User Engagement Analysis

Analyse user engagement to find the most engaging posts and users based on the given conditions.

Retrieve the comprehensive count of likes, comments, and shares garnered by a specific post identified by its unique post ID

Calculate the mean number of reactions, encompassing likes, comments, and shares per distinct user within a designated

Identify the three most engaging posts, measured by the aggregate sum of reactions, within the preceding week

Posts:

post_id	post_content	post_date
1	Lorem ipsum dolor sit amet...	2023-08-25 10:00:00
2	Exploring the beauty of nature...	2023-08-26 15:30:00
3	Unveiling the latest tech trends...	2023-08-27 12:00:00
4	Journey into the world of literature...	2023-08-28 09:45:00
5	Capturing the essence of city life...	2023-08-29 16:20:00

UserReactions:

reaction_id	user_id	post_id	reaction_type	reaction_date
1	101	1	like	2023-08-25 10:15:00
2	102	1	comment	2023-08-25 11:30:00
3	103	1	share	2023-08-26 12:45:00
4	101	2	like	2023-08-26 15:45:00
5	102	2	comment	2023-08-27 09:20:00
6	104	2	like	2023-08-27 10:00:00
7	105	3	comment	2023-08-27 14:30:00
8	101	3	like	2023-08-28 08:15:00
9	103	4	like	2023-08-28 10:30:00
10	105	4	share	2023-08-29 11:15:00
11	104	5	like	2023-08-29 16:30:00
12	101	5	comment	2023-08-30 09:45:00

```
create table posts

(post_id int,post_content VARCHAR(100),post_date date);


insert into posts (post_id,post_content,post_date)

values(1,'Lorem ipsum dolor sit amet...','2023-08-25 10:00:00'),
```

```
(2,'Exploring the beauty of nature...','2023-08-26 15:30:00'),
(3,'Unveiling the latest tech trends...','2023-08-27 12:00:00'),
(4,'Journey into the world of literature...','2023-08-28 09:45:00'),
(5,'Capturing the essence of city life...','2023-08-29 16:20:00');

create TABLE UserReaction
(reaction_id int ,user_id int ,post_id int ,reaction_type varchar(100),reaction_date DATE);

insert into UserReaction(reaction_id,user_id,post_id,reaction_type,reaction_date)
VALUES
(1,101,1,'like','2023-08-25 10:15:00'),
(2,102,1,'comment','2023-08-25 11:30:00'),
(3,103,1,'share','2023-08-26 12:45:00'),
(4,101,2,'like','2023-08-26 15:45:00'),
(5,102,2,'comment','2023-08-27 09:20:00'),
(6,104,2,'like','2023-08-27 10:00:00'),
(7,105,3,'comment','2023-08-27 14:30:00'),
(8,101,3,'like','2023-08-28 08:15:00'),
(9,103,4,'like','2023-08-28 10:30:00'),
(10,105,4,'share','2023-08-29 11:15:00'),
(11,104,5,'like','2023-08-29 16:30:00'),
(12,101,5,'comment','2023-08-30 09:45:00');



SELECT p.post_id ,p.post_content,
count(case when u.reaction_type = 'like' then 1 end )as likes,
count(case when u.reaction_type ='comment' then  1 end) as comments,
count(case when u.reaction_type = 'share' then 1 end) as shares
from posts as p
```

```
left join UserReaction as u on p.post_id=u.post_id
group by p.post_id,p.post_content
order by p.post_id;


 SELECT
    DATE(u.reaction_date) AS reaction_day,
    COUNT(DISTINCT u.user_id) AS distinct_users,
    COUNT(*) AS total_reactions,
    AVG(COUNT(*)) OVER (PARTITION BY DATE(u.reaction_date)) AS avg_reactions_per_user

FROM  UserReaction as u
WHERE u.reaction_date BETWEEN '2023-08-25' AND '2023-08-29'
GROUP BY reaction_day;


create view PostEngagement as (
SELECT p.post_id, p.post_content,p.post_date,count(u.reaction_id) as reactions from posts as p
left join UserReaction as u on p.post_id=u.post_id
where  u.reaction_date >= CURRENT_DATE - interval '1 WEEK' and u.reaction_date <= CURRENT_DATE
group by p.post_id,p.post_content, p.post_date);

select post_id, post_content, post_date,reactions from PostEngagement
order by reactions desc
limit 3;
```