**Project Title:** User Engagement Analysis on a Social Media Platform

**Objective:** Perform various analytical queries to derive insights from an Instagram-like platform's user activity, including identifying the most active users, finding potential bots, and analyzing hashtag usage.

**Dataset:**

* **users**: Contains user information.
  + Columns: id (int, AI, PK), username (varchar(255))
* **photos**: Contains photo information.
  + Columns: id (int, AI, PK), image\_url (varchar(255)), user\_id (int)
* **likes**: Contains information on photo likes.
  + Columns: user\_id (int, PK), photo\_id (int, PK)
* **follows**: Contains follow relationships between users.
  + Columns: follower\_id (int, PK), followee\_id (int, PK)
* **tags**: Contains tag information.
  + Columns: id (int, AI, PK), tag\_name (varchar(255))
* **photo\_tags**: Contains associations between photos and tags.
  + Columns: photo\_id (int, PK), tag\_id (int, PK)
* **comments**: Contains photo comments.
  + Columns: id (int, AI, PK), comment\_text (varchar(255)), photo\_id (int), user\_id (int)

**Key Queries and Insights:**

1. **Identify the 5 Oldest Users:**

SELECT id, username, created\_at

FROM users

ORDER BY created\_at

LIMIT 5;

1. **Find Inactive Users (Never Posted a Photo):**

SELECT id, username

FROM users

WHERE id NOT IN (SELECT user\_id FROM photos);

1. **Contest Winner (Most Likes on a Photo):**

SELECT u.id, username, COUNT(\*) AS like\_count

FROM users u

JOIN photos p ON u.id = p.user\_id

JOIN likes l ON p.id = l.photo\_id

GROUP BY u.id, username

ORDER BY like\_count DESC

LIMIT 3;

1. **Average Number of Posts per User:**

SELECT (SELECT COUNT(\*) FROM photos) / (SELECT COUNT(DISTINCT user\_id) FROM photos) AS average\_posts\_per\_user;

1. **Top 5 Most 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;

1. **Identify Potential Bots (Users Who Liked Every Photo):**

SELECT l.user\_id, u.username, COUNT(\*)

FROM users u

JOIN likes l ON u.id = l.user\_id

JOIN photos p ON l.photo\_id = p.id

GROUP BY l.user\_id, u.username

HAVING COUNT(DISTINCT l.photo\_id) = COUNT(DISTINCT p.id)

ORDER BY COUNT(\*) DESC;

1. **Newest Users Joined in May:**

SELECT id, username, created\_at

FROM users

WHERE MONTH(created\_at) = 5

ORDER BY created\_at DESC

LIMIT 5;

1. **Users with Specific Username Pattern and Activity:**

SELECT DISTINCT u.id, username

FROM users u

JOIN photos p ON u.id = p.user\_ID

JOIN likes l ON u.id = l.user\_id

WHERE username RLIKE '^c.\*[0-9]$';

1. **Top 30 Users by Photo Post Count (3 to 5 Photos):**

SELECT u.id, username, COUNT(\*) AS post\_count

FROM users u

JOIN photos p ON u.id = p.user\_id

GROUP BY u.id, username

HAVING post\_count BETWEEN 3 AND 5

ORDER BY post\_count

LIMIT 30;

**Key Achievements:**

* Designed an ER diagram to represent the relationships between users, photos, likes, comments, tags, and follows.
* Implemented complex SQL queries to extract meaningful insights and trends from the data.
* Identified key user behaviors, including the most active users, inactive users, and potential bots.
* Analyzed hashtag usage to provide recommendations for brand campaigns.

This project demonstrates proficiency in SQL, including the use of joins, subqueries, grouping, and analytical functions to derive insights from a relational database.