Causes of Death with respect to Countries by Year

Dataset link:
https://www.kaggle.com/datasets/ivanchvez/causes-of-death-our-world-in-data

People die from certain causes. Most of the time, the causes are recorded. What did they die from? This dataset might answer some questions.

Quick Look at The Dataset:

In [1]:
```
import pandas as pd
file_path = "20220327 annual-number-of-deaths-by-cause.csv"
df = pd.read_csv(file_path)
df.head()
```

Out[1]:

	Entity	Code	Year	Number of executions (Amnesty International)	Deaths - Meningitis - Sex: Both - Age: All Ages (Number)	Deaths - Neoplasms - Sex: Both - Age: All Ages (Number)	Deaths - Fire, heat, and hot substances - Sex: Both - Age: All Ages (Number)	Deaths - Malaria - Sex: Both - Age: All Ages (Number)	Deaths - Drowning - Sex: Both - Age: All Ages (Number)	Deaths - Interpersonal violence - Sex: Both - Age: All Ages (Number)	...	Deaths - Protein-energy malnutrition - Sex: Both - Age: All Ages (Number)	Terrorism (deaths)	Deaths - Cardiovascular diseases - Sex: Both - Age: All Ages (Number)	
0	Afghanistan	AFG	2007	15	2933.0	15925.0	481.0	393.0	2127.0	3657.0	...	2439.0	1199.0	53962.0	
1	Afghanistan	AFG	2008	17	2731.0	16148.0	462.0	255.0	1973.0	3785.0	...	2231.0	1092.0	54051.0	
2	Afghanistan	AFG	2009	0	2460.0	16383.0	448.0	239.0	1852.0	3874.0	...	1998.0	1065.0	53964.0	
3	Afghanistan	AFG	2011	2	2327.0	17094.0	448.0	390.0	1775.0	4170.0	...	1805.0	1525.0	54347.0	
4	Afghanistan	AFG	2012	14	2254.0	17522.0	445.0	94.0	1716.0	4245.0	...	1667.0	3521.0	54868.0	

5 rows × 36 columns

Column names are too long for future visualizations. Mapping would solve this problem.

In [2]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8254 entries, 0 to 8253
Data columns (total 36 columns):
 #   Column                                                                                Non-Null Count  Dtype
---  ------                                                                                --------------  -----
 0   Entity                                                                                8254 non-null   object
 1   Code                                                                                  6206 non-null   object
 2   Year                                                                                  8254 non-null   int64
 3   Number of executions (Amnesty International)                                          267 non-null    object
 4   Deaths - Meningitis - Sex: Both - Age: All Ages (Number)                              8010 non-null   float64
 5   Deaths - Neoplasms - Sex: Both - Age: All Ages (Number)                               8010 non-null   float64
 6   Deaths - Fire, heat, and hot substances - Sex: Both - Age: All Ages (Number)          8010 non-null   float64
 7   Deaths - Malaria - Sex: Both - Age: All Ages (Number)                                 8010 non-null   float64
 8   Deaths - Drowning - Sex: Both - Age: All Ages (Number)                                8010 non-null   float64
 9   Deaths - Interpersonal violence - Sex: Both - Age: All Ages (Number)                  8010 non-null   float64
 10  Deaths - HIV/AIDS - Sex: Both - Age: All Ages (Number)                                8010 non-null   float64
 11  Deaths - Drug use disorders - Sex: Both - Age: All Ages (Number)                      8010 non-null   float64
 12  Deaths - Tuberculosis - Sex: Both - Age: All Ages (Number)                            8010 non-null   float64
 13  Deaths - Road injuries - Sex: Both - Age: All Ages (Number)                           8010 non-null   float64
 14  Deaths - Maternal disorders - Sex: Both - Age: All Ages (Number)                      8010 non-null   float64
```

Expected value counts for every column: 8254. We have missing values.

Possible solutions for our missing values:

- Filling with 0s
- Filling with mean value
- Filling with every country's respective mean value for that column
- Filling with some other constant value
- Filling with mode/median of the column
- Forward filling

- Backward filling
- Using interpolation to fill the missing values
- Dropping the rows that have missing values

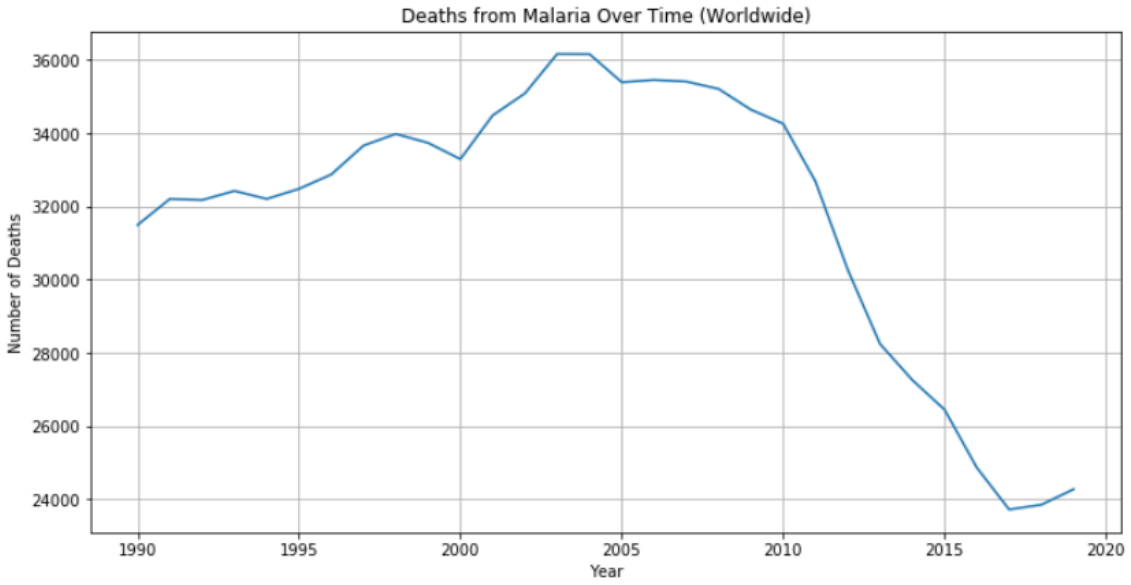
Depending on the nature of the dataset and the feature information, the choice can differ. In an industrial setting, the domain knowledge and comments from a senior data analyst comes into play here.

For our case, Number of executions (Amnesty International) can be filled with 0. Because, the absence of data suggests that there were no deaths by this reason, considering the event.

For the other missing values, 244 over 8254 is rather small. We could choose any approach and it would not disrupt the analysis catastrophically. But to be conservative, we will fill with mean values with respect to the country of that missing value.

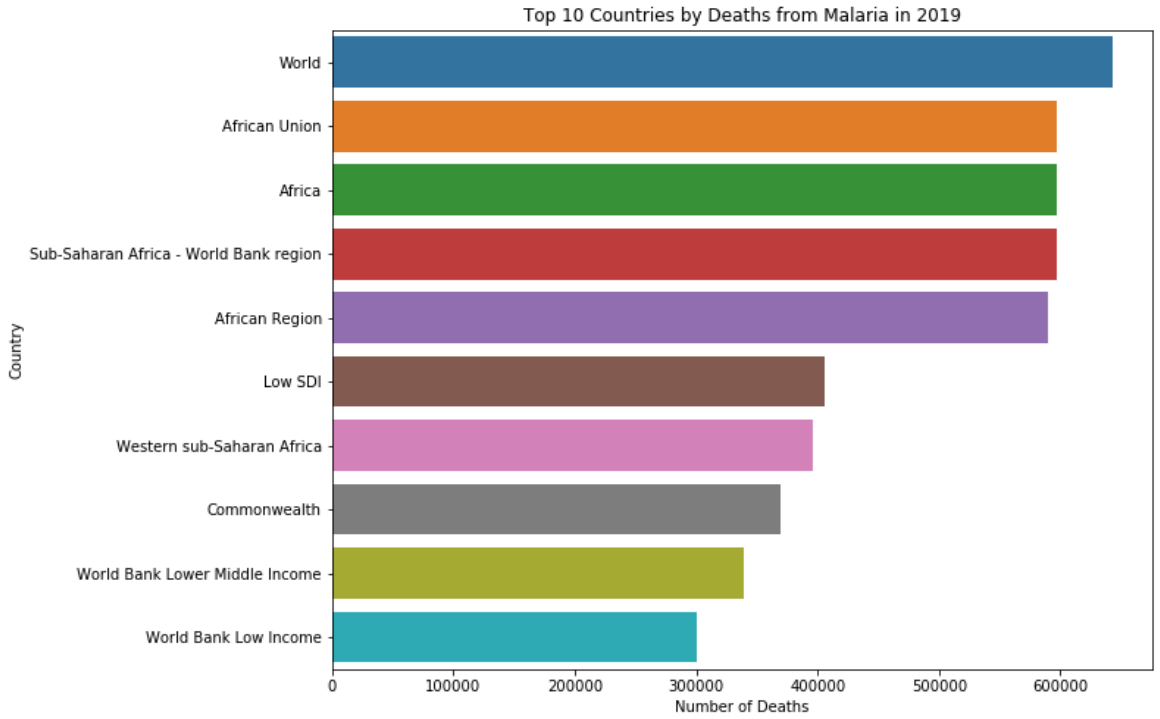
After having a fully populated dataframe, inspecting some descriptions and statistics, we can do quick visualizations to get a better feeling about the data.

Malaria, for example:



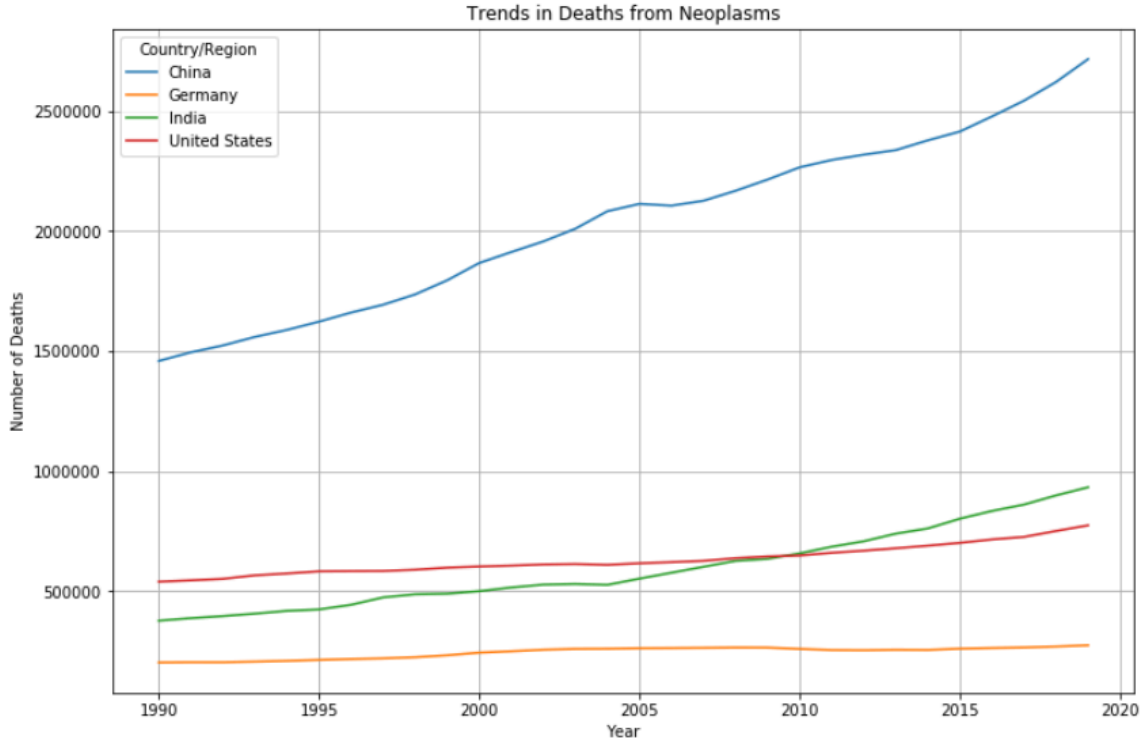
Since we know that the population of the world increases over these years, even on a per capita basis, whatever the world is doing, it works. We can safely say that deaths caused by malaria are decreasing.

For 2019, to inspect top 10 countries and compare, we can use a bar plot. For reference, we will keep the world in the figure.



Many more can be done. You can inspect the code from my github link.

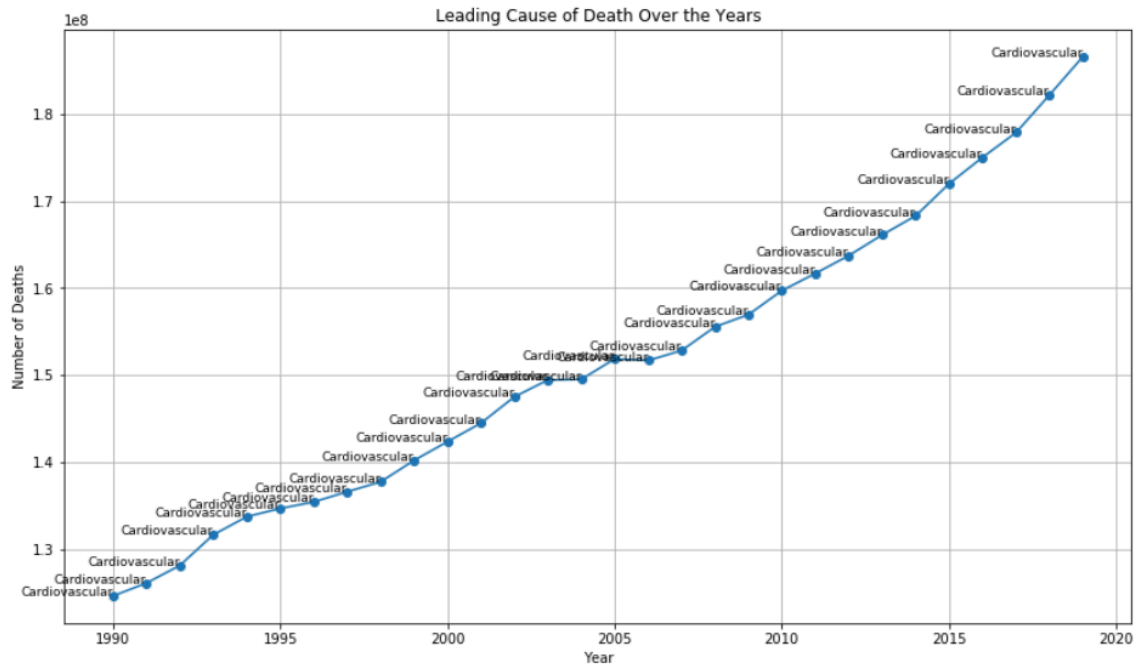
There is a column called neoplasms. A quick search gives the necessary domain knowledge: Neoplasms, commonly known as tumors, are abnormal growths of tissue that arise from uncontrolled, progressive multiplication of cells. These growths can be benign (non-cancerous) or malignant (cancerous). In practice, non-technical people might call “cancer” to all of the cases.



It is obvious that the trend shows an increase in deaths from neoplasms irrespective of the country. Maybe faster in some countries than others.

Other than these quick exploratory looks, one might have possible, specific questions:

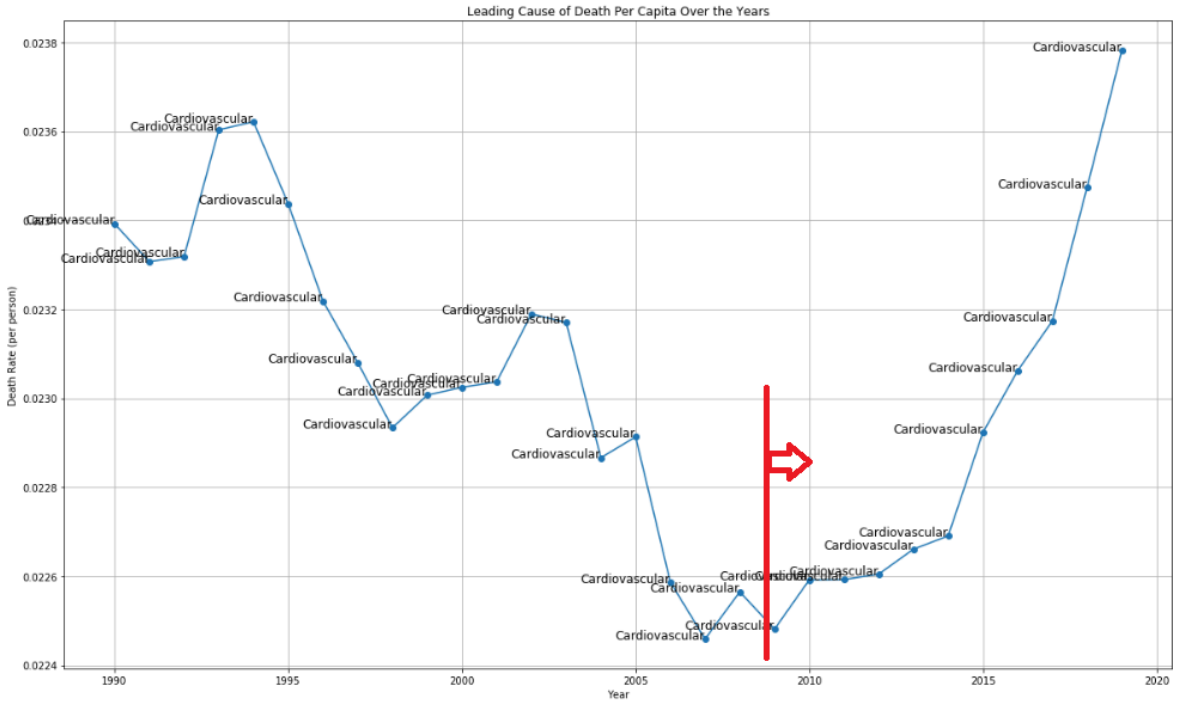
What is the leading cause of death irrespective of the country, throughout all years ? To answer this:



It can be seen that since 1990, the leading cause of death throughout the world is cardiovascular diseases.

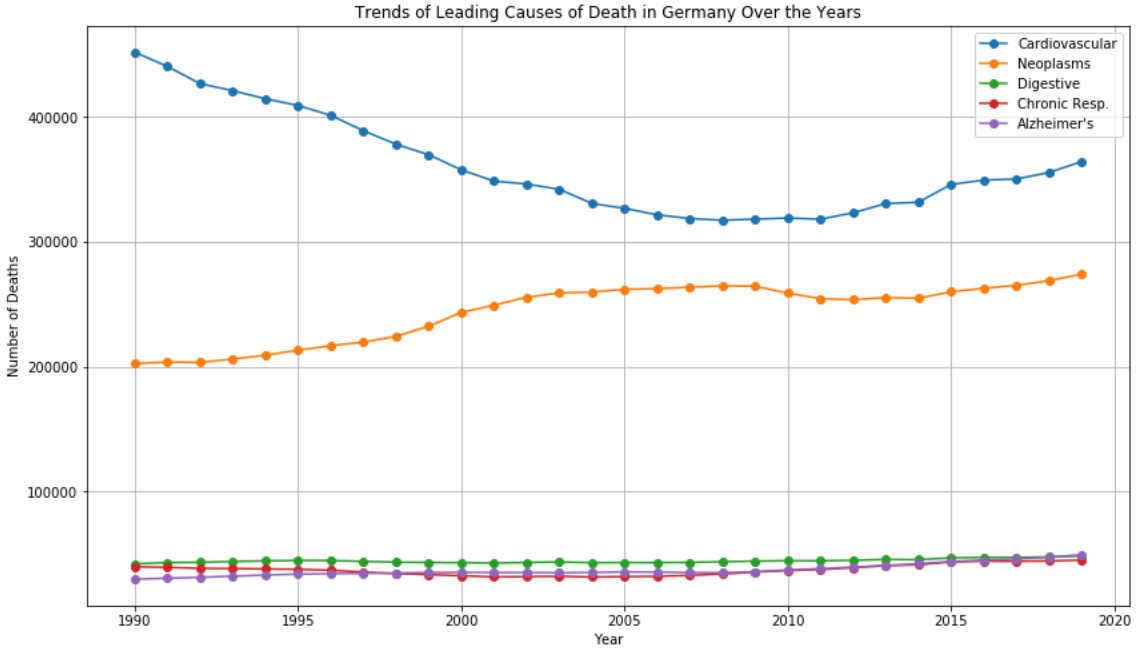
But we know that the world population increases. Maybe it's just that we have more people, so it is seen more often, but it is not a mathematically significant increase ?

Simple google search for world population every year, dividing the death counts with the respective year and plotting the result would answer this question.



We can safely say that after 2009, deaths from cardiovascular diseases increased even on a per capita level.

Since I am based in Germany, I was curious about the data related to Germany. I wanted to know the leading causes of death in Germany throughout the years, and if the top 5 causes changes or not.



With some melting and sorting, it was surprising to see, consistently the top 5 leading causes of deaths do not change over the years:

```
In [26]: # Frequency of causes appearing in top 5
         top5_frequency = top5_deaths_germany['Cause'].value_counts()
         top5_frequency

Out[26]: Cardiovascular    30
         Neoplasms         30
         Digestive         30
         Chronic Resp.     30
         Alzheimer's       30
         Name: Cause, dtype: int64
```

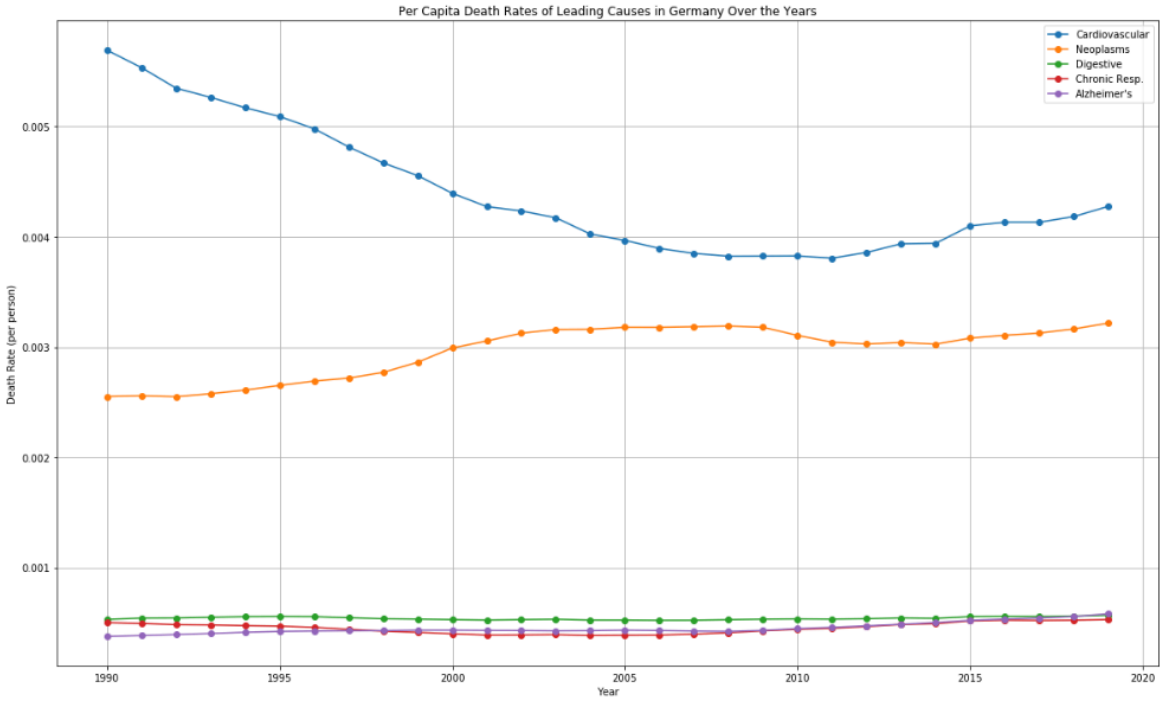
In [32]:
```
# Melt the dataframe to long format
melted_germany = germany_data.melt(id_vars=['Entity', 'Code', 'Year'],
                                   var_name='Cause',
                                   value_name='Death_Count')

# Ensure Death_Count is numeric
melted_germany['Death_Count'] = pd.to_numeric(melted_germany['Death_Count'], errors='coerce')

# Group by Year and Cause, then sum the Death_Count
yearly_deaths_germany = melted_germany.groupby(['Year', 'Cause'])['Death_Count'].sum().reset_index()

# Find top 5 leading causes of death for each year
top5_deaths_germany = yearly_deaths_germany.groupby('Year').apply(
    lambda x: x.nlargest(5, 'Death_Count')
).reset_index(drop=True)

top5_deaths_germany
```

Out[32]:

	Year	Cause	Death_Count
0	1990	Cardiovascular	451910.0
1	1990	Neoplasms	202621.0
2	1990	Digestive	42169.0
3	1990	Chronic Resp.	39888.0
4	1990	Alzheimer's	29984.0
...	...	...	...
145	2019	Cardiovascular	364285.0
146	2019	Neoplasms	274088.0
147	2019	Alzheimer's	49557.0
148	2019	Digestive	48420.0
149	2019	Chronic Resp.	45165.0

150 rows × 3 columns

Since Germany's population is rather stable, even we plot the per capita lines, the trends will not change that much, but just to be sure:



Many more can be done with this dataset. I am curious and will continue to play around with this dataset. Feel free to give me feedback or if you have interesting questions that we can answer through this data, let me know.

Aytekin Yenilmez.