Phase 2 Report

Vamsi Draksharam-16291789

Divya Reddy - 16281700

Sai Charan Kottapalli -16247878

Abstract:

1. Performing analysis on the Twitter data being collected by the means of various queries.
2. Visualization of data.

Used Technologies:

1. Python
2. Spark

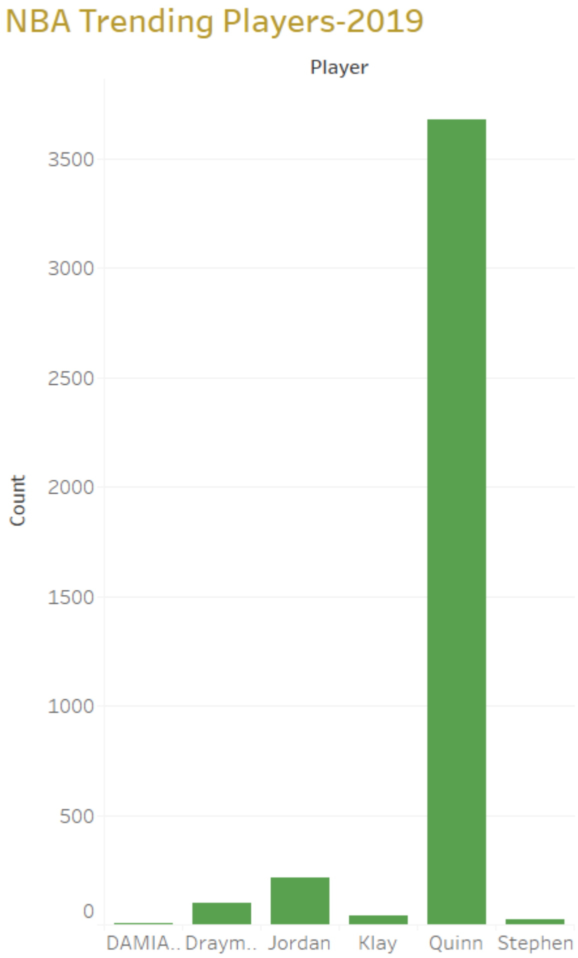
Tools:

Spyder(Python 3.7), Tableau

Queries:

Query 1: List of Trending Players of National Basketball Association(NBA) and the visualization through bar graph.

```
df=spark.read.json("C:/Users/Vamsi Draksharam/PycharmProjects/PB-Vamsi/phase2/data2.json")

df.createOrReplaceTempView("NBA")

   sqlhash = spark.sql("SELECT 'Quinn' player,count(text) as count  \

      FROM NBA\

      WHERE 1=1\

      AND (upper(text) LIKE '%COOK%' or upper(text) LIKE '%QUINN%' or upper(text) LIKE '%QUI%')\

      GROUP BY player\

      UNION\
      SELECT 'Klay' player,count(text) as count \
      FROM NBA\
      WHERE 1=1\
      AND (upper(text) LIKE '%KLAY%' or upper(text) LIKE '%THOMPSON%')\
      GROUP BY player\
      UNION\
      SELECT 'Stephen' player,count(text) as count \
      FROM NBA\
      WHERE 1=1\
      AND (upper(text) LIKE '%STEPHEN%' or text LIKE '%stephen%')\
      GROUP BY player\
      UNION\
      SELECT 'Draymond' player,count(text) as count\
      FROM NBA\
      WHERE 1=1\
      AND (upper(text) LIKE '%DRAYMOND%' or upper(text) LIKE '%GREEN%')\
      GROUP BY player\
      UNION\
      SELECT 'DAMIAN' player,count(text) as count \
      FROM NBA\
      WHERE 1=1\
      AND (upper(text) LIKE '%DAMIAN%' or text LIKE
  '%damian%')\
      GROUP BY player\
      UNION\
      SELECT 'Jordan' player,count(text) as count \
      FROM NBA\
      WHERE 1=1\
      AND (upper(text) LIKE '%JORDAN BELL%' or
  upper(text) LIKE '%JORDAN%' or upper(text) LIKE
  '%BELL%')\
      GROUP BY player")
    sqlhash.show()
    sqlhash.toPandas().to_csv('1.csv')
```

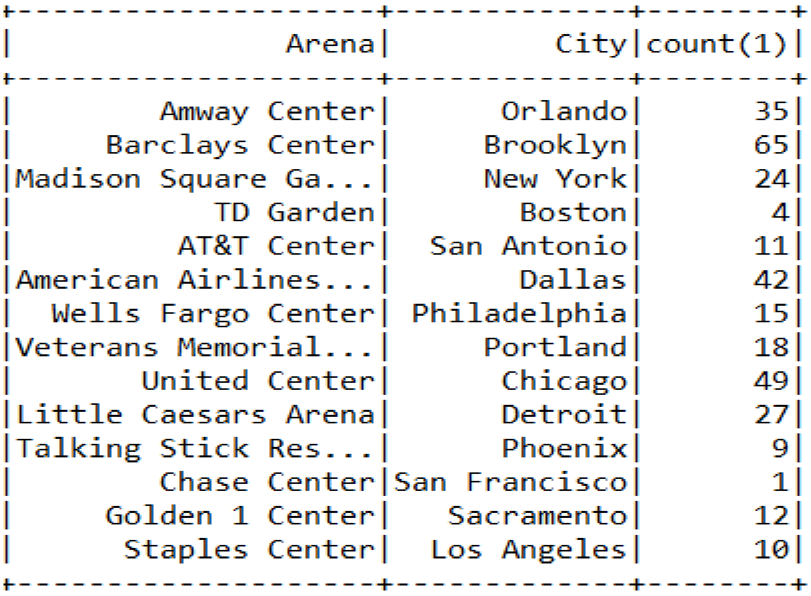


Query2: NBA 2019 - Number of matches being held in various cities and the visualization using pie-chart.

```
df=spark.read.json("C:/Users/Vamsi Draksharam/PycharmProjects/PB-Vamsi/phase2/data2.json")
df.createOrReplaceTempView("nba")
```

```
sqldf= spark.sql("SELECT 'Staples Center' Arena,'Los Angeles' City,count(*) FROM nba WHERE upper(text) LIKE '%LOS ANGELES%' or text like '%los angeles%' \
UNION \
SELECT 'Amway Center' Arena,'Orlando' City,count(*) FROM nba WHERE upper(text) LIKE '%ORLANDO%' or text like '%orlando%' \
UNION \
SELECT 'TD Garden' Arena,'Boston' City,count(*) FROM nba WHERE upper(text) LIKE '%BOSTON' or text like '%boston%' \
UNION \
SELECT 'American Airlines Center' Arena,'Dallas' City,count(*) FROM nba WHERE upper(text) LIKE '%DALLAS%' or text like '%dallas%' \
UNION \
SELECT 'Madison Square Garden' Arena,'New York' City,count(*) FROM nba WHERE upper(text) LIKE '%NEW YORK%' or text like '%new york%' \
UNION \
SELECT 'Veterans Memorial Coliseum' Arena,'Portland' City,count(*) FROM nba WHERE upper(text) LIKE '%PORTLAND%' or text like '%portland%' \
UNION \
SELECT 'Wells Fargo Center' Arena,'Philadelphia' City,count(*) FROM nba WHERE upper(text) LIKE '%PHILADELPHIA%' or text like '%philadelphia%' \
UNION \
SELECT 'Golden 1 Center' Arena,'Sacramento' City,count(*) FROM nba WHERE upper(text) LIKE '%SACRAMENTO%' or text like '%sacramento%' \
UNION \
SELECT 'Barclays Center' Arena,'Brooklyn' City,count(*) FROM nba WHERE upper(text) LIKE '%BROOKLYN%' or text like '%brooklyn%' \
UNION \
SELECT 'AT&T Center' Arena,'San Antonio' City,count(*) FROM nba WHERE upper(text) LIKE '%SAN ANTONIO%' or text like '%san antonio%' \
UNION \
SELECT 'Little Caesars Arena' Arena,'Detroit' City,count(*)
FROM nba WHERE upper(text) LIKE '%DETROIT%' or text
like '%detroit%' \

UNION \

SELECT 'Chase Center' Arena,'San Francisco' City,count(*)
FROM nba WHERE upper(text) LIKE '%SAN FRANCISCO%'
or text like '%san francisco%' \

UNION \

SELECT 'Talking Stick Resort Arena' Arena,'Phoenix'
City,count(*) FROM nba WHERE upper(text) LIKE
'%PHOENIX%' or text like '%phoenix%' \

UNION \

SELECT 'United Center' Arena,'Chicago' City,count(*)
FROM nba WHERE upper(text) LIKE '%CHICAGO%' or text
like '%chicago%'")

sqldf.show(150)


sqldf.toPandas().to_csv('2.csv')
```

Output:



```
+--------------------+-------------+--------+
|               Arena|         City|count(1)|
+--------------------+-------------+--------+
|        Amway Center|      Orlando|      35|
|     Barclays Center|     Brooklyn|      65|
|Madison Square Ga...|     New York|      24|
|           TD Garden|       Boston|       4|
|         AT&T Center|  San Antonio|      11|
|American Airlines...|       Dallas|      42|
|  Wells Fargo Center| Philadelphia|      15|
|Veterans Memorial...|     Portland|      18|
|      United Center|      Chicago|      49|
|Little Caesars Arena|      Detroit|      27|
|Talking Stick Res...|      Phoenix|       9|
|        Chase Center|San Francisco|       1|
|     Golden 1 Center|   Sacramento|      12|
|      Staples Center|  Los Angeles|      10|
+--------------------+-------------+--------+
```

NBA 2019 - Number of matches being conducted in various cities

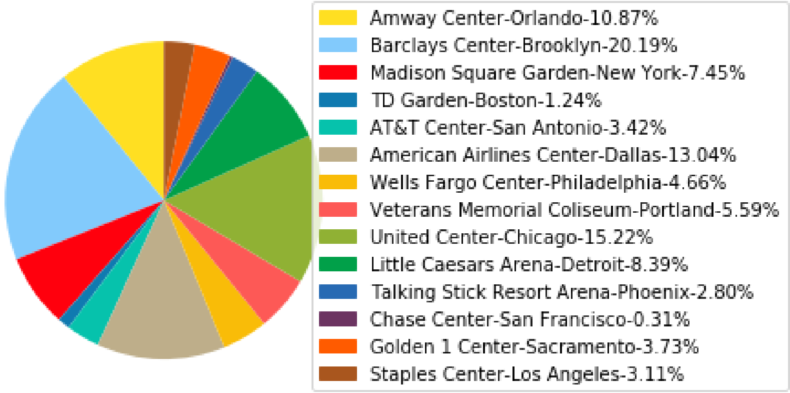
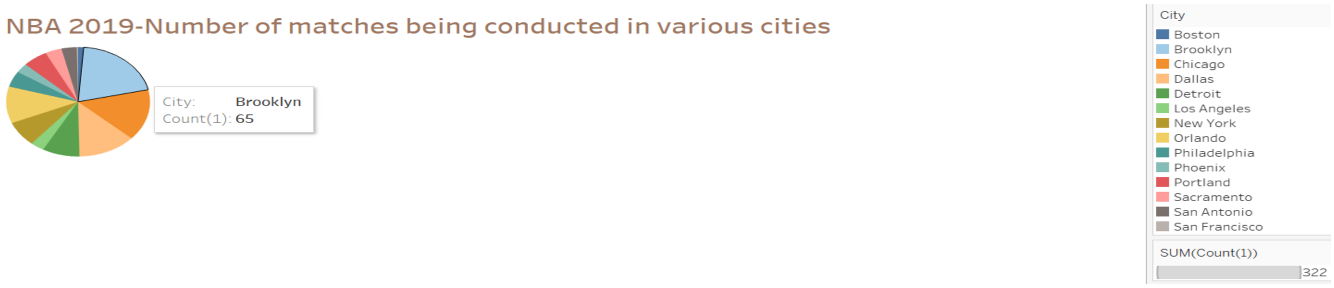


Tableau Visualization

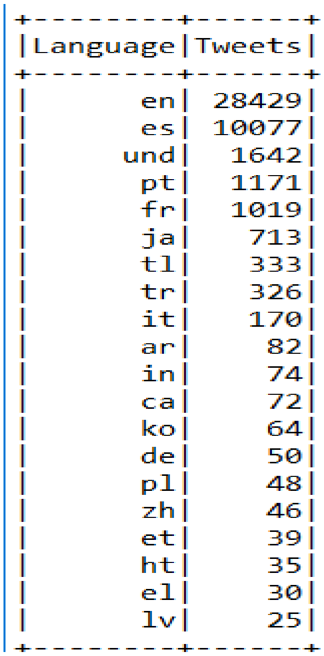


NBA 2019-Number of matches being conducted in various cities

Query3: Display of Tweets from top 20 languages and the visualization using bar-graph

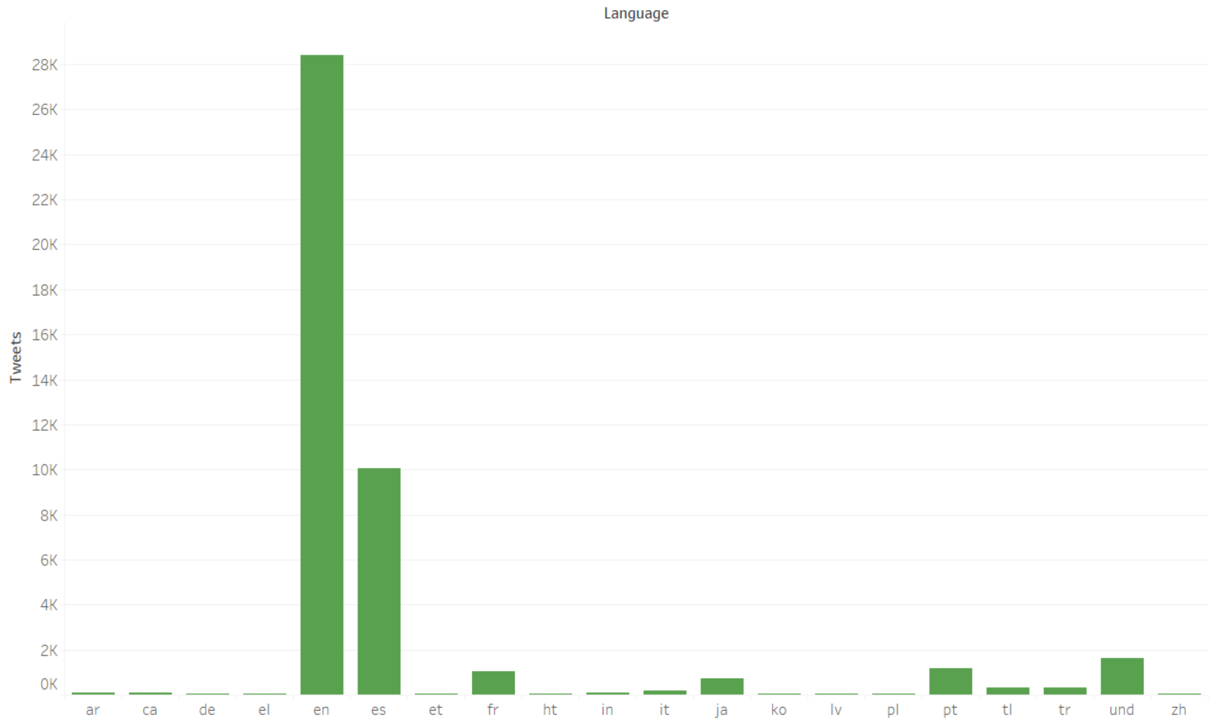
```
df=spark.read.json("C:/Users/Vamsi Draksharam/PycharmProjects/PB-Vamsi/phase2/data2.json")
        df.createOrReplaceTempView("nba")
         sqldf= spark.sql("SELECT nba.lang Language,count(*) Tweets  FROM nba WHERE nba.lang is NOT NULL  GROUP BY nba.lang ORDER BY 2 DESC limit 20")
         sqldf.show(150)
```

Output:



```
+--------+------+
|Language|Tweets|
+--------+------+
|      en| 28429|
|      es| 10077|
|     und|  1642|
|      pt|  1171|
|      fr|  1019|
|      ja|   713|
|      tl|   333|
|      tr|   326|
|      it|   170|
|      ar|    82|
|      in|    74|
|      ca|    72|
|      ko|    64|
|      de|    50|
|      pl|    48|
|      zh|    46|
|      et|    39|
|      ht|    35|
|      el|    30|
|      lv|    25|
+--------+------+
```

Tweets from top 20 languages

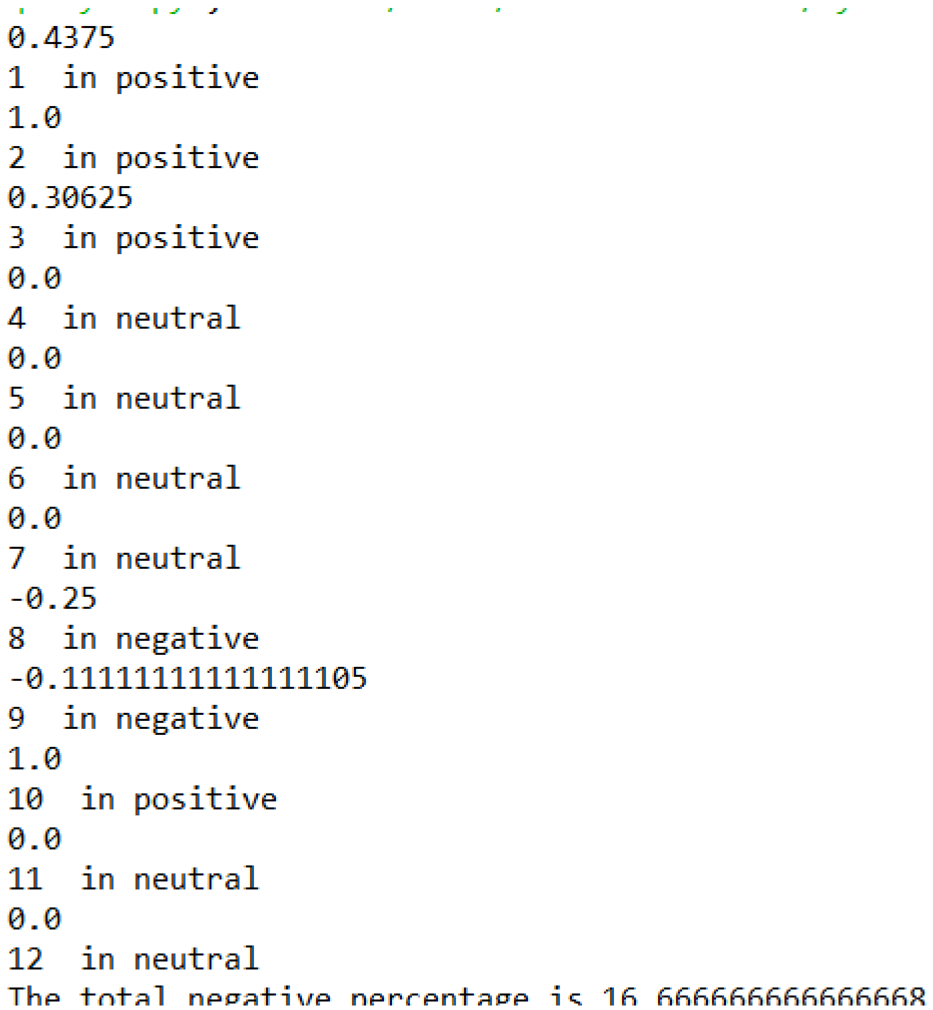


Query 4: Displaying the supporters and hatters of LeBron James in NBA-2019 and the visualization using Donut Pie-Chart.

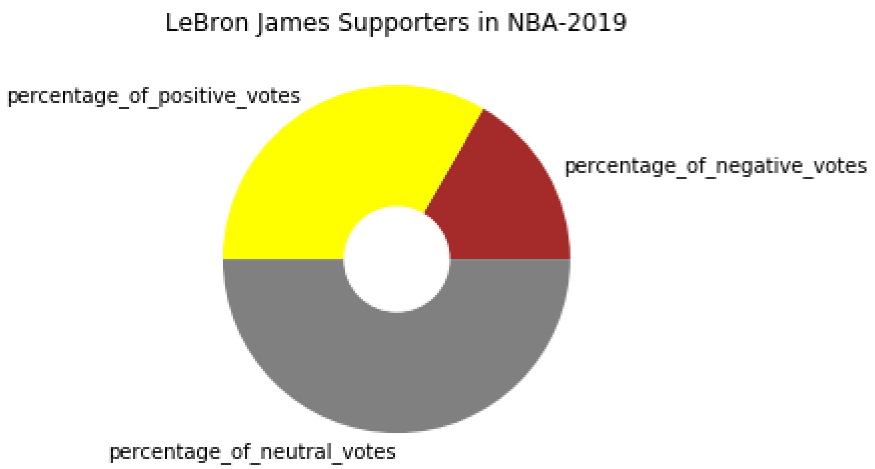
```
df=spark.read.json("C:/Users/Vamsi Draksharam/PycharmProjects/PB-Vamsi/phase2/data2.json")

date= df.select("created_at")

def dateMTest(dateval):

    dt=datetime.datetime.strptime(dateval, '%a %b %d %H:%M:%S +0000 %Y')

    return dt

d = udf(dateMTest , DateType())

df=df.withColumn("created_date",d(date.created_at))

df.createOrReplaceTempView("nba")

sqldf= spark.sql("SELECT id,text,created_date  FROM nba WHERE 1=1 AND (upper(text) LIKE '%LEBRON%'AND text LIKE '%nba%')")

i=0

positive=0
```

```
neutral=0
negative=0
for t in sqldf.select("text").collect():
    i=i+1

    analysis = TextBlob(str((t.text).encode('ascii', 'ignore')))
    print(analysis.sentiment.polarity)
    if (analysis.sentiment.polarity<0):
        negative=negative+1
        print(i," in negative")
    elif(analysis.sentiment.polarity==0.0):
        neutral=neutral+1
        print(i," in neutral")
    elif(analysis.sentiment.polarity>0):
        positive=positive+1
        print(i," in positive")
print("The total negative percentage is",((negative)*100)/i)
print("The total neutral percentage is",((neutral)*100)/i)
print("The total positive percentage is",((positive)*100)/i)
percentage_of_negative_votes=((negative)*100)/i
percentage_of_positive_votes=((positive)*100)/i
percentage_of_neutral_votes=((neutral)*100)/i
```

Output:



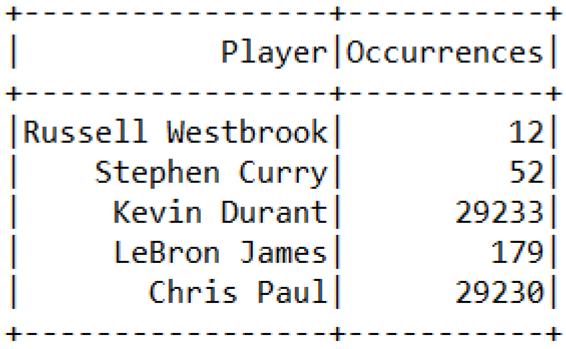
```
0.4375
1  in positive
1.0
2  in positive
0.30625
3  in positive
0.0
4  in neutral
0.0
5  in neutral
0.0
6  in neutral
0.0
7  in neutral
-0.25
8  in negative
-0.11111111111111105
9  in negative
1.0
10  in positive
0.0
11  in neutral
0.0
12  in neutral
The total negative percentage is 16.666666666666668
```



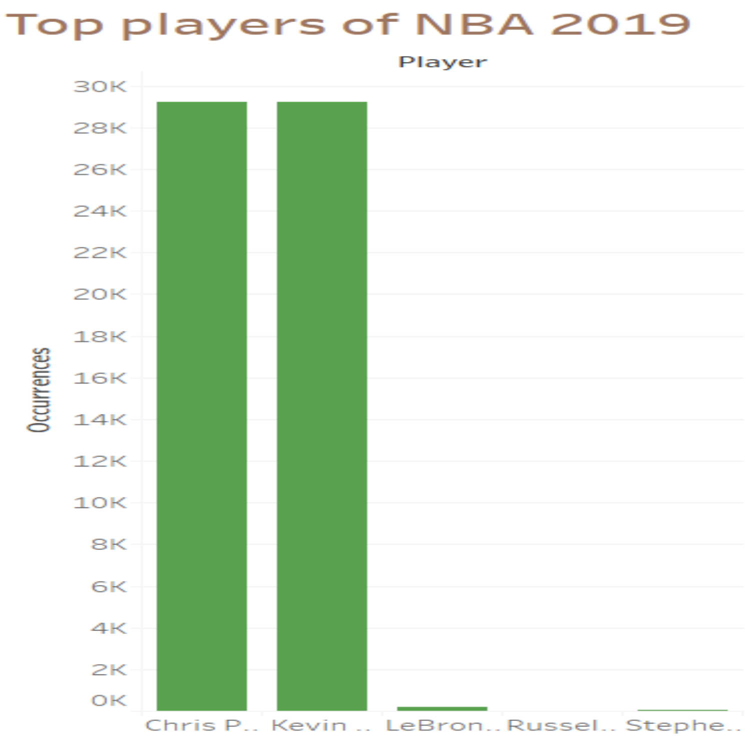
Query5: List of top NBA players and their occurrences.

```
df=spark.read.json("C:/Users/Vamsi
Draksharam/PycharmProjects/PB-
Vamsi/phase2/data2.json")


df.createOrReplaceTempView("NBA")
sqlDF = spark.sql("SELECT 'Chris Paul' as Player, count(*)
as Occurrences from nba where text like '%chris paul%'
or text like '%nba%' or upper(text) like '%CHRIS PAUL%'
or upper(text) like '%NBA%'\
    UNION\
    SELECT 'Stephen Curry' as Player, count(*) as Occurrences from nba where text like '%curry%' or upper(text) like '%CURRY%'\
    UNION\
    SELECT 'Kevin Durant' as Player, count(*) as Occurrences from nba where text like '%kevin durant%' or upper(text) like '%KEVIN DURANT%' or text like '%nba%' or upper(text) like '%NBA%'  UNION\
    SELECT 'LeBron James' as Player, count(*) as Occurrences from nba where text like '%lebron%' or upper(text) like '%LEBRON%' or text like '%LeBron James' UNION\
    SELECT 'Russell Westbrook' as Player, count(*) as Occurrences from nba where text like '%westbrook%' or upper(text) like '%WESTBROOK%'")
pd = sqlDF.toPandas()
```

Output:



```
+-----------------+-----------+
|           Player|Occurrences|
+-----------------+-----------+
|Russell Westbrook|         12|
|    Stephen Curry|         52|
|     Kevin Durant|      29233|
|     LeBron James|        179|
|       Chris Paul|      29230|
+-----------------+-----------+
```

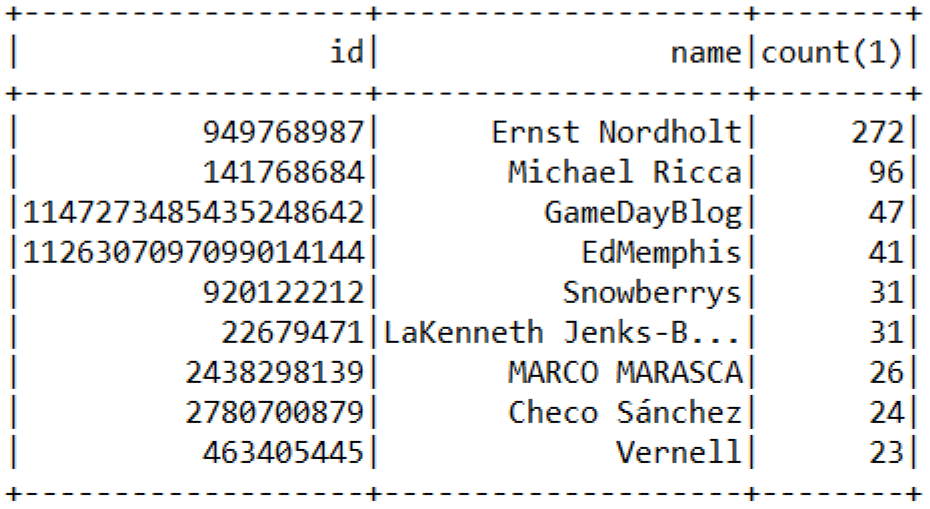


Top players of NBA 2019

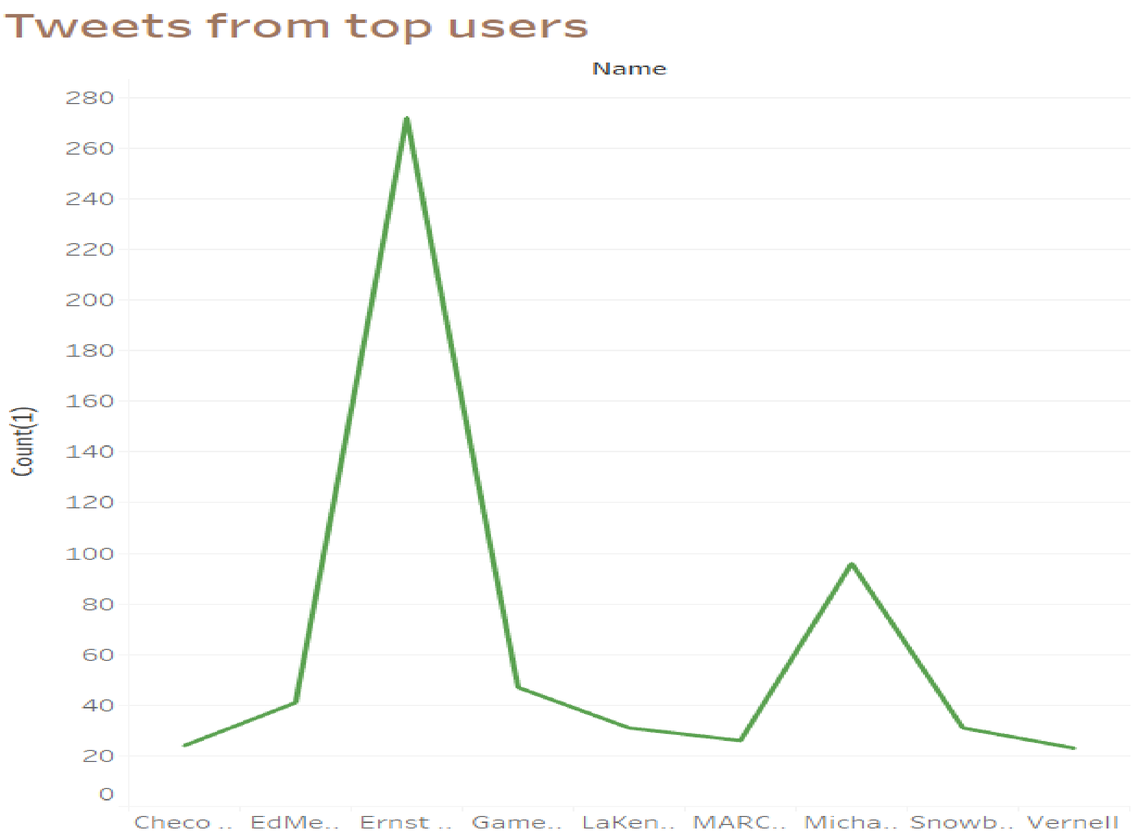
Query6: Tweets from top Users and the visualization using line-graph.

```
df=spark.read.json("C:/Users/Vamsi
Draksharam/PycharmProjects/PB-
Vamsi/phase2/data2.json")
df.createOrReplaceTempView("Users")
sqldf = spark.sql(
    "SELECT user.id,user.name,count(*) FROM Users"
    " WHERE (user.id is not null and user.name is not null)
group by user.id,user.name order by 3 desc limit 9")
sqldf.show(150)
```

Output:



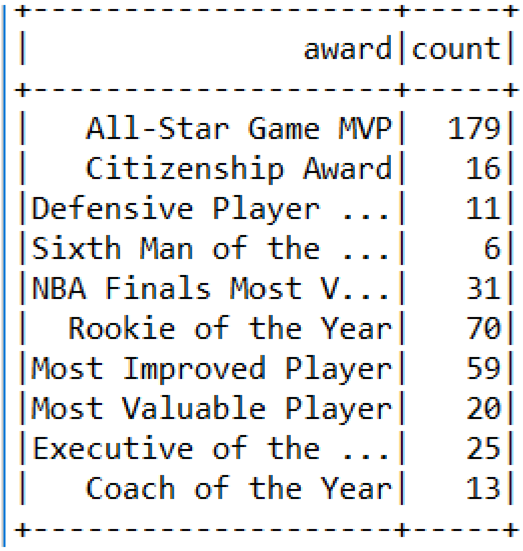
```
+-------------------+--------------------+--------+
|                 id|                name|count(1)|
+-------------------+--------------------+--------+
|          949768987|      Ernst Nordholt|     272|
|          141768684|       Michael Ricca|      96|
|1147273485435248642|         GameDayBlog|      47|
|1126307097099014144|           EdMemphis|      41|
|          920122212|          Snowberrys|      31|
|           22679471|LaKenneth Jenks-B...|      31|
|         2438298139|       MARCO MARASCA|      26|
|         2780700879|       Checo Sánchez|      24|
|          463405445|             Vernell|      23|
+-------------------+--------------------+--------+
```



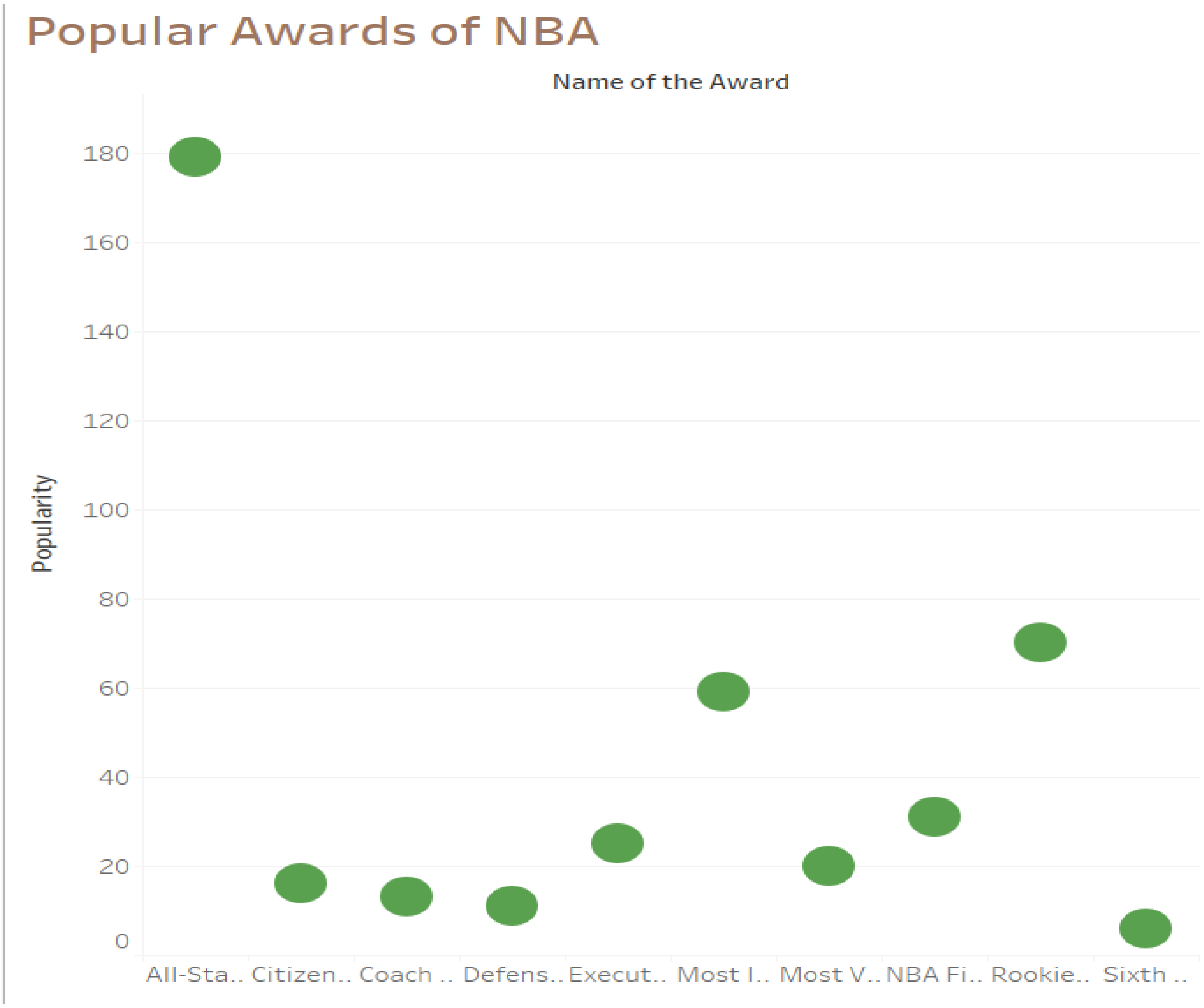
Query7: List of Popular Awards of NBA.

```
df=spark.read.json("C:/Users/Vamsi Draksharam/PycharmProjects/PB-Vamsi/phase2/data2.json")

df.createOrReplaceTempView("nba")

sqldf = spark.sql("SELECT 'All-Star Game MVP' award,count(text) as count  \

   FROM nba\

   WHERE 1=1\

   AND (upper(text) LIKE '%MVP%' or upper(text) LIKE '%LEBRON JAMES%' or text like '%LeBron James%' or upper(text) LIKE '%LEBRON%' or text like '%LeBron%')\

   GROUP BY award\

   UNION\

   SELECT 'Rookie of the Year' award,count(text) as count \

   FROM nba\

   WHERE 1=1\
   AND (upper(text) LIKE '%ROOKIE%' or (upper(text) LIKE '%STEPHEN%' or upper(text) LIKE '%CURRY%' or text like '%stephen%')\
   GROUP BY award     UNION\
   SELECT 'Most Valuable Player' award,count(text) as count \
   FROM nba\
   WHERE 1=1\
   AND (upper(text) LIKE '%MOST VALUABLE PLAYER%' or upper(text) LIKE '%KEVIN DURANT%' or text like '%kevin durant%' or text like '%Durant%')\
   GROUP BY award   UNION\
   SELECT 'Coach of the Year' award,count(text) as count \
   FROM nba\
   WHERE 1=1\
   AND (upper(text) LIKE '%COACH OF THE YEAR%' or upper(text) LIKE '%ANTHONY DAVIS%' or text like '%anthony davis%')\
   GROUP BY award   UNION\
   SELECT 'NBA Finals Most Valuable Player' award,count(text) as count \
   FROM nba\
   WHERE 1=1\
   AND (upper(text) LIKE '%NBA FINALS MOST VALUABLE%' or upper(text) LIKE '%JAMES HARDEN%' or text like '%james harden%')\
   GROUP BY award   UNION\
   SELECT 'Executive of the Year' award,count(text) as count \
   FROM nba\
   WHERE 1=1\
   AND (upper(text) LIKE '%EXECUTIVE OF THE YEAR%' or upper(text) LIKE '%ANTETOKOUNMPO%' or text like '%antetokounmpo%')\
   GROUP BY award  UNION\
   SELECT 'Citizenship Award' award,count(text) as count \
   FROM nba\
   WHERE 1=1\
   AND (upper(text) LIKE '%CITIZENSHIP AWARD%' or upper(text) LIKE '%EMBIID%' or text like '%Embiid%')\
   GROUP BY award  UNION\
   SELECT 'Defensive Player of the Year' award,count(text) as count \
   FROM nba\
   WHERE 1=1\
   AND (upper(text) LIKE '%DEFENSIVE PLAYER%' or upper(text) LIKE '%RUSSELL WESTBROOK%' or text like '%Westbrook%')\
   GROUP BY award  UNION\
   SELECT 'Sixth Man of the Year' award,count(text) as count \
   FROM nba\
   WHERE 1=1\
   AND (upper(text) LIKE '%SIXTH MAN OF THE YEAR%' or upper(text) LIKE '%PAUL GEORGE%' or text like '%Paul George%')\
   GROUP BY award  UNION\
   SELECT 'Most Improved Player' award,count(text) as count \
    FROM nba\
    WHERE 1=1\
    AND (upper(text) LIKE '%MOST IMPROVED PLAYER%' or upper(text) LIKE '%KAWHI LEONARD%' or text like '%Kawhi Leonard%' or text like '%Kawhi%')\
    GROUP BY award")
sqldf.show(150)

sqldf.toPandas().to_csv('7.csv')
```

Output:



```
+--------------------+-----+
|               award|count|
+--------------------+-----+
|   All-Star Game MVP|  179|
|   Citizenship Award|   16|
|Defensive Player ...|   11|
|Sixth Man of the ...|    6|
|NBA Finals Most V...|   31|
|  Rookie of the Year|   70|
|Most Improved Player|   59|
|Most Valuable Player|   20|
|Executive of the ...|   25|
|   Coach of the Year|   13|
+--------------------+-----+
```

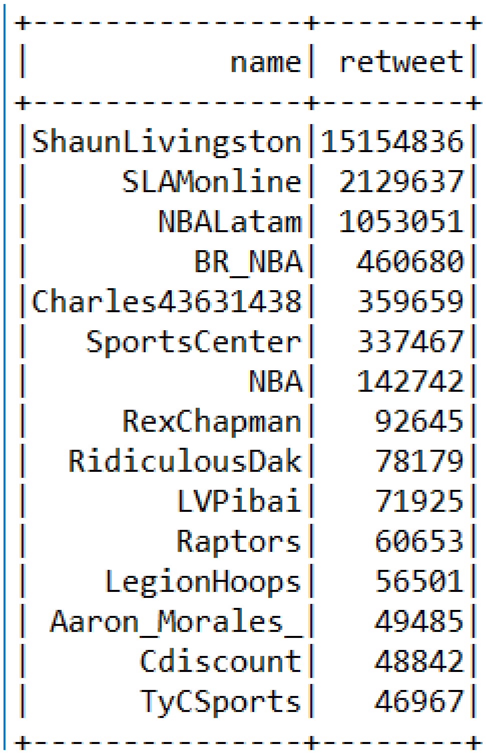


Query8: To display the number of retweets from top pages and visualization using pie-chart.

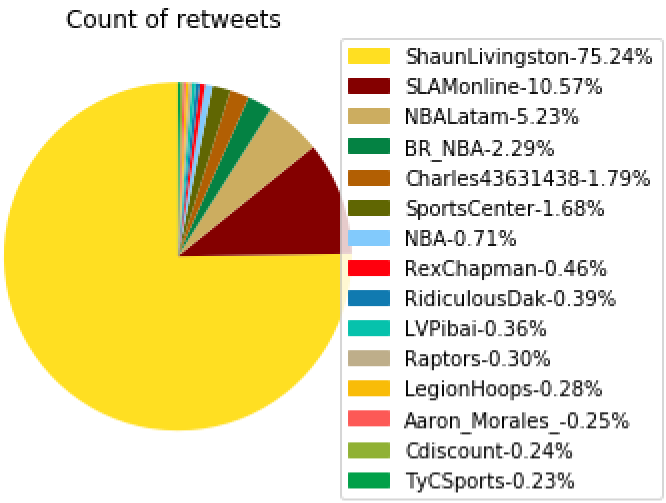
```
df=spark.read.json("C:/Users/Vamsi Draksharam/PycharmProjects/PB-Vamsi/phase2/data2.json")
```

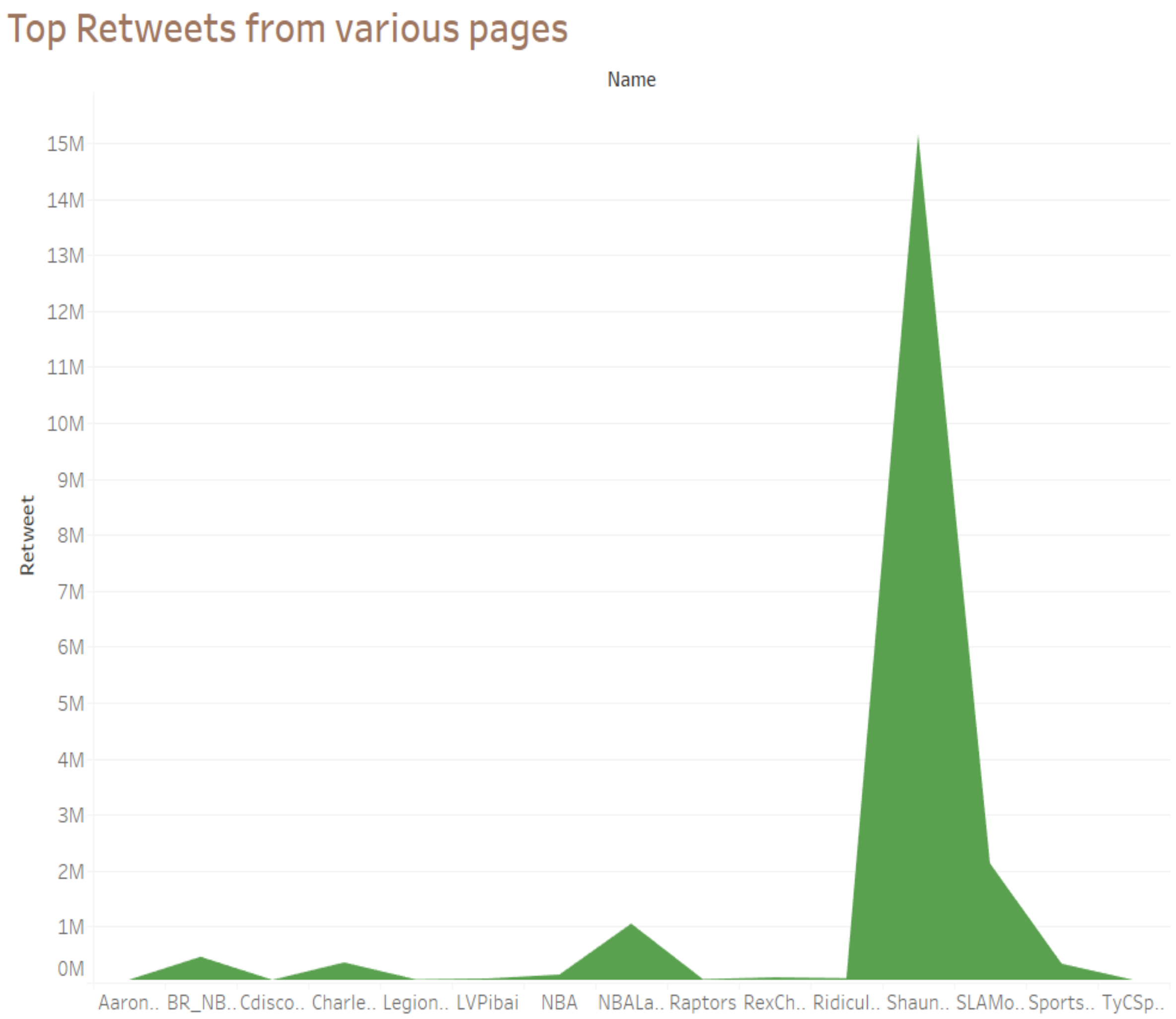
```
df.createOrReplaceTempView("nba")
sqldf = spark.sql(
   "SELECT name,SUM(cnt) as retweet FROM (SELECT
quoted_status.user.screen_name AS
name,quoted_status.retweet_count AS cnt FROM nba
WHERE quoted_status.retweet_count>0)GROUP BY
name ORDER BY retweet DESC LIMIT 15")
sqldf.show(150)
```

Output:



```
+---------------+--------+
|           name| retweet|
+---------------+--------+
|ShaunLivingston|15154836|
|      SLAMonline| 2129637|
|        NBALatam| 1053051|
|          BR_NBA|  460680|
|Charles43631438|  359659|
|    SportsCenter|  337467|
|             NBA|  142742|
|      RexChapman|   92645|
|   RidiculousDak|   78179|
|         LVPibai|   71925|
|         Raptors|   60653|
|     LegionHoops|   49485|
|  Aaron_Morales_|   49485|
|       Cdiscount|   48842|
|       TyCSports|   46967|
+---------------+--------+
```



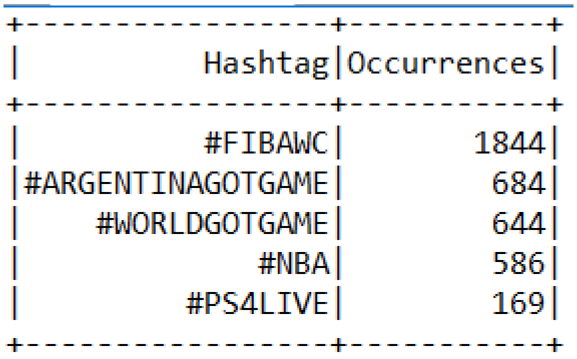


Top Retweets from various pages

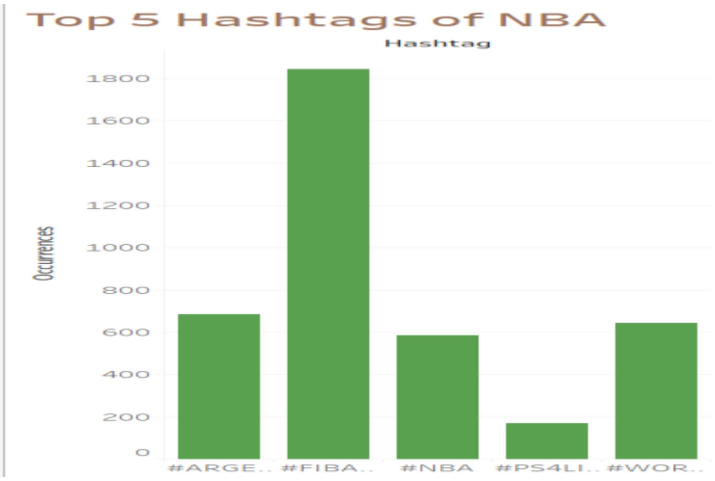
Query9: The list of Top 5 Hashags of NBA 2019 and visualization using bar-graph.

```
df=spark.read.json("C:/Users/Vamsi
Draksharam/PycharmProjects/PB-
Vamsi/phase2/data2.json")
    words = df.select(
        explode(
            split(df.text, " ")
        ).alias("word")
    )
    def extract_tags(word):
        if word.lower().startswith("#"):
            return word
        else:
            return "nonTag"
    extract_tags_udf = udf(extract_tags, StringType())
    resultDF = words.withColumn("tags",
extract_tags_udf(words.word))
    resultDF.createOrReplaceTempView("hashtag_count")
    sqlhash = spark.sql("SELECT Hashtag,\
        Occurrences\
        FROM (SELECT upper(tags) Hashtag,\
        count(*) Occurrences\
        FROM hashtag_count\
        WHERE 1=1\
        AND tags!='nonTag'\
        GROUP BY upper(tags)\
        ORDER BY Occurrences desc, Hashtag asc) limit 5")
    sqlhash.show(70)

    sqlhash.toPandas().to_csv('9.csv')
```

Output:



```
+-----------------+-----------+
|          Hashtag|Occurrences|
+-----------------+-----------+
|          #FIBAWC|       1844|
|#ARGENTINAGOTGAME|        684|
|     #WORLDGOTGAME|        644|
|             #NBA|        586|
|          #PS4LIVE|        169|
+-----------------+-----------+
```

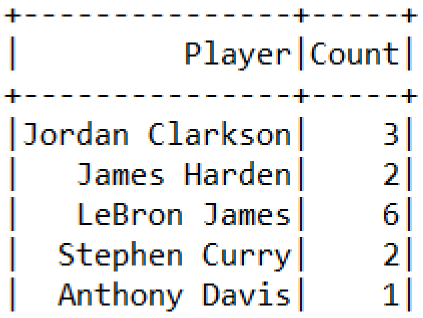


Query10: Player recognition in NBA and the visualization using bar-graph.

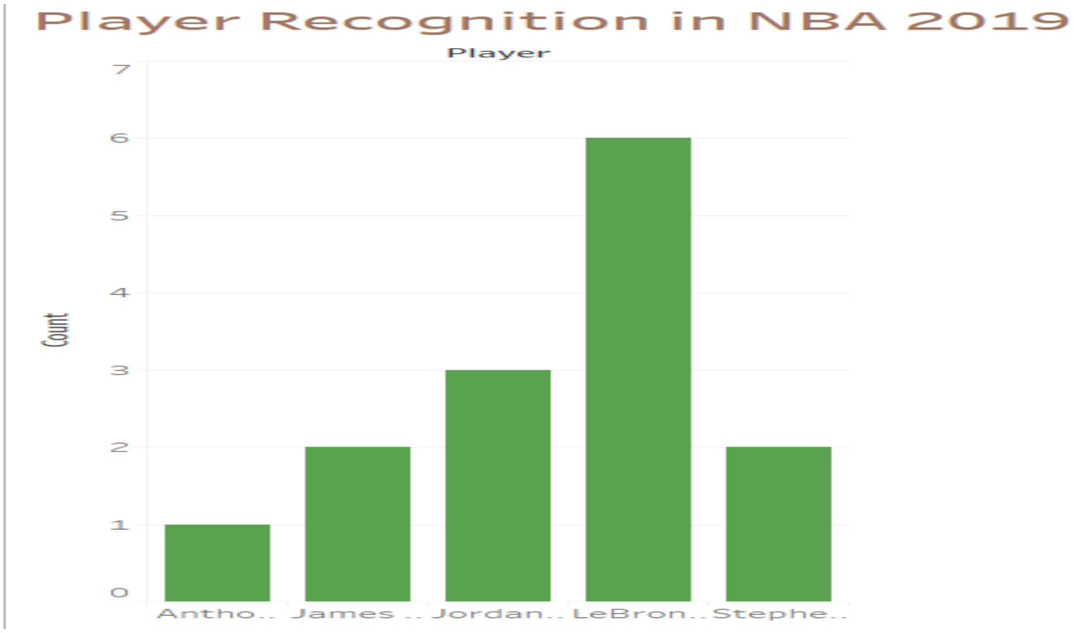
df=spark.read.json("C:/Users/Vamsi Draksharam/PycharmProjects/PB-Vamsi/phase2/data2.json")

```
df.createOrReplaceTempView("nba")
sqlDF = spark.sql("SELECT 'Jordan Clarkson' as Player,
count(*) as Count from nba where text like '%jordan%'
and text like '%nba%'\
        UNION\
        SELECT 'Stephen Curry' as Player, count(*) as Count
from nba where text like '%curry%' and text like
'%nba%'\
        UNION\
        SELECT 'LeBron James' as Player, count(*) as Count
from nba where text like '%lebron%' and text like
'%nba%' UNION\
        SELECT 'James Harden' as Player, count(*) as Count
from nba where text like '%harden%' and text like
'%nba%' UNION\
        SELECT 'Anthony Davis' as Player, count(*) as Count
from nba where text like '%anthony%' and text like
'%nba%'")
pd = sqlDF.toPandas()
pd.to_csv('10.csv', index=False)
```

Output:



```
+---------------+-----+
|         Player|Count|
+---------------+-----+
|Jordan Clarkson|    3|
|   James Harden|    2|
|   LeBron James|    6|
|  Stephen Curry|    2|
|  Anthony Davis|    1|
```


Player Recognition in NBA 2019
Player
Count
Antho..
James ..
Jordan..
LeBron..
Stephe..