

# INSTAGRAM USER ANALYTICS

**PROJECT DESCRIPTION:** This project is all about the Instagram insights. I have converted the data of Instagram into meaningful data using some of the MySQL queries by performing the given tasks.

**APPROACH:** Firstly, I have reviewed the data given so that I can get an idea of it. I have gone through the tables to know which are primary keys and foreign keys so that I can connect the tables. I have mainly used grouping and sorting functions to complete the tasks.

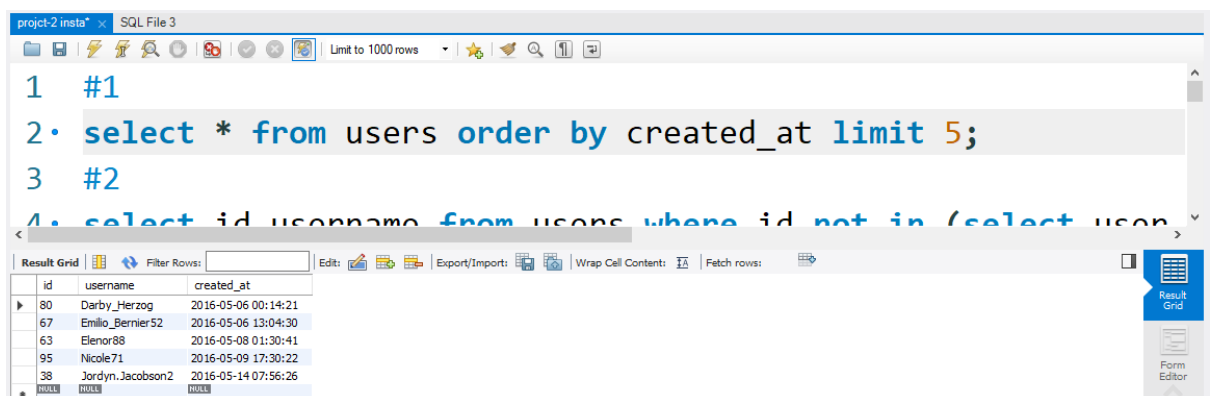
**TECH-STACK USED:** I have used MySQL workbench 8.0 CE as a database management tool. It provided a robust environment for designing and managing the data.

**INSIGHTS:** I have extracted valuable insights from the Instagram user analytics project, such as the oldest user on Instagram, and the most likes for a single photo. I also identified peak days for user engagement, helping optimize posting schedules for better reach.

## **RESULTS:**

### **A) MARKETING ANALYSIS:**

**1) Loyal User Reward:** Identify the five oldest users of Instagram from the database



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

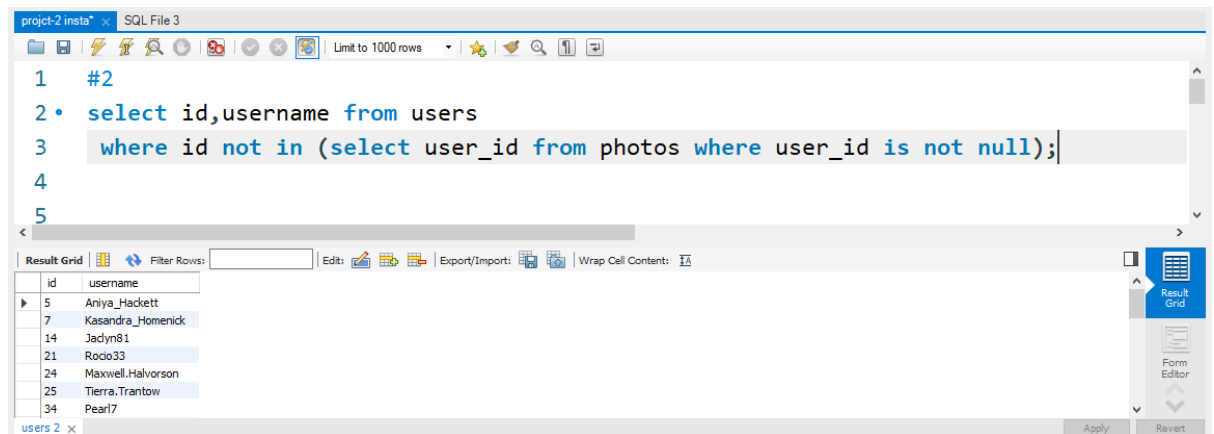
```
1 #1
2 • select * from users order by created_at limit 5;
3 #2
4 • select id, username from users where id not in (select user_id from likes)
```

The 'Result Grid' at the bottom displays the results of the first query, showing the five oldest users:

	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

**Result:** Darby\_Herzog, Emilio\_Bernier52, Elenor88, Nicole71, Jordyn.Jacobson2 these are the oldest users of Instagram.

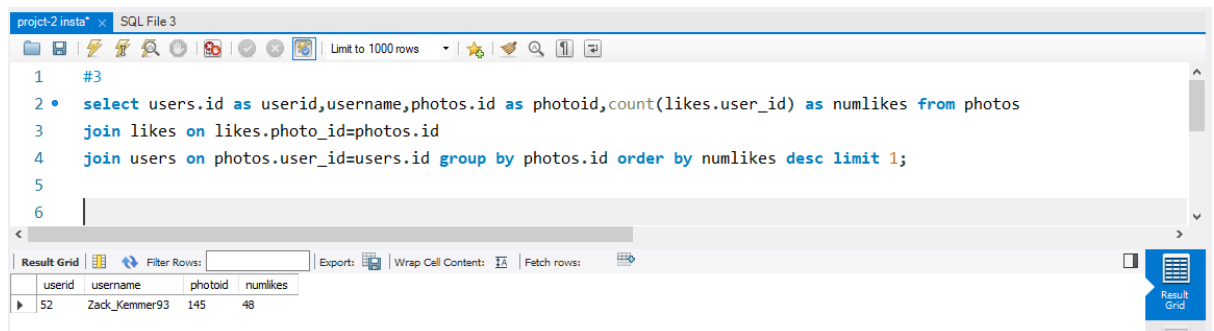
2) **Inactive User Engagement:** Identify users who never posted on Instagram.



**Result:** Below below-mentioned IDs and usernames have zero posts.

id	username
5	Aniya_Hackett
7	Kasandra_Homenick
14	Jaclyn81
21	Rocio33
24	Maxwell.Halvorson
25	Tierra.Trantow
34	Pearl7
36	Ollie_Ledner37
41	Mckenna17
45	David.Osinski47
49	Morgan.Kassulke
53	Linnea59
54	Duane60
57	Julien_Schmidt
66	Mike.Auer39
68	Franco_Keebler64
71	Nia_Haag
74	Hulda.Macejkovic
75	Leslie67
76	Janelle.Nikolaus81
80	Darby_Herzog
81	Esther.Zulauf61
83	Bartholome.Bernhard
89	Jessyca_West
90	Esmeralda.Mraz57
91	Bethany20

3) **Context Winner Declaration:** The user with the most likes on a single photo.

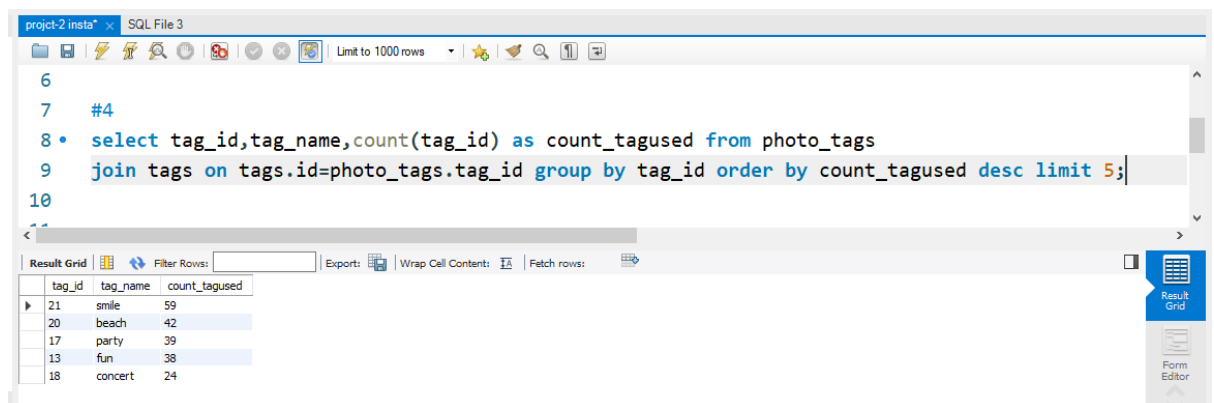


```
1 #3
2 • select users.id as userid,username,photos.id as photoid,count(likes.user_id) as numlikes from photos
3 join likes on likes.photo_id=photos.id
4 join users on photos.user_id=users.id group by photos.id order by numlikes desc limit 1;
5
6
```

userid	username	photoid	numlikes
52	Zack_Kemmer93	145	48

**Result:** The user with the **id-52** and username-**Zack\_kemmer93** has the most likes on a single photo which has an id-**145**, and has **48** likes.

4) **Hashtag Research:** Top 5 most commonly used hashtags on Instagram.



```
6
7 #4
8 • select tag_id,tag_name,count(tag_id) as count_tagused from photo_tags
9 join tags on tags.id=photo_tags.tag_id group by tag_id order by count_tagused desc limit 5;
10
```

tag_id	tag_name	count_tagused
21	smile	59
20	beach	42
17	party	39
13	fun	38
18	concert	24

**Result:** **smile, beach, party, fun, concert** are the five most commonly used hashtags on Instagram

5) **Ad Campaign Launch:** Determine the day of the week when most users register on Instagram.



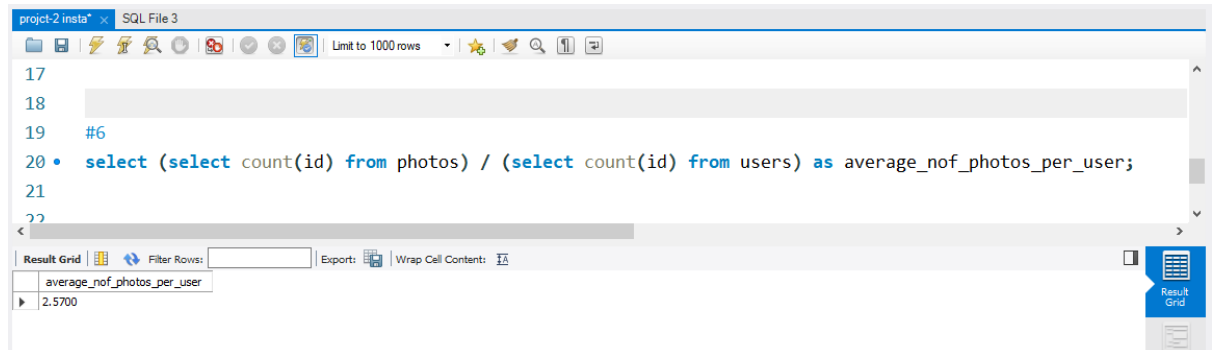
```
12
13 #5
14 • select dayname(created_at) as day_name, count(id) as num from users
15 group by day_name order by num desc limit 1;
16
17
```

day_name	num
Thursday	16

**Result:** **Thursday** is the day when most of the users got registered on Instagram.

## B) Investor metric:

- 1) **User engagement:** Provide the total number of photos on Instagram divided by total number of users on Instagram.



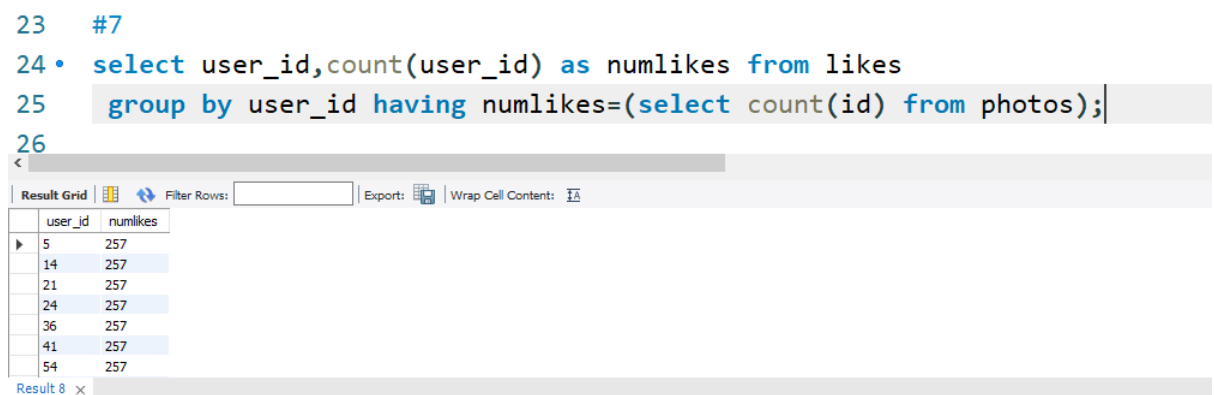
The screenshot shows a SQL IDE window titled 'project-2 insta\*' and 'SQL File 3'. The query editor contains the following SQL code:

```
17
18
19 #6
20 • select (select count(id) from photos) / (select count(id) from users) as average_nof_photos_per_user;
21
22
```

The results pane at the bottom shows a single row with the value 2.5700 for the column 'average\_nof\_photos\_per\_user'.

**Result: 2.5700** is the average

- 2) **Bots & Fake Account:** Identify users who liked every single photo on Instagram



The screenshot shows a SQL IDE window titled 'project-2 insta\*' and 'SQL File 3'. The query editor contains the following SQL code:

```
23 #7
24 • select user_id, count(user_id) as numlikes from likes
25 group by user_id having numlikes=(select count(id) from photos);
26
```

The results pane at the bottom shows a table with two columns: 'user\_id' and 'numlikes'. The table contains 8 rows of data, all with 'numlikes' equal to 257.

user_id	numlikes
5	257
14	257
21	257
24	257
36	257
41	257
54	257
57	257

**Result:** Below provided user **IDs** are potential bots who have liked all photos on Instagram.

user_id	numlikes
5	257
14	257
21	257
24	257
36	257
41	257
54	257
57	257
66	257
71	257
75	257
76	257
91	257