SPRAWOZDANIE

Zajęcia: Eksploracja i wizualizacja danych
Prowadzący: prof. dr hab. Vasyl Martsenyuk

Laboratorium: 3
Data: 23.03.2023
Temat: "Użycie biblioteki PySpark dla dużych zbiorów danych"
Wariant: 7

Michał Stajerski
Informatyka II stopień,
stacjonarne, semestr 3,
Gr. 1
https://github.com/MichalStajerski/eiwd

1. Polecenie

Celem zajęć jest eksploracja i wizualizacja danych z użyciem API Spark przez bibliotekę pyspark. Kolejnym etapem zajęć jest pobieranie danych z pliku CSV lokalnie oraz z użyciem URL. Podstawowe kroki eksploracji przedstawione są w pliku Zajecie3_Spark.ipynb. Dane są określone wariantem z zadania 1, które zostały pobrane ze strony https://ghdx.healthdata.org/ihme_data. Wariant wybrany w zadaniu jest wariant 7:

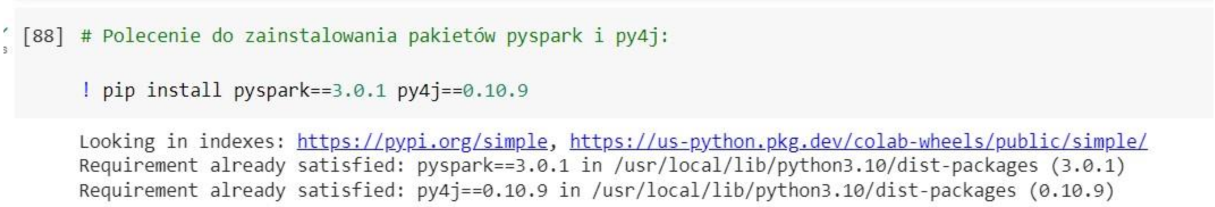
Global Burden of Disease Study 2019 (GBD 2019) Smoking Tobacco Use Prevalence 1990-
2019

2. Zadania

1 - Użycie PySpark w celu eksploracji Big Data

1.1 - Konfigurowanie środowiska w Anaconda lub Google Colab

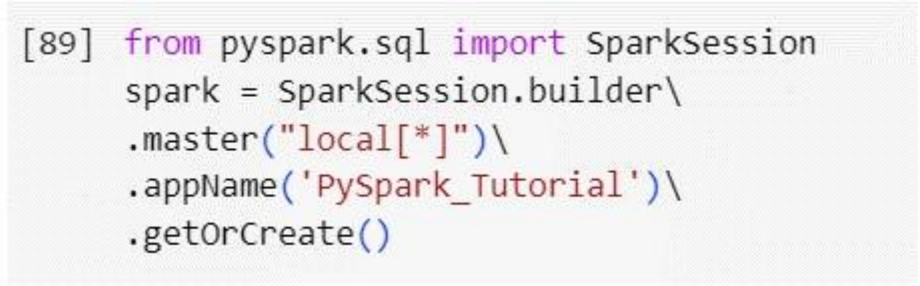
Do wykonywania zadań został wybrany Google Colaboratory



```
[88] # Polecenie do zainstalowania pakietów pyspark i py4j:

     ! pip install pyspark==3.0.1 py4j==0.10.9

     Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
     Requirement already satisfied: pyspark==3.0.1 in /usr/local/lib/python3.10/dist-packages (3.0.1)
     Requirement already satisfied: py4j==0.10.9 in /usr/local/lib/python3.10/dist-packages (0.10.9)
```

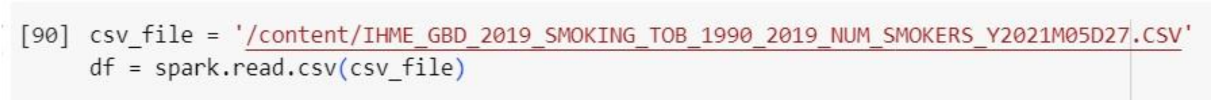
1.2 - Sesja Spark

1.3 - Tworzenie SparkSession



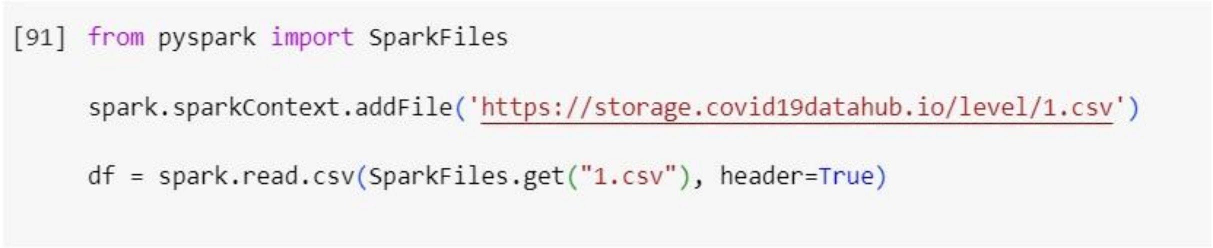
```
[89] from pyspark.sql import SparkSession
     spark = SparkSession.builder\
     .master("local[*]")\
     .appName('PySpark_Tutorial')\
     .getOrCreate()
```

1.4 - Czytanie danych



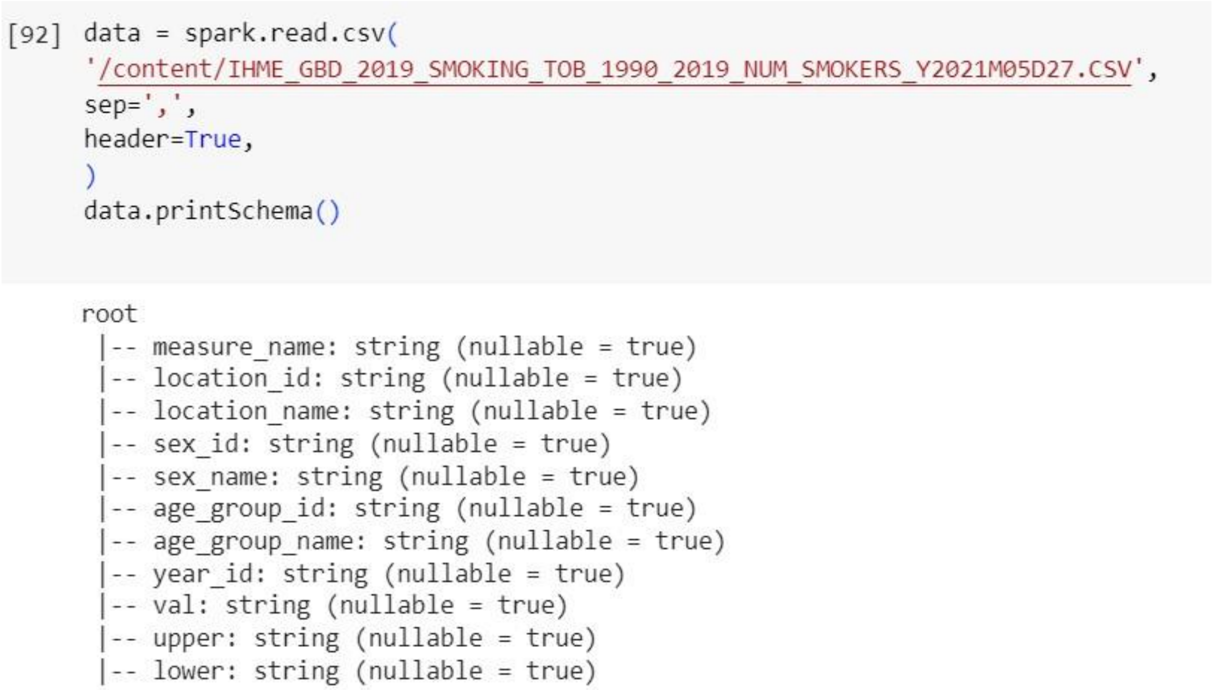
```
[90] csv_file = '/content/IHME_GBD_2019_SMOKING_TOB_1990_2019_NUM_SMOKERS_Y2021M05D27.CSV'
     df = spark.read.csv(csv_file)
```

1.5 - Pobieranie danych za pomocą URL



```
[91] from pyspark import SparkFiles

     spark.sparkContext.addFile('https://storage.covid19datahub.io/level/1.csv')

     df = spark.read.csv(SparkFiles.get("1.csv"), header=True)
```

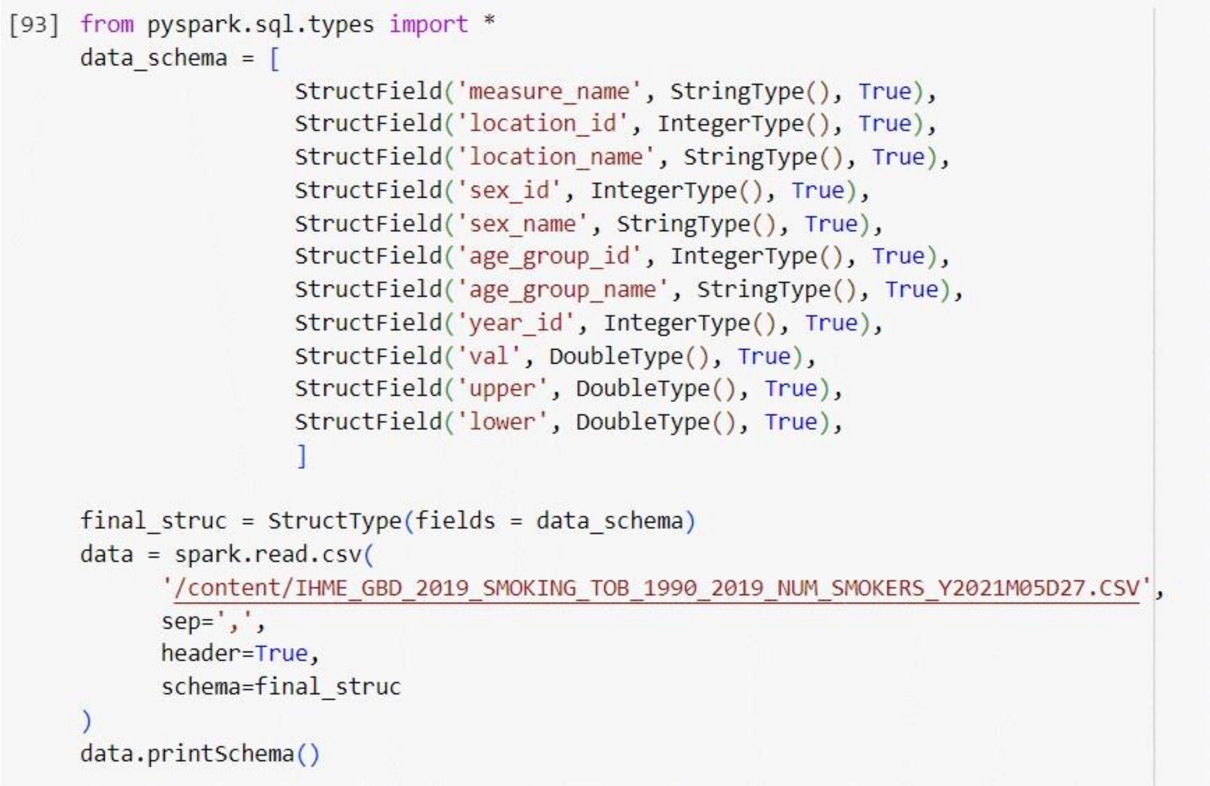
2 - Strukturyzacja danych za pomocą schematu Spark Kod do odczytu danych w formacie pliku CSV:



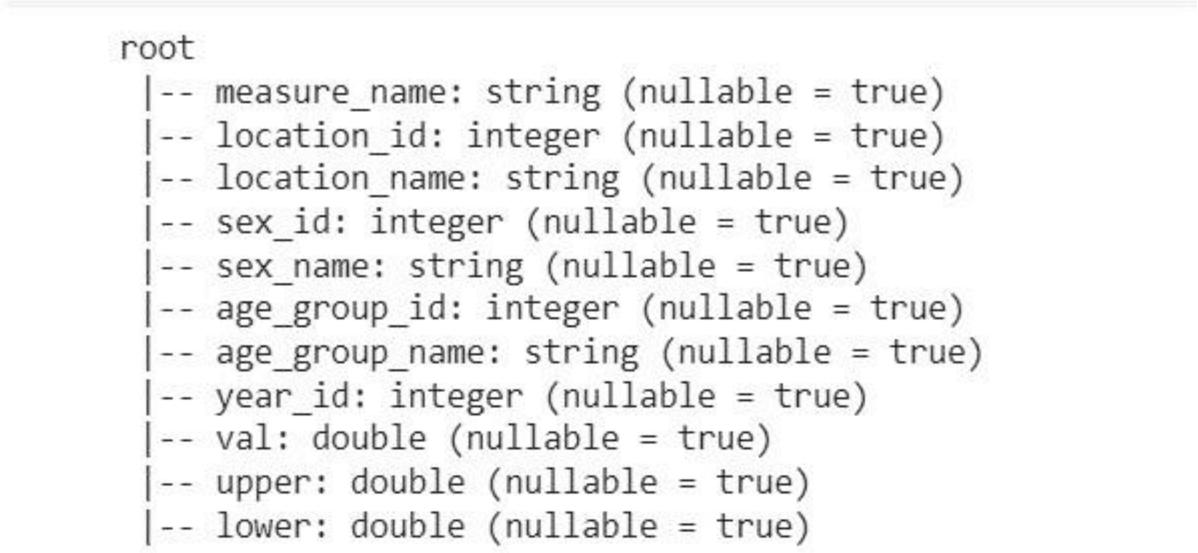
```
[92] data = spark.read.csv(
     '/content/IHME_GBD_2019_SMOKING_TOB_1990_2019_NUM_SMOKERS_Y2021M05D27.CSV',
     sep=',',
     header=True,
     )
     data.printSchema()
```

```
root
 |-- measure_name: string (nullable = true)
 |-- location_id: string (nullable = true)
 |-- location_name: string (nullable = true)
 |-- sex_id: string (nullable = true)
 |-- sex_name: string (nullable = true)
 |-- age_group_id: string (nullable = true)
 |-- age_group_name: string (nullable = true)
 |-- year_id: string (nullable = true)
 |-- val: string (nullable = true)
 |-- upper: string (nullable = true)
 |-- lower: string (nullable = true)
```

Precyzowanie struktury danych



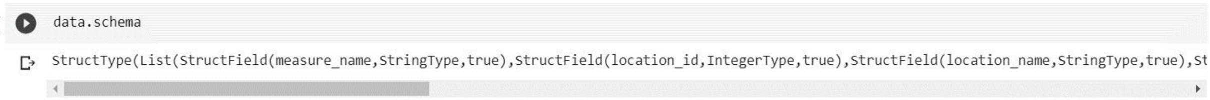
```
[93] from pyspark.sql.types import *
     data_schema = [
                     StructField('measure_name', StringType(), True),
                     StructField('location_id', IntegerType(), True),
                     StructField('location_name', StringType(), True),
                     StructField('sex_id', IntegerType(), True),
                     StructField('sex_name', StringType(), True),
                     StructField('age_group_id', IntegerType(), True),
                     StructField('age_group_name', StringType(), True),
                     StructField('year_id', IntegerType(), True),
                     StructField('val', DoubleType(), True),
                     StructField('upper', DoubleType(), True),
                     StructField('lower', DoubleType(), True),
                     ]

     final_struc = StructType(fields = data_schema)
     data = spark.read.csv(
           '/content/IHME_GBD_2019_SMOKING_TOB_1990_2019_NUM_SMOKERS_Y2021M05D27.CSV',
           sep=',',
           header=True,
           schema=final_struc
     )
     data.printSchema()
```



```
root
 |-- measure_name: string (nullable = true)
 |-- location_id: integer (nullable = true)
 |-- location_name: string (nullable = true)
 |-- sex_id: integer (nullable = true)
 |-- sex_name: string (nullable = true)
 |-- age_group_id: integer (nullable = true)
 |-- age_group_name: string (nullable = true)
 |-- year_id: integer (nullable = true)
 |-- val: double (nullable = true)
 |-- upper: double (nullable = true)
 |-- lower: double (nullable = true)
```

3 - Różne metody kontroli danych

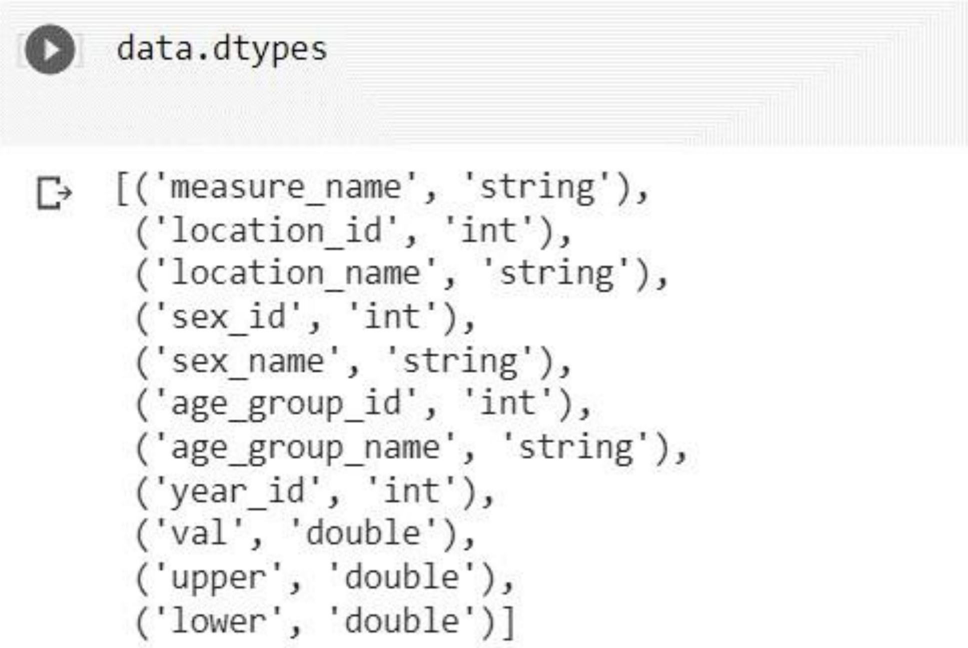
3.1 - schema(): Ta metoda zwraca schemat danych (ramka danych).



```
data.schema
```

```
StructType(List(StructField(measure_name,StringType,true),StructField(location_id,IntegerType,true),StructField(location_name,StringType,true),St
```

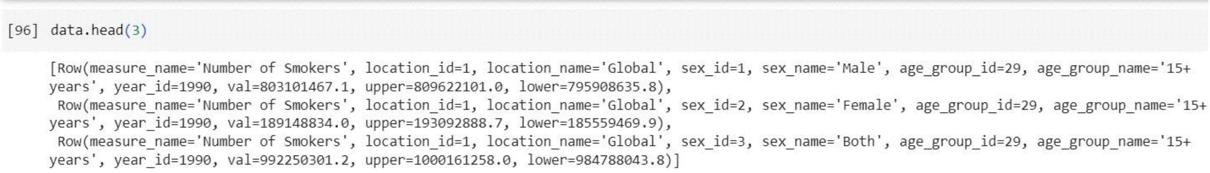
3.2 - dtypes zwraca listę krotek z nazwami kolumn i typami danych.



```
data.dtypes
```

```
[('measure_name', 'string'),
 ('location_id', 'int'),
 ('location_name', 'string'),
 ('sex_id', 'int'),
 ('sex_name', 'string'),
 ('age_group_id', 'int'),
 ('age_group_name', 'string'),
 ('year_id', 'int'),
 ('val', 'double'),
 ('upper', 'double'),
 ('lower', 'double')]
```

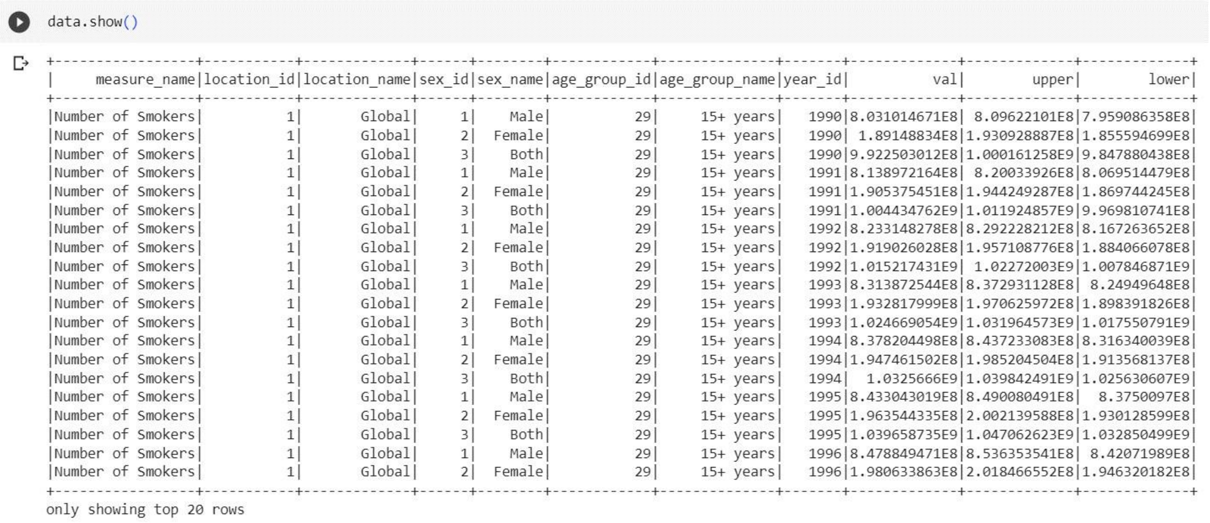
3.3 - head(n) zwraca n wierszy jako listę.



```
[96] data.head(3)
```

```
[Row(measure_name='Number of Smokers', location_id=1, location_name='Global', sex_id=1, sex_name='Male', age_group_id=29, age_group_name='15+
years', year_id=1990, val=803101467.1, upper=809622101.0, lower=795908635.8),
 Row(measure_name='Number of Smokers', location_id=1, location_name='Global', sex_id=2, sex_name='Female', age_group_id=29, age_group_name='15+
years', year_id=1990, val=189148834.0, upper=193092888.7, lower=185559469.9),
 Row(measure_name='Number of Smokers', location_id=1, location_name='Global', sex_id=3, sex_name='Both', age_group_id=29, age_group_name='15+
years', year_id=1990, val=992250301.2, upper=1000161258.0, lower=984788043.8)]
```

3.4 - show() domyślnie wyświetla pierwsze 20 wierszy, a także przyjmuje liczbę jako parametr określający ich liczbę

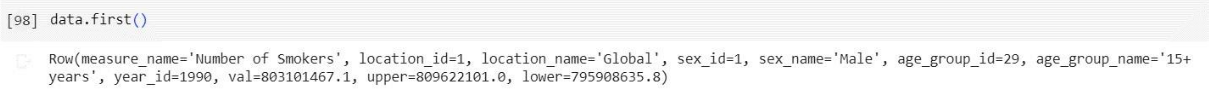


```
data.show()
```

```
+-----------------+-----------+-------------+------+--------+------------+--------------+-------+-------------+-------------+-------------+
|     measure_name|location_id|location_name|sex_id|sex_name|age_group_id|age_group_name|year_id|          val|        upper|        lower|
+-----------------+-----------+-------------+------+--------+------------+--------------+-------+-------------+-------------+-------------+
|Number of Smokers|          1|       Global|     1|    Male|          29|     15+ years|   1990|8.031014671E8| 8.09622101E8|7.959086358E8|
|Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   1990| 1.89148834E8|1.930928887E8|1.855594699E8|
|Number of Smokers|          1|       Global|     3|    Both|          29|     15+ years|   1990|9.922503012E8|1.000161258E9|9.847880438E8|
|Number of Smokers|          1|       Global|     1|    Male|          29|     15+ years|   1991|8.138972164E8| 8.20033926E8|8.069514479E8|
|Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   1991|1.905375451E8|1.944249287E8|1.869744245E8|
|Number of Smokers|          1|       Global|     3|    Both|          29|     15+ years|   1991|1.004434762E9|1.011924857E9|9.969810741E8|
|Number of Smokers|          1|       Global|     1|    Male|          29|     15+ years|   1992|8.233148278E8|8.292228212E8|8.167263652E8|
|Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   1992|1.919026028E8|1.957108776E8|1.884066078E8|
|Number of Smokers|          1|       Global|     3|    Both|          29|     15+ years|   1992|1.015217431E9| 1.02272003E9|1.007846871E9|
|Number of Smokers|          1|       Global|     1|    Male|          29|     15+ years|   1993|8.313872544E8|8.372931128E8| 8.24949648E8|
|Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   1993|1.932817999E8|1.970625972E8|1.898391826E8|
|Number of Smokers|          1|       Global|     3|    Both|          29|     15+ years|   1993|1.024669054E9|1.031964573E9|1.017550791E9|
|Number of Smokers|          1|       Global|     1|    Male|          29|     15+ years|   1994|8.378204498E8|8.437233083E8|8.316340039E8|
|Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   1994|1.947461502E8|1.985204504E8|1.913568137E8|
|Number of Smokers|          1|       Global|     3|    Both|          29|     15+ years|   1994|  1.0325666E9|1.039842491E9|1.025630607E9|
|Number of Smokers|          1|       Global|     1|    Male|          29|     15+ years|   1995|8.433043019E8|8.490080491E8|  8.3750097E8|
|Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   1995|1.963544335E8|2.002139588E8|1.930128599E8|
|Number of Smokers|          1|       Global|     3|    Both|          29|     15+ years|   1995|1.039658735E9|1.047062623E9|1.032850499E9|
|Number of Smokers|          1|       Global|     1|    Male|          29|     15+ years|   1996|8.478849471E8|8.536353541E8| 8.42071989E8|
|Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   1996|1.980633863E8|2.018466552E8|1.946320182E8|
+-----------------+-----------+-------------+------+--------+------------+--------------+-------+-------------+-------------+-------------+
only showing top 20 rows
```

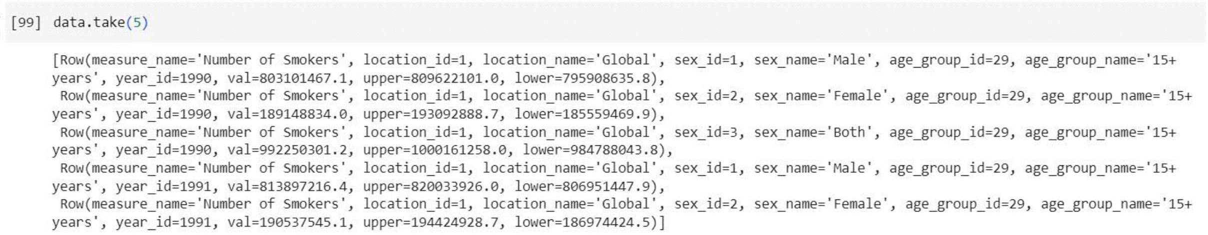
-

3.5 first() zwraca pierwszy wiersz danych.



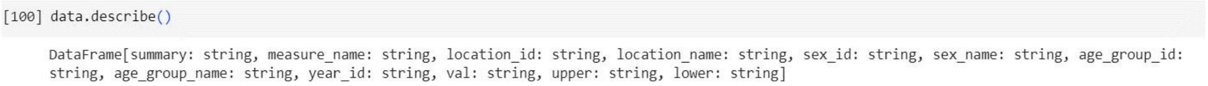
```
[98] data.first()

Row(measure_name='Number of Smokers', location_id=1, location_name='Global', sex_id=1, sex_name='Male', age_group_id=29, age_group_name='15+
years', year_id=1990, val=803101467.1, upper=809622101.0, lower=795908635.8)
```

3.6 - take(n) zwraca pierwsze n wierszy



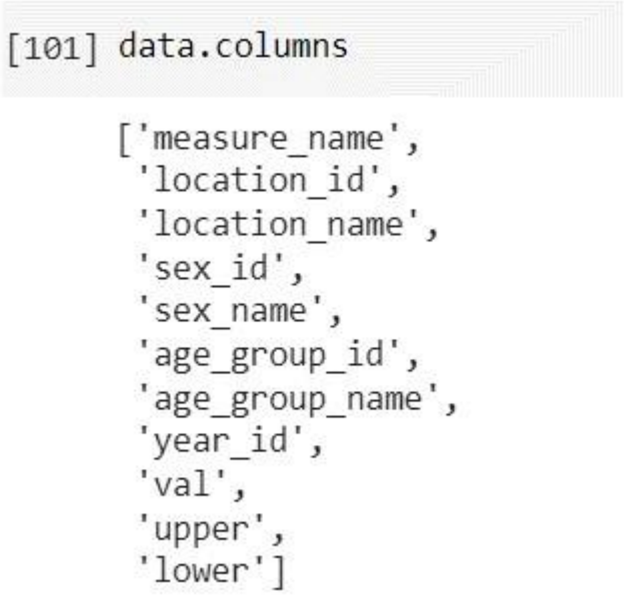
```
[99] data.take(5)

[Row(measure_name='Number of Smokers', location_id=1, location_name='Global', sex_id=1, sex_name='Male', age_group_id=29, age_group_name='15+
years', year_id=1990, val=803101467.1, upper=809622101.0, lower=795908635.8),
 Row(measure_name='Number of Smokers', location_id=1, location_name='Global', sex_id=2, sex_name='Female', age_group_id=29, age_group_name='15+
years', year_id=1990, val=189148834.0, upper=193092888.7, lower=185559469.9),
 Row(measure_name='Number of Smokers', location_id=1, location_name='Global', sex_id=3, sex_name='Both', age_group_id=29, age_group_name='15+
years', year_id=1990, val=992250301.2, upper=1000161258.0, lower=984788043.8),
 Row(measure_name='Number of Smokers', location_id=1, location_name='Global', sex_id=1, sex_name='Male', age_group_id=29, age_group_name='15+
years', year_id=1991, val=813897216.4, upper=820033926.0, lower=806951447.9),
 Row(measure_name='Number of Smokers', location_id=1, location_name='Global', sex_id=2, sex_name='Female', age_group_id=29, age_group_name='15+
years', year_id=1991, val=190537545.1, upper=194424928.7, lower=186974424.5)]
```

3.7 - describe() oblicza niektóre wartości statystyczne dla kolumn liczbowych.



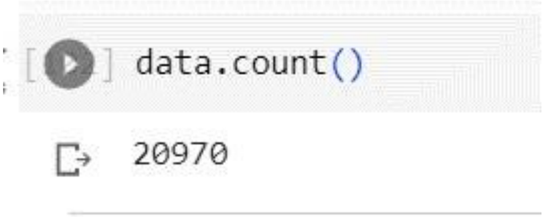
```
[100] data.describe()

DataFrame[summary: string, measure_name: string, location_id: string, location_name: string, sex_id: string, sex_name: string, age_group_id:
string, age_group_name: string, year_id: string, val: string, upper: string, lower: string]
```

3.8 - columns zwraca listę zawierającą nazwy kolumn.



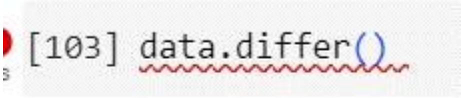
```
[101] data.columns

['measure_name',
 'location_id',
 'location_name',
 'sex_id',
 'sex_name',
 'age_group_id',
 'age_group_name',
 'year_id',
 'val',
 'upper',
 'lower']
```

3.9 - count() zwraca całkowitą liczbę wierszy w zestawie danych



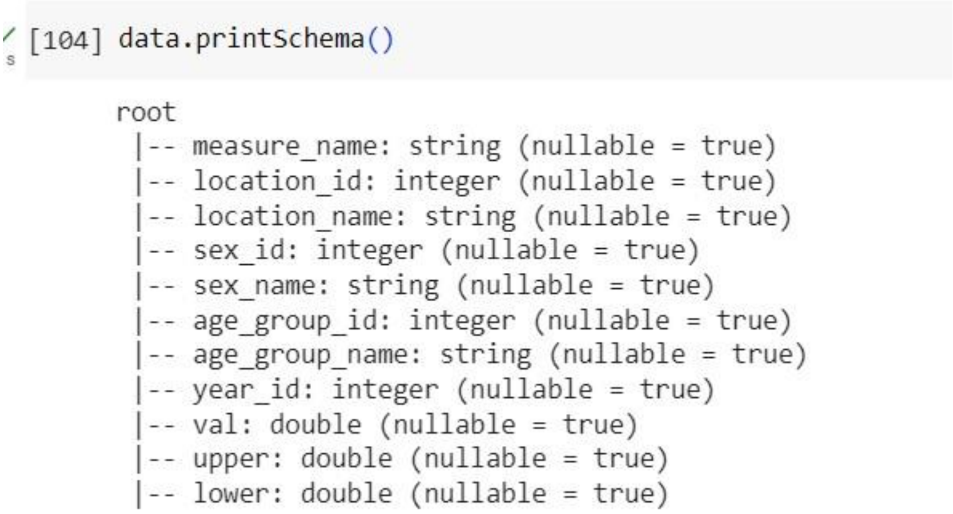
```
data.count()

20970
```

3.10 - differ() to liczba odmiennych wierszy w używanym zbiorze danych.



```
[103] data.differ()
```

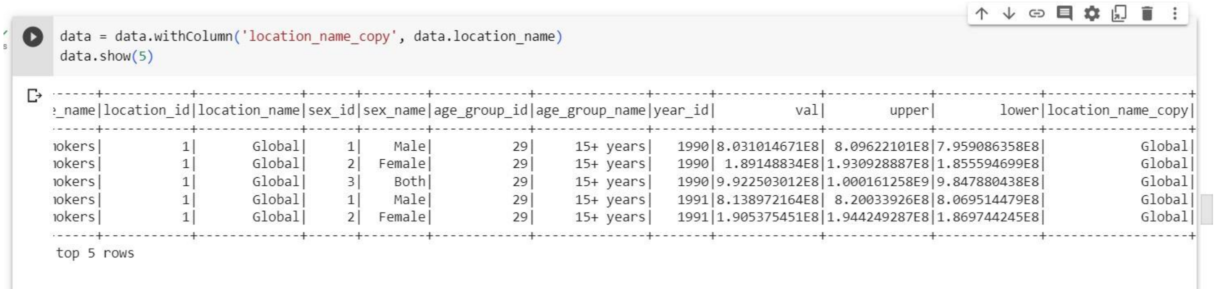
3.11 - printSchema() wyświetla schemat danych.



```
[104] data.printSchema()

root
 |-- measure_name: string (nullable = true)
 |-- location_id: integer (nullable = true)
 |-- location_name: string (nullable = true)
 |-- sex_id: integer (nullable = true)
 |-- sex_name: string (nullable = true)
 |-- age_group_id: integer (nullable = true)
 |-- age_group_name: string (nullable = true)
 |-- year_id: integer (nullable = true)
 |-- val: double (nullable = true)
 |-- upper: double (nullable = true)
 |-- lower: double (nullable = true)
```

4 - Manipulacja kolumnami

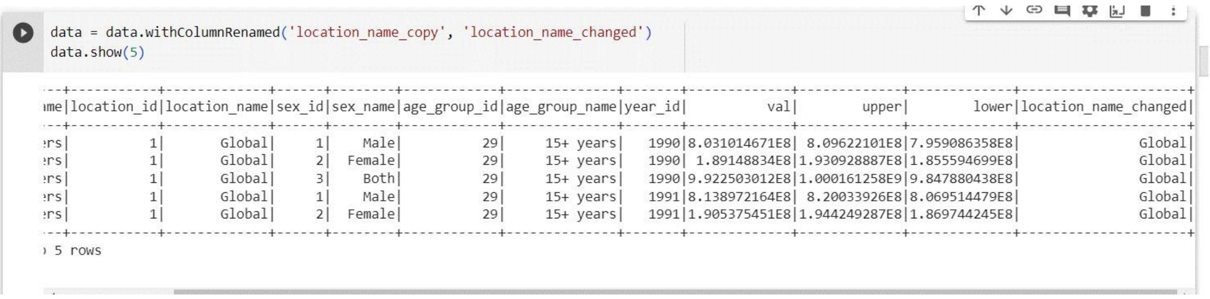
4.1 - Dodawanie kolumny:



```
data = data.withColumn('location_name_copy', data.location_name)
data.show(5)
```

```
-----+-----------+-------------+------+--------+------------+--------------+-------+------------+------------+------------+------------------+
_name|location_id|location_name|sex_id|sex_name|age_group_id|age_group_name|year_id|         val|       upper|       lower|location_name_copy|
-----+-----------+-------------+------+--------+------------+--------------+-------+------------+------------+------------+------------------+
okers|          1|       Global|     1|    Male|          29|     15+ years|   1990|8.031014671E8| 8.09622101E8|7.959086358E8|            Global|
okers|          1|       Global|     2|  Female|          29|     15+ years|   1990| 1.89148834E8|1.930928887E8|1.855594699E8|            Global|
okers|          1|       Global|     3|    Both|          29|     15+ years|   1990|9.922503012E8|1.000161258E9|9.847880438E8|            Global|
okers|          1|       Global|     1|    Male|          29|     15+ years|   1991|8.138972164E8| 8.20033926E8|8.069514479E8|            Global|
okers|          1|       Global|     2|  Female|          29|     15+ years|   1991|1.905375451E8|1.944249287E8|1.869744245E8|            Global|
-----+-----------+-------------+------+--------+------------+--------------+-------+------------+------------+------------+------------------+
top 5 rows
```

4.2 - Aktualizacja kolumny:

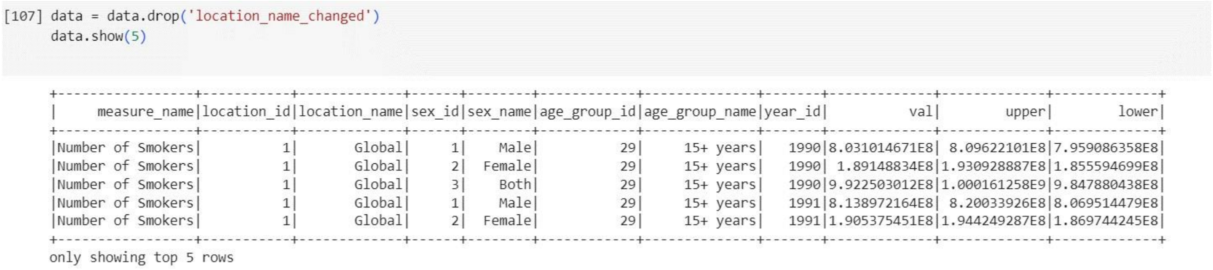


```
data = data.withColumnRenamed('location_name_copy', 'location_name_changed')
data.show(5)
```

```
--+-----------+-------------+------+--------+------------+--------------+-------+------------+------------+------------+---------------------+
me|location_id|location_name|sex_id|sex_name|age_group_id|age_group_name|year_id|         val|       upper|       lower|location_name_changed|
--+-----------+-------------+------+--------+------------+--------------+-------+------------+------------+------------+---------------------+
rs|          1|       Global|     1|    Male|          29|     15+ years|   1990|8.031014671E8| 8.09622101E8|7.959086358E8|               Global|
rs|          1|       Global|     2|  Female|          29|     15+ years|   1990| 1.89148834E8|1.930928887E8|1.855594699E8|               Global|
rs|          1|       Global|     3|    Both|          29|     15+ years|   1990|9.922503012E8|1.000161258E9|9.847880438E8|               Global|
rs|          1|       Global|     1|    Male|          29|     15+ years|   1991|8.138972164E8| 8.20033926E8|8.069514479E8|               Global|
rs|          1|       Global|     2|  Female|          29|     15+ years|   1991|1.905375451E8|1.944249287E8|1.869744245E8|               Global|
--+-----------+-------------+------+--------+------------+--------------+-------+------------+------------+------------+---------------------+
 5 rows
```

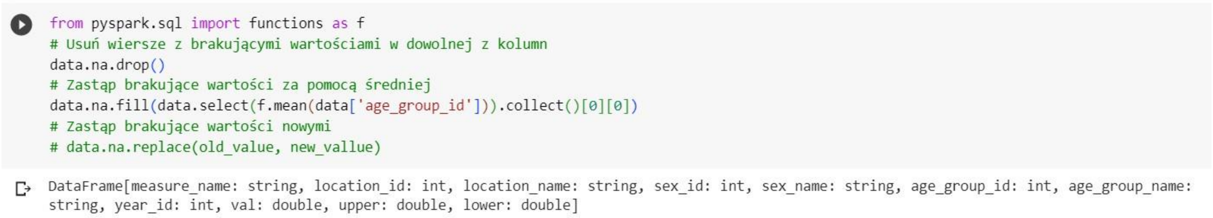
4.3 Upuszczanie kolumny:

-



```
[107] data = data.drop('location_name_changed')
      data.show(5)

+-----------------+-----------+-------------+------+--------+------------+--------------+-------+-------------+-------------+-------------+
|     measure_name|location_id|location_name|sex_id|sex_name|age_group_id|age_group_name|year_id|          val|        upper|        lower|
+-----------------+-----------+-------------+------+--------+------------+--------------+-------+-------------+-------------+-------------+
|Number of Smokers|          1|       Global|     1|    Male|          29|     15+ years|   1990|8.031014671E8| 8.09622101E8|7.959086358E8|
|Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   1990| 1.89148834E8|1.930928887E8|1.855594699E8|
|Number of Smokers|          1|       Global|     3|    Both|          29|     15+ years|   1990|9.922503012E8|1.000161258E9|9.847880438E8|
|Number of Smokers|          1|       Global|     1|    Male|          29|     15+ years|   1991|8.138972164E8| 8.20033926E8|8.069514479E8|
|Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   1991|1.905375451E8|1.944249287E8|1.869744245E8|
+-----------------+-----------+-------------+------+--------+------------+--------------+-------+-------------+-------------+-------------+
only showing top 5 rows
```

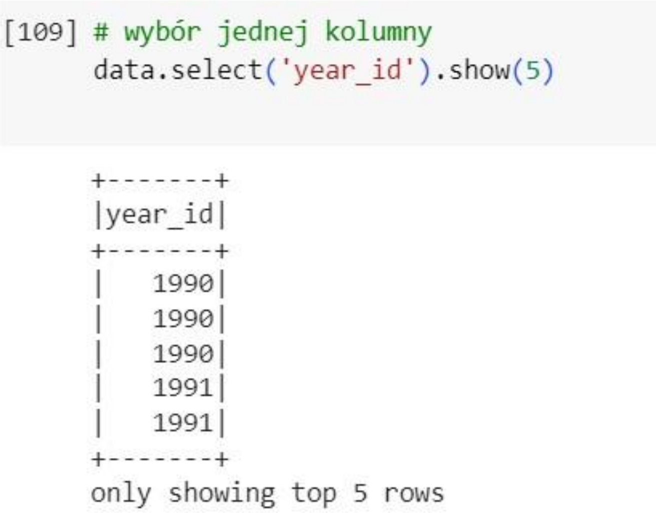
5 - Radzenie sobie z brakującymi wartościami



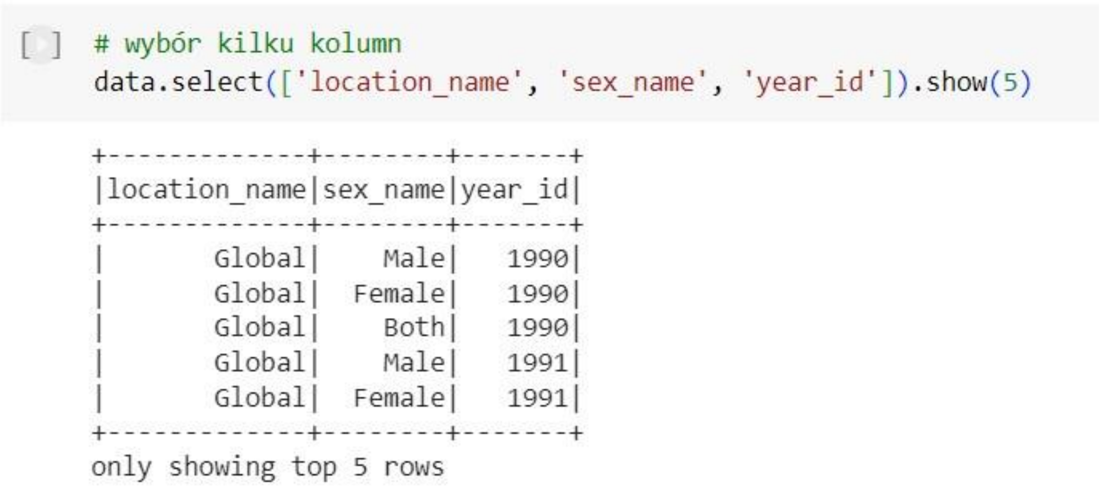
```
from pyspark.sql import functions as f
# Usuń wiersze z brakującymi wartościami w dowolnej z kolumn
data.na.drop()
# Zastąp brakujące wartości za pomocą średniej
data.na.fill(data.select(f.mean(data['age_group_id'])).collect()[0][0])
# Zastąp brakujące wartości nowymi
# data.na.replace(old_value, new_vallue)

DataFrame[measure_name: string, location_id: int, location_name: string, sex_id: int, sex_name: string, age_group_id: int, age_group_name:
string, year_id: int, val: double, upper: double, lower: double]
```

6 - Pobieranie danych

6.1 - Select

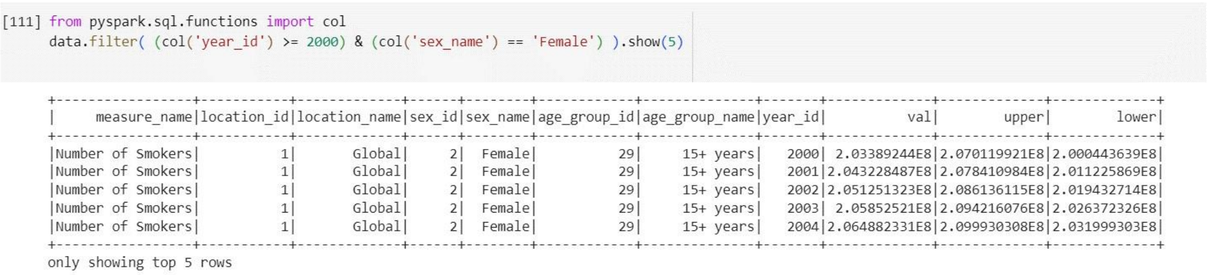


```
[109] # wybór jednej kolumny
      data.select('year_id').show(5)

+-------+
|year_id|
+-------+
|   1990|
|   1990|
|   1990|
|   1991|
|   1991|
+-------+
only showing top 5 rows
```



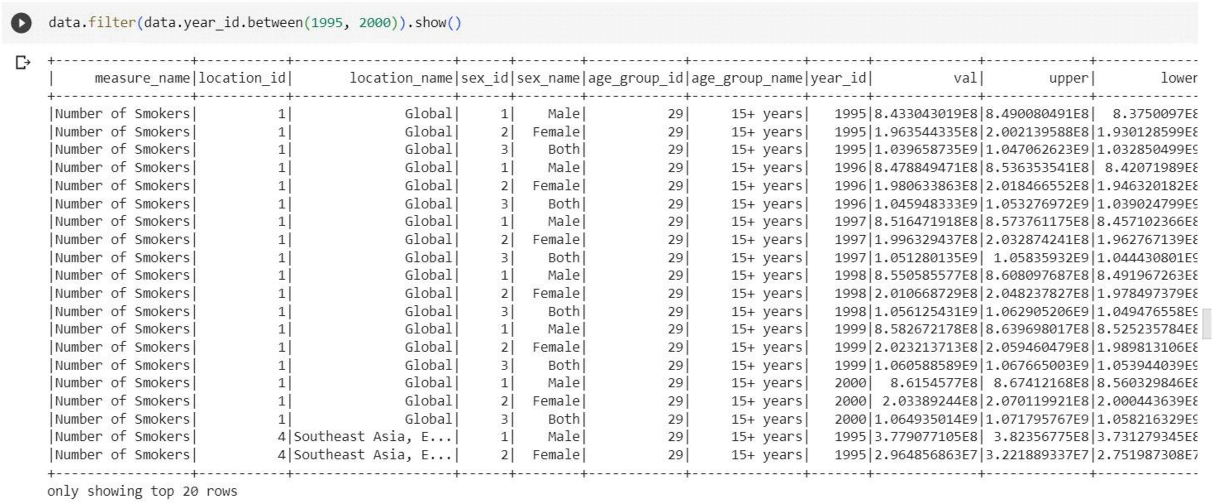
```
[ ] # wybór kilku kolumn
    data.select(['location_name', 'sex_name', 'year_id']).show(5)

    +-------------+--------+-------+
    |location_name|sex_name|year_id|
    +-------------+--------+-------+
    |       Global|    Male|   1990|
    |       Global|  Female|   1990|
    |       Global|    Both|   1990|
    |       Global|    Male|   1991|
    |       Global|  Female|   1991|
    +-------------+--------+-------+
    only showing top 5 rows
```

6.2 - Filter



```
[111] from pyspark.sql.functions import col
      data.filter( (col('year_id') >= 2000) & (col('sex_name') == 'Female') ).show(5)

      +-----------------+-----------+-------------+------+--------+------------+--------------+-------+-------------+-------------+-------------+
      |     measure_name|location_id|location_name|sex_id|sex_name|age_group_id|age_group_name|year_id|          val|        upper|        lower|
      +-----------------+-----------+-------------+------+--------+------------+--------------+-------+-------------+-------------+-------------+
      |Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   2000| 2.03389244E8|2.070119921E8|2.000443639E8|
      |Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   2001|2.043228487E8|2.078410984E8|2.011225869E8|
      |Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   2002|2.051251323E8|2.086136115E8|2.019432714E8|
      |Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   2003| 2.05852521E8|2.094216076E8|2.026372326E8|
      |Number of Smokers|          1|       Global|     2|  Female|          29|     15+ years|   2004|2.064882331E8|2.099930308E8|2.031999303E8|
      +-----------------+-----------+-------------+------+--------+------------+--------------+-------+-------------+-------------+-------------+
      only showing top 5 rows
```

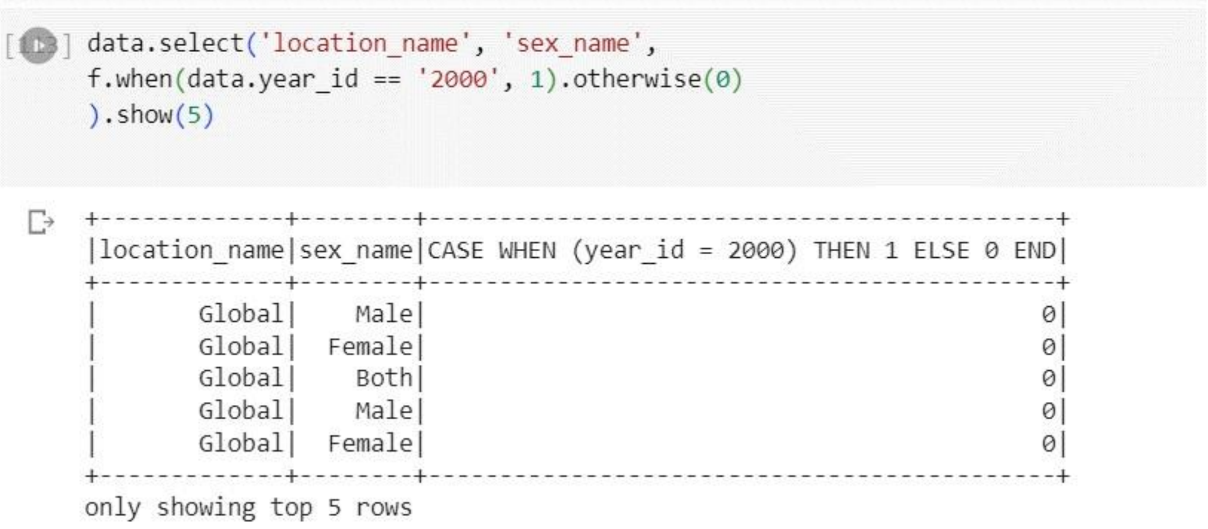
6.3 - Between



```
data.filter(data.year_id.between(1995, 2000)).show()

+-----------------+-----------+--------------------+------+--------+------------+--------------+-------+-------------+-------------+------------
|     measure_name|location_id|       location_name|sex_id|sex_name|age_group_id|age_group_name|year_id|          val|        upper|        lower
+-----------------+-----------+--------------------+------+--------+------------+--------------+-------+-------------+-------------+------------
|Number of Smokers|          1|              Global|     1|    Male|          29|     15+ years|   1995|8.433043019E8|8.490080491E8|  8.3750097E8
|Number of Smokers|          1|              Global|     2|  Female|          29|     15+ years|   1995|1.963544335E8|2.002139588E8|1.930128599E8
|Number of Smokers|          1|              Global|     3|    Both|          29|     15+ years|   1995|1.039658735E9|1.047062623E9|1.032850499E9
|Number of Smokers|          1|              Global|     1|    Male|          29|     15+ years|   1996|8.478849471E8|8.536353541E8| 8.42071989E8
|Number of Smokers|          1|              Global|     2|  Female|          29|     15+ years|   1996|1.980633863E8|2.018466552E8|1.946320182E8
|Number of Smokers|          1|              Global|     3|    Both|          29|     15+ years|   1996|1.045948333E9|1.053276972E9|1.039024799E9
|Number of Smokers|          1|              Global|     1|    Male|          29|     15+ years|   1997|8.516471918E8|8.573761175E8|8.457102366E8
|Number of Smokers|          1|              Global|     2|  Female|          29|     15+ years|   1997|1.996329437E8|2.032874241E8|1.962767139E8
|Number of Smokers|          1|              Global|     3|    Both|          29|     15+ years|   1997|1.051280135E9| 1.05835932E9|1.044430801E9
|Number of Smokers|          1|              Global|     1|    Male|          29|     15+ years|   1998|8.550585577E8|8.608097687E8|8.491967263E8
|Number of Smokers|          1|              Global|     2|  Female|          29|     15+ years|   1998|2.010668729E8|2.048237827E8|1.978497379E8
|Number of Smokers|          1|              Global|     3|    Both|          29|     15+ years|   1998|1.056125431E9|1.062905206E9|1.049476558E9
|Number of Smokers|          1|              Global|     1|    Male|          29|     15+ years|   1999|8.582672178E8|8.639698017E8|8.525235784E8
|Number of Smokers|          1|              Global|     2|  Female|          29|     15+ years|   1999|2.023213713E8|2.059460479E8|1.989813106E8
|Number of Smokers|          1|              Global|     3|    Both|          29|     15+ years|   1999|1.060588589E9|1.067665003E9|1.053944039E9
|Number of Smokers|          1|              Global|     1|    Male|          29|     15+ years|   2000|  8.6154577E8| 8.67412168E8|8.560329846E8
|Number of Smokers|          1|              Global|     2|  Female|          29|     15+ years|   2000| 2.03389244E8|2.070119921E8|2.000443639E8
|Number of Smokers|          1|              Global|     3|    Both|          29|     15+ years|   2000|1.064935014E9|1.071795767E9|1.058216329E9
|Number of Smokers|          4|Southeast Asia, E...|     1|    Male|          29|     15+ years|   1995|3.779077105E8| 3.82356775E8|3.731279345E8
|Number of Smokers|          4|Southeast Asia, E...|     2|  Female|          29|     15+ years|   1995|2.964856863E7|3.221889337E7|2.751987308E7
+-----------------+-----------+--------------------+------+--------+------------+--------------+-------+-------------+-------------+------------
only showing top 20 rows
```

6.4 When

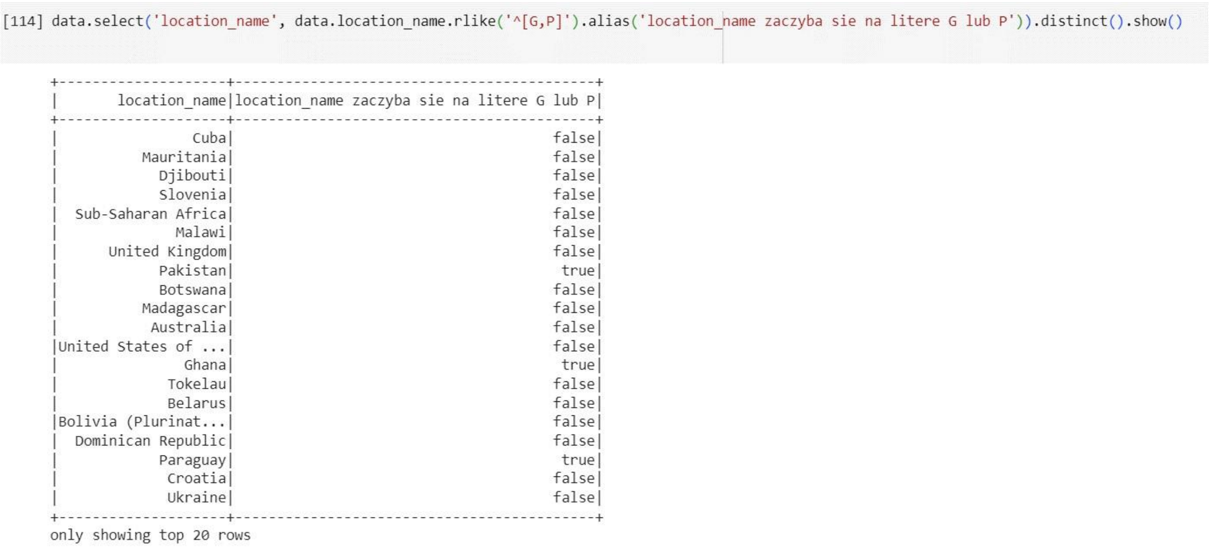
-



```
data.select('location_name', 'sex_name',
f.when(data.year_id == '2000', 1).otherwise(0)
).show(5)
```

```
+-------------+--------+--------------------------------------------+
|location_name|sex_name|CASE WHEN (year_id = 2000) THEN 1 ELSE 0 END|
+-------------+--------+--------------------------------------------+
|       Global|    Male|                                           0|
|       Global|  Female|                                           0|
|       Global|    Both|                                           0|
|       Global|    Male|                                           0|
|       Global|  Female|                                           0|
+-------------+--------+--------------------------------------------+
only showing top 5 rows
```

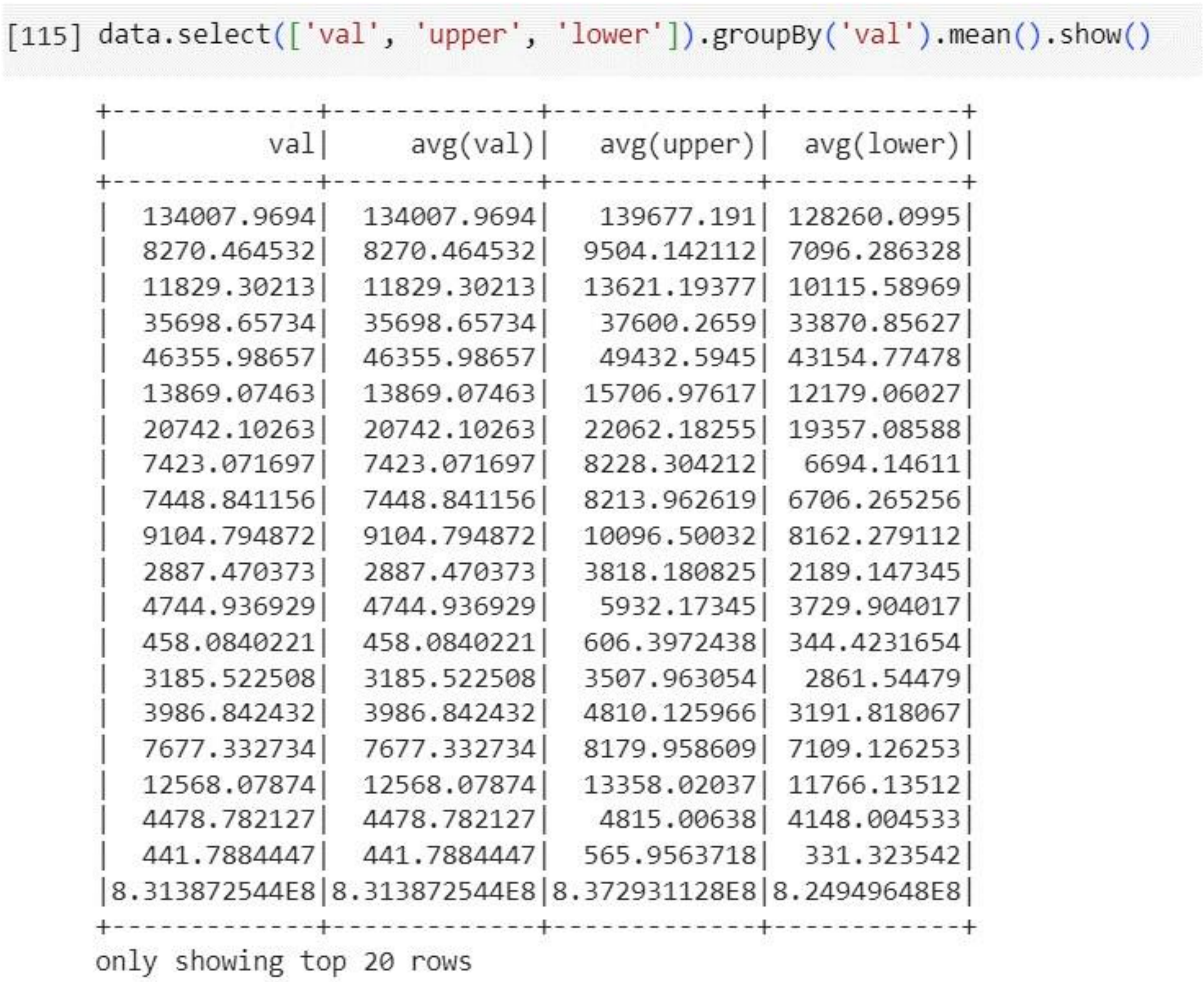
6.5 - Like



```
[114] data.select('location_name', data.location_name.rlike('^[G,P]').alias('location_name zaczyba sie na litere G lub P')).distinct().show()
```

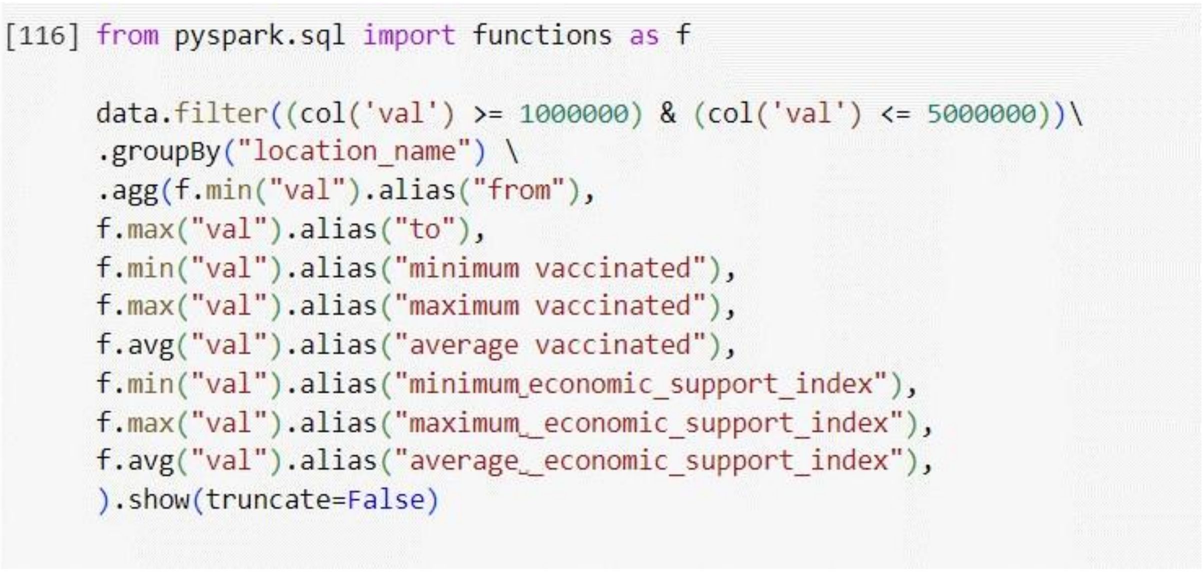
```
+-------------------+-------------------------------------------+
|      location_name|location_name zaczyba sie na litere G lub P|
+-------------------+-------------------------------------------+
|               Cuba|                                      false|
|         Mauritania|                                      false|
|           Djibouti|                                      false|
|           Slovenia|                                      false|
| Sub-Saharan Africa|                                      false|
|             Malawi|                                      false|
|     United Kingdom|                                      false|
|           Pakistan|                                       true|
|           Botswana|                                      false|
|         Madagascar|                                      false|
|          Australia|                                      false|
|United States of ...|                                      false|
|              Ghana|                                       true|
|            Tokelau|                                      false|
|            Belarus|                                      false|
|Bolivia (Plurinat...|                                      false|
| Dominican Republic|                                      false|
|           Paraguay|                                       true|
|            Croatia|                                      false|
|            Ukraine|                                      false|
+-------------------+-------------------------------------------+
only showing top 20 rows
```

6.6 - GroupBy

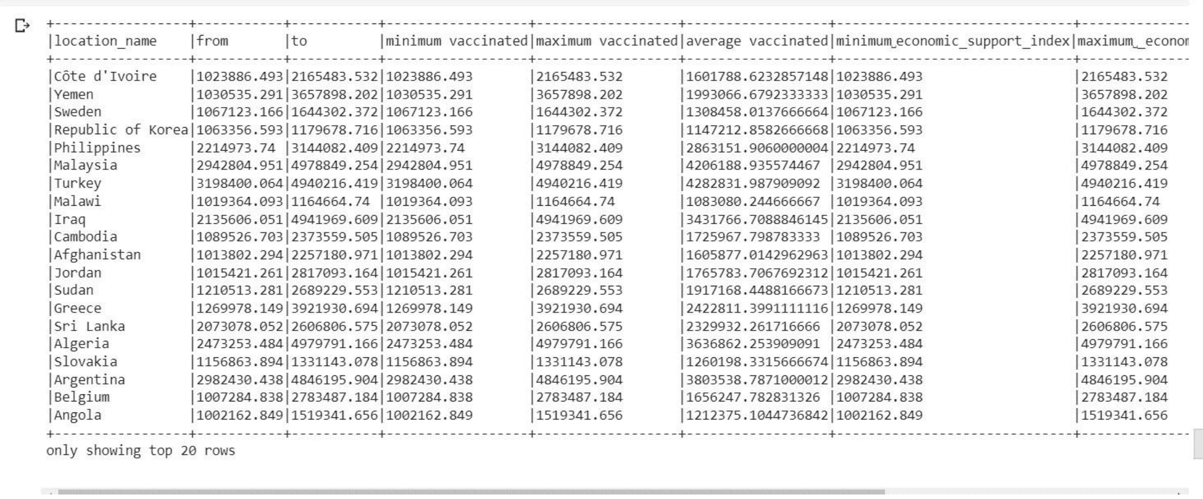


```
[115] data.select(['val', 'upper', 'lower']).groupBy('val').mean().show()

      +------------+------------+------------+-----------+
      |         val|    avg(val)|  avg(upper)| avg(lower)|
      +------------+------------+------------+-----------+
      | 134007.9694| 134007.9694|  139677.191|128260.0995|
      | 8270.464532| 8270.464532| 9504.142112|7096.286328|
      | 11829.30213| 11829.30213| 13621.19377|10115.58969|
      | 35698.65734| 35698.65734|  37600.2659|33870.85627|
      | 46355.98657| 46355.98657|  49432.5945|43154.77478|
      | 13869.07463| 13869.07463| 15706.97617|12179.06027|
      | 20742.10263| 20742.10263| 22062.18255|19357.08588|
      | 7423.071697| 7423.071697| 8228.304212| 6694.14611|
      | 7448.841156| 7448.841156| 8213.962619|6706.265256|
      | 9104.794872| 9104.794872| 10096.50032|8162.279112|
      | 2887.470373| 2887.470373| 3818.180825|2189.147345|
      | 4744.936929| 4744.936929|  5932.17345|3729.904017|
      | 458.0840221| 458.0840221| 606.3972438|344.4231654|
      | 3185.522508| 3185.522508| 3507.963054| 2861.54479|
      | 3986.842432| 3986.842432| 4810.125966|3191.818067|
      | 7677.332734| 7677.332734| 8179.958609|7109.126253|
      | 12568.07874| 12568.07874| 13358.02037|11766.13512|
      | 4478.782127| 4478.782127|  4815.00638|4148.004533|
      | 441.7884447| 441.7884447| 565.9563718| 331.323542|
      |8.313872544E8|8.313872544E8|8.372931128E8|8.24949648E8|
      +------------+------------+------------+-----------+
      only showing top 20 rows
```

6.7 - Agregacja

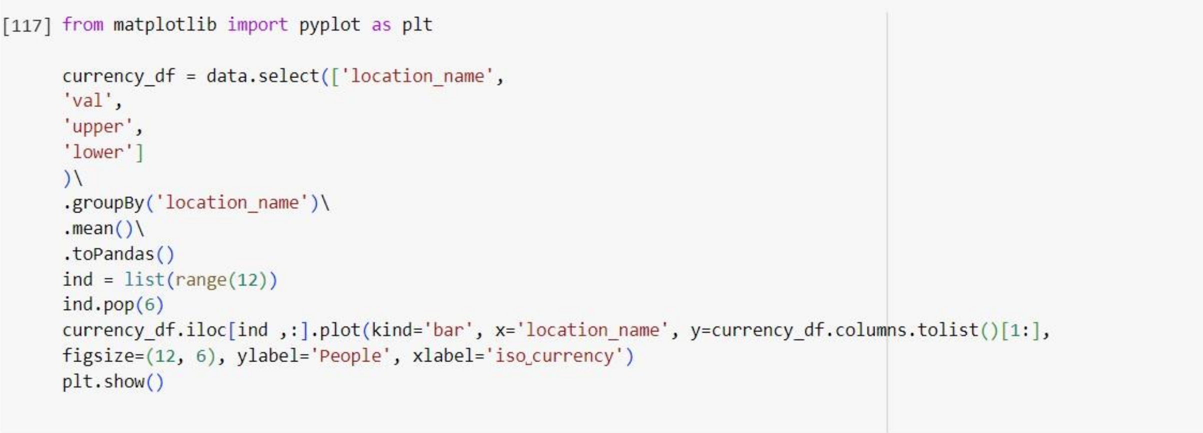


```
[116] from pyspark.sql import functions as f

      data.filter((col('val') >= 1000000) & (col('val') <= 5000000))\
      .groupBy("location_name") \
      .agg(f.min("val").alias("from"),
      f.max("val").alias("to"),
      f.min("val").alias("minimum vaccinated"),
      f.max("val").alias("maximum vaccinated"),
      f.avg("val").alias("average vaccinated"),
      f.min("val").alias("minimum_economic_support_index"),
      f.max("val").alias("maximum__economic_support_index"),
      f.avg("val").alias("average__economic_support_index"),
      ).show(truncate=False)
```

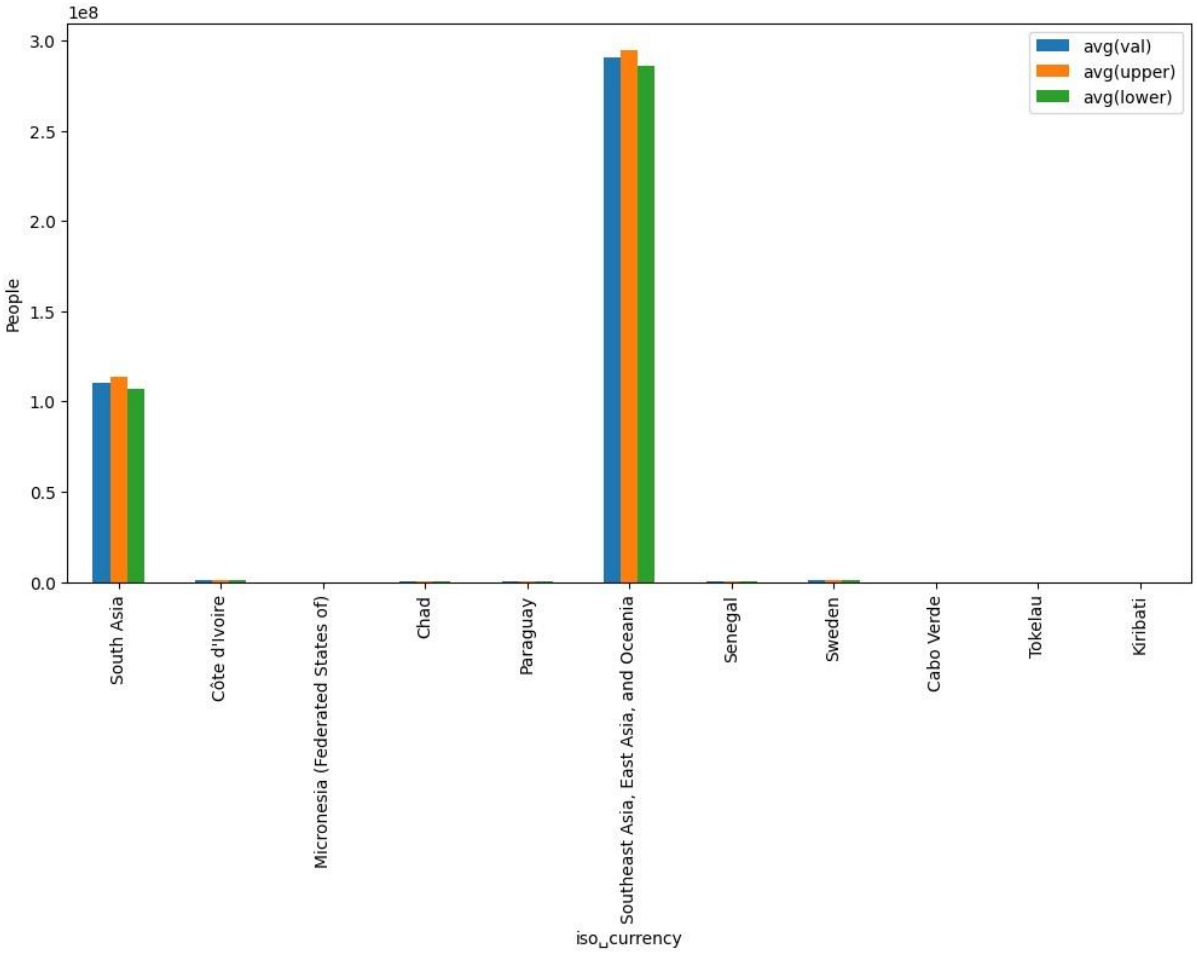


```
+-----------------+-----------+-----------+------------------+------------------+------------------+------------------------------+--------------
|location_name    |from       |to         |minimum vaccinated|maximum vaccinated|average vaccinated|minimum_economic_support_index|maximum_econom
+-----------------+-----------+-----------+------------------+------------------+------------------+------------------------------+--------------
|Côte d'Ivoire    |1023886.493|2165483.532|1023886.493       |2165483.532       |1601788.6232857148|1023886.493                   |2165483.532
|Yemen            |1030535.291|3657898.202|1030535.291       |3657898.202       |1993066.6792333333|1030535.291                   |3657898.202
|Sweden           |1067123.166|1644302.372|1067123.166       |1644302.372       |1308458.0137666664|1067123.166                   |1644302.372
|Republic of Korea|1063356.593|1179678.716|1063356.593       |1179678.716       |1147212.8582666668|1063356.593                   |1179678.716
|Philippines      |2214973.74 |3144082.409|2214973.74        |3144082.409       |2863151.9060000004|2214973.74                    |3144082.409
|Malaysia         |2942804.951|4978849.254|2942804.951       |4978849.254       |4206188.935574467 |2942804.951                   |4978849.254
|Turkey           |3198400.064|4940216.419|3198400.064       |4940216.419       |4282831.987909092 |3198400.064                   |4940216.419
|Malawi           |1019364.093|1164664.74 |1019364.093       |1164664.74        |1083080.244666667 |1019364.093                   |1164664.74
|Iraq             |2135606.051|4941969.609|2135606.051       |4941969.609       |3431766.7088846145|2135606.051                   |4941969.609
|Cambodia         |1089526.703|2373559.505|1089526.703       |2373559.505       |1725967.798783333 |1089526.703                   |2373559.505
|Afghanistan      |1013802.294|2257180.971|1013802.294       |2257180.971       |1605877.0142962963|1013802.294                   |2257180.971
|Jordan           |1015421.261|2817093.164|1015421.261       |2817093.164       |1765783.7067692312|1015421.261                   |2817093.164
|Sudan            |1210513.281|2689229.553|1210513.281       |2689229.553       |1917168.4488166673|1210513.281                   |2689229.553
|Greece           |1269978.149|3921930.694|1269978.149       |3921930.694       |2422811.3991111116|1269978.149                   |3921930.694
|Sri Lanka        |2073078.052|2606806.575|2073078.052       |2606806.575       |2329932.261716666 |2073078.052                   |2606806.575
|Algeria          |2473253.484|4979791.166|2473253.484       |4979791.166       |3636862.253909091 |2473253.484                   |4979791.166
|Slovakia         |1156863.894|1331143.078|1156863.894       |1331143.078       |1260198.3315666674|1156863.894                   |1331143.078
|Argentina        |2982430.438|4846195.904|2982430.438       |4846195.904       |3803538.7871000012|2982430.438                   |4846195.904
|Belgium          |1007284.838|2783487.184|1007284.838       |2783487.184       |1656247.782831326 |1007284.838                   |2783487.184
|Angola           |1002162.849|1519341.656|1002162.849       |1519341.656       |1212375.1044736842|1002162.849                   |1519341.656
+-----------------+-----------+-----------+------------------+------------------+------------------+------------------------------+--------------
only showing top 20 rows
```

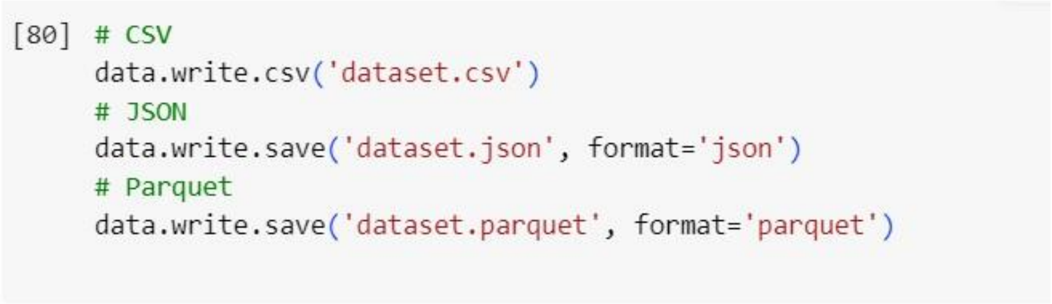
7 - wizualizacja danych



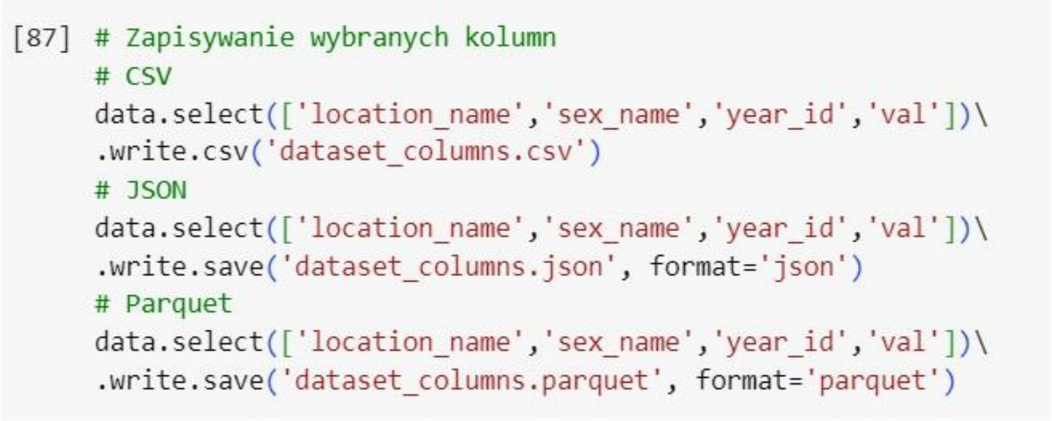
```
[117] from matplotlib import pyplot as plt

      currency_df = data.select(['location_name',
      'val',
      'upper',
      'lower']
      )\
      .groupBy('location_name')\
      .mean()\
      .toPandas()
      ind = list(range(12))
      ind.pop(6)
      currency_df.iloc[ind ,:].plot(kind='bar', x='location_name', y=currency_df.columns.tolist()[1:],
      figsize=(12, 6), ylabel='People', xlabel='iso_currency')
      plt.show()
```



8 - Zapisywanie/zapisywanie danych do pliku



```
[80] # CSV
     data.write.csv('dataset.csv')
     # JSON
     data.write.save('dataset.json', format='json')
     # Parquet
     data.write.save('dataset.parquet', format='parquet')
```



```
[87] # Zapisywanie wybranych kolumn
     # CSV
     data.select(['location_name','sex_name','year_id','val'])\
     .write.csv('dataset_columns.csv')
     # JSON
     data.select(['location_name','sex_name','year_id','val'])\
     .write.save('dataset_columns.json', format='json')
     # Parquet
     data.select(['location_name','sex_name','year_id','val'])\
     .write.save('dataset_columns.parquet', format='parquet')
```

3. Wnioski

Na podstawie otrzymanego wyniku można stwierdzić, że biblioteka PySpark używana jest do przetwarzania dużych zbiorów danych. Używana podczas zajęć biblioteka pozwala na wczytywanie danych z różnych formatów: CSV, JSON, Parquet, jak również za pomocą URL. Biblioteka PySpark pozwala manipulować kolumnami: dodawać kolumnę, zmienić nazwę kolumnie, oraz usunąć kolumnę. PySpark i PySpark SQL zapewniają szeroki zakres metod i funkcji do łatwego wyszukiwania danych.