Trainity Assignment-2

Muthiah Sivavelan
Ph:8525021258

Project Description:

This project involves analyzing user interactions and engagement with the Instagram app to provide valuable insights that can help the business grow. The insights derived from this analysis can be used by various teams within the business. For example, the marketing team might use these insights to launch a new campaign, the product team might use them to decide on new features to build, and the development team might use them to improve the overall user experience. In this project, I used SQL and MySQL Workbench as my tool to analyze Instagram user data and answer questions posed by the management team.

The approach I took for providing the insights regarding various queries by marketing team and development team are as follows:

- I created a database and created various tables in it.
- I filled every table with their given data
- I carefully analyzed the attributes in each table and their relationships with other table (which helped in joining the tables).
- I meticulously drafted sql queries based on the questions given. I created views in some queries to make the queries easy to read and simple.

Tasks:

A) Marketing Analysis:

1) **Loyal User Reward:** The marketing team wants to reward the most loyal users, i.e., those who have been using the platform for the longest time. Your Task: *Identify the five oldest users on Instagram from the provided database.*

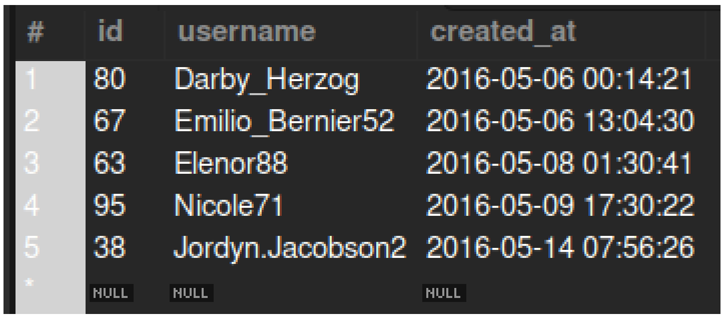
Approach:

My task is to identify the five oldest users on Instagram from the database. I knew that the users table contains the user id, username and the date at which the user created his/her profile(created_at attribute). So I decided to sort the rows based on the created_at attribute and then select all the attributes from the result set and limit the output by maximum of 5 rows(using the LIMIT keyword).

Query:

```
/* First qn: 5 oldest users of instagram_ the loyal user reward*/

select * from users
order by created_at LIMIT 5;
```

Output:



#	id	username	created_at
1	80	Darby_Herzog	2016-05-06 00:14:21
2	67	Emilio_Bernier52	2016-05-06 13:04:30
3	63	Elenor88	2016-05-08 01:30:41
4	95	Nicole71	2016-05-09 17:30:22
5	38	Jordyn.Jacobson2	2016-05-14 07:56:26
*	NULL	NULL	NULL

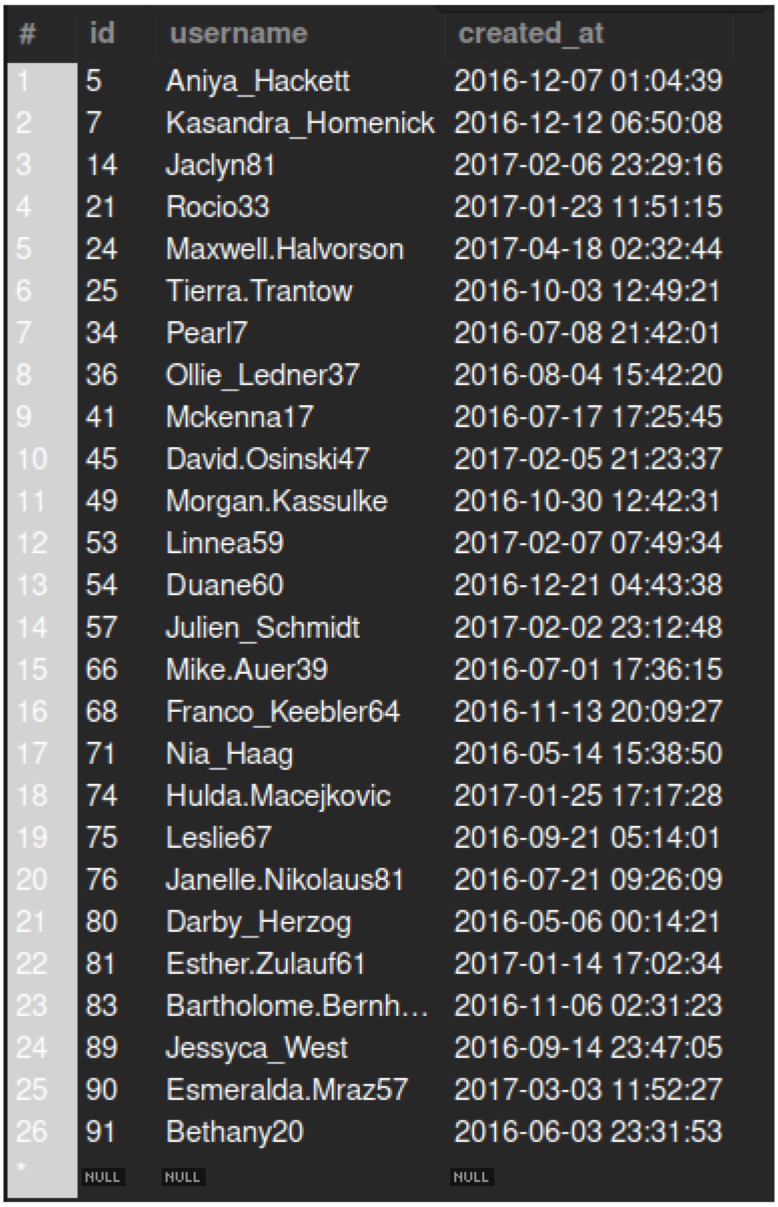
2) **Inactive User Engagement:** The team wants to encourage inactive users to start posting by sending them promotional emails. Your Task: Identify users who have never posted a single photo on Instagram.

Approach:
I need to identify the users who have never posted a single photo on Instagram. Joining users and the photos table based on the user id would give me a table that contains users and the photo they have posted. Therefore all those users whose id is not present in this table are those users who have never posted a single photo on Instagram. Hence, the following query was drafted which solves this problem.

Query:

```
/* Identify users who have never posted a single photo on instagram */

select * from users
where users.id not in
    (select users.id from users inner join photos
    on users.id=photos.user_id);
```

Output:



#	id	username	created_at
1	5	Aniya_Hackett	2016-12-07 01:04:39
2	7	Kasandra_Homenick	2016-12-12 06:50:08
3	14	Jaclyn81	2017-02-06 23:29:16
4	21	Rocio33	2017-01-23 11:51:15
5	24	Maxwell.Halvorson	2017-04-18 02:32:44
6	25	Tierra.Trantow	2016-10-03 12:49:21
7	34	Pearl7	2016-07-08 21:42:01
8	36	Ollie_Ledner37	2016-08-04 15:42:20
9	41	Mckenna17	2016-07-17 17:25:45
10	45	David.Osinski47	2017-02-05 21:23:37
11	49	Morgan.Kassulke	2016-10-30 12:42:31
12	53	Linnea59	2017-02-07 07:49:34
13	54	Duane60	2016-12-21 04:43:38
14	57	Julien_Schmidt	2017-02-02 23:12:48
15	66	Mike.Auer39	2016-07-01 17:36:15
16	68	Franco_Keebler64	2016-11-13 20:09:27
17	71	Nia_Haag	2016-05-14 15:38:50
18	74	Hulda.Macejkovic	2017-01-25 17:17:28
19	75	Leslie67	2016-09-21 05:14:01
20	76	Janelle.Nikolaus81	2016-07-21 09:26:09
21	80	Darby_Herzog	2016-05-06 00:14:21
22	81	Esther.Zulauf61	2017-01-14 17:02:34
23	83	Bartholome.Bernh...	2016-11-06 02:31:23
24	89	Jessyca_West	2016-09-14 23:47:05
25	90	Esmeralda.Mraz57	2017-03-03 11:52:27
26	91	Bethany20	2016-06-03 23:31:53
*	NULL	NULL	NULL

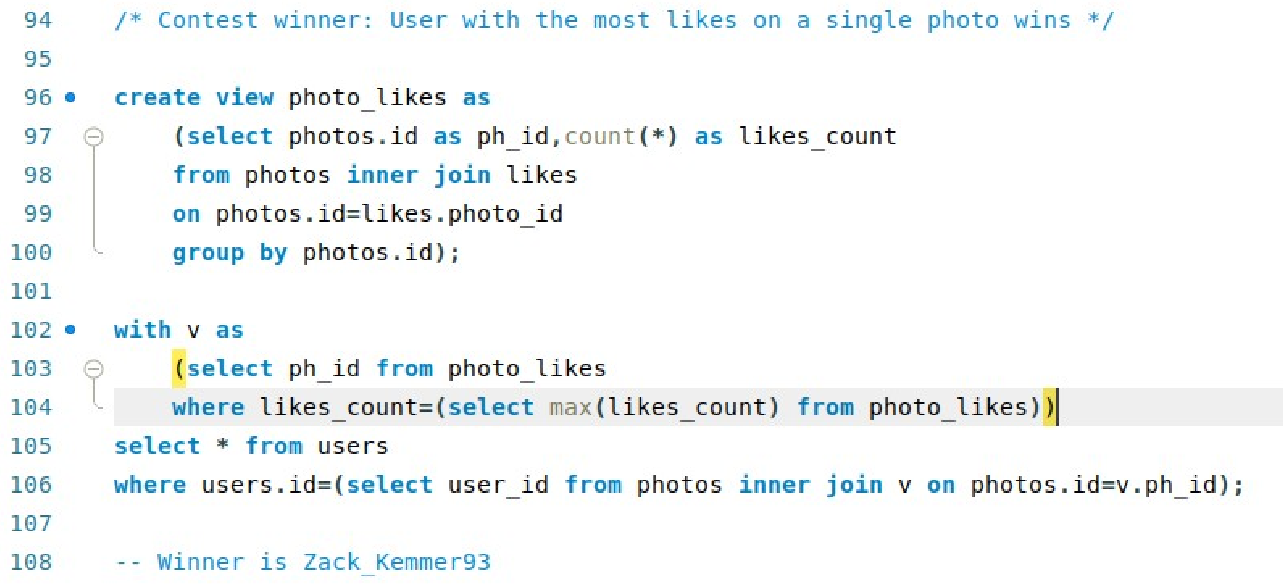
3) **Contest Winner Declaration:** The team has organized a contest where the user with the most likes on a single photo wins.
Your Task: Determine the winner of the contest and provide their details to the team.

Approach:

I need to first find the photo with the maximum likes. Then I will get the user id from the resultant table and based on the user id, I will get the details of the user from the user table. The steps I took for the above approach are as follows:

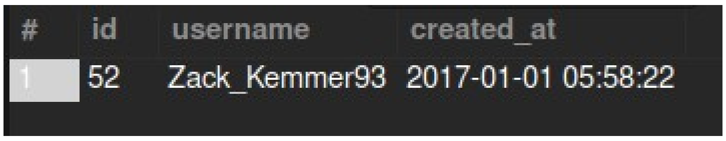
- I first created a view named photo_likes which contains the photo id and the number of likes in that photo(join photos and likes table and then group by photo id).
- I then created a temporary table from the above view selecting only the photo id which contains the maximum number of likes. I joined this temporary table with photos to find the details of the photo that has this photo id. From the details I found out the user id that has posted this photo and based on this user id, I found out the details of the user (using a nested query). This user is the winner of the competition.

Query:



```sql
/* Contest winner: User with the most likes on a single photo wins */

create view photo_likes as
    (select photos.id as ph_id,count(*) as likes_count
    from photos inner join likes
    on photos.id=likes.photo_id
    group by photos.id);

with v as
    (select ph_id from photo_likes
    where likes_count=(select max(likes_count) from photo_likes))
select * from users
where users.id=(select user_id from photos inner join v on photos.id=v.ph_id);

-- Winner is Zack_Kemmer93
```

Output:



#	id	username	created_at
1	52	Zack_Kemmer93	2017-01-01 05:58:22

The contest winner is Zack_Kemmer93.

4) **Hashtag Research:** A partner brand wants to know the most popular hashtags to use in their posts to reach the most people. Your Task: Identify and suggest the top five most commonly used hashtags on the platform.

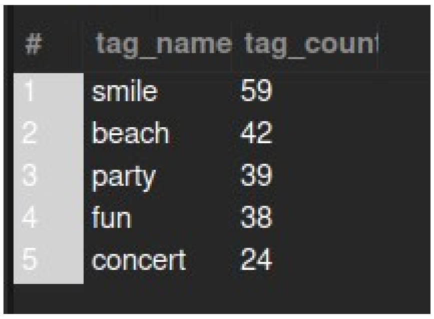
Approach:

I joined the tags and photo_tags table based on the tag id and filtered the tag id and the number of rows with that tag id (using group by tag_id) (which I named it as tag_count). Now I want the tag name for the 5 most commonly used hashtags. Therefore from the above table I decided to sort the resultant table based on the tag_count in descending order and I limited the number of rows by 5 since we require only the 5 most commonly used hashtags.

Query:

```
/* Find the top five most commonly used hastags */

select tags.tag_name, count(*) as tag_count
from tags inner join photo_tags
on photo_tags.tag_id=tags.id
group by tag_id
order by tag_count desc LIMIT 5;

-- Top 5 most commonly used hashtags are smile,beach,party,fun and concert
```

Output:



#	tag_name	tag_count
1	smile	59
2	beach	42
3	party	39
4	fun	38
5	concert	24

5) **Ad Campaign Launch:** The team wants to know the best day of the week to launch ads. Your Task: Determine the day of the week when most users register on Instagram. Provide insights on when to schedule an ad campaign.

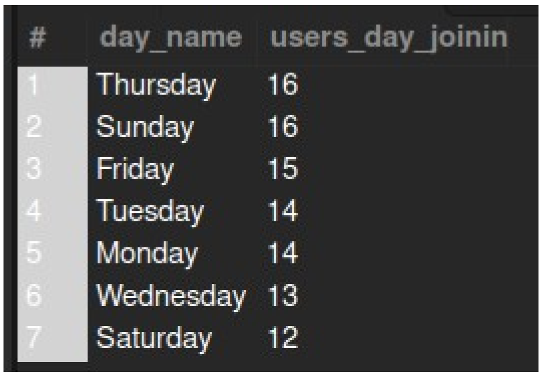
Approach:

The dayname function gives the day name of a given date in mysql. Hence, I decided to use the dayname function on created_at attribute of users table to know about the day the users are joining. I used group by day_name and found the number of users joining on each day of the week (using count(*)). Then I sorted the resultant table in descending order based on the number of users joining on each day of the week.

Query:

```
/* Determine the day of the week when most users register on Instagram */

SELECT dayname(created_at) as day_name, count(*) as users_day_joining
from users
group by day_name
order by users_day_joining desc;

-- Result: Ad campaign can be held mainly on Thursdays and Sundays.
```

Output:



#	day_name	users_day_joinin
1	Thursday	16
2	Sunday	16
3	Friday	15
4	Tuesday	14
5	Monday	14
6	Wednesday	13
7	Saturday	12

The days that has the more users joining are Thursday and Sunday. If we need more days for the ad campaign then we can choose the days from top to bottom as our priority for ad campaign.

B) Investor Metrics:

1) User Engagement: Investors want to know if users are still active and posting on Instagram or if they are making fewer posts. Your Task: Calculate the average number of posts per user on Instagram. Also, provide the total number of photos on Instagram divided by the total number of users.

Approach:

I created a view named user_posts_2 which combines the users and photos table based on the user id and selects the user_id and count(user_id) as noOfPosts which gives the number of rows containing that user id. This view will help in answering both the above questions.

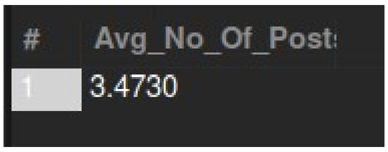
For calculating the average number of posts per user on Instagram, this will include only those users who post (active users). Hence, we can directly select the avg(noOfPosts) from the above view which gives us the average number of posts per user on Instagram.

We are then required to find the total number of photos on Instagram divided by the total number of users. For this, we find the total number of posts from the above created post and then we find the total number of users from the users table (using with subquery clause) and then we find the ratio of both from the cartesian product of the view and users table.

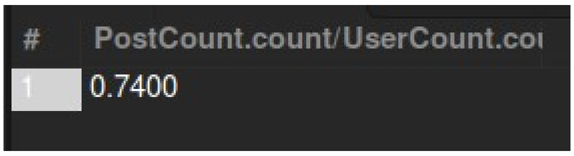
Query:

```sql
/* Calculate the average number of posts per user on Instagram.
Also, provide the total number of photos on Instagram divided by
the total number of users */

create view user_posts_2 as (
select photos.user_id as userId,(count(*)) as noOfPosts
from users inner join photos
on users.id=photos.user_id
group by photos.user_id);

select avg(noOfPosts) as Avg_No_Of_Posts from user_posts_2;

with PostCount as (select count(noOfPosts) as count from user_posts_2),
UserCount as (select count(*) as count from users)
select PostCount.count/UserCount.count from PostCount,UserCount;
```

Output:



```
#   Avg_No_Of_Post
1   3.4730
```



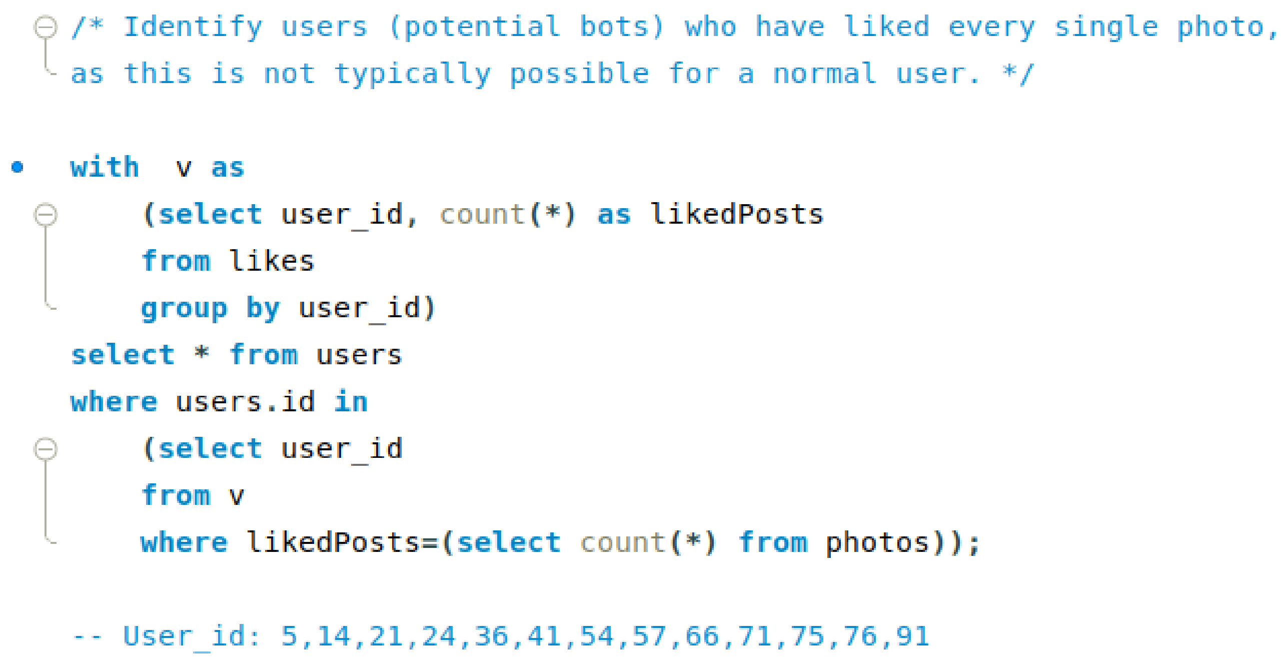
```
#   PostCount.count/UserCount.co
1   0.7400
```

From the above query, I found out that the average number of posts per user on Instagram is 3.4730 and the total number of posts divided by total number of users on Instagram is 0.7400

2) **Bots & Fake Accounts:** Investors want to know if the platform is crowded with fake and dummy accounts. Your Task: Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.

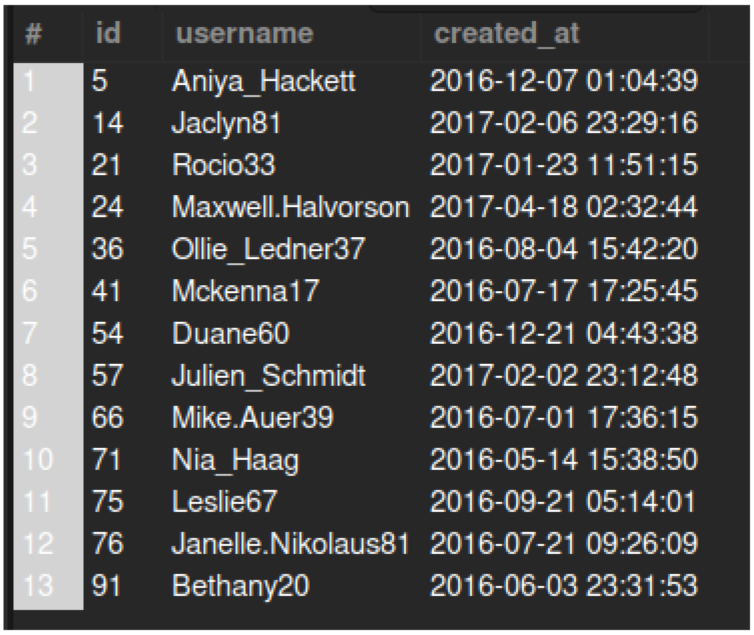
Approach:
I created a temporary table named v which contains the user id along with the number of rows present in the likes table for that user id as likedPosts. This gives us the idea of how many photos are liked by a given user id. Now, I filter out those user id from the above table who has the count of likes equal to the total count of photos in the photos table. Then, I select the details of the users with this user id and give it to development team as they are potentially bots/fake accounts.

Query:



```
/* Identify users (potential bots) who have liked every single photo,
as this is not typically possible for a normal user. */

with  v as
    (select user_id, count(*) as likedPosts
    from likes
    group by user_id)
select * from users
where users.id in
    (select user_id
    from v
    where likedPosts=(select count(*) from photos));

-- User_id: 5,14,21,24,36,41,54,57,66,71,75,76,91
```

Output:



#	id	username	created_at
1	5	Aniya_Hackett	2016-12-07 01:04:39
2	14	Jaclyn81	2017-02-06 23:29:16
3	21	Rocio33	2017-01-23 11:51:15
4	24	Maxwell.Halvorson	2017-04-18 02:32:44
5	36	Ollie_Ledner37	2016-08-04 15:42:20
6	41	Mckenna17	2016-07-17 17:25:45
7	54	Duane60	2016-12-21 04:43:38
8	57	Julien_Schmidt	2017-02-02 23:12:48
9	66	Mike.Auer39	2016-07-01 17:36:15
10	71	Nia_Haag	2016-05-14 15:38:50
11	75	Leslie67	2016-09-21 05:14:01
12	76	Janelle.Nikolaus81	2016-07-21 09:26:09
13	91	Bethany20	2016-06-03 23:31:53

Tech Stack Used:

I have used MySQL Workbench for my project. MySQL Workbench is cross platform and it offers a modern and intuitive graphical user interface (GUI) that combines all the functionalities in a single window. It provides visual representation of databases, schema design tools, SQL editor, and query execution capabilities.

Insights:

- The 5 oldest users of instagram are the users with user id: 80, 67, 63, 95 and 38 who can be awarded the loyal user reward.
- I have gained insights on the users who have never posted even a single photo and the details of the 26 users can be passed on to marketing team who can send them promotional emails.
- Zack_Kemmer93 (user id: 52) is the contest winner, i.e, the user with the most likes on a single photo.
- The top 5 most commonly used hashtags are smile, beach, party, fun and concert.
- The Ad campaign can be held mainly on Thursdays and Sundays.
- The average number of posts by active users is 3.4730 and the total average number of posts considering all the users on instagram is 0.74
- The users with the following user id are categorized as bots/fake accounts:
  User_id: 5, 14, 21, 24, 36, 41, 54, 57, 66, 71, 75, 76, 91

Result:

This project imitates the real world scenario and has given me an opportunity to test my knowledge on SQL. I have found various insights and these insights should actually help the team a lot if it is a real world case.