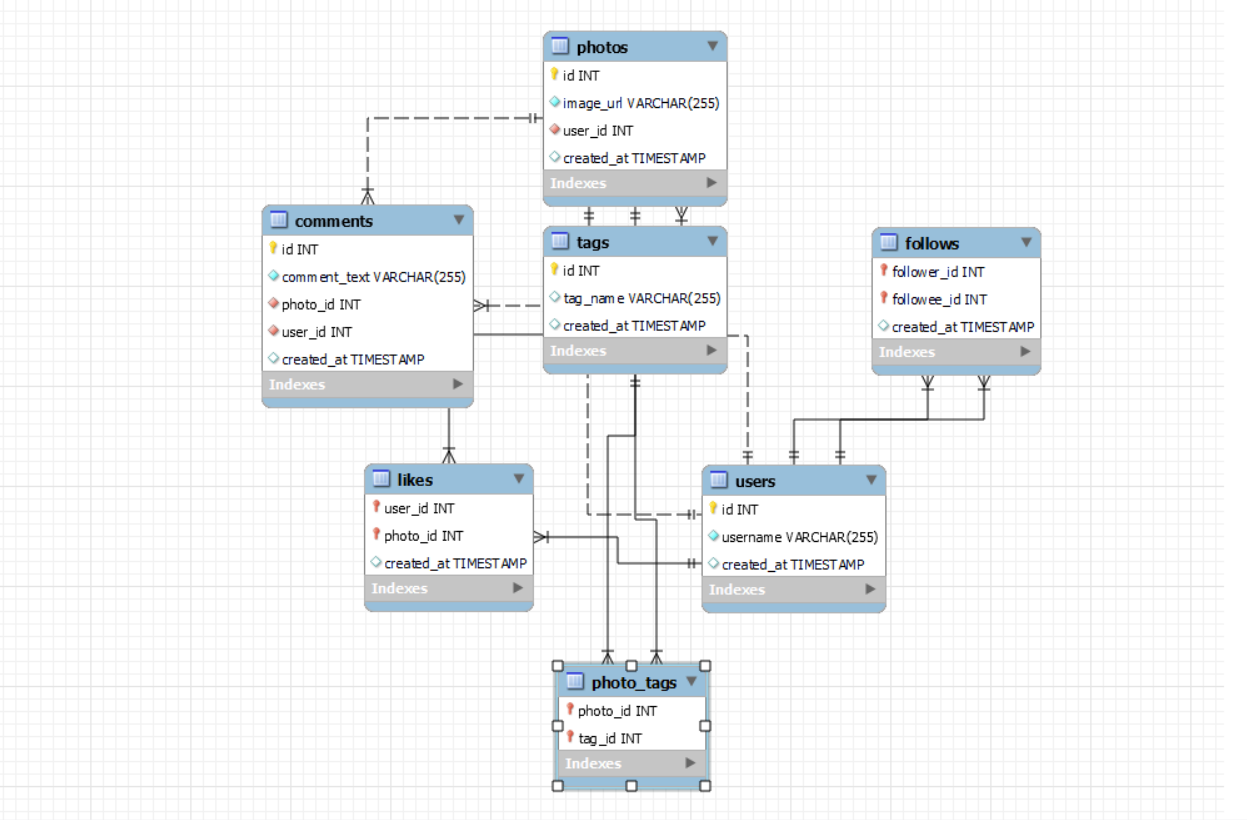
1. Create an ER diagram or draw a schema for the given database.



Select \* from comments;

select \* from follows;

select \* from likes;

select \* from photo\_tags;

select \* from photos;

select \* from tags;

select \* from users;

**TASK-2:** We want to reward the user who has been around the longest, Find the 5 oldest users.

Select \* from users order by created\_at limit 5;

**TASK-3:** To target inactive users in an email ad campaign, find the users who have never posted a photo.

select distinct(user\_id) from photos;

select \* from users u left join photos p on u.id=p.user\_id where p.id is null;

select \* from users where id not in (select distinct(user\_id) from photos);

**TASK-4:** Suppose you are running a contest to find out who got the most likes on a photo. Find out who won?

select \* from users;

select \* from likes;

select \* from photos;

select distinct(photo\_id) from likes;

select photo\_id,count(user\_id) AS total\_likes from likes group by photo\_id order by total\_likes desc limit 1;

select photo\_id from (select photo\_id,count(user\_id) AS total\_likes from likes group by photo\_id order by total\_likes desc limit 1) As photo;

SELECT

\*

FROM

users u

JOIN

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

WHERE

p.id = (SELECT

photo\_id

FROM

(SELECT

photo\_id, COUNT(user\_id) AS total\_likes

FROM

likes

GROUP BY photo\_id

ORDER BY total\_likes DESC

LIMIT 1) AS photo);

**TASK-5:** The investors want to know how many times does the average user post.

select \* from users;

select \* from photos;

select user\_id,count(id) AS no\_of\_posts from photos group by user\_id;

select sum(no\_of\_posts)/count(user\_id) AS Average\_posts from (select user\_id,count(id) AS no\_of\_posts from photos group by user\_id) As posts;

**TASK-6:** A brand wants to know which hashtag to use on a post, and find the top 5 most used hashtags.

select \* from tags;

select \* from photo\_tags;

select tag\_id,count(photo\_id)As count\_of\_tagged\_photos from photo\_tags group by tag\_id order by count\_of\_tagged\_photos desc limit 5;

select tag\_id from (select tag\_id,count(photo\_id)As count\_of\_tagged\_photos from photo\_tags group by tag\_id order by count\_of\_tagged\_photos desc limit 5) As tag\_pics;

select id,tag\_name from tags where id IN (select tag\_id from (select tag\_id,count(photo\_id)As count\_of\_tagged\_photos from photo\_tags group by tag\_id order by count\_of\_tagged\_photos desc limit 5) As tag\_pics);

**TASK-7:** To find out if there are bots, find users who have liked every single photo on the site.

select \* from users;

select \* from likes;

select \* from photos;

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

select user\_id,count(user\_id) As total\_likes\_userwise from likes group by user\_id;

select user\_id,count(user\_id) As total\_likes\_userwise from likes group by user\_id having total\_likes\_userwise = (select count(distinct(photo\_id)) from likes);

**TASK-8:** Find the users who have created instagramid in may and select top 5 newest joinees from it?

select \* from users;

select \* from users where month(created\_at)=5 order by created\_at desc limit 5;

**TASK-9:** Can you help me find the users whose name starts with c and ends with any number and have posted the photos as well as liked the photos?

select \* from users;

select \* from photos;

select \* from likes;

select \* from users where username regexp "^c" and username regexp "[0123456789]$";

select \* from users u join photos p on u.id=p.user\_id where username regexp "^c" and username regexp "[0123456789]$";

select \* from users u join photos p on u.id=p.user\_id join likes l on u.id=l.user\_id where username regexp "^c" and username regexp "[0123456789]$";

select u.id,u.username,p.id,p.image\_url,l.photo\_id from users u join photos p on u.id=p.user\_id join likes l on u.id=l.user\_id where username regexp "^c" and username regexp "[0123456789]$";

**TASK-10:** Demonstrate the top 30 usernames to the company who have posted photos in the range of 3 to 5.

select user\_id,count(id) AS no\_of\_posts from photos group by user\_id;

select user\_id,count(id) AS no\_of\_posts from photos group by user\_id having no\_of\_posts between 3 AND 5;

select p.user\_id,count(p.id) AS no\_of\_posts,u.username from photos p join users u on u.id=p.user\_id group by user\_id having no\_of\_posts between 3 AND 5;