HOMEWORK 09

- Anirudh Narayanan(an9425)

README:

RUN HW09_an9425.py after setting the proper path for the specified tsv files.

```
# Get data from tsv files
name_basics = spark.read.format("csv").option("delimiter", "\t").option("header", True).load(
    "/Users/anirudhramesh/Desktop/INTRO_TO_BIGDATA/ASSN9/Data/name_basics.tsv")
title_basics = spark.read.format("csv").option("delimiter", "\t").option("header", True).load(
    "/Users/anirudhramesh/Desktop/INTRO_TO_BIGDATA/ASSN9/Data/title_basics.tsv")
title_principals = spark.read.format("csv").option("delimiter", "\t").option("header", True).load(
    "/Users/anirudhramesh/Desktop/INTRO_TO_BIGDATA/ASSN9/Data/title_principals.tsv")
```

PYSPARK QUERIES AND OUTPUT:

1.

```
# QUERY 1
# Alive actors whose name starts with “Phi” and did not participate in any movie in 2014.
s1 = time.time()
q1_result = name_basics.join(title_principals, on="nconst", how="inner") \
    .select("primaryName", "nconst", "tconst", "primaryProfession", "deathYear") \
    .filter(((col("deathYear") == 'Null') | (col("deathYear") == '\\N')) &
            (col("primaryName").startswith('Phi')) & (col("primaryProfession")).contains('actor'))


q1_resultfinal = q1_result.join(title_basics, on="tconst", how="inner") \
    .select("primaryName", "startYear", "primaryTitle") \
    .filter((col("startYear").cast("int") != 2014))
q1_resultfinal.show(10)
print("Time taken for Query 1: " + str(time.time() - s1) + " seconds.")
```

Output:

```
+------------------+---------+--------------------+
|       primaryName|startYear|        primaryTitle|
+------------------+---------+--------------------+
|      Phil Proctor|     1983|Nick Danger in th...|
|       Phil Sumner|     1985|    One Summer Again|
|Philip J. Spinelli|     1986|Hovering Over the...|
|   Phillip Connery|     1994|     Fast Getaway II|
|      Phil Katzman|     1997|         Mr. Vincent|
|      Philip Quast|     1985|            Emoh Ruo|
|      Philippe Bas|     1999|Les coquelicots s...|
|      Phil Forrest|     1985|        Submit to Me|
|    Phil Ridarelli|     2000|You Don't Know Ja...|
|  Philippe Bréjean|     1979|Monique et Julie,...|
+------------------+---------+--------------------+
only showing top 10 rows

Time taken for Query 1: 31.565123081207275 seconds.
```

2.

```
# QUERY 2
# Producers who have produced the most talk shows in 2017 and whose name contains “Gill”.
s2 = time.time()
q2_result = name_basics.join(title_principals, on="nconst") \
    .select("nconst", "tconst", "primaryName").distinct() \
    .filter((col("primaryName").contains("Gill")) &
            (col("category") == "producer"))

q2_resultfinal = q2_result.join(title_basics, on="tconst") \
    .select("nconst", "tconst", "primaryName", "startYear", split("genres", ",").alias("Genre_array")) \
    .filter((col("startYear") == "2017"))

q2_output = q2_resultfinal.filter(array_contains(col("Genre_array"), "Talk-Show"))

q2_output2 = q2_output.groupBy(col("primaryName")) \
    .agg(count(col("tconst")).alias("Count")) \
    .sort(desc("Count"))

q2_output2.show(10)
print("Time taken for Query 2: " + str(time.time() - s2) + " seconds.")
```

Output:

```
+----------------+-----+
|     primaryName|Count|
+----------------+-----+
|       Ryan Gill|   81|
|Dominic Gillette|   73|
|Corinne Gilliard|   14|
|      Shane Gill|   13|
|   Gilles Bérard|    1|
+----------------+-----+


Time taken for Query 2: 30.456218242645264 seconds.
```

3.

```
# QUERY 3
# Alive producers with the greatest number of long-run titles produced (runtime greater than 120 minutes).
s3 = time.time()
q3_result = name_basics.join(title_principals, on="nconst") \
    .select("tconst", "nconst", "primaryProfession", "primaryName") \
    .filter(((col("deathYear") == 'Null') | (col("deathYear") == '\\N')) &
            (col("primaryProfession").contains('producer')) & (col("category") == "producer"))

q3_result2 = q3_result.join(title_basics, on="tconst") \
    .select("tconst", "primaryName") \
    .filter((col("runtimeMinutes").cast("int") > 120)).groupBy("primaryName") \
    .agg(count(col("tconst")).alias("Counts")) \
    .sort(desc("Counts"))

q3_result2.show(10)
print("Time taken for Query 2: " + str(time.time() - s3) + " seconds.")
```

Output:

```
+------------------+------+
|       primaryName|Counts|
+------------------+------+
|      Acun Ilicali|   241|
|        Bree Mills|   183|
|     Maxwell James|   176|
|   Wade Baverstock|   143|
|     Vince McMahon|   141|
|John Michael Flynn|   114|
|Christopher Lockey|   114|
|        Kyle Shire|   103|
|         Efe Irvül|   102|
|      Nick Rylance|   102|
+------------------+------+
only showing top 10 rows

Time taken for Query 3: 21.854208946228027 seconds.
```

4.

```
# QUERY 4
# Alive actors who have portrayed Jesus Christ (look for both words independently).
s4 = time.time()
q4_result = name_basics.join(title_principals, on="nconst") \
    .select("primaryName", "characters") \
    .filter(((col("deathYear") == 'Null') | (col("deathYear") == '\\N')) &
            (col("primaryProfession").contains('actor')) & (col("category") == "actor") \
            & (col("characters").like("% Jesus %") |
               (col("characters").like("% Christ %"))))

q4_result.show(10)
print("Time taken for Query 2: " + str(time.time() - s4) + " seconds.")
```

Output:

```
+----------------+--------------------+
|     primaryName|          characters|
+----------------+--------------------+
|  Stephen Kunken|["James Jesus Ang...|
|  Stephen Kunken|["James Jesus Ang...|
|Martin A. Molina|["Dr. Jesus Chris...|
|Martin A. Molina|["Dr. Jesus Chris...|
|   Héctor Holten|["Agente de la Gr...|
| Skully Shemwell|["Jesus Christ Ju...|
|   Jesus Dobbins|["Plaintiff Jesus...|
|      John Joyce| ["The Jesus Freak"]|
| François Brunet|["1er Photographe...|
|   Rafael Torres|["Jesus Christ in...|
+----------------+--------------------+
only showing top 10 rows
```