Project Name**: Instagram User Analytics**

A] Marketing Analysis

1. Loyal User Reward:

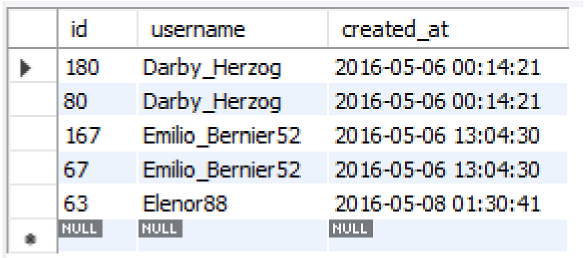
Task: Identify the five oldest users on Instagram from the provided database.

Solution:

Query:

```
select * from ig_clone.users order by created_at asc limit 5;
```

Query Result:



id	username	created_at
180	Darby_Herzog	2016-05-06 00:14:21
80	Darby_Herzog	2016-05-06 00:14:21
167	Emilio_Bernier52	2016-05-06 13:04:30
67	Emilio_Bernier52	2016-05-06 13:04:30
63	Elenor88	2016-05-08 01:30:41
NULL	NULL	NULL

The usernames and creation dates of the five users who have been on the site the longest will be chosen by this query and arranged in ascending order based on their creation dates.

2. Inactive User Engagement

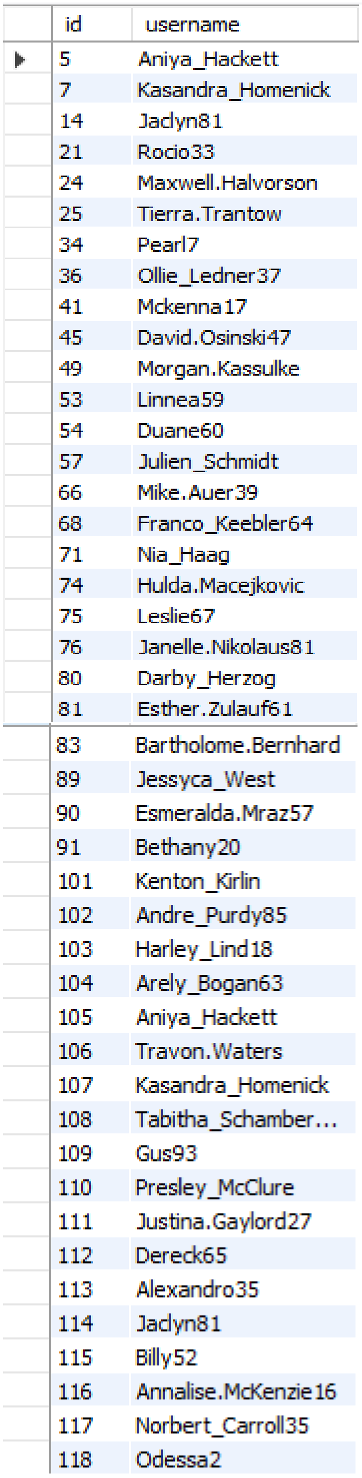
Task: Identify users who have never posted a single photo on Instagram.

Solution:

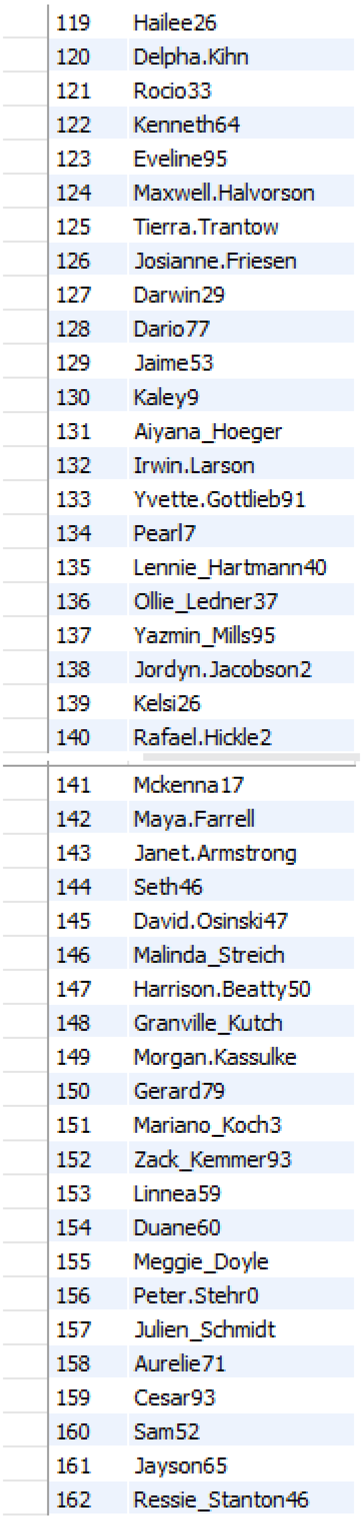
Query:

```
SELECT id, username FROM users WHERE id NOT IN (SELECT DISTINCT user_id FROM photos);
```

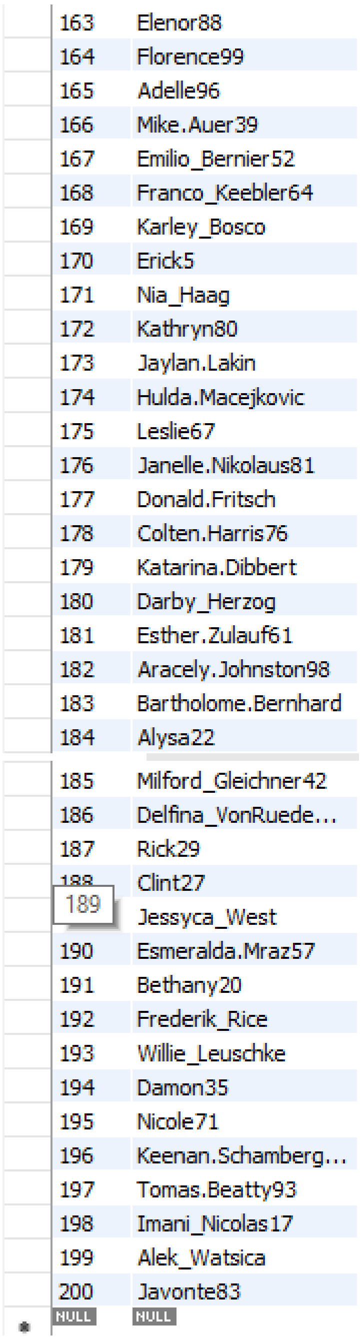
Query Result:



id	username
5	Aniya_Hackett
7	Kasandra_Homenick
14	Jaclyn81
21	Rocio33
24	Maxwell.Halvorson
25	Tierra.Trantow
34	Pearl7
36	Ollie_Ledner37
41	Mckenna17
45	David.Osinski47
49	Morgan.Kassulke
53	Linnea59
54	Duane60
57	Julien_Schmidt
66	Mike.Auer39
68	Franco_Keebler64
71	Nia_Haag
74	Hulda.Macejkovic
75	Leslie67
76	Janelle.Nikolaus81
80	Darby_Herzog
81	Esther.Zulauf61
83	Bartholome.Bernhard
89	Jessyca_West
90	Esmeralda.Mraz57
91	Bethany20
101	Kenton_Kirlin
102	Andre_Purdy85
103	Harley_Lind18
104	Arely_Bogan63
105	Aniya_Hackett
106	Travon.Waters
107	Kasandra_Homenick
108	Tabitha_Schamber...
109	Gus93
110	Presley_McClure
111	Justina.Gaylord27
112	Dereck65
113	Alexandro35
114	Jaclyn81
115	Billy52
116	Annalise.McKenzie16
117	Norbert_Carroll35
118	Odessa2



119	Hailee26
120	Delpha.Kihn
121	Rocio33
122	Kenneth64
123	Eveline95
124	Maxwell.Halvorson
125	Tierra.Trantow
126	Josianne.Friesen
127	Darwin29
128	Dario77
129	Jaime53
130	Kaley9
131	Aiyana_Hoeger
132	Irwin.Larson
133	Yvette.Gottlieb91
134	Pearl7
135	Lennie_Hartmann40
136	Ollie_Ledner37
137	Yazmin_Mills95
138	Jordyn.Jacobson2
139	Kelsi26
140	Rafael.Hickle2
141	Mckenna17
142	Maya.Farrell
143	Janet.Armstrong
144	Seth46
145	David.Osinski47
146	Malinda_Streich
147	Harrison.Beatty50
148	Granville_Kutch
149	Morgan.Kassulke
150	Gerard79
151	Mariano_Koch3
152	Zack_Kemmer93
153	Linnea59
154	Duane60
155	Meggie_Doyle
156	Peter.Stehr0
157	Julien_Schmidt
158	Aurelie71
159	Cesar93
160	Sam52
161	Jayson65
162	Ressie_Stanton46



163	Elenor88
164	Florence99
165	Adelle96
166	Mike.Auer39
167	Emilio_Bernier52
168	Franco_Keebler64
169	Karley_Bosco
170	Erick5
171	Nia_Haag
172	Kathryn80
173	Jaylan.Lakin
174	Hulda.Macejkovic
175	Leslie67
176	Janelle.Nikolaus81
177	Donald.Fritsch
178	Colten.Harris76
179	Katarina.Dibbert
180	Darby_Herzog
181	Esther.Zulauf61
182	Aracely.Johnston98
183	Bartholome.Bernhard
184	Alysa22
185	Milford_Gleichner42
186	Delfina_VonRuede...
187	Rick29
188	Clint27
189	Jessyca_West
190	Esmeralda.Mraz57
191	Bethany20
192	Frederik_Rice
193	Willie_Leuschke
194	Damon35
195	Nicole71
196	Keenan.Schamberg...
197	Tomas.Beatty93
198	Imani_Nicolas17
199	Alek_Watsica
200	Javonte83
NULL	NULL

The users from the "users" table whose IDs and usernames do not have corresponding entries in the "photos" table a sign that they have never uploaded a photo to Instagram are chosen by this query.

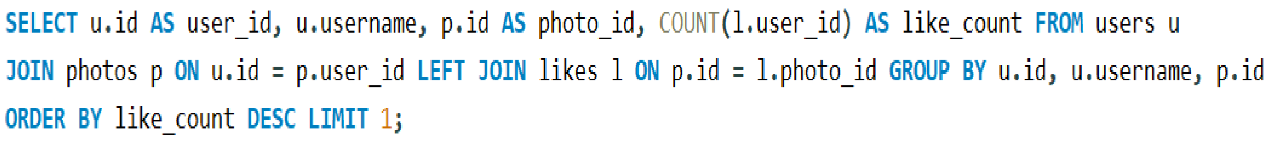
To get the list of inactive users who have never posted any images, run this SQL query on your database. After that, you can utilize this data to send them emails with promotions to get them to start posting.

3. Contest Winner Declaration

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

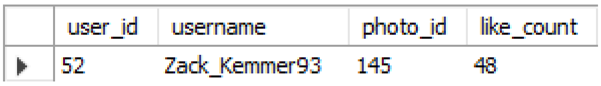
Solution:

Query:



```
SELECT u.id AS user_id, u.username, p.id AS photo_id, COUNT(l.user_id) AS like_count FROM users u
JOIN photos p ON u.id = p.user_id LEFT JOIN likes l ON p.id = l.photo_id GROUP BY u.id, u.username, p.id
ORDER BY like_count DESC LIMIT 1;
```

Query Result:



user_id	username	photo_id	like_count
52	Zack_Kemmer93	145	48

The "users", "photos", and "likes" tables are joined in this query to determine the number of likes for each user's photo. The user with the most likes on a particular photo is represented by the top row, which is selected after the result determined by the like count is sorted in descending order.

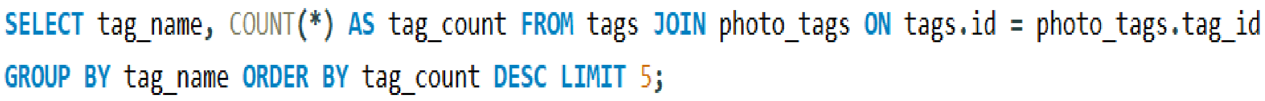
Run this SQL query over your database to find the contest winner and give the team their information.

4. Hashtag Research

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

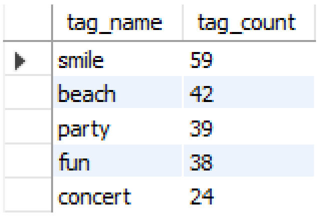
Solution:

Query:



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

Query Result:



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

To count the likes for each photo, this query first joins the "photos" and "users" tables with a left join to the "likes" table. The results are grouped by user and photo, and then they are arranged in descending order of like count. The person with the most likes on a particular photo is represented by the top row, which is the final selection.

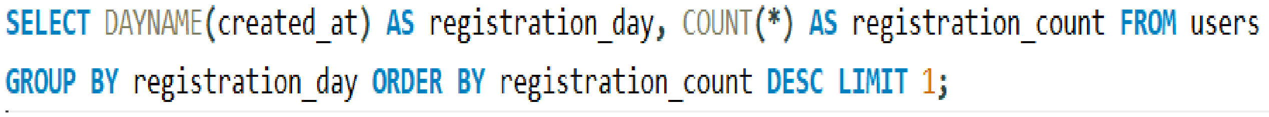
Run this SQL query over your database to find the contest winner and give the team their information.

5. Ad Campaign Launch

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

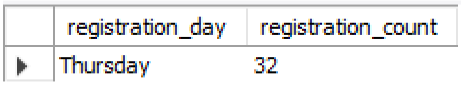
Solution:

Query:



```
SELECT DAYNAME(created_at) AS registration_day, COUNT(*) AS registration_count FROM users
GROUP BY registration_day ORDER BY registration_count DESC LIMIT 1;
```

Query Result:



	registration_day	registration_count
▶	Thursday	32

This query counts the number of registrations for each day and extracts the day of the week (using the DAYNAME function) from the created_at column of the user's table. The results are then grouped by day of the week and arranged in descending order of registration count. Lastly, it restricts the output to the day with the most registrations on top.

Run this SQL query on your database to find out which day of the week the majority of Instagram users sign up. With this information, the team can better plan when to run their advertising campaign to get the most exposure and interaction.

B] Investor Metrics

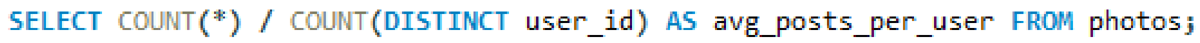
1. User Engagement

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.

Solution:

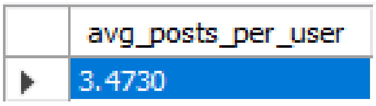
Average number of posts per user:

Query:



```
SELECT COUNT(*) / COUNT(DISTINCT user_id) AS avg_posts_per_user FROM photos;
```

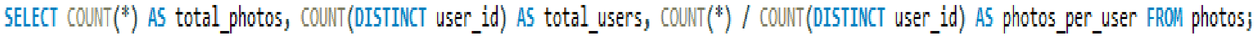
Query Result:



	avg_posts_per_user
▶	3.4730

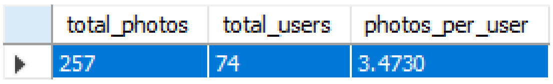
The average number of postings per user is calculated by taking the total number of photos and dividing it by the total number of unique user IDs.
Total number of photos divided by the total number of users:

Query:



```
SELECT COUNT(*) AS total_photos, COUNT(DISTINCT user_id) AS total_users, COUNT(*) / COUNT(DISTINCT user_id) AS photos_per_user FROM photos;
```

Query Result:



total_photos	total_users	photos_per_user
257	74	3.4730

To find the average number of photographs per user, this query counts the total number of photos and different user IDs. It then divides the total number of photos by the total number of users.

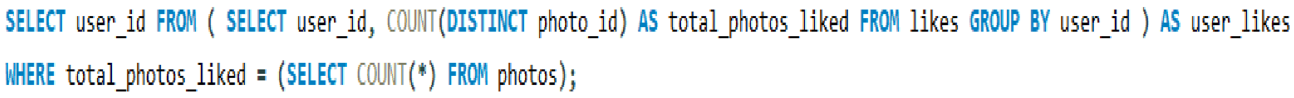
Run these SQL queries via your database to get information about Instagram user engagement. Investors can use these indicators to determine whether users are still using the site and making regular posts.

2. Bots & Fake 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.

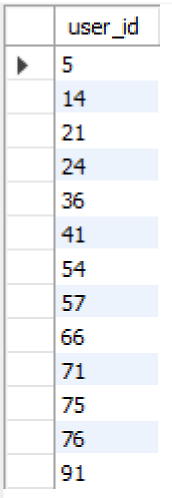
Solution:

Query:



```
SELECT user_id FROM ( SELECT user_id, COUNT(DISTINCT photo_id) AS total_photos_liked FROM likes GROUP BY user_id ) AS user_likes
WHERE total_photos_liked = (SELECT COUNT(*) FROM photos);
```

Query Result:



user_id
5
14
21
24
36
41
54
57
66
71
75
76
91

The number of unique photographs that each user has liked is first determined by this query, which then chooses user IDs that have liked the same number of unique photos as there are overall on the website.

Run your database through this SQL query to find any possible bots or fake accounts that have liked every single picture on the website. In order to ascertain whether these users are in fact bots or fraudulent accounts, more research into them may be necessary.

<u>Note: I used the same database and all the table which are provided into the attachment.</u>

1. Project Description:

The project's main objective is to get insights for investment metrics and marketing tactics by analyzing data from a platform that resembles Instagram. It entails doing things like finding loyal followers, interacting with inactive people, announcing contest winners, looking for hashtags, and spotting possible bots or fraudulent accounts. The purpose of these assignments is to offer insightful data about platform engagement, user behavior, and authenticity.

2. Approach:

Database Creation: Using SQL commands, the project started with the creation of a MySQL database structure. Like an Instagram-like platform, the schema has tables for users, photos, comments, likes, follows, tags, and photo_tags.

Data Analysis: To accomplish the tasks listed in the project description, pertinent data was extracted from the database using SQL queries, and various analyses were carried out. These inquiries included metrics computation, data aggregation, and record filtering according to predetermined standards.

3. Tech-stack used

MySQL was used as the database management system for the project. SQL queries were run with the aid of programs like phpMyAdmin and MySQL Workbench. MySQL was selected due to its robustness, ease of use, and broad use in database administration. I used the MYSQL Workbench 8.0 CE version. Because this version is more suitable to my personal computer.

4. Insights

Marketing Analysis: To reward loyalty, the platform's five oldest users were identified. Identified inactive individuals for targeted engagement who have never uploaded a single photo. Found the winner of the contest whose photo had the most likes. Recommended the top five most popular hashtags to maximize reach. Based on patterns in user registration, identified the optimal day of the week to start advertising campaigns.

Investor Metrics: To determine user engagement, the average number of posts made by each user was calculated. Based on unusual like behavior, suspected bots or false accounts were identified.

5. Result

1. Marketing teams and investors received practical insights through the analysis of user data and engagement metrics.
2. The project outlined potential concerns related to fraudulent accounts and identified areas where user interaction techniques could be improved.
3. These insights aid in the development of well-informed decisions, which results in better marketing initiatives and an improved comprehension of platform dynamics.

6. Drive Link

1. After the project was finished, investors and marketing teams had access to actionable data and metrics that contribute that helped them make wise decisions.

2. The project enhanced customer satisfaction and platform performance by optimizing marketing strategies, user engagement initiatives, and ad campaign scheduling through the resolution of diverse marketing analysis tasks.
3. Investor trust was increased and investment decisions were supported by the useful information that investor metrics provide for evaluating the platform's growth, user engagement, and authenticity.
4. By identifying possible bots or false accounts, the platform's legitimacy and integrity were preserved, protecting user confidence and long-term viability.

In general, users and stakeholders together gain from the project's insights and analysis, which help the Instagram-like platform succeed and continue to be improved.