SPRAWOZDANIE

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

Laboratorium: 1
Temat: " Ustalenia platformu Jupyter. Użycie biblioteki pandas w celu eksploracji i wizualizacji danych "
Wariant: 2

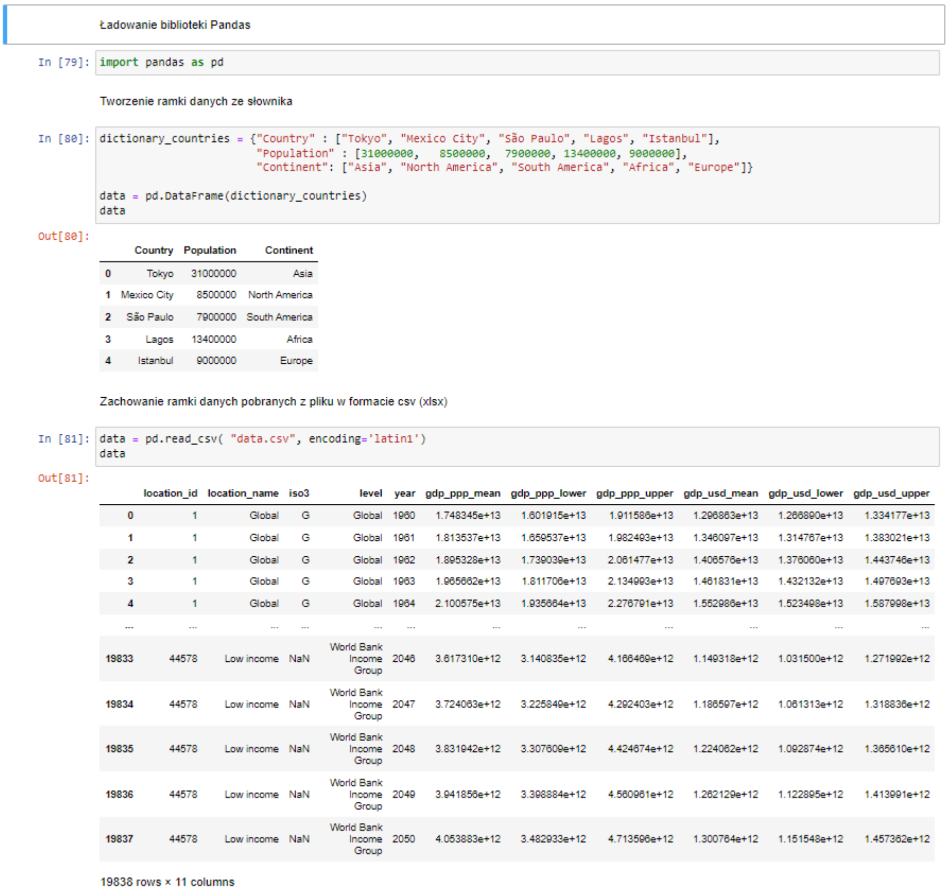
Link do repozytorium: https://github.com/jozek24/Eiwd

Józef Salik
Informatyka II stopień,
niestacjonarne,
3 semestr

1. Polecenie:

Dane do zadania będą pobrane ze strony http://ghdx.healthdata.org/ ihme_data która przedstawia sobą¸ repozytorium danych socjoekonomicznych. One zawieraja dane badań statystycznych w zakresie gospodarki, demografii oraz sluzby zdrowia. Dane mogą być przedstawione w postaci plików formatów .csv lub .xslx.

2. Wykonanie:



Ładowanie biblioteki Pandas

```
In [79]: import pandas as pd
```

Tworzenie ramki danych ze słownika

```
In [80]: dictionary_countries = {"Country" : ["Tokyo", "Mexico City", "São Paulo", "Lagos", "Istanbul"],
                                 "Population" : [31000000,   8500000,  7900000, 13400000, 9000000],
                                 "Continent": ["Asia", "North America", "South America", "Africa", "Europe"]}

         data = pd.DataFrame(dictionary_countries)
         data
```

Out[80]:

	Country	Population	Continent
0	Tokyo	31000000	Asia
1	Mexico City	8500000	North America
2	São Paulo	7900000	South America
3	Lagos	13400000	Africa
4	Istanbul	9000000	Europe

Zachowanie ramki danych pobranych z pliku w formacie csv (xlsx)

```
In [81]: data = pd.read_csv( "data.csv", encoding='latin1')
         data
```

Out[81]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
0	1	Global	G	Global	1960	1.748345e+13	1.601915e+13	1.911586e+13	1.296863e+13	1.266890e+13	1.334177e+13
1	1	Global	G	Global	1961	1.813537e+13	1.659537e+13	1.982493e+13	1.346097e+13	1.314767e+13	1.383021e+13
2	1	Global	G	Global	1962	1.895328e+13	1.739039e+13	2.061477e+13	1.406576e+13	1.376060e+13	1.443746e+13
3	1	Global	G	Global	1963	1.965662e+13	1.811706e+13	2.134993e+13	1.461831e+13	1.432132e+13	1.497693e+13
4	1	Global	G	Global	1964	2.100575e+13	1.935664e+13	2.276791e+13	1.552986e+13	1.523498e+13	1.587998e+13
...	...	...	...	...	...	...	...	...	...	...	...
19833	44578	Low income	NaN	World Bank Income Group	2046	3.617310e+12	3.140835e+12	4.166469e+12	1.149318e+12	1.031500e+12	1.271992e+12
19834	44578	Low income	NaN	World Bank Income Group	2047	3.724063e+12	3.225849e+12	4.292403e+12	1.186597e+12	1.061313e+12	1.318836e+12
19835	44578	Low income	NaN	World Bank Income Group	2048	3.831942e+12	3.307609e+12	4.424674e+12	1.224062e+12	1.092874e+12	1.365610e+12
19836	44578	Low income	NaN	World Bank Income Group	2049	3.941856e+12	3.398884e+12	4.560961e+12	1.262129e+12	1.122895e+12	1.413991e+12
19837	44578	Low income	NaN	World Bank Income Group	2050	4.053883e+12	3.482933e+12	4.713596e+12	1.300764e+12	1.151548e+12	1.457362e+12

19838 rows × 11 columns

Tworzenie ramki danych z listy list

In [82]:

```
lists = [["Tokyo", "Mexico City", "São Paulo", "Lagos", "Istanbul"],
[31000000,   8500000,  7900000, 13400000, 9000000]]

pd.DataFrame(lists)
```

Out[82]:

	0	1	2	3	4
0	Tokyo	Mexico City	São Paulo	Lagos	Istanbul
1	31000000	8500000	7900000	13400000	9000000

Transponowanie (wymieniamy kolumny a wierszy)

In [83]:

```
pd.DataFrame(data).T
```

Out[83]:

	0	1	2	3	4	5
location_id	1	1	1	1	1	1
location_name	Global	Global	Global	Global	Global	Global
iso3	G	G	G	G	G	G
level	Global	Global	Global	Global	Global	Global
year	1960	1961	1962	1963	1964	1965
gdp_ppp_mean	17483449774122.900391	18135370554950.5	18953278607513.5	19656620517295.898438	21005747228643.398438	22024586645615.199219
gdp_ppp_lower	16019146112388.800781	16595371585758.199219	17390391432341.599609	18117057797516.5	19356640986099.699219	20345846562097.898438
gdp_ppp_upper	19115862416823.5	19824927264221.5	20614772322197.601562	21349934484879.699219	22767910934166.101562	23822754092401.5
gdp_usd_mean	12968625317543.800781	13460972883451.599609	14065757980933.900391	14618310920876.400391	15529862054649.199219	16289721213918.900391
gdp_usd_lower	12668903338177.199219	13147665079303.800781	13760596066680.599609	14321321298044.400391	15234981973069.599609	15987267849238.599609
gdp_usd_upper	13341765801289.300781	13830213685062.900391	14437458446538.0	14976927145314.400391	15879980043956.900391	16633103517118.599609

11 rows × 19838 columns

Wyświetlić pierwsze 10 wierszy ramki danych

In [84]:

```
pd.DataFrame(data).T
data.head(10)
```

Out[84]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
0	1	Global	G	Global	1960	1.748345e+13	1.601915e+13	1.911586e+13	1.296863e+13	1.266890e+13	1.334177e+13
1	1	Global	G	Global	1961	1.813537e+13	1.659537e+13	1.982493e+13	1.346097e+13	1.314767e+13	1.383021e+13
2	1	Global	G	Global	1962	1.895328e+13	1.739039e+13	2.061477e+13	1.406576e+13	1.376060e+13	1.443746e+13
3	1	Global	G	Global	1963	1.965662e+13	1.811706e+13	2.134993e+13	1.461831e+13	1.432132e+13	1.497693e+13
4	1	Global	G	Global	1964	2.100575e+13	1.935664e+13	2.276791e+13	1.552986e+13	1.523498e+13	1.587998e+13

Wyświetlić ostatnie 10 wierszy ramki danych

In [85]:

```
data.tail(10)
```

Out[85]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
19828	44578	Low income	NaN	World Bank Income Group	2041	3.120963e+12	2.724077e+12	3.582807e+12	9.752426e+11	8.875033e+11	1.068693e+12
19829	44578	Low income	NaN	World Bank Income Group	2042	3.216988e+12	2.801335e+12	3.686394e+12	1.008813e+12	9.169149e+11	1.107239e+12
19830	44578	Low income	NaN	World Bank Income Group	2043	3.314031e+12	2.886768e+12	3.815672e+12	1.042881e+12	9.461940e+11	1.147550e+12
19831	44578	Low income	NaN	World Bank Income Group	2044	3.413020e+12	2.968361e+12	3.933135e+12	1.077714e+12	9.735487e+11	1.188093e+12
19832	44578	Low income	NaN	World Bank Income Group	2045	3.514244e+12	3.055623e+12	4.049325e+12	1.113207e+12	1.003241e+12	1.228145e+12
19833	44578	Low income	NaN	World Bank Income Group	2046	3.617310e+12	3.140835e+12	4.166469e+12	1.149318e+12	1.031500e+12	1.271992e+12
19834	44578	Low income	NaN	World Bank Income Group	2047	3.724063e+12	3.225849e+12	4.292403e+12	1.186597e+12	1.061313e+12	1.318836e+12
19835	44578	Low income	NaN	World Bank Income Group	2048	3.831942e+12	3.307609e+12	4.424674e+12	1.224062e+12	1.092874e+12	1.365610e+12
19836	44578	Low income	NaN	World Bank Income Group	2049	3.941856e+12	3.398884e+12	4.560961e+12	1.262129e+12	1.122895e+12	1.413991e+12
19837	44578	Low income	NaN	World Bank Income Group	2050	4.053883e+12	3.482933e+12	4.713596e+12	1.300764e+12	1.151548e+12	1.457362e+12

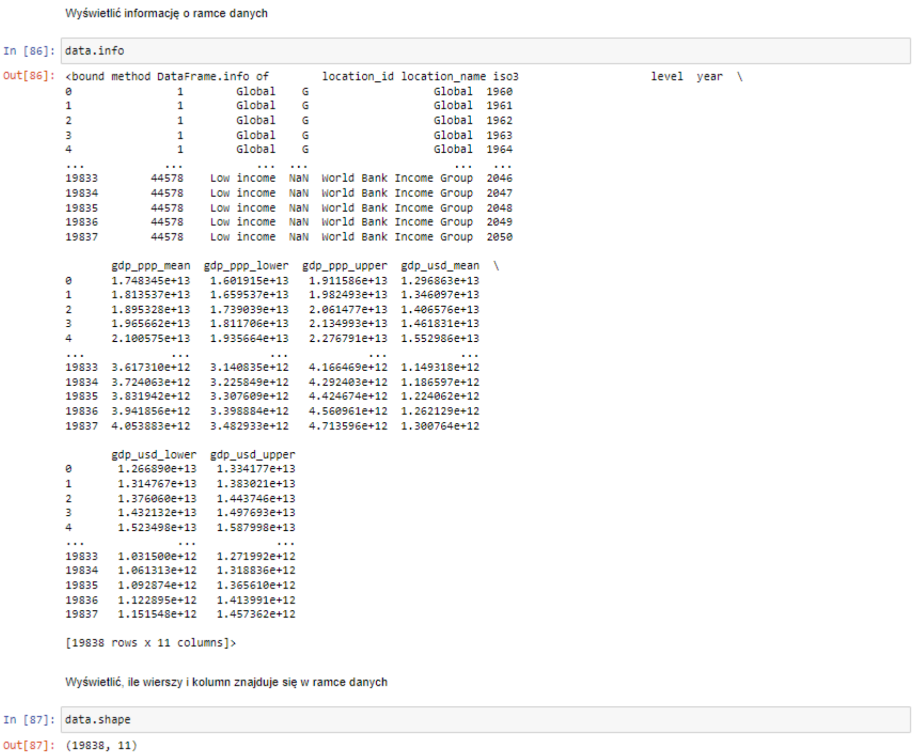
Wyświetlić informację o ramce danych

In [86]:

```
data.info
```

Out[86]:

```
<bound method DataFrame.info of        location_id location_name iso3                    level  year  \
0                1        Global    G                   Global  1960
1                1        Global    G                   Global  1961
2                1        Global    G                   Global  1962
3                1        Global    G                   Global  1963
4                1        Global    G                   Global  1964
...            ...           ...  ...                      ...   ...
19833        44578    Low income  NaN  World Bank Income Group  2046
19834        44578    Low income  NaN  World Bank Income Group  2047
19835        44578    Low income  NaN  World Bank Income Group  2048
19836        44578    Low income  NaN  World Bank Income Group  2049
19837        44578    Low income  NaN  World Bank Income Group  2050
```



Wyświetlić informację o ramce danych

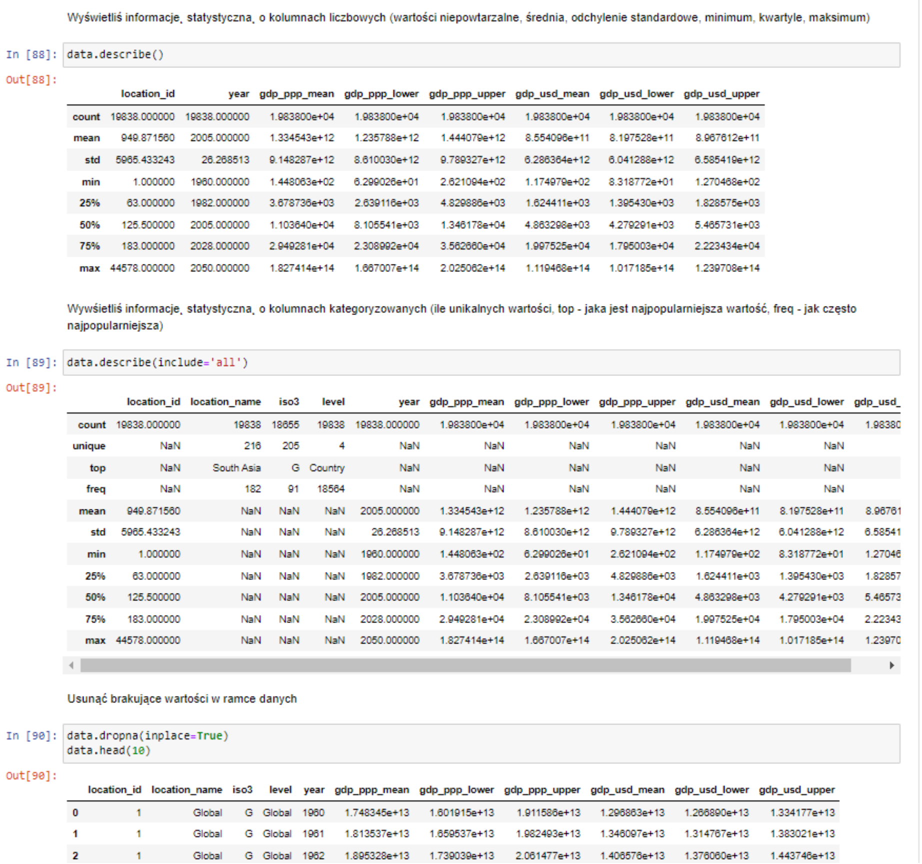
In [86]:
```
data.info
```

Out[86]:
```
<bound method DataFrame.info of        location_id location_name iso3                level  year  \
0                1        Global    G               Global  1960
1                1        Global    G               Global  1961
2                1        Global    G               Global  1962
3                1        Global    G               Global  1963
4                1        Global    G               Global  1964
...            ...           ...  ...                  ...   ...
19833        44578    Low income  NaN  World Bank Income Group  2046
19834        44578    Low income  NaN  World Bank Income Group  2047
19835        44578    Low income  NaN  World Bank Income Group  2048
19836        44578    Low income  NaN  World Bank Income Group  2049
19837        44578    Low income  NaN  World Bank Income Group  2050

       gdp_ppp_mean  gdp_ppp_lower  gdp_ppp_upper  gdp_usd_mean  \
0      1.748345e+13   1.601915e+13   1.911586e+13  1.296863e+13
1      1.813537e+13   1.659537e+13   1.982493e+13  1.346097e+13
2      1.895328e+13   1.739039e+13   2.061477e+13  1.406576e+13
3      1.965662e+13   1.811706e+13   2.134993e+13  1.461831e+13
4      2.100575e+13   1.935664e+13   2.276791e+13  1.552986e+13
...             ...            ...            ...           ...
19833  3.617310e+12   3.140835e+12   4.166469e+12  1.149318e+12
19834  3.724063e+12   3.225849e+12   4.292403e+12  1.186597e+12
19835  3.831942e+12   3.307609e+12   4.424674e+12  1.224062e+12
19836  3.941856e+12   3.398884e+12   4.560961e+12  1.262129e+12
19837  4.053883e+12   3.482933e+12   4.713596e+12  1.300764e+12

       gdp_usd_lower  gdp_usd_upper
0       1.266890e+13   1.334177e+13
1       1.314767e+13   1.383021e+13
2       1.376060e+13   1.443746e+13
3       1.432132e+13   1.497693e+13
4       1.523498e+13   1.587998e+13
...              ...            ...
19833   1.031500e+12   1.271992e+12
19834   1.061313e+12   1.318836e+12
19835   1.092874e+12   1.365610e+12
19836   1.122895e+12   1.413991e+12
19837   1.151548e+12   1.457362e+12

[19838 rows x 11 columns]>
```

Wyświetlić, ile wierszy i kolumn znajduje się w ramce danych

In [87]:
```
data.shape
```

Out[87]:
```
(19838, 11)
```



Wyświetliś informacje, statystyczna, o kolumnach liczbowych (wartości niepowtarzalne, średnia, odchylenie standardowe, minimum, kwartyle, maksimum)

In [88]:
```
data.describe()
```

Out[88]:

	location_id	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
count	19838.000000	19838.000000	1.983800e+04	1.983800e+04	1.983800e+04	1.983800e+04	1.983800e+04	1.983800e+04
mean	949.871560	2005.000000	1.334543e+12	1.235788e+12	1.444079e+12	8.554096e+11	8.197528e+11	8.967612e+11
std	5965.433243	26.268513	9.148287e+12	8.610030e+12	9.789327e+12	6.286364e+12	6.041288e+12	6.585419e+12
min	1.000000	1960.000000	1.448063e+02	6.299026e+01	2.621094e+02	1.174979e+02	8.318772e+01	1.270468e+02
25%	63.000000	1982.000000	3.678736e+03	2.639116e+03	4.829886e+03	1.624411e+03	1.395430e+03	1.828575e+03
50%	125.500000	2005.000000	1.103640e+04	8.105541e+03	1.346178e+04	4.863298e+03	4.279291e+03	5.465731e+03
75%	183.000000	2028.000000	2.949281e+04	2.308992e+04	3.562660e+04	1.997525e+04	1.795003e+04	2.223434e+04
max	44578.000000	2050.000000	1.827414e+14	1.667007e+14	2.025062e+14	1.119468e+14	1.017185e+14	1.239708e+14

Wywśietliś informacje, statystyczna, o kolumnach kategoryzowanych (ile unikalnych wartości, top - jaka jest najpopularniejsza wartość, freq - jak często najpopularniejsza)

In [89]:
```
data.describe(include='all')
```

Out[89]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_
count	19838.000000	19838	18655	19838	19838.000000	1.983800e+04	1.983800e+04	1.983800e+04	1.983800e+04	1.983800e+04	1.98380
unique	NaN	216	205	4	NaN	NaN	NaN	NaN	NaN	NaN	
top	NaN	South Asia	G	Country	NaN	NaN	NaN	NaN	NaN	NaN	
freq	NaN	182	91	18564	NaN	NaN	NaN	NaN	NaN	NaN	
mean	949.871560	NaN	NaN	NaN	2005.000000	1.334543e+12	1.235788e+12	1.444079e+12	8.554096e+11	8.197528e+11	8.96761
std	5965.433243	NaN	NaN	NaN	26.268513	9.148287e+12	8.610030e+12	9.789327e+12	6.286364e+12	6.041288e+12	6.58541
min	1.000000	NaN	NaN	NaN	1960.000000	1.448063e+02	6.299026e+01	2.621094e+02	1.174979e+02	8.318772e+01	1.27046
25%	63.000000	NaN	NaN	NaN	1982.000000	3.678736e+03	2.639116e+03	4.829886e+03	1.624411e+03	1.395430e+03	1.82857
50%	125.500000	NaN	NaN	NaN	2005.000000	1.103640e+04	8.105541e+03	1.346178e+04	4.863298e+03	4.279291e+03	5.46573
75%	183.000000	NaN	NaN	NaN	2028.000000	2.949281e+04	2.308992e+04	3.562660e+04	1.997525e+04	1.795003e+04	2.22343
max	44578.000000	NaN	NaN	NaN	2050.000000	1.827414e+14	1.667007e+14	2.025062e+14	1.119468e+14	1.017185e+14	1.23970

Usunąć brakujące wartości w ramce danych

In [90]:
```
data.dropna(inplace=True)
data.head(10)
```

Out[90]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
0	1	Global	G	Global	1960	1.748345e+13	1.601915e+13	1.911586e+13	1.296863e+13	1.266890e+13	1.334177e+13
1	1	Global	G	Global	1961	1.813537e+13	1.659537e+13	1.982493e+13	1.346097e+13	1.314767e+13	1.383021e+13
2	1	Global	G	Global	1962	1.895328e+13	1.739039e+13	2.061477e+13	1.406576e+13	1.376060e+13	1.443746e+13

Usunąć brakujące wartości w ramce danych

In [90]:

```
data.dropna(inplace=True)
data.head(10)
```

Out[90]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
0	1	Global	G	Global	1960	1.748345e+13	1.601915e+13	1.911586e+13	1.296863e+13	1.266890e+13	1.334177e+13
1	1	Global	G	Global	1961	1.813537e+13	1.659537e+13	1.982493e+13	1.346097e+13	1.314767e+13	1.383021e+13
2	1	Global	G	Global	1962	1.895328e+13	1.739039e+13	2.061477e+13	1.406576e+13	1.376060e+13	1.443746e+13
3	1	Global	G	Global	1963	1.965662e+13	1.811706e+13	2.134993e+13	1.461831e+13	1.432132e+13	1.497693e+13
4	1	Global	G	Global	1964	2.100575e+13	1.935664e+13	2.276791e+13	1.552986e+13	1.523498e+13	1.587998e+13
5	1	Global	G	Global	1965	2.202459e+13	2.034585e+13	2.382275e+13	1.628972e+13	1.598727e+13	1.663310e+13
6	1	Global	G	Global	1966	2.306193e+13	2.136085e+13	2.489782e+13	1.708885e+13	1.678223e+13	1.742396e+13
7	1	Global	G	Global	1967	2.391268e+13	2.217842e+13	2.577837e+13	1.770884e+13	1.740660e+13	1.804193e+13
8	1	Global	G	Global	1968	2.516723e+13	2.340479e+13	2.698215e+13	1.865379e+13	1.833216e+13	1.898399e+13
9	1	Global	G	Global	1969	2.642403e+13	2.464521e+13	2.831984e+13	1.955395e+13	1.921164e+13	1.987990e+13

Przedstawić wybór wierszy i kolumny używając nazw oraz indeksów na różne sposoby

In [91]:

```
data["location_name"]
```

Out[91]:

```
0         Global
1         Global
2         Global
3         Global
4         Global
          ...
19469      Sudan
19470      Sudan
19471      Sudan
19472      Sudan
19473      Sudan
Name: location_name, Length: 18655, dtype: object
```

In [92]:

```
data.location_name
```

Out[92]:

```
0         Global
1         Global
2         Global
3         Global
4         Global
          ...
19469      Sudan
19470      Sudan
19471      Sudan
19472      Sudan
19473      Sudan
Name: location_name, Length: 18655, dtype: object
```

In [93]:

```
data[["location_id", "location_name", "year"]]
```

Out[93]:

	location_id	location_name	year
0	1	Global	1960

In [94]:

```
data.loc[1:3,"location_id":"year"]
```

Out[94]:

	location_id	location_name	iso3	level	year
1	1	Global	G	Global	1961
2	1	Global	G	Global	1962
3	1	Global	G	Global	1963

In [95]:

```
data.loc[:, "location_id":"year"]
```

Out[95]:

	location_id	location_name	iso3	level	year
0	1	Global	G	Global	1960
1	1	Global	G	Global	1961
2	1	Global	G	Global	1962
3	1	Global	G	Global	1963
4	1	Global	G	Global	1964
...	...	...	...	...	...
19469	522	Sudan	SDN	Country	2046
19470	522	Sudan	SDN	Country	2047
19471	522	Sudan	SDN	Country	2048
19472	522	Sudan	SDN	Country	2049
19473	522	Sudan	SDN	Country	2050

18655 rows × 5 columns

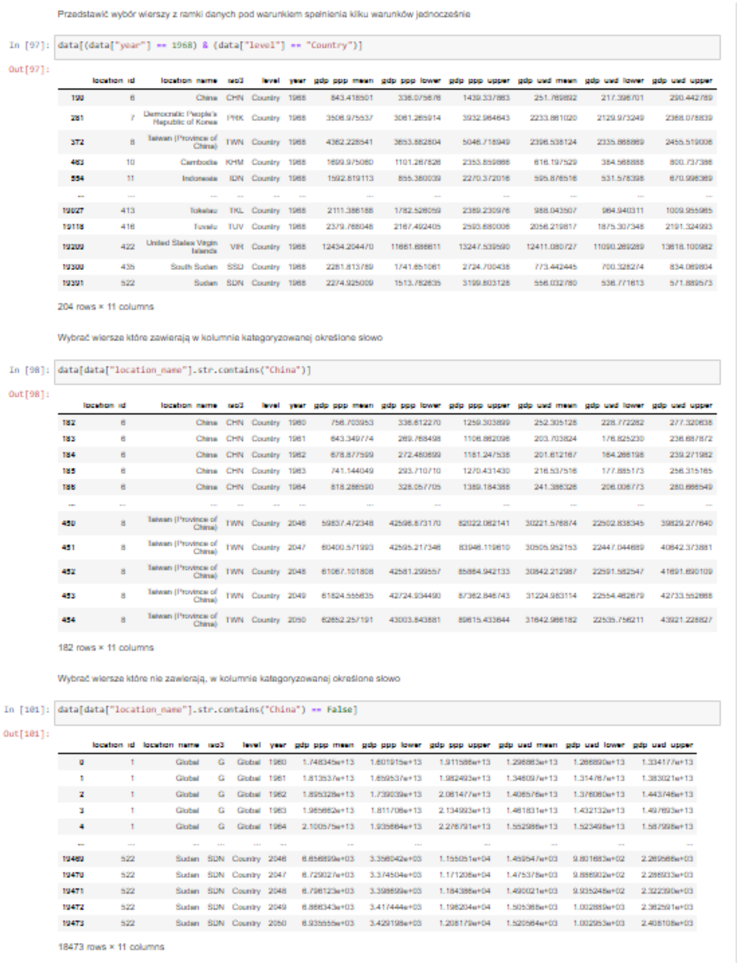
Przedstawić wybór wierszy z ramki danych pod warunkiem odnośnie określonej wartości kolumny

In [96]:

```
data[data["year"] == 1968]
```

Out[96]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
8	1	Global	G	Global	1968	2.516723e+13	2.340479e+13	2.698215e+13	1.865379e+13	1.833216e+13	1.898399e+13
190	6	China	CHN	Country	1968	8.434185e+02	3.360757e+02	1.439338e+03	2.517699e+02	2.173967e+02	2.904428e+02
281	7	Democratic People's Republic of Korea	PRK	Country	1968	3.506976e+03	3.061266e+03	3.932965e+03	2.233861e+03	2.129973e+03	2.368079e+03
372	8	Taiwan (Province of China)	TWN	Country	1968	4.362229e+03	3.653883e+03	5.046719e+03	2.396538e+03	2.335869e+03	2.455519e+03
463	10	Cambodia	KHM	Country	1968	1.699975e+03	1.101268e+03	2.353860e+03	6.161975e+02	3.845689e+02	8.007374e+02
...	...	...	...	...	...	...	...	...	...	...	...
19027	413	Tokelau	TKL	Country	1968	2.111386e+03	1.782526e+03	2.389231e+03	9.880435e+02	9.649403e+02	1.009956e+03
19118	416	Tuvalu	TUV	Country	1968	2.379768e+03	2.167492e+03	2.593680e+03	2.056220e+03	1.875307e+03	2.191325e+03
19209	422	United States Virgin Islands	VIR	Country	1968	1.243420e+04	1.166169e+04	1.324754e+04	1.241108e+04	1.109027e+04	1.361810e+04
19300	435	South Sudan	SSD	Country	1968	2.281814e+03	1.741651e+03	2.724700e+03	7.734424e+02	7.003283e+02	8.340698e+02
19391	522	Sudan	SDN	Country	1968	2.274925e+03	1.513783e+03	3.199803e+03	5.560328e+02	5.367716e+02	5.718896e+02



Przedstawić wybór wierszy z ramki danych pod warunkiem spełnienia kilku warunków jednocześnie

```
In [97]: data[(data["year"] == 1968) & (data["level"] == "Country")]
```

Out[97]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
190	6	China	CHN	Country	1968	843.418501	336.075676	1439.337883	251.769892	217.398701	290.442789
281	7	Democratic People's Republic of Korea	PRK	Country	1968	3506.975537	3061.265914	3932.984643	2233.881020	2129.973249	2368.078839
372	8	Taiwan (Province of China)	TWN	Country	1968	4362.228541	3853.882804	5046.718949	2398.538124	2335.888889	2455.519006
463	10	Cambodia	KHM	Country	1968	1699.975060	1101.267826	2353.859886	616.197529	384.568888	800.737386
554	11	Indonesia	IDN	Country	1968	1592.819113	855.380039	2270.372016	595.876516	531.578398	670.998389
...	...	...	...	...	...	...	...	...	...	...	...
19027	413	Tokelau	TKL	Country	1968	2111.388188	1782.528059	2389.230976	988.043507	984.940311	1009.955985
19118	416	Tuvalu	TUV	Country	1968	2379.768048	2167.492405	2593.680006	2056.219817	1875.307348	2191.324993
19209	422	United States Virgin Islands	VIR	Country	1968	12434.204470	11661.686811	13247.539590	12411.080727	11090.289289	13818.100982
19300	435	South Sudan	SSD	Country	1968	2281.813789	1741.651061	2724.700438	773.442445	700.328274	834.089804
19391	522	Sudan	SDN	Country	1968	2274.925009	1513.782635	3199.803128	556.032780	538.771613	571.889573

204 rows × 11 columns

Wybrać wiersze które zawierają w kolumnie kategoryzowanej określone słowo

```
In [98]: data[data["location_name"].str.contains("China")]
```

Out[98]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
182	6	China	CHN	Country	1980	756.703953	336.612270	1259.303899	252.305128	228.772282	277.320638
183	6	China	CHN	Country	1981	643.349774	289.768498	1106.882098	203.703824	178.825230	238.687872
184	6	China	CHN	Country	1982	678.877599	272.480699	1181.247538	201.612167	164.286198	239.271982
185	6	China	CHN	Country	1983	741.144049	293.710710	1270.431430	216.537516	177.885173	256.315165
186	6	China	CHN	Country	1984	818.288590	328.057705	1389.184388	241.388326	206.008773	280.888549
...	...	...	...	...	...	...	...	...	...	...	...
450	8	Taiwan (Province of China)	TWN	Country	2046	59837.472348	42596.873170	82022.082141	30221.576874	22502.838345	39829.277640
451	8	Taiwan (Province of China)	TWN	Country	2047	60400.571993	42595.217348	83948.119810	30505.952153	22447.044689	40842.373881
452	8	Taiwan (Province of China)	TWN	Country	2048	61067.101808	42581.299557	85864.942133	30842.212987	22591.582547	41691.690109
453	8	Taiwan (Province of China)	TWN	Country	2049	61824.555835	42724.934490	87382.848743	31224.983114	22554.482879	42733.552888
454	8	Taiwan (Province of China)	TWN	Country	2050	62652.257191	43003.843881	89615.433844	31642.988182	22535.756211	43921.228827

182 rows × 11 columns

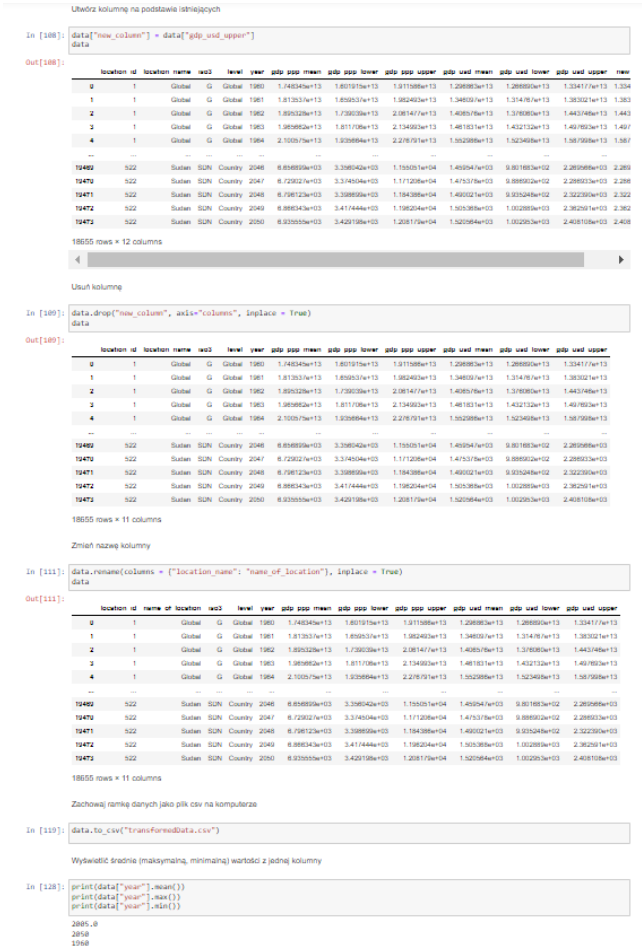
Wybrać wiersze które nie zawierają, w kolumnie kategoryzowanej określone słowo

```
In [101]: data[data["location_name"].str.contains("China") == False]
```

Out[101]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
0	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298883e+13	1.268890e+13	1.334177e+13
1	1	Global	G	Global	1981	1.813537e+13	1.659537e+13	1.982493e+13	1.348097e+13	1.314767e+13	1.383021e+13
2	1	Global	G	Global	1982	1.895328e+13	1.739039e+13	2.061477e+13	1.408578e+13	1.376080e+13	1.443748e+13
3	1	Global	G	Global	1983	1.965662e+13	1.811708e+13	2.134993e+13	1.461831e+13	1.432132e+13	1.497693e+13
4	1	Global	G	Global	1984	2.100575e+13	1.935664e+13	2.276791e+13	1.552986e+13	1.523498e+13	1.587998e+13
...	...	...	...	...	...	...	...	...	...	...	...
19469	522	Sudan	SDN	Country	2046	6.656699e+03	3.356042e+03	1.155051e+04	1.459547e+03	9.801683e+02	2.289566e+03
19470	522	Sudan	SDN	Country	2047	6.729027e+03	3.374504e+03	1.171206e+04	1.475378e+03	9.886902e+02	2.298933e+03
19471	522	Sudan	SDN	Country	2048	6.798123e+03	3.398699e+03	1.184386e+04	1.490021e+03	9.935248e+02	2.322390e+03
19472	522	Sudan	SDN	Country	2049	6.868343e+03	3.417444e+03	1.198204e+04	1.505388e+03	1.002889e+03	2.382591e+03
19473	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520564e+03	1.002953e+03	2.408108e+03

18473 rows × 11 columns



Utwórz kolumnę na podstawie istniejących

```
In [108]: data["new_column"] = data["gdp_usd_upper"]
          data
```

Out[108]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper	new
0	1	Global	G	Global	1980	1.748345e+13	1.801915e+13	1.911586e+13	1.298883e+13	1.286890e+13	1.334177e+13	1.334
1	1	Global	G	Global	1981	1.813537e+13	1.859537e+13	1.982493e+13	1.348097e+13	1.314787e+13	1.383021e+13	1.383
2	1	Global	G	Global	1982	1.895328e+13	1.739039e+13	2.081477e+13	1.408578e+13	1.378080e+13	1.443748e+13	1.443
3	1	Global	G	Global	1983	1.985882e+13	1.811708e+13	2.134993e+13	1.481831e+13	1.432132e+13	1.497893e+13	1.497
4	1	Global	G	Global	1984	2.100575e+13	1.935884e+13	2.276791e+13	1.552986e+13	1.523498e+13	1.587998e+13	1.587
...	...	...	...	...	...	...	...	...	...	...	...	
19469	522	Sudan	SDN	Country	2046	6.858899e+03	3.358042e+03	1.155051e+04	1.459547e+03	9.801883e+02	2.289588e+03	2.289
19470	522	Sudan	SDN	Country	2047	6.729027e+03	3.374504e+03	1.171208e+04	1.475378e+03	9.888902e+02	2.288933e+03	2.288
19471	522	Sudan	SDN	Country	2048	6.798123e+03	3.398899e+03	1.184388e+04	1.490021e+03	9.935248e+02	2.322390e+03	2.322
19472	522	Sudan	SDN	Country	2049	6.888343e+03	3.417444e+03	1.198204e+04	1.505388e+03	1.002889e+03	2.382591e+03	2.382
19473	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002953e+03	2.408108e+03	2.408

18655 rows × 12 columns

Usuń kolumnę

```
In [109]: data.drop("new_column", axis="columns", inplace = True)
          data
```

Out[109]:

	location_id	location_name	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
0	1	Global	G	Global	1980	1.748345e+13	1.801915e+13	1.911586e+13	1.298883e+13	1.286890e+13	1.334177e+13
1	1	Global	G	Global	1981	1.813537e+13	1.859537e+13	1.982493e+13	1.348097e+13	1.314787e+13	1.383021e+13
2	1	Global	G	Global	1982	1.895328e+13	1.739039e+13	2.081477e+13	1.408578e+13	1.378080e+13	1.443748e+13
3	1	Global	G	Global	1983	1.985882e+13	1.811708e+13	2.134993e+13	1.481831e+13	1.432132e+13	1.497893e+13
4	1	Global	G	Global	1984	2.100575e+13	1.935884e+13	2.276791e+13	1.552986e+13	1.523498e+13	1.587998e+13
...	...	...	...	...	...	...	...	...	...	...	...
19469	522	Sudan	SDN	Country	2046	6.858899e+03	3.358042e+03	1.155051e+04	1.459547e+03	9.801883e+02	2.289588e+03
19470	522	Sudan	SDN	Country	2047	6.729027e+03	3.374504e+03	1.171208e+04	1.475378e+03	9.888902e+02	2.288933e+03
19471	522	Sudan	SDN	Country	2048	6.798123e+03	3.398899e+03	1.184388e+04	1.490021e+03	9.935248e+02	2.322390e+03
19472	522	Sudan	SDN	Country	2049	6.888343e+03	3.417444e+03	1.198204e+04	1.505388e+03	1.002889e+03	2.382591e+03
19473	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002953e+03	2.408108e+03

18655 rows × 11 columns

Zmień nazwę kolumny

```
In [111]: data.rename(columns = {"location_name": "name_of_location"}, inplace = True)
          data
```

Out[111]:

	location_id	name_of_location	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
0	1	Global	G	Global	1980	1.748345e+13	1.801915e+13	1.911586e+13	1.298883e+13	1.286890e+13	1.334177e+13
1	1	Global	G	Global	1981	1.813537e+13	1.859537e+13	1.982493e+13	1.348097e+13	1.314787e+13	1.383021e+13
2	1	Global	G	Global	1982	1.895328e+13	1.739039e+13	2.081477e+13	1.408578e+13	1.378080e+13	1.443748e+13
3	1	Global	G	Global	1983	1.985882e+13	1.811708e+13	2.134993e+13	1.481831e+13	1.432132e+13	1.497893e+13
4	1	Global	G	Global	1984	2.100575e+13	1.935884e+13	2.276791e+13	1.552986e+13	1.523498e+13	1.587998e+13
...	...	...	...	...	...	...	...	...	...	...	...
19469	522	Sudan	SDN	Country	2046	6.858899e+03	3.358042e+03	1.155051e+04	1.459547e+03	9.801883e+02	2.289588e+03
19470	522	Sudan	SDN	Country	2047	6.729027e+03	3.374504e+03	1.171208e+04	1.475378e+03	9.888902e+02	2.288933e+03
19471	522	Sudan	SDN	Country	2048	6.798123e+03	3.398899e+03	1.184388e+04	1.490021e+03	9.935248e+02	2.322390e+03
19472	522	Sudan	SDN	Country	2049	6.888343e+03	3.417444e+03	1.198204e+04	1.505388e+03	1.002889e+03	2.382591e+03
19473	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002953e+03	2.408108e+03

18655 rows × 11 columns

Zachowaj ramkę danych jako plik csv na komputerze

```
In [119]: data.to_csv("transformedData.csv")
```

Wyświetlić średnie (maksymalną, minimalną) wartości z jednej kolumny

```
In [128]: print(data["year"].mean())
          print(data["year"].max())
          print(data["year"].min())

2005.0
2050
1960
```

Wyświetlić liczbę wierszy

```
In [130]: len(data)

Out[130]: 18655
```

Wyświetlić wartości unikatowe w kolumnie

```
In [132]: data['name_of_location'].unique()

Out[132]: array(['Global', 'China', "Democratic People's Republic of Korea",
               'Taiwan (Province of China)', 'Cambodia', 'Indonesia',
               "Lao People's Democratic Republic", 'Malaysia', 'Maldives',
               'Myanmar', 'Philippines', 'Sri Lanka', 'Thailand', 'Timor-Leste',
               'Viet Nam', 'Fiji', 'Kiribati', 'Marshall Islands',
               'Micronesia (Federated States of)', 'Papua New Guinea', 'Samoa',
               'Solomon Islands', 'Tonga', 'Vanuatu', 'Armenia', 'Azerbaijan',
               'Georgia', 'Kazakhstan', 'Kyrgyzstan', 'Mongolia', 'Tajikistan',
               'Turkmenistan', 'Uzbekistan', 'Albania', 'Bosnia and Herzegovina',
               'Bulgaria', 'Croatia', 'Czechia', 'Hungary', 'North Macedonia',
               'Montenegro', 'Poland', 'Romania', 'Serbia', 'Slovakia',
               'Slovenia', 'Belarus', 'Estonia', 'Latvia', 'Lithuania',
               'Republic of Moldova', 'Russian Federation', 'Ukraine',
               'Brunei Darussalam', 'Japan', 'Republic of Korea', 'Singapore',
               'Australia', 'New Zealand', 'Andorra', 'Austria', 'Belgium',
               'Cyprus', 'Denmark', 'Finland', 'France', 'Germany', 'Greece',
               'Iceland', 'Ireland', 'Israel', 'Italy', 'Luxembourg', 'Malta',
               'Netherlands', 'Norway', 'Portugal', 'Spain', 'Sweden',
               'Switzerland', 'United Kingdom', 'Argentina', 'Chile', 'Uruguay',
               'Canada', 'United States of America', 'Antigua and Barbuda',
               'Bahamas', 'Barbados', 'Belize', 'Cuba', 'Dominica',
               'Dominican Republic', 'Grenada', 'Guyana', 'Haiti', 'Jamaica',
               'Saint Lucia', 'Saint Vincent and the Grenadines', 'Suriname',
               'Trinidad and Tobago', 'Bolivia (Plurinational State of)',
               'Ecuador', 'Peru', 'Colombia', 'Costa Rica', 'El Salvador',
               'Guatemala', 'Honduras', 'Mexico', 'Nicaragua', 'Panama',
               'Venezuela (Bolivarian Republic of)', 'Brazil', 'Paraguay',
               'Algeria', 'Bahrain', 'Egypt', 'Iran (Islamic Republic of)',
               'Iraq', 'Jordan', 'Kuwait', 'Lebanon', 'Libya', 'Morocco',
               'Palestine', 'Oman', 'Qatar', 'Saudi Arabia',
               'Syrian Arab Republic', 'Tunisia', 'Turkey',
               'United Arab Emirates', 'Yemen', 'Afghanistan', 'Bangladesh',
               'Bhutan', 'India', 'Nepal', 'Pakistan', 'Angola',
               'Central African Republic', 'Congo',
               'Democratic Republic of the Congo', 'Equatorial Guinea', 'Gabon',
               'Burundi', 'Comoros', 'Djibouti', 'Eritrea', 'Ethiopia', 'Kenya',
               'Madagascar', 'Malawi', 'Mauritius', 'Mozambique', 'Rwanda',
               'Seychelles', 'Somalia', 'United Republic of Tanzania', 'Uganda',
               'Zambia', 'Botswana', 'Lesotho', 'Namibia', 'South Africa',
               'Eswatini', 'Zimbabwe', 'Benin', 'Burkina Faso', 'Cameroon',
               'Cabo Verde', 'Chad', "CÃ´te d'Ivoire", 'Gambia', 'Ghana',
               'Guinea', 'Guinea-Bissau', 'Liberia', 'Mali', 'Mauritania',
               'Niger', 'Nigeria', 'Sao Tome and Principe', 'Senegal',
               'Sierra Leone', 'Togo', 'American Samoa', 'Bermuda',
               'Cook Islands', 'Greenland', 'Guam', 'Monaco', 'Nauru', 'Niue',
               'Northern Mariana Islands', 'Palau', 'Puerto Rico',
               'Saint Kitts and Nevis', 'San Marino', 'Tokelau', 'Tuvalu',
               'United States Virgin Islands', 'South Sudan', 'Sudan'],
              dtype=object)
```

Wyświetlić liczby rekordów odpowiadających do wartości

```
In [134]: data['year'].value_counts()

Out[134]: 1960    205
          2020    205
          2026    205
          2025    205
          2024    205
                 ...
          1980    205
          1987    205
          1986    205
          1985    205
          2050    205
          Name: year, Length: 91, dtype: int64
```

Sortowanie wierszy ramki danych według wartości określonej kolumny (malejąco, rosnąco)

```
In [136]: data.sort_values(['year'], ascending = True)

Out[136]:
```

	location id	name of location	iso3	level	year	gdp ppp mean	gdp ppp lower	gdp ppp upper	gdp usd mean	gdp usd lower	gdp usd upper
0	1	Global	G	Global	1960	1.748345e+13	1.801915e+13	1.911588e+13	1.298983e+13	1.268890e+13	1.334177e+13
15561	193	Botswana	BWA	Country	1960	9.717015e+02	3.788083e+02	1.749355e+03	4.495982e+02	3.505335e+02	6.193884e+02
4095	53	Serbia	SRB	Country	1960	6.817093e+03	5.081034e+03	8.569332e+03	3.012123e+03	2.684823e+03	3.294584e+03
16380	203	Cabo Verde	CPV	Country	1960	1.297658e+03	1.033272e+03	1.775715e+03	6.479192e+02	4.383848e+02	8.844288e+02
13195	163	India	IND	Country	1960	1.130893e+03	9.932036e+02	1.277588e+03	3.246846e+02	3.009429e+02	3.493888e+02
...	...	...	...	...	...	...	...	...	...	...	...
16379	202	Cameroon	CMR	Country	2050	5.920282e+03	4.822332e+03	7.497971e+03	2.434588e+03	1.908823e+03	3.037938e+03
12102	150	Oman	OMN	Country	2050	2.498143e+04	1.442400e+04	3.989478e+04	1.299342e+04	9.151710e+03	1.805737e+04

```
In [138]: data.sort_values(['year'], ascending = False)
```

Out[138]:

	location_id	name_of_location	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
19473	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002953e+03	2.408108e+03
909	14	Maldives	MDV	Country	2050	2.793245e+04	1.815198e+04	4.623992e+04	1.489534e+04	9.979481e+03	2.180008e+04
18563	374	Niue	NIU	Country	2050	1.602765e+04	1.085071e+04	2.221278e+04	2.497005e+04	1.787741e+04	3.412818e+04
1000	15	Myanmar	MMR	Country	2050	9.542253e+03	6.358301e+03	1.419022e+04	2.355690e+03	1.607945e+03	3.333577e+03
14013	172	Equatorial Guinea	GNQ	Country	2050	3.959889e+04	2.046880e+04	7.092959e+04	1.501183e+04	9.538289e+03	2.290789e+04
...	...	...	...	...	...	...	...	...	...	...	...
14924	184	Mozambique	MOZ	Country	1980	9.553399e+02	2.598885e+02	2.900925e+03	1.544818e+02	1.471128e+02	1.611889e+02
4823	62	Russian Federation	RUS	Country	1980	1.414160e+04	6.885333e+03	2.348298e+04	6.972232e+03	6.358317e+03	7.758518e+03
14833	183	Mauritius	MUS	Country	1980	4.073388e+03	3.202992e+03	5.181822e+03	1.457139e+03	1.313790e+03	1.587527e+03
4914	63	Ukraine	UKR	Country	1980	1.117381e+04	7.776325e+03	1.697983e+04	3.584988e+03	1.958490e+03	5.210702e+03
0	1	Global	G	Global	1980	1.748345e+13	1.801915e+13	1.911588e+13	1.298883e+13	1.288890e+13	1.334177e+13

18655 rows × 11 columns

Wyświetlić wierszy dla 10 największych (najmniejszych) wartości określonej kolumny

```
In [143]: data.nlargest(10,'location_id')
```

Out[143]:

	location_id	name_of_location	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
19383	522	Sudan	SDN	Country	1980	2547.179302	1644.073039	3828.841689	633.882757	601.384519	665.294016
19384	522	Sudan	SDN	Country	1981	2482.585119	1612.927210	3533.338963	617.083884	585.488008	645.948818
19385	522	Sudan	SDN	Country	1982	2574.844128	1695.153232	3827.890114	639.502180	607.460748	669.311805
19386	522	Sudan	SDN	Country	1983	2441.718832	1607.123912	3483.482837	605.459753	576.710823	630.304819
19387	522	Sudan	SDN	Country	1984	2355.892315	1568.218099	3351.024821	582.982548	558.338878	605.232759
19388	522	Sudan	SDN	Country	1985	2442.227514	1640.723484	3447.380812	603.387933	578.818879	625.398808
19389	522	Sudan	SDN	Country	1986	2332.188841	1584.792089	3291.410283	571.208739	549.375485	587.721988
19390	522	Sudan	SDN	Country	1987	2244.125320	1479.767928	3198.282514	555.178834	531.231822	574.328948
19391	522	Sudan	SDN	Country	1988	2274.925009	1513.782835	3199.803128	558.032780	538.771813	571.889573
19392	522	Sudan	SDN	Country	1989	2301.478529	1548.580837	3228.924550	554.887210	539.859275	574.814931

```
In [144]: data.nsmallest(10, 'location_id')
```

Out[144]:

	location_id	name_of_location	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
0	1	Global	G	Global	1980	1.748345e+13	1.801915e+13	1.911588e+13	1.298883e+13	1.288890e+13	1.334177e+13
1	1	Global	G	Global	1981	1.813537e+13	1.659537e+13	1.982493e+13	1.348097e+13	1.314787e+13	1.383021e+13
2	1	Global	G	Global	1982	1.895328e+13	1.739039e+13	2.081477e+13	1.408578e+13	1.378080e+13	1.443748e+13
3	1	Global	G	Global	1983	1.985882e+13	1.811708e+13	2.134993e+13	1.461831e+13	1.432132e+13	1.497893e+13
4	1	Global	G	Global	1984	2.100575e+13	1.935884e+13	2.278791e+13	1.552988e+13	1.523498e+13	1.587998e+13
5	1	Global	G	Global	1985	2.202459e+13	2.034585e+13	2.382275e+13	1.628972e+13	1.598727e+13	1.663310e+13
6	1	Global	G	Global	1986	2.308193e+13	2.138085e+13	2.489782e+13	1.708885e+13	1.678223e+13	1.742398e+13
7	1	Global	G	Global	1987	2.391288e+13	2.217842e+13	2.577837e+13	1.770884e+13	1.740880e+13	1.804193e+13
8	1	Global	G	Global	1988	2.518723e+13	2.340479e+13	2.898215e+13	1.865379e+13	1.833218e+13	1.898399e+13
9	1	Global	G	Global	1989	2.642403e+13	2.484521e+13	2.831984e+13	1.955395e+13	1.921184e+13	1.987990e+13

Wyświetlić wierszy dla 10 największych wartości określonej kolumny pod warunkiem określonych wartości innej kolumny

```
In [148]: data[(data['location_id'].isin([1,522])) & (data['year'] == 1985)].nlargest(10,'location_id')
```

Out[148]:

	location_id	name_of_location	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
19388	522	Sudan	SDN	Country	1985	2.442228e+03	1.640723e+03	3.447381e+03	6.033879e+02	5.788187e+02	6.253988e+02
5	1	Global	G	Global	1985	2.202459e+13	2.034585e+13	2.382275e+13	1.628972e+13	1.598727e+13	1.663310e+13

Grupowanie wierszy według wartości kolumny kategoryzowanej, potem uśrednienie wartości wszystkich kolumn w grupie - MultiIndex

```
In [155]: data.groupby('year').agg('mean')
```

Out[155]:

	location_id	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
year							
1980	135.839024	8.528513e+10	7.814218e+10	9.324812e+10	6.328159e+10	6.179953e+10	6.508179e+10
1981	135.839024	8.848523e+10	8.095304e+10	9.670897e+10	6.568329e+10	6.413498e+10	6.748448e+10
1982	135.839024	9.245503e+10	8.483118e+10	1.005599e+11	6.881348e+10	6.712488e+10	7.042883e+10
1983	135.839024	9.588598e+10	8.837590e+10	1.041480e+11	7.130884e+10	6.986011e+10	7.305819e+10
1984	135.839024	1.024871e+11	9.442264e+10	1.110830e+11	7.575543e+10	7.431699e+10	7.748333e+10
...	...	...	...	...	...	...	...
2046	135.839024	8.583220e+11	7.915827e+11	9.409579e+11	5.275873e+11	4.882889e+11	5.759145e+11

Grupowanie wierszy według wartości kolumny kategoryzowanej, potem uśrednienie wartości dla pewnych kolumn, liczba wartości i mediana dla pozostałych kolumn w grupach

In [168]:
```
grouped_data = data.groupby('name_of_location').agg({'year': ['max'],
                           'gdp_ppp_mean': ['mean', 'median']})
grouped_data
```

Out[168]:

	year	gdp_ppp_mean	
	max	mean	median
name_of_location			
Afghanistan	2050	1941.160286	2119.759138
Albania	2050	9092.515182	8081.130761
Algeria	2050	8820.271149	10103.908020
American Samoa	2050	15340.385197	13820.538346
Andorra	2050	25139.562251	27089.148048
...	...	...	...
Venezuela (Bolivarian Republic of)	2050	10594.142490	11671.502834
Viet Nam	2050	5737.873814	4273.495870
Yemen	2050	2837.237249	2888.837485
Zambia	2050	3107.029470	3138.130808
Zimbabwe	2050	2925.918098	2825.947945

205 rows × 3 columns

Wyświetlić nazwy kolumn indeksu złożonego

In [163]:
```
grouped_data.columns
```

Out[163]:
```
MultiIndex([(        'year',    'max'),
            ('gdp_ppp_mean',   'mean'),
            ('gdp_ppp_mean', 'median')],
           )
```

Sortować kolumnę indeksu złożonego

In [166]:
```
grouped_data['gdp_ppp_mean']['mean'].sort_values(ascending = False)
```

Out[166]:
```
name_of_location
Global                      9.203498e+13
Monaco                      1.053996e+05
United Arab Emirates        8.433651e+04
Luxembourg                  7.652696e+04
Greenland                   7.523594e+04
                                ...
Central African Republic    1.229873e+03
Niger                       1.182638e+03
Malawi                      9.931087e+02
Burundi                     9.816384e+02
Somalia                     2.185428e+02
Name: mean, Length: 205, dtype: float64
```

Stworzyć tabelę przystawną (pivot table) na podstawie ramki danych

In [185]:
```
data_pivot = data.pivot_table(values='gdp_ppp_mean', index='name_of_location', columns='year', aggfunc='count',
                     margins=False, dropna=True, fill_value=None) # tabela podsumowująca
data_pivot
```

Out[185]:

year	1960	1961	1962	1963	1964	1965	1966	1967	1968	1969	...	2041	2042	2043	2044	2045	2046	2047	2048	2049	2050
name_of_location																					
Afghanistan	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Albania	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Algeria	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
American Samoa	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Andorra	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Venezuela (Bolivarian Republic of)	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Viet Nam	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Yemen	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Zambia	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Zimbabwe	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1

205 rows × 91 columns

Wyświetlić indeksy i kolumny tabeli przystawnej

In [186]:
```
data_pivot.index
```

Wyświetlić indeksy i kolumny tabeli przystawnej

In [186]:
```
data_pivot.index
```

Out[186]:
```
Index(['Afghanistan', 'Albania', 'Algeria', 'American Samoa', 'Andorra',
       'Angola', 'Antigua and Barbuda', 'Argentina', 'Armenia', 'Australia',
       ...
       'United States Virgin Islands', 'United States of America', 'Uruguay',
       'Uzbekistan', 'Vanuatu', 'Venezuela (Bolivarian Republic of)',
       'Viet Nam', 'Yemen', 'Zambia', 'Zimbabwe'],
      dtype='object', name='name_of_location', length=205)
```

Utwórz indeks złożony tabeli przystawnej i wyświetl go

In [187]:
```
data_pivot.columns
```

Out[187]:
```
Int64Index([1960, 1961, 1962, 1963, 1964, 1965, 1966, 1967, 1968, 1969, 1970,
            1971, 1972, 1973, 1974, 1975, 1976, 1977, 1978, 1979, 1980, 1981,
            1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989, 1990, 1991, 1992,
            1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003,
            2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014,
            2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025,
            2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036,
            2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047,
            2048, 2049, 2050],
           dtype='int64', name='year')
```

Zaimportuj moduł pyplot z biblioteki matplotlib

In [188]:
```
import matplotlib.pyplot as plt
```

Wskazać, że wykresy należy rysować bezpośrednio w zeszycie, a nie w osobnej zakładce

In [189]:
```
%matplotlib inline
```

In [190]:
```
data
```

Out[190]:

	location id	name of location	iso3	level	year	gdp ppp mean	gdp ppp lower	gdp ppp upper	gdp usd mean	gdp usd lower	gdp usd upper
0	1	Global	G	Global	1960	1.748345e+13	1.601915e+13	1.911588e+13	1.296863e+13	1.266890e+13	1.334177e+13
1	1	Global	G	Global	1961	1.813537e+13	1.659537e+13	1.982493e+13	1.348097e+13	1.314787e+13	1.383021e+13
2	1	Global	G	Global	1962	1.895328e+13	1.739039e+13	2.081477e+13	1.408578e+13	1.378080e+13	1.443748e+13
3	1	Global	G	Global	1963	1.965882e+13	1.811708e+13	2.134993e+13	1.461831e+13	1.432132e+13	1.497693e+13
4	1	Global	G	Global	1964	2.100575e+13	1.935884e+13	2.276791e+13	1.552986e+13	1.523498e+13	1.587998e+13
...	...	...	...	...	...	...	...	...	...	...	...
19469	522	Sudan	SDN	Country	2046	6.656899e+03	3.358042e+03	1.155051e+04	1.459547e+03	9.801683e+02	2.269566e+03
19470	522	Sudan	SDN	Country	2047	6.729027e+03	3.374504e+03	1.171208e+04	1.475378e+03	9.886902e+02	2.286933e+03
19471	522	Sudan	SDN	Country	2048	6.798123e+03	3.398899e+03	1.184388e+04	1.490021e+03	9.935248e+02	2.322390e+03
19472	522	Sudan	SDN	Country	2049	6.868343e+03	3.417444e+03	1.198204e+04	1.505388e+03	1.002889e+03	2.382591e+03
19473	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002963e+03	2.408108e+03

18655 rows × 11 columns

In [191]:
```
data_pivot
```

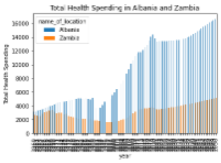
Out[191]:

year	**1960**	**1961**	**1962**	**1963**	**1964**	**1965**	**1966**	**1967**	**1968**	**1969**	**...**	**2041**	**2042**	**2043**	**2044**	**2045**	**2046**	**2047**	**2048**	**2049**	**2050**
name of location																					
Afghanistan	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Albania	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Algeria	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
American Samoa	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Andorra	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Venezuela (Bolivarian Republic of)	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Viet Nam	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Yemen	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Zambia	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
Zimbabwe	1	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1

205 rows × 91 columns

Narysować histogram na podstawie wartości kolumny

```
In [195]: df_bar = data[(data['name_of_location'].isin(['Albania','Zambia']))].pivot_table(values='gdp_ppp_mean',
                           index='year', columns='name_of_location', aggfunc='mean',
                                   fill_value=None, margins=False, dropna=True)
          df_bar.plot(kind = 'bar')
          plt.ylabel('Total Health Spending')
          plt.title('Total Health Spending in Albania and Zambia')

Out[195]: Text(0.5, 1.0, 'Total Health Spending in Albania and Zambia')
```



Przedstawić sposoby łączenia ramek danych za pomocą metod merge i concat

```
In [200]: data2 = pd.read_csv('transformedData.csv', encoding='latin1')
          data2

Out[200]:
```

	Unnamed: 0	location id	name of location	iso3	level	year	gdp ppp mean	gdp ppp lower	gdp ppp upper	gdp usd mean	gdp usd lower	gdp us
0	0	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298863e+13	1.268890e+13	1.334
1	1	1	Global	G	Global	1981	1.813537e+13	1.659537e+13	1.982493e+13	1.348097e+13	1.314767e+13	1.383
2	2	1	Global	G	Global	1982	1.895328e+13	1.739039e+13	2.061477e+13	1.408578e+13	1.376060e+13	1.443
3	3	1	Global	G	Global	1983	1.985882e+13	1.811706e+13	2.134993e+13	1.461831e+13	1.432132e+13	1.497
4	4	1	Global	G	Global	1984	2.100575e+13	1.935884e+13	2.276791e+13	1.552988e+13	1.523498e+13	1.587
...	...	...	...	...	...	...	...	...	...	...	...	
18650	19469	522	Sudan	SDN	Country	2046	6.656899e+03	3.356042e+03	1.155051e+04	1.459547e+03	9.801683e+02	2.269
18651	19470	522	Sudan	SDN	Country	2047	6.729027e+03	3.374504e+03	1.171208e+04	1.475378e+03	9.888902e+02	2.288
18652	19471	522	Sudan	SDN	Country	2048	6.798123e+03	3.398899e+03	1.184388e+04	1.490021e+03	9.935248e+02	2.322
18653	19472	522	Sudan	SDN	Country	2049	6.888343e+03	3.417444e+03	1.198204e+04	1.505388e+03	1.002889e+03	2.362
18654	19473	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002953e+03	2.408

18655 rows × 12 columns

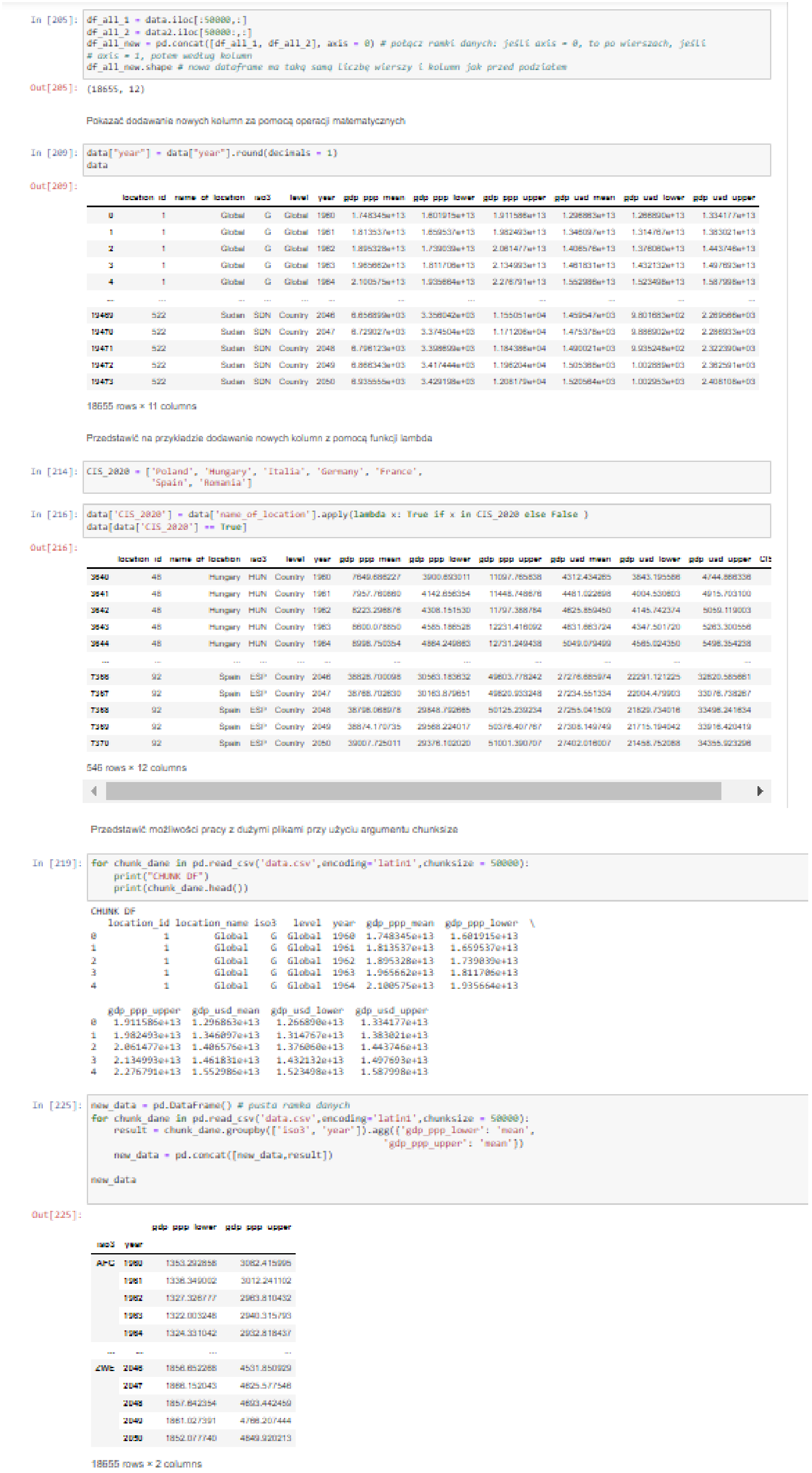
```
In [201]: pd.merge (data, data2, on = ['iso3'], how = 'inner') # bierze tylko wiersze, które pasują do obu ramek danych

Out[201]:
```

	location id x	name of location x	iso3	level x	year x	gdp ppp mean x	gdp ppp lower x	gdp ppp upper x	gdp usd mean x	gdp usd lower x
0	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298863e+13	1.268890e+13
1	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298863e+13	1.268890e+13
2	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298863e+13	1.268890e+13
3	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298863e+13	1.268890e+13
4	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298863e+13	1.268890e+13
...	...	...	...	...	...	...	...	...	...	...
1697600	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002953e+03
1697601	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002953e+03
1697602	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002953e+03
1697603	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002953e+03
1697604	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002953e+03

1697605 rows × 22 columns

```
In [203]: pd.merge (data, data2, on = ['iso3'], how = 'outer') # wszystkie wiersze ze wszystkich ramek danych, nie ma znaczenia, czy pasuj

Out[203]:
```

	location id x	name of location x	iso3	level x	year x	gdp ppp mean x	gdp ppp lower x	gdp ppp upper x	gdp usd mean x	gdp usd lower x
0	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298863e+13	1.268890e+13
1	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298863e+13	1.268890e+13
2	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298863e+13	1.268890e+13
3	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298863e+13	1.268890e+13
4	1	Global	G	Global	1980	1.748345e+13	1.601915e+13	1.911588e+13	1.298863e+13	1.268890e+13
...	...	...	...	...	...	...	...	...	...	...
1697600	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520584e+03	1.002953e+03



In [205]:

```
df_all_1 = data.iloc[:50000,:]
df_all_2 = data2.iloc[50000:,:]
df_all_new = pd.concat([df_all_1, df_all_2], axis = 0) # połącz ramki danych: jeśli axis = 0, to po wierszach, jeśli
# axis = 1, potem według kolumn
df_all_new.shape # nowa dataframe ma taką samą liczbę wierszy i kolumn jak przed podziałem
```

Out[205]:

```
(18655, 12)
```

Pokazać dodawanie nowych kolumn za pomocą operacji matematycznych

In [209]:

```
data["year"] = data["year"].round(decimals = 1)
data
```

Out[209]:

	location_id	name_of_location	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper
0	1	Global	G	Global	1960	1.748345e+13	1.601915e+13	1.911586e+13	1.296863e+13	1.266890e+13	1.334177e+13
1	1	Global	G	Global	1961	1.813537e+13	1.659537e+13	1.982493e+13	1.346097e+13	1.314767e+13	1.383021e+13
2	1	Global	G	Global	1962	1.895328e+13	1.739039e+13	2.061477e+13	1.406576e+13	1.376060e+13	1.443746e+13
3	1	Global	G	Global	1963	1.965662e+13	1.811706e+13	2.134993e+13	1.461831e+13	1.432132e+13	1.497693e+13
4	1	Global	G	Global	1964	2.100575e+13	1.935664e+13	2.276791e+13	1.552986e+13	1.523498e+13	1.587998e+13
...	...	...	...	...	...	...	...	...	...	...	...
19469	522	Sudan	SDN	Country	2046	6.656899e+03	3.356042e+03	1.155051e+04	1.459547e+03	9.801683e+02	2.289566e+03
19470	522	Sudan	SDN	Country	2047	6.729027e+03	3.374504e+03	1.171206e+04	1.475378e+03	9.886902e+02	2.298933e+03
19471	522	Sudan	SDN	Country	2048	6.798123e+03	3.398899e+03	1.184386e+04	1.490021e+03	9.935248e+02	2.322390e+03
19472	522	Sudan	SDN	Country	2049	6.866343e+03	3.417444e+03	1.196204e+04	1.505388e+03	1.002889e+03	2.362591e+03
19473	522	Sudan	SDN	Country	2050	6.935555e+03	3.429198e+03	1.208179e+04	1.520564e+03	1.002953e+03	2.408108e+03

18655 rows × 11 columns

Przedstawić na przykladzie dodawanie nowych kolumn z pomocą funkcji lambda

In [214]:

```
CIS_2020 = ['Poland', 'Hungary', 'Italia', 'Germany', 'France',
            'Spain', 'Romania']
```

In [216]:

```
data['CIS_2020'] = data['name_of_location'].apply(lambda x: True if x in CIS_2020 else False )
data[data['CIS_2020'] == True]
```

Out[216]:

	location_id	name_of_location	iso3	level	year	gdp_ppp_mean	gdp_ppp_lower	gdp_ppp_upper	gdp_usd_mean	gdp_usd_lower	gdp_usd_upper	CIS
3640	48	Hungary	HUN	Country	1980	7649.686227	3900.893011	11097.765838	4312.434265	3843.195586	4744.886338	
3641	48	Hungary	HUN	Country	1981	7957.760880	4142.856354	11446.748676	4481.022698	4004.530603	4915.703100	
3642	48	Hungary	HUN	Country	1982	8223.298878	4308.151530	11797.388784	4625.859450	4145.742374	5059.119003	
3643	48	Hungary	HUN	Country	1983	8600.078850	4585.186526	12231.416092	4831.663724	4347.501720	5263.300556	
3644	48	Hungary	HUN	Country	1984	8998.750354	4884.249883	12731.249438	5049.079499	4565.024350	5498.354238	
...	...	...	...	...	...	...	...	...	...	...	...	
7366	92	Spain	ESP	Country	2046	38828.700098	30563.183832	49803.778242	27276.685974	22291.121225	32820.585861	
7367	92	Spain	ESP	Country	2047	38768.702830	30183.879851	49820.933248	27234.551334	22004.479903	33076.738287	
7368	92	Spain	ESP	Country	2048	38798.068978	29848.792885	50125.239234	27255.041509	21829.734016	33498.241834	
7369	92	Spain	ESP	Country	2049	38874.170735	29568.224017	50376.407787	27308.149749	21715.194042	33916.420419	
7370	92	Spain	ESP	Country	2050	39007.725011	29376.102020	51001.390707	27402.018007	21458.752088	34355.923298	

546 rows × 12 columns

Przedstawić możliwości pracy z dużymi plikami przy użyciu argumentu chunksize

In [219]:

```
for chunk_dane in pd.read_csv('data.csv',encoding='latin1',chunksize = 50000):
    print("CHUNK DF")
    print(chunk_dane.head())
```

```
CHUNK DF
   location_id location_name iso3   level  year  gdp_ppp_mean  gdp_ppp_lower  \
0            1        Global    G  Global  1960  1.748345e+13   1.601915e+13
1            1        Global    G  Global  1961  1.813537e+13   1.659537e+13
2            1        Global    G  Global  1962  1.895328e+13   1.739039e+13
3            1        Global    G  Global  1963  1.965662e+13   1.811706e+13
4            1        Global    G  Global  1964  2.100575e+13   1.935664e+13

   gdp_ppp_upper  gdp_usd_mean  gdp_usd_lower  gdp_usd_upper
0   1.911586e+13  1.296863e+13   1.266890e+13   1.334177e+13
1   1.982493e+13  1.346097e+13   1.314767e+13   1.383021e+13
2   2.061477e+13  1.406576e+13   1.376060e+13   1.443746e+13
3   2.134993e+13  1.461831e+13   1.432132e+13   1.497693e+13
4   2.276791e+13  1.552986e+13   1.523498e+13   1.587998e+13
```

In [225]:

```
new_data = pd.DataFrame() # pusta ramka danych
for chunk_dane in pd.read_csv('data.csv',encoding='latin1',chunksize = 50000):
    result = chunk_dane.groupby(['iso3', 'year']).agg({'gdp_ppp_lower': 'mean',
                                                        'gdp_ppp_upper': 'mean'})
    new_data = pd.concat([new_data,result])

new_data
```

Out[225]:

		gdp_ppp_lower	gdp_ppp_upper
iso3	**year**		
AFG	**1980**	1353.292858	3082.415996
	1981	1338.349002	3012.241102
	1982	1327.328777	2983.810432
	1983	1322.003248	2940.315793
	1984	1324.331042	2932.818437
...	...	...	...
ZWE	**2046**	1856.852268	4531.850929
	2047	1888.152043	4625.577546
	2048	1857.842354	4693.442469
	2049	1861.027391	4768.207444
	2050	1852.077740	4849.920213

18655 rows × 2 columns

3. Wnioski:

- Biblioteka pandas umożliwia analizę i manipulację danymi w Pythonie
- Główna struktura danych to DataFrame
- DataFrame przechowuje dane w formie tabelarycznej
- Pandas zapewnia szereg funkcji do modyfikacji i manipulacji tą strukturą