Instagram User Analytics

**Project Description:**

In this project, we use our SQL knowledge to find Instagram user analytics. User analysis is the process by which we track how users engage and interact with our digital product (software or mobile application) to derive business insights for marketing, product & development teams.

Here are the tasks we need to perform for the different teams:

* **Marketing**: The marketing team wants to launch some campaigns, and they need your help with the following
  + ***Rewarding Most Loyal Users:*** People who have been using the platform for the longest time.

***Our Task***: Find the 5 oldest users of Instagram from the database provided.

* + ***Remind Inactive Users to Start Posting:*** By sending them promotional emails to post their 1st photo.

***Our Task***: Find the users who have never posted a single photo on Instagram.

* + ***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.

***Our Task***: Identify the winner of the contest and provide their details to the team.

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

***Our Task:*** Identify and suggest the top 5 most used hashtags on the platform.

* + ***Launch AD Campaign:*** The team wants to know, which day would be the best day to launch ADs.

***Our Task:*** What day of the week do most users register on? Provide insights on when to schedule an ad campaign.

* **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.
  + ***User Engagement:*** Are users still as active and post on Instagram or they are making fewer posts?

***Our Task:*** Provide how many times does average user posts on Instagram. Also, provide the total number of photos on Instagram/total number of users.

* + ***Bots & Fake Accounts:*** The investors want to know if the platform is crowded with fake and dummy accounts.

***Our 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).



[This Photo](http://www.bettertechtips.com/social-media/instagram-analytics-tools/) by Unknown Author is licensed under [CC BY-SA](https://creativecommons.org/licenses/by-sa/3.0/)

**Approach:**

We are provided with the data set. We assume the data to be clean. We explored the schema of all the provided tables in the MySQL workbench and understood how the tables relate to each other. The database was cautiously converted into SQL tables to ensure most benefit from the database provided. Then we used the database and executed some of the SQL queries of our knowledge to obtain the required solution.

**Tech-Stack Used:**

The software used for this project is ***MySQL Workbench 8.0 CE.***

Graphical user interface, application

Description automatically generated

**Insights:**

***Rewarding Most Loyal Users***: To know about the most loyal users we need to find the users whose accounts were create very long time ago.

Here is the following query we used to find the most loyal users:

SELECT

username, created\_at

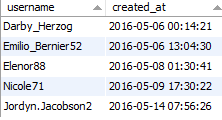
FROM

users

ORDER BY created\_at

LIMIT 5;

This query resulted in the following output:



These are the 5 most loyal users of Instagram

***Reminding Inactive users to start posting:*** The inactive users are the ones who do not post on their account in this context the photos.created\_at is null.

Here is the query we used to find the inactive users:

SELECT

users.id, users.username

FROM

users

LEFT JOIN

photos ON users.id = photos.user\_id

WHERE

created\_dat IS NULL;

This query resulted in the following output:

Table

Description automatically generated

The above are the users who had not posted a single photo till now

***Declaring Contest Winner:*** To know about the contest winner we need to count the no of likes on each photo of users and find the highest liked photo.

Here is the following query we used to find the highest liked photo:

SELECT

photos.user\_id,

users.username,

likes.photo\_id,

COUNT(likes.user\_id) AS number\_of\_Likes

FROM

likes

JOIN

photos

JOIN

users ON users.id = photos.user\_id

AND photos.id = likes.photo\_id

GROUP BY likes.photo\_id

ORDER BY number\_of\_Likes DESC

LIMIT 1;

This query resulted in the following output:



***Hashtag Researching:***To find about the top hashtags used in the posts we need to find the no of times the hashtag is used in the photos

Here is the query to find the highest used hashtag that is used in the photos:

SELECT

tags.id,

tags.tag\_name,

COUNT(photo\_tags.photo\_id) AS no\_of\_times\_tag\_used

FROM

photo\_tags

JOIN

tags ON photo\_tags.tag\_id = tags.id

WHERE

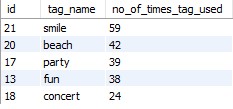
photo\_tags.photo\_id

GROUP BY tag\_id

ORDER BY no\_of\_times\_tag\_used DESC

LIMIT 5;

This query resulted in the following output:



***Launch AD Campaign:***

To launch a ad campaign we need to convey the message to most no of audience, so we need to find the day when most candidates register on ?

Here is the query to find the day when the most no of users register:

SELECT

DAYOFWEEK(created\_at) AS day\_of\_the\_week,

COUNT(users.id) AS number\_of\_Accounts\_Registered

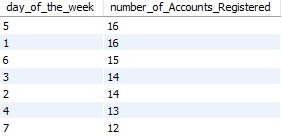
FROM

users

GROUP BY day\_of\_the\_week

ORDER BY number\_of\_Accounts\_Registered DESC;

The query resulted in the following output:



***User Engagement:*** To provide how many times does average user posts on Instagram and the total number of photos on Instagram/total number of users. To do that we need the count of total photos on the Instagram and the total no of users.

Here is the query :

SELECT

COUNT(DISTINCT users.id) as 'total users',

COUNT(photos.id) as 'total no of photos',

ROUND((COUNT(photos.id) / COUNT(DISTINCT users.id)), 2) as 'total\_photos/total\_users'

FROM

users

LEFT JOIN

photos ON users.id = photos.user\_id;

The query resulted in the following output:



We can also use data to find the avg of posts by users who have posted on their timeline:

SELECT

COUNT(distinct users.id) AS 'no of users who post',

COUNT(photos.id) AS 'total no of post',

COUNT(photos.id)/count(distinct users.id) AS 'no of times avg users post'

FROM

users

JOIN

photos ON users.id = photos.user\_id;

The query resulted in the following output:



***Bots and Fake accounts***:

To 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).

Here is the query for our task:

SELECT

users.username,

likes.user\_id,

COUNT(likes.created\_at) AS “number of photos liked”

FROM

likes

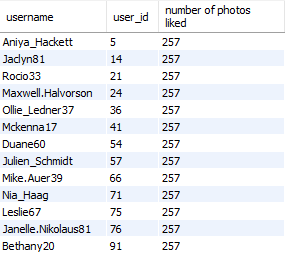
JOIN

users ON users.id = likes.user\_id

GROUP BY likes.user\_id

HAVING COUNT(likes.created\_at) = 257;

The query resulted in the following output:



These usernames are that of

***Result:*** By completing the project, I am feeling more confident in my SQL knowledge. It really helped me to brush up on my concepts related to Sub-queries and Aggregate functions. It also helped me to understand the table schema and how normalization can better help to understand the dataset.

BY

THOTAMCHETTY RANJITH

THANKYOU