Instagram User Analytics

Project Description:

The aim of this project was to analyse user activity data from an Instagram-like platform to evaluate user engagement, identify inactive or fake users, and generate meaningful insights for investors and decision-makers. The focus was on understanding posting trends, like behaviours, and detecting potential anomalies (such as bot activity) using SQL.

Approach:

The analysis was conducted using **SQL queries** in **MySQL Workbench** to extract and analyse data from a relational database containing user, photo, like, and tag information.
The process involved:

- Understanding database schema and relationships.
- Writing optimized SQL queries to answer business questions.
- Interpreting data patterns to derive insights.

Tech-Stack Used:

Used SQL and MySQL Workbench as tool to analyse Instagram user data and answer questions posted by the management team.

Insights:

Through this project, several key insights were discovered using only SQL fundamentals:

- Average posts per user and the ratio of photos to users, which helped evaluate overall activity.
- Identification of top 5 oldest loyal users based on account creation and posting activity.
- Detection of inactive users (users with no photo posts).
- Flagging of potential fake or bot users, particularly those who liked every single photo on the platform — behaviour not typical of genuine user engagement.

This analysis showed how powerful and insightful raw data can be when queried effectively.

Result:

- Successfully created SQL queries to answer investor-focused questions.
- Gained hands-on experience in data extraction and pattern recognition using MySQL.
- Developed a deeper understanding of identifying anomalies and engagement trends in user behaviour.
- The project reinforced how simple, structured queries can unlock high-value business insights from raw data.

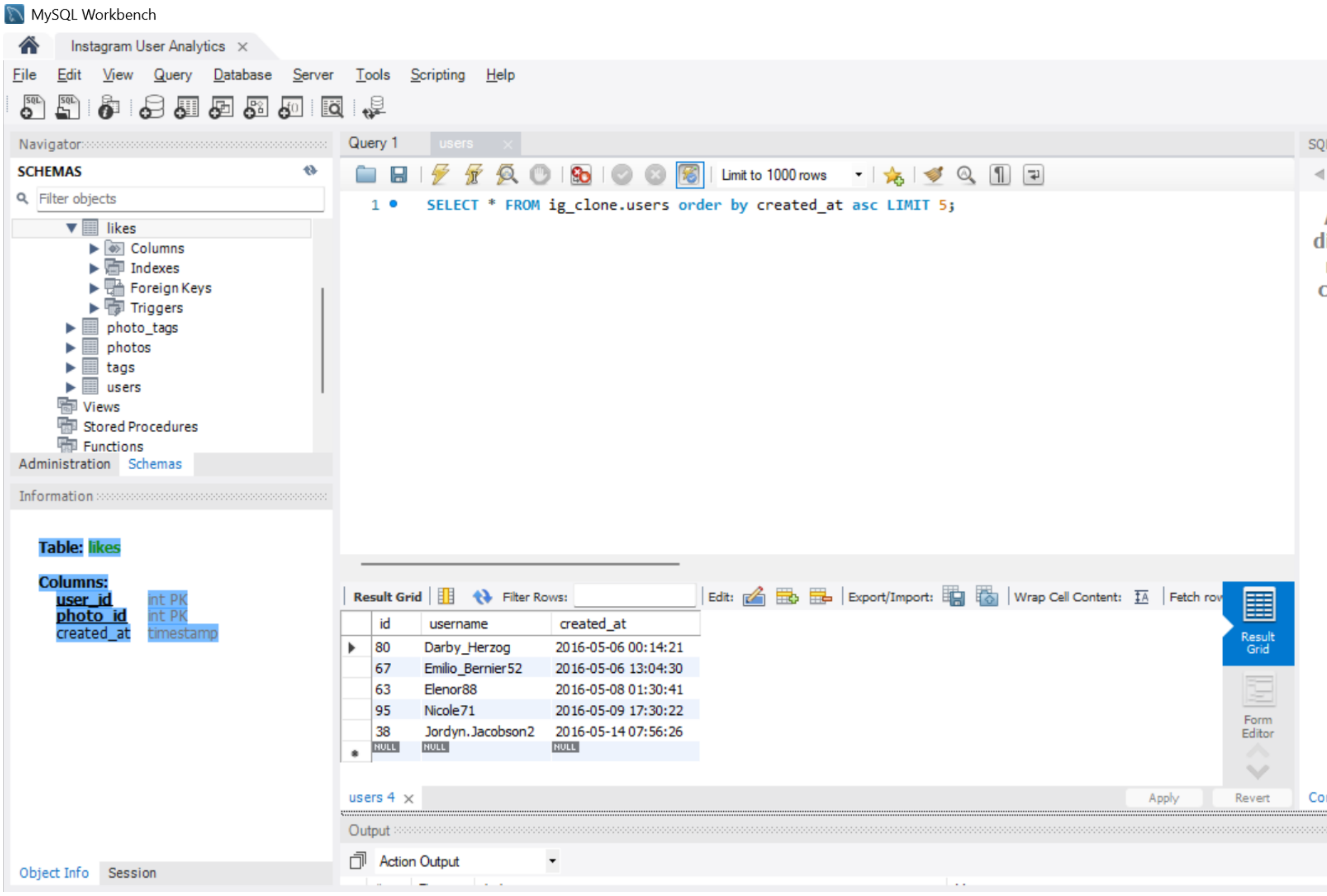
This analysis can support decision-making around user engagement strategies, bot detection, and platform health monitoring.

Drive Link:

https://drive.google.com/drive/folders/14xSZWZNHClp0jGWfHGRozjF67axRZMNB?usp=drive_link

SQL Tasks: - Marketing Analysis

1. Loyal User Reward task: Identify the five oldest users on Instagram from the provided database:



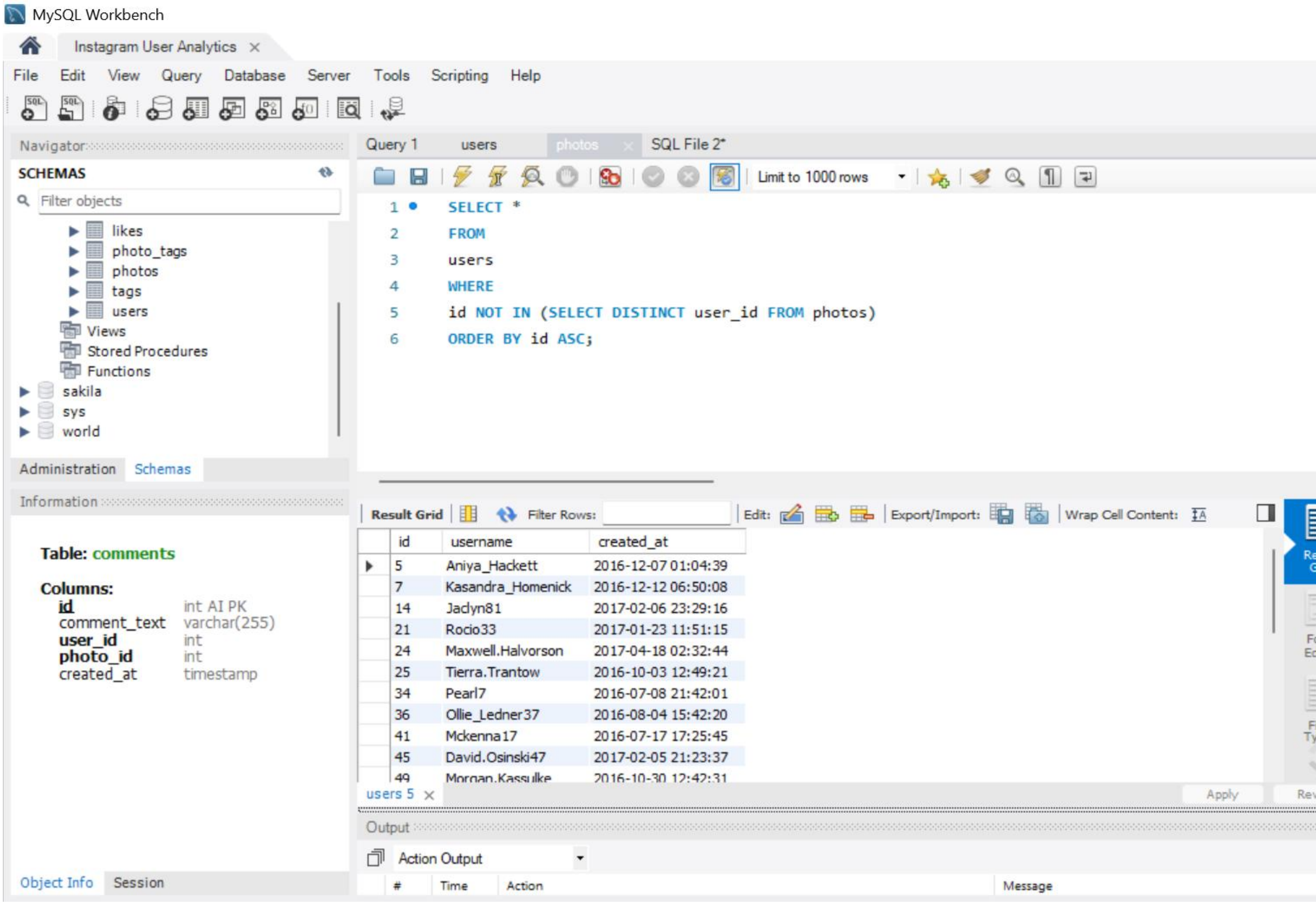
```
SELECT * FROM ig_clone.users order by created_at asc LIMIT 5;
```

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

Table: likes

Columns:
user_id int PK
photo_id int PK
created_at timestamp

2. Inactive User Engagement Task: Identify users who have never posted a single photo on Instagram.



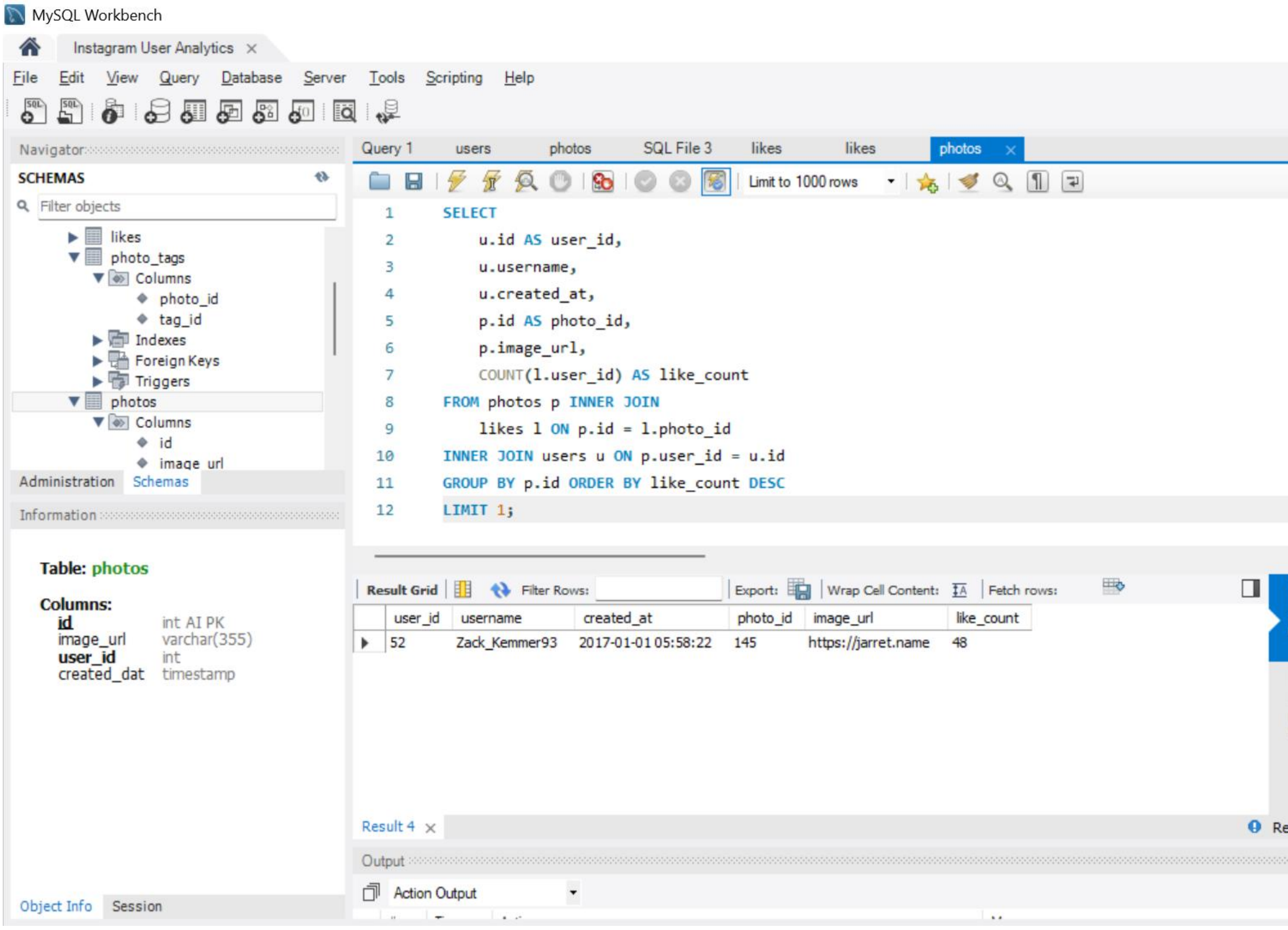
```sql
SELECT *
FROM
users
WHERE
id NOT IN (SELECT DISTINCT user_id FROM photos)
ORDER BY id ASC;
```

Table: comments

Columns:
id int AI PK
comment_text varchar(255)
user_id int
photo_id int
created_at timestamp

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

3. Contest Winner Declaration Task: The team has organized a contest where the user with the most likes on a single photo wins. Determine the winner of the contest and provide their details to the team.



```sql
SELECT
    u.id AS user_id,
    u.username,
    u.created_at,
    p.id AS photo_id,
    p.image_url,
    COUNT(l.user_id) AS like_count
FROM photos p INNER JOIN
    likes l ON p.id = l.photo_id
INNER JOIN users u ON p.user_id = u.id
GROUP BY p.id ORDER BY like_count DESC
LIMIT 1;
```

user_id	username	created_at	photo_id	image_url	like_count
52	Zack_Kemmer93	2017-01-01 05:58:22	145	https://jarret.name	48

Table: photos

Columns:

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

Insights: The winner of the contest for the user with the most likes on a single photo is **Zack_Kemmer93**

4. HashTags Research Task: A partner brand wants to know the most popular hashtags to use in their posts to reach the most people. Identify and suggest the top five most commonly used hashtags on the platform.

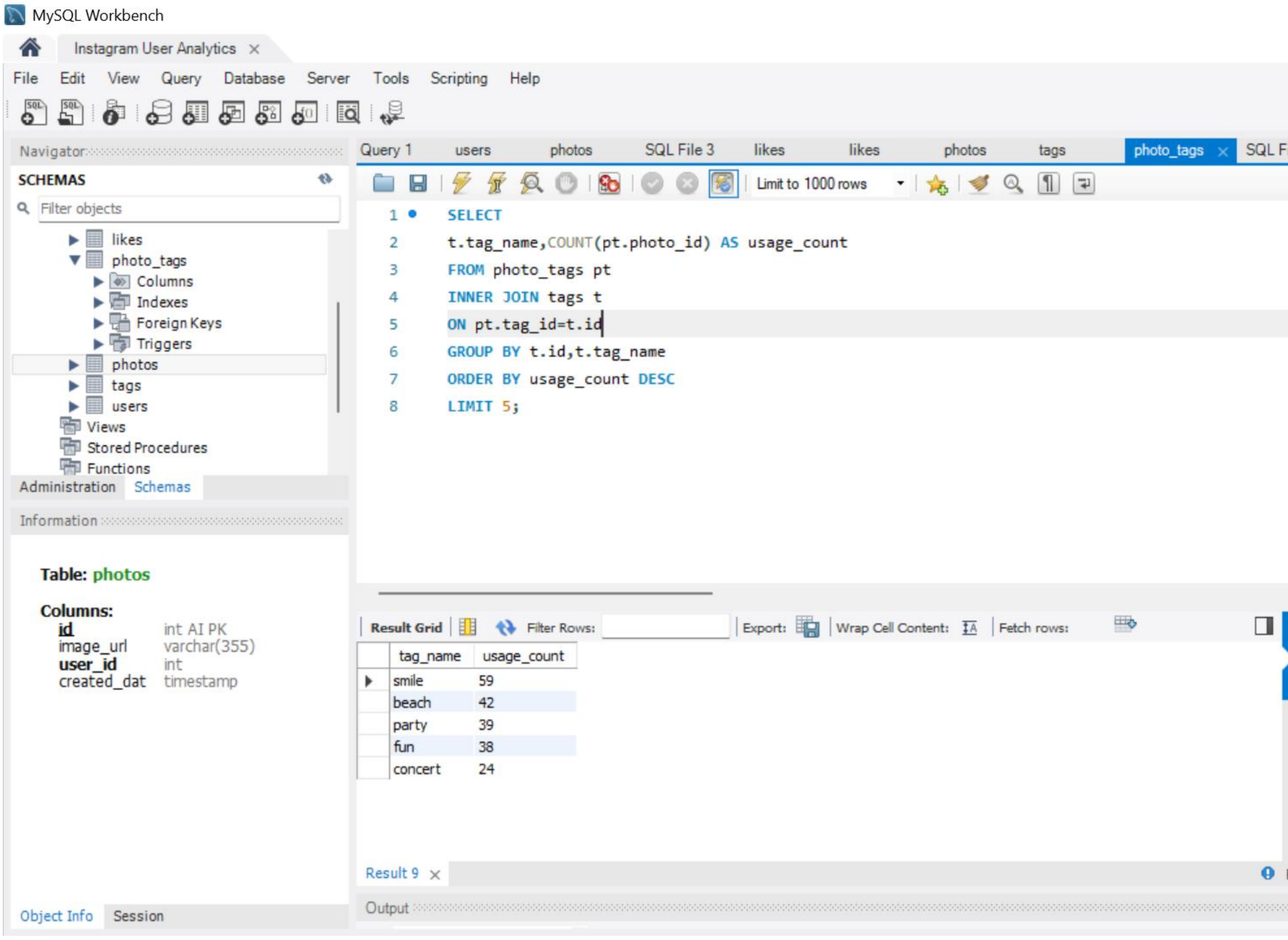


Table: photos

Columns:

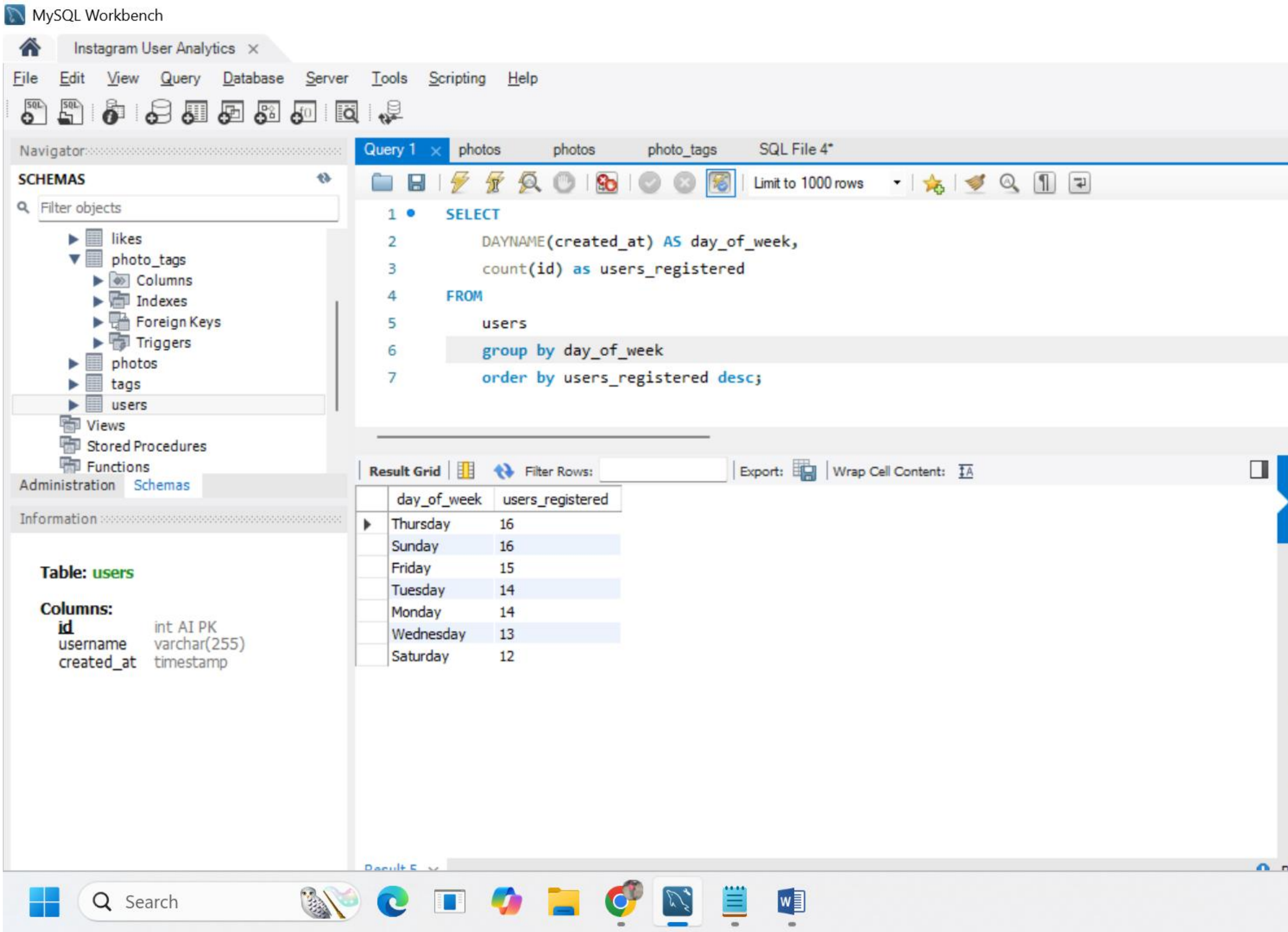
id	int AI PK
image_url	varchar(355)
user_id	int
created_dat	timestamp

```
SELECT
t.tag_name,COUNT(pt.photo_id) AS usage_count
FROM photo_tags pt
INNER JOIN tags t
ON pt.tag_id=t.id
GROUP BY t.id,t.tag_name
ORDER BY usage_count DESC
LIMIT 5;
```

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

Insights: The Five most popular hashtags used in the posts to reach maximum people are as: **#smile, #beach, #party, #fun, #concert**

5. Ad Campaign Launch Task: The team wants to know the best day of the week to launch ads. 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_of_week,
    count(id) as users_registered
FROM
    users
    group by day_of_week
    order by users_registered desc;
```

day_of_week	users_registered
Thursday	16
Sunday	16
Friday	15
Tuesday	14
Monday	14
Wednesday	13
Saturday	12

Table: users

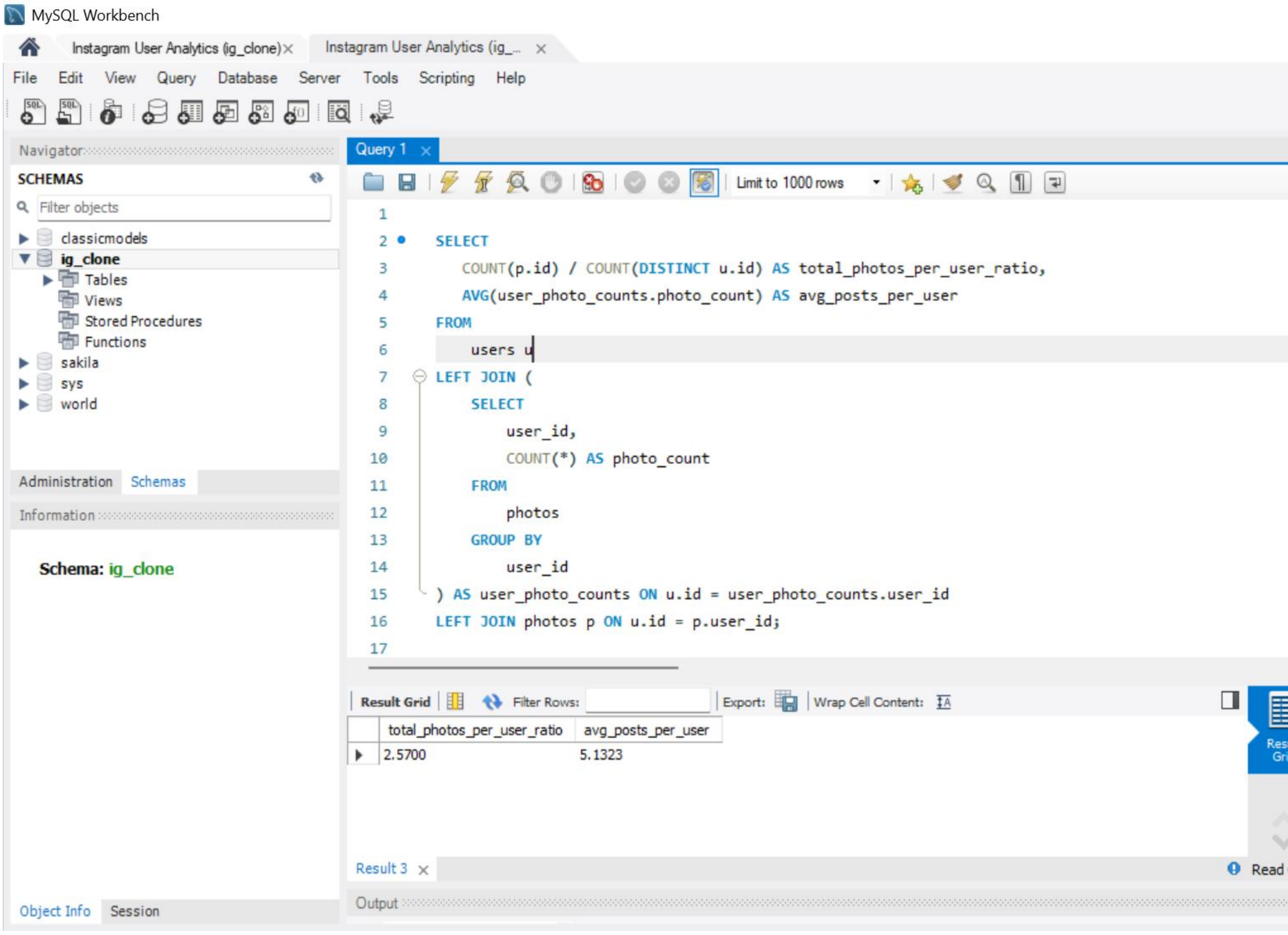
Columns:
id int AI PK
username varchar(255)
created_at timestamp

Insights: Most of the users have registered on Instagram are on Thursdays and Sundays. Hence best day of a week to schedule an AD Campaign would be **Thursday and Sunday**.

SQL Tasks: - Investors Metrics

1. User Engagement: Investors want to know if users are still active and posting on Instagram or if they are making fewer posts.

Your Task: 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
    COUNT(p.id) / COUNT(DISTINCT u.id) AS total_photos_per_user_ratio,
    AVG(user_photo_counts.photo_count) AS avg_posts_per_user
FROM
    users u
LEFT JOIN (
    SELECT
        user_id,
        COUNT(*) AS photo_count
    FROM
        photos
    GROUP BY
        user_id
) AS user_photo_counts ON u.id = user_photo_counts.user_id
LEFT JOIN photos p ON u.id = p.user_id;
```

total_photos_per_user_ratio	avg_posts_per_user
2.5700	5.1323

Insights: Average number of posts per user on Instagram is **5.1323** and the total number of photos on Instagram divided by the total number of users is **2.5700**

Bots & Fake Accounts: Investors want to know if the platform is crowded with fake and dummy accounts.

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

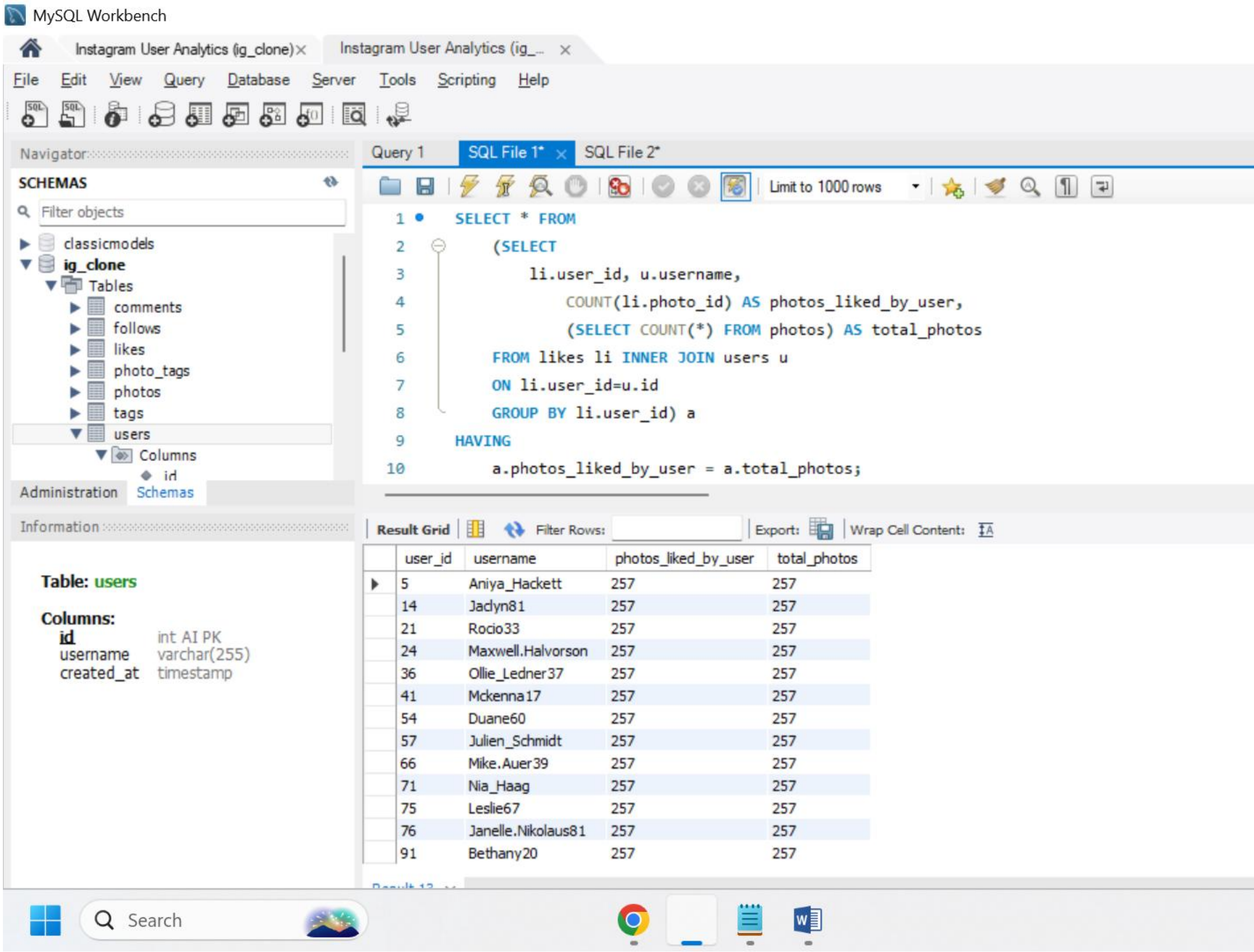


Table: users

Columns:

id	int AI PK
username	varchar(255)
created_at	timestamp

```
SELECT * FROM
    (SELECT
        li.user_id, u.username,
            COUNT(li.photo_id) AS photos_liked_by_user,
            (SELECT COUNT(*) FROM photos) AS total_photos
    FROM likes li INNER JOIN users u
    ON li.user_id=u.id
    GROUP BY li.user_id) a
HAVING
    a.photos_liked_by_user = a.total_photos;
```

user_id	username	photos_liked_by_user	total_photos
5	Aniya_Hackett	257	257
14	Jaclyn81	257	257
21	Rocio33	257	257
24	Maxwell.Halvorson	257	257
36	Ollie_Ledner37	257	257
41	Mckenna17	257	257
54	Duane60	257	257
57	Julien_Schmidt	257	257
66	Mike.Auer39	257	257
71	Nia_Haag	257	257
75	Leslie67	257	257
76	Janelle.Nikolaus81	257	257
91	Bethany20	257	257