Instagram User Analytics

SQL Fundamentals

PROJECT DESCRIPTION

As the project name is Instagram User Analytics that means simply analyzing the user data that is related to instagram application which contains the users information about the usage of instagram , account creation, posts, tags, etc.And send it to the marketing, product and development team to improve the user experience and help business grow.

For implementation of this project , I am going to use DML commands which includes aggregate functions, sorting functions,joins.

This project will be helpful to find out how should the instagram marketing team to launch the campaigns and to increase the investors metrics.

APPROACH

Instagram User Analytics:I thought about the user activities done that are likes, posts account opening , tags used ,etc.

Approach for this project is straightforward:

1. Created a database as per the provided resources in MySQL.
2. Read the question asked and started drafting on, which query can be used.
3. Input the query in MySQL and run that to get output for question asked.
4. The approach is quite simple as it contains the use of sql fundamentals that have been learnt during the session and used for completing this project.

TECH-STACK USED

For implementation of this project I have used MySQL Command Line Client , Version:8.0.27

Purpose of using MySQL is that it is already installed and I know how to use it.Also MySQL is mostly used to write and implement the sql queries.It is easy to use if installed once as compared to suggested installations.

INSIGHTS

While making this project I gained a knowledge about the queries to be used when any question comes related to data analytics. Also it gave me an insight that at what type of question which queries can be followed up.

Rewarding Most Loyal Users:

```
mysql> show tables;
+--------------------+
| Tables_in_ig_clone |
+--------------------+
| comments           |
| follows            |
| likes              |
| photo_tags         |
| photos             |
| tags               |
| users              |
+--------------------+
7 rows in set (0.61 sec)

mysql> SELECT * FROM users ORDER BY created_at LIMIT 5;
+----+------------------+---------------------+
| 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 |
+----+------------------+---------------------+
5 rows in set (0.14 sec)
```

```
SELECT *
FROM users
ORDER BY created_at
LIMIT 5 ;
```

Remind Inactive Users to Start Posting:

```
mysql> -- find the users who have never posted a photo
mysql> SELECT username FROM users LEFT JOIN photos ON users.id=photos.user_id WHERE photos.id IS NULL;
+---------------------+
| 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           |
+---------------------+
26 rows in set (0.04 sec)
```

```
SELECT username
FROM users
LEFT JOIN photos
ON users.id=photos.user_id
WHERE photos.id IS NULL;
```

Declaring Contest Winner:

```
mysql> SELECT
    -> username,
    -> photos.id,
    ->  photos.image_url,
    -> count(likes.user_id) 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;
+---------------+-----+---------------------+-------+
| username      | id  | image_url           | total |
+---------------+-----+---------------------+-------+
| Zack_Kemmer93 | 145 | https://jarret.name |    48 |
+---------------+-----+---------------------+-------+
1 row in set (0.23 sec)
```

```
SELECT
        username,
        photos.id,
        photos.image_url,
         count(likes.user_id) 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;
```

Hash Tag Researching:

```
mysql> 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;
+----------+-------+
| tag_name | total |
+----------+-------+
| smile    |    59 |
| beach    |    42 |
| party    |    39 |
| fun      |    38 |
| concert  |    24 |
+----------+-------+
5 rows in set (0.08 sec)
```

```
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;
```

Launch AD campaign:

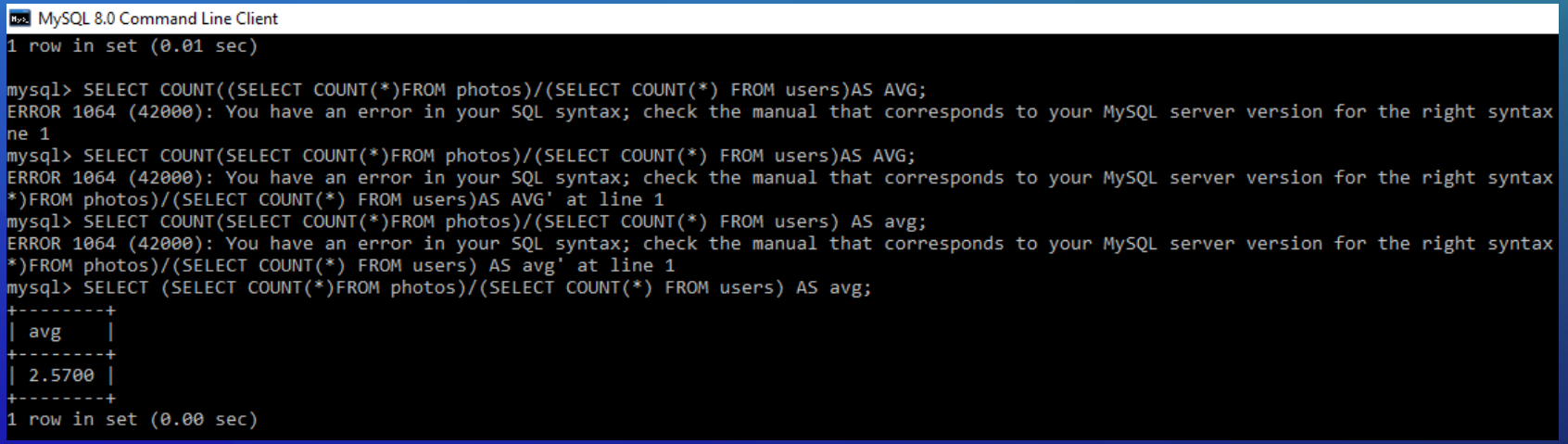
```
mysql> SELECT
    ->     DAYNAME(created_at) AS day,
    ->     count(*) as total
    -> FROM users
    -> GROUP BY day
    -> ORDER BY total DESC;
+-----------+-------+
| day       | total |
+-----------+-------+
| Thursday  |    16 |
| Sunday    |    16 |
| Friday    |    15 |
| Tuesday   |    14 |
| Monday    |    14 |
| Wednesday |    13 |
| Saturday  |    12 |
+-----------+-------+
7 rows in set (0.07 sec)

mysql> SELECT
    ->     DAYNAME(created_at) AS day,
    ->     count(*) as total
    -> FROM users
    -> GROUP BY day
    -> ORDER BY total DESC
    -> LIMIT 2;
+----------+-------+
| day      | total |
+----------+-------+
| Thursday |    16 |
| Sunday   |    16 |
+----------+-------+
```

```
SELECT
 DAYNAME(created_at) AS day,
 COUNT(*) as total
FROM users
GROUP BY day
ORDER BY total DESC;
```

User Engagement:

SELECT (SELECT COUNT(*)FROM photos)/(SELECT COUNT(*) FROM users) AS avg;



Bots and Fake Accounts:

```
mysql> SELECT users.id,username, COUNT(users.id) As total_likes_by_user
    -> FROM users
    -> JOIN likes ON users.id = likes.user_id
    -> GROUP BY users.id
    -> HAVING total_likes_by_user = (SELECT COUNT(*) FROM photos);
+----+--------------------+---------------------+
| id | username           | total_likes_by_user |
+----+--------------------+---------------------+
|  5 | Aniya_Hackett      |                 257 |
| 14 | Jaclyn81           |                 257 |
| 21 | Rocio33            |                 257 |
| 24 | Maxwell.Halvorson  |                 257 |
| 36 | Ollie_Ledner37     |                 257 |
| 41 | Mckenna17          |                 257 |
| 54 | Duane60            |                 257 |
| 57 | Julien_Schmidt     |                 257 |
| 66 | Mike.Auer39        |                 257 |
| 71 | Nia_Haag           |                 257 |
| 75 | Leslie67           |                 257 |
| 76 | Janelle.Nikolaus81 |                 257 |
| 91 | Bethany20          |                 257 |
+----+--------------------+---------------------+
13 rows in set (0.30 sec)
```

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

RESULTS

While making this project I am able get hands on real time project like Instagram User Analysis.Also I came to know how actually this mobile applications like facebook,instagram,etc manage the data and how can I apply the sql to get particular output as per required.

I am also able to apply my learning like sql fundamentals on any real time application and eagar to learn more thing and get an experience by working on real time applications.

Thanks!