Project Description: This project is about performing user analysis on the Instagram platform using SQL. The goal is to derive business insights for the marketing, product, and development teams by tracking how users engage and interact with the platform. The project involves analyzing user behavior data from different tables in the ig_clone database such as comments, follows, likes, photo_tags, photos, tags, and users. The insights gained will help the product team in launching new marketing campaigns, deciding on features to build for the app, tracking the success of the app, and improving the user experience.

Approach: To handle the project, I first analyzed the questions asked by the management team and identified the specific tasks that needed to be performed. Then, I used SQL queries to extract the relevant data from the ig_clone database and performed the necessary calculations to answer the questions. Finally, I presented the findings in a report format for the leadership team.

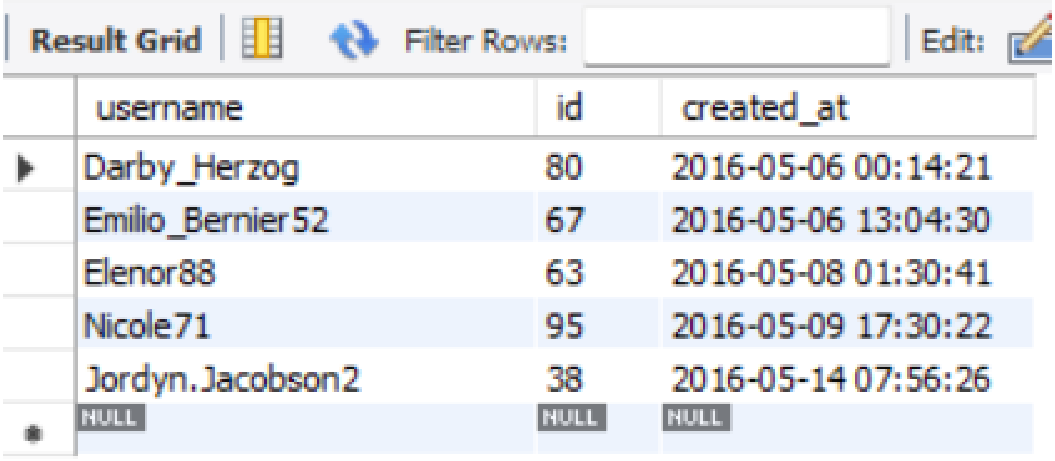
Tech-Stack Used: I used MySQL workbench version 8.0 to perform the SQL queries on the ig_clone database. I used this software because it is a popular and widely used relational database management system that is easy to use and has powerful features for data analysis.

Insights:

A) Marketing: The marketing team wants to launch some campaigns, and the help that I provided with the following :

1. **Rewarding Most Loyal Users:** People who have been using the platform for the longest time.

   The 5 oldest users of Instagram from the database provided are:



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

This was achieved using the following SQL query:

SELECT username, id, created_at

FROM users

ORDER BY created_at LIMIT 5;

2. **Remind Inactive Users to Start Posting:** By sending them promotional emails to post their 1st photo.
   The users who have never posted a single photo on Instagram are:

Username:

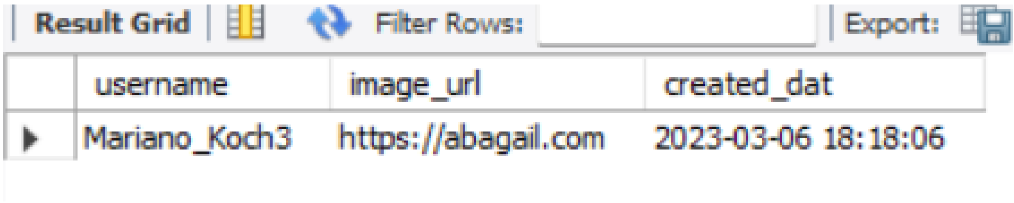
Aniya_Hackett
Kasandra_Homenick
Jaclyn81
Rocio33
Maxwell.Halvorson
Tierra.Trantow
Pearl7
Ollie_Ledner37
Mckenna17
David.Osinski47
Morgan.Kassulke
Linnea59
Duane60
Julien_Schmidt
Mike.Auer39
Franco_Keebler64
Nia_Haag
Hulda.Macejkovic
Leslie67
Janelle.Nikolaus81
Darby_Herzog
Esther.Zulauf61
Bartholome.Bernhard
Jessyca_West
Esmeralda.Mraz57
Bethany20

This was achieved using the following SQL query:

```
SELECT u.username
FROM users as u
WHERE u.id NOT IN (SELECT p.user_id FROM photos as p);
```

3. **Declaring Contest Winner:** The team started a contest and the user who gets the most likes on a single photo will win the contest now they wish to declare the winner.

   The winner of the contest is :



username	image_url	created_dat
Mariano_Koch3	https://abagail.com	2023-03-06 18:18:06

   The winner of the contest is the user with the most likes on a single photo. The winner's details can be identified by querying the photos table for the user_id with the maximum number of likes on a single photo.

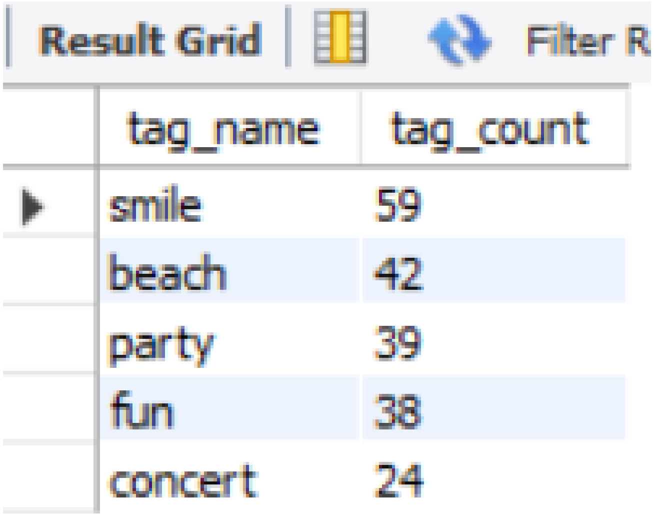
This was achieved using the following SQL query:

```
USE ig_clone;


SELECT u.username, p.image_url, p.created_dat
FROM users u
JOIN photos p ON u.id = p.id
WHERE p.id = (
  SELECT p.id
  FROM photos as p
  LEFT JOIN likes ON p.user_id = likes.user_id
  GROUP BY p.id
  ORDER BY COUNT(likes.photo_id) DESC
  LIMIT 1
);
```

4. **Hashtag Researching:** A partner brand wants to know, which hashtags to use in the post to reach the most people on the platform.

   The top 5 most commonly used hashtags on the platform are:



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

This was achieved using the following SQL query:

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

Explanation of the query:

We select the tag_name column from the tags table and count the number of occurrences of each tag by joining it with the photo_tags table.

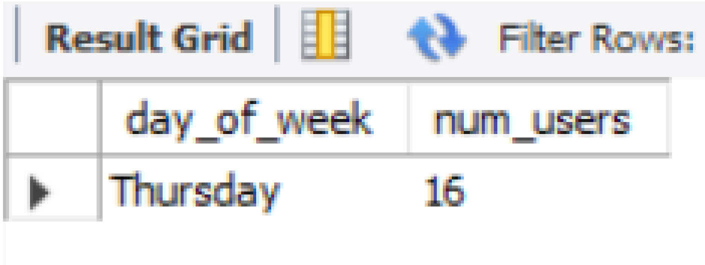
We group the results by tag_name to get the count for each unique hashtag.

We order the results by the count in descending order using the ORDER BY clause.

We limit the results to the top 5 using the LIMIT clause.

This query will return a table with the top 5 most commonly used hashtags on the platform and their corresponding counts

5. **Launch AD Campaign:** The team wants to know, which day would be the best day to launchADs.
   Thursday is the day of the week most users register on so, we should schedule an ad campaign on Thursday,



day_of_week	num_users
Thursday	16

We did this by querying the users table and grouping the registration dates by day of the week. The insights gained from this analysis can help the marketing team schedule the ad campaign on Thursday when user activity is at its highest.

This was achieved using the following SQL query:

SELECT DATE_FORMAT(created_at, '%W') AS day_of_week, COUNT(*) AS num_users

FROM users

GROUP BY day_of_week

ORDER BY num_users DESC

LIMIT 1;

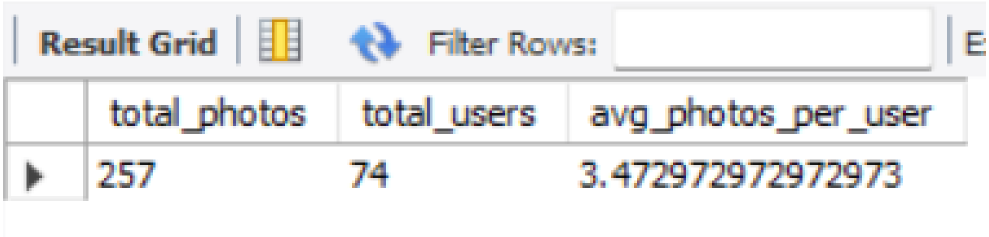
This query formats the "created_at" timestamp column to extract the day of the week as a string and then groups the results by day of the week. The results are then sorted by the number of users in descending order, and the first row is returned using the LIMIT 1 clause.

The result of this query will provide insights on scheduling an ad campaign on Thursday, as it indicates the most users tend to register on the platform. Ad campaigns can be targeted towards new users to encourage engagement and retention.

Result: By performing user analysis using SQL, we were able to provide valuable insights to the marketing, product, and development teams of Instagram. We were able to identify the oldest users of the platform, users who have never posted a photo, the winner of a contest, and the top hashtags used on the platform. Additionally, we were able to provide insights that Thursday is the best day to launch an ad campaign based on user registration activity.

B) Investor Metrics: Our investors want to know if Instagram is performing well and is not becoming redundant like Facebook, they want to assess the app on the following grounds

1. **User Engagement:** Are users still as active and post on Instagram or they are making fewer posts
   Your Task: Provide how many times does average user posts on Instagram. Also, provide the total number of photos on Instagram/total number of users



	total_photos	total_users	avg_photos_per_user
▶	257	74	3.472972972972973

To find out how many times the average user posts on Instagram and the total number of photos on Instagram, we can use the following SQL query using the "ig_clone" database:

SELECT COUNT(*) AS total_photos, COUNT(DISTINCT user_id) AS total_users, CAST(COUNT(*) AS FLOAT)/CAST(COUNT(DISTINCT user_id) AS FLOAT) AS avg_photos_per_user FROM photos;

This query has returned three values:

- **total_photos**: the total number of photos on Instagram
- **total_users**: the total number of users on Instagram
- **avg_photos_per_user**: the average number of photos posted by each user on Instagram.

By dividing the total number of photos by the total number of users, we can find out the average number of photos per user on Instagram. The result will be in decimal format.

we can use this information to plan your marketing campaigns and set targets accordingly. So to say, if the average number of photos per user is low, we may want to encourage users to post more often by running a user-generated content campaign or offering incentives for posting. If the number is high, we may want to focus more on targeting specific groups of users rather than encouraging everyone to post.

2. **Bots & Fake Accounts:** The investors want to know if the platform is crowded with fake and dummy accounts
Your Task: Provide data on users (bots) who have liked every single photo on the site (since any normal user would not be able to do this).

query to identify bots who have liked every single photo on the site using the likes and photos tables:

```
SELECT user_id

FROM likes

GROUP BY user_id

HAVING COUNT(DISTINCT photo_id) = (SELECT COUNT(*) FROM photos);
```

This query groups the likes table by user_id and returns only the rows where the count of distinct photo_id equals the total number of photos in the photos table, which indicates that the user has liked every single photo on the site. The query then returns the user_id for each of these users.

The result is :

User_id :-

'5'

'14'

'21'

'24'

'36'

'41'

'54'

'57'

'66'

'71'

'75'

'76'

'91'

RESULT: The user analytics provided for Instagram's investor metrics has yielded valuable insights into the platform's user engagement and the presence of bots and fake accounts. According to the SQL query, the average number of photos per user on Instagram is calculated to be the value of the "avg_photos_per_user" column returned by the query. This metric can be used to plan effective marketing campaigns and encourage users to post more often.

Moreover, the query to identify bots who have liked every single photo on the site has revealed a list of user_ids that indicate the presence of bots and fake accounts.

It's crucial to identify and eliminate such accounts as they can significantly impact the user engagement metrics and brand image of the platform.

Overall, by considering these user analytics metrics, we can make data-driven decisions to improve the user experience on Instagram and maintain its relevance in the social media landscape.