Instagram User Analytics

Using sql fundamentals

Yashi Gupta

Project Description

This project focuses on analyzing Instagram user data to provide insights that will help product and marketing teams improve platform performance and user engagement. Key tasks include measuring user activity patterns, identifying the most active users, and understanding trends around photos, likes, and hashtags. The project also focuses on identifying users with negative patterns and identifying suspicious behaviors such as bots or fake accounts and protect the real users and the data from such. By querying the data using SQL, the goal is to gain better insights that can improve decision-making for Instagram marketing, products, and investment teams.

Approach

To approach this project, I followed a structured methodology:

1. **Data Exploration**: First, I studied the provided database and identified the key tables: users, photos, comments, likes, follows, and tags. From those tables, I identified primary keys, foreign keys, and composite keys for better understanding.
2. **SQL Queries**: I executed SQL queries for the following tasks:
   A) Marketing Analysis
   B) Investor Metrics
3. **Concepts used for querying**: I used aggregation techniques, such as COUNT, GROUP BY, and HAVING, to calculate metrics like the most common days of the week users engage on Instagram and the top-performing users.
4. For every question, I tried thinking of different approaches to improve my concept understanding.

5.For any given question, the main tasks are to:
1] Identify the tables involved
2] The required output
3] The function which will help in finding the required output

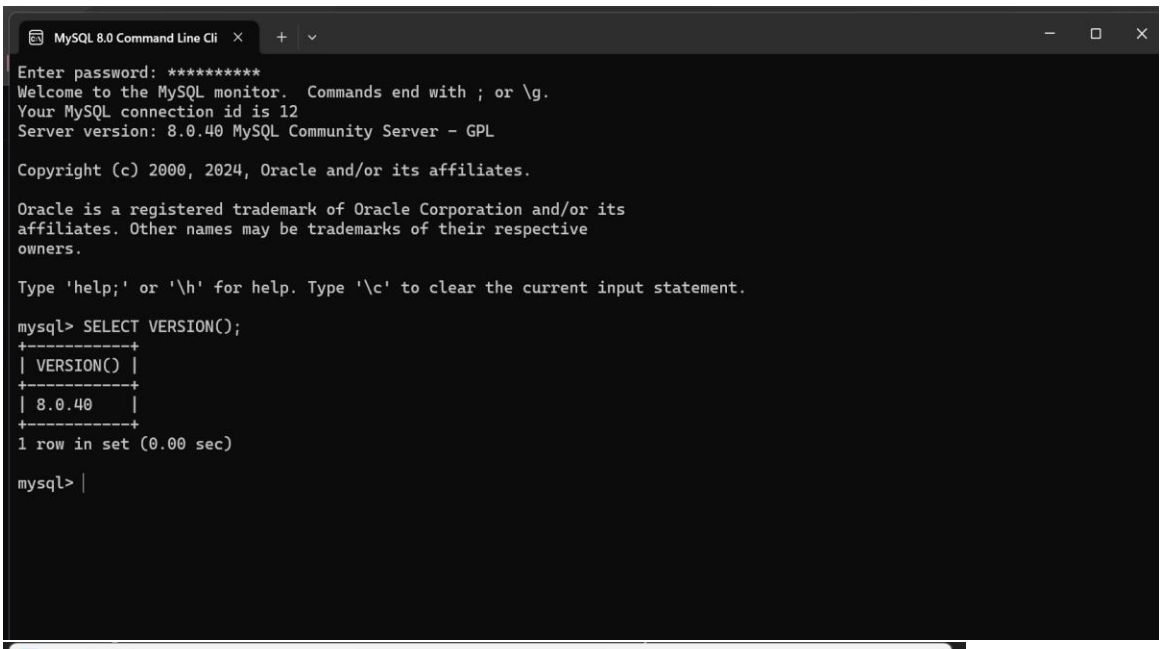
Tech-Stack Used

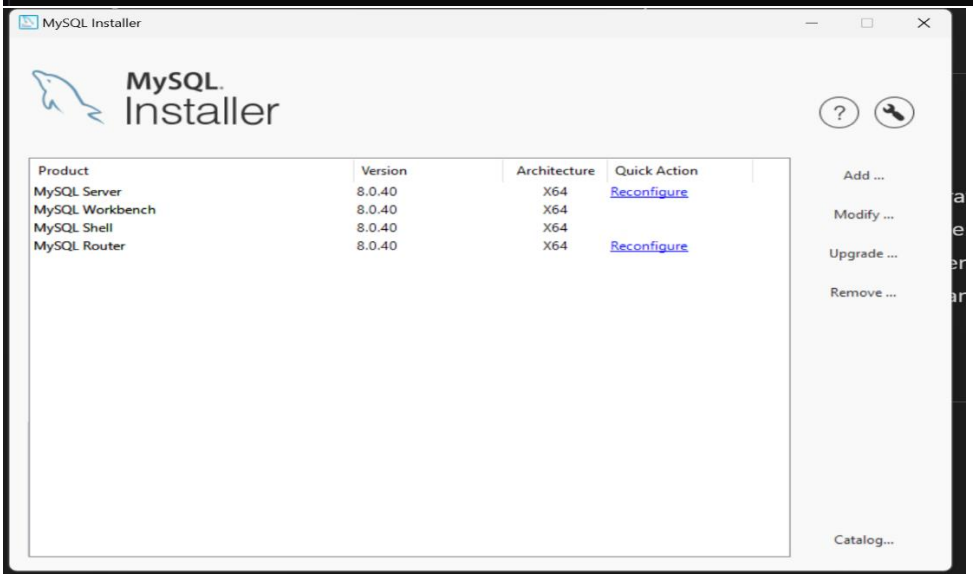
1. MySQL Workbench
2. MySQL
3. Microsoft Word.

MySQL Workbench: This integrated development environment (IDE) was used for SQL development, allowing for efficient database management, query writing.

MySQL: The relational database management system (RDBMS) used for storing and manipulating the data. It was ideal for querying and analyzing large datasets.

Microsoft Word: Used for documenting the project. For versions-





Insights

While working on this project, several insights were derived:

1. **User Activity Patterns**:
   I analyzed the dataset to identify which day of the week had the highest user interaction. This insight can be used to determine the best day for launching ad campaigns to maximize success.

2. **Active and Inactive Users**:
   By examining the number of posts by users, I identified both active and inactive users. Inactive users were defined as those who have not posted even once. This information can help target such users for promotional campaigns.

3. **Popular Photos and Hashtags**:
   I identified the photos and tags with the most likes. The marketing team can use these popular hashtags for better audience reach and engagement.

4. **Suspicious Accounts (Bots)**:
   I identified potential bot accounts by detecting users who liked every single photo on the platform. These suspicious accounts could be flagged to maintain the platform's integrity and protect real users from fake accounts.

5. **Average Engagement and Activity**:
   I calculated the average number of posts per user to assess engagement levels. This analysis helped determine whether user activity is increasing or decreasing over time.

Result and Achievements

This project has not only helped me improve my SQL skills but also helped me use SQL for extracting valuable insights from a dataset, thinking analytically to improve the integrity and user experience of the platform. After the completion of this project, I was able to create complex queries that included joins, group by, having clauses, and using these queries, I was able to analyze this larger dataset.

Overall, this project helped me improve my critical thinking and problem-solving abilities. I have learned new approaches to different problems and which approach to follow for the most efficient answers. These skills are crucial for decision-making for big companies to improve their strategies. This project not only helped improve my technical skills, but I also learned how to analyze a dataset and query it.

At the end of this project, I was able to recognize active and inactive users, determine on which days of the week users were active the most, identify bots, and find the most popular and trending hashtags.

Screenshots and explanation

Primary Keys Summary:

1. **users.id is the primary key of the users table.**
2. **photos.id is the primary key of the photos table.**
3. **comments.id is the primary key of the comments table.**
4. **likes has a composite primary key on (user_id, photo_id).**
5. **follows has a composite primary key on (follower_id, followee_id).**
6. **tags.id is the primary key of the tags table.**
7. **photo_tags has a composite primary key on (photo_id, tag_id).**

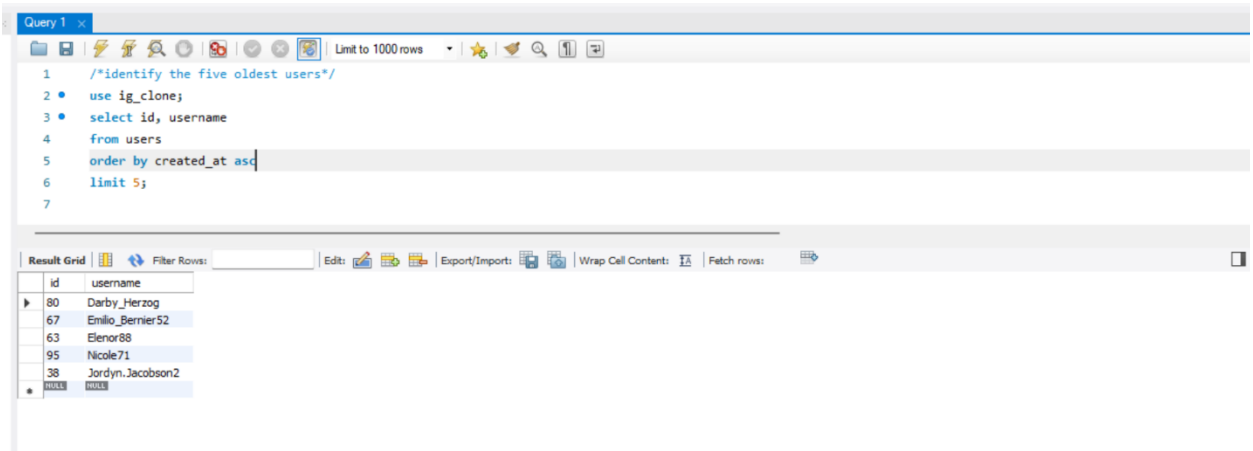
Foreign Keys Summary:

1. **photos.user_id** references **users.id**
2. **comments.user_id** references **users.id**
3. **comments.photo_id** references **photos.id**
4. **likes.user_id** references **users.id**
5. **likes.photo_id** references **photos.id**
6. **follows.follower_id** references **users.id**
7. **follows.followee_id** references **users.id**
8. **photo_tags.photo_id** references **photos.id**
9. **photo_tags.tag_id** references **tags.id**

Tablename.columnname

A) Marketing Analysis:

1. **Loyal User Reward:** The marketing team wants to reward the most loyal users, i.e., those who have been using the platform for the longest time.
   Your Task: Identify the five oldest users on Instagram from the provided database.



Explanation- we have used the column created as to find the 5 oldest users .it has the datatype as timestamp, in which the smallest value will be the entry of the user made first and the largest will be the most recent entry.

To get 5 oldest users we have used limit 5 which will give us the first 5 entries .

2. **Inactive User Engagement:** The team wants to encourage inactive users to start posting by sending them promotional emails.
Your Task: Identify users who have never posted a single photo on Instagram.

```
where p.user_id is null;

select id, username
from users
where id not in (
    select distinct user_id
    from photos
);
```

Result Grid | Filter Rows: | Edit:

id	username
5	Aniya_Hackett
7	Kasandra_Homenick
14	Jaclyn81
21	Rocio33
24	Maxwell.Halvorson
25	Tierra.Trantow
34	Pearl7
36	Ollie_Ledner37
41	Mckenna17
45	David.Osinski47
49	Morgan.Kassulke
53	Linnea59
54	Duane60

LEFT JOIN returns all rows from the left table (users) and the matching rows from the right table (photos).

If no matching row is found in the photos table for a given user (i.e., if the user hasn't posted any photos), the query will still include that user, but the fields from the photos table (such as p.user_id) will be NULL.

where p.user_id is null

This condition is the filter that identifies users who haven't posted any photos.

```
select u.id, u.username
from users u
left join photos p on u.id = p.user_id
where p.user_id is null;
```

Result Grid | Filter Rows: | Export: | Wrap Cel

id	username
5	Aniya_Hackett
7	Kasandra_Homenick
14	Jaclyn81
21	Rocio33
24	Maxwell.Halvorson
25	Tierra.Trantow
34	Pearl7
36	Ollie_Ledner37
41	Mckenna17
45	David.Osinski47
49	Morgan.Kassulke
53	Linnea59
54	Duane60

select distinct user_id from photos;

The subquery selects the user_id values from the photos table, but only those that are distinct (i.e., no duplicates). This effectively creates a list of all users who have posted at least one photo.

where id not in

The outer query will exclude users whose id is found in the list returned by the subquery. As a result, only users who do not have any photos will be selected.

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

CONSIDERING THAT IN THE LIKES TABLE THE user_id of the person who has liked the photo is registered.

In the likes table the photo id which has been appeared most will be the photo with the most likes so from that photo id we can fetch the user to which the photo belongs to:

```
);
/* the photo_id which has occured the most in the likes table*/
select photo_id, count(*) as like_count
from likes
group by photo_id
order by like_count desc
limit 1;
```

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

photo_id	like_count
145	48

Now we will fetch the user to which the photo belongs to:

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

Result Grid | Filter Rows: | Export:

user_id	username
52	Zack_Kemmer93

So the winner of the contest is Zack_Kemmer93 with user_id 52

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

Join the photo_tags table with the tags table .

Count the occurrences of each tag in the photo_tags table.

Sort tags by their frequency in descending order.

Limit the result to the top five most frequently used hashtags

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

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

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

Most used tags are

Smile,beach,party,fun,concert.

5. **Ad Campaign Launch:** The team wants to know the best day of the week to launch ads.
Your Task: Determine the day of the week when most users register on Instagram. Provide insights on when to schedule an ad campaign.

Extract the day of the week from the created_at timestamp in the users table

where 1 = Sunday, 2 = Monday, and so on, up to 7 = Saturday.

Count how many users registered on each day of the week.

Group the results by the day of the week.

Order the results to identify the day with the highest number of registrations.

```
select
    dayofweek(created_at) as registration_day,
    count(*) as registration_count
from users
group by registration_day
order by registration_count desc;
```

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

registration_day	registration_count
5	16
1	16
6	15
3	14
2	14
4	13
7	12

1 refers to Sunday and 5 refers to Thursday.

To launch an ad campaign, we need to determine the day people are most active. We can calculate this using photo comments and likes tables.

```
SELECT DAYOFWEEK(created_dat) AS day_of_week,
       COUNT(*) AS occurrences
FROM photos
GROUP BY day_of_week
ORDER BY occurrences DESC
LIMIT 1;
```

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

day_of_week	occurrences
5	257

```
select * from comments;
SELECT
    DAYOFWEEK(created_at) AS day_of_week,
    COUNT(*) AS occurrences
FROM comments
GROUP BY day_of_week
ORDER BY occurrences DESC
LIMIT 1;
```

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

day_of_week	occurrences
5	7488

```
LIMIT 1;
select * from likes;
SELECT
    DAYOFWEEK(created_at) AS day_of_week,
    COUNT(*) AS occurrences
FROM likes
GROUP BY day_of_week
ORDER BY occurrences DESC
LIMIT 1;
```

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

day_of_week	occurrences
5	8782

We can see that the 5th day of the week has the most number of registered users and the highest number of interactions. So, Thursday is the best day to launch an ad campaign.

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.

1)We can calculate the total number of photos in the photos table and divide it by the total number of unique users in the users table.(avg)

Setup

```
select
    count(p.id) / count(distinct u.id) as total
from users u
left join photos p on u.id = p.user_id;
```

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

	total
▶	2.5700

```
select
    count(p.id) / count(distinct u.id) as avg_posts_per_user
from users u
left join photos p on u.id = p.user_id;


select
    count(p.id) / count(distinct u.id) as total
from users u
left join photos p on u.id = p.user_id;
```

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

	avg_posts_per_user
▶	2.5700

2. **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.

I can think of 2 ways we can perform this particular task:

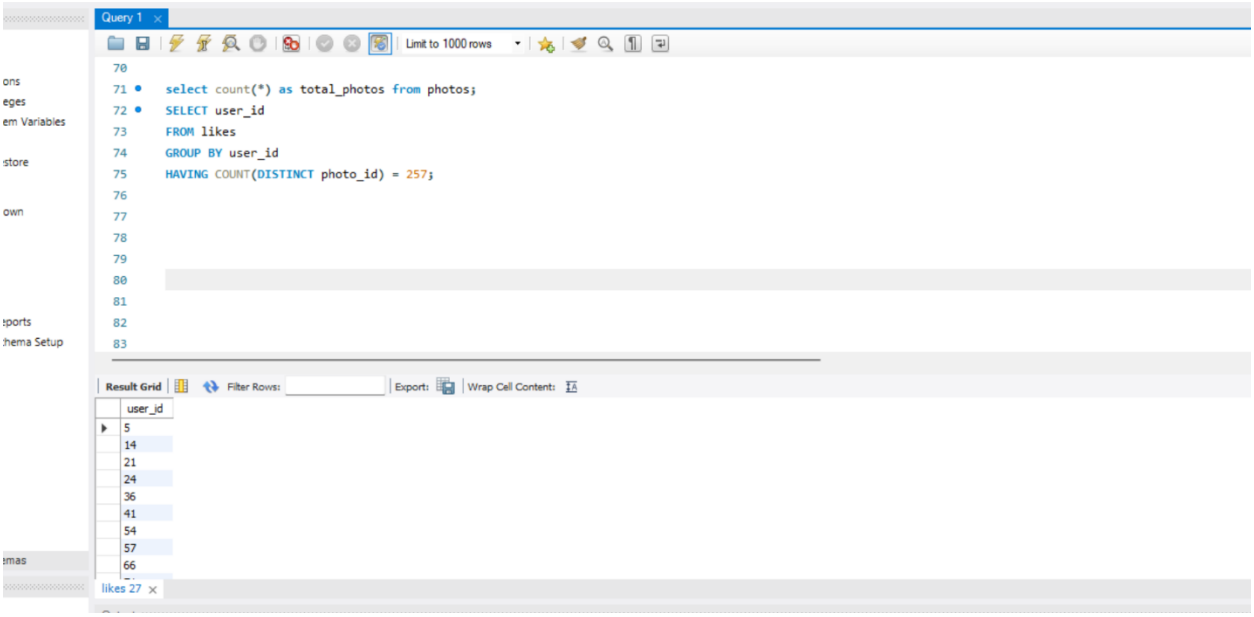
1]

```
select count(*) as total_photos from photos;
```

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

total_photos
257

Count the total no of photos.



Checking which user_id has liked exactly 257 unique photos in the likes table.

This approach calculates how many distinct photos each user has liked. If the user has liked every single photo (257 photos in this case), the query will return them.

2]

```
from users u
left join photos p on u.id = p.user_id;

select count(*) as total_photos from photos;
select l.user_id
from likes l
group by l.user_id
having count(distinct l.photo_id) = (select count(*) from photos);
```

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

user_id
5
14
21
24
36
41
54
57
66

Result 28 | likes 29

Output

It first counts the number of distinct photo_ids liked by each user (COUNT(DISTINCT l.photo_id)).

It then checks if this count is equal to the total number of photos (which is determined by the subquery (SELECT COUNT(*) FROM photos)).

Result Grid

user_id
5
14
21
24
36
41
54
57
66
71
75
76
91

Bots user_id