

Instagram User Analytics

Description:

I assisted in the attempt to generate business insights for the marketing, product, and development teams in this project concerning Instagram user analysis by sending the data to them using SQL (structured query language). During this process, I'm using various SQL queries to collect the needed data. By using the data provided, I was able to determine the top-ranked oldest user, whether the user had posted any pictures or not, the most popular hashtag, the total number of users, and many other insights about Instagram.

Approach:

I first analyzed the objective and looked for the actual data that the team required, and then I imported the data into SQL and ran numerous queries to understand the data and discover the insights that the team need for business benefits.

Tech Stack Used:

DB Fiddle

Version – My SQL 8.0

I ran various SQL commands to gain insights and gain understanding of how to do real-time SQL queries on this project. I deduced several insights from the data provided by Instagram users, including the oldest users, users who never posted a single photo, the contest winner, the most commonly used hashtags, the day of the week on which most users registered, how many times the average user posts, and the bot accounts.

Result:

A) Marketing:

1) Rewarding Most Loyal Users: People who have been using the platform for the longest time. Task: Find the 5 oldest users of the Instagram from the database provided.

Query:

```
1 SELECT
2 username,
3 created_at
4 FROM
5 ig_clone.users
6 ORDER BY
7 created_at
8 LIMIT 5
```

Output:

username	created_at
Darby_Herzog	2016-05-06 00:14:21
Emilio_Bernier52	2016-05-06 13:04:30
Elenor88	2016-05-08 01:30:41
Nicole71	2016-05-09 17:30:22
Jordyn.Jacobson2	2016-05-14 07:56:26

2) Remind Inactive Users to Start Posting: By sending them promotional emails to post their first photo.

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

Query:

```
1 select
2 u.username
3 from
4 ig_clone.users u
5 left join
6 ig_clone.photos p
7 on u.id = p.user_id
8 where
9 p.user_id is null
10 order by
11 u.username;
```

Output:

Aniya_Hackett	Julien_Schmidt
Bartholome.Bernhard	Kasandra_Homenick
Bethany20	Leslie67
Darby_Herzog	Linnea59
David.Osinski47	Maxwell.Halvorson
Duane60	Mckenna17
Esmeralda.Mraz57	Mike.Auer39
Esther.Zulauf61	Morgan.Kassulke
Franco_Keebler64	Nia_Haag
Hulda.Macejkovic	Ollie_Ledner37
Jaclyn81	Pearl7
Janelle.Nikolaus81	Rocio33
Jessyca_West	Tierra.Trantow

3) 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.

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

Query:

```
1 with base as ( select likes.photo_id ,
2                 users.username ,
3                 count(likes.User_id) as like_User
4                 from ig_clone.likes likes
5                 inner join
6                 ig_clone.photos photos on
7                 likes.photo_id = photos.id
8                 inner join
9                 ig_clone.users users on
10                photos.user_id = users.id group by
11                likes.Photo_id,
12                users.username order by Like_user desc limit 1 )
13 select username from base;
```

Output:

username
Zack_Kemmer93

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

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

Query:

```
1 SELECT t.tag_name,
2 COUNT(p.photo_id)AS num_tags
3 FROM ig_clone.photo_tags p INNER JOIN
4 ig_clone.tags t ON p.tag_id = t.id
5 GROUP BY tag_name ORDER BY
6 num_tags DESC LIMIT 5
```

Output:

tag_name	num_tags
smile	59
beach	42
party	39
fun	38
concert	24

5) Launch AD Campaign: The team wants to know, which day would be the best day to launch ADs. Task: What day of the week do most users register on? Provide insights on when to schedule an ad campaign .

Query:

```
1 SELECT WEEKDAY(created_at)
2 AS weekday,COUNT(username)
3 AS num_users
4 FROM ig_clone.users GROUP
5 BY 1 ORDER BY 2 DESC
```

Output:

weekday	num_users
3	16
6	16
4	15
1	14
0	14
2	13
5	12

B) Investor metrics:

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

Query:

```
1 WITH CTE AS (SELECT u.id AS
2               userid, COUNT(p.id)
3               AS photoid FROM ig_clone.users u
4               LEFT JOIN
5               ig_clone.photos p ON u.id = p.user_id
6               GROUP BY u.id)
7   SELECT SUM(photoid) AS
8   total_photos,
9   COUNT(userid) AS total_users,
10  SUM(photoid)/COUNT(userid) AS
11  Photos_per_user FROM CTE
```

Output:

total_photos	total_users	Photos_per_user
257	100	2.5700

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

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).

Query:

```
1 WITH photo_count AS (SELECT
2   user_id,COUNT(photo_id) AS num_like
3   FROM ig_clone.likes GROUP BY
4   user_id ORDER BY num_like DESC)
5   SELECT * FROM photo_count WHERE num_like =
6   (SELECT count(*) FROM ig_clone.photos)
```

Output:

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