Instagram User Analytics

---SQL Fundamentals

Project Description - This is a project about Instagram User Analytics by Trainity in which I have worked on to gather business insights for marketing, product & development teams by providing the data to the team by using SQL.

In this process, I have used several SQL queries to get the desired data through the project. I queried the top oldest user, whether the user posted any photos or not, most commonly used hashtags, total number of users and many several insights of the Instagram platform from the data/schema provided to me.

Approach- I understood the task and what data the team needs. I imported the data/Schema into SQL and performed several queries to understand the data then finding the insights which the team needs to perform their marketing and development works.

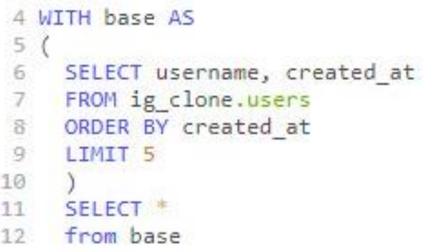
Tech-Stack Used- DB Fiddle (MySQL v8.0)

Insights- I've performed several SQL commands to get insights and achieved a knowledge of how to perform a real time SQL query on this project. I understood that the data given is of Instagram users and I derived several insights about total users of Instagram, oldest user and is there any fake account present on Instagram, etc.

Result- I extracted the relevant data from the given data source and it helped me understand intermediate SQL commands. The results are shown below.

A) ***Marketing-***

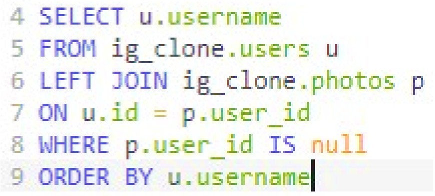
Task 1 - Find the 5 oldest users of the Instagram from the database provided.



```
WITH base AS
(
  SELECT username, created_at
  FROM ig_clone.users
  ORDER BY created_at
  LIMIT 5
  )
  SELECT *
  from base
```

Username	Created_at
Darby_Herzog	2016-05-06 00:14:21
Emilio_Bernier52	2016-05-06 13:04:30
Elenor88	2016-05-08 01:30:41
Nicole71	2016-05-09 17:30:22
Jordyn.Jacobson	2016-05-14 07:56:26

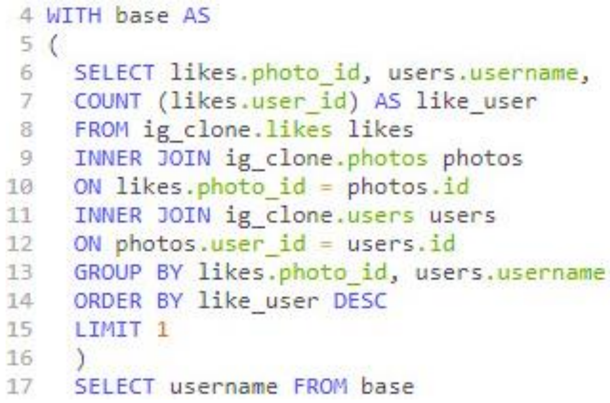
Task 2 - Find the users who have never posted a single photo on Instagram.



```
SELECT u.username
FROM ig_clone.users u
LEFT JOIN ig_clone.photos p
ON u.id = p.user_id
WHERE p.user_id IS null
ORDER BY u.username
```

Username		
Aniya_Hackett	Bartholome.Bernhard	Bethany20
Darby_Herzog	David.Osinski47	Duane60
Esmeralda.Mraz57	Esther.Zulauf61	Franco_Keebler64
Hulda.Macejkovic	Jaclyn81	Janelle.Nikolaus81

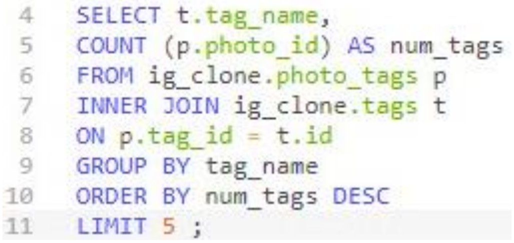
Task 3 - Identify the winner of the contest and provide their details to the team.



```
WITH base AS
(
  SELECT likes.photo_id, users.username,
  COUNT (likes.user_id) AS like_user
  FROM ig_clone.likes likes
  INNER JOIN ig_clone.photos photos
  ON likes.photo_id = photos.id
  INNER JOIN ig_clone.users users
  ON photos.user_id = users.id
  GROUP BY likes.photo_id, users.username
  ORDER BY like_user DESC
  LIMIT 1
  )
  SELECT username FROM base
```

username
Zach_Kemmer93

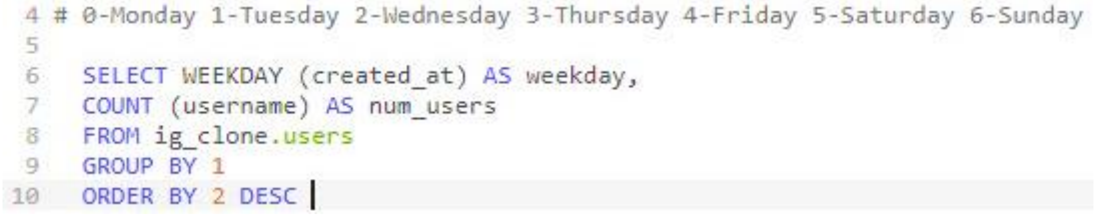
Task 4 - Identify and suggest the top 5 most commonly used hashtags on the platform.



```sql
SELECT t.tag_name,
COUNT (p.photo_id) AS num_tags
FROM ig_clone.photo_tags p
INNER JOIN ig_clone.tags t
ON p.tag_id = t.id
GROUP BY tag_name
ORDER BY num_tags DESC
LIMIT 5 ;
```

Tag_name	**Num_tags**
Smile	59
Beach	42
Party	39
Fun	38
Concert	24

Task 5 - What day of the week do most users register on? Provide insights on when to schedule an ad campaign.



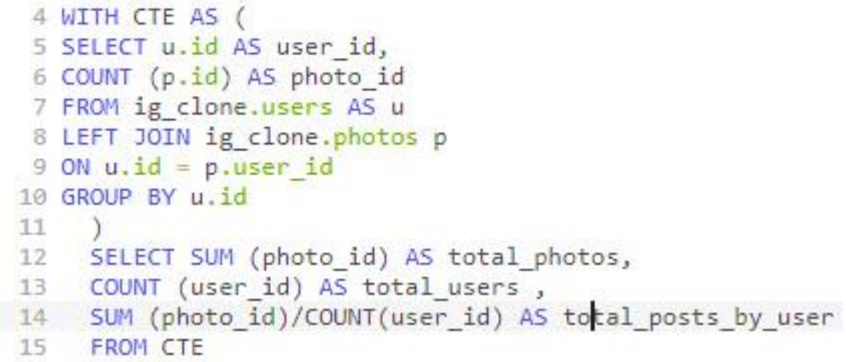
```sql
# 0-Monday 1-Tuesday 2-Wednesday 3-Thursday 4-Friday 5-Saturday 6-Sunday

SELECT WEEKDAY (created_at) AS weekday,
COUNT (username) AS num_users
FROM ig_clone.users
GROUP BY 1
ORDER BY 2 DESC
```

Weekday	Num_users
3	16
6	16
4	15
1	14
0	14
2	13
5	12

B) *Investor Metrics –*

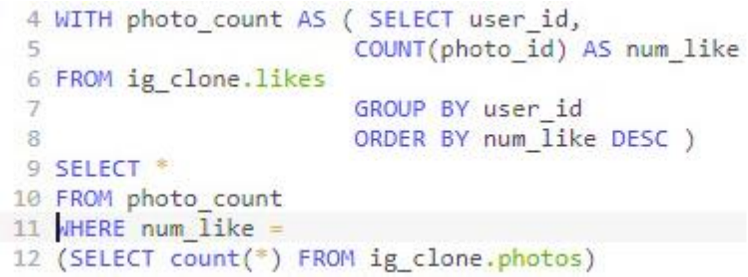
Task 1- Provide how many times does average user posts on Instagram. Also, provide the total number of photos on Instagram/total number of users.



```
WITH CTE AS (
SELECT u.id AS user_id,
COUNT (p.id) AS photo_id
FROM ig_clone.users AS u
LEFT JOIN ig_clone.photos p
ON u.id = p.user_id
GROUP BY u.id
  )
  SELECT SUM (photo_id) AS total_photos,
  COUNT (user_id) AS total_users ,
  SUM (photo_id)/COUNT(user_id) AS total_posts_by_user
  FROM CTE
```

Total_photos	Total_users	Total_posts_by_user
257	100	2.5700

Task 2- Provide data on users (bots) who have liked every single photo on the site (since any normal user would not be able to do this).



```
WITH photo_count AS ( SELECT user_id,
                      COUNT(photo_id) AS num_like
FROM ig_clone.likes
                      GROUP BY user_id
                      ORDER BY num_like DESC )
SELECT *
FROM photo_count
WHERE num_like =
(SELECT count(*) FROM ig_clone.photos)
```

User_id	Num_likes
75	257
21	257
24	257
91	257
36	257
41	257
14	257
76	257
54	257
57	257
66	257
5	257
71	257