Project: Instagram Clone Database Analysis

Author – Akash Prasad Sah

Introduction

- In this project, we will be analyzing the Instagram Clone database ('ig_clone') to extract valuable insights
- And answer specific questions related to user activity, content engagement, and more.
- Our goal is to provide meaningful data-driven recommendations to support decision-making for marketing, user engagement, and overall platform improvement.

'ig_clone' Database Schema

Key Tables

1. **Users Table:**
    - Fields: `id` (Primary Key), `username` (Unique), `created_at` (Timestamp).
    - Stores user information with a unique username and creation timestamp.
2. **Photos Table:**
    - Fields: `id` (Primary Key), `image_url`, `user_id` (Foreign Key), `created_at` (Timestamp).
    - Stores information about user-uploaded photos with image URL, linked to the Users table.
3. **Comments Table:**
    - Fields: `id` (Primary Key), `comment_text`, `photo_id` (Foreign Key), `user_id` (Foreign Key), `created_at` (Timestamp).
    - Stores user comments on photos, linked to both Users and Photos tables.
4. **Likes Table:**
    - Fields: `user_id` (Foreign Key), `photo_id` (Foreign Key), `created_at` (Timestamp).
    - Records likes on photos, with a composite primary key linking Users and Photos tables.
5. **Follows Table:**
    - Fields: `follower_id` (Foreign Key), `followee_id` (Foreign Key), `created_at` (Timestamp).
    - Represents user follow relationships, with a composite primary key linking Users table.
6. **Tags Table:**
    - Fields: `id` (Primary Key), `tag_name` (Unique), `created_at` (Timestamp).
    - Stores unique tags for categorizing photos.
7. **Photo_Tags Table:**
    - Fields: `photo_id` (Foreign Key), `tag_id` (Foreign Key).
    - Connects photos with associated tags, forming a composite primary key.

SQL Queries for Instagram Clone Database

Queries

1. Create an ER diagram or draw a schema for the given database.
2. Find the 5 oldest users.
3. Identify the day of the week most users register on.
4. Target inactive users in an ad campaign.
5. Determine the winner of a photo likes contest.
6. Calculate the average number of posts per user.
7. Find the top 5 most used hashtags.
8. Identify users who liked every photo on the site.
9. Discover users who have never commented on a photo.
10. Find users who have never commented on any photo or have commented on every photo.
11. Demonstrate the top 30 usernames to the company who have posted photos in the range of 3 to 5.
12. Can you help me find the users whose name starts with c and ends with any number and have posted the photos as well as liked the photos?
13. Find the users who have created Instagram id in May and select top 5 newest Joines from it?

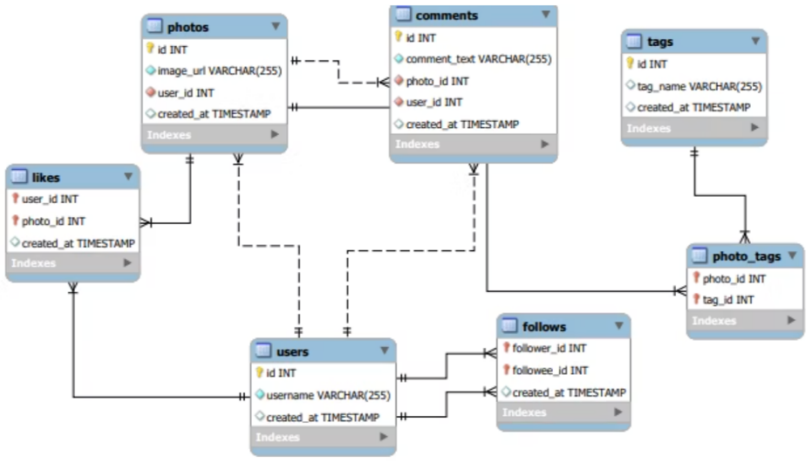
Q.1) ER Diagram for ig_clone Database

Entities:

- Users: Registered users of the app with profile information.
- Photos: Images uploaded by users with associated metadata.
- Tags: Keywords assigned to photos for categorization and organization.
- Follows: User-to-user relationships indicating who follows whom.
- Likes: User-to-photo relationships indicating user preferences.

Relationships:

- One-to-One Relationship: Each user has one profile photo, and each photo belongs to one user.
- One-to-Many Relationship: A user uploads multiple photos, but each photo belongs to only one user.
- Many-to-Many Relationship: A photo has multiple tags, and a tag can be applied to multiple photos.
- One-to-Many Relationship: A user follows multiple users, but each user can only be followed by other users.
- Many-to-Many Relationship: A user likes multiple photos, and a photo can be liked by multiple users.



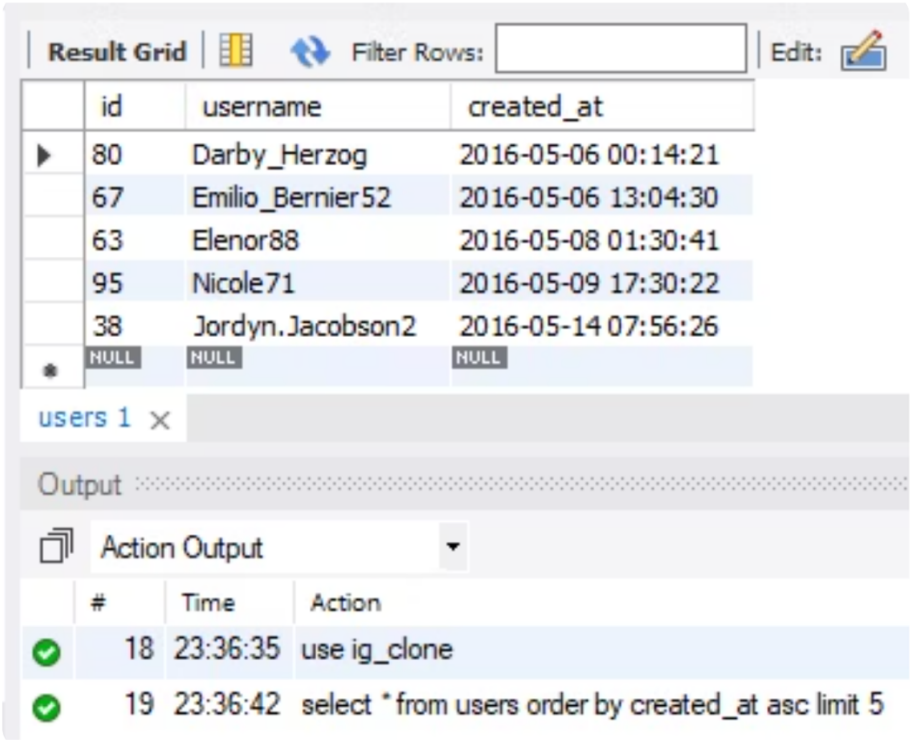
Q.2) We want to reward the user who has been around the longest, Find the 5 oldest users.

Code:

```
SELECT *
FROM users
ORDER BY created_at ASC
LIMIT 5;
```

Insights:

Recognizing and rewarding long-time users is crucial for user retention. These users have been a part of the platform since its early days, showcasing loyalty and engagement.



Q.3) To understand when to run the ad campaign, figure out the day of the week most users register on?

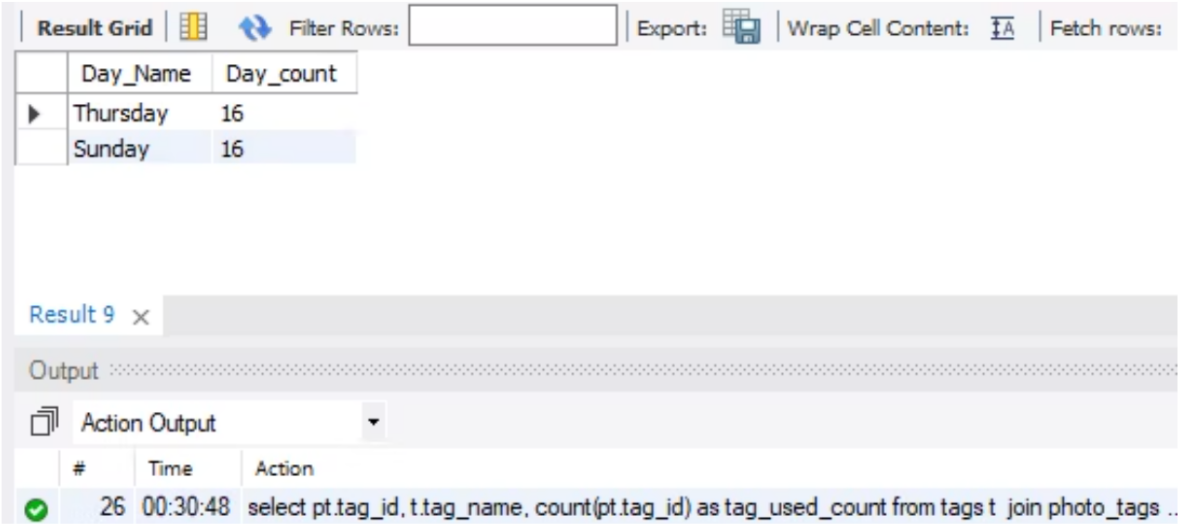
Code:

```
SELECT DAYNAME(created_at) AS Day_Name,
COUNT(*) AS Day_count
FROM users
GROUP BY Day_Name
LIMIT 2;
```

Insights:

Knowing the days of the week when most users register is valuable for planning targeted ad campaigns. This data can

help optimize ad placement and timing to reach a larger audience.



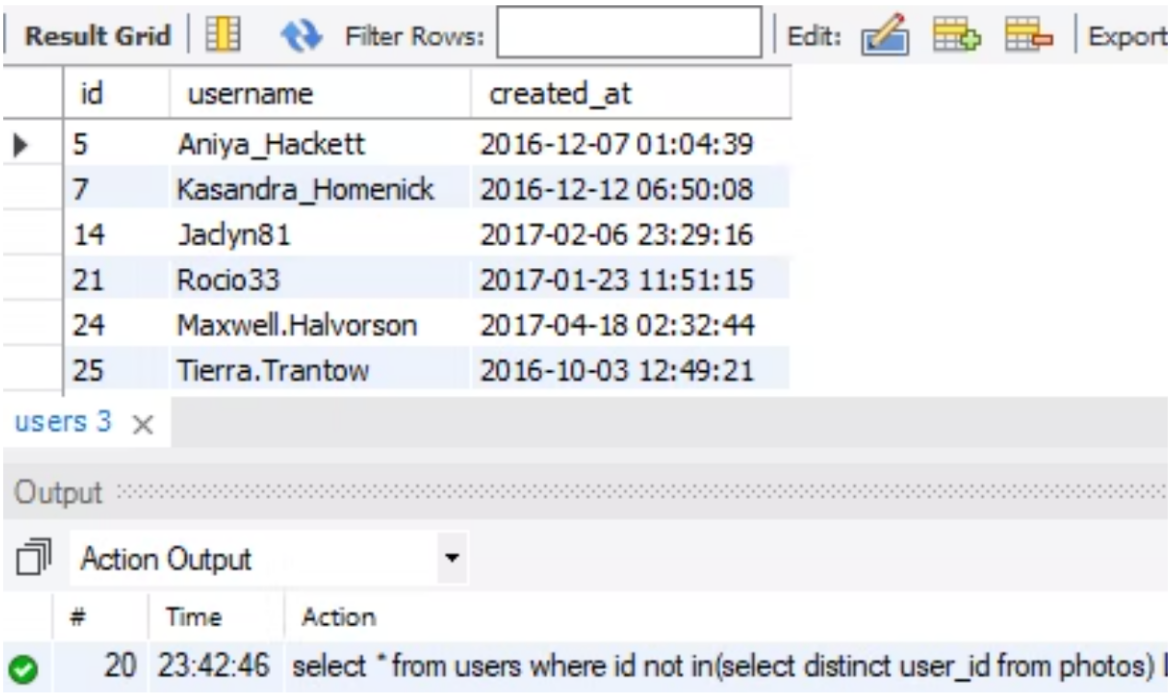
Q.4) To target inactive users in an email ad campaign, find the users who have never posted a photo?

Code:

```
Select * FROM users WHERE id
NOT IN(Select distinct user_id from photos);
```

Insights:

identifying users who have never posted a photo allows for targeted email ad campaigns to re-engage and encourage them to contribute content, enhancing overall platform activity.



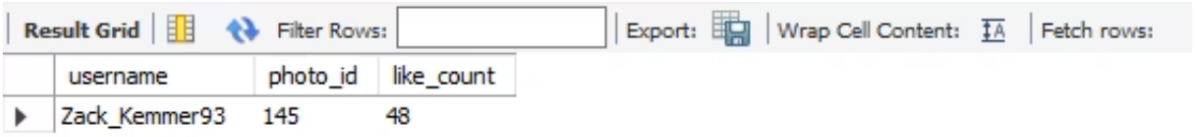
Q.5) Suppose you are running a contest to find out who got the most likes on a photo. Find out who won?

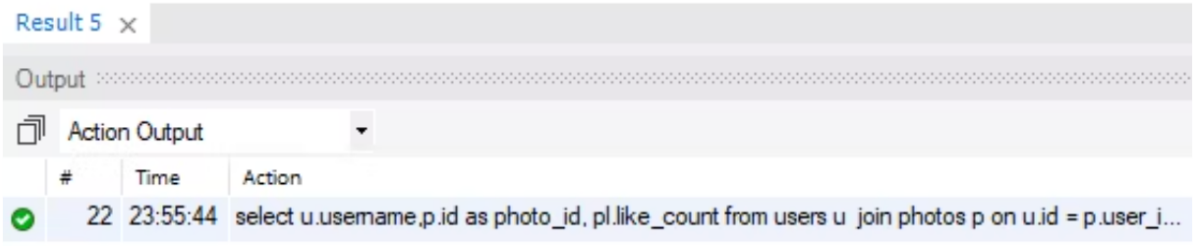
Code:

```
select u.username, p.id as photo_id, pl.like_count from users u
join photos p on u.id = p.user_id
join (select photo_id, count(*) like_count from likes group by
photo_id) as pl on p.id = pl.photo_id order by like_count desc
limit 1;
```

Insights:

Running contests based on likes fosters engagement. Recognizing users with the most likes not only boosts their profile but also encourages healthy competition and interaction on the platform.





Q.6) The investors want to know how many times does the average user post.

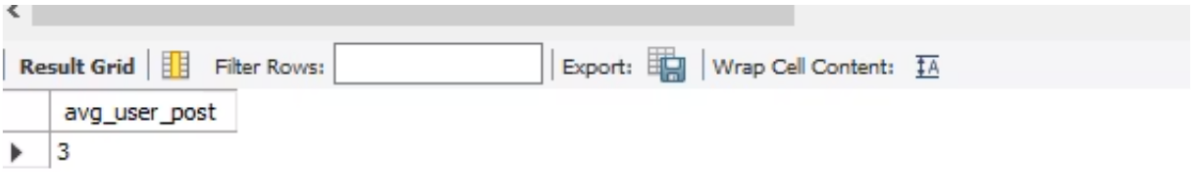
Code:

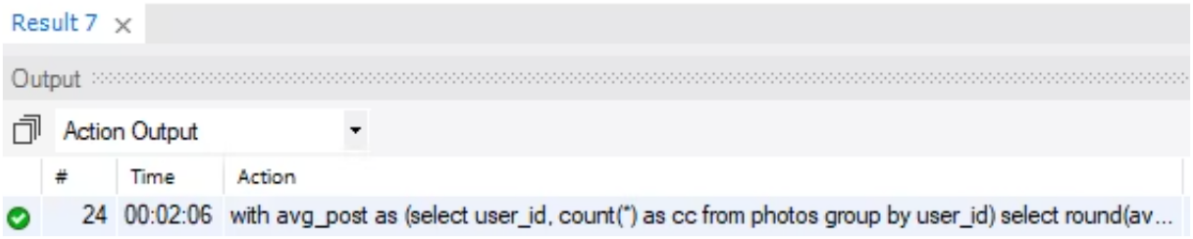
```
With avg_post as (select user_id, count(*) as cc from
photos group by user_id)
select round(avg(cc)) as avg_user_post from avg_post;
```

Insights:

Investors seeking information on user activity find the average post frequency valuable. This metric indicates how

actively users are contributing content, impacting the overall vibrancy of the platform.





Q.7) A brand wants to know which hashtag to use on a post and find the top 5 most used hashtags.

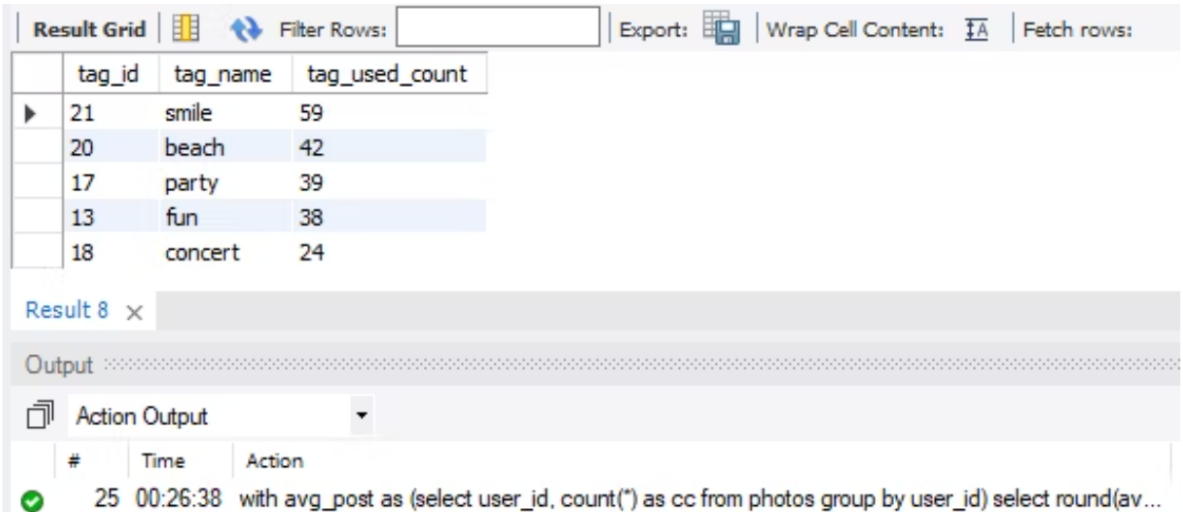
Code:

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

Insights:

For brand strategy, understanding the most used hashtags

provides insights into popular trends and user interests. Brands can leverage these hashtags to maximize the visibility of their content.



Q.8) To find out if there are bots, find users who have liked every single photo on the site.

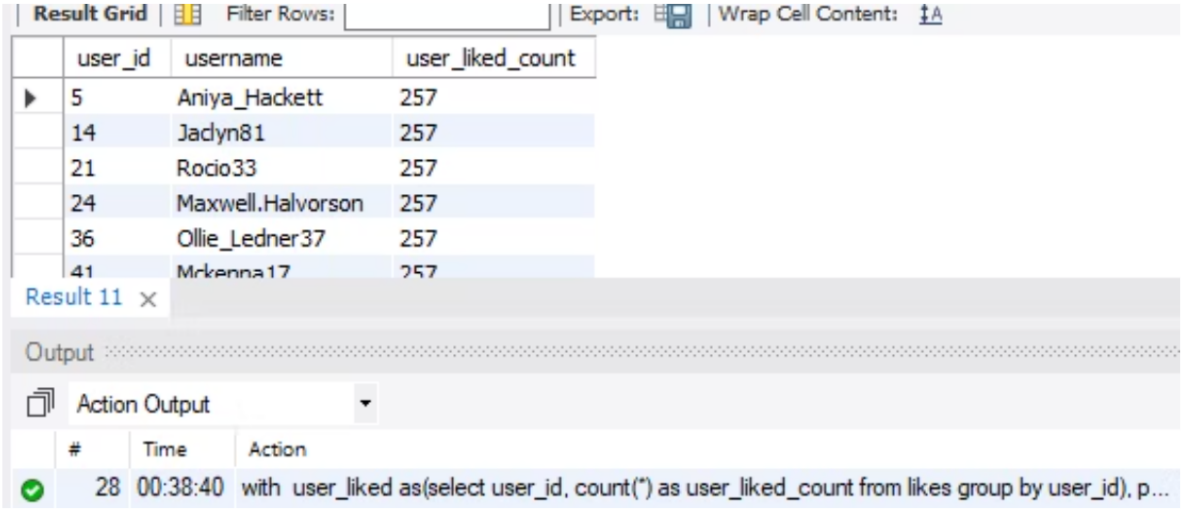
Code:

```
With user_liked as(select user_id, count(*) as user_liked_count
from likes group by user_id),
photos_id as (select count(distinct photo_id) as
photo_id_count from likes)
select ul.user_id, u.username, user_liked_count from users u
join user_liked ul on u.id = ul.user_id
where ul.user_liked_count = (select photo_id_count from
photos_id);
```

Insights:

Identifying users who have liked every single photo raises

awareness about potential bot activity. This information is crucial for maintaining the integrity and authenticity of user engagement on the platform.



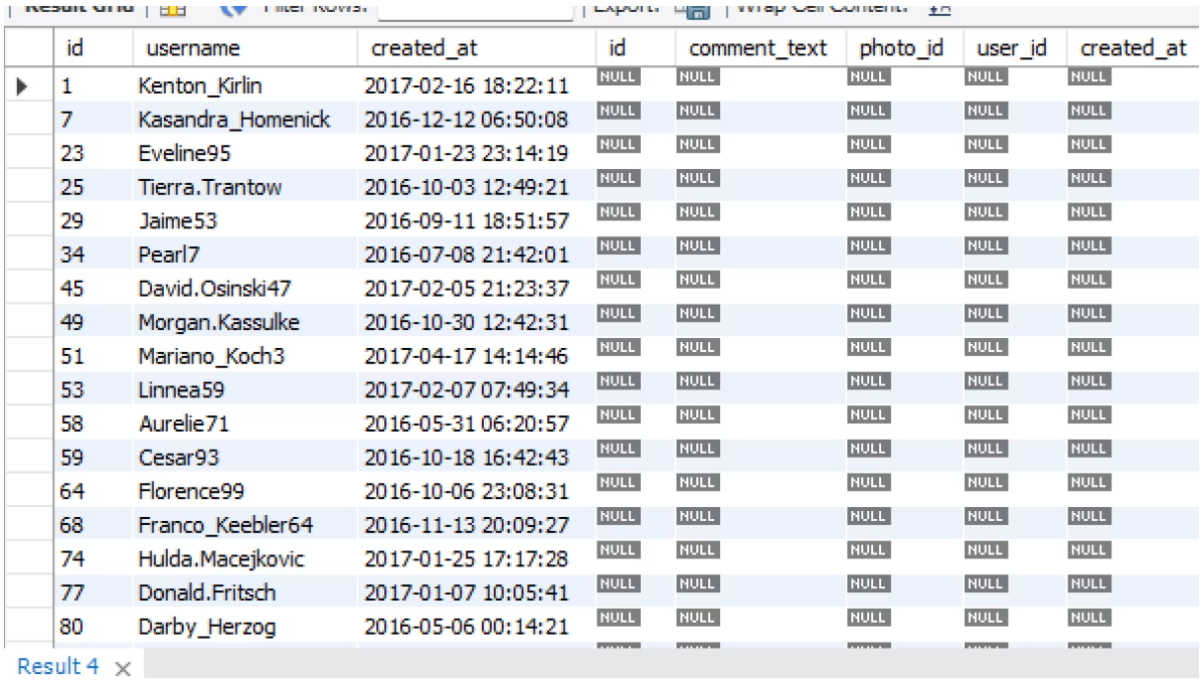
Q.9) To know who the celebrities are, find users who have never commented on a photo.

Code:

```
SELECT *FROM users LEFT JOIN comments ON
users.id = comments.user_id WHERE comments.id IS
NULL;
```

Insights:

Celebrities often have a significant following but may not actively comment. Recognizing users who have never commented helps identify potential celebrities or influencers on the platform.



id	username	created_at	id	comment_text	photo_id	user_id	created_at
1	Kenton_Kirlin	2017-02-16 18:22:11	NULL	NULL	NULL	NULL	NULL
7	Kasandra_Homenick	2016-12-12 06:50:08	NULL	NULL	NULL	NULL	NULL
23	Eveline95	2017-01-23 23:14:19	NULL	NULL	NULL	NULL	NULL
25	Tierra.Trantow	2016-10-03 12:49:21	NULL	NULL	NULL	NULL	NULL
29	Jaime53	2016-09-11 18:51:57	NULL	NULL	NULL	NULL	NULL
34	Pearl7	2016-07-08 21:42:01	NULL	NULL	NULL	NULL	NULL
45	David.Osinski47	2017-02-05 21:23:37	NULL	NULL	NULL	NULL	NULL
49	Morgan.Kassulke	2016-10-30 12:42:31	NULL	NULL	NULL	NULL	NULL
51	Mariano_Koch3	2017-04-17 14:14:46	NULL	NULL	NULL	NULL	NULL
53	Linnea59	2017-02-07 07:49:34	NULL	NULL	NULL	NULL	NULL
58	Aurelie71	2016-05-31 06:20:57	NULL	NULL	NULL	NULL	NULL
59	Cesar93	2016-10-18 16:42:43	NULL	NULL	NULL	NULL	NULL
64	Florence99	2016-10-06 23:08:31	NULL	NULL	NULL	NULL	NULL
68	Franco_Keebler64	2016-11-13 20:09:27	NULL	NULL	NULL	NULL	NULL
74	Hulda.Macejkovic	2017-01-25 17:17:28	NULL	NULL	NULL	NULL	NULL
77	Donald.Fritsch	2017-01-07 10:05:41	NULL	NULL	NULL	NULL	NULL
80	Darby_Herzog	2016-05-06 00:14:21	NULL	NULL	NULL	NULL	NULL

Result 4

Q.10) Now it's time to find both of them together, find the users who have never commented on any photo or have commented on every photo.

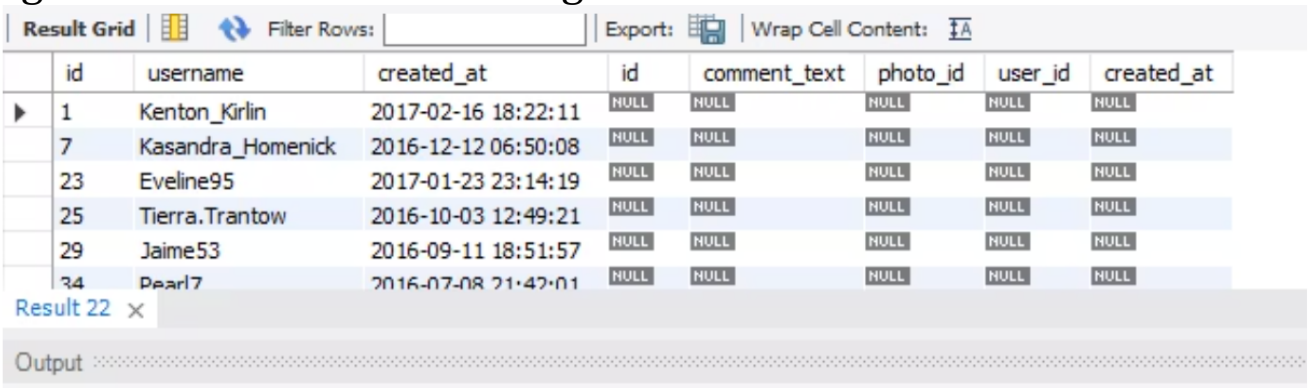
Code:

```
SELECT *FROM users LEFT JOIN comments ON users.id = comments.user_id WHERE comments.id IS NULL UNION ALL SELECT *FROM users LEFT JOIN comments ON users.id = comments.user_id;
```

Insights:

Combining users who have never commented with those who

have commented on every photo provides a nuanced understanding of user engagement patterns. It allows for targeted communication strategies.


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id	username	created_at	id	comment_text	photo_id	user_id	created_at
1	Kenton_Kirlin	2017-02-16 18:22:11	NULL	NULL	NULL	NULL	NULL
7	Kasandra_Homenick	2016-12-12 06:50:08	NULL	NULL	NULL	NULL	NULL
23	Eveline95	2017-01-23 23:14:19	NULL	NULL	NULL	NULL	NULL
25	Tierra.Trantow	2016-10-03 12:49:21	NULL	NULL	NULL	NULL	NULL
29	Jaime53	2016-09-11 18:51:57	NULL	NULL	NULL	NULL	NULL
34	Pearl7	2016-07-08 21:42:01	NULL	NULL	NULL	NULL	NULL

Result 22 ×

Output

Q.11) Demonstrate the top 30 usernames to the company who have posted photos in the range of 3 to 5.

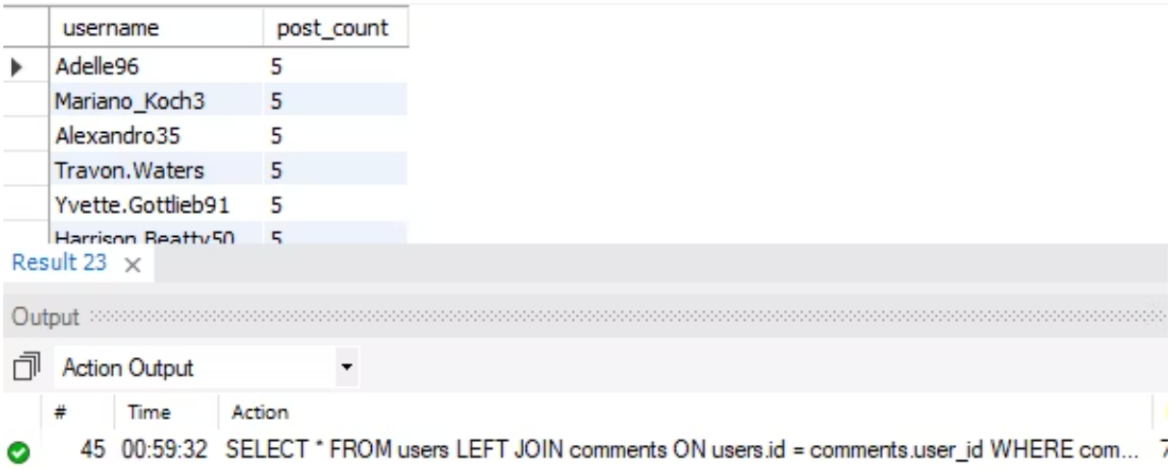
Code:

```
select u.username, count(user_id) as post_count from
photos p join users u on p.user_id = u.id group by user_id
having post_count between 3 and 5
order by post_count desc limit 30;
```

Insights:

The query identifies users posting 3 to 5 photos, emphasizing

their diverse contributions for community building. Visual aids enhance engagement metrics, contributing to a healthier platform. Insights also guide user recognition initiatives, fostering a more inclusive community.



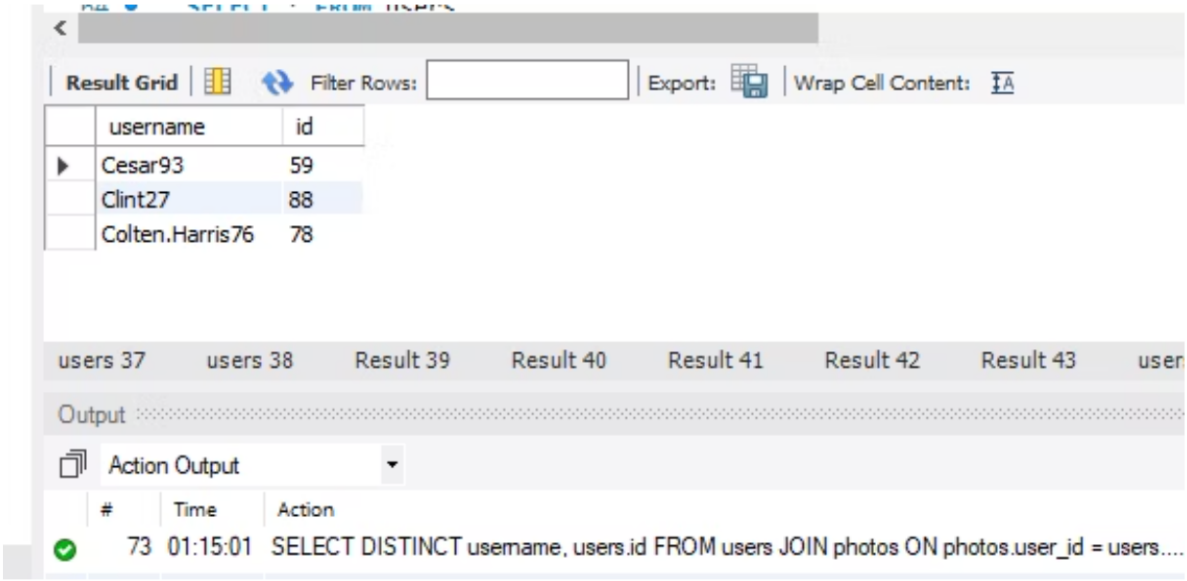
Q.12) Can you help me find the users whose name starts with c and ends with any number and have posted the photos as well as liked the photos?

Code:

```
SELECT DISTINCT username, users.id FROM users JOIN photos ON photos.user_id = users.id JOIN likes ON likes.photo_id = photos.id WHERE username REGEXP '^c' AND username REGEXP '[0-9]$';
```

Insights:

Identify users with names starting with 'c' and ending with a number who have both posted photos and liked others' photos. This query helps pinpoint engaged users with specific username patterns, providing insights for targeted engagement strategies or user recognition program.



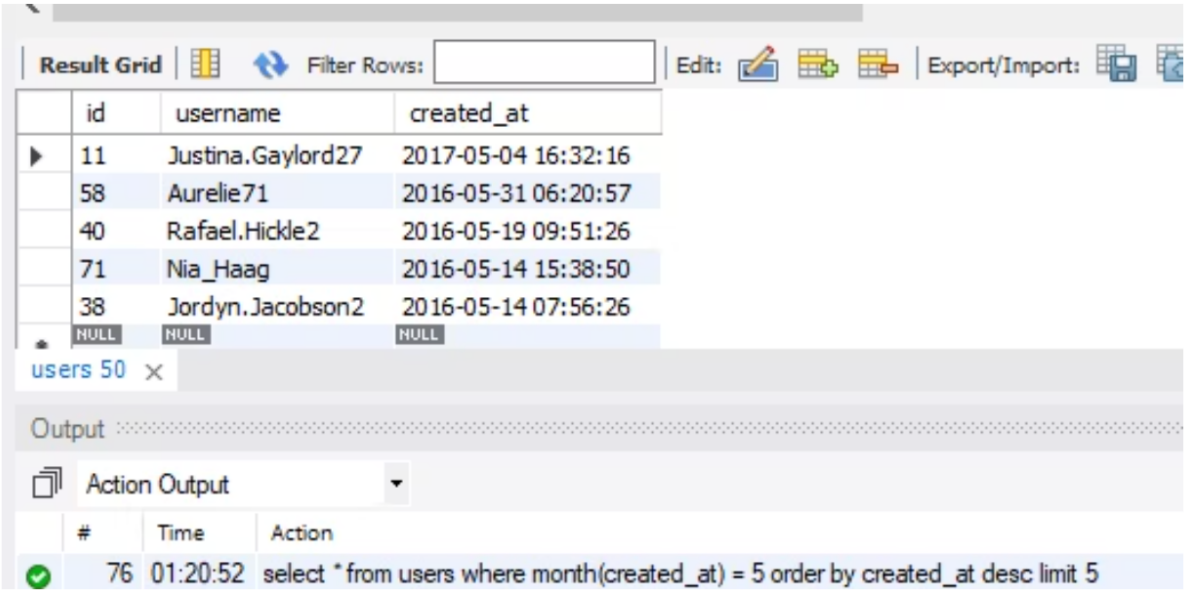
Q.13) Find the users who have created instagramid in may and select top 5 newest joinees from it?

Code:

```
select * from users where month(created_at) = 5 order by created_at desc limit 5;
```

Insights:
Identify users who joined in May, emphasizing the top 5 newest members. Showcase growth trends, highlight platform

appeal in specific months, and consider targeted onboarding or engagement strategies for May joiners.



THANK YOU