# BigQuery Storage & Spark DataFrames

November 24, 2024

## 0.1 Checking the scalar version

[11]:

```
!scala -version
```

```
Scala code runner version 2.12.10 -- Copyright 2002-2019, LAMP/EPFL and
Lightbend, Inc.
```

## 0.2 Creating the Spark Session

[12]:

```
from pyspark.sql import SparkSession

spark = SparkSession.builder \
    .appName('BigQuery Storage & Spark DataFrames') \
    .config('spark.jars.packages', 'com.google.cloud.spark:
 ↪spark-bigquery-with-dependencies_2.12:0.15.1-beta') \
    .getOrCreate()
```

### 0.2.1 Enabling repl.eagerEval

[13]:

```
spark.conf.set("spark.sql.repl.eagerEval.enabled",True)
```

## 0.3 Reading the BigQuery table into Spark DataFrame

Use filter() to query data from a partitioned table.

[14]:

```
table = "bigquery-public-data.wikipedia.pageviews_2020"
df_wiki_pageviews = spark.read \
  .format("bigquery") \
  .option("table", table) \
  .option("filter", "datehour >= '2020-03-01' AND datehour < '2020-03-02'") \
  .load()

df_wiki_pageviews.printSchema()
```

```
root
 |-- datehour: timestamp (nullable = true)
 |-- wiki: string (nullable = true)
 |-- title: string (nullable = true)
 |-- views: long (nullable = true)
```

Select required columns and apply a filter using where() which is an alias for filter() then cache the table

[15]:

```
df_wiki_en = df_wiki_pageviews \
  .select("title", "wiki", "views") \
  .where("views > 1000 AND wiki in ('en', 'en.m')") \
  .cache()

df_wiki_en
```

[15]:

```
+--------------------+----+------+
|               title|wiki| views|
+--------------------+----+------+
|                   -|  en|143159|
|                   -|  en| 14969|
|                   -|  en|186802|
|                   -|  en|131686|
|                   -|  en|213787|
|                   -|  en|211910|
|                   -|  en|186675|
|                   -|  en| 21901|
|                   -|  en|163710|
|                   -|  en| 23527|
|                   -|  en|202621|
|                   -|  en|110524|
|                   -|  en|220543|
|12_Angry_Men_(195…|  en|  1124|
|                   -|  en|195339|
|                   -|  en|151283|
|                   -|  en| 22490|
|                   -|  en|182985|
|                   -|  en| 45182|
|                   -|  en|153327|
+--------------------+----+------+
only showing top 20 rows
```

Grouping by title and order by page views to see the top pages

[16]:

```
import pyspark.sql.functions as F

df_wiki_en_totals = df_wiki_en \
.groupBy("title") \
.agg(F.sum('views').alias('total_views'))

df_wiki_en_totals.orderBy('total_views', ascending=False)
```

```
[16]: +--------------------+-----------+
      |               title|total_views|
      +--------------------+-----------+
      |           Main_Page|   10939337|
      |United_States_Senate|    5619797|
      |                   -|    3852360|
      |      Special:Search|    1538334|
      |2019-20_coronavir…|     407042|
      |2020_Democratic_P…|     260093|
      |         Coronavirus|     254861|
      |The_Invisible_Man…|     233718|
      |       Super_Tuesday|     201077|
      |         Colin_McRae|     200219|
      |         David_Byrne|     189989|
      |2019-20_coronavir…|     156803|
      |        John_Mulaney|     155605|
      |2020_South_Caroli…|     152137|
      |      AEW_Revolution|     140503|
      |       Boris_Johnson|     120957|
      |          Tom_Steyer|     120926|
      |Dyatlov_Pass_inci…|     117704|
      |         Spanish_flu|     108335|
      |2020_coronavirus_…|     107653|
      +--------------------+-----------+
      only showing top 20 rows
```

### 0.4 Writing Spark Dataframe to BigQuery table

```
[18]: # Update to your GCS bucket
      gcs_bucket = 'amali_st_bucket1'

      # Update to your BigQuery dataset name you created
      bq_dataset = 'week_4_lab_dataset'

      # Enter BigQuery table name you want to create or overwite.
      # If the table does not exist it will be created when you run the write function
      bq_table = 'wiki_total_pageviews'

      df_wiki_en_totals.write \
        .format("bigquery") \
        .option("table","{}.{}".format(bq_dataset, bq_table)) \
        .option("temporaryGcsBucket", gcs_bucket) \
        .mode('overwrite') \
        .save()
```

## 0.5 Using BigQuery magic to query table

[20]:
```
%%bigquery
SELECT title, total_views
FROM week_4_lab_dataset.wiki_total_pageviews
ORDER BY total_views DESC
LIMIT 10
```

[20]:
```
                                          title  total_views
0                                     Main_Page     10939337
1                          United_States_Senate      5619797
2                                             -      3852360
3                                Special:Search      1538334
4                  2019-20_coronavirus_outbreak       407042
5  2020_Democratic_Party_presidential_primaries       260093
6                                   Coronavirus       254861
7                 The_Invisible_Man_(2020_film)       233718
8                                 Super_Tuesday       201077
9                                   Colin_McRae       200219
```

[1]:
```
%%bigquery
SELECT title, total_views
FROM week_4_lab_dataset.wiki_total_pageviews
WHERE title LIKE '%United%'
ORDER BY total_views DESC
LIMIT 10
```

[1]:
```
                                               title  total_views
0                               United_States_Senate      5619797
1                                      United_States        17879
2           2020_United_States_presidential_election        17364
3                             Manchester_United_F.C.         3671
4  Third_Amendment_to_the_United_States_Constitution         2330
5            List_of_presidents_of_the_United_States         2153
6           2016_United_States_presidential_election         2018
7  List_of_amendments_to_the_United_States_Consti…         1811
8                Surgeon_General_of_the_United_States         1406
```