Instagram User Analytics Using MySQL

Prepared by: Shanti Jogi

Table of Contents

1. Project Description
2. Approach
3. Tech Stack Used
4. SQL Tasks and Insights
5. Results
6. Conclusion

Project Description

This project focuses on analyzing Instagram user interactions and engagement data using MySQL. As a data analyst, the goal is to derive insights that aid marketing, product, and development teams. The database comprises tables for users, photos, comments, likes, and follow relationships, enabling comprehensive analysis.

Approach

The approach involved creating a structured database, inserting relevant data, and running SQL queries to analyze user interactions. Tasks included identifying loyal users, inactive users, popular hashtags, and calculating user engagement metrics.

Tech Stack Used

MySQL Workbench: Used for database creation and query execution.

SQL Tasks and Insights

Task 1: Loyal User Reward

The marketing team wants to reward the most loyal users. The following query identifies the five oldest users:

SQL Query:

```
SELECT * FROM Users
ORDER BY created_at ASC
LIMIT 5;
```

```
#/MARKETING ANALYSIS/*
# 1) loyal user reward
select * from users
order by created_at
limit 5;
```

Result Grid | Filter Rows:

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

Insight: This query helps identify users who have been with the platform the longest.

Task 2: Inactive User Engagement

SQL Query:

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

```
#/MARKETING ANALYSIS/*
# 2) inactive users on Instagram
select users.username, users.id
from users
left join photos
on users.id=photos.user_id
where photos.id is null;
```

Result Grid | Filter Rows:

username	id
Aniya_Hackett	5
Kasandra_Homenick	7
Jaclyn81	14
Rocio33	21
Maxwell.Halvorson	24
Tierra.Trantow	25
Pearl7	34
Ollie_Ledner37	36
Mckenna17	41
David.Osinski47	45

```
#/MARKETING ANALYSIS/*
# 2) inactive users on Instagram
select Count(username) as inactive_users
from users
left join photos
on users.id=photos.user_id
where photos.id is null;
```

Result Grid | Filter Rows: | Export:

inactive_users
26

Insight: This query identifies users who have not uploaded any photos, highlighting potential inactive users for targeted engagement or re-engagement strategies.

Task 3: Contest Winner Declaration

SQL Query:

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

```
967  #/MARKETING ANALYSIS/*
968  # 3) contest winner declaration
969  select username, photos.id, photos.image_url, count(likes.user_id) as Total_likes
970  from photos
971  inner join users on users.id=photos.user_id
972  inner join likes on likes.photo_id=photos.id
973  group by photos.id
974  order by Total_likes desc
975  limit 1;
```

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

username	id	image_url	Total_likes
Zack_Kemmer93	145	https://jarret.name	48

Insight: This query identifies the photo with the most likes, helping to declare the contest winner based on user engagement.

Task 4: Hashtag Research

SQL Query:

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

```
977  #/MARKETING ANALYSIS/*
978  # 4)Hashtag research
979  select tag_name, count(photo_tags.photo_id) as photoId_count from photo_tags
980  inner join tags
981  on tags.id=photo_tags.tag_id
982  group by tag_name
983  order by photoId_count desc
984  limit 5;
```

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

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

Insight: This query identifies the top 5 most popular hashtags based on their frequency of use in photo uploads.

Task 5: Ad Campaign Launch

SQL Query:

SELECT dayname(created_at) AS Day, COUNT(created_at) AS total
FROM users
GROUP BY day
ORDER BY total DESC;

```
#/MARKETING ANALYSIS/*
# 5)Ad campaign Launch_(best of the day in week)
select dayname(created_at) as Day, count(created_at) as total
from users
group by day
order by total desc;
```

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

Day	total
Thursday	16
Sunday	16
Friday	15
Tuesday	14
Monday	14
Wednesday	13
Saturday	12

Insight: This query identifies the day of the week with the highest user sign-ups, helping to optimize timing for an ad campaign launch.

Task 6: User Engagement

SQL Query:

SELECT round((SELECT count(*) FROM photos)/(SELECT count(*) FROM users)) AS Average_users;

```
#/Investor Metrics/*
# 1) User Engagement - avg. number of posts on IG per user
    Select round((select count(*) from photos)/(select count(*) from users)) as Average_users;
```

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

Average_users
3

Insight: This query calculates the average number of photos uploaded per user, providing insight into overall user engagement.

Task 7: Bots & Fake Accounts

SQL Query:

```
SELECT username, id, count(*) AS Num_Likes
FROM users u
JOIN likes l on u.id=l.user_id
GROUP BY u.id
HAVING num_likes=(select count(*) from photos);
```

```
1002    #/Investor Metrics/*
1003    # 2) Bots and Fake Accounts
1004 •  select username, id, count(*) as Num_Likes
1005    from users u
1006    join likes l on u.id=l.user_id
1007    group by u.id
1008    having num_likes=(select count(*) from photos);
1009
```

Result Grid | Filter Rows: | Export: | Wrap Cc

username	id	Num_Likes
Aniya_Hackett	5	257
Jadyn81	14	257
Rocio33	21	257
Maxwell.Halvorson	24	257
Ollie_Ledner37	36	257
Mckenna17	41	257
Duane60	54	257
Julien_Schmidt	57	257

Result 46

Insight : This query identifies users whose number of likes equals the total number of photos, potentially flagging them as bots or fake accounts exhibiting suspicious activity.

Results

Summarized outcomes of the analysis include identifying loyal users, optimizing ad schedules, and detecting potential bot accounts.

Conclusion

This project demonstrated the utility of SQL in analyzing user interactions and engagement. The insights derived can significantly influence strategic decisions in marketing, product development, and user engagement.