

- **Project Description:**

- In this project , we are supposed to provide a detailed report for marketing and investment metrics department. This analysis will help them make a decision based on different metrics and insights.

- **Approach:**

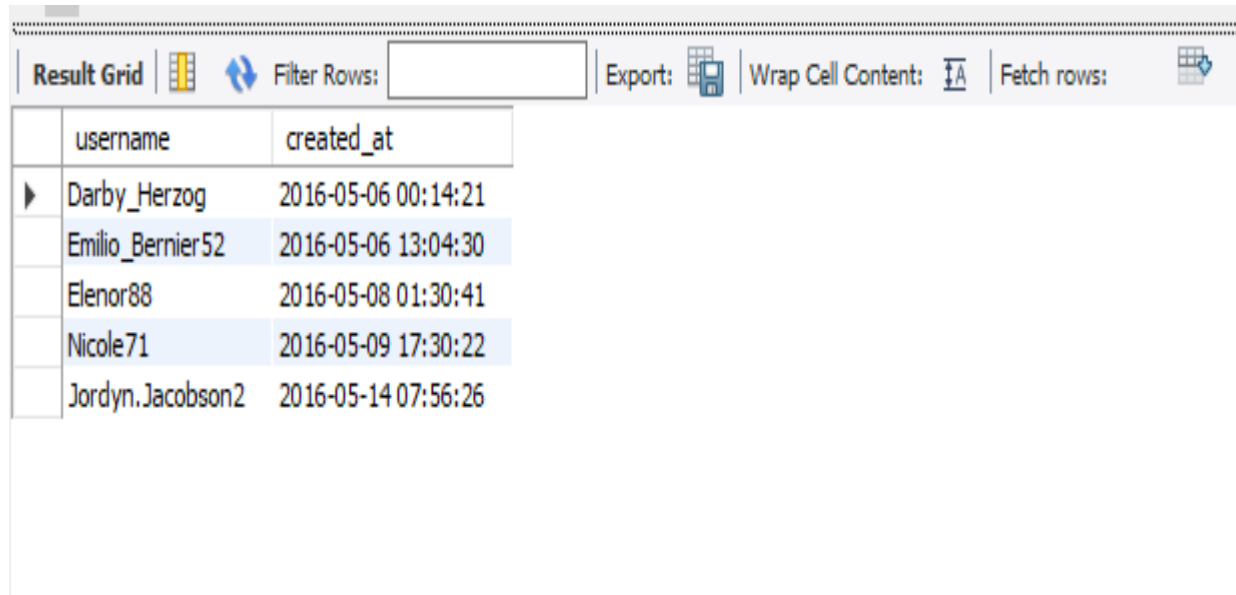
- For this project, I have used My SQL to extract the required data from the given database using the Join function, subqueries, Aggregation, where condition, Group by, Distinct and other functions required. keeping the Primary key and foreign key in consideration provided all the reports asked by the marketing department and Investor metrics department.

- **Tech-Stack Used:**
- MySQL Server: Version 8.0 . MySQL Workbench: Version 8.0CE . I chose MySQL Server as the database management system for the project due to many reasons :1) MySQL is an open-source relational database management system, providing a cost-effective solution for the project.2) Community Support: MySQL has a large and active community of developers and users, tutorials, and forums for support and troubleshooting.
- **Result:**
- This project helped me in understanding importance of data analysis for organizations in making data-driven decisions. In this project based on data from Instagram, I was able to get insights about various questions like which users have been using the platform for the longest, which users are inactive in the platform, which hashtags can be used for promotional contents for maximum reach, how many fake/bot accounts are present, whether the platform is growing or became stagnant in its growth etc.

- **Insights:**
- **A) MARKETING:** The marketing team wants to launch some campaigns, and they need help with the following:
 - **Loyal User Reward:** People who have been using the platform for the longest time.
- **Task:** Find the 5 oldest users of the Instagram from the database provided.
- **Query:**

```
SELECT
    username, created_at
FROM
    users
ORDER BY created_at ASC
LIMIT 5;
```

- **Result:**



The screenshot shows a web-based database interface. At the top, there is a toolbar with the following elements: a 'Result Grid' tab, a grid icon, a 'Filter Rows:' button with a dropdown arrow, an 'Export:' button with a document icon, a 'Wrap Cell Content:' button with a text icon, and a 'Fetch rows:' button with a refresh icon. Below the toolbar is a table with two columns: 'username' and 'created_at'. The table contains five rows of data, each representing a user and their registration timestamp. The first row is highlighted with a blue background.

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

- **Insights:**
- The 5 oldest users in the current database we have are **darby_herzog, Emilo_Berrier52, Elenor88, Ncole71, Jordyn.Jacobson2.**
- We can see that they people are registered their account in may month of 2016.

- **Inactive User Engagement:** By sending them promotional emails to post their 1st photo.
- **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.user_id IS NULL;
```

- **Result:**

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	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			

Result 5 x

Darby_Herzog
Esther.Zulauf61
Bartholome.Bernhard
Jessyca_West
Esmeralda.Mraz57
Bethany20

Result 5 x

- **Insights:**
- The above column username shows all the users who have never posted a single photo in Instagram till the time this dataset was recorded.
- **Contest Winner Declaration:** The team started a contest and the user who gets the most likes on a single photo will win the contest now they wish to declare the winner.
- **Task:** Identify the winner of the contest and provide their details to the team.
- **Query:**

```
WITH base AS
(
SELECT
    likes.photo_id,
    users.username,
    COUNT(likes.user_id) AS Likes,
    users.id AS user_id,
    photos.image_url AS Image_url
FROM
    likes
        INNER JOIN
    photos ON likes.photo_id = photos.id
        INNER JOIN
    users ON photos.user_id = users.id
GROUP BY likes.photo_id , users.username
ORDER BY Likes DESC
LIMIT 1
)
SELECT
    user_id, username, Image_url, Likes
FROM
    base;
```

- **Result:**

	user_id	username	Image_url	Likes
▶	52	Zack_Kemmer93	https://jarret.name	48

- **Insights:**
- The above result shows that User with username **Zack_Kemmer93** has posted a Photo and he has got the most likes (**48**).
- **Hashtag Research:** A partner brand wants to know, which hashtags to use in the post to reach the most people on the platform.
- **Task:** Identify and suggest the top 5 most commonly used hashtags on the platform.
- **Query:**

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

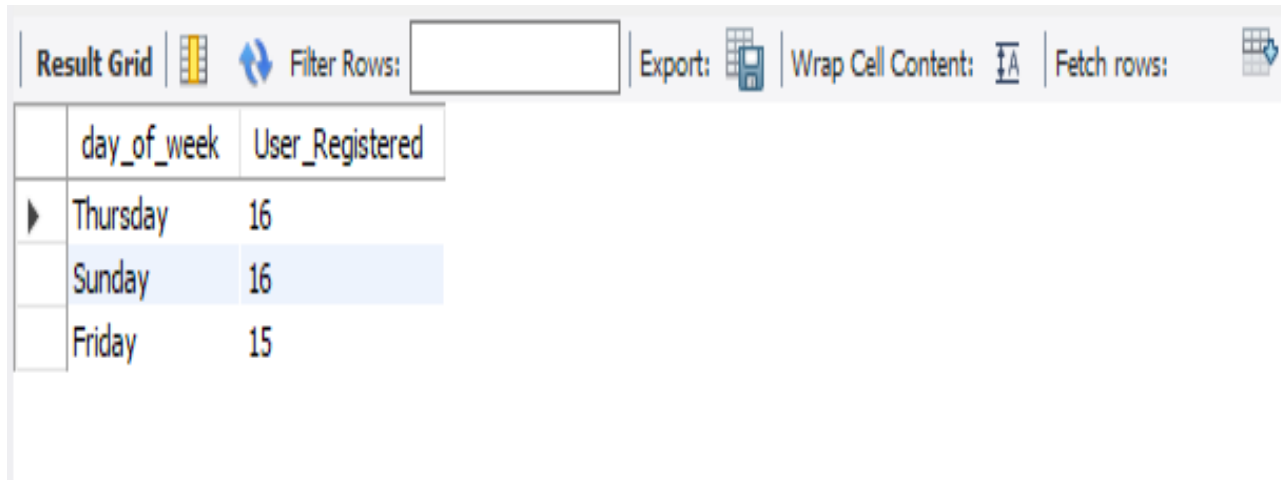
- **Result:**

Result Grid			Filter Rows:	Export:	Wrap Cell Content:	Fetch rows:
	tag_name	total_tags				
▶	smile	59				
	beach	42				
	party	39				
	fun	38				
	concert	24				

- **Insights:**
- From the above result, we can observe that most commonly used hashtags are **smile, beach, party, fun** and **concert**.
- **Launch AD Campaign:** The team wants to know, which day would be the best day to launch ADs.
- **Task:** What day of the week do most users register on? Provide insights on when to schedule an ad campaign.
- **Query:**

```
SELECT
    DAYNAME(created_at) AS day_of_week,
    COUNT(id) AS User_Registered
FROM
    users
GROUP BY day_of_week
ORDER BY User_Registered DESC
LIMIT 3;
```

- **Result:**



The screenshot shows a software interface for data analysis. At the top, there is a toolbar with options: 'Result Grid' (with a grid icon), 'Filter Rows:' (with a funnel icon and an empty text box), 'Export:' (with a save icon), 'Wrap Cell Content:' (with a text wrap icon), and 'Fetch rows:' (with a refresh icon). Below the toolbar is a table with two columns: 'day_of_week' and 'User_Registered'. The table contains three rows of data: Thursday (16), Sunday (16), and Friday (15). The 'Sunday' row is highlighted with a light blue background.

	day_of_week	User_Registered
▶	Thursday	16
	Sunday	16
	Friday	15

- **Insights:**
- From the above data frame, we can observe that most users registered on **Thursday** , **Sunday** and **Friday**.

- **B) INVESTOR METRICS:** Our investors want to know if Instagram is performing well, they want to assess the app on the following grounds:
 - **User Engagement:** Are users still as active and post on Instagram or they are making fewer posts.
 - **Task:** Provide how many times does average user posts on Instagram. Also, provide the total number of photos on Instagram /total number of users.
 - **Query:**

```
SELECT
    AVG(posts_count) AS avg_posts_per_user
FROM
    (SELECT
        user_id, COUNT(*) AS posts_count
    FROM
        photos
    GROUP BY user_id
    ORDER BY posts_count DESC) AS user_posts;
```

- **Result:**

Result Grid			Filter Rows: <input type="text"/>	Export:	Wrap Cell Content:
	avg_posts_per_user				
▶	3.4730				

- total number of photos on Instagram divided by the total number of users.

- **Query:**

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

- **Result:**

	average
▶	3.4730

- **Insights:**
- Thus an **average user posts per user id 3.4730** in Instagram which is the same as the **ratio between number of photos posted in Instagram and total number of users in Instagram.**
- **Bots & Fake Accounts:** The investors want to know if the platform is crowded with fake and dummy accounts.
- **Task:** Provide data on users (bots) who have liked every single photo on the site (since any normal user would not be able to do this).

- Query:

```
SELECT
    user_id, username
FROM
    likes
    INNER JOIN
        users ON likes.user_id = users.id
GROUP BY user_id
HAVING COUNT(photo_id) = (SELECT
    COUNT(*)
FROM
    photos);
```

- Result:

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	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			

- **Insights:**
- The above result shows the list of Users who have liked all photos in the database. These Users can be identified as **Bots**.
- There are **13** number of such Users which account for **13%** of total Users.