Instagram User Analytics

Instagram User Analytics refers to the measurement, collection, analysis, and interpretation of data related to an Instagram user's account and activity. This data can provide insights into the user's audience demographics, engagement rate, content performance, and growth over time. Businesses and individuals can use Instagram analytics to better understand their followers and make informed decisions about their content strategy. Some popular Instagram analytics tools include Iconosquare, Hootsuite, and Sprout Social.

Here is a report on my findings in the Instagram clone dataset given for the 2nd project-

MARKETING:

TASK 1: 5 oldest users of the Instagram from the database provided-

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-

```
SELECT * FROM users ORDER BY created_at LIMIT 5;
```

TASK 2: The users who have never posted a single photo on Instagram - 26

id	username	created_at
5	Aniya_Hackett	2016-12-07 01:04:39
7	Kasandra_Homenick	2016-12-12 06:50:08
14	Jaclyn81	2017-02-06 23:29:16
21	Rocio33	2017-01-23 11:51:15
24	Maxwell.Halvorson	2017-04-18 02:32:44
25	Tierra.Trantow	2016-10-03 12:49:21
34	Pearl7	2016-07-08 21:42:01
36	Ollie_Ledner37	2016-08-04 15:42:20
41	Mckenna17	2016-07-17 17:25:45
45	David.Osinski47	2017-02-05 21:23:37
49	Morgan.Kassulke	2016-10-30 12:42:31
53	Linnea59	2017-02-07 07:49:34
54	Duane60	2016-12-21 04:43:38
57	Julien_Schmidt	2017-02-02 23:12:48
66	Mike.Auer39	2016-07-01 17:36:15
68	Franco_Keebler64	2016-11-13 20:09:27
71	Nia_Haag	2016-05-14 15:38:50
74	Hulda.Macejkovic	2017-01-25 17:17:28
75	Leslie67	2016-09-21 05:14:01
76	Janelle.Nikolaus81	2016-07-21 09:26:09
80	Darby_Herzog	2016-05-06 00:14:21
81	Esther.Zulauf61	2017-01-14 17:02:34
83	Bartholome.Bernhard	2016-11-06 02:31:23
89	Jessyca_West	2016-09-14 23:47:05
90	Esmeralda.Mraz57	2017-03-03 11:52:27
91	Bethany20	2016-06-03 23:31:53

Code-

```
SELECT id FROM users WHERE id NOT IN (SELECT user_id FROM photos GROUP BY user_id);
```

TASK 3: The winner of the contest is -

id	username	created_at
52	Zack_Kemmer93	2017-01-01 05:58:22

With 48 likes(maximum likes) on image –

id	image_url	user_id	created_dat
145	https://jarret.name	52	2023-02-06 11:47:27

Code-

```
CREATE TEMPORARY TABLE temp1 AS (SELECT user_id, photo_id FROM likes ORDER BY
photo_id ASC);
CREATE TEMPORARY TABLE temp2 AS (SELECT COUNT(user_id) AS likes, photo_id FROM
temp1 GROUP BY photo_id);
SET @maxlikes := (SELECT MAX(likes) FROM temp2);
SET @photoid := (SELECT photo_id FROM temp2 WHERE likes = @maxlikes);
SELECT * FROM users WHERE id = (SELECT user_id FROM photos WHERE id = @photoid);
```

TASK 4: The top 5 most commonly used hashtags on the platform are-

id	tag_name	created_at
13	fun	2023-02-06 11:47:28
17	party	2023-02-06 11:47:28
18	concert	2023-02-06 11:47:28
20	beach	2023-02-06 11:47:28
21	smile	2023-02-06 11:47:28

Code-

```
CREATE TEMPORARY TABLE temp3 AS (SELECT COUNT(photo_id) AS tags_count, tag_id FROM
photo_tags GROUP BY tag_id);
CREATE TEMPORARY TABLE temp4 AS (SELECT tag_id FROM temp3 ORDER BY tags_count
DESC LIMIT 5);
SELECT * FROM tags WHERE id IN (SELECT tag_id FROM temp4);
```

TASK 5: Ad campaign should be scheduled on the days of the week on which most users register on are-

MostNewUsers
Thursday
Sunday

Code-

```
CREATE TEMPORARY TABLE temp5 AS (SELECT
DAYOFWEEK(SUBSTRING_INDEX(created_at, ' ', 1)) AS c_day FROM users);
CREATE TEMPORARY TABLE temp6 AS (SELECT c_day, COUNT(c_day) as count FROM temp5
GROUP BY c_day);

SET @maxcount := (SELECT MAX(count) FROM temp6);
CREATE TEMPORARY TABLE temp7 AS (SELECT c_day AS day FROM temp6 WHERE count =
@maxcount);
CREATE TEMPORARY TABLE days_of_week (id INT AUTO_INCREMENT PRIMARY KEY,name
VARCHAR(20));
INSERT INTO days_of_week (name) VALUES
('Sunday'),('Monday'),('Tuesday'),('Wednesday'),('Thursday'),('Friday'),('Saturday');

SELECT name FROM temp7 JOIN days_of_week ON id = day;
```

INVESTOR METRICS:

TASK 1: User Engagement: Are users still as active and post on Instagram or they are making fewer posts? Provide how many times does average user posts on Instagram. Also, provide the total number of photos on Instagram/total number of users-

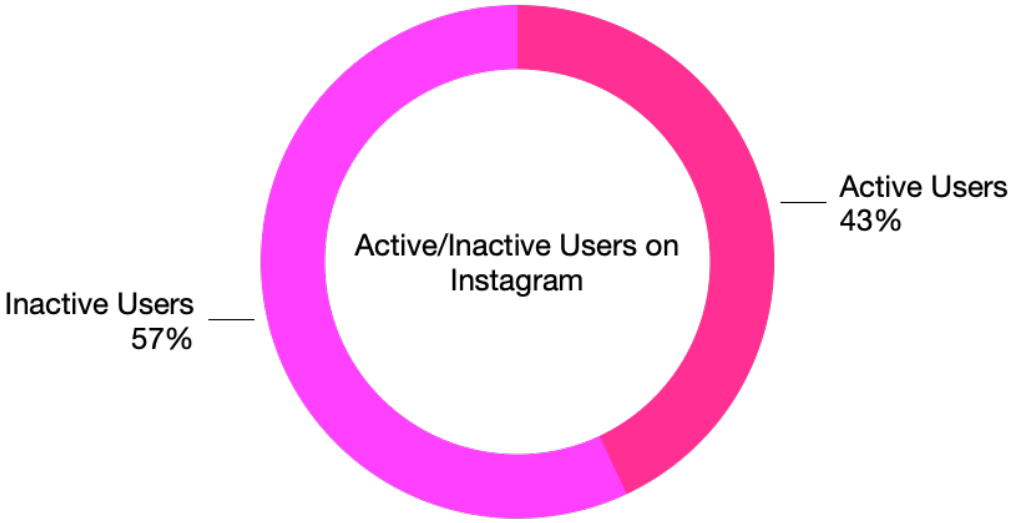
Average posts of a person on Instagram : 3

Average likes on a photo on Instagram : 34

Total Users on Instagram : 100

Total photos posted on Instagram : 257

Total Photos/Total Users : 2.57



Code-

```
CREATE TEMPORARY TABLE temp8 AS (SELECT COUNT(id) as c_posts, user_id FROM photos
GROUP BY user_id);
SET @average_posts := (SELECT ROUND(AVG(c_posts)) FROM temp8);
SET @active_users := (SELECT COUNT(user_id) FROM temp8 WHERE c_posts>=@average_posts);
SET @total_users := (SELECT COUNT(id) FROM users);
SET @total_photos := (SELECT COUNT(id) FROM photos);

CREATE TEMPORARY TABLE temp9 (SELECT COUNT(user_id) AS likes, photo_id FROM likes
GROUP BY photo_id);
SET @avg_likes := (SELECT ROUND(AVG(likes)) FROM temp9);

SELECT @average_posts;
SELECT @avg_likes;
SELECT @total_users;
SELECT @total_photos;
SELECT @total_photos/@total_users;
SELECT @active_users;
```

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

Suspected Bots on Instagram : 13

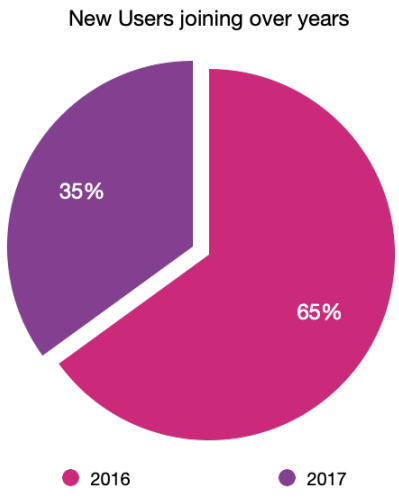
id	username	created_at
5	Aniya_Hackett	2016-12-07 01:04:39
14	Jaclyn81	2017-02-06 23:29:16
21	Rocio33	2017-01-23 11:51:15
24	Maxwell.Halvorson	2017-04-18 02:32:44
36	Ollie_Ledner37	2016-08-04 15:42:20
41	Mckenna17	2016-07-17 17:25:45
54	Duane60	2016-12-21 04:43:38
57	Julien_Schmidt	2017-02-02 23:12:48
66	Mike.Auer39	2016-07-01 17:36:15
71	Nia_Haag	2016-05-14 15:38:50
75	Leslie67	2016-09-21 05:14:01
76	Janelle.Nikolaus81	2016-07-21 09:26:09
91	Bethany20	2016-06-03 23:31:53

Code-

```
CREATE TEMPORARY TABLE temp10 AS (SELECT COUNT(photo_id) as c_liked, user_id FROM
likes GROUP BY user_id);
SET @total_photos := (SELECT COUNT(id) FROM photos);
CREATE TEMPORARY TABLE temp11 AS (SELECT user_id FROM temp10 WHERE
c_liked=@total_photos);
SET @bot_users := (SELECT count(user_id) FROM temp11);
SELECT * FROM users WHERE id IN (SELECT * FROM temp11);
SELECT @bot_users;
```

OTHER:

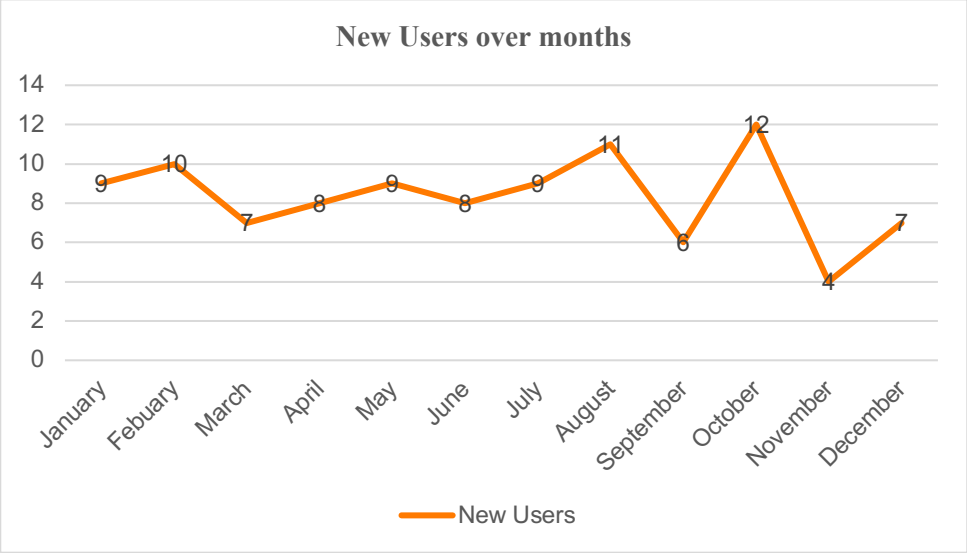
- New Users creating account –
    1. Yearly analysis:



Code-

```
CREATE TEMPORARY TABLE temp12 (SELECT SUBSTRING_INDEX(created_at, '-', 1) AS c_year
FROM users);
SELECT COUNT(c_year), c_year FROM temp12 GROUP BY c_year;
```

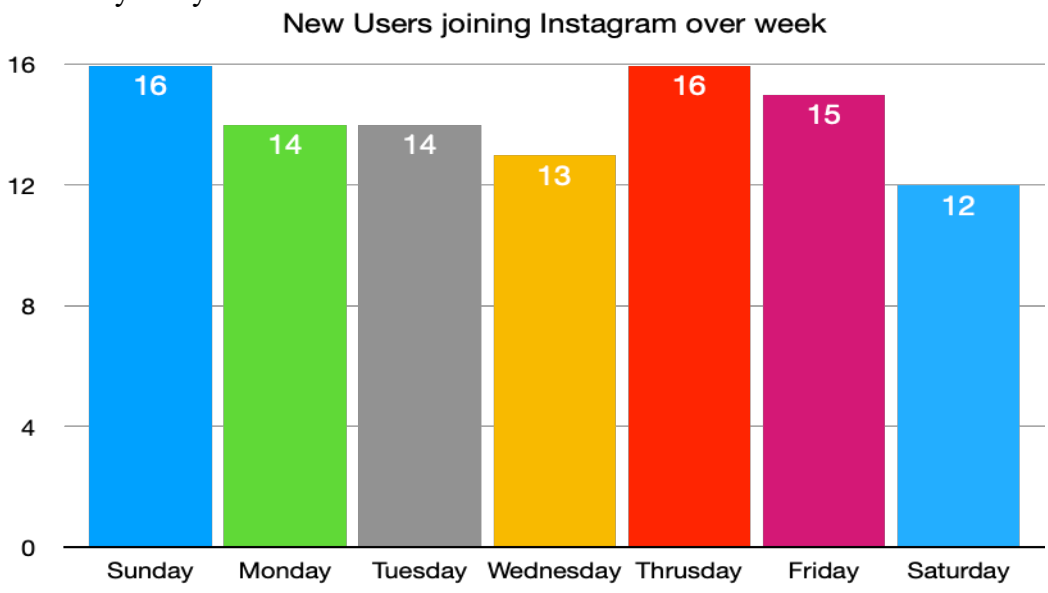
2. Monthly analysis:



Code-

```
CREATE TEMPORARY TABLE temp13 (SELECT MONTH(SUBSTRING_INDEX(created_at, ' ', 1))
AS c_month FROM users);
SELECT COUNT(c_month), c_month FROM temp13 GROUP BY c_month ORDER BY c_month;
```

3. Weekly analysis:



Code-

```
CREATE TEMPORARY TABLE temp14 AS (SELECT
DAYOFWEEK(SUBSTRING_INDEX(created_at, ' ', 1)) AS c_day FROM users);
CREATE TEMPORARY TABLE temp15 AS (SELECT c_day, COUNT(c_day) as count FROM temp14
GROUP BY c_day);
SELECT * FROM temp15;
```

- Most liked photos on Instagram –

photo_id	image_url	user_id	created_dat	likes
145	https://jarret.name	52	2023-02-06 11:47:27	48
127	https://celestine.name	46	2023-02-06 11:47:27	43
182	https://dorcas.biz	65	2023-02-06 11:47:27	43
123	http://shannon.org	44	2023-02-06 11:47:27	42
61	https://dejon.name	20	2023-02-06 11:47:27	41

Code-

```
CREATE TEMPORARY TABLE temp16 (SELECT COUNT(user_id) AS likes, photo_id FROM likes
GROUP BY photo_id);
CREATE TEMPORARY TABLE temp17 (SELECT * FROM temp16 ORDER BY likes DESC LIMIT 5);
SELECT photo_id, image_url, user_id, created_dat, likes FROM photos JOIN temp17 ON id = photo_id
ORDER BY likes DESC, id;
```

- Most liked(popular) and active people based on likes and posts –

user_id	username	created_at	total_likes	posts
23	Eveline95	2017-01-23 23:14:19	420	12
88	Clint27	2016-06-02 21:40:10	361	11
59	Cesar93	2016-10-18 16:42:43	338	10
86	Delfina_VonRueden68	2017-03-21 12:02:14	285	9
58	Aurelie71	2016-05-31 06:20:57	280	8

Code-

```
CREATE TEMPORARY TABLE temp18 (SELECT likes,user_id FROM temp16 a JOIN photos b
WHERE photo_id=id);
SELECT SUM(likes),user_id FROM temp18;
CREATE TEMPORARY TABLE temp19 (SELECT user_id, SUM(likes) as likes, COUNT(photo_id) as
posts FROM temp18 GROUP BY user_id);
CREATE TEMPORARY TABLE temp20 (SELECT * FROM temp19 ORDER BY likes DESC LIMIT 5);
SELECT user_id, username, created_at, likes as total_likes, posts FROM users a JOIN temp20 b WHERE
id=user_id;
```

←----------→