Instagram User Analytics

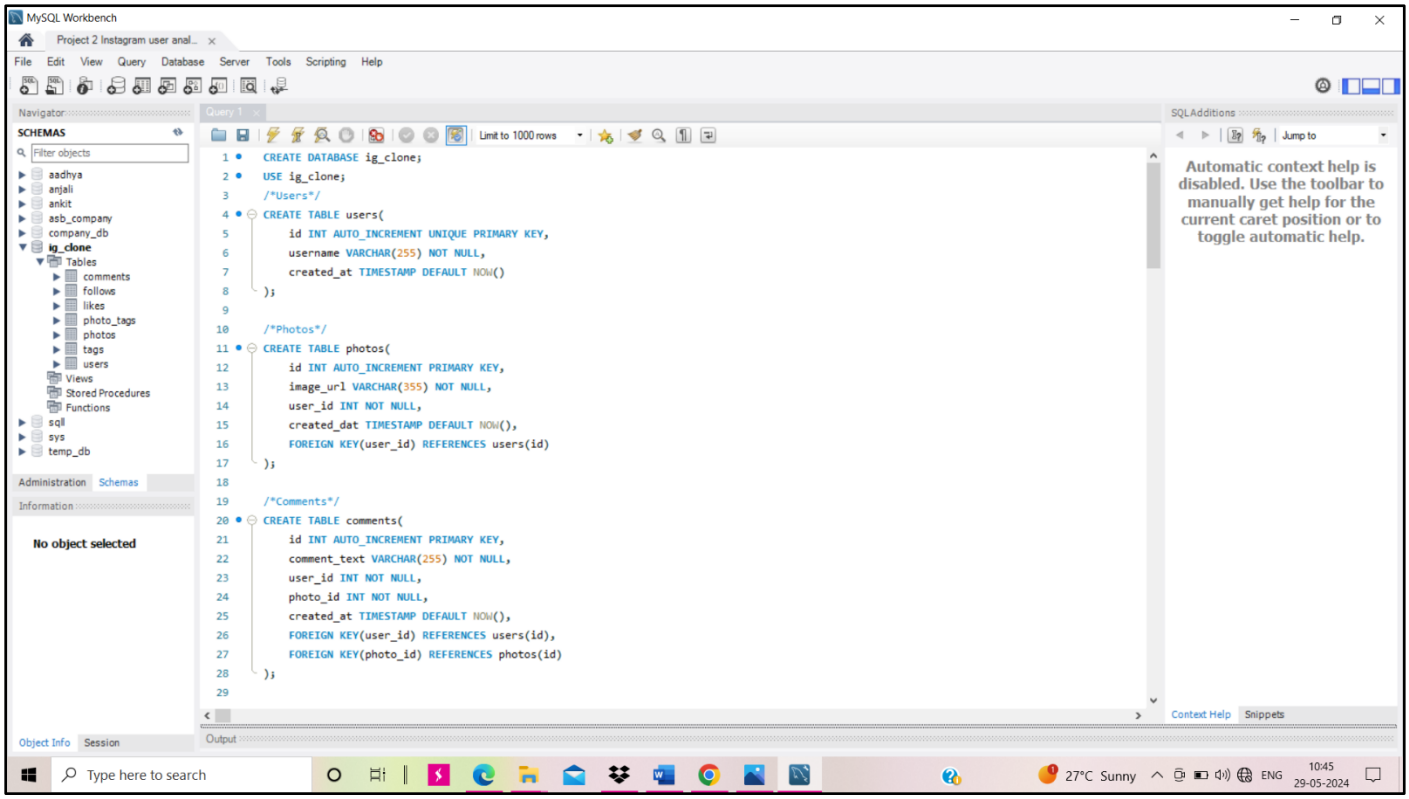
A. Project Description :- The goal of this project is to gather useful insights that can support the business's expansion by studying user interactions and engagement with the Instagram app. This project aims to extract useful insights from raw data/metadata using a variety of database management tools, as well as visualize them, in order to improve platform efficiency.

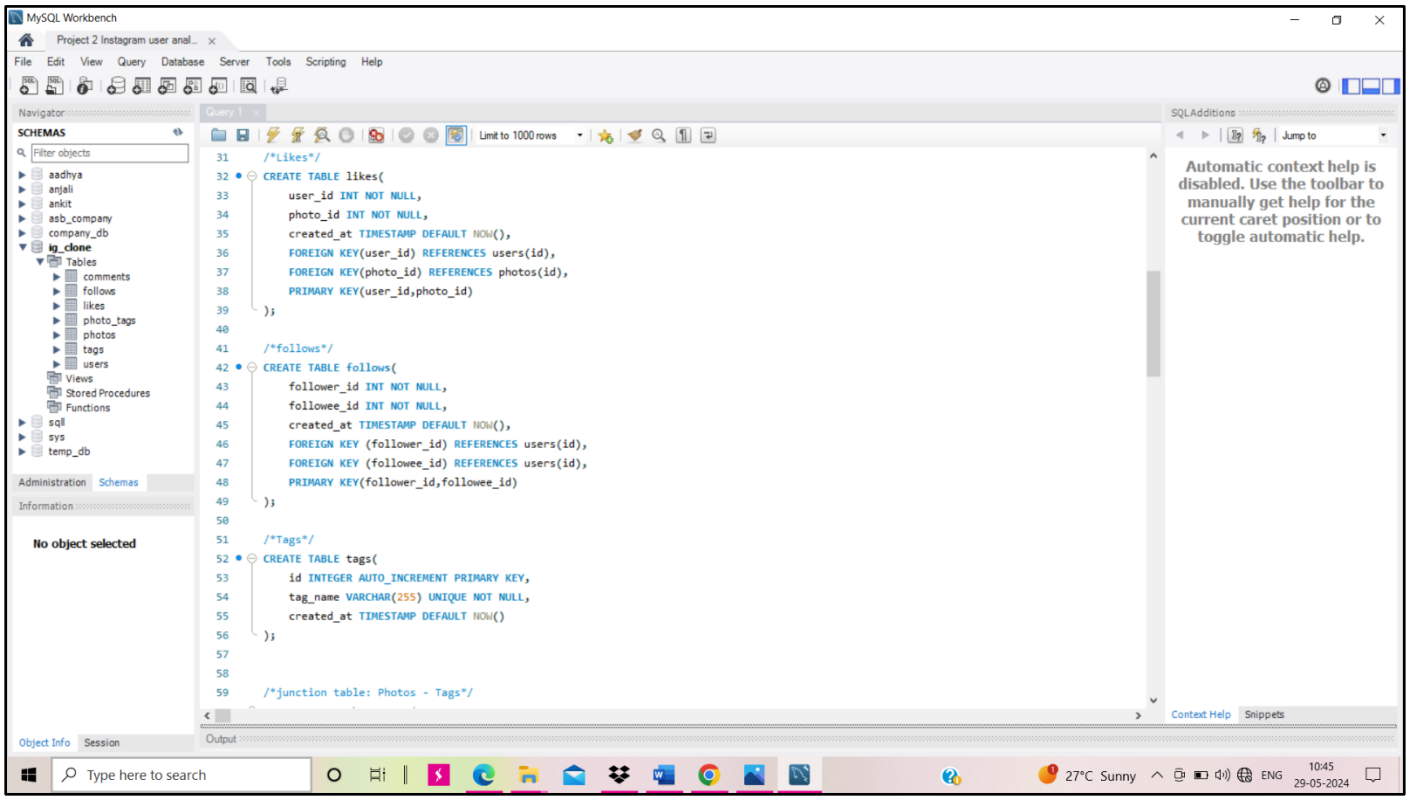
B. Project approach :- SQL was used to create a database from the provided raw data through the execution of queries. After that, queries for data extraction and sorting were used to get the needed information. According to the questions SQL commands are applied and results were obtained.

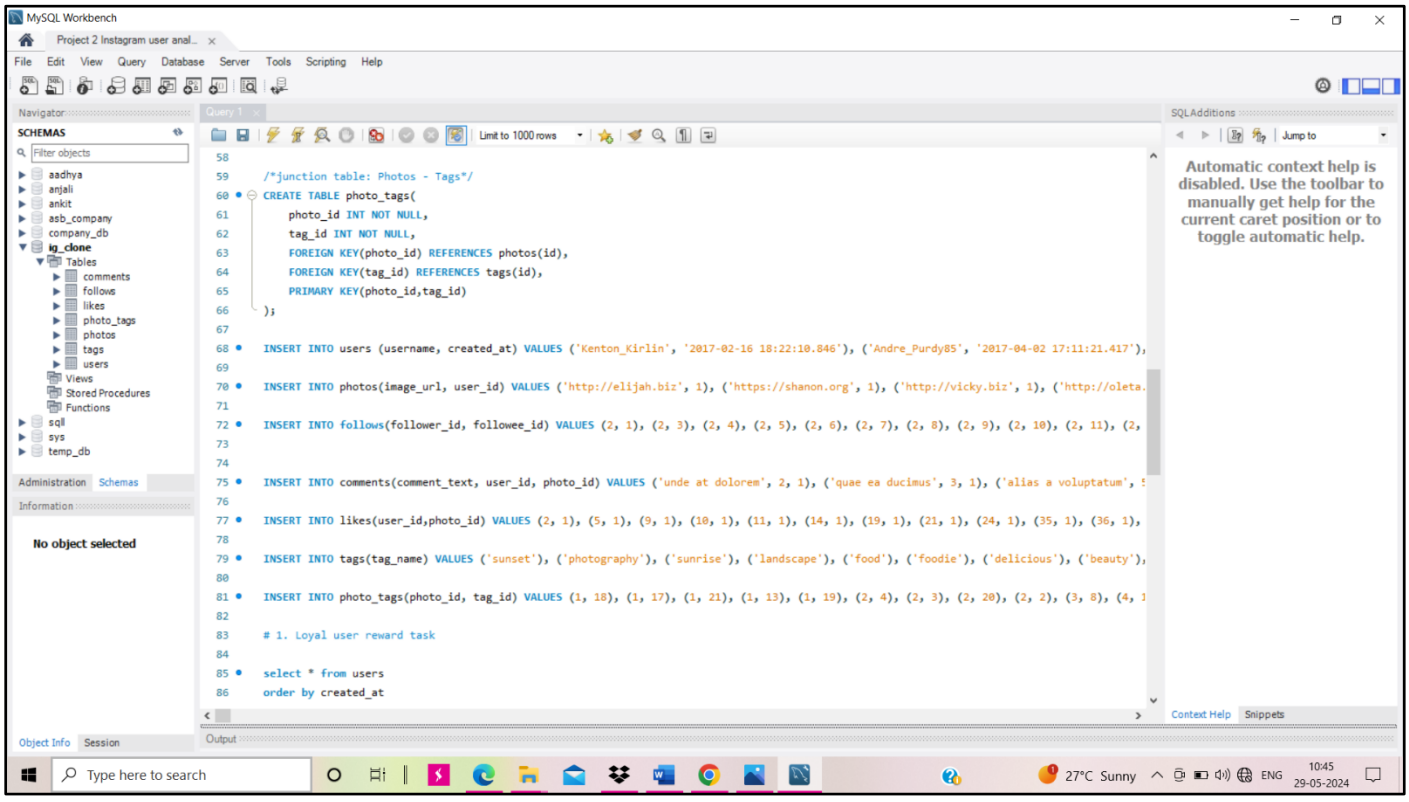
C. Tech-Stack Used :- The tech stack included MySQL Workbench v8.0.30.0, a great tool for database queries because of its accuracy, speed, simplicity, and ease of use. It enables you to work with database objects, design, create, and browse your schemas; it also lets you design and execute SQL queries to manipulate with data that has been stored.

➢ Project Insight

Before going for analysis all the raw data is inserted in MYSQL which can be seen in the following screenshots :-




MySQL Workbench
Project 2 Instagram user anal...
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
aadhya
anjali
ankit
asb_company
company_db
ig_clone
Tables
comments
follows
likes
photo_tags
photos
tags
users
Views
Stored Procedures
Functions
sqll
sys
temp_db
Administration Schemas
Information
No object selected
Object Info Session
Query 1
Limit to 1000 rows
31 /*Likes*/
32 CREATE TABLE likes(
33 user_id INT NOT NULL,
34 photo_id INT NOT NULL,
35 created_at TIMESTAMP DEFAULT NOW(),
36 FOREIGN KEY(user_id) REFERENCES users(id),
37 FOREIGN KEY(photo_id) REFERENCES photos(id),
38 PRIMARY KEY(user_id,photo_id)
39);
40
41 /*follows*/
42 CREATE TABLE follows(
43 follower_id INT NOT NULL,
44 followee_id INT NOT NULL,
45 created_at TIMESTAMP DEFAULT NOW(),
46 FOREIGN KEY (follower_id) REFERENCES users(id),
47 FOREIGN KEY (followee_id) REFERENCES users(id),
48 PRIMARY KEY(follower_id,followee_id)
49);
50
51 /*Tags*/
52 CREATE TABLE tags(
53 id INTEGER AUTO_INCREMENT PRIMARY KEY,
54 tag_name VARCHAR(255) UNIQUE NOT NULL,
55 created_at TIMESTAMP DEFAULT NOW()
56);
57
58
59 /*junction table: Photos - Tags*/
SQLAdditions
Jump to
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Context Help Snippets
Output
Type here to search
27°C Sunny
ENG
10:45
29-05-2024


MySQL Workbench
Project 2 Instagram user anal...
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
aadhya
anjali
ankit
asb_company
company_db
ig_clone
Tables
comments
follows
likes
photo_tags
photos
tags
users
Views
Stored Procedures
Functions
sqll
sys
temp_db
Administration Schemas
Information
No object selected
Object Info Session
Query 1
Limit to 1000 rows
58
59 /*junction table: Photos - Tags*/
60 CREATE TABLE photo_tags(
61 photo_id INT NOT NULL,
62 tag_id INT NOT NULL,
63 FOREIGN KEY(photo_id) REFERENCES photos(id),
64 FOREIGN KEY(tag_id) REFERENCES tags(id),
65 PRIMARY KEY(photo_id,tag_id)
66);
67
68 INSERT INTO users (username, created_at) VALUES ('Kenton_Kirlin', '2017-02-16 18:22:10.846'), ('Andre_Purdy85', '2017-04-02 17:11:21.417'),
69
70 INSERT INTO photos(image_url, user_id) VALUES ('http://elijah.biz', 1), ('https://shanon.org', 1), ('http://vicky.biz', 1), ('http://oleta.
71
72 INSERT INTO follows(follower_id, followee_id) VALUES (2, 1), (2, 3), (2, 4), (2, 5), (2, 6), (2, 7), (2, 8), (2, 9), (2, 10), (2, 11), (2,
73
74
75 INSERT INTO comments(comment_text, user_id, photo_id) VALUES ('unde at dolorem', 2, 1), ('quae ea ducimus', 3, 1), ('alias a voluptatum', 5
76
77 INSERT INTO likes(user_id,photo_id) VALUES (2, 1), (5, 1), (9, 1), (10, 1), (11, 1), (14, 1), (19, 1), (21, 1), (24, 1), (35, 1), (36, 1),
78
79 INSERT INTO tags(tag_name) VALUES ('sunset'), ('photography'), ('sunrise'), ('landscape'), ('food'), ('foodie'), ('delicious'), ('beauty'),
80
81 INSERT INTO photo_tags(photo_id, tag_id) VALUES (1, 18), (1, 17), (1, 21), (1, 13), (1, 19), (2, 4), (2, 3), (2, 20), (2, 2), (3, 8), (4, 1
82
83 # 1. Loyal user reward task
84
85 select * from users
86 order by created_at
SQLAdditions
Jump to
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Context Help Snippets
Output
Type here to search
27°C Sunny
ENG
10:45
29-05-2024

A) Marketing Analysis:

1. Loyal User Reward: To reward people who have been using the platform for the longest time

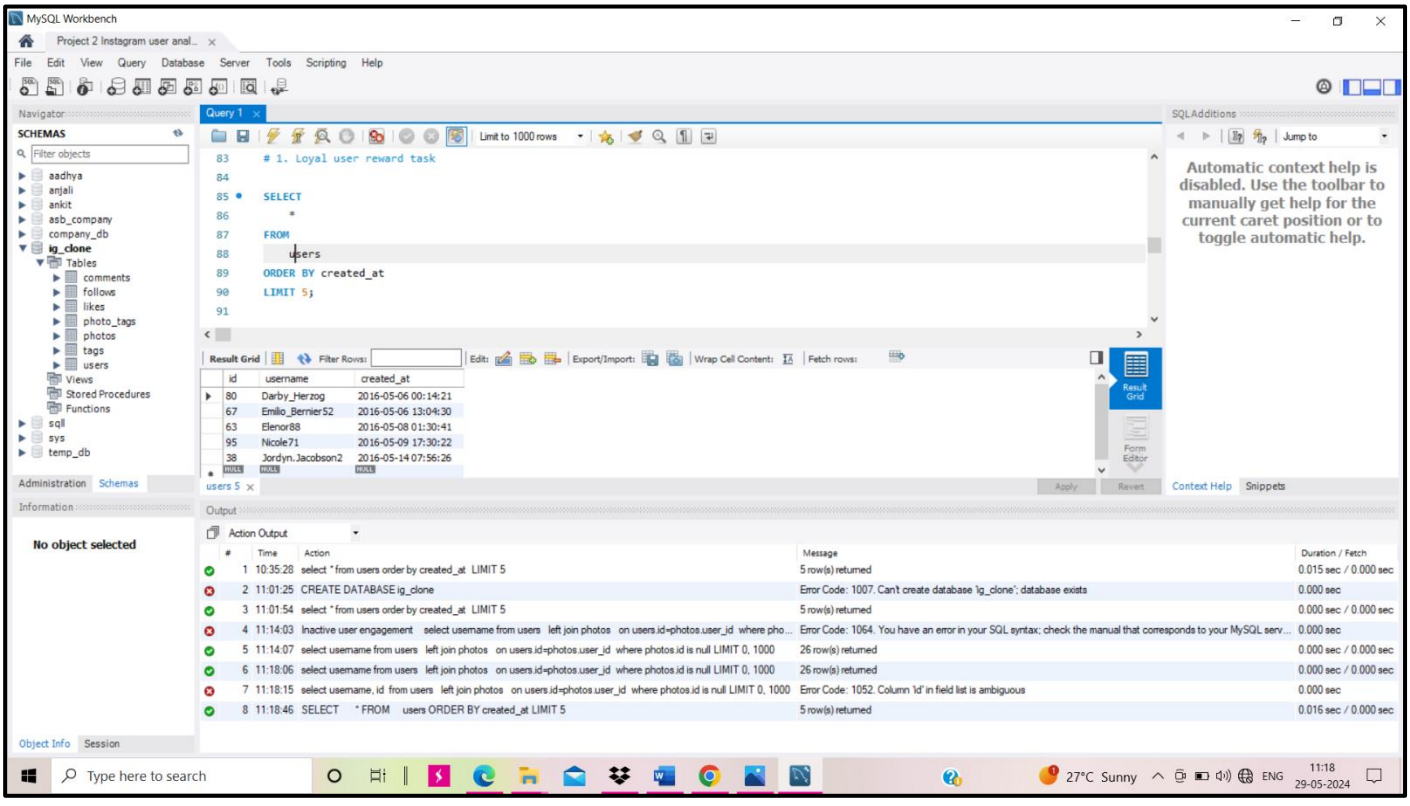
Conclusion /result :- The five oldest users on Instagram are as follows:

ID	Username	Created at
80	Darby_Herzog	2016-05-06 00:14:21
67	Emilio_Bernier52	2016-05-06 13:04:30
63	Elenor88	2016-05-08 01:30:41
95	Nicole71	2016-05-09 17:30:22
38	Jordyn.Jacobson2	2016-05-14 07:56:26

Code used :-

```
select * from users
order by created_at
LIMIT 5 ;
```

- Screenshot of code and after running code its output



-

2. Inactive User Engagement : users who is inactive has to send promotional emails to be active on Instagram, for this we have to identify users who have never posted a single photo on Instagram.

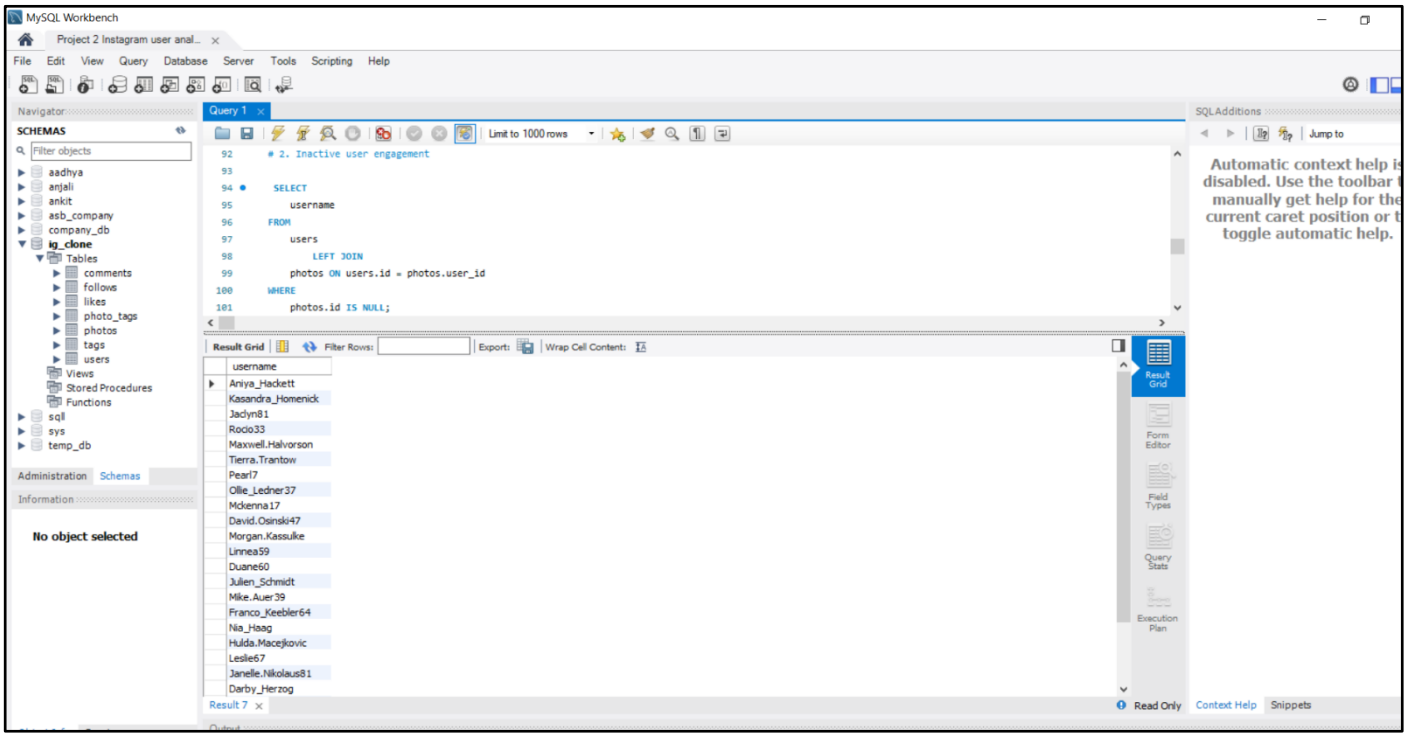
Conclusion /result :- The following users were inactive since long time :

Aniya_Hackett	Esmeralda.Mraz57	Jessyca_West	Mckenna17	Rocio33
Bartholome.Bernhard	Esther.Zulauf61	Julien_Schmidt	Mike.Auer39	Tierra.Trantow
Bethany20	Franco_Keebler64	Kasandra_Homenick	Morgan.Kassulke	
Darby_Herzog	Hulda.Macejkovic	Leslie67	Nia_Haag	
David.Osinski47	Jaclyn81	Linnea59	Ollie_Ledner37	
Duane60	Janelle.Nikolaus81	Maxwell.Halvorson	Pearl7	

Code used :-

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

- Screenshot of code and after running code it output



3. Contest Winner Declaration: The team has organized a contest where the user with the most likes on a single photo wins. We have to determine the winner

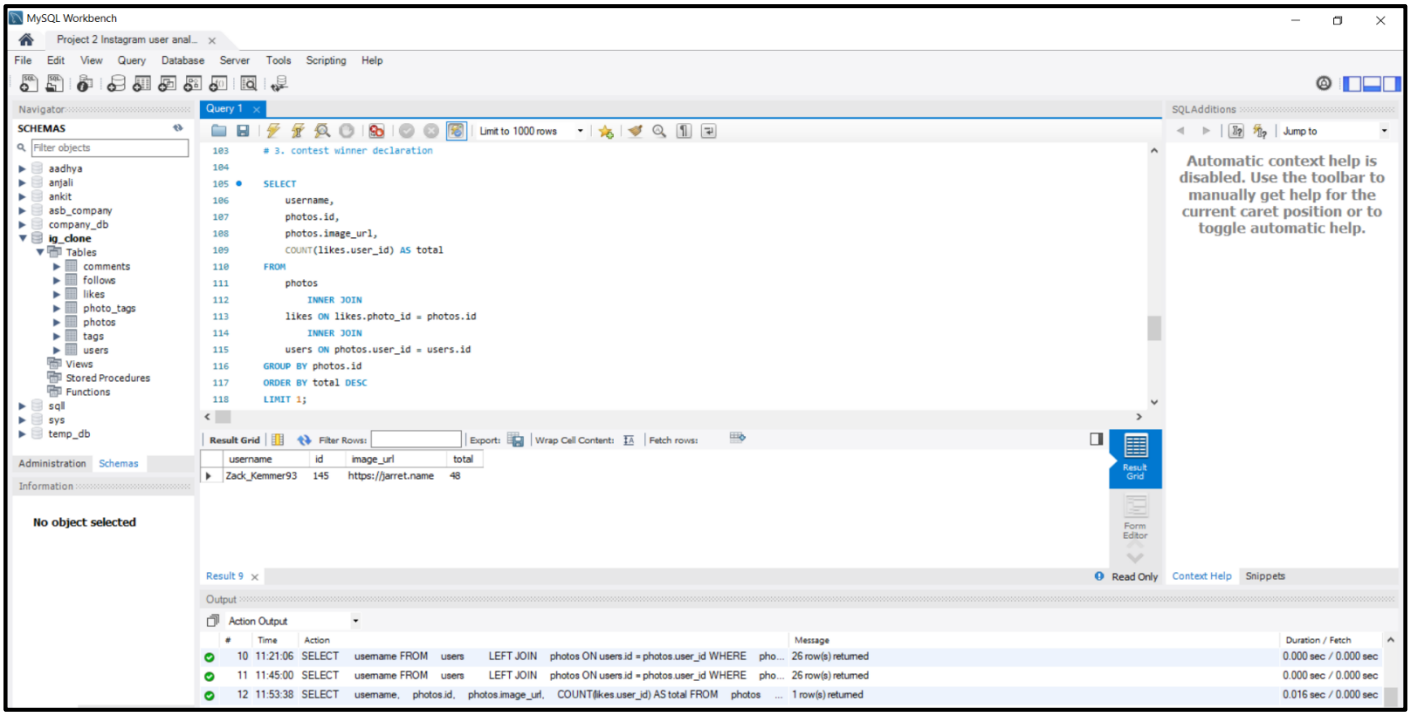
Conclusion /result :- The following user has most likes on his post

username	Id	Image url	total
Zack_Kemmer93	145	https://jarret.name	48

Code used :-

```
SELECT
  username,
  photos.id,
  photos.image_url,
  COUNT(likes.user_id) AS total
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 DESC
LIMIT 1;
```

- Screenshot of code and after running code and its output



4. Hashtag Research: A partner brand wants to know the most popular hashtags to use in their posts to reach the most people we have to Identify and suggest the top five most commonly used hashtags on the platform

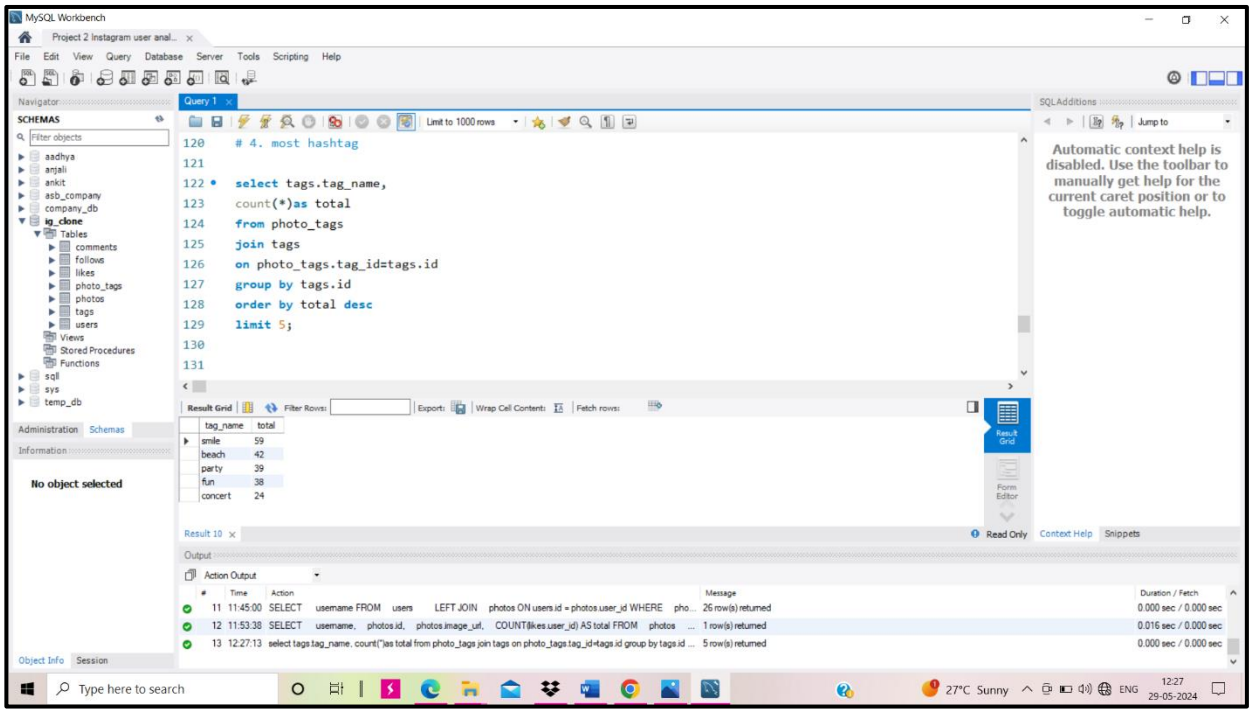
Conclusion/result :- following are top 5 trending hashtag that partner brand can use

Tag_name	Total
smile	59
beach	42
party	39
fun	38
concert	24

Code used :-

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

- Screenshot of code and after running code and its output



5. Ad Campaign Launch:- To know the best day of the week to launch ads.so, we have to Determine the day of the week when most users register on Instagram. & Provide insights on when to schedule an ad campaign.

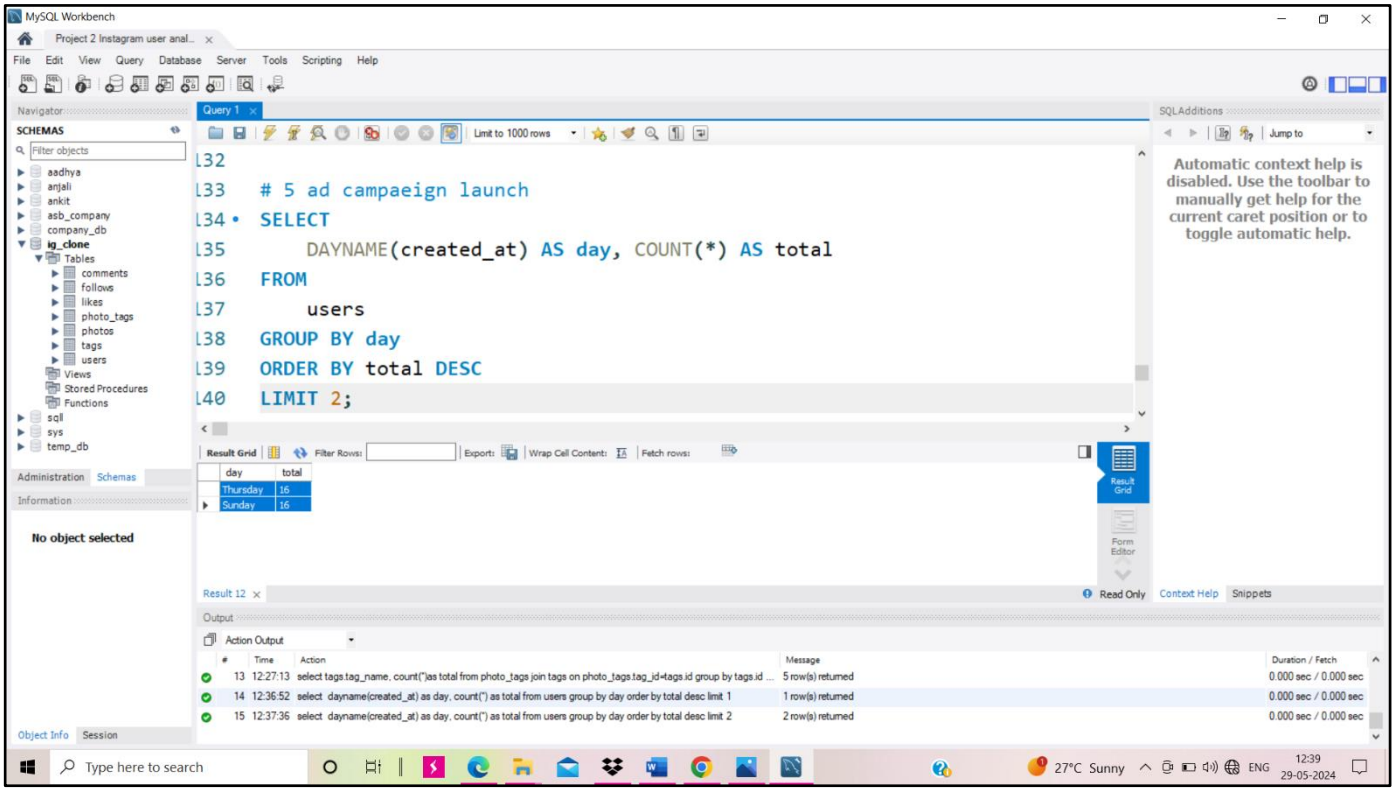
Conclusion/result :- following days were best for ad campaign

Day	Total
Thursday	16
Sunday	16

Code used :-

```
SELECT
   DAYNAME(created_at) AS day, COUNT(*) AS total
FROM
   users
GROUP BY day
ORDER BY total DESC
LIMIT 2;
```

- Screenshot of code and after running code and its output



B) Investor Metrics:

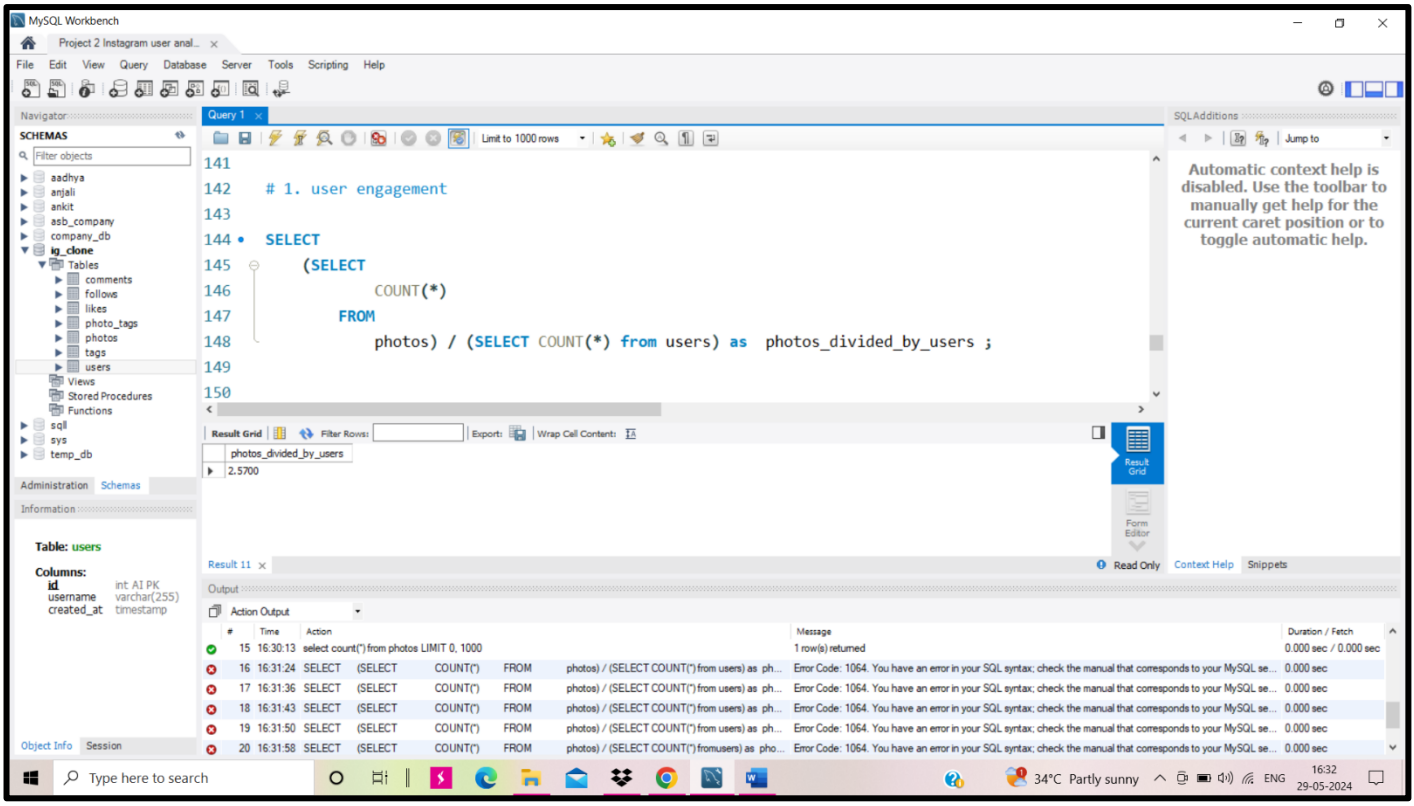
1. **User Engagement:** To find users are still active and posting on Instagram or if they are making fewer posts .for this we have to Calculate the average number of posts per user on Instagram. Also, provide the total number of photos on Instagram divided by the total number of users.

Conclusion/result :- The average number of post per user on Instagram is 2.57

Code used :-

```
SELECT
   (SELECT
         COUNT(*)
      FROM
         photos) / (SELECT COUNT(*) from users) as  photos_divided_by_users ;
```

- Screenshot of code and after running code and its output



2. **Bots & Fake 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 means that account is fake and dummy.

Conclusion/Result :- The table aside shows users who liked every post

Code Used :-

```
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);
```

username	Likes
Aniya_Hackett	257
Bethany20	257
Duane60	257
Jaclyn81	257
Janelle.Nikolaus81	257
Julien_Schmidt	257
Leslie67	257
Maxwell.Halvorson	257
Mckenna17	257
Mike.Auer39	257
Nia_Haag	257
Ollie_Ledner37	257
Rocio33	257

- Screenshot of code and after running code and its output