**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	...	201
0	Aruba	ABW	GDP (current US$)	NY.GDP.MKTP.CD	NaN	NaN	NaN	NaN	NaN	NaN	...	2.962907e+0
1	Africa Eastern and Southern	AFE	GDP (current US$)	NY.GDP.MKTP.CD	1.847810e+10	1.936631e+10	2.050647e+10	2.224273e+10	2.429433e+10	2.661956e+10	...	9.325135e+1
2	Afghanistan	AFG	GDP (current US$)	NY.GDP.MKTP.CD	5.377778e+08	5.488889e+08	5.466667e+08	7.511112e+08	8.000000e+08	1.006667e+09	...	1.913422e+1
3	Africa Western and Central	AFW	GDP (current US$)	NY.GDP.MKTP.CD	1.041165e+10	1.113592e+10	1.195171e+10	1.268581e+10	1.384900e+10	1.487476e+10	...	7.692632e+1
4	Angola	AGO	GDP (current US$)	NY.GDP.MKTP.CD	NaN	NaN	NaN	NaN	NaN	NaN	...	9.049642e+1

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 [5]: gdp_data.info() # Rough picutre of the subset of the data we selected
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 266 entries, 0 to 265
Data columns (total 22 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Country Name  266 non-null    object 
 1   2002          254 non-null    float64
 2   2003          254 non-null    float64
 3   2004          254 non-null    float64
 4   2005          254 non-null    float64
 5   2006          255 non-null    float64
 6   2007          255 non-null    float64
 7   2008          256 non-null    float64
 8   2009          256 non-null    float64
 9   2010          257 non-null    float64
 10  2011          260 non-null    float64
 11  2012          258 non-null    float64
 12  2013          259 non-null    float64
 13  2014          260 non-null    float64
 14  2015          258 non-null    float64
 15  2016          258 non-null    float64
 16  2017          258 non-null    float64
 17  2018          258 non-null    float64
 18  2019          258 non-null    float64
 19  2020          257 non-null    float64
 20  2021          255 non-null    float64
 21  2022          242 non-null    float64
dtypes: float64(21), object(1)
memory usage: 45.8+ KB
```

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

```
Out[6]: array(['Aruba', 'Africa Eastern and Southern', 'Afghanistan',
               'Africa Western and Central', 'Angola', 'Albania', 'Andorra',
               'Arab World', 'United Arab Emirates', 'Argentina', 'Armenia',
               'American Samoa', 'Antigua and Barbuda', 'Australia', 'Austria',
               'Azerbaijan', 'Burundi', 'Belgium', 'Benin', 'Burkina Faso',
               'Bangladesh', 'Bulgaria', 'Bahrain', 'Bahamas, The',
               'Bosnia and Herzegovina', 'Belarus', 'Belize', 'Bermuda',
               'Bolivia', 'Brazil', 'Barbados', 'Brunei Darussalam', 'Bhutan',
               'Botswana', 'Central African Republic', 'Canada',
               'Central Europe and the Baltics', 'Switzerland', 'Channel Islands',
               'Chile', 'China', "Cote d'Ivoire", 'Cameroon', 'Congo, Dem. Rep.',
               'Congo, Rep.', 'Colombia', 'Comoros', 'Cabo Verde', 'Costa Rica',
               'Caribbean small states', 'Cuba', 'Curacao', 'Cayman Islands',
               'Cyprus', 'Czechia', 'Germany', 'Djibouti', 'Dominica', 'Denmark',
               'Dominican Republic', 'Algeria',
               'East Asia & Pacific (excluding high income)',
               'Early-demographic dividend', 'East Asia & Pacific',
               'Europe & Central Asia (excluding high income)',
               'Europe & Central Asia', 'Ecuador', 'Egypt, Arab Rep.',
               'Euro area', 'Eritrea', 'Spain', 'Estonia', 'Ethiopia',
               'European Union', 'Fragile and conflict affected situations',
               'Finland', 'Fiji', 'France', 'Faroe Islands',
               'Micronesia, Fed. Sts.', 'Gabon', 'United Kingdom', 'Georgia',
               'Ghana', 'Gibraltar', 'Guinea', 'Gambia, The', 'Guinea-Bissau',
               'Equatorial Guinea', 'Greece', 'Grenada', 'Greenland', 'Guatemala',
               'Guam', 'Guyana', 'High income', 'Hong Kong SAR, China',
               'Honduras', 'Heavily indebted poor countries (HIPC)', 'Croatia',
               'Haiti', 'Hungary', 'IBRD only', 'IDA & IBRD total', 'IDA total',
               'IDA blend', 'Indonesia', 'IDA only', 'Isle of Man', 'India',
               'Not classified', 'Ireland', 'Iran, Islamic Rep.', 'Iraq',
               'Iceland', 'Israel', 'Italy', 'Jamaica', 'Jordan', 'Japan',
               'Kazakhstan', 'Kenya', 'Kyrgyz Republic', 'Cambodia', 'Kiribati',
               'St. Kitts and Nevis', 'Korea, Rep.', 'Kuwait',
               'Latin America & Caribbean (excluding high income)', 'Lao PDR',
               'Lebanon', 'Liberia', 'Libya', 'St. Lucia',
               'Latin America & Caribbean',
               'Least developed countries: UN classification', 'Low income',
               'Liechtenstein', 'Sri Lanka', 'Lower middle income',
               'Low & middle income', 'Lesotho', 'Late-demographic dividend',
               'Lithuania', 'Luxembourg', 'Latvia', 'Macao SAR, China',
               'St. Martin (French part)', 'Morocco', 'Monaco', 'Moldova',
               'Madagascar', 'Maldives', 'Middle East & North Africa', 'Mexico',
               'Marshall Islands', 'Middle income', 'North Macedonia', 'Mali',
               'Malta', 'Myanmar',
               'Middle East & North Africa (excluding high income)', 'Montenegro',
               'Mongolia', 'Northern Mariana Islands', 'Mozambique', 'Mauritania',
               'Mauritius', 'Malawi', 'Malaysia', 'North America', 'Namibia',
               'New Caledonia', 'Niger', 'Nigeria', 'Nicaragua', 'Netherlands',
               'Norway', 'Nepal', 'Nauru', 'New Zealand', 'OECD members', 'Oman',
               'Other small states', 'Pakistan', 'Panama', 'Peru', 'Philippines',
               'Palau', 'Papua New Guinea', 'Poland', 'Pre-demographic dividend',
               'Puerto Rico', "Korea, Dem. People's Rep.", 'Portugal', 'Paraguay',
               'West Bank and Gaza', 'Pacific island small states',
               'Post-demographic dividend', 'French Polynesia', 'Qatar',
               'Romania', 'Russian Federation', 'Rwanda', 'South Asia',
               'Saudi Arabia', 'Sudan', 'Senegal', 'Singapore', 'Solomon Islands',
               'Sierra Leone', 'El Salvador', 'San Marino', 'Somalia', 'Serbia',
               'Sub-Saharan Africa (excluding high income)', 'South Sudan',
               'Sub-Saharan Africa', 'Small states', 'Sao Tome and Principe',
               'Suriname', 'Slovak Republic', 'Slovenia', 'Sweden', 'Eswatini',
               'Sint Maarten (Dutch part)', 'Seychelles', 'Syrian Arab Republic',
               'Turks and Caicos Islands', 'Chad',
               'East Asia & Pacific (IDA & IBRD countries)',
               'Europe & Central Asia (IDA & IBRD countries)', 'Togo', 'Thailand',
               'Tajikistan', 'Turkmenistan',
               'Latin America & the Caribbean (IDA & IBRD countries)',
               'Timor-Leste', 'Middle East & North Africa (IDA & IBRD countries)',
               'Tonga', 'South Asia (IDA & IBRD)',
               'Sub-Saharan Africa (IDA & IBRD countries)', 'Trinidad and Tobago',
               'Tunisia', 'Turkiye', 'Tuvalu', 'Tanzania', 'Uganda', 'Ukraine',
               'Upper middle income', 'Uruguay', 'United States', 'Uzbekistan',
               'St. Vincent and the Grenadines', 'Venezuela, RB',
               'British Virgin Islands', 'Virgin Islands (U.S.)', 'Viet Nam',
               'Vanuatu', 'World', 'Samoa', 'Kosovo', 'Yemen, Rep.',
               'South Africa', 'Zambia', 'Zimbabwe'], dtype=object)
```

**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	...	3.733
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	...	2.389
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	...	2.381
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	...	2.141
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	...	5.157

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	
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	1
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	
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	
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	
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	1

5 rows × 39 columns

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

In [12]:
```
df2['Country'].unique()
```

Out[12]:
```
array(['Afghanistan', 'Albania', 'Algeria', 'Andorra', 'Angola',
       'Antigua and Barbuda', 'Argentina urban', 'Armenia', 'Australia',
       'Austria', 'Azerbaijan', 'Bahamas', 'Bahrain', 'Bangladesh',
       'Barbados', 'Belarus', 'Belgium', 'Belize', 'Benin', 'Bhutan',
       'Bolivia', 'Bosnia and Herzegovina', 'Botswana', 'Brazil',
       'Brunei Darussalam', 'Bulgaria', 'Burkina Faso', 'Burundi',
       'Cambodia', 'Cameroon', 'Canada', 'Cape Verde',
       'Central African Republic CAR', 'Chad', 'Chili', 'China',
       'Colombia', 'Comoros', 'Congo Brazzaville',
       'Congo Democratic Republic', 'Costa Rica', "Cote d'Ivoire",
       'Croatia', 'Cuba', 'Cyprus', 'Czech Republic', 'Denmark',
       'Djibouti', 'Dominica', 'Dominican Republic', 'Ecuador', 'Egypt',
       'El Salvador', 'Equatorial Guinea', 'Eritrea', 'Estonia',
       'Eswatini', 'Ethiopia', 'Fiji', 'FInland', 'Finland', 'France',
       'French Polynesia', 'Gabon', 'Gambia', 'Georgia', 'Germany',
       'Ghana', 'Gibraltar', 'Greece', 'Greenland', 'Grenada',
       'Guatemala', 'Guinea', 'Guinea Bissau', 'Guyana', 'Haiti',
       'Honduras', 'Hungary', 'Iceland', 'India', 'Indonesia', 'Iran',
       'Iraq', 'Ireland', 'Isle of Man', 'Israel', 'Italy', 'Jamaica',
       'Japan', 'Jersey', 'Jordan', 'Kazakhstan', 'Kenya', 'Kiribati',
       'Kosovo', 'Kuwait', 'Kyrgyzstan', 'Lao', 'Latvia', 'Lebanon',
       'Lesotho', 'Liberia', 'Libya', 'Liechtenstein', 'Lithuania',
       'Luxembourg', 'Madagascar', 'Malawi', 'Malaysia', 'Mali',
       'Mauritania', 'Mauritius', 'Mexico',
       'Micronesia (Federated States of)', 'Moldova', 'Monaco',
       'Mongolia', 'Monte Negro', 'Morocco', 'Mozambique', 'Myanmar',
       'Namibia', 'Nepal', 'Netherlands', 'New Zealand', 'Nicaragua',
       'Niger', 'Nigeria', 'North Macedonia', 'Norway', 'Oman',
       'Pakistan', 'Palestine', 'Panama', 'Papua New Guinea', 'Paraguay',
       'Peru', 'Philippines', 'Poland', 'Portugal', 'Qatar', 'Romania',
       'Russian Federation', 'Rwanda', 'Saint Kitts and Nevis',
       'Saint Lucia', 'Saint Vincent and the Grenadines', 'Samoa',
       'San Marino', 'Sao Tome & Principe', 'Saudi Arabia', 'Senegal',
       'Serbia', 'Sierra Leone', 'Singapore', 'Slovakia', 'Slovenia',
       'Solomon Islands', 'Somalia', 'South Africa', 'South Korea',
       'South Sudan', 'Spain', 'Sri Lanka', 'Sudan', 'Suriname',
       'Svalbard and Jan Mayen', 'Sweden', 'Switzerland', 'Syria',
       'Tajikistan', 'Tanzania', 'Thailand', 'Timor Leste', 'Togo',
       'Tonga', 'Trinidad & Tobago', 'Tunisia', 'Turkey', 'Turkmenistan',
       'Turks & Caicos Islands', 'Uganda', 'Ukraine',
       'United Arab Emirates', 'United Kingdom', 'United States',
       'Uruguay', 'Uzbekistan', 'Vanuatu', 'Vatican City', 'Venezuela',
       'Vietnam', 'Yemen', 'Zambia', 'Zimbabwe'], dtype=object)
```

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 [13]:
```
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 [14]:
```
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=['Leve
```

In [15]:
```
temp_filtered.head()
```

Out[15]:

	Country	2002	2003	2004	2005	2006	2007	2008	2009	2010	...	2013	2014	2015	2016	2017	2018	2019	2020	2
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.
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.
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.
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.
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.

5 rows × 22 columns

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