Instagram User Analysis

Project Description

The aim of this project is to perform an in-depth analysis of Instagram user data to extract meaningful insights about user engagement, platform activity, and marketing opportunities. SQL was utilized as the primary tool for querying the database and retrieving insights. The findings will help businesses and marketers understand user behavior and optimize their strategies.

Approach

1. **Database Creation**: Created a relational database (ig_clone) to store user data, photos, likes, comments, and hashtags.
2. **Data Loading**: Inserted relevant records into the database tables.
3. **Data Analysis**: Ran SQL queries to extract insights about user activity, loyalty, engagement, and marketing trends.
4. **Insights Extraction**: Analyzed the data to generate business recommendations.

Tech-Stack Used

- **MySQL**: Chosen for its efficiency in handling structured data and relational queries.
- **MySQL Workbench**: Used for executing SQL queries and visualizing data.
- **Common Table Expressions (CTEs) & Aggregation Functions**: Used for efficient data retrieval and summarization.

A) Marketing Analysis

1. Rewarding the Most Loyal Users

Identified the five oldest users based on their account creation date.

SQL Query:

```
SELECT username, created_at
FROM users
ORDER BY created_at ASC
LIMIT 5;
```

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

Insight: These users have been on the platform the longest and can be rewarded for their loyalty.

2. Identifying Inactive Users

Found users who have never posted a photo.

SQL Query:

```
SELECT u.username,count (p.user_id)
FROM users AS u
LEFT JOIN photos AS p ON u.id = p.user_id
GROUP BY u.id
having count(p.user_id) = 0;
```

username	count(p.user_id)
Aniya_Hackett	0
Kasandra_Homenick	0
Jaclyn81	0
Rocio33	0
Maxwell.Halvorson	0
Tierra.Trantow	0
Pearl7	0
Ollie_Ledner37	0
Mckenna17	0
David.Osinski47	0
Morgan.Kassulke	0
Linnea59	0
Duane60	0
Julien_Schmidt	0
Mike.Auer39	0
Franco_Keebler64	0
Nia_Haag	0
Hulda.Macejkovic	0
Leslie67	0
Janelle.Nikolaus81	0
Darby_Herzog	0
Esther.Zulauf61	0
Bartholome.Bernhard	0
Jessyca_West	0
Esmeralda.Mraz57	0
Bethany20	0

Insight: These users can be sent promotional emails encouraging them to start posting.

Declaring the Contest Winner

Identified the user with the most likes on a single photo.

SQL Query:

```
SELECT u.username, p.id AS photo_id, COUNT(l.photo_id) AS total_likes
FROM users AS u
JOIN photos AS p ON u.id = p.user_id
JOIN likes AS l ON p.id = l.photo_id
GROUP BY u.username, p.id
ORDER BY total_likes DESC
LIMIT 1;
```

username	Photo_id	total_likes
Zack_Kemmer93	52	48

Insight: Zack_Kemmer93 is the winner of the contest.

4. Most Popular Hashtags

Determined the top five hashtags used.

SQL Query:

```
SELECT t.tag_name, COUNT(pt.tag_id) AS usage_count
FROM photo_tags AS pt
JOIN tags AS t ON pt.tag_id = t.id
GROUP BY t.tag_name
ORDER BY usage_count DESC
LIMIT 5;
```

Hashtag	Count
smile	59
beach	42
party	39
fun	38
concert	24

Insight: Brands can use these hashtags to increase engagement.

5. Best Day to Launch an Ad Campaign

Identified the weekday with the highest number of new user registrations.

SQL Query:

```
SELECT DAYNAME(created_at) AS day_of_week, COUNT(*) AS registrations
FROM users
GROUP BY day_of_week
ORDER BY registrations DESC
LIMIT 1;
```

days_of_week	registrations
Thursday	16

Insight: The best day to launch an ad campaign is **Thursday**.

B) Investor Metrics

6. User Engagement: Posts Per User

Calculated the average number of posts per user.

SQL Query:

```
SELECT (SELECT COUNT(*) FROM photos) / (SELECT COUNT(*) FROM users) AS avg_posts_per_user;
```

Avg_post_per_user
2.57

Insight: The average Instagram user posts **less than 3 times**, indicating that many users are not very active.

7. Detecting Potential Bot Accounts

Identified users who liked every single photo.

SQL Query:

```
SELECT u.user_id, u.username, COUNT(l.photo_id) AS total_likes
FROM users AS u
JOIN likes AS l ON u.id = l.user_id
GROUP BY u.user_id
HAVING total_likes = (SELECT COUNT(*) FROM photos);
```

user_id	username	total_likes
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
91	Bethany20	257

Insight: These accounts are likely **bots** because they have liked every photo on the platform.