Instalacja wymaganych pakietów

```
! pip install pyspark==3.0.1 py4j==0.10.9
```

```
Requirement already satisfied: pyspark==3.0.1 in d:\programdata\
anaconda3\lib\site-packages (3.0.1)
Requirement already satisfied: py4j==0.10.9 in d:\programdata\
anaconda3\lib\site-packages (0.10.9)
```

Tworzenie Spark session

```
from pyspark.sql import SparkSession
spark = SparkSession.builder\
        .master("local[*]")\
        .appName('PySpark_Tutorial')\
        .getOrCreate()
# gdzie "*" znaczy wszystkie rdzenie procesora.
```

Czytanie danych

```
# Czytanie CSV plika
csv_file = 'IHME-GBD_2019_DATA-15798851-2.csv'
df = spark.read.csv(csv_file)
```

Strukturyzacja danych za pomocą schematu Spark

```
data = spark.read.csv(
      "IHME-GBD_2019_DATA-15798851-2.csv",
      sep=',',
      header=True,
      )
data.printSchema()
```

```
root
 |-- measure: string (nullable = true)
 |-- location: string (nullable = true)
 |-- sex: string (nullable = true)
 |-- age: string (nullable = true)
 |-- cause: string (nullable = true)
 |-- metric: string (nullable = true)
 |-- year: string (nullable = true)
 |-- val: string (nullable = true)
 |-- upper: string (nullable = true)
 |-- lower: string (nullable = true)
```

Manualna strukturyzacja danych

```
from pyspark.sql.types import *

data_schema = [
    StructField('measure', StringType(), True), #czy dopuszczalna jest
```

```
wartość null
    StructField('location', StringType(), True),
    StructField('year', IntegerType(), False),
    StructField('var', DoubleType(), False),
]

final_struc = StructType(fields = data_schema)
data2 = spark.read.csv(
      "IHME-GBD_2019_DATA-15798851-2.csv",
      sep=',',
      header=True,
      schema=final_struc
)

data2.printSchema()
```

```
root
 |-- measure: string (nullable = true)
 |-- location: string (nullable = true)
 |-- year: integer (nullable = true)
 |-- var: double (nullable = true)
```

Kontrola danych

```
data2.schema
```

```
StructType(List(StructField(measure,StringType,true),StructField(locat
ion,StringType,true),StructField(year,IntegerType,true),StructField(va
r,DoubleType,true)))
```

```
data2.dtypes
```

```
[('measure', 'string'),
 ('location', 'string'),
 ('year', 'int'),
 ('var', 'double')]
```

```
data2.head
```

```
<bound method DataFrame.head of DataFrame[measure: string, location:
string, year: int, var: double]>
```

Manipulacja kolumnami

```
data = data.withColumn('copy_location', data.location)
data.show(5)
```

```
+--------------------+--------+------+--------+--------------------
+------+----+-----------------+-----------------+-----------------
+-------------+
|             measure|location|   sex|     age|               cause|
```

```
metric|year|              val|             upper|             lower|
copy_location|
+--------------------+--------+------+--------+--------------------
+------+----+------------------+------------------+------------------
+-------------+
|DALYs (Disability...|  Gambia|Female|All Ages|Maternal and neon...|
Rate|2012| 7475.212699705153| 9104.773540846287| 6157.428602624385|
Gambia|
|DALYs (Disability...|  Gambia|  Both|All Ages|Maternal and neon...|
Rate|2012| 7814.344518002015| 9667.960848348446| 6289.146374740097|
Gambia|
|DALYs (Disability...|  Gambia|  Male|All Ages|Substance use dis...|
Number|2012| 1659.038707247863| 2126.829520886102|1239.1726985245457|
Gambia|
|DALYs (Disability...|  Gambia|Female|All Ages|Substance use dis...|
Number|2012| 874.4324658085982|1186.5605963880798| 618.2717801609034|
Gambia|
|DALYs (Disability...|  Gambia|  Both|All Ages|Substance use dis...|
Number|2012|2533.4711730564563| 3231.220866423626|1868.2046086417708|
Gambia|
+--------------------+--------+------+--------+--------------------
+------+----+------------------+------------------+------------------
+-------------+
only showing top 5 rows


data = data.withColumnRenamed('copy_location', 'copy_location2')
#zmiana nazwy
data.show(5)

+--------------------+--------+------+--------+--------------------
+------+----+------------------+------------------+------------------
+--------------+
|             measure|location|   sex|     age|               cause|
metric|year|              val|             upper|             lower|
copy_location2|
+--------------------+--------+------+--------+--------------------
+------+----+------------------+------------------+------------------
+--------------+
|DALYs (Disability...|  Gambia|Female|All Ages|Maternal and neon...|
Rate|2012| 7475.212699705153| 9104.773540846287| 6157.428602624385|
Gambia|
|DALYs (Disability...|  Gambia|  Both|All Ages|Maternal and neon...|
Rate|2012| 7814.344518002015| 9667.960848348446| 6289.146374740097|
Gambia|
|DALYs (Disability...|  Gambia|  Male|All Ages|Substance use dis...|
Number|2012| 1659.038707247863| 2126.829520886102|1239.1726985245457|
Gambia|
|DALYs (Disability...|  Gambia|Female|All Ages|Substance use dis...|
Number|2012| 874.4324658085982|1186.5605963880798| 618.2717801609034|
```

```
Gambia|
|DALYs (Disability...|  Gambia|  Both|All Ages|Substance use dis...|
Number|2012|2533.4711730564563| 3231.220866423626|1868.2046086417708|
Gambia|
+--------------------+--------+------+--------+--------------------
+------+----+------------------+------------------+------------------
+--------------+
only showing top 5 rows


data = data.drop('copy_location2') #kasacja
data.show(5)

+--------------------+--------+------+--------+--------------------
+------+----+------------------+------------------+------------------+
|             measure|location|   sex|     age|               cause|
metric|year|               val|             upper|             lower|
+--------------------+--------+------+--------+--------------------
+------+----+------------------+------------------+------------------+
|DALYs (Disability...|  Gambia|Female|All Ages|Maternal and neon...|
Rate|2012| 7475.212699705153| 9104.773540846287| 6157.428602624385|
|DALYs (Disability...|  Gambia|  Both|All Ages|Maternal and neon...|
Rate|2012| 7814.344518002015| 9667.960848348446| 6289.146374740097|
|DALYs (Disability...|  Gambia|  Male|All Ages|Substance use dis...|
Number|2012| 1659.038707247863| 2126.829520886102|1239.1726985245457|
|DALYs (Disability...|  Gambia|Female|All Ages|Substance use dis...|
Number|2012| 874.4324658085982|1186.5605963880798| 618.2717801609034|
|DALYs (Disability...|  Gambia|  Both|All Ages|Substance use dis...|
Number|2012|2533.4711730564563| 3231.220866423626|1868.2046086417708|
+--------------------+--------+------+--------+--------------------
+------+----+------------------+------------------+------------------+
only showing top 5 rows
```

Radzenie sobie z brakującymi wartościami

```
data.show
```

```
<bound method DataFrame.show of DataFrame[measure: string, location:
string, sex: string, age: string, cause: string, metric: string, year:
string, val: string, upper: string, lower: string]>
```

```
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['val'])).collect()[0][0])
# Zastąp brakujące wartości nowymi
#data.na.replace(old_value, new_vallue)
```

```
DataFrame[measure: string, location: string, sex: string, age: string,
cause: string, metric: string, year: string, val: string, upper:
string, lower: string]
```

Pobieranie danych

```
data.select('year').show(5)
```

```
+----+
|year|
+----+
|2012|
|2012|
|2012|
|2012|
|2012|
+----+
only showing top 5 rows
```

```
# wybór kilku kolumn
data.select(['location', 'year', 'val']).show(10)
```

```
+--------+----+--------------------+
|location|year|                 val|
+--------+----+--------------------+
|  Gambia|2012|   7475.212699705153|
|  Gambia|2012|   7814.344518002015|
|  Gambia|2012|   1659.038707247863|
|  Gambia|2012|   874.4324658085982|
|  Gambia|2012|  2533.4711730564563|
|  Gambia|2012|0.003798563072089447|
|  Gambia|2012|0.002202396944719217|
|  Gambia|2012|0.003038293155919...|
|  Gambia|2012|  179.49365989601577|
|  Gambia|2012|   91.40054305937956|
+--------+----+--------------------+
only showing top 10 rows
```

Filter

```
from pyspark.sql.functions import col
data.filter( (col('val') >= 1000) & (col('upper') <= 10000000) )
data.show(5)
```

```
+--------------------+--------+------+--------+--------------------
+------+----+------------------+------------------+------------------+
|             measure|location|   sex|     age|               cause|
metric|year|               val|             upper|             lower|
+--------------------+--------+------+--------+--------------------
+------+----+------------------+------------------+------------------+
```

```
|DALYs (Disability...|  Gambia|Female|All Ages|Maternal and neon...|
Rate|2012| 7475.212699705153| 9104.773540846287| 6157.428602624385|
|DALYs (Disability...|  Gambia|  Both|All Ages|Maternal and neon...|
Rate|2012| 7814.344518002015| 9667.960848348446| 6289.146374740097|
|DALYs (Disability...|  Gambia|  Male|All Ages|Substance use dis...|
Number|2012| 1659.038707247863| 2126.829520886102|1239.1726985245457|
|DALYs (Disability...|  Gambia|Female|All Ages|Substance use dis...|
Number|2012| 874.4324658085982|1186.5605963880798| 618.2717801609034|
|DALYs (Disability...|  Gambia|  Both|All Ages|Substance use dis...|
Number|2012|2533.4711730564563| 3231.220866423626|1868.2046086417708|
+--------------------+--------+------+--------+-------------------
+------+----+------------------+------------------+------------------+
only showing top 5 rows
```

Between

```
data.filter(data.val.between(1000000, 5000000)).show()
```

```
+--------------------+------------------+------+--------
+--------------------+------+----+------------------
+------------------+------------------+
|             measure|          location|   sex|     age|
cause|metric|year|               val|             upper|
lower|
+--------------------+------------------+------+--------
+--------------------+------+----+------------------
+------------------+------------------+
|DALYs (Disability...|Russian Federation|  Male|All Ages|  Transport
injuries|Number|2013|1563818.2525840718|1730665.1380830628|
1422922.2224031396|
|DALYs (Disability...|Russian Federation|  Both|All Ages|  Transport
injuries|Number|2013|2348269.3054279066|2665389.1249823105|
2068902.188470523|
|DALYs (Disability...|          Thailand|  Both|All Ages|Diabetes and
kidn...|Number|2013|1215245.6178658458| 1387336.100516322|
1055619.0224713387|
|DALYs (Disability...|        Mozambique|  Male|All Ages|HIV/AIDS and
sexu...|Number|2011|2174300.0441490347|2741039.0572092957|
1763039.0810975558|
|DALYs (Disability...|        Mozambique|Female|All Ages|HIV/AIDS and
sexu...|Number|2011|   2895587.7646052|3706408.5720942672|
2306640.354706672|
|DALYs (Disability...|Russian Federation|  Male|All Ages|
Neoplasms|Number|2013| 4090226.802378882| 4155359.464282655|
4005121.2019642727|
|DALYs (Disability...|Russian Federation|Female|All Ages|
Neoplasms|Number|2013|3414103.8508302304|3489028.7047208236|
3310484.1497073723|
|DALYs (Disability...|Russian Federation|  Male|All Ages|  Digestive
diseases|Number|2013|2002567.5882994307|2073526.2177201917|
```

```
1947072.2245896638|
|DALYs (Disability...|Russian Federation|Female|All Ages|  Digestive
diseases|Number|2013|1345203.4673331394|1444258.8697134608|
1266022.689466917|
|DALYs (Disability...|Russian Federation|  Both|All Ages|  Digestive
diseases|Number|2013| 3347771.055632565| 3516635.859878875|
3219696.985016068|
|DALYs (Disability...|              Iraq|  Both|All Ages|Self-harm and
int...|Number|2012| 1256318.149434527|1462101.5208953691|
1089259.7260939889|
|DALYs (Disability...|            Mexico|  Male|All Ages|Diabetes and
kidn...|Number|2012|1858981.4749701458| 2036028.626377141|
1700214.864927907|
|DALYs (Disability...|            Mexico|Female|All Ages|Diabetes and
kidn...|Number|2012|1861185.8003768912|2054662.3057531568|
1681539.0798739342|
|DALYs (Disability...|            Mexico|  Both|All Ages|Diabetes and
kidn...|Number|2012| 3720167.275347037| 4087246.320550209|
3386894.5316912797|
|DALYs (Disability...|          Viet Nam|  Both|All Ages|Neurological
diso...|Number|2012|1069819.6566094689|1928188.7872334127|
501455.5305686765|
|DALYs (Disability...|          Ethiopia|  Both|All Ages|Unintentional
inj...|Number|2012|1378821.0629198356|1693361.7187471045|
1115172.7929996278|
|DALYs (Disability...|          Pakistan|  Male|All Ages|Respiratory
infec...|Number|2011| 4863470.709816628|5817474.0356123205|
4014095.045441964|
|DALYs (Disability...|          Pakistan|Female|All Ages|Respiratory
infec...|Number|2011| 4784176.011975201|  5569335.86283441|
4083265.6903291587|
|DALYs (Disability...|          Pakistan|  Male|All Ages|  Enteric
infections|Number|2011|4099624.9035108723| 5279170.728818483|
3021833.1130831456|
|DALYs (Disability...|          Pakistan|Female|All Ages|  Enteric
infections|Number|2011|3865104.1411546906| 4906030.420062346|
3022509.8843845455|
+--------------------+------------------+------+--------
+--------------------+------+----+------------------
+------------------+------------------+
only showing top 20 rows
```

When

```
data.select('year', 'val',
f.when(data.year == '2012', 1).otherwise(0)
).show(25)
```

```
+----+--------------------+----------------------------------------+
|year|                 val|CASE WHEN (year = 2012) THEN 1 ELSE 0 END|
```

```
+----+--------------------+----------------------------------------+
|2012|   7475.212699705153|                                       1|
|2012|   7814.344518002015|                                       1|
|2012|   1659.038707247863|                                       1|
|2012|   874.4324658085982|                                       1|
|2012|  2533.4711730564563|                                       1|
|2012|0.003798563072089447|                                       1|
|2012|0.002202396944719217|                                       1|
|2012|0.003038293155919...|                                       1|
|2012|  179.49365989601577|                                       1|
|2012|   91.40054305937956|                                       1|
|2012|   134.6880361780433|                                       1|
|2012|   8646.107047081588|                                       1|
|2012|   7619.998216253456|                                       1|
|2012|   16266.10526333505|                                       1|
|2012| 0.01977817597188412|                                       1|
|2012| 0.01919796110415631|                                       1|
|2012| 0.01950424101322522|                                       1|
|2012|    935.434110701281|                                       1|
|2012|   796.4845797817376|                                       1|
|2012|   864.7620693235743|                                       1|
|2011|   22301.33040284042|                                       0|
|2011|  17657.116392824933|                                       0|
|2011|   39958.44679566539|                                       0|
|2011| 0.04799795589878509|                                       0|
|2011|0.042699118371464485|                                       0|
+----+--------------------+----------------------------------------+
only showing top 25 rows
```

Like

```
data.select(
'val',
data.val.rlike('^[9,7]').alias('iso_urrency zaczyba sie na 9
lub␣7')).distinct().show()
```

```
+--------------------+---------------------------------+
|                 val|iso_urrency zaczyba sie na 9 lub␣7|
+--------------------+---------------------------------+
|0.007332988142041626|                            false|
|  0.2713034153257434|                            false|
| 0.06367565466684018|                            false|
| 0.19342156900472102|                            false|
|0.003835061317120...|                            false|
|0.032178074330139694|                            false|
|   7093.222503810944|                             true|
|   7502.956806363028|                             true|
| 0.08536612762995582|                            false|
|  260706.33550143513|                            false|
|  1024.1460052312696|                            false|
```

```
|  1336.6902679035634|                                 false|
|  227114.41419074405|                                 false|
|  1332.9571704727955|                                 false|
|  22580246.363224022|                                 false|
| 0.06428783851451203|                                 false|
|  7222.4012756562925|                                  true|
|0.011527400193312977|                                 false|
|  2120.0737120596864|                                 false|
|   6715.655340340026|                                 false|
+--------------------+-----------------------------------+
only showing top 20 rows
```

GroupBy

```
data.groupBy('year').count().show()
```

```
+----+-----+
|year|count|
+----+-----+
|2016|40392|
|2012|24781|
|2019|40392|
|2017|40392|
|2014|40392|
|2013|39483|
|2018|40392|
|2011| 1224|
|2015|40392|
+----+-----+
```

Agregacja

```
from pyspark.sql import functions as f

data.groupBy("year").agg(f.mean("val")).show()
#grupowanie i obliczanie wartości dla grup
```

```
+----+------------------+
|year|          avg(val)|
+----+------------------+
|2016|125761.03225030862|
|2012|121301.00826382442|
|2019|126113.55401705152|
|2017|125589.93283513733|
|2014|125758.36794698932|
|2013|127595.00698396392|
|2018|125696.43818551343|
```

```
|2011|128072.01051363212|
|2015|125897.23935898131|
+----+------------------+
```

Wizualizacja danych

```
from pyspark.sql.functions import col, min, max

df = data.select('year', 'val')\
      .groupBy("year")\
      .agg(min("val").alias("val_min"),
           max("val").alias("val_max"))\
      .toPandas()
df.head(10)
```

```
   year                 val_min            val_max
0  2016  0.00010176921862111648  999610.4490388336
1  2012  0.00011744952252551429   9994.58618079852
2  2019  0.00010105487549400276  9997.116815557363
3  2017  0.00010662263746395744  99996.76636858631
4  2014  0.00010864893662670056  9997.115100476538
5  2013  0.00010071083440063533  99999.98918410507
6  2018  0.00010488734069313558  9996.475911845675
7  2011   0.0008754420500049157   997.923977543644
8  2015   0.00010292863081047 61  99994.52912391476
```

Zapisywanie danych do pliku

```
# error gdy plik już istnieje
data.write.csv('dataset.csv')
data.write.csv('dataset.json', format='json')
data.write.csv('dataset.parquet', format='parquet')
# wybrane kolumny
data.select(['location_name',
'the_total_mean']).write.csv('dataset.csv')
```

```
---------------------------------------------------------------------------
-----
AnalysisException                         Traceback (most recent call
last)
Cell In[23], line 2
      1 # error gdy plik już istnieje
----> 2 data.write.csv('dataset.csv')
      3 data.write.csv('dataset.json', format='json')
      4 data.write.csv('dataset.parquet', format='parquet')

File D:\ProgramData\anaconda3\lib\site-packages\pyspark\sql\
readwriter.py:1027, in DataFrameWriter.csv(self, path, mode,
compression, sep, quote, escape, header, nullValue, escapeQuotes,
quoteAll, dateFormat, timestampFormat, ignoreLeadingWhiteSpace,
```

```
ignoreTrailingWhiteSpace, charToEscapeQuoteEscaping, encoding,
emptyValue, lineSep)
   1019 self.mode(mode)
   1020 self._set_opts(compression=compression, sep=sep, quote=quote,
escape=escape, header=header,
   1021                nullValue=nullValue, escapeQuotes=escapeQuotes,
quoteAll=quoteAll,
   1022                dateFormat=dateFormat,
timestampFormat=timestampFormat,
   (...)
   1025
charToEscapeQuoteEscaping=charToEscapeQuoteEscaping,
   1026                encoding=encoding, emptyValue=emptyValue,
lineSep=lineSep)
-> 1027 self._jwrite.csv(path)

File D:\ProgramData\anaconda3\lib\site-packages\py4j\
java_gateway.py:1304, in JavaMember.__call__(self, *args)
   1298 command = proto.CALL_COMMAND_NAME +\
   1299     self.command_header +\
   1300     args_command +\
   1301     proto.END_COMMAND_PART
   1303 answer = self.gateway_client.send_command(command)
-> 1304 return_value = get_return_value(
   1305     answer, self.gateway_client, self.target_id, self.name)
   1307 for temp_arg in temp_args:
   1308     temp_arg._detach()

File D:\ProgramData\anaconda3\lib\site-packages\pyspark\sql\
utils.py:134, in capture_sql_exception.<locals>.deco(*a, **kw)
    130 converted = convert_exception(e.java_exception)
    131 if not isinstance(converted, UnknownException):
    132     # Hide where the exception came from that shows a non-
Pythonic
    133     # JVM exception message.
--> 134     raise_from(converted)
    135 else:
    136     raise

File <string>:3, in raise_from(e)

AnalysisException: path
file:/C:/Users/Mikołaj/Desktop/Jupiter/dataset.csv already exists.;
```