

Instagram User Analytics

1. Project Description:

This project aims to analyse user engagement and interactions on the Instagram platform by utilising MySQL Workbench and SQL. The results of the study are intended to offer useful data for decision-making in a variety of departments, such as investor relations, product development, and marketing. The objective is to shape the Instagram app's future and help in its growth by gaining insight into customer behaviour.

2. Approach to the project:

Acknowledge the Layout of the Database:

- I. Understanding the schema of database to know the structure and relations between the tables.
- II. Found important tables such as users, photos, likes, tags, photo tags.

A) Marketing Analysis:

1. Loyal User Reward

Query:

```
select id as old_user_id, username, created_at
from (
    Select * from users
    order by created_at asc ) a
limit 5;
```

Output:

	old_user_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

Explanation:

- a) Inner Subquery (SELECT *FROM users ORDER BY created_at ASC) a):
All of the columns (*) in the users table are selected by the inner subquery. The 'ORDER BY created_at ASC' is used to group results formed on ascending order based on 'created_at' column.
- b) Outer Query (SELECT id AS old_user_id, username, created_at FROM(...) LIMIT 5):
Specific columns (id, username, and created_at) are selected by the outer query from the inner subquery's output. The outer query renames the id column to

old_user_id using the AS keyword and finally the 'limit 5' clause control the output to first five rows based on registration date(created_at).

Main Objective:

The query selects the columns id, username, and created_at from the users table in order to determine the five oldest Instagram users. The results are sorted ascendingly by registration date, and only the top 5 individuals are returned.

Insight:

From the output we can observe that all the users who are using Instagram for long time have registered on the platform in the same month that is MAY of 2016.

2.Inactive User Engagement:

Query:

```
Select u.id, u. username
from users u
left join photos p on u.id = p.user_id
where p.id is null;
```

Output:

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

Explanation:

a) (SELECT u.id, u.username):

The 'select' clause specifies the columns to be fetched in the output that are 'id' and 'username'.

b) (FROM users u):

The users table is the parent table for the data, specified by the FROM clause. The alias u is used for our convenience.

c) (LEFT JOIN photos p ON u.id = p.user_id):

The 'left join' clause combines rows from the users (u) table with matching rows from the photos table (p) based on the condition u.id = p.user_id.

- d) All the rows from 'users' that are fetched in the result and matching rows from the 'photos' table if they are matched. Else the columns from 'photos' filled with 'null'.
- e) WHERE p.id IS NULL:
The results are further filtered by the when clause, which only includes rows in the images table when no matching record can be found.

Main Objective:

Through a left join between the users and photos databases and a filter that only includes users for whom there is no matching record in the photos dataset, the query seeks to find users who have not uploaded any photos to Instagram.

Insight:

From the result we can observe that most(15 users) of the users who are not posted any photo on Instagram has their username ended with integer.

3.contest Winner Declaration:

Query:

```
select us.id as win_user,
p.id as photo_id,
us.username, count(l.user_id) as total_likefrom users us
join photos p on us.id = p.user_id
join likes l on p.id = l.photo_id
group by us.id, p.id,us.username
order by total_likes desc
limit 1;
```

Output:

win_user	photo_id	username	total_likes
52	145	Zack_Kemmer93	48

Explanation:

- a) (select us.id AS win_user, p.id AS photo_id, us.username, COUNT(l.user_id) AS total_likes):
Columns to be obtained in the result set are specified using the SELECT clause.
 - I. us.id AS win_user: Renames the id column from the users table as win_user.

- II. `p.id AS photo_id`: Renames the id column from the pictures table as `photo_id`.
 - III. `us.username`: Retrieves the username column from the users table.
 - IV. `count(l.user_id) AS total_likes`: Determines the total number of likes for every combination of user and photo.
- b) `(FROM users us JOIN photos p ON us.id = p.user_id JOIN likes l ON p.id=l.photo_id)`:
The FROM clause specifies the path from which table data has to be fetched.
- I. `(users us, photos p, likes l)`:
The alias `us`, `p`, `l` are assigns for users, photos, likes respectively. The join condition link the USERS, PHOTOS and LIKES tables based on user IDs and photo IDs.
- c) `(GROUP BY us.id, p.id, us.username)`:
- I. GROUP BY clause groups the output based on certain columns.
 - II. `us.id, p.id, us.username` are columns that output is grouped for each photo and user combination.
- d) `(ORDER BY total_likes DESC)`:
The ORDER BY clause arranges the grouped results in descending order based on total number of likes.
- e) `(LIMIT 1)`:
And the LIMIT clause manages the output to only one top result which selects the user with most likes.

Main Objective:

This query finds the winner of a competition in which the user who receives the most likes on a single photo wins a reward. It does this by adding up all of the likes for every user and photo combination, sorting the results by total likes in decreasing order, and restricting the output to the top result.

Insight:

From the result we can see that the user with username `zack_kemmer93` and user id 52 with post having id 145 has most number of likes (48). He is the winner of the contest.

4.Hashtag Research:

Query:

```
select tg.tag_name, count(ptg.photo_id) as total_tags
from tags tg
join photo_tags ptg on ptg.tag_id = tg.id
group by tg.tag_name
order by total_tags desc
limit 5;
```

Output:

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

Explanation:

- a) (SELECT tg.tag_name, COUNT(ptg.photo_id) AS total_tags):
The SELECT clause specifies columns that has to be fetched in the output.
 - I. (tg.tag_name):
expression retrieves the tag name from 'tags' table.
 - II. (COUNT(ptg.photo_id) AS total_tags):
This expression counts the number of times a tag occurred connected to photo IDs in the 'photo_tags' table.
- b) (FROM tags tg JOIN photo_tags ptg ON ptg.tag_id = tg.id):
The FROM clause indicates the source tables for the data.
 - I. tags tg:
Uses the alias tg to specify the tags table.
 - II. photo_tags ptg: Uses the alias ptg to specify the photo_tags table.
The common condition ptg is used by JOIN conditions to join the tags and photo_tags tables.tag_id = tg.id.
- c) (GROUP BY tg.tag_name) and (ORDER BY total_tags DESC):
 - I. The results are grouped by the tag_name column in the GROUP BY clause.This organises the data such that each tag's occurrences may be counted.
 - II. Based on the number of times each tag appears (total_tags), the aggregated results are arranged in descending order using the ORDER BY clause.
- d) (LIMIT 5):
The limit clause controls the output by selecting only the top results.

Main Objective:

The purpose of the query is to determine and recommend the top five most popular hashtags on the platform. To do this, it counts the instances of each tag in the photo_tags table, sorts the results by descending order based on count, and only returns the top five tags.

Insight:

From the ouput we can observe that 'Smile' is the most popular hashtag on the Instagram with total of 59 tags, beach is the second most popular hashtag with total of 42 tags.

Similarly, Party ,Fun ,Concert have 39,38,24 tags respectively. Unfortunately, concert is the least count tag among the top 5 tags.

5.Ad Campaign Launch

Query:

```
select* from (select dayname(created_at) as popular from users) a
group by popular
order by count(popular) desc
limit 1;
```

Output:

	popular
▶	Thursday

Explanation:

- a) ((SELECT DAYNAME(created_at) AS popular FROM users) a):
 - I. This expression is the inner query which selects day names from the 'created_at' table from 'users' table.
 - II. The DAYNAME() function is used to get the names of the day from the table.
- b) (SELECT * FROM(.....)):
 - I. The outer query selects all the columns from the results of the
 - II. inner query.
- c) (GROUP BY popular):
 - I. The GROUP BY clause groups the data by number of times each day occurred.
- d) (ORDER BY COUNT(popular) DESC) and (LIMIT 1):
 - I. The aggregated results are arranged in descending order by the ORDER BY clause according to the number of occurrences of each day name which helps to find the most popular day of the week.
 - II. The most popular day is chosen by the LIMIT clause, by limiting the output to only the top result.

Main Objective:

The purpose of the query is to find the day of the week that the majority of Instagram users register. To do this, it pulls the day names from the created_at column, groups the results by day, counts the number of times each day appears, sorts the results by count in descending order, and chooses the best result.

Insight:

Thursday is the day of the week where most users registered on Instagram. Thursday is the best day of the week to conduct ad campaign.

B) Investor Metrics:

1.User Engagement:

Query 1:

```
select avg(total_posts_peruser) as avg_count_posts from (select
u.id,u.username,count(p.id) as total_posts_peruser
from users u
left join  photos p on u.id=p.user_id group by u.id) total
```

Output:

	avg_count_posts
▶	2.5700

Explanation:

- a) (Inner query(select u.id,u.username,count(p.id) as total_posts_peruser from users u left join photos p on u.id=p.user_id group by u.id) total):
 - I. From the images table, the inner subquery takes the user_id and post count for each user.
 - II. The number of posts (id) for each user is counted using the COUNT(id) trick.
 - III. Using the GROUP BY clause, the results are grouped by user_id.
- b) (Outer query(SELECT AVG(total_posts_peruser) AS avg_count_posts FROM....)):
 - I. Using the outcome of the inner subquery, the outer query determines the average (AVG) of all posts made by each user.
 - II. The result is presented as a single column named 'avg_count_posts'.

Main Objective:

The query counts the total number of posts for each user on Instagram, then computes the average of these counts to get the average number of posts per user.

Insight:

The average number of posts per user on Instagram is 2.57. That means on an average every user is posting at least 2 posts on Instagram.

Query 2:

```
select count(*)/count(distinct u.id) as taskB_1_2 from photos p join users u on
p.user_id = u.id;
```

Output:

	taskB_1_2
▶	3.4730

Explanation:

- a) (SELECT COUNT(*) / COUNT(DISTINCT u.id) AS taskB_1_2):
The SELECT clause determines the ratio of the overall count of rows in the result set (COUNT(*)) to the count of distinct user IDs (COUNT(DISTINCT u.id)).
- b) (FROM photos p JOIN users u on p.user_id = u.id):
The FROM clause specifies the tables used in the query and the tables 'users' and 'photos' are specified by aliases u and p and joins on condition on p.user_id = u.id.

Main Objective:

This query computes the ratio of the total number of rows in the result set that is produced when the images and users tables are joined to the total number of unique user IDs. The output provides a measure of the correlation between the photos and unique users.

Insight:

By the output we can observe that the ratio of total number of photos on Instagram to total number of users on Instagram is 3.473. That means total number of photos on Instagram is approximately 3.5 times the total number of users on Instagram.

2.Bots & Fake Accounts:

Query:

```
Select u.id AS bot_id, u.username as bot_name
from users u

where not exists (

select 1
from photos p
left join likes l on p.id = l.photo_id and l.user_id = u.id
where l.user_id is null
);
```

Output:

```
5  Aniya_Hackett
14 Jaclyn81
21 Rocio33
24 Maxwell.Halvorson
36 Ollie_Ledner37
41 Mckenna17
```


54 Duane60
57 Julien_Schmidt
66 Mike.Auer39
71 Nia_Haag
75 Leslie67
76 Janelle.Nikolaus81
91 Bethany20

Explanation:

- a) (SELECT u.id AS bot_id, u.username as bot_name):
 - I. The id and username columns from the users table, aliased as bot_id and bot_name, respectively, are the columns that are to be fetched in the result set, specified by the SELECT clause.
- b) WHERE NOT EXISTS (...):
 - I. The NOT EXISTS subquery's condition, which determines whether any results exist in the subquery at all, provides the basis for the WHERE clause's row filtering.
- c) (SELECT 143 FROM photos p LEFT JOIN likes l ON p.id = l.photo_id AND l.user_id = u.id WHERE l.user_id IS NULL):
 - I. The left join makes sure that all records from the photos table are included, and the condition l.user_id IS NULL checks for users who have not liked any photos.
 - II. The subquery chooses the constant value 143 for each user from the users table where there is no matching record in the combination of the photos and likes tables.

Main objective:

The purpose of the query is to find people who have not liked any photographs on the website. It does this by determining which users do not have any records in the combination of their likes and photos databases.

Insight:

We can conclude that there are 13 users (bots) who have liked every photo on the Instagram. Most of the usernames of the bots are ended with an integer.

3.Tech stack used:

- a. Software:
MySQL Workbench 8.0(version 8.0.34).
- b. System Requirements:
Processor : Intel(R) Core(TM) i3-10110U CPU @ 2.10GHz 2.59 GHz
RAM: 8.00 GB
System Type: 64-bit operating system, x64-based processor.
- c. Motivation for Usage:

MySQL Workbench is a visual database design tool and SQL query tool. It provides a user-friendly interface for planning, developing, and administering MySQL databases. It is often used in projects requiring MySQL databases.

4.Insights:

1) Loyal User Reward:

Every long-time Instagram user signed up in May 2016, indicating it may have been a peak month for registrations. A feature release, a marketing effort, or other outside variables could be responsible for this.

2) Inactive User Engagement:

On Instagram, those with usernames that end in numbers are less likely to share photos—15 of these individuals have never posted anything. This pattern might point to a tendency among specific users or account kinds.

3) Contest Winner Declaration:

The user 'zack_kemmer93' with user ID 52 and post ID 145 has the most likes (48), making them the contest winner. Understanding consumer preferences requires knowing which posts are popular and what responses mean.

4) Hashtag Research:

With 59 appearances, "smile" is the most popular hashtag on Instagram. With 42 tags, "Beach" is the second most popular. Examining trending hashtags can influence content strategy and increase interaction.

5) Ad Campaign Launch:

Thursday could be an excellent day to start advertising campaigns because it has the most user registrations. Recognising trends in user behaviour can help improve marketing campaigns.

6) User Engagement:

a. Average posts per user:

Users tend to be engaged and share content often, as seen by the 2.57 average number of posts per user. To measure user engagement and platform utilisation, this statistic is crucial.

b. Photos to users ratio:

With a ratio of 3.473 for total images to total users, it is possible that a large number of photos are contributed by a vibrant and active user base. Trends in content generation and user interaction may be evaluated with the use of this ratio.

7) Bots & Fake accounts:

Every Instagram post has been liked by 13 individuals, many of whose usernames finish in an integer. These users may be bots. Detecting and keeping updated on bot activity is essential to preserving a positive and genuine customer experience.

5.Result:

1. Data Structure Understanding:

Successfully understood the database schema, recognising the relationships between tables. To do insightful analysis, it is essential to have this fundamental understanding.

2. Marketing Insights:

- a. Found that the first people to sign up for Instagram were all in May of 2016. With the possible influence of past user acquisition practices in mind, this data helps direct marketing initiatives.
- b. Discovered dormant users whose usernames ended in digits, offering useful data for focused engagement tactics and user retention initiatives.
- c. Determined a contest winner by looking at likes, showing how important it is to comprehend user engagement indicators for prizes and contests.
- d. 'Smile' was found to be the most commonly used hashtag after an exploration of popular hashtags. Marketing initiatives and the development of content can be guided by such findings.
- e. Found that Thursday was the day with the most new users signing up, which helped marketers determine the best times to run advertising campaigns.

3. Technical Stack Utilization:

Successfully deployed MySQL Workbench for database examination, using its visual design and SQL query capabilities. The project's efficiency was enhanced by the technologies and tool selection.

4. Practical Insights for Decision-Making:

The project's results provide useful information for decision-making across several divisions, such as investor relations, product development, and marketing.

5. Impact on Future Instagram Development:

The study offers a thorough insight of user behaviour, preferences, and engagement patterns, which helps shape the future of the Instagram app.

6. Strategic Content Planning:

Understanding popular hashtags, user engagement metrics, and content creation patterns guides strategic content planning, ensuring alignment with user preferences.

7. Optimised Marketing Campaigns:

Marketers may maximise the reach and efficacy of their campaigns by having a thorough understanding of user registration trends, the best days to launch ads, and interaction data.