Exploring and working with published data from the Global Burden of Disease study started by the World Health Organization (WHO) and continued by the Institute of Health Metrics and Evaluation (IHME). Dataset supplied by Our World in Data.

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

```
df = pd.read_csv('cause_of_deaths.csv')
```

```
df.head()
```

	Country/Territory	Code	Year	Meningitis	Alzheimer's Disease and Other Dementias	Parkinson's Disease	Nutrit. Deficie
0	Afghanistan	AFG	1990	2159	1116	371	
1	Afghanistan	AFG	1991	2218	1136	374	
2	Afghanistan	AFG	1992	2475	1162	378	
3	Afghanistan	AFG	1993	2812	1187	384	
4	Afghanistan	AFG	1994	3027	1211	391	

5 rows × 34 columns

```
#Nice clean dataset
df.isnull().sum()
```

```
Country/Territory                                0
Code                                             0
Year                                             0
Meningitis                                       0
Alzheimer's Disease and Other Dementias          0
Parkinson's Disease                              0
Nutritional Deficiencies                         0
Malaria                                          0
Drowning                                         0
Interpersonal Violence                           0
Maternal Disorders                               0
HIV/AIDS                                         0
Drug Use Disorders                               0
Tuberculosis                                     0
Cardiovascular Diseases                          0
Lower Respiratory Infections                     0
Neonatal Disorders                               0
Alcohol Use Disorders                            0
Self-harm                                        0
Exposure to Forces of Nature                     0
Diarrheal Diseases                               0
Environmental Heat and Cold Exposure             0
Neoplasms                                        0
Conflict and Terrorism                           0
Diabetes Mellitus                                0
Chronic Kidney Disease                           0
Poisonings                                       0
Protein-Energy Malnutrition                      0
Road Injuries                                    0
Chronic Respiratory Diseases                     0
Cirrhosis and Other Chronic Liver Diseases       0
Digestive Diseases                               0
Fire, Heat, and Hot Substances                   0
Acute Hepatitis                                  0
Death Toll                                       0
dtype: int64
```

```
df.duplicated().sum()
```

```
0
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6120 entries, 0 to 6119
Data columns (total 34 columns):
 #   Column                                        Non-Null Count  Dtype 
---  ------                                        --------------  ----- 
 0   Country/Territory                             6120 non-null   object
 1   Code                                          6120 non-null   object
 2   Year                                          6120 non-null   int64 
 3   Meningitis                                    6120 non-null   int64 
 4   Alzheimer's Disease and Other Dementias       6120 non-null   int64 
 5   Parkinson's Disease                           6120 non-null   int64 
 6   Nutritional Deficiencies                      6120 non-null   int64 
 7   Malaria                                       6120 non-null   int64 
 8   Drowning                                      6120 non-null   int64 
 9   Interpersonal Violence                        6120 non-null   int64 
 10  Maternal Disorders                            6120 non-null   int64 
 11  HIV/AIDS                                      6120 non-null   int64 
 12  Drug Use Disorders                            6120 non-null   int64 
 13  Tuberculosis                                  6120 non-null   int64 
 14  Cardiovascular Diseases                       6120 non-null   int64 
 15  Lower Respiratory Infections                  6120 non-null   int64 
 16  Neonatal Disorders                            6120 non-null   int64 
 17  Alcohol Use Disorders                         6120 non-null   int64 
 18  Self-harm                                     6120 non-null   int64 
 19  Exposure to Forces of Nature                  6120 non-null   int64 
 20  Diarrheal Diseases                            6120 non-null   int64 
 21  Environmental Heat and Cold Exposure          6120 non-null   int64 
 22  Neoplasms                                     6120 non-null   int64 
 23  Conflict and Terrorism                        6120 non-null   int64 
 24  Diabetes Mellitus                             6120 non-null   int64 
 25  Chronic Kidney Disease                        6120 non-null   int64 
 26  Poisonings                                    6120 non-null   int64 
 27  Protein-Energy Malnutrition                   6120 non-null   int64 
 28  Road Injuries                                 6120 non-null   int64 
 29  Chronic Respiratory Diseases                  6120 non-null   int64 
 30  Cirrhosis and Other Chronic Liver Diseases    6120 non-null   int64 
 31  Digestive Diseases                            6120 non-null   int64 
 32  Fire, Heat, and Hot Substances                6120 non-null   int64 
 33  Acute Hepatitis                               6120 non-null   int64 
dtypes: int64(32), object(2)
memory usage: 1.6+ MB
```

```
#Create new column for total number of deaths
death_total = [col for col in df.columns if col not in ['Country', 'Code', 'Year']]
df['Death Toll'] = df[death_total].sum(axis=1)
df.head()
```

```
<ipython-input-8-a2715a86e671>:3: FutureWarning: Dropping of nuisance columns
  df['Death Toll'] = df[death_total].sum(axis=1)
```

	Country/Territory	Code	Year	Meningitis	Alzheimer's Disease and Other Dementias	Parkinson's Disease	Nutrit Deficie
0	Afghanistan	AFG	1990	2159	1116	371	
1	Afghanistan	AFG	1991	2218	1136	374	
2	Afghanistan	AFG	1992	2475	1162	378	
3	Afghanistan	AFG	1993	2812	1187	384	
4	Afghanistan	AFG	1994	3027	1211	391	

5 rows × 35 columns

```
#Death total from each disease
#A dizzying amount of death, led by CV disease (a near two length lead!)
disease_df = df[death_total].sum().to_frame().reset_index()
disease_df.rename(columns={"index": "Disease", 0:"Total Cases"}, inplace=True)
disease_df.drop(index=disease_df.index[0], inplace=True)
disease_df
```

	Disease	Total Cases
1	Meningitis	10524572
2	Alzheimer's Disease and Other Dementias	29768839
3	Parkinson's Disease	7179795
4	Nutritional Deficiencies	13792032
5	Malaria	25342676
6	Drowning	10301999
7	Interpersonal Violence	12752839

8	Maternal Disorders	7727046
9	HIV/AIDS	36364419
10	Drug Use Disorders	2656121
11	Tuberculosis	45850603
12	Cardiovascular Diseases	447741982
13	Lower Respiratory Infections	83770038
14	Neonatal Disorders	76860729
15	Alcohol Use Disorders	4819018
16	Self-harm	23713931
17	Exposure to Forces of Nature	1490132
18	Diarrheal Diseases	66235508
19	Environmental Heat and Cold Exposure	1788851
20	Neoplasms	229758538
21	Conflict and Terrorism	3294053
22	Diabetes Mellitus	31448872
23	Chronic Kidney Disease	28911692
24	Poisonings	2601082
25	Protein-Energy Malnutrition	12031885
26	Road Injuries	36296469
27	Chronic Respiratory Diseases	104605334
28	Cirrhosis and Other Chronic Liver Diseases	37479321
29	Digestive Diseases	65638635
30	Fire, Heat, and Hot Substances	3602914
31	Acute Hepatitis	3784791

```
#Total number of deaths per country
#Appears to be a clear and understandable correlation between population size and d
country_df = df.groupby('Country/Territory')['Death Toll'].sum().sort_values(ascend
country_df
```

	Country/Territory	Death Toll
0	China	265408106
1	India	238158165
2	United States	71197802
3	Russia	59591155
4	Indonesia	44046941
...	...	...
199	Cook Islands	3999
200	Tuvalu	2962
201	Nauru	2249
202	Niue	591
203	Tokelau	299

204 rows × 2 columns

```
#Total deaths by year
deaths_by_year = df.groupby('Year')['Death Toll'].sum().reset_index()
deaths_by_year
```

	Year	Death Toll
0	1990	43518516
1	1991	44059729
2	1992	44459130
3	1993	45185713
4	1994	46182613
5	1995	46177018
6	1996	46320827

7	1997	46672370
8	1998	47066088
9	1999	47652090
10	2000	48050317
11	2001	48385692
12	2002	48897031
13	2003	49123952
14	2004	49330171
15	2005	49591909
16	2006	49424521
17	2007	49495216
18	2008	50115740
19	2009	49900666
20	2010	50422775
21	2011	50413303
22	2012	50597654
23	2013	50931550
24	2014	51268375
25	2015	51856393
26	2016	52337435
27	2017	52789758
28	2018	53545244
29	2019	54362920

```
#Group years with country and measure yearly death totals
df_country_group = df.groupby(['Country/Territory','Year']).sum()
df_country_group
```

```
<ipython-input-12-8ad16df488f4>:2: FutureWarning: The default value of numeric
  df_country_group = df.groupby(['Country/Territory','Year']).sum()
```

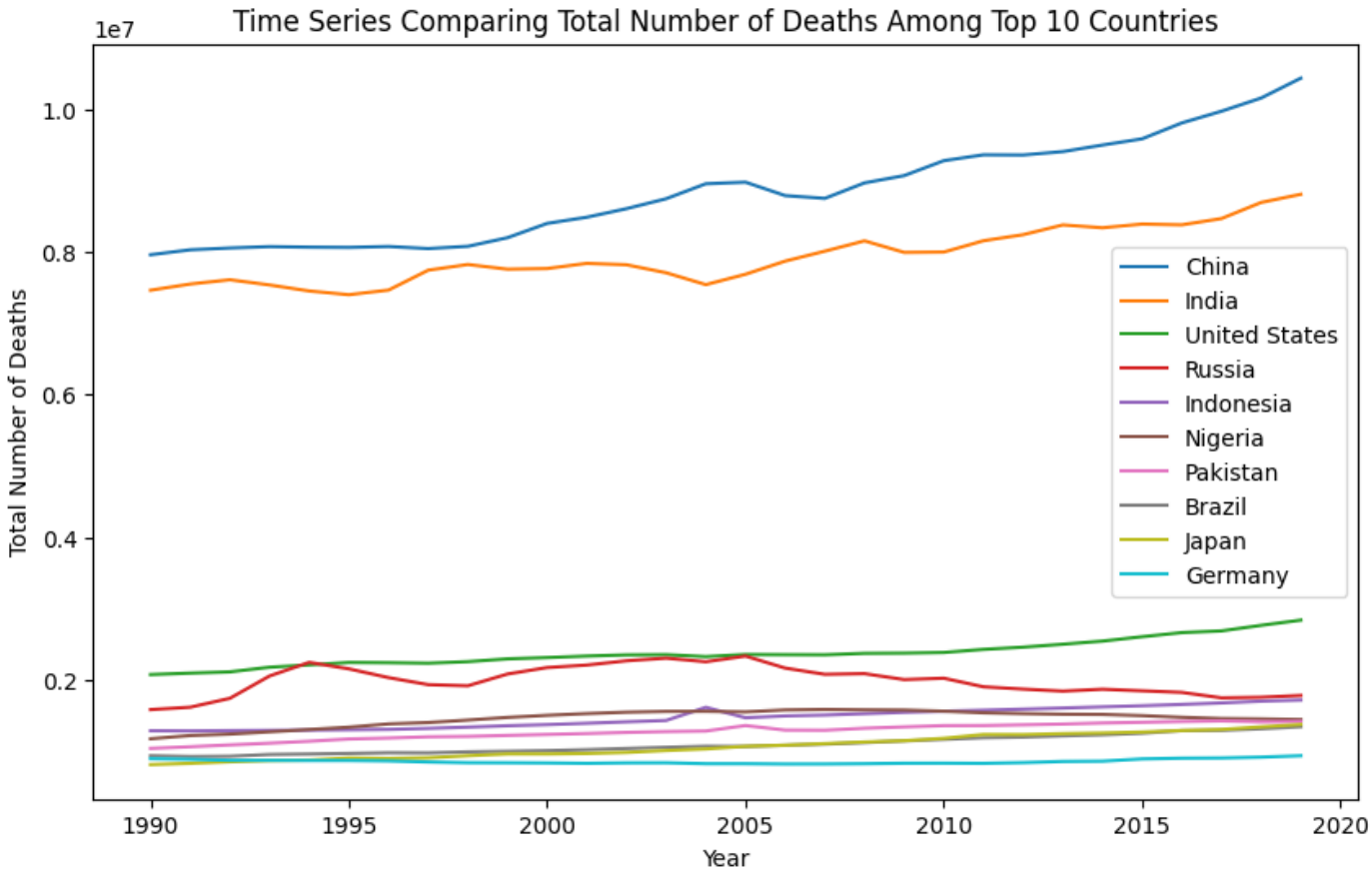
		Meningitis	Alzheimer's Disease and Other Dementias	Parkinson's Disease	Nutritional Deficiencies	Ma
Country/Territory	**Year**					
Afghanistan	**1990**	2159	1116	371	2087	
	1991	2218	1136	374	2153	
	1992	2475	1162	378	2441	
	1993	2812	1187	384	2837	
	1994	3027	1211	391	3081	
...	**...**	...	...	...	...	
Zimbabwe	**2015**	1439	754	215	3019	
	2016	1457	767	219	3056	
	2017	1460	781	223	2990	
	2018	1450	795	227	2918	
	2019	1450	812	232	2884	

6120 rows × 32 columns

```
#Top ten countries by total death
top_10 = df.groupby('Country/Territory')['Death Toll'].sum().sort_values(ascending=
top_10
```

	Country/Territory	Death Toll
0	China	265408106
1	India	238158165
2	United States	71197802
3	Russia	59591155
4	Indonesia	44046941
5	Nigeria	43670014
6	Pakistan	38151878
7	Brazil	32674112
8	Japan	31922807
9	Germany	25559667

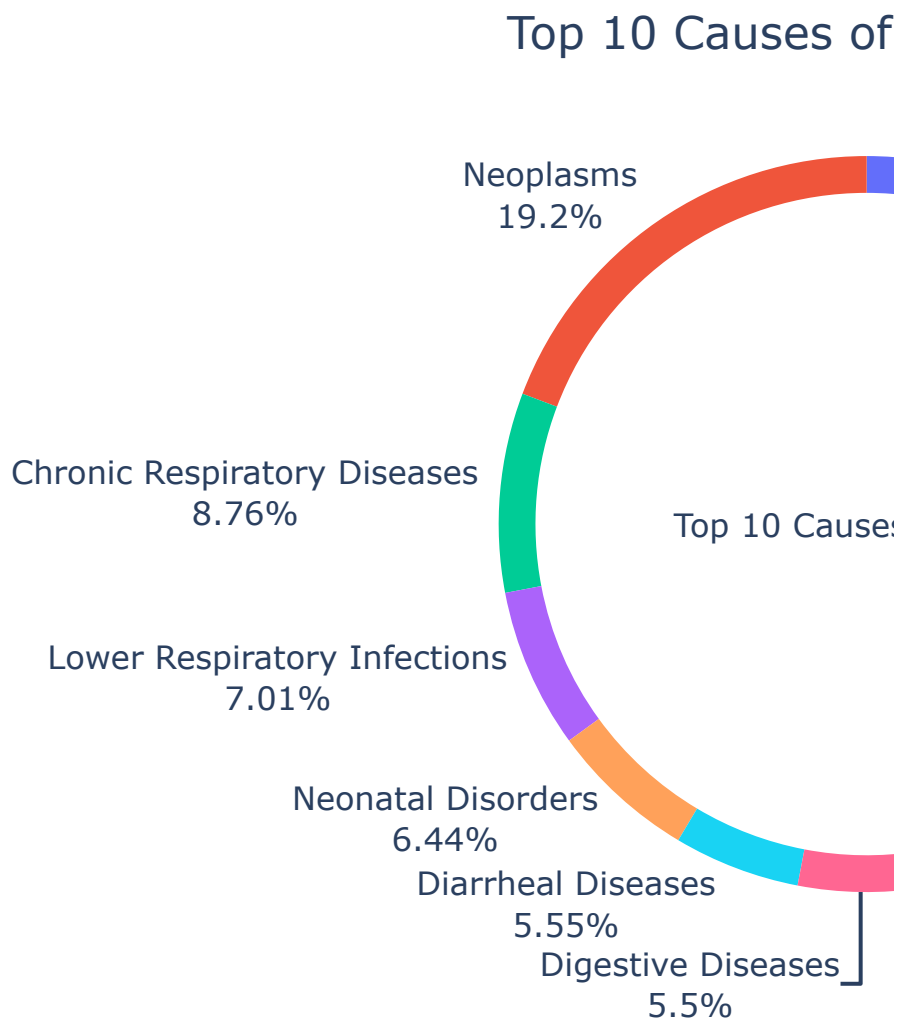
```
#Timeseries of death for the countries with highest number of death
#Shows a relatively steady state that appears to climb in proportionately with grow
plt.figure(figsize=(10,6))

for i in top_10['Country/Territory']:
    a = df[df['Country/Territory']==i]
    sns.lineplot(data=a, x='Year', y='Death Toll', label=i)

plt.xlabel('Year',fontsize =10)
plt.ylabel('Total Number of Deaths',fontsize =10)
plt.title('Time Series Comparing Total Number of Deaths Among Top 10 Countries', fo
plt.legend()
plt.show()
```



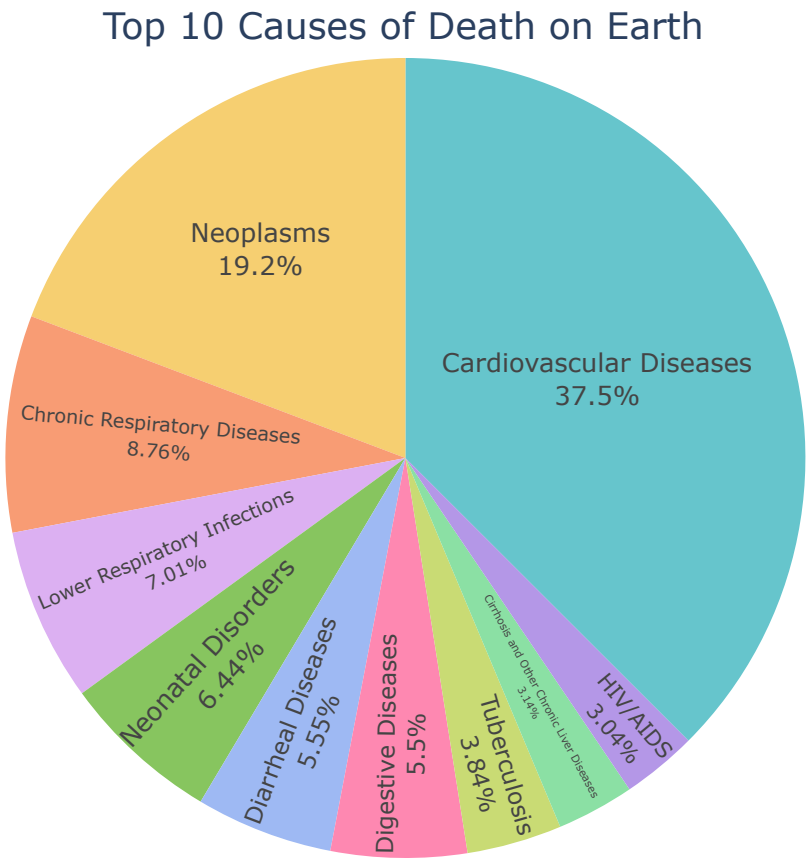
```
#Top causes of death
reapers_top_henchmen = disease_df.groupby('Disease')['Total Cases'].sum().sort_valu
reapers_top_henchmen
```

	Disease	Total Cases
0	Cardiovascular Diseases	447741982
1	Neoplasms	229758538
2	Chronic Respiratory Diseases	104605334
3	Lower Respiratory Infections	83770038
4	Neonatal Disorders	76860729
5	Diarrheal Diseases	66235508
6	Digestive Diseases	65638635
7	Tuberculosis	45850603
8	Cirrhosis and Other Chronic Liver Diseases	37479321
9	HIV/AIDS	36364419

```
#Visualizing with plotly, the top 10 causes of death on earth
import plotly.graph_objects as go
fig = go.Figure(data=[go.Pie(labels=reapers_top_henchmen['Disease'],
                             values=reapers_top_henchmen['Total Cases'],
                             hole=.9,
                             title = 'Top 10 Causes of Death')])

fig.update_layout(title='Top 10 Causes of Death on Earth',title_x=0.5, font_size=15
fig.update_traces(textposition='outside', textinfo='percent+label')
fig.show()
```



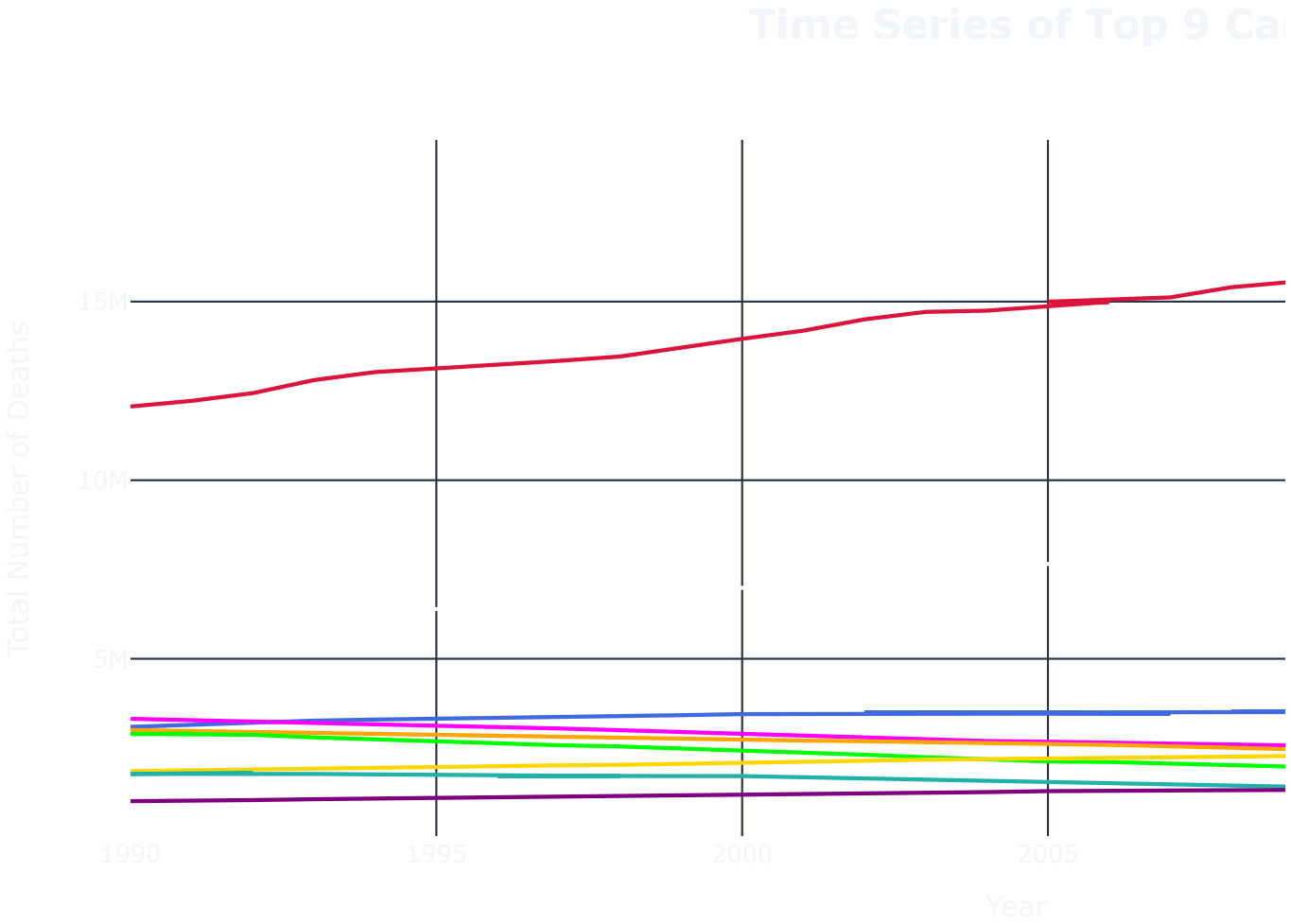
```
#Visualizing with plotly express, the top 10 causes of death on earth
fig = px.pie(reapers_top_henchmen, names = 'Disease' , values = 'Total Cases',
             color_discrete_sequence=px.colors.qualitative.Pastel)
fig.update_traces(textposition='inside', textinfo='percent+label')
fig.update_layout(title_text='Top 10 Causes of Death on Earth',
                  title_x=0.5, title_y=0.99, width=600, height=400,
                  margin=dict(t=25, b=0, l=0, r=0))
fig.update(layout_showlegend=False)
```



```
#Grouping top killers by year in preparation for a time series visualization
TS_CV = df.groupby('Year')['Cardiovascular Diseases'].sum().sort_values(ascending=F
TS_Neo = df.groupby('Year')['Neoplasms'].sum().sort_values(ascending=False).reset_i
TS_CRD = df.groupby('Year')['Chronic Respiratory Diseases'].sum().sort_values(ascen
TS_LLD = df.groupby('Year')['Lower Respiratory Infections'].sum().sort_values(ascen
TS_Neonat = df.groupby('Year')['Neonatal Disorders'].sum().sort_values(ascending=Fa
TS_Diarrhea = df.groupby('Year')['Diarrheal Diseases'].sum().sort_values(ascending=
TS_DD = df.groupby('Year')['Digestive Diseases'].sum().sort_values(ascending=False)
TS_TB = df.groupby('Year')['Tuberculosis'].sum().sort_values(ascending=False).reset
TS_LD = df.groupby('Year')['Cirrhosis and Other Chronic Liver Diseases'].sum().sort
TS_HIV = df.groupby('Year')['HIV/AIDS'].sum().sort_values(ascending=False).reset_in



#Visualizing time series of the top 9 mortal pathologies
#9 instead of 10 due to color scheme and overcrowding, 8 might be better
#Potential decline in the top 3-9, but 1 and 2 interestingly appear to increase (wo
fig = go.Figure()
fig.add_trace(go.Scatter(x = TS_CV['Year'],
                         y = TS_CV['Cardiovascular Diseases'],
                         mode = 'lines',
                         name = 'Cardiovascular Diseases',
                         marker_color = 'Crimson',
                         line = dict(dash = 'solid')))

fig.add_trace(go.Scatter(x = TS_Neo['Year'],
                         y = TS_Neo['Neoplasms'],
                         mode = 'lines',
                         name = 'Neoplasms',
                         marker_color = 'White',
                         line = dict(dash = 'solid')))

fig.add_trace(go.Scatter(x = TS_CRD['Year'],
                         y = TS_CRD['Chronic Respiratory Diseases'],
                         mode = 'lines',
                         name = 'Chronic Respiratory Diseases',
                         marker_color = 'RoyalBlue',
                         line = dict(dash = 'solid')))

fig.add_trace(go.Scatter(x = TS_LLD['Year'],
                         y = TS_LLD["Lower Respiratory Infections"],
                         mode = 'lines',
                         name = "Lower Respiratory Infections",
                         marker_color = 'Magenta',
```

```
                          line = dict(dash = 'solid')))


fig.add_trace(go.Scatter(x = TS_Neonat['Year'],
                         y = TS_Neonat['Neonatal Disorders'],
                         mode = 'lines',
                         name = 'Neonatal Disorders',
                         marker_color = 'Orange',
                         line = dict(dash = 'solid')))

fig.add_trace(go.Scatter(x = TS_Diarrhea['Year'],
                         y = TS_Diarrhea['Diarrheal Diseases'],
                         mode = 'lines',
                         name = 'Diarrheal Diseases',
                         marker_color = 'Lime',
                         line = dict(dash = 'solid')))

fig.add_trace(go.Scatter(x = TS_DD['Year'],
                         y = TS_DD['Digestive Diseases'],
                         mode = 'lines',
                         name = 'Digestive Diseases',
                         marker_color = 'Gold',
                         line = dict(dash = 'solid')))

fig.add_trace(go.Scatter(x = TS_TB['Year'],
                         y = TS_TB['Tuberculosis'],
                         mode = 'lines',
                         name = 'Tuberculosis',
                         marker_color = 'LightSeaGreen',
                         line = dict(dash = 'solid')))

fig.add_trace(go.Scatter(x = TS_LD['Year'],
                         y = TS_LD['Cirrhosis and Other Chronic Liver Diseases'],
                         mode = 'lines',
                         name = 'Cirrhosis and Other Chronic Liver Diseases',
                         marker_color = 'Purple',
                         line = dict(dash = 'solid')))

fig.update_layout(title = '<b>Time Series of Top 9 Causes of Death on Earth<b>',
                  title_x = 0.5,
                  title_font= dict(size = 20),
                  xaxis_title = 'Year',
                  yaxis_title = 'Total Number of Deaths',
                  template = 'plotly_dark')
```

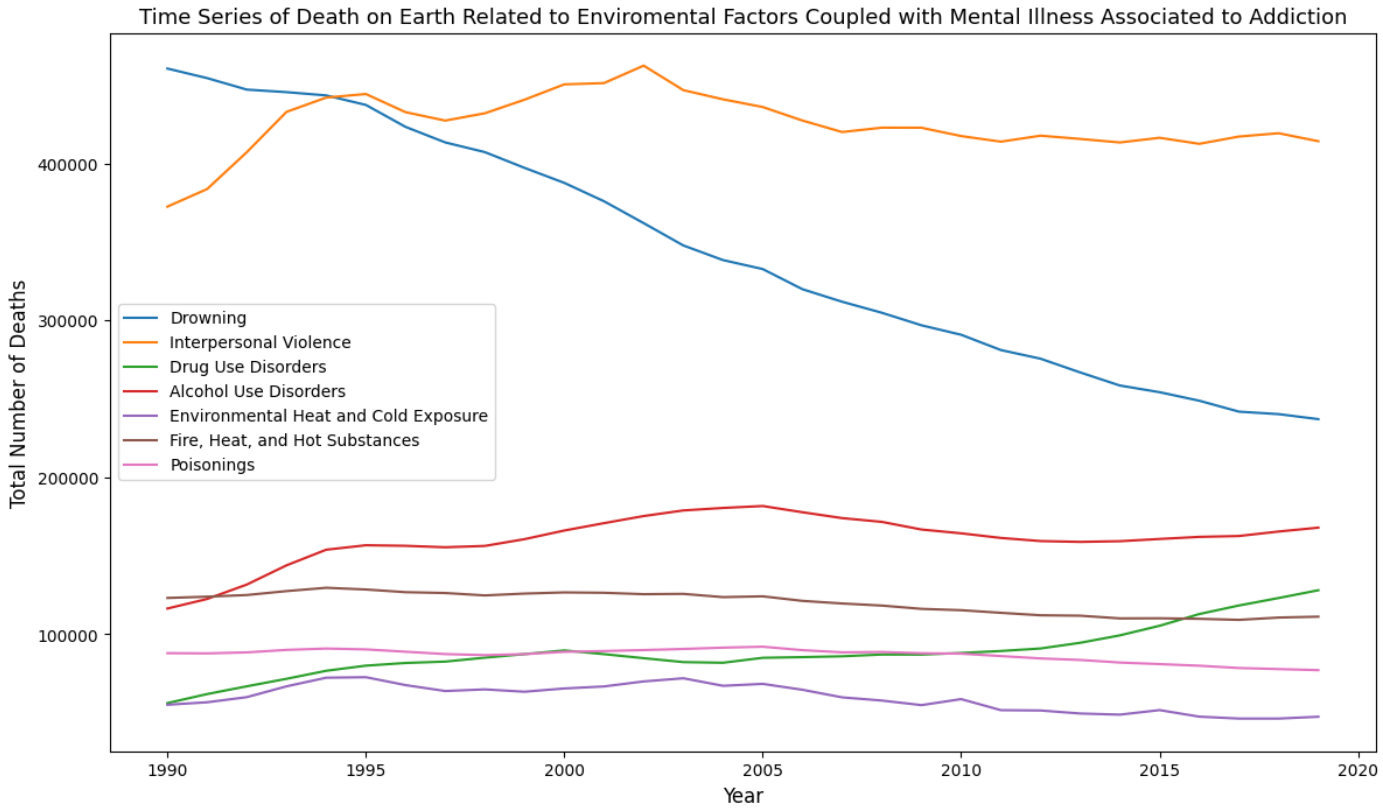
```
fig.show()
```




```
#Looking at some aspects of mental illness and other environmenatal factors that le
#Prep for timeseries
TS_Drown = df.groupby('Year')['Drowning'].sum().sort_values(ascending=False).reset_
TS_Violence = df.groupby('Year')['Interpersonal Violence'].sum().sort_values(ascend
TS_Drugs = df.groupby('Year')['Drug Use Disorders'].sum().sort_values(ascending=Fal
TS_Alcohol = df.groupby('Year')['Alcohol Use Disorders'].sum().sort_values(ascendin
TS_Envi = df.groupby('Year')['Environmental Heat and Cold Exposure'].sum().sort_val
TS_Fire = df.groupby('Year')['Fire, Heat, and Hot Substances'].sum().sort_values(as
TS_Poison = df.groupby('Year')['Poisonings'].sum().sort_values(ascending=False).res
```

```
#Death by environmental factors and addiction (EFaA)
#Timeseries visualization
```

```
#Drowning appears to be on the decline (What is the root cause? A feature that can
#Death from drug use appears to be on the rise (Why?)
EFaA_death = [TS_Drown, TS_Violence, TS_Drugs, TS_Alcohol, TS_Envi, TS_Fire, TS_Poi

plt.figure(figsize=(14,8))

for cause_df in EFaA_death:
    sns.lineplot(data = cause_df,
                 x = 'Year',
                 y = cause_df.columns[1],
                 label = cause_df.columns[1]
                )

plt.xlabel('Year',fontsize =12)
plt.ylabel('Total Number of Deaths',fontsize =12)
plt.title('Time Series of Death on Earth Related to Enviromental Factors Coupled wi
plt.legend()
plt.show()
```




```
countries_top_death_cause = df.drop(columns=['Country/Territory', 'Code', 'Year', '
countries_top_horseperson = countries_top_death_cause.idxmax(axis=1)
df['Top Cause'] = countries_top_horseperson
```

```
#Looking for trends
fig = px.choropleth(df,
              locations="Code",
              color="Top Cause",
              color_continuous_scale='Plasma',
              hover_name="Country/Territory",
              animation_frame="Year",
              width=800
)
fig.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
fig.show()
```