

Sheet

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

import warnings; warnings.filterwarnings("ignore")
```

```
path = 'GDP by Country 1999-2022.csv'
gdp = pd.read_csv(path, sep=',', header=0, thousands=",")

row, col = gdp.shape
print(f"This dataset has {row} rows and {col} columns")
```

This dataset has 180 rows and 24 columns

```
gdp.set_index("Country", inplace=True)
gdp = gdp.T
```

```
gdp.head(10)
```

Country	Afghanistan, Rep. of.	Albania	Algeria	Angola	Antigua and Barbuda	Argentina	Armenia	Australia	Austria	Azerbaijan	...	United Kingdom	United States	Uruguay	Uzbekistan	Vanuatu	Venezuela	Vietnam	Y F c
1999	0.000	3.444	48.845	6.153	0.652	283.523	1.845	401.782	213.390	4.581	...	1467.03	9268.43	20.913	17.041	0.251	97.977	28.684	7
2000	0.000	3.695	54.749	9.135	0.678	284.204	1.912	390.017	194.407	5.273	...	1445.19	9816.98	20.086	13.717	0.245	117.153	31.196	9
2001	0.000	4.096	55.181	8.936	0.710	268.697	2.118	368.191	193.345	5.708	...	1435.63	10127.95	18.561	11.632	0.235	122.872	32.504	9
2002	4.084	4.456	57.053	11.386	0.718	97.732	2.376	412.901	208.566	6.236	...	1574.47	10469.60	12.089	9.657	0.230	92.889	35.148	9
2003	4.585	5.600	68.013	13.956	0.754	127.643	2.807	527.588	255.842	7.276	...	1814.64	10960.75	11.211	10.129	0.280	83.442	39.630	1
2004	5.971	7.452	85.016	19.800	0.818	151.958	3.573	639.356	293.194	8.682	...	2155.16	11712.48	13.268	12.001	0.330	112.800	45.548	1
2005	7.309	8.376	102.380	30.632	0.875	181.549	4.903	712.436	305.338	12.561	...	2230.61	12455.83	16.878	13.670	0.368	143.443	53.053	1
2006	8.399	9.133	114.322	43.759	0.962	212.702	6.410	754.816	321.934	19.817	...	2373.69	13244.55	19.221	16.088	0.387	181.608	60.995	1
2007	9.892	10.163	116.158	55.370	1.026	247.096	7.666	822.089	348.659	28.699	...	2660.66	13770.31	20.725	18.840	0.421	219.372	68.298	2
2008	11.513	11.131	126.889	67.608	1.074	277.912	8.579	854.072	364.580	40.229	...	2822.66	14418.48	22.211	21.727	0.451	231.959	76.414	2

10 rows × 180 columns

EDA

Look at China's growth

```
# Let's look at a particular country China
plt.figure(figsize=(15,8))
sns.set_style('whitegrid')

ax = sns.lineplot(data = gdp[['China']], x=gdp.index, y='China')
plt.title('GDP of China from 1999 to 2022')
plt.ylabel('Gross Domestic Product (billions US dollars)')

plt.show()
```



Look at top 5 countries

```
# let's see who are the top 5 richest countries
gdp22 = gdp.iloc[-1,:].T

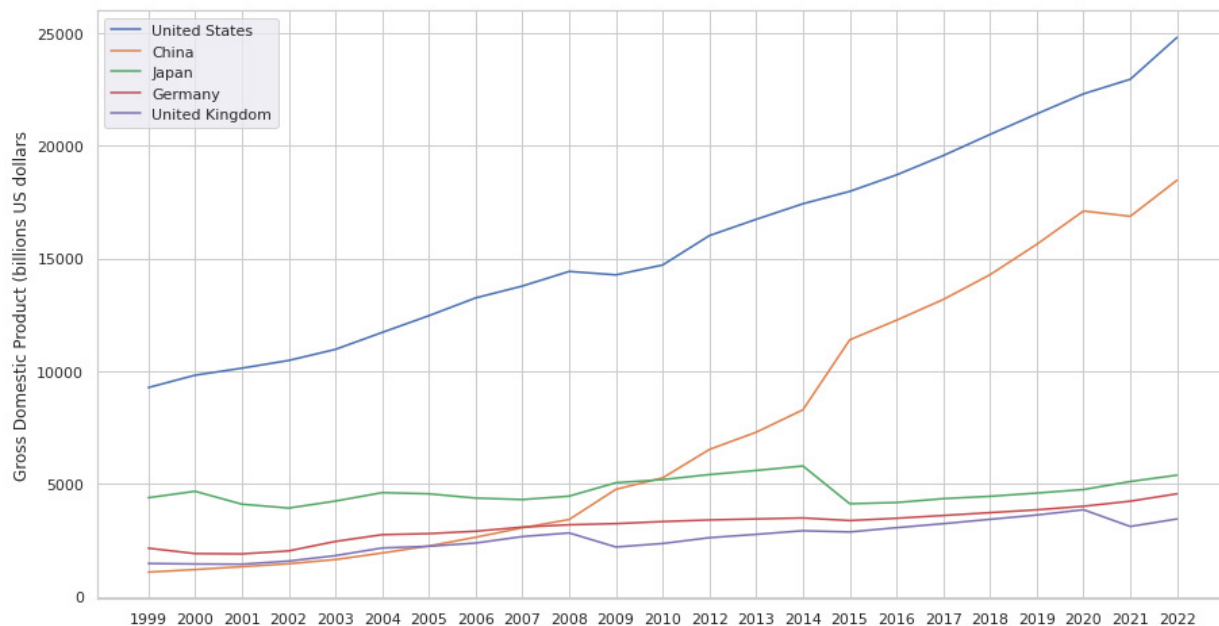
top5 = gdp22.sort_values('2022', ascending=False).head().index
top5 = list(top5)

top5

['United States', 'China', 'Japan', 'Germany', 'United Kingdom']
```

```
fig, ax = plt.subplots(figsize=(15,8))
sns.set(font_scale = 1)
# plot top 5 countries
for i in top5:
    sns.lineplot(data = gdp, x=gdp.index, y=i)

plt.legend(['United States', 'China', 'Japan', 'Germany', 'United Kingdom'])
plt.ylabel('Gross Domestic Product (billions US dollars)')
plt.show()
```



Let's try map visualization

```
# Merge with ISO convention
path = 'https://gist.githubusercontent.com/radcliff/f09c0f88344a7fcef373/raw/2753c482ad091c54b1822288ad2e4811c021d8ec/wikipedia-iso-co
iso = pd.read_csv(path, sep=',')
iso.rename(columns={"English short name lower case": "Country",
                    "Alpha-3 code": "iso_alpha"},
           inplace=True)
iso.set_index("Country", inplace=True)
gdp = gdp.T
gdp = gdp.merge(iso, on="Country", how="left")
iso.head()
```

	Alpha-2 code	iso_alpha	Numeric code	ISO 3166-2
Country				
Afghanistan	AF	AFG	4	ISO 3166-2:AF
Åland Islands	AX	ALA	248	ISO 3166-2:AX
Albania	AL	ALB	8	ISO 3166-2:AL
Algeria	DZ	DZA	12	ISO 3166-2:DZ
American Samoa	AS	ASM	16	ISO 3166-2:AS

Geographical map

```
import plotly.express as px

year = "2022"

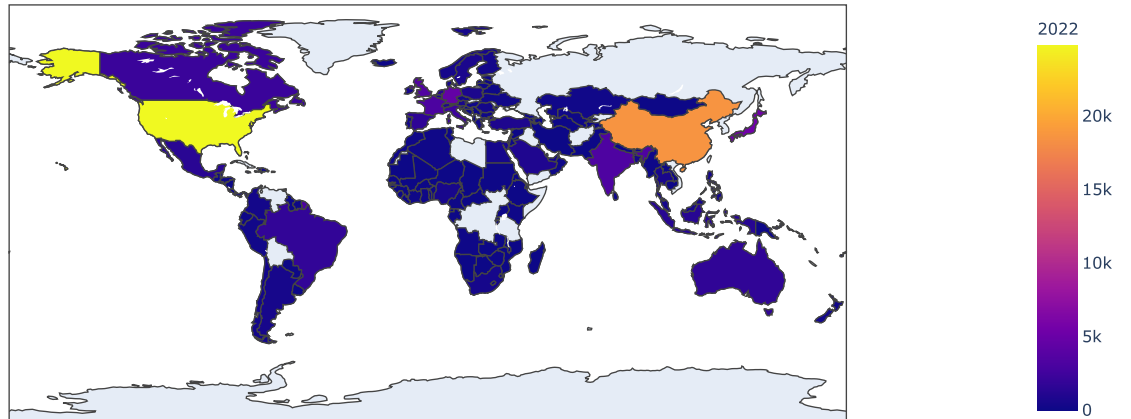
ds = gdp[[year, "iso_alpha"]]
ds.reset_index(inplace=True)

fig = px.choropleth(ds, locations="iso_alpha",
                    color=year,
                    hover_name="Country",
                    color_continuous_scale=px.colors.sequential.Plasma)

fig.update_layout(
    title_text = f'Gross Domestic Product {year}',
    title_pad={"l":700}
)
```

```
fig.show()
```

Gross Domestic Product 2022



Logarithmic scale

```
# plot it in logarithmic scale for better visualization.
ds["log"] = ds[year].apply(lambda x: np.log(x))

fig = px.choropleth(ds, locations="iso_alpha",
                    color="log",
                    hover_name="Country",
                    color_continuous_scale=px.colors.sequential.Plasma)

fig.update_layout(
    title_text = f'Gross Domestic Product {year}',
    title_pad={"l":700}
)

fig.show()
```

Gross Domestic Product 2022

