**Objective Questions**

1. **Are there any tables with duplicate or missing null values? If so, how would you handle them?**

**-- checking duplicates**

SELECT comment\_text, user\_id, photo\_id, created\_at

FROM comments

GROUP BY 1,2,3,4

HAVING COUNT(\*) > 1;

SELECT follower\_id, followee\_id

FROM follows

GROUP BY 1,2

HAVING COUNT(\*) > 1;

SELECT user\_id, photo\_id

FROM likes

GROUP BY 1,2

HAVING COUNT(\*) > 1;

SELECT photo\_id, tag\_id

FROM photo\_tags

GROUP BY 1,2

HAVING COUNT(\*) > 1;

SELECT image\_url, user\_id, created\_dat

FROM photos

GROUP BY 1,2,3

HAVING COUNT(\*) > 1;

SELECT tag\_name, created\_at

FROM tags

GROUP BY 1,2

HAVING COUNT(\*) > 1;

SELECT username, created\_at

FROM users

GROUP BY 1,2

HAVING COUNT(\*) > 1;

**-- NO DUPLICATE VALUES**

**-- Checking NULL Values**

SELECT \* FROM comments

WHERE 1 IS NULL OR 2 IS NULL OR 3 IS NULL OR 4 IS NULL OR 5 IS NULL ;

SELECT \* FROM follows

WHERE 1 IS NULL OR 2 IS NULL OR 3 IS NULL;

SELECT \* FROM likes

WHERE 1 IS NULL OR 2 IS NULL OR 3 IS NULL;

SELECT \* FROM photo\_tags

WHERE 1 IS NULL OR 2 IS NULL;

SELECT \* FROM photos

WHERE 1 IS NULL OR 2 IS NULL OR 3 IS NULL OR 4 IS NULL;

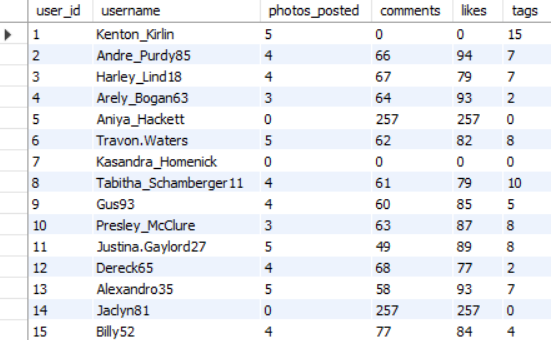
SELECT \* FROM tags

WHERE 1 IS NULL OR 2 IS NULL OR 3 IS NULL;

SELECT \* FROM users

WHERE 1 IS NULL OR 2 IS NULL OR 3 IS NULL;

**-- NO NULL VALUES**

1. **What is the distribution of user activity levels (e.g., number of posts, likes, comments) across the user base?**

SELECT

u.id AS user\_id,

username,

COUNT(DISTINCT p.id) AS photos\_posted,

COUNT(DISTINCT c.photo\_id) AS comments,

COUNT(DISTINCT l.photo\_id) AS likes,

COUNT(DISTINCT pt.tag\_id) AS tags

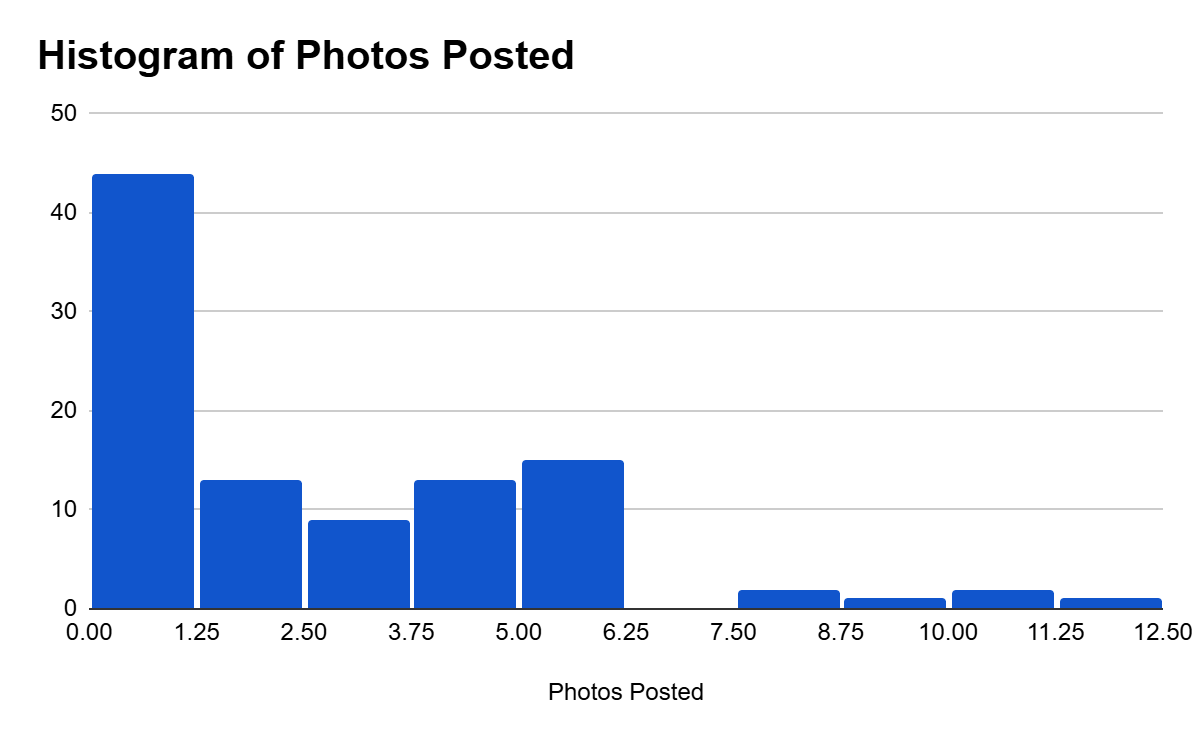
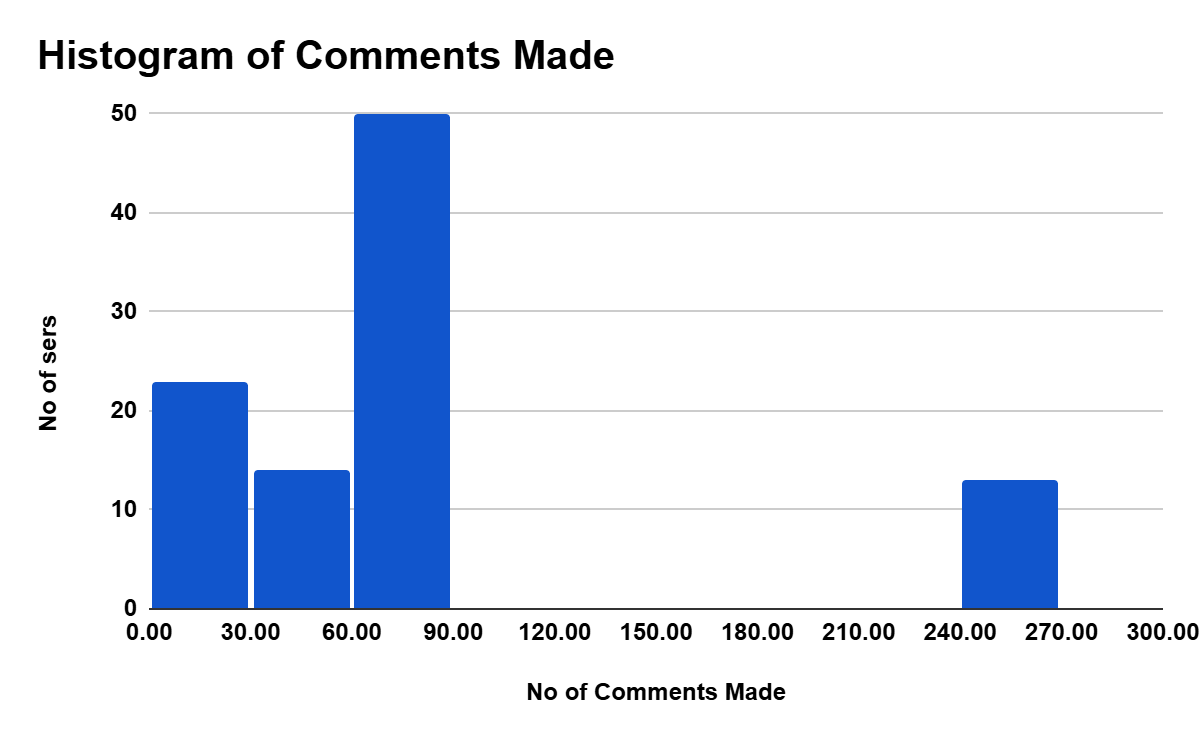
FROM users u

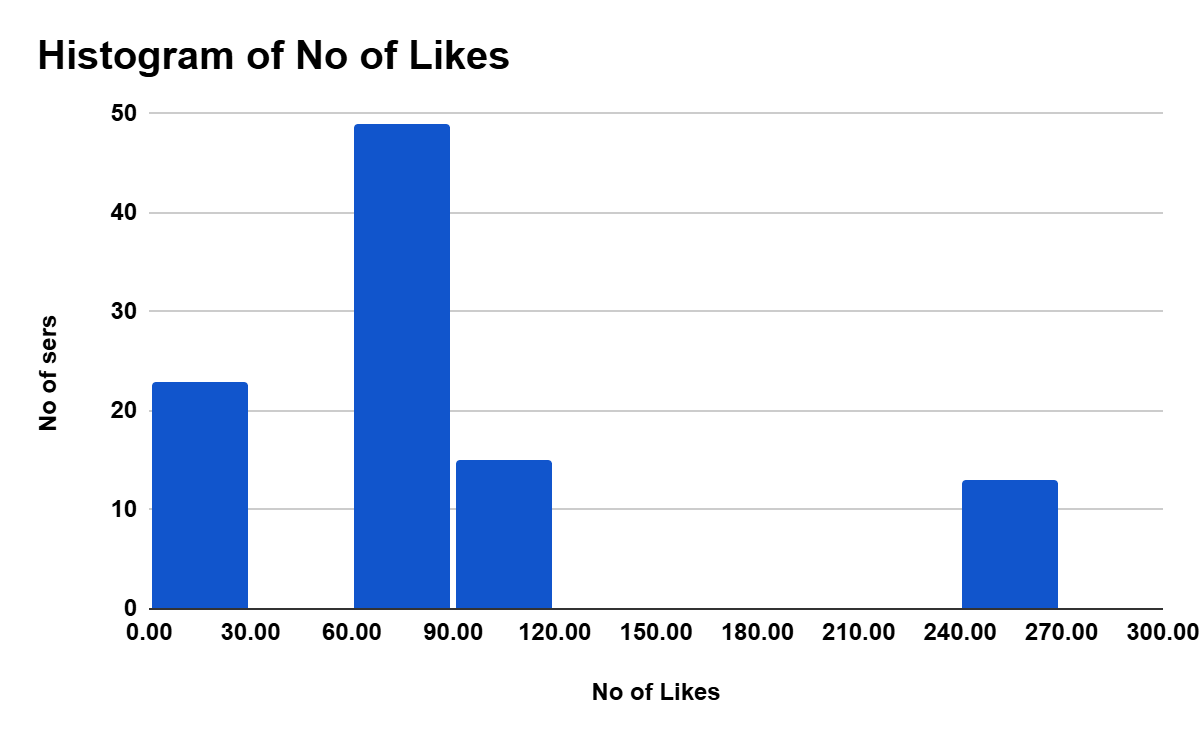
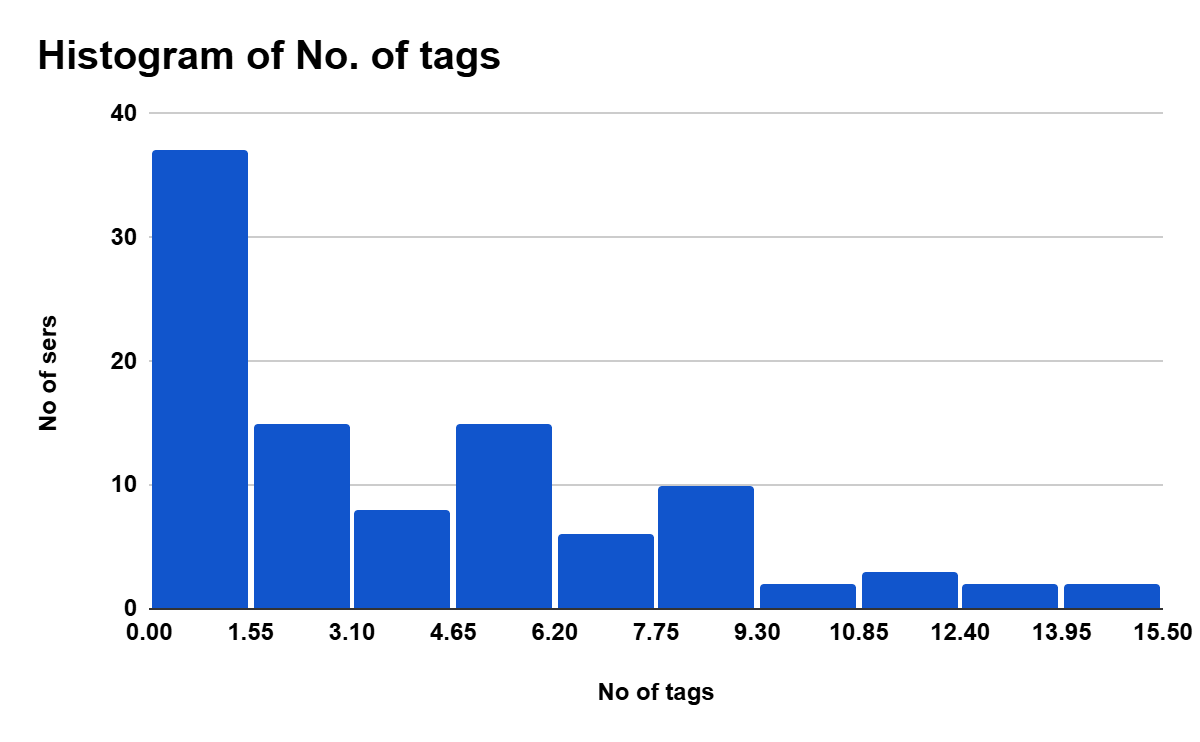
LEFT JOIN photos p ON p.user\_id=u.id

LEFT JOIN comments c ON c.user\_id=u.id

LEFT JOIN likes l ON l.user\_id=u.id

LEFT JOIN photo\_tags pt ON pt.photo\_id=p.id

GROUP BY user\_id, username;



1. **Calculate the average number of tags per post (photo\_tags and photos tables)**



with **cte** as (

select photo\_id, count(tag\_id) as count\_of\_tags

from photo\_tags

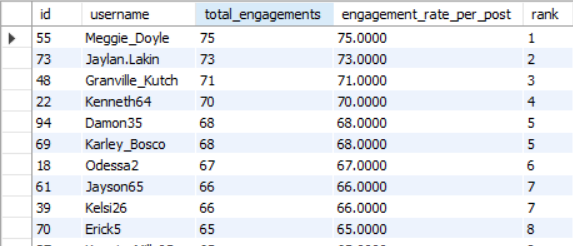
group by photo\_id)

**select** avg(count\_of\_tags) from cte

**= 2.638**

**.**

1. **Identify the top users with the highest engagement rates (likes, comments) on their posts and rank them.**



**Meggie\_Doyle has the Highest Engagement rate**

WITH **engagements** AS(

SELECT

u.id,

username,

COUNT(DISTINCT l.user\_id) AS likes,

COUNT(DISTINCT c.user\_id) AS comments

FROM users u

LEFT JOIN photos p ON u.id = p.user\_id

JOIN likes l ON l.photo\_id = p.id

JOIN comments c ON c.photo\_id=p.id

GROUP BY 1,2

),

**total\_post** as (

select user\_id , COUNT(id) as number\_of\_posts

from photos

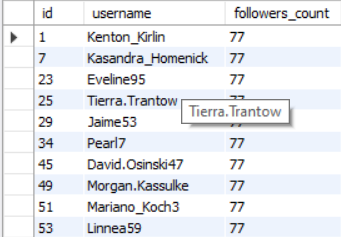
group by user\_id )

**select** id, username, likes+comments as total\_engagements, (likes+comments)/number\_of\_posts as engagement\_rate\_per\_post,

dense\_rank () over (order by (likes+comments)/number\_of\_posts desc) as `rank`

from engagements e

join total\_post t on e.id = t.user\_id

1. **Which users have the highest number of followers and followings?**

**Highest Number of followers:**

WITH **followers\_cte** AS (

SELECT

u.id,

u.username,

COUNT(f.follower\_id) AS followers\_count

FROM users u

LEFT JOIN follows f ON u.id = f.followee\_id

GROUP BY u.id, u.username

),

**ranked\_followers** AS (

SELECT \*,

DENSE\_RANK() OVER (ORDER BY followers\_count DESC) AS count\_rank

FROM followers\_cte

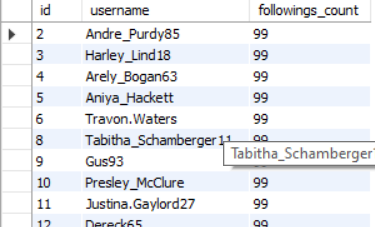
)

**SELECT** id, username, followers\_count

FROM ranked\_followers

WHERE count\_rank = 1

ORDER BY id;



**Highest Number of the following:**

WITH **followings\_cte** AS (

SELECT

u.id,

u.username,

COUNT(f.followee\_id) AS followings\_count

FROM users u

LEFT JOIN follows f ON u.id = f.follower\_id

GROUP BY u.id, u.username

),

**ranked\_followings** AS (

SELECT \*,

DENSE\_RANK() OVER (ORDER BY followings\_count DESC) AS count\_rank

FROM followings\_cte

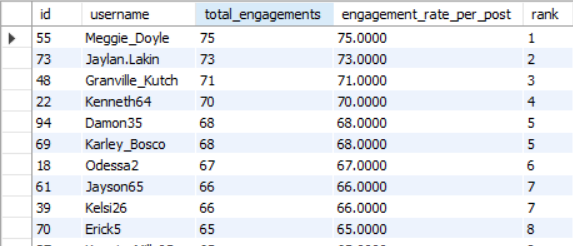
)

**SELECT** id, username, followings\_count

FROM ranked\_followings

WHERE count\_rank = 1

ORDER BY id;

1. **Calculate the average engagement rate (likes, comments) per post for each user.**

WITH **engagements** AS(

SELECT

u.id,

username,

COUNT(DISTINCT l.user\_id) AS likes,

COUNT(DISTINCT c.user\_id) AS comments

FROM users u

LEFT JOIN photos p ON u.id = p.user\_id

JOIN likes l ON l.photo\_id = p.id

JOIN comments c ON c.photo\_id=p.id

GROUP BY 1,2

),

**total\_post** as (

select user\_id , COUNT(id) as number\_of\_posts

from photos

group by user\_id )

Select

Id,

username,

likes+comments as total\_engagements,

(likes+comments)/number\_of\_posts as engagement\_rate\_per\_post,

dense\_rank () over (order by (likes+comments)/number\_of\_posts desc) as `rank`

from engagements e

join total\_post t on e.id = t.user\_id

1. **Get the list of users who have never liked any post (users and likes tables)**

WITH **users\_who\_liked** AS (

SELECT DISTINCT user\_id

FROM likes

)

**SELECT** u.id, u.username

FROM users u

LEFT JOIN users\_who\_liked l ON u.id = l.user\_id

WHERE l.user\_id IS NULL;

1. **How can you leverage user-generated content (posts, hashtags, photo tags) to create more personalized and engaging ad campaigns?**

**User-generated content such as posts, hashtags, and photo tags reveals users' interests and preferences. By analyzing frequently used tags, we can segment audiences and personalize ad campaigns based on trending topics.**

SELECT t.tag\_name, COUNT(\*) AS usage\_count

FROM photo\_tags pt

JOIN tags t ON pt.tag\_id = t.id

GROUP BY t.tag\_name

ORDER BY usage\_count DESC

LIMIT 10;

**Also, to run targeted ads, we analyze users' most frequently used tags from their posts.**

WITH **user\_tags** AS (

SELECT p.user\_id, t.tag\_name, COUNT(\*) AS tag\_count

FROM photos p

JOIN photo\_tags pt ON p.id = pt.photo\_id

JOIN tags t ON pt.tag\_id = t.id

GROUP BY p.user\_id, t.tag\_name

),

**ranked\_tags** AS (

SELECT \*,

ROW\_NUMBER() OVER (PARTITION BY user\_id ORDER BY tag\_count DESC) AS rn

FROM user\_tags

)

**SELECT** user\_id, tag\_name, tag\_count

FROM ranked\_tags

WHERE rn = 1;



1. **Are there any correlations between user activity levels and specific content types (e.g., photos, videos, reels)? How can this information guide content creation and curation strategies?**

WITH **user\_activity** AS (

SELECT u.id AS user\_id,

COUNT(DISTINCT p.id) AS posts,

COUNT(DISTINCT l.photo\_id) AS likes,

COUNT(DISTINCT c.id) AS comments

FROM users u

LEFT JOIN photos p ON u.id = p.user\_id

LEFT JOIN likes l ON u.id = l.user\_id

LEFT JOIN comments c ON u.id = c.user\_id

GROUP BY u.id

),

**user\_tag\_counts** AS (

SELECT p.user\_id, t.tag\_name, COUNT(\*) AS tag\_usage

FROM photos p

JOIN photo\_tags pt ON p.id = pt.photo\_id

JOIN tags t ON pt.tag\_id = t.id

GROUP BY p.user\_id, t.tag\_name

),

**ranked\_user\_tags** AS (

SELECT user\_id, tag\_name, tag\_usage,

RANK() OVER (PARTITION BY user\_id ORDER BY tag\_usage DESC) AS tag\_rank

FROM user\_tag\_counts

),

**top\_tag\_per\_user** AS (

SELECT user\_id, tag\_name

FROM ranked\_user\_tags

WHERE tag\_rank = 1

),

**final\_engagement** AS (

SELECT ua.user\_id,

ua.posts,

ua.likes,

ua.comments,

(ua.posts + ua.likes + ua.comments) AS total\_activity,

tt.tag\_name AS most\_used\_tag

FROM user\_activity ua

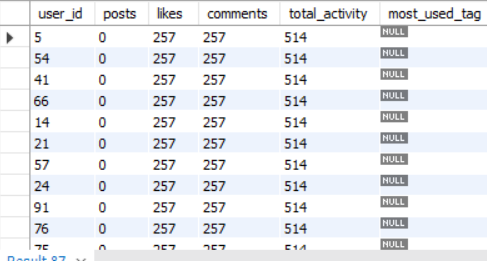
LEFT JOIN top\_tag\_per\_user tt ON ua.user\_id = tt.user\_id

)

**SELECT** \*

FROM final\_engagement

ORDER BY total\_activity DESC;



**Users with the most activity have never posted on Meta !!**

1. **Calculate the total number of likes, comments, and photo tags for each user.**

WITH **user\_photos** AS (

SELECT u.id AS user\_id, u.username, p.id AS photo\_id

FROM users u

JOIN photos p ON u.id = p.user\_id

),

**photo\_stats** AS (

SELECT up.user\_id, up.username, up.photo\_id,

COUNT(DISTINCT l.user\_id) AS likes,

COUNT(DISTINCT c.id) AS comments,

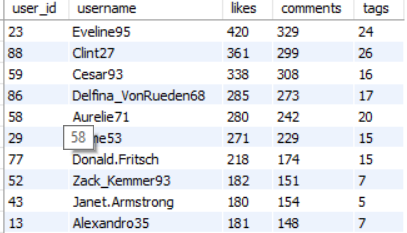
COUNT(DISTINCT pt.tag\_id) AS tags

FROM user\_photos up

LEFT JOIN likes l ON up.photo\_id = l.photo\_id

LEFT JOIN comments c ON up.photo\_id = c.photo\_id

LEFT JOIN photo\_tags pt ON up.photo\_id = pt.photo\_id

GROUP BY up.user\_id, up.username, up.photo\_id

),

**user\_engagement** AS (

SELECT user\_id, username,

SUM(likes) AS likes,

SUM(comments) AS comments,

SUM(tags) AS tags

FROM photo\_stats

GROUP BY user\_id, username

)

**SELECT** \*

FROM user\_engagement

ORDER BY likes + comments + tags DESC;

1. **Rank users based on their total engagement (likes, comments, shares) over a month**.

WITH **engagement** AS (

SELECT

u.id AS user\_id,

u.username,

YEAR(p.created\_dat) AS year,

MONTH(p.created\_dat) AS month,

COUNT(DISTINCT c.id) + COUNT(DISTINCT l.user\_id) AS total\_engagement

FROM users u

LEFT JOIN photos p ON u.id = p.user\_id

LEFT JOIN comments c ON c.photo\_id = p.id

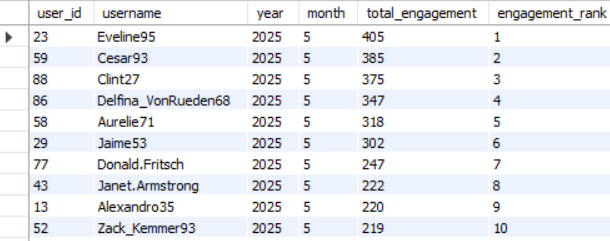
LEFT JOIN likes l ON l.photo\_id = p.id

GROUP BY u.id, u.username, YEAR(p.created\_dat), MONTH(p.created\_dat)

)

**SELECT**

user\_id, username, year, month, total\_engagement,

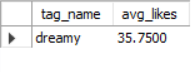
DENSE\_RANK() OVER (PARTITION BY year, month ORDER BY total\_engagement DESC) AS engagement\_rank

FROM engagement

where year is not null and month is not null

ORDER BY year, month, engagement\_rank;

1. **Retrieve the hashtags that have been used in posts with the highest average number of likes. Use a CTE to calculate the average likes for each hashtag first.**



WITH **photo\_like\_counts** AS (

SELECT

p.id AS photo\_id,

COUNT(DISTINCT l.user\_id) AS likes\_count

FROM photos p

LEFT JOIN likes l ON p.id = l.photo\_id

GROUP BY p.id

),

**tag\_likes** AS (

SELECT

t.tag\_name,

plc.likes\_count

FROM photo\_tags pt

JOIN tags t ON pt.tag\_id = t.id

JOIN photo\_like\_counts plc ON pt.photo\_id = plc.photo\_id

),

**avg\_likes\_per\_tag** AS (

SELECT

tag\_name,

AVG(likes\_count) AS avg\_likes

FROM tag\_likes

GROUP BY tag\_name

),

**max\_avg** AS (

SELECT MAX(avg\_likes) AS max\_avg\_likes

FROM avg\_likes\_per\_tag

)

**SELECT**

a.tag\_name,

a.avg\_likes

FROM avg\_likes\_per\_tag a

JOIN max\_avg m ON a.avg\_likes = m.max\_avg\_likes;

1. **Retrieve the users who have started following someone after being followed by that person**

WITH **mutual\_follows** AS (

SELECT 

f1.follower\_id AS user\_id,

f1.followee\_id AS followed\_back\_user\_id,

f1.created\_at AS follow\_time,

f2.created\_at AS followed\_by\_time

FROM follows f1

JOIN follows f2

ON f1.follower\_id = f2.followee\_id

AND f1.followee\_id = f2.follower\_id

)

**SELECT**

user\_id,

followed\_back\_user\_id,

follow\_time,

followed\_by\_time

FROM mutual\_follows

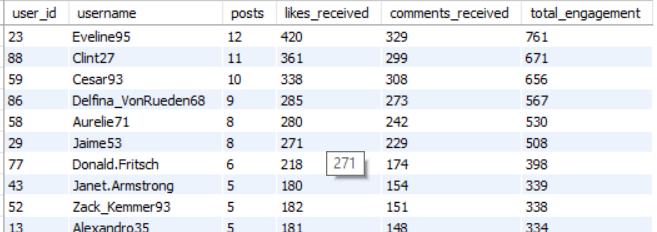
WHERE follow\_time > followed\_by\_time;

**Empty Table !! No, such cases.**

**Subjective Questions**

1. **Based on user engagement and activity levels, which users would you consider the most loyal or valuable? How would you reward or incentivize these users?**

The most loyal or valuable users can be identified by analyzing **high and consistent engagement metrics** such as frequent posts, high likes and comments received, and meaningful interactions like following others back. These users actively contribute to the platform’s ecosystem and drive network effects.They can be rewarded through **exclusive badges, priority support, early feature access, or spotlight opportunities (e.g., “Top Creator of the Month”)**.



WITH user\_posts AS (

SELECT user\_id, COUNT(\*) AS total\_posts

FROM photos

GROUP BY user\_id

),

user\_likes\_received AS (

SELECT p.user\_id, COUNT(l.user\_id) AS total\_likes

FROM photos p

JOIN likes l ON p.id = l.photo\_id

GROUP BY p.user\_id

),

user\_comments\_received AS (

SELECT p.user\_id, COUNT(c.id) AS total\_comments

FROM photos p

JOIN comments c ON p.id = c.photo\_id

GROUP BY p.user\_id

),

total\_engagement AS (

SELECT

u.id AS user\_id,

u.username,

COALESCE(up.total\_posts, 0) AS posts,

COALESCE(ul.total\_likes, 0) AS likes\_received,

COALESCE(uc.total\_comments, 0) AS comments\_received,

COALESCE(up.total\_posts, 0) + COALESCE(ul.total\_likes, 0) + COALESCE(uc.total\_comments, 0) AS total\_engagement

FROM users u

LEFT JOIN user\_posts up ON u.id = up.user\_id

LEFT JOIN user\_likes\_received ul ON u.id = ul.user\_id

LEFT JOIN user\_comments\_received uc ON u.id = uc.user\_id

)

SELECT \*

FROM total\_engagement

ORDER BY total\_engagement DESC

LIMIT 10;

1. **For inactive users, what strategies would you recommend to re-engage them and encourage them to start posting or engaging again?**

To re-engage inactive users, first identify them through low activity metrics (few or no posts, likes, comments, or logins). Then, implement personalized re-engagement campaigns such as email/app notifications with content suggestions based on their past interests.

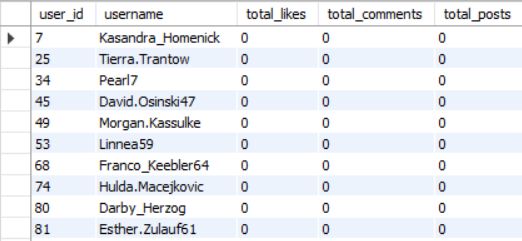
Use incentives like **rewards, badges, or follower boosts** for first re-engagement actions (e.g., posting after a long gap).

Additionally, introduce “catch-up” features (like “See what your friends have been up to”) and AI-driven post ideas or templates to lower the friction for content creation.

Offering **exclusive events, discounts, or creator tools** can further pull them back in.

WITH **user\_likes** AS (

SELECT user\_id, COUNT(\*) AS like\_count

FROM likes

GROUP BY user\_id

),

**user\_comments** AS (

SELECT user\_id, COUNT(\*) AS comment\_count

FROM comments

GROUP BY user\_id

),

**user\_posts** AS (

SELECT user\_id, COUNT(\*) AS post\_count

FROM photos

GROUP BY user\_id

),

**active\_users** AS (

SELECT user\_id FROM likes

UNION

SELECT user\_id FROM comments

UNION

SELECT user\_id FROM photos

)

**SELECT**

u.id AS user\_id,

u.username,

COALESCE(ul.like\_count, 0) AS total\_likes,

COALESCE(uc.comment\_count, 0) AS total\_comments,

COALESCE(up.post\_count, 0) AS total\_posts

FROM users u

LEFT JOIN user\_likes ul ON u.id = ul.user\_id

LEFT JOIN user\_comments uc ON u.id = uc.user\_id

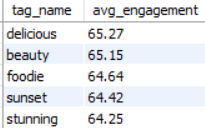
LEFT JOIN user\_posts up ON u.id = up.user\_id

LEFT JOIN active\_users au ON u.id = au.user\_id

WHERE au.user\_id IS NULL;

1. **Which hashtags or content topics have the highest engagement rates? How can this information guide content strategy and ad campaigns?**

WITH **photo\_engagement** AS (

SELECT 

p.id AS photo\_id,

COUNT(DISTINCT l.user\_id) AS like\_count,

COUNT(DISTINCT c.id) AS comment\_count

FROM photos p

LEFT JOIN likes l ON p.id = l.photo\_id

LEFT JOIN comments c ON p.id = c.photo\_id

GROUP BY p.id),

**tag\_engagement** AS (

SELECT

t.tag\_name,

AVG(pe.like\_count + pe.comment\_count) AS avg\_engagement

FROM photo\_tags pt

JOIN tags t ON pt.tag\_id = t.id

JOIN photo\_engagement pe ON pt.photo\_id = pe.photo\_id

GROUP BY t.tag\_name

)

**SELECT**

tag\_name,

ROUND(avg\_engagement, 2) AS avg\_engagement

FROM tag\_engagement

ORDER BY avg\_engagement DESC

LIMIT 5;

Hashtags linked to posts with **high average likes and comments** indicate strong engagement. By analyzing engagement metrics per hashtag, we can identify **high-performing content topics**. These tags represent what resonates most with users — for example, trending topics, popular niches, or seasonal themes.

This insight helps shape a content strategy by encouraging creators and marketers to use **top-performing hashtags** and create content around those themes. It also enables **targeted ad campaigns** by aligning product promotions with users’ current interests

.

1. **Are there any patterns or trends in user engagement based on demographics (age, location, gender) or posting times? How can these insights inform targeted marketing campaigns?**

We can analyze engagement patterns based on **posting times** using timestamps (created\_date in the photos table).

By aggregating likes and comments across different **hours of the day**, **days of the week**, or even **months**, we can identify peak activity periods.

These trends help marketers schedule content or campaigns during **high-engagement windows**, increasing visibility and impact. If demographic data were available, combining it with time-based behavior would enable **hyper-targeted, personalized marketing strategies**.

**SELECT**

p.id AS photo\_id,

u.username,

WEEKDAY(p.created\_dat) AS day\_of\_week,

EXTRACT(HOUR FROM p.created\_dat) AS hour\_of\_day,

COUNT(DISTINCT l.user\_id) AS likes\_received,

COUNT(DISTINCT c.id) AS comments\_received

FROM photos p

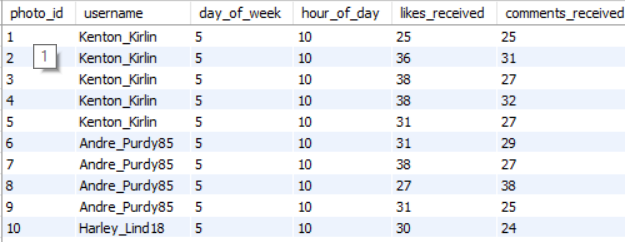
JOIN users u ON u.id = p.user\_id

LEFT JOIN likes l ON p.id = l.photo\_id

LEFT JOIN comments c ON p.id = c.photo\_id

GROUP BY p.id, u.username, day\_of\_week, hour\_of\_day

ORDER BY day\_of\_week, hour\_of\_day;



1. **Based on follower counts and engagement rates, which users would be ideal candidates for influencer marketing campaigns? How would you approach and collaborate with these influencers?**

Influencers with both *reach* and *engagement* drive authentic conversations, boost brand visibility, and enhance trust among their followers, making them powerful marketing allies.

WITH **follower\_counts** AS (

SELECT followee\_id AS user\_id, COUNT(\*) AS total\_followers

FROM follows

GROUP BY followee\_id

),

**engagement\_per\_post** AS (

SELECT

p.user\_id,

p.id AS photo\_id,

COUNT(DISTINCT l.user\_id) AS likes,

COUNT(DISTINCT c.id) AS comments

FROM photos p

LEFT JOIN likes l ON p.id = l.photo\_id

LEFT JOIN comments c ON p.id = c.photo\_id

GROUP BY p.user\_id, p.id

),

**user\_engagement\_summary** AS (

SELECT

user\_id,

COUNT(photo\_id) AS total\_posts,

SUM(likes) AS total\_likes,

SUM(comments) AS total\_comments,

(SUM(likes) + SUM(comments)) / NULLIF(COUNT(photo\_id), 0) AS avg\_engagement\_rate

FROM engagement\_per\_post

GROUP BY user\_id

),

**influencer\_candidates** AS (

SELECT

u.id AS user\_id,

u.username,

fc.total\_followers,

ue.total\_posts,

ue.total\_likes,

ue.total\_comments,

ue.avg\_engagement\_rate

FROM users u

JOIN follower\_counts fc ON u.id = fc.user\_id

JOIN user\_engagement\_summary ue ON u.id = ue.user\_id

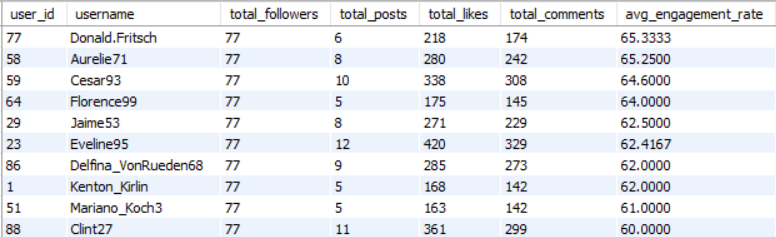
)

**SELECT** \*

FROM influencer\_candidates

ORDER BY total\_followers DESC, avg\_engagement\_rate DESC

LIMIT 10;



**This query identifies potential influencer candidates by calculating each user's total followers and average engagement rate (likes + comments per post). It uses CTEs to organize follower counts and post interactions. Users with both high follower counts and strong engagement are prioritized. The final result ranks the top 10 most influential users for marketing outreach**

1. **Based on user behavior and engagement data, how would you segment the user base for targeted marketing campaigns or personalized recommendations?**

We can segment the user base based on engagement and behavior using the following four key categories:

**Highly Active –** Regular posters and frequent engagers.

**Moderately Active** – Some posts and moderate engagement.

**Low Activity** – Rarely posts or engages.

**Inactive** – No posts, likes, or comments.

WITH **user\_activity** AS (

SELECT

u.id AS user\_id,

u.username,

COALESCE(p.post\_count, 0) AS posts,

COALESCE(l.likes\_given, 0) AS likes,

COALESCE(c.comments\_given, 0) AS comments

FROM users u

LEFT JOIN (

SELECT user\_id, COUNT(\*) AS post\_count

FROM photos

GROUP BY user\_id

) p ON u.id = p.user\_id

LEFT JOIN (

SELECT user\_id, COUNT(\*) AS likes\_given

FROM likes

GROUP BY user\_id

) l ON u.id = l.user\_id

LEFT JOIN (

SELECT user\_id, COUNT(\*) AS comments\_given

FROM comments

GROUP BY user\_id

) c ON u.id = c.user\_id

),

**segmented\_users** AS (

SELECT \*,

CASE

WHEN posts >= 5 AND (likes + comments) >= 15 THEN 'Highly Active'

WHEN posts BETWEEN 2 AND 4 OR (likes + comments) BETWEEN 5 AND 14 THEN 'Moderately Active'

WHEN posts = 1 OR (likes + comments) BETWEEN 1 AND 4 THEN 'Low Activity'

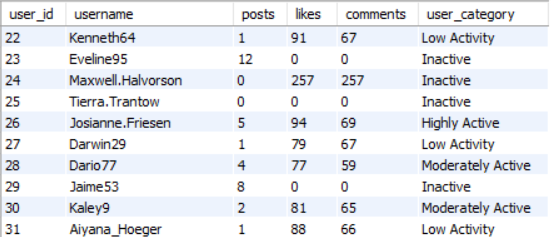
ELSE 'Inactive'

END AS user\_category

FROM user\_activity

)

**SELECT** \* FROM segmented\_users;



1. **If data on ad campaigns (impressions, clicks, conversions) is available, how would you measure their effectiveness and optimize future campaigns?**

To measure the effectiveness of ad campaigns and optimize future ones, we analyze key performance indicators (KPIs) such as impressions, clicks, and conversions.

### **Useful Metrics:**

**Click-Through Rate (CTR) = (Clicks / Impressions) × 100**:This measures how many people are clicking your ad after seeing it. A low CTR may suggest poor ad creative or irrelevant targeting.

**Conversion Rate = (Conversions / Clicks) × 100**: This tells you how effective the ad is at driving meaningful actions (e.g., purchases, sign-ups). A low rate might indicate a disconnect between the ad and the landing page experience.

**Cost per Click (CPC) and Cost per Acquisition (CPA)** : CPC helps monitor spending efficiency per click, while CPA shows how much each conversion is costing you. Higher CPA might require re-evaluating targeting or creative strategy.

**Return on Ad Spend (ROAS) = (Revenue / Cost)**: This is the ultimate profitability metric. It tells you how much return you're getting for every dollar spent.

By continuously measuring these metrics and adjusting campaigns accordingly, marketing teams can significantly improve the efficiency and impact of their ad efforts.

1. **How can you use user activity data to identify potential brand ambassadors or advocates who could help promote Instagram's initiatives or events?**

User activity data can be instrumental in identifying potential brand ambassadors by highlighting individuals who are both highly engaged and influential within the platform.

### 

### 

### **Identifying Brand Ambassadors:**

**High Engagement Rates:** Users whose posts receive consistently high likes and comments relative to their number of followers indicate active, loyal audiences.

**Posting Frequency & Content Quality:** Users who post regularly and use diverse, high-quality content (e.g., using trending hashtags or getting tagged frequently) show dedication and creativity.

**Follower Growth** & Retention: Rapid or steady follower increases and low unfollow rates suggest trust and appeal among their audience.

**Community Interaction** :Users who actively engage with others via likes, comments, and follows are likely to influence peer behavior — a key trait of brand advocates.

### **How to Collaborate with Them:**

**Incentivized Campaigns:** Offer exclusive features, merchandise, or early access to new tools/events.

**Co-Branded Content**: Partner with them to create posts or reels promoting initiatives like safety, inclusivity, or new app features.

**Event Invitations**: Invite top advocates to speak or attend launch events to amplify outreach.

**Recognition & Spotlight:** Feature them on Instagram’s official pages to increase their visibility and deepen loyalty.

By leveraging user activity data smartly, Instagram can identify authentic voices who can naturally amplify its brand and drive deeper community engagement.

1. **How would you approach this problem if the objective and subjective questions weren't given?**

If the objective and subjective questions weren’t provided, I would approach the problem using a systematic, data-driven exploration process:

**Understand Business Goals:** Align with marketing and product teams to define what *engagement* means—likes, comments, time spent, or user connections—and establish relevant KPIs like retention and acquisition.

**Explore the Data Schema:** Examine tables (users, photos, likes, comments, followers, tags), their relationships, and data quality. Understand foreign keys, constraints, and potential data issues like nulls or duplicates.

**Identify Key Focus Areas:** on user activity trends, content performance, follower dynamics, and temporal patterns. This ensures a structured exploration of user behavior and content interaction.

**Formulate Hypotheses:** testable questions, e.g., *Do frequent posters get more likes?* Or *are hashtags linked to engagement?* Hypotheses make the analysis purpose-driven.

**Run SQL-Based Exploration:** queries to test hypotheses—calculate engagement scores, identify top users, analyze tag usage, and understand posting patterns.

**Visualize for Clarity:** tools like Excel, Power BI, or Tableau to create intuitive visuals (heatmaps, bar charts, scatter plots) that uncover patterns and communicate insights.

**Deliver Data-Driven Strategies:** actionable plans: reward top users, optimize post timings, personalize outreach, and re-engage dormant users, anchoring all recommendations in the data.

1. **Assuming there's a "User\_Interactions" table tracking user engagements, how can you update the "Engagement\_Type" column to change all instances of "Like" to "Heart" to align with Instagram's terminology?**

You can update all instances of "Like" to "Heart" in the Engagement\_Type column of the User\_Interactions table using the following SQL query:

UPDATE User\_Interactions

SET Engagement\_Type = 'Heart'

WHERE Engagement\_Type = 'Like';