Task 1:Find the 5 oldest users.

Solution:

SELECT \* FROM `users` ORDER BY created\_at limit 5;

Task 2:

a). What day of the week do most users register on?

b). We need to figure out when to schedule an ad campgain

Solution:

SELECT dayname(created\_at) as `day of the week`,COUNT(\*) as `total registeration` FROM `users` group by `day of the week` order by `total registeration` desc limit 10;

Task 3:

We want to target our inactive users with an email campaign. Find the users who have never posted a photo.

Solution:

SELECT username FROM users left join photos on users.id=photos.user\_id where photos.user\_id is null;

Task 4:

We're running a new contest to see who can get the most likes on a single photo. WHO WON?

Solution:

SELECT username,photos.id AS id,photos.image\_url,COUNT(\*) AS Total\_Likes FROM photos JOIN likes ON photos.id = likes.photo\_id JOIN users ON users.id = photos.user\_id GROUP BY id ORDER BY Total\_Likes DESC LIMIT 1;

Task 5:

Our Investors want to know...How many times does the average user post? (total number of photos/total number of users)

Solution:

select ROUND((SELECT COUNT(\*)FROM photos)/(SELECT COUNT(\*) FROM users),2);

Task 6:

user ranking by postings higher to lower

Solution:

select username,count(photos.image\_url) from users inner join photos on users.id=photos.user\_id group by users.id order by count(photos.image\_url) desc ;

Task 7:

Total Posts by users (longer versionof SELECT COUNT(\*)FROM photos)

Solution:

select count(\*) from photos;

Task 8:

Total numbers of users who have posted at least one time

Solution:

select count(distinct user\_id) from photos;

Task 9:

A brand wants to know which hashtags to use in a post. What are the top 5 most commonly used hashtags?

Solution:

SELECT tag\_name,count(tag\_id)as total FROM tags inner join photo\_tags on tags.id=photo\_tags.tag\_id group by tag\_name order by total DESC ;

Task 10:

We have a small problem with bots on our site. Find users who have liked every single photo on the site

Solution:

SELECT user\_id as id,username,count(\*) as total\_likes\_by\_user FROM likes inner join users on likes.user\_id=users.id GROUP BY user\_id HAVING COUNT(\*) = (SELECT COUNT(\*) FROM photos);

Task 11:

We also have a problem with celebrities. Find users who have never commented on a photo

Solution:

select distinct(username),comment\_text from users left join comments on users.id=comments.user\_id where comment\_text is null;

Task 12:

Are we overrun with bots and celebrity accounts? Find the percentage of our users who have either never commented on a photo or have commented on every photo

Solution:

SELECT tableA.total\_A AS 'Number Of Users who never commented',(tableA.total\_A/(SELECT COUNT(\*) FROM users))\*100 AS '%',tableB.total\_B AS 'Number of Users who likes on every photos',(tableB.total\_B/(SELECT COUNT(\*) FROM users))\*100 AS '%'FROM(SELECT COUNT(\*) AS total\_A FROM(SELECT username,comment\_text FROM users LEFT JOIN comments ON users.id =comments.user\_id GROUP BY users.id HAVING comment\_text IS NULL) AS total\_number\_of\_users\_without\_comments) AS tableA JOIN(SELECT COUNT(\*) AS total\_B FROM(SELECT user\_id as id,username,count(\*) as total\_likes\_by\_user FROM likes inner join users on likes.user\_id=users.id GROUP BY user\_id HAVING COUNT(\*) = (SELECT COUNT(\*) FROM photos)) AS total\_number\_users\_with\_likes)AS tableB;

Task 13:

Find users who have ever commented on a photo

Solution:

SELECT username,comment\_text FROM users LEFT JOIN comments ON users.id = comments.user\_id GROUP BY users.id HAVING comment\_text IS NOT NULL;

Task 14:

Are we overrun with bots and celebrity accounts? Find the percentage of our users who have either never commented on a photo or have commented on photos before.

Solution:

SELECT tableA.total\_A AS 'Number Of Users who never commented',(tableB.total\_B/(SELECT COUNT(\*) FROM users))\*100 AS '%',tableB.total\_B AS 'Number of Users who commented on photos' FROM(SELECT COUNT(\*) AS total\_A FROM (SELECT username,comment\_text FROM users LEFT JOIN comments ON users.id = comments.user\_id GROUP BY users.id HAVING comment\_text IS NULL) AS total\_number\_of\_users\_without\_comments) AS tableA JOIN (SELECT COUNT(\*) AS total\_B FROM(SELECT username,comment\_text FROM users LEFT JOIN comments ON users.id = comments.user\_id GROUP BY users.id HAVING comment\_text IS NOT NULL) AS total\_number\_users\_with\_comments)AS tableB;