Instagram User Analytics

OUTLINE

PROJECT DESCRIPTION

APPROACH

TECH – STACK USED

INSIGHTS

RESULT

Project Description

- The Instagram Analytics project aims to analyze user engagement and detect potential fake accounts.
- By extracting and analyzing data from the Instagram database, we will provide insights on user activity, identify key patterns, and uncover unusual behaviors.
- This information will help stakeholders make informed decisions regarding user engagement, ad campaigns, and bot detection.

Approach

- To analyze the data and find answers to the questions, I executed targeted SQL queries.
- I identified the oldest users by sorting account creation dates and found users with no posts using left joins.
- I determined the most liked photo by counting likes and joining relevant tables.
- For the most popular hashtags, I counted occurrences and ranked them.
 Lastly, I identified the best day for ad campaigns by grouping and analyzing user registrations by the day of the week.

Tech - Stack Used

- For this project, I used **MySQL Ver 8.0.37**.
- MySQL is a reliable and widely-used database system, perfect for handling complex queries and large datasets. It's user-friendly with strong community support and offers an open-source platform that is cost-effective.



Insights

A) Marketing Analysis

1) Loyal User Reward:

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

SELECT

id, username

FROM

users

ORDER BY created_at

LIMIT 5

	id	username
•	80	Darby_Herzog
	67	Emilio_Bernier52
	63	Elenor88
	95	Nicole71
	38	Jordyn.Jacobson2

2) Inactive User Engagement:

Identify users who have never posted a single photo on Instagram

```
select

u.id, u.username , p.image_url

FROM

users u

LEFT JOIN

photos p ON u.id = p.user_id

WHERE

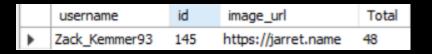
p.image_url IS NULL;
```

	id	username	image_url
•	5	Aniya_Hackett	NULL
	7	Kasandra_Homenick	NULL
	14	Jaclyn81	NULL
	21	Rocio33	NULL
	24	Maxwell.Halvorson	NULL
	25	Tierra.Trantow	NULL
	34	Pearl7	NULL
	36	Ollie_Ledner37	NULL
	41	Mckenna 17	NULL
	45	David.Osinski47	NULL
	49	Morgan.Kassulke	NULL
	53	Linnea59	NULL
	54	Duane60	NULL
	57	Julien_Schmidt	NULL
	66	Mike.Auer39	NULL
	68	Franco_Keebler64	NULL
	71	Nia_Haag	NULL
	74	Hulda.Macejkovic	NULL
	75	Leslie67	NULL
	76	Janelle.Nikolaus81	NULL
	80	Darby_Herzog	NULL
	81	Esther.Zulauf61	NULL
	81	Esther Zulauf61	NULL
	83	Bartholome.Bernhard	NULL
	89	Jessyca_West	NULL
	90	Esmeralda.Mraz57	NULL
	91	Bethany20	NULL

3) Contest Winner Declaration:

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

```
WITH MostLikedPhoto AS (
    SELECT photo_id, COUNT(*) AS TotalLikes
    FROM likes
    GROUP BY photo_id
    ORDER BY TotalLikes DESC
    LIMIT 1
SELECT
    u.username, p.id, p.image_url, COUNT(*) AS Total
    FROM
   MostLikedPhoto mlp
```



INNER JOIN

```
photos p ON mlp.photo_id = p.id
```

INNER JOIN

users u ON p.user_id = u.id

INNER JOIN

likes l ON p.id = l.photo_id

GROUP BY p.id, u.username, p.image_url;

4) Hashtag Research:

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

```
SELECT
    t.tag_name, COUNT(pt.tag_id) AS tag_count
FROM
    tags t
       INNER JOIN
    photo_tags pt ON t.id = pt.tag_id
GROUP BY t.tag_name
ORDER BY tag_count DESC
```

	tag_name	tag_count
•	smile	59
	beach	42
	party	39
	fun	38
	concert	24

5) Ad Campaign Launch:

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

```
DAYNAME(created_at) AS day_of_week,

COUNT(*) AS registrations

FROM

users

GROUP BY day_of_week

ORDER BY registrations DESC

LIMIT 2;
```

	day_of_week	registrations
•	Thursday	16
	Sunday	16

B) Investor Metrics

1) User Engagement:

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.

```
WITH post_counts AS(

SELECT user_id ,COUNT(*) num_posts

FROM photos

GROUP BY user_id)

SELECT

(SELECT

COUNT(*)

FROM

photos) /
```

```
(SELECT

COUNT(*)

FROM

users) AS avg_photos_per_user ,

AVG(num_posts) avg_no_posts

FROM

post_counts;
```

avg_no_posts

3,4730

2) Bots & Fake Accounts:

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

GROUP BY u.id

HAVING COUNT(l.photo_id) = (SELECT

count_of_photos

FROM

total_photos);

	id	username	bot_likes
•	5	Aniya_Hackett	257
	14	Jaclyn81	257
	21	Rocio33	257
	24	Maxwell.Halvorson	257
	36	Ollie_Ledner37	257
	41	Mckenna 17	257
	54	Duane60	257
	57	Julien_Schmidt	257
66	66	Mike.Auer39	257
	71	Nia_Haag	257
	75	Leslie67	257
	76	Janelle.Nikolaus81	257
	91	Bethany20	257

Result

- The project allowed me to refine my data analysis skills and demonstrated how SQL can reveal critical insights.
- By identifying user engagement patterns and potential fake accounts, I
 provided valuable information for optimizing strategies.
- The analysis helped stakeholders make informed decisions, improving platform effectiveness and user experience.