




# *INSTAGRAM USER ANALYTICS*



Aaditya Singh  
TRAINITY

## Table of content

Sr. no.	Content	Pg. no.
1.	Cover Page	(i)
2.	Table of Content	1
3.	List of Figures	2
4.	Project Description	3
5.	Approach	4
6.	Tech-Stack Used	6
7.	Insights	7
8.	Result	24
9.	Drive Link	26

## List of Figures

Sr. no.	Content	Pg. no.
1.1	ER diagram of ig_clone	7
2.1	Loyal User Reward	10
2.2	Inactive User Engagement	12
2.3	Contest Winner Declaration	14
2.4	Hashtag Research	16
2.5	Ad campaign launch	18
3.1	User Engagement query 1	20
3.2	User Engagement query 2	21
3.3	Fake account detection	22

# Project Description

Instagram User Analytics is a project which focuses on highlighting data driven insights to help the employer in making better decision which would in turn increase interactions and engagement of target users with the app. Insights are extracted with the help of Data Analysis which is the science of analyzing raw data to make conclusions about that information.

Various teams within the company can be benefitted from this methodology. This analysis aims to enable Instagram to make accurate and reliable decisions, understand customer preferences, identify inefficiencies, predict trends and detect frauds. The steps which were taken in my data analysis timeline were:

1. Defining the question
2. Collection and Cleaning of Data
3. Analyzing the data
4. Creating insights
5. Report creation
6. Sharing insights

This activity list helped in understanding the business, data preparation, model building, evaluation and result deployment.

# Approach

I accumulated all the knowledge required for solving the business problems presented by the company (Instagram). This was possible by actively learning the required modules from Trainity and implementing what was taught throughout the process by myself in the background.

The problems presented were of 2 measures:

1. Marketing analysis
2. Investor metrics

These tasks were further broken down to specific problems which would help the marketing team, management team and product team to influence the Instagram platform accordingly.

My analysis skills were used to extract meaningful insights such as:

1. **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.
2. **Inactive User Engagement:** The team wants to encourage inactive users to start posting by sending them promotional emails.
3. **Contest Winner Declaration:** The team has organized a contest where the user with the most likes on a single photo wins.
4. **Hashtag Research:** A partner brand wants to know the most popular hashtags to use in their posts to reach the most people.
5. **Ad Campaign Launch:** The team wants to know the best day of the week to launch ads.

6. **User Engagement:** Investors want to know if users are still active and posting on Instagram or if they are making fewer posts.
7. **Bots & Fake Accounts:** Investors want to know if the platform is crowded with fake and dummy accounts.

All the SQL queries and results obtained are shared in the [Insights](#) and Results chapters of this report.

## Tech-stack used

The data was stored in a tabular format so a relational database model was used. A relational database management system stores and fetches data in the form of tables, therefore MySQL was used as the software since it is based on the principle of RDBMS.

The version used was MySQL workbench version 8.0.34. MySQL has characteristics which include ease of service and is simple to use making it extremely manageable. MySQL is openly available for download and free to install.

When we open MySQL workbench for the first time after installation, we are greeted with a welcome page. The steps followed further to load the database was as follows:

1. Go to MySQL Connections
2. Click on 'Plus' icon to setup a new connection.
3. Give the connection a name.
4. Test the connection.
5. Click OK.

Once the database connection was successfully made we can run the database query provided we are presented with a database called 'ig\_clone'. We shall use MySQL to analyze Instagram user data and answer questions posed by the team.

# Insights

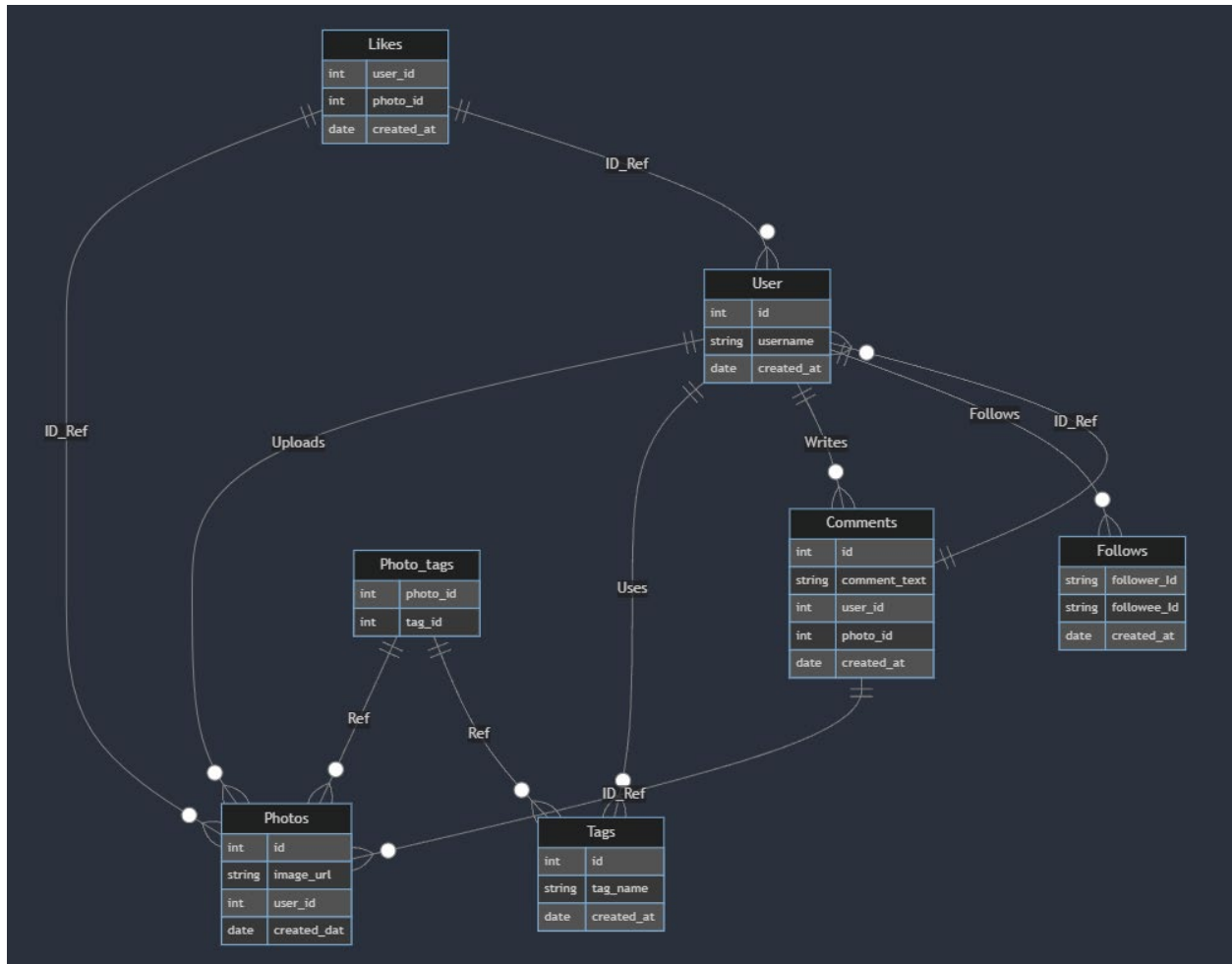


Fig 1.1 ER diagram of `ig_clone`

The ER diagram of `ig_clone` sheds light on how it functions and the connections of various table such as users, photos, comments, likes, follows, tags, and a junction table for the relationship between photos and tags to each other. This schema will help us in establishing certain actions and their references from other table.



## Table Description:

### 'users' table-

1. 'id': unique id for the user. (primary key)
2. 'username': username of the user.
3. 'created\_at': timestamp of when user was created.

### 'photos' table-

1. 'id': unique id of the photo. (primary key)
2. 'image\_url': URL of the photo.
3. 'user\_id': references 'id' in users table. (foreign key)
4. 'created\_at': timestamp of when the photo was created.

### 'comments' table-

1. 'id': unique id for each comment. (primary key)
2. 'comment\_text': The text of the comment.
3. 'user\_id': references 'id' in users table. (foreign key)
4. 'photo\_id': references 'id' in photos table. (foreign key)
5. 'created\_at': timestamp of when the comment was created.

### 'likes' table-

1. 'user\_id': references 'id' in users table. (foreign key)
2. 'photo\_id': references 'id' in photos table. (foreign key)
3. 'created\_at': timestamp of when the user liked the photo.

### 'follows' table-

1. 'follower\_id': references the id of follower in the users table. (foreign key)
2. 'followee\_id': references the id of followee in the users table. (foreign key)
3. 'created\_at': timestamp of when the following took place.

'tags' table-

1. 'id': unique id for the tag. (primary key)
2. 'tag\_name': The name of the tag.
3. 'created\_at': timestamp of the tag.

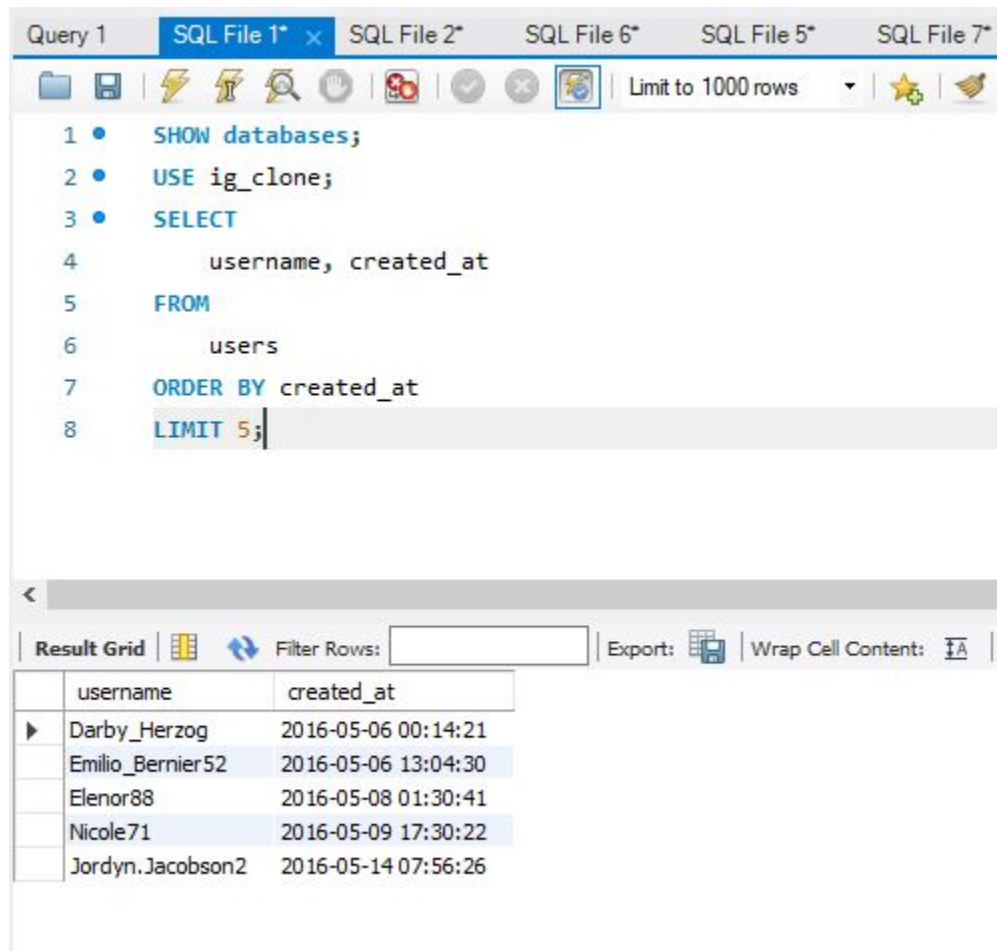
'photos\_tag' table-

1. 'photo\_id': references 'id' in photos table. (foreign key)
2. 'tag\_id': references the id in the tags table. (foreign key)

Now that the table has been described and a simplified structure is represented of the entire database, lets dive into the business problems that have been solved and by what methodology.

## Marketing analysis tasks:

Task 1: Loyal user reward- Identify the five oldest users on Instagram from the provided database.



```
1 • SHOW databases;
2 • USE ig_clone;
3 • SELECT
4     username, created_at
5 FROM
6     users
7 ORDER BY created_at
8 LIMIT 5;
```

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

Fig 2.1: Loyal User Reward query and result

Since we are using 'ig\_clone' database, let's tell MySQL that by using the USE statement. In order of execution (according to MySQL)-

1. FROM users table.
2. SELECT username and created at.
3. ORDER BY the created time
4. LIMIT them to 5

Once we run this query we are presented with the 5 oldest users on Instagram who are –

Darby\_Herzog

Emilio\_Bernier52

Elenor88

Nicole71

Jordyn.Jacobson2

Task 2: Inactive User Engagement- Identify users who have never posted a single photo on Instagram.

Query 1   SQL File 1\*   **SQL File 2\*** x   SQL File 6\*   SQL File 5\*   SQL File 7\*   SQL File 8\*   SQL File 9\*

Limit to 1000 rows

```

2  SELECT
3      *
4  FROM
5      users;
6  • SELECT
7      *
8  FROM
9      photos;
10 • SELECT
11     *
12  FROM
13     users
14     LEFT JOIN
15     photos ON photos.user_id = users.id
16  ORDER BY photos.user_id;

```

Result Grid   Filter Rows:   Export:   Wrap Cell Content:  

	id	username	created_at	id	image_url	user_id	created_dat
▶	5	Aniya_Hackett	2016-12-07 01:04:39	NULL	NULL	NULL	NULL
	7	Kassandra_Homenick	2016-12-12 06:50:08	NULL	NULL	NULL	NULL
	14	Jadyn81	2017-02-06 23:29:16	NULL	NULL	NULL	NULL
	21	Rocio33	2017-01-23 11:51:15	NULL	NULL	NULL	NULL
	24	Maxwell_Halvorson	2017-04-18 02:32:44	NULL	NULL	NULL	NULL
	25	Tierra.Trantow	2016-10-03 12:49:21	NULL	NULL	NULL	NULL
	34	Pearl7	2016-07-08 21:42:01	NULL	NULL	NULL	NULL
	36	Ollie_Ledner37	2016-08-04 15:42:20	NULL	NULL	NULL	NULL
	41	Mckenna17	2016-07-17 17:25:45	NULL	NULL	NULL	NULL
	45	David.Osinski47	2017-02-05 21:23:37	NULL	NULL	NULL	NULL
	49	Morgan.Kassulke	2016-10-30 12:42:31	NULL	NULL	NULL	NULL
	53	Linnea59	2017-02-07 07:49:34	NULL	NULL	NULL	NULL
	54	Duane60	2016-12-21 04:43:38	NULL	NULL	NULL	NULL
	57	Julien_Schmidt	2017-02-02 23:12:48	NULL	NULL	NULL	NULL
	66	Mike.Auer39	2016-07-01 17:36:15	NULL	NULL	NULL	NULL
	68	Franco_Keebler64	2016-11-13 20:09:27	NULL	NULL	NULL	NULL
	71	Nia_Haag	2016-05-14 15:38:50	NULL	NULL	NULL	NULL
	74	Hulda.Macejkovic	2017-01-25 17:17:28	NULL	NULL	NULL	NULL
	75	Leslie67	2016-09-21 05:14:01	NULL	NULL	NULL	NULL
	76	Janelle.Nikolaus81	2016-07-21 09:26:09	NULL	NULL	NULL	NULL
	80	Darby_Herzog	2016-05-06 00:14:21	NULL	NULL	NULL	NULL
	81	Esther.Zulauf61	2017-01-14 17:02:34	NULL	NULL	NULL	NULL
	83	Bartholome.Bernhard	2016-11-06 02:31:23	NULL	NULL	NULL	NULL
	89	Jessyca_West	2016-09-14 23:47:05	NULL	NULL	NULL	NULL
	90	Esmeralda.Mraz57	2017-03-03 11:52:27	NULL	NULL	NULL	NULL
	91	Bethany20	2016-06-03 23:31:53	NULL	NULL	NULL	NULL

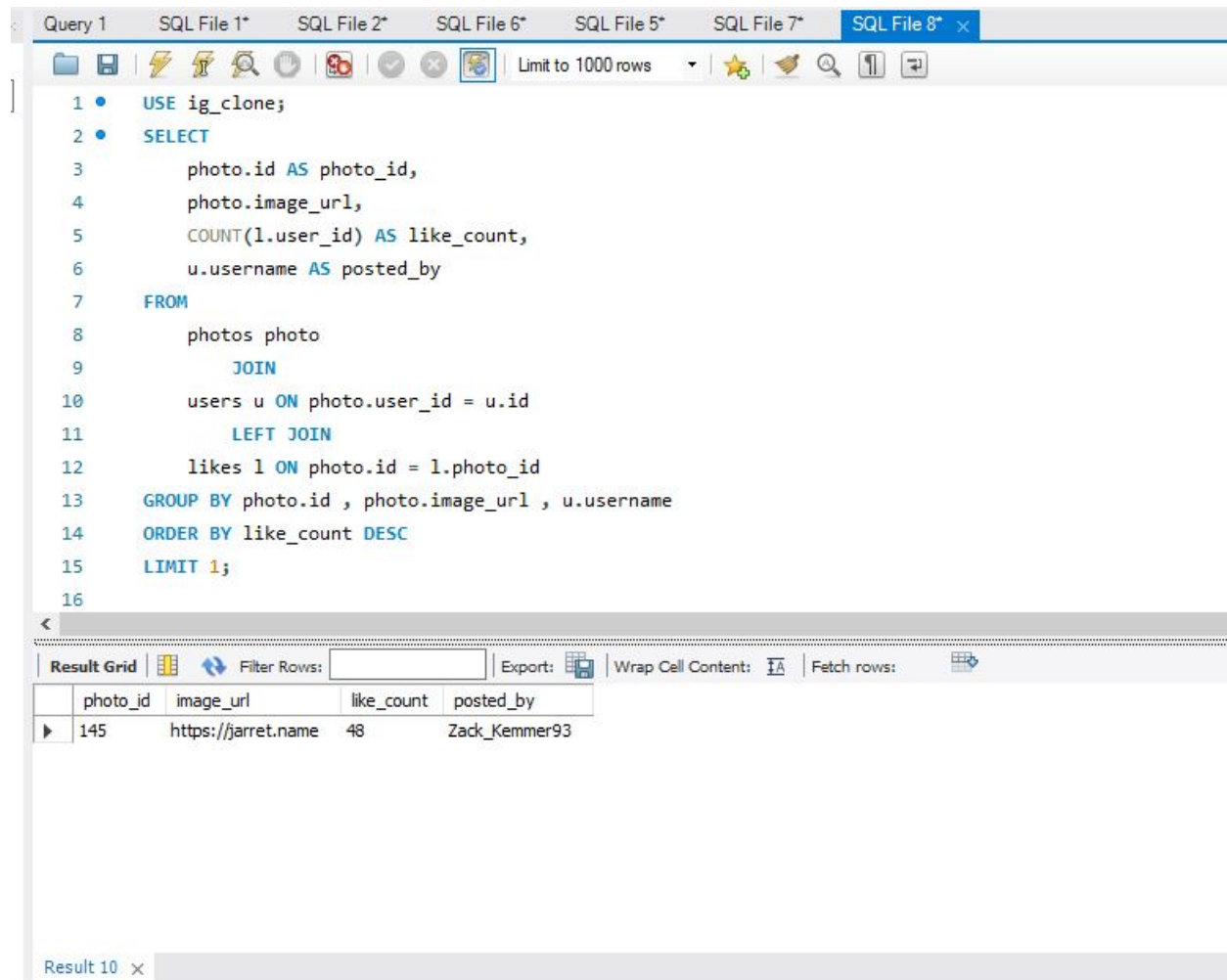
Result 1 x

Fig 2.2: Inactive User Engagement query and result

Here we use JOIN statement which allows us to pull data from multiple table at the same time but they need to have a related column between them. So according to our requirement we need to identify users who have never posted photos. So to do so we LEFT JOIN our id of photos with the id of all the users and order it so that the resulting values are null values. These null values indicate users username who have never posted on Instagram-

Aniya\_Hackett, Kasandra\_Homenick, Jaclyn81, Rocio33,  
Maxwell.Halvorson, Tierra.Trantow, Pearl7, Ollie\_Ledner37, Mckenna17,  
David.Osinski47, Morgan.Kassulke, Linnea59, Duane60, Julien\_Schmidt,  
Mike.Auer39, Franco\_Keebler64, Nia\_Haag, Hulda.Macejkovic, Leslie67,  
Janelle.Nikolaus81, Darby\_Herzog, Esther.Zulauf61,  
Bartholome.Bernhard, Jessyca\_West, Esmeralda.Mraz57, Bethany20

Task 3: Contest Winner Declaration- Determine the user with most likes on a single photo and provide their details to the team.



The screenshot shows a SQL IDE interface with a query editor and a result grid. The query editor contains the following SQL code:

```
1 • USE ig_clone;
2 • SELECT
3     photo.id AS photo_id,
4     photo.image_url,
5     COUNT(l.user_id) AS like_count,
6     u.username AS posted_by
7 FROM
8     photos photo
9     JOIN
10    users u ON photo.user_id = u.id
11    LEFT JOIN
12    likes l ON photo.id = l.photo_id
13 GROUP BY photo.id , photo.image_url , u.username
14 ORDER BY like_count DESC
15 LIMIT 1;
16
```

The result grid shows the following data:

photo_id	image_url	like_count	posted_by
145	https://jarret.name	48	Zack_Kemmer93

Fig 2.3: Contest Winner Declaration query and result

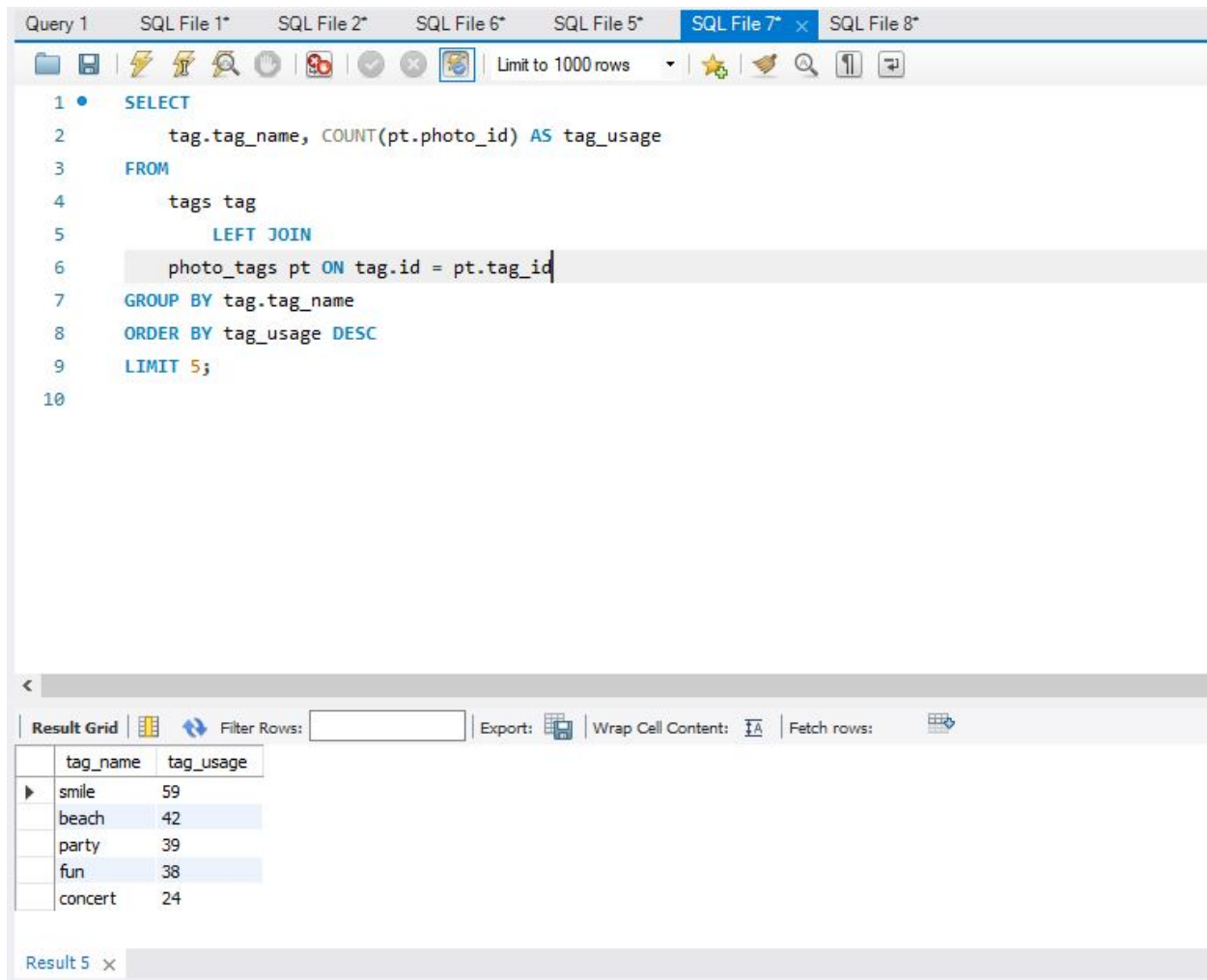
Again, JOIN statement is used for solving this query. This time we need to find a singular user who has the most liked photo on the platform and share all of their information with the team. We share the photo id, image url and the amount of likes the post has along with the username in the SELECT clause. We use LEFT JOIN on the photos table with the like table based on the id of the photo, the group by will give us the result in group of photo id and image url. The order by will order the

result in descending order (DESC) since we need the highest liked photo on Instagram.

Here, the most liked photo\_id 145 was posted by Zack\_Kemmer93 and the amount was 48.



Task 4: Hashtag Research- Identify and suggest the top five most commonly used hashtags on the platform.



```
1 • SELECT
2     tag.tag_name, COUNT(pt.photo_id) AS tag_usage
3 FROM
4     tags tag
5     LEFT JOIN
6     photo_tags pt ON tag.id = pt.tag_id
7 GROUP BY tag.tag_name
8 ORDER BY tag_usage DESC
9 LIMIT 5;
10
```

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

Fig2.4: Hashtag Research query and result

Here we are trying to find out what are the top 5 hashtags used on Instagram and to do so we are using the tag name along with the count of how many photo ids have used it, then ordering it in descending order (DESC) according to the tag usage. Tag\_usage was defined as a column which is the count of photo ids using that particular hashtag. The LEFT JOIN is performed on tag table with photo\_tag based on their tag id. This is then grouped by the tag name. The LIMIT 5 clause only gives us 5 result in order.

The top 5 used hashtags with their "name" and amount are-

Smile 59

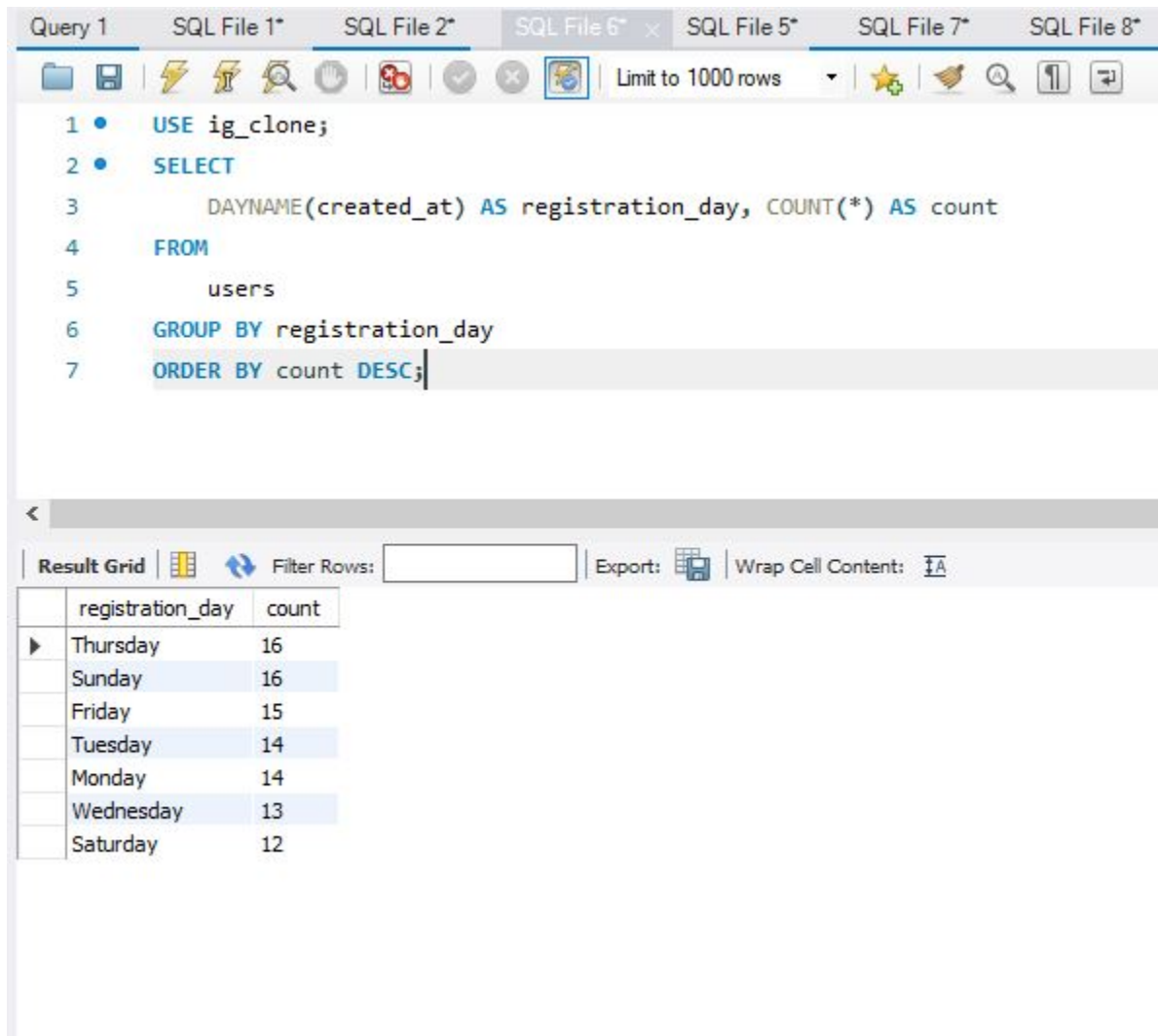
beach 42

party 39

fun 38

concert 24

Task 5: Ad campaign launch- Determine the day of the week when most users register on Instagram. Provide insights on when to schedule an ad campaign.



```
1 • USE ig_clone;
2 • SELECT
3     DAYNAME(created_at) AS registration_day, COUNT(*) AS count
4 FROM
5     users
6 GROUP BY registration_day
7 ORDER BY count DESC;
```

registration_day	count
Thursday	16
Sunday	16
Friday	15
Tuesday	14
Monday	14
Wednesday	13
Saturday	12

Fig 2.5 Ad campaign launch query and result

Finally, we make use of the created\_at attribute given in our users table. The DAYNAME function helps in extracting day of the week from the given created\_at timestamp. We rename this as registration\_day and similar to our other usage of count function we find the number of registrations that take place on each day of the week from the users table. Later on we group these according to registration day and order it

in descending order (DESC) of our count so that the day with the highest registration is on top.

The results were as follows-

Thursday 16

Sunday 16

Friday 15

Tuesday 14

Monday 14

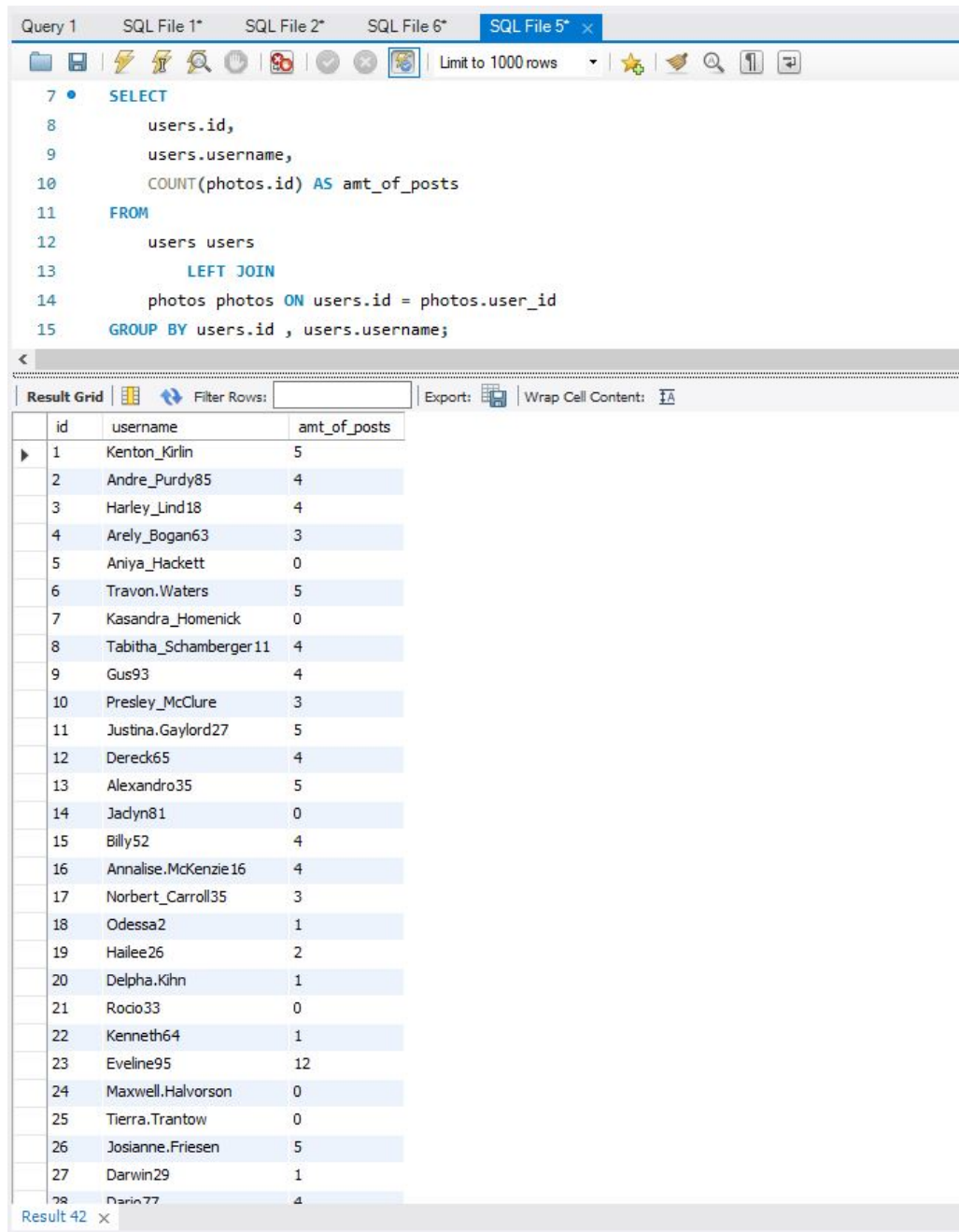
Wednesday 13

Saturday 12

As we can see, an ad campaign should ideally be launched on Thursday and Sunday and less likely on a Saturday.

## Investor metrics

Task1: User Engagement- 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.



```
7 • SELECT
8     users.id,
9     users.username,
10    COUNT(photos.id) AS amt_of_posts
11 FROM
12     users users
13    LEFT JOIN
14    photos photos ON users.id = photos.user_id
15 GROUP BY users.id , users.username;
```

	id	username	amt_of_posts
1	Kenton_Kirlin		5
2	Andre_Purdy85		4
3	Harley_Lind18		4
4	Arely_Bogan63		3
5	Aniya_Hackett		0
6	Travon.Waters		5
7	Kassandra_Homenick		0
8	Tabitha_Schamberger11		4
9	Gus93		4
10	Presley_McClure		3
11	Justina.Gaylord27		5
12	Dereck65		4
13	Alexandro35		5
14	Jadlyn81		0
15	Billy52		4
16	Annalise.McKenzie16		4
17	Norbert_Carroll35		3
18	Odessa2		1
19	Hailee26		2
20	Delpha.Kihn		1
21	Rocio33		0
22	Kenneth64		1
23	Eveline95		12
24	Maxwell.Halvorson		0
25	Tierra.Trantow		0
26	Josianne.Friesen		5
27	Darwin29		1
28	Darin77		4

Fig 3.1: User Engagement query 1 and result

The screenshot shows a SQL IDE with a query editor and a result grid. The query is as follows:

```

1 • USE ig_clone;
2 • SELECT
3     (SELECT COUNT(*) FROM photos) AS photos,
4     (SELECT COUNT(*) FROM users) AS users,
5     (SELECT COUNT(*) FROM photos) / (SELECT COUNT(*) FROM users) AS average_photos_per_user;

```

The result grid displays the following data:

photos	users	average_photos_per_user
257	100	2.5700

Fig 3.2: User Engagement query 2 and result

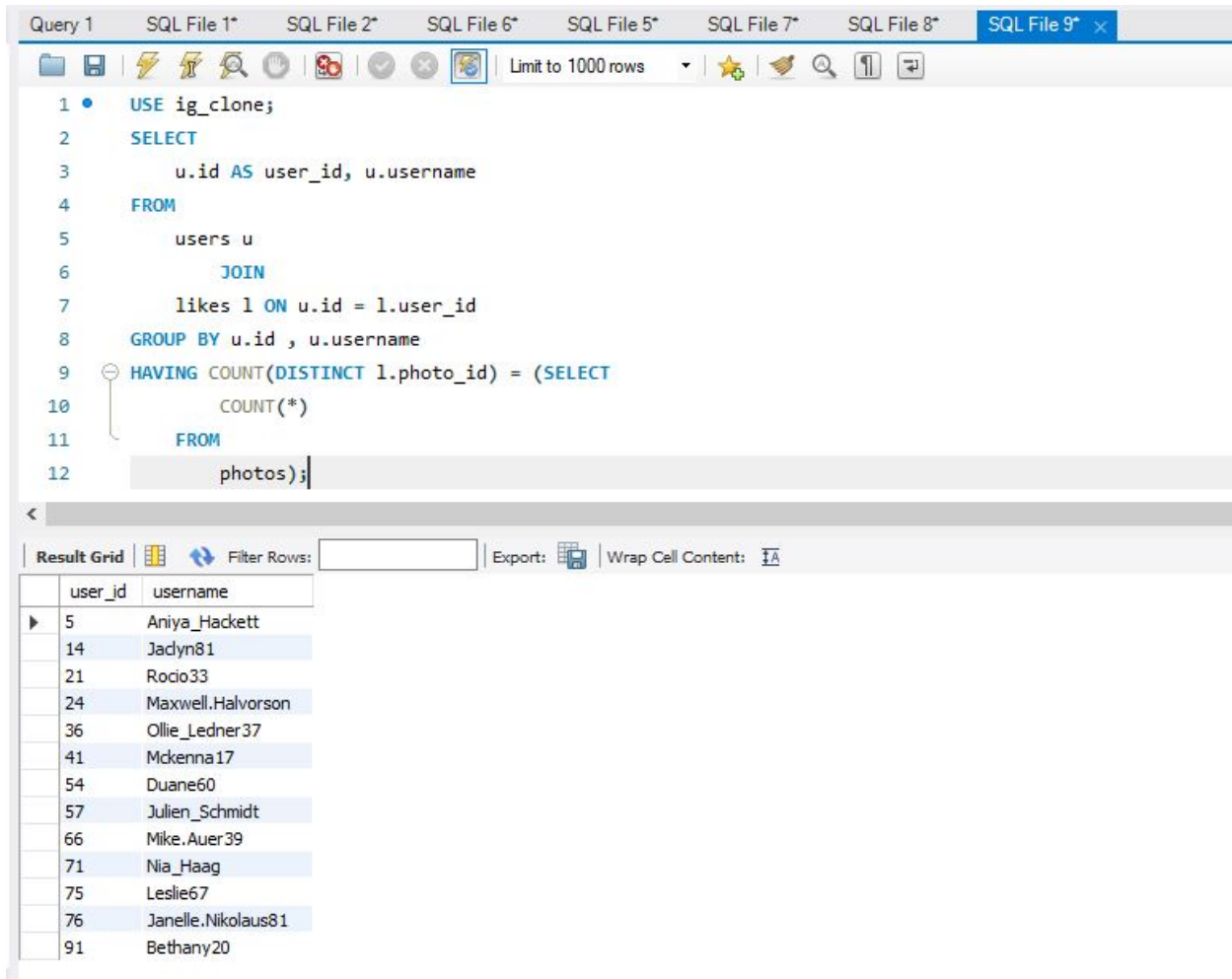
The task was of 2 parts- 1) Calculate the average post per user on Instagram and 2) Give the total number of photos by the total number of users.

For finding out average number of posts per user, we need to LEFT JOIN users and photos table on the user id column. This will provide us the username and with the help of COUNT clause, we can get the amount of photos posted by that id. Later we GROUP BY our output in according to the id.

To solve an arithmetic type of query, we simply need to find total numeric value of required column, in this case total number of photos in the platform which we have found out with the help of COUNT(\*) clause. Similarly we find out total count of users with the COUNT(\*) clause. We need to find out the Average number of photos per user. For this , we can simply divide COUNT(\*) of photos with the COUNT(\*) of users.

Following is the result we obtained- The average number of posts per User can be found out in the drive link provided and the average number of photos per user comes up to 2.5700.

Task2: Bots and Fake account- Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.



```
1 • USE ig_clone;
2 SELECT
3     u.id AS user_id, u.username
4 FROM
5     users u
6 JOIN
7     likes l ON u.id = l.user_id
8 GROUP BY u.id , u.username
9 HAVING COUNT(DISTINCT l.photo_id) = (SELECT
10     COUNT(*)
11 FROM
12     photos);
```

Result Grid

	user_id	username
▶	5	Aniya_Hackett
	14	Jadyn81
	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

Fig 3.3: Fake account detection query and result

To find out users who are made up or have simply liked every photo on Instagram we need to use the JOIN clause on likes and users table. Basically, we will be finding out who HAVE liked every photo id which we have and will cross check this with their user id. A combination of JOIN and HAVING clause comes into play here. 3 attributes of our database are being used here, namely users, likes and photos. We JOIN likes with user based on user id, then GROUP them according to users

id. We filter the result with the help of HAVING clause, here we COUNT the DISTINCT amount of photos liked by user and cross check it with photos available in the app. If the query results true then we can declare such an account as fake account.

The accounts which can be fake/bot are-

Aniya\_Hackett

Jaclyn81

Rocio33

Maxwell.Halvorson

Ollie\_Ledner37

Mckenna17

Duane60

Julien\_Schmidt

Mike.Auer39

Nia\_Haag

Leslie67

Janelle.Nikolaus81

Bethany20



## Results

The following conclusion is what I have come to.

Instagram should reward these 5 users for their valuable time on the app:

Darby\_Herzog

Emilio\_Bernier52

Elenor88

Nicole71

Jordyn.Jacobson2

Instagram can send these users an email or push a notification to remind them to post:

Aniya\_Hackett, Kasandra\_Homenick, Jaclyn81, Rocio33, Maxwell.Halvorson, Tierra.Trantow, Pearl7, Ollie\_Ledner37, Mckenna17, David.Osinski47, Morgan.Kassulke, Linnea59, Duane60, Julien\_Schmidt, Mike.Auer39, Franco\_Keebler64, Nia\_Haag, Hulda.Macejkovic, Leslie67, Janelle.Nikolaus81, Darby\_Herzog, Esther.Zulauf61, Bartholome.Bernhard, Jessyca\_West, Esmeralda.Mraz57, Bethany20

The most liked photo contest winner was:

Photo\_id 145, posted by Zack\_Kemmer93 and the amount of likes on the photo was 48.

The top 5 most used hashtag of all time are:

Smile (59), beach(42), party(39), fun(38), concert(24)

The best time to launch an Ad campaign is Thursday and Sunday.

Investor should know that User engagement is of the following numbers-

The average number of photos per user comes up to 2.5700 and the users who have uploaded amount of photos on Instagram is shared in a CSV file in the drive link shared.

The accounts with suspicious activity of being a bot are as follows:

Aniya\_Hackett

Jaclyn81

Rocio33

Maxwell.Halvorson

Ollie\_Ledner37

Mckenna17

Duane60

Julien\_Schmidt

Mike.Auer39

Nia\_Haag

Leslie67

Janelle.Nikolaus81

Bethany20

## Drive link

[https://drive.google.com/drive/folders/1et906IL1dICtm2Gu5py67Bw6aJdG2uyS?usp=drive\\_link](https://drive.google.com/drive/folders/1et906IL1dICtm2Gu5py67Bw6aJdG2uyS?usp=drive_link)