ART Gallery Management System

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology/Master of Technology

In

Computer Science and Engineering

School of Engineering and Sciences

Submitted by

Praveen Chowdary Vemasani

AP20110010415

Under the Guidance of

Dr._________

SRM University–AP

Neerukonda, Mangalagiri, Guntur

Andhra Pradesh – 522 240

December 2022

Certificate

Date: 08-Dec-22

This is to certify that the work present in this Project entitled "**ART Gallery Management System**" has been carried out by **Vemasani Praveen Chowdary** under my/our supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology/Master of Technology in the **School of Engineering and Sciences**.

Supervisor

(Signature)

Prof. / Dr.________

Professor,

Affiliation.

Acknowledgments

It is myprivilege to express my sincerest regards to my project mentor, Dr. Rajiv Senapati for his valuable inputs, able guidance, encouragement, whole-hearted cooperation and constructive criticism throughout the duration of our project.

I would like to express my profound gratitude to Mr./Mrs.____ (name of the HOD), of ____ (designation and department name) department, and Mr./Mrs. _____ (Dean) of _____ university for their contributions to the completion of my project titled _____

I would like to express my special thanks to my mentor prof.Rajiv Senapati again for his time and efforts he provided throughout the semester.

I take this opportunity to thank all our lecturers who have directly or indirectly helped our project. Last but not the least we express our thanks to our friends for their cooperation and support.

Table of Contents

Abstract

To create an Art Exhibition Management System that keeps records of artists, their paintings, art gallery details, and exhibition details, and showcases pictures of paintings to the customers.
This project intends to include various features related to an art exhibition i.e. information about the gallery, exhibition, artists, their paintings, customers (the ones who buy the paintings), etc.

Art Gallery, as described above, can lead to an error-free, secure, reliable, and fast management system. It can assist the user in concentrating on their other activities rather to concentrate on record keeping. Thus it will help organizations in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant while being able to reach the information.

The aim is to automate its existing manual system with the help of a reliable database management system, fulfilling its requirements so that their valuable data/information can be stored for a longer period with easy access and manipulation of the same. Basically, the project describes how to manage for good performance and better client services.

Abbreviations

DBMS	Data Base Management System
E-R	Entity-Relationship
FFD	Full Functionally Dependency
PFD	Paertial Functional Dependency
TFD	Transitive Functianal Dependency
1NF	First Normal Form
2NF	Seconf Normal Form
3NF	Third Normal Form
BCNF	Boyce Codd Normal Form
4NF	Fourth Normal Form
5NF	Fifth Normal Form
PJNF	Project-Join Normal Form

List of Tables

List of Figures

Introduction

In this modern times, data/information is every thing.Data will help you to improve quality of life for people.Data = Knowledge. Good data provides indisputable evidence, while anecdotal evidence, assumptions, or abstract observation might lead to wasted resources due to taking action based on an incorrect conclusion.

So relying on the older techniques like file managements system is not a good choice.It has many issues and take more manpower and time.

Data Redundancy, Data Inconsistency, Data Isolation, Atomicity, Concurrent access, Security are the few tasks which becomes an anomaly by using files.

Database Management System helps to overcome all these issues.

Art gallery management system is one of the application, where Database management system makes it way more efficient.

In order to create a good data base we need to follow the following steps :

- Determine the purpose of your database
- Find the Entity List
- List Down the attributes for all the entities
- Establish E-R Model
- Draw E-R Model
- Convert the E-R to Relational Model
- Normalise the Relational Model
- Build the SQL

Methodology

Entities

Entity: a “thing” or “object” in the enterprise that is
distinguishable from other objects

- ❖ Described by a set of attributes

List of Entities in the database :

- **Artist:** The person who can showcase his artwork in the exhibition. It contains primary information about the artist like name, phone number, his/her painting style, address, city, etc.

- **Artwork:** The artwork that is being exhibited in the exhibition by any of the artists. It contains basic information about the artwork like the year it was made, title, price, type of painting, etc.

- **Exhibition:** The exhibition or the place where the exhibition is being organized. It contains information like the name of the exhibition/place, the start date and end date of the exhibition, and the address of the exhibition.

- **Stall:** Art Exhibition has several stalls, which provide space for presenting artwork. This entity contains information like Stall No, open time and close time.

- **Order:** The artwork that is sold. It specifies when the artwork sold and at what price it sold.

- **Customer**: This describes the customer who visits the exhibition or buys the artwork or rents the artwork from the exhibition. It contains information like the name of the customer, phone number, address, email, etc.

- **Bill**: This describes the total bill consists of all the orders places by the customer. It contains information like total bill paid, bill details, GST etc.

Artist:

Atribute	Datatype	Constraint
Name	varchar(30)	Not Null
Artist_ID	int	Primary Key
Style	varchar(30)	Not Null
Address	varchar(200)	Not Null
Contact	Int8	Not Null

Artwork:

Atribute	Datatype	Constraint
Art_ID	Int	Primary Key
Title	varchar(30)	Not Null
Year	date	Not Null
Price	int	Not Null
Description	varchar(200)	Not Null
Type	varchar(30)	Not Null

Order_Info:

Atribute	Datatype	Constraint
O_ID	int	Primary Key
O_Date	date	Not Null
O_Price	int	Not Null

Customer:

Atribute	Datatype	Constraint
C_ID	int	Primary Key
E_Mail	varchar(30)	Not Null
Name	varchar(30)	Not Null
Contact	int8	Not Null

Bill:

Atribute	Datatype	Constraint
Bill_ID	int	Primary Key
Bill_Details	varchar(20)	Not Null
Bill_Paid	int	Not Null
Gst	int	Not Null

Stall:

Atribute	Datatype	Constraint
S_No	int	Primary Key
Open_time	time	Not Null
Close_time	time	Not Null

Exhibition:

Atribute	Datatype	Constraint
Expo_ID	int	Primary Key
Address	varchar(30)	Not Null
Start_Date	date	Not Null
End_Date	date	Not Null
Name	varchar(50)	Not Null

Attributes

Attributes — characteristics of an entity, and has an oval symbol.

There are different types of attributes :

- ***Key attribute:*** An attribute uniquely distinguishes the entity in an entity set.
- ***Simple attribute:*** An attribute that cannot be further subdivided into components.
- ***Composite attribute:*** An attribute that can be split into components.
- ***Single-valued attribute:*** The attribute which takes up only a single value for each entity instance.
- ***Multi-valued attribute:*** The attribute which takes up more than a single value for each entity instance.
- ***Stored attribute:*** An attribute that stores the data which can be used to get the derived attribute.
- ***Derived attribute:*** An attribute that can be derived from other attributes.

Attributes for each entity in the art exhibition database:

Artist: Name, phone number, art_style, address, Artist_ID.
Artwork: Art_ID, Title, Price, Description, Type, Year,.
Exhibition: Name, Start_date, End_date, Address, Expo-ID, Address.
Stall: Open_time, Close_time, S_No.
Order: Order_price, Order_date.
Customer: F_name, L_name, Date of birth, phone number, address, city,pincode, state, country, email.
Bill: Bill_ID, Bill_Paid, GST, Bill_Details.

A relationship is an association among several entities
Relationships:

- Artists creating paintings.
- Paintings created by Artist.
- Paintings present at Stalls(Exhibition Stalls).
- Art Exhibition Stalls present Paintings
- Artwork ordered via Order
- Order(Customer) be in agreement to buy Artwork
- Customer placing Order (Artwork)
- Order (Artwork) placed by Customer
- The customer pays Bill.
- Bill Paid by the customer.
- The exhibition has several stalls.
- Stalls are in the Exhibition.

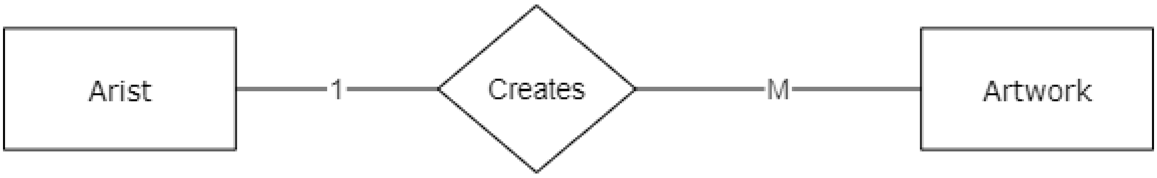
Relations and Cardinality

Cardinality represents the number of times an entity of an entity set participate in a relationship set.

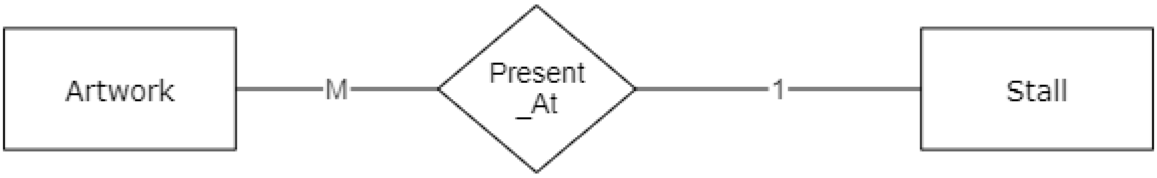
Types of cardinality:

- one to one
- one to many
- many to one
- many to many

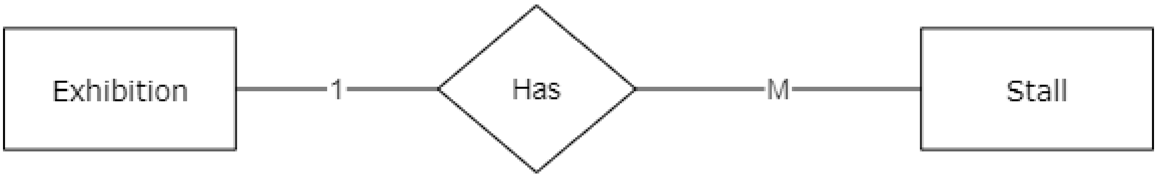
1. **Artist and Artwork**
   Relation: Creates
   Cardinality: one to many



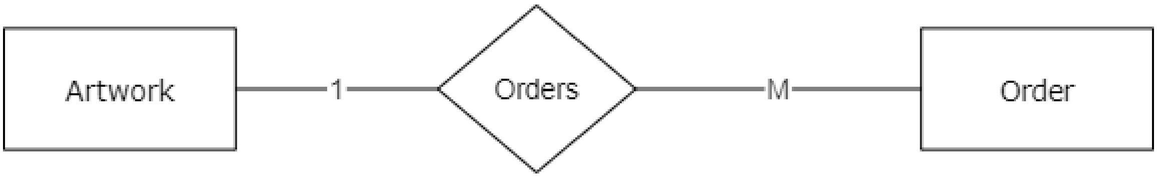
2. **Artwork and Stall**
   Relation: Present_At
   Cardinality: Many to One



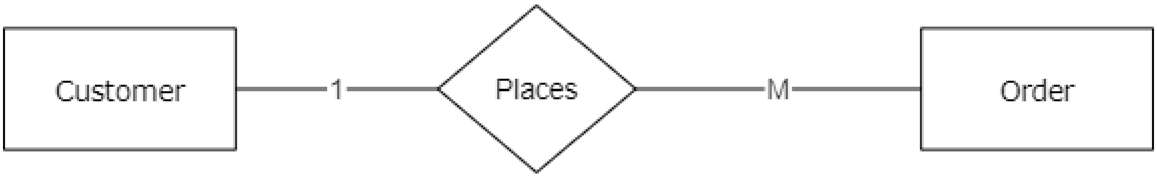
3. **Exhibition and Stall**
   Relation: Has
   Cardinality: One to Many



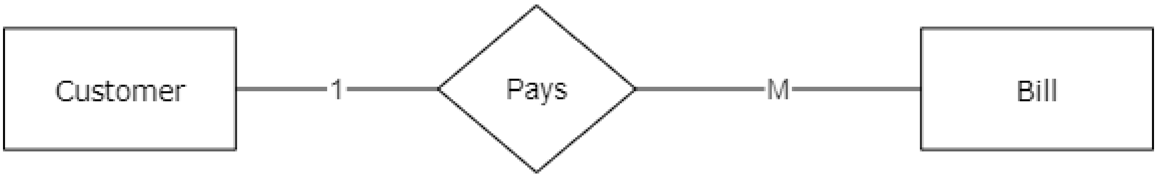
4. **Artwork and Order_Info**
   Relation: Orders
   Cardinality: One to Many



5. **Customer and Order_Info**
   Relation: Places
   Cardinality: One to Many



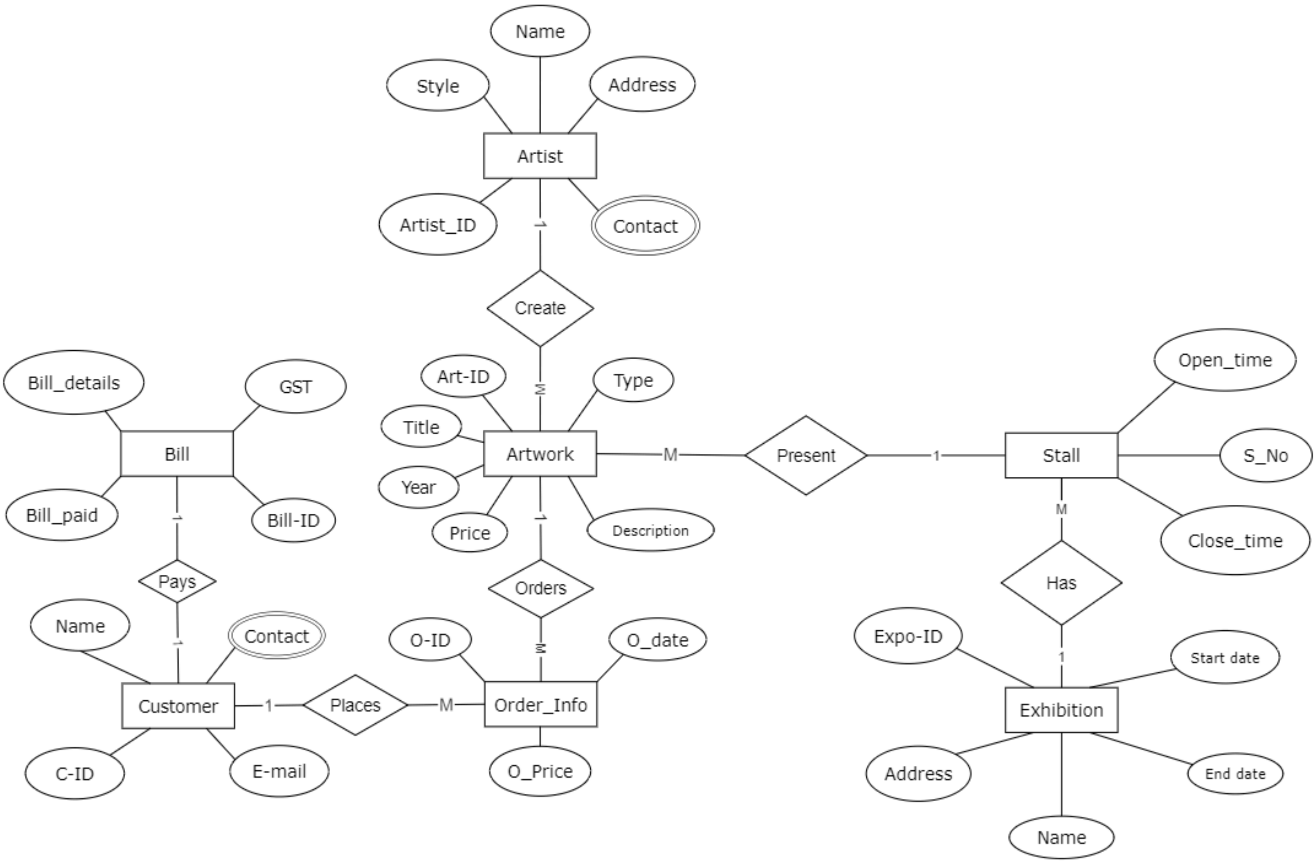
6. **Customer and Bill**
   Relation: Pays
   Cardinality: One to Many



Entity-Relationship Model (ER):

ER model stands for an Entity-Relationship model.
It is a high-level data model. This model is used to define the data elements and relationships for a specified system.

It develops a conceptual design for the database. It also develops a very simple and easy-to-design view of data.



Relational Model:

A relational database is a collection of tables. Atable consists of rows and columns. Each and Every Column header is called attribute. Each and Every row is called as Tuples.

So in relational model we are keeping the information regarding entity in a tuple manner.

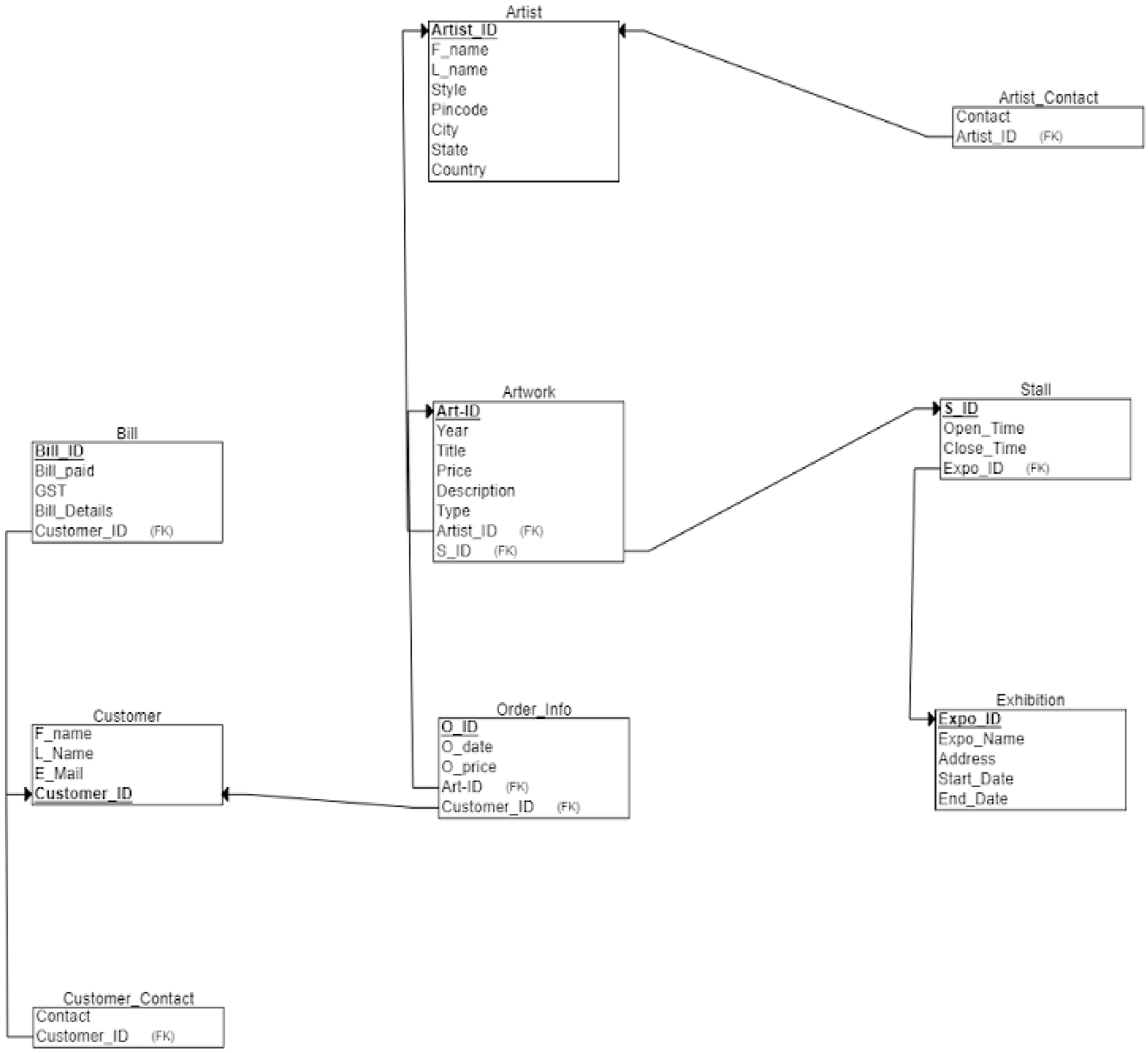
Conversion From ER Model to Relational Model

1. An entity in the ER model is represented by a relational table in the relational model.
2. All the attributes of the ER model are represented in different columns as an attribute.
3. Primary key attribute of the ER model is represented as the primary key in the relational model.
4. Composite key of the ER model is split in different columns in the relational model.
5. Derived attributes must be dropped in relational models.
6. Multi valued attributes need to be kept separate in a new table.
7. If the relationship between two entities is 1:1 then the primary key of one relation becomes foreign key in another relation.
8. If the relationship between entities is 1:M or M:1 then the primary key attribute of one-sided relations becomes a foreign key in many-sided relations.
9. If the relationship between entities is M:M then a new table needs to be created to represent that relation, where the new relation will have the primary key of both relations as the foreign keys.

Schema Diagram :

A database schema, along with primary key and foreign key dependencies, can be expressed pictorially by schema diagram.

Here is the diagram for the Art gallery Managemst system.



Normalization:

Normalization is used to minimize the redundancy from a relation or set of relations.

There are some pitbulls in Relational Database

- Data Redundancy
- Data inconsistency
- Insert, Update, delete anomalies
- wastage of storage space due to data redundancy
- sometimes unable to represent data properly

There are 5 Normal forms:

1. First Normal Form (1NF)

2. Second Normal Form (2NF)

3. Third Normal Form (3NF) or Boyes Cord Normal Form (BCNF)

4. Fourth Normal Form (4NF)

5. Fifth Normal Form (5F) or Project Join Normal Form (PJNF)

FIRST NORMAL FORM (1NF)

• A relation is said to be in 1NF, if it has got no non-atomic attributes i.e., which cannot be subdivided.

SECOND NORMAL FORM (2NF)

• A relation in 1NF is said to be in 2NF, if it satisfies any one of the following conditions. They are,

1. The primary key consists of only one attribute.

2. There exists no non key attribute.

3. Every non key attribute present in relation should functionally depend upon a full set of primary keys.

THIRD NORMAL FORM (3NF)

• A relation which is in 2NF is said to be in 3NF, if there exists no transitive functional dependency of any non-key attribute on the set of primary keys.

• Transitivity says that, if X->Y and Y->Z then X->Z.

Normalization of Database:

1. **Artist(**Artist_ID (key), FirstName, LastName, Pincode, City, State, Country, Style**)**

1NF: Meets the 1NF because it has no non-atomic attribute.
2NF: Meets the 2NF Rule-1 The primary key contains only one attribute.
3NF: This is not in 3N due to the existence of the transitive dependency.

Artist_ID → FirstName, LastName, Style, Pincode.

Pincode→ City, State, Country.

Solution: Split the relation into two relations named Artist_Info and Artist_Address.

Artist_Info(Artist_ID(key), First_Name, Last_Name, Pincode(fk), Style).
Artist_Address(Pincode(key), City, State, Country).

2. **Artist_Contact(**Artist_ID (fk), Contact**)**

1NF: Meets the 1NF because it has no non-atomic attribute.
2NF: Meets the 2NF Rule-1 The primary key contains only one attribute.
3NF: This is in 3N due to the existence of the no transitive dependency.

3. **Artwork(**Art_ID (key), Year, Title, Price, Description, Type, Artist_ID(fk), S_ID(fk)**)**

1NF: This meets the 1NF because it has no non-atomic attribute.
2NF: Meets the 2NF Rule-1 The primary key contains only one attribute.
3NF: This is in 3N due to the existence of the no transitive dependency.

4. **Exhibition(**Expo_ID (key), Gallery_Name, Start_Date, End_Date, Pincode, City, State, Country**)**

1NF: Meets the 1NF because it has no non-atomic attribute.
2NF: Meets the 2NF Rule-1 The primary key contains only one attribute.
3NF: This is not in 3N due to the existence of the transitive dependency.

Exhibition→ Expo_ID (key), Gallery_Name, Start_Date, End_Date, Pincode.

Pincode→ City, State, Country.

Solution: Split the relation into two relations named Artist_Info and Artist_Address.
Exhibition(Expo_ID (key), Expo_Name, Start_Date, End_Date, Pincode(fk)).
Exhibition_Address(Pincode(key), City, State, Country).

5. **Stall(**S_ID (key), Open_time, Close_time, Expo_ID(fk)**)**

1NF: Meets the 1NF because it has no non-atomic attribute.
2NF: Meets the 2NF Rule-1 The primary key contains only one attribute.
3NF: This is in 3N due to the existence of the no transitive dependency.

6. **Order_Info(**Order_ID (key), Order_Date, Price, Art_ID(fk), Customer_ID(fk)**)**

1NF: Meets the 1NF because it has no non-atomic attribute.
2NF: Meets the 2NF Rule-1 The primary key contains only one attribute.
3NF: This is in 3N due to the existence of the no transitive dependency.

7. **Customer(**Customer_ID (key), F_name, L_Name, E-Mail**)**

1NF: Meets the 1NF because it has no non-atomic attribute.
2NF: Meets the 2NF Rule-1 The primary key contains only one attribute.
3NF: This is in 3N due to the existence of the no transitive dependency.

8. **Customer_Contact(**Customer_ID (fk), Contact**)**

1NF: Meets the 1NF because it has no non-atomic attribute.
2NF: Meets the 2NF Rule-1 The primary key contains only one attribute.
3NF: This is in 3N due to the existence of the no transitive dependency.

9. **Bill(**Bill_ID (key), Bill_paid, GST, Bill_Details, Customer_ID(fk)**).**

1NF: Meets the 1NF because it has no non-atomic attribute.
2NF: Meets the 2NF Rule-1 The primary key contains only one attribute.
3NF: This is in 3N due to the existence of the no transitive dependency.

Final relations normalized to the Third Normal Form:

Artist_Info(Artist_ID(key), First_Name, Last_Name, Pincode(fk), Style).

Artist_Address(Pincode(key), City, State, Country).

Artist_Contact(Artist_ID (fk), Contact**)**

Artwork(Art_ID (key), Year, Title, Price, Description, Type, Artist_ID(fk), S_ID(fk)**)**

Exhibition(Expo_ID (key), Gallery_Name, Start_Date, End_Date, Pincode(fk)).

Exhibition_Address(Pincode(key), City, State, Country).

Stall(S_ID (key), Open_time, Close_time, Expo_ID(fk)**)**

Order_Info(Order_ID (key), Order_Date, Price, Art_ID(fk), Customer_ID(fk)**)**

Customer(Customer_ID (key), F_name, L_Name, E-Mail**)**

Customer_Contact(Customer_ID (fk), Contact**)**

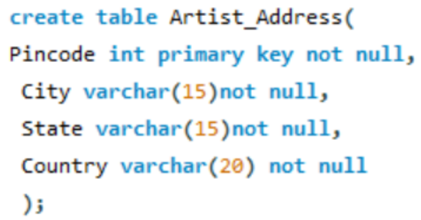
Bill(Bill_ID (key), Bill_paid, GST, Bill_Details, Customer_ID(fk)**).**

SQL

ENTITY RECORDS

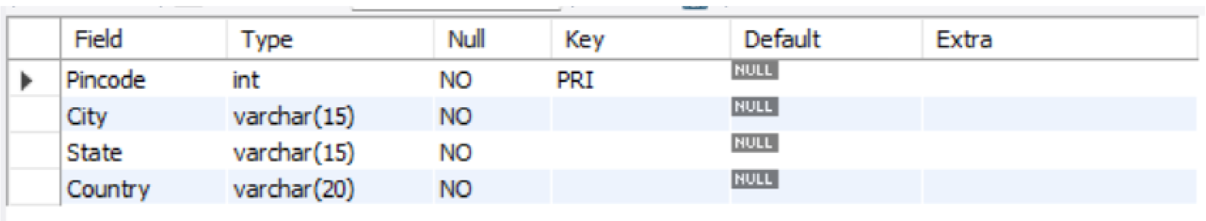
1. Artist_Address Table

Below is the SQL command to Create Artist_Address Table:



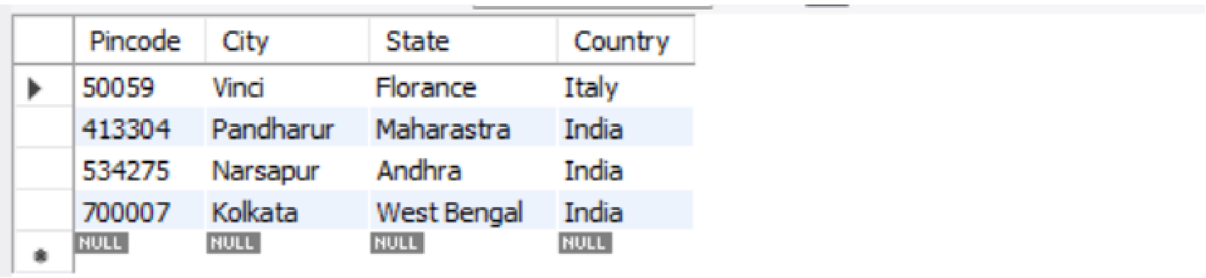
```
create table Artist_Address(
Pincode int primary key not null,
 City varchar(15)not null,
 State varchar(15)not null,
 Country varchar(20) not null
 );
```

Table:



	Field	Type	Null	Key	Default	Extra
▶	Pincode	int	NO	PRI	NULL	
	City	varchar(15)	NO		NULL	
	State	varchar(15)	NO		NULL	
	Country	varchar(20)	NO		NULL	

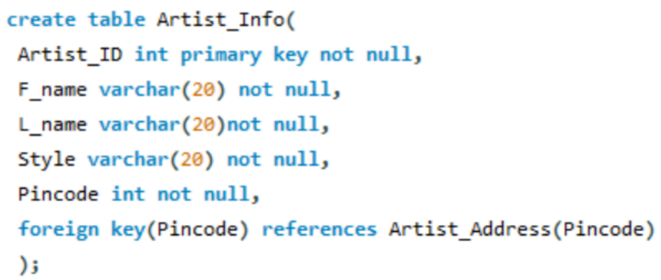
Sample Table:



	Pincode	City	State	Country
▶	50059	Vinci	Florance	Italy
	413304	Pandharur	Maharastra	India
	534275	Narsapur	Andhra	India
	700007	Kolkata	West Bengal	India
*	NULL	NULL	NULL	NULL

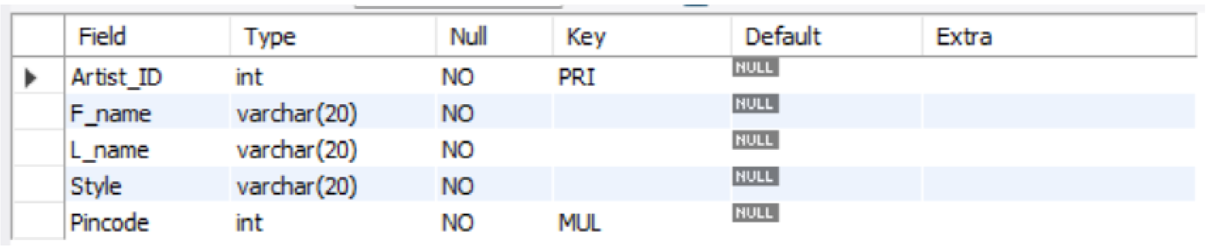
2. Artist_Info Table

Below is the SQL command to Create Artist_Info Table:



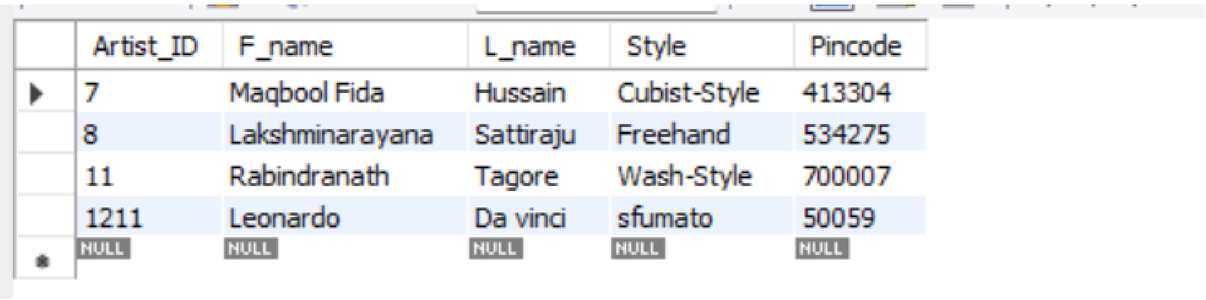
```
create table Artist_Info(
 Artist_ID int primary key not null,
 F_name varchar(20) not null,
 L_name varchar(20)not null,
 Style varchar(20) not null,
 Pincode int not null,
 foreign key(Pincode) references Artist_Address(Pincode)
 );
```

Table:



	Field	Type	Null	Key	Default	Extra
▶	Artist_ID	int	NO	PRI	NULL	
	F_name	varchar(20)	NO		NULL	
	L_name	varchar(20)	NO		NULL	
	Style	varchar(20)	NO		NULL	
	Pincode	int	NO	MUL	NULL	

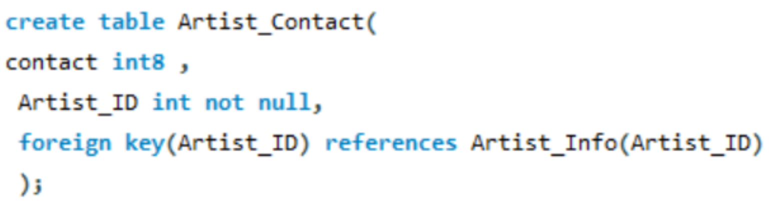
Sample Table :



	Artist_ID	F_name	L_name	Style	Pincode
▶	7	Maqbool Fida	Hussain	Cubist-Style	413304
	8	Lakshminarayana	Sattiraju	Freehand	534275
	11	Rabindranath	Tagore	Wash-Style	700007
	1211	Leonardo	Da vinci	sfumato	50059
*	NULL	NULL	NULL	NULL	NULL

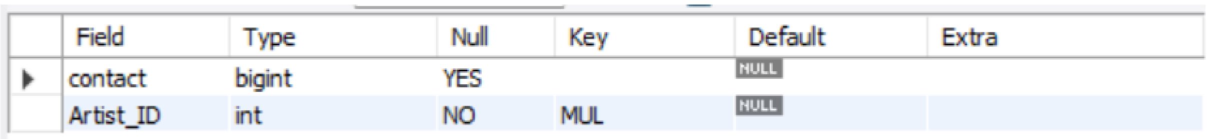
3. Artist_Contact Table

Below is the SQL command to Create the Artist_Contact Table:



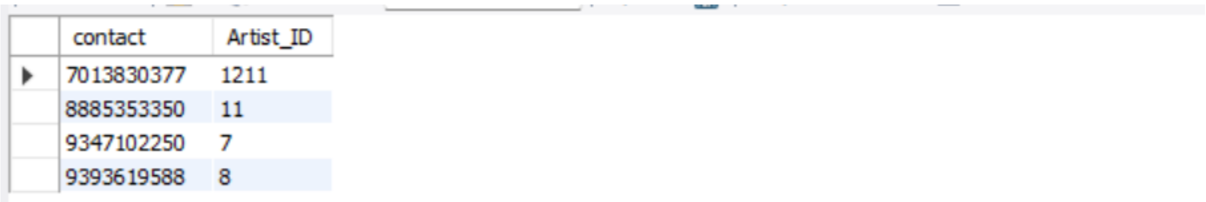
```
create table Artist_Contact(
contact int8 ,
 Artist_ID int not null,
 foreign key(Artist_ID) references Artist_Info(Artist_ID)
 );
```

Table:



	Field	Type	Null	Key	Default	Extra
▶	contact	bigint	YES		NULL	
	Artist_ID	int	NO	MUL	NULL	

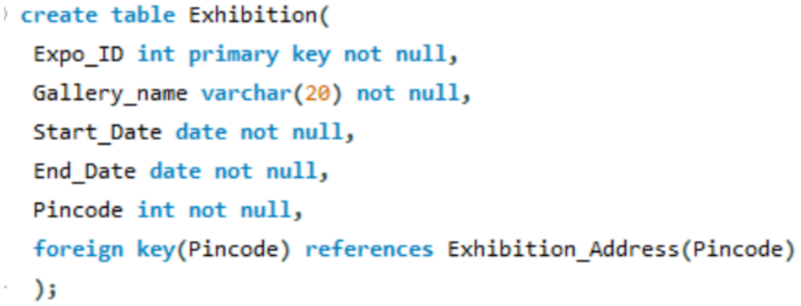
Sample Table :



	contact	Artist_ID
▶	7013830377	1211
	8885353350	11
	9347102250	7
	9393619588	8

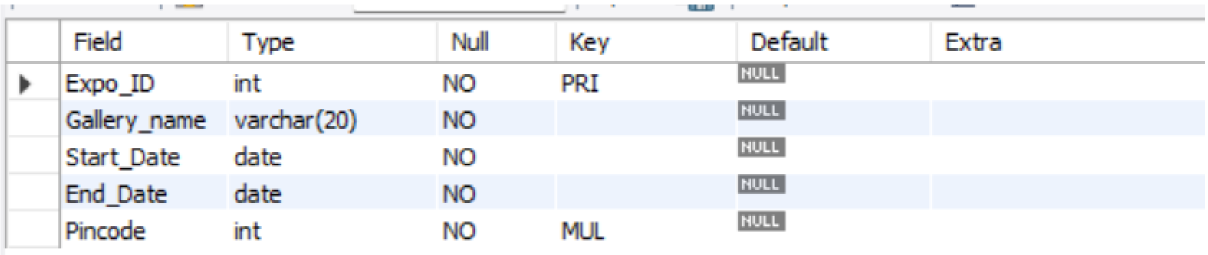
4. Exhibition Table

Below is the SQL command to Create an Exhibition Table:



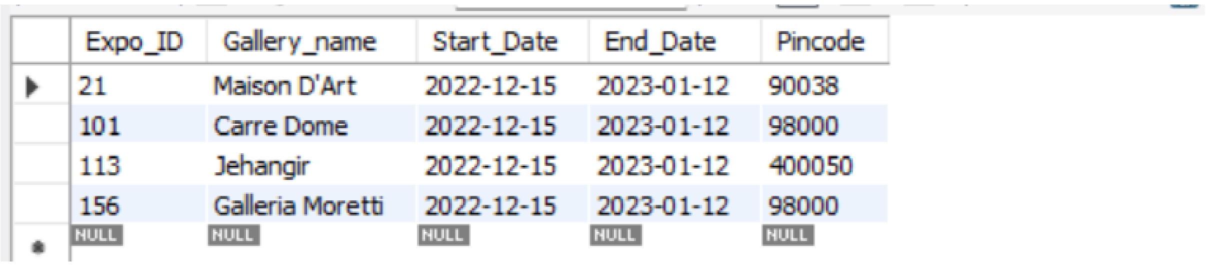
```
create table Exhibition(
 Expo_ID int primary key not null,
 Gallery_name varchar(20) not null,
 Start_Date date not null,
 End_Date date not null,
 Pincode int not null,
 foreign key(Pincode) references Exhibition_Address(Pincode)
 );
```

Table:



	Field	Type	Null	Key	Default	Extra
▶	Expo_ID	int	NO	PRI	NULL	
	Gallery_name	varchar(20)	NO		NULL	
	Start_Date	date	NO		NULL	
	End_Date	date	NO		NULL	
	Pincode	int	NO	MUL	NULL	

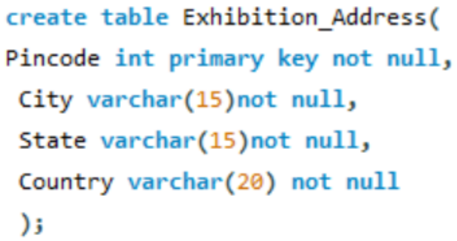
Sample Table:



	Expo_ID	Gallery_name	Start_Date	End_Date	Pincode
▶	21	Maison D'Art	2022-12-15	2023-01-12	90038
	101	Carre Dome	2022-12-15	2023-01-12	98000
	113	Jehangir	2022-12-15	2023-01-12	400050
	156	Galleria Moretti	2022-12-15	2023-01-12	98000
*	NULL	NULL	NULL	NULL	NULL

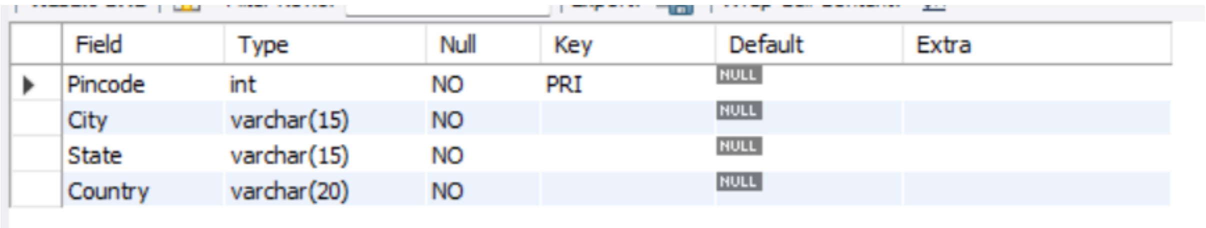
5. Exhibition_Address Table

Below is the SQL command to Create Exhibition_Address Table:



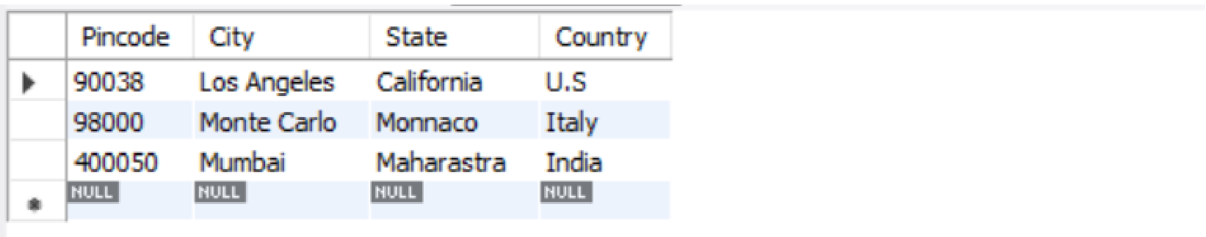
```
create table Exhibition_Address(
Pincode int primary key not null,
 City varchar(15)not null,
 State varchar(15)not null,
 Country varchar(20) not null
 );
```

Table:



	Field	Type	Null	Key	Default	Extra
▶	Pincode	int	NO	PRI	NULL	
	City	varchar(15)	NO		NULL	
	State	varchar(15)	NO		NULL	
	Country	varchar(20)	NO		NULL	

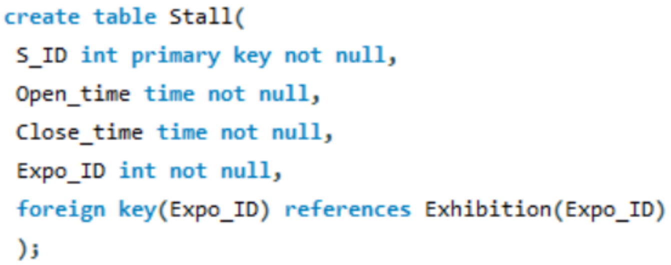
Sample Table :



	Pincode	City	State	Country
▶	90038	Los Angeles	California	U.S
	98000	Monte Carlo	Monnaco	Italy
	400050	Mumbai	Maharastra	India
*	NULL	NULL	NULL	NULL

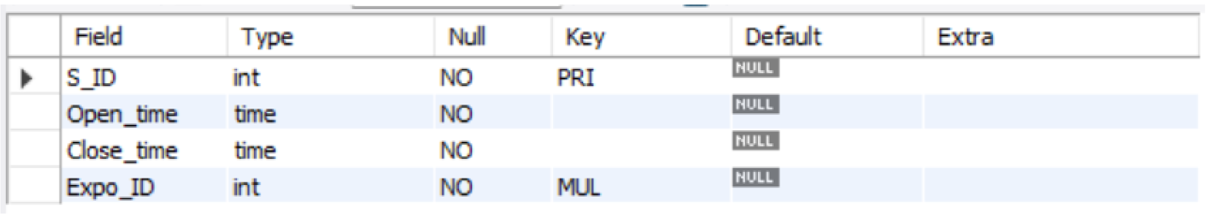
6. Stall Table

Below is the SQL command to Create Stall Table:



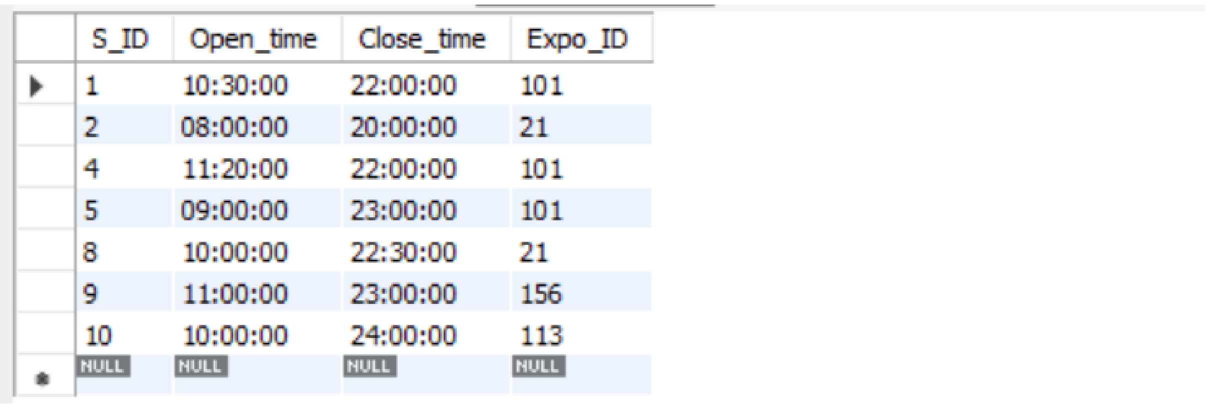
```
create table Stall(
 S_ID int primary key not null,
 Open_time time not null,
 Close_time time not null,
 Expo_ID int not null,
 foreign key(Expo_ID) references Exhibition(Expo_ID)
 );
```

Table:



	Field	Type	Null	Key	Default	Extra
▶	S_ID	int	NO	PRI	NULL	
	Open_time	time	NO		NULL	
	Close_time	time	NO		NULL	
	Expo_ID	int	NO	MUL	NULL	

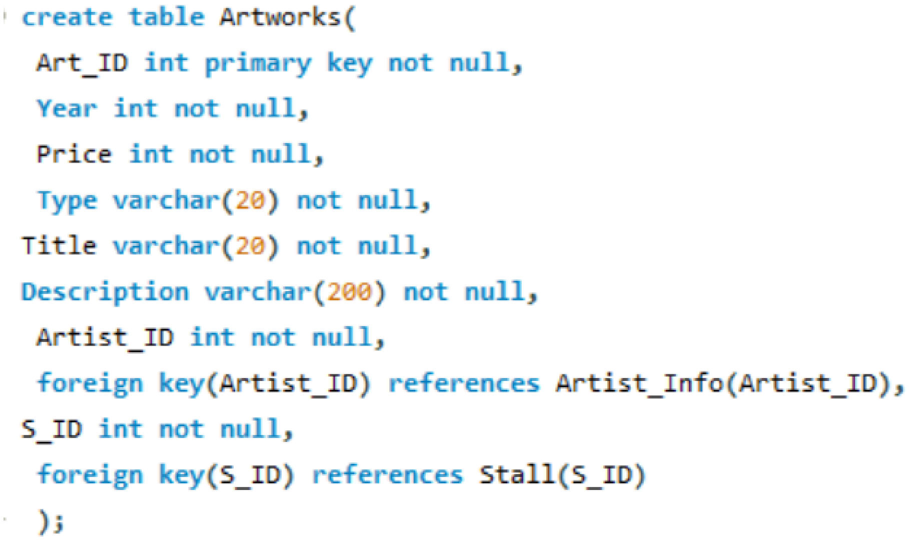
Sample Table :



	S_ID	Open_time	Close_time	Expo_ID
▶	1	10:30:00	22:00:00	101
	2	08:00:00	20:00:00	21
	4	11:20:00	22:00:00	101
	5	09:00:00	23:00:00	101
	8	10:00:00	22:30:00	21
	9	11:00:00	23:00:00	156
	10	10:00:00	24:00:00	113
*	NULL	NULL	NULL	NULL

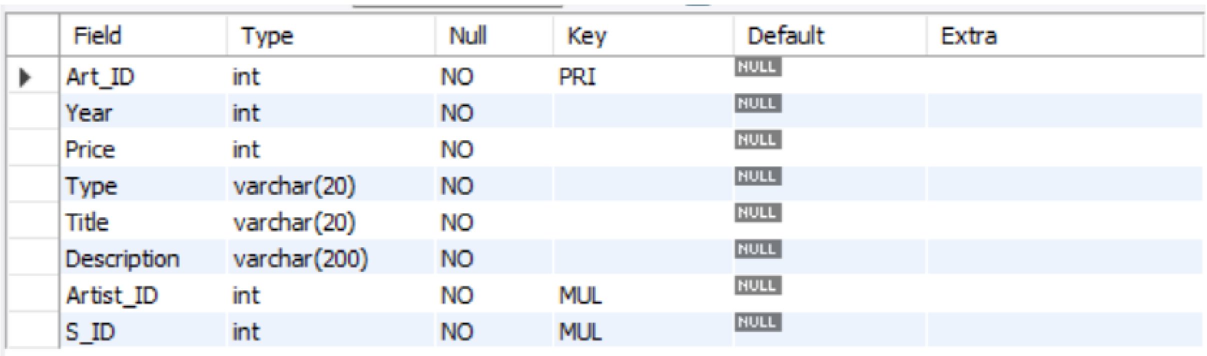
7. Artworks Table

Below is the SQL command to Create an Artworks Table:



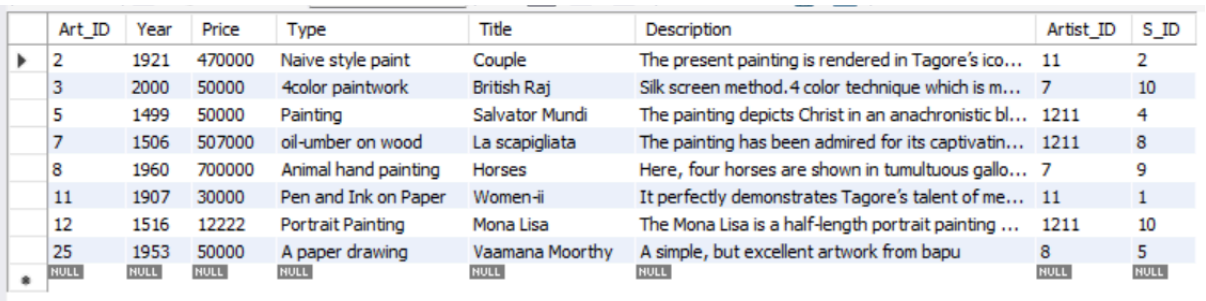
```
create table Artworks(
 Art_ID int primary key not null,
 Year int not null,
 Price int not null,
 Type varchar(20) not null,
Title varchar(20) not null,
Description varchar(200) not null,
 Artist_ID int not null,
 foreign key(Artist_ID) references Artist_Info(Artist_ID),
S_ID int not null,
 foreign key(S_ID) references Stall(S_ID)
 );
```

Table:



	Field	Type	Null	Key	Default	Extra
▶	Art_ID	int	NO	PRI	NULL	
	Year	int	NO		NULL	
	Price	int	NO		NULL	
	Type	varchar(20)	NO		NULL	
	Title	varchar(20)	NO		NULL	
	Description	varchar(200)	NO		NULL	
	Artist_ID	int	NO	MUL	NULL	
	S_ID	int	NO	MUL	NULL	

Sample Table :



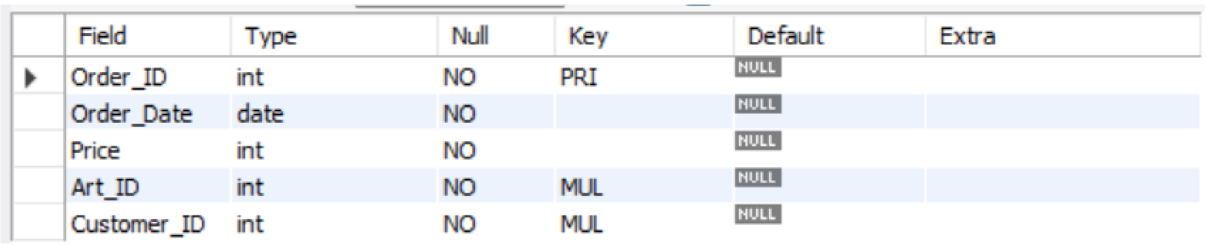
	Art_ID	Year	Price	Type	Title	Description	Artist_ID	S_ID
▶	2	1921	470000	Naive style paint	Couple	The present painting is rendered in Tagore's ico...	11	2
	3	2000	50000	4color paintwork	British Raj	Silk screen method.4 color technique which is m...	7	10
	5	1499	50000	Painting	Salvator Mundi	The painting depicts Christ in an anachronistic bl...	1211	4
	7	1506	507000	oil-umber on wood	La scapigliata	The painting has been admired for its captivatin...	1211	8
	8	1960	700000	Animal hand painting	Horses	Here, four horses are shown in tumultuous gallo...	7	9
	11	1907	30000	Pen and Ink on Paper	Women-ii	It perfectly demonstrates Tagore's talent of me...	11	1
	12	1516	12222	Portrait Painting	Mona Lisa	The Mona Lisa is a half-length portrait painting ...	1211	10
	25	1953	50000	A paper drawing	Vaamana Moorthy	A simple, but excellent artwork from bapu	8	5
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

8. Order_Info Table

Below is the SQL command to Create Order_Info Table:

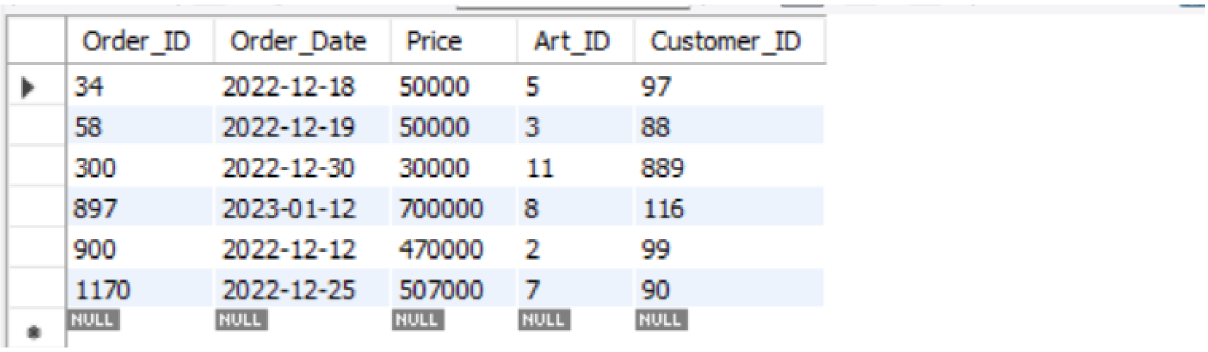
```
create table Order_Info(
 Order_ID int primary key not null,
 Order_Date date not null,
Price int not null,
 Art_ID int not null,
 foreign key(Art_ID) references Artwork(Art_ID),
 Customer_ID int not null,
 foreign key(Customer_ID) references Customer(Customer_ID)
 );
```

Table:



	Field	Type	Null	Key	Default	Extra
▶	Order_ID	int	NO	PRI	NULL	
	Order_Date	date	NO		NULL	
	Price	int	NO		NULL	
	Art_ID	int	NO	MUL	NULL	
	Customer_ID	int	NO	MUL	NULL	

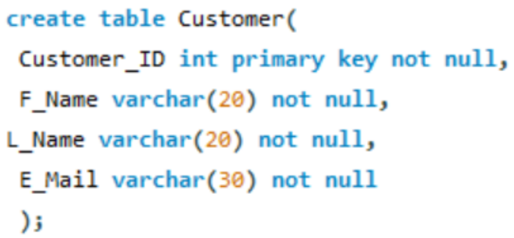
Sample Table :



Order_ID	Order_Date	Price	Art_ID	Customer_ID
34	2022-12-18	50000	5	97
58	2022-12-19	50000	3	88
300	2022-12-30	30000	11	889
897	2023-01-12	700000	8	116
900	2022-12-12	470000	2	99
1170	2022-12-25	507000	7	90
NULL	NULL	NULL	NULL	NULL

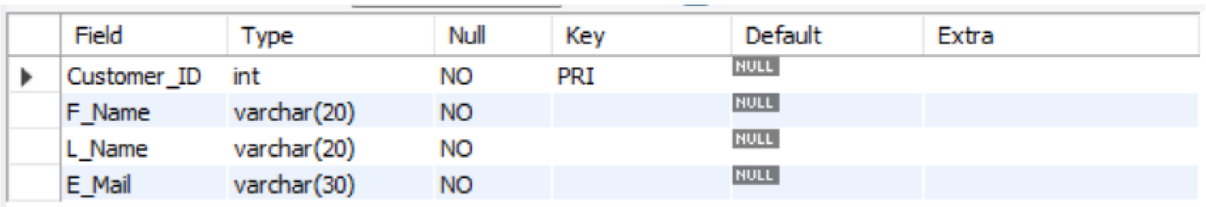
9. Customer Table

Below is the SQL command to Create a Customer Table:



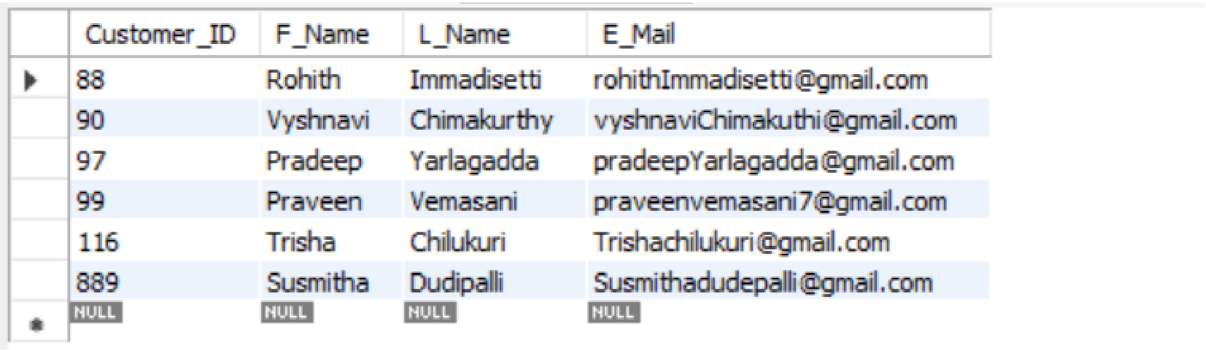
```
create table Customer(
 Customer_ID int primary key not null,
 F_Name varchar(20) not null,
L_Name varchar(20) not null,
 E_Mail varchar(30) not null
 );
```

Table:



Field	Type	Null	Key	Default	Extra
Customer_ID	int	NO	PRI	NULL	
F_Name	varchar(20)	NO		NULL	
L_Name	varchar(20)	NO		NULL	
E_Mail	varchar(30)	NO		NULL	

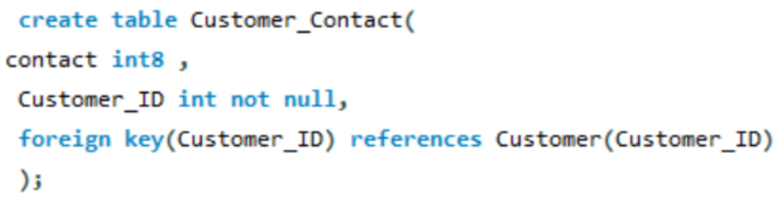
Sample Table :



Customer_ID	F_Name	L_Name	E_Mail
88	Rohith	Immadisetti	rohithImmadisetti@gmail.com
90	Vyshnavi	Chimakurthy	vyshnaviChimakuthi@gmail.com
97	Pradeep	Yarlagadda	pradeepYarlagadda@gmail.com
99	Praveen	Vemasani	praveenvemasani7@gmail.com
116	Trisha	Chilukuri	Trishachilukuri@gmail.com
889	Susmitha	Dudipalli	Susmithadudepalli@gmail.com
NULL	NULL	NULL	NULL

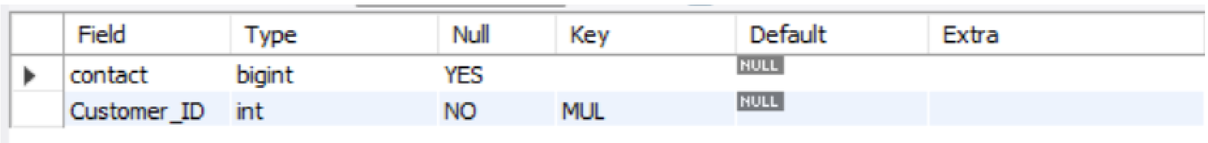
10. Customer_Contact Table

Below is the SQL command to Create the Customer_Contact Table:



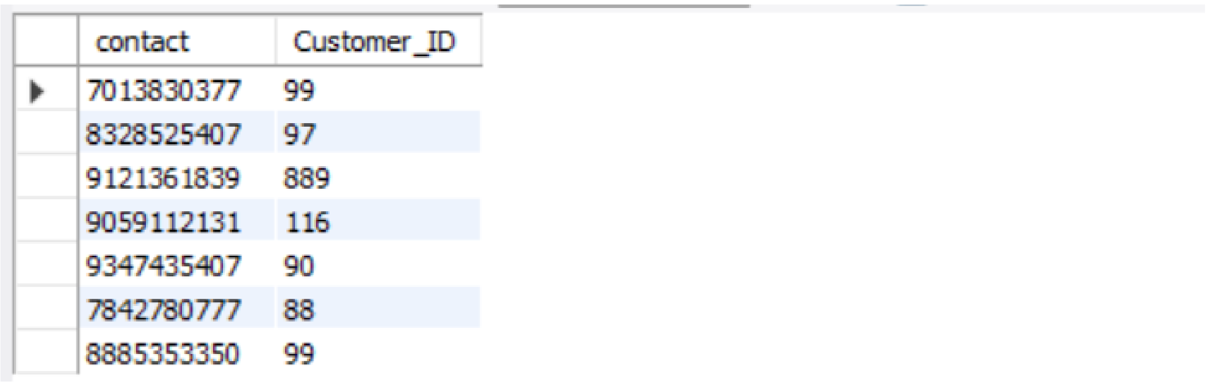
```
 create table Customer_Contact(
contact int8 ,
 Customer_ID int not null,
 foreign key(Customer_ID) references Customer(Customer_ID)
 );
```

Table:



Field	Type	Null	Key	Default	Extra
contact	bigint	YES		NULL	
Customer_ID	int	NO	MUL	NULL	

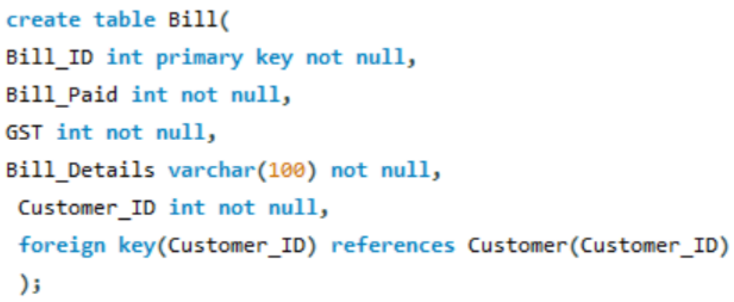
Sample Table:



contact	Customer_ID
7013830377	99
8328525407	97
9121361839	889
9059112131	116
9347435407	90
7842780777	88
8885353350	99

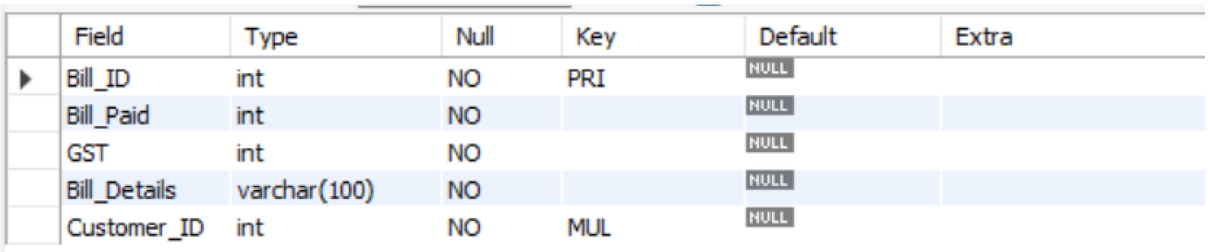
11. Bill Table

Below is the SQL command to Create Bill Table:



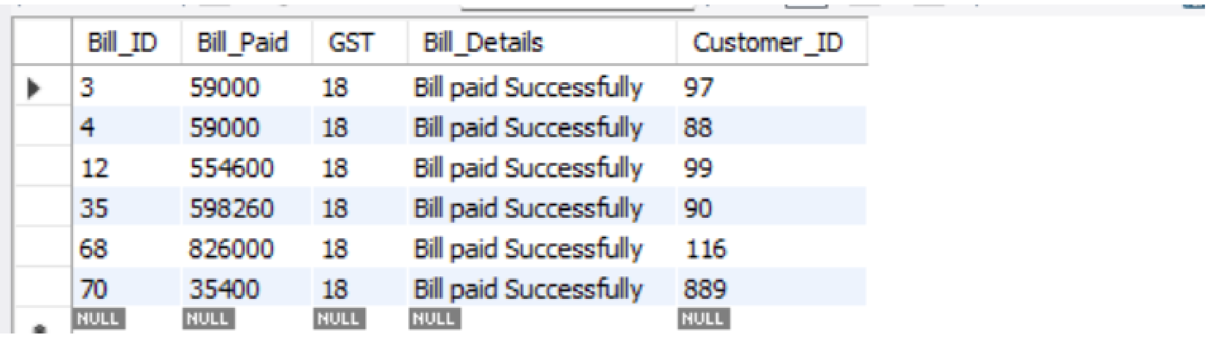
```
create table Bill(
Bill_ID int primary key not null,
Bill_Paid int not null,
GST int not null,
Bill_Details varchar(100) not null,
 Customer_ID int not null,
 foreign key(Customer_ID) references Customer(Customer_ID)
 );
```

Table:



Field	Type	Null	Key	Default	Extra
Bill_ID	int	NO	PRI	NULL	
Bill_Paid	int	NO		NULL	
GST	int	NO		NULL	
Bill_Details	varchar(100)	NO		NULL	
Customer_ID	int	NO	MUL	NULL	

Sample Table :



Bill_ID	Bill_Paid	GST	Bill_Details	Customer_ID
3	59000	18	Bill paid Successfully	97
4	59000	18	Bill paid Successfully	88
12	554600	18	Bill paid Successfully	99
35	598260	18	Bill paid Successfully	90
68	826000	18	Bill paid Successfully	116
70	35400	18	Bill paid Successfully	889
NULL	NULL	NULL	NULL	NULL

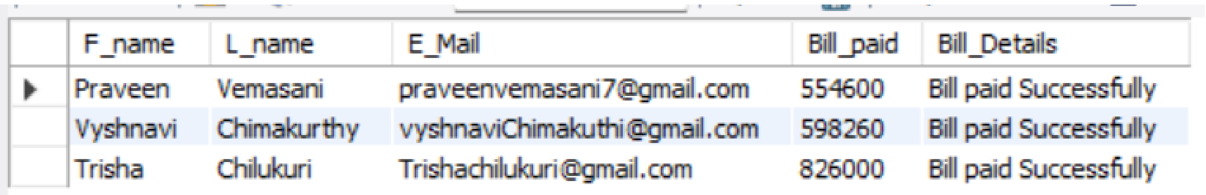
SQL Queries

Query 1: Find the details of the customers who paid bill more than 5,00,000$.

SQL Command:

```
select F_name,L_name,E_Mail,Bill_paid,Bill_Details
from customer
inner join Bill
on customer.Customer_ID=Bill.Customer_ID
where Bill.Bill_paid > 500000;
```

Output:



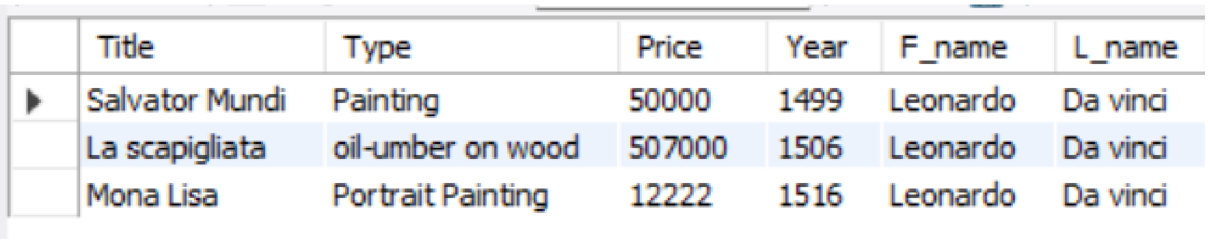
F_name	L_name	E_Mail	Bill_paid	Bill_Details
Praveen	Vemasani	praveenvemasani7@gmail.com	554600	Bill paid Successfully
Vyshnavi	Chimakurthy	vyshnaviChimakuthi@gmail.com	598260	Bill paid Successfully
Trisha	Chilukuri	Trishachilukuri@gmail.com	826000	Bill paid Successfully

Query 2: Find the Artworks drawn by Leonardo Da vinci.

SQL Command :

```
select Title,Type,Price,Year,F_name,L_name
from artist_info
inner join artworks
on artworks.Artist_ID=artist_info.Artist_ID
where artist_info.F_name="Leonardo";
```

Output:



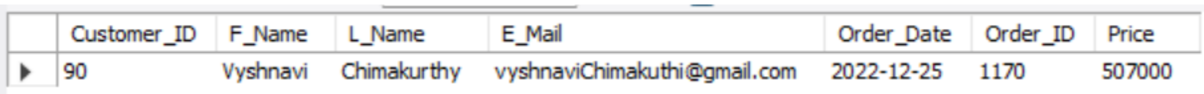
Title	Type	Price	Year	F_name	L_name
Salvator Mundi	Painting	50000	1499	Leonardo	Da vinci
La scapigliata	oil-umber on wood	507000	1506	Leonardo	Da vinci
Mona Lisa	Portrait Painting	12222	1516	Leonardo	Da vinci

Query 3: Find the details of Customer who ordered artwork on 2022-12-25

SQL Command :

```
select customer.Customer_ID,F_Name,L_Name,E_Mail,Order_Date,Order_ID,Price
from customer
inner join order_info
on order_info.Customer_ID=customer.Customer_ID
where order_info.Order_Date="2022-12-25";
```

Output:



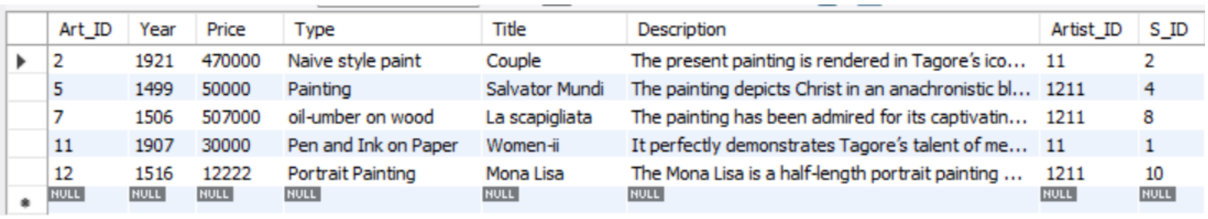
	Customer_ID	F_Name	L_Name	E_Mail	Order_Date	Order_ID	Price
▶	90	Vyshnavi	Chimakurthy	vyshnaviChimakuthi@gmail.com	2022-12-25	1170	507000

Query 4: Find the Artworks, which were made in between 1400 and 1950

SQL Command :

```
select * from artworks
where Year>1400 and year<1950;
```

Output:



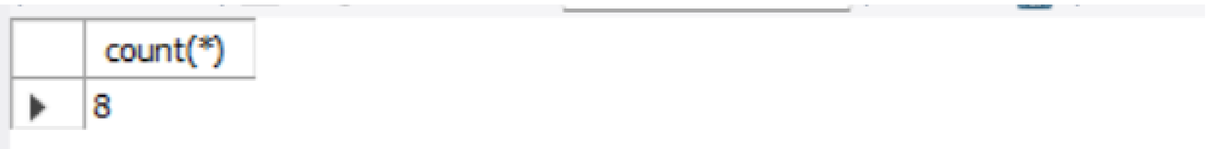
	Art_ID	Year	Price	Type	Title	Description	Artist_ID	S_ID
▶	2	1921	470000	Naive style paint	Couple	The present painting is rendered in Tagore's ico...	11	2
	5	1499	50000	Painting	Salvator Mundi	The painting depicts Christ in an anachronistic bl...	1211	4
	7	1506	507000	oil-umber on wood	La scapigliata	The painting has been admired for its captivatin...	1211	8
	11	1907	30000	Pen and Ink on Paper	Women-ii	It perfectly demonstrates Tagore's talent of me...	11	1
	12	1516	12222	Portrait Painting	Mona Lisa	The Mona Lisa is a half-length portrait painting ...	1211	10
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Query 5: Find the total number of artworks

SQL Command :

```
select count(*)
from artworks;
```

Output:



	count(*)
▶	8

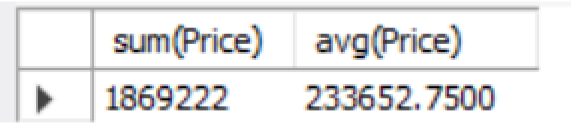
Query 6: Find the sum of prices,average price of artworks

SQL Command :

select sum(Price),avg(Price)

from artworks;

Output:



sum(Price)	avg(Price)
1869222	233652.7500

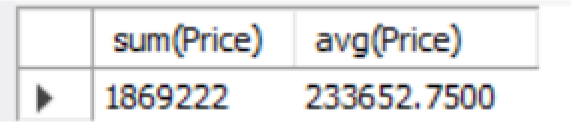
Query 6: Find the sum of prices,average price of artworks

SQL Command :

select sum(Price),avg(Price)

from artworks;

Output:



sum(Price)	avg(Price)
1869222	233652.7500

Query 7: Find the complete details of every Artist;

SQL Command :

select F_name,L_name,Style,Contact,artist_address.Pincode,State,City,Country

from artist_info

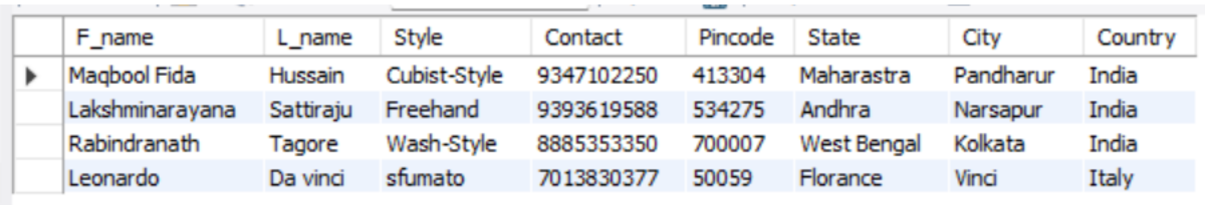
join artist_address

on artist_address.Pincode=artist_info.Pincode

join artist_contact

on artist_contact.Artist_ID=artist_info.Artist_ID;

Output:



F_name	L_name	Style	Contact	Pincode	State	City	Country
Maqbool Fida	Hussain	Cubist-Style	9347102250	413304	Maharastra	Pandharur	India
Lakshminarayana	Sattiraju	Freehand	9393619588	534275	Andhra	Narsapur	India
Rabindranath	Tagore	Wash-Style	8885353350	700007	West Bengal	Kolkata	India
Leonardo	Da vinci	sfumato	7013830377	50059	Florance	Vinci	Italy

References

1. https://www.google.com/url?sa=i&url=https%3A%2F%2Fgithub.com%2FTalRodin%2Fart_gallery_database&psig=AOvVaw0Fzrjgx6ZWhVBKbRL3IxXb&ust=1670610219998000&source=images&cd=vfe&ved=0CA8QjRxqFwoTCMj8qZfS6vsCFQAAAAAdAAAAABAE
2. https://www.javatpoint.com/dbms-normalization
3. www.smartdraw.com
4. https://erdplus.com/