Instagram User Analytics

Project Description:

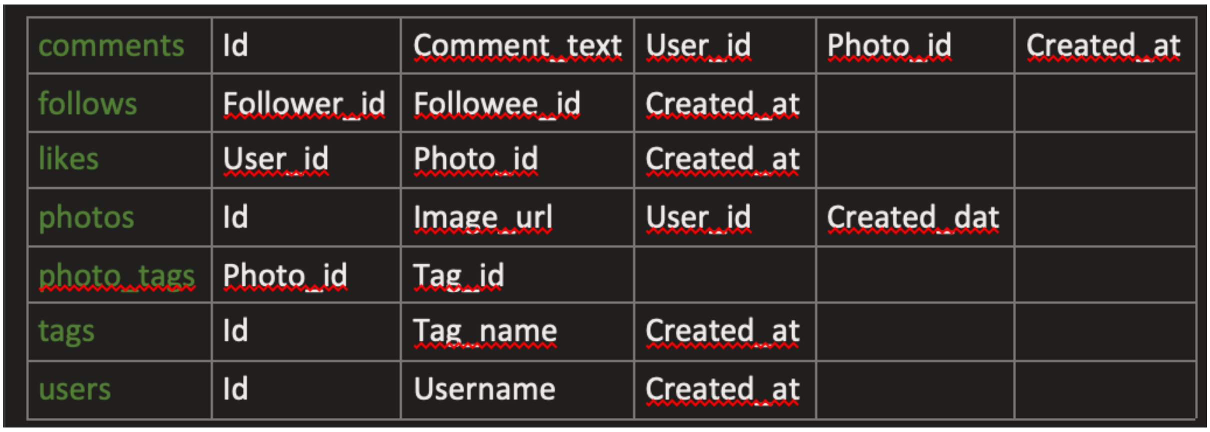
This project focuses on analysing user engagement and providing valuable insights for the Instagram platform.

Through this analysis, I aim to derive valuable information for the **Marketing, Product, and Development teams.**

I will track **User Interactions**, **Engagement Patterns of the customers with Instagram**, and **key metrics of likes, number of photos, what hashtags are being used** etc to gain insights that can guide decision-making for **launching AD campaigns, developing new features, and contributing to overall business growth.**

Approach:

My approach to solving this project was firstly to **analyse the database.** I made this table (picture attached below) to make myself aware of what all columns are there in each table and which are the common columns through which will be able to link the tables.



comments	Id	Comment_text	User_id	Photo_id	Created_at
follows	Follower_id	Followee_id	Created_at		
likes	User_id	Photo_id	Created_at		
photos	Id	Image_url	User_id	Created_dat	
photo_tags	Photo_id	Tag_id			
tags	Id	Tag_name	Created_at		
users	Id	Username	Created_at		

Using this approach helped me to derive meaningful conclusions regarding user behaviour, engagement, contest outcomes, hashtag popularity, and optimal ad campaign launch days.

Tech-Stack Used:

MySQL Workbench Version 8.0.33 for MacBook was used to analyse **Ig_Clone database** and extract insights.

Insights:

Analysing the database helped in answering the questions asked by the Marketing Team

- **Rewarding Most Loyal Users**: Identified the five oldest users on Instagram, recognizing their loyalty and longevity on the platform. Here I checked when was each account created on and then sorted it by ascending to know the five oldest accounts.

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

- **Reminding Inactive Users to Start Posting:** Identifying users who have never posted a single photo and can target them with promotional emails to encourage their active participation. Here I found which all ids are there in Users table but not there in Photos table. These are the users who have never posted any photos.

```
SELECT
   username
FROM
   users
      LEFT JOIN
   photos ON users.id = photos.user_id
WHERE
   photos.user_id IS NULL;
```

- **Declaring Contest Winner:** By analysing , the contest winner was determined by identifying the user with the highest number of likes on a single photo. Here I joined the “likes” table with the “photos” table to know which photo got the most likes.

```
SELECT
   users.username,
   photos.id AS photo_id,
   COUNT(likes.user_id) AS total_likes
FROM
   users
      INNER JOIN
   photos ON users.id = photos.user_id
      INNER JOIN
   likes ON photos.id = likes.photo_id
GROUP BY photos.id
ORDER BY total_likes DESC
LIMIT 1;
```

- **Hashtag Researching:** By examining the data, the top five most commonly used hashtags on the platform were identified, which provided insights for effective content reach. Here we analysed “photo_tags” table and counted the occurrence of each tag and then ranked the top 5 tags.

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

- **Launch AD Campaign:** To determine the best day to launch ads, I analysed the user registration data and identified the day of the week with the highest number of registrations. I used Google here to know search for the function which can tell me the name of the day based on the date provided.

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

- **User Engagement-** Analysing the database helped in answering the question if the users are still as active on Instagram as before or if are they making fewer posts. To know this I calculated the average number of posts per user by dividing the total number of photos by the total number of users

```
SELECT
   COUNT(*) AS total_photos,
   (SELECT
         COUNT(*)
      FROM
         users) AS total_users,
   COUNT(*) / (SELECT
         COUNT(*)
      FROM
         users) AS avg_posts_per_user
FROM
   photos;
```

- **Bots & Fake Accounts-** It is impossible for a human being to like every single photo on Instagram but it can be done through Internet bots. Hence it is very important to analyse the bots/fake accounts to make the platform as authentic and genuine as possible. Here we identified how many likes has each distinct id done, then compared it to the total, if both of them matches, then we conclude that it is a fake account or done by a bot

```
SELECT
    users.id AS user_id,
    users.username,
    COUNT(likes.photo_id) AS liked_photo_count
FROM
    users
        INNER JOIN
    likes ON users.id = likes.user_id
GROUP BY users.id
HAVING COUNT(DISTINCT likes.photo_id) = (SELECT
        COUNT(*)
    FROM
        photos);
```

Result:

Through the project, I not only gained valuable insights into user engagement and behaviour on Instagram, but it also helped me in working with a database which is similar to a live database, it has helped me make the queries by relating different SQL tables with different SQL queries.

After I completed this project it helped me understand SQL more, how to think while working with SQL and how to join two different tables. How to use Google properly to get my answers when I am stuck.

I believe my knowledge of SQL has increased as I have completed this project.