Cause Of Deaths Report

Submitted by: Md Saleem Patel

Content :-

In this Dataset, we have Historical Data of different cause of deaths for all ages around the World. The key features of this Dataset are: Meningitis, Alzheimer's Disease and Other

Dementias, Parkinson's Disease, Nutritional Deficiencies,

Malaria, Drowning, Interpersonal Violence, Maternal Disorders,

HIV/AIDS, Drug Use Disorders, Tuberculosis, Cardiovascular

Diseases, Lower Respiratory Infections, Neonatal Disorders,

Alcohol Use Disorders, Self-harm, Exposure to Forces of

Nature, Diarrheal Diseases, Environmental Heat and Cold

Exposure, Neoplasms, Conflict and Terrorism, Diabetes

Mellitus, Chronic Kidney Disease, Poisonings, Protein-Energy

Malnutrition, Road Injuries, Chronic Respiratory Diseases, Cirrhosis and Other Chronic Liver Diseases, Digestive Diseases, Fire, Heat, and Hot Substances, Acute Hepatitis.

IMPORTING

IMPORTANT LIBARARIES:-

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

```
import plotly.express as px
import plotly.offline as pyo
import plotly.graph_objects as go
from plotly.subplots import make_subplots
```

IMPORTING THE DATASET WITH DISPLAY

MAX COLUMNS AS THERE ARE 34 COLUMNS IN THE DATASET :-

```
df=pd.read_csv('cause_of_deaths.csv')
pd.set_option("display.max_columns",None)
df
```

	Country/Territory	Code	Year	Meningitis	Alzheimer's Disease and Other Dementias	Parkinson's Disease	Nutritional Deficiencies	Malaria	Drowning	Interpersonal Violence	Maternal Disorders	HIV/AIDS	Drug Use Disorders	Tuber
0	Afghanistan	AFG	1990	2159	1116	371	2087	93	1370	1538	2655	34	93	
1	Afghanistan	AFG	1991	2218	1136	374	2153	189	1391	2001	2885	41	102	
2	Afghanistan	AFG	1992	2475	1162	378	2441	239	1514	2299	3315	48	118	
3	Afghanistan	AFG	1993	2812	1187	384	2837	108	1687	2589	3671	56	132	
4	Afghanistan	AFG	1994	3027	1211	391	3081	211	1809	2849	3863	63	142	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
6115	Zimbabwe	ZWE	2015	1439	754	215	3019	2518	770	1302	1355	29162	104	
6116	Zimbabwe	ZWE	2016	1457	767	219	3056	2050	801	1342	1338	27141	110	
6117	Zimbabwe	ZWE	2017	1460	781	223	2990	2116	818	1363	1312	24846	115	
6118	Zimbabwe	ZWE	2018	1450	795	227	2918	2088	825	1396	1294	22106	121	
6119	Zimbabwe	ZWE	2019	1450	812	232	2884	2068	827	1434	1294	20722	127	

6120 rows × 34 columns

DOING SOME SHUFFLING OF THE DATASET TO SEE ANY ABNORMAL VALUES PRESENT IN THE DATASET :-

```
df.tail()
```

	Country/Territory	Code	Year	Meningitis	Alzheimer's Disease and Other Dementias	Parkinson's Disease	Nutritional Deficiencies	Malaria	Drowning	Interpersonal Violence	Maternal Disorders	HIV/AIDS	Drug Use Disorders	Tuber
6115	Zimbabwe	ZWE	2015	1439	754	215	3019	2518	770	1302	1355	29162	104	
6116	Zimbabwe	ZWE	2016	1457	767	219	3056	2050	801	1342	1338	27141	110	
6117	Zimbabwe	ZWE	2017	1460	781	223	2990	2116	818	1363	1312	24846	115	
6118	Zimbabwe	ZWE	2018	1450	795	227	2918	2088	825	1396	1294	22106	121	
6119	Zimbabwe	ZWE	2019	1450	812	232	2884	2068	827	1434	1294	20722	127	

```
df.sample(5)
```

	Country/Territory	Code	Year	Meningitis	Alzheimer's Disease and Other Dementias	Parkinson's Disease	Nutritional Deficiencies	Malaria	Drowning	Interpersonal Violence	Maternal Disorders	HIV/AIDS	Drug Use Disorders	Tuber
1275	Costa Rica	CRI	2005	43	660	120	17	0	137	299	26	150	12	
2396	Hungary	HUN	2016	53	4538	891	34	0	137	143	11	32	51	
1573	Dominican Republic	DOM	2003	299	1148	231	441	16	252	1242	196	3686	15	
3784	New Zealand	NZL	1994	28	919	188	11	0	67	70	7	54	32	
4482	Russia	RUS	2002	2380	27337	7642	635	0	18248	56916	533	9450	9596	

CHECKING OUT THE DATA TYPES OF THE COLUMNS IN THE DATASET :-

```
# Now lets identify which types of data types do they all belongs

df.dtypes
```

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

Dataset contains both categorical columns and numerical columns.. There are only 2 numerical columns in whole dataset

HERE WE CAN SEE THAT THERE ARE

2 OBJECT COLUMNS AND REST ALL THE OTHER COLUMNS ARE NUMERICAL COLUMNS.

LETS CHECK THE INFO OF THE DATASET AND HERE WE GET TO KNOW ABOUT THE DATA TYPE AND COUNTS OF THE COLUMN:-

```
: 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
```

This tell us about columns name null value dtypes of columns and memory usage.. count of every column are equal which means there are no nan present in dataset..it tell dtype of every column and tere are two data type in dataset int64, object where 32 columns are int64 where as 2 column are object..

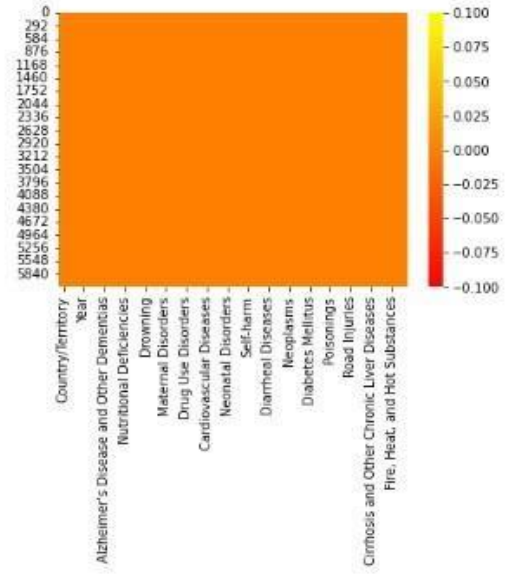
LETS CHECK NULL VALUES IN THE DATASET:-

```
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
dtype: int64
```

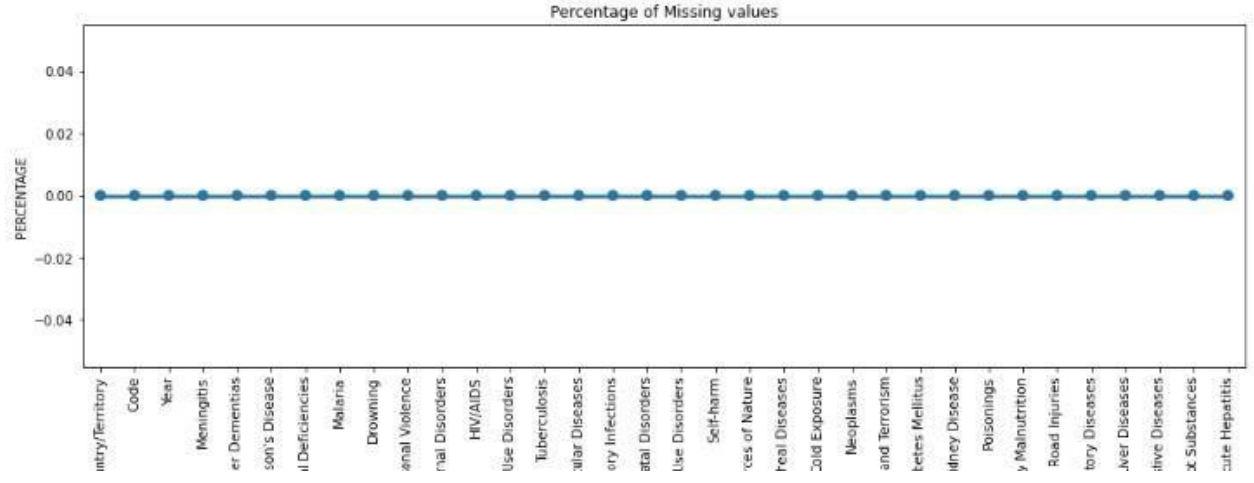
Count of nan is 0 in every column

```
# Let's visualize NaN values
sns.heatmap(df.isnull(),cmap="autumn")
plt.show()
```



dataset is free from nan value

```
missing = pd.DataFrame((df.isnull().sum())*100/df.shape[0]).reset_index()
plt.figure(figsize=(16,5))
ax = sns.pointplot('index',0,data=missing)
plt.xticks(rotation =90,fontsize =11)
plt.title("Percentage of Missing values")
plt.ylabel("PERCENTAGE")
plt.show()
```



here we can see that 0 nan values are present in the dataset.

separating categorical and numerical columns from the dataset :-

Separating numerical and categorcal columns

```
# Checking for categorical columns
categorical_col=[]
for i in df.dtypes.index:
    if df.dtypes[i]=='object':
        categorical_col.append(i)
print("Categorical columns are:\n",categorical_col)
```

```
Categorical columns are:
 ['Country/Territory', 'Code']
```

These two columns are only categorical in dataset

```
# Now checking for numerical columns
numerical_col=[]
for i in df.dtypes.index:
    if df.dtypes[i]!='object':
        numerical_col.append(i)
print("Numerical columns are:\n",numerical_col)
```

```
Numerical columns are:
 ['Year', 'Meningitis', "Alzheimer's Disease and Other Dementias", "Parkinson's Disease", 'Nutritional Deficiencies', 'Malari
a', 'Drowning', 'Interpersonal Violence', 'Maternal Disorders', 'HIV/AIDS', 'Drug Use Disorders', 'Tuberculosis', 'Cardiovascul
ar Diseases', 'Lower Respiratory Infections', 'Neonatal Disorders', 'Alcohol Use Disorders', 'Self-harm', 'Exposure to Forces o
f Nature', 'Diarrheal Diseases', 'Environmental Heat and Cold Exposure', 'Neoplasms', 'Conflict and Terrorism', 'Diabetes Melli
tus', 'Chronic Kidney Disease', 'Poisonings', 'Protein-Energy Malnutrition', 'Road Injuries', 'Chronic Respiratory Diseases',
'Cirrhosis and Other Chronic Liver Diseases', 'Digestive Diseases', 'Fire, Heat, and Hot Substances', 'Acute Hepatitis']
```

These are numerical column of dataset

Here we have described the whole dataset by DESCRIBE command .

We can see the count of all the columns that is 6120 which means no Null value is present in the dataset.

We can see the mean and standard deviation of all the Numeric columns in the dataset.

We can see the Min and Max from all the columns.

We can see Quartiles over here too

VISUALIZATIONS:-

NOW LET’S DIVIDE ALL THE

FACTORS OF DEATH INTO 4 CATAGORIES :-

```
df.columns
```

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

```
deathsBy_Disease = df[["Country/Territory",
                       "Year",
                       "Meningitis",
                       "Alzheimer's Disease and Other Dementias",
                       "Parkinson's Disease",
                       "Digestive Diseases",
                       "Malaria",
                       "Tuberculosis",
                       "Diabetes Mellitus",
                       "HIV/AIDS",
                       "Acute Hepatitis",
                      "Parkinson's Disease",
                     "Nutritional Deficiencies",
                    "Cardiovascular Diseases",
                    "Neoplasms","Neonatal Disorders","Maternal Disorders",
                    "Diarrheal Diseases"]]

deathsBy_Environment_And_Accidental = df[["Country/Territory",
                                  "Year",
                                  "Environmental Heat and Cold Exposure",
                                  "Drowning",
                                  "Road Injuries",
                                  "Exposure to Forces of Nature",
                                  "Protein-Energy Malnutrition"]]

deathsBy_Crimes_Terror_Accident_SelfHarm = df[["Country/Territory",
                                       "Year",
                                       "Interpersonal Violence",
                                       "Drug Use Disorders",
                                       "Alcohol Use Disorders",
                                       "Self-harm",
                                       "Conflict and Terrorism",
                                       "Poisonings"]]

deathsBy_Chronic_Disases = df[["Country/Territory",
                           "Year",
                           "Chronic Kidney Disease",
                           "Chronic Respiratory Diseases",
                           "Cirrhosis and Other Chronic Liver Diseases","Lower Respiratory Infections"]]
```

THESE 4 CATAGORIES ARE :-

DEATH BY DISEASES

DEATH BY ENVIORNMENT AND ACCIDENT.

DEATH BY CRIME ,TERROR,SELF-HARM AND ACCIDENT.

DEATH BY CRONIC DISEASES.

NOW DO THE ANALYSIS AS

PER THE DEATH BY

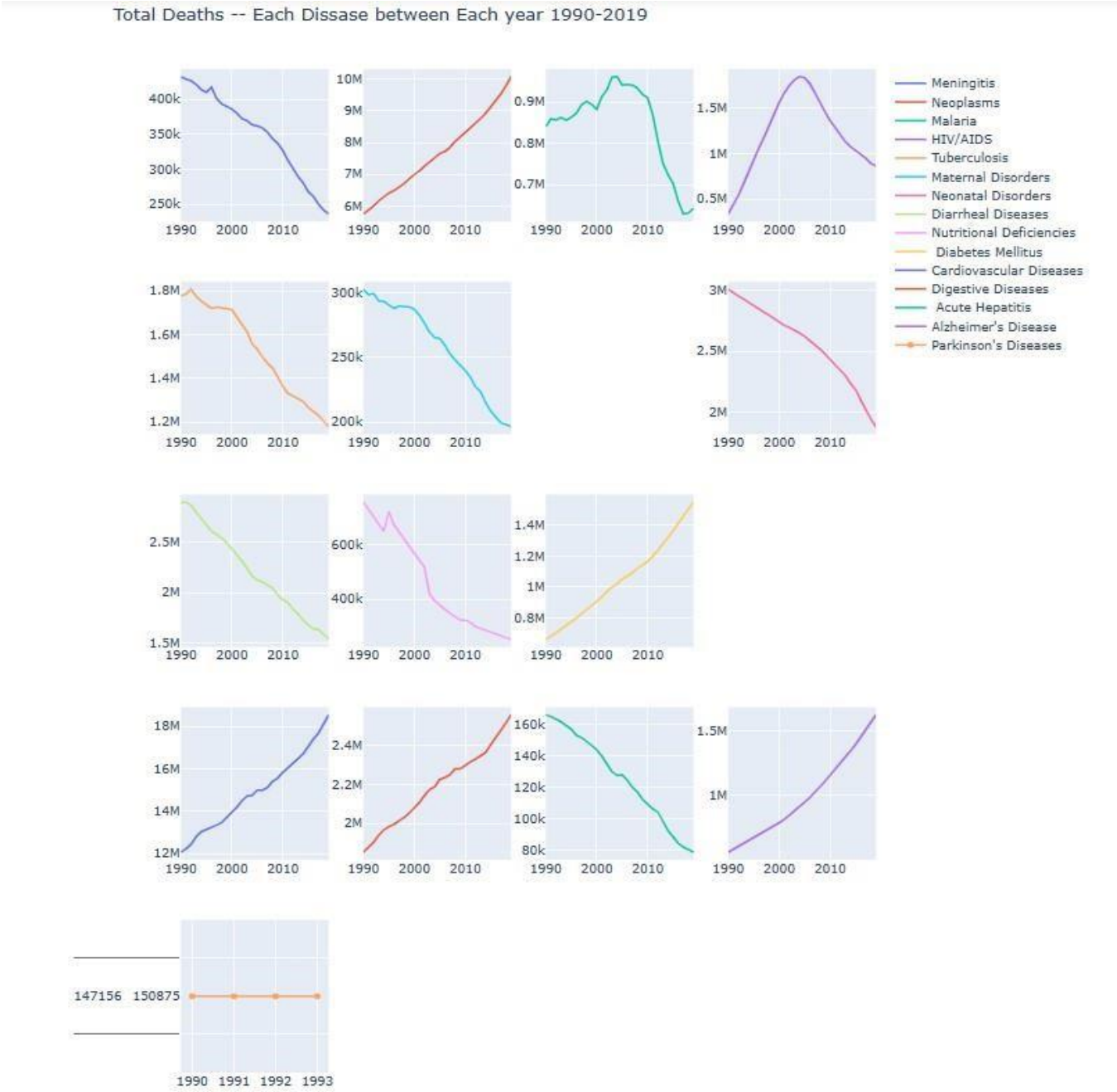
DISEASES :-

HERE I HAVE DONE GROUPING OF YEAR AND COUNTRIES ON THE BASIS OF DISEASES .

```
groupingByYear = deathsBy_Disease.groupby(['Year'])[[
                          "Meningitis",
                          "Alzheimer's Disease and Other Dementias",
                          "Parkinson's Disease",
                          "Digestive Diseases",
                          "Malaria",
                          "Tuberculosis",
                          "Diabetes Mellitus",
                          "HIV/AIDS",
                          "Acute Hepatitis",
                         "Parkinson's Disease",
                        "Nutritional Deficiencies",
                       "Cardiovascular Diseases",
                       "Neoplasms",
                       "Neonatal Disorders",
                       "Maternal Disorders",
                       "Diarrheal Diseases",]].sum().reset_index()

groupingByCountries = deathsBy_Disease.groupby(['Country/Territory'])[[
                          "Meningitis",
                          "Alzheimer's Disease and Other Dementias",
                          "Parkinson's Disease",
                          "Digestive Diseases",
                          "Malaria",
                          "Tuberculosis",
                          "Diabetes Mellitus",
                          "HIV/AIDS",
                          "Acute Hepatitis",
                         "Parkinson's Disease",
                        "Nutritional Deficiencies",
                       "Cardiovascular Diseases",
                       "Neoplasms",
                       "Neonatal Disorders",
                       "Maternal Disorders",
                       "Diarrheal Diseases",]].sum().reset_index()
```

NOW DO PLOTTING OF DEATH BY DISEASES:-

```
fig = make_subplots(rows=5, cols=4)

fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Meningitis'], name = 'Meningitis'),row=1, col=1)
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Neoplasms'], name = 'Neoplasms'),row=1, col=2)
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Malaria'],name='Malaria'),row=1, col=3)
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['HIV/AIDS'],name='HIV/AIDS'),row=1, col=4)
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Tuberculosis'],name='Tuberculosis'),row=2, col=1)
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Maternal Disorders'],name='Maternal Disorders'),row=2, col=2
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Neonatal Disorders'],name='Neonatal Disorders'),row=2, col=4
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Diarrheal Diseases'],name='Diarrheal Diseases'),row=3, col=1
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Nutritional Deficiencies'],name='Nutritional Deficiencies'),
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Diabetes Mellitus'],name=' Diabetes Mellitus'),row=3, col=3)
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Cardiovascular Diseases'],name='Cardiovascular Diseases'),ro
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Digestive Diseases'],name='Digestive Diseases'),row=4, col=2
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear['Acute Hepatitis'],name=' Acute Hepatitis'),row=4, col=3)
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear["Alzheimer's Disease and Other Dementias"],name="Alzheimer's
fig.add_trace(go.Scatter(x=groupingByYear['Year'], y=groupingByYear["Parkinson's Disease"],name="Parkinson's Diseases"),row=5, co


fig.update_layout(height=1200, width=1000, title_text="Total Deaths -- Each Dissase between Each year 1990-2019")
fig.show()
```



This is Plot shows how much Death has taken places by Diseases In all the year since 19902019.

NOW LET’S SEE DEATHS TAKEN PLACE BY ENVIORNMENT AND ACCIDENT.

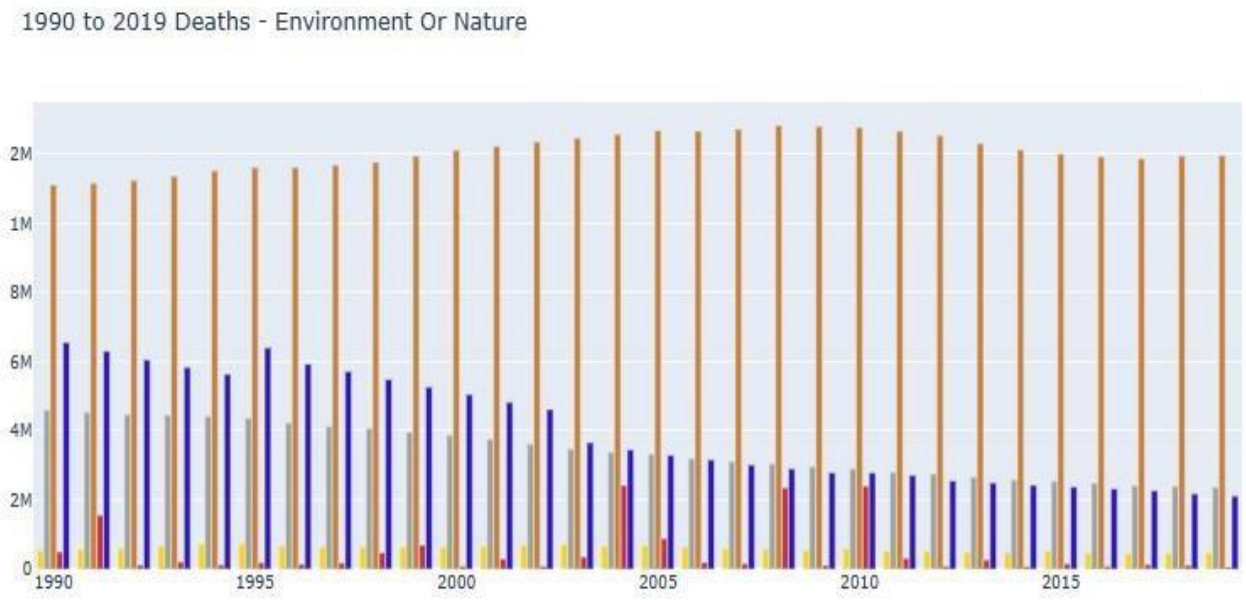
I HAVE DONE GROUPBY OF ALL

THE DEATH ACCORDING TO YEAR WHICH FALLS UNDER THIS CATEGORY.

```
deathsBy_Environment_And_Nature_group_Year = deathsBy_Environment_And_Accidental.groupby('Year')[[
                        "Environmental Heat and Cold Exposure",
                                "Drowning",
                                "Road Injuries",
                                "Exposure to Forces of Nature",
                                "Protein-Energy Malnutrition"]].sum().reset_index()
deathsBy_Environment_And_Nature_group_Year.head()
```

	Year	Environmental Heat and Cold Exposure	Drowning	Road Injuries	Exposure to Forces of Nature	Protein-Energy Malnutrition
0	1990	55072	480460	1112770	50216	655975
1	1991	56858	454375	1117024	156552	631013
2	1992	59928	447056	1125566	12030	606015
3	1993	66812	445434	1137444	21389	583919
4	1994	72305	443350	1153642	12717	564046

```
trace1 = go.Bar(
    x=deathsBy_Environment_And_Nature_group_Year['Year'],
    y=deathsBy_Environment_And_Nature_group_Year['Environmental Heat and Cold Exposure'],
    name = 'Deaths - Enviornmental heat and cold exposure',
    marker=dict(color='#FFD700'))

trace2 = go.Bar(
    x=deathsBy_Environment_And_Nature_group_Year['Year'],
    y=deathsBy_Environment_And_Nature_group_Year['Drowning'],
    name='Deaths - Drowning',
    marker=dict(color='#9EA0A1'))

trace3 = go.Bar(
    x=deathsBy_Environment_And_Nature_group_Year['Year'],
    y=deathsBy_Environment_And_Nature_group_Year['Road Injuries'],
    name='Deaths - Road injuries ',
    marker=dict(color='#CD7F32'))

trace4 = go.Bar(
    x=deathsBy_Environment_And_Nature_group_Year['Year'],
    y=deathsBy_Environment_And_Nature_group_Year['Exposure to Forces of Nature'],
    name='Exposure to forces of nature',
    marker=dict(color='#CD2F32'))

trace5 = go.Bar(
    x=deathsBy_Environment_And_Nature_group_Year['Year'],
    y=deathsBy_Environment_And_Nature_group_Year['Protein-Energy Malnutrition'],
    name='Deaths - PEM',
    marker=dict(color='#2f12cd'))

data = [trace1, trace2, trace3, trace4, trace5]
layout = go.Layout(
    title='1990 to 2019 Deaths - Environment Or Nature',height = 500, width=1400
)
fig = go.Figure(data=data, layout=layout)
fig.show()
```



Here following colour is representing following columns :-

Yellow:- Environmental Heat and Cold Exposure

Grey :- Deaths - Drowning

Orange :- Road Injuries

Blue :- Protein-Energy Malnutrition(PEM)

This plot shows the total number of deaths caused by Environment_And_Accidental in year 1990 TO 2019. Here we can notice the least and the max death that took place in all the 4 categories in all the given year.

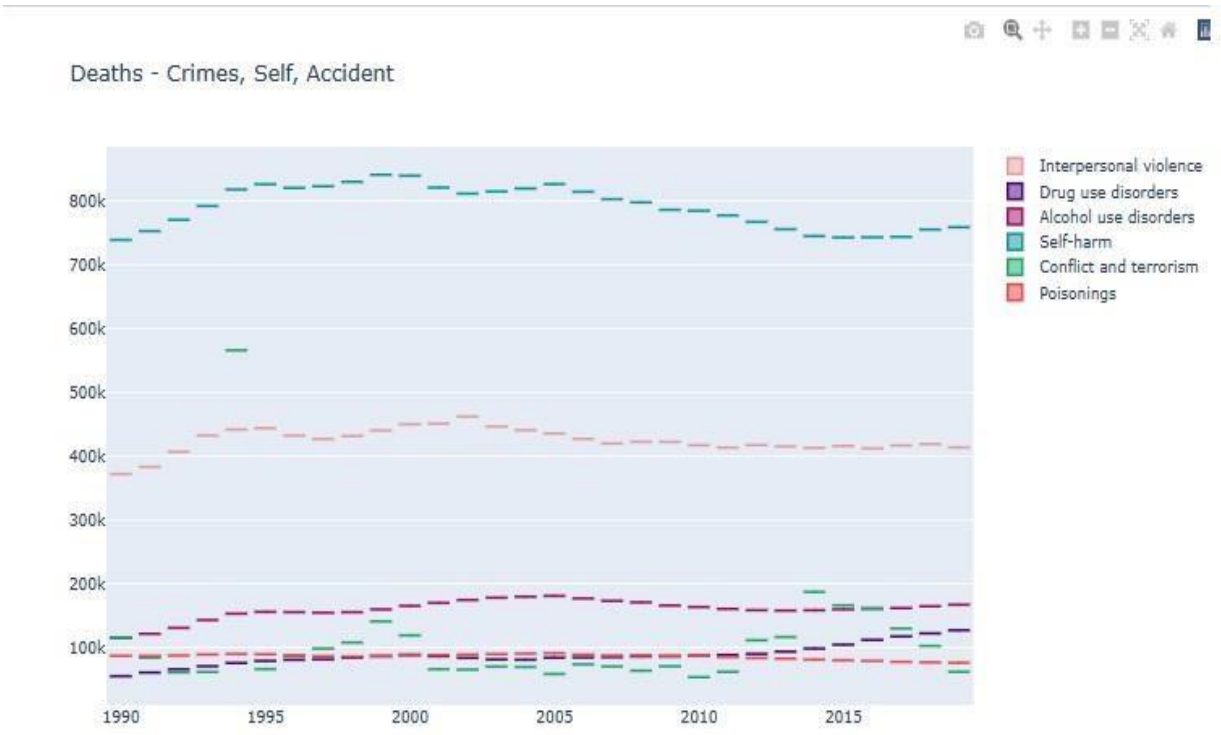
DEATH BY CRIME , TERROR ,SELF-HARM AND ACCIDENT.

```
groupingCrimesTerrorAccidentSelf = deathsBy_Crimes_Terror_Accident_SelfHarm.groupby('Year')[['Interpersonal Violence',
                                                                                          'Drug Use Disorders',
                                                                                          'Alcohol Use Disorders',
                                                                                          'Self-harm',
                                                                                          'Conflict and Terrorism',
                                                                                          'Poisonings'
                                                                                          ]].sum().reset_index()
```

```
groupingCrimesTerrorAccidentSelf.head()
```

	Year	Interpersonal Violence	Drug Use Disorders	Alcohol Use Disorders	Self-harm	Conflict and Terrorism	Poisonings
0	1990	372497	56133	116390	738804	116286	87951
1	1991	383689	61890	122478	752575	85017	87813
2	1992	407176	66826	131665	770286	62063	88435
3	1993	432858	71603	143901	791904	62733	90036
4	1994	441971	76717	153859	817682	566082	90897

```
fig = go.Figure()

fig.add_trace(go.Violin(x= groupingCrimesTerrorAccidentSelf['Year'] ,
                        y= groupingCrimesTerrorAccidentSelf['Interpersonal Violence'],
                        name='Interpersonal violence',
                        line_color='#ea9999'))

fig.add_trace(go.Violin(x= groupingCrimesTerrorAccidentSelf['Year'] ,
                        y= groupingCrimesTerrorAccidentSelf['Drug Use Disorders'],
                        name='Drug use disorders',
                        line_color='#48007c'))

fig.add_trace(go.Violin(x= groupingCrimesTerrorAccidentSelf['Year'] ,
                        y= groupingCrimesTerrorAccidentSelf['Alcohol Use Disorders'],
                        name='Alcohol use disorders',
                        line_color='#a60661'))

fig.add_trace(go.Violin(x= groupingCrimesTerrorAccidentSelf['Year'] ,
                        y= groupingCrimesTerrorAccidentSelf['Self-harm'],
                        name='Self-harm ',
                        line_color='#009999'))

fig.add_trace(go.Violin(x= groupingCrimesTerrorAccidentSelf['Year'] ,
                        y= groupingCrimesTerrorAccidentSelf['Conflict and Terrorism'],
                        name='Conflict and terrorism',
                        line_color='#15a962'))

fig.add_trace(go.Violin(x= groupingCrimesTerrorAccidentSelf['Year'] ,
                        y= groupingCrimesTerrorAccidentSelf['Poisonings'],
                        name='Poisonings',
                        line_color='#ff4040'))



fig.update_traces(meanline_visible=True)
fig.update_layout(title_text='Deaths - Crimes, Self, Accident',violingap=0, violinmode='overlay',height=600,width=1000)
fig.show()
```



We can clearly see in this plot which shows

CRIMES_TERROR_ACCIDENT_SELF-HARM and here

come to know that in all the years the maximum death have been taken place by Conflicts and Terrorism and the max death was in between 1990 and 2000.

Poisoning seems to be constant in all the years.

The second highest death has taken place by Interpersonal violence

And rest all the case of seems to be under 200k in all the given years.

DEATH BY CHRONIC DISEASES

:-

Now do grouping of chronic diseases as per year and relevant diseases.

DEATH BY CRONIC DISEASES

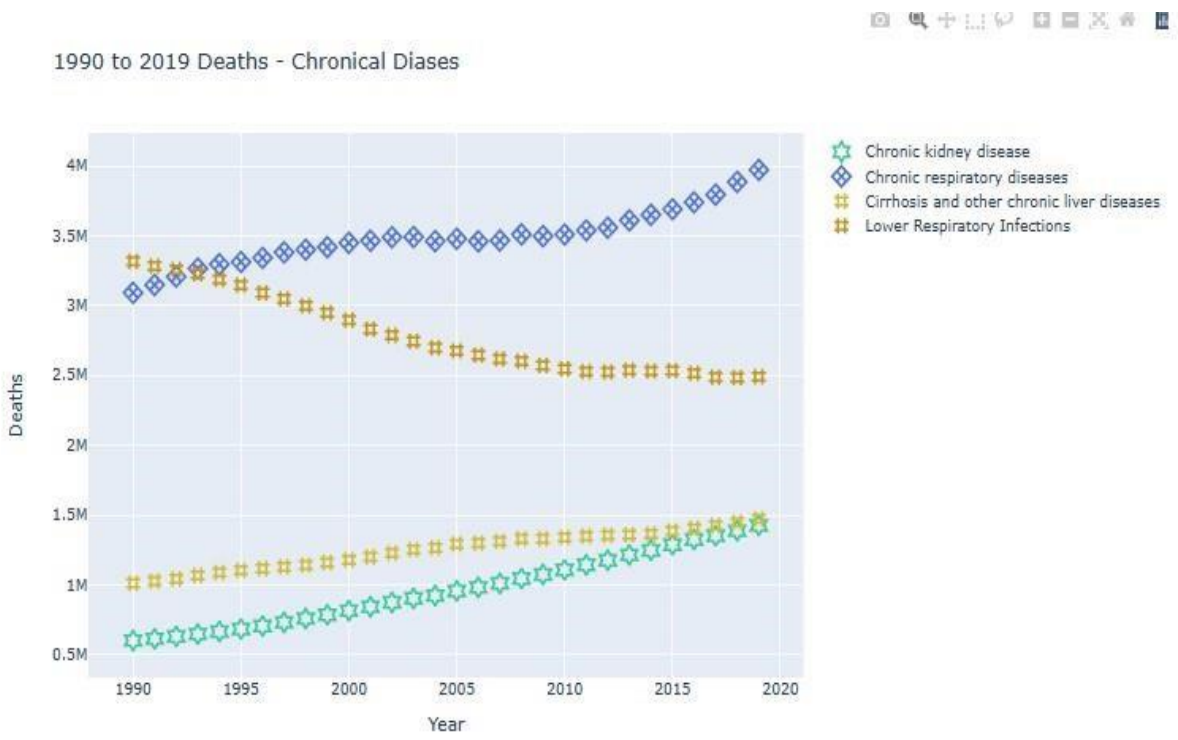
In [40]:
```
chronic_Deaths_GroupingByYear = deathsBy_Chronic_Disases.groupby('Year')[['Chronic Kidney Disease',
                                        'Chronic Respiratory Diseases',
                                        'Cirrhosis and Other Chronic Liver Diseases','Lower Respiratory Infections']].sum().re
```

In [41]:
```
chronic_Deaths_GroupingByYear.head()
```

Out[41]:

	Year	Chronic Kidney Disease	Chronic Respiratory Diseases	Cirrhosis and Other Chronic Liver Diseases	Lower Respiratory Infections
0	1990	600825	3092759	1012423	3318264
1	1991	613589	3148288	1026870	3282941
2	1992	630160	3207816	1042953	3258792
3	1993	647255	3266812	1057730	3228972
4	1994	665365	3297292	1089331	3187285

```
trace0 = go.Scatter(
    x = chronic_Deaths_GroupingByYear['Year'],
    y = chronic_Deaths_GroupingByYear['Chronic Kidney Disease'],
    name = 'Chronic kidney disease',
    mode = 'markers',
    marker = dict(
        size = 12,
        color = 'rgb(51,204,153)',
        symbol = 'hexagram-open',
        line = dict(width = 2)))

trace1 = go.Scatter(
    x = chronic_Deaths_GroupingByYear['Year'],
    y = chronic_Deaths_GroupingByYear['Chronic Respiratory Diseases'],
    name = 'Chronic respiratory diseases',
    mode = 'markers',
    marker = dict(
        size = 12,
        color = 'rgb(77,113,222)',
        symbol = 'diamond-x-open',
        line = dict(width = 2)))

trace2 = go.Scatter(
    x = chronic_Deaths_GroupingByYear['Year'],
    y = chronic_Deaths_GroupingByYear['Cirrhosis and Other Chronic Liver Diseases'],
    name = 'Cirrhosis and other chronic liver diseases',
    mode = 'markers',
    marker = dict(
        size = 12,
        color = 'rgb(211,188,53)',
        symbol = 'hash-open',
        line = dict(width = 2)))

trace3 = go.Scatter(
    x = chronic_Deaths_GroupingByYear['Year'],
    y = chronic_Deaths_GroupingByYear['Lower Respiratory Infections'],
    name = 'Lower Respiratory Infections',
    mode = 'markers',
    marker = dict(
        size = 12,
        color = 'rgb(200,150,20)',
        symbol = 'hash-open',
        line = dict(width = 2)))

data = [trace0,trace1,trace2,trace3]
layout = go.Layout(
    title = '1990 to 2019 Deaths - Chronical Diases',
    xaxis = dict(title = 'Year'),
    yaxis = dict(title = 'Deaths'),
    hovermode ='closest',
    height =600,
    width = 1000
)
fig = go.Figure(data=data, layout=layout)
fig.show()
```



WE CAN SEE THAT THE MAXIMUM DEATH IS CAUSED

BY CHRONIC RESPIRATORY DISEASES AND LEAST DEATH IS CAUSED BY CHRONIC KIDNEY DISEASES IN ALL THE GIVEN YEARS WHICH IS 1990 TO 2019.

CONCLUSION:

Total rows 6120 and 34 columns in the dataset.

I found out that there are many diseases which continuously increasing such as Neoplasms, HIV/AIDS, Diabetes, Cardiovascular Diseases, Digestive disorder and Alzheimer

I Found out that there are many diseases which are continuously decreasing too such as Acute Hepatitis, Diarrheal Diseases, Nutritional Diseases and Meningitis

Parkinson Diseases seems to be constants till 1990 to 1993 after that no data is present for the same.

We can see that in all the given years i.e. 1990 to 2019, Road accident have taken Maximum life's and the least can death can be seen in Exposure to force of Nature

In case of Death by crime, self-harm and Accident ->

Maximum death have been taken place by Conflict and Terrorism and the second highest death have been recorded by -Interpersonal Violence.

Rest all other factors of death are under 200k which can be even further minimized

ALL THE GOVERNMENT AND CONCERNED BODIES

SHOULD TAKE RESONABLE STEP TO ENSURE THAT

ALL THE AREAS WITH MAXIMUM DEATHS CAN BE

MINIMIZED AND PROPER ACTION SHOULD BE TAKEN

IN CASE OF CONFLICT &TERRIOSM AND

INTERPERSONAL VIOLENCE SO THAT IT SHOULD BE REDUCED TO MINIMAL.