Instagram User Analytics

Pranav Vaswani

Project Description

This project on 'Instagram User Analytics' is to be performed on the 'ig_clone' database provided. I am supposed to answer the questions about Instagram by running queries on the given database. I then need to provide the answers to those questions and for the purpose of this particular project, show my queries as well.

Approach

My approach towards this project was pretty straightforward. I first imported the database 'ig_clone' through MySQL workbench. I then ran the queries for all the questions one by one, first in MySQL workbench and once I had the appropriate result, I pasted my commands and the output to this power point presentation which I will eventually convert into a pdf.

Tech-Stack Used

For the purpose of this project, I have used:

MySQL Workbench version 8.0.33 (MySQL Community Server – GPL)

A) Marketing

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

   My task: Find the 5 oldest users of the Instagram from the database provided

```
83    #  Find the 5 oldest users of the Instagram from the database provided
84 •  SELECT
85        *
86    FROM
87        users
88    ORDER BY created_at
89    LIMIT 0 , 5;
```

QUERY

```
SELECT
   *
FROM
   users
ORDER BY created_at
LIMIT 0 , 5;
```

OUTPUT

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

Insights

After running the query and obtaining the result, we infer the names of the first 5 users who joined instagram or in other words, the 5 oldest users of instagram.

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

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

```
91    #  Find the users who have never posted a single photo on Instagram
92    SELECT
93        *
94    FROM
95        users
96    WHERE
97        id NOT IN (SELECT
98                user_id
99            FROM
100               photos);
```

QUERY

```
SELECT
   *
FROM
   users
WHERE
   id NOT IN (SELECT
         user_id
      FROM
         photos);
```

OUTPUT

id	username	created_at
5	Aniya_Hackett	2016-12-07 01:04:39
7	Kasandra_Homenick	2016-12-12 06:50:08
14	Jadyn81	2017-02-06 23:29:16
21	Rocio33	2017-01-23 11:51:15
24	Maxwell.Halvorson	2017-04-18 02:32:44
25	Tierra.Trantow	2016-10-03 12:49:21
34	Pearl7	2016-07-08 21:42:01
36	Ollie_Ledner37	2016-08-04 15:42:20
41	Mckenna17	2016-07-17 17:25:45
45	David.Osinski47	2017-02-05 21:23:37
49	Morgan.Kassulke	2016-10-30 12:42:31
53	Linnea59	2017-02-07 07:49:34
54	Duane60	2016-12-21 04:43:38
57	Julien_Schmidt	2017-02-02 23:12:48
66	Mike.Auer39	2016-07-01 17:36:15
68	Franco_Keebler64	2016-11-13 20:09:27
71	Nia_Haag	2016-05-14 15:38:50
74	Hulda.Macejkovic	2017-01-25 17:17:28
75	Leslie67	2016-09-21 05:14:01
76	Janelle.Nikolaus81	2016-07-21 09:26:09
80	Darby_Herzog	2016-05-06 00:14:21
81	Esther.Zulauf61	2017-01-14 17:02:34
83	Bartholome.Bernhard	2016-11-06 02:31:23
89	Jessyca_West	2016-09-14 23:47:05

90	Esmeralda.Mraz57	2017-03-03 11:52:27
91	Bethany20	2016-06-03 23:31:53
NULL	NULL	NULL

Insights

After running the query and obtaining the result, we get the data of the users who have never posted a single photo on Instagram.

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.

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

QUERY

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

QUERY

```
105 •   SELECT
106         photos.user_id,
107         users.username,
108         photo_id,
109         COUNT(*) AS max_likes
110     FROM
111         likes
112             JOIN
113         users
114             JOIN
115         photos ON users.id = photos.user_id
116             AND photos.id = likes.photo_id
117     GROUP BY likes.photo_id
118     ORDER BY max_likes DESC
119     LIMIT 1;
```

OUTPUT

	user_id	username	photo_id	max_likes
▶	52	Zack_Kemmer93	145	48

Insights

After running the query and obtaining the result, we get the details of the winner of the contest.

4. Hashtag Researching: A partner brand wants to know, which hashtags to use in the post to reach the most people on the platform.
My Task: Identify and suggest the top 5 most commonly used hashtags on the platform.

```
121 •   SELECT
122         id, tag_name, COUNT(tag_id) AS no_of_times_tag_used
123     FROM
124         photo_tags
125             JOIN
126         tags ON tags.id = photo_tags.tag_id
127     GROUP BY tags.id
128     ORDER BY no_of_times_tag_used DESC
129     LIMIT 5;
```

QUERY

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

OUTPUT

id	tag_name	no_of_times_tag_used
21	smile	59
20	beach	42
17	party	39
13	fun	38
18	concert	24

Insights

After running the query and obtaining the result, we get the list of top 5 most commonly used hashtags.

5. Launch AD Campaign: The team wants to know, which day would be the best day to launch ADs.
My Task: What day of the week do most users register on? Provide insights on when to schedule an ad campaign.

```sql
SELECT
    COUNT(id) AS no_of_users_registered,
    DAYNAME(created_at) AS best_day_for_AD_campaign
FROM
    users
GROUP BY best_day_for_AD_campaign
ORDER BY COUNT(id) DESC
LIMIT 1;
```

QUERY

ELECT

COUNT(id) AS no_of_users_registered,

DAYNAME(created_at) AS best_day_for_AD_campaign

FROM

users

GROUP BY best_day_for_AD_campaign

ORDER BY COUNT(id) DESC

LIMIT 1;

OUTPUT

	no_of_users_registered	best_day_for_AD_campaign
▶	16	Thursday

Insights

After running the query and obtainin g the result, we get the best day (when the most users register) to launch an AD campaign.

B) Investor Metrics

1. **User Engagement: Are users still as active and post on Instagram or they are making fewer posts**
   My Task: Provide how many times does average user posts on Instagram. Also, provide the total number of photos on Instagram/total number of users

```
SELECT
    COUNT(DISTINCT (users.id)) AS total_no_of_users,
    COUNT(photos.id) AS total_no_of_photos,
    COUNT(photos.id) / COUNT(DISTINCT (users.id)) AS avg_posts_per_user
FROM
    users
        LEFT JOIN
    photos ON photos.user_id = users.id;
```

QUERY

```
SELECT
    COUNT(DISTINCT (users.id)) AS total_no_of_users,
    COUNT(photos.id) AS total_no_of_photos,
    COUNT(photos.id) / COUNT(DISTINCT (users.id)) AS avg_posts_per_user
FROM
    users
        LEFT JOIN
    photos ON photos.user_id = users.id;
```

OUTPUT

	total_no_of_users	total_no_of_photos	avg_posts_per_user
▶	100	257	2.5700

Insights

After running the query and obtaining the result, we infer that the average user has posted 2-3 times on Instagram.

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

```
158 •   SELECT
159         user_id AS bot_id
160     FROM
161         likes
162     GROUP BY user_id
163     HAVING COUNT(user_id) = (SELECT
164             COUNT(id)
165         FROM
166             photos);
```

QUERY

```
SELECT
    user_id AS bot_id
FROM
    likes
GROUP BY user_id
HAVING COUNT(user_id) = (SELECT
      COUNT(id)
    FROM
      photos);
```

OUTPUT

bot_id
5
14
21
24
36
41
54
57
66
71
75
76
91

Insights

After running the query and obtaining the result, we get the ids of all the bots (not real users)

Result

Through this project, I have successfully brushed up on my MySQL skills and am very confident in writing queries and deriving the outputs, as well as importing databases and finding solutions to problems on my own.