

**PROJECT FOR SQL MODULE**

**ART GALLERY COMPETITION MANAGEMENT**

**Institute name** = Itvedant education Pvt. Ltd.

**Name** = Ayush Dnyaneshwar Gunjal

**Email address** = [ayushgunjal2002@gmail.com](mailto:ayushgunjal2002@gmail.com)

**Date of submission** = 27-07-2024

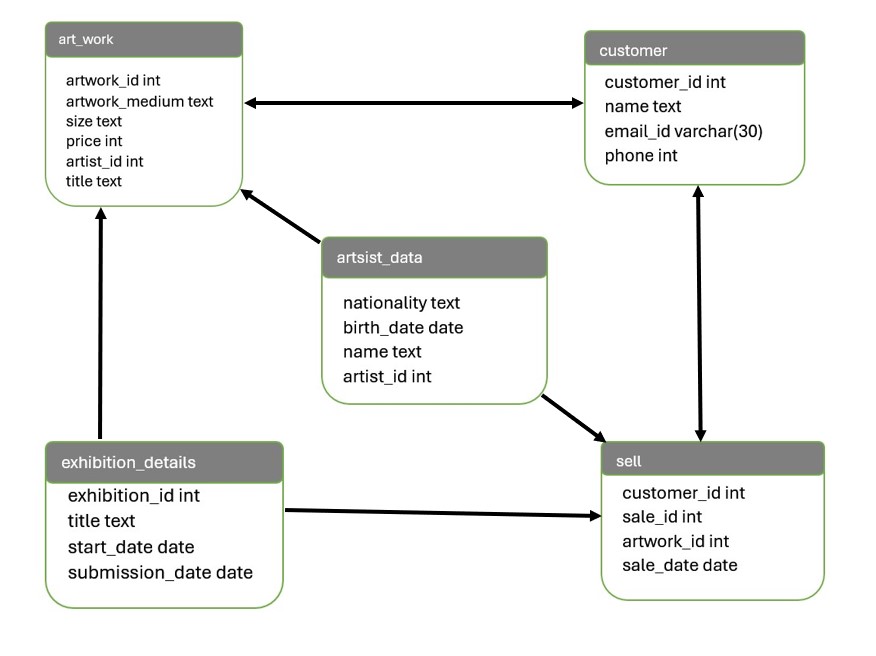
**Project aims:**

* **Artist Commission Management:** Calculate and track artist commissions on artwork sales.
* **Exhibition Scheduling:** Schedule and manage exhibitions, including booking dates and artwork selection.
* **Artwork Conservation Management:** Track and manage artwork conservation and restoration activities.
* **Marketing and Promotion:** Support marketing and promotional activities, including email campaigns and social media integration.
* **Financial Management:** Integrate with financial systems to manage artwork sales, commissions, and expenses.
* **Artist Management:** Store and manage information about artists, including their personal details, artworks, and exhibitions.
* **Artwork Management:** Store and manage information about artworks, including their title, medium, size, price, and artist.
* **Exhibition Management:** Store and manage information about exhibitions, including their title, start and end dates, and participating artworks.
* **Customer Management:** Store and manage information about customers, including their personal details and artwork purchases.

**Project objective:**

The objective of the Art Gallery Management Database project is to design and implement a comprehensive database system that efficiently manages and stores information related to the art gallery's operations, including artist and artwork details, exhibitions, customer interactions, sales, and conservation activities. The system aims to streamline gallery operations, improve data accuracy and accessibility, enhance customer engagement, and support business growth through data-driven insights and reporting. By achieving this objective, the project will provide a robust and scalable solution for the art gallery to manage its diverse activities, make informed decisions, and maintain its competitive edge in the art industry.

**ER diagram of the project:**

****

**ER diagram description:**

**Entities:**

* Artist (Artist\_ID, Name, Birthdate, Nationality)
* Artwork (Artwork\_ID, Title, Medium, Size, Price, Artist\_ID)
* Exhibition (Exhibition\_ID, Title, Start\_Date, End\_Date)
* Customer (Customer\_ID, Name, Email, Phone)
* Sale (Sale\_ID, Artwork\_ID, Customer\_ID, Sale\_Date, Price)

**Relationships:**

* An artist can create many artworks (one-to-many).
* An artwork is created by one artist (many-to-one).
* An exhibition can feature many artworks (one-to-many).
* An artwork can be part of many exhibitions (many-to-many).
* A customer can purchase many artworks (one-to-many).
* A sale is associated with one artwork and one customer (many-to-one).

**Table description:**

1. **art\_work**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** |
| artwork\_id | Int | Yes | Primary key | Null |  |
| artwork\_medium | Text | Yes |  | Null |  |
| size | Text | Yes |  | Null |  |
| price | Int | Yes |  | Null |  |
| artist\_id | Int | Yes | Primary key | Null |  |
| Title | Varchar(100) | yes |  | Null |  |

1. **artsist\_data**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** |
| nationality | text | Yes |  | Null |  |
| birth\_date | date | Yes |  | Null |  |
| name | text | Yes |  | Null |  |
| artist\_id | int | Yes |  | null |  |

1. **customer:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **type** | **Null** | **Key** | **Default** | **Extra** |
| customer\_id | int | Yes | Primary key | Null |  |
| name | text | Yes |  | Null |  |
| email\_id | varchar(50) | Yes |  | Null |  |
| Phone | Int | Yes |  | Null |  |

1. **exhibition\_details**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** |
| Exhibition\_id | Int | Yes | Primary key | Null |  |
| title | Text | Yes |  | Null |  |
| Start\_date | Date | Yes |  | Null |  |
| Submission\_date | Date | Yes |  | Null |  |

1. **sell**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** |
| customer\_id | int | Yes |  | Null |  |
| sale\_id | int | Yes | Primary | Null |  |
| artwork\_id | int | Yes |  | Null |  |
| sale\_date | Date | Yes |  | Null |  |

**Commands:**

**create database exhibition;**

**use exhibition;**

**create table** art\_work(

artwork\_id int primary key,

artwork\_medium text,

size text,

price int,

artist\_id int primary key,

title text

);

**insert into** art\_Work **values**

(2327, ‘oil on canvas’, ”extra large”, 224483, 180, ‘Palette of Ages’),

(2956, ‘water colour’, ‘medium’, 119370, 964, ‘Carved Whispers’),

(2656, ‘history, ‘small’, 415840, 135, ‘Frozen Moments’),

(2328, ‘water colour’, ‘medium’, 293659, 802’, ‘Tangled Threads’),

(2715, ‘sculpture’, ‘medium’, 40536, 557, ‘Aqua Serenade’),

(2464, ‘history’, ‘small’, 356752, 34, ‘Tangled Threads’),

(2584, ‘oil on canvas’, ‘small’, 122590, 607, ‘Luminous Horizons’),

(2060, ‘mixed media’, ‘large’, 21684, 609, ‘Eternal Brushstrokes’),

(2060, ‘mixed media’, ‘small ‘, 272020, 662, ‘Pixel Reverberations’),

(2210, ‘photography’, ‘extra large’, 40601, 381, ‘Pixel Reverberations’);

Select \* from art\_work;

**Output:**

A screenshot of a computer

Description automatically generated

**create table** artsist\_data(

Nationality text,

Birth\_date date,

Name varchar(40),

Artist\_id int

);

**insert into** artsist\_data **values**

(‘Russia’, 15-12-2009, ‘Maggi Cohane’, 196),

(‘Canada’, 10-08-2012, ‘Carmelita Slatten’, 92),

(‘United States’, 14-12-2012, ‘Raul Lightowlers’, 599),

(‘United States’, 15-06-2013, ‘Karie Legon’, 167),

(‘Canada’, 22-03-2011, ‘Verena Cumbridge’, 646),

(‘Russia’, 26-04-1999, ‘Harv Lorait’, 353),

(‘Russia’, 27-01-2004, ‘Sunny Clewett’, 836),

(‘Germany’, 18-09-2006, ‘Giacinta Earsman’, 825),

(‘Russia’, 06-09-2010, ‘Koenraad Cavnor’, 383),

(‘Russia’, 16-05-2000, ‘Arlette Seniour’, 270);

select \* from artsist\_data;

**A screenshot of a computer

Description automatically generatedoutput:**

**create table** customer(

Customer\_id int primary key,

Name varchar(50),

Email\_id varchar(50)’

Phone int

);

**insert into** customer **values**

(3747, ‘Siobhan Morgon’, ‘smorgon0@desdev.cn’ 867 543 0616),

(3341, ‘Bobette Cree’, ‘bcree1@mtv.com’ 663 815 3226),

(3428, ‘Harmon Pahl’, ‘hpahl2@simplemachines.org’ 825 678 0059),

(3506, ‘Esme Horrigan’, ‘ehorrigan3@auda.org.au’ 923 497 8518),

(3763, ‘Celinda Sidaway’, ‘csidaway4@twitpic.com’ 794 880 8433),

(3906, ‘Beaufort Donat’, ‘bdonat5@statcounter.com’ 816 383 5620),

(3785, ‘Rex Chastanet’, ‘rchastanet6@ft.com’ 766 123 7962),

(3742, ‘Kerwin Braddick’, ‘kbraddick7@netvibes.com’ 405 260 5685),

(3749, ‘Livia Ossipenko’, ‘lossipenko8@vistaprint.com’ 424 631 1004),

(3887, ‘Augustine Lightwing’, ‘alightwing9@tuttocitta.it’ 129 520 1144);

Select \* from customer;

**Output:**

A screenshot of a computer

Description automatically generated

**create table** exhibition\_details(

Exhibition\_id int primary key,

Title varchar(100),

Start\_date date,

Submission\_date date

);

**insert into** exhibition\_details **values**

(1020, ‘Palette of Ages’, 06-01-2024, 08-01-2024),

(1470, ‘Carved Whispers’, 03-01-2024, 27-01-2024),

(1158, ‘Frozen Moments’, 03-01-2024, 13-01-2024),

(1535, ‘Tangled Threads’, 07-01-2024, 02-01-2024),

(1208, ‘Aqua Serenade’, 07-01-2024, 06-01-2024),

(1093, ‘Tangled Threads’, 09-01-2024, 06-01-2024),

(1625, ‘Luminous Horizons’, 05-01-2024, 12-01-2024),

(1863, ‘Eternal Brushstrokes’, 05-01-2024, 12-01-2024),

(1537, ‘Pixel Reverberations’, 05-01-2024, 09-01-2024),

(1714, ‘Pixel Reverberations’, 07-01-2024, 14-01-2024);

Select \* from exhibition\_details;

**Output:**

**A screenshot of a computer

Description automatically generated**

**create table** sell(

Customer\_id int ,

Sale\_id int primary key,

Artwork\_id int,

Sale\_date date,

);

**insert into** sell **values**

(3496, 4196, 2431, 18-01-2024),

(3877, 4190, 2566, 27-01-2024),

(3933, 4991, 2902, 20-01-2024),

(3212, 4018, 2157, 04-01-2024),

(3817, 4286, 2949, 10-01-2024),

(3303, 4429, 2878, 19-01-2024),

(3317, 4273, 2512, 20-01-2024),

(3724, 4436, 2748, 29-01-2024),

(3325, 4634, 2669, 23-01-2024),

(3246, 4352, 2712, 19-01-2024);

Select \* from sell;

**Output:**

**A screenshot of a computer

Description automatically generated**

**JOIN QUERIES:**

**1) what is the nationality of the artist having the ID 167**

select nationality, name from art\_work A inner join artsist\_data D on A.artist\_iD = D.artist\_id where A.artist\_id = 167;

**output:**

A screenshot of a computer

Description automatically generated

**2) art having the artwork\_id 2157 was get by which customer**

select name, customer\_id, email\_id, phone from customer inner join sell on customer.customer\_id = sell.customer\_id where sell.artwork\_id = 2157;

**output:**

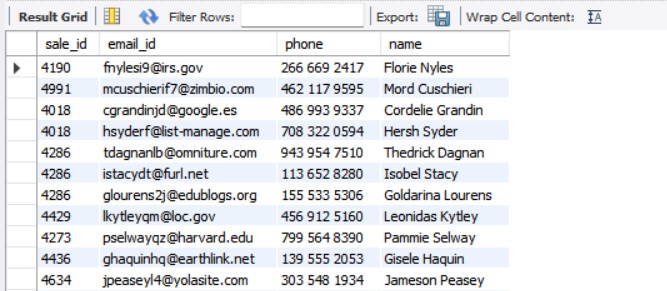
A screenshot of a computer

Description automatically generated

**3) customer name mail phone having sale id using join**

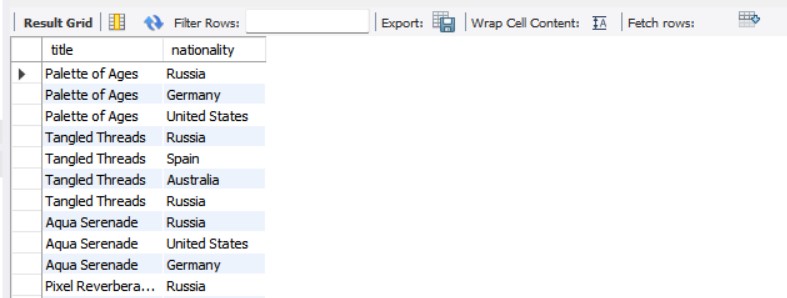
select sale\_id , email\_id, phone, name from customer join sell on customer.customer\_id = sell.customer\_id;

**output:**



**4) title of the project and their nationality**

select title , nationality from art\_work A join artsist\_data D on A.artist\_id = D.artist\_id;

**output:**

**5) find average price of the artwork in the medium category**

select avg(price) , nationality from art\_work inner join artsist\_data on art\_work.artist\_id = artsist\_data.artist\_id where nationality = "Russia";

**output:**

A white rectangular object with a black border

Description automatically generated with medium confidence

**6) right join use of the two table to show name with there medium and nationality**

select name , artwork\_medium , nationality from art\_work A right join artsist\_data AD on A.artist\_id = AD.artist\_id;

**output:**

A screenshot of a computer

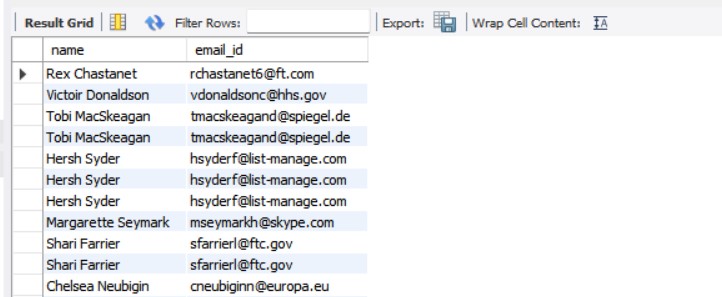
Description automatically generated

**7) joining 3 tables to extract name of the customer having spend more than 300000 on artwork**

SELECT c.name, c.email\_id FROM customer c JOIN sell s ON c.customer\_id = s.customer\_id JOIN art\_work a ON s.artwork\_id = a.artwork\_id

WHERE a.price > 300000;

**Output:**



**SUBQUERIE QUERIES:**

**1) 2nd highest price of the art\_work**

select max(price) from art\_work;

select max(price) from art\_work where price < (select max(price) from art\_work);

**output:**

A screenshot of a computer

Description automatically generated

**2) Retrieve the names of customers who have purchased artworks worth more than $10,000**

select \* from art\_work where price > 10000;

select artwork\_medium from art\_work where price > any(select artist\_id from art\_work where price > 10000);

**output:**

A screenshot of a computer

Description automatically generated

**3) data of the artist whos salary is greater than the one who are from united states**

select nationality, price from artsist\_data join art\_work on artsist\_data.artist\_id = art\_work.artist\_id where nationality ="united states";

select \* from art\_work where price > (select max(price) from artsist\_data join art\_work on artsist\_data.artist\_id = art\_work.artist\_id where nationality ="united states");

**output:**

**A screenshot of a computer

Description automatically generated**

**4) show the data of the artwork who’s price the greater than average price**

select avg(price) from art\_work;

select \* from art\_work where price > (select avg(price) from art\_work);

**output:**

A screenshot of a computer

Description automatically generated

**5) having the name and email\_id of the customer who spend more than 300000 on the art using joins and subqueries**

SELECT c.name, c.email\_id FROM customer c

WHERE c.customer\_id IN (SELECT s.customer\_id

FROM sell s

JOIN art\_work a ON s.artwork\_id = a.artwork\_id

WHERE a.price > 300000);

**Output:**

A screenshot of a computer

Description automatically generated

**BASIC QUESTIONS:**

**1) Find the average price of artworks in a specific size**

select avg(price) from art\_work where size = "small";

**output:**

A white box with black text

Description automatically generated

**2) Find the total value of artworks sold by the artist "Verena Cumbridge"**

select sum(price) from art\_work inner join artsist\_data on art\_work.artist\_id = artsist\_data.artist\_id where artsist\_data.name = "Verena Cumbridge";

**output:**

A white rectangular object with a black border

Description automatically generated

**3) dropping the column from the table**

alter table exhibition\_details drop column artist\_id;

**output:**

**A screenshot of a computer

Description automatically generated**

**4) artwork having the top 5 price**

select \* from art\_work limit 5;

A screenshot of a computer

Description automatically generated**output:**

**5) count of artist by sizes available**

select size, count(artist\_id) from art\_work group by size;

**output:**

A screenshot of a computer

Description automatically generated

**6) selecting particular value**

select \* from sell where artwork\_id = 2157;

select \* from sell;

**output:**

A screenshot of a computer

Description automatically generated

**7) title of a project having size small**

select size, title from art\_work where size = "small";

**output:**

A screenshot of a computer

Description automatically generated

**8) extract month name from the table**

select monthname(sale\_date) from sell;

**output:**

A screenshot of a computer

Description automatically generated

**9) change mobile number of the customer**

update customer set phone = 7447547494 where customer\_id = 3506;

select \* from customer;

**output:**

A screenshot of a computer

Description automatically generated

**10) use of group by clause using joins getting sum of the price in the medium**

select artwork\_medium ,sum(price), count(price) from art\_work inner join sell on art\_work.artwork\_id = sell.artwork\_id group by artwork\_medium ;

**output:**

A screenshot of a computer

Description automatically generated

**11) use of logical operator AND**

select \* from art\_work where price > 100000 AND size = "large";

**output:**

A screenshot of a computer

Description automatically generated

**12) convert the email ID of the particular customer to the uper case**

select upper(name), customer\_id from customer where customer\_id = 3506;

select \* from customer;

A screenshot of a computer

Description automatically generated**output:**

**13) what is the day gap between start date and the date sold**

select datediff(submission\_date,start\_date ) from exhibition\_details;

**output:**

A screenshot of a computer

Description automatically generated

**14) title of the art work having id 1,2,3,4,5**

select title, price from art\_work where artist\_id in (1,2,3,4,5);

**output:**

A screenshot of a computer

Description automatically generated

**15) change the column name and add the city is new york**

alter table sell add city varchar(30);

select \* from sell;

update sell set city = "new york";

**output:**

A screenshot of a computer

Description automatically generated

**CONCLUSION:**

the Art Gallery Management Database project has successfully designed and implemented a comprehensive database system that effectively manages and stores information related to the art gallery's operations. The system has achieved its objectives by providing a user-friendly interface, improving data accuracy and accessibility, enhancing customer engagement, and supporting business growth through data-driven insights and reporting. With its robust and scalable design, the system is well-positioned to support the art gallery's future growth and expansion. The project demonstrates the importance of effective database management in the art industry, and its successful implementation will serve as a valuable resource for the gallery's continued success.