In [5]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [6]:
```
train_df=pd.read_csv(r'C:\Users\cool_adarsh\Downloads\train-csv2.csv')
```

```
C:\Users\cool_adarsh\AppData\Local\Temp\ipykernel_2880\1205194078.py:1: DtypeWarning: Columns (1) have mixed types. Specify dtype option on import or set low_memory=False.
  train_df=pd.read_csv(r'C:\Users\cool_adarsh\Downloads\train-csv2.csv')
```

In [12]:
```
test_df=pd.read_csv(r'C:\Users\cool_adarsh\Downloads\test-csv2.csv')
```

In [13]:
```
train_df.head(20)
```

Out[13]:

	Id	County	Province_State	Country_Region	Population	Weight	Date	Target	TargetValue
0	1	NaN	NaN	Afghanistan	27657145	0.058359	2020-01-23	ConfirmedCases	0
1	2	NaN	NaN	Afghanistan	27657145	0.583587	2020-01-23	Fatalities	0
2	3	NaN	NaN	Afghanistan	27657145	0.058359	2020-01-24	ConfirmedCases	0
3	4	NaN	NaN	Afghanistan	27657145	0.583587	2020-01-24	Fatalities	0
4	5	NaN	NaN	Afghanistan	27657145	0.058359	2020-01-25	ConfirmedCases	0
5	6	NaN	NaN	Afghanistan	27657145	0.583587	2020-01-25	Fatalities	0
6	7	NaN	NaN	Afghanistan	27657145	0.058359	2020-01-26	ConfirmedCases	0
7	8	NaN	NaN	Afghanistan	27657145	0.583587	2020-01-26	Fatalities	0
8	9	NaN	NaN	Afghanistan	27657145	0.058359	2020-01-27	ConfirmedCases	0
9	10	NaN	NaN	Afghanistan	27657145	0.583587	2020-01-27	Fatalities	0
10	11	NaN	NaN	Afghanistan	27657145	0.058359	2020-01-28	ConfirmedCases	0
11	12	NaN	NaN	Afghanistan	27657145	0.583587	2020-01-28	Fatalities	0
12	13	NaN	NaN	Afghanistan	27657145	0.058359	2020-01-29	ConfirmedCases	0
13	14	NaN	NaN	Afghanistan	27657145	0.583587	2020-01-29	Fatalities	0
14	15	NaN	NaN	Afghanistan	27657145	0.058359	2020-01-30	ConfirmedCases	0
15	16	NaN	NaN	Afghanistan	27657145	0.583587	2020-01-30	Fatalities	0
16	17	NaN	NaN	Afghanistan	27657145	0.058359	2020-01-31	ConfirmedCases	0
17	18	NaN	NaN	Afghanistan	27657145	0.583587	2020-01-31	Fatalities	0
18	19	NaN	NaN	Afghanistan	27657145	0.058359	2020-02-01	ConfirmedCases	0
19	20	NaN	NaN	Afghanistan	27657145	0.583587	2020-02-01	Fatalities	0

In [14]:
```
test_df.head(20)
```

Out[14]:

	ForecastId	County	Province_State	Country_Region	Population	Weight	Date	Target
0	1	NaN	NaN	Afghanistan	27657145	0.058359	2020-04-27	ConfirmedCases
1	2	NaN	NaN	Afghanistan	27657145	0.583587	2020-04-27	Fatalities
2	3	NaN	NaN	Afghanistan	27657145	0.058359	2020-04-28	ConfirmedCases
3	4	NaN	NaN	Afghanistan	27657145	0.583587	2020-04-28	Fatalities
4	5	NaN	NaN	Afghanistan	27657145	0.058359	2020-04-29	ConfirmedCases
5	6	NaN	NaN	Afghanistan	27657145	0.583587	2020-04-29	Fatalities
6	7	NaN	NaN	Afghanistan	27657145	0.058359	2020-04-30	ConfirmedCases
7	8	NaN	NaN	Afghanistan	27657145	0.583587	2020-04-30	Fatalities
8	9	NaN	NaN	Afghanistan	27657145	0.058359	2020-05-01	ConfirmedCases
9	10	NaN	NaN	Afghanistan	27657145	0.583587	2020-05-01	Fatalities
10	11	NaN	NaN	Afghanistan	27657145	0.058359	2020-05-02	ConfirmedCases
11	12	NaN	NaN	Afghanistan	27657145	0.583587	2020-05-02	Fatalities
12	13	NaN	NaN	Afghanistan	27657145	0.058359	2020-05-03	ConfirmedCases
13	14	NaN	NaN	Afghanistan	27657145	0.583587	2020-05-03	Fatalities
14	15	NaN	NaN	Afghanistan	27657145	0.058359	2020-05-04	ConfirmedCases
15	16	NaN	NaN	Afghanistan	27657145	0.583587	2020-05-04	Fatalities
16	17	NaN	NaN	Afghanistan	27657145	0.058359	2020-05-05	ConfirmedCases
17	18	NaN	NaN	Afghanistan	27657145	0.583587	2020-05-05	Fatalities
18	19	NaN	NaN	Afghanistan	27657145	0.058359	2020-05-06	ConfirmedCases
19	20	NaN	NaN	Afghanistan	27657145	0.583587	2020-05-06	Fatalities

In [9]:
```
train_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 969640 entries, 0 to 969639
Data columns (total 9 columns):
 #   Column          Non-Null Count   Dtype 
---  ------          --------------   ----- 
 0   Id              969640 non-null  int64 
 1   County          880040 non-null  object
 2   Province_State  917280 non-null  object
 3   Country_Region  969640 non-null  object
 4   Population      969640 non-null  int64 
 5   Weight          969640 non-null  float64
 6   Date            969640 non-null  object
 7   Target          969640 non-null  object
 8   TargetValue     969640 non-null  int64 
dtypes: float64(1), int64(3), object(5)
memory usage: 66.6+ MB
```

In [10]:
```
train_df.describe()
```

Out[10]:

	Id	Population	Weight	TargetValue
count	969640.000000	9.696400e+05	969640.000000	969640.000000
mean	484820.500000	2.720127e+06	0.530870	12.563518
std	279911.101846	3.477771e+07	0.451909	302.524795
min	1.000000	8.600000e+01	0.047491	-10034.000000
25%	242410.750000	1.213300e+04	0.096838	0.000000
50%	484820.500000	3.053100e+04	0.349413	0.000000
75%	727230.250000	1.056120e+05	0.968379	0.000000
max	969640.000000	1.395773e+09	2.239186	36163.000000

In [15]:
```
test_df.isnull().sum()
```

Out[15]:
```
ForecastId          0
County          28800
Province_State  16830
Country_Region      0
Population          0
Weight              0
Date                0
Target              0
dtype: int64
```

In [16]:
```
train_df.isnull().sum()
```

Out[16]:
```
Id                  0
County          89600
Province_State  52360
Country_Region      0
Population          0
Weight              0
Date                0
Target              0
TargetValue         0
dtype: int64
```

In [18]:
```
train_df.shape
```

Out[18]:
```
(969640, 9)
```

In [19]:
```
train_df[train_df['Country_Region']=='China']
```

Out[19]:

	Id	County	Province_State	Country_Region	Population	Weight	Date	Target	TargetValue
15680	15681	NaN	Anhui	China	62550000	0.055706	2020-01-23	ConfirmedCases	8
15681	15682	NaN	Anhui	China	62550000	0.557057	2020-01-23	Fatalities	0
15682	15683	NaN	Anhui	China	62550000	0.055706	2020-01-24	ConfirmedCases	6
15683	15684	NaN	Anhui	China	62550000	0.557057	2020-01-24	Fatalities	0
15684	15685	NaN	Anhui	China	62550000	0.055706	2020-01-25	ConfirmedCases	24
...	...	...	...	...	...	...	...	...	...
25195	25196	NaN	NaN	China	1395773400	0.474908	2020-06-08	Fatalities	0
25196	25197	NaN	NaN	China	1395773400	0.047491	2020-06-09	ConfirmedCases	3
25197	25198	NaN	NaN	China	1395773400	0.474908	2020-06-09	Fatalities	0
25198	25199	NaN	NaN	China	1395773400	0.047491	2020-06-10	ConfirmedCases	11
25199	25200	NaN	NaN	China	1395773400	0.474908	2020-06-10	Fatalities	0

9520 rows × 9 columns

In [21]:
```
train_df[train_df['Country_Region']=='India']
```

Out[21]:

	Id	County	Province_State	Country_Region	Population	Weight	Date	Target	TargetValue
40320	40321	NaN	NaN	India	1295210000	0.04766	2020-01-23	ConfirmedCases	0
40321	40322	NaN	NaN	India	1295210000	0.47660	2020-01-23	Fatalities	0
40322	40323	NaN	NaN	India	1295210000	0.04766	2020-01-24	ConfirmedCases	0
40323	40324	NaN	NaN	India	1295210000	0.47660	2020-01-24	Fatalities	0
40324	40325	NaN	NaN	India	1295210000	0.04766	2020-01-25	ConfirmedCases	0
...	...	...	...	...	...	...	...	...	...
40595	40596	NaN	NaN	India	1295210000	0.47660	2020-06-08	Fatalities	266
40596	40597	NaN	NaN	India	1295210000	0.04766	2020-06-09	ConfirmedCases	10218
40597	40598	NaN	NaN	India	1295210000	0.47660	2020-06-09	Fatalities	277
40598	40599	NaN	NaN	India	1295210000	0.04766	2020-06-10	ConfirmedCases	437
40599	40600	NaN	NaN	India	1295210000	0.47660	2020-06-10	Fatalities	-5

280 rows × 9 columns

In [25]:
```
group_data=train_df.groupby('Country_Region').sum()

countries= group_data.nlargest(10,'Population')['TargetValue']

countries
```

```
C:\Users\cool_adarsh\AppData\Local\Temp\ipykernel_2880\3685022227.py:1: FutureWarning: The default value of numeric_only in DataFrameGroupBy.sum is deprecated. In a future version, numeric_only will defaul
lt to False. Either specify numeric_only or select only columns which should be valid for the function.
  group_data=train_df.groupby('Country_Region').sum()
```

Out[25]:
```
Country_Region
China           176564
India           284328
US             6317214
Indonesia        36275
Brazil          812096
Pakistan        115957
Nigeria          14255
Bangladesh       75877
Russia          499373
```

```
Japan           18066
Name: TargetValue, dtype: int64
```

In [ ]: