PROJECT 2

INSTAGRAM USER ANALYTICS

Project description:

The Instagram user analytics project is to help us put into practice the lessons learnt in module 2 i.e. join, aggregate functions and group by functions.

In this project we learnt to analyse though the data given about Instagram users ad perform various tasks related to marketing and admin side.

Tech stack used: MYSQL v8.0 in db-fiddle.com

Approach: Getting through each task one after; glancing the given data bases and finding relations between the tables.

Insights: we have analysed through the clone data base of 100 Instagram users and found out useful insights into the user behaviour and also about the fake accounts(bots). We found user who are using the platform since 2016 and those who have not posted a single photo which we used to measure an activity. We found that #smile was trending and popular as more and more people used it. We found Zack_Kemmer93 who had a total of 48 likes as the contest winner. We also found that Thursday and Friday are the days where we were registering more and more users compared to the other days; which helped advertisers to pick the perfect day to run their ads campaigns.

Result of this project:

I have learnt the ways in which I can find relationship between data tables and their data. I also learnt how to use different joins and aggregate functions.

I learnt useful lessons on how the big product company use data analytics to improve their product offerings and manage growth.

Marketing :

#TASK A1:REWARDING MOST LOYAL USERS

```
select
username,created_at
from
ig_clone.users
ORDER BY created_at
limit 5
```

#TASK A2:REMIND INACTIVE USERS TO START POSTING

```
select
IGU.username
from
ig_clone.users IGU
left join
ig_clone.photos IGP
on
IGU.id=IGP.user_id
where
IGP.user_id is null
order by
IGU.username;
```

#TASK A3:DECLARING CONTEST WINNER

```
select
users.username,
likes.photo_id,
count(likes.user_id) as user_Likes
from
ig_clone.likes likes
inner join
ig_clone.photos photos
on likes.photo_id=photos.id
inner join
ig_clone.users users
on photos.user_id=users.id
group by likes.photo_id,users.username
order by user_likes desc
limit 1
```

#TASK A4: HASHTAG RESEARCHING

```
select
tag.tag_name as TagName,
count(pt.photo_id) as tagCount
from
ig_clone.photo_tags as pt
inner join
ig_clone.tags as tag
on tag.id=pt.tag_id
group by TagName
order by tagCount desc
limit 5
```

#TASK A5:LAUNCH AD CAMPAIGN

```
select
WEEKDAY(created_at) as day,
count(username) as user_reg
from
ig_clone.users
group by day
order by user_reg desc
```

Investor Metrics:

#TASK B1: USER ENGAGEMENT

```
with CTE as(
select
u.id as userid,
 count(p.id) as photoid
 from
 ig_clone.users u
 left join
 ig_clone.photos p
 on u.id=p.user_id
 group by u.id
)
select
SUM(photoid)  total_photos,
count(userid)  total_users,
sum(photoid)/count(userid) as avgPPU
FROM CTE
where photoid>0;
```

#TASK B2: BOTS & FAKE ACCOUNTS

```
with botfinder as (
 select
 user_id,
 count(photo_id) as total_likes
 from
 ig_clone.likes
 group by
 user_id
 order by total_likes
 )
 select *
 From botfinder
 where
 total_likes=(select count(*) from ig_clone.photos);
```