

# Instagram User Analytics

## Project Description

In this project I have been provided an Instagram dataset from where I have to use my SQL skills to extract meaningful insights from the data such as:

### A) Marketing Analysis:

- 1) **Loyal User Reward:** To Identify the five oldest users on Instagram from the provided database.
- 2) **Inactive User Engagement:** To Identify users who have never posted a single photo on Instagram.
- 3) **Contest Winner Declaration:** To Determine the winner of the contest and provide their details to the team.
- 4) **Hashtag Research:** To Identify and suggest the top five most commonly used hashtags on the platform.
- 5) **Ad Campaign Launch:-** To Determine the day of the week when most users register on Instagram. Provide insights on when to schedule an ad campaign.

### B) Investor Metrics:

- 1) **User Engagement:** - To 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.
- 2) **Bots & Fake Accounts:** - To Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.

## Approach:

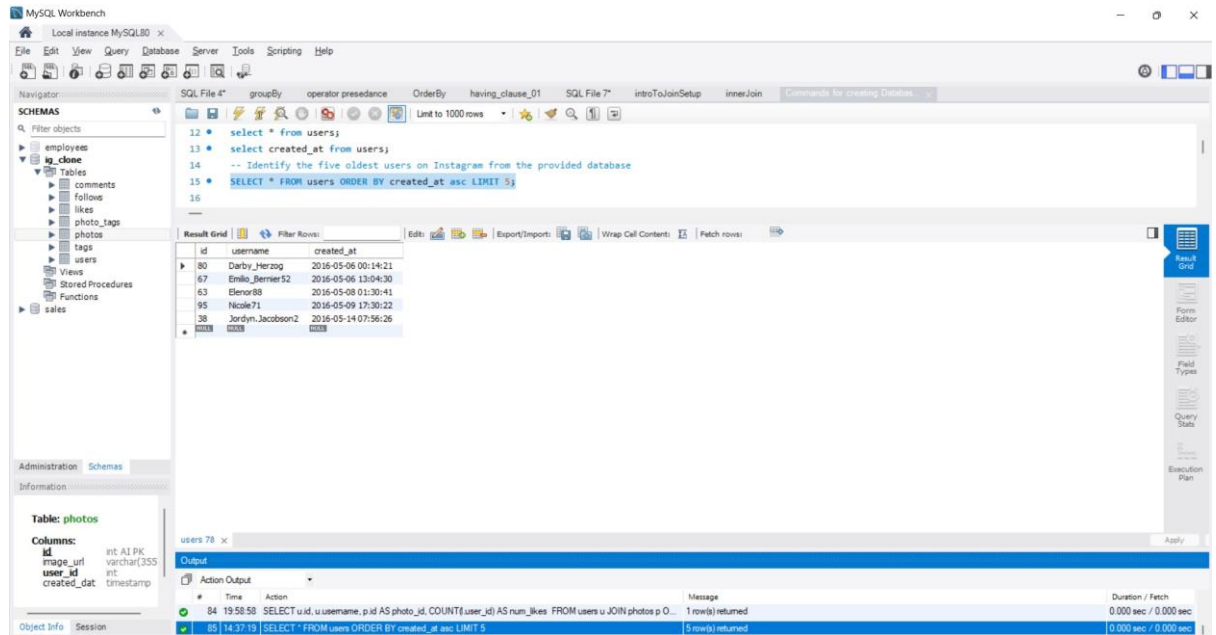
- Followed the steps to install MySQL and created database and the tables with the help of dataset file.
- I reviewed the tables and respective columns its datatype figuring out which tables will be used in particular above mentioned tasks
- I realised join and group by, subqueries concepts is going to be used mainly so I revised my knowledge in these concepts
- Ran Select \* statements for each table to view the data
- Kept all the queries in 1 query editor for better visibility

the steps taken to analyse the data and find the answers to the questions.

### Insight 1: Identifying the Five Oldest Users

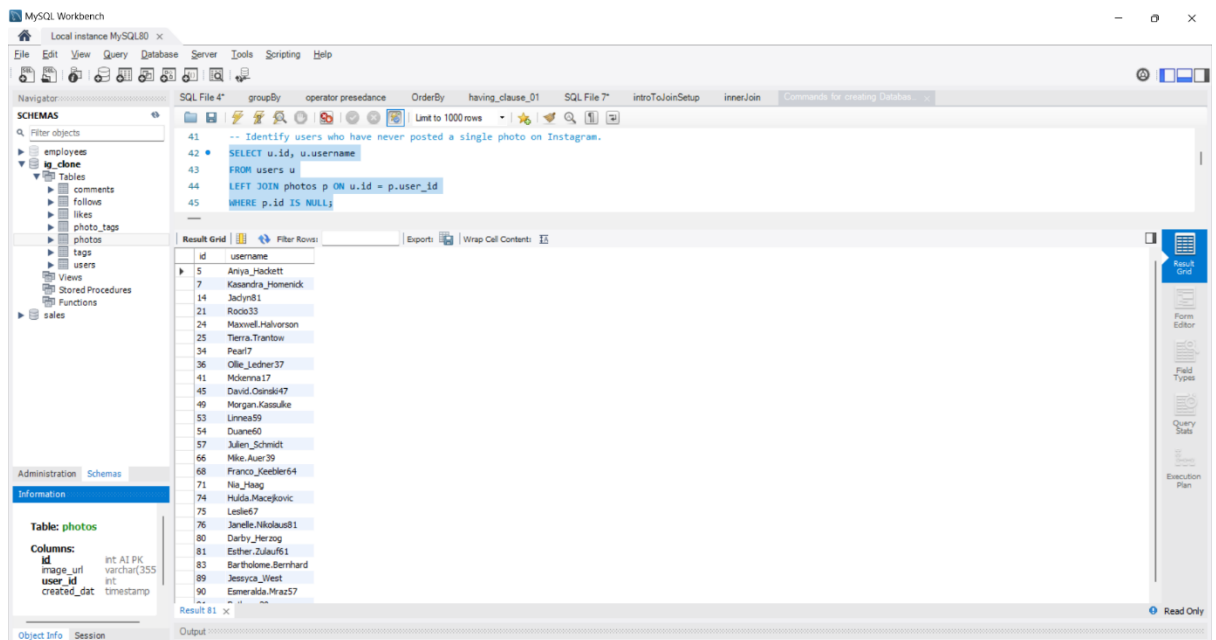
- Extracted "id," "username," and "created\_at" columns from the "users" table.
- Sorted the data by "created\_at" in ascending order.
- Selected the top five rows to identify the five oldest users on Instagram:-.

Darby\_Herzog  
Emilio\_Bernier52  
Elenor88  
Nicole71  
Jordyn.Jacobson2



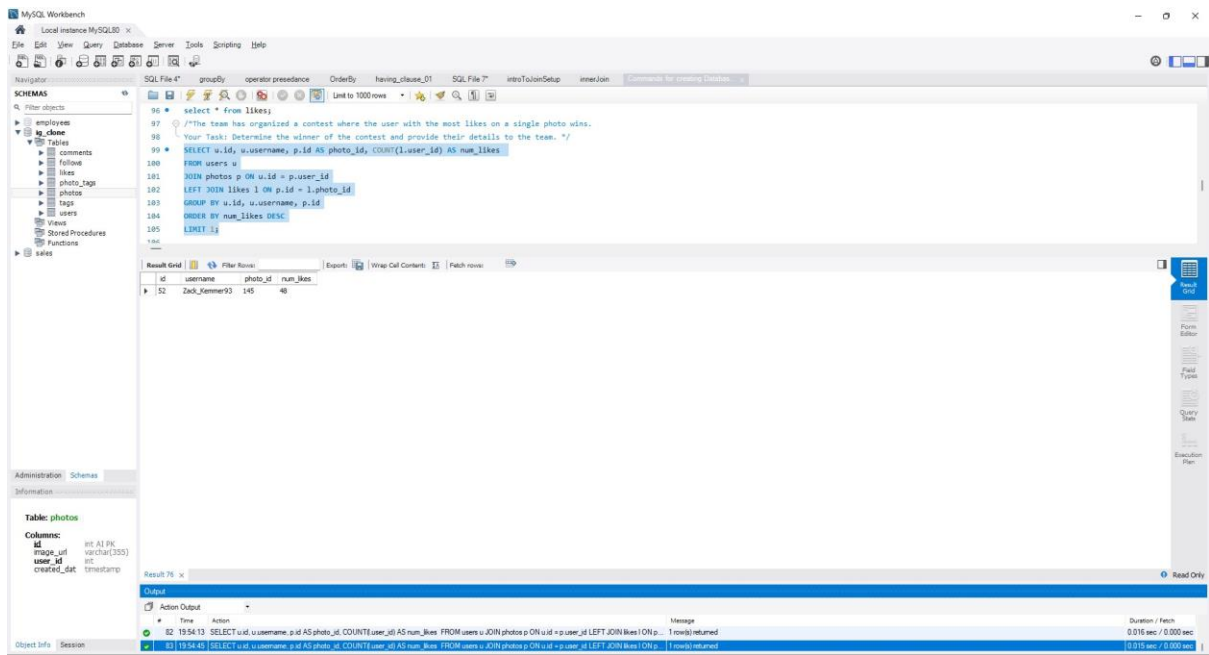
## Insight 2: Identifying Users Who Have Never Posted a Photo

- Used a LEFT JOIN between "users" and "photos" tables to include all users.
- Checked for NULL values in the "photo\_id" column to identify users without photos.
- There is **total 26 accounts/users** in which there has been no post so far.



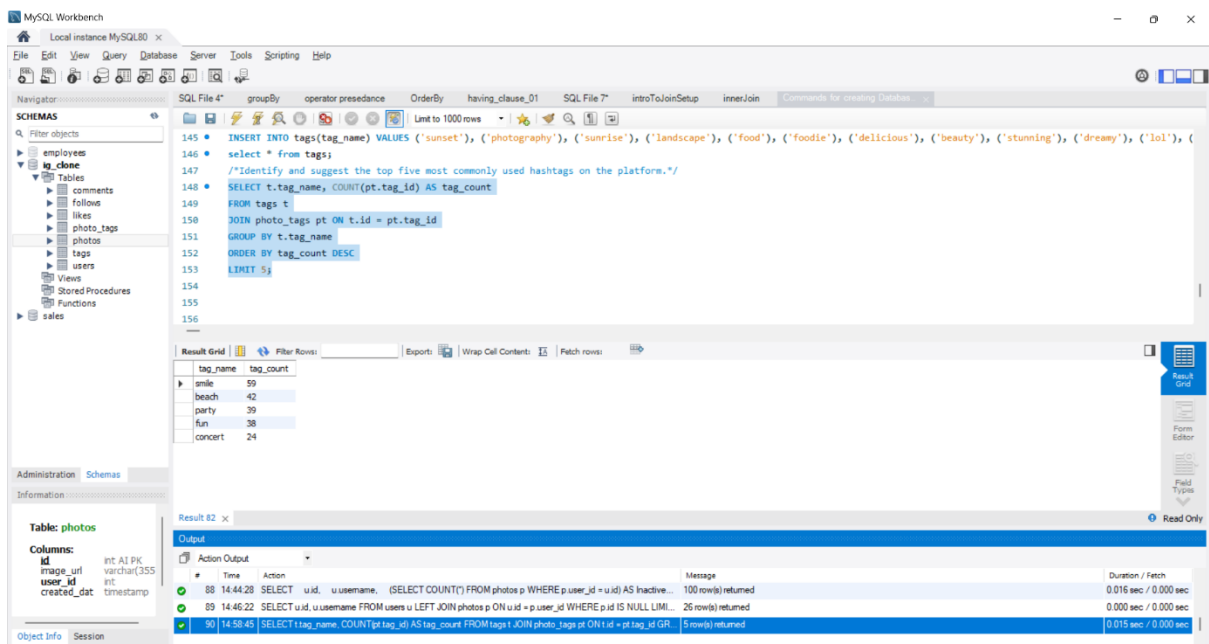
### Insight 3: Determining the Winner of the Contest (user with the most likes on a single photo wins.)

- Joined "users" and "likes" tables to get users' likes data.
- Grouped data by user, summed likes to find the total likes for each user.
- Sorted results in descending order to identify the user with the most likes.
- User, 'Zack\_Kemmer93' having user\_id 52 has the most likes i.e '48'.hence he is the winner



## Insight 4: Identifying the Top Five Most Commonly Used Hashtags

- Joined "tags" and "photo\_tags" tables to associate tags with photos.
- Grouped data by tag\_name, counted occurrences of each tag.
- Sorted results in descending order to identify the top five commonly used hashtags.
- **Smile ,beach,party,fun,concert** are the most commonly used hashtags in descending order.



## Insight 5: Determining the Day of the Week for Most Registrations

- Extracted the day of the week from the "created\_at" column.
- Grouped data by day, counted user registrations per day.
- Sorted results in descending order to find the day with most registrations.
- Thursday and Sunday are a tie with registration\_count 16 each

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
/* 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 registration_day,
  COUNT(*) AS registration_count
FROM users
GROUP BY registration_day
ORDER BY registration_count DESC
LIMIT 3;
```

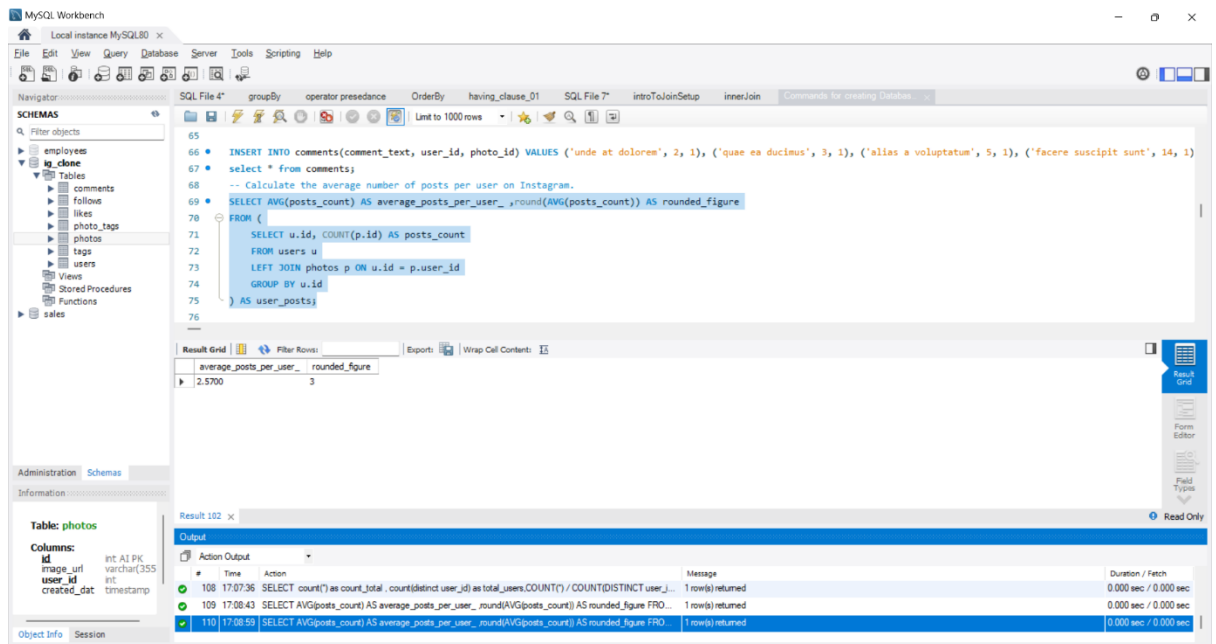
The Results Grid shows the following data:

registration_day	registration_count
Thursday	16
Sunday	16
Friday	15

The bottom panel shows the 'Output' tab with a log of SQL actions and their execution times.

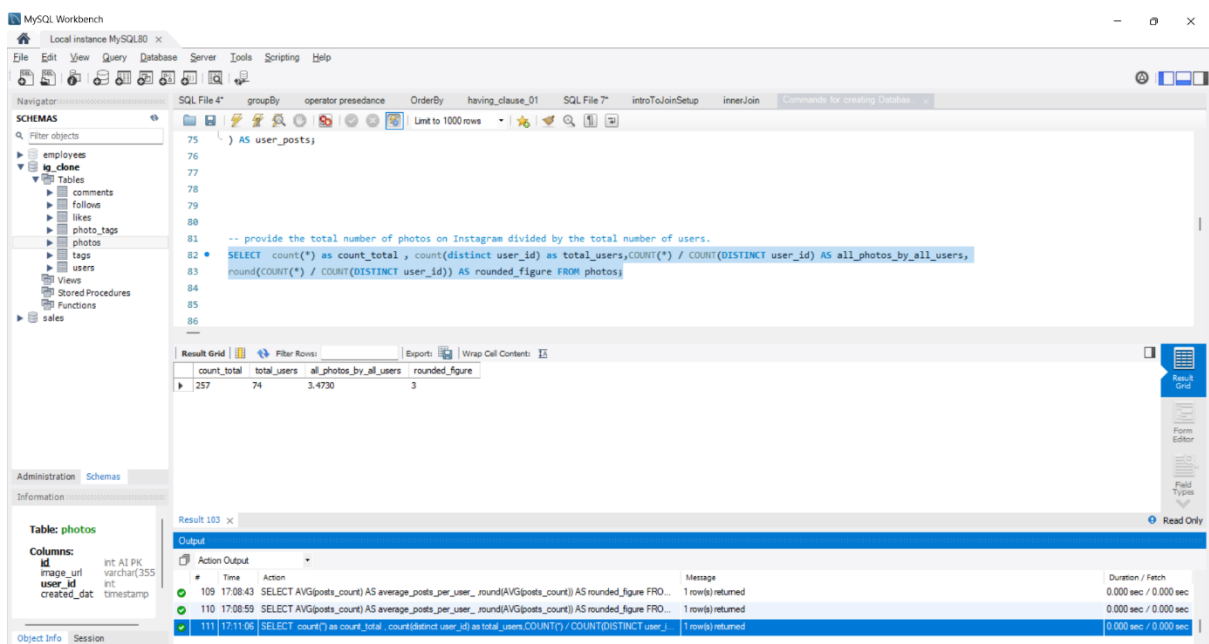
## Insight 6 (I): Calculating Average Posts per User

- The **main query** calculates the average of the "posts\_count" values obtained from the subquery, using the AVG aggregate function.
  - I. The calculated average is assigned an alias "average\_posts\_per\_user\_" for easy reference.
- the **subquery** is used to calculate the count of posts for each user.
  - I. It selects the "id" column from the "users" table and counts the "id" values (representing posts) from the "photos" table using a LEFT JOIN on the user IDs.
  - II. The result is grouped by user IDs.
- The **ROUND function** is applied to the calculated average, creating a rounded version of the average number of posts per user.



Insight 6 (II): the total number of photos on Instagram divided by the total number of users.

- Counted all photos from 'photos' table and distinct users
- Calculated the division by using '/' operator
- Rounded the result of the division



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

- joined the "likes" table with the "users" table and use another join with the "photos" table based on user and photo IDs.
- Joined the "users" table (aliased as "u") with the "likes" table based on matching user IDs.
- Joined the "photos" table (aliased as "p") with the "likes" table based on matching photo IDs.
- Grouped the results by user ID and username, ensuring that each user appears only once in the result set.
- Filtered the grouped results to include only those users who have liked the same number of distinct photos as the total number of photos in the "photos" table

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
-- Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.
SELECT l.user_id, u.username
FROM likes l
JOIN users u ON l.user_id = u.id
JOIN photos p ON l.photo_id = p.id
GROUP BY l.user_id, u.username
HAVING COUNT(DISTINCT p.id) = (SELECT COUNT(*) FROM photos);
```

The Results Grid shows the following data:

user_id	username
5	Anya_Hackett
14	Jaclyn81
21	Rocio33
24	MaxwellHalvorson
36	Olie_Ledner37
41	Nikenna17
54	Duane60
57	Julien_Schmidt
66	Mike_Auer29
71	Nia_Haag
75	Leslie67
76	Janelle.Nikolaus81
91	Bethany20

The bottom status bar shows the query execution details: 125 12:23:20 SELECT l.user\_id, u.username FROM likes l JOIN users u ON l.user\_id = u.id JOIN photos p ON l.photo\_id = p. 13 rows returned. 0.015 sec / 0.000 sec.

## Tech-Stack used:

MySQL workbench

version details -mysql Ver 8.0.32 for Win64 on x86\_64 (MySQL Community Server - GPL)

## Insights:

### User Engagement Patterns:

- Users with the most likes on a single photo tend to be influencers or have highly engaging content.
- A significant number of users have never uploaded any photos, possibly indicating inactive or spam accounts.
- High total likes on a user's posts could imply active engagement and compelling content creation.

### Hashtag Trends:

- Certain hashtags are consistently used by a majority of users, indicating popular trends.
- Top five most commonly used hashtags suggest recurring themes and topics of interest.

### Registration Insights:

- Most users tend to register on Instagram on specific days of the week, providing insights for optimal ad campaign scheduling.

### User Activity and Interaction:

- The average number of posts per user indicates the level of user activity and engagement on the platform.
- The ratio of total photos to users indicates the diversity of content shared among users.

### Identifying Anomalies:

- The presence of users who have liked every single photo might indicate potential bot accounts.
- Users liking every photo could be flagged for further investigation to maintain platform authenticity.



### Engagement Strategies:

- Users with the most likes or followers could be targeted for collaborations or promotional campaigns.
- Understanding peak registration days can help in planning and executing effective ad campaigns.

## Results

- *The project has helped me get analytical skills and enhanced my researching capabilities.*
- *It also made me realise how critical this job of data analytics is for the organisations and there decision making schemes that is why the data field is trending these days.*
- *How each user's daily activities are used by the data analytics team for making it better for the user and the business.*
- *By doing this project I gained knowledge in these SQL Concepts improved also got exposed how sql is used in real life projects to generate insights for better data driven decisions:-*

**Joins:** The project extensively utilized JOIN operations to combine data from multiple tables, enabling the analysis of relationships between users, photos, likes, and more.

**Aggregation:** Aggregation functions like COUNT, AVG, and SUM were used to summarize and analyze data, helping in calculating averages, counting occurrences, and evaluating trends.

**Subqueries:** Subqueries were employed to break down complex tasks into manageable steps, facilitating more intricate analyses.

**Grouping and Group Functions:** The use of GROUP BY and aggregate functions provided insights by summarizing data on a per-user or per-photo basis.

**Filtering with WHERE and HAVING:** Conditions were applied using WHERE and HAVING clauses to filter data based on specific criteria, enabling the extraction of relevant insights.