Instagram User Analytics

Based on SQL fundamentals

Description

This projects aim is to analyze the Instagram user engagement and interactions to derive insights which can help the marketing, product, and investor teams to make data-driven decisions. Using MySQL Workbench, we will extract meaningful insights from Instagram's user data to improve user engagement, identifying trends, and detect potential fake accounts.

Approach

In this project, we will follow a structured SQL-based analytical approach-

- Create database – import the provided database into MySQL Workbench
- Data exploration – understanding the table structures and relationships.
- Query execution – run multiple SQL queries to answer the provided questions.
- Analysis – analysing the results and summarize the key insights.
- Report – document findings with SQL queries and Outputs.

Tech-Stack Used

In this project, we use SQL database tool which is MySQL Workbench version 8.0 CE, because it is efficient, interactive, and widely used over the world for structured data analysis.

SQL Tasks

A) Marketing Analysis:

Loyal User Reward

Task: Identify the five oldest users on Instagram from the provided database.

```
USE ig_clone;
SELECT username, created_at FROM users
ORDER BY created_at DESC
LIMIT 5;
```

Output

username	created_at
Justina.Gaylord27	2017-05-04 16:32:16
Travon.Waters	2017-04-30 13:26:14
Milford_Gleichner42	2017-04-30 07:50:51
Hailee26	2017-04-29 18:53:40
Maxwell.Halvorson	2017-04-18 02:32:44

Inactive User Engagement:
Task: Identify users who have never posted a single photo on Instagram.

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

Output

username
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

Contest Winner Declaration:
Task: Determine the winner of the contest who have maximum no. of likes on a single photo and provide their details to the team.

```
USE ig_clone;
SELECT users.id, users.username, photos.id as photo_id,image_url,
count(likes.user_id) AS total_likes
FROM photos
INNER JOIN likes ON photos.id=likes.photo_id
INNER JOIN users ON photos.user_id=users.id
GROUP BY users.id, photos.id, users.username
ORDER BY total_likes DESC
LIMIT 1;
```

Output

	id	username	photo_id	image_url	total_likes
▶	52	Zack_Kemmer93	145	https://jarret.name	48

above query is good if only single photo has maximum likes. but if we have more than one photo on which maximum likes are same then-

```
USE ig_clone;
with likes_rank as(
select users.id as User_ID, users.username as USERSNAME, photos.id as Photo_ID,
image_url, count(likes.user_id) AS total_likes,
rank() over(order by count(likes.user_id)desc) as rank_number
from photos
INNER JOIN likes ON photos.id=likes.photo_id
INNER JOIN users ON photos.user_id=users.id
GROUP BY users.id, photos.id, users.username)
select User_ID, USERSNAME, Photo_ID,image_url, total_likes from likes_rank
where rank_number=1;
```

Output

	User_ID	USERSNAME	Photo_ID	image_url	total_likes
▶	52	Zack_Kemmer93	145	https://jarret.name	48

Hashtag Research:

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

```
USE ig_clone;
SELECT tags.tag_name, count(photo_tags.photo_id) AS total
FROM tags
INNER JOIN photo_tags
ON photo_tags.tag_id=tags.id
GROUP BY tags.tag_name
ORDER BY total DESC
LIMIT 5;
```

Output

	tag_name	total
▶	smile	59
	beach	42
	party	39
	fun	38
	concert	24

here are some other tags which is used by the same no. of time as others. To find those tags, we proceed with

```
USE ig_clone;
with trending_tags as(
select tags.tag_name as tag_names, count(photo_tags.photo_id) as total,
rank() over (order by count(photo_tags.photo_id) desc) as tag_rank
from photo_tags
inner join tags
on tags.id=photo_tags.tag_id
group by photo_tags.tag_id)
select tag_names, total from trending_tags
where tag_rank between 1 and 5;
```

Output

	tag_names	total
▶	smile	59
	beach	42
	party	39
	fun	38
	concert	24
	food	24
	lol	24

Ad Campaign Launch:

Task: Determine the day of the week when most users register on Instagram. Provide insights on when to schedule an ad campaign.

```
USE ig_clone;
with day_rank as(
select dayname(created_at) as registration_day,
count(id) as total_registered_user,
rank() over (order by count(id) desc) as rank_number
from users
group by registration_day)
select registration_day, total_registered_user
from day_rank
where rank_number=1;
```

Output

	registration_day	total_registered_user
▶	Thursday	16
	Sunday	16

B) Investor Metrics:

User Engagement:

Task: Calculate the average number of posts per user on Instagram. Also, provide the total number of photos on Instagram divided by the total number of users.

- average number of posts per user on Instagram

```
use ig_clone;
SELECT COUNT(id) / COUNT(DISTINCT user_id) AS avg_posts_per_user
FROM photos;
```

Output

	avg_posts_per_user
▶	3.4730

- total number of photos on Instagram divided by the total number of users.

```
USE ig_clone;
SELECT
    (SELECT COUNT(id) FROM photos) /
    (SELECT COUNT(id) FROM users) AS posts_per_user_ratio;
```

Output

	posts_per_user_ratio
▶	2.5700

Bots & Fake Accounts:

Task: Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.

```
USE ig_clone;
SELECT likes.user_id, users.username, count(likes.photo_id) AS total_likes
FROM likes
INNER JOIN users
ON users.id=likes.user_id
GROUP BY likes.user_id, users.username
HAVING total_likes=(SELECT count(id) FROM photos);
```

Output

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

Insights & Findings

- Oldest Users**:** Platform's oldest five users can be rewarded for loyalty.
- Inactive Users**:** Many users never post. They can be targeted for re-engagement campaigns.
- Contest Winner**:** User with the most-liked photo is identified as the winner.
- Best Ad Campaign Day**:** The day with the highest sign-ups is done, is best for marketing campaigns.
- Fake Accounts**:** Certain users have liked every post which is typically not possible, indicates bot activity.

Results and Recommendations

- **For Marketing Team**
  - Use hashtags from top 5 in for brand promotions
  - Target inactive users via email campaigns to boost engagement
  - Schedule apps promotional ads on the busiest registration day for maximum impact
- **For Product Team**
  - Monitor user engagement trends (post per user)
  - Encourage frequent posting via badges, leaderboards etc.
- **For Investor Metrics**
  - Minimise bot activity by implementing CAPTCHA for new accounts
  - Improve user retention by analyzing inactive users' behaviour