CS5540 - Principles of Big Data Management

Project Phase-2

Twitter Data Analysis of Cricket Premier Leagues

By

Yagnasri Chowdary Nalluri - 16293464
Akarsha Yedla - 16293464
Pravallika Reddy Lankapothu- 16293464

Objective:

This part of project deals with Analysis of created data and also its visualization based on various parameters. Here we come with the queries as mentioned in the interim report.

Technologies used:

- JetBrains PyCharm 2018.3.5
- Spark
- Scala
- Tableau 2019.4

Queries and Analysis:

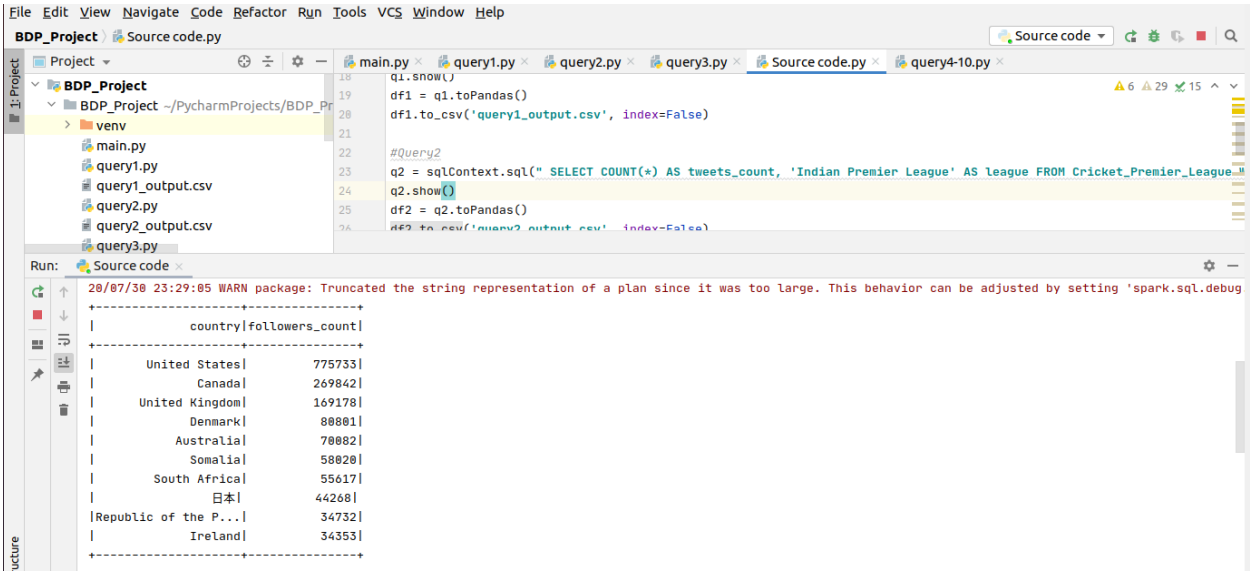
Query 1:

This query collects the top 10 geographical locations from which highest number of tweets had been recorded.

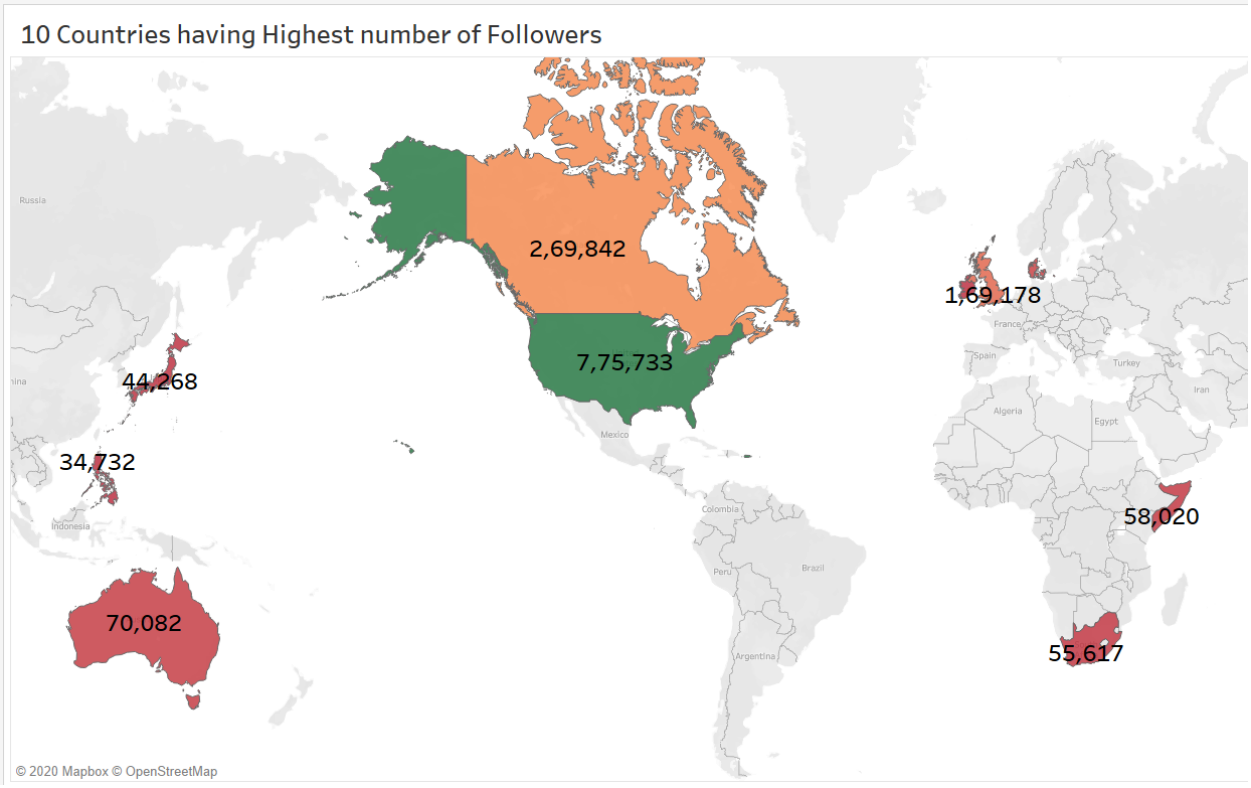
Please find the query below that had been used for extraction

```
q1 = sqlContext.sql("SELECT place.country AS country,
SUM(user.followers_count) AS followers_count FROM Cricket_Premier_League
WHERE place.country != 'null' GROUP BY place.country ORDER BY
followers_count DESC LIMIT 10")
```

Please find the screenshot of the query output.



Please find the screenshot of the data visualization for query 1.



Query 2:

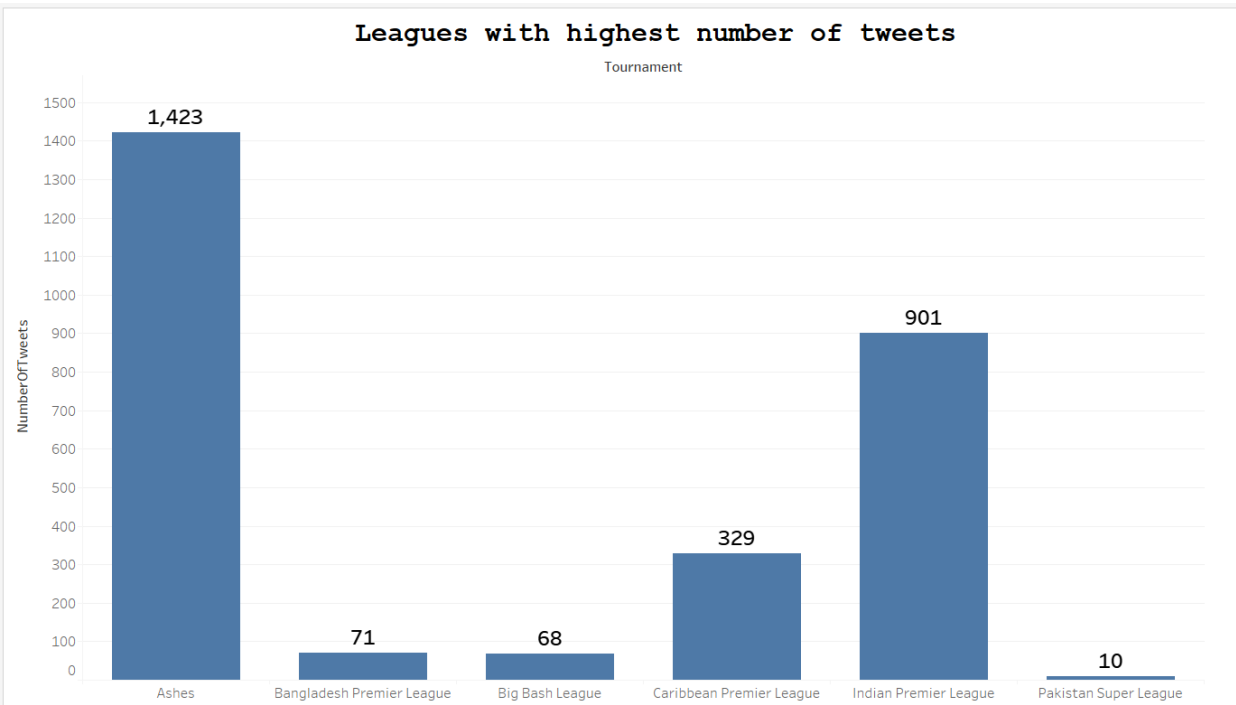
This query is used to collect the number of recorded for the leagues “IPL”, “PSL”, “CPL”, “BPL” and “Ashes” and sort them in the descending order.

Please find the query below that had been used for extraction

```
q2 = sqlContext.sql(" SELECT COUNT(*) AS tweets_count, 'Indian Premier
League' AS league FROM Cricket_Premier_League WHERE TEXT LIKE '%ipl%' OR
TEXT LIKE '%Indian Premier League%' OR  TEXT LIKE '%IPL%' OR TEXT LIKE
'%Ipl%' UNION SELECT COUNT(*) AS tweets_count, 'Pakistan Super League' AS
league FROM Cricket_Premier_League WHERE TEXT LIKE '%psl%' OR TEXT LIKE
'%Pakistan Super League%' OR TEXT LIKE '%PSL%' OR TEXT LIKE '%Psl%' UNION
SELECT COUNT(*) AS tweets_count, 'Caribbean Premier League' AS league FROM
Cricket_Premier_League WHERE TEXT LIKE '%cpl%' OR TEXT LIKE '%Caribbean
Premier League%' OR TEXT LIKE '%CPL%' OR TEXT LIKE '%Cpl%' UNION SELECT
COUNT(*) AS tweets_count, 'Bangladesh Premier League' AS league FROM
Cricket_Premier_League WHERE TEXT LIKE '%bpl%' OR TEXT LIKE '%Bangladesh
Premier League%' OR  TEXT LIKE '%BPL%' OR TEXT LIKE '%Bpl%' UNION SELECT
COUNT(*) AS tweets_count, 'Ashes' AS league FROM Cricket_Premier_League
WHERE TEXT LIKE '%ashes%' OR TEXT LIKE '%Ashes%' OR  TEXT LIKE '%ASHES%'
UNION SELECT COUNT(*) AS tweets_count, 'Big Bash League' AS league FROM
Cricket_Premier_League WHERE TEXT LIKE '%bbl%' OR TEXT LIKE '%Big Bash
League%' OR  TEXT LIKE '%BBL%' OR TEXT LIKE '%Bbl%' OR TEXT LIKE '%big
bash%'")
```

Please find the screenshot of the query output.

Please find the screenshot of the data visualization for query 2.



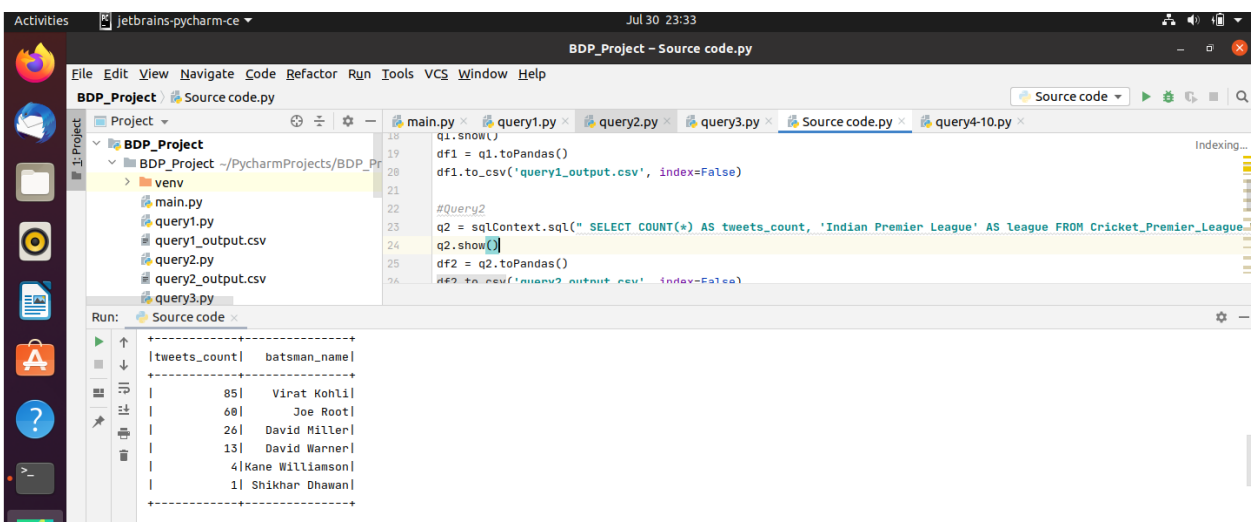
Query 3:

This query is to find the most famous batsman among “Virat Kohli”, “David Warner”, “Kane Williamson”, “Joe Root”, “Shikhar Dhawan” and “David Miller”.

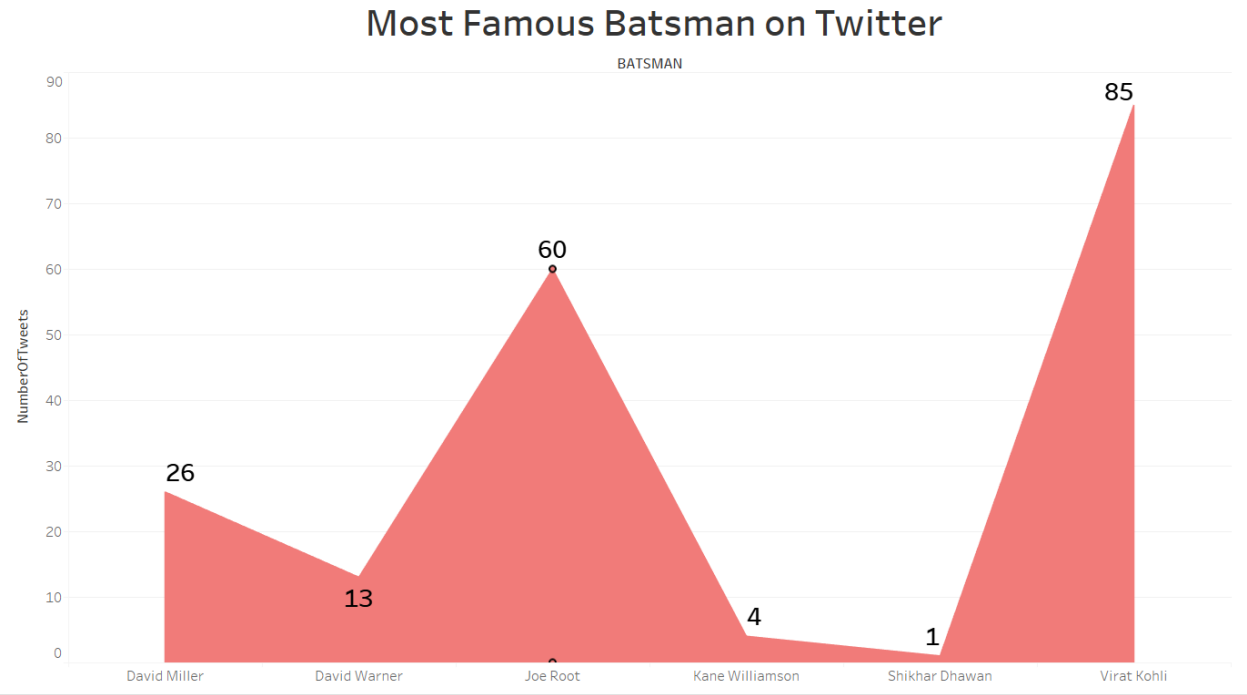
Please find the query below that had been used for extraction

```
q3 = sqlContext.sql(" SELECT COUNT(*) AS tweets_count, 'Virat Kohli' AS
batsman_name FROM Cricket_Premier_League WHERE TEXT LIKE '%virat kohli%'
OR TEXT LIKE '%Virat Kohli%' OR  TEXT LIKE '%virat%' OR TEXT LIKE
'%kohli%' UNION SELECT COUNT(*) AS tweets_count, 'David Warner' AS
batsman_name FROM Cricket_Premier_League WHERE TEXT LIKE '%david warner%'
OR TEXT LIKE '%David Warner%' OR TEXT LIKE '%warner%' OR TEXT LIKE
'%WARNER%' UNION  SELECT COUNT(*) AS tweets_count, 'Kane Williamson' AS
batsman_name FROM Cricket_Premier_League WHERE TEXT LIKE '%Kane
Williamson%' OR TEXT LIKE '%kane williamson%' OR TEXT LIKE '%kane%' OR
TEXT LIKE '%williamson%' UNION SELECT COUNT(*) AS tweets_count, 'Joe Root'
AS batsman_name FROM Cricket_Premier_League WHERE TEXT LIKE '%Joe Root%'
OR TEXT LIKE '%joe root%' OR  TEXT LIKE '%root%' OR TEXT LIKE '%Root%'
UNION SELECT COUNT(*) AS tweets_count, 'Shikhar Dhawan' AS batsman_name
FROM Cricket_Premier_League WHERE TEXT LIKE '%Shikhar Dhawan%' OR TEXT
LIKE '%shikhar dhawan%' OR  TEXT LIKE '%shikhar%' OR  TEXT LIKE '%dhawan%'
OR  TEXT LIKE '%gabbar%' UNION SELECT COUNT(*) AS tweets_count, 'David
Miller' AS batsman_name FROM Cricket_Premier_League WHERE TEXT LIKE
'%David Miller%' OR TEXT LIKE '%david miller%' OR  TEXT LIKE '%Miller%' OR
TEXT LIKE '%miller%' OR TEXT LIKE '%Miller%' ORDER BY tweets_count DESC")
```

Please find the screenshot of the query output.



Please find the screenshot of the data visualization for query 3.



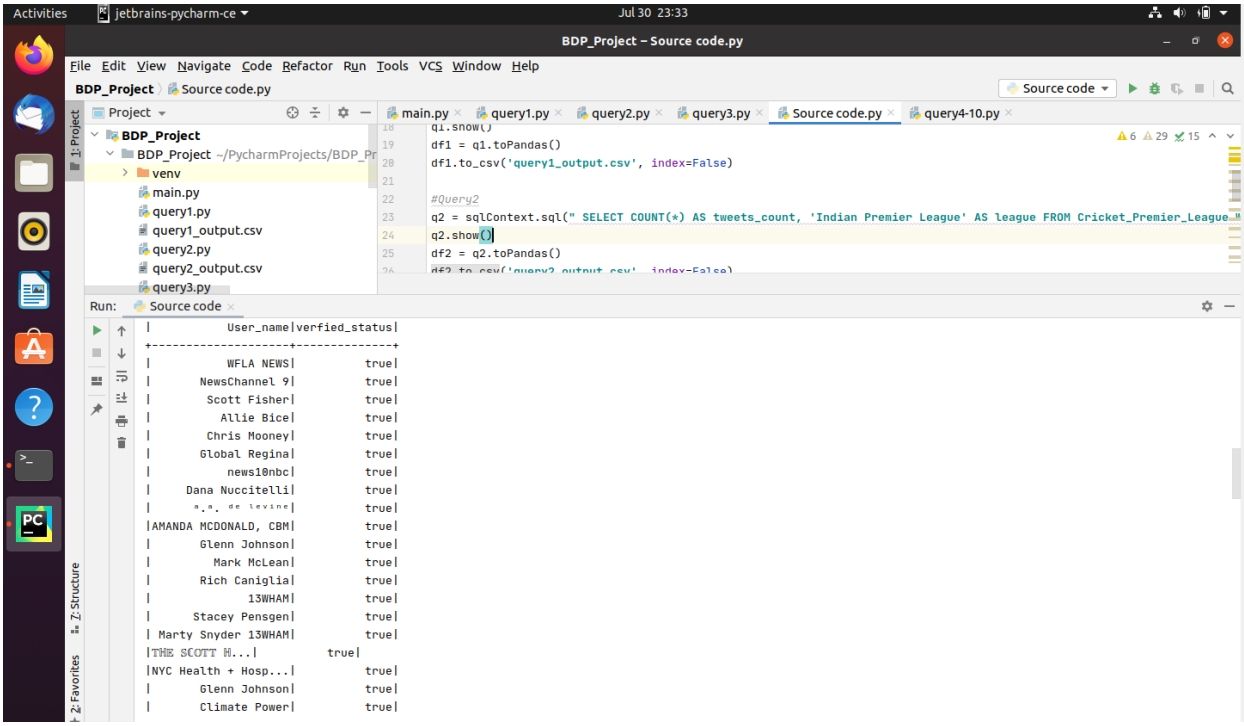
Query 4:

This query is used to extract the list of verified users.

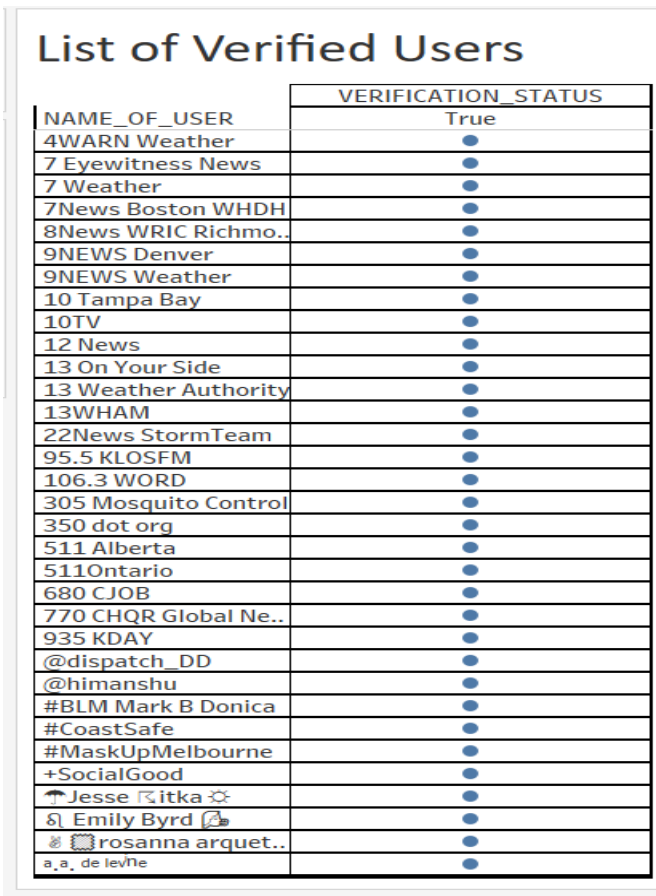
Please find the query below that had been used for extraction

```
q4 = sqlContext.sql("SELECT user.name AS User_name, user.verified as
verfied_status FROM Cricket_Premier_League WHERE user.verified = 'true'")
```

Please find the screenshot of the query output.



Please find the screenshot of the data visualization for query 4.



List of Verified Users

NAME_OF_USER	VERIFICATION_STATUS True
4WARN Weather	●
7 Eyewitness News	●
7 Weather	●
7News Boston WHDH	●
8News WRIC Richmo..	●
9NEWS Denver	●
9NEWS Weather	●
10 Tampa Bay	●
10TV	●
12 News	●
13 On Your Side	●
13 Weather Authority	●
13WHAM	●
22News StormTeam	●
95.5 KLOSFM	●
106.3 WORD	●
305 Mosquito Control	●
350 dot org	●
511 Alberta	●
511Ontario	●
680 CJOB	●
770 CHQR Global Ne..	●
935 KDAY	●
@dispatch_DD	●
@himanshu	●
#BLM Mark B Donica	●
#CoastSafe	●
#MaskUpMelbourne	●
+SocialGood	●
☂Jesse ✎itka ☼	●
♫ Emily Byrd	●
rosanna arquet..	●
a.a. de levine	●

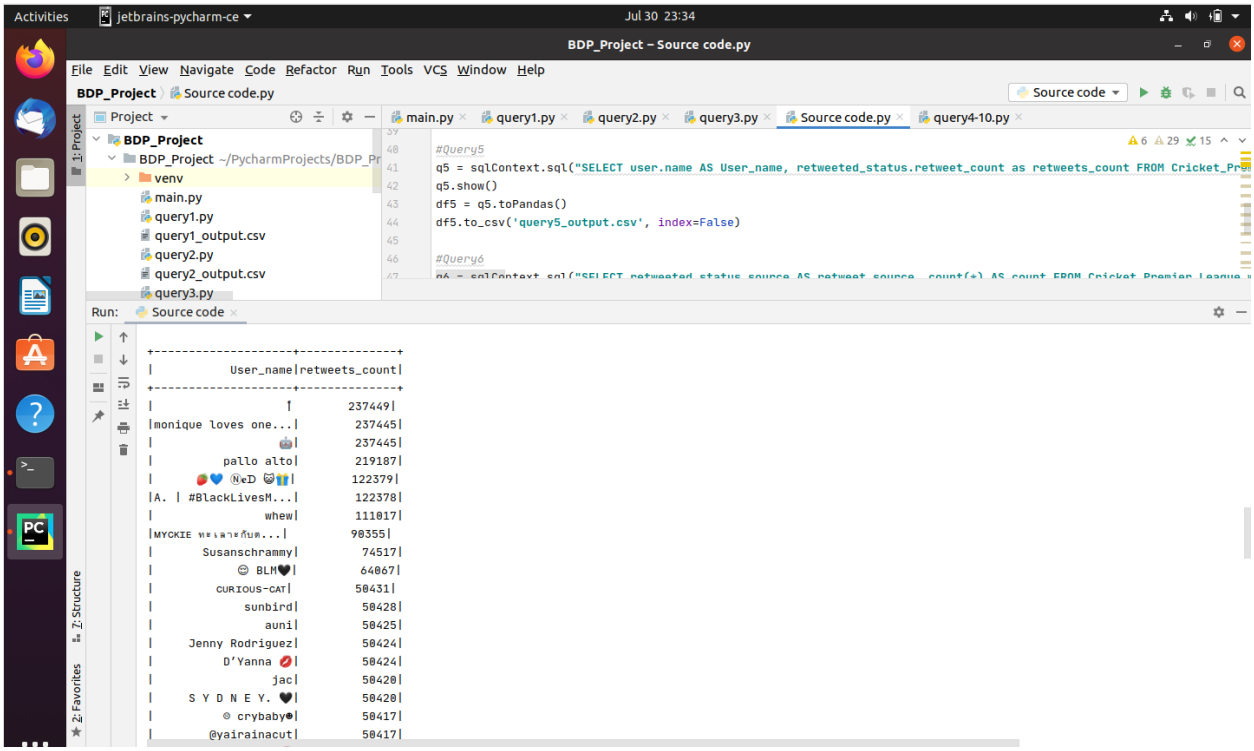
Query 5:

This query is used to extract the list of users who had retweeted for the tweets and their respective number of retweets.

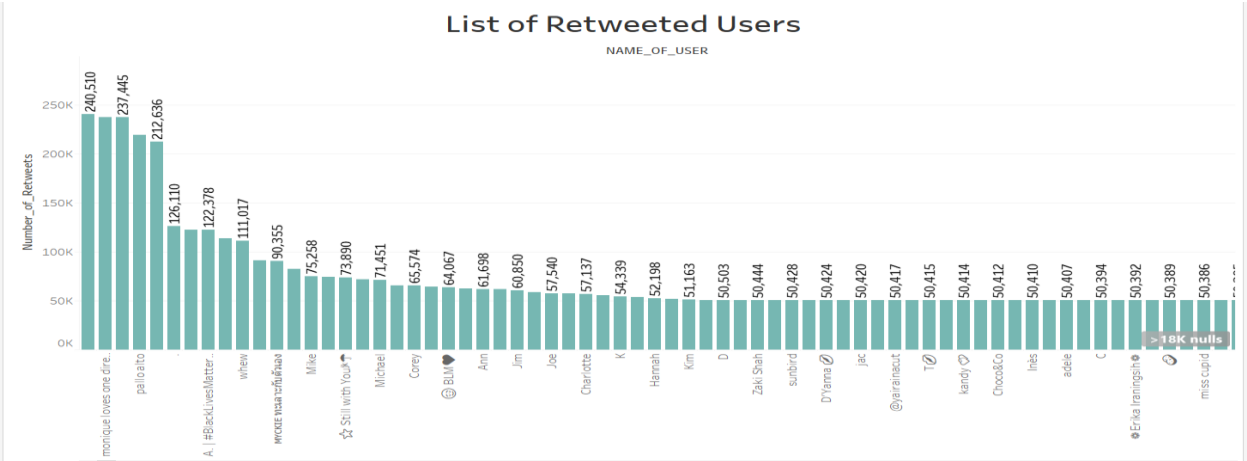
Please find the query below that had been used for extraction

```
q5 = sqlContext.sql("SELECT user.name AS User_name,
retweeted_status.retweet_count as retweets_count FROM
Cricket_Premier_League ORDER BY retweets_count DESC")
```

Please find the screenshot of the query output.



Please find the screenshot of the data visualization for query 5.



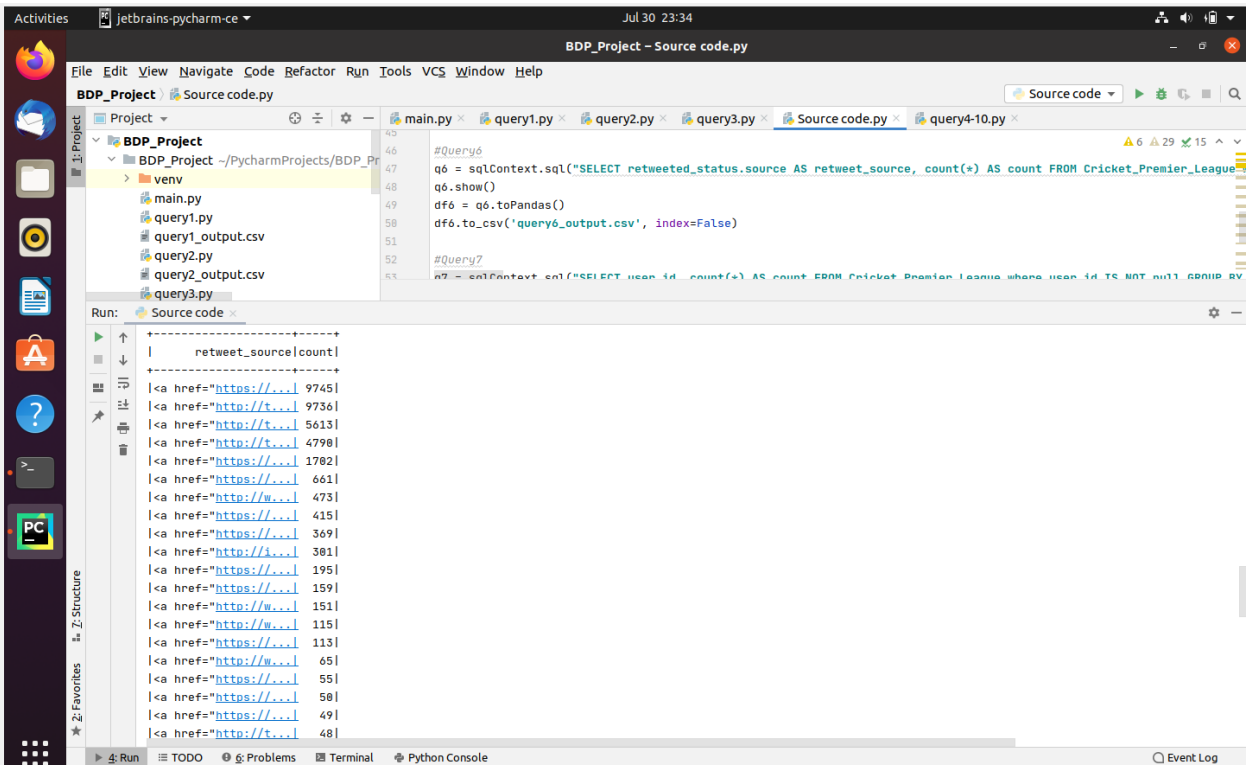
Query 6:

This query is used to extract the list of sources from which the tweets had been collected and their count and sort them from highest to lowest based on their count.

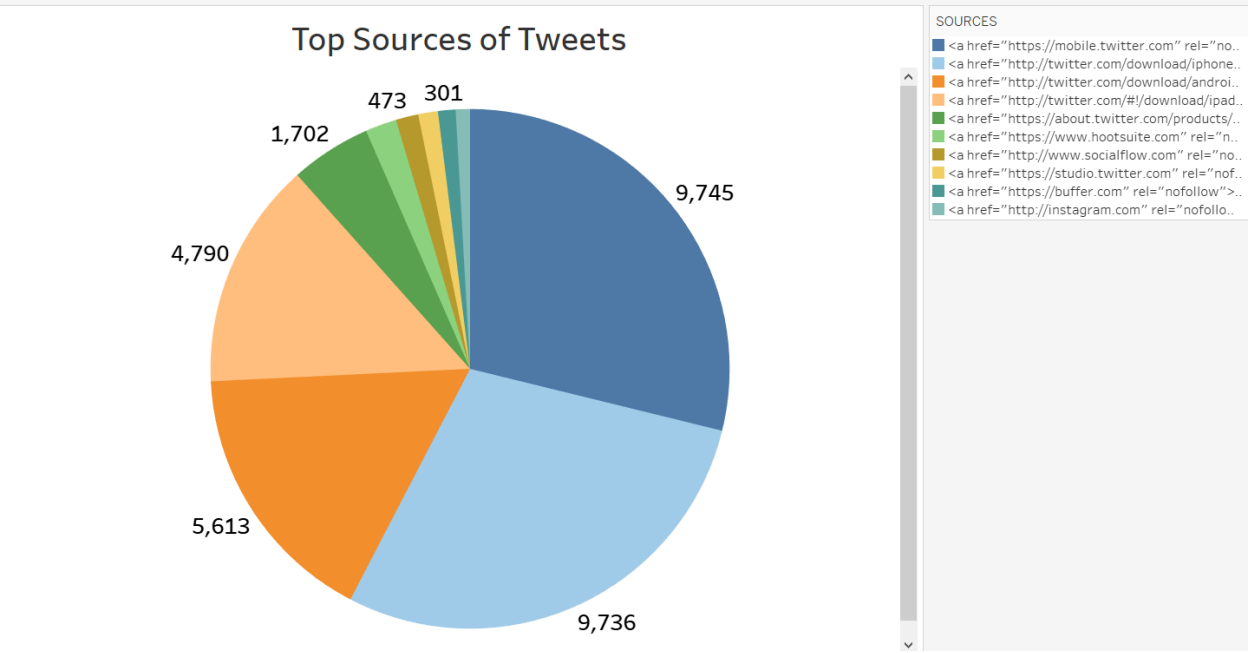
Please find the query below that had been used for extraction

```
q6 = sqlContext.sql("SELECT retweeted_status.source AS retweet_source, count(*) AS count FROM Cricket_Premier_League where retweeted_status.source IS NOT null GROUP BY retweeted_status.source ORDER BY count DESC")
```

Please find the screenshot of the query output.



Please find the screenshot of the data visualization for query 6.



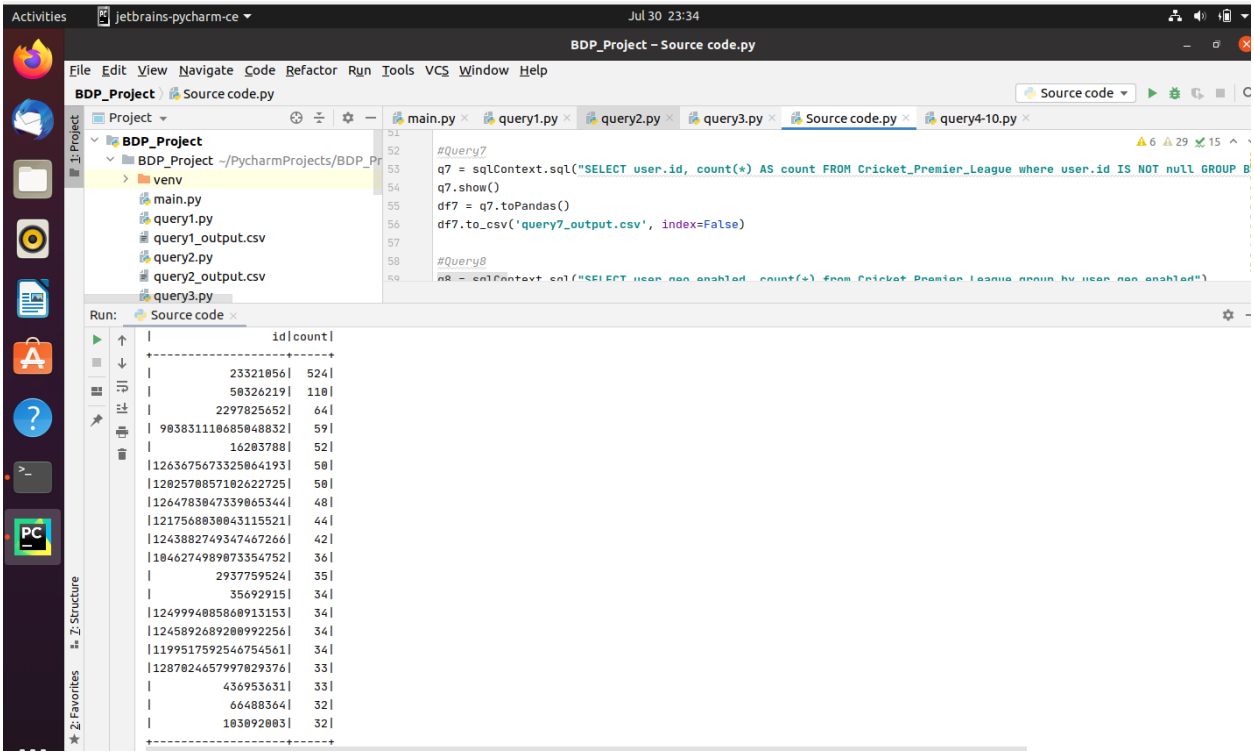
Query 7:

This query is used to find the user who tweeted the most about cricket.

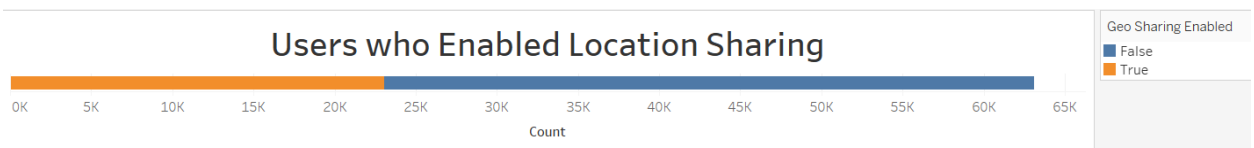
Please find the query below that had been used for extraction

```
q7 = sqlContext.sql("SELECT user.id, count(*) AS count FROM
Cricket_Premier_League where user.id IS NOT null GROUP BY user.id ORDER BY
count DESC")
```

Please find the screenshot of the query output.



Please find the screenshot of the data visualization for query 7.



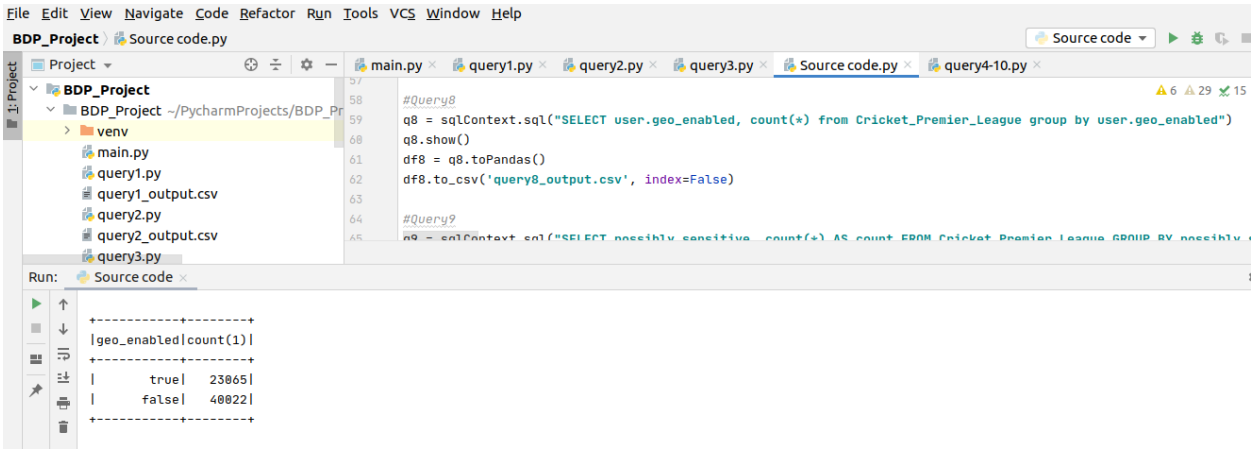
Query 8:

This query is used to extract the number of users who have enabled and not enabled location sharing.

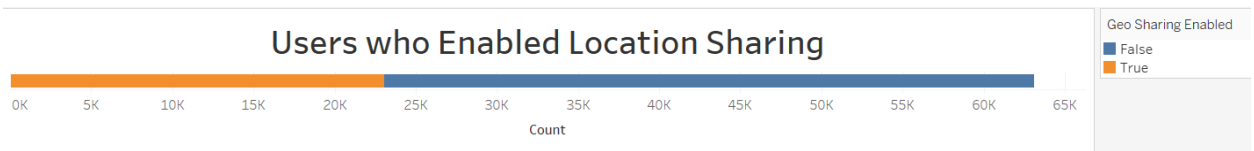
Please find the query below that had been used for extraction

```
q8 = sqlContext.sql("SELECT user.geo_enabled, count(*) from
Cricket_Premier_League group by user.geo_enabled")
```

Please find the screenshot of the query output.



Please find the screenshot of the data visualization for query 8.



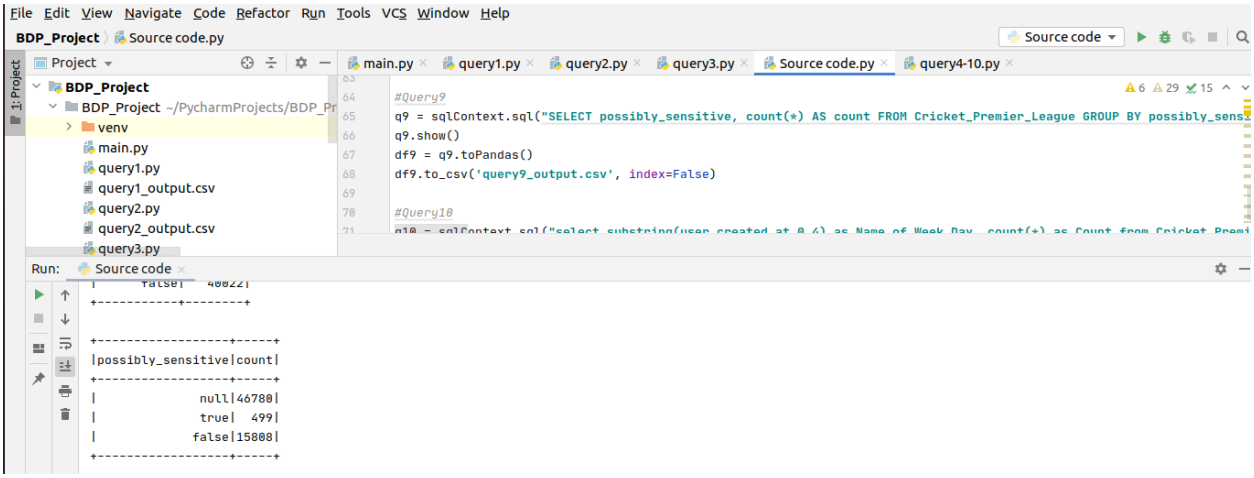
Query 9:

This query is used to test the sensitive data.

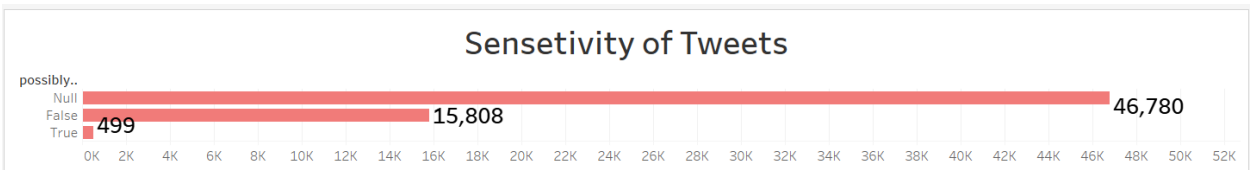
Please find the query below that had been used for extraction

```
q9 = sqlContext.sql("SELECT possibly_sensitive, count(*) AS count FROM
Cricket_Premier_League GROUP BY possibly_sensitive")
```

Please find the screenshot of the query output.



Please find the screenshot of the data visualization for query 9.



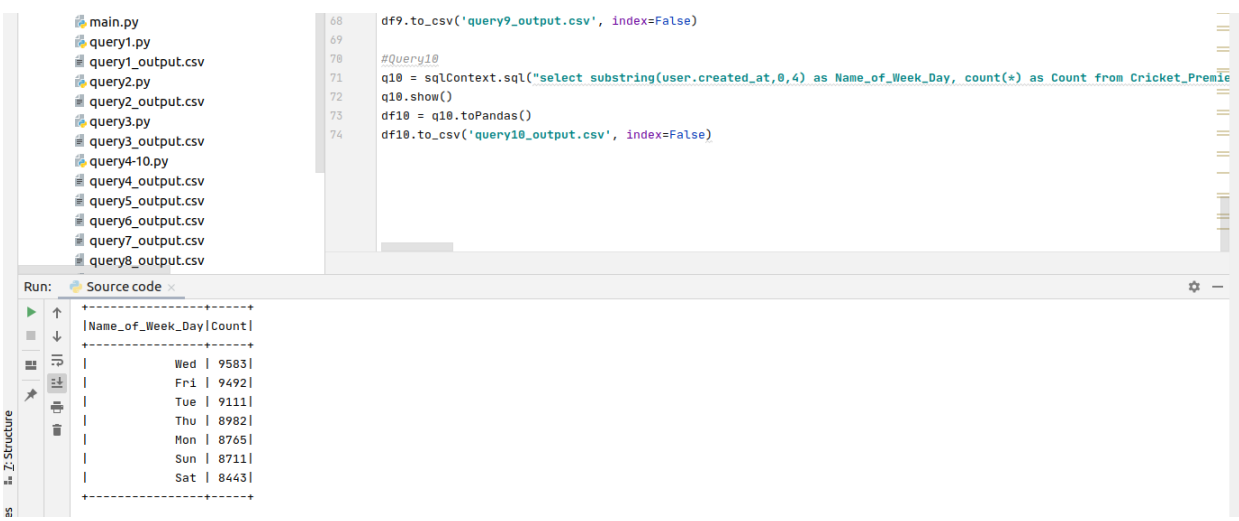
Query 10:

This query is used to find the weekdays on which tweet activity is high.

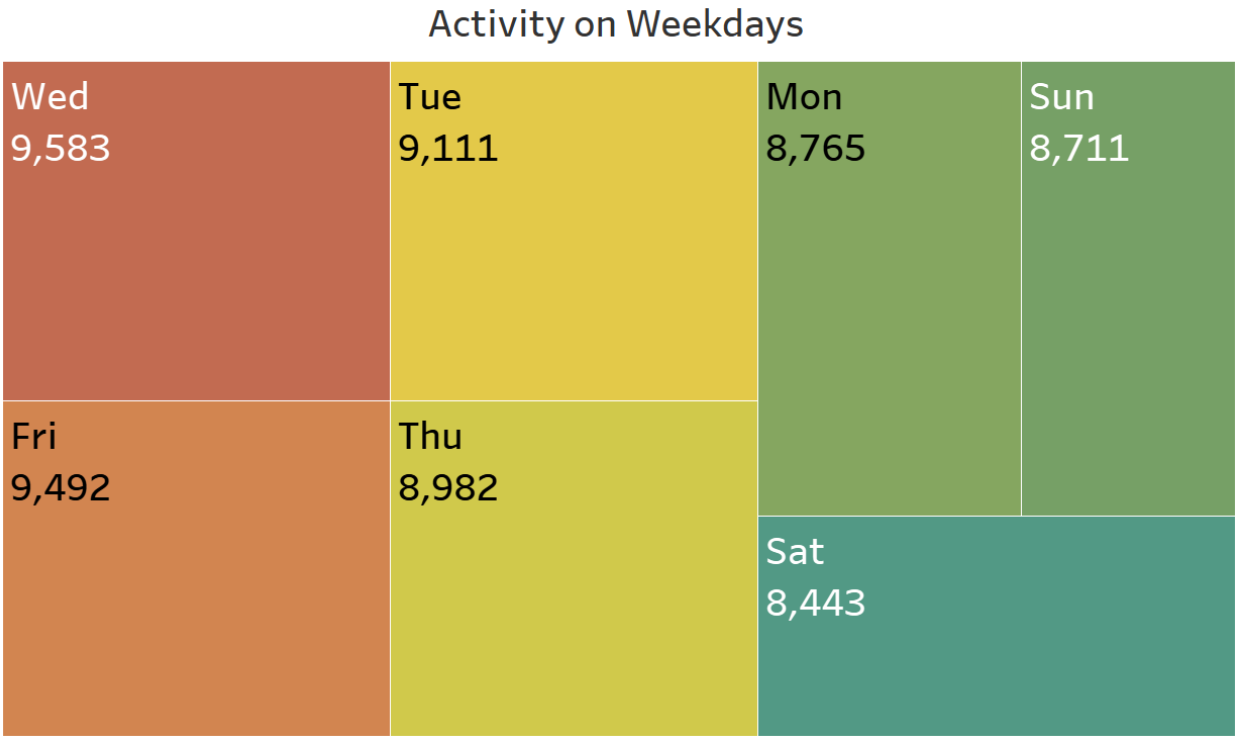
Please find the query below that had been used for extraction

```
q10 = sqlContext.sql("select substring(user.created_at,0,4) as
Name_of_Week_Day, count(*) as Count from Cricket_Premier_League group by
substring(user.created_at,0,4) order by Count desc")
```

Please find the screenshot of the query output.



Please find the screenshot of the data visualization for query 10.



Please find the Github link below

https://github.com/ynkc3/Principles-of-Big-Data-Academic-Project/tree/master/Phase%202