



Instagram User Analytics

Case Study Project-2
for Trainity DA training

Prepared by:
Nilesh Kulkarni

Content/Agenda

2

Project Description

Approach

Tech Stack Used

Insights

Results

01/02/2025

Project Description:



Instagram User Analytics:

With given data on user activities on the social media platform, the aim of the project is to find new business insights those can be used across business teams to launch new marketing campaign, track the success of app using current user engagement, promote user engagement and discover new features that can be built to improve user experience.

The in-scope business insights to find are as listed below:

A) Marketing Analysis:

1. Loyal User Reward – Find most loyal users
2. Inactive Users Engagement – Find inactive users
3. Contest Winner Declaration – Find user & photo getting most likes
4. Hashtag Research – Find most commonly used Hashtags
5. Ad Campaign Launch – Find Best time to launch an Ad

B) Investor Metrics:

1. User Engagement: Find stats about platform usage/user activities
2. Bots & Fake Account: Find potential Bots (Fake Users) on the platform

Project Approach:

High level steps for the Project approach are as outlined below:

- **Database creation:** Create the database using provided DDL statements (SQL)
- **Data Loading:** Load the provided data into database using given DML statements (SQL)
- **Analysis:** Analyse each insights requirement in detail and prepare SQL queries. Select optimal and efficient SQL queries/approach.
- **Extract insights:** Run the SQL queries to extract new insights as required
- **Review:** Review and cross check SQL output to verify it matches with the requirements
- **Document:** Document the insights and results to be shared across business teams

Tech-Stack used:



- ❖ **Operating System: Microsoft Windows 11** Version 22H2
- ❖ **Database: MySQL Community Server** – GPL version 8.0.40. This is free and open source relation database which uses SQL to interact with database
- ❖ **Documentation: Microsoft office 365 & Acrobat PDF**

Insights



Insight No.A.1 – “Loyal User Award”

SQL Query & Output for Top 5 users in order of
joining date-time on Instagram

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.

Task: Identify the five oldest users on Instagram from the provided database.

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'ig_clone' selected. The main editor shows an SQL query to select the top 5 users by creation date. The 'Result Grid' at the bottom displays the output of the query, listing the top 5 most loyal users.

SQL Query:

```
1 SELECT
2     id AS "User ID:",
3     username AS 'Most Loyal Users:',
4     DATE(created_at) AS 'Date Joined Instagram:',
5     TIME(created_at) AS 'Time Joined Instagram:'
6 FROM
7     users
8 ORDER BY
9     created_at
10 LIMIT 5;
```

Result Grid:

User ID:	Most Loyal Users:	Date Joined Instagram:	Time Joined Instagram:
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 No. A.2 - "Inactive Users Engagement"

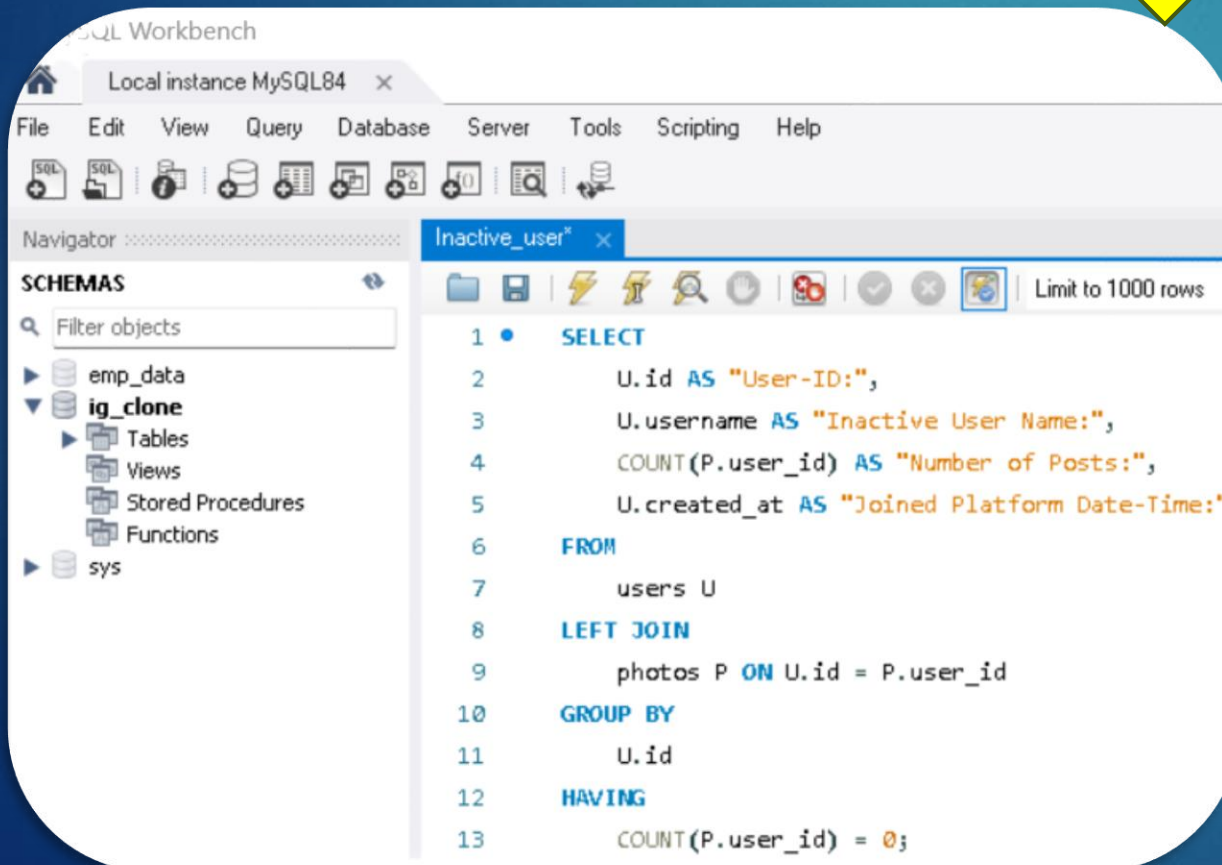
8

01/02/2025

Inactive User Engagement: The team wants to encourage inactive users to start posting by sending them promotional emails.

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

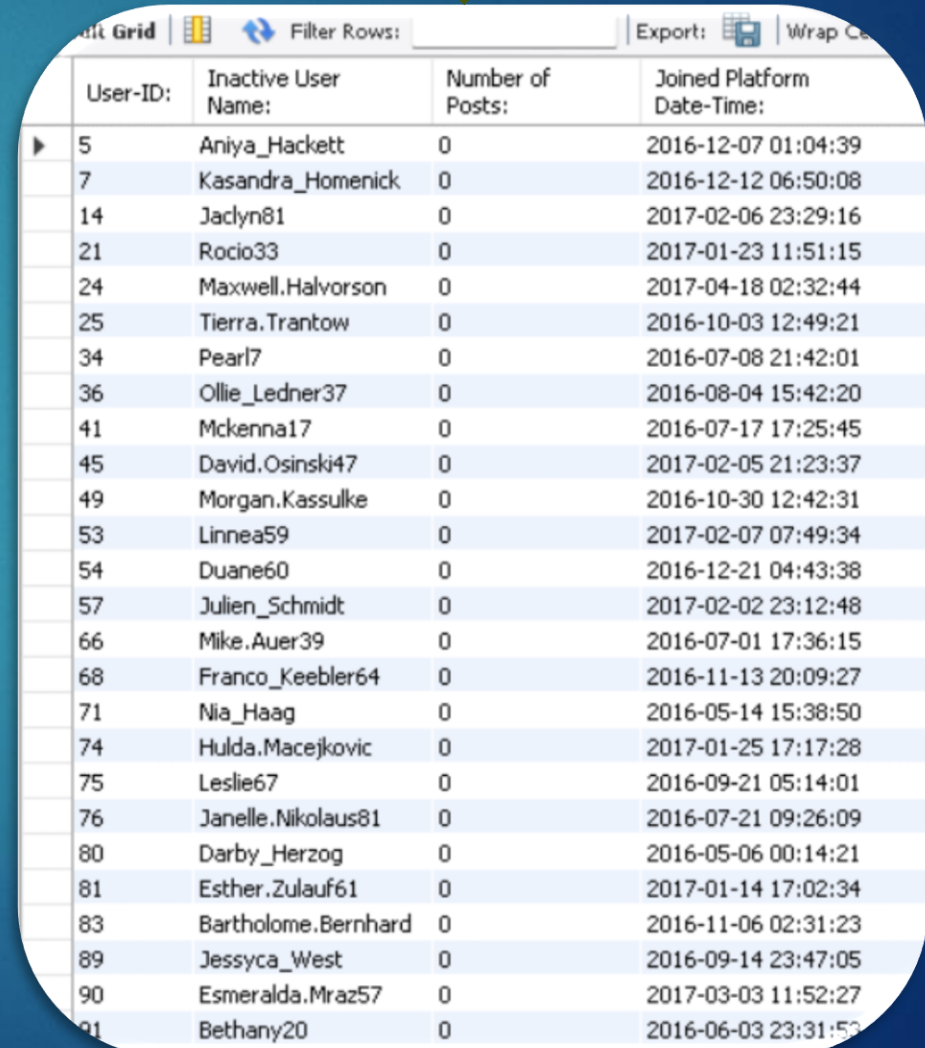
SQL Query for finding "Inactive users"



The screenshot shows the SQL Workbench interface. The 'SCHEMAS' panel on the left shows a database named 'ig_clone' with tables, views, stored procedures, and functions. The main query editor displays the following SQL query:

```
1 SELECT
2     U.id AS "User-ID:",
3     U.username AS "Inactive User Name:",
4     COUNT(P.user_id) AS "Number of Posts:",
5     U.created_at AS "Joined Platform Date-Time:"
6 FROM
7     users U
8 LEFT JOIN
9     photos P ON U.id = P.user_id
10 GROUP BY
11     U.id
12 HAVING
13     COUNT(P.user_id) = 0;
```

List of Inactive Users who have never posted a single photo (Total 26 Inactive users)



The screenshot shows a table with 4 columns: User-ID, Inactive User Name, Number of Posts, and Joined Platform Date-Time. The table contains 26 rows of data, all with 0 posts.

User-ID:	Inactive User Name:	Number of Posts:	Joined Platform Date-Time:
5	Aniya_Hackett	0	2016-12-07 01:04:39
7	Kassandra_Homenick	0	2016-12-12 06:50:08
14	Jaclyn81	0	2017-02-06 23:29:16
21	Rocio33	0	2017-01-23 11:51:15
24	Maxwell.Halvorson	0	2017-04-18 02:32:44
25	Tierra.Trantow	0	2016-10-03 12:49:21
34	Pearl7	0	2016-07-08 21:42:01
36	Ollie_Ledner37	0	2016-08-04 15:42:20
41	Mckenna17	0	2016-07-17 17:25:45
45	David.Osinski47	0	2017-02-05 21:23:37
49	Morgan.Kassulke	0	2016-10-30 12:42:31
53	Linnea59	0	2017-02-07 07:49:34
54	Duane60	0	2016-12-21 04:43:38
57	Julien_Schmidt	0	2017-02-02 23:12:48
66	Mike.Auer39	0	2016-07-01 17:36:15
68	Franco_Keebler64	0	2016-11-13 20:09:27
71	Nia_Haag	0	2016-05-14 15:38:50
74	Hulda.Macejkovic	0	2017-01-25 17:17:28
75	Leslie67	0	2016-09-21 05:14:01
76	Janelle.Nikolaus81	0	2016-07-21 09:26:09
80	Darby_Herzog	0	2016-05-06 00:14:21
81	Esther.Zulauf61	0	2017-01-14 17:02:34
83	Bartholome.Bernhard	0	2016-11-06 02:31:23
89	Jessyca_West	0	2016-09-14 23:47:05
90	Esmeralda.Mraz57	0	2017-03-03 11:52:27
91	Bethany20	0	2016-06-03 23:31:53

Insight No. A.3 – “Contest Winner Declaration ”

Contest Winner Declaration: The team has organized a contest where the user with the most likes on a single photo wins.

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

SQL Query & Output for finding User and Photo details with Most Likes



The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' panel with a tree view containing 'emp_data', 'ig_clone', and 'sys'. The main editor window shows an SQL query for the 'ig_clone' schema. The query is as follows:

```
1 • SELECT
2     U.username AS 'User with Most Likes:',
3     U.id AS 'User ID:',
4     P.id AS 'Photo Id:',
5     P.created_at AS 'Photo Posting Date-Time:'
6 FROM
7     users U
8     JOIN
9     photos P ON U.id = P.user_id
10 WHERE
11     P.id = (SELECT
12             photo_id
13             FROM
14             likes
15             GROUP BY photo_id
16             ORDER BY COUNT(photo_id) DESC
17             LIMIT 1)
18 LIMIT 1;
```

The query is highlighted with a yellow box labeled "SQL Query". Below the query editor, the 'Result Grid' shows the output of the query. The first row is highlighted with a yellow box labeled "User with most likes".

User with Most Likes:	User ID:	Photo Id:	Photo Posting Date-Time:
Zack_Kemmer93	52	145	2024-12-19 23:28:15

Insight No. A.4 – “Hashtag Research”

Hashtag Research: A partner brand wants to know the most popular hashtags to use in their posts to reach the most people.

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

SQL query & the output containing the top five most commonly used hashtags



The screenshot shows the SQL Workbench interface with a MySQL database. The query window displays the following SQL code:

```
1 SELECT
2     T.tag_name AS 'Hashtag Name:',
3     COUNT(T.tag_name) AS 'Hashtag Count:'
4 FROM
5     tags T
6 INNER JOIN
7     photo_tags P ON P.tag_id = T.id
8 GROUP BY
9     T.tag_name
10 ORDER BY
11     count(T.tag_name) DESC
12 LIMIT 5;
```

The results are shown in the Result Grid below:

Hashtag Name:	Hashtag Count:
smile	59
beach	42
party	39
fun	38
concert	24

Annotations on the screenshot include a yellow box around the SQL query text, a yellow box around the result grid, and a yellow box around the text "Top 5 most used Hashtags".

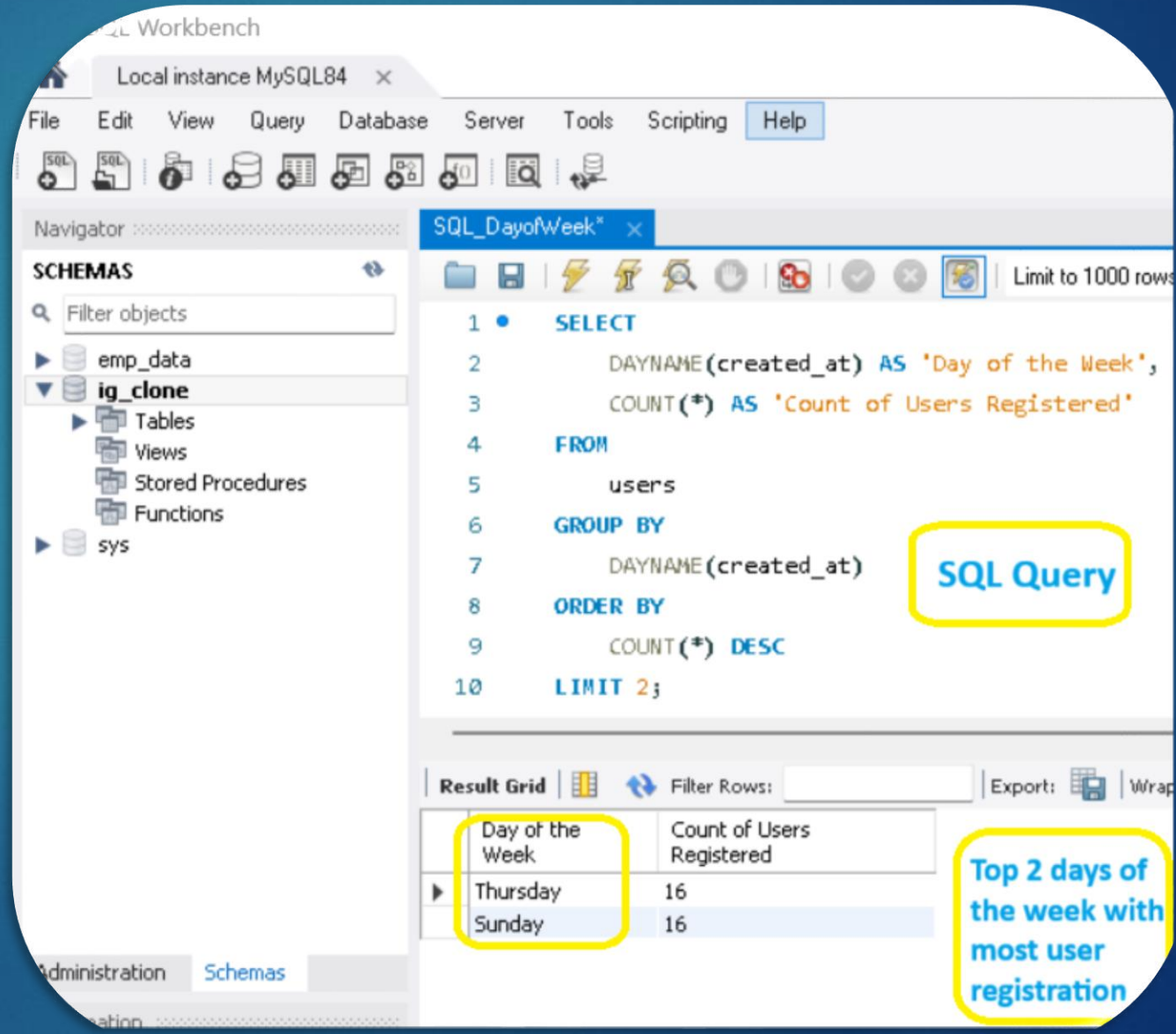
Insight No. A.5 – “Ad Campaign Launch”

SQL Query & output shown for top 2 days of the week when most users are registered

11

Ad Campaign Launch: The team wants to know the best day of the week to launch ads.

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



The screenshot shows the SQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'ig_clone' selected. The main window shows the 'SQL_DayofWeek*' query editor with the following SQL code:

```
1 SELECT
2     DAYNAME(created_at) AS 'Day of the Week',
3     COUNT(*) AS 'Count of Users Registered'
4 FROM
5     users
6 GROUP BY
7     DAYNAME(created_at)
8 ORDER BY
9     COUNT(*) DESC
10 LIMIT 2;
```

The 'SQL Query' text is highlighted with a yellow box. Below the query editor, the 'Result Grid' shows the output:

Day of the Week	Count of Users Registered
Thursday	16
Sunday	16

The 'Day of the Week' column header and the first two rows of data are highlighted with a yellow box. The text 'Top 2 days of the week with most user registration' is highlighted with a yellow box.

01/02/2025

Insight No. B.1 - “User Engagement”

12

SQL query and output showing user engagement stats



01/02/2025

User Engagement: Investors want to know if users are still active and posting on Instagram or if they are making fewer posts.

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.

The screenshot shows the MySQL Workbench interface. The 'Scripting' tab is active, displaying the following SQL query:

```
1 • SELECT
2     ((SELECT COUNT(id) FROM photos) /
3      (SELECT COUNT(DISTINCT user_id) FROM photos) AS "Average Posts/User",
4
5     ((SELECT COUNT(id) FROM photos) /
6      (SELECT COUNT(id) FROM users) AS "Total Photos/Total Users";
```

The result grid below the query shows the following data:

Average Posts/User	Total Photos/Total Users
3.4730	2.5700

Annotations on the screenshot include a yellow box around the SQL query text, a yellow box around the result grid, and a yellow box around the text 'Stats about Users Engagement'.

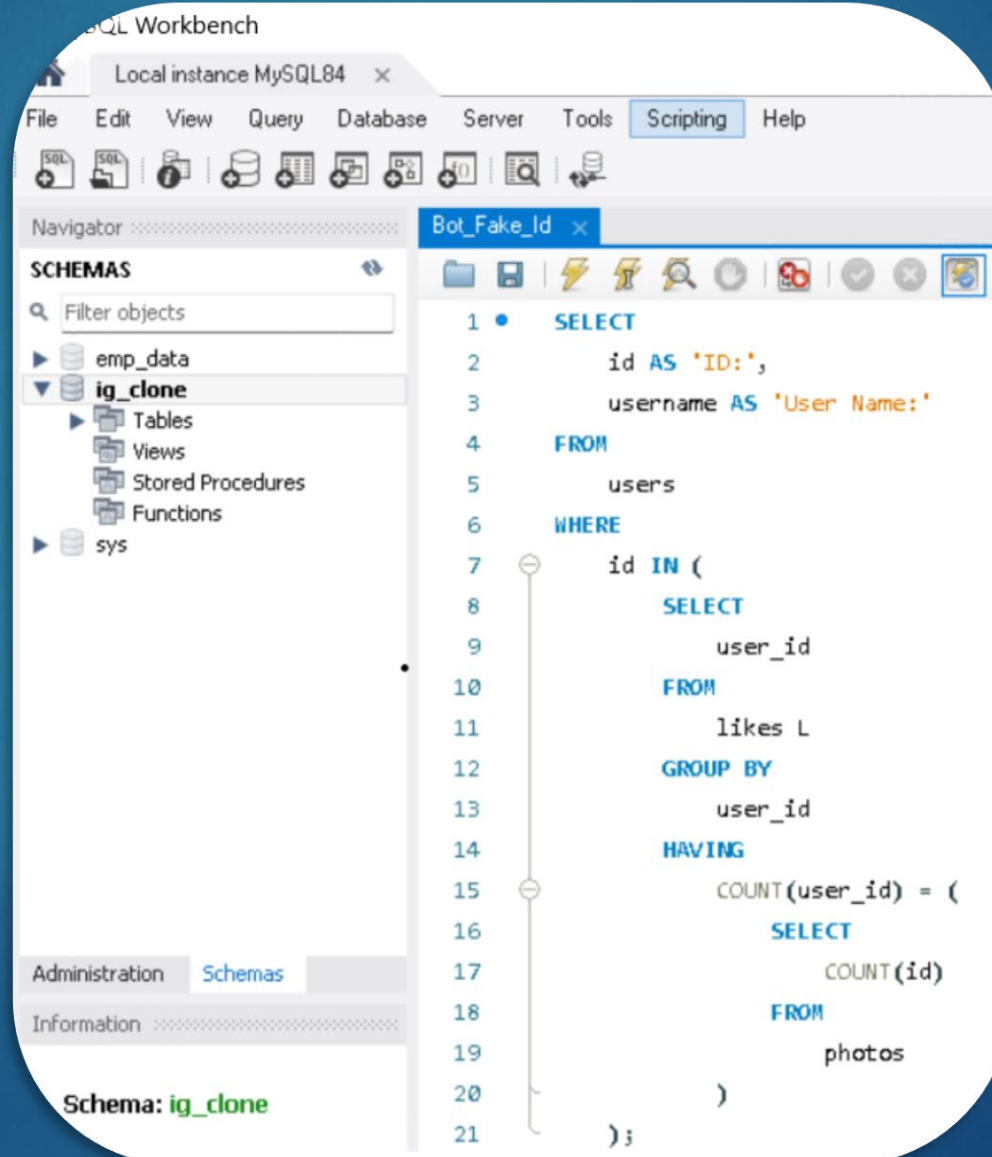
Insight No. B.2 – “Bots & Fake Accounts”

13

Bots & Fake Accounts: Investors want to know if the platform is crowded with fake and dummy accounts

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

SQL query →



The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left shows the 'ig_clone' database selected. The 'Query' editor in the center contains the following SQL query:

```
1 SELECT
2     id AS 'ID:',
3     username AS 'User Name:'
4 FROM
5     users
6 WHERE
7     id IN (
8         SELECT
9             user_id
10        FROM
11            likes L
12        GROUP BY
13            user_id
14        HAVING
15            COUNT(user_id) = (
16                SELECT
17                    COUNT(id)
18                FROM
19                    photos
20            )
21    );
```

SQL Output showing potential bots ↓



The screenshot shows the 'Result Grid' of the SQL query. It displays a list of users who have liked every single photo on the site, identified as potential bots. The table has two columns: 'ID:' and 'User Name:'.

ID:	User Name:
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

01/02/2025

Conclusion for the Analysis – Business Value-Add



- ❖ Business/Marketing team can use provided insights for
 - ❖ Reward most loyal users & Reach out to inactive users – promoting user engagement
 - ❖ Use most popular Hashtags – to collaborate with partner brands
 - ❖ Effective Brand promotions – using most active days on platform
 - ❖ Track app success – using provided user/post/photo stats
- ❖ Business can work on removing or tracking further the users reported as potential Bots to enhance end user experience and optimize app/resource usage
- ❖ Overall, the insights would add significant value to business teams

Personal Achievement/Upskilling

- ❖ Developed skills of data analysis using SQL queries (specifically with MySQL database)
- ❖ Learnt extracting useful business insights like user engagement, track app success, stats on user activities etc using efficient SQL queries
- ❖ Understood intricacies of useful business insights and how data analytics can help business with spot on insights





Thank You!

Prepared by:
Nilesh Kulkarni