listensto(uphone , mediaid, uplay_date, dplay_count)

- uphone is the FOREIGN KEY which is the part of the composite primary key for this table which makes it NOT NULL.
- mediaid is the FOREIGN KEY which is the part of the composite primary key for this table which makes it NOT NULL.
- uplay_date is the third PRIMARY KEY which makes it NOT NULL.
- dplay_count is NOT NULL because it is required to maintain the count for the user when the user listens to any type of the media.
- Referential Integrity: listensto is dependent on User with uphone as Foreign Key, any update and deletion on a particular uphone in User table will have a cascading action on listensto table.
- Referential Integrity: listensto is dependent on Media with mediaid as Foreign Key, any update and deletion on a particular mediaid in Media table will have a cascading action on listensto table.

hostedby(podcastid,episodeno,hostid,host_amount)

- podcastid is the FOREIGN KEY which is the part of the composite primary key for this table which makes it NOT NULL.
- episodeno is the FOREIGN KEY which is the part of the composite primary key for this table which makes it NOT NULL.
- hostid is the third PRIMARY KEY for this table.
- host_amount is NOT NULL because the table has the data for the payment made to the host.
- Referential Integrity: hostedby is dependent on podcastEpisode with podcastid and episodeno as Foreign Key, any update and deletion on a particular podcastid and episodeno in podcastEpisode table will have a cascading action on hostedby table.

- Referential Integrity: hostedby is dependent on podcastHost with hostid as Foreign Key, any update and deletion on a particular hostid in podcastHost table will have a cascading action on hostedby table.

Creators(<u>creatorsid</u>, cf_name, cl_name)

- creatorsid is the PRIMARY KEY which makes it NOT NULL.
- cf_name and cl_name are NOT NULL as it is important to maintain the data for the creators.

artist(<u>creatorsid</u>, labelname, a_status, type, primary_genre, monthly_listeners, a_country)

- creatorsid is the FOREIGN KEY which is the part of the composite primary key for this table which makes it NOT NULL.
- labelname is the FOREIGN KEY which makes it NOT NULL.
- a_status , type, primary_genre, a_country are NOT NULL because it is necessary to maintain the data for the artist.
- monthly_listeners are NOT NULL but its default value is set to 0 in case any new artist joins in.
- Referential Integrity: artist is-a creator with creatorsid as Foreign Key, any update and deletion on a particular creatorsid in Creators table will have a cascading action on artist table.

3. Base Relations:

```
CREATE TABLE Media(
mediaid              VARCHAR(5) NOT NULL,
media_name           VARCHAR(80) NOT NULL,
genre                VARCHAR(80) NOT NULL,
language             VARCHAR(80) DEFAULT NULL,
m_country            VARCHAR(80) DEFAULT NULL,
PRIMARY KEY(mediaID));


CREATE TABLE Song(
mediaid              VARCHAR(5) NOT NULL,
albumid              VARCHAR(5) DEFAULT NULL,
duration             FLOAT NOT NULL,
s_release_date       DATE NOT NULL,
royalty_rate         FLOAT NOT NULL,
royalty_paid         TINYINT(1) NOT NULL DEFAULT 0,
track_no             INT(11) DEFAULT NULL,
splay_count          INT(11) NOT NULL,
PRIMARY KEY(mediaid) ,
FOREIGN KEY(mediaid) REFERENCES Media (mediaid)
ON UPDATE CASCADE ON DELETE CASCADE
FOREIGN KEY(albumid) REFERENCES Album(albumid)
ON UPDATE CASCADE ON DELETE CASCADE
);


CREATE TABLE Podcast(
mediaid              VARCHAR(5) NOT NULL,
episode_count        INT NOT NULL,
hostid               VARCHAR(5) NOT NULL,
PRIMARY KEY(mediaid),
FOREIGN KEY(hostid) REFERENCES podcastHost(creatorsid) ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY(mediaid) REFERENCES Media (mediaid)
ON UPDATE CASCADE);
```

```
CREATE table podcast_record(
podcastid        VARCHAR(5) NOT NULL,
pd_date          DATE NOT NULL,
rating           FLOAT NOT NULL,
total_subscribers INT NOT NULL,
PRIMARY KEY(podcastid,pd_date),
FOREIGN KEY(podcastid) REFERENCES Podcast(mediaid) ON DELETE CASCADE
ON UPDATE CASCADE);


CREATE TABLE sponsors(
sid              VARCHAR(5) NOT NULL,
s_name           VARCHAR(80) NOT NULL,
PRIMARY KEY(sid)) ;



CREATE TABLE sponsoredby (
sid              VARCHAR(5) NOT NULL,
Podcastid        VARCHAR(5) NOT NULL,
PRIMARY KEY(sid, podcastid),
FOREIGN KEY (sid)
        REFERENCES sponsors(sid)
        ON UPDATE CASCADE,
FOREIGN KEY (podcastid)
        REFERENCES Podcast(mediaid)
        ON UPDATE CASCADE);



CREATE TABLE Album(
albumid          VARCHAR(5) NOT NULL,
album_name       VARCHAR(80) NOT NULL,
a_release_year   YEAR(4) NOT NULL,
edition          VARCHAR(50)
                 CHECK(edition in('special','limited','collector\'s edition'),
artistid         VARCHAR(5) DEFAULT NULL,
PRIMARY KEY(albumid);
```

```
CREATE TABLE composedby(
songid       VARCHAR(5) NOT NULL,
artistid     VARCHAR(5) NOT NULL,
lead         ENUM('YES', 'NO') NOT NULL,
PRIMARY KEY(songid, artistid),
FOREIGN KEY(songid) REFERENCES Song(mediaid)
ON UPDATE CASCADE,
FOREIGN KEY(artistid) REFERENCES Artist(creatorsid)
ON UPDATE CASCADE);


CREATE TABLE podcastEpisode (
podcastid          VARCHAR(5) NOT NULL,
episodeno          FLOAT NOT NULL,
title              VARCHAR(80) DEFAULT 'Untitled',
p_duration         FLOAT NOT NULL,
p_release_date     DATE NOT NULL,
PRIMARY KEY(podcastid , episodeno),
FOREIGN KEY (podcastid)
        REFERENCES Podcast(mediaid)
        ON DELETE CASCADE ON UPDATE CASCADE);



CREATE TABLE podcastEpisode_listening (
podcastid          VARCHAR(5) NOT NULL,
episodeno          FLOAT NOT NULL,
pel_date           DATE NOT NULL,
listening_count    INT DEFAULT 0,
ad_count           INT  DEFAULT 0,
PRIMARY KEY(podcastid , episodeno, pel_date),
FOREIGN KEY (podcastid, episodeno)
        REFERENCES podcastEpisode(podcastid, episodeno)
        ON UPDATE CASCADE);
```

```
CREATE TABLE Creators(
creatorsid      VARCHAR(80),
cf_name         VARCHAR(80) NOT NULL,
cl_name         VARCHAR(80) NOT NULL,
PRIMARY KEY(creatorsid),
);


CREATE TABLE recordLabel (
labelname varchar(80),
PRIMARY KEY(labelname)
);



CREATE TABLE Artist(
creatorsid      VARCHAR(5),
labelname       VARCHAR(80) NOT NULL,
a_status        VARCHAR(80)
                 CHECK (a_status IN('Active','Retired')),
type            VARCHAR(80)
                 CHECK (type IN('Band','Musician','Composer')),
primary_genre          VARCHAR(80) NOT NULL,
monthly_listeners      INT NOT NULL,
a_country              VARCHAR(80) NOT NULL,
PRIMARY KEY(creatorsid),
FOREIGN KEY (creatorsid) REFERENCES Creators(creatorsid) ON UPDATE
CASCADE,
FOREIGN KEY (labelname) REFERENCES recordLabel(labelname) ON UPDATE
CASCADE);



CREATE TABLE podcastHost(
creatorsid    VARCHAR(5) NOT NULL,
email         VARCHAR(80) DEFAULT NULL,
phone         INT DEFAULT NULL,
city           VARCHAR(80) DEFAULT NULL,
PRIMARY KEY(creatorsid), FOREIGN KEY(creatorsid)
```

```
REFERENCES Creators(creatorsid))
ON DELETE CASCADE ON UPDATE CASCADE;

CREATE TABLE hostedby(
podcastid       VARCHAR(5) NOT NULL,
episodeno       FLOAT NOT NULL,
hostid          VARCHAR(5) NOT NULL,
host_amount  INT NOT NULL,
PRIMARY KEY(podcastid,epsiodeno,hostid),
FOREIGN KEY(hostid) REFERENCES podcastHost(creatorsid)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY(podcastid,episodeno) REFERENCES
podcastEpisode(podcastid,episodeno)
ON DELETE CASCADE ON UPDATE CASCADE,



CREATE TABLE user_payment(
uid             VARCHAR(5),
uphone      INT NOT NULL,
ups_date    DATE NOT NULL,
PRIMARY KEY(uid),
FOREIGN KEY(uphone) REFERENCES user(uphone) ON UPDATE CASCADE ON
DELETE CASCADE);


CREATE TABLE songPayment(
spayid INT NOT NULL AUTO_INCREMENT,
songid VARCHAR(5) NOT NULL,
artistid VARCHAR(5) NOT NULL,
labelname VARCHAR(80) NOT NULL,
label_payment FLOAT NOT NULL,
artist_payment FLOAT NOT NULL,
spay_date DATE NOT NULL,
Main_label VARCHAR(10) NOT NULL DEFAULT 'No',
PRIMARY KEY (spayid),
FOREIGN KEY (songid) REFERENCES Song(mediaid) ON UPDATE CASCADE,
FOREIGN KEY (labelname) REFERENCES recordLabel(labelname) ON UPDATE
CASCADE,
```

```
FOREIGN KEY (artistid) REFERENCES Artist(creatorsid) ON UPDATE CASCADE
);

CREATE TABLE user(
uphone              INT NOT NULL,
start_date          date     NOT NULL,
end_date            date      NOT NULL,
uf_name             VARCHAR(80) NOT NULL,
ul_name             VARCHAR(80) NOT NULL,
uemail              VARCHAR(80),
u_status            VARCHAR(80) CHECK (u_status IN('Active','Inactive')) ,
sub_fee             FLOAT,
PRIMARY KEY(uphone)
);


CREATE TABLE listensto (
uphone          INT NOT NULL,
mediaid         VARCHAR(5) NOT NULL,
uplay_date      DATE NOT NULL,
dplay_count     INT NOT NULL DEFAULT '0',
PRIMARY KEY (uphone,mediaid,uplay_date),
FOREIGN KEY (uphone) REFERENCES  user(uphone) ON DELETE CASCADE ON
UPDATE CASCADE,
FOREIGN KEY (mediaid) REFERENCES  Media (mediaid) ON DELETE CASCADE ON
UPDATE CASCADE
```

```
[ MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  show tables
+--------------------------+
| Tables_in_hsangha        |
+--------------------------+
| Album                    |
| Artist                   |
| Creators                 |
| Media                    |
| Podcast                  |
| Song                     |
| composedby               |
| hostedby                 |
| listensto                |
| podcastEpisode           |
| podcastEpisode_listening |
| podcastHost              |
| podcast_record           |
| recordLabel              |
| songPayment              |
| sponsoredby              |
| sponsors                 |
| user                     |
| user_payment             |
+--------------------------+
19 rows in set (0.0022 sec)
```

SELECT STATEMENTS:-

SELECT * FROM Media;

```
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select * from Media;
+---------+--------------------+------------+----------+-----------+
| mediaid | media_name         | genre      | language | m_country |
+---------+--------------------+------------+----------+-----------+
| M11     | Bones              | Rock       | English  | USA       |
| M12     | Unholy             | Pop        | English  | USA       |
| M13     | Heaven             | multigenre | English  | USA       |
| M14     | Waking Up Dreaming | Dance      | English  | Canada    |
| M15     | Chammak Challo     | Party      | Hindi    | India     |
| M16     | Toota hua          | romance    | hindi    | India     |
| P11     | Joe Rogan          | Interview  | English  | USA       |
| P12     | Daily              | News       | English  | USA       |
| P13     | Conan              | Comedy     | English  | USA       |
| P14     | Sherlock Holmes    | Fiction    | French   | France    |
| P15     | Echoes of India    | History    | Hindi    | India     |
| P16     | Justice            | comedy     | English  | USA       |
+---------+--------------------+------------+----------+-----------+
12 rows in set (0.0014 sec)
```

SELECT * FROM Song;

```
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select * from Song;
+---------+----------+---------------+--------------+--------------+---------+----------+-------------+
| mediaid | duration | s_release_date | royalty_rate | royalty_paid | albumid | track_no | splay_count |
+---------+----------+---------------+--------------+--------------+---------+----------+-------------+
| M11     |      272 | 2022-02-15    |            4 |            0 | A11     |        1 |           5 |
| M12     |      305 | 2022-07-07    |            2 |            0 | A12     |        1 |           3 |
| M13     |      286 | 2022-03-04    |            1 |            1 | A12     |        2 |           5 |
| M14     |      182 | 2022-01-01    |            7 |            0 | A11     |        2 |          12 |
| M15     |      301 | 2016-04-05    |           12 |            1 | A13     |        1 |           3 |
| M16     |      320 | 2012-02-14    |            8 |            1 | A14     |        1 |           7 |
+---------+----------+---------------+--------------+--------------+---------+----------+-------------+
6 rows in set (0.0018 sec)
```

SELECT * FROM Podcast;

```
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select * from Podcast;
+---------+---------------+---------+
| mediaid | episode_count | hostid  |
+---------+---------------+---------+
| P11     |            10 | PH11    |
| P12     |           100 | PH12    |
| P13     |            25 | PH11    |
| P14     |             5 | PH13    |
| P15     |             1 | PH14    |
+---------+---------------+---------+
5 rows in set (0.0020 sec)
```

SELECT * FROM podcast_record;

```
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select * from podcast_record;

+-----------+------------+--------+-------------------+
| podcastid | pd_date    | rating | total_subscribers |
+-----------+------------+--------+-------------------+
| P11       | 2022-01-01 |      8 |                 2 |
| P11       | 2022-02-01 |      7 |                 3 |
| P12       | 2022-01-01 |      9 |                 3 |
| P13       | 2022-02-01 |      9 |                 4 |
| P14       | 2022-01-01 |      6 |                 1 |
| P15       | 2022-02-02 |      9 |                 5 |
+-----------+------------+--------+-------------------+
6 rows in set (0.0046 sec)
```

SELECT * FROM sponsors;

sid	s_name
S11	Dhruv Patel
S12	Gautam Ambani
S13	Satya Nadella
S14	Sundar Pichai
S15	Harshil Sanghavi

SELECT * FROM sponsoredby;

sid	podcastid
S11	P14
S12	P12
S13	P11
S14	P12
S15	P13

SELECT * FROM Album;

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Album;
+---------+------------+----------------+----------------------+----------+
| albumid | album_name | a_release_year | edition              | artistid |
+---------+------------+----------------+----------------------+----------+
| A11     | Bones      |           2022 | special              | CR12     |
| A12     | Gloria     |           2022 | limited              | NULL     |
| A13     | RaOne      |           2016 | collectors's edition | CR11     |
| A14     | AI2        |           2012 | special              | CR16     |
+---------+------------+----------------+----------------------+----------+
4 rows in set (0.0017 sec)
```

SELECT * FROM composedBy;

songid	artistid	lead
M11	CR12	Yes
M11	CR18	No
M11	CR19	No
M12	CR13	Yes
M13	CR14	Yes
M14	CR12	Yes
M14	CR17	No
M15	CR11	Yes
M15	CR15	No

SELECT * FROM podcastEpsiode;

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcastEpisode;
+-----------+----------+-----------------+------------+----------------+----------+----------+-------+
| podcastid | episodeno | title           | p_duration | p_release_date | flat_fee | ad_count | bonus |
+-----------+----------+-----------------+------------+----------------+----------+----------+-------+
| P11       |        1 | North Carolina  |         21 | 2020-08-01     |       20 |        3 |     6 |
| P12       |        1 | ChatGPT         |         28 | 2023-02-03     |       10 |        1 |     2 |
| P12       |        2 | modern          |         31 | 2023-03-04     |       20 |        4 |     8 |
| P13       |        1 | Need a Friend   |         27 | 2023-03-04     |       35 |        2 |     4 |
| P14       |        1 | Lying Detective |         40 | 2020-07-11     |       30 |        8 |    16 |
| P14       |        2 | modern          |         41 | 2020-08-11     |       20 |        3 |     6 |
| P15       |        1 | Mangalyaan      |         25 | 2022-05-05     |       80 |        1 |     2 |
| P15       |        2 | Chandrayan      |         28 | 2023-01-25     |       40 |        3 |     6 |
| P15       |        3 | Gaganyaan       |         26 | 2023-01-28     |       25 |        3 |     6 |
+-----------+----------+-----------------+------------+----------------+----------+----------+-------+
9 rows in set (0.0018 sec)
```

SELECT * FROM podcastEpsiode_listening;

```
[ MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select * from podcastEpisode_listening;
+-----------+-----------+------------+-----------------+
| podcastid | episodeno | pel_date   | listening_count |
+-----------+-----------+------------+-----------------+
| P11       |         1 | 2023-02-01 |             100 |
| P11       |         1 | 2023-03-01 |             170 |
| P12       |         1 | 2023-02-01 |              80 |
| P12       |         2 | 2023-03-01 |              75 |
| P13       |         1 | 2023-04-01 |             200 |
| P14       |         1 | 2023-02-01 |             120 |
| P14       |         2 | 2023-03-01 |             150 |
| P15       |         1 | 2023-02-01 |             125 |
| P15       |         1 | 2023-03-01 |             250 |
+-----------+-----------+------------+-----------------+
```

SELECT * FROM Creators;

```
[ MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select * from Creators;
+------------+---------+-----------+
| creatorsid | cf_name | cl_name   |
+------------+---------+-----------+
| CR11       | Ravi    | Ghevariya |
| CR12       | Imagine | Dragons   |
| CR13       | Sam     | Smith     |
| CR14       | Kim     | Petras    |
| CR15       | Mitul   | Patel     |
| CR16       | Artharv | Pandit    |
| CR17       | John    | Mckee     |
| CR18       | Dan     | Paris     |
| CR19       | Julia   | Vento     |
| CR20       | Alex    | patel     |
| PH11       | Joe     | Rogan     |
| PH12       | Steve   | Colbert   |
| PH13       | Conan   | Brien     |
| PH14       | Conan   | Dyle      |
| PH15       | Harsha  | Bhogle    |
+------------+---------+-----------+
15 rows in set (0.0016 sec)
```

SELECT * FROM recordLabel;

labelname
AV Productions
LN 32 Music Group
Lorimer Studios
MSeries
Pluto

SELECT * FROM Artist;

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Artist;
```

creatorsid	labelname	a_status	type	primary_genre	monthly_listeners	a_country
CR11	Lorimer Studios	Active	Band	Bollywood	60	India
CR12	Pluto	Active	Band	Rock	10	USA
CR13	LN 32 Music Group	Retired	Musician	Lofi	30	Switzerland
CR14	LN 32 Music Group	Active	Composer	Romance	13	New Zealand
CR15	MSeries	Active	Composer	Rock	17	India
CR16	AV Productions	Active	Composer	Pop	63	India
CR17	Pluto	Active	Composer	Romance	60	USA
CR18	AV Productions	Active	Composer	Rock	17	France
CR19	Pluto	Retired	Musician	Lofi	10	USA
CR20	Pluto	active	musician	lofi	20	India

10 rows in set (0.0020 sec)

SELECT * FROM podcastHost;

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcastHost;
```

creatorsid	email	phone	city
PH11	jrogan@gmail.com	NULL	Raleigh
PH12	scolbert@gmail.com	NULL	New York
PH13	cbriend@gmail.com	2102376873	NULL
PH14	NULL	NULL	London
PH15	hbhogle@gmail.com	2141874387	Mumbai

5 rows in set (0.0017 sec)

SELECT * FROM hostedby;

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from hostedby;
+-----------+-----------+--------+-------------+
| podcastid | episodeno | hostid | host_amount |
+-----------+-----------+--------+-------------+
| P11       |         1 | PH11   |          26 |
| P12       |         1 | PH12   |          12 |
| P12       |         2 | PH12   |          28 |
| P13       |         1 | PH13   |          39 |
| P14       |         1 | PH14   |          46 |
| P14       |         2 | PH14   |          26 |
| P15       |         1 | PH15   |          82 |
| P15       |         2 | PH12   |          46 |
+-----------+-----------+--------+-------------+
8 rows in set (0.0018 sec)
```

SELECT * FROM user_payment;

uid	uphone	ups_date
US11	2131219087	2023-01-01
US12	2131219087	2023-02-01
US13	2014740314	2023-01-01
US14	2146753981	2023-01-01

SELECT * FROM songPayment;

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from songPayment;
+--------+--------+----------+------------------+---------------+-----------------+------------+------------+
| spayid | songid | artistid | labelname        | label_payment | artist_payment | spay_date  | Main_label |
+--------+--------+----------+------------------+---------------+-----------------+------------+------------+
|     17 | M11    | CR12     | Pluto            |            20 |         4.66667 | 2023-02-01 | Yes        |
|     18 | M14    | CR12     | Pluto            |            84 |            29.4 | 2023-02-01 | Yes        |
|     19 | M12    | CR13     | LN 32 Music Group |             6 |             4.2 | 2023-02-01 | Yes        |
|     20 | M14    | CR17     | Pluto            |            84 |            29.4 | 2023-02-01 | No         |
|     21 | M11    | CR18     | AV Productions   |            20 |         4.66667 | 2023-02-01 | No         |
|     22 | M11    | CR19     | Pluto            |            20 |         4.66667 | 2023-02-01 | No         |
|     24 | M15    | CR11     | Lorimer Studios  |            48 |            16.8 | 2023-03-01 | Yes        |
|     25 | M11    | CR12     | Pluto            |            24 |             5.6 | 2023-03-01 | Yes        |
|     26 | M14    | CR12     | Pluto            |            98 |            34.3 | 2023-03-01 | Yes        |
|     27 | M12    | CR13     | LN 32 Music Group |             8 |             5.6 | 2023-03-01 | Yes        |
|     28 | M13    | CR14     | LN 32 Music Group |             7 |             4.9 | 2023-03-01 | Yes        |
|     29 | M15    | CR15     | MSeries          |            48 |            16.8 | 2023-03-01 | No         |
|     30 | M16    | CR16     | AV Productions   |            72 |            50.4 | 2023-03-01 | Yes        |
|     31 | M14    | CR17     | Pluto            |            98 |            34.3 | 2023-03-01 | No         |
|     32 | M11    | CR18     | AV Productions   |            24 |             5.6 | 2023-03-01 | No         |
|     33 | M11    | CR19     | Pluto            |            24 |             5.6 | 2023-03-01 | No         |
+--------+--------+----------+------------------+---------------+-----------------+------------+------------+
16 rows in set (0.0018 sec)
```

SELECT * FROM user;

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from user;
+------------+------------+------------+----------+----------+----------------------+----------+---------+
| uphone     | join_date  | end_date   | uf_name  | ul_name  | uemail               | u_status | sub_fee |
+------------+------------+------------+----------+----------+----------------------+----------+---------+
| 1111111111 | 2023-03-06 | NULL       | Joey     | Tribiani | joey@gmail.com       | Active   |      15 |
| 1111111112 | 2023-03-07 | NULL       | Pete     | Maverick | maverick@gmail.com   | Active   |      15 |
| 1111111113 | 2023-01-04 | NULL       | Tony     | Stark    | NULL                 | Active   |      15 |
| 1111111114 | 2023-01-11 | NULL       | Vishwa   | Gandhi   | vgandhi@lexis.com    | Active   |      15 |
| 1111111115 | 2023-01-09 | NULL       | Dwight   | Schrute  | dschrute@mifflin.com | Active   |      15 |
| 1111111116 | 2023-01-02 | 2023-02-01 | Jethalal | Gada     | jgada@electronics.com| Inactive |       0 |
| 1111111117 | 2023-03-02 | NULL       | Steve    | Rogers   | Srogers@gmail.com    | Active   |      15 |
| 1111111118 | 2023-01-07 | 2023-02-06 | Michael  | Scott    | mscott@dundler.com   | Inactive |       0 |
+------------+------------+------------+----------+----------+----------------------+----------+---------+
8 rows in set (0.0017 sec)
```

SELECT * FROM listensto;

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from listensto;
+------------+---------+------------+-------------+
| uphone     | mediaid | uplay_date | dplay_count |
+------------+---------+------------+-------------+
| 1111111111 | M11     | 2023-02-01 |           2 |
| 1111111111 | M13     | 2023-02-02 |           1 |
| 1111111111 | M15     | 2023-02-01 |           3 |
| 1111111111 | P12     | 2023-02-07 |           3 |
| 1111111111 | P12     | 2023-02-10 |           1 |
| 1111111112 | M11     | 2023-02-05 |           3 |
| 1111111112 | P13     | 2023-02-05 |           1 |
| 1111111112 | P15     | 2023-02-05 |           1 |
| 1111111113 | P12     | 2023-02-01 |           2 |
| 1111111114 | M14     | 2023-02-07 |           4 |
| 1111111114 | P12     | 2023-02-04 |           1 |
| 1111111115 | M14     | 2023-02-04 |           6 |
| 1111111117 | M14     | 2023-02-08 |           2 |
| 1111111117 | P15     | 2023-02-07 |           5 |
+------------+---------+------------+-------------+
14 rows in set (0.0017 sec)
```

4.1) TASKS AND OPERATIONS:-

Task 1: Information Processing

Enter basic information about Songs

To enter the information relating to the song we first need to enter the record into the media entity since the song is in the isa hierarchy of the media thus it is connected to the media table and has common attribute mediaid.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into Media values('M17','Ghost','Pop','English','USA');
Query OK, 1 row affected (0.0033 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into Song values('M17',236,'2022-07-07',12,0,'A14',2,4);
Query OK, 1 row affected (0.0035 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Media natural join Song;
+---------+---------------------+------------+----------+-----------+----------+----------------+--------------+---------------+---------+----------+--------------+
| mediaid | media_name          | genre      | language | m_country | duration | s_release_date | royalty_rate | royalty_paid | albumid | track_no | splay_count |
+---------+---------------------+------------+----------+-----------+----------+----------------+--------------+---------------+---------+----------+--------------+
| M11     | Bones               | Rock       | English  | USA       |      272 | 2022-02-15     |            4 |             0 | A11     |        1 |            5 |
| M12     | Unholy              | Pop        | English  | USA       |      305 | 2022-07-07     |            2 |             0 | A12     |        1 |            3 |
| M13     | Heaven              | multigenre | English  | USA       |      286 | 2022-03-04     |            1 |             1 | A12     |        2 |            5 |
| M14     | Waking Up Dreaming  | Dance      | English  | Canada    |      182 | 2022-01-01     |            7 |             0 | A11     |        2 |            0 |
| M15     | Chammak Challo      | Party      | Hindi    | India     |      301 | 2016-04-05     |           12 |             1 | A13     |        1 |            3 |
| M16     | Toota hua           | romance    | hindi    | India     |      320 | 2012-02-14     |            8 |             1 | A14     |        1 |            7 |
| M17     | Ghost               | Pop        | English  | USA       |      236 | 2022-07-07     |           12 |             0 | A14     |        2 |            4 |
+---------+---------------------+------------+----------+-----------+----------+----------------+--------------+---------------+---------+----------+--------------+
7 rows in set (0.0026 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL
```

Update basic information about Songs

The song has some of the attributes in the media table and some of them in the song table. To update any attribute related to the song that respective table needs to be updated.

```
+---------+---------------------+------------+----------+-----------+----------+----------------+--------------+---------------+---------+----------+--------------+
| mediaid | media_name          | genre      | language | m_country | duration | s_release_date | royalty_rate | royalty_paid | albumid | track_no | splay_count |
+---------+---------------------+------------+----------+-----------+----------+----------------+--------------+---------------+---------+----------+--------------+
| M11     | Bones               | Rock       | English  | USA       |      272 | 2022-02-15     |            4 |             0 | A11     |        1 |            5 |
| M12     | Unholy              | Pop        | English  | USA       |      305 | 2022-07-07     |            2 |             0 | A12     |        1 |            3 |
| M13     | Heaven              | multigenre | English  | USA       |      286 | 2022-03-04     |            1 |             1 | A12     |        2 |            5 |
| M14     | Waking Up Dreaming  | Dance      | English  | Canada    |      182 | 2022-01-01     |            7 |             0 | A11     |        2 |            0 |
| M15     | Chammak Challo      | Party      | Hindi    | India     |      301 | 2016-04-05     |           12 |             1 | A13     |        1 |            3 |
| M16     | Toota hua           | romance    | hindi    | India     |      320 | 2012-02-14     |            8 |             1 | A14     |        1 |            7 |
| M17     | Ghost               | Pop        | English  | USA       |      236 | 2022-07-07     |           12 |             0 | A14     |        2 |            4 |
+---------+---------------------+------------+----------+-----------+----------+----------------+--------------+---------------+---------+----------+--------------+
7 rows in set (0.0026 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Song SET duration=242,s_release_date='2022-05-02',royalty_rate=1,royalty_paid=1,albumid='A12',track_no=3,splay_count=2 where mediaid='M17';
Query OK, 1 row affected (0.0038 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Media natural join Song;
+---------+---------------------+------------+----------+-----------+----------+----------------+--------------+---------------+---------+----------+--------------+
| mediaid | media_name          | genre      | language | m_country | duration | s_release_date | royalty_rate | royalty_paid | albumid | track_no | splay_count |
+---------+---------------------+------------+----------+-----------+----------+----------------+--------------+---------------+---------+----------+--------------+
| M11     | Bones               | Rock       | English  | USA       |      272 | 2022-02-15     |            4 |             0 | A11     |        1 |            5 |
| M12     | Unholy              | Pop        | English  | USA       |      305 | 2022-07-07     |            2 |             0 | A12     |        1 |            3 |
| M13     | Heaven              | multigenre | English  | USA       |      286 | 2022-03-04     |            1 |             1 | A12     |        2 |            5 |
| M14     | Waking Up Dreaming  | Dance      | English  | Canada    |      182 | 2022-01-01     |            7 |             0 | A11     |        2 |            0 |
| M15     | Chammak Challo      | Party      | Hindi    | India     |      301 | 2016-04-05     |           12 |             1 | A13     |        1 |            3 |
| M16     | Toota hua           | romance    | hindi    | India     |      320 | 2012-02-14     |            8 |             1 | A14     |        1 |            7 |
| M17     | Ghost               | Pop        | English  | USA       |      242 | 2022-05-02     |            1 |             1 | A12     |        3 |            2 |
+---------+---------------------+------------+----------+-----------+----------+----------------+--------------+---------------+---------+----------+--------------+
7 rows in set (0.0029 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL

MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Song SET albumid='A11' where mediaid='M17';
Query OK, 1 row affected (0.0032 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Song SET track_no=3 where mediaid='M17';
Query OK, 1 row affected (0.0034 sec)

Rows matched: 1  Changed: 1  Warnings: 0
```

Delete basic information about Songs

To delete any of the records from the song we need to delete it from the media table. On deletion from the media table the record will automatically be deleted from the song table.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Media natural join Song;
+---------+-------------------+-----------+----------+-----------+----------+----------------+--------------+--------------+---------+----------+-------------+
| mediaid | media_name        | genre     | language | m_country | duration | s_release_date | royalty_rate | royalty_paid | albumid | track_no | splay_count |
+---------+-------------------+-----------+----------+-----------+----------+----------------+--------------+--------------+---------+----------+-------------+
| M11     | Bones             | Rock      | English  | USA       |      272 | 2022-02-15     |            4 |            0 | A11     |        1 |           5 |
| M12     | Unholy            | Pop       | English  | USA       |      305 | 2022-07-07     |            2 |            0 | A12     |        1 |           3 |
| M13     | Heaven            | multigenre| English  | USA       |      286 | 2022-03-04     |            1 |            1 | A12     |        2 |           5 |
| M14     | Waking Up Dreaming| Dance     | English  | Canada    |      182 | 2022-01-01     |            7 |            0 | A11     |        2 |           0 |
| M15     | Chammak Challo    | Party     | Hindi    | India     |      301 | 2016-04-05     |           12 |            1 | A13     |        1 |           3 |
| M16     | Toota hua         | romance   | hindi    | India     |      320 | 2012-02-14     |            8 |            1 | A14     |        1 |           7 |
| M17     | Ghost             | Pop       | English  | USA       |      242 | 2022-05-02     |            1 |            1 | A12     |        3 |           2 |
+---------+-------------------+-----------+----------+-----------+----------+----------------+--------------+--------------+---------+----------+-------------+
7 rows in set (0.0029 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL delete from Media where mediaid='M17';
Query OK, 1 row affected (0.0037 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Song;
+---------+----------+----------------+--------------+--------------+---------+----------+-------------+
| mediaid | duration | s_release_date | royalty_rate | royalty_paid | albumid | track_no | splay_count |
+---------+----------+----------------+--------------+--------------+---------+----------+-------------+
| M11     |      272 | 2022-02-15     |            4 |            0 | A11     |        1 |           5 |
| M12     |      305 | 2022-07-07     |            2 |            0 | A12     |        1 |           3 |
| M13     |      286 | 2022-03-04     |            1 |            1 | A12     |        2 |           5 |
| M14     |      182 | 2022-01-01     |            7 |            0 | A11     |        2 |           0 |
| M15     |      301 | 2016-04-05     |           12 |            1 | A13     |        1 |           3 |
| M16     |      320 | 2012-02-14     |            8 |            1 | A14     |        1 |           7 |
+---------+----------+----------------+--------------+--------------+---------+----------+-------------+
6 rows in set (0.0023 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL
```

Enter basic information about Album

It can directly be entered into the album table.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into Album values('A15','ABC','2018','limited','CR11');
Query OK, 1 row affected (0.0039 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Album;
+---------+------------+-----------------+----------------------+-----------+
| albumid | album_name | a_release_year | edition              | artistid |
+---------+------------+-----------------+----------------------+-----------+
| A11     | Bones      |            2022 | special              | NULL     |
| A12     | Gloria     |            2022 | limited              | NULL     |
| A13     | RaOne      |            2016 | collectors's edition | NULL     |
| A14     | AI2        |            2012 | special              | NULL     |
| A15     | ABC        |            2018 | limited              | CR11     |
+---------+------------+-----------------+----------------------+-----------+
5 rows in set (0.0027 sec)
```

Update basic information about Album

It can directly be updated into the Album table using the update statement in sql.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Album SET album_name='DEF',a_release_year='2021',edition='special',artistid='CR12' where albumid='A15';
Query OK, 1 row affected (0.0036 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Album;
+---------+------------+-----------------+----------------------+-----------+
| albumid | album_name | a_release_year | edition              | artistid |
+---------+------------+-----------------+----------------------+-----------+
| A11     | Bones      |            2022 | special              | NULL     |
| A12     | Gloria     |            2022 | limited              | NULL     |
| A13     | RaOne      |            2016 | collectors's edition | NULL     |
| A14     | AI2        |            2012 | special              | NULL     |
| A15     | DEF        |            2021 | special              | CR12     |
+---------+------------+-----------------+----------------------+-----------+
5 rows in set (0.0022 sec)
```

Delete basic information about Album

The record can directly be deleted from the album table using the delete statement in sql.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL delete from Album where albumid='A15';
Query OK, 1 row affected (0.0038 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Album;
+---------+-------------+-----------------+-----------------------+----------+
| albumid | album_name  | a_release_year  | edition               | artistid |
+---------+-------------+-----------------+-----------------------+----------+
| A11     | Bones       |            2022 | special               | NULL     |
| A12     | Gloria      |            2022 | limited               | NULL     |
| A13     | RaOne       |            2016 | collectors's edition  | NULL     |
| A14     | AI2         |            2012 | special               | NULL     |
+---------+-------------+-----------------+-----------------------+----------+
4 rows in set (0.0022 sec)
```

Enter basic information about Artist

To enter the information relating to the artist we first need to enter the record into the creators entity since the artist is in the isa hierarchy of the creators thus it is connected to the creators table and has common attribute creatorsid.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into Creators values('CR20','Joshua','Martin');
Query OK, 1 row affected (0.0172 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into Artist values('CR20','AV Productions','Active','Musician','Pop',5,'France');
Query OK, 1 row affected (0.0034 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL Select * from Artist;
+------------+------------------+----------+----------+--------------+------------------+-------------+
| creatorsid | labelname        | a_status | type     | primary_genre | monthly_listeners | a_country   |
+------------+------------------+----------+----------+--------------+------------------+-------------+
| CR11       | Lorimer Studios  | Active   | Band     | Bollywood    |               60 | India       |
| CR12       | Pluto            | Active   | Band     | Rock         |               10 | USA         |
| CR13       | LN 32 Music Group | Retired  | Musician | Lofi         |                0 | Switzerland |
| CR14       | LN 32 Music Group | Active   | Composer | Romance      |               13 | New Zealand |
| CR15       | MSeries          | Active   | Composer | Rock         |               17 | India       |
| CR16       | AV Productions   | Active   | Composer | Pop          |               63 | India       |
| CR17       | Pluto            | Active   | Composer | Romance      |               60 | USA         |
| CR18       | AV Productions   | Active   | Composer | Rock         |               17 | France      |
| CR19       | Lorimer Studios  | Retired  | Musician | Lofi         |               10 | USA         |
| CR20       | AV Productions   | Active   | Musician | Pop          |                5 | France      |
+------------+------------------+----------+----------+--------------+------------------+-------------+
10 rows in set (0.0023 sec)

MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Creators natural join Artist;
+------------+---------+----------+------------------+----------+----------+--------------+------------------+-------------+
| creatorsid | cf_name | cl_name  | labelname        | a_status | type     | primary_genre | monthly_listeners | a_country   |
+------------+---------+----------+------------------+----------+----------+--------------+------------------+-------------+
| CR11       | Ravi    | Ghevariya | Lorimer Studios  | Active   | Band     | Bollywood    |               60 | India       |
| CR12       | Imagine | Dragons  | Pluto            | Active   | Band     | Rock         |               10 | USA         |
| CR13       | Sam     | Smith    | LN 32 Music Group | Retired  | Musician | Lofi         |                0 | Switzerland |
| CR14       | Kim     | Petras   | LN 32 Music Group | Active   | Composer | Romance      |               13 | New Zealand |
| CR15       | Mitul   | Patel    | MSeries          | Active   | Composer | Rock         |               17 | India       |
| CR16       | Artharv | Pandit   | AV Productions   | Active   | Composer | Pop          |               63 | India       |
| CR17       | John    | Mckee    | Pluto            | Active   | Composer | Romance      |               60 | USA         |
| CR18       | Dan     | Paris    | AV Productions   | Active   | Composer | Rock         |               17 | France      |
| CR19       | Julia   | Vento    | Lorimer Studios  | Retired  | Musician | Lofi         |               10 | USA         |
| CR20       | Joshua  | Martin   | AV Productions   | Active   | Musician | Pop          |                5 | France      |
+------------+---------+----------+------------------+----------+----------+--------------+------------------+-------------+
10 rows in set (0.0103 sec)
```

Update basic information about Artist

The artist has some of the attributes in the creators table and some of them in the artist table. To update any attribute related to the artist that respective table needs to be updated.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Artist SET labelname='MSeries',a_status='Retired',type='Band',primary_genre='Lofi',monthly_listeners=4,a_country='Peru' where creatorsid='CR20';
Query OK, 1 row affected (0.0046 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Creators natural join Artist;
+------------+---------+----------+-------------------+----------+----------+---------------+--------------------+-------------+
| creatorsid | cf_name | cl_name  | labelname         | a_status | type     | primary_genre | monthly_listeners | a_country   |
+------------+---------+----------+-------------------+----------+----------+---------------+--------------------+-------------+
| CR11       | Ravi    | Ghevariya | Lorimer Studios   | Active   | Band     | Bollywood     |                 60 | India       |
| CR12       | Imagine | Dragons  | Pluto             | Active   | Band     | Rock          |                 10 | USA         |
| CR13       | Sam     | Smith    | LN 32 Music Group | Retired  | Musician | Lofi          |                  0 | Switzerland |
| CR14       | Kim     | Petras   | LN 32 Music Group | Active   | Composer | Romance       |                 13 | New Zealand |
| CR15       | Mitul   | Patel    | MSeries           | Active   | Composer | Rock          |                 17 | India       |
| CR16       | Artharv | Pandit   | AV Productions    | Active   | Composer | Pop           |                 63 | India       |
| CR17       | John    | Mckee    | Pluto             | Active   | Composer | Romance       |                 60 | USA         |
| CR18       | Dan     | Paris    | AV Productions    | Active   | Composer | Rock          |                 17 | France      |
| CR19       | Julia   | Vento    | Lorimer Studios   | Retired  | Musician | Lofi          |                 10 | USA         |
| CR20       | Joshua  | Martin   | MSeries           | Retired  | Band     | Lofi          |                  4 | Peru        |
+------------+---------+----------+-------------------+----------+----------+---------------+--------------------+-------------+
10 rows in set (0.0033 sec)
```

Delete basic information about Artist

To delete any of the records from the artist table we need to delete it from the creators table. On deletion from the creators table the record will automatically be deleted from the artist table.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL delete from Creators where creatorsid='CR20';
Query OK, 1 row affected (0.0094 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Creators natural join Artist;
+------------+---------+----------+-------------------+----------+----------+---------------+--------------------+-------------+
| creatorsid | cf_name | cl_name  | labelname         | a_status | type     | primary_genre | monthly_listeners | a_country   |
+------------+---------+----------+-------------------+----------+----------+---------------+--------------------+-------------+
| CR11       | Ravi    | Ghevariya | Lorimer Studios   | Active   | Band     | Bollywood     |                 60 | India       |
| CR12       | Imagine | Dragons  | Pluto             | Active   | Band     | Rock          |                 10 | USA         |
| CR13       | Sam     | Smith    | LN 32 Music Group | Retired  | Musician | Lofi          |                  0 | Switzerland |
| CR14       | Kim     | Petras   | LN 32 Music Group | Active   | Composer | Romance       |                 13 | New Zealand |
| CR15       | Mitul   | Patel    | MSeries           | Active   | Composer | Rock          |                 17 | India       |
| CR16       | Artharv | Pandit   | AV Productions    | Active   | Composer | Pop           |                 63 | India       |
| CR17       | John    | Mckee    | Pluto             | Active   | Composer | Romance       |                 60 | USA         |
| CR18       | Dan     | Paris    | AV Productions    | Active   | Composer | Rock          |                 17 | France      |
| CR19       | Julia   | Vento    | Lorimer Studios   | Retired  | Musician | Lofi          |                 10 | USA         |
+------------+---------+----------+-------------------+----------+----------+---------------+--------------------+-------------+
9 rows in set (0.0025 sec)
```

Enter basic information about Podcast

To enter the information relating to the podcast we first need to enter the record into the media entity since the podcast is in the isa hierarchy of the media thus it is connected to the media table and has common attribute mediaid. Podcast_record has some of the attributes of the podcasts which are maintained every month and hence the information can be inserted into that table.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into Media values ('P17','BBC','Doc','Hindi','India');
Query OK, 1 row affected (0.0042 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into Podcast values('P17',4,'NA');
Query OK, 1 row affected (0.0040 sec)
```

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into podcast_record values ('P17','2023-01-01',9,5);
Query OK, 1 row affected (0.0043 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Media natural join Podcast join podcast_record on (Podcast.mediaid = podcast_record.podcastid);
+---------+-----------------+-----------+----------+-----------+---------------+---------+-----------+------------+--------+-------------------+
| mediaid | media_name      | genre     | language | m_country | episode_count | hostid  | podcastid | pd_date    | rating | total_subscribers |
+---------+-----------------+-----------+----------+-----------+---------------+---------+-----------+------------+--------+-------------------+
| P11     | Joe Rogan       | Interview | English  | USA       |            10 | PH11    | P11       | 2022-01-01 |      8 |                 2 |
| P11     | Joe Rogan       | Interview | English  | USA       |            10 | PH11    | P11       | 2022-02-01 |      7 |                 3 |
| P12     | Daily           | News      | English  | USA       |           100 | PH12    | P12       | 2022-01-01 |      8 |                 1 |
| P13     | Conan           | Comedy    | English  | USA       |            25 | PH11    | P13       | 2022-02-01 |      9 |                 4 |
| P14     | Sherlock Holmes | Fiction   | French   | France    |             5 | PH13    | P14       | 2022-01-01 |      6 |                 1 |
| P15     | Echoes of India | History   | Hindi    | India     |             1 | NA      | P15       | 2022-02-02 |      9 |                 5 |
| P17     | BBC             | Doc       | Hindi    | India     |             4 | NA      | P17       | 2023-01-01 |      9 |                 5 |
+---------+-----------------+-----------+----------+-----------+---------------+---------+-----------+------------+--------+-------------------+
7 rows in set (0.0030 sec)
```

Update basic information about Podcast

The podcast has some of the attributes in the media table and some of them in the podcast_record table. To update any attribute related to the podcast that respective table needs to be updated.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Media set media_name='MSN',genre='News',language='English',m_country='USA' where mediaid='P17';
Query OK, 1 row affected (0.0042 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Podcast set episode_count=8 where mediaid = 'P17';
Query OK, 1 row affected (0.0039 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update podcast_record set rating=8, total_subscribers=35 where (podcastid='P17' and pd_date='2023-01-01');
Query OK, 1 row affected (0.0038 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Media natural join Podcast join podcast_record on (Podcast.mediaid = podcast_record.podcastid);
+---------+-----------------+-----------+----------+-----------+---------------+---------+-----------+------------+--------+-------------------+
| mediaid | media_name      | genre     | language | m_country | episode_count | hostid  | podcastid | pd_date    | rating | total_subscribers |
+---------+-----------------+-----------+----------+-----------+---------------+---------+-----------+------------+--------+-------------------+
| P11     | Joe Rogan       | Interview | English  | USA       |            10 | PH11    | P11       | 2022-01-01 |      8 |                 2 |
| P11     | Joe Rogan       | Interview | English  | USA       |            10 | PH11    | P11       | 2022-02-01 |      7 |                 3 |
| P12     | Daily           | News      | English  | USA       |           100 | PH12    | P12       | 2022-01-01 |      8 |                 1 |
| P13     | Conan           | Comedy    | English  | USA       |            25 | PH11    | P13       | 2022-02-01 |      9 |                 4 |
| P14     | Sherlock Holmes | Fiction   | French   | France    |             5 | PH13    | P14       | 2022-01-01 |      6 |                 1 |
| P15     | Echoes of India | History   | Hindi    | India     |             1 | NA      | P15       | 2022-02-02 |      9 |                 5 |
| P17     | MSN             | News      | English  | USA       |             8 | NA      | P17       | 2023-01-01 |      8 |                35 |
+---------+-----------------+-----------+----------+-----------+---------------+---------+-----------+------------+--------+-------------------+
7 rows in set (0.0028 sec)
```

Delete basic information about Podcast

To delete any of the records from the podcast we need to delete it from the media table. On deletion from the media table the record will automatically be deleted from the podcast and podcast_record table.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL delete from Media where mediaid='P17';
Query OK, 1 row affected (0.0045 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Media natural join Podcast join podcast_record on (Podcast.mediaid = podcast_record.podcastid);
+---------+-----------------+-----------+----------+-----------+---------------+---------+-----------+------------+--------+-------------------+
| mediaid | media_name      | genre     | language | m_country | episode_count | hostid  | podcastid | pd_date    | rating | total_subscribers |
+---------+-----------------+-----------+----------+-----------+---------------+---------+-----------+------------+--------+-------------------+
| P11     | Joe Rogan       | Interview | English  | USA       |            10 | PH11    | P11       | 2022-01-01 |      8 |                 2 |
| P11     | Joe Rogan       | Interview | English  | USA       |            10 | PH11    | P11       | 2022-02-01 |      7 |                 3 |
| P12     | Daily           | News      | English  | USA       |           100 | PH12    | P12       | 2022-01-01 |      8 |                 1 |
| P13     | Conan           | Comedy    | English  | USA       |            25 | PH11    | P13       | 2022-02-01 |      9 |                 4 |
| P14     | Sherlock Holmes | Fiction   | French   | France    |             5 | PH13    | P14       | 2022-01-01 |      6 |                 1 |
| P15     | Echoes of India | History   | Hindi    | India     |             1 | NA      | P15       | 2022-02-02 |      9 |                 5 |
+---------+-----------------+-----------+----------+-----------+---------------+---------+-----------+------------+--------+-------------------+
6 rows in set (0.0029 sec)
```

Enter basic information about Podcast Host

To enter the information relating to the podcast host we first need to enter the record into the creators entity since the podcast host is in the isa hierarchy of the creators thus it is connected to the creators table and has common attribute creatorsid.

```
5 rows in set (0.0025 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into Creators values('CR20','Michael','Jackson');
Query OK, 1 row affected (0.0044 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into podcastHost values('CR20','mich@gmail.com',2001100111,'Texas');
Query OK, 1 row affected (0.0035 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Creators natural join podcastHost;
+------------+---------+---------+--------------------+------------+----------+
| creatorsid | cf_name | cl_name | email              | phone      | city     |
+------------+---------+---------+--------------------+------------+----------+
| CR20       | Michael | Jackson | mich@gmail.com     | 2001100111 | Texas    |
| PH11       | Joe     | Rogan   | jrogan@gmail.com   |       NULL | Raleigh  |
| PH12       | Steve   | Colbert | scolbert@gmail.com |       NULL | New York |
| PH13       | Conan   | Brien   | cbriend@gmail.com  | 2102376873 | NULL     |
| PH14       | Conan   | Dyle    | NULL               |       NULL | London   |
| PH15       | Harsha  | Bhogle  | hbhogle@gmail.com  | 2141874387 | Mumbai   |
+------------+---------+---------+--------------------+------------+----------+
6 rows in set (0.0027 sec)
```

Update basic information about Podcast Host

The podcast host has some of the attributes in the creators table and some of them in the podcast host table. To update any attribute related to the podcast host that respective table needs to be updated.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update podcastHost SET email='mjack@gmail.com',phone=NULL,city='Raleigh' where creatorsid='CR20';
Query OK, 1 row affected (0.0046 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcastHost;
+------------+--------------------+------------+----------+
| creatorsid | email              | phone      | city     |
+------------+--------------------+------------+----------+
| CR20       | mjack@gmail.com    |       NULL | Raleigh  |
| PH11       | jrogan@gmail.com   |       NULL | Raleigh  |
| PH12       | scolbert@gmail.com |       NULL | New York |
| PH13       | cbriend@gmail.com  | 2102376873 | NULL     |
| PH14       | NULL               |       NULL | London   |
| PH15       | hbhogle@gmail.com  | 2141874387 | Mumbai   |
+------------+--------------------+------------+----------+
6 rows in set (0.0022 sec)
```

Delete basic information about Podcast Host

To delete any of the records from the podcast host table we need to delete it from the creators table. On deletion from the creators table the record will automatically be deleted from the podcast host table.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL delete from Creators where creatorsid='CR20';
Query OK, 1 row affected (0.0037 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcastHost;
+------------+--------------------+------------+----------+
| creatorsid | email              | phone      | city     |
+------------+--------------------+------------+----------+
| PH11       | jrogan@gmail.com   |       NULL | Raleigh  |
| PH12       | scolbert@gmail.com |       NULL | New York |
| PH13       | cbriend@gmail.com  | 2102376873 | NULL     |
| PH14       | NULL               |       NULL | London   |
| PH15       | hbhogle@gmail.com  | 2141874387 | Mumbai   |
+------------+--------------------+------------+----------+
5 rows in set (0.0022 sec)
```

Enter basic information about Podcast Episode

The podcast_episode is divided into two different tables in order to maintain the data related to the listening count for every month. The basic information related to the podcast episode is maintained in the podcast_episode table so the new data can be inserted there.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into podcastEpisode values('P15',3,'Friends',34,'2023-01-05',10,2,2*2);
Query OK, 1 row affected (0.0054 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into podcastEpisode_listening values('P15',3,'2023-01-06',25);
Query OK, 1 row affected (0.0045 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcastEpisode natural join podcastEpisode_listening;
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+------------+-----------------+
| podcastid | episodeno | title           | p_duration | p_release_date | flat_fee | ad_count | bonus | pel_date   | listening_count |
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+------------+-----------------+
| P11       |         1 | North Carolina  |         21 | 2020-08-01     |       20 |        3 |     6 | 2023-02-01 |             100 |
| P11       |         1 | North Carolina  |         21 | 2020-08-01     |       20 |        3 |     6 | 2023-03-01 |             170 |
| P12       |         1 | ChatGPT         |         28 | 2023-02-03     |       10 |        1 |     2 | 2023-02-01 |              80 |
| P12       |         2 | modern          |         31 | 2023-03-04     |       20 |        4 |     8 | 2023-03-01 |              75 |
| P13       |         1 | Need a Friend   |         27 | 2023-03-04     |       35 |        2 |     4 | 2023-03-04 |               5 |
| P14       |         1 | Lying Detective |         40 | 2020-07-11     |       30 |        8 |    16 | 2023-02-01 |             120 |
| P14       |         2 | modern          |         41 | 2020-08-11     |       20 |        3 |     6 | 2023-03-01 |             150 |
| P15       |         1 | Mangalyaan      |         25 | 2022-05-05     |       80 |        1 |     2 | 2023-02-01 |             125 |
| P15       |         1 | Mangalyaan      |         25 | 2022-05-05     |       80 |        1 |     2 | 2023-03-01 |             250 |
| P15       |         3 | Friends         |         34 | 2023-01-05     |       10 |        2 |     4 | 2023-01-06 |              25 |
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+------------+-----------------+
10 rows in set (0.0036 sec)
```

Update basic information about Podcast Episode

Any information related to the podcast episode can be updated in the podcast_episode and podcastepisode_listening table with the change in their respective attributes.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update podcastEpisode set title='modern family' where podcastid='P15' and episodeno=3;
Query OK, 1 row affected (0.0046 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update podcastEpisode set p_duration=44 where podcastid='P15' and episodeno=3;
Query OK, 1 row affected (0.0047 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update podcastEpisode_listening set listening_count=40 where podcastid='P15' and episodeno=3;
Query OK, 1 row affected (0.0048 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcastEpisode natural join podcastEpisode_listening;
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+------------+-----------------+
| podcastid | episodeno | title           | p_duration | p_release_date | flat_fee | ad_count | bonus | pel_date   | listening_count |
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+------------+-----------------+
| P11       |         1 | North Carolina  |         21 | 2020-08-01     |       20 |        3 |     6 | 2023-02-01 |             100 |
| P11       |         1 | North Carolina  |         21 | 2020-08-01     |       20 |        3 |     6 | 2023-03-01 |             170 |
| P12       |         1 | ChatGPT         |         28 | 2023-02-03     |       10 |        1 |     2 | 2023-02-01 |              80 |
| P12       |         2 | modern          |         31 | 2023-03-04     |       20 |        4 |     8 | 2023-03-01 |              75 |
| P13       |         1 | Need a Friend   |         27 | 2023-03-04     |       35 |        2 |     4 | 2023-03-04 |               5 |
| P14       |         1 | Lying Detective |         40 | 2020-07-11     |       30 |        8 |    16 | 2023-02-01 |             120 |
| P14       |         2 | modern          |         41 | 2020-08-11     |       20 |        3 |     6 | 2023-03-01 |             150 |
| P15       |         1 | Mangalyaan      |         25 | 2022-05-05     |       80 |        1 |     2 | 2023-02-01 |             125 |
| P15       |         1 | Mangalyaan      |         25 | 2022-05-05     |       80 |        1 |     2 | 2023-03-01 |             250 |
| P15       |         3 | modern family   |         44 | 2023-01-05     |       10 |        2 |     4 | 2023-01-06 |              40 |
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+------------+-----------------+
```

Delete basic information about Podcast Episode

The record can be deleted from the podcastEpisode. This will automatically delete its respective data from the podcastEpisode_listening table.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL delete from podcastEpisode where podcastid='P15' and episodeno=3;
Query OK, 1 row affected (0.0049 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcastEpisode natural join podcastEpisode_listening;
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+------------+-----------------+
| podcastid | episodeno | title           | p_duration | p_release_date | flat_fee | ad_count | bonus | pel_date   | listening_count |
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+------------+-----------------+
| P11       |         1 | North Carolina  |         21 | 2020-08-01     |       20 |        3 |     6 | 2023-02-01 |             100 |
| P11       |         1 | North Carolina  |         21 | 2020-08-01     |       20 |        3 |     6 | 2023-03-01 |             170 |
| P12       |         1 | ChatGPT         |         28 | 2023-02-03     |       10 |        1 |     2 | 2023-02-01 |              80 |
| P12       |         2 | modern          |         31 | 2023-03-04     |       20 |        4 |     8 | 2023-03-01 |              75 |
| P13       |         1 | Need a Friend   |         27 | 2023-03-04     |       35 |        2 |     4 | 2023-03-04 |               5 |
| P14       |         1 | Lying Detective |         40 | 2020-07-11     |       30 |        8 |    16 | 2023-02-01 |             120 |
| P14       |         2 | modern          |         41 | 2020-08-11     |       20 |        3 |     6 | 2023-03-01 |             150 |
| P15       |         1 | Mangalyaan      |         25 | 2022-05-05     |       80 |        1 |     2 | 2023-02-01 |             125 |
| P15       |         1 | Mangalyaan      |         25 | 2022-05-05     |       80 |        1 |     2 | 2023-03-01 |             250 |
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+------------+-----------------+
9 rows in set (0.0036 sec)
```

Assign songs to album

Since we have assumed that every song can belong to at most one album and there can be songs which do not belong to any album it converts the relationship from songs to album into many one. Hence we have the albumid as an attribute in the song table. So whenever we need to assign the songs to the album we need to update the song table and assign the value for the attribute albumid.

```
18 rows in set (0.0025 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Song;
+---------+----------+----------------+--------------+--------------+---------+----------+
| mediaid | duration | s_release_date | royalty_rate | royalty_paid | albumid | track_no |
+---------+----------+----------------+--------------+--------------+---------+----------+
| M11     |      272 | 2022-02-15     |            4 |            0 | A11     |        1 |
| M12     |      305 | 2022-07-07     |            2 |            0 | A12     |        1 |
| M13     |      286 | 2022-03-04     |            1 |            1 | A12     |        2 |
| M14     |      182 | 2022-01-01     |            7 |            0 | NULL    |     NULL |
| M15     |      301 | 2016-04-05     |           12 |            1 | A13     |        1 |
| M16     |      320 | 2012-02-14     |            8 |            1 | A14     |        1 |
+---------+----------+----------------+--------------+--------------+---------+----------+
6 rows in set (0.0022 sec)
```

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Song SET albumid='A11',track_no=2 where mediaid='M14';
Query OK, 1 row affected (0.0040 sec)

Rows matched: 1  Changed: 1  Warnings: 0

MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Song;
+---------+----------+----------------+--------------+--------------+---------+----------+
| mediaid | duration | s_release_date | royalty_rate | royalty_paid | albumid | track_no |
+---------+----------+----------------+--------------+--------------+---------+----------+
| M11     |      272 | 2022-02-15     |            4 |            0 | A11     |        1 |
| M12     |      305 | 2022-07-07     |            2 |            0 | A12     |        1 |
| M13     |      286 | 2022-03-04     |            1 |            1 | A12     |        2 |
| M14     |      182 | 2022-01-01     |            7 |            0 | A11     |        2 |
| M15     |      301 | 2016-04-05     |           12 |            1 | A13     |        1 |
| M16     |      320 | 2012-02-14     |            8 |            1 | A14     |        1 |
+---------+----------+----------------+--------------+--------------+---------+----------+
6 rows in set (0.0027 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL
```

Assign artists to album

Based on our assumptions, the album belongs to only one artist. The artist can collaborate with different artists in the songs for that album. We have many to one relationship and hence the key artistid is present

in the album table. So we can update the album table and assign any album to the respective artist by updating the artistid.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Album;
+---------+------------+----------------+----------------------+----------+
| albumid | album_name | a_release_year | edition              | artistid |
+---------+------------+----------------+----------------------+----------+
| A11     | Bones      |           2022 | special              | NULL     |
| A12     | Gloria     |           2022 | limited              | NULL     |
| A13     | RaOne      |           2016 | collectors's edition | NULL     |
| A14     | AI2        |           2012 | special              | NULL     |
+---------+------------+----------------+----------------------+----------+
4 rows in set (0.0024 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Album SET artistid='CR11' where albumid='A13';
Query OK, 1 row affected (0.0034 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Album;
+---------+------------+----------------+----------------------+----------+
| albumid | album_name | a_release_year | edition              | artistid |
+---------+------------+----------------+----------------------+----------+
| A11     | Bones      |           2022 | special              | NULL     |
| A12     | Gloria     |           2022 | limited              | NULL     |
| A13     | RaOne      |           2016 | collectors's edition | CR11     |
| A14     | AI2        |           2012 | special              | NULL     |
+---------+------------+----------------+----------------------+----------+
4 rows in set (0.0023 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL
```

Assign artists to record label

Based on our assumptions, the artist is in contract with only one record label. Hence we have the labelname as an attribute in the artist table. So the artist can be assigned to any record label by updating the value for labelname in the artist table.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Artist;
+------------+------------------+----------+----------+---------------+------------------+-------------+
| creatorsid | labelname        | a_status | type     | primary_genre | monthly_listeners | a_country   |
+------------+------------------+----------+----------+---------------+------------------+-------------+
| CR11       | Lorimer Studios  | Active   | Band     | Bollywood     |               60 | India       |
| CR12       | Pluto            | Active   | Band     | Rock          |               10 | USA         |
| CR13       | LN 32 Music Group | Retired  | Musician | Lofi          |                0 | Switzerland |
| CR14       | LN 32 Music Group | Active   | Composer | Romance       |               13 | New Zealand |
| CR15       | MSeries          | Active   | Composer | Rock          |               17 | India       |
| CR16       | AV Productions   | Active   | Composer | Pop           |               63 | India       |
| CR17       | Pluto            | Active   | Composer | Romance       |               60 | USA         |
| CR18       | AV Productions   | Active   | Composer | Rock          |               17 | France      |
| CR19       | Pluto            | Retired  | Musician | Lofi          |               10 | USA         |
| CR20       | Unassigned       | active   | musician | lofi          |               20 | India       |
+------------+------------------+----------+----------+---------------+------------------+-------------+
10 rows in set (0.0029 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Artist set labelname='Pluto' where creatorsid = 'CR20';
Query OK, 1 row affected (0.0043 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Artist;
+------------+------------------+----------+----------+---------------+------------------+-------------+
| creatorsid | labelname        | a_status | type     | primary_genre | monthly_listeners | a_country   |
+------------+------------------+----------+----------+---------------+------------------+-------------+
| CR11       | Lorimer Studios  | Active   | Band     | Bollywood     |               60 | India       |
| CR12       | Pluto            | Active   | Band     | Rock          |               10 | USA         |
| CR13       | LN 32 Music Group | Retired  | Musician | Lofi          |                0 | Switzerland |
| CR14       | LN 32 Music Group | Active   | Composer | Romance       |               13 | New Zealand |
| CR15       | MSeries          | Active   | Composer | Rock          |               17 | India       |
| CR16       | AV Productions   | Active   | Composer | Pop           |               63 | India       |
| CR17       | Pluto            | Active   | Composer | Romance       |               60 | USA         |
| CR18       | AV Productions   | Active   | Composer | Rock          |               17 | France      |
| CR19       | Pluto            | Retired  | Musician | Lofi          |               10 | USA         |
| CR20       | Pluto            | active   | musician | lofi          |               20 | India       |
+------------+------------------+----------+----------+---------------+------------------+-------------+
10 rows in set (0.0024 sec)
```

Assign podcast episodes to podcast.

Within the podcast every podcast episode no is unique. So for assigning the podcast episode we first need to insert that record into the podcast episode table. Then that needs to be updated in the podcast table.

```
Rows matched: 5  Changed: 5  Warnings: 0
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select * from Podcast;
+----------+---------------+--------+
| mediaid  | episode_count | hostid |
+----------+---------------+--------+
| P11      |             1 | PH11   |
| P12      |             2 | PH12   |
| P13      |             1 | PH11   |
| P14      |             2 | PH13   |
| P15      |             2 | PH14   |
+----------+---------------+--------+
5 rows in set (0.0033 sec)
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select * from podcastEpisode;
+------------+------------+-----------------+------------+----------------+----------+----------+-------+
| podcastid  | episodeno  | title           | p_duration | p_release_date | flat_fee | ad_count | bonus |
+------------+------------+-----------------+------------+----------------+----------+----------+-------+
| P11        |          1 | North Carolina  |         21 | 2020-08-01     |       20 |        3 |     6 |
| P12        |          1 | ChatGPT         |         28 | 2023-02-03     |       10 |        1 |     2 |
| P12        |          2 | modern          |         31 | 2023-03-04     |       20 |        4 |     8 |
| P13        |          1 | Need a Friend   |         27 | 2023-03-04     |       35 |        2 |     4 |
| P14        |          1 | Lying Detective |         40 | 2020-07-11     |       30 |        8 |    16 |
| P14        |          2 | modern          |         41 | 2020-08-11     |       20 |        3 |     6 |
| P15        |          1 | Mangalyaan      |         25 | 2022-05-05     |       80 |        1 |     2 |
| P15        |          2 | Chandrayan      |         28 | 2023-01-25     |       40 |        3 |     6 |
+------------+------------+-----------------+------------+----------------+----------+----------+-------+
8 rows in set (0.0038 sec)
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  insert into podcastEpisode values ('P15',3,'Gaganyaan',26,'2023-01-28',25,3,3*2);
Query OK, 1 row affected (0.0046 sec)
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  update Podcast set episode_count = (select count(episodeno) from podcastEpisode where podcastEpisode.podcastid=Podcast.mediaid  group by podcastid);
Query OK, 1 row affected (0.0066 sec)

Rows matched: 5  Changed: 1  Warnings: 0
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select * from Podcast;
+----------+---------------+--------+
| mediaid  | episode_count | hostid |
+----------+---------------+--------+
| P11      |             1 | PH11   |
| P12      |             2 | PH12   |
| P13      |             1 | PH11   |
| P14      |             2 | PH13   |
| P15      |             3 | PH14   |
+----------+---------------+--------+
5 rows in set (0.0035 sec)
```

Assign podcast host to podcast.

Podcast host and podcasts are connected through table hostedby. Hence for assigning podcast host to podcasts the hostedby table needs to be updated.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from hostedby;
+------------+------------+--------+-------------+
| podcastid | episodeno | hostid | host_amount |
+------------+------------+--------+-------------+
| P11       |          1 | PH11   |          26 |
| P12       |          1 | PH12   |          12 |
| P12       |          2 | PH12   |          28 |
| P13       |          1 | PH13   |           0 |
| P14       |          1 | PH14   |          46 |
| P14       |          2 | PH14   |          26 |
| P15       |          1 | PH15   |          82 |
| P15       |          2 | PH12   |          46 |
+------------+------------+--------+-------------+
8 rows in set (0.0033 sec)

MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update hostedby set hostid='PH13' where podcastid='P13';
Query OK, 1 row affected (0.0047 sec)

Rows matched: 1  Changed: 1  Warnings: 0

MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update hostedby set host_amount = (select (flat_fee + bonus) from podcastEpisode where podcastid = 'P13') where podcastid = 'P13';
Query OK, 1 row affected (0.0049 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from hostedby;
+-----------+-----------+--------+-------------+
| podcastid | episodeno | hostid | host_amount |
+-----------+-----------+--------+-------------+
| P11       |         1 | PH11   |          26 |
| P12       |         1 | PH12   |          12 |
| P12       |         2 | PH12   |          28 |
| P13       |         1 | PH13   |          39 |
| P14       |         1 | PH14   |          46 |
| P14       |         2 | PH14   |          26 |
| P15       |         1 | PH15   |          82 |
| P15       |         2 | PH12   |          46 |
+-----------+-----------+--------+-------------+
8 rows in set (0.0036 sec)
```

TASK 2:- Maintaining metadata and records:

Enter playcount for Songs

For calculating the playcount we have collected the data from the listensto table. The listensto table has the data regarding the user listening to media and also has the timestamp for it. The playcount for a particular song is updated every month by counting the user listening to that particular media in the listensto table. Hence the playcount for that songs are entered into the listensto table.

```
Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Song;
+---------+----------+----------------+--------------+--------------+---------+----------+--------------+
| mediaid | duration | s_release_date | royalty_rate | royalty_paid | albumid | track_no | splay_count |
+---------+----------+----------------+--------------+--------------+---------+----------+--------------+
| M11     |      272 | 2022-02-15     |            4 |            0 | A11     |        1 |            5 |
| M12     |      305 | 2022-07-07     |            2 |            0 | A12     |        1 |            3 |
| M13     |      286 | 2022-03-04     |            1 |            1 | A12     |        2 |            5 |
| M14     |      182 | 2022-01-01     |            7 |            0 | A11     |        2 |            0 |
| M15     |      301 | 2016-04-05     |           12 |            1 | A13     |        1 |            3 |
| M16     |      320 | 2012-02-14     |            8 |            1 | A14     |        1 |            7 |
+---------+----------+----------------+--------------+--------------+---------+----------+--------------+
6 rows in set (0.0022 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into listensto values (1111111114,'M14','2023-02-07',4);
Query OK, 1 row affected (0.0040 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into listensto values (1111111115,'M14','2023-02-04',6);
Query OK, 1 row affected (0.0035 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into listensto values (1111111117,'M14','2023-02-08',2);
Query OK, 1 row affected (0.0043 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from listensto;
+------------+---------+------------+-------------+
| uphone     | mediaid | uplay_date | dplay_count |
+------------+---------+------------+-------------+
| 1111111111 | M11     | 2023-02-01 |           2 |
| 1111111111 | M13     | 2023-02-02 |           1 |
| 1111111111 | M15     | 2023-02-01 |           3 |
| 1111111112 | M11     | 2023-02-05 |           3 |
| 1111111112 | P13     | 2023-02-05 |           1 |
| 1111111112 | P15     | 2023-02-05 |           1 |
| 1111111114 | M14     | 2023-02-07 |           4 |
| 1111111115 | M14     | 2023-02-04 |           6 |
| 1111111117 | M14     | 2023-02-08 |           2 |
+------------+---------+------------+-------------+
9 rows in set (0.0022 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL
```

Update playcount for Songs

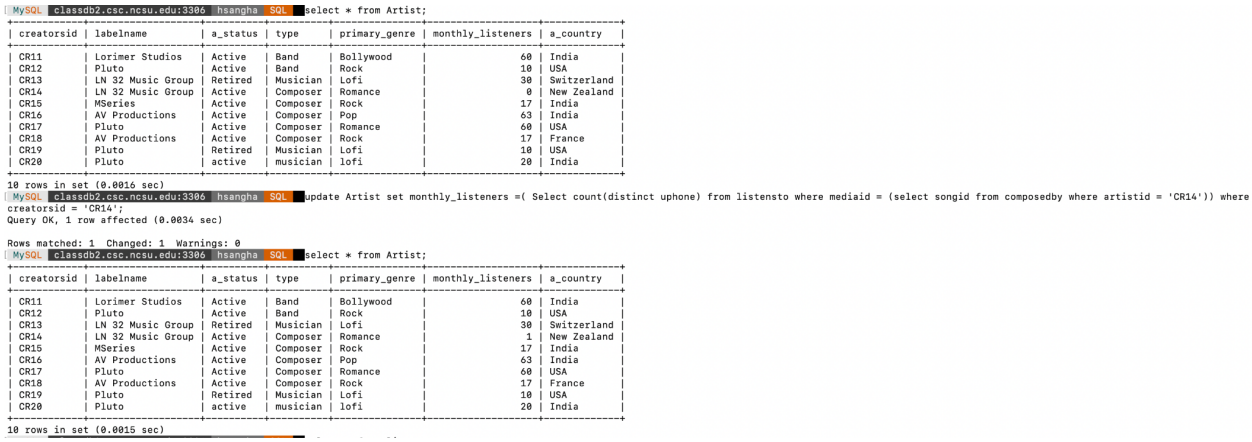
The monthly playcounts are updated in the songs table on the basis of its data from the listensto table.

```
9 rows in set (0.0022 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Song SET splay_count = (select sum(dplay_count) from listensto where  mediaid='M14' group by(month(uplay_date))) where mediaid='M14';
Query OK, 1 row affected (0.0045 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Song;
+---------+----------+----------------+--------------+--------------+---------+----------+--------------+
| mediaid | duration | s_release_date | royalty_rate | royalty_paid | albumid | track_no | splay_count |
+---------+----------+----------------+--------------+--------------+---------+----------+--------------+
| M11     |      272 | 2022-02-15     |            4 |            0 | A11     |        1 |            5 |
| M12     |      305 | 2022-07-07     |            2 |            0 | A12     |        1 |            3 |
| M13     |      286 | 2022-03-04     |            1 |            1 | A12     |        2 |            5 |
| M14     |      182 | 2022-01-01     |            7 |            0 | A11     |        2 |           12 |
| M15     |      301 | 2016-04-05     |           12 |            1 | A13     |        1 |            3 |
| M16     |      320 | 2012-02-14     |            8 |            1 | A14     |        1 |            7 |
+---------+----------+----------------+--------------+--------------+---------+----------+--------------+
6 rows in set (0.0023 sec)
```

Update count of monthly listener's count for Artist

The monthly listeners count is determined by the count of the unique active user listening to the all the media of the given artist. For this three relations are used one is listens to which has the data for the user listening to the media. The composedby relation has the record for the different songs created by the artist and then the montly_listeners are updated in the artist table.



```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Artist;
+------------+-----------------+----------+----------+---------------+------------------+-------------+
| creatorsid | labelname       | a_status | type     | primary_genre | monthly_listeners | a_country   |
+------------+-----------------+----------+----------+---------------+------------------+-------------+
| CR11       | Lorimer Studios | Active   | Band     | Bollywood     |               60 | India       |
| CR12       | Pluto           | Active   | Band     | Rock          |               10 | USA         |
| CR13       | LN 32 Music Group | Retired | Musician | Lofi          |               30 | Switzerland |
| CR14       | LN 32 Music Group | Active   | Composer | Romance       |                0 | New Zealand |
| CR15       | MSeries         | Active   | Composer | Rock          |               17 | India       |
| CR16       | AV Productions  | Active   | Composer | Pop           |               63 | India       |
| CR17       | Pluto           | Active   | Composer | Romance       |               60 | USA         |
| CR18       | AV Productions  | Active   | Composer | Rock          |               17 | France      |
| CR19       | Pluto           | Retired  | Musician | Lofi          |               10 | USA         |
| CR20       | Pluto           | active   | musician | lofi          |               20 | India       |
+------------+-----------------+----------+----------+---------------+------------------+-------------+
10 rows in set (0.0016 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update Artist set monthly_listeners =( Select count(distinct uphone) from listensto where mediaid = (select songid from composedby where artistid = 'CR14')) where
creatorsid = 'CR14';
Query OK, 1 row affected (0.0034 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from Artist;
+------------+-----------------+----------+----------+---------------+------------------+-------------+
| creatorsid | labelname       | a_status | type     | primary_genre | monthly_listeners | a_country   |
+------------+-----------------+----------+----------+---------------+------------------+-------------+
| CR11       | Lorimer Studios | Active   | Band     | Bollywood     |               60 | India       |
| CR12       | Pluto           | Active   | Band     | Rock          |               10 | USA         |
| CR13       | LN 32 Music Group | Retired | Musician | Lofi          |               30 | Switzerland |
| CR14       | LN 32 Music Group | Active   | Composer | Romance       |                1 | New Zealand |
| CR15       | MSeries         | Active   | Composer | Rock          |               17 | India       |
| CR16       | AV Productions  | Active   | Composer | Pop           |               63 | India       |
| CR17       | Pluto           | Active   | Composer | Romance       |               60 | USA         |
| CR18       | AV Productions  | Active   | Composer | Rock          |               17 | France      |
| CR19       | Pluto           | Retired  | Musician | Lofi          |               10 | USA         |
| CR20       | Pluto           | active   | musician | lofi          |               20 | India       |
+------------+-----------------+----------+----------+---------------+------------------+-------------+
10 rows in set (0.0015 sec)
```

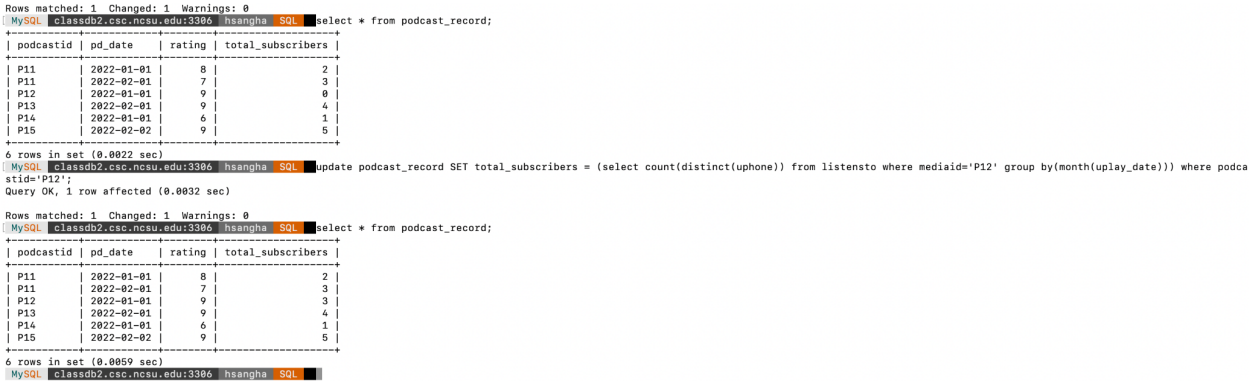
Enter total subscribers for Podcasts

To maintain the data for total subscribers we maintain the listens to table where the records are updated.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into listensto values (1111111113,'P12','2023-02-01',2);
Query OK, 1 row affected (0.0034 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into listensto values (1111111114,'P12','2023-02-04',1);
Query OK, 1 row affected (0.0032 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into listensto values (1111111111,'P12','2023-02-07',3);
Query OK, 1 row affected (0.0034 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from listensto;
+------------+---------+------------+-------------+
| uphone     | mediaid | uplay_date | dplay_count |
+------------+---------+------------+-------------+
| 1111111111 | M11     | 2023-02-01 |           2 |
| 1111111111 | M13     | 2023-02-02 |           1 |
| 1111111111 | M15     | 2023-02-01 |           3 |
| 1111111111 | P12     | 2023-02-07 |           3 |
| 1111111112 | M11     | 2023-02-05 |           3 |
| 1111111112 | P13     | 2023-02-05 |           1 |
| 1111111112 | P15     | 2023-02-05 |           1 |
| 1111111113 | P12     | 2023-02-01 |           2 |
| 1111111114 | M14     | 2023-02-07 |           4 |
| 1111111114 | P12     | 2023-02-04 |           1 |
| 1111111115 | M14     | 2023-02-04 |           6 |
| 1111111117 | M14     | 2023-02-08 |           2 |
| 1111111117 | P15     | 2023-02-07 |           5 |
+------------+---------+------------+-------------+
13 rows in set (0.0027 sec)
```

Update total subscribers for Podcasts

The listensto table is used to maintain the record of the total subscribers. It is then updated in the podcast_record table.



Update rating for Podcasts

The rating is updated in the podcast_record table based on the various different reviews.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcast_record;
+------------+------------+--------+--------------------+
| podcastid | pd_date    | rating | total_subscribers |
+------------+------------+--------+--------------------+
| P11        | 2022-01-01 |      8 |                  2 |
| P11        | 2022-02-01 |      7 |                  3 |
| P12        | 2022-01-01 |      8 |                  6 |
| P13        | 2022-02-01 |      9 |                  4 |
| P14        | 2022-01-01 |      6 |                  1 |
| P15        | 2022-02-02 |      9 |                  5 |
+------------+------------+--------+--------------------+
6 rows in set (0.0022 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update podcast_record SET rating =9 where podcastid='P12';
Query OK, 1 row affected (0.0110 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcast_record;
+------------+------------+--------+--------------------+
| podcastid | pd_date    | rating | total_subscribers |
+------------+------------+--------+--------------------+
| P11        | 2022-01-01 |      8 |                  2 |
| P11        | 2022-02-01 |      7 |                  3 |
| P12        | 2022-01-01 |      9 |                  6 |
| P13        | 2022-02-01 |      9 |                  4 |
| P14        | 2022-01-01 |      6 |                  1 |
| P15        | 2022-02-02 |      9 |                  5 |
+------------+------------+--------+--------------------+
6 rows in set (0.0023 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL
```

Enter the listening count for Podcast Episodes

The listening count for the podcast episodes are updated in the podcast episodes table.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcastEpisode_listening;
+-----------+-----------+------------+-----------------+
| podcastid | episodeno | pel_date   | listening_count |
+-----------+-----------+------------+-----------------+
| P11       |         1 | 2023-02-01 |             100 |
| P11       |         1 | 2023-03-01 |             170 |
| P12       |         1 | 2023-02-01 |              80 |
| P12       |         2 | 2023-03-01 |              75 |
| P14       |         1 | 2023-02-01 |             120 |
| P14       |         2 | 2023-03-01 |             150 |
| P15       |         1 | 2023-02-01 |             125 |
| P15       |         1 | 2023-03-01 |             250 |
+-----------+-----------+------------+-----------------+
8 rows in set (0.0035 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcastEpisode;
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+
| podcastid | episodeno | title           | p_duration | p_release_date | flat_fee | ad_count | bonus |
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+
| P11       |         1 | North Carolina  |         21 | 2020-08-01     |       20 |        3 |     6 |
| P12       |         1 | ChatGPT         |         28 | 2023-02-03     |       10 |        1 |     2 |
| P12       |         2 | modern          |         31 | 2023-03-04     |       20 |        4 |     8 |
| P13       |         1 | Need a Friend   |         27 | 2023-03-04     |       35 |        2 |     4 |
| P14       |         1 | Lying Detective |         40 | 2020-07-11     |       30 |        8 |    16 |
| P14       |         2 | modern          |         41 | 2020-08-11     |       20 |        3 |     6 |
| P15       |         1 | Mangalyaan      |         25 | 2022-05-05     |       80 |        1 |     2 |
| P15       |         2 | Chandrayan      |         28 | 2023-01-25     |       40 |        3 |     6 |
| P15       |         3 | Gaganyaan       |         26 | 2023-01-28     |       25 |        3 |     6 |
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+
9 rows in set (0.0035 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL insert into podcastEpisode_listening values('P13',1,'2023-04-01',200);
Query OK, 1 row affected (0.0047 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcastEpisode_listening;
+-----------+-----------+------------+-----------------+
| podcastid | episodeno | pel_date   | listening_count |
+-----------+-----------+------------+-----------------+
| P11       |         1 | 2023-02-01 |             100 |
| P11       |         1 | 2023-03-01 |             170 |
| P12       |         1 | 2023-02-01 |              80 |
| P12       |         2 | 2023-03-01 |              75 |
| P13       |         1 | 2023-04-01 |             200 |
| P14       |         1 | 2023-02-01 |             120 |
| P14       |         2 | 2023-03-01 |             150 |
| P15       |         1 | 2023-02-01 |             125 |
| P15       |         1 | 2023-03-01 |             250 |
+-----------+-----------+------------+-----------------+
9 rows in set (0.0033 sec)
```

Find songs given artist

The information related to which song is sung by which artist is maintained in the composedby table. Hence the song is joined with that table along with the artist table to obtain the desired output.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select media_name, cf_name, cl_name from Media natural join Song
join composedby on mediaid=songid join Artist on artistid=creatorsid natural join Creators where creatorsid='CR15';
+---------------+---------+---------+
| media_name    | cf_name | cl_name |
+---------------+---------+---------+
| Chammak Challo | Mitul   | Patel   |
+---------------+---------+---------+
1 row in set (0.0019 sec)
```

Find songs given album

The song and album can be combined on the albumid since we have assumed that every song belongs to at most one album.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select media_name, album_name from Media natural join Song natura
l join Album where albumid='A13';
+----------------+------------+
| media_name     | album_name |
+----------------+------------+
| Chammak Challo | RaOne      |
+----------------+------------+
1 row in set (0.0016 sec)
```

Find songs given artist and album

The song has the information of the album in the song table itself. Hence we only need to join artist and song along with creators and media in order to get the information needed.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select media_name, album_name, cf_name, cl_name from Media natural join
 Song natural join Album join Artist on artistid=creatorsid natural join Creators where albumid='A14' and creatorsid='CR16
';
+------------+------------+---------+---------+
| media_name | album_name | cf_name | cl_name |
+------------+------------+---------+---------+
| Toota hua  | AI2        | Artharv | Pandit  |
+------------+------------+---------+---------+
1 row in set (0.0020 sec)
```

Find podcast episodes given podcast

The information for the podcastepisode can be obtained from the podcastid in the podcastepisode table.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select title, episodeno, media_name from Media natural join Podca
st join podcastEpisode on mediaid=podcastid where mediaid='P11';
+----------------+-----------+------------+
| title          | episodeno | media_name |
+----------------+-----------+------------+
| North Carolina |         1 | Joe Rogan  |
+----------------+-----------+------------+
1 row in set (0.0016 sec)
```

TASK 3:- Maintaining Payments:

Receive payment from subscribers

The payments are received from the users and are stored in the user_payment table. The user can pay for various different months but the paymentid will be generated every time the user pays. Hence the paymentid is unique and is associated with only one user but users can have many paymentids.

```
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select * from user natural join user_payment;
+------------+------------+------------+----------+----------+------------------------+----------+---------+------+------------+
| uphone     | join_date  | end_date   | uf_name  | ul_name  | uemail                 | u_status | sub_fee | uid  | ups_date   |
+------------+------------+------------+----------+----------+------------------------+----------+---------+------+------------+
| 1111111114 | 2023-01-11 | NULL       | Vishwa   | Gandhi   | vgandhi@lexis.com      | Active   |      15 | US11 | 2023-01-01 |
| 1111111114 | 2023-01-11 | NULL       | Vishwa   | Gandhi   | vgandhi@lexis.com      | Active   |      15 | US12 | 2023-02-01 |
| 1111111111 | 2023-03-06 | NULL       | Joey     | Tribiani | joey@gmail.com         | Active   |      15 | US13 | 2023-01-01 |
| 1111111116 | 2023-01-02 | 2023-02-01 | Jethalal | Gada     | jgada@electronics.com  | Inactive |       0 | US14 | 2023-01-01 |
+------------+------------+------------+----------+----------+------------------------+----------+---------+------+------------+
4 rows in set (0.0016 sec)
```

Generate monthly royalties for songs

Royalties are calculated based on the play count. Firstly the status for royalty_paid is checked and if the status is 0 then the royalties are generated based on the various different tables.

```
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select mediaid,media_name,creatorsid,cf_name,royalty_rate,royalty_paid,splay_count,labelname,royalty_rate*splay_count AS total_amount from Media natural
join Song join composedby on mediaid=songid join Artist on artistid=creatorsid natural join Creators where lead='Yes' AND royalty_paid=0;
+---------+-----------------+------------+---------+--------------+--------------+--------------+-----------------+--------------+
| mediaid | media_name      | creatorsid | cf_name | royalty_rate | royalty_paid | splay_count | labelname       | total_amount |
+---------+-----------------+------------+---------+--------------+--------------+--------------+-----------------+--------------+
| M11     | Bones           | CR12       | Imagine |            4 |            0 |            5 | Pluto           |           20 |
| M12     | Unholy          | CR13       | Sam     |            2 |            0 |            3 | LN 32 Music Group |            6 |
| M14     | Waking Up Dreaming | CR12    | Imagine |            7 |            0 |           12 | Pluto           |           84 |
+---------+-----------------+------------+---------+--------------+--------------+--------------+-----------------+--------------+
3 rows in set (0.0044 sec)
```

Monthly Record for Record label payment per month

For maintaining the label_payment we have added the column for main_label which shows whether the song which is being recorded by multiple artists has a label that owns it. Royalties are only paid to the label of the song which are the owners of it.

```
Records: 10  Duplicates: 0  Warnings: 0
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select * from songPayment;
+---------+--------+-----------+-----------------+---------------+-----------------+------------+--------------+
| spayid  | songid | artistid  | labelname       | label_payment | artist_payment  | spay_date  | Main_label   |
+---------+--------+-----------+-----------------+---------------+-----------------+------------+--------------+
|      17 | M11    | CR12      | Pluto           |            20 |         4.66667 | 2023-02-01 | Yes          |
|      18 | M14    | CR12      | Pluto           |            84 |            29.4 | 2023-02-01 | Yes          |
|      19 | M12    | CR13      | LN 32 Music Group |             6 |             4.2 | 2023-02-01 | Yes          |
|      20 | M14    | CR17      | Pluto           |            84 |            29.4 | 2023-02-01 | No           |
|      21 | M11    | CR18      | AV Productions  |            20 |         4.66667 | 2023-02-01 | No           |
|      22 | M11    | CR19      | Pluto           |            20 |         4.66667 | 2023-02-01 | No           |
|      24 | M15    | CR11      | Lorimer Studios |            48 |            16.8 | 2023-03-01 | Yes          |
|      25 | M11    | CR12      | Pluto           |            24 |             5.6 | 2023-03-01 | Yes          |
|      26 | M14    | CR12      | Pluto           |            98 |            34.3 | 2023-03-01 | Yes          |
|      27 | M12    | CR13      | LN 32 Music Group |             8 |             5.6 | 2023-03-01 | Yes          |
|      28 | M13    | CR14      | LN 32 Music Group |             7 |             4.9 | 2023-03-01 | Yes          |
|      29 | M15    | CR15      | MSeries         |            48 |            16.8 | 2023-03-01 | No           |
|      30 | M16    | CR16      | AV Productions  |            72 |            50.4 | 2023-03-01 | Yes          |
|      31 | M14    | CR17      | Pluto           |            98 |            34.3 | 2023-03-01 | No           |
|      32 | M11    | CR18      | AV Productions  |            24 |             5.6 | 2023-03-01 | No           |
|      33 | M11    | CR19      | Pluto           |            24 |             5.6 | 2023-03-01 | No           |
+---------+--------+-----------+-----------------+---------------+-----------------+------------+--------------+
16 rows in set (0.0015 sec)
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL  select labelname, sum(label_payment) from songPayment where Main_label='Yes' AND month(spay_date)=3 group by labelname;
+-------------------+---------------------+
| labelname         | sum(label_payment) |
+-------------------+---------------------+
| AV Productions    |                  72 |
| LN 32 Music Group |                  15 |
| Lorimer Studios   |                  48 |
| Pluto             |                 122 |
+-------------------+---------------------+
4 rows in set (0.0020 sec)
MySQL  classdb2.csc.ncsu.edu:3306  hsangha  SQL
```

Monthly record for Artist payment

The monthly record for the artist payment is maintained in the song payment table itself.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from songPayment;
+---------+--------+----------+-------------------+---------------+----------------+------------+------------+
| spayid | songid | artistid | labelname         | label_payment | artist_payment | spay_date  | Main_label |
+---------+--------+----------+-------------------+---------------+----------------+------------+------------+
|      17 | M11    | CR12     | Pluto             |            20 |        4.66667 | 2023-02-01 | Yes        |
|      18 | M14    | CR12     | Pluto             |            84 |           29.4 | 2023-02-01 | Yes        |
|      19 | M12    | CR13     | LN 32 Music Group |             6 |            4.2 | 2023-02-01 | Yes        |
|      20 | M14    | CR17     | Pluto             |            84 |           29.4 | 2023-02-01 | No         |
|      21 | M11    | CR18     | AV Productions    |            20 |        4.66667 | 2023-02-01 | No         |
|      22 | M11    | CR19     | Pluto             |            20 |        4.66667 | 2023-02-01 | No         |
|      24 | M15    | CR11     | Lorimer Studios   |            48 |           16.8 | 2023-03-01 | Yes        |
|      25 | M11    | CR12     | Pluto             |            24 |            5.6 | 2023-03-01 | Yes        |
|      26 | M14    | CR12     | Pluto             |            98 |           34.3 | 2023-03-01 | Yes        |
|      27 | M12    | CR13     | LN 32 Music Group |             8 |            5.6 | 2023-03-01 | Yes        |
|      28 | M13    | CR14     | LN 32 Music Group |             7 |            4.9 | 2023-03-01 | Yes        |
|      29 | M15    | CR15     | MSeries           |            48 |           16.8 | 2023-03-01 | No         |
|      30 | M16    | CR16     | AV Productions    |            72 |           50.4 | 2023-03-01 | Yes        |
|      31 | M14    | CR17     | Pluto             |            98 |           34.3 | 2023-03-01 | No         |
|      32 | M11    | CR18     | AV Productions    |            24 |            5.6 | 2023-03-01 | No         |
|      33 | M11    | CR19     | Pluto             |            24 |            5.6 | 2023-03-01 | No         |
+---------+--------+----------+-------------------+---------------+----------------+------------+------------+
16 rows in set (0.0014 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select artistid,sum(artist_payment) from songPayment where month(spay_date)=3 group by artistid;
+----------+---------------------+
| artistid | sum(artist_payment) |
+----------+---------------------+
| CR11     |  16.799999237060547 |
| CR12     |  39.899999141693115 |
| CR13     |   5.599999904632568 |
| CR14     |   4.900000095367432 |
| CR15     |  16.799999237060547 |
| CR16     |  50.400001525878906 |
| CR17     |   34.29999923706055 |
| CR18     |   5.599999904632568 |
| CR19     |   5.599999904632568 |
+----------+---------------------+
9 rows in set (0.0015 sec)
```

Make payment to podcast hosts

Based on our assumptions for every ad_count podcast host receives a bonus of 2 dollars. The payment is made to the podcast hosts by adding the flat_fee and bonus for every released episode.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from podcastEpisode;;
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+
| podcastid | episodeno | title           | p_duration | p_release_date | flat_fee | ad_count | bonus |
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+
| P11       |         1 | North Carolina  |         21 | 2020-08-01     |       20 |        3 |     6 |
| P12       |         1 | ChatGPT         |         28 | 2023-02-03     |       10 |        1 |     2 |
| P12       |         2 | modern          |         31 | 2023-03-04     |       20 |        4 |     8 |
| P13       |         1 | Need a Friend   |         27 | 2023-03-04     |       35 |        2 |     4 |
| P14       |         1 | Lying Detective |         40 | 2020-07-11     |       30 |        8 |    16 |
| P14       |         2 | modern          |         41 | 2020-08-11     |       20 |        3 |     6 |
| P15       |         1 | Mangalyaan      |         25 | 2022-05-05     |       80 |        1 |     2 |
| P15       |         2 | Chandrayan      |         28 | 2023-01-25     |       40 |        3 |     6 |
| P15       |         3 | Gaganyaan       |         26 | 2023-01-28     |       25 |        3 |     6 |
+-----------+-----------+-----------------+------------+----------------+----------+----------+-------+
9 rows in set (0.0033 sec)
ERROR: 1065 (42000): Query was empty
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from hostedby;
+-----------+-----------+--------+-------------+
| podcastid | episodeno | hostid | host_amount |
+-----------+-----------+--------+-------------+
| P11       |         1 | PH11   |          26 |
| P12       |         1 | PH12   |          12 |
| P12       |         2 | PH12   |          28 |
| P13       |         1 | PH13   |          39 |
| P14       |         1 | PH14   |          46 |
| P14       |         2 | PH14   |          26 |
| P15       |         1 | PH15   |          82 |
| P15       |         2 | PH12   |           0 |
+-----------+-----------+--------+-------------+
8 rows in set (0.0033 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL update hostedby SET host_amount = ( select flat_fee + bonus from podcastEpisode where podcastid='P15' AND episodeno=2)  where podcastid='P15' AND episode
no=2;
Query OK, 1 row affected (0.0045 sec)

Rows matched: 1  Changed: 1  Warnings: 0
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from hostedby;
+-----------+-----------+--------+-------------+
| podcastid | episodeno | hostid | host_amount |
+-----------+-----------+--------+-------------+
| P11       |         1 | PH11   |          26 |
| P12       |         1 | PH12   |          12 |
| P12       |         2 | PH12   |          28 |
| P13       |         1 | PH13   |          39 |
| P14       |         1 | PH14   |          46 |
| P14       |         2 | PH14   |          26 |
| P15       |         1 | PH15   |          82 |
| P15       |         2 | PH12   |          46 |
+-----------+-----------+--------+-------------+
8 rows in set (0.0033 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL
```

TASK 4. Reports:-

Monthly play count per song

The report can be generated by selecting the play counts column in the song table.

```
6 rows in set (0.0013 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select mediaid,media_name,splay_count from Media natural join Song;
+---------+---------------------+--------------+
| mediaid | media_name          | splay_count |
+---------+---------------------+--------------+
| M11     | Bones               |            5 |
| M12     | Unholy              |            3 |
| M13     | Heaven              |            5 |
| M14     | Waking Up Dreaming  |           12 |
| M15     | Chammak Challo      |            3 |
| M16     | Toota hua           |            7 |
+---------+---------------------+--------------+
6 rows in set (0.0021 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL
```

Monthly play count per album

The monthly play count for the album can be obtained by adding the play counts of the all songs of the album for that particular month.

```
4 rows in set (0.0014 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select albumid,album_name,sum(splay_count) from Song natural join Album group by (albumid);
+---------+------------+-------------------+
| albumid | album_name | sum(splay_count) |
+---------+------------+-------------------+
| A11     | Bones      |                17 |
| A12     | Gloria     |                 8 |
| A13     | RaOne      |                 3 |
| A14     | AI2        |                 7 |
+---------+------------+-------------------+
4 rows in set (0.0018 sec)
```

Monthly play count per artist.

The monthly play count per artist is updated by calculating the total play counts for all the songs of that artist for the given month.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select creatorsid,cf_name,sum(splay_count) from Media natural join Song join composedby on mediaid=songid join Artist on artistid=creatorsid natural join
Creators group by creatorsid;
+------------+---------+-------------------+
| creatorsid | cf_name | sum(splay_count) |
+------------+---------+-------------------+
| CR11       | Ravi    |                 3 |
| CR12       | Imagine |                17 |
| CR13       | Sam     |                 3 |
| CR14       | Kim     |                 5 |
| CR15       | Mitul   |                 3 |
| CR16       | Artharv |                 7 |
| CR17       | John    |                12 |
| CR18       | Dan     |                 5 |
| CR19       | Julia   |                 5 |
+------------+---------+-------------------+
9 rows in set (0.0020 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL
```

Calculate total payments made out to the host per a given time period.a

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from hostedby natural join podcastEpisode;
+-----------+-----------+--------+-------------+-----------------+------------+----------------+----------+----------+-------+
| podcastid | episodeno | hostid | host_amount | title           | p_duration | p_release_date | flat_fee | ad_count | bonus |
+-----------+-----------+--------+-------------+-----------------+------------+----------------+----------+----------+-------+
| P11       |         1 | PH11   |          26 | North Carolina  |         21 | 2020-08-01     |       20 |        3 |     6 |
| P12       |         1 | PH12   |          12 | ChatGPT         |         28 | 2023-02-03     |       10 |        1 |     2 |
| P12       |         2 | PH12   |          28 | modern          |         31 | 2023-03-04     |       20 |        4 |     8 |
| P13       |         1 | PH13   |          39 | Need a Friend   |         27 | 2023-03-04     |       35 |        2 |     4 |
| P14       |         1 | PH14   |          46 | Lying Detective |         40 | 2020-07-11     |       30 |        8 |    16 |
| P14       |         2 | PH14   |          26 | modern          |         41 | 2020-08-11     |       20 |        3 |     6 |
| P15       |         1 | PH15   |          82 | Mangalyaan      |         25 | 2022-05-05     |       80 |        1 |     2 |
| P15       |         2 | PH12   |          46 | Chandrayan      |         28 | 2023-01-25     |       40 |        3 |     6 |
+-----------+-----------+--------+-------------+-----------------+------------+----------------+----------+----------+-------+
8 rows in set (0.0023 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select hostid,sum(host_amount),p_release_date from hostedby natural join podcastEpisode where hostid='PH14' group by podcastid,episodeno;
+--------+------------------+----------------+
| hostid | sum(host_amount) | p_release_date |
+--------+------------------+----------------+
| PH14   |               46 | 2020-07-11     |
| PH14   |               26 | 2020-08-11     |
+--------+------------------+----------------+
2 rows in set (0.0016 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select hostid,sum(host_amount),p_release_date from hostedby natural join podcastEpisode where hostid='PH14' AND p_release_date BETWEEN CAST('2020-07-01'
AS DATE) AND CAST('2020-08-01' AS DATE) group by podcastid,episodeno;
+--------+------------------+----------------+
| hostid | sum(host_amount) | p_release_date |
+--------+------------------+----------------+
| PH14   |               46 | 2020-07-11     |
+--------+------------------+----------------+
1 row in set (0.0018 sec)
```

Calculate total payments made out to the artist per a given time period.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from songPayment;
+---------+--------+----------+---------------+---------------+----------------+------------+------------+
| spayid  | songid | artistid | labelname     | label_payment | artist_payment | spay_date  | Main_label |
+---------+--------+----------+---------------+---------------+----------------+------------+------------+
|      17 | M11    | CR12     | Pluto         |            20 |        4.66667 | 2023-02-01 | Yes        |
|      18 | M14    | CR12     | Pluto         |            84 |           29.4 | 2023-02-01 | Yes        |
|      19 | M12    | CR13     | LN 32 Music Group |             6 |            4.2 | 2023-02-01 | Yes        |
|      20 | M14    | CR17     | Pluto         |            84 |           29.4 | 2023-02-01 | No         |
|      21 | M11    | CR18     | AV Productions |            20 |        4.66667 | 2023-02-01 | No         |
|      22 | M11    | CR19     | Pluto         |            20 |        4.66667 | 2023-02-01 | No         |
|      24 | M15    | CR11     | Lorimer Studios |            48 |           16.8 | 2023-03-01 | Yes        |
|      25 | M11    | CR12     | Pluto         |            24 |            5.6 | 2023-03-01 | Yes        |
|      26 | M14    | CR12     | Pluto         |            98 |           34.3 | 2023-03-01 | Yes        |
|      27 | M12    | CR13     | LN 32 Music Group |             8 |            5.6 | 2023-03-01 | Yes        |
|      28 | M13    | CR14     | LN 32 Music Group |             7 |            4.9 | 2023-03-01 | Yes        |
|      29 | M15    | CR15     | MSeries       |            48 |           16.8 | 2023-03-01 | No         |
|      30 | M16    | CR16     | AV Productions |            72 |           50.4 | 2023-03-01 | Yes        |
|      31 | M14    | CR17     | Pluto         |            98 |           34.3 | 2023-03-01 | No         |
|      32 | M11    | CR18     | AV Productions |            24 |            5.6 | 2023-03-01 | No         |
|      33 | M11    | CR19     | Pluto         |            24 |            5.6 | 2023-03-01 | No         |
+---------+--------+----------+---------------+---------------+----------------+------------+------------+
16 rows in set (0.0015 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select artistid,sum(artist_payment) from songPayment where month(spay_date)=3 AND artistid='CR16' group by artistid;
+----------+---------------------+
| artistid | sum(artist_payment) |
+----------+---------------------+
| CR16     |  50.400001525878906 |
+----------+---------------------+
```

Calculate total payments made out to the record label per a given time period.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from songPayment;
+---------+--------+----------+---------------+---------------+----------------+------------+------------+
| spayid  | songid | artistid | labelname     | label_payment | artist_payment | spay_date  | Main_label |
+---------+--------+----------+---------------+---------------+----------------+------------+------------+
|      17 | M11    | CR12     | Pluto         |            20 |        4.66667 | 2023-02-01 | Yes        |
|      18 | M14    | CR12     | Pluto         |            84 |           29.4 | 2023-02-01 | Yes        |
|      19 | M12    | CR13     | LN 32 Music Group |             6 |            4.2 | 2023-02-01 | Yes        |
|      20 | M14    | CR17     | Pluto         |            84 |           29.4 | 2023-02-01 | No         |
|      21 | M11    | CR18     | AV Productions |            20 |        4.66667 | 2023-02-01 | No         |
|      22 | M11    | CR19     | Pluto         |            20 |        4.66667 | 2023-02-01 | No         |
|      24 | M15    | CR11     | Lorimer Studios |            48 |           16.8 | 2023-03-01 | Yes        |
|      25 | M11    | CR12     | Pluto         |            24 |            5.6 | 2023-03-01 | Yes        |
|      26 | M14    | CR12     | Pluto         |            98 |           34.3 | 2023-03-01 | Yes        |
|      27 | M12    | CR13     | LN 32 Music Group |             8 |            5.6 | 2023-03-01 | Yes        |
|      28 | M13    | CR14     | LN 32 Music Group |             7 |            4.9 | 2023-03-01 | Yes        |
|      29 | M15    | CR15     | MSeries       |            48 |           16.8 | 2023-03-01 | No         |
|      30 | M16    | CR16     | AV Productions |            72 |           50.4 | 2023-03-01 | Yes        |
|      31 | M14    | CR17     | Pluto         |            98 |           34.3 | 2023-03-01 | No         |
|      32 | M11    | CR18     | AV Productions |            24 |            5.6 | 2023-03-01 | No         |
|      33 | M11    | CR19     | Pluto         |            24 |            5.6 | 2023-03-01 | No         |
+---------+--------+----------+---------------+---------------+----------------+------------+------------+
16 rows in set (0.0023 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select labelname, sum(label_payment) from songPayment where Main_label='Yes' AND month(spay_date)=3 AND labelname='Pluto' group by labelname;
+-----------+---------------------+
| labelname | sum(label_payment) |
+-----------+---------------------+
| Pluto     |                 122 |
+-----------+---------------------+
```

Total revenue of the streaming service per month

```
1 row in set (0.0017 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from user_payment;
+------+------------+------------+
| uid  | uphone     | ups_date   |
+------+------------+------------+
| US11 | 1111111112 | 2023-03-07 |
| US12 | 1111111114 | 2023-01-11 |
| US13 | 1111111114 | 2023-02-11 |
| US14 | 1111111116 | 2023-01-02 |
| US15 | 1111111111 | 2023-03-06 |
| US16 | 1111111113 | 2023-03-04 |
| US17 | 1111111113 | 2023-02-04 |
| US18 | 1111111113 | 2023-01-04 |
| US19 | 1111111115 | 2023-01-09 |
| US20 | 1111111115 | 2023-02-09 |
| US21 | 1111111115 | 2023-03-09 |
| US22 | 1111111117 | 2023-03-02 |
| US23 | 1111111118 | 2023-01-07 |
| US24 | 1111111114 | 2023-03-11 |
+------+------------+------------+
14 rows in set (0.0013 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select (select avg(sub_fee) from user where u_status='Active') * (select count(*) from user_payment where ups_date BETWEEN CAST('2023-01-02' AS DATE) AND
 CAST('2023-02-28' AS DATE) ) as Feb_Revenue;
+-------------+
| Feb_Revenue |
+-------------+
|         120 |
+-------------+
1 row in set (0.0015 sec)
```

Total revenue of the streaming service per year.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select * from user_payment;
+------+------------+------------+
| uid  | uphone     | ups_date   |
+------+------------+------------+
| US11 | 1111111112 | 2023-03-07 |
| US12 | 1111111114 | 2023-01-11 |
| US13 | 1111111114 | 2023-02-11 |
| US14 | 1111111116 | 2023-01-02 |
| US15 | 1111111111 | 2023-03-06 |
| US16 | 1111111113 | 2023-03-04 |
| US17 | 1111111113 | 2023-02-04 |
| US18 | 1111111113 | 2023-01-04 |
| US19 | 1111111115 | 2023-01-09 |
| US20 | 1111111115 | 2023-02-09 |
| US21 | 1111111115 | 2023-03-09 |
| US22 | 1111111117 | 2023-03-02 |
| US23 | 1111111118 | 2023-01-07 |
| US24 | 1111111114 | 2023-03-11 |
+------+------------+------------+
14 rows in set (0.0012 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select 15 * (select count(*) from user_payment where ups_date BETWEEN CAST('2023-01-02' AS DATE) AND CAST('2023-02-28' AS DATE) ) as Feb_Revenue;
+-------------+
| Feb_Revenue |
+-------------+
|         120 |
+-------------+
1 row in set (0.0012 sec)
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select 15 * (select count(*) from user_payment where ups_date BETWEEN CAST('2023-01-01' AS DATE) AND CAST('2023-12-31' AS DATE) ) as Yearly_Revenue;
+----------------+
| Yearly_Revenue |
+----------------+
|            210 |
+----------------+
```

Report all songs given by an artist

The information related to which song is sung by which artist is maintained in the composedby table. Hence the song is joined with that table along with the artist table to obtain the desired output.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select media_name, cf_name, cl_name from Media natural join Song
join composedby on mediaid=songid join Artist on artistid=creatorsid natural join Creators where creatorsid='CR12';
+--------------------+---------+---------+
| media_name         | cf_name | cl_name |
+--------------------+---------+---------+
| Bones              | Imagine | Dragons |
| Waking Up Dreaming | Imagine | Dragons |
+--------------------+---------+---------+
2 rows in set (0.0024 sec)
```

Report all songs given an album

The song and album can be combined on the albumid since we have assumed that every song belongs to at most one album.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select media_name, album_name from Media natural join Song natura
l join Album where albumid='A11';
+--------------------+------------+
| media_name         | album_name |
+--------------------+------------+
| Bones              | Bones      |
| Waking Up Dreaming | Bones      |
+--------------------+------------+
2 rows in set (0.0017 sec)
```

Report all songs given an artist and album

The song has the information of the album in the song table itself. Hence we only need to join artist and song along with creators and media in order to get the information needed.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select media_name, album_name, cf_name, cl_name from Media natura
l join Song natural join Album join Artist on artistid=creatorsid natural join Creators where creatorsid='CR12' and
albumid='A11';
+--------------------+------------+---------+---------+
| media_name         | album_name | cf_name | cl_name |
+--------------------+------------+---------+---------+
| Bones              | Bones      | Imagine | Dragons |
| Waking Up Dreaming | Bones      | Imagine | Dragons |
+--------------------+------------+---------+---------+
2 rows in set (0.0020 sec)
```

Report all podcast episodes given podcast

The information for the podcastepisode can be obtained from the podcastid in the podcastepisode table.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select title, episodeno, media_name from Media natural join Podca
st join podcastEpisode on mediaid=podcastid where mediaid='P15';
+------------+-----------+-----------------+
| title      | episodeno | media_name      |
+------------+-----------+-----------------+
| Mangalyaan |         1 | Echoes of India |
| Chandrayan |         2 | Echoes of India |
| Gaganyaan  |         3 | Echoes of India |
+------------+-----------+-----------------+
3 rows in set (0.0020 sec)
```

4.2 Use the EXPLAIN directive in MariaDB to study execution plans for your queries (from item 4.1 above). Find two queries for which EXPLAIN shows full table scans; print the EXPLAIN outputs for these queries. For these two queries, create appropriate indexes; print the EXPLAIN output that shows the use of the indexes. For each query, submit the printouts of: (1) the SQL query, (2) the execution plan (EXPLAIN) for the query which shows at least one full table scan, (3) your index-creation statement in SQL, and (4) the execution plan (EXPLAIN) for the query where the full table scan has been replaced with a use of your index.

QUERY 1:-

1.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select artistid,sum(artist_payment) from songPayment where month(spay_date)=3 AND artistid='CR16' group by artistid;
+-----------+----------------------+
| artistid | sum(artist_payment) |
+-----------+----------------------+
| CR16     |  50.400001525878906 |
+-----------+----------------------+
1 row in set (0.0019 sec)
```

2.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL EXPLAIN select artistid,sum(artist_payment) from songPayment where month(spay_date)=3 AND artistid='CR16' group by artistid;
+----+-------------+-------------+------+---------------+----------+---------+-------+------+-----------------------------------------+
| id | select_type | table       | type | possible_keys | key      | key_len | ref   | rows | Extra                                   |
+----+-------------+-------------+------+---------------+----------+---------+-------+------+-----------------------------------------+
|  1 | SIMPLE      | songPayment | ref  | artistid      | artistid | 7       | const |    1 | Using index condition; Using where |
+----+-------------+-------------+------+---------------+----------+---------+-------+------+-----------------------------------------+
```

3.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL create INDEX ar_idx ON songPayment(artistid);
Query OK, 0 rows affected (0.0515 sec)

Records: 0  Duplicates: 0  Warnings: 0
```

4.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL EXPLAIN select artistid,sum(artist_payment) from songPayment where month(spay_date)=3 AND artistid='CR16' group by artistid;
+----+-------------+-------------+------+---------------+--------+---------+-------+------+-----------------------------------------+
| id | select_type | table       | type | possible_keys | key    | key_len | ref   | rows | Extra                                   |
+----+-------------+-------------+------+---------------+--------+---------+-------+------+-----------------------------------------+
|  1 | SIMPLE      | songPayment | ref  | ar_idx        | ar_idx | 7       | const |    1 | Using index condition; Using where |
+----+-------------+-------------+------+---------------+--------+---------+-------+------+-----------------------------------------+
1 row in set (0.0015 sec)
```

QUERY2:-

1.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL select labelname, sum(label_payment) from songPayment where Main_label='Yes' AND month(spay_date)=3 group by labelname;
+--------------------+---------------------+
| labelname          | sum(label_payment) |
+--------------------+---------------------+
| AV Productions     |                  72 |
| LN 32 Music Group  |                  15 |
| Lorimer Studios    |                  48 |
| Pluto              |                 122 |
+--------------------+---------------------+
4 rows in set (0.0021 sec)
```

2.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL EXPLAIN select labelname, sum(label_payment) from songPayment where Main_label='Yes' AND month(spay_date)=3 group by labelname;
+----+-------------+-------------+-------+---------------+-----------+---------+------+------+--------------+
| id | select_type | table       | type  | possible_keys | key       | key_len | ref  | rows | Extra        |
+----+-------------+-------------+-------+---------------+-----------+---------+------+------+--------------+
|  1 | SIMPLE      | songPayment | index | NULL          | labelname | 82      | NULL |   16 | Using where |
+----+-------------+-------------+-------+---------------+-----------+---------+------+------+--------------+
1 row in set (0.0016 sec)
```

3.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL create INDEX rl_idx ON songPayment(labelname);
Query OK, 0 rows affected (0.0694 sec)

Records: 0  Duplicates: 0  Warnings: 0
```

4.

```
MySQL classdb2.csc.ncsu.edu:3306 hsangha SQL EXPLAIN select labelname, sum(label_payment) from songPayment where Main_label='Yes' AND month(spay_date)=3 group by labelname;
+----+-------------+-------------+-------+---------------+--------+---------+------+------+--------------+
| id | select_type | table       | type  | possible_keys | key    | key_len | ref  | rows | Extra        |
+----+-------------+-------------+-------+---------------+--------+---------+------+------+--------------+
|  1 | SIMPLE      | songPayment | index | NULL          | rl_idx | 82      | NULL |   16 | Using where |
+----+-------------+-------------+-------+---------------+--------+---------+------+------+--------------+
1 row in set (0.0016 sec)
```

4.3 For any two of your SQL queries ("select" statements only, rather than "insert/delete/update") with joins, explain why the queries are correct - one explanation per query. If you come up with erroneous solutions prior to the correct one, also include those as part of the explanation. For each query, submit the query itself and your explanation. Your explanations will consist of two parts:

- **the corresponding relational algebra expression, and**
- **a *correctness proof* that shows that the query answer always corresponds to the query specification;**

QUERY 1:-

Specification:- Calculate total payments made out to the artist per a given time period.

Query:

```
select artistid,sum(artist_payment)

from songPayment

where month(spay_date)=3 AND artistid='CR16'

group by artistid;
```

Relational Algebra:

$\gamma_{artistid,sum(artist\_payment)}(\sigma_{month(spay\_date)=3\ AND\ artistid==\text{'}CR16\text{'}}(songPayment))$

Correctness Proof:

The songPayment stores the record for each media and contains the artistid. Suppose there is a tuple in the artist relation with artistid CR16, then there exists a same record with artistid CR16 in songPayment table. All the corresponding paymentid related to artistid CR16 are combined together through sum. Therefore total payments for the particular artist is received which is exactly what we wanted.This proves that the above query is correct.

QUERY 2:-

Specification :- Calculate total payments made out to the Record Label per a given time period.

Query:

```
select labelname, sum(label_payment)
from songPayment
where Main_label='Yes' AND month(spay_date)=3
group by labelname;
```

Relational Algebra:

$\gamma_{labelname,sum(labelt\_payment)}(\sigma_{month(spay\_date)=3 \text{ AND } Main\_label==\text{'Yes'}}(\text{songPayment}))$

Correctness Proof:

The songPayment stores the record for each Record Label and contains the labelname. Suppose there is a tuple in the record label relation with labelname as the primary key then there exists a same record with the same primary key in the songPayment table. All the corresponding paymentid related to all the record label which have owned any song are summed up to give the total payment in the songPayment table for that particular record label. This is exactly what we wanted and thus the above query is correct.

5. Peer evaluations (via submit board). For each project report, if you (individually) wish to get a grade for the report, you must submit your own peer evaluation of each member of your team. Please submit a single plain-text file per student via the submit board, with your own evaluation of all members of your team including yourself, using the values (such as "excellent", "ordinary", or "superficial", to grade your team members' participation in the teamwork) listed in the peer evaluation form. No signature is needed in your submit-board submission.

Peer evaluations are submitted via the submit board.