INSTAGRAM USER ANALYTICS

PROJECT DESCRIPTION :- In this project we extract insight from given raw data. This operation we perform to increase the efficiency of the Instagram. And to help all the department (**marketing, product manager)** so that they can get the correct insight and take a better decision.

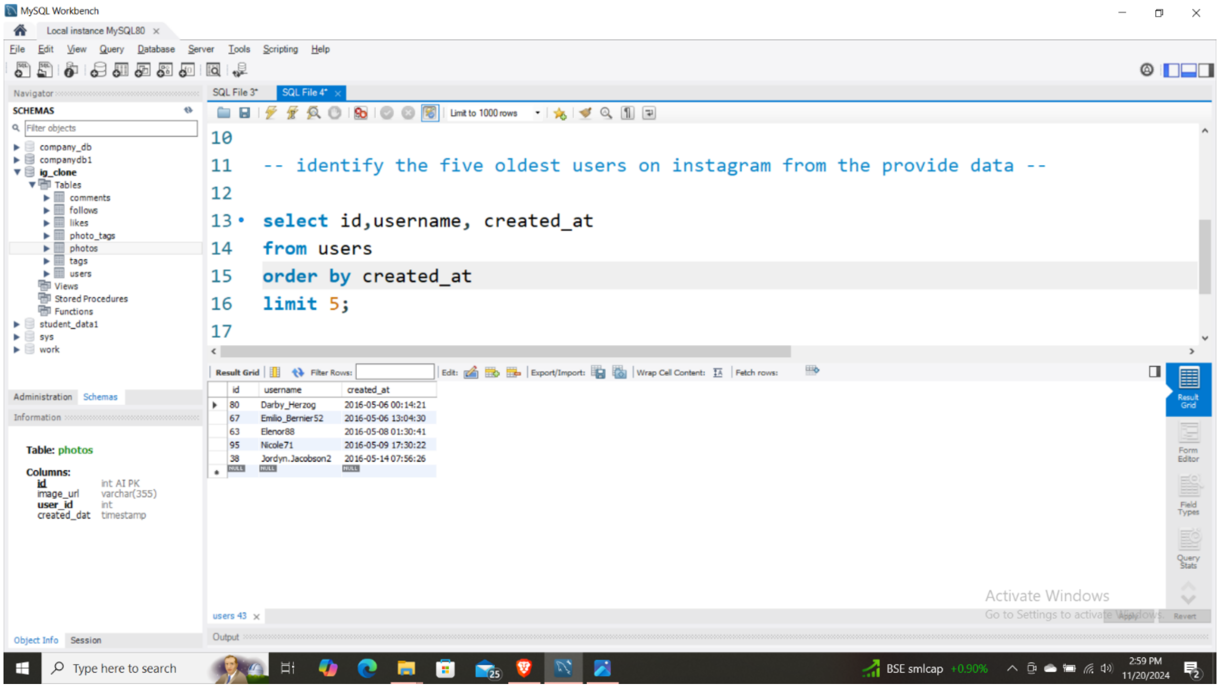
Project approach:- I made this project on using mysql by using the given data providing by you. Where first I create the database by raw data and execute all the given code. then I perform all the Query to extract the data from database and create useful insight.

Tech stack used :- In this project I use mysql workbench 8.0 CE . this is good tool to perform all type of task, ease to use as GUI and very eassy to troubleshoot.

Insight SQL Tasks :

Marketing Analysis:-

1. **Identify the five oldest user on instagram from provided database.**



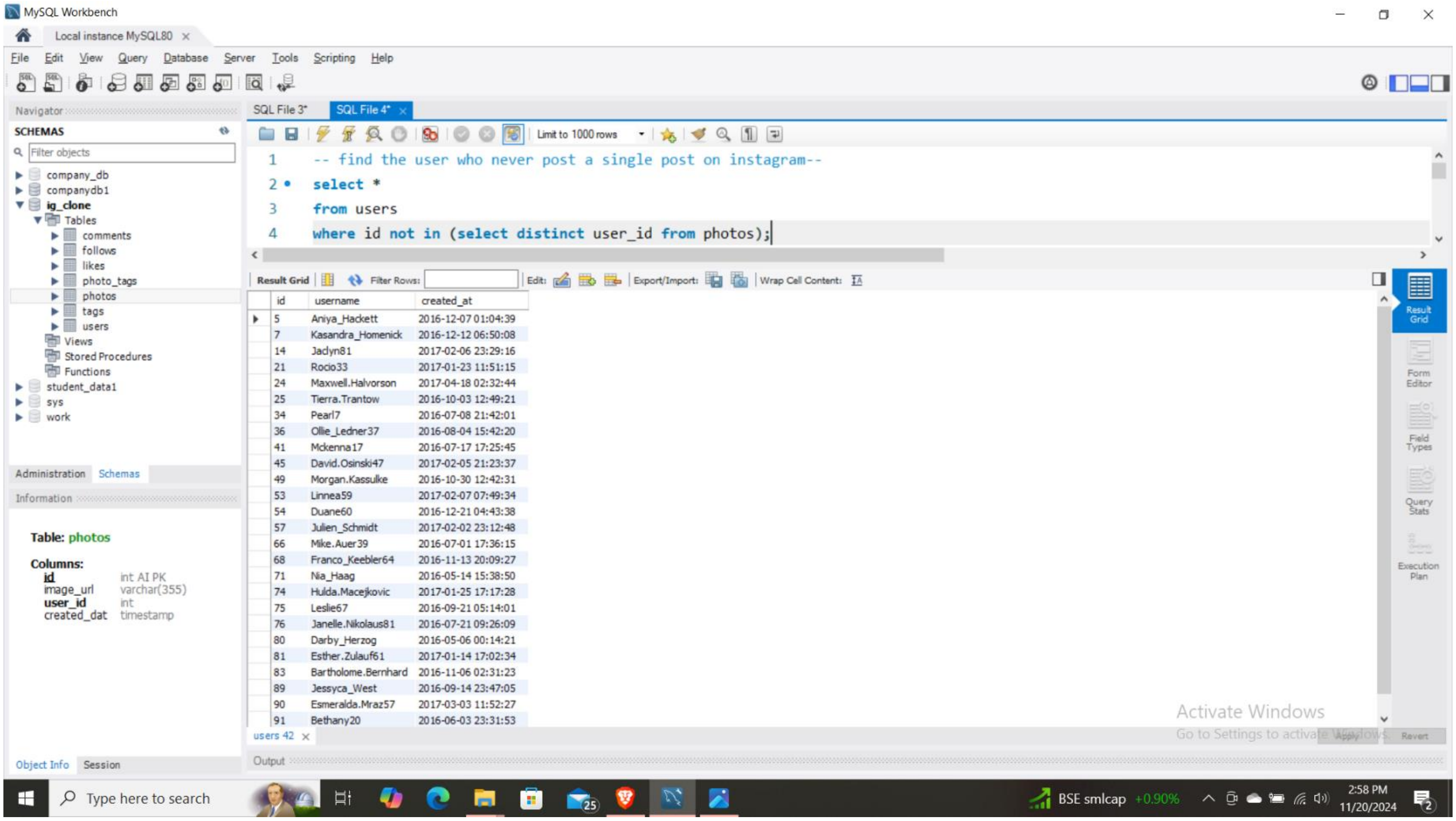
```
-- identify the five oldest users on instagram from the provide data --

select id,username, created_at
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

Result:- In this we have to find five oldest user on Instagram so for that first we extract the users table And find out how we can extract the five oldest so for that we extract the user details first after that We use order by clause . **note :** we didn't use desc with order by because we want the oldest user And for date function we ascending find us the oldest users then we use **limit 5** coz we want to take only five users.

2. Identify the users who never post the single post on Instagram.



```
-- find the user who never post a single post on instagram--
select *
from users
where id not in (select distinct user_id from photos);
```

id	username	created_at
5	Aniya_Hackett	2016-12-07 01:04:39
7	Kasandra_Homenick	2016-12-12 06:50:08
14	Jaclyn81	2017-02-06 23:29:16
21	Rocio33	2017-01-23 11:51:15
24	Maxwell.Halvorson	2017-04-18 02:32:44
25	Tierra.Trantow	2016-10-03 12:49:21
34	Pearl7	2016-07-08 21:42:01
36	Ollie_Ledner37	2016-08-04 15:42:20
41	Mckenna17	2016-07-17 17:25:45
45	David.Osinski47	2017-02-05 21:23:37
49	Morgan.Kassulke	2016-10-30 12:42:31
53	Linnea59	2017-02-07 07:49:34
54	Duane60	2016-12-21 04:43:38
57	Julien_Schmidt	2017-02-02 23:12:48
66	Mike.Auer39	2016-07-01 17:36:15
68	Franco_Keebler64	2016-11-13 20:09:27
71	Nia_Haag	2016-05-14 15:38:50
74	Hulda.Macejkovic	2017-01-25 17:17:28
75	Leslie67	2016-09-21 05:14:01
76	Janelle.Nikolaus81	2016-07-21 09:26:09
80	Darby_Herzog	2016-05-06 00:14:21
81	Esther.Zulauf61	2017-01-14 17:02:34
83	Bartholome.Bernhard	2016-11-06 02:31:23
89	Jessyca_West	2016-09-14 23:47:05
90	Esmeralda.Mraz57	2017-03-03 11:52:27
91	Bethany20	2016-06-03 23:31:53

Table: photos

Columns:

id	int AI PK
image_url	varchar(355)
user_id	int
created_dat	timestamp

Result: In this we have to find the user who never post a single post on instagram for that we have to take the value from two table coz user table have the **users details** and **photos** table have the data about the users who post on instagram and both table are have same **attribute** called **user_id** in **photos** and id in users so we just use **where condition** with **not in.** if users those who are not present in the **photos table** user it means they never post a single post on instagram.

3. Determine the user on instagram who has got the most like on single post.

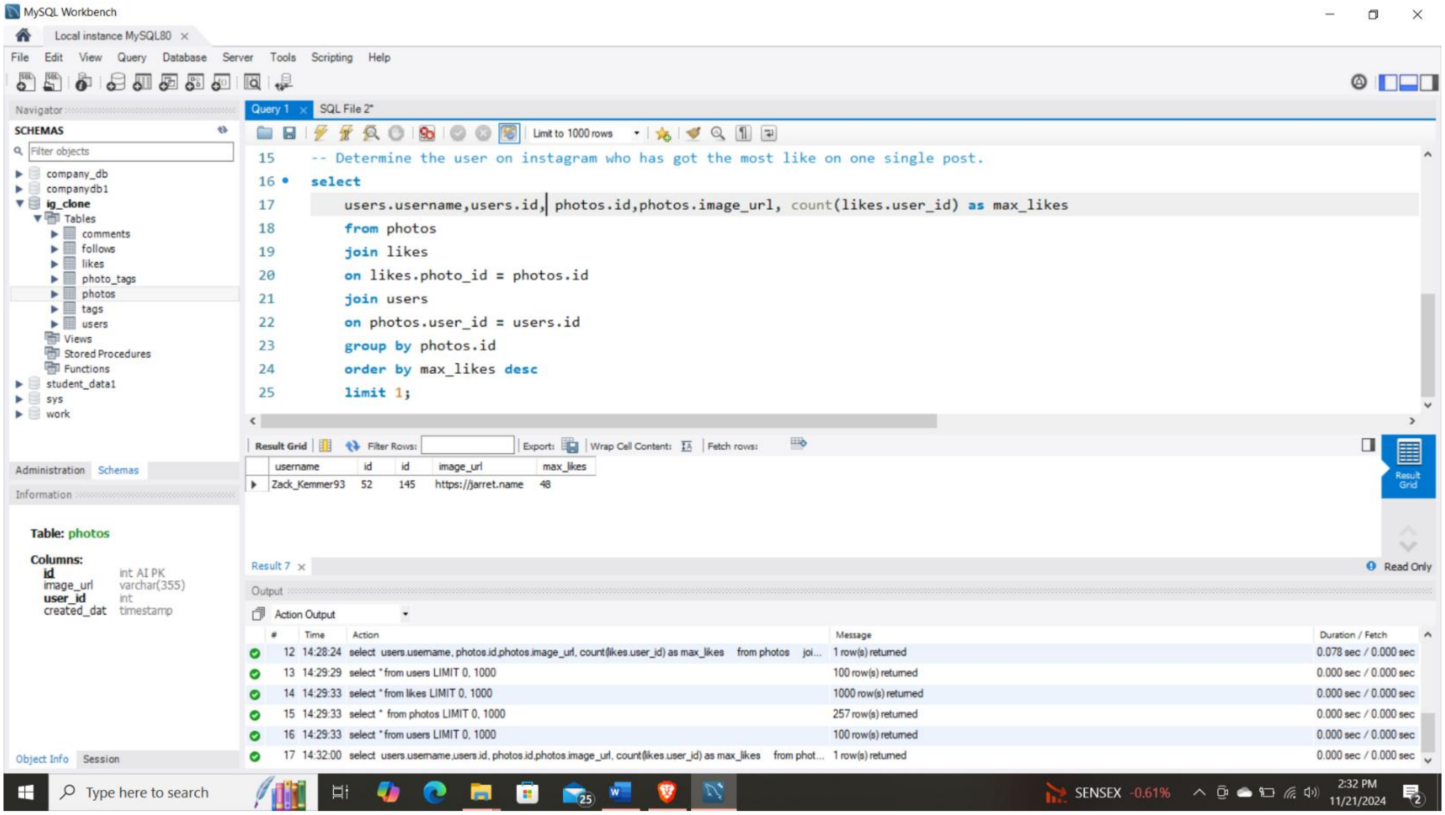


Table: photos

Columns:

id	int AI PK
image_url	varchar(355)
user_id	int
created_dat	timestamp

```sql
-- Determine the user on instagram who has got the most like on one single post.
select
    users.username,users.id, photos.id,photos.image_url, count(likes.user_id) as max_likes
    from photos
    join likes
    on likes.photo_id = photos.id
    join users
    on photos.user_id = users.id
    group by photos.id
    order by max_likes desc
    limit 1;
```

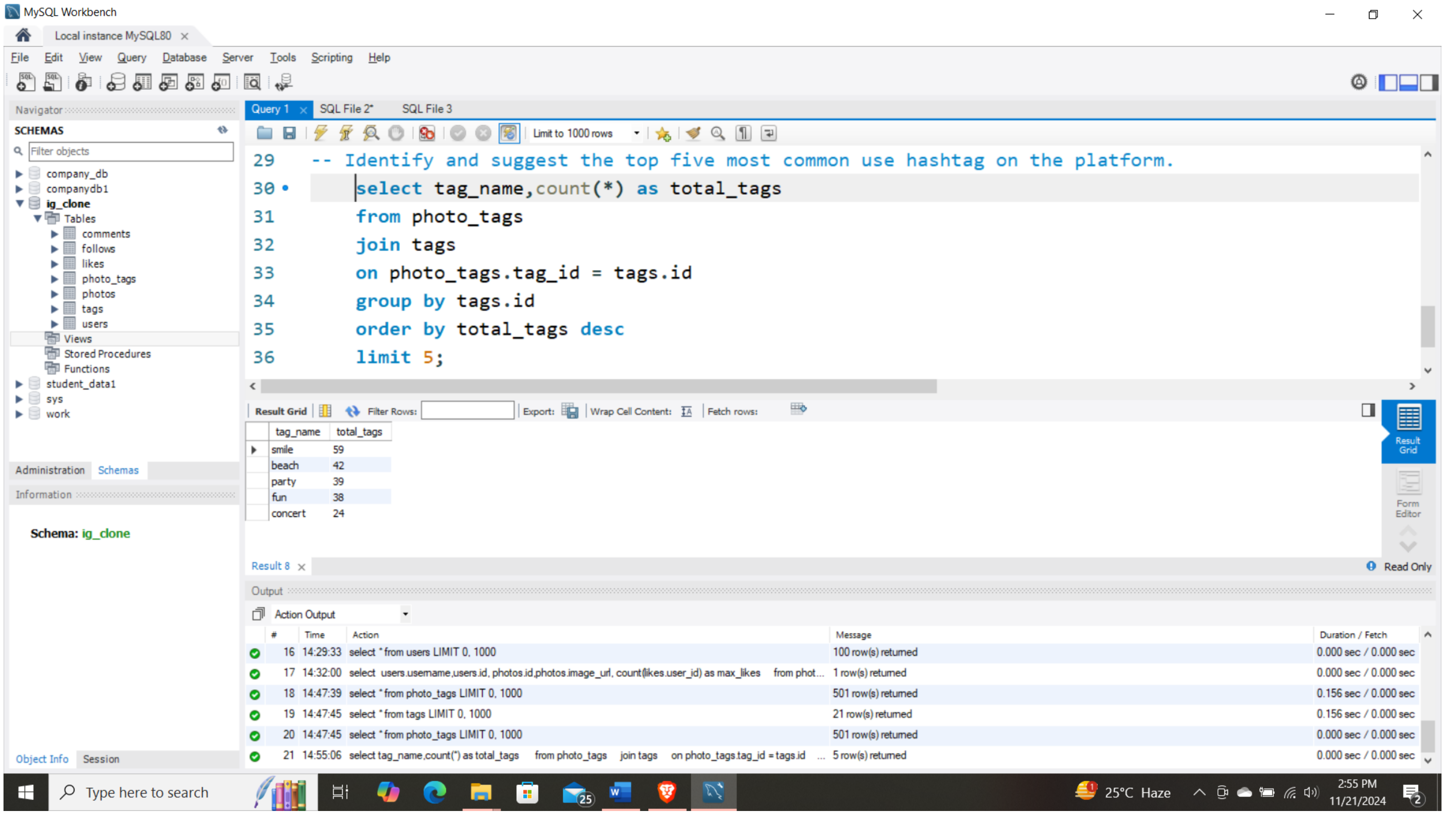
	username	id	id	image_url	max_likes
▶	Zack_Kemmer93	52	145	https://jarret.name	48

	#	Time	Action	Message	Duration / Fetch
✓	12	14:28:24	select users.username, photos.id,photos.image_url, count(likes.user_id) as max_likes from photos joi...	1 row(s) returned	0.078 sec / 0.000 sec
✓	13	14:29:29	select * from users LIMIT 0, 1000	100 row(s) returned	0.000 sec / 0.000 sec
✓	14	14:29:33	select * from likes LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec
✓	15	14:29:33	select * from photos LIMIT 0, 1000	257 row(s) returned	0.000 sec / 0.000 sec
✓	16	14:29:33	select * from users LIMIT 0, 1000	100 row(s) returned	0.000 sec / 0.000 sec
✓	17	14:32:00	select users.username,users.id, photos.id,photos.image_url, count(likes.user_id) as max_likes from phot...	1 row(s) returned	0.000 sec / 0.000 sec

Result: This query is little bit complicate coz we use **multiple inner join function** in this query. In this query they want to know which user has got the most like on one single post for we need retrieve information from three different table coz **likes table** has got the data about likes on each post and **photos** got the data about each post **and user table** got the user information so we just extract all the necessary attribute in the select clause join them using **inner join** and count the **likes.user_id** so that we can count and use **order by desc** function to find **the maximum likes** on single post . and in the last we use **limit 1** coz we need only **top one users**.

4. Identify and suggest the top five most common use hashtag on the platform.

Answer :-



```
-- Identify and suggest the top five most common use hashtag on the platform.
select tag_name,count(*) as total_tags
from photo_tags
join tags
on photo_tags.tag_id = tags.id
group by tags.id
order by total_tags desc
limit 5;
```

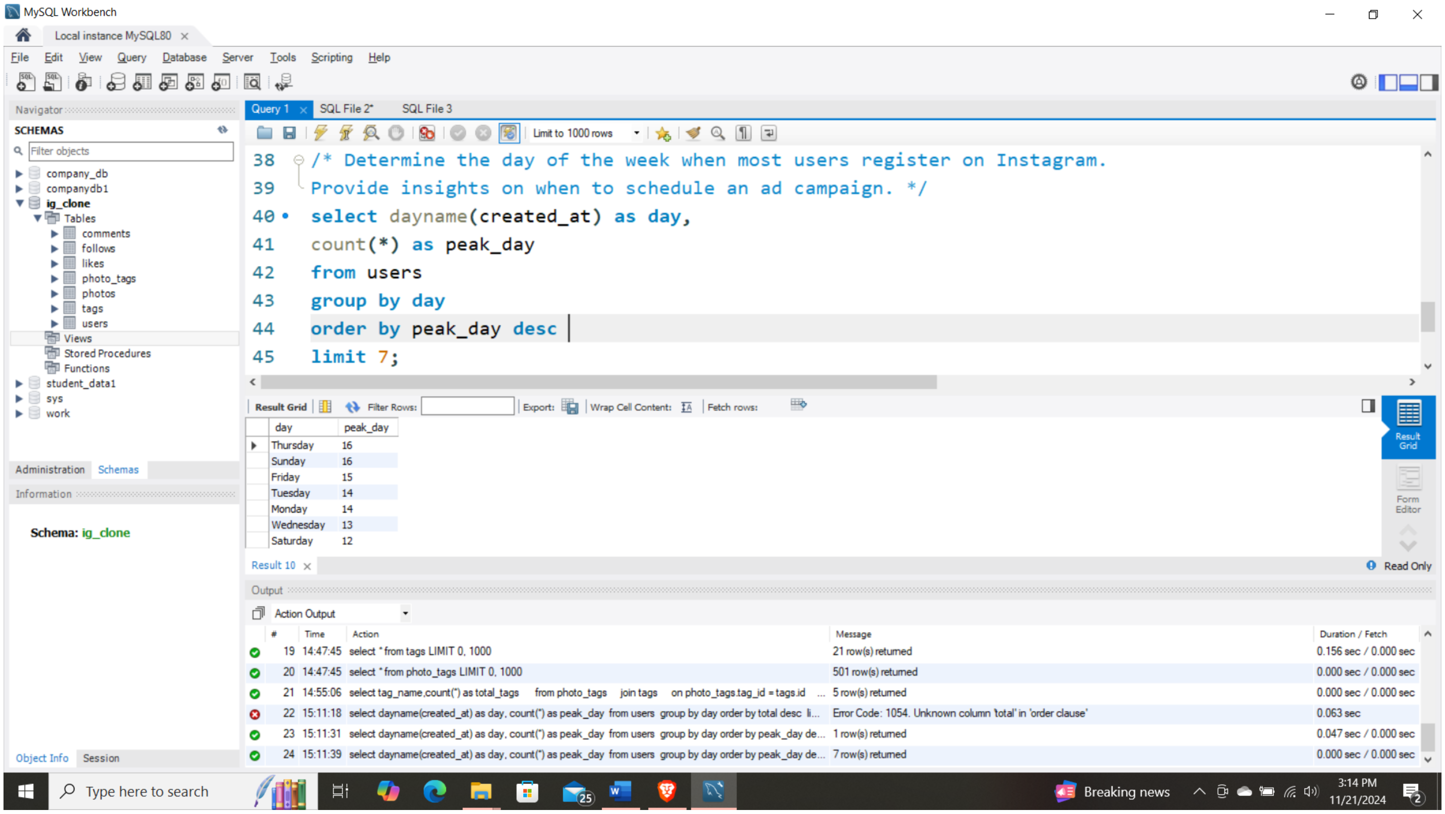
tag_name	total_tags
smile	59
beach	42
party	39
fun	38
concert	24

```
#   Time      Action                                                                                      Message             Duration / Fetch
16  14:29:33  select * from users LIMIT 0, 1000                                                           100 row(s) returned 0.000 sec / 0.000 sec
17  14:32:00  select  users.username,users.id, photos.id,photos.image_url, count(likes.user_id) as max_likes   from phot...  1 row(s) returned   0.000 sec / 0.000 sec
18  14:47:39  select * from photo_tags LIMIT 0, 1000                                                      501 row(s) returned 0.156 sec / 0.000 sec
19  14:47:45  select * from tags LIMIT 0, 1000                                                            21 row(s) returned  0.156 sec / 0.000 sec
20  14:47:45  select * from photo_tags LIMIT 0, 1000                                                      501 row(s) returned 0.000 sec / 0.000 sec
21  14:55:06  select tag_name,count(*) as total_tags   from photo_tags   join tags   on photo_tags.tag_id = tags.id   ...  5 row(s) returned   0.000 sec / 0.000 sec
```

Result:- this query is same as above query in this query we also use inner join and group by and order by this query is very simple as we have to find top most **commonly tag use** on the platform so what we did just **count tag_name** use count clause and give **alisas** to it **total_tags** and then use **order by clause desc** to find the **top hashtag** and group by **tag_id .**

5. Determine the day of the week when most users register on Instagram. Provide insights on when to schedule an ad campaign.

Answer:-



```
/* Determine the day of the week when most users register on Instagram.
Provide insights on when to schedule an ad campaign. */
select dayname(created_at) as day,
count(*) as peak_day
from users
group by day
order by peak_day desc
limit 7;
```

day	peak_day
Thursday	16
Sunday	16
Friday	15
Tuesday	14
Monday	14
Wednesday	13
Saturday	12

#	Time	Action	Message	Duration / Fetch
19	14:47:45	select * from tags LIMIT 0, 1000	21 row(s) returned	0.156 sec / 0.000 sec
20	14:47:45	select * from photo_tags LIMIT 0, 1000	501 row(s) returned	0.000 sec / 0.000 sec
21	14:55:06	select tag_name,count(*) as total_tags from photo_tags join tags on photo_tags.tag_id = tags.id ...	5 row(s) returned	0.000 sec / 0.000 sec
22	15:11:18	select dayname(created_at) as day, count(*) as peak_day from users group by day order by total desc li...	Error Code: 1054. Unknown column 'total' in 'order clause'	0.063 sec
23	15:11:31	select dayname(created_at) as day, count(*) as peak_day from users group by day order by peak_day de...	1 row(s) returned	0.047 sec / 0.000 sec
24	15:11:39	select dayname(created_at) as day, count(*) as peak_day from users group by day order by peak_day de...	7 row(s) returned	0.000 sec / 0.000 sec

Result: In this we have to find the peak day of the weeks so that we ad campian launch their product

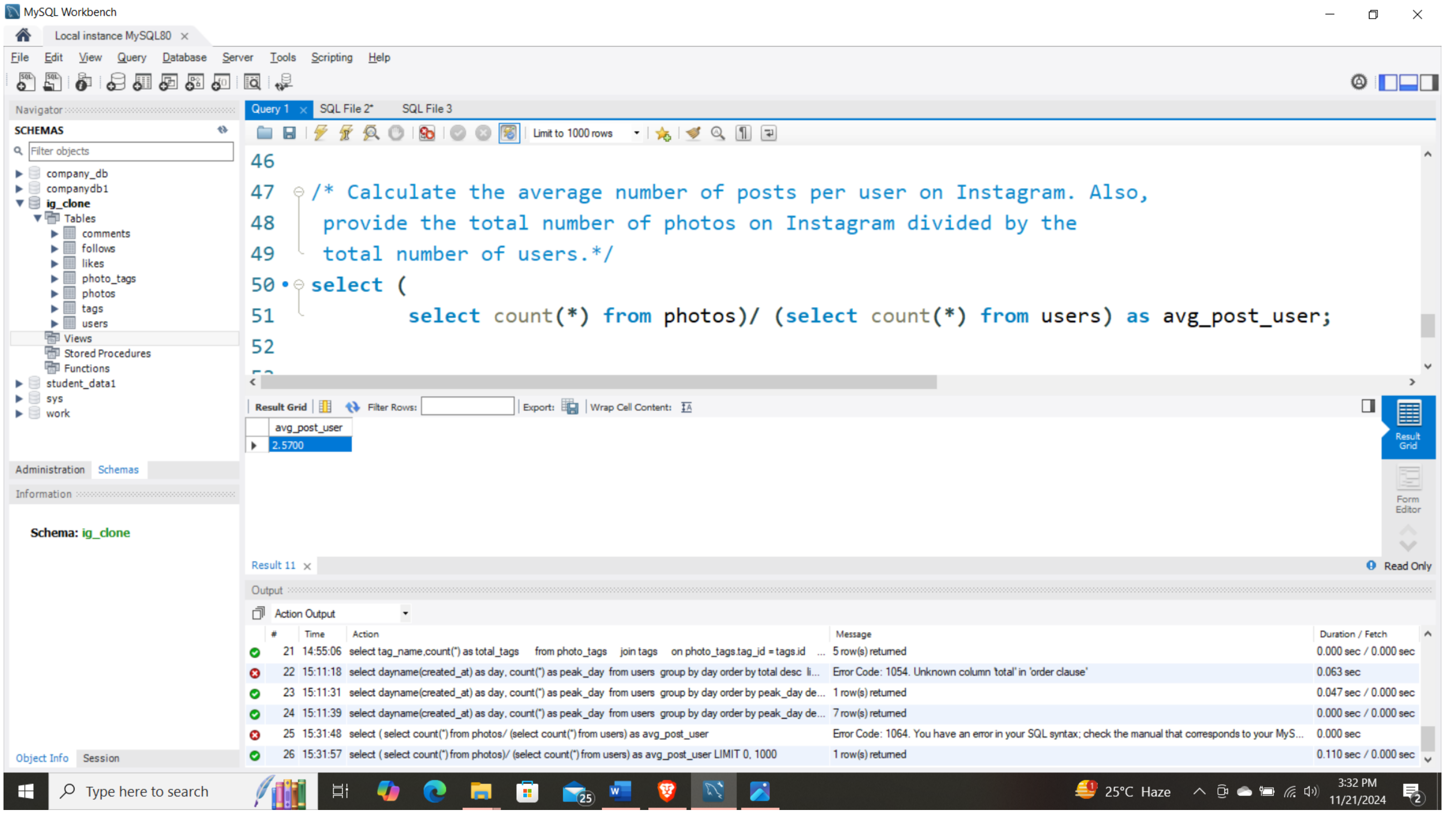
According to the need. So to **find the day on which the user register the most** first we use **dayname clause** coz and put in **created_at attrubute as an argument** coz **created_at** attribute has **the date of the users on which they register** and we give **alisas as day** and **group by day** in the last we **order by peak_day desc**, **peak_day is the alisas** that we created for **count the day** from the **dayname(created_at)** and then we use **limit 7** so that we find all seven days on which day users register the most.

B) INVESTOR METRICS

1. USER ENGAGEMENT

2. Calculate the average number of posts per user on Instagram. Also, provide the total number of photos on Instagram divided by the total number of users.

ANSWER:



```
/* Calculate the average number of posts per user on Instagram. Also,
 provide the total number of photos on Instagram divided by the
 total number of users.*/
select (
        select count(*) from photos)/ (select count(*) from users) as avg_post_user;
```

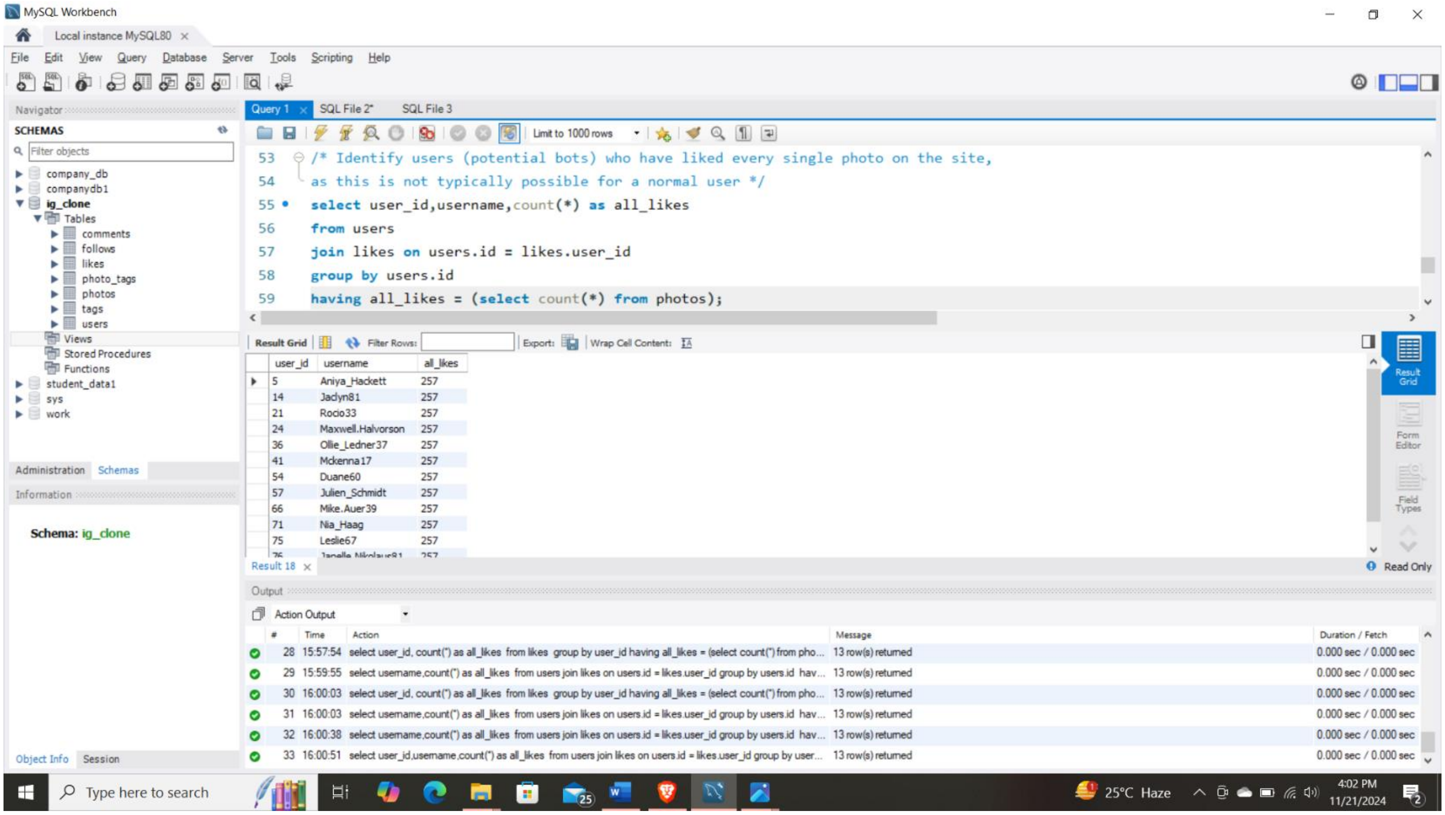
avg_post_user
2.5700

```
#   Time      Action                                                                                         Message                                                                          Duration / Fetch
21  14:55:06  select tag_name,count(*) as total_tags   from photo_tags   join tags   on photo_tags.tag_id = tags.id   ...  5 row(s) returned                                              0.000 sec / 0.000 sec
22  15:11:18  select dayname(created_at) as day, count(*) as peak_day  from users  group by day order by total desc  li...  Error Code: 1054. Unknown column 'total' in 'order clause'   0.063 sec
23  15:11:31  select dayname(created_at) as day, count(*) as peak_day  from users  group by day order by peak_day de...  1 row(s) returned                                              0.047 sec / 0.000 sec
24  15:11:39  select dayname(created_at) as day, count(*) as peak_day  from users  group by day order by peak_day de...  7 row(s) returned                                              0.000 sec / 0.000 sec
25  15:31:48  select ( select count(*) from photos/ (select count(*) from users) as avg_post_user   Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MyS...   0.000 sec
26  15:31:57  select ( select count(*) from photos)/ (select count(*) from users) as avg_post_user LIMIT 0, 1000   1 row(s) returned                                       0.110 sec / 0.000 sec
```

Result: In this we have **to find the average post** that post by user so for that we first **count the photos that post by user and then divide it to the total number of users** so that we can find the average post By the users.

7. Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.

Answer:-



```
/* Identify users (potential bots) who have liked every single photo on the site,
as this is not typically possible for a normal user */
select user_id,username,count(*) as all_likes
from users
join likes on users.id = likes.user_id
group by users.id
having all_likes = (select count(*) from photos);
```

user_id	username	all_likes
5	Aniya_Hackett	257
14	Jaclyn81	257
21	Rocio33	257
24	Maxwell.Halvorson	257
36	Ollie_Ledner37	257
41	Mckenna17	257
54	Duane60	257
57	Julien_Schmidt	257
66	Mike.Auer39	257
71	Nia_Haag	257
75	Leslie67	257
76	Janelle.Nikolaus81	257

```
 #   Time      Action                                                                                                   Message             Duration / Fetch
28   15:57:54  select user_id, count(*) as all_likes  from likes  group by user_id having all_likes = (select count(*) from pho...   13 row(s) returned   0.000 sec / 0.000 sec
29   15:59:55  select username,count(*) as all_likes  from users join likes on users.id = likes.user_id group by users.id  hav...   13 row(s) returned   0.000 sec / 0.000 sec
30   16:00:03  select user_id, count(*) as all_likes  from likes  group by user_id having all_likes = (select count(*) from pho...   13 row(s) returned   0.000 sec / 0.000 sec
31   16:00:03  select username,count(*) as all_likes  from users join likes on users.id = likes.user_id group by users.id  hav...   13 row(s) returned   0.000 sec / 0.000 sec
32   16:00:38  select username,count(*) as all_likes  from users join likes on users.id = likes.user_id group by users.id  hav...   13 row(s) returned   0.000 sec / 0.000 sec
33   16:00:51  select user_id,username,count(*) as all_likes  from users join likes on users.id = likes.user_id group by user...   13 row(s) returned   0.000 sec / 0.000 sec
```

Result :- In this we have to find the bot account so to find that we use logic is that those user who like all the post on the platform possibly they are bot account because it's not possible for user to like all the post on the platform. So to find the user who like all the post first we extract data from **users table and then likes table**, **likes table gives the information about the user who likes all the post** and through **users table we can extract the complete information of the users** who likes all the posts in the query we just join both table using **inner join then** count all the likes on the posts after that we use **having clause to compare it** with all the post in **the photos** .