

Project: Social Media Data Analysis

Objective Questions:

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

Ans →

-- Checking for duplicate values in all the table

```
SELECT user_name, COUNT (*) as duplicate_count FROM users GROUP BY user_name having duplicate_count>1;
```

```
SELECT image_url, COUNT (*) as duplicate_count FROM photos GROUP BY image_url having duplicate_count>1;
```

```
SELECT user_id, photo_id, COUNT (*) as duplicate_count FROM comments GROUP BY user_id, photo_id having duplicate_count>1;
```

```
SELECT user_id, photo_id, COUNT (*) as duplicate_count FROM likes GROUP BY user_id, photo_id having duplicate_count>1;
```

```
SELECT follower_id, followee_id, COUNT (*) as duplicate_count FROM followers GROUP BY follower_id, followee_id having duplicate_count>1;
```

```
SELECT tag_name, COUNT (*) as duplicate_count FROM tags GROUP BY tag_name having duplicate_count>1;
```

-- Checking for null values for crucial columns in each table

```
SELECT COUNT (*) as users_null_count FROM users WHERE user_name is null or created_at is null;
```

```
SELECT COUNT (*) as photos_null_count FROM photos WHERE image_url is null or user_id is null or created_at is null;
```

```
SELECT COUNT (*) as comments_null_count FROM comments WHERE comment_txt is null or user_id is null or photo_id is null or created_at is null;
```

```
SELECT COUNT (*) as photos_tags_null_count FROM photo_tags WHERE photo_id is null or tag_id is null;
```

```
SELECT COUNT (*) as likes_null_count FROM likes WHERE user_id is null or photo_id is null or created_at is null;
```

```
SELECT COUNT (*) as tags_null_count FROM tags WHERE tag_name is null or created_at is null;
```

```
SELECT COUNT (*) as followers_null_count FROM followers WHERE follower_id is null or followee_id is null or created_at is null;
```

Observation:

- Checked for duplicates present in the data across all the tables using primary key, foreign key and there's no duplicates present in the data.
- Similarly, checked for null values using count (*) and taking column name into condition as "IS NULL". Having no null values across tables in data.

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

Ans →

-- Distribution of the number of posts (photos) by each user

WITH Posts as (**SELECT** user_id, **COUNT** (*) as post_count **FROM** photos **GROUP BY** user_id),

-- Distribution of number of likes by each user

Likes as (**SELECT** user_id, **COUNT** (*) as likes_count **FROM** likes **GROUP BY** user_id),

-- Distribution of number of comments by each user

Comments as (**SELECT** user_id, **COUNT** (*) as comment_count **FROM** comments **GROUP BY** user_id)

SELECT u.id, u.user_name, ifnull (p. post_count,0) as Post_count, ifnull (l. likes_count,0) as Likes_count, ifnull(c.comment_count,0) as Comments_count

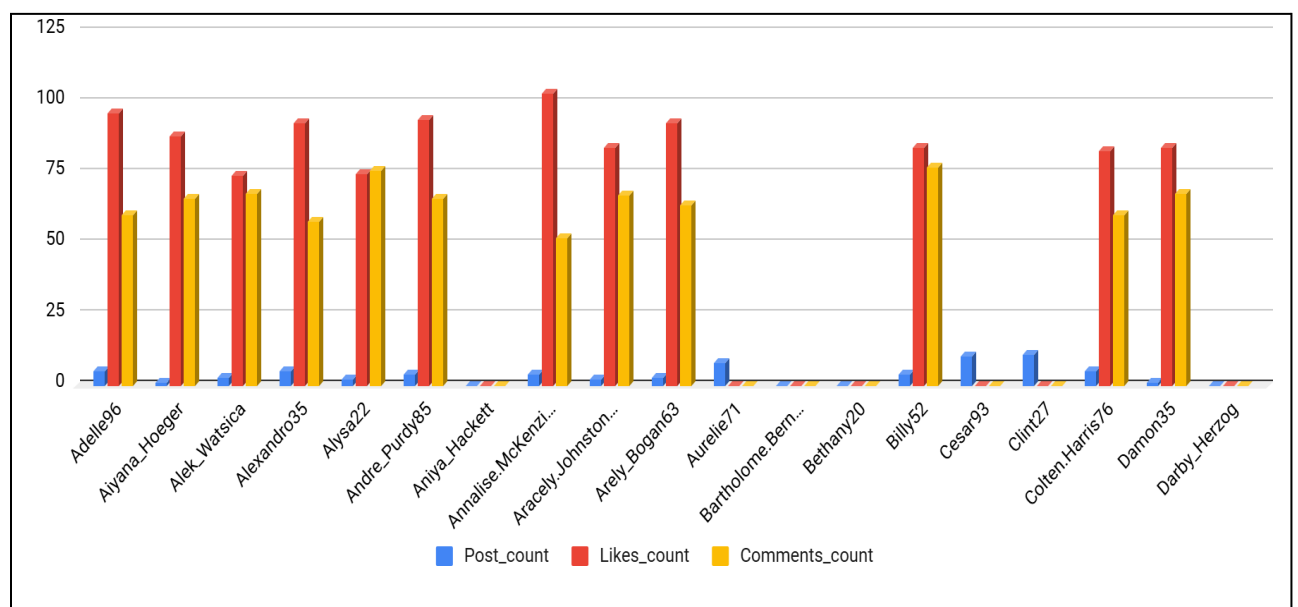
FROM users u **LEFT JOIN** Posts p **ON** u.id = p.user_id

LEFT JOIN comments c **ON** p.user_id = c.user_id

LEFT JOIN likes l **ON** c.user_id = l.user_id

ORDER BY u.user_name;

	id	user_name	Post_count	Likes_count	Comments_count
▶	65	Adelle96	5	96	60
	31	Aiyana_Hoeger	1	88	66
	99	Alek_Watsica	3	74	68
	13	Alexandro35	5	93	58
	84	Alysa22	2	75	76
	2	Andre_Purdy85	4	94	66
	5	Aniya_Hackett	0	0	0
	16	Annalise.McKenzie16	4	103	52
	82	Aracely.Johnston98	2	84	67
	4	Arely_Bogan63	3	93	64
	58	Aurelie71	8	0	0
	83	Bartholome.Bernhard	0	0	0
	91	Bethany20	0	0	0
	15	Billy52	4	84	77
	59	Cesar93	10	0	0
	88	Clint27	11	0	0
	78	Colten.Harris76	5	83	60
	94	Damon35	1	84	68
	80	Darby_Herzog	0	0	0
	28	Dario77	4	77	59



Observation:

- The column chart indicates that user's engagement levels vary significantly, with some users showing high interaction even with a similar number of posts.
- The user base splits mostly into low- and high-engagement groups, highlighting a blend of passive and active users.
- This suggests that certain users are consistently more interactive, while others engage minimally, creating distinct activity patterns across the base.

3. Calculate the average number of tags per post (photo_tags and photos tables).

Ans →

```
WITH Tags_per_post as (
SELECT photo_id,
COUNT (tag_id) as tags
FROM photo_tags GROUP BY photo_id)

SELECT ROUND (avg(tags),2) as Avg_tags_post
FROM Tags_per_post;
```

	Avg_tags_post
▶	2.64

Observation:

- Insight suggests on average, posts include around 2-3 tags, showing that users often add multiple relevant tags to their photos.
- This moderate tagging practice suggests users are keen on categorizing their posts or boosting their visibility.

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

Ans →

```
WITH likes_count AS (SELECT user_id, COUNT (*) AS total_likes FROM likes GROUP BY user_id),
comment_count AS (SELECT user_id, COUNT (*) AS total_comments FROM comments GROUP BY user_id),
```

-- Combine likes and comments to compute engagement rate

```
engagement AS (SELECT u.id AS user_id, u.user_name, COALESCE (l.total_likes, 0) +
COALESCE (c.total_comments, 0) AS engagement_rate
FROM users u
```

```
LEFT JOIN likes_count l ON u.id = l.user_id
LEFT JOIN comment_count c ON u.id = c.user_id),
```

-- Rank users by engagement rate

```
ranked_engagement AS (SELECT *, RANK () OVER (ORDER BY engagement_rate DESC) AS
engagement_rank FROM engagement)
```

```
SELECT * FROM ranked_engagement ORDER BY engagement_rank ASC;
```

	post_id	username	total_likes	total_comments	engagement_rate	engagement_rank
▶	13	Harley_Lind18	40	39	79	1
	145	Zack_Kemmer93	48	27	75	2
	147	Meggie_Doyle	41	34	75	2
	118	Janet_Armstrong	39	35	74	3
	227	Rick29	39	35	74	3
	123	Seth46	42	31	73	4
	127	Malinda_Streich	43	30	73	4
	196	Jaylan_Lakin	38	35	73	4
	225	Rick29	37	36	73	4
	97	Irwin_Larson	40	32	72	5
	153	Aurelie71	40	32	72	5

Observation:

- Curated table represents list of top users with highest engagement rate to their posts and rankings.
- These users consistently generate high engagement, posts strong likes and comments reflect high follower interest and activity.

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

Ans →

```
WITH followers_count AS (SELECT followee_id AS user_id, COUNT (follower_id) AS followers FROM followers
GROUP BY followee_id),
```

```
following_count AS (SELECT follower_id AS user_id, COUNT (followee_id) AS followings FROM followers GROUP
BY follower_id)
```

```
SELECT
```

```
u.user_name,
```

```
COALESCE (flwr.followers , 0) AS followers,
```

```
COALESCE (flwe.followings, 0) AS followings
```

```
FROM users u
```

```
LEFT JOIN followers_count flwr ON flwr.user_id = u.id
```

```
LEFT JOIN following_count flwe ON flwe.user_id = u.id;
```

	user_id	username	followers_count	followings_count
▶	2	Andre_Purdy85	76	99
	3	Harley_Lind18	76	99
	4	Arely_Bogan63	76	99
	5	Aniya_Hackett	76	99
	6	Travon.Waters	76	99
	8	Tabitha_Schamberger11	76	99
	9	Gus93	76	99
	10	Presley_McClure	76	99

Observation:

- The tables mention a list of users with the highest number of followers and followings from the meta user data.
- Users with a high follower count are ideal candidates for brand partnerships or influence-driven marketing.
- Users with high followings may represent highly engaged users who actively follow others' content.

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

Ans →

WITH like_count **AS** (**SELECT** photo_id, **COUNT** (**DISTINCT** user_id) **AS** total_likes **FROM** likes **GROUP BY** photo_id),

comment_count **AS** (**SELECT** photo_id, **COUNT** (**DISTINCT** user_id) **AS** total_comments **FROM** comments **GROUP BY** photo_id),

engagement_per_photo **AS** (
SELECT
p.id **AS** photo_id,
p.user_id,
COALESCE(l.total_likes,0) + **COALESCE** (c.total_comments, 0) **AS** engagement_score
FROM photos p
LEFT JOIN like_count l **ON** p.id = l.photo_id
LEFT JOIN comment_count c **ON** p.id = c.photo_id
),

engagement_per_user **AS** (
SELECT user_id,
SUM (engagement_score) **AS** total_engagement,
COUNT (photo_id) **AS** total_post
FROM engagement_per_photo **GROUP BY** user_id
)

SELECT
user_name,
e.total_engagement,
e.total_post,
ROUND (e.total_engagement / e.total_post, 2) **AS** avg_engagement_per_post
FROM engagement_per_user e
LEFT JOIN users u **ON** u.id = e.user_id
ORDER BY avg_engagement_per_post **DESC**;

	post_id	username	avg_engagement	engagement_rank
▶	13	Harley_Lind18	79	1
	145	Zack_Kemmer93	75	2
	147	Meggie_Doyle	75	2
	118	Janet.Armstrong	74	3
	227	Rick29	74	3
	123	Seth46	73	4
	127	Malinda_Streich	73	4
	196	Jaylan.Lakin	73	4
	225	Rick29	73	4
	97	Irwin.Larson	72	5

Observation:

- The tables represent the list of users with a large number of followers and a high average engagement rate to posts by these users, who might be well-known public figure.
- Users with a high average engagement rate per post are highly engaged, indicating potential for influencer marketing.
- Users with low average engagement may benefit from content engagement strategies to increase their visibility or post quality.

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

Ans →

```
SELECT u.id, u.user_name
FROM users u
LEFT JOIN likes l ON u.id = l.user_id
WHERE l.user_id IS NULL;
```

	id	username
▶	1	Kenton_Kirlin
	7	Kasandra_Homenick
	23	Eveline95
	25	Tierra.Trantow
	29	Jaime53
	34	Pearl7
	45	David.Osinski47
	49	Morgan.Kassulke
	51	Mariano_Koch3
	53	Linnea59
	58	Aurelie71
	59	Cesar93
	64	Florence99
	68	Franco_Keebler64
	74	Hulda.Macejkovic
	77	Donald.Fritsch
	80	Darby_Herzog
	81	Esther.Zulauf61
	83	Bartholome.Bernhard
	86	Delfina_VonRuede...
	88	Clint27
	89	Jessyca_West
	90	Esmeralda.Mraz57

Observation:

- Here is the list of users who have never liked any post on the platform.
- These users show low engagement in liking content, possibly preferring to consume passively or focus on other interactions.

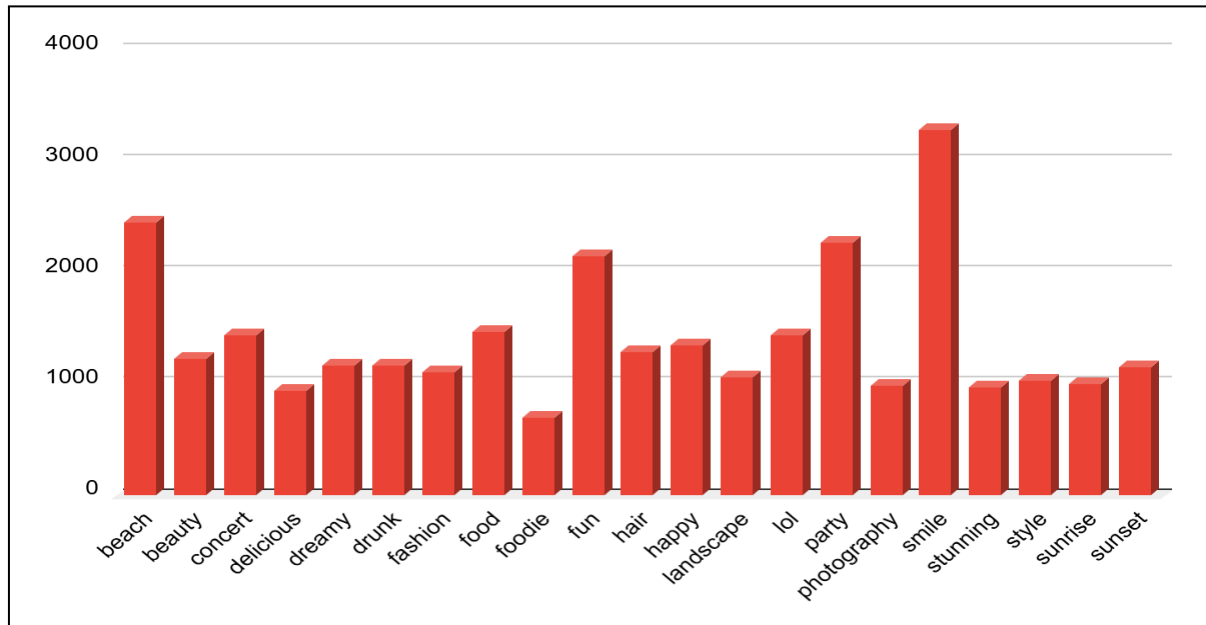
8. How can you leverage user-generated content (posts, hashtags, photo tags) to create more personalized and engaging ad campaigns?

Ans →

```
WITH tag_count AS (SELECT
    p.user_id,
    pt.tag_id,
    COUNT (*) AS tag_count
FROM photo_tags pt
LEFT JOIN photos p ON p.id = pt.photo_id
GROUP BY p.user_id, pt.tag_id
),
top_tags AS (SELECT
    tc.user_id,
    t.tag_name,
    tc.tag_count,
    RANK () OVER (PARTITION BY tc.user_id ORDER BY tc.tag_count DESC) AS tag_rank
FROM tag_count tc
JOIN tags t ON tc.tag_id = t.id
)

SELECT
    u.id AS user_id,
    tt.tag_name,
    tt.tag_count
FROM users u
JOIN top_tags tt ON tt.user_id = u.id
WHERE tt.tag_rank <= 3 -- get top 3 tags per user
ORDER BY u.id, tt.tag_count DESC;
```

	user_id	hashtag	tag_count	avg_engagement
▶	88	beach	30	229
	23	smile	13	211
	88	dreamy	14	167
	59	fun	18	161
	9	stunning	1	152
	2	fun	2	148
	2	smile	2	148
	23	lol	7	144
	8	smile	4	143
	65	smile	29	143



Observation:

- To leverage user-generated content for personalized ad campaigns, we identified popular tags, tag usage using a column chart, and average engagement to those tags.
- These tags help us identify users' interest in the specific content type, which helps create personalized ads with good engagement.

9. 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?

Ans →

```

tag_name AS content_type,
COUNT (DISTINCT user_id) AS unique_users,
COUNT (*) AS total_ WITH interacted_posts AS (
SELECT
    user_id,
    photo_id
FROM likes
WHERE user_id IN (SELECT user_id FROM comments)
),

```

```

photo_tags_ WITH_user AS (
SELECT
    i.user_id,
    i.photo_id,
    pt.tag_id
FROM interacted_posts i
JOIN photo_tags pt ON i.photo_id = pt.photo_id
),

```

```

tagged_content AS (
SELECT
    ptwu.user_id,
    ptwu.photo_id,
    ptwu.tag_id,
    t.tag_name
FROM photo_tags_ WITH_user ptwu
JOIN tags t ON ptwu.tag_id = t.id
),

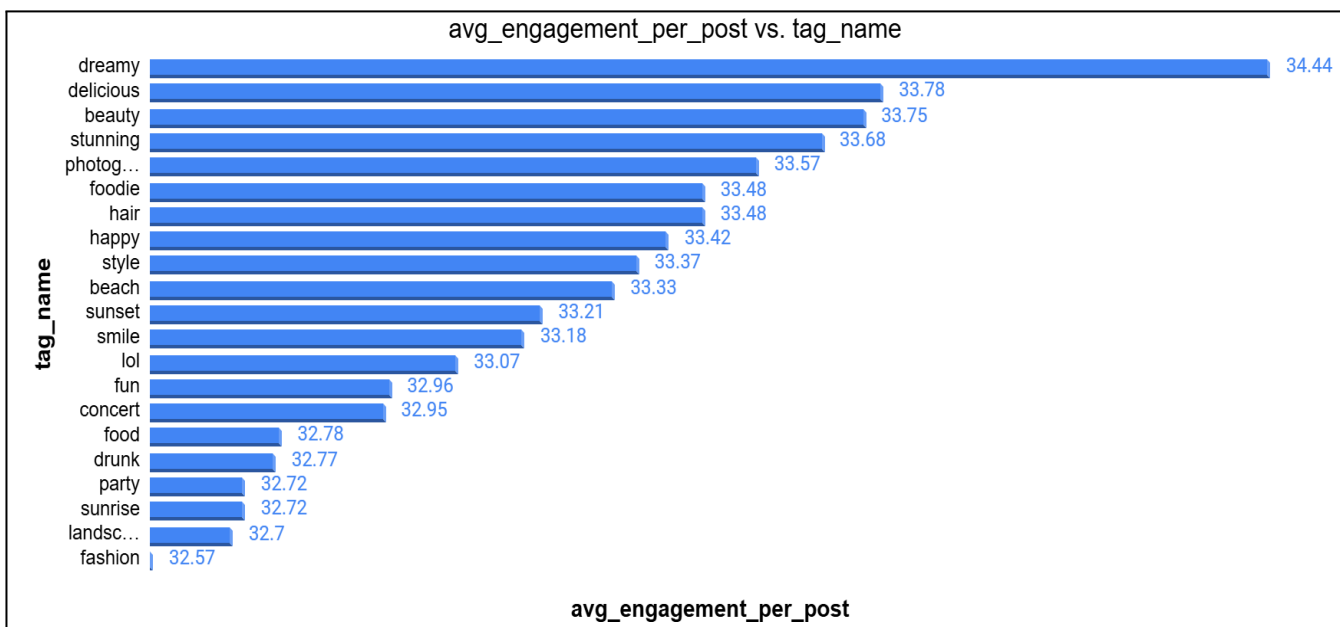
```



```
content_stats AS (
SELECT
interactions,
ROUND (COUNT (*) * 1.0 / COUNT (DISTINCT user_id), 2) AS avg_interactions_per_user
FROM tagged_content
WHERE tag_name IS NOT NULL
GROUP BY tag_name
)
```

```
SELECT * FROM content_stats
ORDER BY avg_interactions_per_user DESC;
```

	tag_name	post_count	total_likes	total_comments	avg_engagement_per_post
▶	dreamy	20077	20	563	34.44
	delicious	15876	15	455	33.78
	beauty	21058	20	604	33.75
	stunning	16335	16	469	33.68
	photography	16383	16	472	33.57
	foodie	11382	11	329	33.48
	hair	22602	23	652	33.48
	happy	21620	22	625	33.42
	style	16917	17	490	33.37
	beach	41935	42	1216	33.33
	sunset	19692	19	574	33.21
	smile	59199	59	1725	33.18
	lol	23643	24	691	33.07
	fun	37145	38	1089	32.96
	concert	23065	24	676	32.95
	food	24619	24	727	32.78
	drunk	18678	19	551	32.77
	party	38939	39	1151	32.72
	sunrise	16819	17	497	32.72
	landscape	17067	17	505	32.70
	fashion	18470	19	548	32.57



Observation:

- The table shows that 'dreamy' tag has a maximum number of engagements even though 'smile' tag has more postings.
- So, yes, there is a correlation between user activity levels and specific content types, as certain genres receive significantly more engagement than others.
- Based on the content type and average engagement per post, user's interest-specific content strategy can be made.
- Recommend ad formats that align with popular content types to increase campaign effectiveness.

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

Ans →

```
WITH total_likes_per_post AS (
  SELECT
    photo_id,
    COUNT (user_id) AS likes_per_photo
  FROM likes
  GROUP BY photo_id
),
total_comments_per_post AS (
  SELECT
    photo_id,
    COUNT (user_id) AS comments_per_photo
  FROM comments
  GROUP BY photo_id
),
total_tags_per_post AS (
  SELECT
    photo_id,
    COUNT (tag_id) AS tags_per_photo
  FROM photo_tags
  GROUP BY photo_id
),
total_interactiON AS (
  SELECT
    p.user_id,
    p.id AS post_id,
    COALESCE(l.likes_per_photo, 0) AS likes_per_photo,
    COALESCE(c.comments_per_photo, 0) AS comments_per_photo,
    COALESCE(t.tags_per_photo, 0) AS tags_per_photo
  FROM photos p
  LEFT JOIN total_likes_per_post l ON p.id = l.photo_id
  LEFT JOIN total_comments_per_post c ON p.id = c.photo_id
  LEFT JOIN total_tags_per_post t ON p.id = t.photo_id
)
SELECT
  ti.user_id,
  u.user_name,
  SUM(ti.likes_per_photo) AS total_likes,
  SUM(ti.comments_per_photo) AS total_comments,
  SUM(ti.tags_per_photo) AS total_tags
FROM total_interactiON ti
JOIN users u ON ti.user_id = u.id
GROUP BY ti.user_id, u.user_name;
```

	id	username	total_likes	total_comments	total_tags
▶	1	Kenton_Kirlin	0	0	15
	2	Andre_Purdy85	94	66	7
	3	Harley_Lind18	79	67	7
	4	Arely_Bogan63	93	64	2
	5	Aniya_Hackett	257	257	0
	6	Travon.Waters	82	82	8
	7	Kasandra_Homenick	0	0	0
	8	Tabitha_Schamberger11	79	61	10
	9	Gus93	85	60	5
	10	Presley_McClure	87	63	8

Observation:

- The table shows that there are a variety of users and their engagement data, some might be fake accounts just used for tagging and post boost.
- Some users show high engagement in terms of likes and comments but have not used any photo tags.
- This suggests they actively engage **WITH** content but may underuse tagging to categorize or promote their posts.

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

Ans →

```

WITH total_likes_per_post AS (
  SELECT
    photo_id,
    MONTH(created_at) AS mnth,
    COUNT(user_id) AS likes_per_photo
  FROM likes
  GROUP BY photo_id, mnth
),
total_comments_per_post AS (
  SELECT
    photo_id,
    MONTH(created_at) AS mnth,
    COUNT(user_id) AS comments_per_photo
  FROM comments
  GROUP BY photo_id, mnth
),
total_tags_per_post AS (
  SELECT
    photo_id,
    COUNT(tag_id) AS tags_per_photo
  FROM photo_tags
  GROUP BY photo_id
),
total_interactiON AS (
  SELECT
    p.user_id,
    MONTH(p.created_at) AS mnth,
    COALESCE(l.likes_per_photo, 0) AS likes_per_photo,
    COALESCE(c.comments_per_photo, 0) AS comments_per_photo,
    COALESCE(t.tags_per_photo, 0) AS tags_per_photo
  FROM photos p
  LEFT JOIN total_likes_per_post l ON p.id = l.photo_id AND MONTH(p.created_at) = l.mnth

```

```

LEFT JOIN total_comments_per_post c ON p.id = c.photo_id AND MONTH (p.created_at) = c.mnth
LEFT JOIN total_tags_per_post t ON p.id = t.photo_id
),
monthly_engagement AS (
  SELECT
    user_id,
    mnth,
    SUM(likes_per_photo + comments_per_photo + tags_per_photo) AS total_engagement
  FROM total_interaction
  GROUP BY user_id, mnth
)
SELECT
  me.user_id,
  u.user_name,
  me.mnth AS month,
  me.total_engagement,
  RANK() OVER (PARTITION BY me.mnth ORDER BY me.total_engagement DESC) AS engagement_rank
FROM monthly_engagement me
JOIN users u ON me.user_id = u.id;

```

	user_id	username	month	total_engagement	engagement_rank
▶	23	Eveline95	2024-09	405	1
	59	Cesar93	2024-09	385	2
	88	Clint27	2024-09	375	3
	86	Delfina_VonRueden68	2024-09	347	4
	58	Aurelie71	2024-09	318	5
	29	Jaime53	2024-09	302	6
	77	Donald.Fritsch	2024-09	247	7
	43	Janet.Armstrong	2024-09	222	8
	13	Alexandro35	2024-09	220	9
	52	Zack_Kemmer93	2024-09	219	10

Observation:

- The table highlights users who generate the most engagement (likes and comments) **ON** their posts each month, identifying high-performing accounts that **cON**sistently drive activity.
- Users who rank high in monthly engagement could be ideal candidates for brand collaborations or ambassador roles.
- Analyzing the content types of high-ranking users can reveal themes or post formats that resonate well with the audience, guiding future content strategies.

11. 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.

Ans →

```

WITH photo_with_likes AS (
  SELECT
    photo_id,
    COUNT(*) AS likes_count
  FROM likes
  GROUP BY photo_id
),
likes_per_hashtag AS (

```

```

SELECT
    pt.tag_id,
    pl.likes_count
FROM photo_tags pt
JOIN photo_WITH_likes pl ON pt.photo_id = pl.photo_id
)

```

```

SELECT
    t.tag_name,
    ROUND (AVG (lph.likes_count), 2) AS avg_likes_per_tag
FROM likes_per_hashtag lph
JOIN tags t ON lph.tag_id = t.id
GROUP BY t.tag_name
ORDER BY avg_likes_per_tag DESC;

```

	id	Hashtags	Avg_likes
▶	10	dreamy	35.75

Observation:

- The query identifies the hashtag that is associated **WITH** posts receiving the highest average number of likes, which highlights the most engaging hashtag on the platform.
- Understanding which hashtags drive the most likes allows for more strategic hashtag usage in campaigns, maximizing reach and visibility.
- This insight can guide content strategy, influence marketing campaigns, and improve user engagement on the platform.

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

Ans →

13.

```

SELECT
    f1.follower_id AS user_id,
    f1.followee_id AS followed_back_user
FROM followers f1
JOIN followers f2
ON f1.follower_id = f2.followee_id
AND f1.followee_id = f2.follower_id
AND f1.created_at > f2.created_at;

```

	followed_back	followed_by

Observation:

- The results show no users who followed someone back after being followed, resulting in no output.

Subjective Questions:

- 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?

Ans →

```
WITH post_per_user AS (
  SELECT user_id, COUNT(id) AS total_post
  FROM photos
  GROUP BY user_id),
```

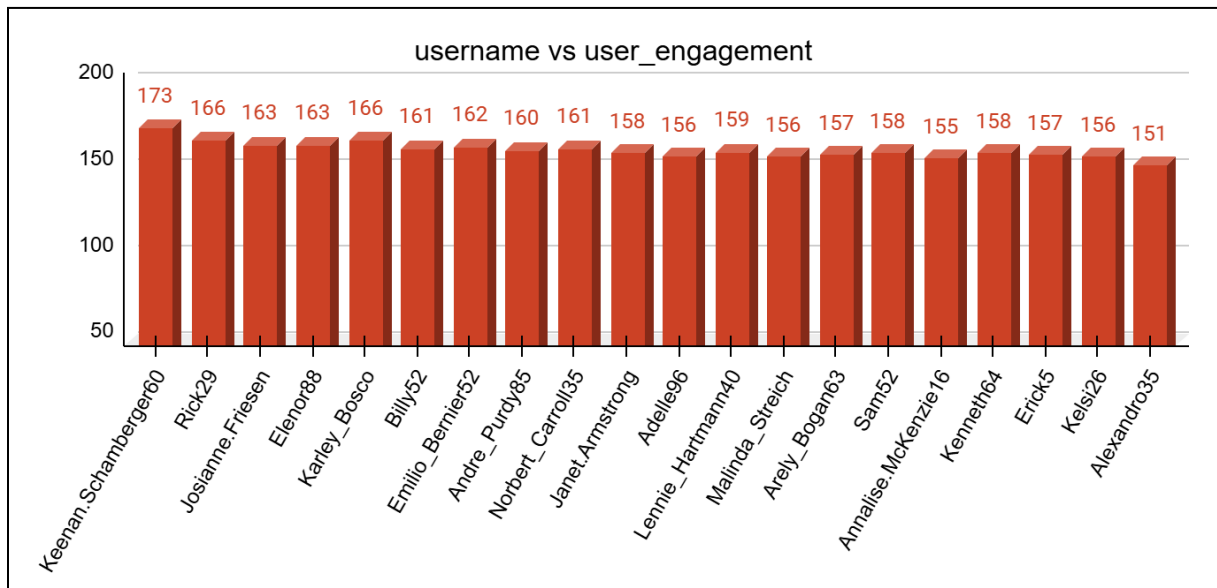
```
likes_per_user AS (
  SELECT user_id, COUNT(photo_id) AS total_likes
  FROM likes
  GROUP BY user_id),
```

```
comment_per_user AS (
  SELECT user_id, COUNT(id) AS total_comments
  FROM comments
  GROUP BY user_id)
```

-- user activity & engagement and ranking user on loyalty

```
SELECT u.id, u.user_name, SUM(total_post + total_likes + total_comments) AS user_activity_level,
SUM(total_likes + total_comments) AS user_engagement_rate,
DENSE_RANK() OVER(ORDER BY SUM(total_post + total_likes + total_comments) DESC,
SUM(total_likes + total_comments)) AS user_loyalty_rank
FROM users u JOIN post_per_user p
ON u.id = p.user_id
JOIN likes_per_user l ON u.id = l.user_id
JOIN comment_per_user c ON u.id = c.user_id
GROUP BY u.id;
```

	id	username	user_activity_level	user_engagement_rate	user_loyalty_rank
▶	96	Keenan.Schamberger60	176	173	1
	87	Rick29	170	166	2
	26	Josianne.Friesen	168	163	3
	63	Elenor88	167	163	4
	69	Karley_Bosc Karley_Bosco	57	166	5
	15	Billy52	165	161	6
	67	Emilio_Bernier52	165	162	7
	2	Andre_Purdy85	164	160	8
	17	Norbert_Carroll35	164	161	9
	43	Janet.Armstrong	163	158	10



Observation:

- Based on the query results & bar chart, users **WITH** the highest user_activity_level and user_engagement_rate frequently posting, liking, and commenting can be considered the most loyal or valuable.

Here's how I'd approach recognizing and rewarding these top users:

- Special Recognition:** Grant them VIP status **WITH** exclusive profile badges or features, acknowledging their loyalty and engagement.
- Early Access to New Features:** Offer these users priority access to new tools, updates, or beta features to keep them engaged and feeling valued.
- Special Content and Events:** Provide access to exclusive content, webinars, or events tailored to their interests as a thank-you for their engagement.
- Loyalty Rewards Program:** Create a rewards program where points earned through engagement can be redeemed for perks like profile boosts, content shout-outs, or even discounts on partnered services.
- Influencer or Ambassador Opportunities:** Invite top users to become brand ambassadors, allowing them to represent and promote Instagram initiatives and gain more followers.

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

Ans →

```
SELECT
    u.id,
    u.user_name,
    (COUNT(DISTINCT p.id) + COUNT(DISTINCT l.photo_id) + COUNT(DISTINCT c.id)) AS
total_engagement
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, u.user_name
HAVING total_engagement = 0;
```

	id	username	engagement
▶	7	Kassandra_Homenick	0
	25	Tierra.Trantow	0
	34	Pearl7	0
	45	David.Osinski47	0
	49	Morgan.Kassulke	0
	53	Linnea59	0
	68	Franco_Keebler64	0
	74	Hulda.Macejkovic	0
	80	Darby_Herzog	0
	81	Esther.Zulauf61	0
	83	Bartholome.Bernhard	0
	89	Jessyca_West	0
	90	Esmeralda.Mraz57	0

Observation:

- The tables show the list of users with no activity on the platform in terms of posts, likes, and comments.

To re-engage inactive users, here are personalized strategies using a more direct and engaging Approach:

- **Personalized Outreach:** Send friendly, personalized messages that highlight how much they're missed and acknowledge their past contributions.
- **Exclusive Offers:** Provide special incentives, like discounts or early access to new features, to encourage a return.

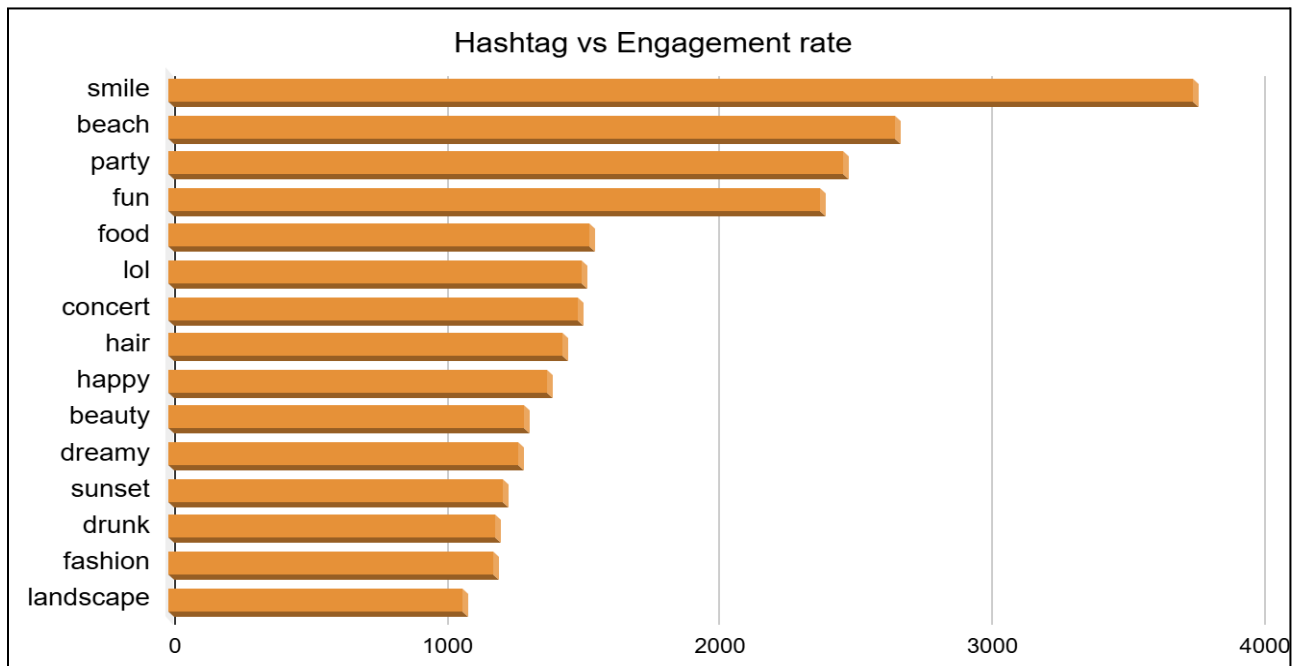
- **Feedback Surveys:** Invite inactive users to share why they left and what would improve their experience, making them feel valued.
- **Gamified Challenges:** Introduce fun challenges, leaderboards, or achievement badges to make re-engagement enjoyable and rewarding.
- **Curated Content Highlights:** Share trending or personalized content based on their past interests to spark curiosity and re-engage them.

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

Ans →

```
WITH likes_per_post AS (
  SELECT
    photo_id,
    COUNT(user_id) AS likes
  FROM likes
  GROUP BY photo_id
),
comments_per_post AS (
  SELECT
    photo_id,
    COUNT(user_id) AS comments
  FROM comments
  GROUP BY photo_id
),
total_engagement_per_post AS (
  SELECT
    pt.photo_id,
    pt.tag_id,
    COALESCE(l.likes, 0) + COALESCE(c.comments, 0) AS total_engagement
  FROM photo_tags pt
  LEFT JOIN likes_per_post l ON l.photo_id = pt.photo_id
  LEFT JOIN comments_per_post c ON c.photo_id = pt.photo_id
),
engagement_per_tag AS (
  SELECT
    tag_id,
    COUNT(photo_id) AS total_posts,
    SUM(total_engagement) AS total_engagement
  FROM total_engagement_per_post
  GROUP BY tag_id
)
SELECT
  t.tag_name,
  ROUND(e.total_engagement * 1.0 / e.total_posts, 2) AS average_engagement_per_tag
FROM engagement_per_tag e
JOIN tags t ON t.id = e.tag_id
ORDER BY average_engagement_per_tag DESC;
```

	tag_name	post_engagement_rate	tag_engagement_rank
▶	smile	3758	1
	beach	2664	2
	party	2474	3
	fun	2390	4
	food	1539	5
	lol	1512	6
	concert	1501	7
	hair	1446	8
	happy	1386	9
	beauty	1303	10
	dreamy	1278	11
	sunset	1224	1224
	drunk	1198	13
	fashion	1188	14
	landscape	1076	15
	style	1076	15
	sunrise	1071	16
	stunning	1028	17
	photogra...	1024	18
	delicious	979	19
	foodie	711	20



Observation:

- The Bar chart clearly shows that the hashtag "smile" has the highest engagement rate, followed by "beach" and "party".
- We can prioritize these tags in posts and campaigns to boost reach and interaction.
- **Content Strategy Alignment:** We can Personalize content to align with the most engaging topics or themes. We will focus on hashtags such as smile, beach, party, and fun to drive engagement that also resonates with the audience.
- **Targeted Ad Campaigns:** We can use high-engagement tags to target users with specific interests, improving ad relevance and effectiveness.

4. 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?

Ans →

```
WITH post_engagement AS (
SELECT
  p.id AS post_id,
  p.user_id,
  p.created_at AS post_time,
  COUNT(l.user_id) AS total_likes,
  COUNT(c.id) AS total_comments,
  (COUNT(l.user_id) + COUNT(c.id)) AS total_engagement
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),
```

```
-- Extracted hour and day of week FROM post_time and calculate average engagement for each
engagement_by_time AS (
SELECT
  hour(post_time) AS post_hour,
  DAYOFWEEK(post_time) AS post_day,
  COUNT(post_id) AS total_post,
```

```
ROUND (AVG (total_engagement)) AS avg_engagement_per_post
FROM post_engagement
GROUP BY post_hour, post_day
ORDER BY avg_engagement_per_post DESC)
```

```
SELECT
post_hour,
post_day,
total_post,
avg_engagement_per_post
FROM engagement_by_time;
```

	post_hour	post_day	total_post	avg_engagement_per_post
▶	2	7	257	1989

Observation:

- These Identified specific hours denote the highest average engagement. This can help in scheduling posts or marketing campaigns during peak engagement times.
- Days of week 7 show higher engagement, allowing campaigns to be focused on those days.
- Insights help us know when users are most active and allow for tailored content delivery, such as posting promotions or sending notifications during high-engagement windows to maximize visibility and user response.

5. 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?

Ans →

```
WITH user_follower_count AS (
SELECT
followee_id AS user, COUNT (follower_id) AS followers_count
FROM followers
GROUP BY followee_id),

likes_per_user AS (
SELECT user_id, COUNT (photo_id) AS total_likes
FROM likes
GROUP BY user_id),

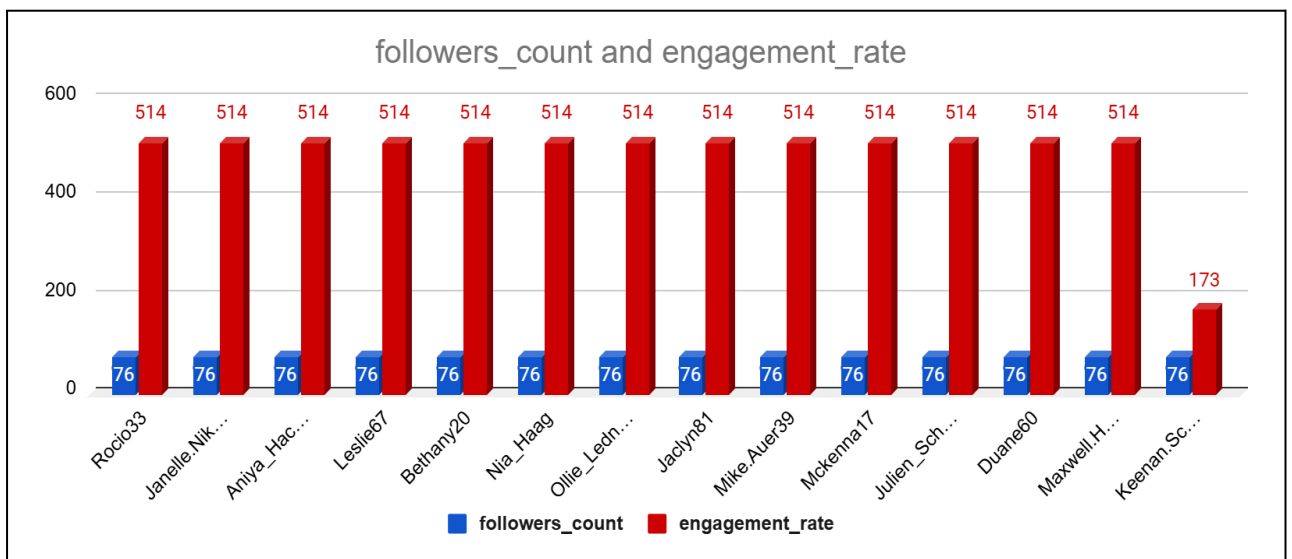
comment_per_user AS (
SELECT user_id, COUNT (id) AS total_comments
FROM comments
GROUP BY user_id)

-- ideal candidate for influencer marketing rank

SELECT u.user_name, followers_count, SUM ( l.total_likes + c.total_comments) AS engagement_rate,
DENSE_RANK() OVER(ORDER BY followers_count DESC, SUM ( l.total_likes + c.total_comments) DESC)
AS candidate_rank
FROM users u
JOIN user_follower_count uf ON u.id = uf.user
JOIN likes_per_user l ON u.id = l.user_id
JOIN comment_per_user c ON u.id = c.user_id
GROUP BY 1, 2
```

ORDER BY followers_count DESC, engagement_rate DESC;

	username	followers_count	engagement_rate	candidate_rank
▶	Rocio33	76	514	1
	Janelle.Nikolaus81	76	514	1
	Aniya_Hackett	76	514	1
	Leslie67	76	514	1
	Bethany20	76	514	1
	Nia_Haag	76	514	1
	Ollie_Ledner37	76	514	1
	Jaclyn81	Jaclyn81	514	1
	Mike.Auer39	76	514	1
	Mckenna17	76	514	1



Observation:

- Users with large followings and high engagement (likes + comments) are ideal for influencer marketing, maximizing reach and impact.
- These influencers show potential for consistent, authentic brand partnerships due to their strong connection with followers.

To approach and collaborate these influencers, consider the following:

1. **Tailored Outreach:** Connect with influencers directly, using a personalized message that acknowledges their unique impact and shows how your brand resonates with their audience.
2. **Provide Mutual Benefits:** Suggest partnerships that offer value for their followers, such as special offers, giveaways, or co-creating engaging content.

3. **Develop Authentic Relationships:** Build genuine connections by actively engaging with their content and showing real interest in their audience.
4. **Set Transparent Goals:** Clearly outline your campaign objectives, expectations, and collaboration terms to ensure alignment.

This approach encourages meaningful, mutually beneficial relationships with influencers.

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

Ans →

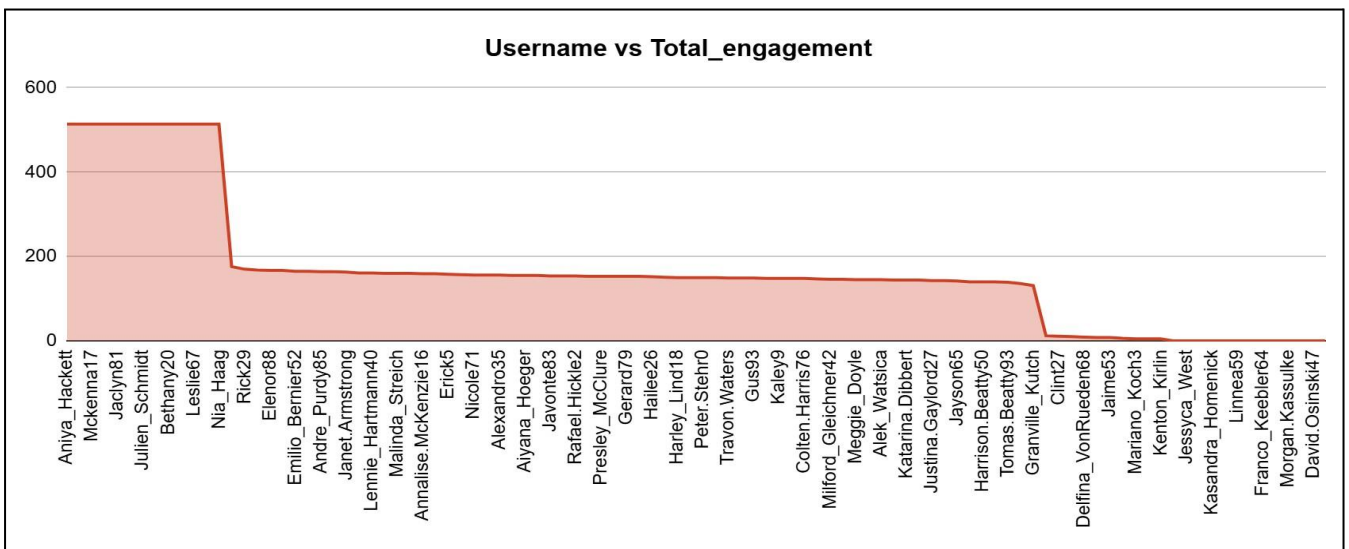
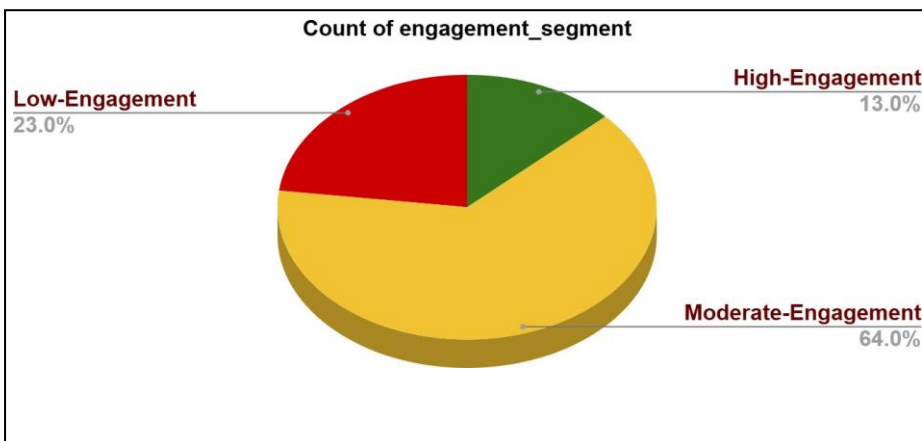
```
WITH post_per_user AS (
SELECT user_id, COUNT(id) AS total_post
FROM photos
GROUP BY user_id),

likes_per_user AS (
SELECT user_id, COUNT(photo_id) AS total_likes
FROM likes
GROUP BY user_id),

comment_per_user AS (
SELECT user_id, COUNT(id) AS total_comments
FROM comments
GROUP BY user_id)

SELECT u.id, u.user_name,
(COALESCE(l.total_likes, 0) + COALESCE(c.total_comments, 0) + COALESCE(p.total_post, 0)) AS
total_engagement,
CASE
WHEN (COALESCE(l.total_likes, 0) + COALESCE(c.total_comments, 0) + COALESCE(p.total_post, 0)) >=
200 THEN 'High-Engagement'
WHEN (COALESCE(l.total_likes, 0) + COALESCE(c.total_comments, 0) + COALESCE(p.total_post, 0))
BETWEEN 100 AND 200 THEN 'Moderate-Engagement'
ELSE 'Low-Engagement'
END AS engagement_segment
FROM users u
LEFT JOIN post_per_user p ON u.id = p.user_id
LEFT JOIN likes_per_user l ON u.id = l.user_id
LEFT JOIN comment_per_user c ON u.id = c.user_id
ORDER BY total_engagement DESC;
```


	id	username	total_engagement	engagement_segment
▶	5	Aniya_Hackett	514	High-Engagement
	54	Duane60	514	High-Engagement
	41	Mckenna17	514	High-Engagement
	66	Mike.Auer39	514	High-Engagement
	14	Jadyn81	514	High-Engagement
	21	Rocio33	514	High-Engagement
	57	Julien_Schmidt	514	High-Engagement
	24	Maxwell.Halvorson	514	High-Engagement
	91	Bethany20	514	High-Engagement
	76	Janelle.Nikolaus81	514	High-Engagement
	75	Leslie67	514	High-Engagement
	36	Ollie_Ledner37	514	High-Engagement
	71	Nia_Haag	514	High-Engagement
	96	Keenan.Schamber...	176	Moderate-Engagement
	87	Rick29	170	Moderate-Engagement
	26	Josianne.Friesen	168	Moderate-Engagement



Observation:

- **High-Engagement Users:** Users with 200+ engagements are highly active. We can target them for ambassador programs or exclusive offers to sustain activity.

- **Moderate-Engagement Users:** with 100-200 engagements, these users need slight incentives to increase activity. Personalized recommendations can drive further engagement.
- **Low-Engagement Users:** Users with under 100 engagements may need onboarding support. Tutorials or follow suggestions could help boost their interaction.
- **Personalized Campaigns:** Each segment enables specific marketing approaches from exclusive events for high engagers to activity prompts for low-engagers.
- **Retention and Growth:** Monitoring movement across segments helps refine strategies for user retention and engagement growth in each group.

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

Ans →

To measure ad campaign effectiveness and optimize future campaigns, we can focus on these key metrics and strategies:

- **Click-Through Rate (CTR):** Measure CTR (clicks/impressions) to gauge ad relevance. Adjust ad content or targeting if CTR is low.
- **Conversion Rate:** Calculate conversions/clicks to assess ad-to-action effectiveness. Low rates suggest refining ad messaging or landing pages.
- **Cost per Conversion (CPC, CPA):** Track CPC and CPA to optimize budget. High CPA may indicate the need to tweak targeting or bidding strategies.
- **Return on Ad Spend (ROAS):** Find ROAS by dividing revenue from conversions by ad spend, guiding budget allocation toward high-return campaigns.
- **A/B Testing:** Test variations in ad elements to identify top performers. Implement best options for incremental improvements in effectiveness.

By regularly analyzing these metrics and using data-driven insights for adjustments, we can maximize ad performance and drive higher returns from each campaign.

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

Ans →

```
WITH user_activity AS (
SELECT
  u.id AS user_id, u.user_name,
  COUNT (DISTINCT p.id) AS total_posts,
  COUNT (DISTINCT l.photo_id) AS total_likes_received,
  COUNT(DISTINCT c.id) AS total_comments_received,
  (SELECT COUNT(*) FROM followers f WHERE f.followee_id = u.id) AS total_followers
FROM users u
LEFT JOIN photos p 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 u.id),

high_engagement_users **AS**(

SELECT

user_id,

user_name,

total_posts,

total_likes_received,

total_comments_received,

total_followers,

(total_posts + total_likes_received + total_comments_received + total_followers) **AS** engagement_score

FROM user_activity

)

SELECT

user_id,

user_name,

engagement_score,

RANK() **OVER** (**ORDER BY** engagement_score **DESC**) **AS** ambassador_rank

FROM high_engagement_users

ORDER BY engagement_score **DESC**

LIMIT 10; -- select top 10 potential brand ambassadors

	user_id	username	engagement_score	ambassador_rank
▶	23	Eveline95	430	1
	59	Cesar93	405	2
	88	Clint27	398	3
	86	Delfina_VonRueden68	368	4
	58	Aurelie71	335	5
	29	Jaime53	322	6
	77	Donald.Fritsch	263	7
	43	Janet.Armstrong	240	8
	52	Zack_Kemmer93	237	9
	13	Alexandro35	234	10

Observation:

- We analyzed trends in likes, comments, and posts to identify active users and then flagged users with high engagement as potential brand advocates.
- Users with the highest engagement_score are most likely to be effective ambassadors, as they have a strong presence and engagement.
- Analyzing top ambassadors' posts can reveal which types of content (e.g., lifestyle, travel) receive the most engagement, helping to tailor campaigns.

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

Ans → Here's how I would approach the Instagram data analysis problem:

- **Understand the Data and Scope:** I'd start by familiarizing myself with the dataset's structure and defining the main goals, like improving user engagement or identifying trends.
- **Define Key Focus Areas:** I would identify essential aspects to analyze, such as user engagement, content popularity, user demographics, or time-based trends.
- **Set Key Performance Indicators (KPIs):** I'd choose metrics that guide my analysis, such as likes per post, follower growth, active users, or top tags.
- **Break Down Objectives:** I'd segment the analysis into categories, including user demographics, engagement patterns, content effectiveness, and social influence.
- **Develop Hypotheses:** I'd formulate possible trends or behaviors (like certain content types performing better on weekends) to test through the data.
- **Design Analytical Approach:** I'd plan the queries and computations needed for each focus area, like calculating engagement rates, identifying high-influence users, or finding peak posting times.
- **Analyze and Refine:** I'd review my initial findings, adjust metrics or hypotheses if necessary, and make sure my results align with the business goals.
- **Summarize and Present Insights:** Finally, I'd compile actionable insights and recommendations, focusing on how these findings can support Instagram's strategic objectives.

10. 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?

Ans →

```
UPDATE User_interactions
SET Engagement_Type = "Heart"
WHERE Engagement_Type = "Like";
```

Observation:

- This mentioned SQL query will find all rows in the "User_Interactions" table where the "Engagement_Type" is "Like" and update them to "Heart".