

Instagram User Analytics

G.Sai Ravi Chandra

4/09/2024

Project Description

The purpose of this project is to analyze the Instagram database and derive insights from the data. The approach involves executing various SQL queries to extract relevant information, identify patterns, and answer specific questions about the users, photos, likes, and tags in the database.

APPROACH

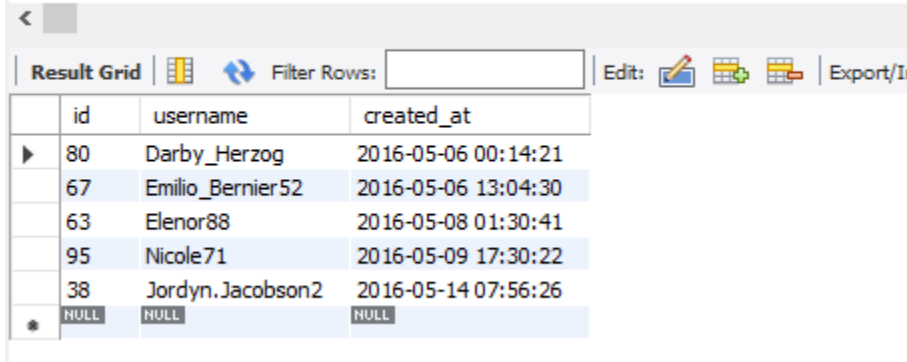
A) Marketing Analysis

1. Loyal User reward:

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

As the data contains the timestamp of the users we can simply use the statement ORDERBY and get the top 5 oldest customers from the dataset.

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



The screenshot shows a database interface with a query editor at the top and a result grid below. The query is `SELECT * FROM users ORDER BY created_at ASC LIMIT 5;`. The result grid displays the following data:

	id	username	created_at
▶	80	Darby_Herzog	2016-05-06 00:14:21
	67	Emilio_Bernier52	2016-05-06 13:04:30
	63	Elenor88	2016-05-08 01:30:41
	95	Nicole71	2016-05-09 17:30:22
	38	Jordyn.Jacobson2	2016-05-14 07:56:26
*	NULL	NULL	NULL

2. Inactive user Engagement:

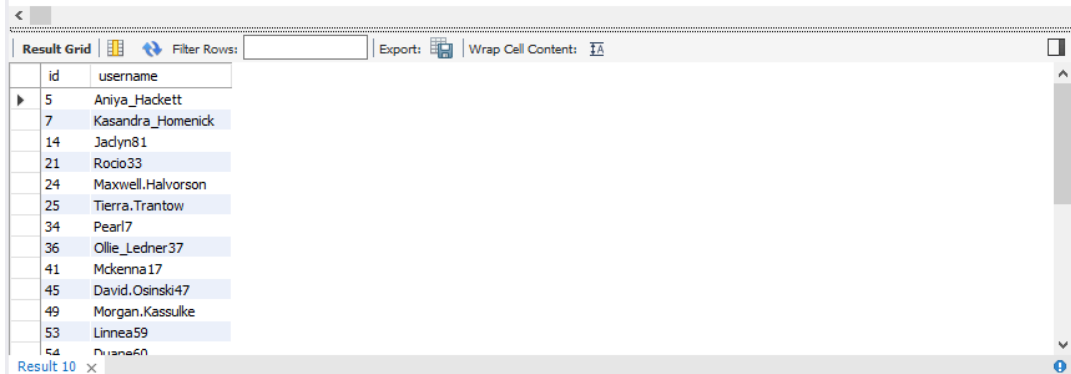
Your Task: Identify users who have never posted a single photo on Instagram.

Here we have first selected ID and username from the users table and LEFT joined it with photos where it will add all the data from users which will match with user_id from photos table and finally display the results where user_id is NULL in that table.

```

97 • SELECT u.id, u.username
98 FROM users u
99 LEFT JOIN photos p ON u.id = p.user_id
100 WHERE p.user_id IS NULL;
101

```



id	username
5	Aniya_Hackett
7	Kassandra_Homenick
14	Jadyn81
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

3. Contest Winner Declaration:

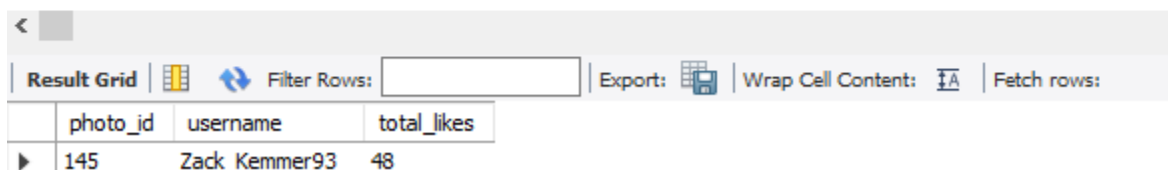
Your Task: Determine the winner of the contest and provide their details to the team.

In this task we have to find a user who has the highest likes on a single picture, so first we have to select photo_id from likes and username from users and also count the total likes by the user_id. Now join likes, photos and user tables based on the respective ID's and finally use group_by function to aggregate the total likes per photo and order by descending to show the highest likes and limit with one person with the highest likes. From the analysis, Zack_Kemmer93 has the highest likes for his photo with the photo_id 145.

```

107 • SELECT l.photo_id,u.username,count(l.user_id) as total_likes
108 from likes l join photos p on l.photo_id=p.id
109 join users u on p.user_id=u.id
110 group by l.photo_id,u.username order by total_likes desc limit 1;
111
112

```



photo_id	username	total_likes
145	Zack_Kemmer93	48

4. Hashtag Research:

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

This is almost the same as the above task, where we join photo_tags and tags table and group it by tag_name to get the aggregate value of each tag and arrange by descending order to get the highest value on the top.

```

113 • SELECT t.tag_name,count(p.photo_id) as total_tags
114 from tags t inner join photo_tags p on t.id=p.tag_id
115 group by t.tag_name order by total_tags desc LIMIT 1;
116
117

```

<	Result Grid	Filter Rows:	Export:	Wrap Cell Content:
	tag_name	total_tags		
▶	smile	59		

5. Ad Campaign Launch:

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

Here we have to use function date_format("%W") to convert timestamp to days and aggregate the days using group by to get total user registering to instagram on a particular day.

```

117 • SELECT DATE_FORMAT(created_at,'%W') as days,count(username)
118 from users
119 group by 1 order by 2 desc;

```

<	Result Grid	Filter Rows:	Export:	Wrap Cell Content:
	days	count(username)		
▶	Thursday	16		
	Sunday	16		
	Friday	15		
	Tuesday	14		
	Monday	14		
	Wednesday	13		
	Saturday	12		

B) Investor Metrics:

1.User Engagement:

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.

Select id from users and count number of photos per id and now join the photos table and user table by taking user Id as common field and group the rows by user id.

Now sum all the pictures as total photos and count user id as total users and divide these two values to get the average post per user.

```

126 WITH base AS (
127     SELECT u.id AS userid,
128            COUNT(p.id) AS photoid
129     FROM users u
130     LEFT JOIN photos p ON p.user_id = u.id
131     GROUP BY u.id
132 )
133 SELECT SUM(photoid) AS total_photos,
134        COUNT(userid) AS total_users,
135        SUM(photoid) / NULLIF(COUNT(userid), 0) AS average_posts_per_user
136 FROM base;
137
138

```

total_photos	total_users	average_posts_per_user
257	100	2.5700

2. Bots and Fake account

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.

For this task we have to find out the number of likes for each user and compare it with total photos to know if the user has liked every photo.

```

138
139 SELECT u.username, COUNT(l.photo_id) AS total_likes
140 FROM likes l
141 INNER JOIN users u ON u.id = l.user_id
142 GROUP BY u.username
143 HAVING COUNT(l.photo_id) = (SELECT COUNT(*) FROM photos)
144 ORDER BY u.username;

```

username	total_likes
Aniya_Hackett	257
Bethany20	257
Duane60	257
Jadyn81	257
Janelle.Nikolaus81	257
Julien_Schmidt	257
Leslie67	257
Maxwell.Halvorson	257
Mckenna17	257
Mike.Auer39	257
Nia_Haag	257
Ollie_Ledner37	257
Rocio33	257

Tech-Stack Used

MySQL Workbench: This software was used to execute the SQL queries and analyze the data. It provides a user-friendly interface for interacting with the database and visualizing the results.

Insights

A) Marketing Analysis:

- **Loyal User Reward:**
 - To identify the five oldest users on Instagram, we can query the `users` table and order the results by the `created_at` column in ascending order, limiting the output to 5 rows. This will provide the usernames and registration dates of the oldest users, who can be considered for the loyal user reward program.
- **Inactive User Engagement:**
 - The query to identify users who have never posted a photo involves a left join between the `users` and `photos` tables, filtering for users where the `user_id` in the `photos` table is `NULL`. This insight will help target inactive users with promotional emails to encourage them to start posting content.
- **Contest Winner Declaration:**
 - To determine the winner of the contest, we can join the `likes`, `photos`, and `users` tables to find the photo with the most likes. Grouping by `photo_id` and ordering by the `total_likes` count in descending order, while limiting to 1 row, will provide the details of the contest winner.
- **Hashtag Research:**
 - To identify the top five most commonly used hashtags, we can join the `tags` and `photo_tags` tables, group by `tag_name`, order by the `total_tags` count in descending order, and limit the output to 5 rows. This will provide the most popular hashtags for the partner brand to use in their posts.
- **Ad Campaign Launch:**
 - To determine the best day of the week to launch ads, we can query the `users` table, extract the day of the week from the `created_at` column using `DATE_FORMAT`, group by the day of the week, order by the `total_registrations` count in descending order, and limit to 1 row. This insight will help schedule the ad campaign on the day with the highest user registration activity.

B) Investor Metrics:

- **User Engagement:**
 - To calculate the average number of posts per user, we can perform a left join between the `users` and `photos` tables, group by `user_id`, and calculate the

average of the `photo_count` for each user. To get the total number of photos divided by the total number of users, we can use a subquery to count the total photos and divide it by the count of distinct `user_id` values from the `users` table. These metrics will provide insights into user engagement and activity levels on the platform.

- **Bots & Fake Accounts:**

- To identify potential bot accounts that have liked every single photo, we can join the `likes` and `users` tables, group by `username`, and use the `HAVING` clause to filter for users whose `total_likes` count equals the total number of photos. This query will highlight accounts that have interacted with every available photo, which is unlikely for a normal user and may indicate bot or fake accounts.

Impact of the Analysis

The analysis conducted during this project has significant implications for both the marketing team and investors. By providing detailed insights into user behavior, engagement patterns, and content trends, the project equips stakeholders with the information needed to make informed decisions.

- **For the Marketing Team:** The insights gained can drive targeted campaigns, enhance user engagement, and optimize content strategies, ultimately leading to improved user retention and satisfaction.
- **For Investors:** The metrics on user engagement and the identification of potential issues with fake accounts provide transparency and assurance regarding the platform's health and growth potential.

Overall, this project has not only enhanced my SQL skills and data analysis capabilities but has also contributed to a deeper understanding of user dynamics within the platform. The actionable insights derived from the analysis will serve as a foundation for future marketing strategies and user engagement initiatives, ensuring continued growth and success for the platform.