```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
print('Modules are imported.')
```

```
Modules are imported.
```

```
df=pd.read_csv("covid19_Confirmed_dataset.csv")
df.head()
```

	Province/State	Country/Region	Lat	Long	1/22/20	1/23/20	1/24/20	1/
0	NaN	Afghanistan	33.0000	65.0000	0	0	0	
1	NaN	Albania	41.1533	20.1683	0	0	0	
2	NaN	Algeria	28.0339	1.6596	0	0	0	
3	NaN	Andorra	42.5063	1.5218	0	0	0	
4	NaN	Angola	-11.2027	17.8739	0	0	0	

5 rows × 104 columns

```
df.shape
```

```
(266, 104)
```

```
df.drop(["Lat","Long"],axis=1,inplace=True)
```

```
df.head()
```

	Province/State	Country/Region	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/
0	NaN	Afghanistan	0	0	0	0	0	

```
aggregating=df.groupby("Country/Region").sum()
```

```
aggregating.head()
```

Country/Region	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	1/28/20	1/2
Afghanistan	0	0	0	0	0	0	0	
Albania	0	0	0	0	0	0	0	
Algeria	0	0	0	0	0	0	0	
Andorra	0	0	0	0	0	0	0	
Angola	0	0	0	0	0	0	0	

5 rows × 100 columns

```
aggregating.shape
```

```
(187, 100)
```

Visualizing data related to a country for example China visualization always helps for better understanding of our data.

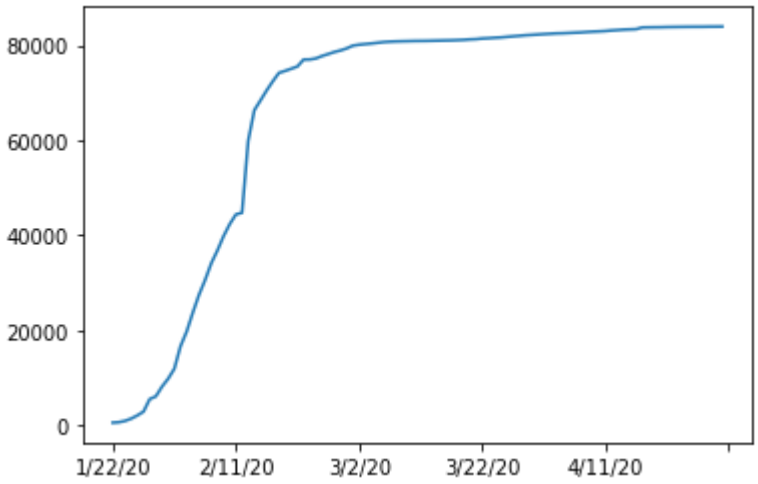
```
aggregating.loc["China"].plot()
aggregating.loc["Italy"].plot()
aggregating.loc["Spain"].plot()
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7f29b04b38e0>
```

Calculating a good measure we need to find a good measure reperestend as a number, describing the spread of the virus in a country.

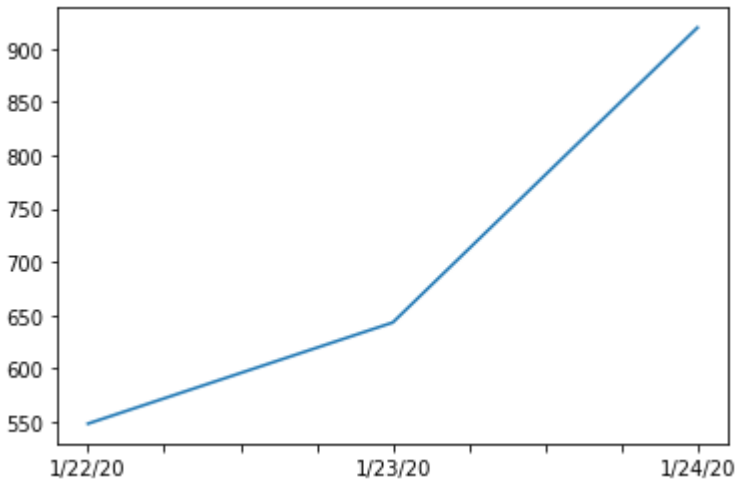

```
aggregating.loc['China'].plot()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f29b03fde50>
```




```
aggregating.loc['China'][:3].plot()
```

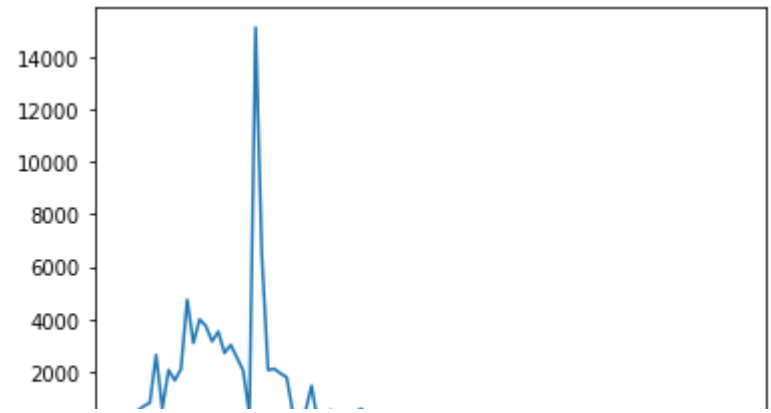
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f29afef8c40>
```



caculating the first derivative of the curve

```
aggregating.loc['China'].diff().plot()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f29aff139a0>
```



find maxmimum infection rate for China

```
aggregating.loc['China'].diff().max()
```

```
15136.0
```

```
aggregating.loc['Italy'].diff().max()
```

```
6557.0
```

```
aggregating.loc['Spain'].diff().max()
```

```
9630.0
```

```
aggregating.describe()
```

	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	
count	187.000000	187.000000	187.000000	187.000000	187.000000	187.000000	1
mean	2.967914	3.497326	5.032086	7.668449	11.326203	15.652406	
std	40.071697	47.017656	67.271230	102.808451	151.725366	210.370186	4
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
25%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
50%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
75%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
max	548.000000	643.000000	920.000000	1406.000000	2075.000000	2877.000000	55

8 rows × 100 columns

Colab paid products - Cancel contracts here

0s completed at 10:30 AM