Table of Contents

[**Part 1: Design**](#_pmyn1qt652bs) **2**

[Project Narrative](#_2f4ietx1oos8) 2

[Data Dictionary](#_fzcmqc3941ar) 2-3

[Data Questions](#_smbvkhqdxx11) 3-4

[Entity Relationship Diagram](#_hw31dhgo5bjh) 4

[Logical Model Diagram](#_jkexl3efnk2k) 5

[Normalized Model](#_6wf7rhx8omgg) 5

[**Part 2: Implementation**](#_spfj7uy2zkwz) **6**

[SQL DDL](#_2kprgnbxmo9w) 6

[SQL DML](#_wsdi3utu5iag) 7

[INSERT Statements](#_4trtr3797uke) 8

[Artist Table](#_oef9anykaxnj) 8

[Website Table](#_6itk4pih4983) 8-9

[Product Table](#_a4meu2cilof2) 9-10

[Performance Table](#_fg3wfiqzmbno) 10

[MusicTitle Table](#_m4ilmgp89zbc) 11

[MusicSales Table](#_j6429ypr4alq) 11

Data Manipulation statements 12

[Answers to Data Questions (SQL SELECT Statements)](#_2vclphkmrhup) 13-19

[GUI Prototype](#_wpwklx1y3r05) 20-23

[Reflection](#_mx5id0nsesu4) 23

**Part 1: Design**

## Project Narrative

Milk Boys Art House is a new, small concert hall, art gallery, and mini-store located right behind the University of Maryland’s main campus. Within its first year of opening, the Art House has hosted a number of bands. Andrew, the owner of the venue is a designer who sells custom-made Headbands, Posters, and T-shirts of the Art House’s mini store wants to streamline the planning of booking a solo musician by making it more efficient so he can focus on designing merchandise and running the venue. He has joined a deal with a few record labels that allows him to book solo artists from bands for a set price for the year 2019. Andrew hopes to create a database that makes the Musician Booking process more efficient. By creating a database that can show him all the artists he can book, the cost of booking them for a performance, the type of performer they are, where the artist is located. Additionally, he wants the database linked to a website called mymusic.com that will populate the artists banner page, contact information, image, and biography. He also wants to know which artists have posters and other merchandise he can buy and sell in his mini-store, alongside his own products. Overall, Andrew wants to see information pertaining to specific musicians in order to plan the atmosphere of the Art House for that event in order to make their performance more of an experience to attendees.

## Data Dictionary

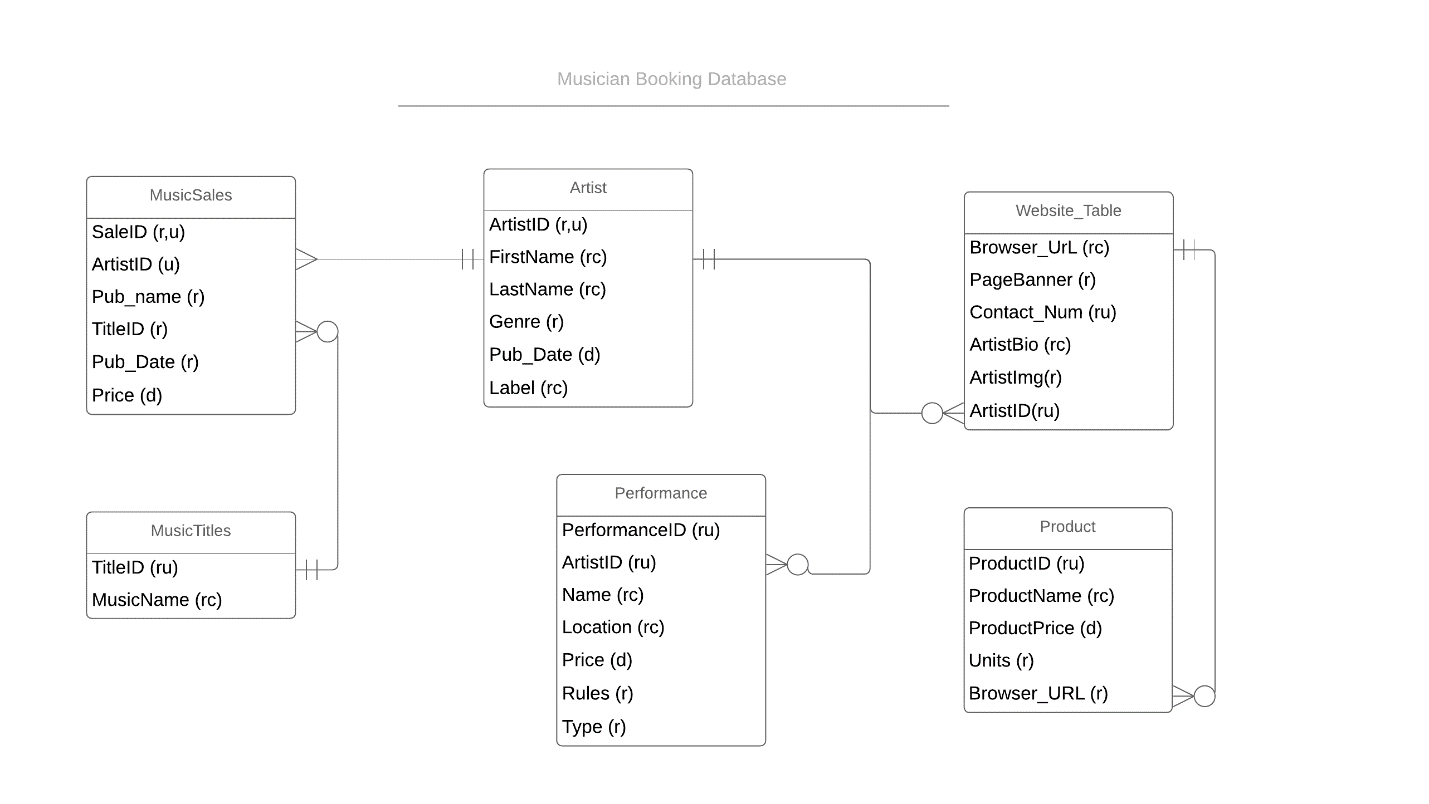
Working with Andrew, we have compiled a following list of entities and attributes he seeks in his database.

|  |  |  |
| --- | --- | --- |
| **Entity** | **Attribute** | **Properties** |
| Artist | ArtistID | Required. Unique |
|  | FirstName | Required. |
|  | LastName | Required |
|  | Genre | Required. From list of music genres |
|  | Label | Required. |
| Website | Browser\_URL | Required. Unique |
|  | ArtistID | Required. |
|  | PageBanner | Updated from website. |
|  | Contact\_Num | Required. |
|  | ArtistBio | Required.Updated from website |
|  | ArtistImg | Updated from website |
| Product | ProductID | Unique. Required |
|  | ProductName | Required |
|  | ProductPrice | Required. Price for single product |
|  | Units | Required. Sum of units available of product |
|  | Browser\_Url | Required. |
| Performance | PerformanceID | Required. Unique |
|  | ArtistID | Required |
|  | Name | Required. Name of the artists tour or performance |
|  | Location | Required |
|  | Price | Required. Price of single performance |
|  | Rules | Required.Rules of hosting said artist compiled. |
|  | TYPE | Compilation of the type of performance given by single artist. |
| MusicTitles | TitleID | Unique. Required |
|  | MusicName | Required. Single sample of artitsts song, |
| Music Sales | SaleID | Required.Unique |
|  | ArtistID | Required |
|  | Pub\_Name | Derived from a list of used music publisherrs |
|  | TitleID | Unique. Required |
|  | Price | Required. Sum of music sales for said artist |
|  | Pub\_Date | Date of musics released |

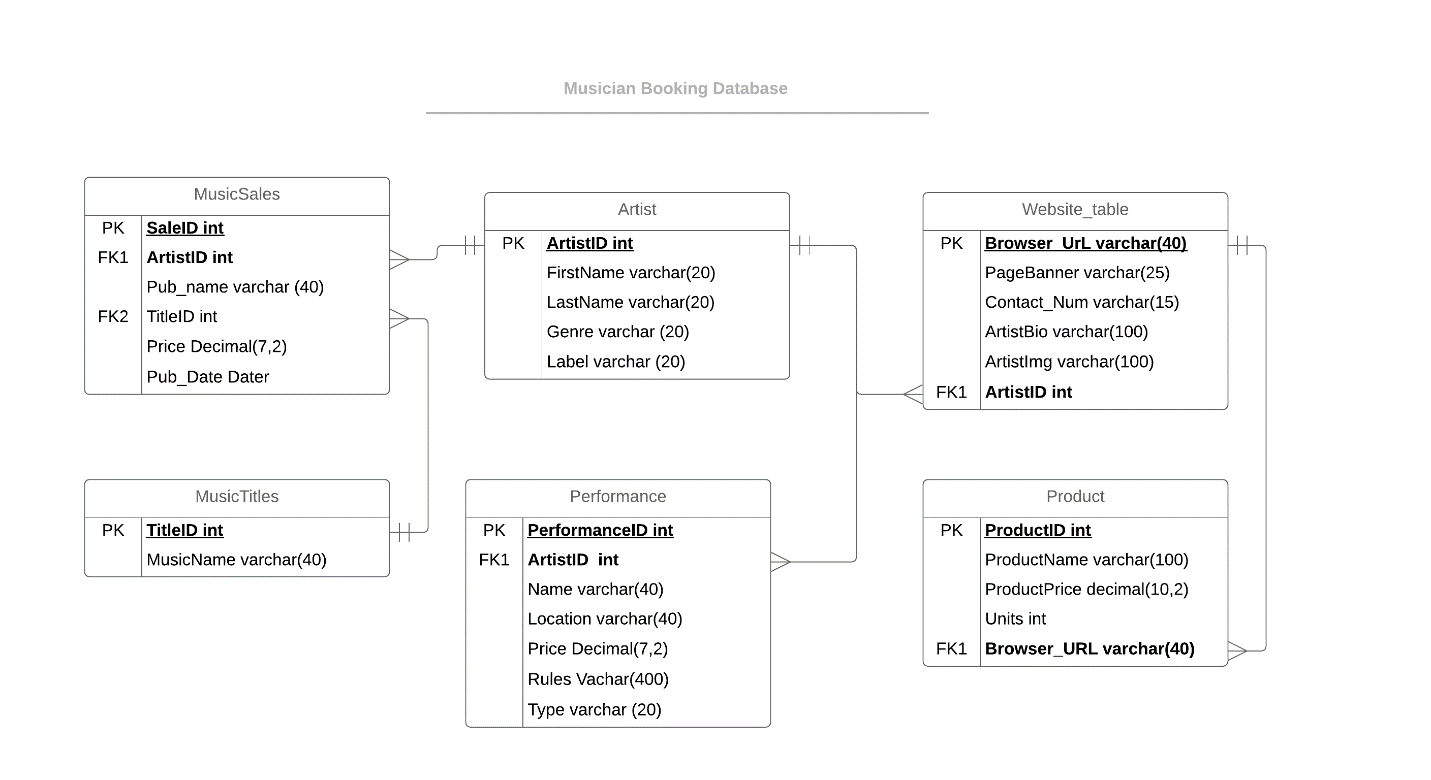
## Data Questions

1. What are the artists available to book, what is their musical genres, current location, and price per performance?
2. What artists have posters Milk Boy Art House can buy and sell? What is the quantity available, price per poster, and URL to the poster’s link.
3. What are the music sales of artists, and the date they published their songs. Also wish to include their label’s name and genre.
4. List of available performances ordered by the price of said performance and the type of performance given by artist.
5. List of songs with the artist and its published date.
6. What music sales are are over $1,000
7. Who are the artists with publicans released between 1/1/2001 and 12/31/2005
8. What are the average prices of the products sold.
9. Who are the artist who have never performed.
10. What is the contact information and bio for artist with products available.
11. What are the products with musician information and website URL
12. Who are the artist who have higher than average music sales. With their contact information and their records company
13. What artists don’t have an album listed in the music sales table?
14. What is the URL of the artists who are not able to perform this year?

# Entity Relationship Diagram



## Logical Model Diagram



## Normalized Model

The logical model above was able to be normalized in at least the third normal form, therefore, no additional normalization is needed.

Part 2: Implementation

### SQL DDL

--Create table artist

create table artist

(

ArtistID int identity,

FirstName VARCHAR (20) not null,

LastName VARCHAR (20) not null,

Genre VARCHAR (20) not null,

Label VARCHAR (20) not null,

--Create constraint on table artist

constraint PK\_artist PRIMARY KEY(ArtistID)

)

--End creating table for artist

--Create table for website

create table website\_table

(

Browser\_URL VARCHAR(40),

PageBanner VARCHAR (25),

Contact\_Num VARCHAR(15),

ArtistBio VARCHAR(100),

ArtistImg VARCHAR(15),

ArtistID int NOT NULL,

--Create constraint on table artist

constraint PK\_BrowserURL primary key(Browser\_URL),

constraint FKArtistID foreign key (ArtistID) references artist(ArtistID)

)

--end creating table for website

--create table for product

create table Product

(

ProductID int identity,

ProductName VARCHAR(100),

ProductPrice DECIMAL(10,2),

Units int,

Browser\_URL VARCHAR(40) NOT NULL,

--Create Constraints on table product

constraint PK\_Product primary key(ProductID),

constraint FK1\_Product foreign key (Browser\_URL) references website\_table(Browser\_URL)

)

--end creating table for product

--Create Table for Performances

create table Performances

(

PerformanceID int identity,

ArtistID int NOT NULL,

Name VARCHAR(40),

Location VARCHAR(40),

Price DECIMAL(7,2),

Rules VARCHAR(400),

--Create contraints for table performances

constraint PK\_Performances primary key(PerformanceID),

constraint FKPerformerID foreign key (ArtistID) references artist(ArtistID)

);

--End Creating table performance

--Create table MusicTitles

create table MusicTitles

(

TitleID int identity,

MusicName VARCHAR(40),

--Create constraints for table musictitles

constraint PK\_MusicTitles primary key(TitleID)

);

--End creating table Music Titles

--Create table Musicsales

create table MusicSales

(

SaleID int identity,

ArtistID int NOT NULL,

Pub\_Name VARCHAR(40),

TitleID int NOT NULL,

Price DECIMAL(7,2),

Pub\_Date DATE,

--create constraints music sales

constraint PK\_MusicSales primary key(SaleID),

constraint FK1\_MusicSales foreign key (ArtistID) references artist(ArtistID),

constraint FK2\_MusicSales foreign key (TitleID) references MusicTitles(TitleID)

);

--end creating Music Sales

### SQL DML

--Alter performances table column name (due to error in initital entry/execution)

alter table dbo.Performances

ADD TYPE varchar(20);

--Insert data---

--insert data into Artist table--

insert into artist (FirstName,LastName,Genre,Label) values ('Billy','Power','Punk','Tooth & Nail Records');

insert into artist (FirstName,LastName,Genre,Label) values ('Ed','Power','Punk','Tooth & Nail Records');

insert into artist (FirstName,LastName,Genre,Label) values ('Stephen','Keech','Metalcore','Solid State');

insert into artist (FirstName,LastName,Genre,Label) values ('Jimmy','Ryan','Christian metal','Solid State');

insert into artist (FirstName,LastName,Genre,Label) values ('Nicholas','Moore','Post-hardcore','InVogue');

insert into artist (FirstName,LastName,Genre,Label) values('Anthony','Damschroder','Alternative metal','Rise');

insert into artist (FirstName,LastName,Genre,Label) values ('Larry','Farkas','Christian metal','Vengeance Rising');

insert into artist (FirstName,LastName,Genre,Label) values ('Eli','Prinsen','Christian metal','Intense');

insert into artist (FirstName,LastName,Genre,Label) values ('Bruce','Swift','heavy metal','Intense');

insert into artist (FirstName,LastName,Genre,Label) values ('Gleen','Mancaruso','christian rock','Joshua');

--query check on artist table--

select \* from artist;

--Insert into website table

insert into website\_table(Browser\_URL,PageBanner,Contact\_Num,ArtistBio,ArtistImg,ArtistID) values

('http://www.mymusics.com/billypower','ToothNail Banner','9123412345','He is member of Blenderhead band','billy.jpg','2');

insert into website\_table(Browser\_URL,PageBanner,Contact\_Num,ArtistBio,ArtistImg,ArtistID) values

('http://www.mymusics.com/edcari','ToothNail Banner','7208912345','He is guitar player of Blenderhead band','edi.jpg','3');

insert into website\_table(Browser\_URL,PageBanner,Contact\_Num,ArtistBio,ArtistImg,ArtistID) values

('http://www.mymusics.com/stephen','SolidState Banner','7123412345','He is member of HASTE THE DAY','stephen.jpg','4');

insert into website\_table(Browser\_URL,PageBanner,Contact\_Num,ArtistBio,ArtistImg,ArtistID) values

('http://www.mymusics.com/jimmy','SolidState Banner','9468372567','He is member of HASTE THE DAY','jimmy.jpg','5');

insert into website\_table(Browser\_URL,PageBanner,Contact\_Num,ArtistBio,ArtistImg,ArtistID) values

('www.facebook.com/beforetheireyesband','Before there Eyes Banner','9198972567','He is member of Before there eyes post hardcore band','moore.jpg','6');

insert into website\_table(Browser\_URL,PageBanner,Contact\_Num,ArtistBio,ArtistImg,ArtistID) values

('http://www.mymusics.com/anthony','Eyes Banner','7298972567','He is member of Before there eyes post hardcore band','anthony.jpg','7');

insert into website\_table(Browser\_URL,PageBanner,Contact\_Num,ArtistBio,ArtistImg,ArtistID) values

('http://www.mymusics.com/larry','DieBanner','7298971234','He is member of Die happy band','larry.jpg','8');

insert into website\_table(Browser\_URL,PageBanner,Contact\_Num,ArtistBio,ArtistImg,ArtistID) values

('http://www.mymusics.com/eli','sacredWarriorBanner','9256781234','He is member of Sacred Warrior band','eli.jpg','9');

insert into website\_table(Browser\_URL,PageBanner,Contact\_Num,ArtistBio,ArtistImg,ArtistID) values

('http://www.mymusics.com/bruce','sacredWarriorBanner','5676781234','He is member of Sacred Warrior band','bruce.jpg','10');

insert into website\_table(Browser\_URL,PageBanner,Contact\_Num,ArtistBio,ArtistImg,ArtistID) values

('http://www.mymusics.com/gleen','DieBanner','7256781234','He is member of Die happy band','gleen.jpg','1');

--query the website table

select \* from website\_table

--Insert data into Product table

insert into Product(ProductName,ProductPrice,Units,Browser\_URL) values ('T-Shirt','50','10','http://www.mymusics.com/stephen');

insert into Product(ProductName,ProductPrice,Units,Browser\_URL) values ('T-Shirt','350','5','http://www.mymusics.com/larry');

insert into Product(ProductName,ProductPrice,Units,Browser\_URL) values ('Headband','100','15','http://www.mymusics.com/larry');

insert into Product(ProductName,ProductPrice,Units,Browser\_URL)values ('Headband','150','10','http://www.mymusics.com/billypower');

insert into Product(ProductName,ProductPrice,Units,Browser\_URL) values ('Poster','25','100','http://www.mymusics.com/billypower');

insert into Product(ProductName,ProductPrice,Units,Browser\_URL) values ('Poster','10','1000','http://www.mymusics.com/gleen');

insert into Product(ProductName,ProductPrice,Units,Browser\_URL)values ('Headband','50','100','http://www.mymusics.com/gleen');

insert into Product(ProductName,ProductPrice,Units,Browser\_URL) values ('T-Shirt','200','10','http://www.mymusics.com/gleen');

insert into Product(ProductName,ProductPrice,Units,Browser\_URL) values ('T-Shirt','250','100','http://www.mymusics.com/anthony');

insert into Product(ProductName,ProductPrice,Units,Browser\_URL) values ('T-Shirt','50','100','http://www.mymusics.com/eli');

--query product table

select \* from Product

--Insert data into performances table

insert into Performances(TYPE,ArtistID,Name,Location,Price,Rules) values ('Vocals','2','Ninety Pound Wuss','America','100','Enjoy Tooth and Nail songs by Billy');

insert into Performances(TYPE,ArtistID,Name,Location,Price,Rules) values ('Bass','2','Roadside Monument','America','1000','Enjoy Tooth and Nail songs by Billy');

insert into Performances(TYPE,ArtistID,Name,Location,Price,Rules) values ('Guitar','3',' Ninety Pound Wuss','New York','500','Enjoy Tooth and Nail songs by Carrigan');

insert into Performances(TYPE,ArtistID,Name,Location,Price,Rules) values ('lead-vocals','4','The Anti–Mother Tour','Indianapolis','500','Enjoy Haste the days songs by Keech');

insert into Performances(TYPE,ArtistID,Name,Location,Price,Rules) values ('clean-vocals','5','Napalm & Noise Tour','Indianapolis','700','Enjoy Haste the days songs by Jimmy');

insert into Performances(TYPE,ArtistID,Name,Location,Price,Rules) values ('lead singer','6','Spring Break Your Heart Tour','California','2000','Enjoy BEFORE THE EYES songs by Nicholas Moore');

insert into Performances(TYPE,ArtistID,Name,Location,Price,Rules) values ('Drumer','7','Shipwreck in the sand Tour','New York','1000','Enjoy BEFORE THE EYES songs by Anthony Damschroder');

insert into Performances(TYPE,ArtistID,Name,Location,Price,Rules) values ('Drum and Guitar','7','One Moment Management Tour','Oceana','3000','Enjoy BEFORE THE EYES songs by Anthony Damschroder');

insert into Performances(TYPE,ArtistID,Name,Location,Price,Rules) values ('Guitarist','4','Saints and Sinners Tour','Johannesburg','1500','Enjoy Haste the day songs by Keech');

insert into Performances(TYPE,ArtistID,Name,Location,Price,Rules) values ('Keyboard and vocal','5','Scream the Prayer Tour','Cape town','2500','Enjoy Haste the day songs by Jimmy');

--Query Performance table

select \* from Performances

--Insert data into MusicTitle table

insert into MusicTitles(MusicName) values ('Prime Candidate for Burnout');

insert into MusicTitles(MusicName) values ('Burning Bridges');

insert into MusicTitles(MusicName) values ('When Everything Falls');

insert into MusicTitles(MusicName) values ('Pressure the Hinges');

insert into MusicTitles(MusicName) values ('Its a Punk Rock Christmas');

insert into MusicTitles(MusicName) values ('The Dawn of My Death');

insert into MusicTitles(MusicName) values ('Untouchable');

insert into MusicTitles(MusicName) values ('The Hi-Fi Horizon');

insert into MusicTitles(MusicName) values ('Rebellion');

insert into MusicTitles(MusicName) values ('Visions of Hell');

--query music titles

select \* from MusicTitles

--insert data into music sales

insert into MusicSales(ArtistID,PUB\_NAME,TITLEID,PRICE,PUB\_DATE) values ('2','Tooth & Nail Records','1','200.50',('19940117'));

insert into MusicSales(ArtistID,PUB\_NAME,TITLEID,PRICE,PUB\_DATE) values ('4',' Solid State Records','2','500',('20040527'));

insert into MusicSales(ArtistID,PUB\_NAME,TITLEID,PRICE,PUB\_DATE) values ('4',' Solid State Records','3','600',('20050610'));

insert into MusicSales(ArtistID,PUB\_NAME,TITLEID,PRICE,PUB\_DATE) values ('5',' Solid State Records','4','250',('20070223'));

insert into MusicSales(ArtistID,PUB\_NAME,TITLEID,PRICE,PUB\_DATE) values ('5',' Solid State Records','5','750',('20081127'));

insert into MusicSales(ArtistID,PUB\_NAME,TITLEID,PRICE,PUB\_DATE) values ('6','InVogue Records','6','750',('20081028'));

insert into MusicSales(ArtistID,PUB\_NAME,TITLEID,PRICE,PUB\_DATE) values ('7','InVogue Records','7','1050',('20100316'));

insert into MusicSales(ArtistID,PUB\_NAME,TITLEID,PRICE,PUB\_DATE) values ('7','InVogue Records','8','1500',('20121112'));

insert into MusicSales(ArtistID,PUB\_NAME,TITLEID,PRICE,PUB\_DATE) values ('8','Master Command Records','9','1000',('20011207'));

insert into MusicSales(ArtistID,PUB\_NAME,TITLEID,PRICE,PUB\_DATE) values ('9','Die Happy Records','10','400',('20000807'));

--query music sales table

select \* from MusicSales

--Queries for data manipulation

/\* Select all columns and all rows from one table \*/

select \* from artist;

/\* Select 5 columns and all rows from one table \*/

select ArtistID, FirstName, LastName, Genre, Label

from artist;

/\*Create views\*/

--Create view called BuyPoster

Create view BuyPoster\_view as

select \*

from dbo.Product

where ProductName='Poster';

--Query View BuyPoster

Select

\* from dbo.Product

where ProductName='Poster';

--Create Performers\_View to see the name and information of performers

create view Performers\_view as

select p.Name,p.Location,p.Price,a.FirstName,a.LastName,a.artistID

from Performances p inner join artist a

on p.artistID=a.artistID;

--Query performers view

Select

p.Name,p.Location,p.Price,a.FirstName,a.LastName,a.artistID

from Performances p inner join artist a

on p.artistID=a.artistID

order by artistID asc;

/\*Need to update the data \*/

--query before the update

select \* from MusicSales;

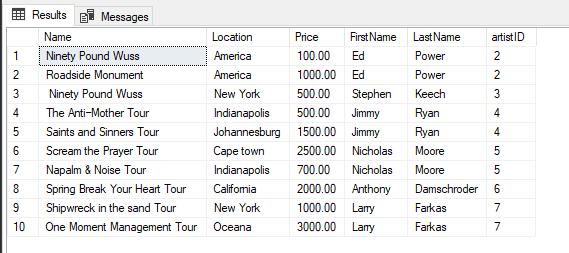
--update price by increasing 5\* price already set for music sales

update musicsales set price=price+5\*price;

select \* from MusicSales;

### Answers to Data Questions:

1. What are the artists available to book, what is their musical genres, current location, and price per performance?



--Create Performers\_View to see the name and information of performers

create view Performers\_view as

select p.Name,p.Location,p.Price,a.FirstName,a.LastName,a.artistID

from Performances p inner join artist a

on p.artistID=a.artistID;

--Query performers view

Select

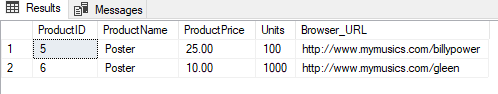
p.Name,p.Location,p.Price,a.FirstName,a.LastName,a.artistID

from Performances p inner join artist a

on p.artistID=a.artistID

order by artistID asc;

1. What artists has have posters Milk Boy Art House can buy and sell? What is the quantity available, price per poster, and URL to the poster’s link.



--Create view called BuyPoster

Create view BuyPoster\_view as

select \*

from dbo.Product

where ProductName='Poster';

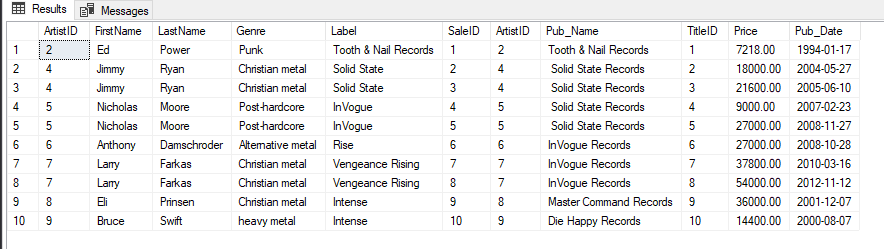
--Query View BuyPoster

Select

\* from dbo.Product

where ProductName='Poster';

1. What are the music sales of artists, and the date they published their songs. Also wish to include their label’s name and genre.



/\* What are the music sales of artists, and the date they published their songs.

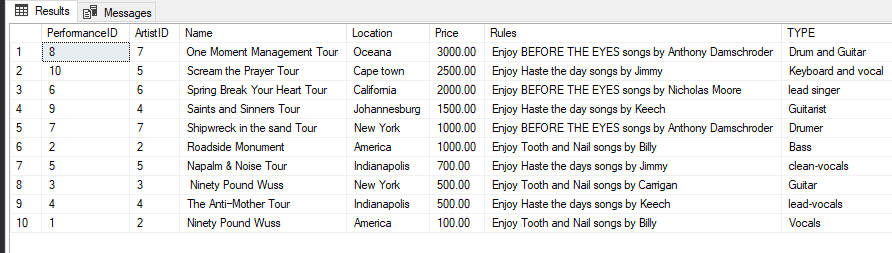
Also wish to include their label’s name and genre \*/

select artist.ArtistID, artist.FirstName, artist.LastName, Genre, Label, MusicSales.SaleID,

musicsales.ArtistID, Pub\_Name, TitleID, Price, Pub\_Date

from artist inner join musicsales on artist.ArtistID= musicsales.ArtistID;;

1. List of available performances, ordered by the price of said performance and the type of performance given by artist.



/\* List of available performances, ordered by the price of said

performance and the type of performance given by artist. \*/

select \* from performances order by Price desc;

1. List of songs with the artist and its published date.



/\* list of songs with artist and published daate \*/

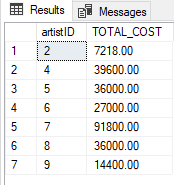
select artist.artistID,firstname,pub\_name,musicname,pub\_date

from (musicsales inner join artist on musicsales.artistID=artist.artistID) inner join musictitles on

Musictitles.titleID=musicsales.titleID

Where MusicSales.TitleID<11;

1. What music sales are are over $1,000

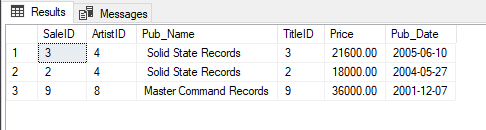


/\* list artists with music sales above 1000\*/

select artistID,SUM(Price) AS TOTAL\_COST

from musicsales GROUP BY artistID having SUM(price)>1000;

1. Who are the artists with publicans released between 1/1/2001 and 12/31/2005



/\* What artists publications released between January 1, 2001 to December 31,2005? \*/

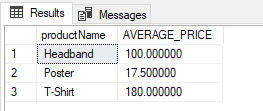
select \*

from musicsales

where pub\_date>('20010101') and pub\_date<('20051231')

order by Pub\_Date desc;

1. What are the average prices of the products sold.



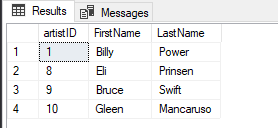
/\* aggregate function to calculate average price of products sold \*/

select productName,avg(productPrice) AS AVERAGE\_PRICE

from product

group by(productName);

1. Who are the artist who have never performed.



--Listing the ID and names of artists who had never performed\*/

SELECT artist.artistID, FirstName,LastName

FROM artist

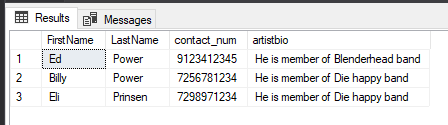
WHERE NOT EXISTS

(SELECT artistID

FROM performances

WHERE performances.artistID=artist.artistID);

1. What is the contact information and bio for artist with products available.



--Contact Info for the artists with more than one product in market.

SELECT FirstName , LastName,contact\_num,artistbio

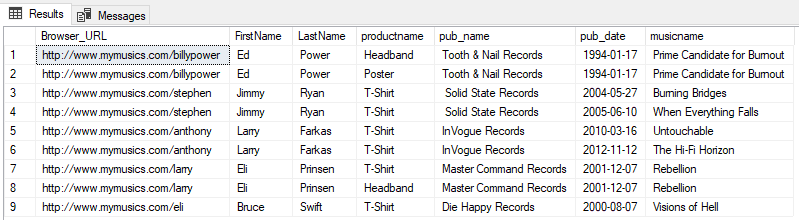
FROM WEBSITE\_TABLE INNER JOIN artist ON WEBSITE\_TABLE.artistID=artist.artistID

WHERE (SELECT count(\*) as count

FROM product

WHERE product.browser\_url=WEBSITE\_TABLE.browser\_url)>=2;

1. What are the products with muscian information and website URL



-- Using JOIN. What are the products with muscian information and website URL

SELECT website\_table.Browser\_URL, FirstName, LastName, productname, pub\_name,pub\_date,musicname

FROM (((website\_table

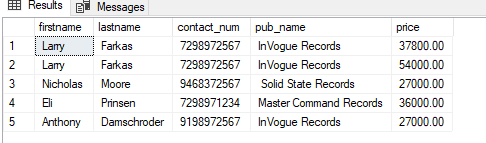
INNER JOIN artist on website\_table.artistID=artist.artistID)

INNER JOIN product on product.browser\_url=website\_table.browser\_url)

INNER JOIN musicsales on musicsales.artistID=website\_table.artistID)

INNER JOIN musictitles on musictitles.titleID=musicsales.titleID;

1. Who are the artist who have higher than average music sales. With their contact information and their records company



--List the publications of the artists that have music sales of more than average\*/

select firstname, lastname,contact\_num,pub\_name,price

from (artist INNER JOIN website\_table on artist.artistID=website\_table.artistID) INNER JOIN musicsales on musicsales.artistID=artist.artistID

where price> (select avg(price) from musicsales);

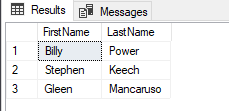
1. What artists don’t haave an album listed in the music sales table?

--List the first and last name of the Artists who don't have any album listed in the musicsales table.

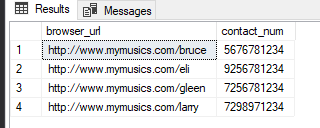
SELECT FirstName, LastName

FROM artist LEFT JOIN musicsales on musicsales.artistid=artist.artistID

WHERE musicsales.artistID is NULL;



1. What is the URL of the artists who are not able to perform this year?



--List browser urls and contacts of artists that are not performers this year\*/

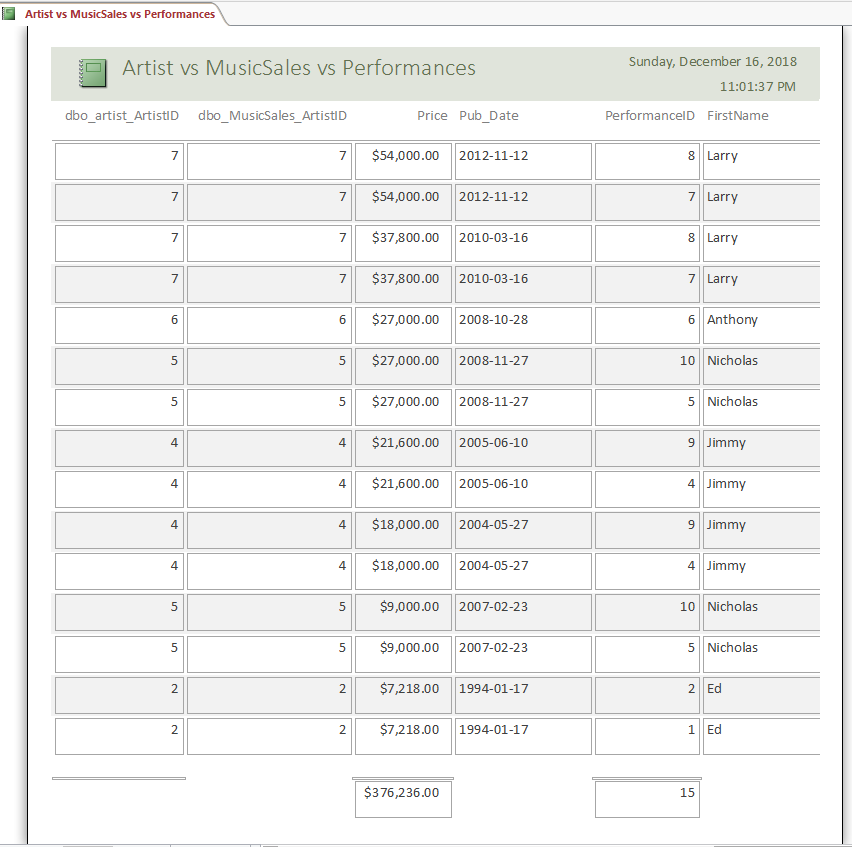
select browser\_url,contact\_num

from website\_table where artistID not in (select artaistID from performances);

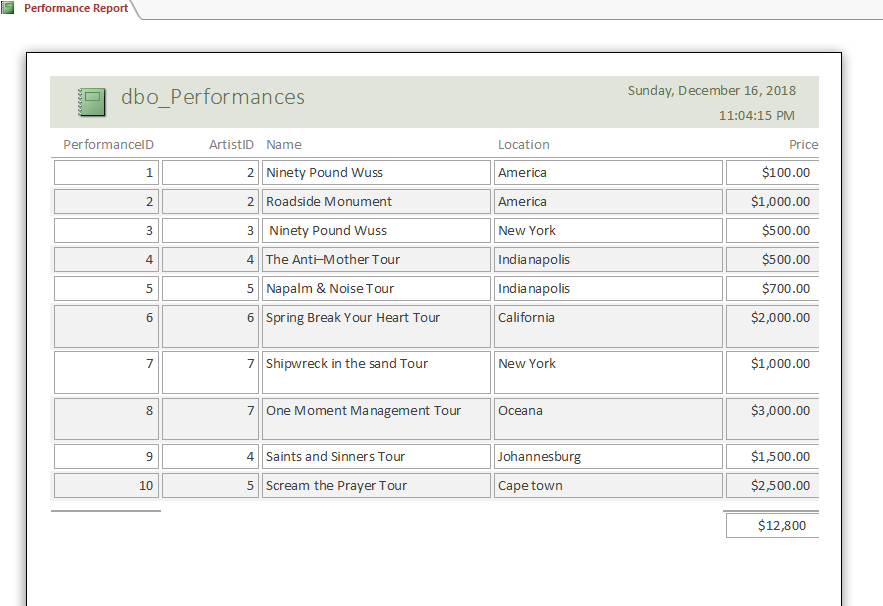
### GUI Protype:

I decided to use Access for my front end interface.

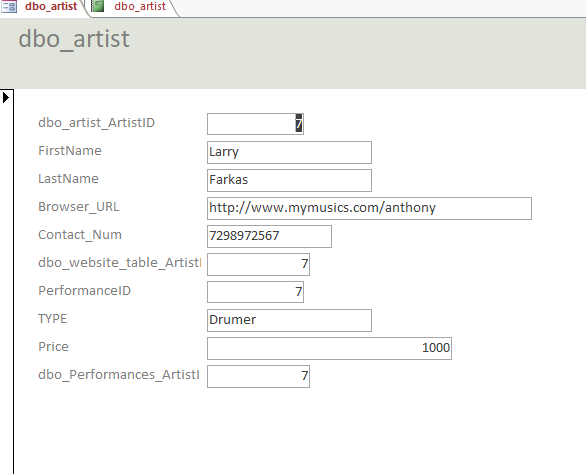
**Report that compares Artist to music sales and their cost of their music sales.**



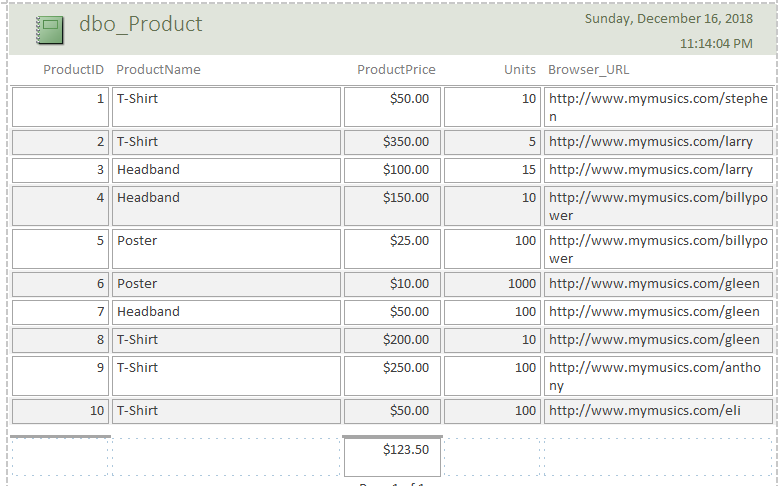
**A report that looks at artist performances and sums the total cost for the year if Andrew hired all the musicians once.**



**A form for artists that give all contact, performance type, and website information .**



**A product report for andrew to use when estimating average merchandise cost and products.**



### Reflection:

Creating a database to to help make booking bands easier is a hobby my roomate Andrew and I enjoy. He is actually the owner of the local Milk Boy Art House concert and mini-store, so I chose this project because I could see a practical use for its final product. I changed the names of the bands and songs that he books, but otherwise, looking back at this project, I wish I had chosen this topic initially. Because, I believe I could have truly gone father with my interface and database capabilities. Initially, I had chosen to do my final project regarding the restaurant industry, a place that I work in and am quite familiar with. However, as I dove into the project earlier this semeter, I could not find the interest or vision to continue it. Looking at my goals for this project, I believe that I could have utilized visio and asp.net to create an acutal website that Andew could use when planning which artists to book. Additionally, I believe that this would allow for samples of the artists’ songs to be uploaded, along with potentiall creating a “checkout’ style store where Andrews venue customers and concert-goers could go to after or before the concert in order to buy merchandise. I believe that this project has a lot of potential, and avenues it could expand upon because its focus is very straightforward. The goal of this project was not to tackle every single process related to concerts at a venue and artist planning. The main objective was the deliver a solution to the major problem- how can Andrew plan the bands and his merchandise more efficienrly. I believe through the use of the Access forms, he will be able to manage it for now. I would like to take this one step further, after the semester is over, and potentially see the website created.