

Instagram User Analytics

Description:

Imagine you're a data analyst working with the product team at Instagram. Your role involves analyzing user interactions and engagement with the Instagram app to provide valuable insights that can help the business grow. User analysis involves tracking how users engage with a digital product, such as a software application or a mobile app. 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, you'll be using SQL and MySQL Workbench as your tool to analyze Instagram user data and answer questions posed by the management team. Your insights will help the product manager and the rest of the team make informed decisions about the future direction of the Instagram app.

Remember, the goal of this project is to use your SQL skills to extract meaningful insights from the data. Your findings could potentially influence the future development of one of the world's most popular social media platforms.

SQL Tasks:

A) Marketing Analysis:

- 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.
- 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.
- 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.
- 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.

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.

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.
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:

To do this project, I have used the knowledge of the SQL that I have learned during the course with additional reading as well. I have extracted the data from the given database using the various functions I have learnt such as where condition, Group by and other related functions. I have found out the answers to the questions asked by both the marketing departments and metrics department with the SQL queries and executed them in the MYSQL workbench. In order for the report to look visually pleasing I have used Canva to display the answers to the questions so that the project looks little more attractive.

Tech-Stack Used:

The software I have used to do this project to execute the queries is MYSQL workbench. I have used this particular software as it works best with the relational database in creating database and also most popular database system and smooth to execute the queries.

Marketing department answers:

1. Identify the five oldest users on Instagram from the provided database.

Query used: The query used to solve this question and the answer to the question i.e the output both are shown in the below image

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
1 use ig_clone;
2 #1st solution
3 select username,created_at
4 from users
5 order by created_at
6 limit 5;
```

The Results window displays the output of the query, showing the five oldest users:

username	created_at
Darby_Herzog	2016-05-06 00:14:21
Emilio_Berrier52	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

The bottom panel shows the Action Output window, which lists the execution steps and their results:

#	Time	Action	Message	Duration / Fetch
12	10:05:40	select tag_name,COUNT(tag_name) AS total FROM tags JOIN photo_tags ON tags.id = photo_tags.tag_id GR...	21 row(s) returned	0.000 sec / 0.000 sec
13	10:05:53	select tag_name,COUNT(tag_name) AS total FROM tags JOIN photo_tags ON tags.id = photo_tags.tag_id GR...	5 row(s) returned	0.000 sec / 0.000 sec
14	10:08:51	select username,created_at from users order by created_at asc limit 5	5 row(s) returned	0.000 sec / 0.000 sec
15	10:09:03	select username,created_at from users order by created_at limit 5	5 row(s) returned	0.000 sec / 0.000 sec
16	10:10:10	use ig_clone	0 row(s) affected	0.000 sec
17	10:10:14	select username,created_at from users order by created_at limit 5	5 row(s) returned	0.000 sec / 0.000 sec

The Most Loyal Top 5 Users of INSTAGRAM



1. **Darby_Herzog-(2016-05-06)**
2. **Emilio_Bernier52-(2016-06-06)**
3. **Elenor88-(2016-05-08)**
4. **Nicole71-(2016-05-09)**
5. **Jordyn_Jacobson2-(2016-05-14)**

2. Identify users who have never posted a single photo on Instagram.

Query used: The query used to solve this question and the answer to the question i.e the output both are shown in the below image.

The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL query:

```
#2nd solution
select id,username
from users
where id not in(
    select distinct user_id
    from photos
    where image_url is not null
);
```

The result grid displays the following data:

id	username
53	Linnea99
54	Duane60
57	Julien_Schmidt
66	Mike_Auer39
68	Franco_Keebler64
71	Nia_Haag
74	Hulda_Macejovic
75	Leslie67
76	Janelle_Nikolaus81
80	Darby_Herzog
81	Esther_Zulauf61
83	Bartholome_Bernhard
89	Jessyca_West
90	Esmeralda_Mraz57
91	Bethany20

The bottom status bar shows the query execution details: 79 16:31:13 select id,username from users where id not in(select distinct id from photos where image_url is not null ... 0 row(s) returned 0.000 sec / 0.000 sec.

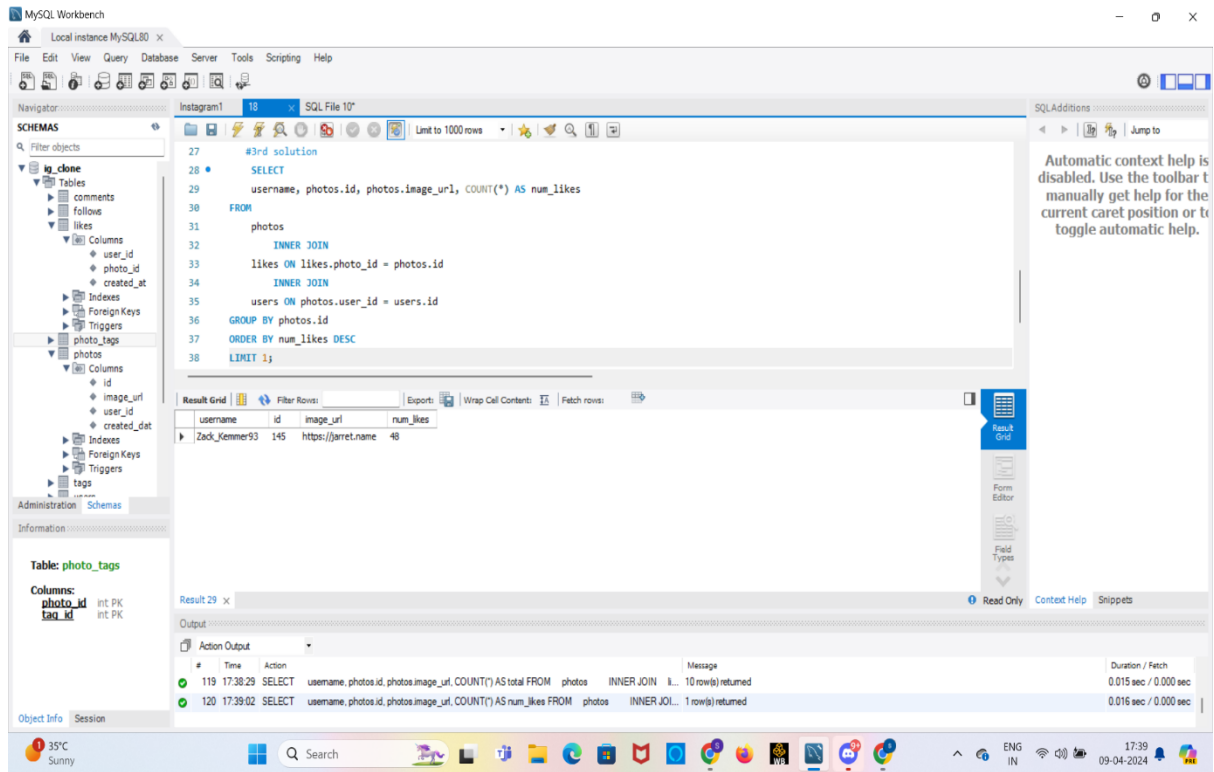
Encouraging Inactive Users to Start Posting



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

3. Determine the winner of the contest and provide their details to the team.

Query used: The query used to solve this question and the answer to the question i.e the output both are shown in the below image.



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
#3rd solution
SELECT
  username, photos.id, photos.image_url, COUNT(*) AS num_likes
FROM
  photos
  INNER JOIN
  likes ON likes.photo_id = photos.id
  INNER JOIN
  users ON photos.user_id = users.id
GROUP BY photos.id
ORDER BY num_likes DESC
LIMIT 1;
```

The Result Grid shows the following output:

username	id	image_url	num_likes
Zack_Kemmer93	145	https://jaret.name	48

The bottom status bar shows the query execution details:

#	Time	Action	Message	Duration / Fetch
119	17:38:29	SELECT	username, photos.id, photos.image_url, COUNT(*) AS total FROM photos INNER JOIN likes ON likes.photo_id = photos.id INNER JOIN users ON photos.user_id = users.id GROUP BY photos.id ORDER BY num_likes DESC LIMIT 1;	0.015 sec / 0.000 sec
120	17:39:02	SELECT	username, photos.id, photos.image_url, COUNT(*) AS num_likes FROM photos INNER JOIN likes ON likes.photo_id = photos.id INNER JOIN users ON photos.user_id = users.id GROUP BY photos.id ORDER BY num_likes DESC LIMIT 1;	0.016 sec / 0.000 sec

Declaring the Contest Winner



4. Identify and suggest the top five most commonly used hashtags on the platform.

Query used: The query used to solve this question and the answer to the question i.e the output both are shown in the below image.

The screenshot displays the MySQL Workbench interface. On the left, the 'SCHEMAS' pane shows a database named 'ig_clone' with various tables and columns. The main editor window contains the following SQL query:

```
SELECT
  tag_name, COUNT(tag_name) AS total
FROM
  tags
JOIN
  photo_tags ON tags.id = photo_tags.tag_id
GROUP BY tags.id
ORDER BY total DESC
limit 5;
```

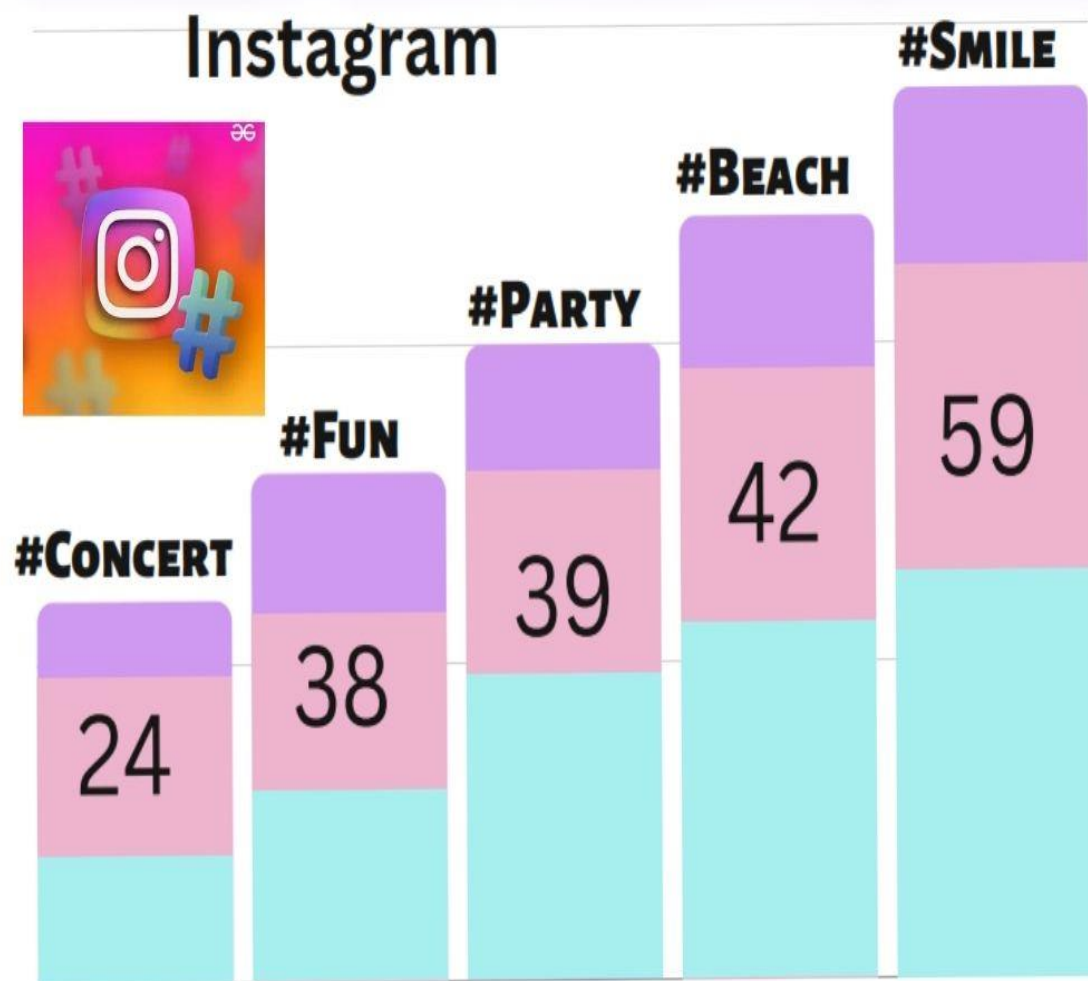
Below the query editor, the 'Result Grid' shows the output of the query:

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

At the bottom, the 'Action Output' pane shows the execution details of the query, including the time taken and the number of rows returned.

#	Time	Action	Message	Duration / Fetch
133	17:49:51	SELECT	tag_name, COUNT(tag_name) AS total FROM tags JOIN photo_tags ON tags.id = phot...	7 row(s) returned 0.000 sec / 0.000 sec
134	17:51:37	SELECT	tag_name, COUNT(tag_name) AS total FROM tags JOIN photo_tags ON tags.id = phot...	5 row(s) returned 0.000 sec / 0.000 sec

TOP 5 HASHTAGS USED ON INSTAGRAM



5. Determine the day of the week when most users register on Instagram.
Provide insights on when to schedule an ad campaign.

Query used: The query used to solve this question and the answer to the question i.e the output both are shown in the below image.

The screenshot shows the MySQL Workbench interface. The central pane displays a SQL query labeled "#5th solution". The query is as follows:

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

The bottom pane shows the "Result Grid" with the following data:

days_of_week	total_registrations
Thursday	16
Sunday	16
Friday	15
Tuesday	14
Monday	14
Wednesday	13
Saturday	12

The left sidebar shows the "SCHEMAS" pane with a tree view of databases including "ig_clone", "sakila", "sys", "temp_db", and "world". The "ig_clone" database is expanded, showing tables like "comments", "follows", "likes", "photo_tags", "photos", "tags", and "users". The "Table: photo_tags" is selected, and its columns "photo_id" and "tag_id" are listed as integer primary keys.

The bottom status bar shows the system temperature as 29°C, sunny weather, and the date and time as 10:27 on 10-04-2024.

Ad Campaign Launch



Best time to schedule an advertisement campaign

Thursday
16

Sunday
16



Investor Metric department answers:

1. 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.

Query used: The query used to solve this question and the answer to the question i.e the output both are shown in the below image.

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
#1st solution for investor metrics
SELECT
  (SELECT
    COUNT(id)
  FROM
    photos) / (SELECT
    COUNT(DISTINCT user_id)
  FROM
    photos) AS Average_posts_per_User,
  (SELECT
    COUNT(id)
  FROM
    photos) / (SELECT
    COUNT(id)
  FROM
    users) AS Ratio_of_Total_Posts_to_Total_Users;
```

The Results window displays the output of the query:

Average_posts_per_User	Ratio_of_Total_Posts_to_Total_Users
3.4730	2.5700

The bottom status bar shows the execution details: 34 rows returned, 11.2758 seconds duration, and 0.000 seconds fetch time.

2. Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.

Query used: The query used to solve this question and the answer to the question i.e the output both are shown in the below image.

The screenshot displays the MySQL Workbench interface. The central SQL editor contains the following query:

```
23 photos) / (SELECT
24 COUNT(id)
25 FROM
26 users) AS Ratio_of_Total_Posts_to_Total_Users;
27
28 #2nd solution for investor metrics
29 SELECT id,
30 username
31 FROM users
32 WHERE id IN (SELECT user_id
33 FROM likes
34 GROUP BY user_id
35 HAVING Count(user_id) = (SELECT Count(id)
36 FROM photos));
```

The 'Result Grid' at the bottom shows the output of the query, listing user IDs and usernames:

id	username
5	Anya_Hackett
14	Jadyn81
21	Roco33
24	Maxwell_Halvorsen
36	Olie_Ledner37
41	Mikenna17
54	Duane60
57	Julien_Schmidt
66	Mike_Auer39
71	Nia_Haag
75	Leslie67
76	Janele_Nikolaus81
91	Bethany20
9999	

The left sidebar shows the 'SCHEMAS' panel with a tree view of the database structure, including tables like 'comments', 'follows', 'likes', 'photo_tags', 'photos', 'tags', and 'users'. The 'photo_tags' table is selected, showing its columns: 'photo_id' (int PK) and 'tag_id' (int PK).

Insights: By doing this project I have gained a lot of understanding on how the data sets can be analysed using SQL queries and also how to get the required information out from the given dataset. It also helped me in trying different approaches I can apply to a particular question to be solved with various query commands. Furthermore, it helped me understand how does the analytic process is applied to the datasets and how they are solved in real time scenarios.

Results:

The outcomes required for both the marketing team and the investor metric team has been solved using the SQL queries in MYSQL workbench and the outputs have been provided thoroughly so that the respective teams can take best possible decision based on the results and insights provided.