**Is GDP(Gross Domestic Product) of a country a direct contributer to the climate change?

**I have used two data sources for builing my pipeline:

source1: GDP Data of countries(World Bank IBRD-IDA) url: https://data.worldbank.org/indicator/NY.GDP.MKTP.CD?end=2022&start=2021&view=chart (https://data.worldbank.org/indicator/NY.GDP.MKTP.CD?end=2022&start=2021&view=chart) licence:CC-BY 4.0 https://datacatalog.worldbank.org/public-licenses#cc-by (https://datacatalog.worldbank.org/public-licenses#cc-by)

source2: Surface temperature data of countries(GlobalDataLab) url:https://globaldatalab.org/geos/download/surfacetempyear/ (https://globaldatalab.org/geos/download/surfacetempyear/)

Firstly we make the necessary imports

In [1]:

```
import pandas as pd
import sqlite3
```

we read the CSV file using Pandas as pandas has numerous features to clean and transform our data.

In [2]:

```
df1 = pd.read_csv(
    r"C:\Users\YUGAL\Downloads\API_NY.GDP.MKTP.CD_DS2_en_csv_v2_584499\API_NY.GDP.MKTP.CD_DS2_en_csv_v2_584499.csv",
    sep=',',
    skiprows=4,
    error_bad_lines=False,
    low_memory=False
)
```

In [3]:

```
df1.head() # To get a rough picture of the data we are dealing with
```

Out[3]:

	Country Name	Country Code	Indicator Name	Indicator Code	1960	1961	1962	1963	1964	1965	...	2015
0	Aruba	ABW	GDP (current US$)	NY.GDP.MKTP.CD	NaN	NaN	NaN	NaN	NaN	NaN	...	2.962907e+09
1	Africa Eastern and Southern	AFE	GDP (current US$)	NY.GDP.MKTP.CD	1.847810e+10	1.936631e+10	2.050647e+10	2.224273e+10	2.429433e+10	2.661956e+10	...	9.325135e+11
2	Afghanistan	AFG	GDP (current US$)	NY.GDP.MKTP.CD	5.377778e+08	5.488889e+08	5.466667e+08	7.511112e+08	8.000000e+08	1.006667e+09	...	1.913422e+10
3	Africa Western and Central	AFW	GDP (current US$)	NY.GDP.MKTP.CD	1.041165e+10	1.113592e+10	1.195171e+10	1.268581e+10	1.384900e+10	1.487476e+10	...	7.692632e+11
4	Angola	AGO	GDP (current US$)	NY.GDP.MKTP.CD	NaN	NaN	NaN	NaN	NaN	NaN	...	9.049642e+10

5 rows × 69 columns

**Currently let us target at the data from the years (2002-2022) and we only need the "Country Name" column and get rid of the other columns.

In [4]:

```
year_range = range(2002,2023)
years = [str(i) for i in year_range]
cat_cols = ['Country Name']
cols_to_keep = cat_cols + years
gdp_data = df1[cols_to_keep]
```

In [6]:
```
print(set(gdp_data['Country Name'].unique()))
```

```
{'Africa Eastern and Southern', 'Monaco', 'Fragile and conflict affected situations', 'Central Europe and the Baltics', 'Suda
n', 'Cayman Islands', 'El Salvador', 'South Africa', 'Slovenia', 'Lithuania', 'Zimbabwe', 'Brunei Darussalam', 'French Polyne
sia', 'Gabon', "Cote d'Ivoire", 'Indonesia', 'Least developed countries: UN classification', 'Euro area', 'Brazil', 'Armeni
a', 'Vanuatu', 'Latin America & Caribbean (excluding high income)', 'Botswana', 'Ecuador', 'Djibouti', 'Early-demographic div
idend', 'Gibraltar', 'Serbia', 'Lao PDR', 'North America', 'San Marino', 'Fiji', 'Nepal', 'Central African Republic', 'Hondur
as', 'IDA only', 'Gambia, The', 'Switzerland', 'Guinea', 'Middle East & North Africa (excluding high income)', 'Paraguay', 'C
roatia', 'West Bank and Gaza', 'Europe & Central Asia (IDA & IBRD countries)', 'Zambia', 'Angola', 'Chile', 'Nauru', 'Arab Wo
rld', 'Guyana', 'Ghana', 'East Asia & Pacific (excluding high income)', 'Liechtenstein', 'Mozambique', 'East Asia & Pacific
(IDA & IBRD countries)', 'Uganda', 'Spain', 'Bulgaria', 'Singapore', 'Madagascar', 'Panama', 'Peru', 'St. Martin (French par
t)', 'Tajikistan', 'Belize', 'Curacao', 'Mali', 'Pacific island small states', 'Bermuda', 'Solomon Islands', 'Bahrain', 'Not
classified', 'Turkmenistan', 'New Caledonia', 'St. Kitts and Nevis', 'Latin America & the Caribbean (IDA & IBRD countries)',
'Ukraine', 'Namibia', 'St. Lucia', 'Puerto Rico', 'Greenland', 'Upper middle income', 'Papua New Guinea', 'Late-demographic d
ividend', 'Macao SAR, China', 'IDA blend', 'Estonia', 'Argentina', 'Tanzania', 'Sierra Leone', 'Uruguay', 'Lower middle incom
e', 'Liberia', 'Austria', 'Maldives', 'Luxembourg', 'Congo, Dem. Rep.', 'Burkina Faso', 'Small states', 'Algeria', 'Tuvalu',
'Finland', 'Myanmar', 'Portugal', 'France', 'IBRD only', 'Somalia', 'East Asia & Pacific', 'Cuba', 'Norway', 'Senegal', 'Faro
e Islands', 'Slovak Republic', 'Isle of Man', 'Sub-Saharan Africa (IDA & IBRD countries)', 'Egypt, Arab Rep.', 'Marshall Isla
nds', 'Europe & Central Asia (excluding high income)', 'European Union', 'Haiti', 'North Macedonia', 'World', 'Low income',
'Guinea-Bissau', 'New Zealand', 'Lebanon', 'Grenada', 'Venezuela, RB', 'Eswatini', 'Guatemala', 'Samoa', 'Canada', 'Channel I
slands', 'Germany', 'Middle income', 'Israel', 'Cyprus', 'Heavily indebted poor countries (HIPC)', 'Saudi Arabia', 'High inco
me', 'Kenya', 'Latin America & Caribbean', 'Timor-Leste', 'Kosovo', 'Mexico', 'Australia', 'OECD members', 'Poland', 'Nigeri
a', 'Sri Lanka', 'Bhutan', 'Kiribati', 'Netherlands', 'Sweden', 'South Asia (IDA & IBRD)', 'Turkiye', 'IDA total', 'United St
ates', 'Cameroon', 'Suriname', 'Hong Kong SAR, China', 'Japan', 'Costa Rica', 'Burundi', 'Bosnia and Herzegovina', 'Georgia',
'Guam', 'Denmark', 'Colombia', 'India', 'Morocco', 'Lesotho', 'Middle East & North Africa', 'Andorra', 'Belgium', 'Caribbean
small states', 'Greece', 'United Kingdom', 'Nicaragua', 'Dominica', 'St. Vincent and the Grenadines', 'Sub-Saharan Africa',
'Trinidad and Tobago', 'Uzbekistan', 'Bangladesh', 'Post-demographic dividend', 'South Asia', 'Antigua and Barbuda', 'Microne
sia, Fed. Sts.', 'Niger', 'Comoros', 'Cabo Verde', 'Mongolia', 'Ireland', 'Congo, Rep.', 'Romania', 'Iraq', 'Sub-Saharan Afri
ca (excluding high income)', 'Iceland', 'Middle East & North Africa (IDA & IBRD countries)', 'Europe & Central Asia', 'South
Sudan', 'Pakistan', "Korea, Dem. People's Rep.", 'Latvia', 'China', 'Iran, Islamic Rep.', 'Tunisia', 'British Virgin Island
s', 'Africa Western and Central', 'IDA & IBRD total', 'Jamaica', 'Azerbaijan', 'Kyrgyz Republic', 'Albania', 'Malta', 'Philip
pines', 'Afghanistan', 'Barbados', 'Kuwait', 'Korea, Rep.', 'Palau', 'Sint Maarten (Dutch part)', 'Viet Nam', 'American Samo
a', 'Dominican Republic', 'Jordan', 'Mauritius', 'Montenegro', 'Ethiopia', 'Sao Tome and Principe', 'Qatar', 'Chad', 'Maurita
nia', 'Eritrea', 'Bahamas, The', 'Hungary', 'Russian Federation', 'Tonga', 'Moldova', 'Oman', 'Low & middle income', 'Equator
ial Guinea', 'Kazakhstan', 'Malawi', 'Italy', 'Malaysia', 'Other small states', 'Thailand', 'Togo', 'Yemen, Rep.', 'Bolivia',
'Cambodia', 'United Arab Emirates', 'Belarus', 'Aruba', 'Northern Mariana Islands', 'Seychelles', 'Rwanda', 'Syrian Arab Repu
blic', 'Pre-demographic dividend', 'Benin', 'Czechia', 'Turks and Caicos Islands', 'Virgin Islands (U.S.)', 'Libya'}
```

**selcting 5 European countries for our study

In [7]:
```
countries_to_study = ['Germany','Ireland','Poland','Greece','Italy']
gdp_filtered = gdp_data[gdp_data['Country Name'].isin(countries_to_study)]
```

In [8]:
```
gdp_filtered.head()
```

Out[8]:

	Country Name	2002	2003	2004	2005	2006	2007	2008	2009	2010	...	
55	Germany	2.078485e+12	2.501640e+12	2.814354e+12	2.846864e+12	2.994704e+12	3.425578e+12	3.745264e+12	3.411261e+12	3.399668e+12	...	3.733805
89	Greece	1.545642e+11	2.023701e+11	2.409636e+11	2.478754e+11	2.735467e+11	3.189028e+11	3.559087e+11	3.313085e+11	2.971250e+11	...	2.389077
111	Ireland	1.285960e+11	1.646708e+11	1.943721e+11	2.118770e+11	2.321806e+11	2.700793e+11	2.754475e+11	2.364431e+11	2.219136e+11	...	2.381125
116	Italy	1.276769e+12	1.577622e+12	1.806543e+12	1.858217e+12	1.949552e+12	2.213102e+12	2.408655e+12	2.199929e+12	2.136100e+12	...	2.141924
190	Poland	1.990704e+11	2.178287e+11	2.551073e+11	3.061459e+11	3.446267e+11	4.290208e+11	5.335998e+11	4.397316e+11	4.756966e+11	...	5.157620

5 rows × 22 columns

**Read the surface temperature data from the second dataset source

In [9]:
```
df2 = pd.read_csv(r"C:\Users\YUGAL\Downloads\GDL-Yearly-Average-Surface-Temperature-(ºC)-data.csv")
```

In [10]:
```
df2.head() # Rough picture of the data
```

Out[10]:

	Country	Continent	ISO_Code	Level	GDLCODE	Region	1990	1991	1992	1993	...	2013	2014	2015	2016	2017	2018	20
0	Afghanistan	Asia/Pacific	AFG	National	AFGt	Total	9.466	8.369	8.203	8.754	...	9.752	9.400	9.999	11.000	10.730	10.910	10.0
1	Afghanistan	Asia/Pacific	AFG	Subnat	AFGr101	Central (Kabul Wardak Kapisa Logar Parwan Panj...	5.755	4.302	4.228	5.013	...	5.216	5.403	5.920	7.382	6.887	7.200	5.7
2	Afghanistan	Asia/Pacific	AFG	Subnat	AFGr102	Central Highlands (Bamyan Daikundi)	4.144	3.013	2.819	3.501	...	4.200	3.920	4.462	5.528	5.369	5.909	4.3
3	Afghanistan	Asia/Pacific	AFG	Subnat	AFGr103	East (Nangarhar Kunar Laghman Nooristan)	8.965	7.778	7.696	8.362	...	8.808	8.618	8.942	10.320	10.090	9.796	9.1
4	Afghanistan	Asia/Pacific	AFG	Subnat	AFGr104	North (Samangan Sar-e-Pul Balkh Jawzjan Faryab)	11.460	10.840	10.560	10.660	...	12.070	11.520	12.330	12.910	12.660	12.750	12.6

5 rows × 39 columns

In [11]:
```
df2 = df2.dropna() # dropping all the missing values
```

Selecting the range of years for which we intend to conduct our study(data should match with the earlier dataframes from other sources) and selecting which columns to keep from the whole DataFrame.

In [12]:
```
year_range = range(2002,2023)
years = [str(i) for i in year_range]
cat_cols = ['Country','Level']
cols_to_keep = cat_cols + years
temp_data = df2[cols_to_keep]
```

Selecting the countries we intend to study(should match with the other dataframes)

In [13]:
```
untries_to_study = ['Germany','Ireland','Poland','Greece','Italy']
mp_filtered = temp_data[temp_data['Country'].isin(countries_to_study) & (temp_data['Level']=="National")].drop(columns=['Level'
```

In [14]:
```
temp_filtered.head()
```

Out[14]:

	Country	2002	2003	2004	2005	2006	2007	2008	2009	2010	...	2013	2014	2015	2016	2017	2018	2019	2020	202
933	Germany	9.677	9.541	9.181	9.307	9.864	10.150	9.780	9.458	8.055	...	8.926	10.580	10.140	9.775	9.819	10.680	10.55	10.71	9.39
962	Greece	15.530	15.420	15.380	15.120	15.100	15.980	15.980	15.890	16.350	...	16.250	16.160	15.860	16.190	15.850	16.490	16.35	16.24	16.22
1171	Ireland	10.100	10.140	10.050	10.200	10.280	10.460	9.726	9.693	8.707	...	9.707	10.230	9.609	9.903	10.220	10.050	10.09	10.04	10.25
1182	Italy	12.490	12.640	12.120	11.560	12.300	12.700	12.570	12.570	11.880	...	12.390	13.040	13.090	12.870	12.810	13.160	13.06	13.04	12.73
1812	Poland	9.201	8.363	8.378	8.353	8.750	9.407	9.472	8.624	7.567	...	8.550	9.731	9.864	9.272	9.006	9.905	10.33	10.03	8.66

5 rows × 22 columns

with this we are done with the data cleaning and preproccsing.