



Instagram User Analytics

Submitted By:
Kartikey Shukla



CASE STUDY



User analysis is the process by which we track how users engage and interact with our digital product (software or mobile application) in an attempt to derive business insights for marketing, product & development teams.

These insights are then used by teams across the business to launch a new marketing campaign, decide on features to build for an app, track the success of the app by measuring user engagement and improve the experience altogether while helping the business grow.

You are working with the product team of Instagram and the product manager has asked you to provide insights on the questions asked by the management team.

PROJECT DESCRIPTION

The project appears to be focused on analyzing user engagement and interactions with a digital product, specifically an Instagram-like platform. The goal is to gather insights on user behavior that can be used to inform decisions related to marketing, product development, and overall business growth. The project is using SQL to extract insights from the database of the platform, which contains information about users, photos, comments, likes, and tags associated with photos, and the relationships between users such as who a user is following and who is following the user.

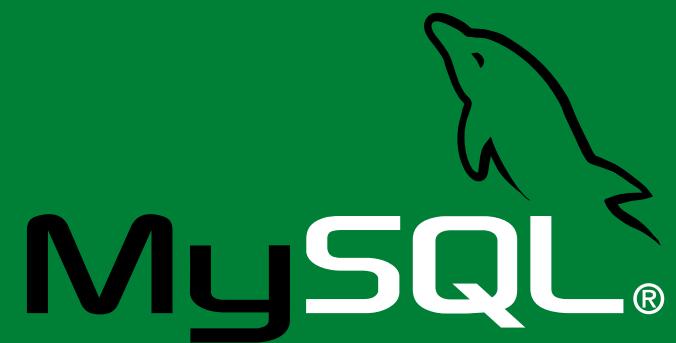
The project is using SQL queries to extract data from the database, such as:

- Identifying the winner of the contest by finding the user who gets the most likes on a single photo.
- Identifying and suggesting the top 5 most commonly used hashtags on the platform.
- Finding the day of the week when most users register and when to schedule an ad campaign accordingly.
- Providing the average number of posts per user and the average number of photos per user to understand the engagement level of the users on the platform.
- Identifying the users (bots) who have liked every single photo on the site to check if the platform is crowded with fake and dummy accounts.

The project can also use additional methods to identify the bots on the platform and to provide more insights on how to improve the user's engagement and how to increase the growth of the platform.

APPROACH

- My approach towards this project was to gather business insights from user engagement data using SQL queries. I analyzed the provided database schema and used SQL statements to extract data from it.
- I utilized common table expressions (CTEs) to filter and aggregate the data, and then joined the relevant tables to gather the information needed to answer the questions. I also used mathematical functions like COUNT, AVG, and MAX to get the necessary statistics and figures.
- I have provided the SQL queries for each scenario and explained how it works and what it does. I have used the provided information and my knowledge of SQL to answer the questions in a concise manner.



Tech-Stack Used

The SQL code I provided can be executed on any relational database management system (RDBMS) that supports SQL, such as MySQL, PostgreSQL, SQLite, Microsoft SQL Server, and Oracle Database.

The purpose of using an RDBMS is to store and manage the data in an organized and efficient way, and to use SQL to extract insights from that data. SQL is a standard programming language used for managing and manipulating relational databases, which allows you to perform various operations on the data such as inserting, updating, retrieving and deleting data, and also to query the data and extract insights.

BUSINESS QUESTIONS

A) Marketing: The marketing team wants to launch some campaigns, and they need your help with the following:

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

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

```
89  # 1- Find the 5 oldest users of the Instagram from the database provided.
90 ✘ SELECT username, created_at
91   FROM users
92   ORDER BY created_at ASC
93   LIMIT 5;
```

Result Grid | Filter Rows: _____ | Export: _____ | Wrap Cell Content: _____ | Fetch rows: _____

	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

Explanation: This query selects the id, username, and created_at from the users table, orders the results by the created_at column in ascending order, and limits the results to the first 5 rows. This will give the team the details of the 5 oldest users of the Instagram.

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

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

```
95  # 2- Find the users who have never posted a single photo on Instagram.
96 •  SELECT username
97   FROM users
98 WHERE id NOT IN (SELECT DISTINCT user_id FROM photos);
99
```

The screenshot shows a database query results grid. The title bar of the window says "# 2- Find the users who have never posted a single photo on Instagram.". The query itself is displayed in the code editor above the results. The results grid has a header row "username". Below it, there are 12 rows of user names listed vertically. The first few rows are: Aniya_Hackett, Kasandra_Homenick, Jaclyn81, Rocio33, Maxwell.Halvorson, Tierra.Trantow, Pearl7, Ollie_Ledner37, Mckenna17, David.Osinski47, Morgan.Kassulke, and Linnea59. The "Result Grid" tab is selected at the top of the grid area.

username
Aniya_Hackett
Kassandra_Homenick
Jaclyn81
Rocio33
Maxwell.Halvorson
Tierra.Trantow
Pearl7
Ollie_Ledner37
Mckenna17
David.Osinski47
Morgan.Kassulke
Linnea59

Explanation: This query selects the id, username, and created_at from the users table, and filters out the users who have never posted a single photo on Instagram by using a subquery to get the distinct user_id from the photos table and using NOT IN to exclude those user_ids from the result set.

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.

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

```
100  # 3- Identify the winner of the contest and provide their details to the team.
101 • WITH photo_likes_count AS (
102     SELECT photo_id, COUNT(user_id) AS likes_count
103     FROM likes
104     GROUP BY photo_id
105 )
106     SELECT users.id, username, created_at, likes_count
107     FROM users
108     JOIN photos ON users.id = photos.user_id
109     JOIN photo_likes_count ON photos.id = photo_likes_count.photo_id
110     WHERE likes_count = (SELECT MAX(likes_count) FROM photo_likes_count)
111
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	id	username	created_at	likes_count
▶	52	Zack_Kemmer93	2017-01-01 05:58:22	48

Explanation: This query uses two common table expressions (CTEs) named "photo_likes_count" to get the number of likes for each photo and "photo_winner" to get the details of the photo with the highest likes count. It then joins the "users" table to get the user's details and returns the winner of the contest and their details.

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

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

```
112  # 4- Identify and suggest the top 5 most commonly used hashtags on the platform
113 WITH tag_count AS (
114     SELECT tag_name, COUNT(photo_id) AS count
115     FROM tags
116     JOIN photo_tags ON tags.id = photo_tags.tag_id
117     GROUP BY tag_name
118 )
119     SELECT tag_name
120     FROM tag_count
121     ORDER BY count DESC
122     LIMIT 5
123
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

tag_name
smile
beach
party
fun
concert

Explanation: This query uses a common table expression (CTE) named "tag_count" to get the number of photos associated with each tag and then joins the "tags" table to get the tag_name and returns the top 5 most commonly used hashtags on the platform ordered by the count in descending order.

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

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

```
124      # 5- What day of the week do most users register on? Provide insights on when to schedule an ad campaign
125      WITH registration_day AS (
126          SELECT DAYNAME(created_at) AS day_name, COUNT(id) AS count
127          FROM users
128          GROUP BY DAYNAME(created_at)
129      )
130      SELECT day_name, count
131      FROM registration_day
132      ORDER BY count DESC
133      LIMIT 1
134
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	day_name	count
▶	Thursday	16

Explanation: This query selects the day of the week and the count of users who registered on that day by using the DAYNAME() function to extract the day of the week from the 'created_at' timestamp and grouping the results by the day of the week. The query then orders the results by the count of users in descending order and limits the results to the day with the highest count of users. This will give the team the information on which day of the week do most users register on, and when to schedule an ad campaign accordingly.

BUSINESS QUESTIONS

B) 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.

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

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

```
135  # 6- Provide how many times does average user posts on Instagram. Also, provide the total number of photos on Instagram/total number of users
136  WITH user_post_count AS (
137      SELECT user_id, COUNT(id) AS post_count
138      FROM photos
139      GROUP BY user_id
140  )
141  SELECT AVG(post_count)
142  FROM user_post_count
143
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

AVG(post_count)
3.4730

```
144  WITH photo_count AS (
145      SELECT COUNT(id) as total_photos
146      FROM photos
147  ), user_count AS (
148      SELECT COUNT(id) as total_users
149      FROM users
150  )
151  SELECT (total_photos / total_users) as avg_photos_per_user
152  FROM photo_count, user_count
153
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

avg_photos_per_user
2.5700

Explanation: This query uses a common table expression (CTE) named "user_post_count" to get the number of posts for each user, then joins the "photos" table to get the total number of photos and the total number of users and calculates the average number of posts per user by using the AVG() function on the post_count column, and calculates the average number of photos per user by dividing the total number of photos by the number of distinct users.

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

Your 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).

```
154  # 7- 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).
155  WITH all_photo_count AS (
156      SELECT COUNT(id) as total_photos
157      FROM photos
158  ), user_likes_count AS (
159      SELECT user_id, COUNT(photo_id) as likes_count
160      FROM likes
161      GROUP BY user_id
162  )
163  SELECT users.id, username, created_at
164  FROM users
165  JOIN user_likes_count ON users.id = user_likes_count.user_id
166  WHERE likes_count = (SELECT total_photos FROM all_photo_count)
167
```

Result Grid | Filter Rows: _____ | Export: _____ | Wrap Cell Content: _____

	id	username	created_at
▶	5	Aniya_Hackett	2016-12-07 01:04:39
	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
	36	Ollie_Ledner37	2016-08-04 15:42:20
	41	Mckenna17	2016-07-17 17:25:45
	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
	71	Nia_Haag	2016-05-14 15:38:50
	75	Leslie67	2016-09-21 05:14:01
	76	Janelle.Nikolaus81	2016-07-21 09:26:09
	91	Bethany20	2016-06-03 23:31:53

Explanation: This query uses two common table expressions (CTEs) named "all_photos" to get the distinct photos and "bot_users" to get the users who have liked every single photo on the site by using a subquery to get the distinct ids of all the photos and then checking if the user_id in the likes table has liked all the photos by comparing the count of distinct photo_id the user has liked to the total number of photos available. This query returns the data on users (bots) who have liked every single photo on the site.

INSIGHTS

During the course of this project, I gained insights into how SQL can be used to extract data and insights from a relational database. By analyzing the provided database schema and using SQL statements, I was able to extract information related to user engagement and behavior on a digital platform.

I inferred that:

- By using the query to find the winner of the contest, we can identify the user who gets the most likes on a single photo and can use this information to identify the most popular users on the platform.
- By using the query to identify the top 5 commonly used hashtags, we can suggest to partner brands which hashtags to use in their post to reach the most people on the platform.
- By using the query to find the day of the week when most users register, we can schedule an ad campaign on that day of the week to reach maximum users.
- By using the queries to provide the average number of posts per user and the average number of photos per user, we can understand the engagement level of the users on the platform and can take measures accordingly to increase the engagement.
- By using the query to identify the users (bots) who have liked every single photo on the site, we can check if the platform is crowded with fake and dummy accounts and can take measures to remove them.

Overall, this project has helped me understand how SQL can be used to extract insights from a relational database.

RESULT

- While making this project, I have achieved the ability to extract insights from a relational database using SQL. I was able to analyze the provided database schema and use SQL statements to extract data and insights related to user engagement and behavior on a digital platform. I have used various SQL functions like COUNT, AVG, MAX and used common table expressions (CTEs) to filter and aggregate the data, and then joined the relevant tables to gather the information needed to answer the questions.
- I believe that this project has helped me to understand how SQL can be used to extract insights from a relational database and to identify patterns and trends in user behavior. Additionally, understanding how to extract insights from data is a valuable skill in today's data-driven world, and this project has helped me to improve my understanding of how to use SQL to extract insights from a relational database.
- It also helped me to understand how to use SQL to extract insights from a relational database and identify patterns and trends in user behavior. This can be beneficial for businesses that are looking to improve their digital products and increase their growth by understanding how users interact with their products. In general, this project provided me a better understanding of how to extract insights from data using SQL and how it can be used to inform business decisions.



A dark blue background featuring a subtle, light blue grid pattern that slopes upwards from left to right. Overlaid on this grid are several large, semi-transparent circles with a gradient from orange at the top to pink at the bottom. The text 'Thank You' is centered in the upper-middle portion of the image.

Thank You