**PROJECT ON INSTAGRAM DATABASE**

**(by Nisha Jagtap)**

1**) How many times does the average user post?**

**select avg(post) from (select count(photos.id) as post from users**

**join photos on users.id = photos.user\_id group by username) as h ;**

**2) Find the top 5 most used hashtags**

**select tag\_name, count(\*) as c from photo\_tags join tags on**

**photo\_tags.tag\_id = tags.id group by tag\_id, tag\_name order by c desc limit 5;**

**3) Find users who have liked every single photo on the site**

**select u.username, count(photo\_id) as no\_of\_photos from likes l join photos**

**p on l.photo\_id = p.id join users u on l.user\_id = u.id**

**group by username having no\_of\_photos =**

**(select count(distinct photo\_id) from likes);**

**4) Retrieve a list of users along with their usernames and**

**the rank of their account creation, ordered by the creation date in ascending order;**

**select user\_id, username, users.created\_at,**

**rank () over (order by users.created\_at) as 'rank' from users**

**inner join photos on users.id = photos.user\_id group by user\_id, username ;**

**5) List the comments made on photos with their comment texts, photo URLs, and usernames of users who posted the comments. Include the comment count for each photo;**

**select u.username, c.comment\_text, p.image\_url from photos p**

**join comments c on p.id = c.photo\_id**

**join users u on c.user\_id = u.id**

**group by u.username, c.comment\_text, p.image\_url;**

**select image\_url,photo\_id, count(comment\_text) as c\_count from photos**

**join comments on photos.id = comments.photo\_id group by image\_url, photo\_id;**

**6) For each tag, show the tag name and the number of photos associated with that tag. Rank the tags by the number of photos in descending order;**

**select t.tag\_name , count(image\_url) as no\_of\_photos from tags t**

**join photo\_tags pt on t.id = pt.tag\_id**

**join photos p on pt.photo\_id = p.id**

**join users u on p.user\_id = u.id**

**group by t.tag\_name order by no\_of\_photos desc;**

**7) List the usernames of users who have posted photos along with the count**

**of photos they have posted. Rank them by the number of photos in descending**

**order.**

**select u.username ,count(p.image\_url) as no\_of\_photos from photos p**

**join users u on p.user\_id = u.id group by u.username**

**order by no\_of\_photos desc;**

**8) Display the username of each user along with the creation date of**

**their first posted photo and the creation date of their next posted photo.**

**select username, created\_at,**

**lag(created\_at) over (order by created\_at) as first\_post,**

**lead(created\_at) over (order by created\_at) as next\_post from users;**

**9) For each comment, show the comment text, the username of the commenter,**

**and the comment text of the previous comment made on the same photo.**

**select comments.user\_id, username, image\_url, comment\_text,**

**lag(comment\_text) over (partition by image\_url) as previous\_comment from comments**

**join users on comments.user\_id = users.id**

**join photos on users.id = photos.user\_id**

**group by comments.user\_id, username, image\_url, comment\_text ;**

**10) Show the username of each user along with the number of photos they have posted and the number of photos posted by the user before them and after them, based on the creation date.**

**select p.user\_id, u.username , count(p.image\_url) as no\_of\_photos,**

**lag(count(p.image\_url)) over(order by 'p.created\_at') as before\_users ,**

**lead(count(p.image\_url)) over(order by 'p.created\_at') as after\_users from photos p**

**join users u on p.user\_id = u.id group by p.user\_id, u.username;**