DATABASE MANAGEMENT SYSTEM

PROJECT

ART MUSEUM AND SHOWROOM

MANAGEMENT SYSTEM

SUBMITTED TO: **SUBMITTED BY:**

Mr. Rohit Beniwal   **Vansh Dhawan - 2K22/CO/489**

                              Yash Jain   - 2K22/CO/511

INDEX

INTRODUCTION

SQL, short for Structured Query Language, serves as a means to retrieve data from databases, as well as to insert, update, or delete data within them, or manipulate database metadata.

In contrast to procedural languages, which typically necessitate specifying operational details like opening and closing tables, loading and searching indexes, or managing buffers and writing data to file systems, SQL is often described as non-procedural. It operates at a higher conceptual level, abstracting away such low-level details.

Commonly used statements are grouped into the following categories:

Data Query Language (DQL):

- SELECT: Used to retrieve certain records from one or more tables.

Data Manipulation Language (DML):

- INSERT: Used to create a record.
- UPDATE: Used to change certain records.
- DELETE: Used to delete certain records.

Data Definition Language (DDL):

- CREATE: Used to create a new table, a view of a table, or other object in the database.
- ALTER: Used to modify an existing database object, such as a table.
- DROP: Used to delete an entire table, a view of a table, or other object in the database.

Data Control Language (DCL):

- GRANT: Used to give a privilege to someone.
- REVOKE: Used to take back privileges granted to someone.

The Art Museum and Showroom Database Management System is designed to facilitate the efficient management of art galleries, encompassing both user and gallery databases. This project streamlines the handling of orders and showcases comprehensive details pertaining to customers, artists, and artworks. Included within is an SQL file for straightforward integration into the database module.

OBJECTIVE OF THE PROJECT

The main objective of creating an Art Museum and Showroom database project is :-

- The Art Museum and Showroom database project aims to efficiently manage the details of galleries, exhibitions, artworks, and artists while overseeing sales and inventory within the gallery. Its primary objective is to develop an application program that minimizes manual tasks.

- The system tracks sales details of artworks, including customer information, to streamline gallery management. By providing comprehensive information and descriptions of available artworks, it enhances gallery management efficiency, enabling the organization to maintain a computerized record of its art collection.

- Moreover, the system assists in resource utilization by maintaining a list of customers, their purchased artworks, and associated investments. Additionally, it records exhibition details and sales made during them. The project's ultimate goal is to reduce paperwork and save time in art gallery database management, thereby enhancing efficiency and reducing workload.

ER DIAGRAM

a) Purpose of ER Model:

- Outlines interconnected elements within a domain for understanding complex structures.
- Facilitates comprehension of complex knowledge structures.

b) Entity Types and Relationships:

- Entity types classify elements of interest.
- Relationships define associations between instances of these types.

c) Definition of Entities:

- Entities represent distinct, independently identifiable elements.
- Serve as abstractions from domain complexities.

d) Representation of Attributes:

- Attributes depicted as ovals.
- Linked to single entity or relationship sets.
- Delineate characteristics within the model.

e) ER Diagram Usage:

- ER diagrams visually represent entities and interconnections.
- Crucial for organizing data within databases or information systems.

f) Cardinality Constraints:

- Line styles and symbols convey participation, key constraints, and bijectivity.
- Facilitate comprehension of data relationships.

g) Graphical Representation:

- ER models offer graphical portrayal of complex data structures.
- Aid in analysis and understanding of intricate data relationships.

ER DIAGRAM of Art Museum and Showroom DATABASE
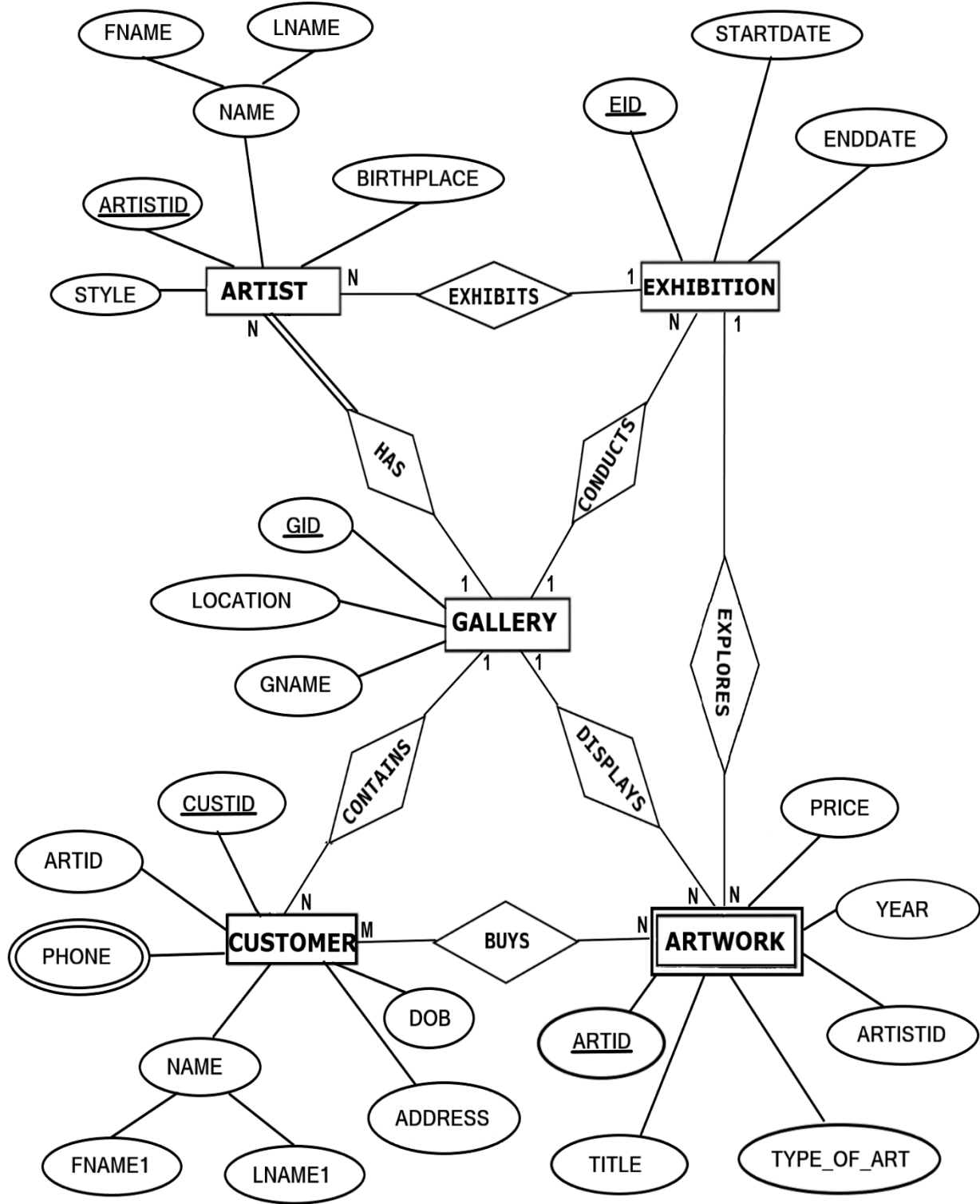
MAPPING OF ER DIAGRAM TO RELATIONS

STEP 1: Mapping of Regular Entities

For each regular entity type E in the ER schema, create relation R that includes all simple attributes of E.

GALLERY

GID	GNAME	LOCATION

EXHIBITION

EID	STARTDATE	ENDDATE

ARTIST

ARTISTID	FNAME	LNAME	BIRTHPLACE	STYLE

CUSTOMER

CUSTID	ARTID	FNAME1	LNAME1	ADDRESS	PHONE	DOB
	FK					

STEP 2 : Mapping of Weak Entity Types

ARTWORK

ARTID	ARTISTID	TITLE	TYPE_OF_ART	YEAR	PRICE
	FK				

STEP 3: Mapping of 1:1 Relationship

Identify the relation S that represents the participating entity type at the 1-side of the relationship type.

Include as foreign key in S the primary key of the relations T that represents the other entity type participating in R.

For each binary 1:1 relationship type R in ER schema, identify the relations S and T that correspond to the entity types participating in R if any.

There are **no** 1:1 relationship.

STEP 4 : Mapping of 1:N Relationship

EXHIBITION

EID	STARTDATE	ENDDATE	GID
			FK

ARTIST

ARTISTID	FNAME	LNAME	BIRTHPLACE	STYLE	EID	GID	CUSTID
					FK	FK	FK

CUSTOMER

CUSTID	ARTID	FNAME1	LNAME1	ADDRESS	DOB	GID
	FK					FK

ARTWORK

ARTID	ARTISTID	TITLE	TYPE_OF_ART	YEAR	PRICE	EID	GID
	FK					FK	FK

STEP 5 : Mapping of M:N Relationship

Create a new relation S to represent R.

Include as foreign key attributes in S the primary key of the relations that represents the participating entity types their combination will form the primary key of S.

Also, include any simple attributes of the M:N relationship type as attributes of S.

STEP 6: Mapping of Multi-Valued Attributes

For each multivalued attributes A, create a new relation R. This relation R will include an attribute corresponding to A, plus the primary key attribute K-as a foreign key in R-of the relation that represents the entity type of relationship type that has A as an attribute.

The Primary Key of R is the combination of A and K. If the multivalued attribute is composite, we include its simple components.

CONTACTS

CUSTID	PHONE

STEP 7: Mapping of N-Ary Relationship Types

For each n-ary relationship type R, where n>2 create a new relationship S to represent R. λ include as foreign key attributes in S the primary keys of the relations that represent the participating entity types.

λ also includes any simple attributes of the n-ary relationship type (or simple components of composite attributes) as attributes of S.

There are **no** n-ary relationship types.

IMPLEMENTATION

CREATION OF TABLES

1. CREATING GALLERY TABLE

```
CREATE TABLE GALLERY
(GID VARCHAR(20) PRIMARY KEY,
GNAME CHAR(20),
LOCATION CHAR(20));
```

```
mysql> desc gallery;
+----------+-------------+------+-----+----------+-------+
| Field    | Type        | Null | Key | Default  | Extra |
+----------+-------------+------+-----+----------+-------+
| gid      | varchar(26) | NO   | PRI | not null |       |
| gname    | varchar(24) | YES  |     | NULL     |       |
| location | varchar(26) | YES  |     | NULL     |       |
+----------+-------------+------+-----+----------+-------+
3 rows in set (0.00 sec)
```

2. CREATE EXHIBITION TABLE

```
CREATE TABLE EXHIBITION
(EID VARCHAR(20) PRIMARY KEY,
GID VARCHAR(20),
STARTDATE DATE,
ENDDATE DATE,
FOREIGN KEY(GID) REFERENCES GALLERY(GID) ON DELETE CASCADE);
```

```
mysql> desc exhibition;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| eid       | varchar(20) | NO   | PRI | NULL    |       |
| gid       | varchar(20) | YES  | MUL | NULL    |       |
| startdate | date        | YES  |     | NULL    |       |
| enddate   | date        | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

3. CREATE ARTWORK TABLE

```
CREATE TABLE ARTWORK
(ARTID VARCHAR(20) PRIMARY KEY,
TITLE VARCHAR(20),
YEAR INT,
```

TYPE_OF_ART VARCHAR(20),
PRICE INT,
EID VARCHAR(20), GID VARCHAR(20),

FOREIGN KEY(EID) REFERENCES EXHIBITION(EID) ON DELETE CASCADE,
FOREIGN KEY(GID) REFERENCES GALLERY(GID) ON DELETE CASCADE);

```
mysql> desc artwork;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| artid       | varchar(20) | NO   | PRI | NULL    |       |
| title       | varchar(20) | YES  |     | NULL    |       |
| year        | varchar(5)  | YES  |     | NULL    |       |
| type_of_art | varchar(20) | YES  |     | NULL    |       |
| price       | varchar(15) | YES  |     | NULL    |       |
| eid         | varchar(20) | YES  | MUL | NULL    |       |
| gid         | varchar(20) | YES  | MUL | NULL    |       |
| artistid    | varchar(20) | YES  | MUL | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
8 rows in set (0.00 sec)
```

4. CREATE CUSTOMER TABLE

CREATE TABLE CUSTOMER
(CUSTID VARCHAR(20) PRIMARY KEY,
GID VARCHAR(20),
ARTID VARCHAR(20),
FNAME1 CHAR(20),
LNAME1 CHAR(20),
DOB DATE,
ADDRESS CHAR(20),
FOREIGN KEY(GID) REFERENCES GALLERY(GID) ON DELETE CASCADE,
FOREIGN KEY(ARTID) REFERENCES GALLERY(ARTID) ON DELETE CASCADE);

5.
```
mysql> desc customer;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| custid  | varchar(20) | NO   | PRI | NULL    |       |
| gid     | varchar(20) | YES  | MUL | NULL    |       |
| artid   | varchar(20) | YES  | MUL | NULL    |       |
| fname   | char(25)    | YES  |     | NULL    |       |
| lname   | char(25)    | YES  |     | NULL    |       |
| dob     | date        | YES  |     | NULL    |       |
| address | char(25)    | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
7 rows in set (0.00 sec)
```

CREATE ARTIST TABLE

CREATE TABLE ARTIST
(ARTISTID VARCHAR(20) PRIMARY KEY,

```
GID VARCHAR(20),
CUSTID VARCHAR(20),
EID VARCHAR(20),
FNAME CHAR(20),

LNAME CHAR(20),
BIRTHPLACE CHAR(20),
STYLE CHAR(20),
FOREIGN KEY(GID) REFERENCES GALLERY(GID) ON DELETE CASCADE,
FOREIGN KEY (CUSTID) REFERENCES CUSTOMER(CUSTID) ON DELETE
CASCADE,
FOREIGN KEY(EID) REFERENCES EXHIBITION(EID) ON DELETE CASCADE);

ALTER TABLE ARTWORK ADD ARTISTID VARCHAR(20);

ALTER TABLE ARTWORK
ADD FOREIGN KEY (ARTISTID) REFERENCES ARTIST(ARTISTID) ON DELETE
CASCADE;
```

```
mysql> desc artist;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| artistid  | varchar(20) | NO   | PRI | NULL    |       |
| gid       | varchar(20) | YES  | MUL | NULL    |       |
| custid    | varchar(20) | YES  | MUL | NULL    |       |
| eid       | varchar(20) | YES  | MUL | NULL    |       |
| fname1    | char(25)    | YES  |     | NULL    |       |
| lname1    | char(25)    | YES  |     | NULL    |       |
| birthplace| char(25)    | YES  |     | NULL    |       |
| style     | char(25)    | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
8 rows in set (0.00 sec)
```

1. CREATE CONTACTS TABLE

```
CREATE TABLE CONTACTS
(CUSTID VARCHAR(20),
PHONE VARCHAR(12),
FOREIGN KEY (CUSTID) REFERENCES CUSTOMER(CUSTID) ON DELETE
CASCADE);
```

```
mysql> desc contacts;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| CUSTID | varchar(20) | YES  | MUL | NULL    |       |
| PHONE  | varchar(12) | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

INSERTION OF TUPLES

1. INSERTION OF GALLERY TABLE

INSERT INTO GALLERY VALUES('NG123','National Gallery', 'Washington');
INSERT INTO GALLERY VALUES('BM123','British Museum', 'London');
INSERT INTO GALLERY VALUES('JG123','Jahangir Gallery', 'Mumbai');
INSERT INTO GALLERY VALUES('TLM123','The Louvre Museum', 'Paris');
INSERT INTO GALLERY VALUES('MM123','Metropolitan Museum', 'New York');

```
mysql> select * from gallery;
+--------+---------------------+------------+
| gid    | gname               | location   |
+--------+---------------------+------------+
| NG123  | NATIONAL GALLERY    | Washington |
| BM123  | BRITISH MUSEUM      | London     |
| JG123  | JAHANGIR GALLERY    | Mumbai     |
| TLM123 | THE LOUVRE MUSEUM   | Paris      |
| MM123  | METROPOLITAN MUSEUM | New York   |
+--------+---------------------+------------+
5 rows in set (0.00 sec)
```

2. INSERTION OF EXHIBITION TABLE

INSERT INTO EXHIBITION VALUES('G123','NG123','2018-12-01','2018-12-15');
INSERT INTO EXHIBITION VALUES('H123','BM123','2018-12-21','2019-01-05');
INSERT INTO EXHIBITION VALUES('I123','MM123','2019-01-25','2019-02-05');
INSERT INTO EXHIBITION VALUES('J123','TLM123','2018-12-15','2019-01-15');
INSERT INTO EXHIBITION VALUES('K123','JG123','2019-03-09','2019-03-27');

```
mysql> select * from exhibition;
+------+--------+------------+------------+
| eid  | gid    | startdate  | enddate    |
+------+--------+------------+------------+
| H123 | BM123  | 2018-12-21 | 2019-01-05 |
| I123 | MM123  | 2019-01-25 | 2019-02-05 |
| G123 | NG123  | 2018-12-01 | 2018-12-15 |
| J123 | TLM123 | 2018-12-15 | 2019-01-15 |
| K123 | JG123  | 2019-03-09 | 2019-03-27 |
+------+--------+------------+------------+
5 rows in set (0.00 sec)
```

3. INSERTION OF ARTWORK TABLE

INSERT INTO ARTWORK
VALUES('AW12','Mona Lisa','1503','Painting','10,00,00,000','G123','NG123','AD11');
INSERT INTO ARTWORK

VALUES('AW34','Poppies','1873','Painting','1,50,00,000','H123','MM123','AD22');
INSERT INTO ARTWORK
VALUES('AW56','Guernica','1937','Painting','2,50,00,000','I123','TLM123','AD55');

INSERT INTO ARTWORK
VALUES('AW78','The Night Watch','1642','Painting','90,00,000','J123','BM123','AD88');
INSERT INTO ARTWORK
VALUES('AW90','Two Sisters','2010','Sculpture','2,00,000','K123','JG123','AD00');

```
mysql> select * from artwork;
+-------+----------------+------+-------------+--------------+------+--------+----------+
| artid | title          | year | type_of_art | price        | eid  | gid    | artistid |
+-------+----------------+------+-------------+--------------+------+--------+----------+
| AW12  | Mona Lisa      | 1503 | Painting    | 10,00,00,000 | G123 | NG123  | AD11     |
| AW34  | Poppies        | 1873 | Painting    | 1,50,00,000  | H123 | MM123  | AD22     |
| AW56  | Guernica       | 1937 | Painting    | 2,50,00,000  | I123 | TLM123 | AD55     |
| AW78  | The Night Watch| 1642 | Painting    | 90,00,000    | J123 | BM123  | AD88     |
| AW90  | Two Sisters    | 2010 | Sculpture   | 2,00,000     | K123 | JG123  | AD00     |
+-------+----------------+------+-------------+--------------+------+--------+----------+
5 rows in set (0.00 sec)
```

4. INSERTION OF CUSTOMER TABLE

INSERT INTO CUSTOMER VALUES
('AT2000','MM123','AD22','Akshay','Thakur','2000-04-16','New York');
INSERT INTO CUSTOMER
VALUES('AR1998','TLM123','AD55','Ashutosh','Ranjan','1998-02-04','Paris');
INSERT INTO CUSTOMER
VALUES('AD1998','BM123','AD88','Ayush','Dhar','1998-09-28','London');
INSERT INTO CUSTOMER
VALUES('AM1994','JG123','AD00','Avanish','Mehta','1994-10-05','Mumbai');
INSERT INTO CUSTOMER VALUES
('PM1996','NG123','AD11','Prashant','Mehta','1996-06-18','Washington');

```
mysql> select * from customer;
+--------+--------+-------+----------+--------+------------+------------+
| custid | gid    | artid | fname    | lname  | dob        | address    |
+--------+--------+-------+----------+--------+------------+------------+
| AT2000 | MM123  | AD22  | Akshay   | Thakur | 2000-04-16 | New York   |
| AR1998 | TLM123 | AD55  | Ashutosh | Ranjan | 1998-02-04 | Paris      |
| AD1998 | BM123  | AD88  | Ayush    | Dhar   | 1998-09-28 | London     |
| AM1994 | JG123  | AD00  | Avanish  | Mehta  | 1994-10-05 | Mumbai     |
| PM1996 | NG123  | AD11  | Prashant | Mehta  | 1996-06-18 | Washington |
+--------+--------+-------+----------+--------+------------+------------+
5 rows in set (0.00 sec)
```

5. INSERTION OF ARTIST TABLE

INSERT INTO ARTIST
VALUES('ART1','MM123','AT2000','AD22','Georgia','O Keeffe','USA','Oil on Canvas');
INSERT INTO ARTIST
VALUES('ART2','TLM123','AR1998','AD55','Pablo','Picasso','Spain','Analytic Cubism');
INSERT INTO ARTIST VALUES
('ART3','BM123','AD1998','AD88','Rembrandt','van Rijn','Netherlands','Oil Painting');

INSERT INTO ARTIST
VALUES('ART4','JG123','AM1994','AD00','Theodore','Chasseriau','France','Oil Painting');
INSERT INTO ARTIST
VALUES('ART5','NG123','PM1996','AD11','Leonardo','da Vinci','Italy','High Renaissance');

```
mysql> select * from artist;
+----------+--------+--------+------+-----------+------------+-------------+-----------------+
| artistid | gid    | custid | eid  | fname1    | lname1     | birthplace  | style           |
+----------+--------+--------+------+-----------+------------+-------------+-----------------+
| ART1     | MM123  | AT2000 | AD22 | Georgia   | O Keeffe   | USA         | Oil on Canvas   |
| ART2     | TLM123 | AR1998 | AD55 | Pablo     | Picasso    | Spain       | Analytic Cubism |
| ART3     | BM123  | AD1998 | AD88 | Rembrandt | van Rijn   | Netherlands | Oil Painting    |
| ART4     | JG123  | AM1994 | AD00 | Theodore  | Chasseriau | France      | Oil Painting    |
| ART5     | NG123  | PM1996 | AD11 | Leonardo  | da Vinci   | Italy       | High Renaissance|
+----------+--------+--------+------+-----------+------------+-------------+-----------------+
5 rows in set (0.00 sec)
```

6. INSERTION OF CONTACTS TABLE

INSERT INTO CONTACTS VALUES('AT2000', '9456805776');
INSERT INTO CONTACTS VALUES('AR1998', '8073271337');
INSERT INTO CONTACTS VALUES('AD1998', '9980904736');
INSERT INTO CONTACTS VALUES('AM1994', '7737564076');
INSERT INTO CONTACTS VALUES('PM1996', '8002391707');

```
mysql> select * from contacts;
+--------+------------+
| CUSTID | PHONE      |
+--------+------------+
| AT2000 | 9456805776 |
| AR1998 | 8073271337 |
| AD1998 | 9980904736 |
| AM1994 | 7737564076 |
| PM1996 | 8002391707 |
+--------+------------+
5 rows in set (0.00 sec)
```

CONCLUSION

The implementation of the Art Museum and Showroom Database Management System has yielded significant results in enhancing the efficiency and organization of art gallery operations. By centralizing the management of users and gallery databases, the system has streamlined processes related to order management and the display of comprehensive customer, artist, and artwork details. Through the systematic storage of data on artists, artworks, and customers, the project has provided gallery administrators with a cohesive platform for easy access to crucial information, facilitating informed decision-making and smoother operations.

Moreover, the project's features, including the ability to store detailed information such as artist names, birthplaces, styles of art, artwork titles, creation years, types, and prices, have enriched the gallery management experience. This comprehensive data management capability enables gallery administrators to maintain a thorough record of their inventory and clientele, fostering better communication and engagement with customers and artists alike. Overall, the Art Museum and Showroom Database Management System has proven to be a valuable asset in optimizing gallery operations, enhancing customer satisfaction, and fostering a more organized and efficient art gallery environment.

REFERENCES

I.   Fundamentals of Database System, 7th Edition

          -By Elmasri Ramez and Navathe Shamkanth

II.   Database System Concepts

          -By Abraham Silberschatz

III.   For MySQL references

      a.   https://www.w3schools.com/

      b.   https://www.stackoverflow.com

      c.   https://www.youtube.com