Instagram User Analytics

PROJECT NO.2

NAME :- Ayush Ravindra Vaidande

DESCRIPTION

- In this project we are going to use SQL to perform various Data analysis

  to give us insights in the given dataset.
- Here i have to give various insights about the dataset to the product team of instagram and answer the given questions assigned by product management team.
- Here we are going to use MySQL workshop to perform data analysis on given dataset.
- During the analysis we are going to find insights on dataset about their oldest user on instagram, user inactivity,contest winner, hashtag researching,AD campaigns,user engagement,bots and fake account.

APPROACH

- Here we are going to use SQL to perform data analysis for software we are going to use MySQL workbench
- We are going to download the given dataset and then load it in MySQL workbench
- Then we are going to look at questions assigned by product manager team

  and then we will find the best approach for the given question. Then we will run SQL commands on the following dataset.
- After extracting the information we will send the insights to the product manager team of instagram.

Tech-Stack Used

- MySQL workbench version- 8.0.2

  I used MySQL workbench to run SQL commands , by first uploading the dataset to MySQL workbench and then running the SQL commands to gain insight’s in the dataset.

INSIGHTS(Marketing)

1st QUESTION :- Rewarding Most Loyal Users: People who have been using the platform for the longest time.

-My Task: Find the 5 oldest users of the Instagram from the database provided

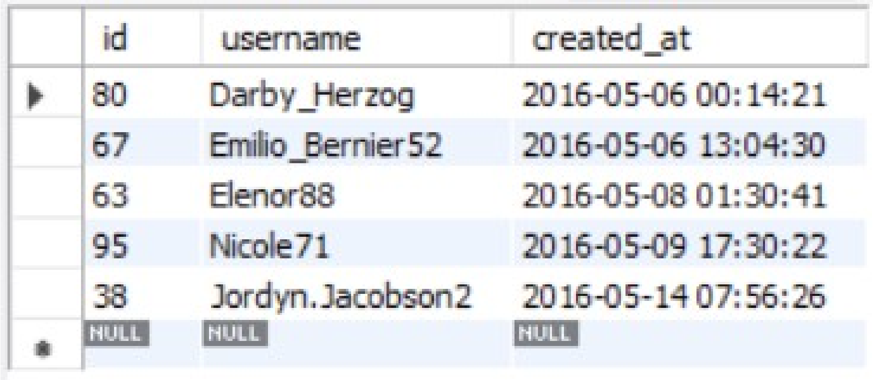
- Here we have executed the following SQL command :-

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

- SELECT * – ‘*’ to show all columns
- FROM users – using table *‘users’*
- ORDER BY created_at – sorting by ascending order by

  column *‘created_at’*
- LIMIT 5 – showing only top 5 rows by setting value by 5.

OUTPUT



	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

INSIGHTS :-

Darby_Herzog is the 1st and oldest instagram user and then at
2nd Emilio_Bernier52, 3rd Elenor88, 4th Nicole71,
5th Jordyn.Jacobson2

2nd QUESTION :- Remind Inactive Users to Start Posting: By sending them promotional emails to post their 1st photo.

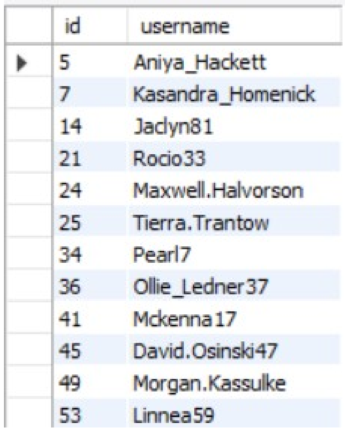
-My Task: Find the users who have never posted a single photo on Instagram

- Here we have executed the following SQL command :-

```
SELECT id, username FROM users WHERE id NOT IN (SELECT user_id FROM photos)
```

SELECT id, username	– showing columns *id, username*
FROM users	– using table *‘users’*
WHERE id NOT IN	– filtering records where *id* is not present
SELECT user_id FROM photos	– subquery to extract *user_id* from table *photos*

OUTPUT



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

INSIGHTS :-

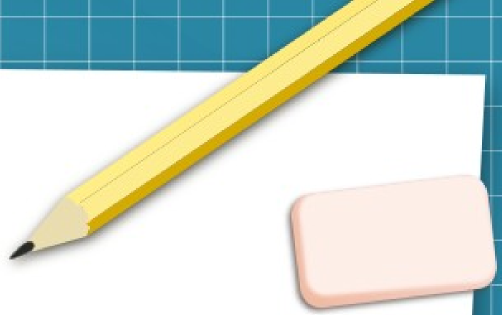
Here as you can see in the table we have all the inactive users whom we have to send promotional email to post their first photo.

- 3rd Question :- Declaring Contest Winner: 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.

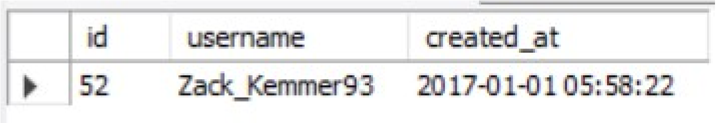
  - My Task: Identify the winner of the contest and provide their details to the team

- Here we have executed the following SQL command :-

select u.id, u.username, created_at	– showing columns u.id, u.username
from users u join photos p on u.id = p.user_id	– using table users and photos by joining them using JOIN function on columns id and user_id.
where p.id =	– filtering records where id is present
(select photo_id from likes	
join photos on likes.photo_id = photos.id	
group by photo_id order by count(photo_id) desc limit 1)	– subquery to extract photo_id from likes and photos table by joining them using JOIN function and sorting by descending order by count of photo_id and showing only the first record by using LIMIT 1



OUTPUT



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

INSIGHTS :-

Here the user Zack_kemmer93 has highest number of likes on a single photo ,so he is the winner of the contest

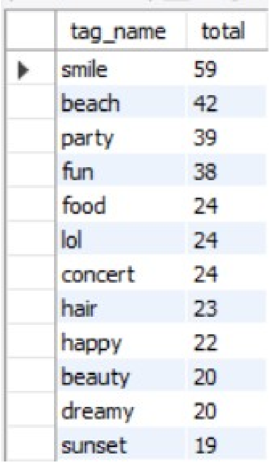
- 4th QUESTION :- Hashtag Researching: A partner brand wants to know, which hashtags to use in the post to reach the most people on the platform.

  - My Task: Identify and suggest the top 5 most commonly used hashtags on the platform

- Here we have executed the following SQL command :-

```
SELECT tag_name, COUNT(tag_name) AS total                – showing columns tag_name and count of
                                                           tag_name named  as total
FROM tags JOIN photo_tags ON tags.id = photo_tags.tag_id – using table tags and photo_tags
                                                           by joining them using JOIN function on
                                                           column id and tag_id
GROUP BY tags.id                                          – grouping rows by column id
ORDER BY total DESC                                       – sorting in descending order by total
```

OUTPUT



	tag_name	total
▶	smile	59
	beach	42
	party	39
	fun	38
	food	24
	lol	24
	concert	24
	hair	23
	happy	22
	beauty	20
	dreamy	20
	sunset	19

INSIGHTS :-

Here in the given table :- 1.smile
2.beach
3.party
4.fun
5.food
Are the most commonly used hastags on the platform

- 5th QUESTION :- Launch AD Campaign: The team wants to know, which day would be the best day to launch ADs.

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

- Here we have executed the following SQL command :-

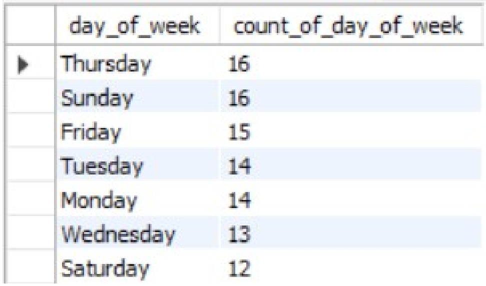
select date_format(created_at,'%W') as day_of_week, count(dayofweek(created_at)) as count_of_day_of_week – showing columns day_of_week (by extracting day from the created_at) and count_of_day_of_week(by using COUNT function on day_of_week)

from users – using table users

group by day_of_week – grouping rows by day_of_week

order by count_of_day_of_week desc – sorting in descending order by count_of_day_of_week

OUTPUT



day_of_week	count_of_day_of_week
Thursday	16
Sunday	16
Friday	15
Tuesday	14
Monday	14
Wednesday	13
Saturday	12

INSIGHTS :- According to the insights most users register on Thursday, so Thursday will be the best day to launch an AD campaign

INSIGHTS(Investor Metrics)

- 1st QUESTION :- User Engagement: Are users still as active and post on Instagram or they are making fewer posts

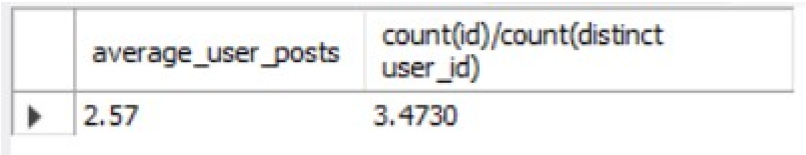
  - My task : Provide how many times does average user posts on Instagram. Also, provide the total number of photos on Instagram/total number of users

- Here we have executed the following SQL command :-

```
SELECT ROUND((SELECT COUNT(*)FROM photos)/(SELECT COUNT(*) FROM users),2) as
average_user_posts, count(id)/count(distinct user_id) from photos
```

using subquery to extract the count of records from photos table divide by count of records from users table and rounding them by 2 decimal places and naming it by average_user_posts. Then showing count of id divide by distinct user_id from photos table.

OUTPUT



	average_user_posts	count(id)/count(distinct user_id)
▶	2.57	3.4730

INSIGHTS :- so as you can see in the table the average number of user post on instagram is 2.57

- 2nd QUESTION :- Bots & Fake Accounts: The investors want to know if the platform is crowded with fake and dummy accounts

  - My 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).

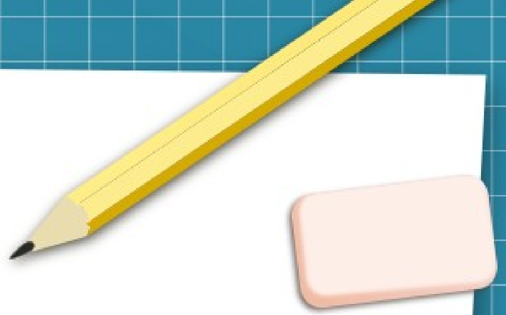
- Here we have executed the following SQL command :-

```
SELECT users.id,username, COUNT(users.id) As total_likes_by_user
```
– showing columns id, username and count of id named as total_likes_by_user

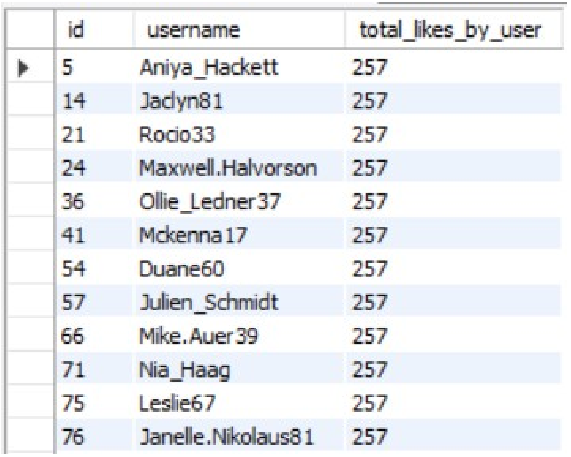
```
FROM users JOIN likes ON users.id = likes.user_id
```
– using table users and likes by joining them using JOIN function on columns id and user_id

```
GROUP BY users.id
```
– grouping rows by id

```
HAVING total_likes_by_user = (SELECT COUNT(*) FROM photos)
```
– showing only those records where total_likes_by_user is equal to count of records from photos table.



OUTPUT



id	username	total_likes_by_user
5	Aniya_Hackett	257
14	Jaclyn81	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

INSIGHTS :-

Here during insights we have found these users who have liked each and every single photo on the site. So we can consider the user accounts as bots.

THE RESULT

- While solving the given problems i have learned a lot about Data analysis like finding insights on given dataset, using SQL commands to extract whatever information we want from given data , i learned to solve real-world tasks and i learned lot about SQL.

THE END