Instagram User Analytics

Project description:

This is a project about Instagram User analysis in which I helped the attempt to derive business insights for marketing, product & development teams by providing the data to them by using SQL(structured query language) On this process am using several SQL queries to get the desired data Through the project I founded the top oldest user, whether the user posted any picture or not, most commonly used hashtag, total number of users and many several insights of the Instagram from the data provided

Approach:

Firstly, I went through all the tables given in the dataset to get some clarity about what the data is about and what we could obtain or interpret from the given dataset. Then I imported the data in my Mysql software to understand the data further in order to find the insights of the data required for growth of the company

Tech stack used:

I used MySQL Workbench 8 0 CE

Insights:

I executed different queries to provide solution for the given tasks. I got to know how to use my sql skills to provide solutions for different questions of the company which would be very much useful for their further development of business

Result:

After I executed queries in my mysql, I took the results data and saved it

A) Marketing:

1) Rewarding Most loyal users:

People who have been using the platform for the longest time.

Query:

```
USE ig_clone;

SELECT username

FROM users

ORDER BY created_at

LIMIT 5;
```

Result:

username
Darby_Herzog
Emilio_Bernier52
Elenor88
Nicole71
Jordyn.Jacobson2

2) Remind Inactive users to start posting:

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

Query:

```
USE ig_clone;

SELECT u.username

FROM users u

LEFT JOIN photos p ON u.id=p.user_id

WHERE p.user_id IS NULL

ORDER BY u.username;
```

Result:

I exported the retrieved data in an excel sheet and took snapshot

username	username2
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**:

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

Query:

```
USE ig_clone;

SELECT users.username, COUNT(*) AS total_likes

FROM likes

JOIN photos ON photos.id=likes.photo_id

JOIN users ON users.id=likes.photo_id

GROUP BY photos.id

ORDER BY total_likes desc

LIMIT 1;
```

Result:

username	total_likes
Kaley9	41

4) **Hashtag researching**:

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

Query:

```
USE ig_clone;

SELECT t.tag_name,COUNT(p.photo_id) as num_tag

FROM photo_tags p

JOIN tags t ON p.tag_id=t.id

GROUP BY tag_name

ORDER BY num_tag DESC

LIMIT 5;
```

Result:

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

5) **Launch AD Campaign**:

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

Query:

```
USE ig_clone;

SELECT DAYNAME(created_at) AS Days,COUNT(*) AS total

FROM users

GROUP BY Days

ORDER BY total DESC

LIMIT 2;
```

Result:

Days	total
Thursday	16
Sunday	16

B) INVESTOR METRICS

1) User Engagement:

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

Query:

```
WITH CTE AS (SELECT u.id AS userid , COUNT(p.id) AS photoid

FROM users u

LEFT JOIN photos p ON u.id=p.user_id

GROUP BY u.id)

SELECT SUM(photoid) AS total photos ,COUNT(userid) AS total_users ,
SUM(photoid)/COUNT(userid)

FROM CTE
```

Result:

Result Grid | Filter Rows: | Export:

total_photos	total_users	photos_per_user
257	100	2.5700

2) Bots & Fake accounts:

Provide data on users (bots) who have liked every single photo on the site.

Query:

```
USE ig_clone;

WITH photo_count AS(SELECT user_id,COUNT(photo_id) AS num_like

FROM likes

GROUP BY user_id

ORDER BY num_like DESC)

SELECT *

FROM photo_count

WHERE num_like =(SELECT COUNT(*) FROM photos)
```

Result:

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