

INSTAGRAM USER ANALYTICS

DESCRIPTION: Analysing the dataset so that we can get valuable information about certain trends or important insights from the dataset. The dataset is about Instagram Users, with tables of users, photos, comments, likes, etc. These tables can be joined to get some insights. Some important information/insights are shown by the below queries.

APPROACH: -

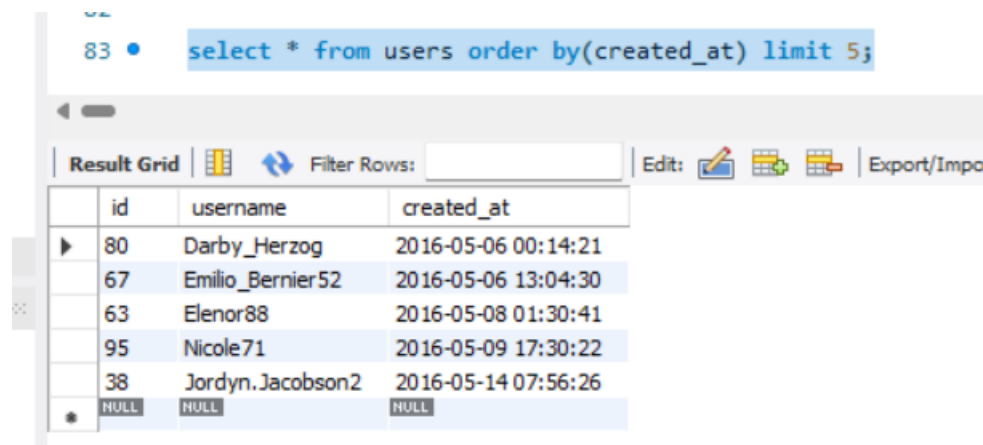
Analysis done on the following parts:-

Part (1). Marketing:- 1. Rewarding Most Loyal Users 2. Remind Inactive Users to Start Posting 3. Declaring the contest winners 4. Hashtag Researching 5. Launch AD Campaign

Part (2). Investor Metrics:- 1. User Engagement 2. Bots and Fake Accounts

TECH STACK: MYSQL USED

5 oldest users of the Instagram



The screenshot shows a MySQL query editor with the following SQL query: `select * from users order by(created_at) limit 5;`. Below the query, the 'Result Grid' displays the results of the query. The grid has columns for 'id', 'username', and 'created_at'. The results 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
*	NULL	NULL	NULL

ABOVE WE GOT 5 OLDEST USERS OF INSTAGRAM FROM THE DB.

the users who have never posted a single photo on Instagram

select username,users.id


from users left join photos

on users.id=photos.user_id

where photos.id IS NULL

order by users.id;

Result Grid



Filter Rows:

username	id
Jadyn81	14
Rocio33	21
Maxwell.Halvorson	24
Tierra.Trantow	25
Pearl7	34
Ollie_Ledner37	36
Mckenna17	41
David.Osinski47	45
Morgan.Kassulke	49
Linnea59	53
Duane60	54
Julien_Schmidt	57
Mike.Auer39	66
Franco_Keebler64	68
Nia_Haag	71
Hulda.Macejkovic	74
Leslie67	75
Janelle.Nikolaus81	76
Darby_Herzog	80
Esther.Zulauf61	81
Bartholome.Bernhard	83
Jessyca_West	89
Esmeralda.Mraz57	90
Bethany20	91

WE USED LEFT JOIN TO GET NULL RECORDS FROM THE PHOTOS TABLE WHICH REPRESENTS NO PHOTO WAS POSTED BY THAT PARTICULAR USER.

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.

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

```

91 • select users.id as user_id,photos.id as photo_id,photos.image_url,users.username,count(photo_id) as count
92 from users join photos on users.id=photos.user_id join likes on photos.id=likes.photo_id
93 group by photos.id
94 order by count DESC
95 limit 1;
96

```

WE JOINED 3 TABLES I.E. USERS, PHOTOS, AND LIKES, AND THEN GROUPED THEM BASED ON PHOTO ID AND THEN ORDERED THEM BY COUNT TO GET THE PHOTO ID WITH THE MOST LIKES.

A partner brand wants to know, which hashtags to use in the post to reach the most people on the platform.

Your Task: Identify and suggest the top 5 most commonly used hashtags on the platform.

```
select tags.tag_name,count(tags.tag_name) as total
```

```
from tags join photo_tags
```

on tags.id=photo_tags.tag_id

group by tags.tag_name

order by total desc

limit 5;

```
97 • select tags.tag_name,count(tags.tag_name) as total
98   from tags join photo_tags
99   on tags.id=photo_tags.tag_id
100  group by tags.tag_name
101  order by total desc
102  limit 5;
103
```

Result Grid			Filter Rows:	Export:	Wrap Cell Content
	tag_name	total			
	smile	59			
	beach	42			
	party	39			
	fun	38			
	concert	24			

WE GROUP TAGS BY TAGNAME AND SORT THEM BASED ON THEIR COUNTS TO GET THE TOP 5 MOST COMMONLY USED HASHTAGS.

The team wants to know, which day would be the best day to launch ADs.

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

```
103
104 • select dayname(created_at) as dayn , count(dayname(created_at)) as count
105   from users
106  group by dayn
107  order by count desc;
```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	dayn	count			
▶	Thursday	16			
	Sunday	16			
	Friday	15			
	Tuesday	14			
	Monday	14			
	Wednesday	13			
	Saturday	12			

WE GROUP THEM BY DAYNAMES AND THEN SORTED THEM BY THEIR COUNT. WE GET THURSDAY AND SUNDAY AS DAYS HAVING MAXIMUM COUNT.

Are users still as active and post on Instagram or they are making fewer posts
Your Task: Provide how many times does average user posts on Instagram. Also, provide the total number of photos on Instagram/total number of users

108

109 • `select ((select count(*) from photos)/(select count(*) from users)) as result;`

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	result
▶	2.5700

110 • `select user_id,count(user_id) as post_count from photos group by user_id order by user_id;`

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	user_id	post_count
▶	1	5
	2	4
	3	4
	4	3
	6	5
	8	4
	9	4
	10	3
	11	5
	12	4

Result 23 x

Output

ABOVE COUNTS OF PHOTOS AND USERS ARE DIVIDED TO GET DESIRED OUTPUT.

THEN GROUPING BY USER_ID WE GET COUNT OF USER POSTS.

The investors want to know if the platform is crowded with fake and dummy accounts
Your 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).

```

112 • select user_id,username,count(*) as total_likes
113 from users join likes on users.id=likes.user_id
114 group by users.id
115 having total_likes=(select count(*) from photos);
116
117

```

Result Grid			
Filter Rows:			
Export: Wrap Cell Conte			
	user_id	username	total_likes
▶	5	Aniya_Hackett	257
	14	Jadyn81	257
	21	Rocio33	257
	24	Maxwell.Halvorson	257
	36	Ollie_Ledner37	257
	41	Mckenna17	257
	54	Duane60	257
	57	Julien_Schmidt	257
	66	Mike.Auer39	257
	71	Nia_Haag	257
	75	Leslie67	257
	76	Janelle.Nikolaus81	257
	91	Bethany20	257

ABOVE WE JOIN USERS AND LIKES TABLES AND THEN GROUP BY USER ID, CTHE OUNT OF USER ID GIVES THE TOTAL NO. OF LIKES DONE BY THAT USER. IF THIS COUNT IS THE SAME AS THE TOTAL COUNT OF ALL PHOTOS I.E. USER LIKED ALL PHOTOS THEN THE USER IS A BOT.