



# Instagram User Analytics

PAVITHRA K R



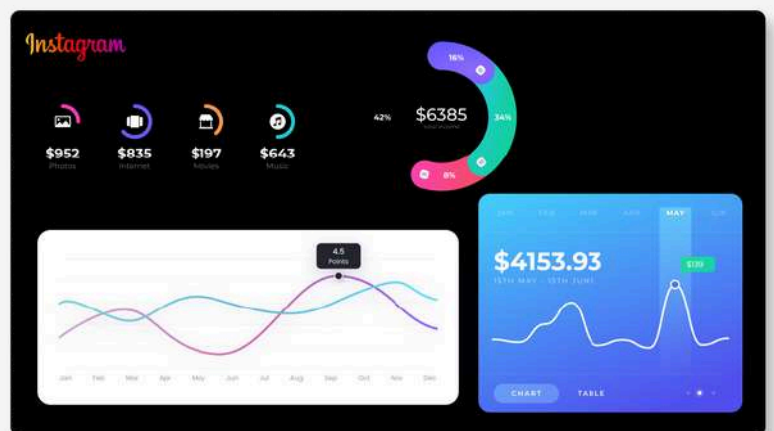
# Project Description and Approach

PAGE 1

The project aims to find out how much Instagram has reached users and take data-driven decisions on how to further engage more users on the platform.

In the first part of the project, which involves marketing analysis, my role is to provide answers to the questions asked about the oldest user on Instagram, inactive users, photos with the most likes, the most commonly used tags, and when most users create an account on Instagram. I have used MySQL to run queries to find the answers so that the management team can plan more activities on the platform.

The second part of the project involves statistics: finding the total number of posts by each user and the average number of posts made by a user on the platform. It also requires me to detect any fake accounts or bot accounts. I have used advanced SQL concepts like subqueries and Common Table Expressions to retrieve data efficiently and provide useful insights.



## TECH STACK USED

MySQL has been used in this project because of its ease of use, MySQL has a lot of useful functions making data analytics easier. It is a reliable tool and can be easily learnt.



## Marketing Analysis

- 01 Loyal User Reward: The marketing team wants to reward the most loyal users, i.e., those who have been using the platform for the longest time.**

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

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

5 rows in set (0.06 sec)

From the analysis, we come to know that the oldest user has been on the platform since May 6,2016.

- 02 Inactive User Engagement: The team wants to encourage inactive users to start posting by sending them promotional emails.**

```
select users.id, username from users left join photos on
users.id=photos.user_id where photos.user_id IS NULL;
```

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

26 rows in set (0.02 sec)

The above-given 26 users out of the total 100 users have never posted a single photo. So 26% of the users are inactive on the platform, which implies the platform is doing really well and requires only a small amount of promotion and marketing.

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

```
with abc as(
select photos.user_id as id, photos.id as photo_id, count(*) as
like_count
from likes join photos on likes.photo_id=photos.id
group by photos.id, photos.user_id order by like_count DESC limit 1)
select users.username,abc.like_count,abc.photo_id from users join
abc on users.id=abc.id;
```

```

+-----+-----+-----+
| username | like_count | photo_id |
+-----+-----+-----+
| Zack_Kemmer93 | 48 | 145 |
+-----+-----+-----+
1 row in set (0.02 sec)

```

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

select tags.tag\_name, count(\*) as no\_of\_times\_used from tags join photo\_tags on photo\_tags.tag\_id=tags.id group by tag\_name order by no\_of\_times\_used desc limit 5;

```

+-----+-----+
| tag_name | no_of_times_used |
+-----+-----+
| smile    | 59                |
| beach    | 42                |
| party    | 39                |
| fun      | 38                |
| concert  | 24                |
+-----+-----+
5 rows in set (0.01 sec)

```

The above-provided tags have been the five most popular tags and can be used by the partner brand to reach the most people.

#### 05 **Ad Campaign Launch: The team wants to know the best day of the week to launch ads. Determine the day of the week when most users register on Instagram. Provide insights on when to schedule an ad campaign.**

```
SELECT CASE
WHEN dayy=1 then "SUN"
when dayy=2 then "MON"
when dayy=3 then "TUE"
when dayy=4 then "WED"
when dayy=5 then "THU"
when dayy=6 then "FRI"
else "SAT"
end as day_in_week, no_of_registrations, concat(time_created,"
",case when time_created<12 then "AM" else "PM" end) as
avg_reg_time from(
SELECT
    DAYOFWEEK(created_at) AS dayy,
    truncate(avg(hour(created_at)),0) as time_created,
    COUNT(*) AS no_of_registrations
FROM
    users
GROUP BY dayy
ORDER BY no_of_registrations DESC)
as qq;
```

day_in_week	no_of_registrations	avg_reg_time
THU	16	13 PM
SUN	16	12 PM
FRI	15	16 PM
TUE	14	12 PM
MON	14	15 PM
WED	13	11 AM
SAT	12	9 AM

7 rows in set (0.01 sec)



From the query, we can see that the most registrations have happened on Thursday and Sunday, and hence the campaign can be scheduled on these days for maximum user engagement. Further, the campaign can be scheduled between 12:00 p.m. and 4:00 p.m. for maximum benefits, as on average, many users have registered between that timestamp, as evident from the queried table.

## Investor Metrics

**01 User Engagement: Investors want to know if users are still active and posting on Instagram or if they are making fewer posts. 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.**

### TO CALCULATE NUMBER OF POSTS PER USER:

```
select users.id, users.username, count(photos.id) as
no_of_posts_per_user from users left join photos on
photos.user_id=users.id group by users.id, users.username;
```

id	username	no_of_posts_per_user
1	Kenton_Kirlin	5
2	Andre_Purdy85	4
3	Harley_Lind18	4
4	Arely_Bogan63	3
5	Aniya_Hackett	0
6	Travon.Waters	5
7	Kassandra_Homenick	0
8	Tabitha_Schamberger11	4
9	Gus93	4
10	Presley_McClure	3
11	Justina.Gaylord27	5
12	Dereck65	4
13	Alexandro35	5
14	Jaclyn81	0
15	Billy52	4

**TO CALCULATE AVERAGE NUMBER OF POSTS:**

```
with pics as (select count(*) as no_of_pics from photos),
no_users as (select count(*) as no_of_user from users)
select no_of_pics, no_of_pics/no_of_user as result from
pics,no_users;
```

no_of_pics	result
257	2.5700

From the above queries, we see that about 74 users have been engaging on the platform well, posting photos now and then, and the 74 users have posted 257 posts in total. In order to increase user engagement, we need to make the 26 users who have never posted at all start posting so that our platform scales to a bigger level of success.

## 02 **Bots & Fake Accounts: Investors want to know if the platform is crowded with fake and dummy accounts. Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.**

```
with abc as(select user_id, count(*) as likes_per_user from likes group
by user_id), def as (select count(*) as no_of_pics from photos)
select users.username as
bot_account_name,abc.user_id,abc.likes_per_user from abc join users
on abc.user_id=users.id where likes_per_user=(select no_of_pics from
def);
```

bot_account_name	user_id	likes_per_user
Aniya_Hackett	5	257
Jaclyn81	14	257
Rocio33	21	257
Maxwell.Halvorson	24	257
Ollie_Ledner37	36	257
Mckenna17	41	257
Duane60	54	257
Julien_Schmidt	57	257
Mike.Auer39	66	257
Nia_Haag	71	257
Leslie67	75	257
Janelle.Nikolaus81	76	257
Bethany20	91	257

13 rows in set (0.02 sec)



There are 13 bot accounts on the platform, as these accounts have liked every single post on the platform, and we need to take measures regarding this as they may be a threat to other genuine users' safety on the site.

From the above queries, we see that about 74 users have been engaging on the platform well, posting photos now and then, and the 74 users have posted 257 posts in total. In order to increase user engagement, we need to make the 26 users who have never posted at all start posting so that our platform scales to a bigger level of success.

## 02 Bots & Fake Accounts: Investors want to know if the platform is crowded with fake and dummy accounts. Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.

```
with abc as(select user_id, count(*) as likes_per_user from likes group
by user_id), def as (select count(*) as no_of_pics from photos)
select users.username as
bot_account_name,abc.user_id,abc.likes_per_user from abc join users
on abc.user_id=users.id where likes_per_user=(select no_of_pics from
def);
```

bot_account_name	user_id	likes_per_user
Aniya_Hackett	5	257
Jaclyn81	14	257
Rocio33	21	257
Maxwell.Halvorson	24	257
Ollie_Ledner37	36	257
Mckenna17	41	257
Duane60	54	257
Julien_Schmidt	57	257
Mike.Auer39	66	257
Nia_Haag	71	257
Leslie67	75	257
Janelle.Nikolaus81	76	257
Bethany20	91	257

13 rows in set (0.02 sec)

## Further Insights

### 01 5 users who have the highest number of comments on a single post

with abc as (select comments.photo\_id as photo\_id, count(\*) as no\_of\_comments, photos.user\_id as userid from comments join photos on comments.photo\_id=photos.id group by comments.photo\_id, photos.user\_id order by no\_of\_comments desc)  
select users.username, abc.photo\_id, abc.no\_of\_comments from users join abc on users.id=abc.userid order by no\_of\_comments desc limit 5;

username	photo_id	no_of_comments
Harley_Lind18	13	39
Cesar93	157	39
Keenan.Schamberger60	247	39
Andre_Purdy85	8	38
Zack_Kemmer93	146	37

This result when compared with the result of 5 users with most liked post (Marketing analysis, question 3), we get that the user Zack\_Kemmer93 is on both the lists, which means his photos (photo id: 145 and 146) are one of the most liked and most commented posts.

username	like_count	photo_id
Zack_Kemmer93	48	145
Malinda_Streich	43	127
Adelle96	43	182
Seth46	42	123
Delpha.Kihn	41	61

The management team may acknowledge the user in some engaging way so that many other users get motivated as well to post more photos and those who never post at all may also get encouraged seeing this.

## 02 Users with most number of followers

```
select tag_id from photo_tags where photo_id=146;
with abc as (select followee_id user_id, count(*) as no_of_followers
from follows group by user_id) select users.username, abc.user_id,
abc.no_of_followers from abc join users on users.id=abc.user_id
where abc.no_of_followers=(select max(no_of_followers) from abc) order
by abc.no_of_followers desc;
```

username	user_id	no_of_followers
Kenton_Kirlin	1	77
Kasandra_Homenick	7	77
Eveline95	23	77
Tierra.Trantow	25	77
Jaime53	29	77
Pearl7	34	77
David.Osinski47	45	77
Morgan.Kassulke	49	77
Mariano_Koch3	51	77
Linnea59	53	77
Aurelie71	58	77
Cesar93	59	77
Florence99	64	77
Franco_Keebler64	68	77
Hulda.Macejkovic	74	77
Donald.Fritsch	77	77
Darby_Herzog	80	77
Esther.Zulauf61	81	77
Bartholome.Bernhard	83	77
Delfina_VonRueden68	86	77
Clint27	88	77
Jessyca_West	89	77
Esmeralda.Mraz57	90	77

23 rows in set (0.01 sec)

These 23 users have the most number of followers on the platform and hence they can be approached for any brand collaborations as they have about 77% reach (77 followers out of total 100 users) making the brand collaboration successful.

### 03 Most used comment texts

```
select comment_text,count(*) as count from comments group by
comment_text order by count(*) DESC limit 10;
```

comment_text	count
et et et	3
aut et et	2
voluptas suscipit quidem	2
et qui dolores	2
sed ut qui	2
rerum et omnis	2
qui explicabo qui	2
et rerum velit	2
excepturi consequatur qui	2
doloremque sit totam	2

10 rows in set (0.02 sec)

### 04 Most active hours on instagram based on post timestamps

```
select hour(created_dat) as hour_posted, count(*) as post_count from
photos group by hour_posted order by post_count desc;
```

hour_posted	post_count
9	257

We see that all the posts have been created at 9:00 a.m.

### 05 Most active day on instagram based on likes timestamps

```
SELECT CASE
WHEN dayy=1 then "SUN"
when dayy=2 then "MON"
when dayy=3 then "TUE"
when dayy=4 then "WED"
when dayy=5 then "THU"
```

```

when day=6 then "FRI"
else "SAT"
end as day_in_week, no_of_likes from(
SELECT
    DAYOFWEEK(created_at) AS day,
    COUNT(*) AS no_of_likes
FROM
    likes
GROUP BY day
ORDER BY no_of_likes DESC)
as qq;

```

day_in_week	no_of_likes
TUE	8782

Hence, there has been 8782 likes on tuesday making tuesday the most active day.

## 05 Mutual followers

```

select a.follower_id user_1,b.follower_id user_2 from follows a join follows b
on a.follower_id=b.followee_id and a.followee_id=b.follower_id;

```

## 06 Likes to comments ratio

```

with abc as(
select photos.id as photo_id,
(select count(*) from likes where likes.photo_id=photos.id) as like_count,
(select count(*) from comments where comments.photo_id=photos.id) as
comment_count
from photos)
select abc.photo_id, abc.like_count,
abc.comment_count,abc.like_count/abc.comment_count as ratio from abc
WHERE like_count>0 and comment_count>0
order by ratio desc;

```

photo_id	like_count	comment_count	ratio
145	48	54	0.8889
61	41	48	0.8542
144	39	46	0.8478
199	39	46	0.8478
16	37	44	0.8409
230	37	44	0.8409
205	39	48	0.8125
69	38	48	0.7917
197	38	48	0.7917
17	36	46	0.7826
101	36	46	0.7826
182	43	56	0.7679
44	39	52	0.7500
106	36	48	0.7500
107	39	52	0.7500
211	36	48	0.7500
257	36	48	0.7500
57	37	50	0.7400
59	37	50	0.7400
87	37	50	0.7400
248	37	50	0.7400
23	38	52	0.7308

From the query, it is evident that for majority of the posts, the number of comments is greater than the number of likes. More users are interested to comment rather than liking. The lowest like : comment ratio is 0.3553 with 27 likes and 76 comments and the highest ratio is 0.8889 with 48 likes and 54 comments.

## 07 Frequency of tags per user

```
select users.username,tags.tag_name,count(*) as tag_freq from users join photos
on users.id=photos.user_id
join photo_tags on photos.id=photo_tags.photo_id join tags on
tags.id=photo_tags.tag_id group by users.id,tags.id order by tag_freq desc limit 5;
```

username	tag_name	tag_freq
Clint27	beach	6
Eveline95	smile	5
Clint27	dreamy	4
Andre_Purdy85	fun	3
Eveline95	hair	3

5 rows in set (0.01 sec)



## 08 Most tagging users

```
select users.username, count(photo_tags.photo_id) as tag_count from users join
photos on users.id=photos.user_id join photo_tags on
photo_tags.photo_id=photos.id group by users.id order by tag_count desc;
```

username	tag_count
Clint27	26
Eveline95	24
Aurelie71	20
Kenton_Kirlin	18
Delfina_VonRueden68	17
Cesar93	16
Adelle96	15
Jaime53	15
Donald.Fritsch	15
Tabitha_Schamberger11	13

These users have been utilizing tags the most on the platform.

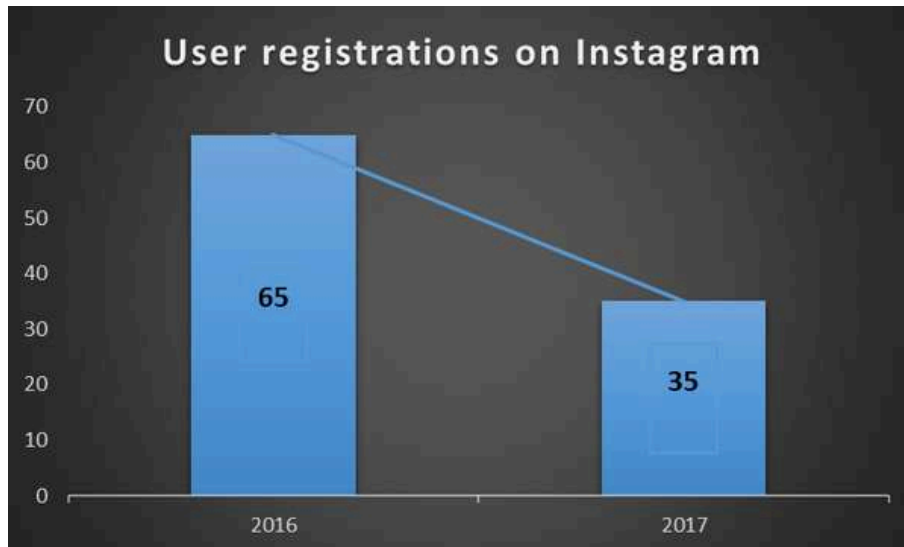
## 09 Users who have made most number of comments

```
select user_id,count(*) as no_of_comments from comments group by user_id
order by no_of_comments desc;
```

user_id	no_of_comments
21	257
54	257
5	257
76	257
75	257
91	257
36	257
14	257
71	257
66	257
41	257
57	257
24	257
17	83
63	80
15	77
84	76
67	76
96	75
87	74
60	72
43	72
100	70

The users whose number of comments is 27 can be potential bot accounts and require the management's attention if necessary.

## 12 Users Registration Trends



The number of new users on the platform has reduced drastically by about 53% and it is therefore essential for the marketing team and advertising team to attract more users with efficient strategies and promotional activities.



## Result



From this project, I have gained hands-on experience on real-world data analytics problems. I have implemented the data analysis using basic to advanced SQL concepts to derive answers to the questions asked, as well as I have made my own observations to provide valuable insights to help the management team make data-driven decisions.

I hope my insights serve useful to the decision-makers.

It was a wonderful opportunity to explore various ways to manipulate data, and the real-world example just made it an even better experience. I have learnt how to make efficient data queries using common table expressions, subqueries, and joins, especially.