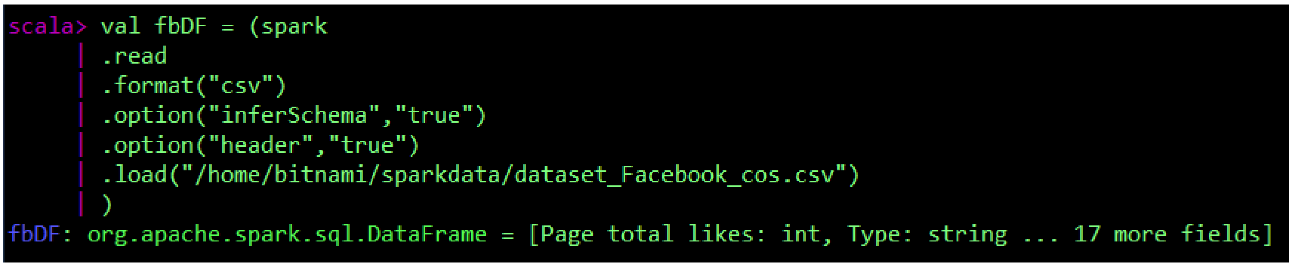
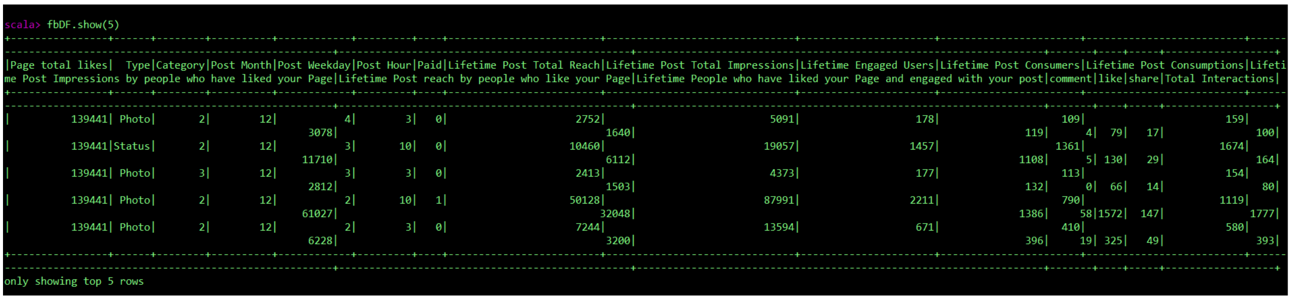
Spark Capstone Project (Facebook Digital Marketing Data)

Loading the data

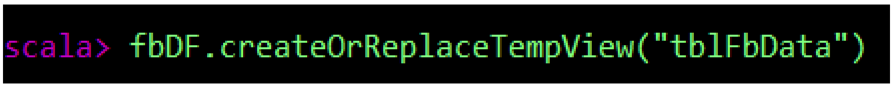


```
scala> val fbDF = (spark
     | .read
     | .format("csv")
     | .option("inferSchema","true")
     | .option("header","true")
     | .load("/home/bitnami/sparkdata/dataset_Facebook_cos.csv")
     | )
fbDF: org.apache.spark.sql.DataFrame = [Page total likes: int, Type: string ... 17 more fields]
```

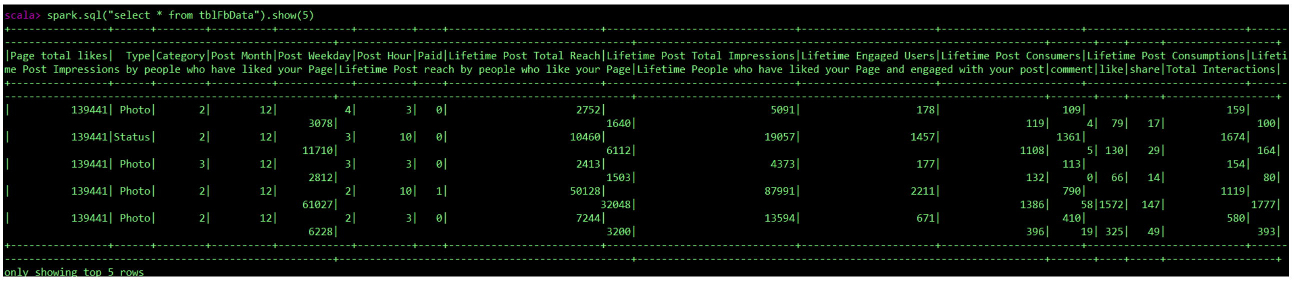


```
scala> fbDF.show(5)
+----------------+------+--------+----------+------------+---------+----+-------------------------+-------------------------------+----------------------+-----------------------+--------------------------+------
----------------------------------------------------+-----------------------------------------------+-------------------------------------------------------------------+-------+----+-----+------------------+
|Page total likes|  Type|Category|Post Month|Post Weekday|Post Hour|Paid|Lifetime Post Total Reach|Lifetime Post Total Impressions|Lifetime Engaged Users|Lifetime Post Consumers|Lifetime Post Consumptions|Lifeti
me Post Impressions by people who have liked your Page|Lifetime Post reach by people who like your Page|Lifetime People who have liked your Page and engaged with your post|comment|like|share|Total Interactions|
+----------------+------+--------+----------+------------+---------+----+-------------------------+-------------------------------+----------------------+-----------------------+--------------------------+------
----------------------------------------------------+-----------------------------------------------+-------------------------------------------------------------------+-------+----+-----+------------------+
|          139441| Photo|       2|        12|           4|        3|   0|                     2752|                           5091|                   178|                    109|                       159|
                                                3078|                                           1640|                                                                119|      4|  79|   17|               100|
|          139441|Status|       2|        12|           3|       10|   0|                    10460|                          19057|                  1457|                   1361|                      1674|
                                               11710|                                           6112|                                                               1108|      5| 130|   29|               164|
|          139441| Photo|       3|        12|           3|        3|   0|                     2413|                           4373|                   177|                    113|                       154|
                                                2812|                                           1503|                                                                132|      0|  66|   14|                80|
|          139441| Photo|       2|        12|           2|       10|   1|                    50128|                          87991|                  2211|                    790|                      1119|
                                               61027|                                          32048|                                                               1386|     58|1572|  147|              1777|
|          139441| Photo|       2|        12|           2|        3|   0|                     7244|                          13594|                   671|                    410|                       580|
                                                6228|                                           3200|                                                                396|     19| 325|   49|               393|
+----------------+------+--------+----------+------------+---------+----+-------------------------+-------------------------------+----------------------+-----------------------+--------------------------+------
----------------------------------------------------+-----------------------------------------------+-------------------------------------------------------------------+-------+----+-----+------------------+
only showing top 5 rows
```

Creating a view



```
scala> fbDF.createOrReplaceTempView("tblFbData")
```

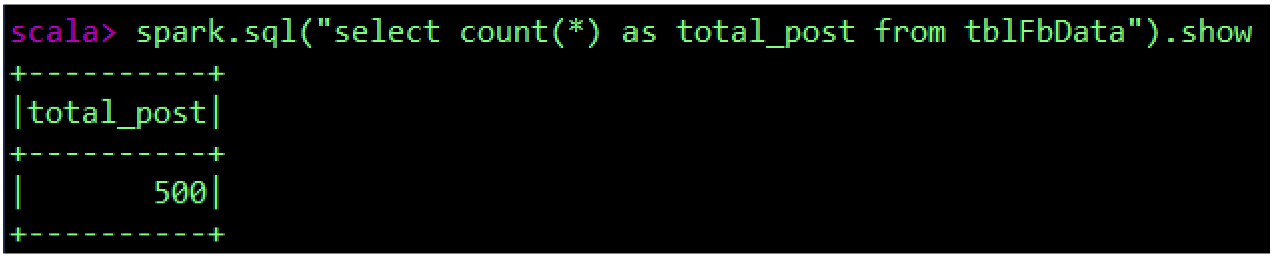


```
scala> spark.sql("select * from tblFbData").show(5)
+----------------+------+--------+----------+------------+---------+----+-------------------------+-------------------------------+----------------------+-----------------------+--------------------------+------
----------------------------------------------------+-----------------------------------------------+-------------------------------------------------------------------+-------+----+-----+------------------+
|Page total likes|  Type|Category|Post Month|Post Weekday|Post Hour|Paid|Lifetime Post Total Reach|Lifetime Post Total Impressions|Lifetime Engaged Users|Lifetime Post Consumers|Lifetime Post Consumptions|Lifeti
me Post Impressions by people who have liked your Page|Lifetime Post reach by people who like your Page|Lifetime People who have liked your Page and engaged with your post|comment|like|share|Total Interactions|
+----------------+------+--------+----------+------------+---------+----+-------------------------+-------------------------------+----------------------+-----------------------+--------------------------+------
----------------------------------------------------+-----------------------------------------------+-------------------------------------------------------------------+-------+----+-----+------------------+
|          139441| Photo|       2|        12|           4|        3|   0|                     2752|                           5091|                   178|                    109|                       159|
                                                3078|                                           1640|                                                                119|      4|  79|   17|               100|
|          139441|Status|       2|        12|           3|       10|   0|                    10460|                          19057|                  1457|                   1361|                      1674|
                                               11710|                                           6112|                                                               1108|      5| 130|   29|               164|
|          139441| Photo|       3|        12|           3|        3|   0|                     2413|                           4373|                   177|                    113|                       154|
                                                2812|                                           1503|                                                                132|      0|  66|   14|                80|
|          139441| Photo|       2|        12|           2|       10|   1|                    50128|                          87991|                  2211|                    790|                      1119|
                                               61027|                                          32048|                                                               1386|     58|1572|  147|              1777|
|          139441| Photo|       2|        12|           2|        3|   0|                     7244|                          13594|                   671|                    410|                       580|
                                                6228|                                           3200|                                                                396|     19| 325|   49|               393|
+----------------+------+--------+----------+------------+---------+----+-------------------------+-------------------------------+----------------------+-----------------------+--------------------------+------
----------------------------------------------------+-----------------------------------------------+-------------------------------------------------------------------+-------+----+-----+------------------+
only showing top 5 rows
```

Find out or solve the following:

1. The total number of posts made

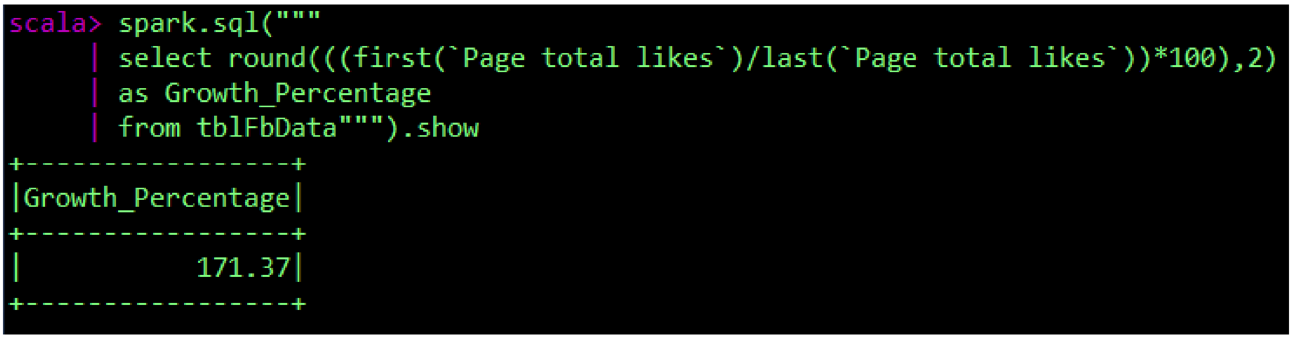
Solution:



```
scala> spark.sql("select count(*) as total_post from tblFbData").show
+----------+
|total_post|
+----------+
|       500|
+----------+
```

2. The percentage of the growth or decline of the page, in terms of likes (subscriptions on the page), from the first post to the latest post

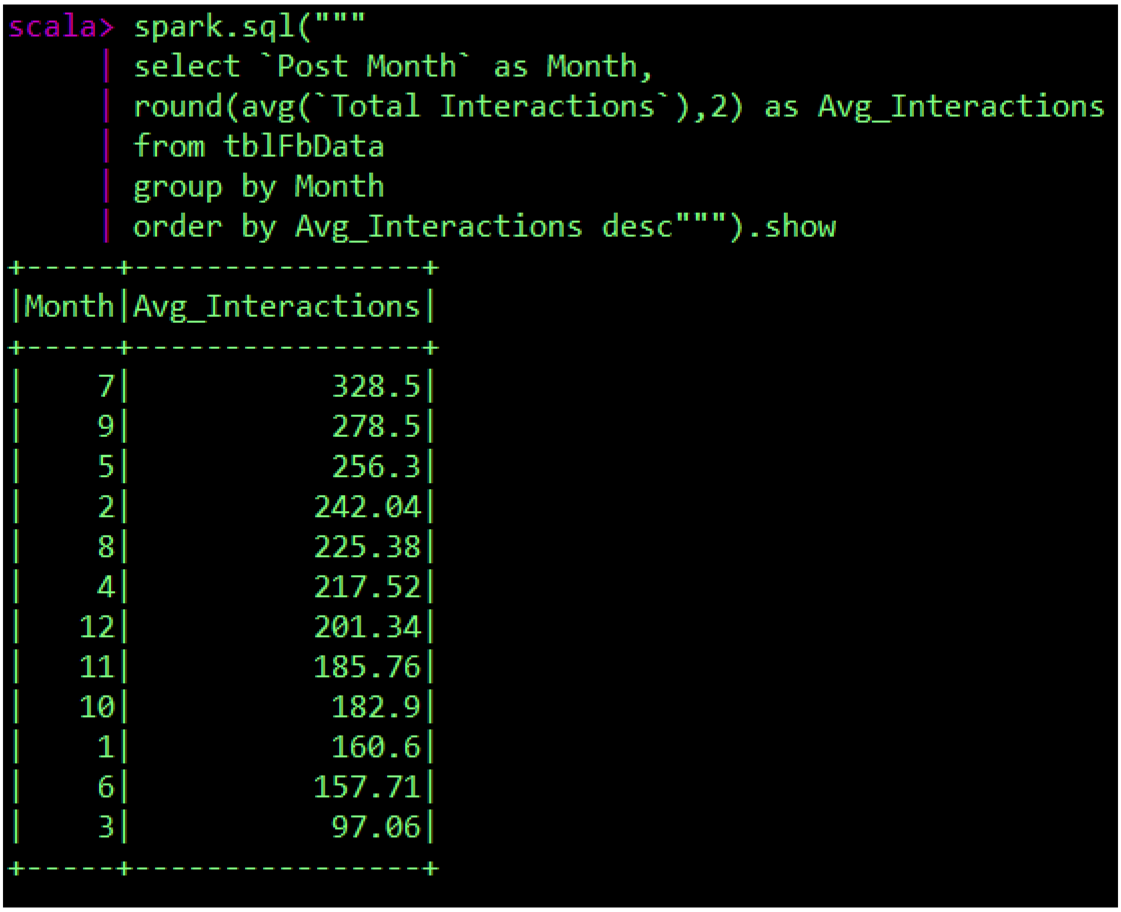
Hint: The first record of the dataset represents the latest post, and the last record of the dataset represents the first post.



```
scala> spark.sql("""
     | select round(((first(`Page total likes`)/last(`Page total likes`))*100),2)
     | as Growth_Percentage
     | from tblFbData""").show
+-----------------+
|Growth_Percentage|
+-----------------+
|           171.37|
+-----------------+
```

3. Which month, on average, has the highest number of post interactions?

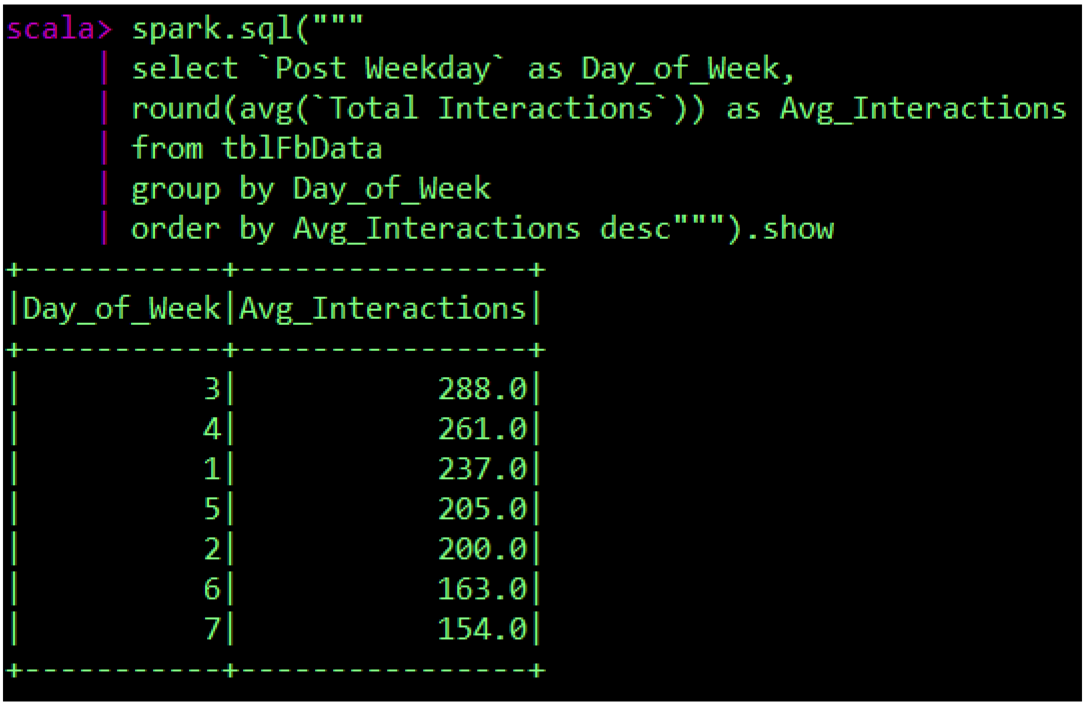
Solution:



```
scala> spark.sql("""
     | select `Post Month` as Month,
     | round(avg(`Total Interactions`),2) as Avg_Interactions
     | from tblFbData
     | group by Month
     | order by Avg_Interactions desc""").show
+-----+----------------+
|Month|Avg_Interactions|
+-----+----------------+
|    7|           328.5|
|    9|           278.5|
|    5|           256.3|
|    2|          242.04|
|    8|          225.38|
|    4|          217.52|
|   12|          201.34|
|   11|          185.76|
|   10|           182.9|
|    1|           160.6|
|    6|          157.71|
|    3|           97.06|
+-----+----------------+
```

4. Which day of the week, on average, has the highest number of post interactions?

Solution:

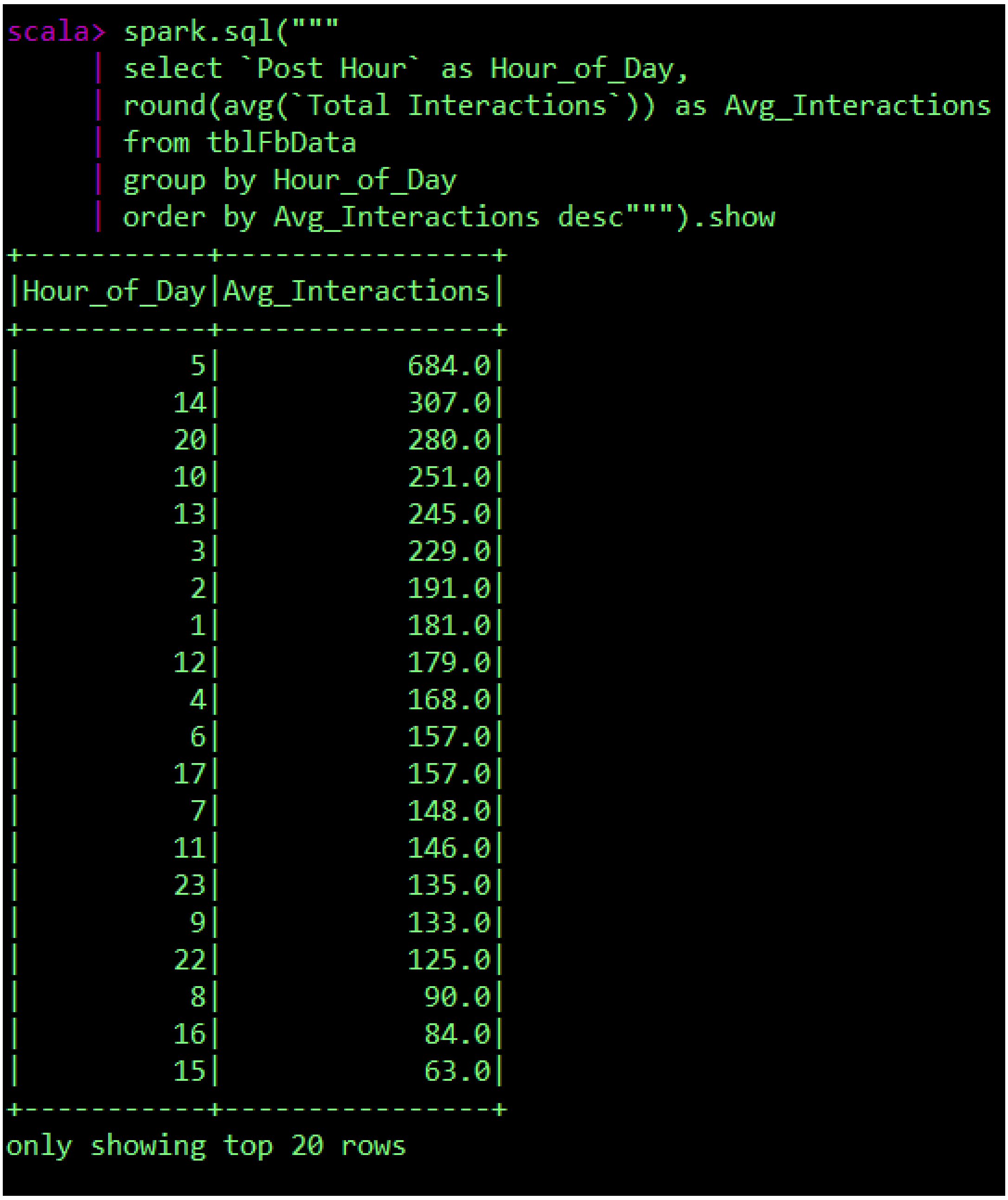


```
scala> spark.sql("""
     | select `Post Weekday` as Day_of_Week,
     | round(avg(`Total Interactions`)) as Avg_Interactions
     | from tblFbData
     | group by Day_of_Week
     | order by Avg_Interactions desc""").show
+-----------+----------------+
|Day_of_Week|Avg_Interactions|
+-----------+----------------+
|          3|           288.0|
|          4|           261.0|
|          1|           237.0|
|          5|           205.0|
|          2|           200.0|
|          6|           163.0|
|          7|           154.0|
+-----------+----------------+
```

5. Which hour of the day, on average, has the highest number of post interactions?

Hint: You can use numbers present in the dataset to define the months, weekdays, and hours in your answer documentation. You don't have to be concerned with naming (e.g., use '12' instead of 'December')

Solution:

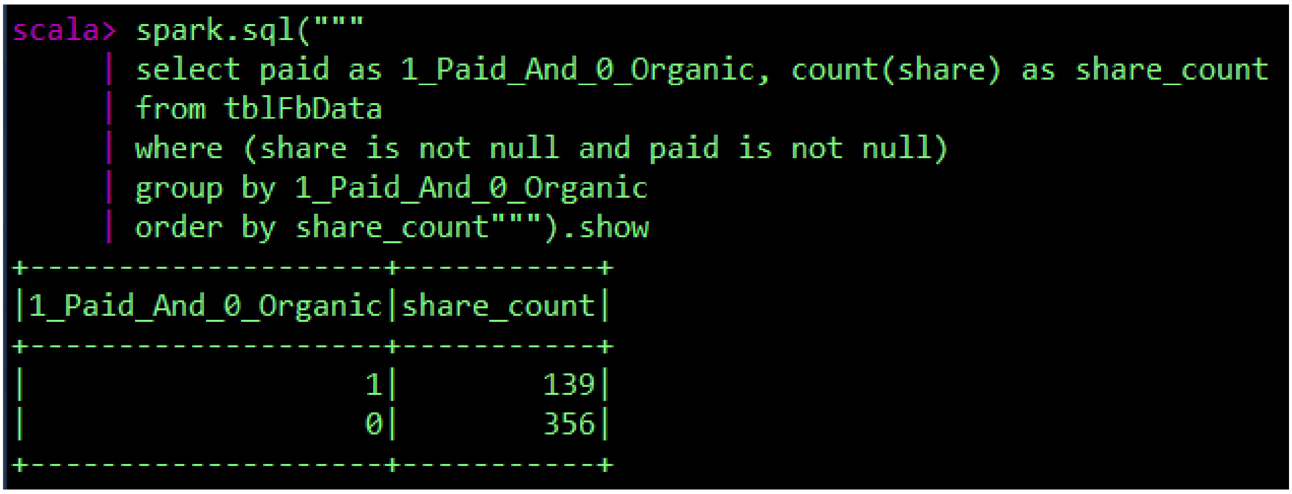


```
scala> spark.sql("""
     | select `Post Hour` as Hour_of_Day,
     | round(avg(`Total Interactions`)) as Avg_Interactions
     | from tblFbData
     | group by Hour_of_Day
     | order by Avg_Interactions desc""").show
+-----------+----------------+
|Hour_of_Day|Avg_Interactions|
+-----------+----------------+
|          5|           684.0|
|         14|           307.0|
|         20|           280.0|
|         10|           251.0|
|         13|           245.0|
|          3|           229.0|
|          2|           191.0|
|          1|           181.0|
|         12|           179.0|
|          4|           168.0|
|          6|           157.0|
|         17|           157.0|
|          7|           148.0|
|         11|           146.0|
|         23|           135.0|
|          9|           133.0|
|         22|           125.0|
|          8|            90.0|
|         16|            84.0|
|         15|            63.0|
+-----------+----------------+
only showing top 20 rows
```

6. Determine if paid (promoted) posts have a higher correlation with a large number of post shares when compared to the post shares of organic (non-promoted) posts.

This is to determine the commercial viability of investing in paid posts for promoting cosmetic products. Answer with either a Yes or a No, and provide the methodology of how you reached your conclusion

Solution:



```
scala> spark.sql("""
     | select paid as 1_Paid_And_0_Organic, count(share) as share_count
     | from tblFbData
     | where (share is not null and paid is not null)
     | group by 1_Paid_And_0_Organic
     | order by share_count""").show
+--------------------+-----------+
|1_Paid_And_0_Organic|share_count|
+--------------------+-----------+
|                   1|        139|
|                   0|        356|
+--------------------+-----------+
```

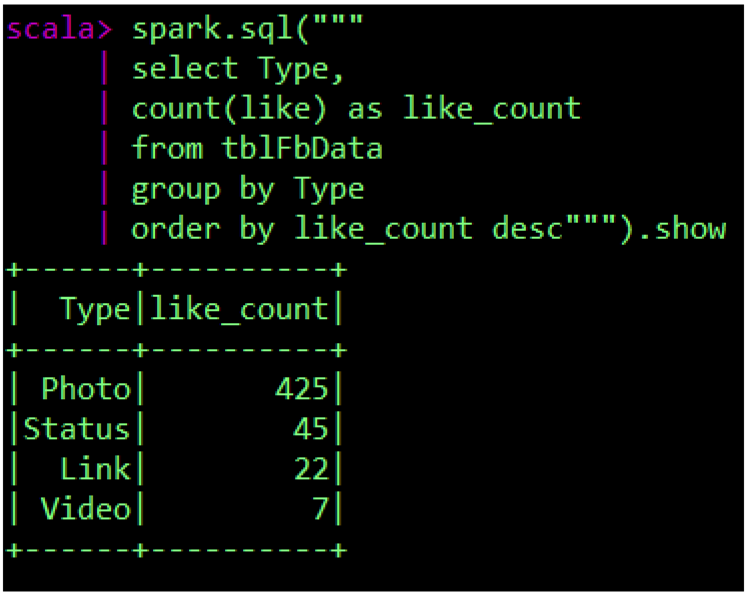
Yes.

From the above statistics it can be inferred that there is a correlation between whether a post is paid or not and the number of posts shared.

It can be seen that organic posts have a higher share count when compared to posts that were paid.

7. Which post type (photo, video, status, or link) is the most attractive to people who have subscribed to your page (people who have liked the page)?

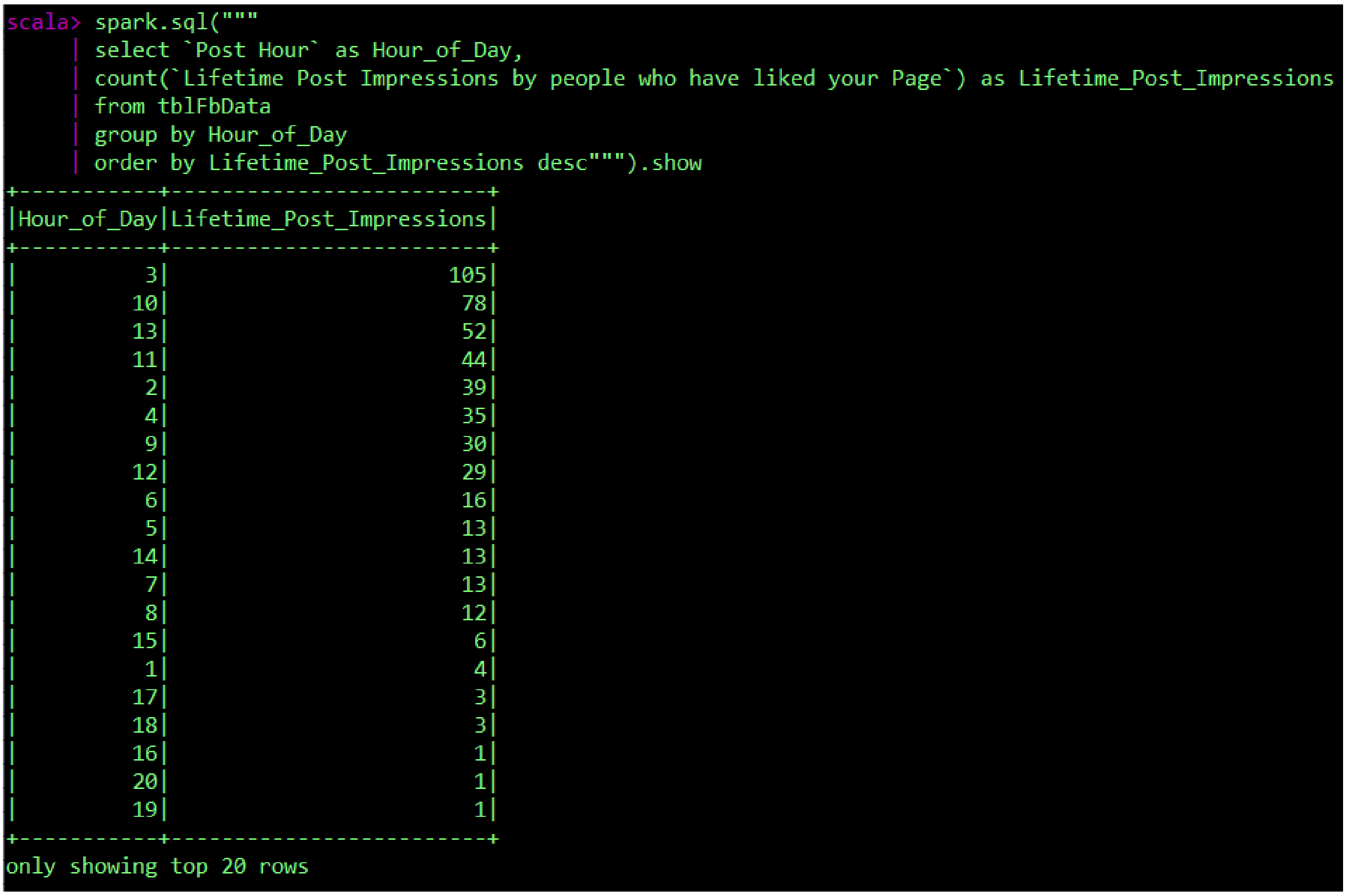
Solution:



```
scala> spark.sql("""
     | select Type,
     | count(like) as like_count
     | from tblFbData
     | group by Type
     | order by like_count desc""").show
+------+----------+
|  Type|like_count|
+------+----------+
| Photo|       425|
|Status|        45|
|  Link|        22|
| Video|         7|
+------+----------+
```

8. Which hour of the day is ideal for posting photographic content? Arrange the hours of the day according to the order of the Lifetime Post Impressions column?

Solution:

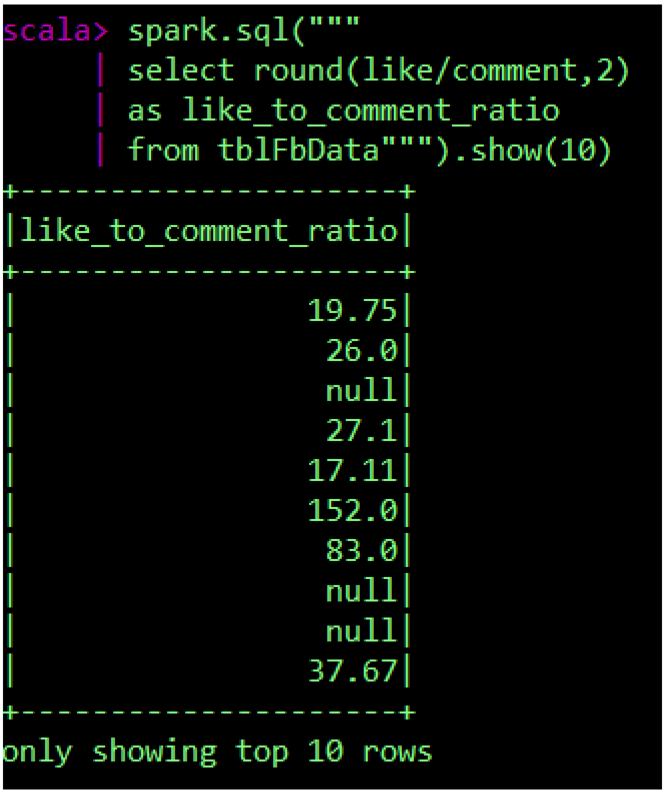


```
scala> spark.sql("""
     | select `Post Hour` as Hour_of_Day,
     | count(`Lifetime Post Impressions by people who have liked your Page`) as Lifetime_Post_Impressions
     | from tblFbData
     | group by Hour_of_Day
     | order by Lifetime_Post_Impressions desc""").show
+-----------+-------------------------+
|Hour_of_Day|Lifetime_Post_Impressions|
+-----------+-------------------------+
|          3|                      105|
|         10|                       78|
|         13|                       52|
|         11|                       44|
|          2|                       39|
|          4|                       35|
|          9|                       30|
|         12|                       29|
|          6|                       16|
|          5|                       13|
|         14|                       13|
|          7|                       13|
|          8|                       12|
|         15|                        6|
|          1|                        4|
|         17|                        3|
|         18|                        3|
|         16|                        1|
|         20|                        1|
|         19|                        1|
+-----------+-------------------------+
only showing top 20 rows
```

9. Create an additional column with the name Likes-to-comment Ratio, with the column values having the equation:
likes to comment ratio = like / comment

Hint: Make sure the ratio is in a decimal format, and correct it to 2 decimal places

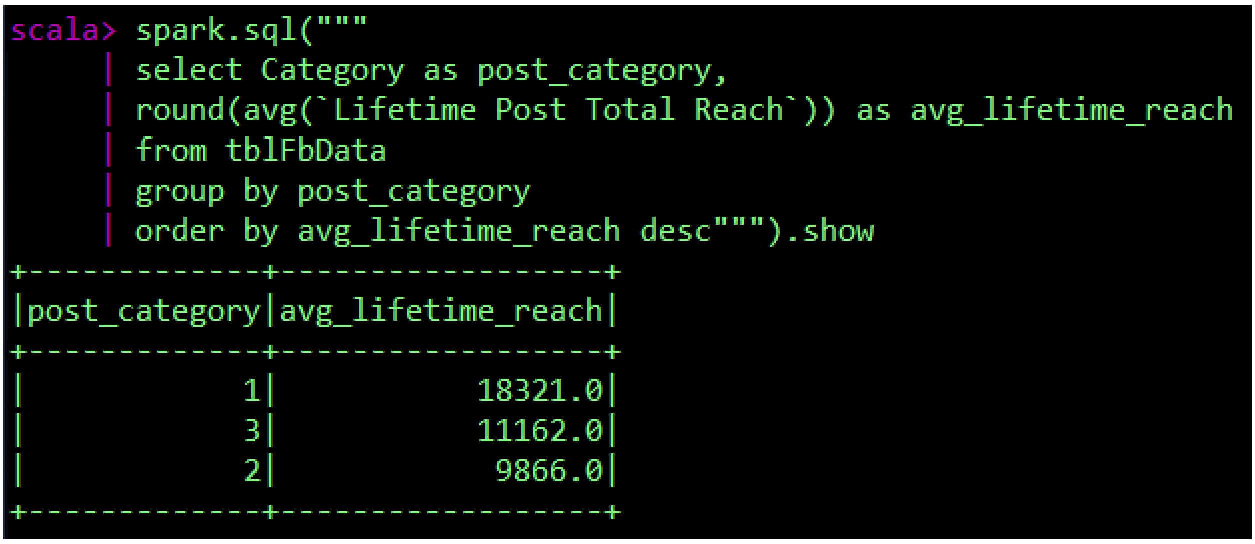
Solution:



```
scala> spark.sql("""
     | select round(like/comment,2)
     | as like_to_comment_ratio
     | from tblFbData""").show(10)
+---------------------+
|like_to_comment_ratio|
+---------------------+
|                19.75|
|                 26.0|
|                 null|
|                 27.1|
|                17.11|
|                152.0|
|                 83.0|
|                 null|
|                 null|
|                37.67|
+---------------------+
only showing top 10 rows
```

10. Arrange post categories (1,2,3) in the descending order of the reach that they can accumulate on average

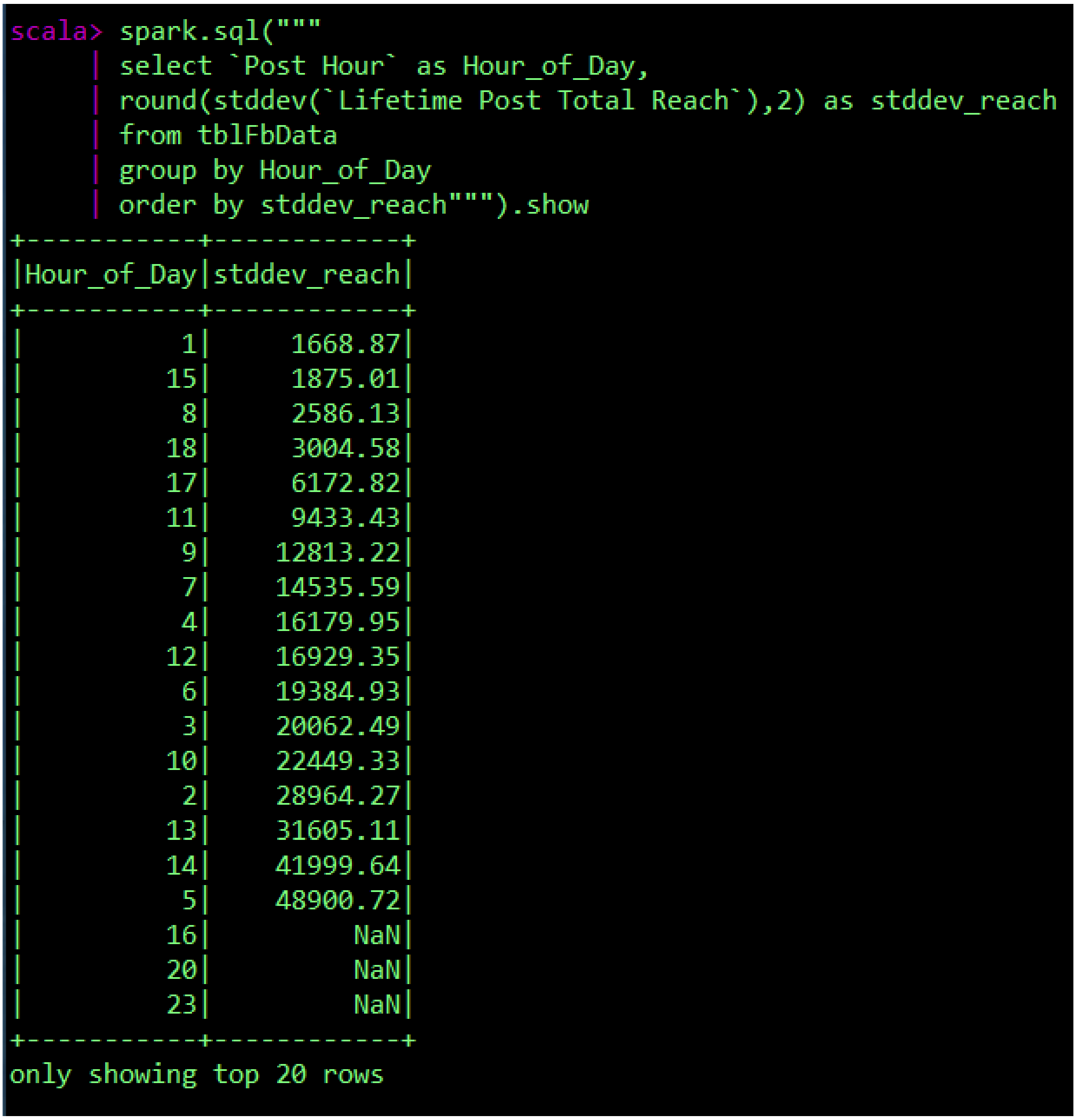
Solution:



```
scala> spark.sql("""
     | select Category as post_category,
     | round(avg(`Lifetime Post Total Reach`)) as avg_lifetime_reach
     | from tblFbData
     | group by post_category
     | order by avg_lifetime_reach desc""").show
+-------------+------------------+
|post_category|avg_lifetime_reach|
+-------------+------------------+
|            1|           18321.0|
|            3|           11162.0|
|            2|            9866.0|
+-------------+------------------+
```

11. Determine the standard deviation of the average post reach for each of the day hours. This is to determine if the time of the day is an ideal criterion to identify when to create posts

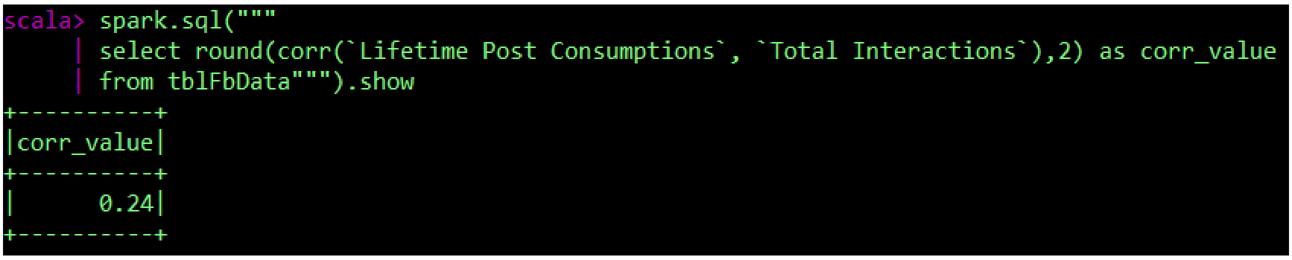
Solution:



```
scala> spark.sql("""
     | select `Post Hour` as Hour_of_Day,
     | round(stddev(`Lifetime Post Total Reach`),2) as stddev_reach
     | from tblFbData
     | group by Hour_of_Day
     | order by stddev_reach""").show
+-----------+------------+
|Hour_of_Day|stddev_reach|
+-----------+------------+
|          1|     1668.87|
|         15|     1875.01|
|          8|     2586.13|
|         18|     3004.58|
|         17|     6172.82|
|         11|     9433.43|
|          9|    12813.22|
|          7|    14535.59|
|          4|    16179.95|
|         12|    16929.35|
|          6|    19384.93|
|          3|    20062.49|
|         10|    22449.33|
|          2|    28964.27|
|         13|    31605.11|
|         14|    41999.64|
|          5|    48900.72|
|         16|         NaN|
|         20|         NaN|
|         23|         NaN|
+-----------+------------+
only showing top 20 rows
```

12. Is there any correlation between the number of post consumptions and the total interactions on the post?

Solution:

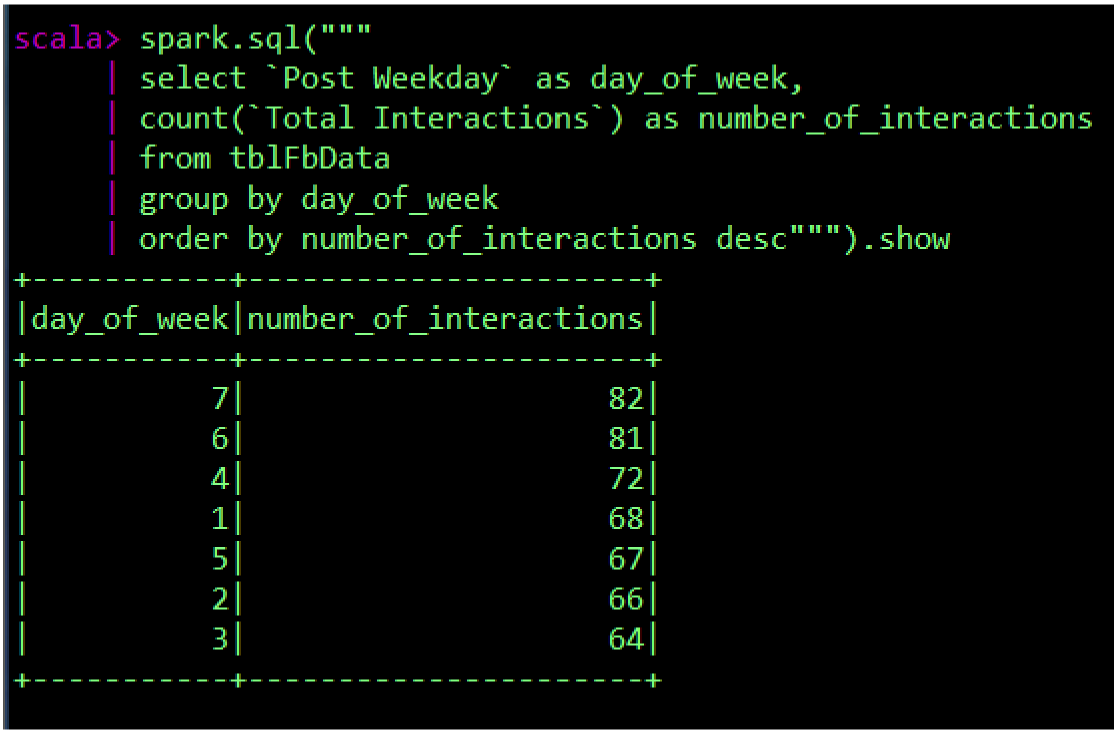


```
scala> spark.sql("""
     | select round(corr(`Lifetime Post Consumptions`, `Total Interactions`),2) as corr_value
     | from tblFbData""").show
+----------+
|corr_value|
+----------+
|      0.24|
+----------+
```

13. Determine the two best days in a week to create posts, when people are extremely active on social media, based on the data that you have

Hint: Question 13 can have a subjective answer. You are free to choose your own approach to determine the best days to post in a week. Make sure to validate your claims with the relevant code and explanation of your approach.

Solution:



```
scala> spark.sql("""
     | select `Post Weekday` as day_of_week,
     | count(`Total Interactions`) as number_of_interactions
     | from tblFbData
     | group by day_of_week
     | order by number_of_interactions desc""").show
+-----------+----------------------+
|day_of_week|number_of_interactions|
+-----------+----------------------+
|          7|                    82|
|          6|                    81|
|          4|                    72|
|          1|                    68|
|          5|                    67|
|          2|                    66|
|          3|                    64|
+-----------+----------------------+
```

Concluding from the above statistics, we can see that on the 7th and 6th days of the week there are significantly more number of post interactions that occur.

That makes 7th and 6th days as the two best days in a week to create posts since there is significantly more likelihood of the post reaching and being seen by users compared to rest of the days in a week.