Instagram user analytics

Project Description:

The purpose of an Instagram user analytics project is to analyze and understand the behavior, preferences, and interactions of specific users on the platform. This type of project focuses on individual user-level data and provides insights that can be valuable for personalized content, targeted marketing, and enhancing user engagement.

Approach:

Our approach to this project involves utilizing SQL Workbench as our primary tool for data analysis. By harnessing the power of SQL queries, we can efficiently extract relevant information such as user engagement metrics, demographic insights, and content performance. This approach enables us to derive actionable insights and make informed decisions based on the data extracted from the Instagram database.

Tech-stack used:

MySQL Workbench 8.0 CE for my Instagram user analytics project is a sound decision due to its open-source nature, compatibility with MySQL databases, powerful SQL querying capabilities, and support for visualization and integration with other tools.

PROJECT INSIGHTS:

➢ MARKETING ANALYSIS

A) LOYAL USER REWARD: Users those who have been using the Instagram platform for the longest time.

Code :

```
SELECT username, created_at
FROM users
```

```
ORDER BY username, created_at ASC
LIMIT 5;
```

Result:

Username	Created_at
Adelle96	2016-10-01 00:37:57
Aiyana_Hoeger	2016-09-29 20:28:12
Alek_Watsica	2016-12-10 07:43:58
Alexandro35	2017-03-29 17:09:02
Alysa22	2017-01-01 17:44:43

B) INACTIVE USER ENGAGEMENT: Users who have never posted a single photo on Instagram.

Code:

```
SELECT username
FROM users
LEFT JOIN photos ON users.id = photos.user_id
WHERE photos.user_id IS NULL;
```

Result:

Aniya_Hackett
Kasandra_Homenick
Jaclyn81
Rocio33
Maxwell.Halvorson
Tierra.Trantow
Pearl7
Ollie_Ledner37
Mckenna17
David.Osinski47

Morgan.Kassulke
Linnea59
Duane60
Julien_Schmidt
Mike.Auer39
Franco_Keebler64
Nia_Haag
Hulda.Macejkovic
Leslie67
Janelle.Nikolaus81
Darby_Herzog
Esther.Zulauf61
Bartholome.Bernhard
Jessyca_West
Esmeralda.Mraz57
Bethany20

c) CONTEST WINNER ENGAGEMENT: The team has organize a contest where the user with the most likes on a single photo wins..

Code:

```
SELECT username, photos.id, photos.image_url,
COUNT(likes.user_id) as like_count
FROM photos
INNER JOIN likes ON likes.photo_id=photos.id
INNER JOIN users ON users.id=photos.user_id
GROUP BY photo_id
ORDER BY like_count DESC
LIMIT 1;
```

Result: The WINNER is

Username	id	image_url	like_count
Zack_Kemmer93	145	https://jarret.name	48

D) HASHTAGS RESEARCH: The most popular hashtags to be used by users in their posts.

Code:

```
SELECT  tag_name ,count(*) as hashtag_count
FROM tags
JOIN photo_tags ON photo_tags.tag_id= tags.id
GROUP BY tag_name
ORDER BY hashtag_count DESC
LIMIT 5;
```

Result :

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

E) AD COMPAIGN LAUNCH: The team wishes to know the day of the week when most users register on Instagram.

Code:

```
SELECT  DAYNAME(created_at) AS registration_day,
```

```
    COUNT(*) AS registration_count
FROM users
GROUP BY registration_day
ORDER BY registration_count DESC
limit 2;
```

Result: The days of the week when most of the users register on Instagram.

Registration_day	Registration_count
Thursday	16
Sunday	16

Sunday is preferred over Thursday for the ad campaign. Users may have more free time on weekends, leading to increased engagement.

➢ *Investor metrics*

1. USER ENGAGEMENT: Calculating the average number of posts per user on Instagram.

   Code:

```
SELECT AVG(photo_count) AS average_posts_per_user
FROM (
   SELECT user_id, COUNT(*) AS photo_count
   FROM photos
   GROUP BY user_id
```

)user_photos_count;

Result: The average is 3.4730.

Total number of photos on Instagram divided by total number of users is

Code:

```
SELECT COUNT(*) AS total_photos, COUNT(DISTINCT user_id) AS total_users,
    COUNT(*) / COUNT(DISTINCT user_id) AS average_photos_per_user
FROM photos;
```

Result:

Total_photos	Total_users	Average_posts_per_users
257	74	3.4730

2. BOTS AND FAKE ACCOUNTS: Investors want to know if the platform is crowded with fake and dummy accounts.

Users who have liked every single photo on the site

Code:

```
SELECT id, username
FROM users
```

```
JOIN (
    SELECT user_id
    FROM likes
    GROUP BY user_id
    HAVING COUNT(DISTINCT likes.photo_id) = (SELECT
COUNT(DISTINCT image_url) FROM photos)
) likes ON users.id = likes.user_id
ORDER BY username ASC;
```

Result:

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