**TRAINITY PROJECT 2: DATA ANALYTICS PROCESS**

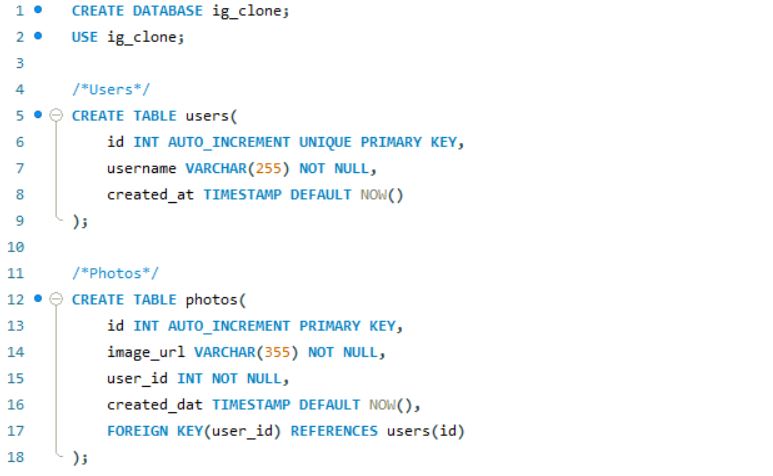
##### TOPIC: Instagram User Analytics Using MySQL Workbench

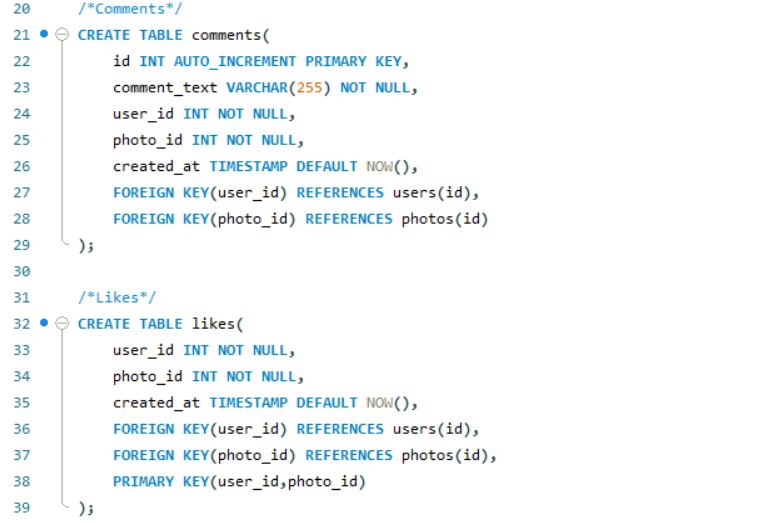
**PROJECT DESCRIPTION:**

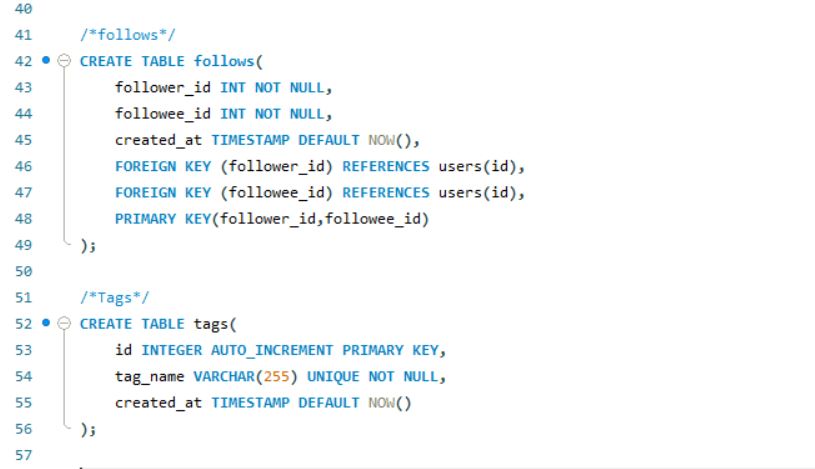
As a data analyst working closely with the product team at Instagram, my primary responsibility is to explore and analyze user behaviors and engagement on the Instagram app. These analyses are vital for formulating data-driven decisions that further the growth and success of the platform. Utilizing SQL and MySQL Workbench as my tools, I strive to uncover valuable insights from Instagram user data, which empowers numerous teams across the business.

This project aims to examine user engagement on Instagram and offer valuable insights that can aid the product team in refining the platform. By thoroughly analyzing user interactions, including likes, comments, follows, and posts, we can gain a deeper comprehension of user behaviors and preferences. This knowledge can be leveraged by various teams, encompassing marketing, product development, and customer support, to optimize the user experience and impel business growth.

**APPROACH:**

**Create a Database:**The database file was provided in the attachments.



****

****

**Perform Analysis:** I utilised MySQL Workbench to perform the analysis and answer the questions mentioned in the project description. I used SQL queries to extract the required information from the database, ensuring the accuracy and efficiency of the queries.

**TECH- STACK USED:**

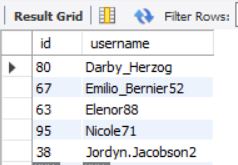
The software I am using to do the analysis is MySQL Workbench, Version: 8.0.34 (MySQL Community Server – GPL). I used this software for the project as it is easy to use and freely available.

**INSIGHTS:**

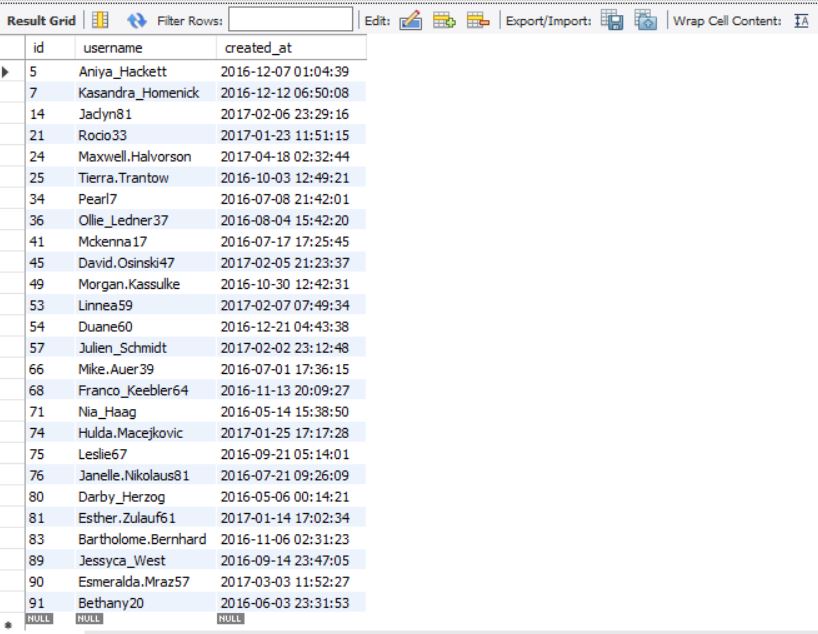
**A) Marketing Analysis:**

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.  
   To identify the five oldest users on Instagram from the provided database:

The SQL query provided retrieves rows from the **users** table, orders them by the **created\_at** column in ascending order (earliest to latest), and limits the result to the first 5 rows.



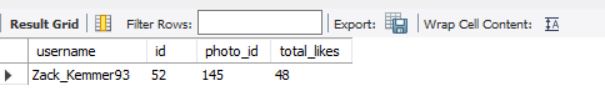
1. **Inactive User Engagement:** The team wants to encourage inactive users to start posting by sending them promotional emails.  
   To identify users who have never posted a single photo on Instagram:



1. **Contest Winner Declaration:** The team has organized a contest where the user with the most likes on a single photo win.  
   To determine the winner of the contest and provide their details to the team:

To retrieve the winner details

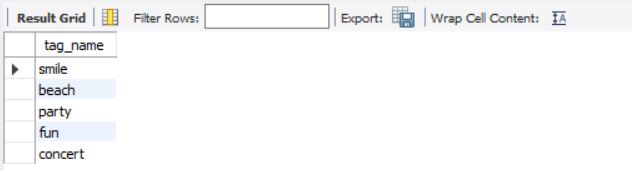
In the first part, we are querying the **likes** table to identify the photo that has received the most likes. The **SELECT** statement retrieves the **photo\_id** from the **likes** table. The **GROUP BY** clause groups the data by **photo\_id**. The **ORDER BY** clause arranges the grouped results in descending order of the count of **user\_id** (which represents the likes). So, the most liked photo will be at the top. The **LIMIT 1** clause ensures that you only get the top result, which is the photo with the most likes.

In the second part, we are building upon the previous query to retrieve the details of the user who posted the most liked photo. The **WITH** clause creates a common table expression (CTE) named **MostLikedPhoto**. This CTE uses the same logic as the first query to find the most liked photo but also calculates the total number of likes for that photo and stores it in the **total\_likes** column. Now we're joining the **MostLikedPhoto** CTE with the **photos** and **users** tables. This allows us to fetch details like the username (**u.username**), user ID (**u.id**), photo ID (**p.id**), and the total number of likes (**MostLikedPhoto.total\_likes**).

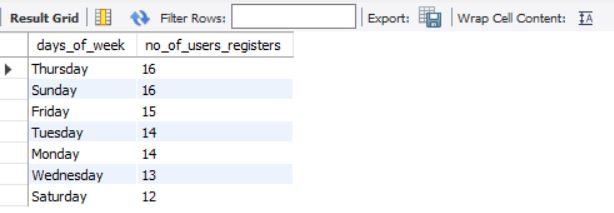
1. **Hashtag Research:** A partner brand wants to know the most popular hashtags to use in their posts to reach the most people.  
   To identify and suggest the top five most commonly used hashtags on the platform:

Here we are using **WITH** clause creating a Common Table Expression (CTE) named **top\_tags** to identify the top five most commonly used hashtags. From photo\_tags table, the **GROUP BY** clause groups the data by **tag\_id** counting how many times each tag has been used. The **ORDER BY** clause arranges the groups in descending order of the count of **tag\_id** occurrences, which means the most commonly used tags will be at the top. The **LIMIT 5** clause ensures you only retrieve the top five tags.

In the final part of the query, we're joining the **top\_tags** CTE with the **tags** table to retrieve the names of the top tags. This allows us to present the actual hashtag names that are the most commonly used on the platform.



1. **Ad Campaign Launch:** The team wants to know the best day of the week to launch ads.  
   To determine the day of the week when most users register on Instagram and to provide insights on when to schedule an ad campaign.

In the above query, to determine the date of the week, we use **DAYNAME** function which is applied to the **created\_at** column as it's a timestamp column. The **count(\*)** function calculates the number of users registered on each day of the week. The **GROUP BY** clause groups the data by the day of the week to aggregate the registration counts. The **ORDER BY** clause arranges the results in descending order based on the count of registered users.

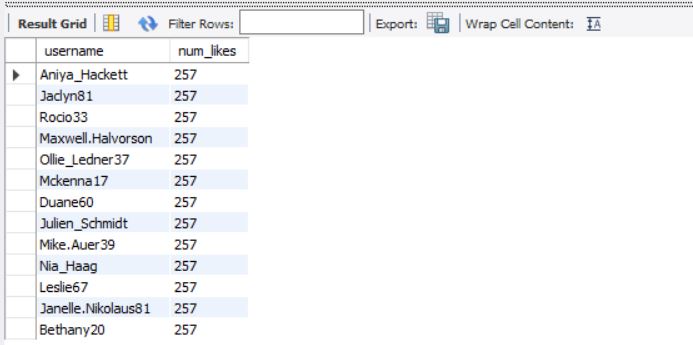
The insights gained from this query will help us determine the best day of the week to launch ad campaigns, as we'll be able to identify the day when most users tend to register on Instagram. This information can guide your ad campaign scheduling strategy for maximum impact.

**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.  
   To calculate the average number of posts per user on Instagram and to provide the total number of photos on Instagram divided by the total number of users.



1. **Bots & Fake Accounts:** Investors want to know if the platform is crowded with fake and dummy accounts.  
   To identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user:

It joins the users and likes table, then groups the data by **user\_id**, and filters the results to only include users who have liked the same number of photos as the total number of photos available in the photos table.

**RESULTS:**

I gained a lot of valuable insights from this project as it gave a glimpse of how business projects work and how a company derive its data from SQL queries and implement them in real time which results in increasing their overall productivity. The results of the analysis will be helpful to various departments.

This Instagram user analytics project provided insights on marketing, user engagements, investor metrics. These insights can be used by Instagram product team to launch new campaigns, track users’ engagement and improve user experience.

The insights I derived in these projects:

1. Top 5 Oldest Users of Instagram
2. Users who never posted photos on Instagram
3. Most liked photo on Instagram
4. Top 5 most commonly used hash-tags on Instagram
5. Total number of users on Instagram
6. Total number of photos on Instagram
7. Average number of photos per user
8. Bots and fake accounts on Instagram