

# Data Visualisation representing the Fatality Rate of Most Affected Countries by Covid 19

Reading the Data from John-Hopkins University showing the daily statistics of Covid 19 for each country in duration 22-Jan-2020 to 28-April-2020. For US, Statewise Covid 19 Statistics are collected.

```
In [1]: import csv
path = 'C:\Sanjay\Coursera\Python\Course1_assignment\johns-hopkins-covid-19-daily-dashboards-cases-over-time.csv'
f = open(path)
reader = csv.reader(f, delimiter = ',')
header = next(reader)
```

```
In [2]: header
```

```
Out[2]: ['country_region',
'last_update',
'confirmed',
'deaths',
'recovered',
'active',
'delta_confirmed',
'delta_recovered',
'incident_rate',
'people_tested',
'people_hospitalized',
'province_state',
'fips',
'uid',
'iso3',
'report_date_string']
```

Creating dataset containing list of dictionaries

```
In [3]: dataset = []
```

```
for line in reader:
    d = dict(zip(header,line))
    dataset.append(d)
len(dataset)
```

Out[3]: 23814

Deriving Month and Year in Month\_YYYY format from Reporting Date field

```
In [4]: import time
for i in range(len(dataset)):
    dataset[i]['month_year'] = time.strftime('%B_%Y',time.strptime(data
set[i]['report_date_string'].strip(), '%Y-%m-%d'))
```

Segragting Non-US Countries Data and US data as US data is represented with States

```
In [5]: NonUSdata = []
USdata = []

for i in range(len(dataset)):
    if dataset[i]['country_region'] != 'US':
        NonUSdata.append(dataset[i])
    else:
        USdata.append(dataset[i])
```

Sorting US data based on Reporting Date

```
In [6]: USdata = sorted(USdata, key=lambda k: (k['report_date_string']))
```

Sorting Non-US Countries Data based on Country and Reporting Date

```
In [7]: NonUSdata = sorted(NonUSdata, key=lambda k: (k['country_region'], k['re
port_date_string']))
```

Creating new dataset from Non-US countries having data only from last day of each Month

```
In [8]: dataset2 = []
prevMonth = None
prevCountry = None

for i in range(len(NonUSdata)):
```

```

if NonUSdata[i]['country_region'] == prevCountry:
    if NonUSdata[i]['month_year'] != prevMonth:
        dataset2.append(NonUSdata[i-1])
elif NonUSdata[i]['country_region'] != prevCountry:
    dataset2.append(NonUSdata[i-1])
prevCountry = NonUSdata[i]['country_region']
prevMonth = NonUSdata[i]['month_year']

```

Calculating Monthwise Fatality Rate for each Non-US Country

```

In [9]: for i in range(len(dataset2)):
        if int(dataset2[i]['confirmed']) != 0:
            dataset2[i]['fatality_rate'] = (int(dataset2[i]['deaths'])/int(
dataset2[i]['confirmed']))*100
            #print(dataset2[i]['country_region']+' '+dataset2[i]['month_year']+' '+dataset2[i]['confirmed']+' '+dataset2[i]['deaths']+' '+str(dataset2[i]['recovery_rate']))
        else:
            dataset2[i]['fatality_rate'] = 0.0

```

Doing Monthwise Summation of all Confirmed Cases and Death Cases for US States and appending this data to Non-US Countries dataset

```

In [10]: prevMonth = None
sumConf = 0
sumDeath = 0

for i in range(len(USdata)):
    if USdata[i]['month_year'] != prevMonth:
        if prevMonth == None:
            sumConf = sumConf + int(USdata[i]['confirmed'])
            sumDeath = sumDeath + int(USdata[i]['deaths'])
        else:
            dataset2.append({'country_region': 'US', 'month_year': prevMonth, 'confirmed': sumConf, 'deaths': sumDeath, 'fatality_rate': (sumDeath/sumConf)*100})
            sumConf = int(USdata[i]['confirmed'])
            sumDeath = int(USdata[i]['deaths'])
    elif USdata[i]['month_year'] == prevMonth:
        sumConf = sumConf + int(USdata[i]['confirmed'])

```

```

        sumDeath = sumDeath + int(USdata[i]['deaths'])
        if i == (len(USdata)-1):
            dataset2.append({'country_region': 'US', 'month_year': prevMonth, 'confirmed': sumConf, 'deaths': sumDeath, 'fatality_rate': (sumDeath/sumConf)*100})
            prevMonth = USdata[i]['month_year']

```

Dataset containing both US and Non-US countries will be sorted based on Country and Confirmed Cases

```

In [12]: dataset2 = sorted(dataset2, key=lambda k: (k['country_region'], int(k['confirmed'])))

```

Countries having more than 100000 Confirmed Cases will be considered for Visualization

```

In [14]: topCovidAffectedCountry = []
        for i in range(len(dataset2)):
            if int(dataset2[i]['confirmed']) > 100000:
                topCovidAffectedCountry.append(dataset2[i]['country_region'])
                #print(dataset2[i]['country_region']+' '+dataset2[i]['month_year']+' '+str(dataset2[i]['confirmed'])+' '+str(dataset2[i]['deaths'])+' '+str(dataset2[i]['fatality_rate']))

        topCovidAffectedCountry = list(set(topCovidAffectedCountry))

        print(topCovidAffectedCountry)

        ['Turkey', 'US', 'Italy', 'Spain', 'Germany', 'United Kingdom', 'France']

```

Creating Dataset having Monthly information for Mostly infected Countries

```

In [15]: finalDataSet = []

        for i in range(len(dataset2)):
            if dataset2[i]['country_region'] in topCovidAffectedCountry:
                finalDataSet.append(dataset2[i])

```

Sorting Dataset based on Country Name and Confirmed Cases. This dataset will be used for Visualization

```

In [16]: finalDataSet = sorted(finalDataSet, key=lambda k: (k['country_region'], int(k['confirmed'])))

```

```
for i in range(len(finalDataSet)):
    print(finalDataSet[i]['country_region']+' '+finalDataSet[i]['month_year']+' '+str(finalDataSet[i]['confirmed'])+' '+str(finalDataSet[i]['deaths'])+' '+str(finalDataSet[i]['fatality_rate']))
```

```
France January_2020 5 0 0.0
France February_2020 100 2 2.0
France March_2020 52827 3532 6.68597497491813
France April_2020 169053 23694 14.015722879806924
Germany January_2020 5 0 0.0
Germany February_2020 79 0 0.0
Germany March_2020 71808 775 1.0792669340463457
Germany April_2020 159912 6314 3.948421631897544
Italy January_2020 2 0 0.0
Italy February_2020 1128 29 2.5709219858156027
Italy March_2020 105792 12428 11.747580157289777
Italy April_2020 201505 27359 13.577330587330339
Spain January_2020 0 0 0.0
Spain February_2020 45 0 0.0
Spain March_2020 95923 8464 8.823744044702522
Spain April_2020 232128 23822 10.26244141163496
Turkey January_2020 0 0 0.0
Turkey February_2020 0 0 0.0
Turkey March_2020 13531 214 1.5815534698100657
Turkey April_2020 114653 2992 2.6096133550801115
US January_2020 76 0 0.0
US February_2020 1000 2 0.2
US March_2020 2173108 51071 2.3501363024755326
US April_2020 34795702 1784175 5.127572939899301
United Kingdom January_2020 2 0 0.0
United Kingdom February_2020 23 0 0.0
United Kingdom March_2020 25481 1793 7.036615517444371
United Kingdom April_2020 162350 21745 13.393902063443178
```

Data Visualization representing Fatality Rate of the Countries which are most affected by Covid 19.

```
In [17]: import matplotlib.pyplot as plt

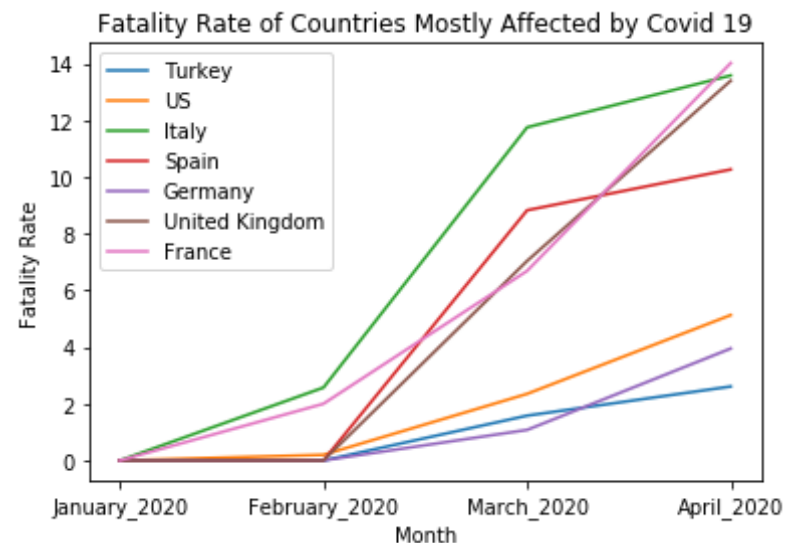
for val in topCovidAffectedCountry:
```

```

xvalue = []
yvalue = []
for i in range(len(finalDataSet)):
    if val == finalDataSet[i]['country_region']:
        xvalue.append(finalDataSet[i]['fatality_rate'])
        yvalue.append(finalDataSet[i]['month_year'])
plt.plot(yvalue, xvalue, label=val)

plt.ylabel('Fatality Rate')
plt.xlabel('Month')
plt.legend()
plt.title('Fatality Rate of Countries Mostly Affected by Covid 19')
plt.show()

```



Following inferences can be drawn from the above Visualization. i. Turkey and Germany has lower fatality rate when compared with other countries. These Countries have performed well while containing pandemic. ii. The Fatality Rate for United States is increased linearly after February 2020. But still less than other Italy, Spain, France and UK. iii. Italy and Spain are able to control fatality rate after March 2020. iv. Out of all 7 mostly affected countries, France and United Kingdom are not able to control fatality rate after February 2020.