7_data_wrangling-sql

July 11, 2021

1 Spark SQL Examples

Run the code cells below. This is the same code from the previous screencast.

```
In [1]: from pyspark.sql import SparkSession
        from pyspark.sql.functions import udf
        from pyspark.sql.types import StringType
        from pyspark.sql.types import IntegerType
        from pyspark.sql.functions import desc
        from pyspark.sql.functions import asc
        from pyspark.sql.functions import sum as Fsum

        import datetime

        import numpy as np
        import pandas as pd
        %matplotlib inline
        import matplotlib.pyplot as plt

In [2]: spark = SparkSession \
            .builder \
            .appName("Data wrangling with Spark SQL") \
            .getOrCreate()

In [3]: path = "data/sparkify_log_small.json"
        user_log = spark.read.json(path)

In [4]: user_log.take(1)

Out[4]: [Row(artist='Showaddywaddy', auth='Logged In', firstName='Kenneth', gender='M', itemInSe

In [5]: user_log.printSchema()

root
 |-- artist: string (nullable = true)
 |-- auth: string (nullable = true)
 |-- firstName: string (nullable = true)
 |-- gender: string (nullable = true)
```

```
 |-- itemInSession: long (nullable = true)
 |-- lastName: string (nullable = true)
 |-- length: double (nullable = true)
 |-- level: string (nullable = true)
 |-- location: string (nullable = true)
 |-- method: string (nullable = true)
 |-- page: string (nullable = true)
 |-- registration: long (nullable = true)
 |-- sessionId: long (nullable = true)
 |-- song: string (nullable = true)
 |-- status: long (nullable = true)
 |-- ts: long (nullable = true)
 |-- userAgent: string (nullable = true)
 |-- userId: string (nullable = true)
```

2 Create a View And Run Queries

The code below creates a temporary view against which you can run SQL queries.

```
In [6]: user_log.createOrReplaceTempView("user_log_table")
```

```
In [7]: spark.sql("SELECT * FROM user_log_table LIMIT 2").show()
```

```
+-------------+---------+---------+------+-------------+--------+---------+-----+--------------
|       artist|     auth|firstName|gender|itemInSession|lastName|   length|level|           loc
+-------------+---------+---------+------+-------------+--------+---------+-----+--------------
|Showaddywaddy|Logged In|  Kenneth|     M|          112|Matthews|232.93342| paid|Charlotte-Conco
|   Lily Allen|Logged In|Elizabeth|     F|            7|   Chase|195.23873| free|Shreveport-Boss
+-------------+---------+---------+------+-------------+--------+---------+-----+--------------
```

```
In [8]: spark.sql('''
                  SELECT *
                  FROM user_log_table
                  LIMIT 2
                  '''
                  ).show()
```

```
+-------------+---------+---------+------+-------------+--------+---------+-----+--------------
|       artist|     auth|firstName|gender|itemInSession|lastName|   length|level|           loc
+-------------+---------+---------+------+-------------+--------+---------+-----+--------------
|Showaddywaddy|Logged In|  Kenneth|     M|          112|Matthews|232.93342| paid|Charlotte-Conco
|   Lily Allen|Logged In|Elizabeth|     F|            7|   Chase|195.23873| free|Shreveport-Boss
+-------------+---------+---------+------+-------------+--------+---------+-----+--------------
```

```
In [9]: spark.sql('''
                  SELECT COUNT(*)
                  FROM user_log_table
                  '''
                  ).show()

+--------+
|count(1)|
+--------+
|   10000|
+--------+
```

```
In [10]: spark.sql('''
                   SELECT userID, firstname, page, song
                   FROM user_log_table
                   WHERE userID == '1046'
                   '''
                   ).collect()

Out[10]: [Row(userID='1046', firstname='Kenneth', page='NextSong', song='Christmas Tears Will Fa
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Be Wary Of A Woman'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Public Enemy No.1'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Reign Of The Tyrants'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Father And Son'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='No. 5'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Seventeen'),
          Row(userID='1046', firstname='Kenneth', page='Home', song=None),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='War on war'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Killermont Street'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Black & Blue'),
          Row(userID='1046', firstname='Kenneth', page='Logout', song=None),
          Row(userID='1046', firstname='Kenneth', page='Home', song=None),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Heads Will Roll'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Bleed It Out [Live At M
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Clocks'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Love Rain'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song="Ry Ry's Song (Album Ver
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='The Invisible Man'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Catch You Baby (Steve P
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Ask The Mountains'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Given Up (Album Version
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='El Cuatrero'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Hero/Heroine'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Spring'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Rising Moon'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Tough Little Boys'),
```

```
          Row(userID='1046', firstname='Kenneth', page='NextSong', song="Qu'Est-Ce Que T'Es Bell
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Secrets'),
          Row(userID='1046', firstname='Kenneth', page='NextSong', song='Under The Gun')]
```

```
In [11]: spark.sql('''
                    SELECT DISTINCT page
                    FROM user_log_table
                    ORDER BY page ASC
                    '''
                    ).show()
```

```
+----------------+
|            page|
+----------------+
|           About|
|       Downgrade|
|           Error|
|            Help|
|            Home|
|           Login|
|          Logout|
|        NextSong|
|   Save Settings|
|        Settings|
|Submit Downgrade|
|  Submit Upgrade|
|         Upgrade|
+----------------+
```

3 User Defined Functions

```
In [13]: spark.udf.register("get_hour", lambda x: int(datetime.datetime.fromtimestamp(x / 1000.0
```

```
Out[13]: <function __main__.<lambda>(x)>
```

```
In [14]: spark.sql('''
                    SELECT *, get_hour(ts) AS hour
                    FROM user_log_table
                    LIMIT 1
                    '''
                    ).collect()
```

```
Out[14]: [Row(artist='Showaddywaddy', auth='Logged In', firstName='Kenneth', gender='M', itemInS
```

```
In [15]: songs_in_hour = spark.sql('''
                    SELECT get_hour(ts) AS hour, COUNT(*) as plays_per_hour
```

```
                    FROM user_log_table
                    WHERE page = "NextSong"
                    GROUP BY hour
                    ORDER BY cast(hour as int) ASC
                    '''
                    )
```

```
In [16]: songs_in_hour.show()
```

```
+----+--------------+
|hour|plays_per_hour|
+----+--------------+
|   0|           456|
|   1|           454|
|   2|           382|
|   3|           302|
|   4|           352|
|   5|           276|
|   6|           348|
|   7|           358|
|   8|           375|
|   9|           249|
|  10|           216|
|  11|           228|
|  12|           251|
|  13|           339|
|  14|           462|
|  15|           479|
|  16|           484|
|  17|           430|
|  18|           362|
|  19|           295|
+----+--------------+
only showing top 20 rows
```

4 Converting Results to Pandas

```
In [17]: songs_in_hour_pd = songs_in_hour.toPandas()
```

```
In [18]: print(songs_in_hour_pd)
```

```
   hour  plays_per_hour
0     0             456
1     1             454
2     2             382
3     3             302
```

```
4      4              352
5      5              276
6      6              348
7      7              358
8      8              375
9      9              249
10    10              216
11    11              228
12    12              251
13    13              339
14    14              462
15    15              479
16    16              484
17    17              430
18    18              362
19    19              295
20    20              257
21    21              248
22    22              369
23    23              375
```

In [ ]: