In [1]:
```
Group_2 = "Assignment_1 : Data Management"
```

I. INTRODUCTION

This report presents an analysis on mortality data and death rates of people living in the United States. The data provides information on mortality patterns that have been observed year-by-year based on variables such as the cause of death, geographic location, etc. It is based on information from all resident death reports filed in the 50 U.S. states and the District of Columbia.

The scope of this analysis does not extend to factors such as gender, age, race, or nationality.

For purpose of analysis, we've made use of two public datasets—Leading Causes of Death in the United States From 1999 to 2016 (obtained from NCHS) and Annual Estimates of the Resident Population for the United States, Regions, States, and Puerto Rico (2010-2018). Data has been analysed using Pandas, a software library written for the Python programming language for data manipulation and analysis. In order to gain an understanding of available data, a tabular format has been adopted wherever required. In addition, the data has been visualised using Python plotting libraries such as Matplotlib and Seaborn.

Import Relevant Libraries and Retrieve Data

We start by importing the data-analytics libraries that are relevant to our work.

In [2]:
```
# Import pandas
import pandas as pd

# Import NumPy
import numpy as np

# Import Matplotlib for statistical visualizations
import matplotlib.pyplot as plt

# Import Seaborn to create statistical visualizations
import seaborn as sns

# Import Tabulate to print tabular data in well-formatted tables
from tabulate import tabulate
```

In [12]:
```
# Read the first dataset (a csv file) into data frame.
data1 = pd.read_csv('NCHS_-_Leading_Causes_of_Death__United_States.csv')
```

In [13]:
```
# Return the first n rows for the object based on position.
data1.head()
```

	Year	113 Cause Name	Cause Name	State	Deaths	Age-adjusted Death Rate
0	2012	Nephritis, nephrotic syndrome and nephrosis (N...	Kidney disease	Vermont	21	2.6
1	2016	Nephritis, nephrotic syndrome and nephrosis (N...	Kidney disease	Vermont	30	3.7
2	2013	Nephritis, nephrotic syndrome and nephrosis (N...	Kidney disease	Vermont	30	3.8
3	2000	Intentional self-harm (suicide) (*U03,X60-X84,...	Suicide	District of Columbia	23	3.8
4	2014	Nephritis, nephrotic syndrome and nephrosis (N...	Kidney disease	Arizona	325	4.1

In [14]:
```
# Read the second dataset (an excel file) into data frame.
data2 = pd.read_excel("nst-est2018-01.xlsx")
```

II. Description of Datasets Used in the Analysis

Dataset 1: NCHS – Leading Causes of Death in the United States (1999-2016)

This dataset gives an insight into the age-adjusted death rates for 10 (unique) leading causes of death in the United States from 1999 to 2016. The data is obtained from the U.S. government agency, NCHS and is based on information from all 50 U.S. states and the District of Columbia. Further, this national data gives an overview of the state-wise number of deaths and respective causes for various years. The dataset also identifies 'all causes' as one of the factors leading to death, along with 'U.S. states' (in its entirety) as a component of states.

Dataset 2: Annual Estimates of the Resident Population for the United States, Regions, States, and Puerto Rico (April 1, 2010 to July 1, 2018)

This dataset consists of annual population estimates for the residents of the United States based on their geographic location. The estimates are for the entire nation, followed by its census regions, states, and for Puerto Rico (data years 2010-2018). The data was released in December 2018 by the U.S. Census Bureau, Population Division. It is in the form of numerical figures.

In [15]:
```
# The describe() method here returns description of the data in the DataFrame.
data1.describe(include=object)
```

	113 Cause Name	Cause Name	State
count	10296	10296	10296
unique	11	11	52
top	Nephritis, nephrotic syndrome and nephrosis (N...	Kidney disease	Vermont
freq	936	936	198

In [16]:
```
data2.describe(include=object)
```

	table with row headers in column A and column headers in rows 3 through 4. (leading dots indicate sub-parts)	Unnamed: 1	Unnamed: 2	Unnamed: 3
count	64	59	58	59
unique	64	59	58	59
top	Table 1. Annual Estimates of the Resident Popu...	2010-04-01 00:00:00	Estimates Base	Population Estimate (as of July 1)
freq	1	1	1	1

III. Data Cleaning Techniques Used Prior Analysis

It is imperative that we deal with messy data to avoid false conclusions. This data could be in the form of missing values, inconsistent formatting, malformed records, or nonsensical outliers. The first dataset (data1) has been used as is.

In the next steps, we have implemented the following data cleaning processes on the second dataset (data2):

- Column names have been renamed to a more recognizable set of labels.
- Unnecessary columns have been dropped from the data frame.

1. Renaming Column Names in data2

- The first column has been renamed as 'Geography'.
- All the unnamed columns have been suitably labelled.

2. Drop Unnecessary Columns from data2

- The first three columns (indexes 0,1,2 respectively) consist of NaN values and irrelevant data that has been dropped.

In [17]:
```
# rename() function is used for all unnamed column names.
data2.rename({'table with row headers in column A and column headers in rows 3 through
             , axis=1, inplace=True)
```

```
data2.rename({
    'Unnamed: 1': 'Census', 'Unnamed: 2': 'Estimates Base', 'Unnamed: 3': '2010', 'Unn
    'Unnamed: 5': '2012',
    'Unnamed: 6': '2013', 'Unnamed: 7': '2014', 'Unnamed: 8': '2015', 'Unnamed: 9': '2
    'Unnamed: 11': '2018'}, axis=1, inplace=True)

# return the first 3 rows of the data frame.
data2.head(3)
```

	Geographic Region	Census	Estimates Base	2010	2011	2012	2013	2014	2015	2016	2017
0	Table 1. Annual Estimates of the Resident Popu...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	Geographic Area	2010-04-01 00:00:00	NaN	Population Estimate (as of July 1)	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	NaN	Census	Estimates Base	2010	2011.0	2012.0	2013.0	2014.0	2015.0	2016.0	2017.0

In [18]:
```
data3 = data2.drop(data2.index[[0, 1, 2]]) # drop() function is used to remove the fir

data3.reset_index(drop=True, inplace=True) # reset_index() method is used to reset the

data3.head()
```

	Geographic Region	Census	Estimates Base	2010	2011	2012	2013	201
0	United States	308745538	308758105	309326085	311580009.0	313874218.0	316057727.0	318386421.
1	Northeast	55317240	55318430	55380645	55600532.0	55776729.0	55907823.0	56015864.
2	Midwest	66927001	66929743	66974749	67152631.0	67336937.0	67564135.0	67752238.
3	South	114555744	114563045	114867066	116039399.0	117271075.0	118393244.0	119657737.
4	West	71945553	71946887	72103625	72787447.0	73489477.0	74192525.0	74960582.

IV. Data Analysis

We will be answering a few questions on this dataset based on our data analysis. These are:

1. Are Americans facing an increasing, decreasing, or steady likelihood of death?

2. What are the four leading causes of death for Americans?

3. Do individual states show the same four leading causes of death?

4. Are there year-by-year changes in the four leading causes of death nationwide?

1. Are Americans facing an increasing, decreasing, or steady likelihood of death?

Coding Algorithm:

1. Retrieve the first n rows of dataset-1: NCHS_Leading_Causes_of_Death_United_States.csv.
2. Locate the state 'United States' and cause name 'all causes' in the dataset, with the condition that the year is greater than/equal to 2010 because dataset-2 is for the years 2010-2018.
3. In dataset-2 (nst-est2018-01.xlsx), locate the geographic location 'United States' and store related data with population estimates in a new variable.
4. Drop the following columns from the new variable: 'Census', 'Estimate Base' and 'Geographic region'.
5. Take a transpose of the DataFrame and Reset the index, followed by Renaming it.
6. The two DataFrames are then merged based on 'Year'.
7. Take the ratio of values in 'Deaths' column with the values in 'Population' column.
8. Sort the dataframe by 'Average Death Rate' in ascending order.
9. Reset the index and display merged data in tabular form. 10.Plot the table in the form of a line graph.

In [19]:
```
# Locate state = 'United States' and cause name = 'all causes' in NHS dataset, with th
df = data1.loc[(data1['State'] == 'United States')& (data1['Cause Name'] == 'All cause
df
```

	Year	113 Cause Name	Cause Name	State	Deaths	Age-adjusted Death Rate
9585	2014	All Causes	All causes	United States	2626418	724.6
9601	2016	All Causes	All causes	United States	2744248	728.8
9616	2013	All Causes	All causes	United States	2596993	731.9
9620	2012	All Causes	All causes	United States	2543279	732.8
9622	2015	All Causes	All causes	United States	2712630	733.1
9641	2011	All Causes	All causes	United States	2515458	741.3
9664	2010	All Causes	All causes	United States	2468435	747.0

In [20]:
```
data4 = data3.loc[(data3['Geographic Region'] == 'United States')] # Locate 'United St

data4.head()
```

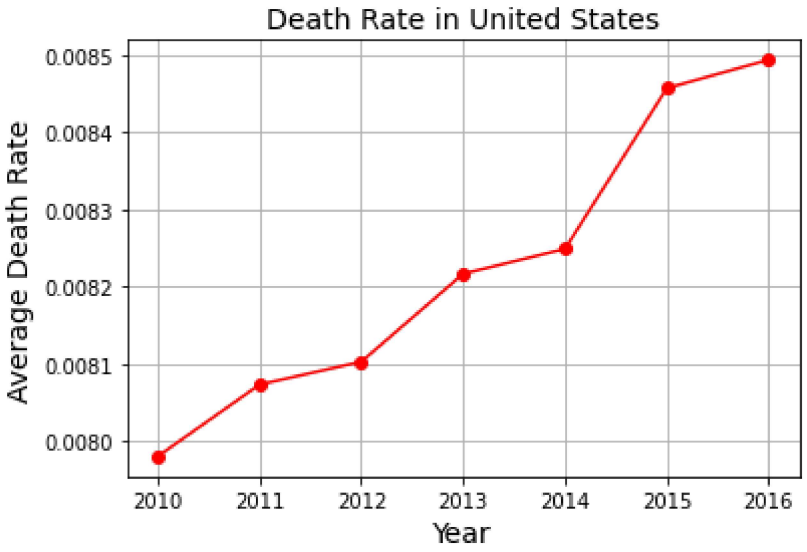
	Geographic Region	Census	Estimates Base	2010	2011	2012	2013	201
0	United States	308745538	308758105	309326085	311580009.0	313874218.0	316057727.0	318386421.

In [21]:
```
data4 = data4.drop(['Census', 'Estimates Base' ,'Geographic Region'], axis=1)#Dropping
data4 = data4.T #taking a transpose of the dataframe
data4 = data4.reset_index() #resetting index
data4.rename({ 'index' : 'Year' , 0 : 'Population'}, axis=1, inplace=True) #renaming i
data4['Year'] = data4['Year'].astype(str).astype(int) #changing 'Year' column data typ
data4['Population'] = data4['Population'].astype(str).astype(float) #changing 'Populat
mergedData = df.merge(data4, left_on='Year', right_on='Year') #merging two dataframes
mergedData['Average Death Rate'] =  mergedData['Deaths']/mergedData['Population'] #tak
mergedData = mergedData.sort_values('Average Death Rate') #sorting dataframe by 'Avera
mergedData = mergedData.reset_index(drop = True) #resetting index

mergedData #displaying mergedData
```

	Year	113 Cause Name	Cause Name	State	Deaths	Age-adjusted Death Rate	Population	Average Death Rate
0	2010	All Causes	All causes	United States	2468435	747.0	309326085.0	0.007980
1	2011	All Causes	All causes	United States	2515458	741.3	311580009.0	0.008073
2	2012	All Causes	All causes	United States	2543279	732.8	313874218.0	0.008103
3	2013	All Causes	All causes	United States	2596993	731.9	316057727.0	0.008217
4	2014	All Causes	All causes	United States	2626418	724.6	318386421.0	0.008249
5	2015	All Causes	All causes	United States	2712630	733.1	320742673.0	0.008457
6	2016	All Causes	All causes	United States	2744248	728.8	323071342.0	0.008494

In [22]:
```
plt.plot(mergedData['Year'], mergedData['Average Death Rate'], color='red', marker='o'
plt.title('Death Rate in United States', fontsize=14) #giving a title to the figure
plt.xlabel('Year', fontsize=14) #labelling x-axis
plt.ylabel('Average Death Rate', fontsize=14) #labelling y-axis
plt.grid(True) #giving grid lines to the graph
plt.show() #displaying the graph
```



Result:

The line graph illustrates an increasing trend in the average death rate between the years 2010 and 2016. A sharp rise is noticeable at some points, whereas there is a steady increase at others. For example, the time period between 2014-2015 displays a prominent rise in the number of deaths by 0.0002, or 0.02%.

Thus, it can be concluded that:

Americans are facing an increasing likelihood of death with each passing year.

2. What are the four leading causes of death for Americans?

Coding Algorithm:

1. We start by grouping 'Cause Name' column with 'Deaths' using the sum() function on 'Deaths' column.
2. The first row in the dataframe is dropped i.e 'All causes'.
3. Sort the data by 'no. of deaths' and display it in a descending order, for the first 10 leading causes.
4. Display the results in a tabular format and as a visual plot.

In [23]:
```
df= data1.groupby(["Cause Name"]).sum()["Deaths"].reset_index()#grouped 'Cause Name' c
df.head()
```

	Cause Name	Deaths
0	All causes	89830132
1	Alzheimer's disease	2746824
2	CLRD	4869452
3	Cancer	20489072
4	Diabetes	2632758

In [24]:
```
df= df.drop(df.index[0]) #dropped first row in the dataframe i.e 'All causes'
df.head()
```

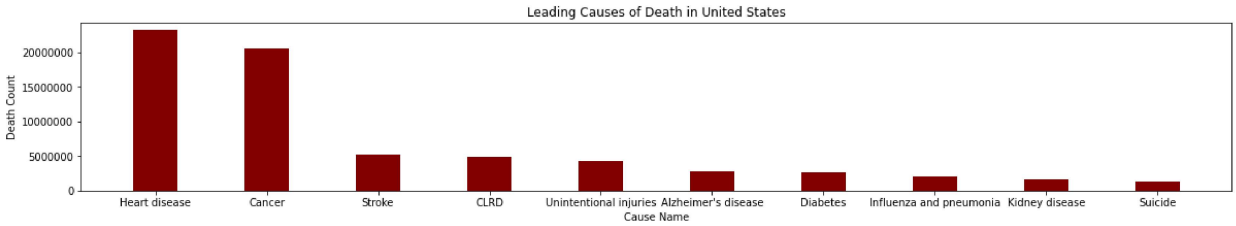
	Cause Name	Deaths
1	Alzheimer's disease	2746824
2	CLRD	4869452
3	Cancer	20489072
4	Diabetes	2632758
5	Heart disease	23150366

In [25]:
```
df = df.sort_values(by='Deaths', ascending=False).reset_index(drop=True)# Sorting the
df.head()
```

	Cause Name	Deaths
0	Heart disease	23150366
1	Cancer	20489072
2	Stroke	5160280
3	CLRD	4869452
4	Unintentional injuries	4355768

The table above displays the number of deaths for the leading causes of death in the United States. The same has been visualised below using a bar chart.

In [26]:
```
plt.rcParams["figure.figsize"] = (20,3) #setting figure size
plt.bar(df['Cause Name'], df['Deaths'], color ='maroon',width = 0.4) #plotting a bar c
plt.ticklabel_format(style='plain', axis='y') #removing scientific annotation from y-c
plt.xlabel("Cause Name") #Labeling x-axis
plt.ylabel("Death Count") #labelling y-axis
plt.title("Leading Causes of Death in United States") #giving heading to the figure
plt.show() #displaying plot
```



Result:

The bar graph shows the 10 leading causes of death in the United States, making use of 'death count' as reference. Heart disease tops the list with 23,150,366 cases, followed by cancer. Suicide is the least significant cause of death in the U.S. and about 1,299,686 cases have been reported.

We can conclude:

Heart disease, Cancer, Stroke, and CLRD are the four leading causes of death for Americans.

3. Do individual states show the same four leading causes of death?

Coding Algorithm:

1. First, we group the States and Cause Name by summing deaths.
2. Then, we sort deaths in descending order to get leading causes of deaths.
3. We then display 'New York' state's data from the dataframe, and remove all causes.
4. The top 4 causes are stored in a new variable.
5. The United States row is dropped from the state index.
6. Create an empty dictionary, an empty array and use 'for' loop to iterate 'cause name' of every 'state'.
7. A boolean comparison of one row with another consecutive row is performed to check whether every 'State' has the same top 4 causes or not.
8. The loop is run in a similar fashion for subsequent entries.

In [28]:
```
df2 = data1.groupby(['State','Cause Name'])['Deaths'].sum() #grouping the State, Cause
df2 = pd.DataFrame(df2) #storing as a dataframe
```

In [29]:
```
df2 = df2.sort_values('Deaths', ascending = False) #sorting deaths in descending order
```

In [30]:
```
df2 = df2.drop(df2.groupby(['State']).head(1).index) #removing all causes
df2 = df2.groupby(['State']).head(4) #storing top 4 causes of death
df2=df2.drop('United States', level=0, axis=0) #droped United States row from the stat
df2.head() # displaying some rows to get heads up of a dataframe
```

		Deaths
State	**Cause Name**	
California	**Heart disease**	1141776
	Cancer	1002719
New York	**Heart disease**	895080
Florida	**Heart disease**	816162
	Cancer	737552

In [31]:
```
l2 = df2.index.get_level_values(0).unique().tolist() #getting a list of desired values
l1 = {} # creating an empty dictionary
l5 = [] # creating an empty array
#using 'for' loop to iterate 'Cause name' of every 'State'
#doing a boolean comparison with one row with another consecutive row to check whether
for i in range(0,len(df2.index.get_level_values(0).unique().tolist())-1):
    l3=set(df2.loc[[l2[i]]].index.get_level_values(1).tolist()) #stores list of four c
    l4 =set(df2.loc[[l2[i+1]]].index.get_level_values(1).tolist()) #stores list of fou
    l5.append(l3==l4) #boolean comparison and appending to array
    l1[df2.index.get_level_values(0).unique().tolist()[i]] = df2.loc[[l2[i]]].index.ge
if 'False' not in l5: #checking is there any false in the list of boolean values
    print('Every States is not having the same four leading causes of death')
```

```
Every States is not having the same four leading causes of death
```

In [32]:
```
l6 = [k for k,v in l1.items() if sorted(v) == sorted(['Heart disease', 'Cancer', 'Strc
l7 = [k for k,v in l1.items() if sorted(v) == sorted(["Alzheimer's disease", 'Cancer',
l8 = [k for k,v in l1.items() if sorted(v) == sorted(['Heart disease', 'Cancer', 'Unir
l9 = [k for k,v in l1.items() if sorted(v) == sorted(['Heart disease', 'Cancer', 'Strc
dict2 = {} # creating an empty dictionary
for i in l8 : #for loop for getting key-value pairs of a dictionary using list of keys
    dict2[i] =l1[i] #storing ith key and its ith value of a dictionary in a new dictio
print(tabulate(dict2.items()))#printing formatted table of a dictionary
print("There are "+str(len(l8))+" States which have 'Heart disease', 'Cancer', 'Uninte
```

```
-------------  ------------------------------------------------------------
Arizona        ['Heart disease', 'Cancer', 'Unintentional injuries', 'CLRD']
Kentucky       ['Heart disease', 'Cancer', 'CLRD', 'Unintentional injuries']
Colorado       ['Cancer', 'Heart disease', 'Unintentional injuries', 'CLRD']
West Virginia  ['Heart disease', 'Cancer', 'CLRD', 'Unintentional injuries']
Nevada         ['Heart disease', 'Cancer', 'CLRD', 'Unintentional injuries']
New Mexico     ['Heart disease', 'Cancer', 'Unintentional injuries', 'CLRD']
Montana        ['Heart disease', 'Cancer', 'CLRD', 'Unintentional injuries']
Vermont        ['Cancer', 'Heart disease', 'CLRD', 'Unintentional injuries']
Wyoming        ['Heart disease', 'Cancer', 'CLRD', 'Unintentional injuries']
-------------  ------------------------------------------------------------
There are 9 States which have 'Heart disease', 'Cancer', 'Unintentional injuries' and
'CLRD' as top four leading causes.
```

In [33]:
```
dict1 = {} # creating an empty dictionary
for i in l9 : #for loop for getting key-value pairs of a dictionary using list of keys
    dict1[i] =l1[i] #storing ith key and its ith value of a dictionary in a new dictio
print(tabulate(dict1.items()))#printing formatted table of a dictionary
print("There are "+str(len(l9))+" States have top four causes of deaths as Heart disea
```

```
-------------  ------------------------------------------
California     ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
New York       ['Heart disease', 'Cancer', 'CLRD', 'Stroke']
Florida        ['Heart disease', 'Cancer', 'CLRD', 'Stroke']
Pennsylvania   ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Ohio           ['Heart disease', 'Cancer', 'CLRD', 'Stroke']
Illinois       ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Michigan       ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
New Jersey     ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
North Carolina ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Tennessee      ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Missouri       ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Indiana        ['Heart disease', 'Cancer', 'CLRD', 'Stroke']
Virginia       ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Massachusetts  ['Cancer', 'Heart disease', 'Stroke', 'CLRD']
Alabama        ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Washington     ['Cancer', 'Heart disease', 'Stroke', 'CLRD']
Maryland       ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Oklahoma       ['Heart disease', 'Cancer', 'CLRD', 'Stroke']
Connecticut    ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Arkansas       ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Oregon         ['Cancer', 'Heart disease', 'Stroke', 'CLRD']
Iowa           ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Kansas         ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
Nebraska       ['Heart disease', 'Cancer', 'CLRD', 'Stroke']
Maine          ['Cancer', 'Heart disease', 'CLRD', 'Stroke']
Rhode Island   ['Heart disease', 'Cancer', 'CLRD', 'Stroke']
New Hampshire  ['Cancer', 'Heart disease', 'CLRD', 'Stroke']
Idaho          ['Heart disease', 'Cancer', 'CLRD', 'Stroke']
Delaware       ['Heart disease', 'Cancer', 'CLRD', 'Stroke']
South Dakota   ['Heart disease', 'Cancer', 'Stroke', 'CLRD']
-------------  ------------------------------------------
There are 30 States have top four causes of deaths as Heart disease, Cancer, Stroke a
nd CLRD
```

In [35]:
```
dict3 = {}# creating an empty dictionary
for i in l7 :#for loop for getting key-value pairs of a dictionary using list of keys
    dict3[i] =l1[i] #storing ith key and its ith value of a dictionary in a new dictio
dict3
print("North Dakota is the only State which has Heart disease', 'Cancer', 'Stroke', 'A
```

```
North Dakota is the only State which has Heart disease', 'Cancer', 'Stroke', 'Alzheim
er's disease as the top four leading causes of death.
```

In [36]:
```
dict4 = {}# creating an empty dictionary
for i in l6 :#for loop for getting key-value pairs of a dictionary using list of keys
    dict4[i] =l1[i] #storing ith key and its ith value of a dictionary in a new dictio
print(tabulate(dict4.items()))#printing formatted table of a dictionary
print("There are "+str(len(l6))+" States which have the top four leading causes as 'He
```

```
-------------------  ------------------------------------------------------------
Texas                ['Heart disease', 'Cancer', 'Stroke', 'Unintentional injuries']
Georgia              ['Heart disease', 'Cancer', 'Stroke', 'Unintentional injuries']
Wisconsin            ['Heart disease', 'Cancer', 'Stroke', 'Unintentional injuries']
Louisiana            ['Heart disease', 'Cancer', 'Unintentional injuries', 'Stroke']
South Carolina       ['Heart disease', 'Cancer', 'Stroke', 'Unintentional injuries']
Minnesota            ['Cancer', 'Heart disease', 'Stroke', 'Unintentional injuries']
Mississippi          ['Heart disease', 'Cancer', 'Unintentional injuries', 'Stroke']
Utah                 ['Heart disease', 'Cancer', 'Unintentional injuries', 'Stroke']
Hawaii               ['Heart disease', 'Cancer', 'Stroke', 'Unintentional injuries']
District of Columbia ['Heart disease', 'Cancer', 'Stroke', 'Unintentional injuries']
-------------------  ------------------------------------------------------------
There are 10 States which have the top four leading causes as 'Heart disease', 'Cance
r', 'Stroke' and 'Unintentional injuries'.
```

Result:

We came up with the following results on applying the algorithm:

- 9 States have 'Heart disease', 'Cancer', 'Unintentional injuries' and 'CLRD' as top four leading causes.
- 30 'State' have top four causes of deaths as Heart disease, Cancer, Stroke and CLRD.
- North Dakota is the only State which has Heart disease', 'Cancer', 'Stroke', "Alzheimer's disease as the top four leading causes of death.
- 10 States have the top four leading causes as 'Heart disease', 'Cancer', 'Stroke' and 'Unintentional injuries'.

4. Are there year-by-year changes in the four leading causes of death nationwide?

Coding Algorithm:

1. The dataframe is first sorted with year.
2. It is then grouped in the dataframe by 'Year' and 'Cause' by taking a sum of 'Deaths'.
3. Create a dataframe of an output from step 2.
4. Sort values of 'deaths' column in descending order.
5. Remove 'All causes' from 'Cause Name' column and plot data.

In [37]:
```
df5 = data1.sort_values(by=['Year']) #sorting the dataframe with year
df5.head()
```

	Year	113 Cause Name	Cause Name	State	Deaths	Age-adjusted Death Rate
10295	1999	All Causes	All causes	District of Columbia	6076	1087.3
1911	1999	Alzheimer's disease (G30)	Alzheimer's disease	Massachusetts	1182	16.5
6407	1999	Chronic lower respiratory diseases (J40-J47)	CLRD	Iowa	1643	46.8
1906	1999	Nephritis, nephrotic syndrome and nephrosis (N...	Kidney disease	West Virginia	345	16.4
8906	1999	Malignant neoplasms (C00-C97)	Cancer	North Carolina	15815	207.1

In [38]:
```
df5 = df5.groupby(['Year','Cause Name'])['Deaths'].sum() # grouping the dataframe by Y
df5 = pd.DataFrame(df5) # creating a dataframe of a output from above sentence
df5.head()
```

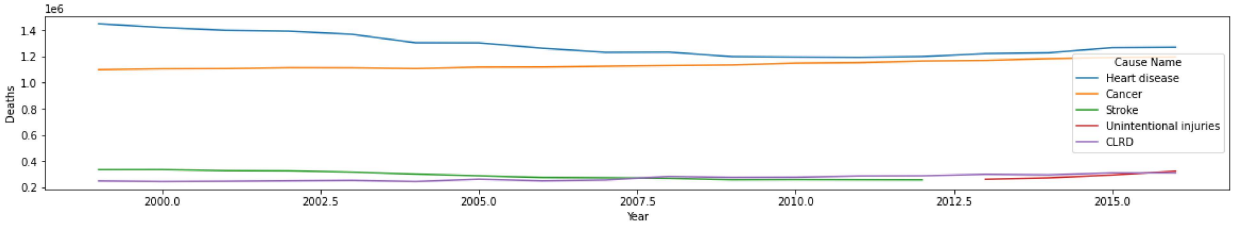
		Deaths
Year	**Cause Name**	
1999	**All causes**	4782798
	Alzheimer's disease	89072
	CLRD	248362
	Cancer	1099676
	Diabetes	136798

In [39]:
```
df6 = df5.sort_values(['Deaths'], ascending = False) # sorting values of deaths columr
df6 = df6.drop('All causes', level=1, axis=0) # remove 'All causes' from 'Cause Name'
df7 = df6.groupby(['Year']).head(4) # storing only top 4 causes of deaths by groupby c
df7.head()
```

		Deaths
Year	**Cause Name**	
1999	**Heart disease**	1450384
2000	**Heart disease**	1421520
2001	**Heart disease**	1400284
2002	**Heart disease**	1393894
2003	**Heart disease**	1370178

In [40]:
```
sns.lineplot(x='Year', y='Deaths', hue='Cause Name',
                         data=df7) #plotting a line graph from searborn library and usi
```

```
<AxesSubplot:xlabel='Year', ylabel='Deaths'>
```



Result:

The line graph above shows the top four causes of deaths over the years. The third leading cause was stroke till 2015 but changed to unintentional injuries after 2012; others like heart disease, cancer and CLRD remained the same in the chart of top leading causes of deaths. Further, heart disease and cancer are the highest of all, and cancer is increasing over time. However, heart disease and stroke significantly decreased with time. Also, if we see overall cancer, unintentional injuries, and CLRD, are on the rise.

VI. Potential Questions For Future Research

1. Do men live longer than women? This question could be addressed by examining gender differences in mortality and the genetics at play for both, men and women, that change with age. Additional data reqd:

- Gender of people dying in the United States.
- Did these people suffer a family history of health issues?
- Any genetic issues encountered during their lifetime?

1. We could also discuss health differences across historical time. For e.g. mortality rates during the colonial times vs now.
2. We could analyse this on a global level. With data on mortality from different countries. This could answer whether a country's economic power and health policies play a role.

V. Conclusion

The above analysis concludes that the United States is facing an increasing average death rate year by year in which heart disease and cancer are far above on the chart of number of deaths per year in the United States. Other than heart disease and cancer, stroke and CLRD also have a significant impact on deaths in the States.Also, till 2012, stroke was the fourth leading cause of mortality, although after 2012, unintentional injuries became the fourth leading cause, but after 2016, it crossed CLRD and became the third leading cause.

Furthermore, every state's individual causes of deaths are different. However, more than half of the States in the United States has Heart disease, Cancer, Stroke and CLRD as the top leading causes.