

PROJECT 2

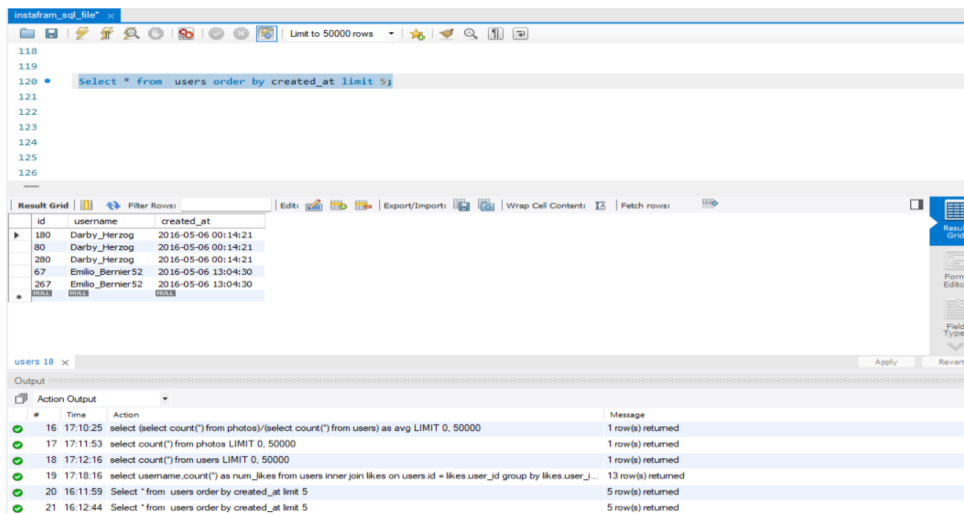
Instagram User Analytics

SQL Tasks:

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.

Answer= **Select * from users order by created_at limit 5;**

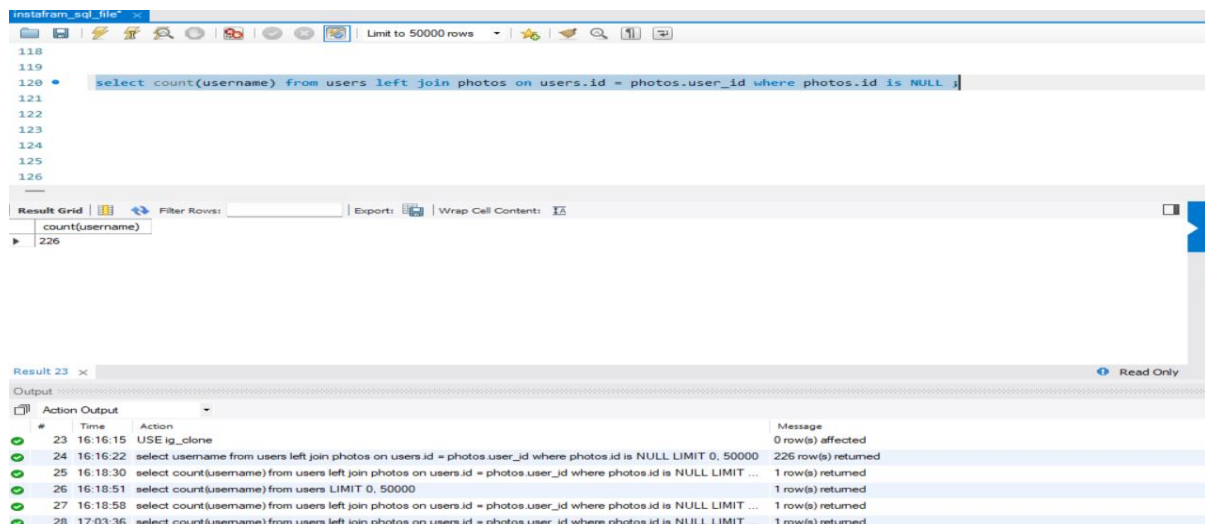


id	username	created_at
180	Darby_Herzog	2016-05-06 00:14:21
80	Darby_Herzog	2016-05-06 00:14:21
280	Darby_Herzog	2016-05-06 00:14:21
67	Emilio_Bernier52	2016-05-06 13:04:30
267	Emilio_Bernier52	2016-05-06 13:04:30

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.

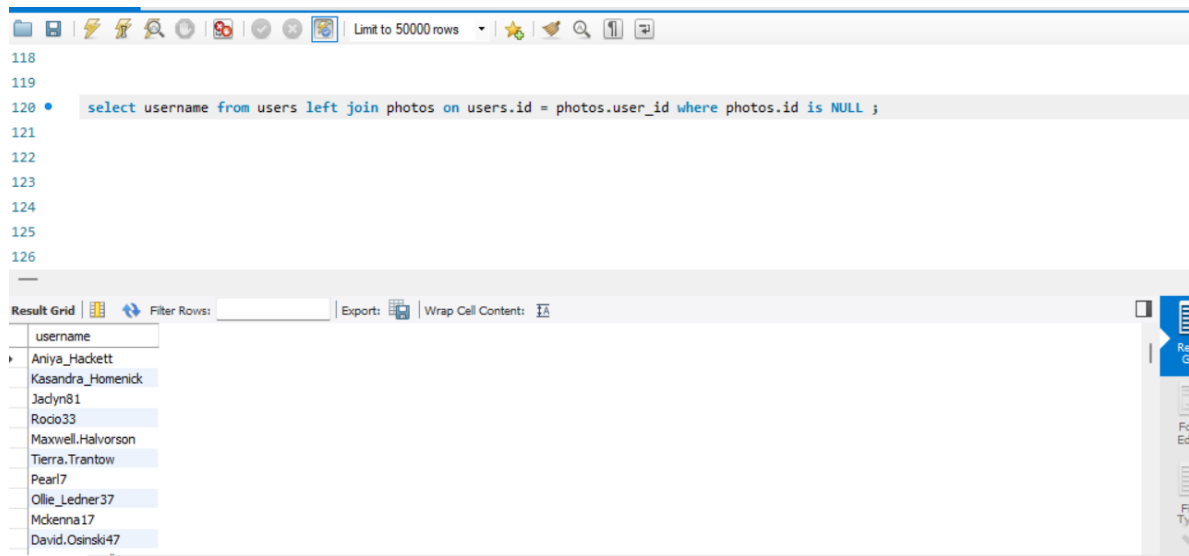
Answer=

select count(username) from users left join photos on users.id = photos.user_id where photos.id is NULL ;



count(username)
226

select username from users left join photos on users.id = photos.user_id where photos.id is NULL ;



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.

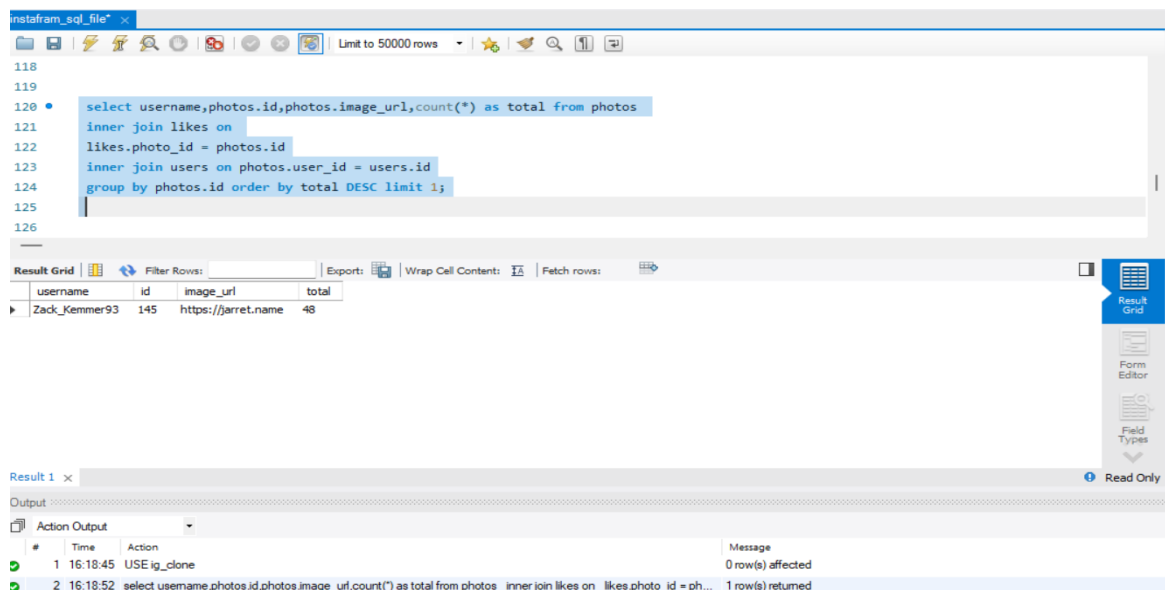
Answer= **select username,photos.id,photos.image_url,count(*) as total from photos**

inner join likes on

likes.photo_id = photos.id

inner join users on photos.user_id = users.id

group by photos.id order by total DESC limit 1;



4.Hashtag Research: A partner brand wants to know the most popular hashtags to use in their posts to reach the most people.

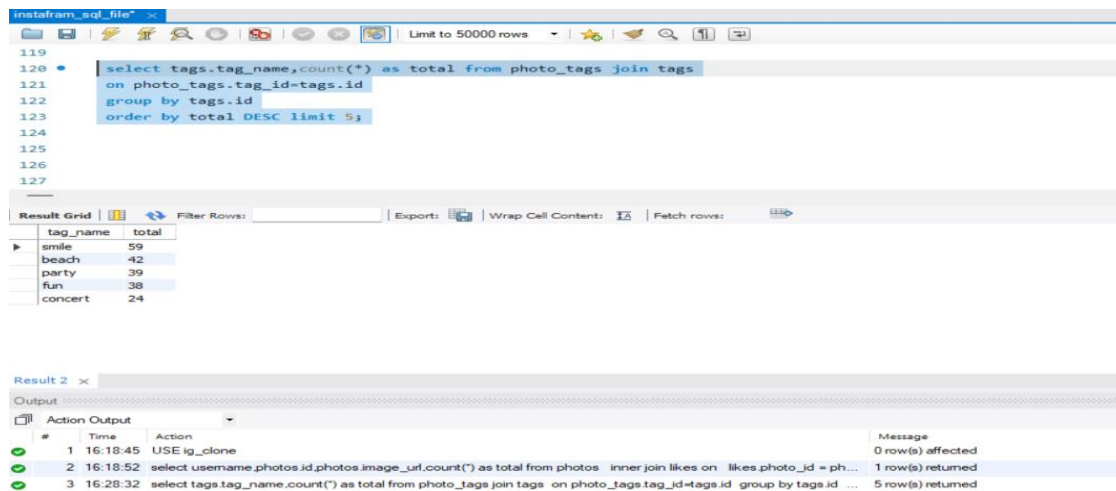
Your Task: Identify and suggest the top five most commonly used hashtags on the platform.

Answer= **select tags.tag_name,count(*) as total from photo_tags join tags**

on photo_tags.tag_id=tags.id

group by tags.id

order by total DESC limit 5;



The screenshot shows a SQL IDE window titled 'instagram_sql_file'. The query editor contains the following SQL code:

```
119
120 • select tags.tag_name,count(*) as total from photo_tags join tags
121   on photo_tags.tag_id=tags.id
122   group by tags.id
123   order by total DESC limit 5;
124
125
126
127
```

Below the query editor, the 'Result Grid' shows the following data:

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

Below the result grid, the 'Output' tab shows the execution log:

#	Time	Action	Message
1	16:18:45	USE ig_clone	0 row(s) affected
2	16:18:52	select username,photos.id,photos.image_url,count(*) as total from photos inner join likes on likes.photo_id = ph...	1 row(s) returned
3	16:28:32	select tags.tag_name,count(*) as total from photo_tags join tags on photo_tags.tag_id=tags.id group by tags.id ...	5 row(s) returned

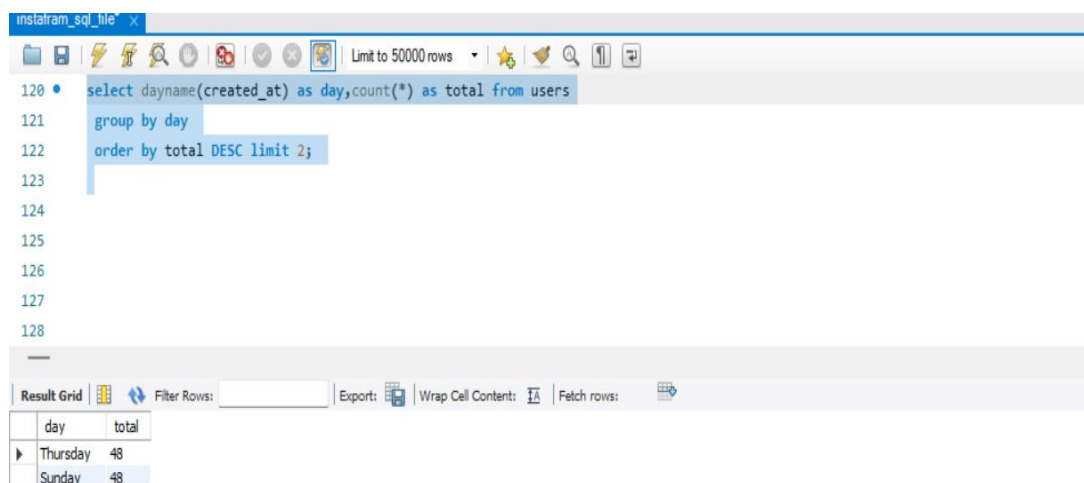
5.Ad Campaign Launch: The team wants to know the best day of the week to launch ads.

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

Answer= **select dayname(created_at) as day,count(*) as total from users**

group by day

order by total DESC limit 2;



The screenshot shows a SQL IDE window titled 'instagram_sql_file'. The query editor contains the following SQL code:

```
120 • select dayname(created_at) as day,count(*) as total from users
121   group by day
122   order by total DESC limit 2;
123
124
125
126
127
128
```

Below the query editor, the 'Result Grid' shows the following data:

day	total
Thursday	48
Sunday	48

B) Investor 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.

Answer= **select (select count(*) from photos)/(select count(*) from users) as avg;**

The screenshot shows a SQL IDE interface. The top pane displays the SQL query: `select (select count(*) from photos)/(select count(*) from users) as avg;`. The bottom pane shows the 'Result Grid' with a single row containing the value `0.8567`. Below the result grid, the 'Action Output' pane shows a log of database actions, including the execution of the query and the resulting message: '1 row(s) returned'.

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

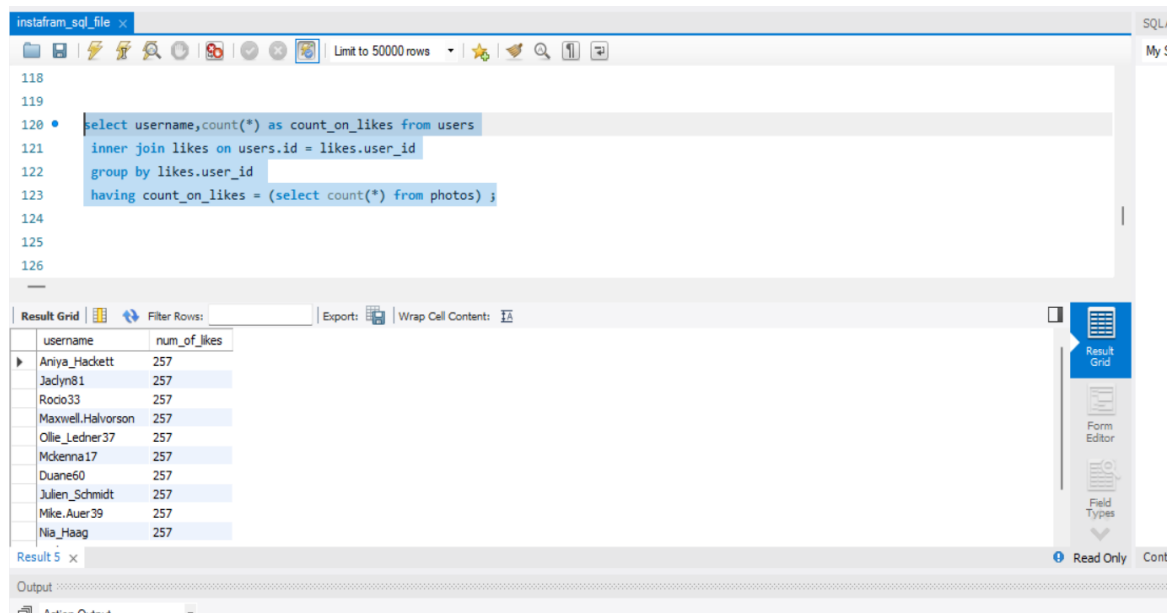
Your Task: Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.

Answer= **select username,count(*) as count_on_likes from users**

inner join likes on users.id = likes.user_id

group by likes.user_id

having count_on_likes = (select count(*) from photos) ;



Approach:

The data set was given in the project . First I have created a data base and use that data base and then created a tables in it . And then inserted the values in the tables using MYSQL Workbench. I have created the table ,inserted the values and altered the table when needed. And them worked on the given queries. I used my sql skills to solve the problems. I worked on joins , i.e inner joins, outer joins etc.

Tech Stack:

MYSQL WORKBENCH 8.0 CE

Used:

I used MYSQL WORKBENCH for this project because I am very familiar with this tool, I am working for almost 4 years of now, I feel comfortable in writing the SQL queries. The software is very interactive and very easy to use . The compiler is so easy to run and interactive .

Insights:

In this project I came to learn more about joins , it was very helpful to work on and feel comfortable on joins queries. I worked on real-life Instagram data , Which is helped me to build confidence on the joins queries and also the fundamental queries.I made a beautiful insights from a data . while writing queries.

Results:

The project was the best way to boost up with my sql knowledge. I have worked on joins , I came to know about the data cleaning and data formatting in this . I am able to get the desired output while writing the queries.