databricks ASP 4.3L - De-Duping Data Lab

De-Duping Data Lab

In this exercise, we're doing ETL on a file we've received from a customer. That file contains data about people, including:

- first, middle and last names
- gender
- birth date
- Social Security number
- salary

But, as is unfortunately common in data we get from this customer, the file contains some duplicate records. Worse:

- In some of the records, the names are mixed case (e.g., "Carol"), while in others, they are uppercase (e.g., "CAROL").
- The Social Security numbers aren't consistent either. Some of them are hyphenated (e.g., "992-83-4829"), while others are missing hyphens ("992834829").

If all of the name fields match -- if you disregard character case -- then the birth dates and salaries are guaranteed to match as well, and the Social Security Numbers *would* match if they were somehow put in the same format.

Your job is to remove the duplicate records. The specific requirements of your job are:

- Remove duplicates. It doesn't matter which record you keep; it only matters that you keep one of them.
- Preserve the data format of the columns. For example, if you write the first name column in all lowercase, you haven't met this requirement.

The initial dataset contains 103,000 records. The de-duplicated result has 100,000 records.

Next, write the results in **Delta** format as a **single data file** to the directory given by the variable **delta_dest_dir**.

Remember the relationship between the number of partitions in a DataFrame and the number of files written.

Methods

- DataFrameReader (https://spark.apache.org/docs/latest/api/python/reference/pyspark.sql.html#input-and-output)
- DataFrame (https://spark.apache.org/docs/latest/api/python/reference/api/pyspark.sql.DataFrame.html)
- Built-In Functions (https://spark.apache.org/docs/latest/api/python/reference/pyspark.sql.html?#functions)
- DataFrameWriter (https://spark.apache.org/docs/latest/api/python/reference/pyspark.sql.html#input-and-output)

```
%run ../../Includes/Classroom-Setup
```

```
Deleted the working directory dbfs:/user/odl_user_534131@databrickslabs.com/dbacademy/aspwd/asp_4_3l_de_duping_data_la
b
```

```
Your working directory is
dbfs:/user/odl_user_534131@databrickslabs.com/dbacademy/aspwd

The source for this dataset is
wasbs://courseware@dbacademy.blob.core.windows.net/apache-spark-programming-with-databricks/v02/

Skipping install of existing dataset to
dbfs:/user/odl_user_534131@databrickslabs.com/dbacademy/aspwd/datasets

Out[5]: DataFrame[key: string, value: string]
```

It's helpful to look at the file first, so you can check the format with `dbutils.fs.head()`.

```
# TODO
from pyspark.sql.functions import *
source_file = f"{datasets_dir}/people/people-with-dups.txt"
delta_dest_dir = working_dir + "/people"

# In case it already exists
dbutils.fs.rm(delta_dest_dir, True)

# Complete your work here...
deduped_df = (df1
             .select(col("*"),
                     lower(col("firstName")).alias("lcFirstName"),
                     lower(col("lastName")).alias("lcLastName"),
                     lower(col("middleName")).alias("lcMiddleName"),
                     translate(col("ssn"), "-", "").alias("ssnNums")
                     # regexp_replace(col("ssn"), "-", "").alias("ssnNums")  # An alternate function to strip the
hyphens
                     # regexp_replace(col("ssn"), """^(\d{3})(\d{2})(\d{4})$""", "$1-$2-$3").alias("ssnNums")  # An
alternate that adds hyphens if missing
                    )
             .dropDuplicates(["lcFirstName", "lcMiddleName", "lcLastName", "ssnNums", "gender", "birthDate",
"salary"])
             .drop("lcFirstName", "lcMiddleName", "lcLastName", "ssnNums").repartition(1)
            )
deduped_df.write.format("delta").mode("overwrite").save(delta_dest_dir)
```

```
dbutils.fs.ls(delta_dest_dir)
```

```
Out[50]: [FileInfo(path='dbfs:/user/odl_user_534131@databrickslabs.com/dbacademy/aspwd/asp_4_3l_de_duping_data_lab/peo
ple/_delta_log/', name='_delta_log/', size=0),
 FileInfo(path='dbfs:/user/odl_user_534131@databrickslabs.com/dbacademy/aspwd/asp_4_3l_de_duping_data_lab/people/part-
00000-8f481373-c0cb-4fdc-91d0-34f4fbfb7c1e-c000.snappy.parquet', name='part-00000-8f481373-c0cb-4fdc-91d0-34f4fbfb7c1e
-c000.snappy.parquet', size=2691564)]
```

CHECK YOUR WORK

```
verify_files = dbutils.fs.ls(delta_dest_dir)
verify_delta_format = False
verify_num_data_files = 0
for f in verify_files:
    if f.name == "_delta_log/":
        verify_delta_format = True
    elif f.name.endswith(".parquet"):
        verify_num_data_files += 1

assert verify_delta_format, "Data not written in Delta format"
assert verify_num_data_files == 1, "Expected 1 data file written"

verify_record_count = spark.read.format("delta").load(delta_dest_dir).count()
assert verify_record_count == 100000, "Expected 100000 records in final result"

del verify_files, verify_delta_format, verify_num_data_files, verify_record_count
```

Clean up classroom

Run the cell below to clean up resources.

```
classroom_cleanup()
```

```
Dropped the database dbacademy_odl_user_534131_databrickslabs_com_aspwd_asp_4_3l_de_duping_data_lab
Deleted the working directory dbfs:/user/odl_user_534131@databrickslabs.com/dbacademy/aspwd/asp_4_3l_de_duping_data_la
b
```

Privacy Policy (https://databricks.com/privacy-policy) | Terms of Use (https://databricks.com/terms-of-use) | Support