PROJECT 2

Instagram User Analytics

Name: Arsh Chawhan **Email:** Chawhanarsh@gmai.com

Project Description

In this project, we are going to analyze Instagram User data and answer the following questions:

A) Marketing:

1. **Rewarding Most Loyal Users:** Find 5 oldest users of the Instagram from the database provided.
2. **Remind Inactive Users to Start Posting:** Find the users who have never posted a single photo on Instagram.
3. **Declaring Contest Winner:** Identify the winner of the contest and provide their details to the team.
4. **Hashtag Researching:** Identify and suggest the top 5 most commonly used hashtags on the platform
5. **Launch AD Campaign:** What day of the week do most users register on? Provide insights on when to schedule an ad campaign

B) Investor Metrics:

1. **User Engagement:** Provide how many times does average user posts on Instagram. Also, provide the total number of photos on Instagram/total number of users
2. **Bots & Fake Accounts:** 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).

Instagram User Analytics

Approach

First, I run the given commands to create the database in MySQL Workbench, then I analyzed the database. Then I observed all the tables, columns carefully and then I started solving the questions.

Tech – Stack Used

In this project, I have used **MySQL Workbench 8.0 CE**. I used MySQL Workbench as it is freely available, run smoothly on my system and it is faster compares to other freely available SQL software.

Detailed Project Report / Insights

Marketing

The marketing team wants to launch some campaigns, and they need help with the following -

1. **Rewarding Most Loyal Users:** People who have been using the platform for the longest time.

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

QUERY:

```
SELECT
    username, created_at
FROM
    users
ORDER BY created_at ASC
LIMIT 5;
```

Instagram User Analytics

- First, I have selected ‘username’, ‘created_at’ from the table named ‘users’.
- Then I used ORDER BY clause on ‘created_at’. ORDER BY clause is used to sort records in ascending or descending order.
- As I wanted only 5 oldest users, I used LIMIT. LIMIT is used to restrict the number of rows and shows the result in a limited manner.

OUTPUT:

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.Jacobson2	2016-05-14 07:56:26

2. **Remind Inactive Users to Start Posting:** By sending them promotional emails to post their 1st photo.

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

QUERY:

```
SELECT
    users.id, username
FROM
    users
        LEFT JOIN photos
        ON users.id = photos.user_id
WHERE
        photos.id IS NULL;
```

Instagram User Analytics

- In this query, I have selected ‘username’,’users.id’ from the table named ‘users’
- Then I used LEFT JOIN keyword to join ‘photos’ table. A LEFT JOIN is used when a user wants to extract data of left table only. It returns all the record from the left table (users), and the matching records from the right table (photos).
- Then I used WHERE clause to filter the record from the table ‘photos’.
- I used IS NULL to exclude rows where the data is missing or null.

OUTPUT:

id	username
5	Aniya_Hackett
7	Kasandra_Homenick
14	Jaclyn81
21	Rocio33
24	Maxwell.Halvorson
25	Tierra.Trantow
34	Pearl7
36	Ollie_Ledner37
41	Mckenna17
45	David.Osinski47
49	Morgan.Kassulke
53	Linnea59
54	Duane60
57	Julien_Schmidt
66	Mike.Auer39
68	Franco_Keebler64
71	Nia_Haag
74	Hulda.Macejkovic
75	Leslie67
76	Janelle.Nikolaus81
80	Darby_Herzog
81	Esther.Zulauf61
83	Bartholome.Bernhard
89	Jessyca_West
90	Esmeralda.Mraz57
91	Bethany20

Instagram User Analytics

3. **Declaring Contest Winner:** The team started a contest and the user who gets the most likes on a single photo will win the contest now they wish to declare the winner.

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

QUERY:

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

- Here, I selected ‘users.id’ AS user_id, ‘users.username’, ‘likes.photo_id’, ‘count(likes.user_id) AS ‘Total_likes’ from ‘likes’ table.
- AS (ALIAS) is used to give name which will be shown in the result.
- COUNT function is used to return the total number of items in a group.
- Then I used INNER JOIN to join two tables, i.e. likes and photos.
- I used INNER JOIN again to join two tables, i.e. photos and users.

Instagram User Analytics

- After joining, I used GROUP BY clause. When we use an aggregation, GROUP BY clause is used. Here, I have GROUP BY 'likes.photo_id'.
- Then I used ORDER BY on Total_likes in descending order.
- I used LIMIT at 1. So that it will show only 1 record, i.e. the user who have the most numbers of likes on a single photo.

OUTPUT:

user_id	username	photo_id	Total_likes
52	Zack_Kemmer93	145	48

Here, the winner for the contest is "Zack_Kemmer93" who have a photo with 48 likes on it.

4. **Hashtag Researching:** A partner brand wants to know, which hashtags to use in the post to reach the most people on the platform.

Task: Identify and suggest the top 5 most commonly used hashtags on the platform

QUERY:

```
SELECT
   tags.tag_name AS tag_name, COUNT(*) AS tags_used
FROM
   tags
      INNER JOIN photo_tags
      ON tags.id = photo_tags.tag_id
GROUP BY tags.tag_name
ORDER BY tags_used DESC
LIMIT 5;
```

Instagram User Analytics

- In this query, I have selected ‘tags.tag_name’ AS ‘tag_name’, count(*) AS ‘tags_used’ from ‘tags’ table. Asterisk symbol (*) is used when you want to select everything from a particular table.
- Then I INNER JOIN ‘photos_tags’ table on ‘tags.id’ and ‘photo_tags.tag_id’.
- After joining, I have used GROUP BY on ‘tags.tags_name’ and ORDER BY on tags_used in descending order.
- LIMIT is 5, i.e. It will show top 5 most used hashtags.

OUTPUT:

tag_name	tags_used
smile	59
beach	42
party	39
fun	38
concert	24

5. **Launch AD Campaign:** The team wants to know, which day would be the best day to launch ADs.

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

Instagram User Analytics

QUERY:

```
SELECT
    DAYNAME(created_at) AS `day`, COUNT(*) AS users_registered
FROM
    users
GROUP BY `day`
ORDER BY users_registered DESC;
```

- Here, I have selected, ‘created_at’ AS ‘day’, count(*) AS ‘users_registered’ from table ‘users’.
- I used Dayname() function to show the name of the day.
- Then I GROUP BY ‘day’
- And used ORDER BY on ‘user_registered’ in a descending order.

OUTPUT:

day	users_registered
Thursday	16
Sunday	16
Friday	15
Tuesday	14
Monday	14
Wednesday	13
Saturday	12

Here, we can see that the most numbers of users registered on Instagram on THURSDAY and SUNDAY. So, Thursday and Sunday will be the best days to launch Ads.

Instagram User Analytics

Investor Metrics

Our investors want to know if Instagram is performing well and is not becoming redundant like Facebook, they want to assess the app on the following grounds

1. **User Engagement:** Are users still as active and post on Instagram or they are making fewer posts

Task: Provide how many times does average user posts on Instagram.

Also, provide the total number of photos on Instagram/total number of users

QUERY:

```
WITH main AS
( SELECT users.id AS user_id, count(photos.id) AS photo_id
 FROM users
      LEFT JOIN photos
      ON photos.user_id = users.id
GROUP BY users.id )

SELECT sum(photo_id) AS `Total no of photos`,
count(user_id) AS `Total no of users`,
sum(photo_id) / count(user_id) AS
`Average no of photos per user`
FROM main;
```

- In this query, I have created a temporary table named ‘main’ using the WITH clause.
- In the temporary table, I have selected ‘users.id’ and count(photos.id) as ‘photo_id’ from ‘users’ table.
- Then I LEFT JOIN ‘photos’ table on ‘photos.user_id’ and ‘users.id’

Instagram User Analytics

- Then I used GROUP BY on ‘users.id’
- Then outside the temporary table, I have written another query.
- In the outer query, I have selected ‘photo_id’ inside a SUM function as ‘Total no of photos’, ‘user_id’ inside a COUNT function as ‘Total no of users’.
- And to get the ‘average no of photos per user’, I have divided sum(photo_id) by count(user_id) from the temporary table named ‘main’.

OUTPUT:

Total no of photos	Total no of users	Average no of photos per user
257	100	2.5700

2. **Bots & Fake Accounts:** The investors want to know if the platform is crowded with fake and dummy accounts.

Task: 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).

QUERY:

```
SELECT username , count(*) AS likes_count
FROM users
    INNER JOIN likes
    ON users.id = likes.user_id
GROUP BY likes.user_id
HAVING likes_count = (
    SELECT count(*)
    FROM photos );
```

Instagram User Analytics

- In this query, I have selected ‘username’ and count(*) as ‘likes_count’ from ‘users’ table.
- Then I INNER JOIN ‘likes’ table to ‘users’ table.
- I used GROUP BY clause on likes.user_id and HAVING clause on likes_count
- Inside HAVING clause, I have selected count(*) from ‘photos’ table.

OUTPUT:

username	likes_count
Aniya_Hackett	257
Jaclyn81	257
Rocio33	257
Maxwell.Halvorson	257
Ollie_Ledner37	257
Mckenna17	257
Duane60	257
Julien_Schmidt	257
Mike.Auer39	257
Nia_Haag	257
Leslie67	257
Janelle.Nikolaus81	257
Bethany20	257

Here, 13 users are bot users, as they have liked every single picture on the site, which is not possible for a normal user.

Instagram User Analytics

Result

This project “Instagram User Analytics” helped me to understand the concept of MySQL in a deep and useful way. This project challenged me to pushed my limits to solve the problems. Because of this project I learned a lot of new terms and functions in MySQL. I have analyzed the tables well and provided the solutions to every question with their output.