INSTAGRAM USER ANALYTICS

PROJECT DESCRIPTION

To achieve these goals, I want to look at the 5 oldest Instagram users and find those who haven't posted recently. Another task is to determine the winner of the contest based on the number of likes on a photo, while determining the top 5 hashtags used. It will also find the most suitable day to run the campaign based on the day with the most users on the platform.

In addition, investors who want to invest in Instagram want to know if the platform is in good standing, so they need to know something about it. So, to find an understanding for this, I need to do some questions related to the question

TASKS

- Find the 5 oldest users of Instagram from the database.

```
Select* from users
Order by created_at  ASC
Limit 5;
```

- Find the users who have never posted a single photo on Instagram.

```
select users.id, username from users left JOIN
on users.id photos.user_id
where image_url is Null;
```

- To identify the winner of the contest and provide their details to the team.

```
select users.id, username, likes.photo_id, count(likes.user_id) as Most_Likes From users
join photos
on photos.user_id = users.id
join likes
on photos.id = likes.photo_id
group by photo_id
order by count(user_id) DESC
limit 1;
```

- Identify the 5 most suitable hashtags that are most commonly used on the platform.

```
SELECT TAGS.ID, TAGS.TAG_NAME, COUNT(TAG_ID) FROM TAGS
JOIN PHOTO_TAGS
ON TAGS.ID = PHOTO_TAGS.TAG_ID
GROUP BY TAG_ID
ORDER BY COUNT(TAG_ID) DESC
Limit 5;
```

- What day of the week do most user register on ? Provide insights on when to schedule the launch campaign of the Ads.

```
SELECT DAYNAME(created_at), count(*)
FROM users
GROUP BY DAYNAME(created_at)
Order BY COUNT(*) DESC;
```

- Provide how many times does average user post on Instagram . Also , provide total no of photos on Instagram /total no of users.

```
select avg(cnt) as average
from (select user_id, count(*) as cnt
from photos
group by user_id
order by cnt desc)sub
```

- Total number of posts

```
select count(id) from photos;
```

- Total number of users

```
select count(id) from users;
```

- Provide data on users (bots) who have liked every single photo on the site (since the account is been created as normal user cannot be able to do this.

```
select username as Bots, count(*) as likes_on_photos from likes
inner join users on users.id = user_id
group by user_id
having likes_on_photos = max(photo_id);
```