Instagram is a popular social media platform that allows users to share photos and videos with their followers. It provides a visually appealing interface with a range of creative filters and editing tools to enhance content. Users can follow accounts, like and comment on posts, and send direct messages. Instagram also features stories, a temporary photo and video sharing feature, and IGTV for longer videos. It is widely used by individuals, celebrities, and businesses to showcase their lives, talents, and products, making it a vibrant and engaging online community for visual storytelling and digital connections.

As we proceed above, I created a database called ig_clone. The database consists of 7 tables –

1. USER TABLE
2. PHOTOS TABLE
3. COMMENTS TABLE
4. LIKES TABLE
5. FOLLOWS TABLE
6. TAGS TABLE
7. PHOTOS-TAGS TABLE

The user table consists of 3 attributes, id being the primary. The photos table has its own primary key by the name ID. Different tables were connected to it with the help of Foreign Key such as likes, comments, photo-tags and tags. There is a table called follows, which stores the followers and the following ID.

Approach towards the project was quite simple, I cloned the project and executed the commands as given in the dataset. Later I wrote all the queries needed to find out the solutions for the asked questions.

Tech stack used was MySQL version 8.0.3.2 and later installed MySQL Workbench 8.0.3.2. It provides a visual console to easily administer the MySQL environments and gain better visibility of the databases.

DETAILED REPORT

1. MARKETING – The team wants to launch some campaigns, and they need some help with the following questions –
   1.1. Rewarding the most loyal users: It shows the people who have been using the platform for the longest time.

   Task – Find the 5 oldest users of Instagram from the database provided by the team.

   Query – SELECT * from users ORDER BY created_at LIMIT 5;

   Select – tells the database that you want to select data

   FROM – User tells the database to select the data from the user table

   (*) tells the database that you want to see all the columns in the table.

   ORDER BY – With this expression, simply specify a column on which the data will be sorted.

LIMIT n – Returns the first n rows from the result. This is much more efficient than returning all the data from the database.

RESULT –

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
NULL	NULL	NULL

2. Inactive users on Instagram
   Task – Find the users who have never posted a single photo on Instagram.
   QUERY – Select username FROM users LEFT JOIN photos on users.id = photos.user_id WHERE photos.id is NULL;
   LEFT JOIN works in the following way: it returns all rows from the left table (the first table in the query) plus all matching rows from the right table (the second table in the query).
   RESULT –

username
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

3. Declaring Contest Winner:
   Task: Identify the winner of the contest and provide their details to the team. To do this task, we need to find the most popular photo with most likes and user who created it.
   QUERY - **select username, photos.id, photos.image_url, count(*) as total from photos**
   inner join likes on likes.photo_id = photos.id
   inner join users on photos.user_id = users.id
   group by photos.id order by total desc limit 1;

INNER JOIN (also known as JOIN) only joins those rows from the two tables where there is a match between the columns.

RESULT –

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

Result shows the user photo is most liked on Instagram.

4. HASHTAG RESEARCHING: Hashtag helps the user to reach a wide range of people. It is used to draw attention, organize, promote and connect.

   TASK – To identify the top 5 commonly used hastags on Instagram.

   QUERY - select tags.tag_name,
   count(*) as total
   from photo_tags
   join tags
   on photo_tags.tag_id=tags.id
   group by tags.id
   order by total desc
   limit 5;

   COUNT() – function returns the no. of rows that matches the specific criteria.

   AS – new keyword AS and we put the new name after it (person_id). We can repeat the process with every column.

   AS – The new name is just an alias, which means its temporary and doesn't change the column name in the database. It only influences the way the column is shown in the result of the specific query. This technique is often used when there are a few columns with the same name coming from different tables.

   Normally when SQL displays column in the result, there is no information about the table that a specific column is part of.

   RESULT –

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

   Query shows the most used hashtags by the users on Instagram.

5. **LAUNCH AD CAMPAIGN**

   TASK – To find out the day of week when most users register on Instagram.

   QUERY - select dayname(created_at) as Day,
   count(*) as total
   from users
   group by day
   order by total desc
   limit 2;

   RESULT –

Day	total
Thursday	16
Sunday	16

The query result shows two days of the week when the user register mostly. According to me, the most suitable day to launch the ad campaign would be Sunday because most users mostly are free on Sundays and hence, more interaction would take place on Sunday.

B. INVESTOR METRICS

6. USER ENGAGEMENT – investors wants to know that Instagram is not becoming redundant like Facebook, so they want to check the frequency of how much the users are engaging on the platform.

TASK – To provide how many times an average user posts on Instagram.

QUERY - select(select count(*)

from photos) / (select count(*) from users) as avg;

RESULT –

avg
2.5700

BOTS & FAKE ACCOUNTS – It is reported that there are lots of bots and fake accounts on the platform. The investors want to know if there are fake and dummy accounts.

TASK – to provide data on users(bots) who have liked every single photo on the site(normal user would not be able to do this)

QUERY – select username, count(*) as num_likes from users

inner join likes on users.id = likes.user_id

group by likes.user_id

having num_likes = (select count(*) from photos);

RESULT –

username	num_likes
Aniya_Hackett	257
Jaclyn81	257
Rocio33	257
Maxwell.Halvorson	257
Ollie_Ledner37	257
Mckenna17	257
Duane60	257
Julien_Schmidt	257
Mike.Auer39	257
Nia_Haag	257
Leslie67	257
Janelle.Nikolaus81	257
Bethany20	257