**Instagram User Analytics**

**SQL Fundamentals**

Project Description

* This project aims to extract useful insights from raw data/metadata. Using various database management tools and even visualise them to increase the platform’s efficiency.

Project Approach

* The project was executed using SQL where queries were utilized to create a database from the provided raw data. Sorting and data extracting queries were then implemented to obtain the required data/insights.

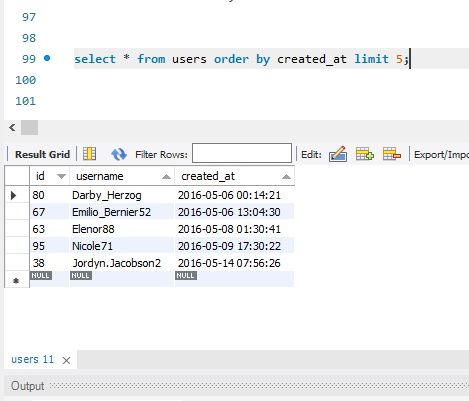
**Tech Stack Used**

* The Tech stack used included MySQL Workbench v8.0.30.0 which was an excellent tool for querying the database, thanks to its ease of access, simple setup and GUI as well as its troubleshooting support.

Project Insights:

A) Marketing:

* 1.) Rewarding most loyal users: People who have been using the platform for the longest time.
* CODE:-
* select \* from users order by created\_at limit 5;
* RESULT:-



2.) Remind inactive users to start posting: By sending them promotional emails to post their photos.

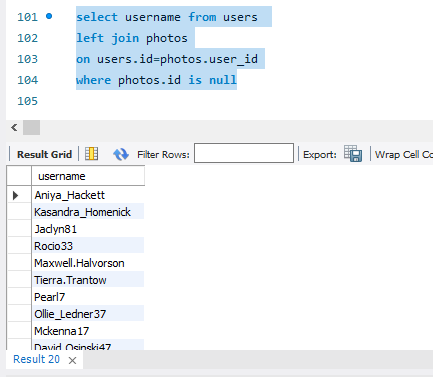
CODE:’-

* select username from users left join photoson users.id=photos.user\_idwhere photos.id is null

RESULT:-

# 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.Niklaus81
* Darby\_Herzog
* Esther.Zulauf61
* Bartholome.Bernhard
* Jessyca\_West
* Esmeralda.Mraz57
* Bethany20.



3.) Contest Winner Declaration:  The team has organized a contest where the user with the most likes on a single photo wins.

Code:- select username , photos.id , photos.image\_url , count(likes.user\_id) as total\_likes from photos

inner join likes on likes.photo\_id=photos.id

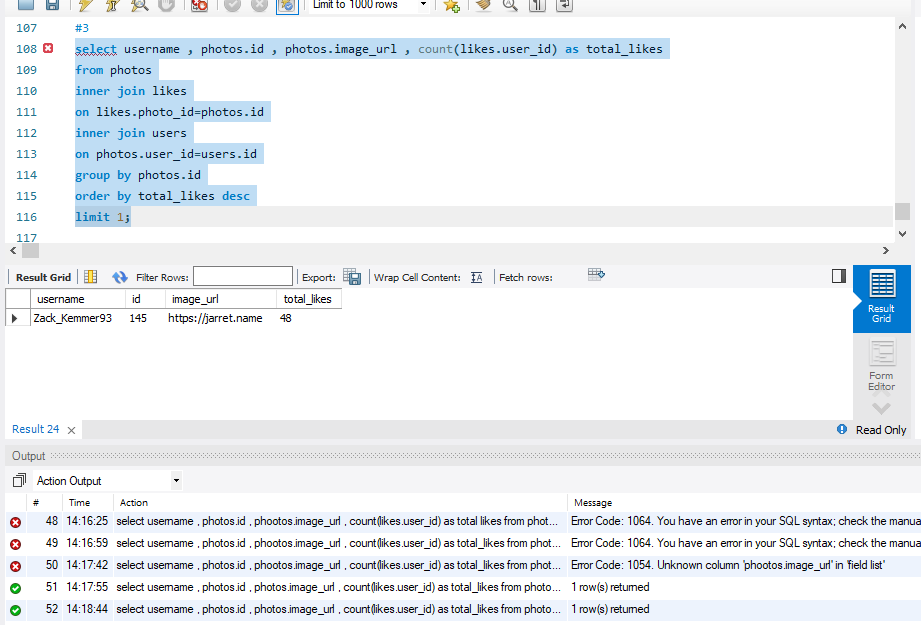
inner join users on photos.user\_id=users.id

group by photos.id order by total\_likes desc limit 1;

* Result:-

|  |  |  |  |
| --- | --- | --- | --- |
| username | id | Photo\_url | Total\_likes |

|  |  |  |  |
| --- | --- | --- | --- |
| Zack\_Kemmer93 | 145 | https://jarret.name | 48 |



4.) Hashtag Research: A partner brand wants to know the most popular hashtags to use in their posts to reach the most people.

* CODE:-

select tag\_name , count(\*) as total

from tags

join photo\_tags

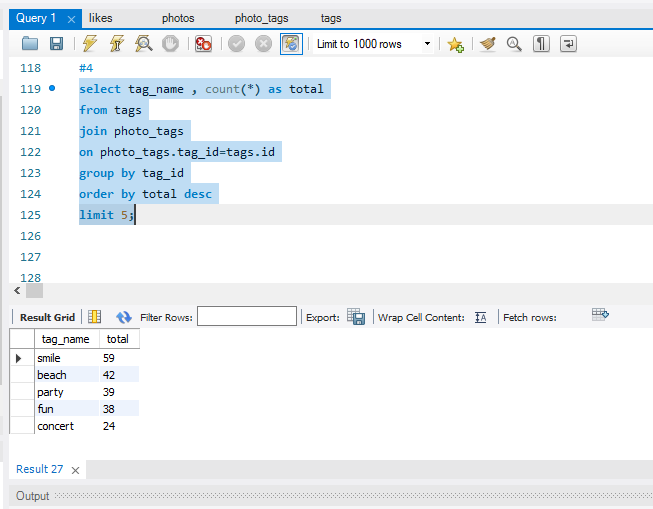
on photo\_tags.tag\_id=tags.id

group by tag\_id

order by total desc

limit 5;

RESULT:-



5). Ad Campaign Launch: The team wants to know the best day of the week to launch ads.

CODE:-

select dayname(created\_at) as days , count(\*) as total

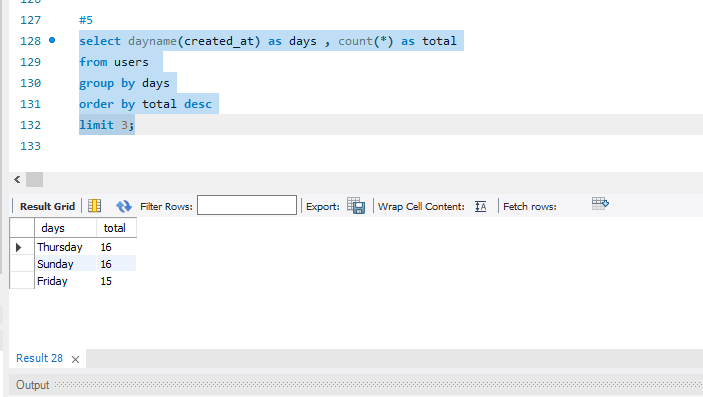
from users

group by days

order by total desc

limit 3;

RESULT:-



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.

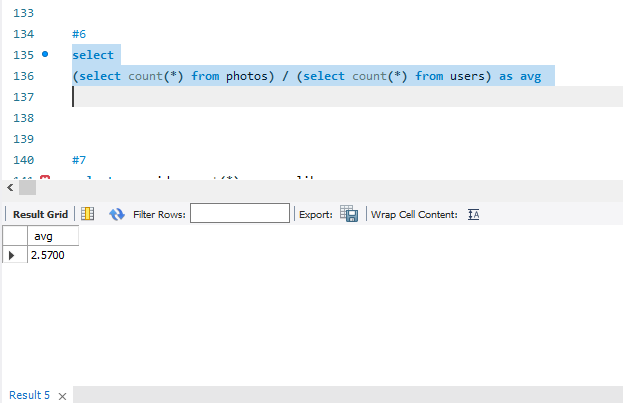
CODE:-

Select

(select count(\*) from photos) / (select count(\*) from users) as avg

RESULT:-

257.00



2). Bots & Fake Accounts: Investors want to know if the platform is crowded with fake and dummy accounts.

CODE:-

select user\_id, count(\*) as num\_likes

from likes

group by user\_id

having num\_likes = (select count(\*) from photos);

select u.username, count(\*) as num\_likes

from users u

join likes l on u.id = l.user\_id

group by u.id

having num\_likes = (select count(\*) from photos);

RESULT:-

|  |  |
| --- | --- |
| username | num\_likes |

|  |  |
| --- | --- |
| Aniya\_Hackett | 257 |
| Jaclyn81 | 257 |
| Rocio33 | 257 |
| Maxwell.Halvorson | 257 |
| Ollie\_Ledner37 | 257 |
| Mckenna17 | 257 |
| Duane60 | 257 |
| Julien\_Schmidt | 257 |
| Mike.Auer39 | 257 |
| Nia\_Haag | 257 |
| Leslie67 | 257 |
| Janelle.Nikolaus81 | 257 |
| Bethany20 | 257 |

