Music Database :-

* Notown Record has decided to store information about musicians who perform on its albums. Each musician that records at Notown has an SSN, a name, an address, and a phone number. No musician has more than one phone. Each instrument used in songs recorded at Notown has a unique identification number, a name (e.g guitar, synthesizer, and flute) and a musical key (e.g C, B- flat, E- flat). Each album recorded on the Notown label has a unique identification number, a title, a copyright date and a format (e.g. CD or MC). Each song recorded at Notown has a id title and an author. Each musician may play several instruments; and a given instrument may be played by several musicians. Each album has a number of songs on it, but no song may appear on more than one album Each song is performed by one or more musicians and a musician may perform a number of songs. Each album has exactly one musician who acts as its producer. A musician may produce several albums

Step 1 : Identification of entities.
Step 2 : Identifications of attribute for each of the entities
Step 3 : Relationship between the entities.
Step 4 : cardinality constraint and participate constraint
Step 5 : ER Diagram
Step 6 : Relationship Schema Diagram

* Entities :-

Musician, Album, Instrument, Song

* attribute

* Entities

MUSICIAN, ALBUM, INSTRUMENT, SONG

* Attributes :-

MUSICIAN :- SSN, Name, Address, phone_no

* ALBUM :- AID, title, copyright_Date

INSTRUMENT :-

I_ID, Name, key (c, B-flat, E-flat)

SONG :-

SID, Author, Title

(3) Relationship :

MUSICIAN and INSTRUMENT

plays

MUSICIAN and ALBUM

produces

MUSICIAN and SONG

performs

* ALBUM and SONG

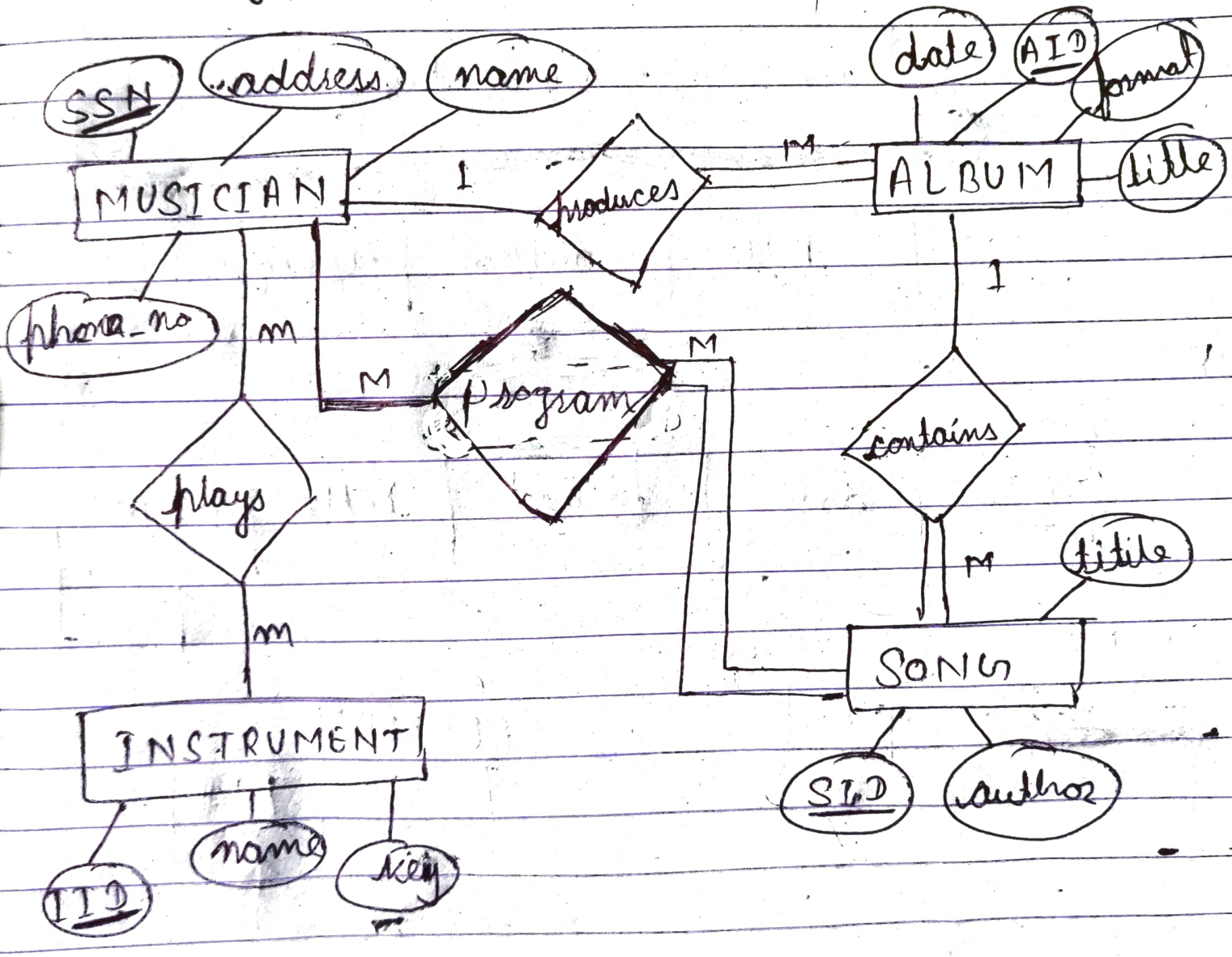
Appears / contains

④ Cardinality of relationship :-

* Many to many [MUSICIAN, INSTRUMENT]
* Many to many [MUSICIAN, SONG]
* one to many [MUSICIAN, ALBUM]
* one to many [ALBUM, SONG]

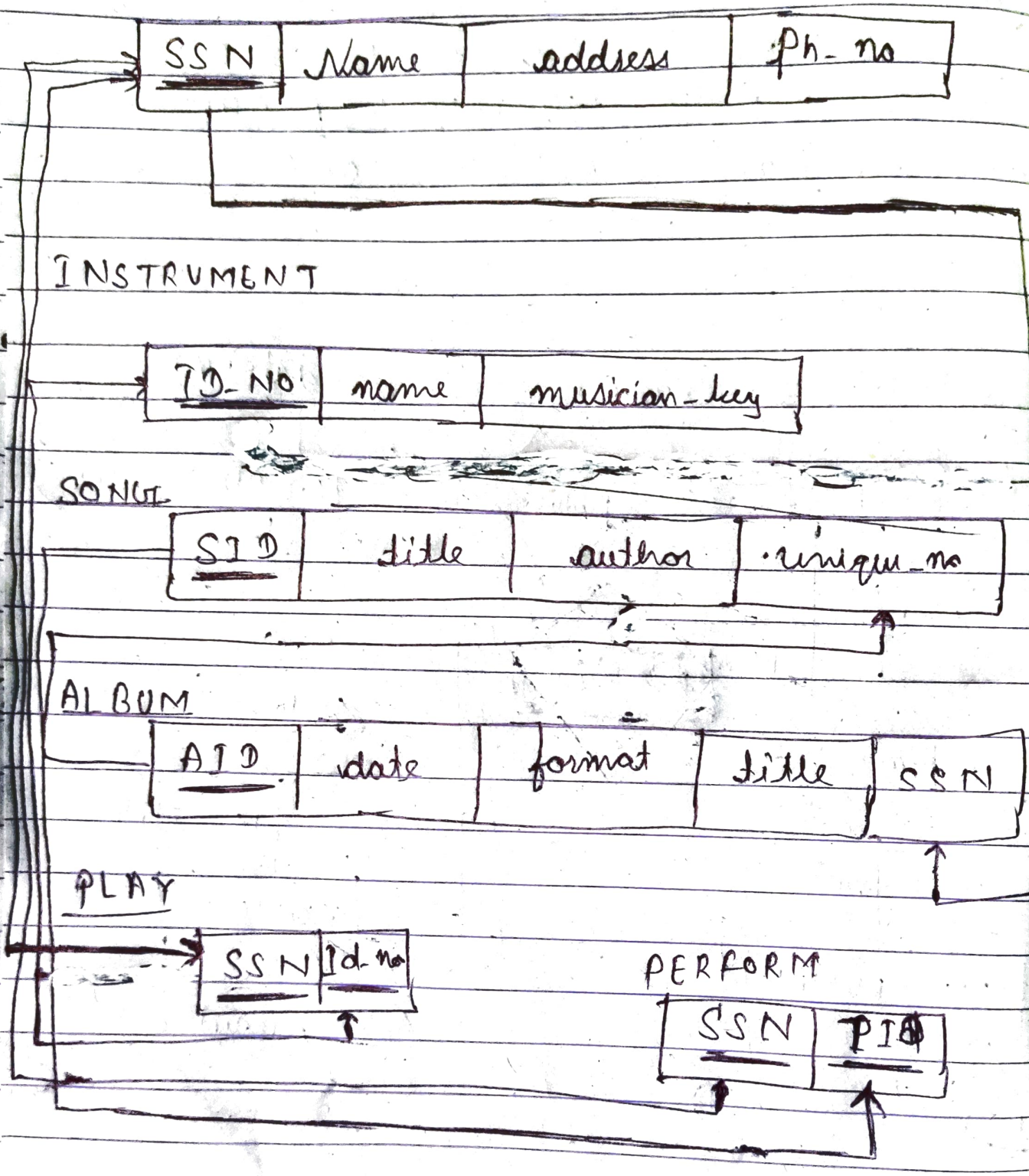
Step - 5 :-

* ER- Diagram



Step :- 6

① MUSICIAN



```
mysql> create database lab1;
Query OK, 1 row affected (0.23 sec)
mysql> use rvce;
Database changed
```

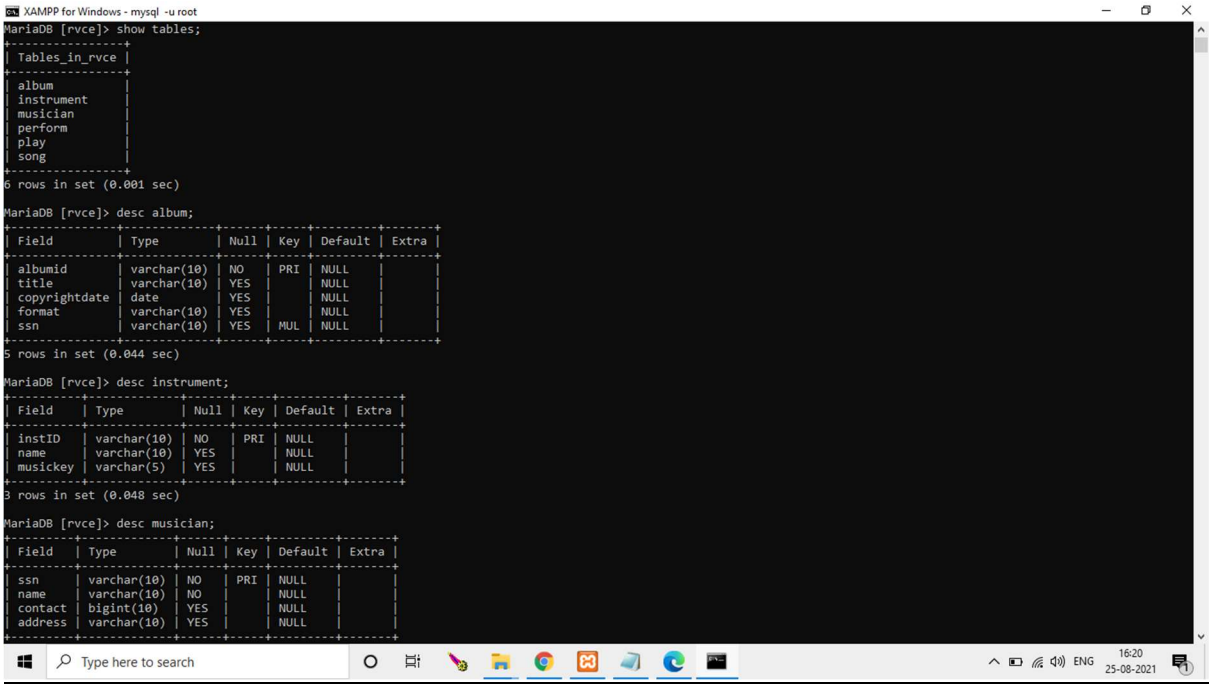
Table Creation

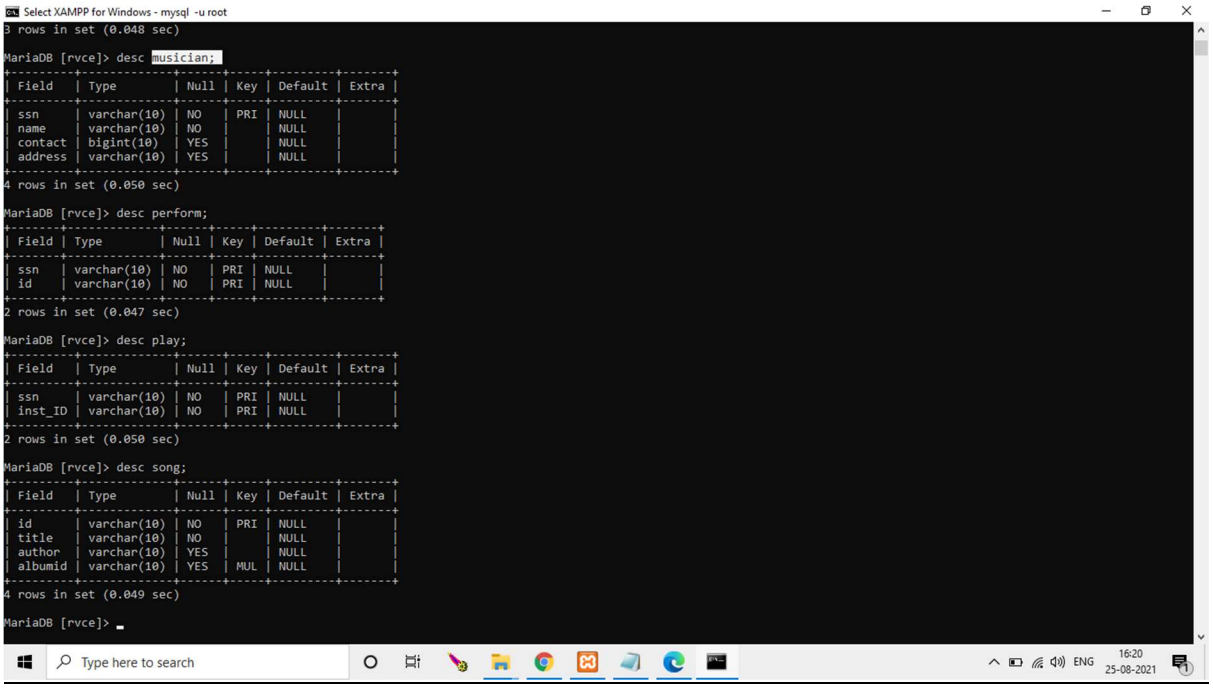
```
mysql> create table MUSICIAN(ssn varchar(10) primary key,name varchar (10) NOT NULL,contact bigint(10),address varchar(10));
Query OK, 0 rows affected, 1 warning (1.34 sec)

mysql> desc MUSICIAN;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| ssn     | varchar(10) | NO   | PRI | NULL    |       |
| name    | varchar(10) | NO   |     | NULL    |       |
| contact | bigint      | YES  |     | NULL    |       |
| address | varchar(10) | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
4 rows in set (0.05 sec)

mysql> create table INSTRUMENT(instID varchar(10)primary key,name varchar(10),musickey varchar(5));

mysql> create table ALBUM(albumid varchar(10)primary key,title varchar(10),copyrightdate date,ssn varchar(10),foreign key(ssn)
references MUSICIAN(ssn));

mysql> create table PLAY(ssn varchar(10),inst_ID varchar(10),foreign key(ssn) references MUSICIAN(ssn),foreign key(inst_ID) references
INSTRUMENT(instID),primary key(ssn,inst_ID));

mysql> create table SONG(id varchar(10)primary key,title varchar(10) NOT NULL,author varchar(10),albumid varchar(10),foreign
key(albumid) references ALBUM(albumid));

mysql> create table PERFORM(ssn varchar(10),id varchar(10),foreign key(ssn) references MUSICIAN(ssn), foreign key(id) references
SONG(id),primary key(ssn,id));
```

Insert Data into a Musician Table:-

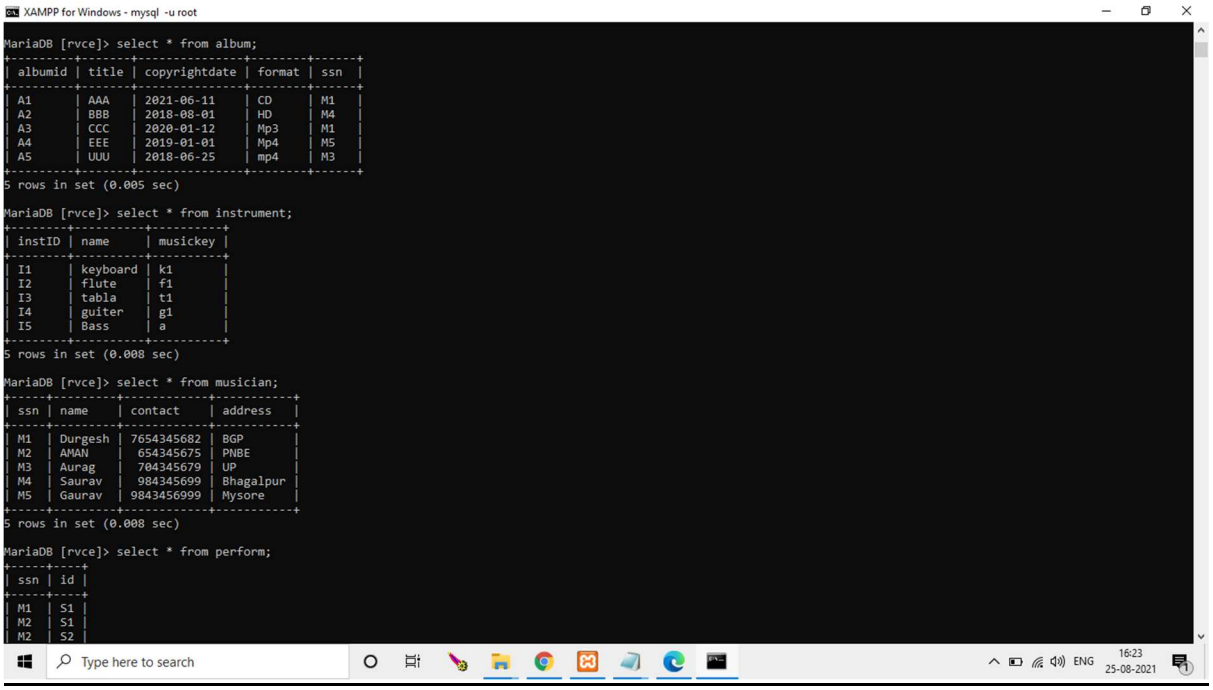
```
Sql>insert into musician(ssn,name,contact,address) values("M1","Durgesh",7564533456,"BGP");
```

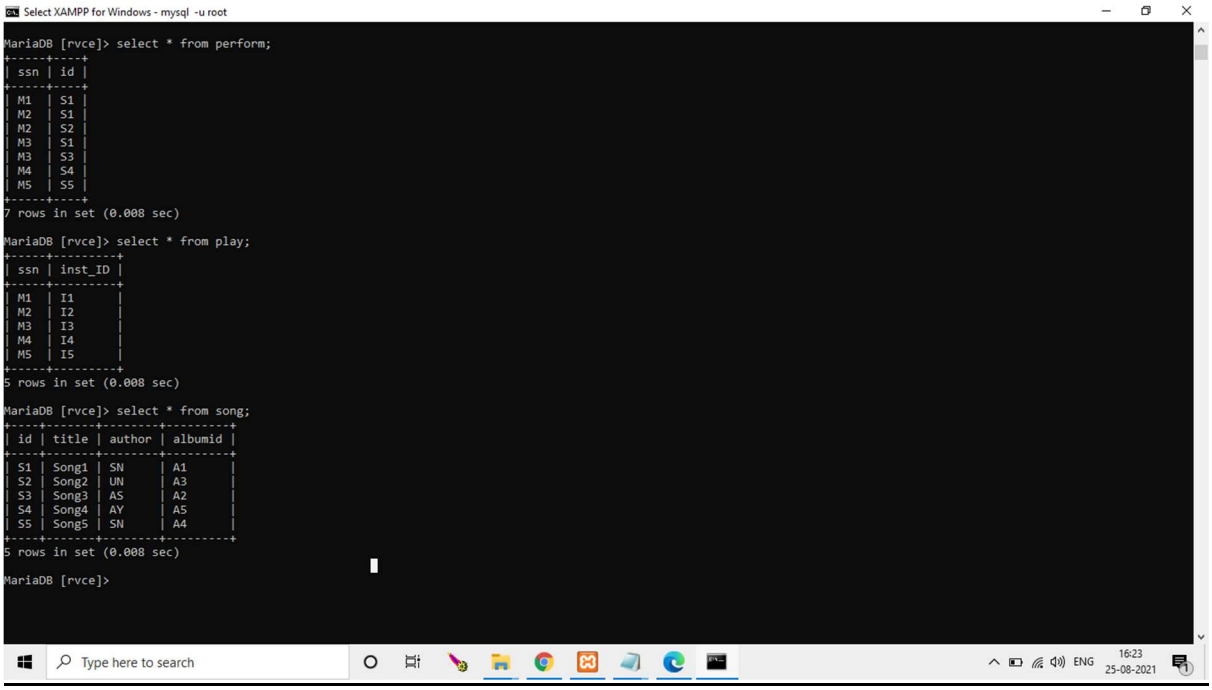
Check the Description of a table:-





Show Data of all Table:-





Class Work:-

1.a:- List musician name, title of the song which he has played, the album in which song has occulted.
Answer:-select m.name,s.title,a.albumid from musician m,perform p,song s,album a where m.ssn=p.ssn and p.id=s.id and s.albumid=a.albumid;
or
select m.name,s.title,a.title from musician m,perform p,song s,album a where m.ssn=p.ssn and p.id=s.id and s.albumid=a.albumid;

2.b:-List the musicians who have not produced any album.
Answer:-select m.ssn,m.name from musician m where m.ssn not in(select m.ssn from musician m,album a where m.ssn=a.ssn);

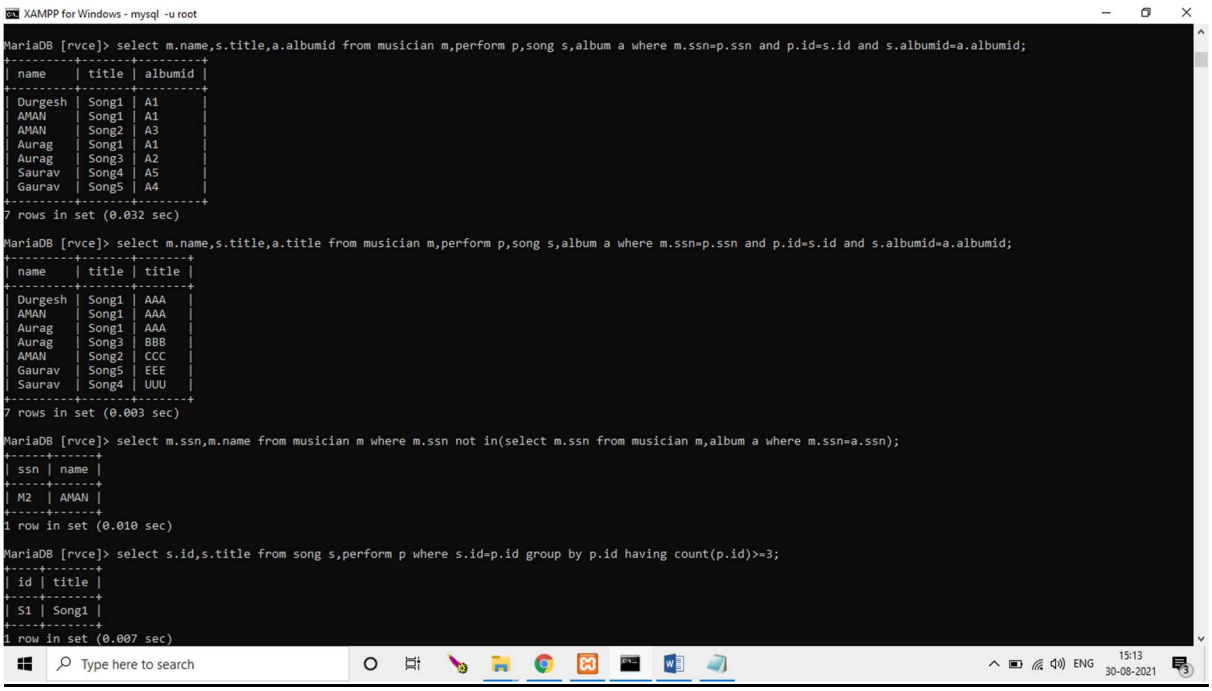
3.c:-List the details of songs which are performed bymore than 3 musicians.
Answer:-select s.id,s.title from song s,perform p where s.id=p.id group by p.id having count(p.id)>=3;

4.d:-List the different instruments played by the musicians and the average number of musicians who play the instrument.
Answer:-select count(p.inst_id), i.name,count(p.Inst_ID)/(select count(*) from musician) as 'Musician name' from play p ,instrument i where i.instID=p.inst_id group by p.inst_id ;

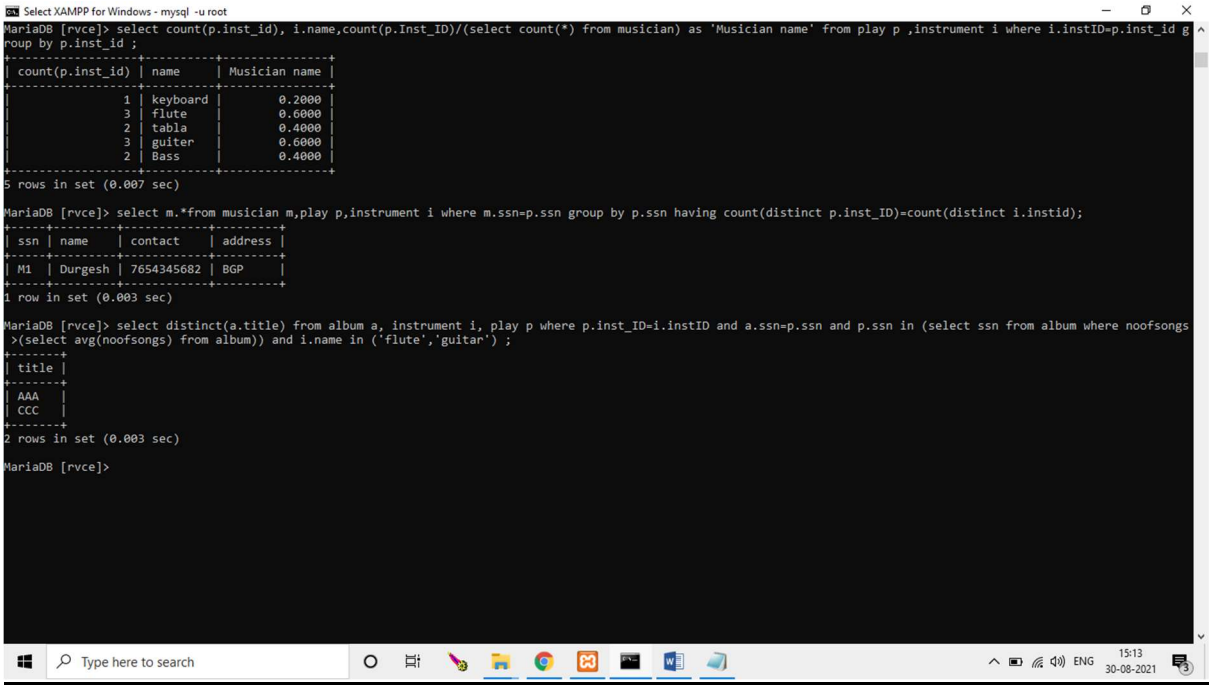
5:-Retrieve album title produced by the producer who plays guitar as well as flute and has produced no of songs greater than the average songsproduced by all producers.
Answer:-select m.*from musician m,play p,instrument i where m.ssn=p.ssn group by p.ssn having count(distinct p.inst_ID)=count(distinct i.instid);

6.f:-List the details of musicians who can play all the instruments present.
Answer:-select distinct(a.title) from album a, instrument i, play p where p.inst_ID=i.instID and a.ssn=p.ssn and p.ssn in (select ssn from album where noofsongs >(select avg(noofsongs) from album)) and i.name in ('flute','guitar') ;

Screenshot of class Work:-



```
XAMPP for Windows - mysql  -u root

MariaDB [rvce]> select m.name,s.title,a.albumid from musician m,perform p,song s,album a where m.ssn=p.ssn and p.id=s.id and s.albumid=a.albumid;
+---------+-------+---------+
| name    | title | albumid |
+---------+-------+---------+
| Durgesh | Song1 | A1      |
| AMAN    | Song1 | A1      |
| AMAN    | Song2 | A3      |
| Aurag   | Song1 | A1      |
| Aurag   | Song3 | A2      |
| Saurav  | Song4 | A5      |
| Gaurav  | Song5 | A4      |
+---------+-------+---------+
7 rows in set (0.032 sec)

MariaDB [rvce]> select m.name,s.title,a.title from musician m,perform p,song s,album a where m.ssn=p.ssn and p.id=s.id and s.albumid=a.albumid;
+---------+-------+-------+
| name    | title | title |
+---------+-------+-------+
| Durgesh | Song1 | AAA   |
| AMAN    | Song1 | AAA   |
| Aurag   | Song1 | AAA   |
| Aurag   | Song3 | BBB   |
| AMAN    | Song2 | CCC   |
| Gaurav  | Song5 | EEE   |
| Saurav  | Song4 | UUU   |
+---------+-------+-------+
7 rows in set (0.003 sec)

MariaDB [rvce]> select m.ssn,m.name from musician m where m.ssn not in(select m.ssn from musician m,album a where m.ssn=a.ssn);
+-----+------+
| ssn | name |
+-----+------+
| M2  | AMAN |
+-----+------+
1 row in set (0.010 sec)

MariaDB [rvce]> select s.id,s.title from song s,perform p where s.id=p.id group by p.id having count(p.id)>=3;
+----+-------+
| id | title |
+----+-------+
| S1 | Song1 |
+----+-------+
1 row in set (0.007 sec)
```



```
Select XAMPP for Windows - mysql  -u root

MariaDB [rvce]> select count(p.inst_id), i.name,count(p.Inst_ID)/(select count(*) from musician) as 'Musician name' from play p ,instrument i where i.instID=p.inst_id g
roup by p.inst_id ;
+-----------------+----------+--------------+
| count(p.inst_id) | name     | Musician name |
+-----------------+----------+--------------+
|               1 | keyboard |       0.2000 |
|               3 | flute    |       0.6000 |
|               2 | tabla    |       0.4000 |
|               3 | guiter   |       0.6000 |
|               2 | Bass     |       0.4000 |
+-----------------+----------+--------------+
5 rows in set (0.007 sec)

MariaDB [rvce]> select m.*from musician m,play p,instrument i where m.ssn=p.ssn group by p.ssn having count(distinct p.inst_ID)=count(distinct i.instid);
+-----+---------+------------+---------+
| ssn | name    | contact    | address |
+-----+---------+------------+---------+
| M1  | Durgesh | 7654345682 | BGP     |
+-----+---------+------------+---------+
1 row in set (0.003 sec)

MariaDB [rvce]> select distinct(a.title) from album a, instrument i, play p where p.inst_ID=i.instID and a.ssn=p.ssn and p.ssn in (select ssn from album where noofsongs
 >(select avg(noofsongs) from album)) and i.name in ('flute','guitar') ;
+-------+
| title |
+-------+
| AAA   |
| CCC   |
+-------+
2 rows in set (0.003 sec)

MariaDB [rvce]>
```