

DBMS LAB3

NAME: NAGAVENI L G

SRN: PES2UG21CS315

SEC: 5F

```
mysql> show databases;
+-----+
| Database |
+-----+
| art_gallery |
| cricketleaguemanagementsystem |
| information_schema |
| lab1 |
| mysql |
| performance_schema |
| pes2ug21cs315_cricket_league_management_db |
| sys |
+-----+
8 rows in set (0.01 sec)

mysql> use art_gallery;
Database changed
mysql> show tables;
+-----+
| Tables_in_art_gallery |
+-----+
| art |
| art_backup |
| art_order |
| artist |
| comments |
| customer |
| exhibited |
| exhibition |
| gallery |
| payment |
| purchase |
| specialization |
+-----+
12 rows in set (0.01 sec)
```

TASK 1:

There is a small case study on the Café database that has been given. You must understand the case study, the set of records in the tables, then you are supposed to answer, whether the task1 commands given are executable or not. Also justify the same with appropriate reasons. (NO NEED FOR ANY EXECUTION)

- 1) In the given SQL query, there seems to be an issue in the 'GROUP BY' clause. It groups the results by the price column, which is not the appropriate operation to calculate the average price per cafe. Instead, it should be grouped by the Café_name column.
- 2) The given SQL query is correct. The result of the query is the list of names (V1.Name) of individuals who have visited both 'Café A' and 'Café B'.
- 3) Since an aggregate function (SUM) has been used in the SELECT clause, we need to add a 'GROUP BY' clause at the end of the "Name" attribute.
- 4) The given SQL query is correct. The result of this query will be a list of names (from the "Visits" table) of individuals who have visited two or more different cafes, as indicated by the condition V.cafecount >= 2
- 5) The given SQL query is correct. This query will retrieve the distinct names of individuals who have visited a cafe by joining the "Person" and "Visits" tables using the "Name" column as the join condition.

TASK2

Convert 6 th and 7 th question relational algebra into SQL query execute it and attach a screen .

6)

>SELECT p.order_id as Order_ID,p.art_id as Art_ID,ao.amount as Amount,ao.order_description as Order_Desc,py.p_time as Payment_time,py.p_status as Payment_Status

-> From purchase p right outer join art_order ao on ao.order_id=p.order_id left outer join payment py on py.order_id=ao.order_id;

```
mysql> prompt PES2UG21CS315>
PROMPT set to 'PES2UG21CS315>'
PES2UG21CS315>SELECT p.order_id as Order_ID,p.art_id as Art_ID,ao.amount as Amount,ao.order_description as Order_Desc,py.p_time as Payment_time,py.p_status
as Payment_Status
-> From purchase p right outer join art_order ao on ao.order_id=p.order_id left outer join payment py on py.order_id=ao.order_id;
```

Order_ID	Art_ID	Amount	Order_Desc	Payment_time	Payment_Status
ORD001	ART001	500	Order for artwork	2023-09-10 10:00:00	Success
ORD002	ART003	750	Order for sculpture	2023-09-12 11:00:00	Success
ORD003	ART005	1000	Order for painting	2023-09-15 12:00:00	Pending
ORD004	ART007	1250	Order for artwork	2023-09-18 13:00:00	Failed
ORD005	ART009	1500	Order for sculpture	2023-09-20 14:00:00	Success
ORD006	ART011	1750	Order for painting	2023-09-23 15:00:00	Pending
ORD007	ART013	2000	Order for artwork	2023-09-26 16:00:00	Success
ORD008	ART002	2250	Order for sculpture	2023-09-28 17:00:00	Failed
NULL	NULL	2500	Order for painting	2023-10-01 18:00:00	Success
ORD010	ART006	2750	Order for artwork	2023-10-03 19:00:00	Pending

10 rows in set (0.00 sec)

7)

```
PES2UG21CS315>SELECT p.cust_id AS Customer_ID, SUM(pa.amount) AS Customer_Income FROM purchase p JOIN payment pa ON p.order_id = pa.order_id GROUP BY p.cust_id;
```

Customer_ID	Customer_Income
C002	6500
C003	750
C004	1000
C006	1500
C007	1750
C009	2250

6 rows in set (0.01 sec)

TASK3

8) Tally the customer count in each location. This data assists in understanding the distribution of customers across various geographical areas, allowing the art gallery to tailor its services and promotions to cater effectively to different regions and demographics.

```
PES2UG21CS315>SELECT location,Count(*) as customer_count  
-> from customer  
-> group by location;
```

location	customer_count
Bangaluru	1
Mysuru	1
Hubli	1
Belgaum	1
Mangaluru	1
Gulbarga	1
Bidar	1
Udupi	1
Davangere	1

```
9 rows in set (0.02 sec)
```

9) In the world of art galleries, understanding the financial performance of each gallery is paramount. To achieve this, we embark on a quest to unveil the total revenue generated by each gallery, shedding light on their respective contributions to the gallery's economy .list should be sorted based on decreasing order of revenue .

```

PES2UG21CS315>SELECT
->     g.g_name AS gallery_name,
->     SUM(ao.amount) AS total_revenue
-> FROM
->     exhibition AS e
-> INNER JOIN
->     gallery AS g ON e.g_id = g.g_id
-> LEFT JOIN
->     exhibited AS ex ON e.ex_id = ex.ex_id
-> LEFT JOIN
->     purchase AS p ON ex.art_id = p.art_id
-> LEFT JOIN
->     art_order AS ao ON p.order_id = ao.order_id
-> GROUP BY
->     g.g_name
-> ORDER BY
->     total_revenue DESC;

```

gallery_name	total_revenue
Bengaluru Chitra Kala Parishat	4500
Mysuru Art Haven	3750
Gulbarga Sculpture Park	2750
Belgaum Art Gallery	1750
Mangaluru Art Studio	1000
Hubli Art Center	NULL
Udupi Digital Art Showcase	NULL
Bidar Illustration Gallery	NULL
Davangere Art Studio	NULL

9 rows in set (0.01 sec)

10) Retrieve galleries that exhibit over 1 art works. This information aids in identifying galleries with a substantial collection, making it easier for visitors to explore diverse art pieces and enhancing the gallery's reputation as a vibrant artistic hub.

```

PES2UG21CS315>
PES2UG21CS315>
PES2UG21CS315>SELECT
->      g.g_name AS gallery_name,
->      COUNT(ex.art_id) AS artworks_exhibited
-> FROM
->      gallery AS g
-> LEFT JOIN
->      exhibited AS ex ON g.g_id = ex.ex_id
-> GROUP BY
->      g.g_name
-> HAVING
->      COUNT(ex.art_id) > 1;
Empty set (0.00 sec)

```

11) In the bustling world of art galleries, it's essential to recognize exhibitions that truly stand out those that feature the most artworks, showcasing the pinnacle of artistic expression. Imagine you're the curator of a renowned gallery, and you want to spotlight the exhibition that has exhibited the highest number of artworks. By doing so, you can celebrate the artists, generate buzz around your gallery, and provide visitors with a remarkable experience.

```

PES2UG21CS315>
PES2UG21CS315>
PES2UG21CS315>SELECT
->      ex.ex_id AS exhibition_id,
->      e.g_id AS gallery_id,
->      e.s_date AS start_date,
->      e.e_date AS end_date,
->      COUNT(ex.art_id) AS total_artworks_exhibited
-> FROM
->      exhibited AS ex
-> JOIN
->      exhibition AS e ON ex.ex_id = e.ex_id
-> GROUP BY
->      ex.ex_id, e.g_id, e.s_date, e.e_date
-> ORDER BY
->      total_artworks_exhibited DESC
-> LIMIT 1;
+-----+-----+-----+-----+-----+
| exhibition_id | gallery_id | start_date | end_date   | total_artworks_exhibited |
+-----+-----+-----+-----+-----+
| EX001        | G002      | 2023-09-01 | 2023-09-30 | 4 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

PES2UG21CS315>

```