Data Processing using Pyspark

In [1]:
```
#import SparkSession
from pyspark.sql import SparkSession
```

```
Starting Spark application
```

ID	YARN Application ID	Kind	State	Spark UI	Driver log	User	Current session?
0	application_1716817798215_0001	pyspark	idle	Link	Link	None	✔

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
SparkSession available as 'spark'.
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [2]:
```
#create spar session object
spark=SparkSession.builder.appName('data_processing').getOrCreate()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [3]:
```
# Load csv Dataset
df=spark.read.csv('s3://cristianbucket-labs-telematica/datasets/sample_data.csv',in
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [4]:
```
#columns of dataframe
df.columns
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
['ratings', 'age', 'experience', 'family', 'mobile']
```

In [5]:
```
#check number of columns
len(df.columns)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
5
```

In [6]:
```
#number of records in dataframe
df.count()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
33
```

In [7]:
```
#shape of dataset
print((df.count(),len(df.columns)))
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

```
(33, 5)
```

In [8]:
```
#printSchema
df.printSchema()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
root
 |-- ratings: integer (nullable = true)
 |-- age: integer (nullable = true)
 |-- experience: double (nullable = true)
 |-- family: integer (nullable = true)
 |-- mobile: string (nullable = true)
```

In [9]:
```
#fisrt few rows of dataframe
df.show(5)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+---+----------+------+-------+
|ratings|age|experience|family| mobile|
+-------+---+----------+------+-------+
|      3| 32|       9.0|     3|   Vivo|
|      3| 27|      13.0|     3|  Apple|
|      4| 22|       2.5|     0|Samsung|
|      4| 37|      16.5|     4|  Apple|
|      5| 27|       9.0|     1|     MI|
+-------+---+----------+------+-------+
only showing top 5 rows
```

In [10]:
```
#select only 2 columns
df.select('age','mobile').show(5)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+---+-------+
|age| mobile|
+---+-------+
| 32|   Vivo|
| 27|  Apple|
| 22|Samsung|
| 37|  Apple|
| 27|     MI|
+---+-------+
only showing top 5 rows
```

In [11]:
```
#info about dataframe
df.describe().show()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

```
+-------+------------------+------------------+------------------+------------------
+------+
|summary|           ratings|               age|        experience|            family
|mobile|
+-------+------------------+------------------+------------------+------------------
+------+
|  count|                33|                33|                33|                33
|    33|
|   mean|3.5757575757575757|30.484848484848484|10.303030303030303|1.8181818181818181
|  null|
| stddev|1.1188806636071336|  6.18527087180309| 6.770731351213326|1.8448330794164254
|  null|
|    min|                 1|                22|               2.5|                 0
| Apple|
|    max|                 5|                42|              23.0|                 5
|  Vivo|
+-------+------------------+------------------+------------------+------------------
+------+
```

In [12]:
```
from pyspark.sql.types import StringType,DoubleType,IntegerType
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [13]:
```
#with column
df.withColumn("age_after_10_yrs",(df["age"]+10)).show(10,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+---+----------+------+-------+---------------+
|ratings|age|experience|family|mobile |age_after_10_yrs|
+-------+---+----------+------+-------+---------------+
|3      |32 |9.0       |3     |Vivo   |42             |
|3      |27 |13.0      |3     |Apple  |37             |
|4      |22 |2.5       |0     |Samsung|32             |
|4      |37 |16.5      |4     |Apple  |47             |
|5      |27 |9.0       |1     |MI     |37             |
|4      |27 |9.0       |0     |Oppo   |37             |
|5      |37 |23.0      |5     |Vivo   |47             |
|5      |37 |23.0      |5     |Samsung|47             |
|3      |22 |2.5       |0     |Apple  |32             |
|3      |27 |6.0       |0     |MI     |37             |
+-------+---+----------+------+-------+---------------+
only showing top 10 rows
```

In [14]:
```
df.withColumn('age_double',df['age'].cast(DoubleType())).show(10,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

```
+-------+---+----------+------+-------+----------+
|ratings|age|experience|family|mobile |age_double|
+-------+---+----------+------+-------+----------+
|3      |32 |9.0       |3     |Vivo   |32.0      |
|3      |27 |13.0      |3     |Apple  |27.0      |
|4      |22 |2.5       |0     |Samsung|22.0      |
|4      |37 |16.5      |4     |Apple  |37.0      |
|5      |27 |9.0       |1     |MI     |27.0      |
|4      |27 |9.0       |0     |Oppo   |27.0      |
|5      |37 |23.0      |5     |Vivo   |37.0      |
|5      |37 |23.0      |5     |Samsung|37.0      |
|3      |22 |2.5       |0     |Apple  |22.0      |
|3      |27 |6.0       |0     |MI     |27.0      |
+-------+---+----------+------+-------+----------+
only showing top 10 rows
```

In [15]:
```
#with column
df.withColumn("age_after_10_yrs",(df["age"]+10)).show(10,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+---+----------+------+-------+---------------+
|ratings|age|experience|family|mobile |age_after_10_yrs|
+-------+---+----------+------+-------+---------------+
|3      |32 |9.0       |3     |Vivo   |42             |
|3      |27 |13.0      |3     |Apple  |37             |
|4      |22 |2.5       |0     |Samsung|32             |
|4      |37 |16.5      |4     |Apple  |47             |
|5      |27 |9.0       |1     |MI     |37             |
|4      |27 |9.0       |0     |Oppo   |37             |
|5      |37 |23.0      |5     |Vivo   |47             |
|5      |37 |23.0      |5     |Samsung|47             |
|3      |22 |2.5       |0     |Apple  |32             |
|3      |27 |6.0       |0     |MI     |37             |
+-------+---+----------+------+-------+---------------+
only showing top 10 rows
```

In [16]:
```
#filter the records
df.filter(df['mobile']=='Vivo').show()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+---+----------+------+------+
|ratings|age|experience|family|mobile|
+-------+---+----------+------+------+
|      3| 32|       9.0|     3|  Vivo|
|      5| 37|      23.0|     5|  Vivo|
|      4| 37|       6.0|     0|  Vivo|
|      5| 37|      13.0|     1|  Vivo|
|      4| 37|       6.0|     0|  Vivo|
+-------+---+----------+------+------+
```

In [17]:
```
#filter the records
df.filter(df['mobile']=='Vivo').select('age','ratings','mobile').show()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

```
+---+-------+------+
|age|ratings|mobile|
+---+-------+------+
| 32|      3|  Vivo|
| 37|      5|  Vivo|
| 37|      4|  Vivo|
| 37|      5|  Vivo|
| 37|      4|  Vivo|
+---+-------+------+
```

In [18]:
```
#filter the multiple conditions
df.filter(df['mobile']=='Vivo').filter(df['experience'] >10).show()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+---+----------+------+------+
|ratings|age|experience|family|mobile|
+-------+---+----------+------+------+
|      5| 37|      23.0|     5|  Vivo|
|      5| 37|      13.0|     1|  Vivo|
+-------+---+----------+------+------+
```

In [19]:
```
#filter the multiple conditions
df.filter((df['mobile']=='Vivo')&(df['experience'] >10)).show()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+---+----------+------+------+
|ratings|age|experience|family|mobile|
+-------+---+----------+------+------+
|      5| 37|      23.0|     5|  Vivo|
|      5| 37|      13.0|     1|  Vivo|
+-------+---+----------+------+------+
```

In [20]:
```
#Distinct Values in a column
df.select('mobile').distinct().show()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+
| mobile|
+-------+
|Samsung|
|     MI|
|   Oppo|
|  Apple|
|   Vivo|
+-------+
```

In [21]:
```
#distinct value count
df.select('mobile').distinct().count()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
5
```

In [22]:
```
df.groupBy('mobile').count().show(5,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+-----+
|mobile |count|
+-------+-----+
|Samsung|6    |
|MI     |8    |
|Oppo   |7    |
|Apple  |7    |
|Vivo   |5    |
+-------+-----+
```

In [23]:
```
# Value counts
df.groupBy('mobile').count().orderBy('count',ascending=False).show(5,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+-----+
|mobile |count|
+-------+-----+
|MI     |8    |
|Oppo   |7    |
|Apple  |7    |
|Samsung|6    |
|Vivo   |5    |
+-------+-----+
```

In [24]:
```
# Value counts
df.groupBy('mobile').mean().show(5,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+------------------+-----------------+-----------------+-----------------
+
|mobile |avg(ratings)      |avg(age)         |avg(experience)  |avg(family)
|
+-------+------------------+-----------------+-----------------+-----------------
+
|Samsung|4.166666666666667 |28.666666666666668|8.666666666666666 |1.8333333333333333
|
|MI     |3.5               |30.125           |10.1875          |1.375
|
|Oppo   |2.857142857142857 |28.428571428571427|10.357142857142858|1.4285714285714286
|
|Apple  |3.4285714285714284|30.571428571428573|11.0             |2.7142857142857144
|
|Vivo   |4.2               |36.0             |11.4             |1.8
|
+-------+------------------+-----------------+-----------------+-----------------
+
```

In [25]:
```
df.groupBy('mobile').sum().show(5,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

```
+-------+------------+--------+---------------+-----------+
|mobile |sum(ratings)|sum(age)|sum(experience)|sum(family)|
+-------+------------+--------+---------------+-----------+
|Samsung|25          |172     |52.0           |11         |
|MI     |28          |241     |81.5           |11         |
|Oppo   |20          |199     |72.5           |10         |
|Apple  |24          |214     |77.0           |19         |
|Vivo   |21          |180     |57.0           |9          |
+-------+------------+--------+---------------+-----------+
```

In [26]:
```
# Value counts
df.groupBy('mobile').max().show(5,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+------------+--------+---------------+-----------+
|mobile |max(ratings)|max(age)|max(experience)|max(family)|
+-------+------------+--------+---------------+-----------+
|Samsung|5           |37      |23.0           |5          |
|MI     |5           |42      |23.0           |5          |
|Oppo   |4           |42      |23.0           |2          |
|Apple  |4           |37      |16.5           |5          |
|Vivo   |5           |37      |23.0           |5          |
+-------+------------+--------+---------------+-----------+
```

In [27]:
```
# Value counts
df.groupBy('mobile').min().show(5,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+------------+--------+---------------+-----------+
|mobile |min(ratings)|min(age)|min(experience)|min(family)|
+-------+------------+--------+---------------+-----------+
|Samsung|2           |22      |2.5            |0          |
|MI     |1           |27      |2.5            |0          |
|Oppo   |2           |22      |6.0            |0          |
|Apple  |3           |22      |2.5            |0          |
|Vivo   |3           |32      |6.0            |0          |
+-------+------------+--------+---------------+-----------+
```

In [28]:
```
#Aggregation
df.groupBy('mobile').agg({'experience':'sum'}).show(5,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+---------------+
|mobile |sum(experience)|
+-------+---------------+
|Samsung|52.0           |
|MI     |81.5           |
|Oppo   |72.5           |
|Apple  |77.0           |
|Vivo   |57.0           |
+-------+---------------+
```

In [29]:
```
# UDF
from pyspark.sql.functions import udf
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [30]:
```
#normal function
def price_range(brand):
    if brand in ['Samsung','Apple']:
        return 'High Price'
    elif brand =='MI':
        return 'Mid Price'
    else:
        return 'Low Price'
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [31]:
```
#create udf using python function
brand_udf=udf(price_range,StringType())
#apply udf on dataframe
df.withColumn('price_range',brand_udf(df['mobile'])).show(10,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
+-------+---+----------+------+-------+-----------+
|ratings|age|experience|family|mobile |price_range|
+-------+---+----------+------+-------+-----------+
|3      |32 |9.0       |3     |Vivo   |Low Price  |
|3      |27 |13.0      |3     |Apple  |High Price |
|4      |22 |2.5       |0     |Samsung|High Price |
|4      |37 |16.5      |4     |Apple  |High Price |
|5      |27 |9.0       |1     |MI     |Mid Price  |
|4      |27 |9.0       |0     |Oppo   |Low Price  |
|5      |37 |23.0      |5     |Vivo   |Low Price  |
|5      |37 |23.0      |5     |Samsung|High Price |
|3      |22 |2.5       |0     |Apple  |High Price |
|3      |27 |6.0       |0     |MI     |Mid Price  |
+-------+---+----------+------+-------+-----------+
only showing top 10 rows
```

In [32]:
```
#using lambda function
age_udf = udf(lambda age: "young" if age <= 30 else "senior", StringType())
#apply udf on dataframe
df.withColumn("age_group", age_udf(df.age)).show(10,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

```
+-------+---+----------+------+-------+---------+
|ratings|age|experience|family|mobile |age_group|
+-------+---+----------+------+-------+---------+
|3      |32 |9.0       |3     |Vivo   |senior   |
|3      |27 |13.0      |3     |Apple  |young    |
|4      |22 |2.5       |0     |Samsung|young    |
|4      |37 |16.5      |4     |Apple  |senior   |
|5      |27 |9.0       |1     |MI     |young    |
|4      |27 |9.0       |0     |Oppo   |young    |
|5      |37 |23.0      |5     |Vivo   |senior   |
|5      |37 |23.0      |5     |Samsung|senior   |
|3      |22 |2.5       |0     |Apple  |young    |
|3      |27 |6.0       |0     |MI     |young    |
+-------+---+----------+------+-------+---------+
only showing top 10 rows
```

In [33]:
```
#pandas udf
from pyspark.sql.functions import pandas_udf, PandasUDFType
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [34]:
```
#create python function
def remaining_yrs(age):
    yrs_left=100-age

    return yrs_left
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [35]:
```
#create udf using python function
length_udf = pandas_udf(remaining_yrs, IntegerType())
#apply pandas udf on dataframe
df.withColumn("yrs_left", length_udf(df['age'])).show(10,False)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

```
An error was encountered:
Pandas >= 1.0.5 must be installed; however, it was not found.
Traceback (most recent call last):
  File "/mnt/yarn/usercache/livy/appcache/application_1716817798215_0001/container_1
716817798215_0001_01_000001/pyspark.zip/pyspark/sql/pandas/functions.py", line 336,
in pandas_udf
    require_minimum_pandas_version()
  File "/mnt/yarn/usercache/livy/appcache/application_1716817798215_0001/container_1
716817798215_0001_01_000001/pyspark.zip/pyspark/sql/pandas/utils.py", line 36, in re
quire_minimum_pandas_version
    ) from raised_error
ImportError: Pandas >= 1.0.5 must be installed; however, it was not found.
```

In [36]:
```
#udf using two columns
def prod(rating,exp):
    x=rating*exp
    return x
```

FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(height='25px', width='50%'),…

In [37]:
```
#create udf using python function
prod_udf = pandas_udf(prod, DoubleType())
#apply pandas udf on multiple columns of dataframe
df.withColumn("product", prod_udf(df['ratings'],df['experience'])).show(10,False)
```

FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(height='25px', width='50%'),…

```
An error was encountered:
Pandas >= 1.0.5 must be installed; however, it was not found.
Traceback (most recent call last):
  File "/mnt/yarn/usercache/livy/appcache/application_1716817798215_0001/container_1716817798215_0001_01_000001/pyspark.zip/pyspark/sql/pandas/functions.py", line 336, in pandas_udf
    require_minimum_pandas_version()
  File "/mnt/yarn/usercache/livy/appcache/application_1716817798215_0001/container_1716817798215_0001_01_000001/pyspark.zip/pyspark/sql/pandas/utils.py", line 36, in require_minimum_pandas_version
    ) from raised_error
ImportError: Pandas >= 1.0.5 must be installed; however, it was not found.
```

In [38]:
```
#duplicate values
df.count()
```

FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(height='25px', width='50%'),…

```
33
```

In [39]:
```
#drop duplicate values
df=df.dropDuplicates()
```

FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(height='25px', width='50%'),…

In [40]:
```
#validate new count
df.count()
```

FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(height='25px', width='50%'),…

```
26
```

In [41]:
```
#drop column of dataframe
df_new=df.drop('mobile')
```

FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(height='25px', width='50%'),…

In [42]:
```
df_new.show(10)
```

FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(height='25px', width='50%'),…

```
+-------+---+----------+------+
|ratings|age|experience|family|
+-------+---+----------+------+
|      4| 22|       2.5|     0|
|      4| 22|       6.0|     1|
|      3| 27|       6.0|     0|
|      2| 32|      16.5|     2|
|      4| 27|       9.0|     0|
|      3| 37|      16.5|     5|
|      4| 27|       6.0|     1|
|      4| 37|       9.0|     2|
|      3| 22|       2.5|     0|
|      3| 32|       9.0|     3|
+-------+---+----------+------+
only showing top 10 rows
```

In [43]:
```
# saving file (csv)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [44]:
```
#current working directory
pwd
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

```
An error was encountered:
name 'pwd' is not defined
Traceback (most recent call last):
NameError: name 'pwd' is not defined
```

In [45]:
```
#target directory
write_uri='s3://cristianbucket-labs-telematica/datasets/df_csv'
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [46]:
```
#save the dataframe as single csv
df.coalesce(1).write.format("csv").option("header","true").save(write_uri)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [47]:
```
# parquet
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [48]:
```
#target location
parquet_uri='s3://<bucket/dir>/df_parquet'
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

In [49]:
```
#save the data into parquet format
df.write.format('parquet').save(parquet_uri)
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(he
ight='25px', width='50%'),…
```

```
An error was encountered:
Illegal character in authority at index 5: s3://<bucket
Traceback (most recent call last):
  File "/mnt/yarn/usercache/livy/appcache/application_1716817798215_0001/container_1
716817798215_0001_01_000001/pyspark.zip/pyspark/sql/readwriter.py", line 1398, in sa
ve
    self._jwrite.save(path)
  File "/mnt/yarn/usercache/livy/appcache/application_1716817798215_0001/container_1
716817798215_0001_01_000001/py4j-0.10.9.7-src.zip/py4j/java_gateway.py", line 1323,
in __call__
    answer, self.gateway_client, self.target_id, self.name)
  File "/mnt/yarn/usercache/livy/appcache/application_1716817798215_0001/container_1
716817798215_0001_01_000001/pyspark.zip/pyspark/errors/exceptions/captured.py", line
175, in deco
    raise converted from None
pyspark.errors.exceptions.captured.IllegalArgumentException: Illegal character in au
thority at index 5: s3://<bucket
```

In [ ]: