TRAINITY PROJECT II- INSTAGRAM USER ANALYTICS

BY- Shaurya Gairola

❒ **Project Description** : As per the initial project report , I have been assigned the job to gather and provide insights to the product team of Instagram based on the questions they have asked. I have to work on the data from the provided database and collect useful insights for Instagram to launch appropriate marketing campaigns as well as help the investors who want to know the performance of Instagram.

❒ **Approach** :The project was executes using MYSQL where queries were used to create a raw database from the provided data . According to the instructions provided to create the database and the corresponding tables , I feeded the data into MySQL and executed the appropriate queries to get the required insights.

❒ **Tech Stack Used** : 1. MySQL Workbench 8.0 CE

2. Microsoft Word (Office 365)

❒ **Insights**: I had basic knowledge of SQL before and after working on this project I have gained a lot of valuable knowledge of SQL. This Instagram user analytics project helped me to dig deeper into the world of SQL and helped me understand how complex queries work and how to cultivate business insights from given data . It enabled me to ask the right necessary questions and narrow down solutions to the given problems.

❒ **Results:** Here are the query statements which I executed and the corresponding results.

A) Marketing: The marketing team wants to launch some campaigns, and they need your help with the following

1. **Rewarding Most Loyal Users**: People who have been using the platform for the longest time. Your Task: Find the 5 oldest users of the Instagram from the database provided.

QUERY-

```
#1-loyal user reward

select * from users

order by created_at

limit 5;
```

SELECT – it tells your database that you want to select data.

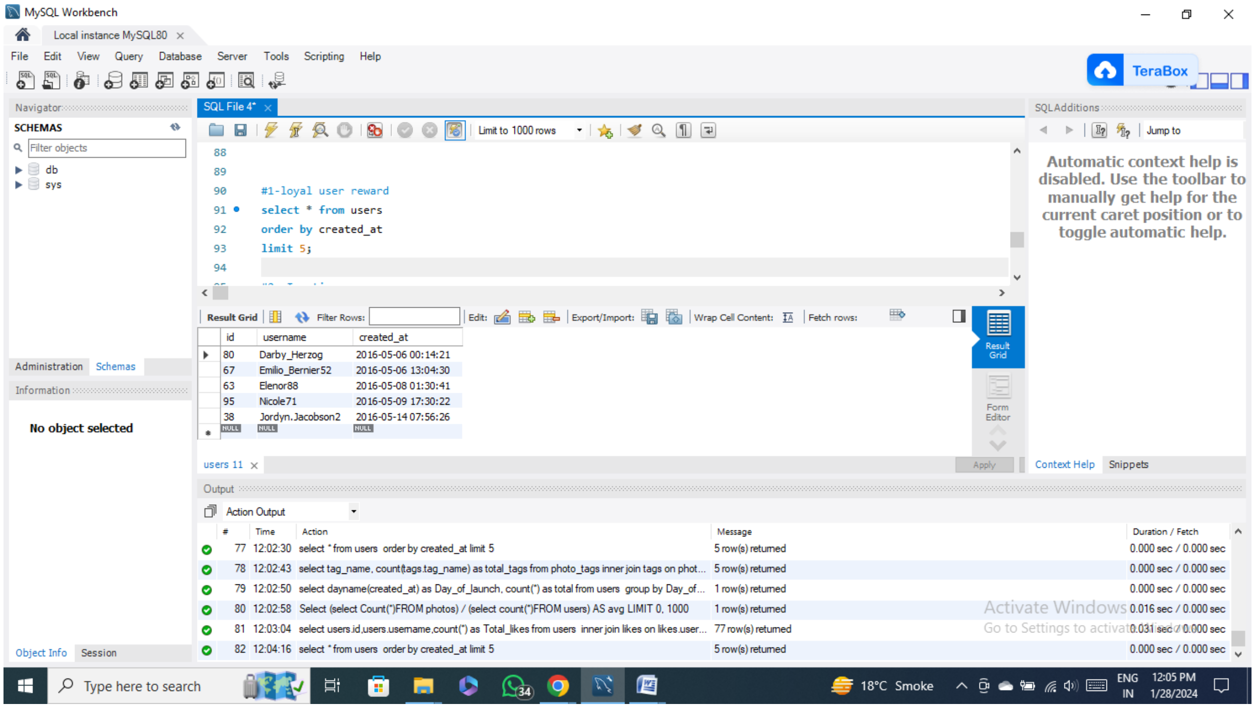
FROM- users tells the database to select data from the user table.

(*) tells the database that you want to see all columns in this table.

ORDER BY- It is used to sort the data in ascending or descending order.

LIMIT n - returns the first n rows from the result. This is much more efficient than returning all the data from the database

RESULT-



```
#1-loyal user reward
select * from users
order by created_at
limit 5;
```

id	username	created_at
80	Darby_Herzog	2016-05-06 00:14:21
67	Emilio_Bernier52	2016-05-06 13:04:30
63	Elenor88	2016-05-08 01:30:41
95	Nicole71	2016-05-09 17:30:22
38	Jordyn.Jacobson2	2016-05-14 07:56:26
NULL	NULL	NULL

```
#   Time      Action                                                                                  Message             Duration / Fetch
77  12:02:30  select * from users  order by created_at limit 5                                        5 row(s) returned   0.000 sec / 0.000 sec
78  12:02:43  select tag_name, count(tags.tag_name) as total_tags from photo_tags inner join tags on phot...   5 row(s) returned   0.000 sec / 0.000 sec
79  12:02:50  select dayname(created_at) as Day_of_launch, count(*) as total from users  group by Day_of...   1 row(s) returned   0.000 sec / 0.000 sec
80  12:02:58  Select (select Count(*)FROM photos) / (select count(*)FROM users) AS avg LIMIT 0, 1000      1 row(s) returned   0.016 sec / 0.000 sec
81  12:03:04  select users.id,users.username,count(*) as Total_likes from users  inner join likes on likes.user...   77 row(s) returned   0.031 sec / 0.000 sec
82  12:04:16  select * from users  order by created_at limit 5                                        5 row(s) returned   0.000 sec / 0.000 sec
```

2. Remind Inactive Users to Start Posting: By sending them promotional emails to post their 1st photo.

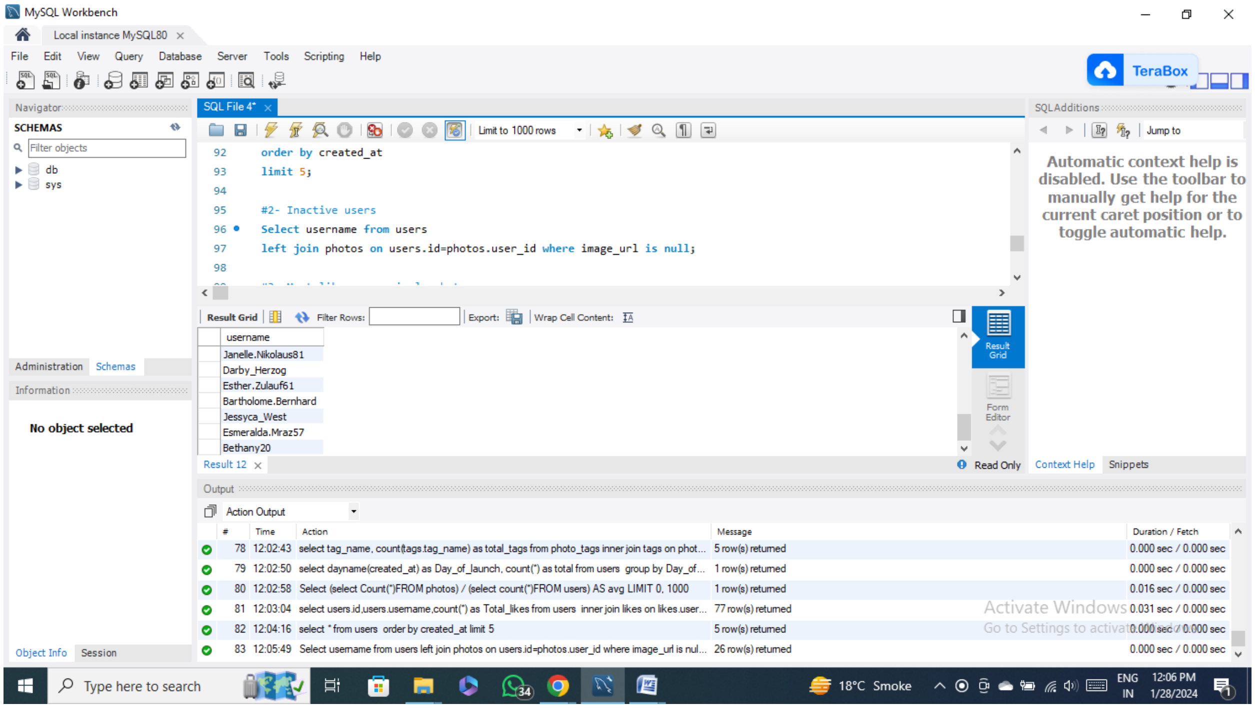
Task: Find the users who have never posted a single photo on Instagram

QUERY-

```
#2- Inactive users
Select username from users
left join photos on users.id=photos.user_id
where image_url is null;
```

LEFT JOIN- It returns all rows from the left table (the first table in the query) plus all matching rows from the right table (the second table in the query).

RESULT-



```
order by created_at
limit 5;

#2- Inactive users
Select username from users
left join photos on users.id=photos.user_id where image_url is null;
```

username
Janelle.Nikolaus81
Darby_Herzog
Esther.Zulauf61
Bartholome.Bernhard
Jessyca_West
Esmeralda.Mraz57
Bethany20

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
 #   Time      Action                                                                                   Message            Duration / Fetch
 78  12:02:43  select tag_name, count(tags.tag_name) as total_tags from photo_tags inner join tags on phot...  5 row(s) returned  0.000 sec / 0.000 sec
 79  12:02:50  select dayname(created_at) as Day_of_launch, count(*) as total from users  group by Day_of...  1 row(s) returned  0.000 sec / 0.000 sec
 80  12:02:58  Select (select Count(*)FROM photos) / (select count(*)FROM users) AS avg LIMIT 0, 1000        1 row(s) returned  0.016 sec / 0.000 sec
 81  12:03:04  select users.id,users.username,count(*) as Total_likes from users  inner join likes on likes.user...  77 row(s) returned  0.031 sec / 0.000 sec
 82  12:04:16  select * from users  order by created_at limit 5                                          5 row(s) returned  0.000 sec / 0.000 sec
 83  12:05:49  Select username from users left join photos on users.id=photos.user_id where image_url is nul...  26 row(s) returned  0.000 sec / 0.000 sec
```

3. Declaring Contest Winner:

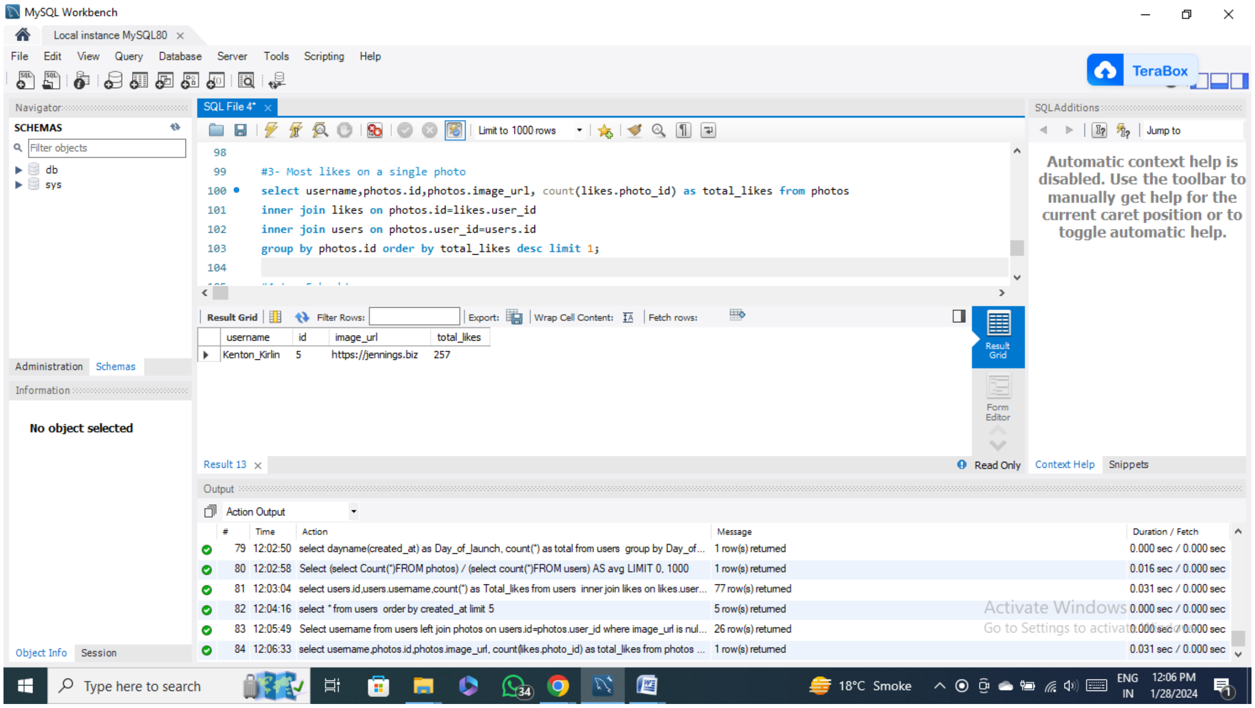
Task: Identify the winner of the contest and provide their details to the team.To do this task we need to find the most popular photo with most likes and user who created it.

QUERY-

```
#3- Most likes on a single photo


select username,photos.id,photos.image_url, count(likes.photo_id) as total_likes from photos

inner join likes on photos.id=likes.user_id

inner join users on photos.user_id=users.id

group by photos.id

order by total_likes desc

limit 1;
```

INNER JOIN - only shows those rows from the two tables where there is a match between the columns. In other words, you can only see those pieces of equipment which have a room assigned and vice versa

RESULT-



```sql
#3- Most likes on a single photo
select username,photos.id,photos.image_url, count(likes.photo_id) as total_likes from photos
inner join likes on photos.id=likes.user_id
inner join users on photos.user_id=users.id
group by photos.id order by total_likes desc limit 1;
```

username	id	image_url	total_likes
Kenton_Kirlin	5	https://jennings.biz	257

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
#   Time      Action                                                                           Message           Duration / Fetch
79  12:02:50  select dayname(created_at) as Day_of_launch, count(*) as total from users  group by Day_of...   1 row(s) returned    0.000 sec / 0.000 sec
80  12:02:58  Select (select Count(*)FROM photos) / (select count(*)FROM users) AS avg LIMIT 0, 1000       1 row(s) returned    0.016 sec / 0.000 sec
81  12:03:04  select users.id,users.username,count(*) as Total_likes from users  inner join likes on likes.user...   77 row(s) returned   0.031 sec / 0.000 sec
82  12:04:16  select * from users  order by created_at limit 5                                       5 row(s) returned    0.000 sec / 0.000 sec
83  12:05:49  Select username from users left join photos on users.id=photos.user_id where image_url is nul...   26 row(s) returned   0.000 sec / 0.000 sec
84  12:06:33  select username,photos.id,photos.image_url, count(likes.photo_id) as total_likes from photos ...   1 row(s) returned    0.031 sec / 0.000 sec
```

4. HASHTAG RESEARCHING: Hashtag helps the user to reach to wide range of people. It is used to draw attention ,organise, promote and connect.

Task- to identify the top 5 most commonly used hashtags on instagram.

QUERY-

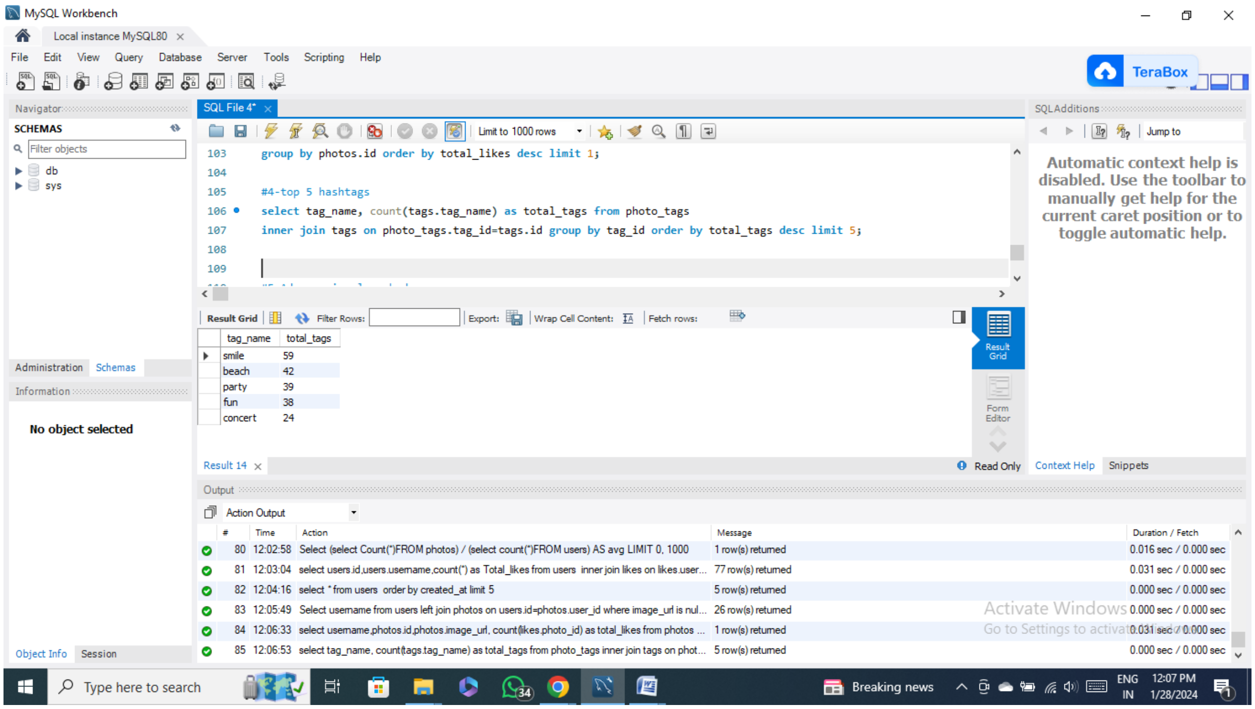
#4-top 5 hashtags

```
select tag_name, count(tags.tag_name) as total_tags from photo_tags
inner join tags on photo_tags.tag_id=tags.id
group by tag_id
order by total_tags desc
limit 5;
```

COUNT()- function returns the number of rows that matches the specific criteria

AS- The new name is a temporary name and doesn't change the actual column name in the database. It only influences the way the column is shown in the result of the specific query. This technique is often used when there are a few columns with the same name coming from different tables. . We can repeat this process with every column.

RESULT-



```
group by photos.id order by total_likes desc limit 1;

#4-top 5 hashtags
select tag_name, count(tags.tag_name) as total_tags from photo_tags
inner join tags on photo_tags.tag_id=tags.id group by tag_id order by total_tags desc limit 5;
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

tag_name	total_tags
smile	59
beach	42
party	39
fun	38
concert	24

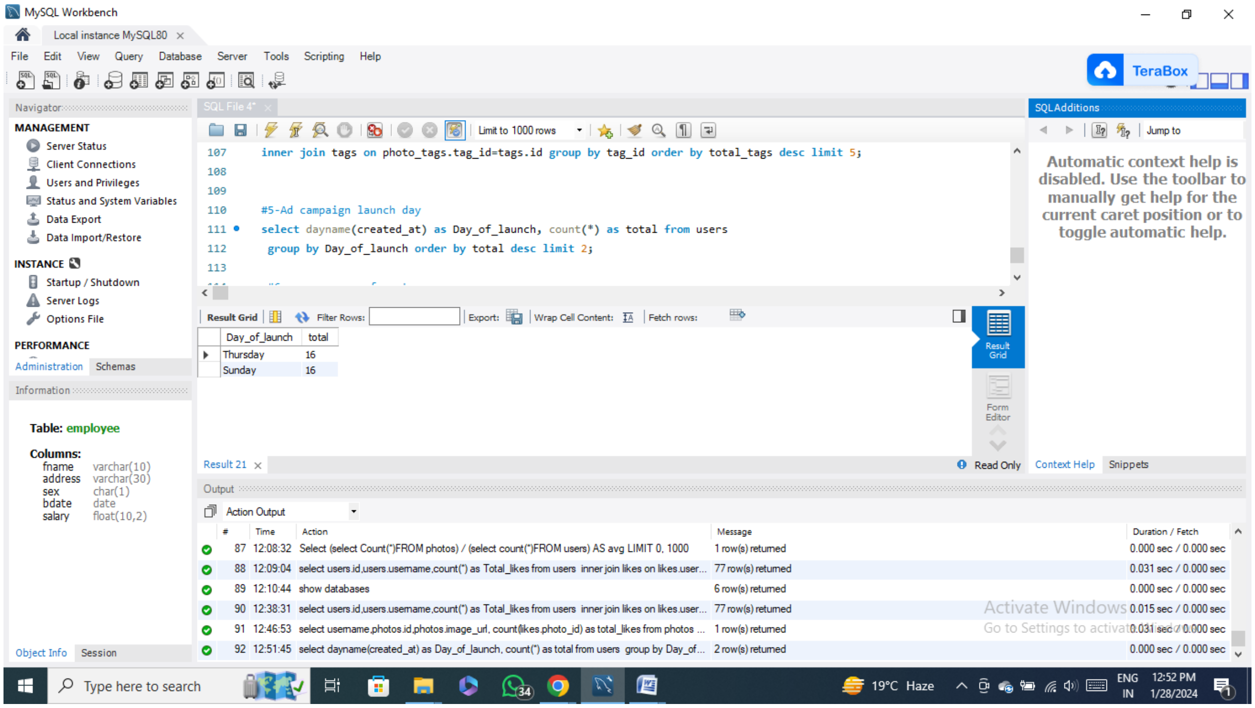
```
 #  Time      Action                                                                              Message             Duration / Fetch
 80 12:02:58  Select (select Count(*)FROM photos) / (select count(*)FROM users) AS avg LIMIT 0, 1000  1 row(s) returned   0.016 sec / 0.000 sec
 81 12:03:04  select users.id,users.username,count(*) as Total_likes from users  inner join likes on likes.user...  77 row(s) returned  0.031 sec / 0.000 sec
 82 12:04:16  select * from users  order by created_at limit 5                                     5 row(s) returned   0.000 sec / 0.000 sec
 83 12:05:49  Select username from users left join photos on users.id=photos.user_id where image_url is nul...  26 row(s) returned  0.000 sec / 0.000 sec
 84 12:06:33  select username,photos.id,photos.image_url, count(likes.photo_id) as total_likes from photos ...  1 row(s) returned  0.031 sec / 0.000 sec
 85 12:06:53  select tag_name, count(tags.tag_name) as total_tags from photo_tags inner join tags on phot...  5 row(s) returned  0.000 sec / 0.000 sec
```

5. LAUNCH AD CAMPAIGNTASK – To find out the day of week when most users register’s on Instagram.

QUERY-

```
#5-Ad campaign launch day

select dayname(created_at) as Day_of_launch, count(*) as total from users

group by Day_of_launch

order by total desc

limit 2;
```

RESULT- I found out that there are 2 days of the week in which the new users register the most on instagram.



```
inner join tags on photo_tags.tag_id=tags.id group by tag_id order by total_tags desc limit 5;


#5-Ad campaign launch day
select dayname(created_at) as Day_of_launch, count(*) as total from users
 group by Day_of_launch order by total desc limit 2;
```

Day_of_launch	total
Thursday	16
Sunday	16

Table: employee

Columns:

fname	varchar(10)
address	varchar(30)
sex	char(1)
bdate	date
salary	float(10,2)

```
#   Time      Action                                                                         Message            Duration / Fetch
87  12:08:32  Select (select Count(*)FROM photos) / (select count(*)FROM users) AS avg LIMIT 0, 1000   1 row(s) returned   0.000 sec / 0.000 sec
88  12:09:04  select users.id,users.username,count(*) as Total_likes from users  inner join likes on likes.user...   77 row(s) returned   0.031 sec / 0.000 sec
89  12:10:44  show databases                                                                 6 row(s) returned   0.000 sec / 0.000 sec
90  12:38:31  select users.id,users.username,count(*) as Total_likes from users  inner join likes on likes.user...   77 row(s) returned   0.015 sec / 0.000 sec
91  12:46:53  select username,photos.id,photos.image_url, count(likes.photo_id) as total_likes from photos ...   1 row(s) returned   0.031 sec / 0.000 sec
92  12:51:45  select dayname(created_at) as Day_of_launch, count(*) as total from users  group by Day_of...   2 row(s) returned   0.000 sec / 0.000 sec
```

B. INVESTER MATRICS

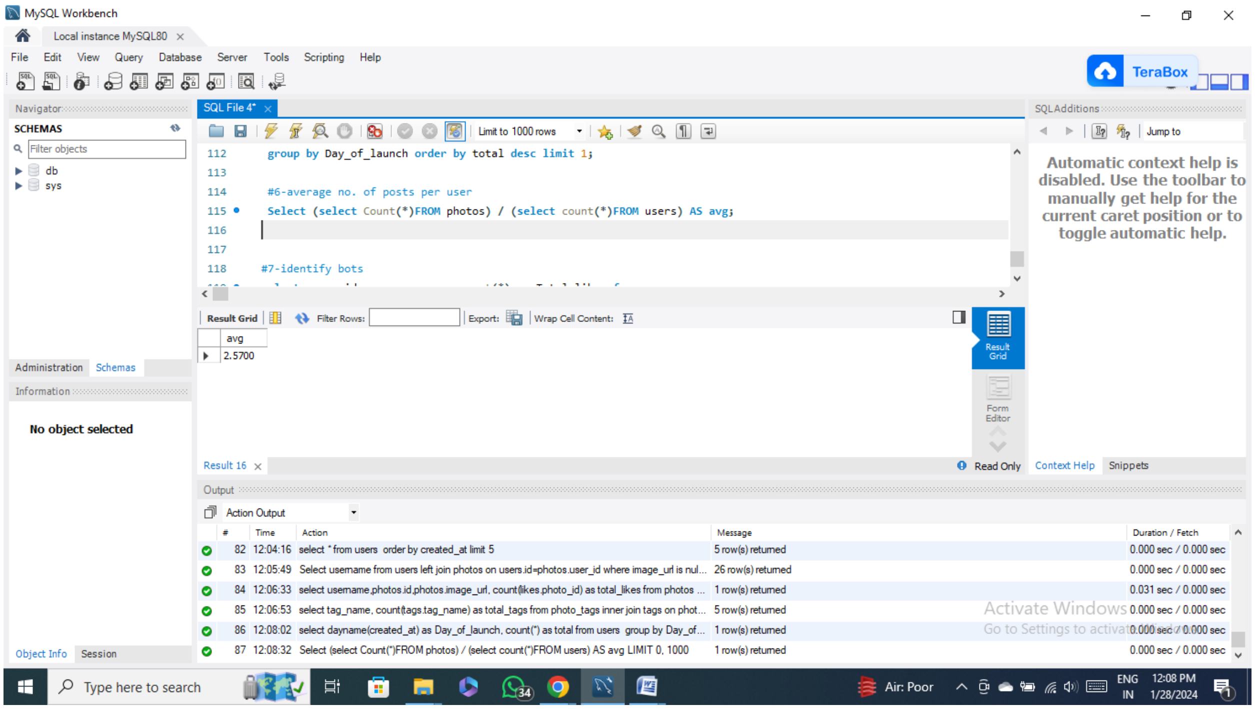
6. USER ENGAGEMENT- Investers want to know that instagram is not becoming redundantlike facebook, so they want to check the frequency of how much the users are engaging onthe platform.

Task- To provide how many times an average user post on instagram.

QUERY-

```
#6-average no. of posts per user

Select (select Count(*)FROM photos) / (select count(*)FROM users) AS avg;
```

RESULT-



```
group by Day_of_launch order by total desc limit 1;

#6-average no. of posts per user
Select (select Count(*)FROM photos) / (select count(*)FROM users) AS avg;


#7-identify bots
```

avg
2.5700

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

#	Time	Action	Message	Duration / Fetch
82	12:04:16	select * from users order by created_at limit 5	5 row(s) returned	0.000 sec / 0.000 sec
83	12:05:49	Select username from users left join photos on users.id=photos.user_id where image_url is nul...	26 row(s) returned	0.000 sec / 0.000 sec
84	12:06:33	select username,photos.id,photos.image_url, count(likes.photo_id) as total_likes from photos ...	1 row(s) returned	0.031 sec / 0.000 sec
85	12:06:53	select tag_name, count(tags.tag_name) as total_tags from photo_tags inner join tags on phot...	5 row(s) returned	0.000 sec / 0.000 sec
86	12:08:02	select dayname(created_at) as Day_of_launch, count(*) as total from users group by Day_of...	1 row(s) returned	0.000 sec / 0.000 sec
87	12:08:32	Select (select Count(*)FROM photos) / (select count(*)FROM users) AS avg LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

Activate Windows
Go to Settings to activate Windows.

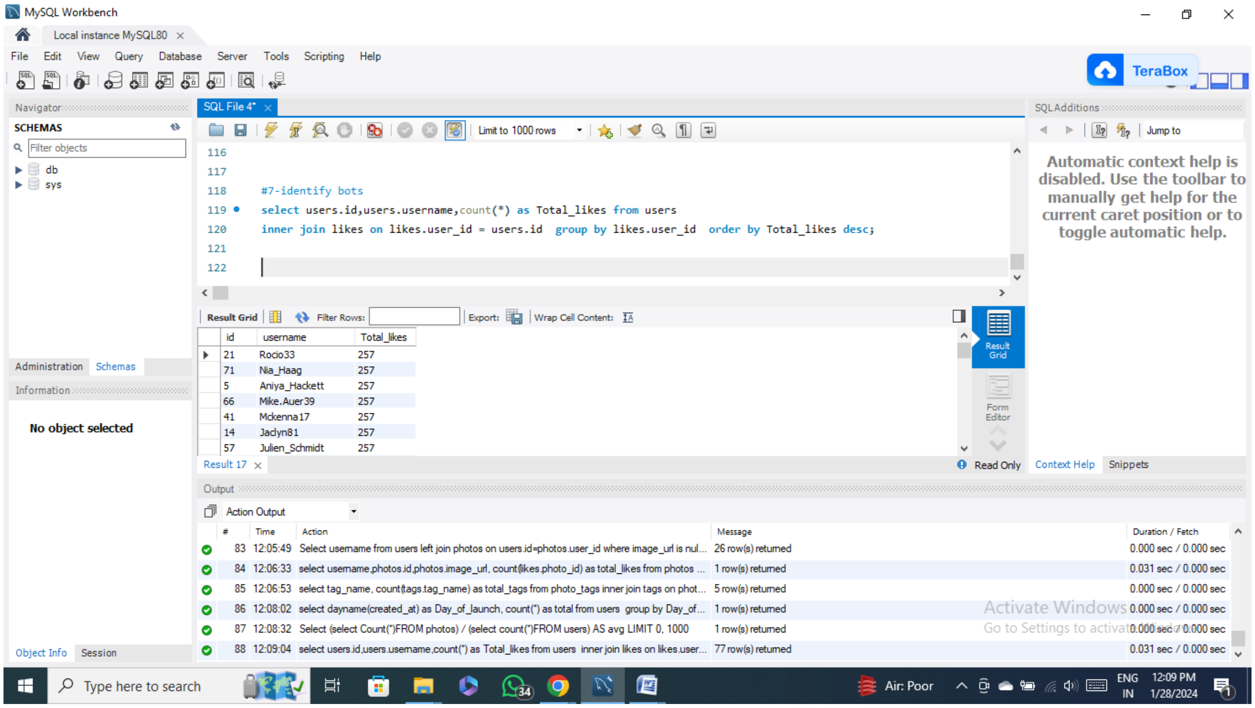
7. BOTS& FAKE ACCOUNTS- It is reported that there are lot of bots and fake accounts on the platform. The investors wants to know if there are fake and dummy accounts.

Task – To Provide data on users(bots) who have liked every single photo on the site.

QUERY-

```
#7-identify bots

select users.id,users.username,count(*) as Total_likes from users

inner join likes on likes.user_id = users.id

group by likes.user_id

order by Total_likes desc;
```

RESULT- From the query, I found that there are 13 bots.



```
#7-identify bots
select users.id,users.username,count(*) as Total_likes from users
inner join likes on likes.user_id = users.id  group by likes.user_id  order by Total_likes desc;
```

id	username	Total_likes
21	Rocio33	257
71	Nia_Haag	257
5	Aniya_Hackett	257
66	Mike.Auer39	257
41	Mckenna17	257
14	Jaclyn81	257
57	Julien_Schmidt	257

```
#   Time      Action                                                                                  Message              Duration / Fetch
83  12:05:49  Select username from users left join photos on users.id=photos.user_id where image_url is nul...  26 row(s) returned   0.000 sec / 0.000 sec
84  12:06:33  select username,photos.id,photos.image_url, count(likes.photo_id) as total_likes from photos ...  1 row(s) returned    0.031 sec / 0.000 sec
85  12:06:53  select tag_name, count(tags.tag_name) as total_tags from photo_tags inner join tags on phot...   5 row(s) returned    0.000 sec / 0.000 sec
86  12:08:02  select dayname(created_at) as Day_of_launch, count(*) as total from users  group by Day_of...     1 row(s) returned    0.000 sec / 0.000 sec
87  12:08:32  Select (select Count(*)FROM photos) / (select count(*)FROM users) AS avg LIMIT 0, 1000          1 row(s) returned    0.000 sec / 0.000 sec
88  12:09:04  select users.id,users.username,count(*) as Total_likes from users  inner join likes on likes.user...  77 row(s) returned   0.031 sec / 0.000 sec
```

OVERALL RESULT – While doing this project I found out so many important terms of MYSQL that helped me in solving complex problems irrespective of how large the database is. I got to learn about using mysql and whereabouts of mysql workbench and i believe it will help me in future a lot. I have provided the solutions to every questions asked and i believe they are correct to the best of my knowledge and it solves all the query. Overall , this Instagram user analytics project has helped me understand SQL and its working right from the basic to advanced concepts involved in it .

THANK YOU

BY-

SHAURYA GAIROLA