Instagram User Analytics

SQL Fundamentals

Project Description:

This Project involves analysing user interactions and engagement with the Instagram app to provide valuable insights that can help the business grow. User analysis involves tracking how users engage with a digital product, such as a software application or a mobile app. The insights derived from this analysis can be used by various teams within the business. For example, the marketing team might use these insights to launch a new campaign, the product team might use them to decide on new features to build, and the development team might use them to improve the overall user experience.

Overall, it will help the product manager and the rest of the team make informed decisions about the future direction of the Instagram app.

Approach:

I have used my SQL skills to extract meaningful insights from the data.

The steps I took to analyse the data and find the answers to the questions provided:

1. **Create and Connect to Database:** Open **MySQL Workbench** and created a database named ig_clone then connected to this database by writing use ig_clone query.

CREATE DATABASE ig_clone;
USE ig_clone;

- 2. **Create Tables and populate It:** Create the schemas of each table including its column, data types and constraints mentioned in dataset. After this insert the values into each table correctly.
- 3. **Formulate queries:** Determined the specific questions I wanted to answer and tried to understand what result this query wants. Formulated SQL queries that retrieve the relevant data to answer those questions. Started with simple queries and gradually build more complex ones as needed.
- 4. **Write Queries:** Used the SQL editor in **MySQL Workbench** to write and execute SQL queries. Ensured that the queries are accurate, efficient, and properly structured to extract the required information from the database.
- 5. **Execute Queries:** Ran SQL queries against the database. Reviewed the query results to ensure they match with question expectations. Again, Refined queries as necessary to get the desired results.

Tech-Stack Used:

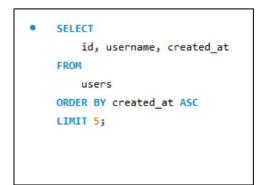
In this project, I have used MySQL Workbench 8.0 as my tool to analyse Instagram user data and answer questions posed by the management team. I have chosen this software for answering MySQL queries for several reasons:

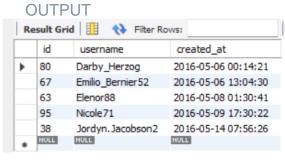
1. **User Friendly Interface:** MySQL Workbench provides a visually appealing and intuitive interface for interacting with MySQL databases. Its graphical user interface (GUI) makes it easy to write and execute queries, view query results, and manage database objects.

- 2. **Query Editor:** MySQL Workbench includes a powerful query editor with syntax highlighting, code completion, and error detection features. This helps users write queries more efficiently and accurately and also beautify the code.
- 3. **Database Design and Modelling:** MySQL Workbench offers tools for designing and modelling databases, including the ability to create and modify database schemas, tables, and relationships visually. This can be helpful for understanding the database structure and planning queries accordingly.
- Insights: While working on this project I gained hands on experience with real life use cases. I gained knowledge on SQL queries and used it to answer some really interesting questions on Instagram users. Solving the queries with subquery or without subquery was a little bit challenging. I learned where using subqueries can be avoided and joins could be used and tried to make it more optimized.
 - Instagram user analytics project provided me some actionable insights to optimize Instagram presence, improve engagement, and better understand user's preferences and behaviour on the platform. It also revealed that on which day of the week user register mostly.
- **Result**: This section provides Question posted, the SQL query and its output snapshots.

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. **SQL Task:** Identify the five oldest users on Instagram from the provided database.





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

```
• SELECT

id, username

FROM

users

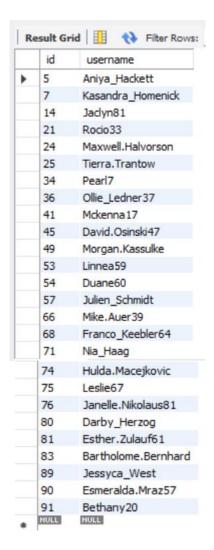
WHERE

⇒ id NOT IN (SELECT DISTINCT

user_id

FROM

photos);
```



OUTPUT

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

```
with photo_likes as (SELECT

photo_id as pic_id, COUNT(photo_id) as no_of_likes

FROM

likes

GROUP BY photo_id

ORDER BY no_of_likes DESC limit 1 )

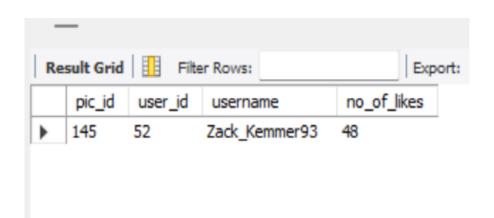
select pic_id , user_id, username ,no_of_likes

from

photo_likes inner join photos

on photo_likes.pic_id=photos.id

inner join users on photos.user_id=users.id;
```



OUTPUT

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

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

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

	Re	Result Grid Filter Rows:				
		tag_id	tag_name	Total_no_of_tags		
•	•	21	smile	59		
		20	beach	42		
		17	party	39		
		13	fun	38		
		18	concert	24		

OUTPUT

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

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

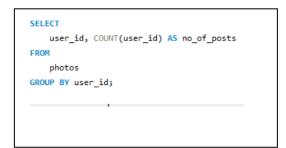
```
SELECT DAYNAME(created_at) AS day_of_week, COUNT(*) AS best_day
FROM users
GROUP BY day_of_week
ORDER BY best_day DESC
LIMIT 1;
```



OUTPUT

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.
 - **SQL 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.



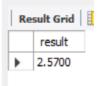
user_id	no_of_posts	user_id	no_of_po
1	5	22	1
2	4	23	12
3	4	26	5
4	3	27	1
6	5	28	4
8	4	29	8
9	4	30	2
10	3	31	1
11	5	32	4
12	4	33	5
13	5	35	2
15	4	37	1
16	4	38	2
17	3	39	1
18	1	40	1
19	2	42	3
20	1	43	5
-			

user_id	no_of_posts		user_id	no_of_posts
44	4		67	3
46	4		69	1
47	5		70	1
48	1		72	5
50	3	•	73	1
51	5		77	6
52	5		78	5
55	1		79	1
56	1		82	2
58	8		84	2
59	10		85	2
60	2		86	9
61	1		87	4
62	2		88	11
63	4		92	3
64	5		93	2
65	5		94	1

95	2
96	3
97	2
98	1
99	3
100	2

OUTPUT

```
SELECT
(SELECT COUNT(id) FROM photos) /
(SELECT COUNT(id) FROM users) AS result;
```

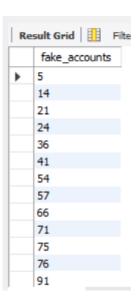


OUTPUT

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

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

```
with fake_accounts as
  (select user_id , count(photo_id) as no_of_pics_liked from likes group by user_id)
select user_id as fake_accounts from fake_accounts
where no_of_pics_liked=(select count(*) from photos);
```



OUTPUT