LONDON
METROPOLITAN
UNIVERSITY

ITAHARI
INTERNATIONAL
COLLEGE

Module Code & Module Title

CC4057NT Introduction to Information System

Assessment Weightage & Type

30% Individual Coursework

Year and Semester

2019-20 Spring

Student Name: Mishan Rai

London Met ID: 19033566

College ID: np05cp4s200023

Assignment Due Date: 2020/04/17

Assignment Submission Date: 2020/04/16

Title (Where Required):

Word Count (Where Required):

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Table of Contents

1. INTRODUCTION 1
2. DATA MODEL 2
3. ERD (ENTITY RELATIONSHIP DIAGRAM) 7
4. DATA DICTIONARY 13
5. DATABASE CREATION 18
6. DATABASE QUERIES 22
7. CONCLUSION 27
8. BIBLIOGRAPHY 28

Table of tables

Table 1 of Genres datatype 2
Table 2 of Songs datatype 2
Table 3 of Artists datatype 3
Table 4 of Customers datatype 3
Table 5 of Bills datatype 3
Table 6 of Genres 4
Table 7 of Songs 5
Table 8 of Artists 5
Table 9 of Customers 6
Table 10 of Bills 6
Table 11 Data Dictionary of Genres 13
Table 12 Data Dictionary of Songs 14
Table 13 Data Dictionary of Artists 15
Table 14 Data Dictionary of Customers 16
Table 15 Data Dictionary of Bills 17

Table of figure

Figure 1 of Genres 7
Figure 2 of Songs 8
Figure 3 of Artists 8
Figure 4 of Customers 8
Figure 5 of Bills 9
Figure 6 of ERD (Entity Relation Diagram) 11
Figure 7 of RD (Relational Diagram) 12
Figure 8 Database creation MusicStore 18
Figure 9 Creating Genres table 18
Figure 10 Creating Songs table 18
Figure 11 Creating Artists table 19
Figure 12 Creating Customers table 19
Figure 13 Creating Bills table 19
Figure 14 Inserting values in Genres 19
Figure 15 Inserting values in Songs 20
Figure 16 Inserting values in Artists 20
Figure 17 Inserting values in Customers 21
Figure 18 Inserting values in Bills 21
Figure 19 Using Max and Min query 22
Figure 20 Using Show tables query 22

Figure 21 Using Select now () query..23
Figure 22 Using Limit query 23
Figure 23 Using Where query 24
Figure 24 Using Order By query 24
Figure 25 Using Like query 25
Figure 26 Using Between query...25
Figure 27 Using Group By query..26
Figure 28 Using Count query 26

1. INTRODUCTION

Before starting this coursework, we should know about database. Database is a collection of data which helps to reduce ones effort making easier for people to search about the data of the people working under them either governmental or ungovernmental works. We were provided a coursework related to this database. And we also needed to create a data model, ERD (Entity Diagram Relationship), RD (Relational Database), data dictionary. It contained 30% marks.

In the first coursework, I had created a database about department store. So, in this second coursework I tried to make a different database and made a database of MusicStore. This database has different entities with each entities having different attributes. The entities of this database are: Genres, Songs, Artists, Customers and Bills where Genres has GenreId and Name attributes, Songs has SongId, Name, Price and ReleaseDate attributes, Artists has ArtistId, FirstName, LastName and Song attributes, Customers has CustomerId, FirstName, LastName and Contact attributes and lastly Bills has BillId attributes.

All the entities has a suitable primary key and their relations with the attributes is interlinked using suitable foreign key. The attibutes of each entities is controlled using suitable constraints like unique, not null, auto increment etc. At last, I used some of the queries to check the changes in that occurred in the database.

2. DATA MODEL

A data model is a data flow or logical inter-relationship between different data models. It helps to show how the data is connected to eachother and also shows how the data are processed, stored and retrieved. It also helps to enhance the communication business as well as technical development. It can also help to represent what type of data is required varying different formats which is used for business and other purposes. (Anon., 2017), (Anon., 2019)

Hence, I have prepared a data model for MusicStore. And it's tables are:

a. Genres

GenreId	INT	PRIMARY KEY
Name	VARCHAR(225)	NULL

Table 1 of Genres datatype

b. Songs

SongId	INT	PRIMARY KEY
Name	VARCHAR(225)	NULL
Price	INT	NULL
ReleaseDate	DATE	NULL
Genre	INT	FOREIGN KEY

Table 2 of Songs datatype

c. Artists

ArtistId	INT	PRIMARY KEY
FirstName	VARCHAR(225)	NULL
LastName	VARCHAR(225)	NULL
Song	INT	FOREIGN KEY

Table 3 of Artists datatype

d. Customers

CustomerId	INT	PRIMARY KEY
FirstName	VARCHAR(225)	NULL
LastName	VARCHAR(225)	NULL
Contact	INT	NULL
Artist	INT	FOREIGN KEY

Table 4 of Customers datatype

e. Bills

BillId	INT	PRIMARY KEY
Customer	INT	FOREIGN KEY

Table 5 of Bills datatype

The values of the above tables are:

a. Genres

GenreId	Name
11	Country
22	Classic
33	Hip-Hop
44	Jazz
55	R&B
66	Electronic
77	Pop

Table 6 of Genres

In this table, GenreId is the primary key.

b. Songs

SongId	Name	Price	Genre	ReleaseDate
10	Take me home	90	11	1971-04-10
20	Stand by me	50	22	1961-04-24
30	Lose yourself	110	33	2002-10-28
40	What a wonderful world	55	44	1967-01-01
50	See you again	130	55	2015-03-10
60	Faded	120	66	2015-12-03

70	Dusk till dawn	100	77	2017-09-07

Table 7 of Songs

In this table, SongId is the primary key and Genre is the foreign key.

c. Artists

ArtistId	FirstName	LastName	Song
12	John	Denver	10
23	John	Lennon	20
34	Marshall	Mathers	30
45	Louis	Armstrong	40
56	Wiz	Khalifa	50
67	Alan	Walker	60
71	Zayn	Malik	70

Table 8 of Artists

In this table, ArtistId is the primary key and Song is the foreign key.

d. Customers

CustomerId	FirstName	LastName	Contact	Artist
1	Snoop	Dog	890-234-234	12
2	Charlie	puth	213-345-435	23
3	Ice	Cube	876-123-098	34
4	Pink	Guy	231-234-342	45

5	Rich	Brian	897-123-243	56
6	John	Legend	234-112-234	67
7	Kevin	Hart	653-324-547	71

Table 9 of Customers

In this table, CustomerId is the primary key and Artist is the foreign key.

e. Bills

BillId	Customer
001	1
002	2
003	3
004	4
005	5
006	6
007	7

Table 10 of Bills

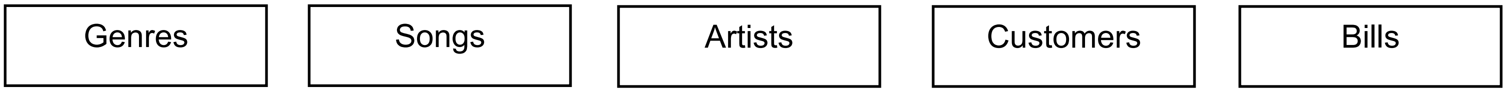
In this table, BillId is the primary key and Customer is the foreign key.

3. ERD (ENTITY RELATIONSHIP DIAGRAM)

An ERD (Entity Relationship Diagram) is a data modelling technique of graphical representation of the entities and their relationship between each entities (Beal, n.d.), (Anon., 2017). Some of the components of database are:

a. Entity

An entity is a type of object. Entity can be refereed as people, object or events which stores data. Each entity has different attributes. They are represented in rectangular boxes. The entities which I have created in the database MusicStore are:



b. Attributes

Attributes are the property of an entity. Some of the attributes of the above entity are:

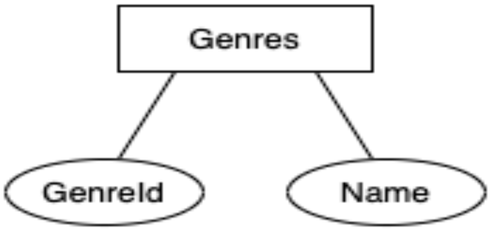


Figure 1 of Genres

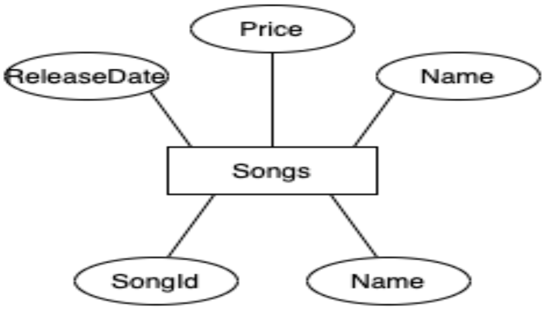


Figure 2 of Songs

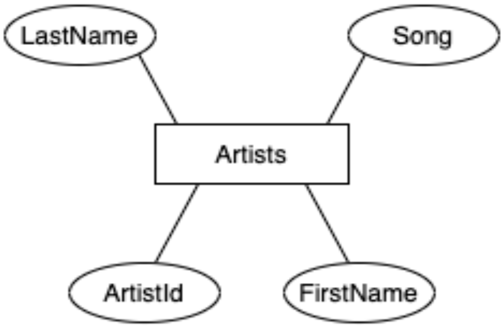


Figure 3 of Artists

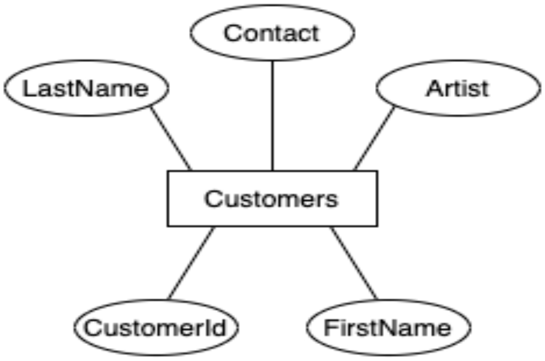


Figure 4 of Customers

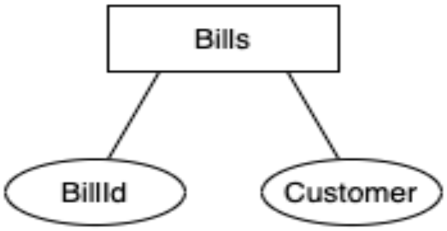
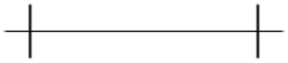


Figure 5 of Bills

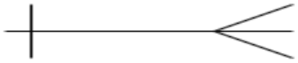
c. Relationship

It means relating two or more entities which take part in relationship. It is represented in diamond symbol. Some of its indicators are:

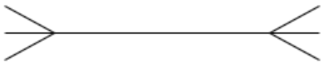
I. One to one:

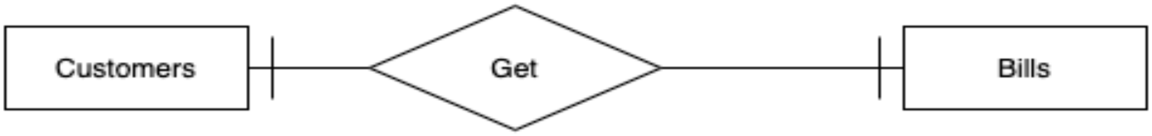


II. One to many:

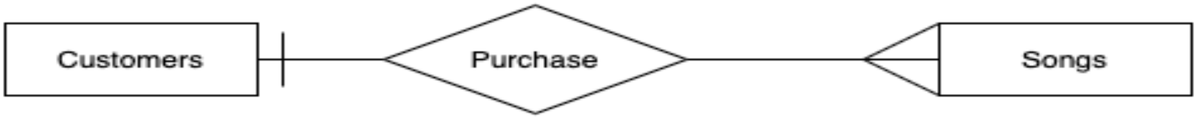


III. Many to many:

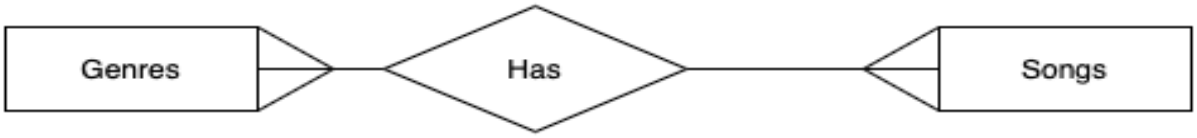




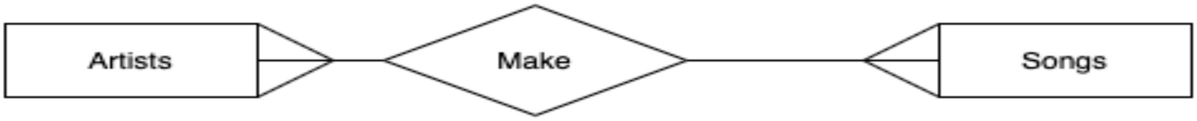
One customer can get one bill.



One customer can buy many songs.



Many songs has many genres.



Many artists makes many songs.

d. ERD of database MusicStore

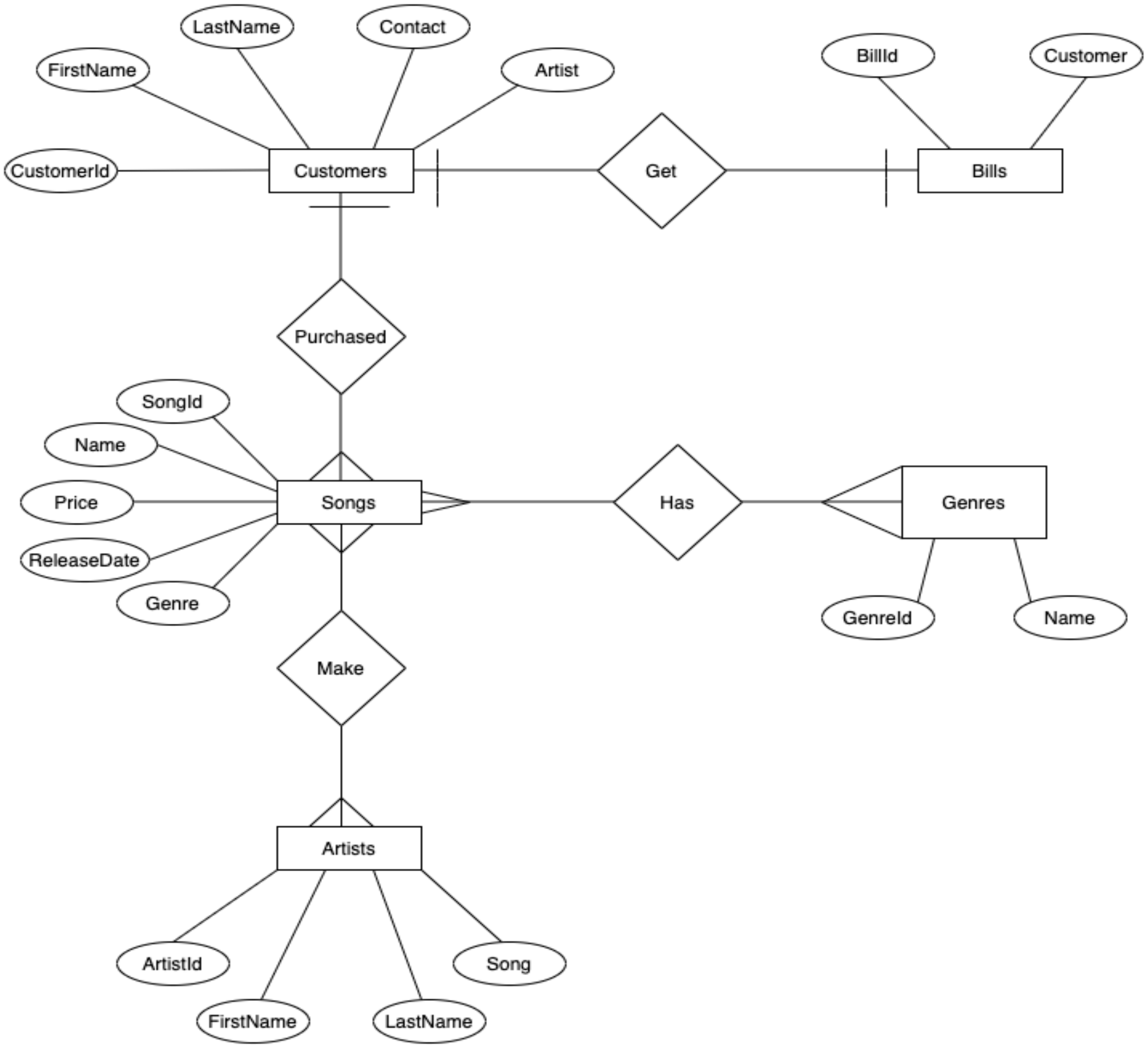


Figure 6 of ERD (Entity Relation Diagram)

In the above figure, an ERD of database MusicStore is shown as ERD stands for Entity Relationship Diagram. In the figure, each entities are inter-connected with each other and each entities has attributes which differs from one another.

e. Relational Database

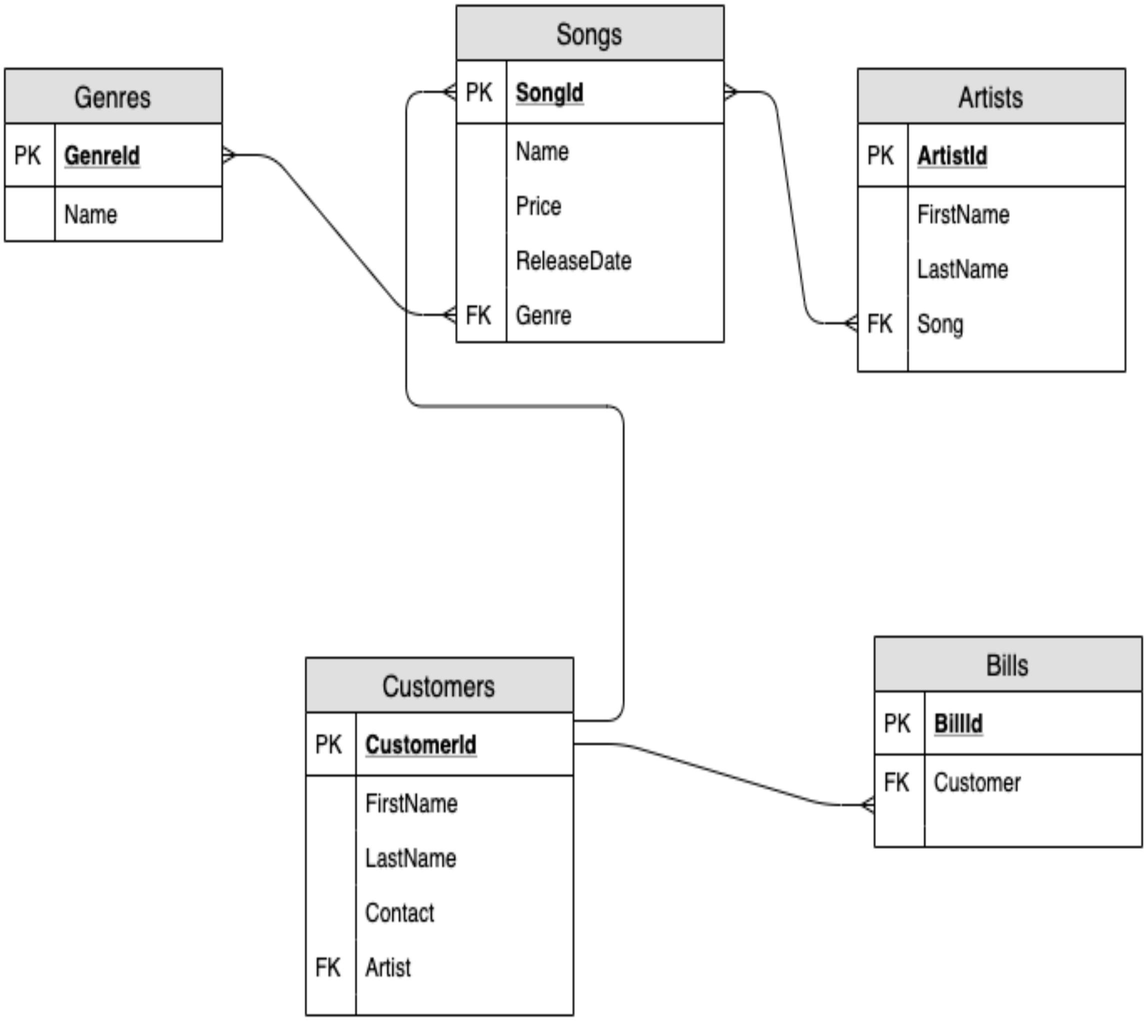


Figure 7 of RD (Relational Diagram)

The above figure is the RD (Relational Diagram) of the database MusicStore.

4. DATA DICTIONARY

A data dictionary is a collection or set of data containing database which gives the information of other data. According to the type of database, One can input different suitable attributes which can help to define their database. Data dictionary can also help an organization to operate smoothly by storing all the data which are necessary for smooth operation. (Anon., 2011), (Anon., 2005) The data dictionary of MusicStore attributes are given below:

a. Genres

Entity Name	Entity Description	Column Name	Column Description	Data Type	Length	Primary Key	Foreign Key	Nullable	Unique	Notes
Genres	Category	GenreId	Id for unique identification	INT	11	True	False	False	False	Auto Incremented
		Name	Identification of anything	VARCHAR	225	False	False	True	True	

Table 11 Data Dictionary of Genres

b. Songs

Entity Name	Entity Description	Column Name	Column Description	Data Type	Length	Primary Key	Foreign Key	Nullable	Unique	Notes
Songs	Collection of music	SongId	Id for unique identification	INT	11	True	False	False	False	Auto incremented
		Name	Identification of anything	VARCHAR	225	False	False	True	False	
		Price	Amount of money	INT	11	False	False	True	False	
		Genre	category	INT	11	False	True	True	False	References to GenreId of Genres table
		Release Date	Published time	DATE		False	False	True	False	

Table 12 Data Dictionary of Songs

c. Artists

Entity Name	Entity Description	Column Name	Column Description	Data Type	Length	Primary Key	Foreign Key	Nullable	Unique	Notes
Artists	People who write or sing songs	ArtistId	Id for unique identification	INT	11	True	False	False	False	Auto incremented
		FirstName	Way of identifying a person	VARCHAR	225	False	False	True	False	
		LastName	Surname of a person	VARCHAR	225	False	False	True	False	
		Song	Collection of music	INT	11	False	True	True	False	References to SongId of Songs table

Table 13 Data Dictionary of Artists

d. Customers

Entity Name	Entity Description	Column Name	Column Description	Data Type	Length	Primary Key	Foreign Key	Nullable	Unique	Notes
Customers	Person who buys goods	CustomerId	Id for unique identification	INT	11	True	False	False	False	Auto incremented
		FirstName	Way of identifying a person	VARCHAR	225	False	False	True	False	
		LastName	Surname of a person	VARCHAR	225	False	False	True	False	
		Contact	Way of communication	INT	11	False	False	True	True	
		Artist	Person who write or sing songs	INT	11	False	True	True	False	References to ArtistId of Artists table

Table 14 Data Dictionary of Customers

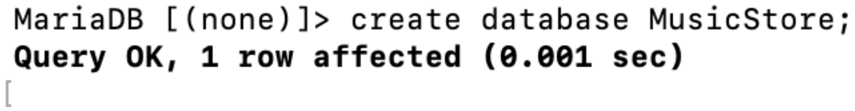
e. Bills

Entity Name	Entity Description	Column Name	Column Description	Data Type	Length	Primary Key	Foreign Key	Nullable	Unique	Notes
Bills	Receipt got after buying goods	BillId	Id for unique identification	INT	11	True	False	False	False	Auto incremented
		Customer	Person who buys goods	INT	11	False	True	True	False	References to Customer Id of Customers table

Table 15 Data Dictionary of Bills

5. DATABASE CREATION

I have created a database named MusicStore using XAMPP. Following are the process I followed to create a database. They are:



```
MariaDB [(none)]> create database MusicStore;
Query OK, 1 row affected (0.001 sec)
```

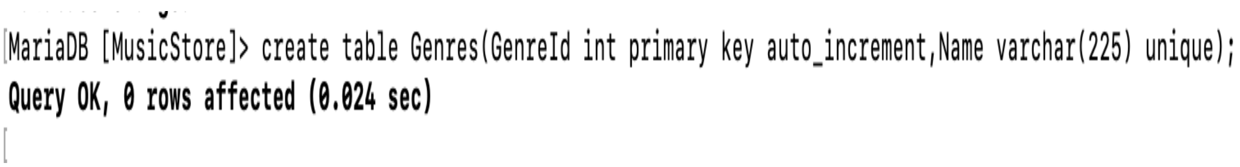
Figure 8 Database creation MusicStore

This above database has five entities. They are:

a. Genres
b. Songs
c. Artists
d. Customers
e. Bills

The process for creating each entity (table) are:

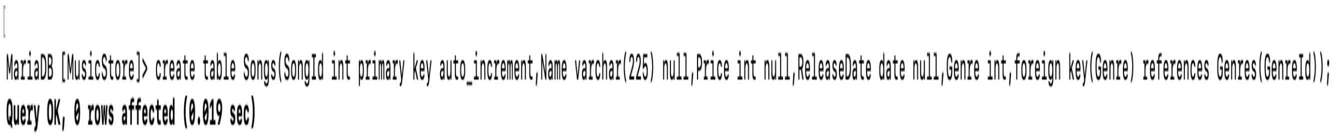
a. Genres



```
MariaDB [MusicStore]> create table Genres(GenreId int primary key auto_increment,Name varchar(225) unique);
Query OK, 0 rows affected (0.024 sec)
```

Figure 9 Creating Genres table

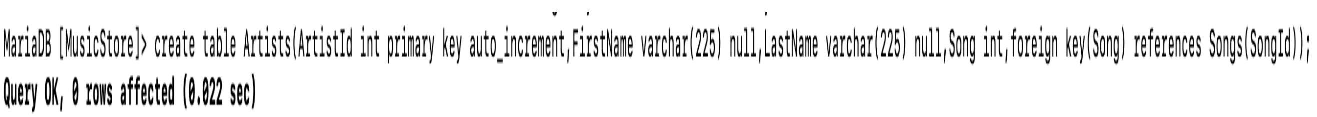
b. Songs



```
MariaDB [MusicStore]> create table Songs(SongId int primary key auto_increment,Name varchar(225) null,Price int null,ReleaseDate date null,Genre int,foreign key(Genre) references Genres(GenreId));
Query OK, 0 rows affected (0.019 sec)
```

Figure 10 Creating Songs table

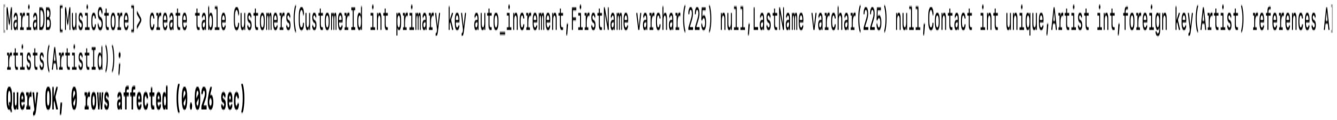
c. Artists



```
MariaDB [MusicStore]> create table Artists(ArtistId int primary key auto_increment,FirstName varchar(225) null,LastName varchar(225) null,Song int,foreign key(Song) references Songs(SongId));
Query OK, 0 rows affected (0.022 sec)
```

Figure 11 Creating Artists table

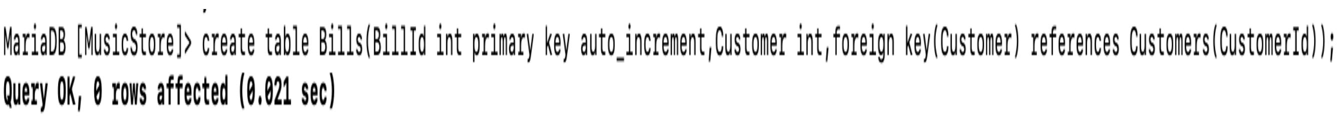
d. Customers



```
[MariaDB [MusicStore]> create table Customers(CustomerId int primary key auto_increment,FirstName varchar(225) null,LastName varchar(225) null,Contact int unique,Artist int,foreign key(Artist) references A]
rtists(ArtistId));
Query OK, 0 rows affected (0.026 sec)
```

Figure 12 Creating Customers table

e. Bills

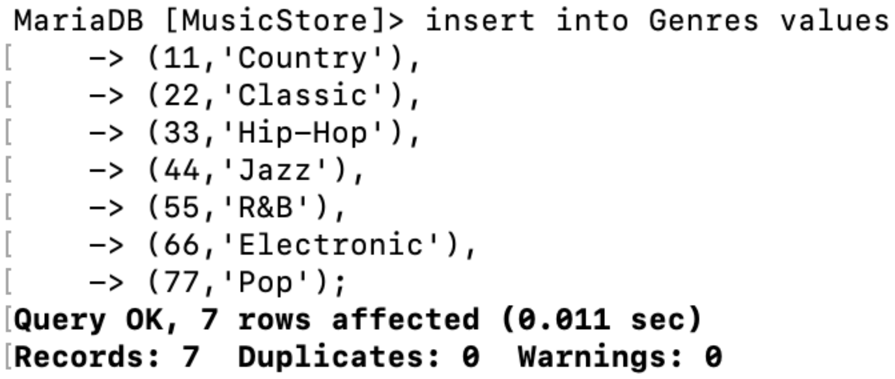


```
MariaDB [MusicStore]> create table Bills(BillId int primary key auto_increment,Customer int,foreign key(Customer) references Customers(CustomerId));
Query OK, 0 rows affected (0.021 sec)
```

Figure 13 Creating Bills table

The process for inserting values in each table are:

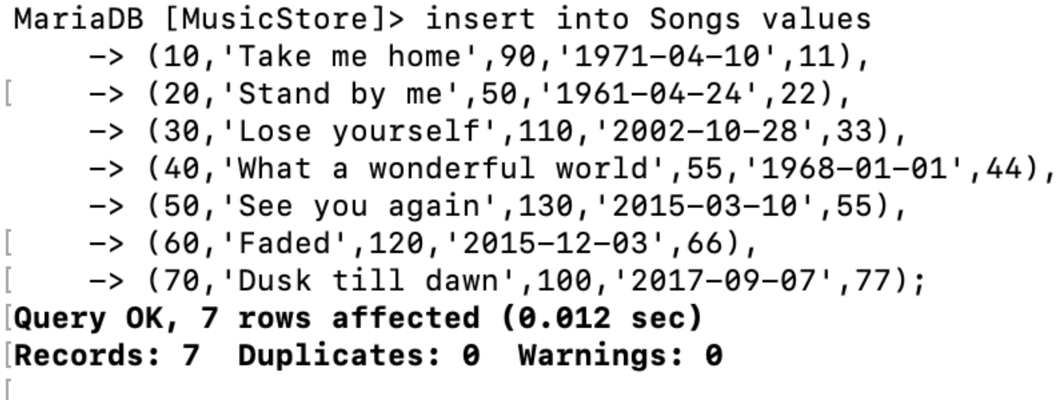
a. Genres



```
 MariaDB [MusicStore]> insert into Genres values
[    -> (11,'Country'),
[    -> (22,'Classic'),
[    -> (33,'Hip-Hop'),
[    -> (44,'Jazz'),
[    -> (55,'R&B'),
[    -> (66,'Electronic'),
[    -> (77,'Pop');
[Query OK, 7 rows affected (0.011 sec)
[Records: 7  Duplicates: 0  Warnings: 0
```

Figure 14 Inserting values in Genres

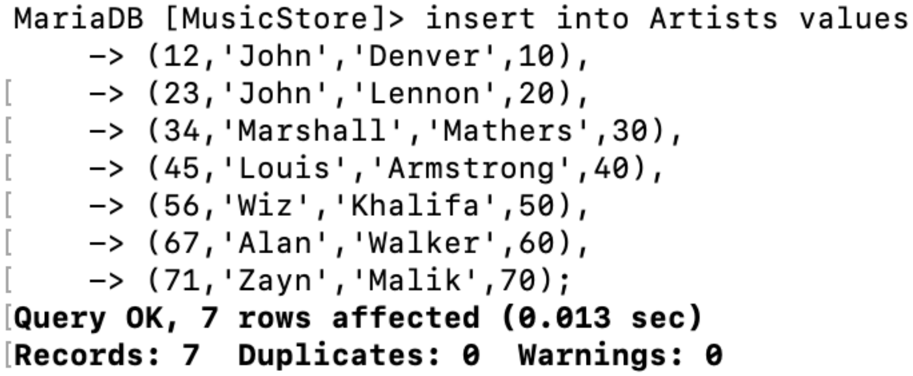
b. Songs



```
 MariaDB [MusicStore]> insert into Songs values
     -> (10,'Take me home',90,'1971-04-10',11),
[    -> (20,'Stand by me',50,'1961-04-24',22),
     -> (30,'Lose yourself',110,'2002-10-28',33),
     -> (40,'What a wonderful world',55,'1968-01-01',44),
     -> (50,'See you again',130,'2015-03-10',55),
[    -> (60,'Faded',120,'2015-12-03',66),
[    -> (70,'Dusk till dawn',100,'2017-09-07',77);
[Query OK, 7 rows affected (0.012 sec)
[Records: 7  Duplicates: 0  Warnings: 0
[
```

Figure 15 Inserting values in Songs

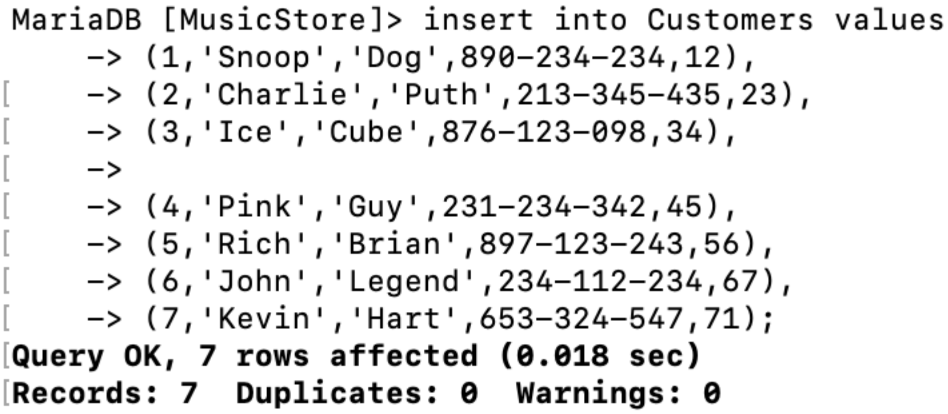
c. Artists



```
 MariaDB [MusicStore]> insert into Artists values
     -> (12,'John','Denver',10),
[    -> (23,'John','Lennon',20),
[    -> (34,'Marshall','Mathers',30),
[    -> (45,'Louis','Armstrong',40),
[    -> (56,'Wiz','Khalifa',50),
[    -> (67,'Alan','Walker',60),
[    -> (71,'Zayn','Malik',70);
[Query OK, 7 rows affected (0.013 sec)
[Records: 7  Duplicates: 0  Warnings: 0
```

Figure 16 Inserting values in Artists

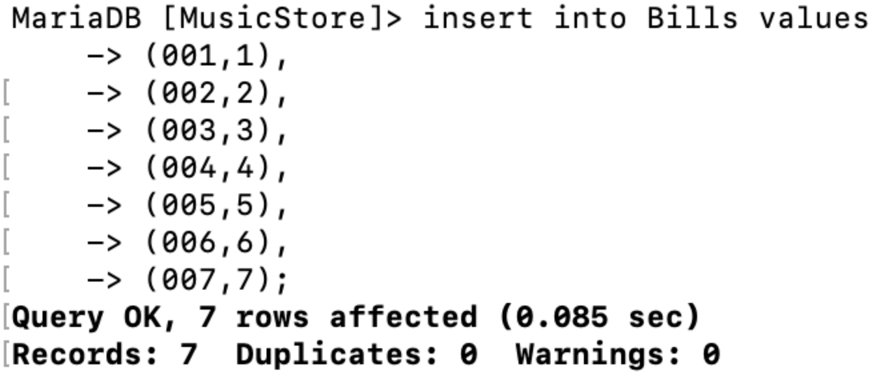
d. Customers



```
 MariaDB [MusicStore]> insert into Customers values
     -> (1,'Snoop','Dog',890-234-234,12),
[    -> (2,'Charlie','Puth',213-345-435,23),
[    -> (3,'Ice','Cube',876-123-098,34),
[    ->
[    -> (4,'Pink','Guy',231-234-342,45),
[    -> (5,'Rich','Brian',897-123-243,56),
[    -> (6,'John','Legend',234-112-234,67),
[    -> (7,'Kevin','Hart',653-324-547,71);
[Query OK, 7 rows affected (0.018 sec)
[Records: 7  Duplicates: 0  Warnings: 0
```

Figure 17 Inserting values in Customers

e. Bills



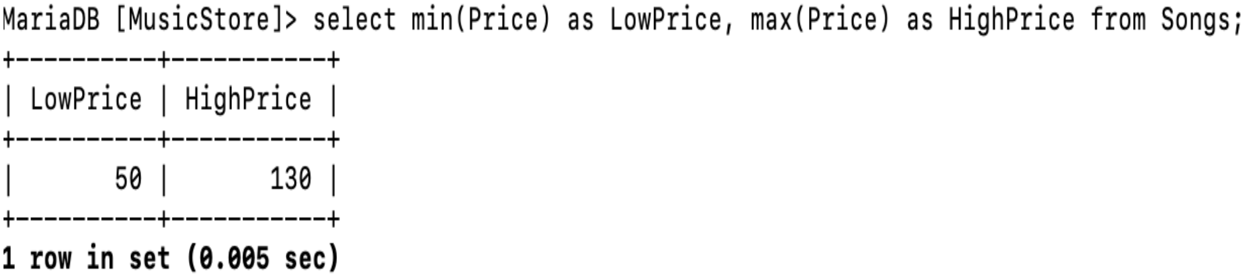
```
 MariaDB [MusicStore]> insert into Bills values
     -> (001,1),
[    -> (002,2),
[    -> (003,3),
[    -> (004,4),
[    -> (005,5),
[    -> (006,6),
[    -> (007,7);
[Query OK, 7 rows affected (0.085 sec)
[Records: 7  Duplicates: 0  Warnings: 0
```

Figure 18 Inserting values in Bills

6. DATABASE QUERIES

a. Max and Min

This query helps to display the high and low price of the items.

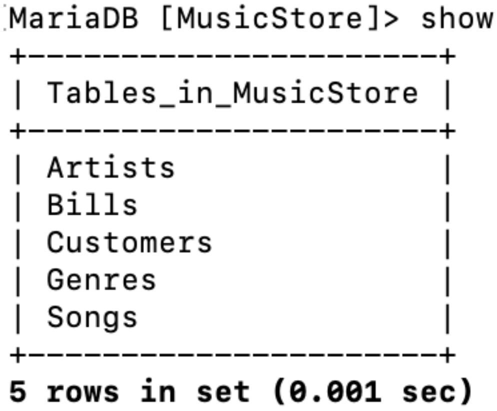


```
MariaDB [MusicStore]> select min(Price) as LowPrice, max(Price) as HighPrice from Songs;
+----------+-----------+
| LowPrice | HighPrice |
+----------+-----------+
|       50 |       130 |
+----------+-----------+
1 row in set (0.005 sec)
```

Figure 19 Using Max and Min query

b. Show tables

This query helps to display all the entities (table) of the database.

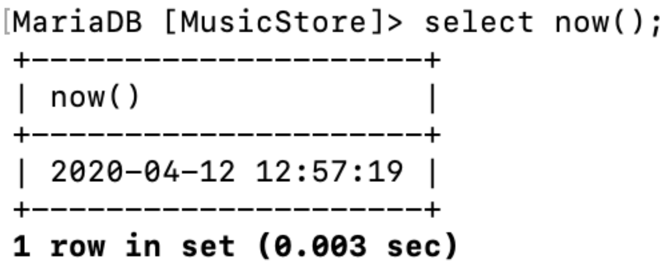


```
MariaDB [MusicStore]> show tables;
+----------------------+
| Tables_in_MusicStore |
+----------------------+
| Artists              |
| Bills                |
| Customers            |
| Genres               |
| Songs                |
+----------------------+
5 rows in set (0.001 sec)
```

Figure 20 Using Show tables query

c. Select now ()

   This query helps to display the current date and time.

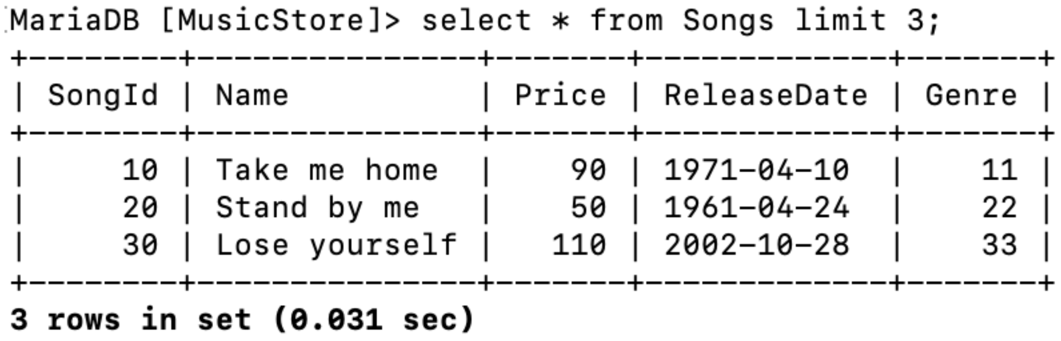


```
MariaDB [MusicStore]> select now();
+---------------------+
| now()               |
+---------------------+
| 2020-04-12 12:57:19 |
+---------------------+
1 row in set (0.003 sec)
```

Figure 21 Using Select now () query

d. Limit

   This query helps to display the limited number of data as per the user command.

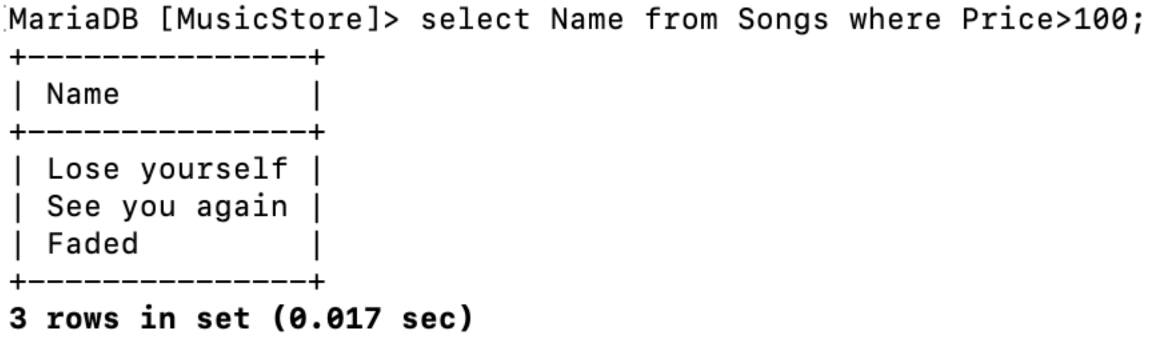


```
MariaDB [MusicStore]> select * from Songs limit 3;
+--------+---------------+-------+-------------+-------+
| SongId | Name          | Price | ReleaseDate | Genre |
+--------+---------------+-------+-------------+-------+
|     10 | Take me home  |    90 | 1971-04-10  |    11 |
|     20 | Stand by me   |    50 | 1961-04-24  |    22 |
|     30 | Lose yourself |   110 | 2002-10-28  |    33 |
+--------+---------------+-------+-------------+-------+
3 rows in set (0.031 sec)
```

Figure 22 Using Limit query

e. Where

This query helps to display the data which the user requires.

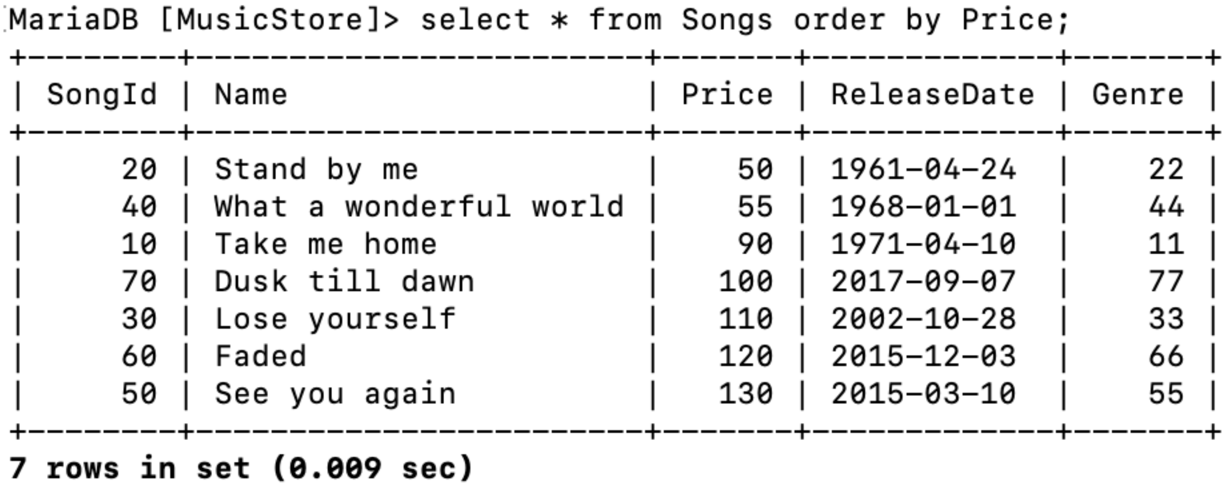


```
MariaDB [MusicStore]> select Name from Songs where Price>100;
+---------------+
| Name          |
+---------------+
| Lose yourself |
| See you again |
| Faded         |
+---------------+
3 rows in set (0.017 sec)
```

Figure 23 Using Where query

f. Order By

This query helps to display the data in ascending order.

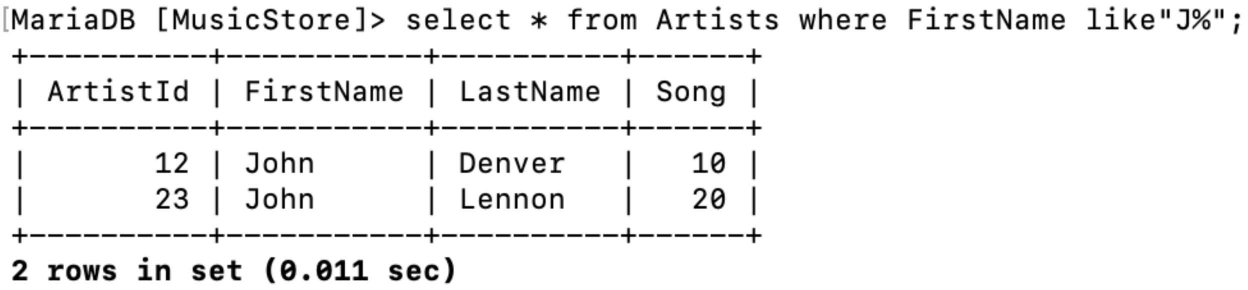


```
MariaDB [MusicStore]> select * from Songs order by Price;
+--------+------------------------+-------+-------------+-------+
| SongId | Name                   | Price | ReleaseDate | Genre |
+--------+------------------------+-------+-------------+-------+
|     20 | Stand by me            |    50 | 1961-04-24  |    22 |
|     40 | What a wonderful world |    55 | 1968-01-01  |    44 |
|     10 | Take me home           |    90 | 1971-04-10  |    11 |
|     70 | Dusk till dawn         |   100 | 2017-09-07  |    77 |
|     30 | Lose yourself          |   110 | 2002-10-28  |    33 |
|     60 | Faded                  |   120 | 2015-12-03  |    66 |
|     50 | See you again          |   130 | 2015-03-10  |    55 |
+--------+------------------------+-------+-------------+-------+
7 rows in set (0.009 sec)
```

Figure 24 Using Order By query

g. Like

This query helps to display specific data which the user requires.

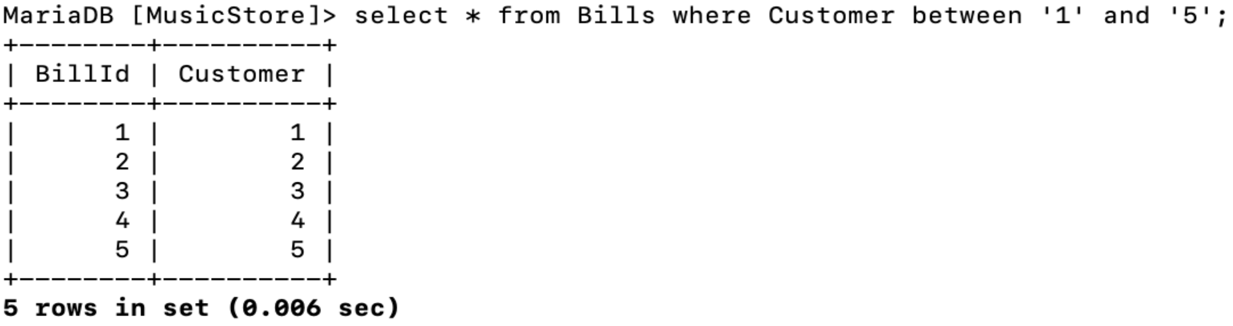


```
MariaDB [MusicStore]> select * from Artists where FirstName like"J%";
+----------+-----------+----------+------+
| ArtistId | FirstName | LastName | Song |
+----------+-----------+----------+------+
|       12 | John      | Denver   |   10 |
|       23 | John      | Lennon   |   20 |
+----------+-----------+----------+------+
2 rows in set (0.011 sec)
```

Figure 25 Using Like query

h. Between

This query helps to display the data within the required range.

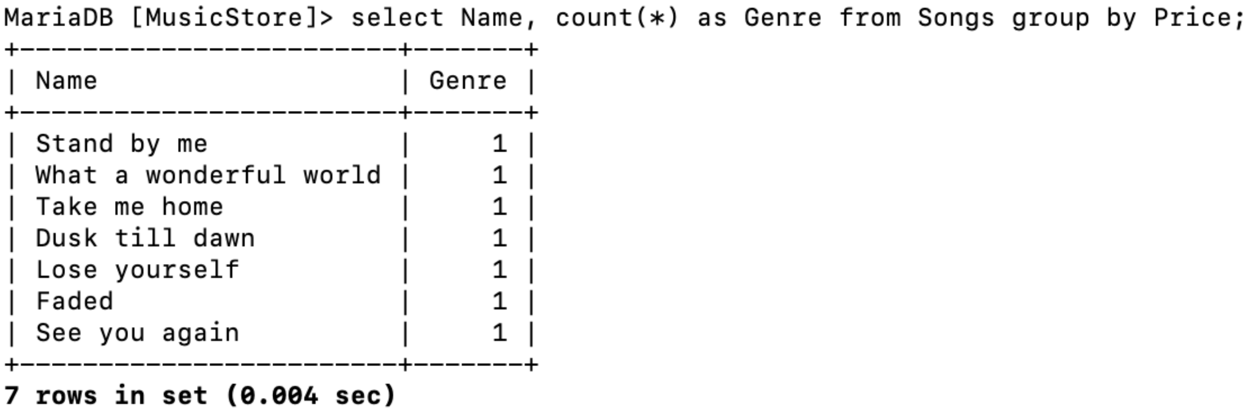


```
MariaDB [MusicStore]> select * from Bills where Customer between '1' and '5';
+--------+----------+
| BillId | Customer |
+--------+----------+
|      1 |        1 |
|      2 |        2 |
|      3 |        3 |
|      4 |        4 |
|      5 |        5 |
+--------+----------+
5 rows in set (0.006 sec)
```

Figure 26 Using Between query

i. Group By

This query helps to group the rows that have same value.

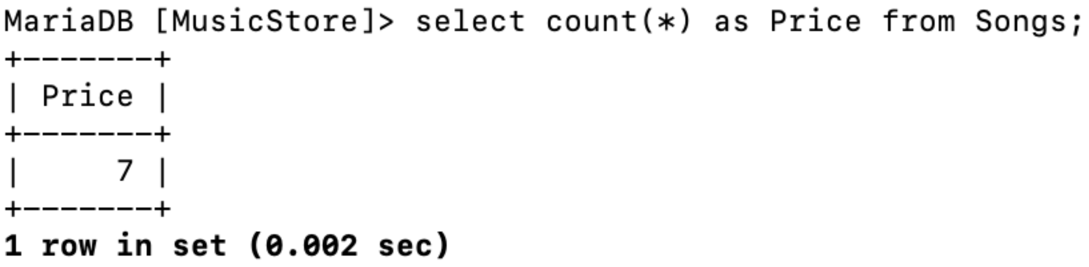


```
MariaDB [MusicStore]> select Name, count(*) as Genre from Songs group by Price;
+------------------------+-------+
| Name                   | Genre |
+------------------------+-------+
| Stand by me            |     1 |
| What a wonderful world |     1 |
| Take me home           |     1 |
| Dusk till dawn         |     1 |
| Lose yourself          |     1 |
| Faded                  |     1 |
| See you again          |     1 |
+------------------------+-------+
7 rows in set (0.004 sec)
```

Figure 27 Using Group By query

j. Count

This query helps to display the number of rows in the table.



```
MariaDB [MusicStore]> select count(*) as Price from Songs;
+-------+
| Price |
+-------+
|     7 |
+-------+
1 row in set (0.002 sec)
```

Figure 28 Using Count query

7. CONCLUSION

Finally, I completed my second coursework of IS. This coursework was easy to complete than first coursework. Thanks to our first coursework I didn’t had many problems completing it. As our first coursework covered 30% of our module marks, this our second coursework also covers 30% module mark.

Just like the first coursework we were again given similar questions. So, the database that I have created in this coursework is about MusicStore. This database has five different entities. They are: Genres, Songs, Artists, Customers and Bills. Each entity has different attributes. After that I created an ERD (Entity Relationship Diagram) of the database MusicStore. In the ERD, I defined about entity, attributes and their relation. Then I created a data dictionary of the database MusicStore describing each entities briefly. After the completion of data dictionary, I described the process I used for creating the database. After that I used different queries to check what changes would it bring to my database. Then I also described the roles of the queries which I had used. I also gave captions to all the tables and figures that I had used. Finally, I gave references to the site from where I had written some definations.

Even though I had less problems solving this coursework thanks to the first coursework. I still got some problems which I couldn’t solve. So, I asked some of my friends to help me in completing this coursework. Thanks to them I could understand the things which I couldn’t understand at first. I know that to become a successful person one must struggle like their life depended on it. So, I am gonna struggle till I get what I wanted. And at last, I wanna say that this coursework helped me in learning new things too even though I had problems at first.

8. Bibliography

Anon., 2005. *Techtarget.* [Online]
Available at: https://searchapparchitecture.techtarget.com/definition/data-dictionary
[Accessed 03 2020].

Anon., 2011. *Technopedia.* [Online]
Available at: https://www.techopedia.com/definition/27752/data-dictionary
[Accessed 03 2020].

Anon., 2017. *Technopedia.* [Online]
Available at: https://www.techopedia.com/definition/18702/data-model
[Accessed 03 2020].

Anon., 2017. *Technopedia.* [Online]
Available at: https://www.techopedia.com/definition/1200/entity-relationship-diagram-erd
[Accessed 03 2020].

Anon., 2019. *Tutorialspoint.* [Online]
Available at: https://www.tutorialspoint.com/dbms/dbms_data_models.htm
[Accessed 03 2020].

Beal, V., n.d. *Webopedia.* [Online]
Available at: https://www.webopedia.com/TERM/E/entity_relationship_diagram.html
[Accessed 03 2020].