In [8]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

import plotly.graph_objects as go
import plotly.express as px

import warnings
warnings.filterwarnings('ignore')
```

In [9]:

```
import os
```

In [10]:

```
files=os.listdir('D:\Lito\Project 3 Covid-19 Data Analysis\Covid-19 Data Analysis\Covid-19')
files
```

Out[10]:

```
['country_wise_latest.csv',
 'covid_19_clean_complete.csv',
 'day_wise.csv',
 'full_grouped.csv',
 'usa_country_wise.csv',
 'worldometer_data.csv']
```

In [11]:

```
def read_data(path,filename):
    return pd.read_csv(path+'/'+filename)
```

In [12]:

```
path='D:\Lito\Project 3 Covid-19 Data Analysis\Covid-19 Data Analysis\Covid-19'
world_data=read_data(path,'worldometer_data.csv')
```

In [13]:

```
world_data.head()
```

Out[13]:

	Country/Region	Continent	Population	TotalCases	NewCases	TotalDeaths	NewDeaths	TotalRecovered	NewRecovere
0	USA	North America	3.311981e+08	5032179	NaN	162804.0	NaN	2576668.0	Na
1	Brazil	South America	2.127107e+08	2917562	NaN	98644.0	NaN	2047660.0	Na
2	India	Asia	1.381345e+09	2025409	NaN	41638.0	NaN	1377384.0	Na
3	Russia	Europe	1.459409e+08	871894	NaN	14606.0	NaN	676357.0	Na
4	South Africa	Africa	5.938157e+07	538184	NaN	9604.0	NaN	387316.0	Na

In [14]:

```
day_wise=read_data(path,files[2])
```

In [15]:

```
group_data=read_data(path,files[3])
```

In [16]:

```
usa_data=read_data(path,files[4])
```

In [17]:

```
province_data=read_data(path,files[1])
```

In [18]:

```
province_data.shape
```

Out[18]:

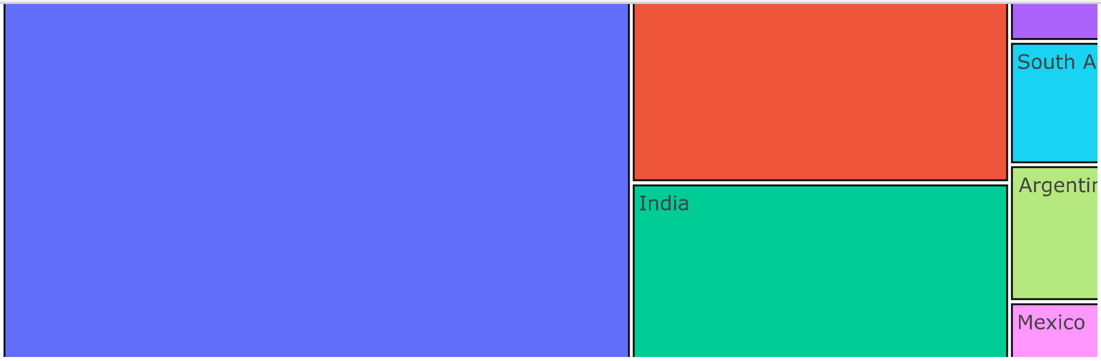
```
(49068, 10)
```

Which country has maximum Total cases, Deaths, Recovered Active Cases?

In [19]:
```
world_data.columns
```

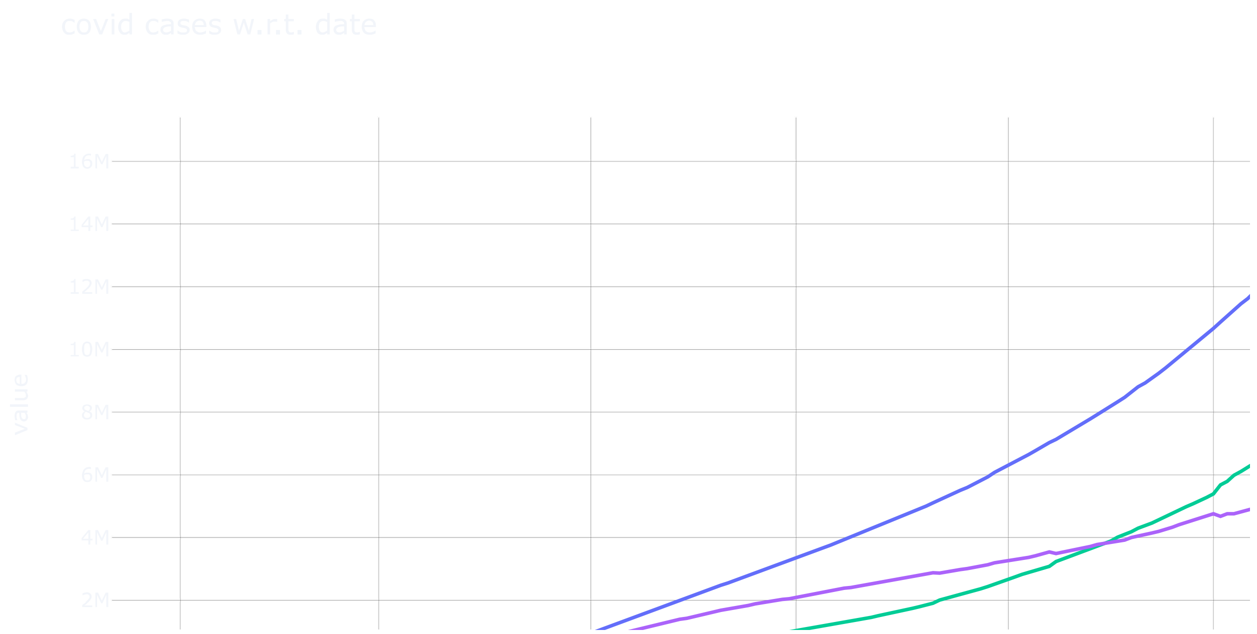
Out[19]:
```
Index(['Country/Region', 'Continent', 'Population', 'TotalCases', 'NewCases',
       'TotalDeaths', 'NewDeaths', 'TotalRecovered', 'NewRecovered',
       'ActiveCases', 'Serious,Critical', 'Tot Cases/1M pop', 'Deaths/1M pop',
       'TotalTests', 'Tests/1M pop', 'WHO Region'],
      dtype='object')
```

In [20]:
```
columns=['TotalCases','TotalDeaths','TotalRecovered','ActiveCases']
for i in columns:
    fig=px.treemap(world_data[0:20],values=i,path=['Country/Region'],template="plotly_dark",title="<
    fig.show()
```



What is the trend of confirmed Deaths Recovered Active Cases.

In [21]:
```
fig=px.line(day_wise,x="Date",y=["Confirmed","Deaths","Recovered","Active"],title="covid cases w.r.t
fig.show()
```



Find 20 most effected countries

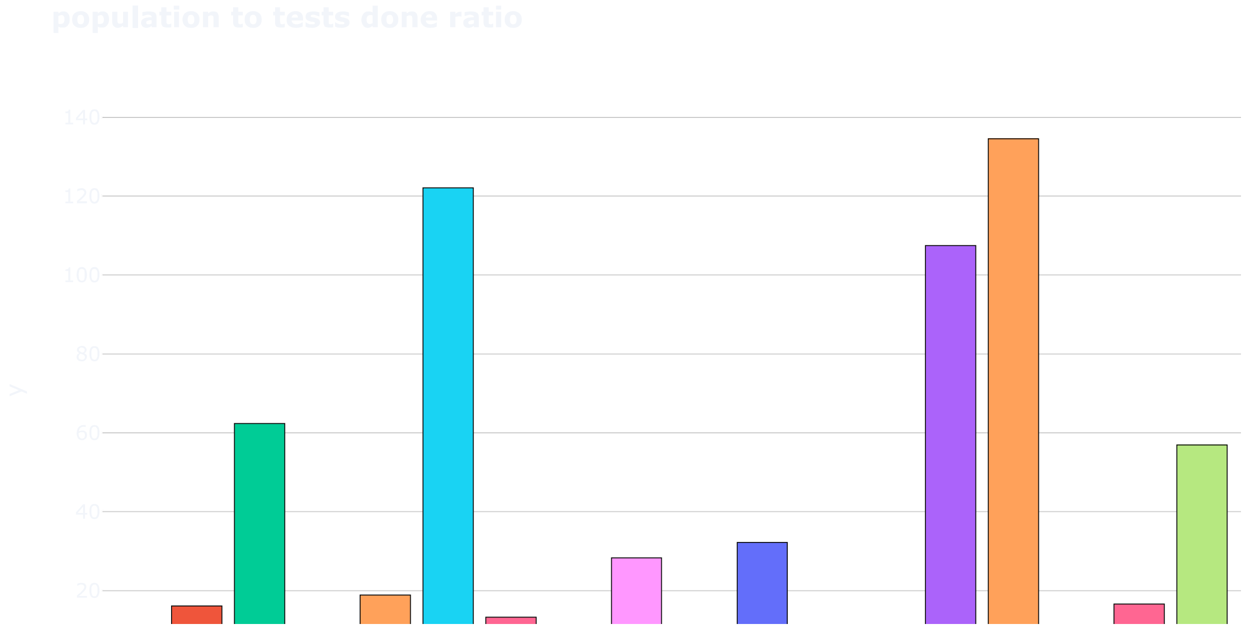
BarPlot Representation of Population to Tests Done Ratio

In [22]:
```
pop_test_ratio=world_data.iloc[0:20]['Population']/world_data.iloc[0:20]['TotalTests']
```

In [23]:
```
pop_test_ratio
```

Out[23]:
```
0       5.245489
1      16.106896
2      62.365033
3       4.911040
4      18.852446
5     122.115932
6      13.241331
7      10.866949
8      28.269105
9       6.618696
10     32.187237
11      3.877883
12      9.589865
13    107.484026
14    134.558952
15      8.514790
16     16.613857
17     56.934398
18      9.760649
19     16.353942
dtype: float64
```

In [24]:
```
fig=px.bar(world_data.iloc[0:20],color='Country/Region',y=pop_test_ratio,x='Country/Region',template
fig.show()
```



20 Countries that are badly affected by Corona.

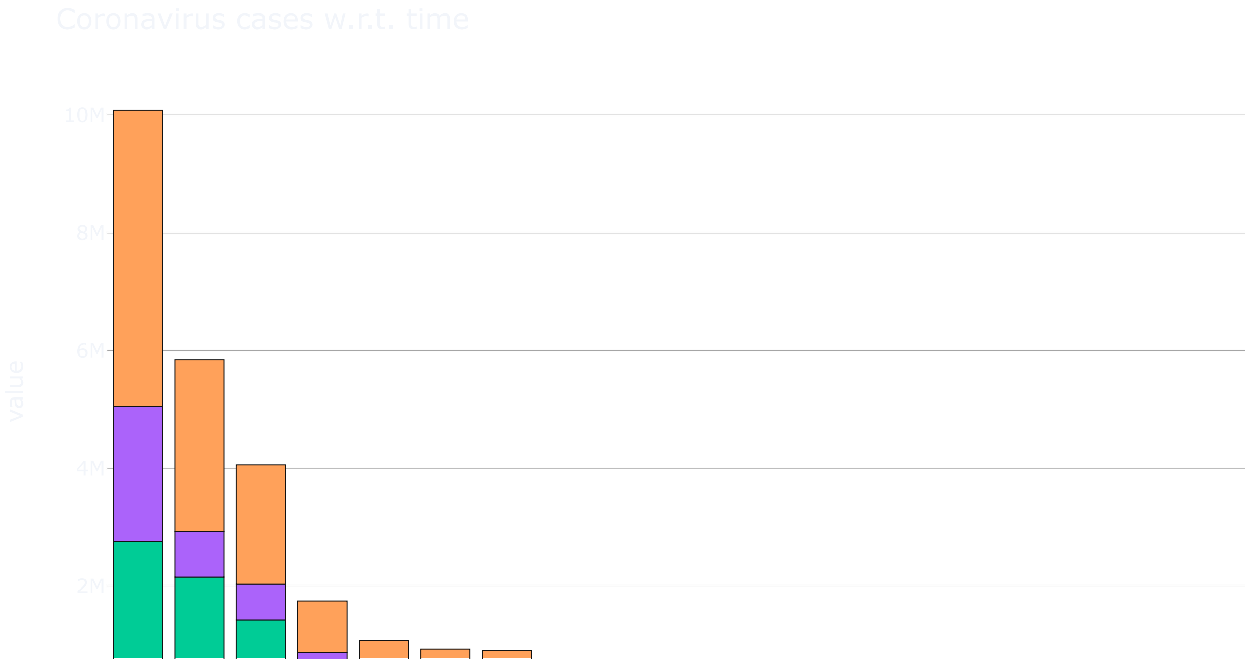
BarPlot the represenation of CoronaViruses Cases w.r.t Time

In [25]:

```
fig=px.bar(world_data.iloc[0:20],x='Country/Region',y=['Serious,Critical','TotalDeaths','TotalRecove
```

In [26]:

```
fig.update_layout({'title':"Coronavirus cases w.r.t. time"})
fig.show()
```



Top 20 Countries of Total Confirmed Cases, Total Recovered Cases, Total Deaths, Total Active Cases

In [27]:
```
world_data.head()
```

Out[27]:

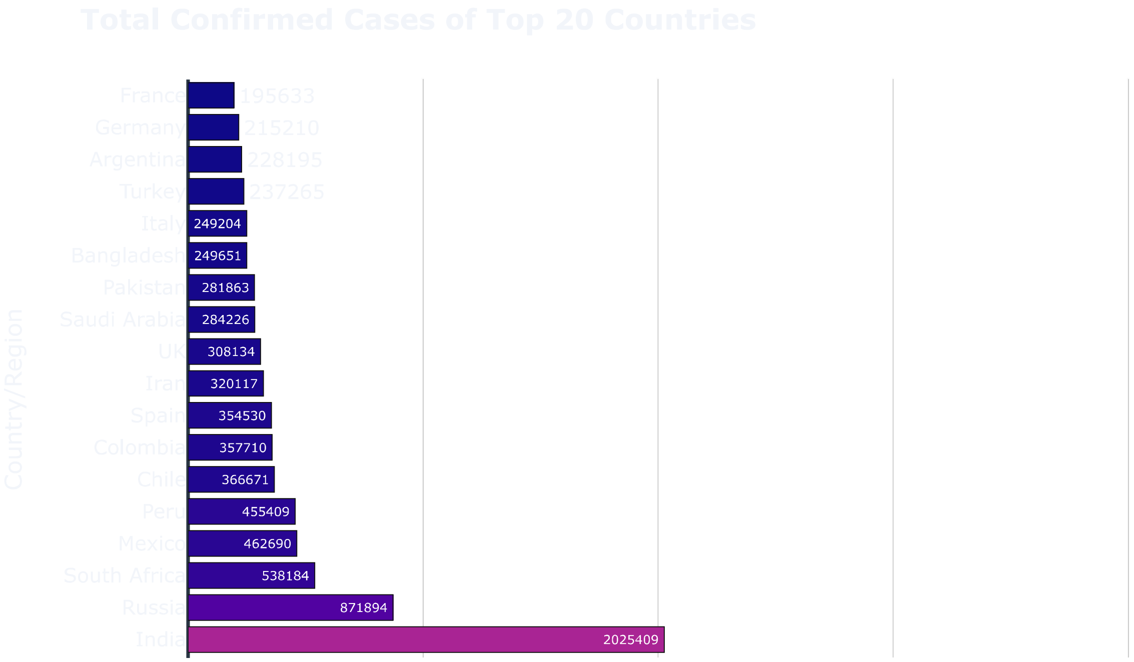
	Country/Region	Continent	Population	TotalCases	NewCases	TotalDeaths	NewDeaths	TotalRecovered	NewRecovere
0	USA	North America	3.311981e+08	5032179	NaN	162804.0	NaN	2576668.0	Na
1	Brazil	South America	2.127107e+08	2917562	NaN	98644.0	NaN	2047660.0	Na
2	India	Asia	1.381345e+09	2025409	NaN	41638.0	NaN	1377384.0	Na
3	Russia	Europe	1.459409e+08	871894	NaN	14606.0	NaN	676357.0	Na
4	South Africa	Africa	5.938157e+07	538184	NaN	9604.0	NaN	387316.0	Na

In [28]:
```
world_data['Country/Region'].nunique()
```

Out[28]:
```
209
```

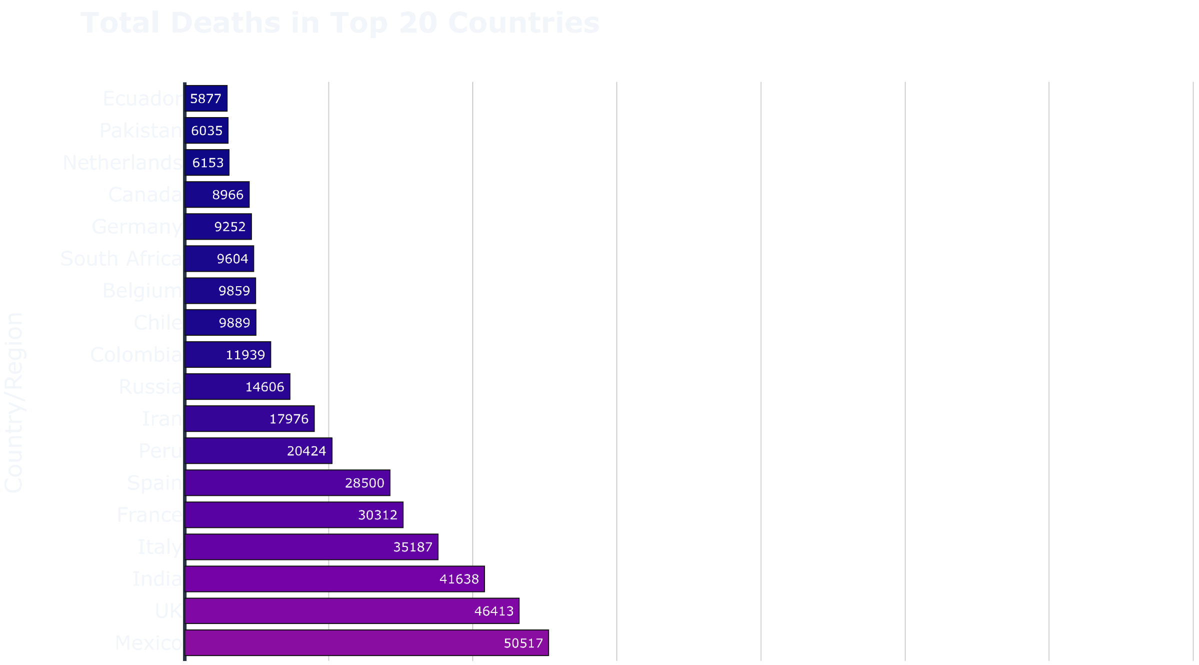
(1) Total Confirmed Cases of Top 20 Countries

In [29]:
```
fig=px.bar(world_data.iloc[0:20],y='Country/Region',x='TotalCases',color='TotalCases',text="TotalCas
fig.update_layout(template="plotly_dark",title_text="<b>Total Confirmed Cases of Top 20 Countries</b
fig.show()
```



Total Deaths in Top 20 Countries

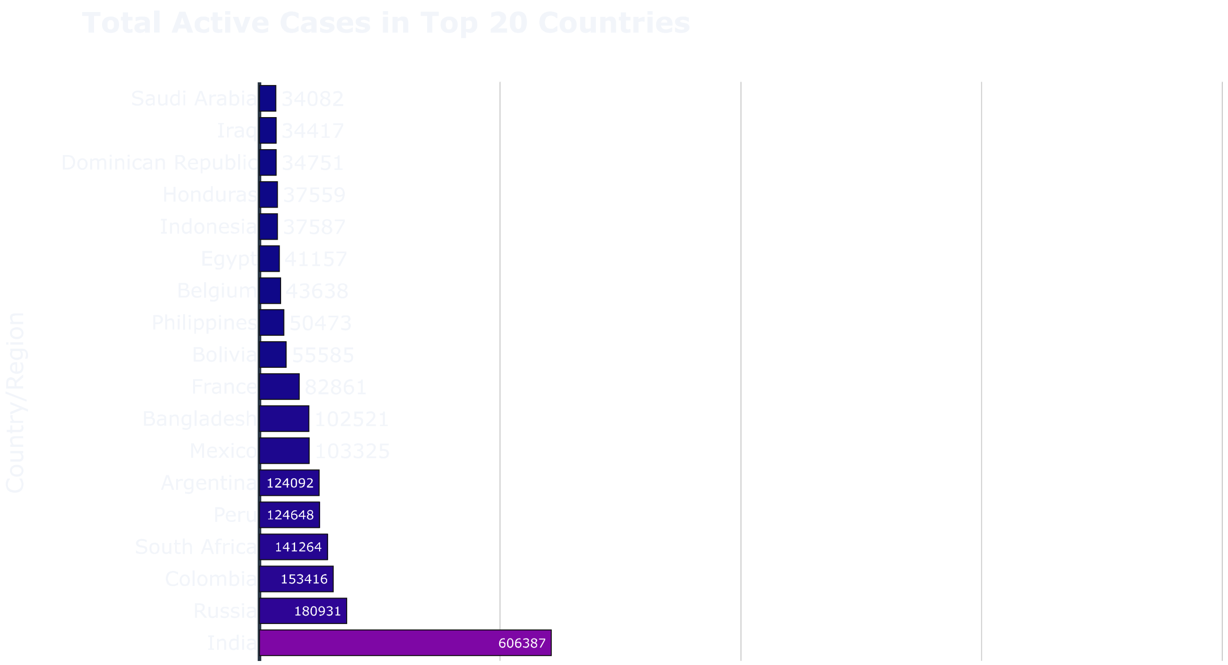
```
In [30]: fig=px.bar(world_data.sort_values(by='TotalDeaths',ascending=False)[0:20],y='Country/Region',x='Tota
         fig.update_layout(template="plotly_dark",title_text="<b>Total Deaths in Top 20 Countries</b>")
         fig.show()
```



Total Active Cases in Top 20 Countries

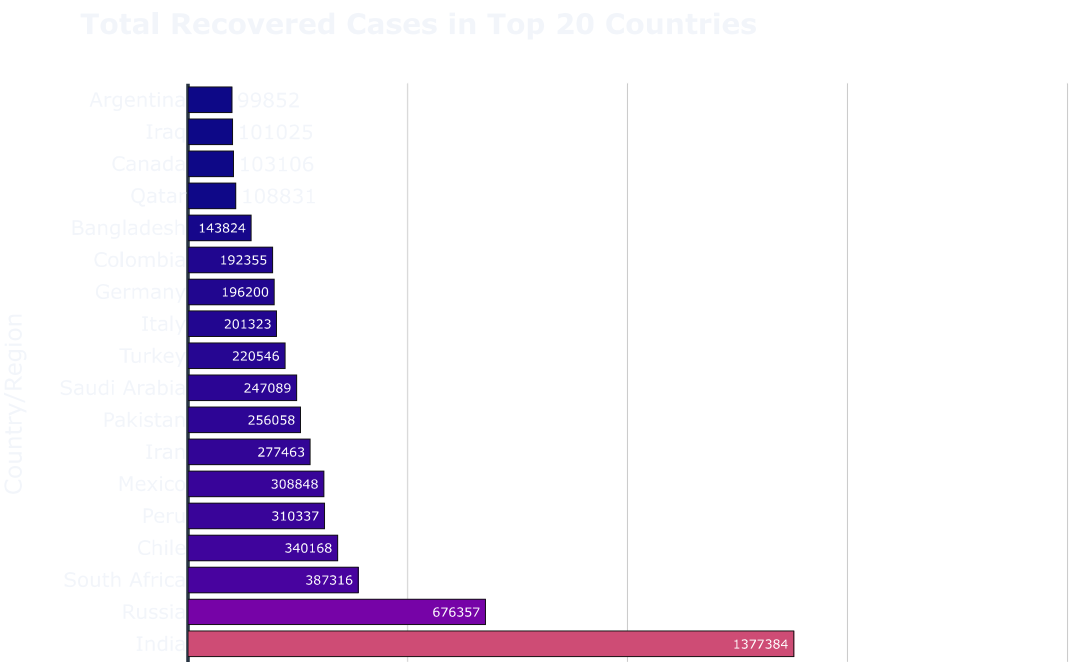
In [31]:

```
fig=px.bar(world_data.sort_values(by='ActiveCases',ascending=False)[0:20], y='Country/Region',x='Act
fig.update_layout(template="plotly_dark",title_text="<b>Total Active Cases in Top 20 Countries")
fig.show()
```



Total Recovered Cases in Top 20 Countries

In [32]:
```
fig=px.bar(world_data.sort_values(by='TotalRecovered',ascending=False)[:20],y='Country/Region',x='To
fig.update_layout(template="plotly_dark",title_text="<b>Total Recovered Cases in Top 20 Countries")
fig.show()
```



In [33]:
```
world_data.columns
```

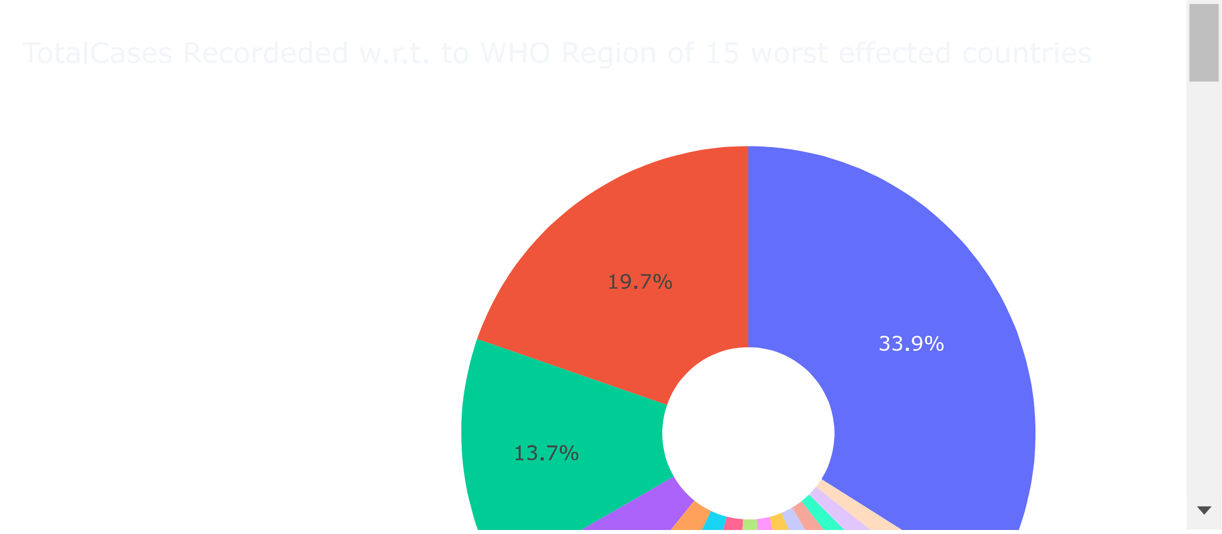
Out[33]:
```
Index(['Country/Region', 'Continent', 'Population', 'TotalCases', 'NewCases',
       'TotalDeaths', 'NewDeaths', 'TotalRecovered', 'NewRecovered',
       'ActiveCases', 'Serious,Critical', 'Tot Cases/1M pop', 'Deaths/1M pop',
       'TotalTests', 'Tests/1M pop', 'WHO Region'],
      dtype='object')
```

In [34]:
```
world_data[0:15]['Country/Region'].values
```

Out[34]:
```
array(['USA', 'Brazil', 'India', 'Russia', 'South Africa', 'Mexico',
       'Peru', 'Chile', 'Colombia', 'Spain', 'Iran', 'UK', 'Saudi Arabia',
       'Pakistan', 'Bangladesh'], dtype=object)
```

Pie Chart Representation of stats of worst affected countries

In [35]:
```
labels=world_data[0:15]['Country/Region'].values
cases=['TotalCases','TotalDeaths','TotalRecovered','ActiveCases']
for i in cases:
    fig=px.pie(world_data[0:15],values=i,names=labels,template="plotly_dark",hole=0.3,title=" {} Rec
    fig.show()
```



Check Death to Confirmed Ratio

In [36]:
```
world_data.head()
```

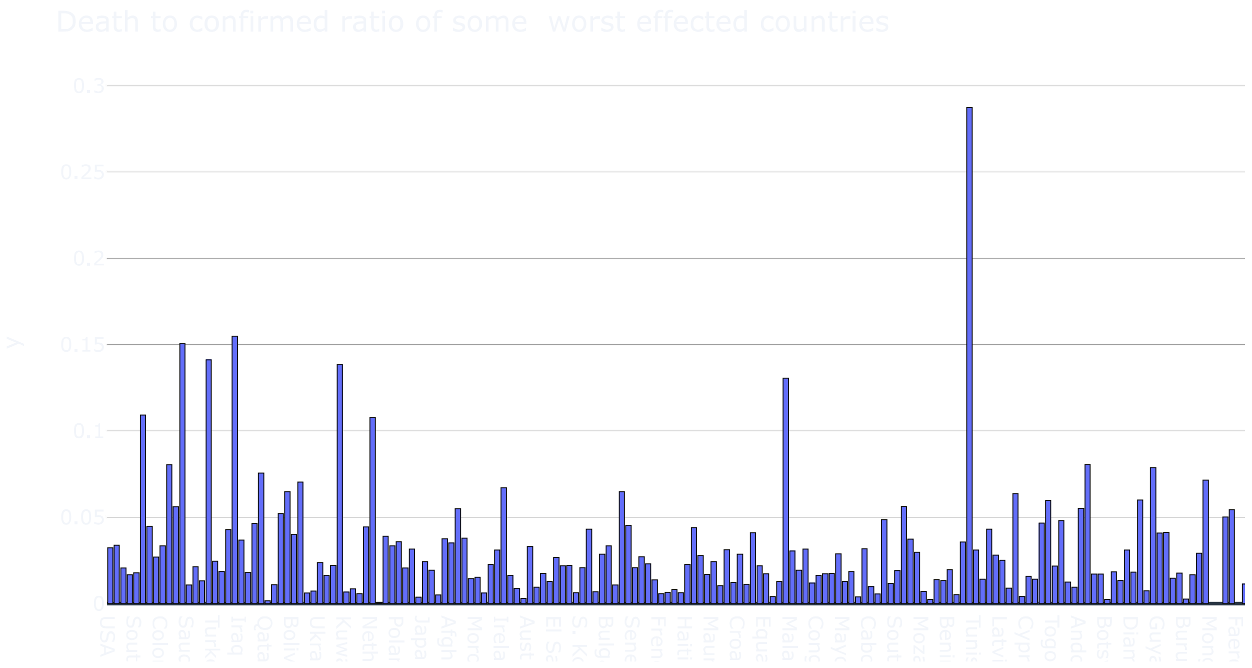
Out[36]:

	Country/Region	Continent	Population	TotalCases	NewCases	TotalDeaths	NewDeaths	TotalRecovered	NewRecovere
0	USA	North America	3.311981e+08	5032179	NaN	162804.0	NaN	2576668.0	Na
1	Brazil	South America	2.127107e+08	2917562	NaN	98644.0	NaN	2047660.0	Na
2	India	Asia	1.381345e+09	2025409	NaN	41638.0	NaN	1377384.0	Na
3	Russia	Europe	1.459409e+08	871894	NaN	14606.0	NaN	676357.0	Na
4	South Africa	Africa	5.938157e+07	538184	NaN	9604.0	NaN	387316.0	Na

In [37]:
```
deaths_to_confirmed = world_data['TotalDeaths']/world_data['TotalCases']
deaths_to_confirmed
```

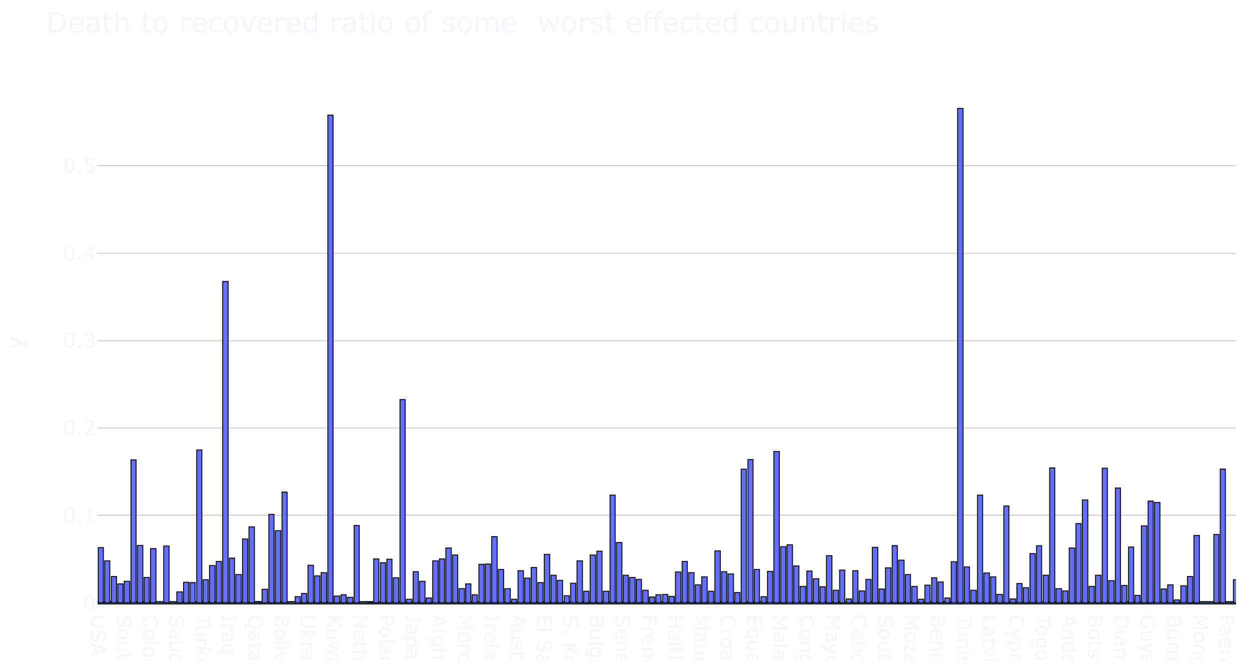
Out[37]:
```
0      0.032353
1      0.033810
2      0.020558
3      0.016752
4      0.017845
         ...
204    0.076923
205         NaN
206         NaN
207         NaN
208    0.100000
Length: 209, dtype: float64
```

In [38]:
```
deaths_to_confirmed=((world_data['TotalDeaths']/world_data['TotalCases']))
fig = px.bar(world_data,x='Country/Region',y=deaths_to_confirmed)
fig.update_layout(title={'text':"Death to confirmed ratio of some  worst effected countries",'xancho
fig.show()
```



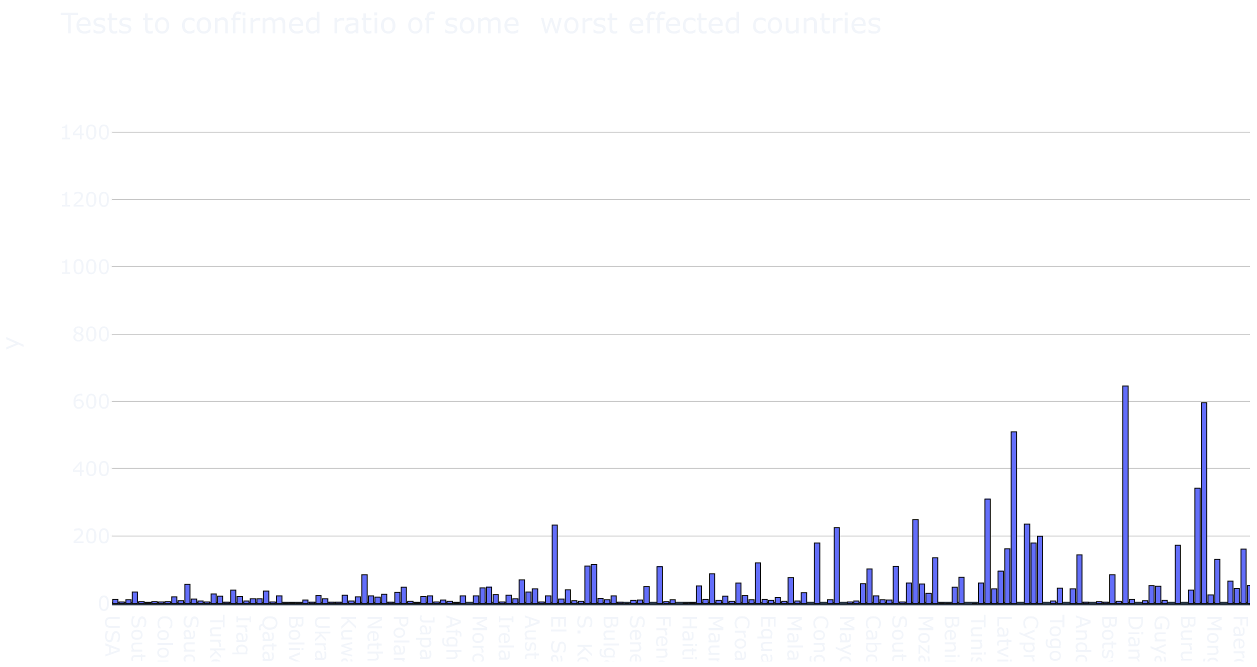
Check Death to Recovered Ratio

In [39]:
```
deaths_to_recovered=((world_data['TotalDeaths']/world_data['TotalRecovered']))
fig = px.bar(world_data,x='Country/Region',y=deaths_to_recovered)
fig.update_layout(title={'text':"Death to recovered ratio of some  worst effected countries",'xancho
fig.show()
```



Show Test to Confirmed Ratio

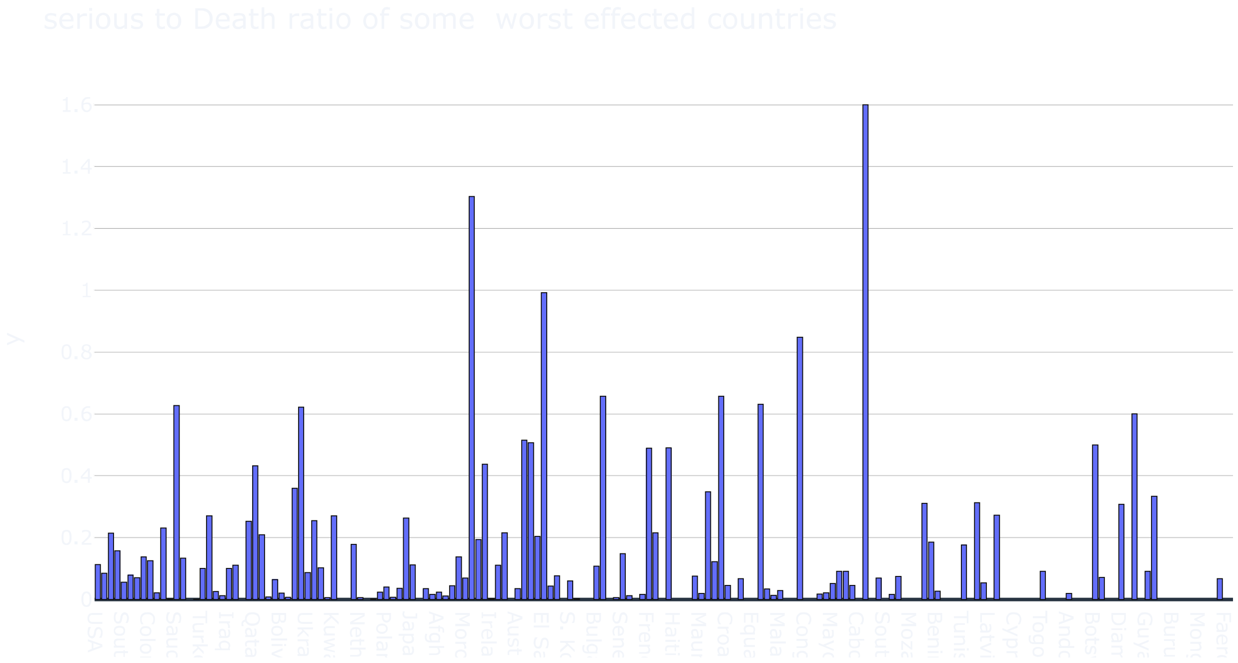
```
In [40]: tests_to_confirmed=((world_data['TotalTests']/world_data['TotalCases']))
         fig = px.bar(world_data,x='Country/Region',y=tests_to_confirmed)
         fig.update_layout(title={'text':"Tests to confirmed ratio of some  worst effected countries",'xancho
         fig.show()
```



Show Serious to Death Ratio

In [41]:

```
serious_to_death=((world_data['Serious,Critical']/world_data['TotalDeaths']))
fig = px.bar(world_data,x='Country/Region',y=serious_to_death)
fig.update_layout(title={'text':"serious to Death ratio of some  worst effected countries",'xanchor'
fig.show()
```



Visulaize Confirmed, Active, Recovered, Deaths Cases of a particular country

In [42]:

```
group_data.head()
```

Out[42]:

	Date	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	WHO Region
0	2020-01-22	Afghanistan	0	0	0	0	0	0	0	Eastern Mediterranean
1	2020-01-22	Albania	0	0	0	0	0	0	0	Europe
2	2020-01-22	Algeria	0	0	0	0	0	0	0	Africa
3	2020-01-22	Andorra	0	0	0	0	0	0	0	Europe
4	2020-01-22	Angola	0	0	0	0	0	0	0	Africa

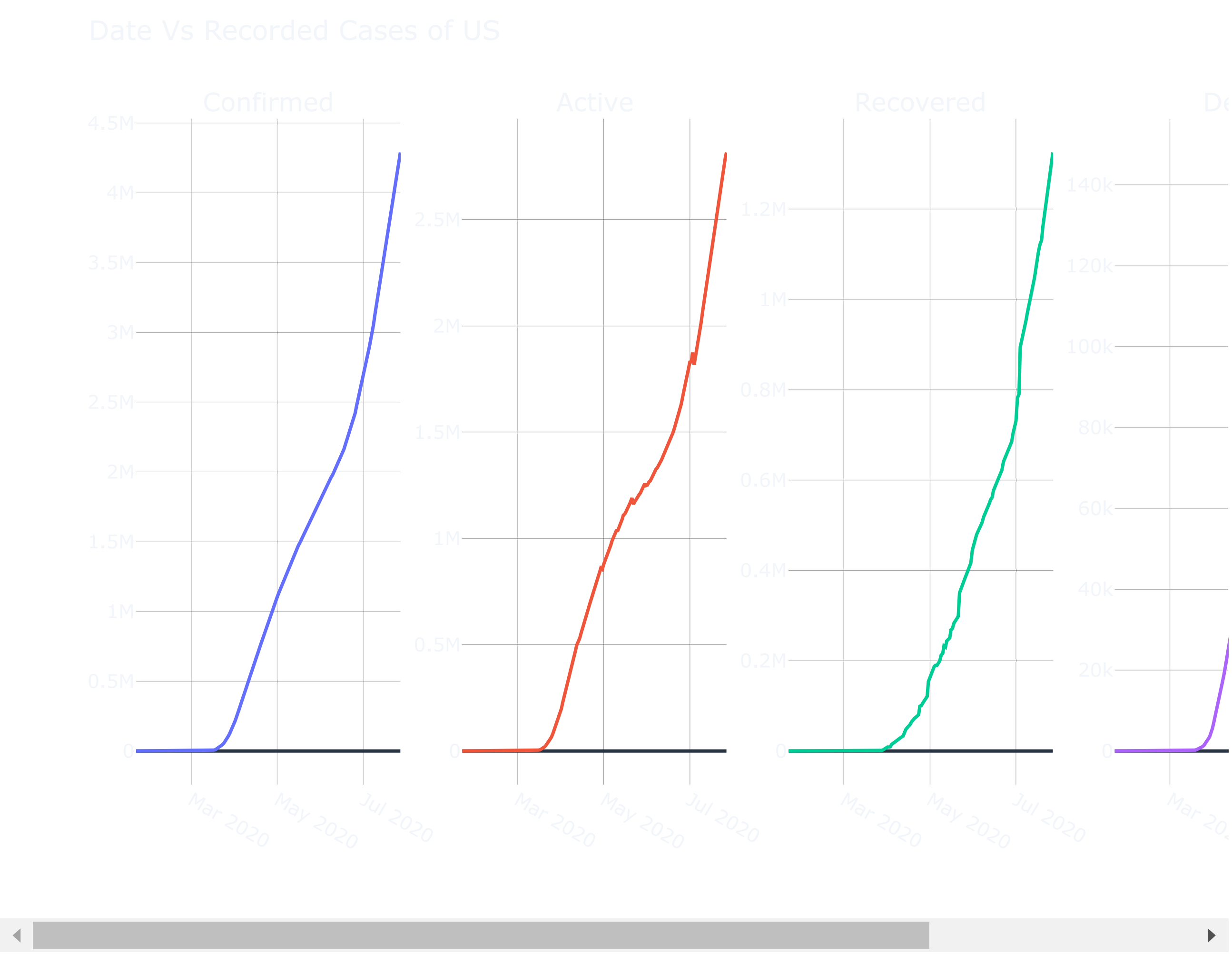
In [43]:

```
from plotly.subplots import make_subplots  ## for creating subplots in plotly
import plotly.graph_objects as go
```

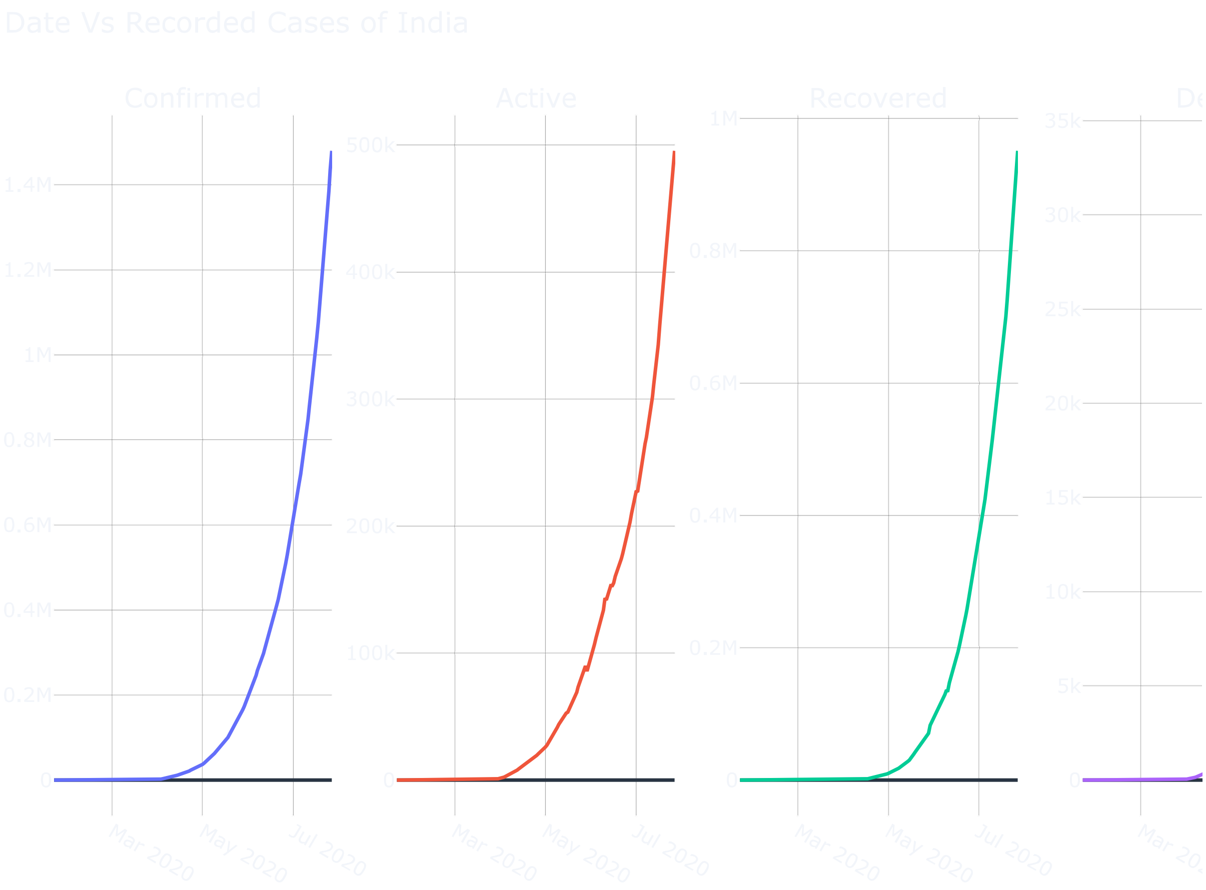
In [44]:

```
def country_visualization(group_data,country):

    data=group_data[group_data['Country/Region']==country]
    df=data.loc[:,['Date','Confirmed','Deaths','Recovered','Active']]
    fig = make_subplots(rows=1, cols=4,subplot_titles=("Confirmed", "Active", "Recovered",'Deaths'))
    fig.add_trace(
        go.Scatter(name="Confirmed",x=df['Date'],y=df['Confirmed']),
        row=1, col=1
    )

    fig.add_trace(
        go.Scatter(name="Active",x=df['Date'],y=df['Active']),
        row=1, col=2
    )
    fig.add_trace(
        go.Scatter(name="Recovered",x=df['Date'],y=df['Recovered']),
        row=1, col=3
    )

    fig.add_trace(
        go.Scatter(name="Deaths",x=df['Date'],y=df['Deaths']),
        row=1, col=4
    )

    fig.update_layout(height=600, width=1000, title_text="Date Vs Recorded Cases of {}".format(count
    fig.show()
```

```
In [45]: country_visualization(group_data,'US')
```



In [46]:
```
country_visualization(group_data,'India')
```



In [ ]: