https://tinyurl.com/wz2rbso

1. Nowe kolumny

```
cdf=cdf.withColumn('Area (sq mi)', (cdf['Area (sq km)']*0.3861).cast(IntegerType())).\
      withColumn('Pop Density (per sq mi)', cdf['Pop Density (per sq km)']/0.3861)

cdf.select('Country','Area (sq mi)', 'Area (sq km)', 'Pop Density (per sq mi)', 'Pop Density
(per sq km)').toPandas()
```

2 Wczytanie pliku airports - skopiować z arkusza Zadania 1

```
airports = spark.read.csv("airports.csv",inferSchema=True,header=False).\
toDF("id","airport","city","country","iata","icao","latitude","longitude","altitude","timezo
ne","dst","tz_timezone","type","data_source")
```

Uwaga: właściwy plik airports.csv do pobrania stąd:
https://drive.google.com/open?id=1gISWlbZ19UOXwJioUUhLHAnGv0WqHbrm

Liczba lotnisk w poszczególnych krajach

```
from pyspark.sql.functions import count

a=airports.groupBy('country').agg(count('airport').alias('AirportCount')).sort(col('AirportC
ount').desc()).alias('a')
b=cdf.alias('b')

# złączona ramka
gp_joined=a.join(b, a.country==b.Country).select('b.Country', 'b.Area (sq km)',
'a.AirportCount')
gp_joined.toPandas()
```

Wykres:

```
gp=gp_joined.toPandas()
ax=gp.plot.scatter('Area (sq km)', 'AirportCount')
for k, v in gp.iterrows():
      x=v['Area (sq km)']
      y=v['AirportCount']
      label=v['Country']
      plt.scatter(x,y,s=50)
```

Dane smogowe

https://openaq-fetches.s3.amazonaws.com/index.html

Formaty danych:

- CSV
- JSON
- XML
- Inne (protobuf, YAML, …)

ndjson - newline-delimited JSON — każdy wiersz pliku zawiera poprawny dokument JSON (np. pomiar)

Uprawnienia: Należy utworzyć **rolę** (w usłudze IAM) z uprawnieniami S3 Full access i przypisać ją do naszej instancji EC2.

“Spłaszczenie” schematu danych ładowanych z JSON:

```
smog3_df=smog2_df.select("location", "city", "parameter", "unit", "value",\
                         col("date.local").alias('date_local'),\
                         col('date.utc').alias('date_utc'),\
                         'coordinates.*')
smog3_df.printSchema()
```