

COVID-19

DATA ANALYSIS USING PYTHON



A Jupyter Notebook Analysis of the COVID-19 Pandemic

Data Import & Cleaning

- Loading Datasets
- Handling Missing Values



Exploratory Data Analysis

- Global Trends
- Case & Death Rates



Visualizations

- Time Series Plots
- Choropleth Maps
- Growth Charts



Key Insights & Predictions

- Mortality Analysis
- Forecasting Spread



Pandas



NumPy



Matplotlib



Plotly

PYTHON DATA SCIENCE PROJECT

In [2]:

```
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib import style
style.use('ggplot')
%matplotlib inline

import plotly
import plotly.express as px
import plotly.graph_objects as go
# plt.rcParams['figure.figsize']=17,8
import cufflinks as cf
import plotly.offline as pyo
from plotly.offline import init_notebook_mode,plot,iplot

import folium
```

In [5]:

```
pyo.init_notebook_mode(connected=True)
cf.go_offline()
```

In [6]:

```
df=pd.read_excel("C:/Users/Dell Laptop/Downloads/Covid cases in India.xlsx")
```

In [7]:

```
df
```

Out[7]:

S. No.	Name of State / UT	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death
0	1 Andhra Pradesh	12	0	1	0
1	2 Chhattisgarh	6	0	0	0
2	3 Delhi	38	1	6	1
3	4 Gujarat	43	0	0	3
4	5 Haryana	16	14	11	0
5	6 Himachal Pradesh Karnataka	4	0	0	1
6	7 Kerala	20	0	3	2
7	8 Madhya Pradesh	131	7	11	0
8	9 Maharashtra	23	0	0	1
9	10 Odisha	144	3	15	4
10	11 Puducherry	3	0	0	0
11	12 Punjab	1	0	0	0
12	13 Rajasthan	29	0	0	1
13	14 Tamil Nadu	41	2	3	0
14	15 Telengana	32	3	1	1
15	16	34	11	1	0

S. No.	Name of State / UT	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death
16	17 Chandigarh	7	0	0	0
17	18 Jammu and Kashmir Ladakh	18	0	1	1
18	19 Uttar Pradesh	13	0	0	0
19	20 Uttarakhand	42	1	11	0
20	21 West Bengal	4	0	0	0
21	22 Bihar	11	0	0	1
22	23 Mizoram	7	0	0	1
23	24 Goa	1	0	0	0
24	25 Manipur	6	0	0	0
25	26	1	0	0	0

In [8]:

```
df.drop(['S. No.'],axis=1,inplace=True)
```

In [9]:

```
df
```

Out[9]:

	Name of State / UT	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death
0	Andhra Pradesh	12	0	1	0
1	Chhattisgarh	6	0	0	0
2	Delhi	38	1	6	1
3	Gujarat	43	0	0	3
4	Haryana	16	14	11	0
5	Himachal Pradesh	4	0	0	1
6	Karnataka	20	0	3	2
7	Kerala	131	7	11	0
8	Madhya Pradesh	23	0	0	1
9	Maharashtra	144	3	15	4
10	Odisha	3	0	0	0
11	Puducherry	1	0	0	0
12	Punjab	29	0	0	1
13	Rajasthan	41	2	3	0
14	Tamil Nadu	32	3	1	1
15	Telengana	34	11	1	0
16	Chandigarh	7	0	0	0

	Name of State / UT	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death
17	Jammu and Kashmir	18		0	1 1
	Ladakh				
18	Uttar Pradesh	13		0	0 0
19	Uttarakhand	42		1	11 0
20	West Bengal	4		0	0 0
21	Bihar	11		0	0 1
22	Mizoram	7		0	0 1
23	Goa	1		0	0 0
24	Manipur	6		0	0 0
25		1		0	0 0

In [10]:

```
df['Total Cases']=df['Total Confirmed cases (Indian National)']+df['Total Confirmed case']
```

In [11]:

```
df
```

Out[11]:

	Name of State / UT	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death	Total Cases
0	Andhra Pradesh	12		0	1 0	12
1	Chhattisgarh	6		0	0 0	6
2	Delhi	38		1	6 1	39
3	Gujarat	43		0	0 3	43
4	Haryana	16		14	11 0	30
5	Himachal Pradesh			0	0 1	4
	Karnataka	4				
6	Kerala	20		0	3 2	20
7	Madhya Pradesh	131		7	11 0	138
8	Maharashtra	23		0	0 1	23
9	Odisha	144		3	15 4	147
10	Puducherry	3		0	0 0	3
11	Punjab	1		0	0 0	1
12	Rajasthan	29		0	0 1	29
13	Tamil Nadu	41		2	3 0	43
14	Telengana	32		3	1 1	35
15	Chandigarh	34		11	1 0	45
16		7		0	0 0	7

Name of State / UT	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death	Total Cases
17 Jammu and Kashmir	18	0	1	1	18
18 Ladakh					
18 Uttar Pradesh	13	0	0	0	13
19 Uttarakhand	42	1	11	0	43
20 West Bengal	4	0	0	0	4
21 Bihar	11	0	0	1	11
22 Mizoram	7	0	0	1	7
23 Goa	1	0	0	0	1
24 Manipur	6	0	0	0	6
25	1	0	0	0	1

In [12]:

```
total_cases_overall=df['Total Cases'].sum()
print('The total number of cases till now in India is ',total_cases_overall)
```

The total number of cases till now in India is 729

In [13]:

```
df['Active Cases']=df['Total Cases']-(df['Death']+df['Cured'])
```

In [14]:

```
df
```

Out[14]:

Name of State / UT	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death	Total Cases	Active Cases
0 Andhra Pradesh	12	0	1	0	12	11
Chhattisgarh						
1 Delhi	6	0	0	0	6	6
2 Gujarat	38	1	6	1	39	32
3 Haryana	43	0	0	3	43	40
4 Himachal Pradesh	16	14	11	0	30	19
5 Karnataka	4	0	0	1	4	3
6 Kerala	20	0	3	2	20	15
7 Madhya Pradesh	131	7	11	0	138	127
8 Maharashtra	23	0	0	1	23	22
9 Odisha	144	3	15	4	147	128
10 Puducherry	3	0	0	0	3	3
11 Punjab	1	0	0	0	1	1
12	29	0	0	1	29	28

Name of State / UT	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death	Total Cases	Active Cases
13 Rajasthan	41	2	3	0	43	40
14 Tamil Nadu	32	3	1	1	35	33
15 Telengana	34	11	1	0	45	44
16 Chandigarh	7	0	0	0	7	7
Jammu and Kashmir Ladakh	18	0	1	1	18	16
18 Uttar Pradesh	13	0	0	0	13	13
19 Uttarakhand	42	1	11	0	43	32
20 West Bengal	4	0	0	0	4	4
21 Bihar	11	0	0	1	11	10
22 Mizoram	7	0	0	1	7	6
23 Goa	1	0	0	0	1	1
24 Manipur	6	0	0	0	6	6
25	1	0	0	0	1	1

In [15]:

```
df.style.background_gradient(cmap='Reds')
```

Out[15]:

Name of State / UT	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death	Total Cases	Active Cases
0 Andhra Pradesh	12	0	1	0	12	11
Chhattisgarh						
1 Delhi	6	0	0	0	6	6
2 Gujarat	38	1	6	1	39	32
3 Haryana	43	0	0	3	43	40
4 Himachal	16	14	11	0	30	19
Pradesh						
5 Karnataka	4	0	0	1	4	3
6 Kerala	20	0	3	2	20	15
7 Madhya Pradesh	131	7	11	0	138	127
8 Maharashtra	23	0	0	1	23	22
9 Odisha	144	3	15	4	147	128
Puducherry	3	0	0	0	3	3
11 Punjab	1	0	0	0	1	1
12 Rajasthan	29	0	0	1	29	28
13	41	2	3	0	43	40

	Name of State / UT	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death	Total Cases	Active Cases
14	Tamil Nadu	32	3	1	1	35	33
15	Telengana	34	11	1	0	45	44
16	Chandigarh	7	0	0	0	7	7
17	Jammu and Kashmir	18	0	1	1	18	16
18	Ladakh	13	0	0	0	13	13
19	Uttar Pradesh	42	1	11	0	43	32
20	West Bengal	4	0	0	0	4	4
21	Bihar	11	0	0	1	11	10
22	Mizoram	7	0	0	1	7	6
23	Goa	1	0	0	0	1	1
24	Manipur	6	0	0	0	6	6
25		1	0	0	0	1	1

In [16]:

```
Total_Active_Cases = df.groupby('Name of State / UT')['Total Cases'].sum().sort_values(a
```

In [17]:

```
Total_Active_Cases
```

Out[17]:

Total Cases

Name of State / UT	Total Cases
Maharashtra	147
Kerala	138
Telengana	45
Rajasthan	43
Gujarat	43
Uttar Pradesh	43
Delhi	39
Tamil Nadu	35
Haryana	30
Punjab	29
Madhya Pradesh	23
Karnataka	20
Jammu and Kashmir	18
Ladakh	13
Andhra Pradesh	12

Total Cases

Name of State / UT

West Bengal	11
Bihar	7
Chandigarh	7
Chhattisgarh	6
Goa	6
Himachal Pradesh	4
Uttarakhand	4
Odisha	3
Manipur	1
Mizoram	1
Puducherry	1

In [18]:

```
Total_Active_Cases.style.background_gradient(cmap='Reds')
```

Out[18]:

Total Cases

Name of State / UT

Maharashtra	147
Kerala	138
Telengana	45
Rajasthan	43
Gujarat	43
Uttar Pradesh	43
Delhi	39
Tamil Nadu	35
Haryana	30
Punjab	29
Madhya Pradesh	23
Karnataka	20
Jammu and Kashmir	18
Ladakh	13
Andhra Pradesh	12
West Bengal	11
Bihar	7
Chandigarh	7

Total Cases

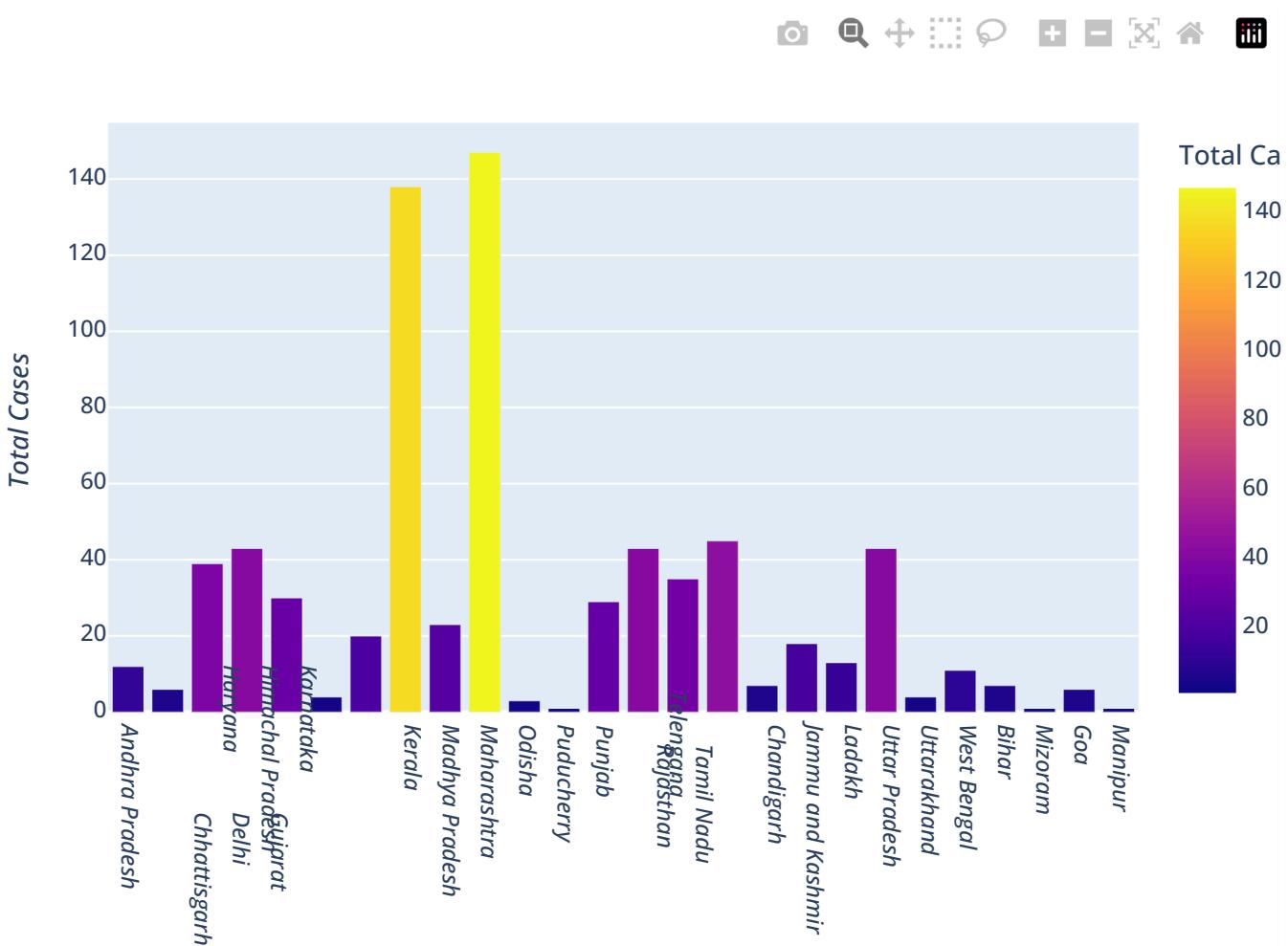
Name of State / UT	
Chhattisgarh	6
Goa	6
Himachal Pradesh	4
Uttarakhand	4
Odisha	3
Manipur	1
Mizoram	1
Puducherry	1

In [19]:

Graphical Representation

In [20]:

```
fig = px.bar(df, x='Name of State / UT', y='Total Cases', color='Total Cases')
fig.show()
```

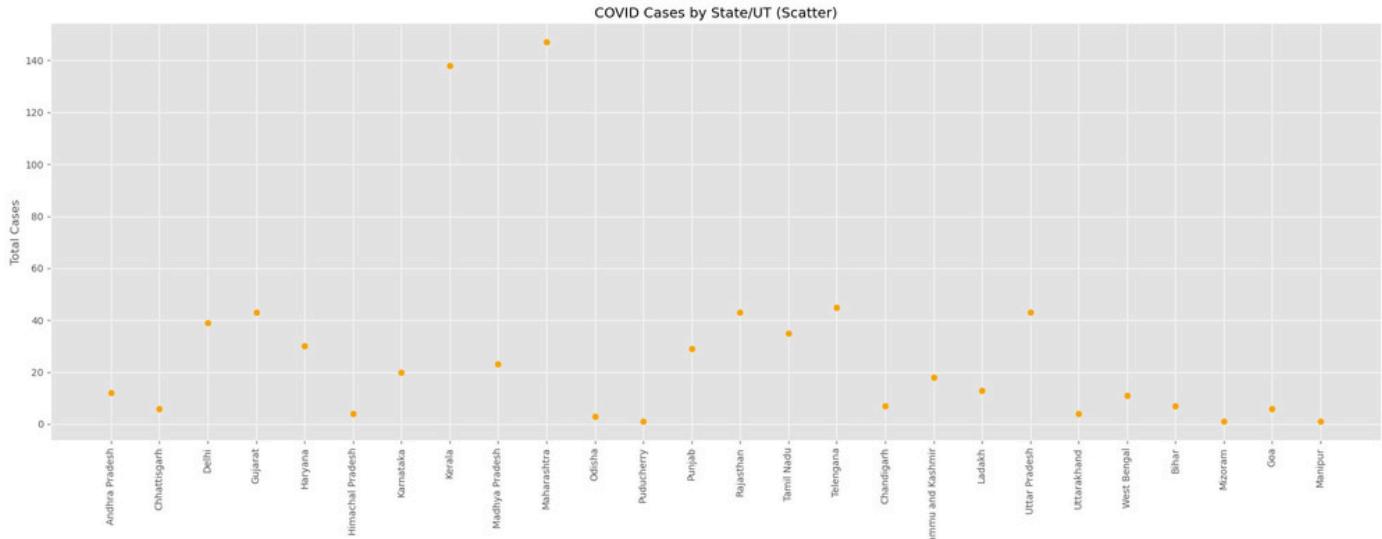


In [21]:

```

plt.figure(figsize=(20,8))
plt.scatter(df['Name of State / UT'], df['Total Cases'], color='orange')
plt.xticks(rotation=90, ha='center') # labels vertical
plt.ylabel("Total Cases")
plt.title("COVID Cases by State/UT (Scatter)")
plt.tight_layout()
plt.show()

```



In [22]:

```
Indian_Cord=pd.read_excel("C:/Users/Dell Laptop/Downloads/Indian Coordinates.xlsx")
```

In [23]:

```
Indian_Cord
```

Out[23]:

	Name of State / UT	Latitude	Longitude
0	Andaman And Nicobar	11.667026	92.735983
1	Andhra Pradesh	14.750429	78.570026
2	Arunachal Pradesh	27.100399	93.616601
3	Assam	26.749981	94.216667
4	Bihar	25.785414	87.479973
5	Chandigarh	30.719997	76.780006
6	Chhattisgarh	22.090420	82.159987
7	Dadra And Nagar Haveli	20.266578	73.016618
8	Delhi	28.669993	77.230004
9	Goa	15.491997	73.818001
10	Haryana	28.450006	77.019991
11	Himachal Pradesh	31.100025	77.166597
12	Union Territory of Jammu and Kashmir	33.450000	76.240000
13	Jharkhand	23.800393	86.419986

	Name of State / UT	Latitude	Longitude
14	Karnataka	12.570381	76.919997
15	Kerala	8.900373	76.569993
16	Lakshadweep	10.562573	72.636867
17	Madhya Pradesh	21.300391	76.130019
18	Maharashtra	19.250232	73.160175
19	Manipur	24.799971	93.950017
20	Meghalaya	25.570492	91.880014
21	Mizoram	23.710399	92.720015
22	Nagaland	25.666998	94.116570
23	Orissa	19.820430	85.900017
24	Puducherry	11.934994	79.830000
25	Punjab	31.519974	75.980003
26	Rajasthan	26.449999	74.639981
27	Sikkim	27.333330	88.616647
28	Telengana	18.112400	79.019300
29	Tamil Nadu	12.920386	79.150042
30	Tripura	23.835404	91.279999
31	Uttar Pradesh	27.599981	78.050006
32	Uttarakhand	30.320409	78.050006
33	West Bengal	22.580390	88.329947
34	Union Territory of Ladakh	34.100000	77.340000

In [24]:

```
df_full=pd.merge(Indian_Cord,df,on='Name of State / UT')
```

In [25]:

```
df_full
```

Out[25]:

	Name of State / UT	Latitude	Longitude	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death	Total Cases	Active Cases
0	Andhra Pradesh	14.750429	78.570026	12	0	1	0	12	11
1	Delhi	28.669993	77.230004	38	1	6	1	39	32
2	Haryana	28.450006	77.019991	16	14	11	0	30	19
3	Karnataka	12.570381	76.919997	20	0	3	2	20	15
4	Kerala	8.900373	76.569993	131	7	11	0	138	127

Name of State / UT	Latitude	Longitude	Total Confirmed cases (Indian National)	Total Confirmed cases (Foreign National)	Cured	Death	Total Cases	Active Cases
5 Maharashtra	19.250232	73.160175	144	3	15	4	147	128
6 Punjab	31.519974	75.980003	29	0	0	1	29	28
7 Rajasthan	26.449999	74.639981	41	2	3	0	43	40
8 Telengana	18.112400	79.019300	34	11	1	0	45	44
9 Tamil Nadu	12.920386	79.150042	32	3	1	1	35	33
10 Uttar Pradesh	27.599981	78.050006	42	1	11	0	43	32
11 Uttarakhand	30.320409	78.050006	4	0	0	0	4	4

In [36]:

```
m = folium.Map(
    location=[22.5, 78.9],           #India center
    zoom_start=5,
    tiles='OpenStreetMap'
)

# Add circles
for lat, lon, value, name in zip(
    df_full['Latitude'],
    df_full['Longitude'],
    df_full['Total Cases'],
    df_full['Name of State / UT']
):
    folium.CircleMarker(
        location=[lat, lon],
        radius=value / 30,      # adjust size
        popup=f"<b>State:</b> {name}<br><b>Total Cases:</b> {value}",
        color='red',
        fill=True,
        fill_color='red',
        fill_opacity=0.4
    ).add_to(m)
m
```

Out[36]:

Make this Notebook Trusted to load map: File -> Trust Notebook

In [31]:

```
dbd_India=pd.read_excel(r"C:\Users\DELL Laptop\Downloads\per_day_cases.xlsx",parse_dates=True)
dbd_Italy=pd.read_excel(r"C:\Users\DELL Laptop\Downloads\per_day_cases.xlsx",parse_dates=True)
dbd_korea=pd.read_excel(r"C:\Users\DELL Laptop\Downloads\per_day_cases.xlsx",parse_dates=True)
dbd_Wuhan=pd.read_excel(r"C:\Users\DELL Laptop\Downloads\per_day_cases.xlsx",parse_dates=True)
```

In [32]:

```
dbd_India
```

Out[32]:

	Date	Total Cases	New Cases	Days after surpassing 100 cases
0	2020-01-30	1	1	NaN
1	2020-01-31	1	0	NaN
2	2020-02-01	1	0	NaN
3	2020-02-02	2	1	NaN
4	2020-02-03	3	1	NaN
5	2020-02-04	3	0	NaN
6	2020-02-05	3	0	NaN
7	2020-02-06	3	0	NaN
8	2020-02-07	3	0	NaN
9	2020-02-08	3	0	NaN
10	2020-02-09	3	0	NaN
11	2020-02-10	3	0	NaN
12	2020-02-11	3	0	NaN

	Date	Total Cases	New Cases	Days after surpassing 100 cases
13	2020-02-12	3	0	NaN
14	2020-02-13	3	0	NaN
15	2020-02-14	3	0	NaN
16	2020-02-15	3	0	NaN
17	2020-02-16	3	0	NaN
18	2020-02-17	3	0	NaN
19	2020-02-18	3	0	NaN
20	2020-02-19	3	0	NaN
21	2020-02-20	3	0	NaN
22	2020-02-21	3	0	NaN
23	2020-02-22	3	0	NaN
24	2020-02-23	3	0	NaN
25	2020-02-24	3	0	NaN
26	2020-02-25	3	0	NaN
27	2020-02-26	3	0	NaN
28	2020-02-27	3	0	NaN
29	2020-02-28	3	0	NaN
30	2020-02-29	3	0	NaN
31	2020-03-01	3	0	NaN
32	2020-03-02	6	3	NaN
33	2020-03-03	9	3	NaN
34	2020-03-04	28	19	NaN
35	2020-03-05	30	2	NaN
36	2020-03-06	31	1	NaN
37	2020-03-07	34	3	NaN
38	2020-03-08	39	5	NaN
39	2020-03-09	43	4	NaN
40	2020-03-10	56	13	NaN
41	2020-03-11	62	6	NaN
42	2020-03-12	73	11	NaN
43	2020-03-13	82	9	NaN
44	2020-03-14	102	20	0.0
45	2020-03-15	113	11	1.0
46	2020-03-16	119	6	2.0
47	2020-03-17	142	23	3.0

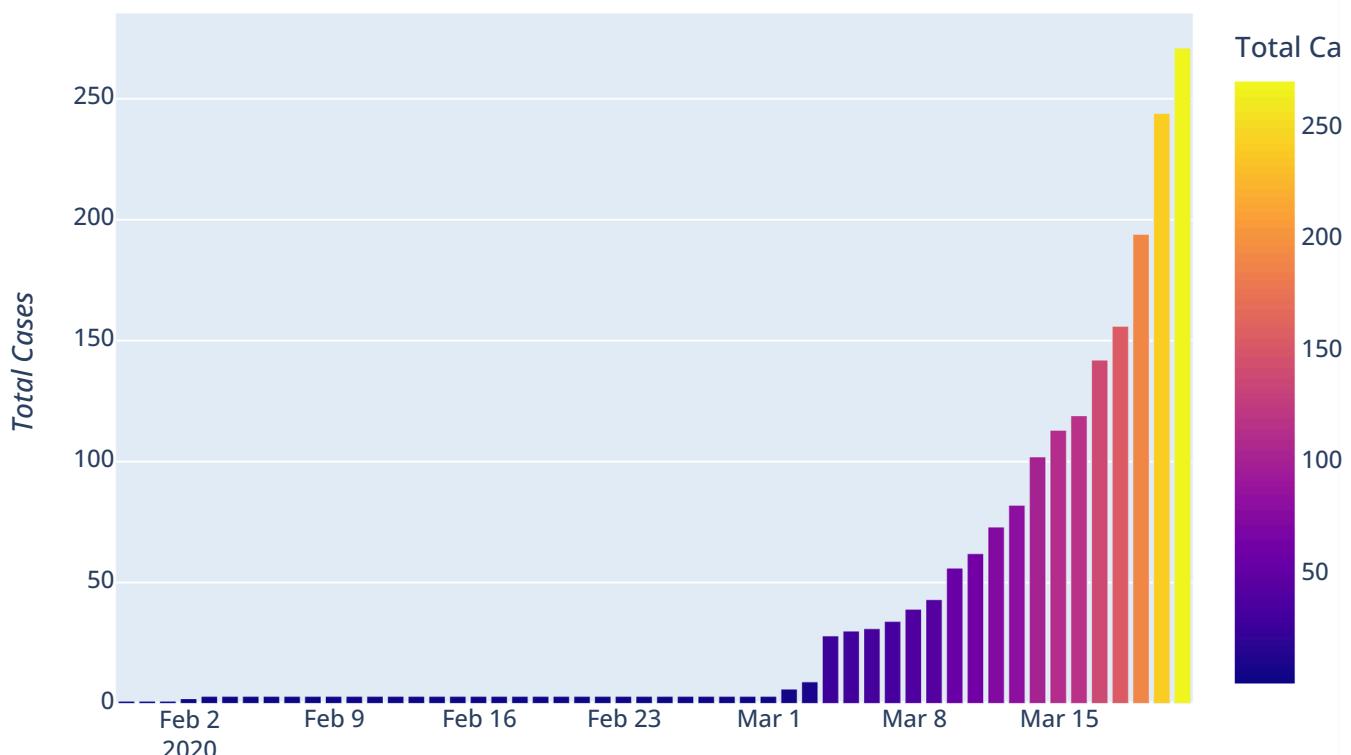
	Date	Total Cases	New Cases	Days after surpassing 100 cases
48	2020-03-18	156	14	4.0
49	2020-03-19	194	38	5.0
50	2020-03-20	244	50	6.0
51	2020-03-21	271	27	7.0

In [33]:

```
fig=px.bar(dbd_India,x="Date",y="Total Cases",color='Total Cases',title='Confirmed cases'
fig.show()
```



Confirmed cases in India



In [34]:

```
fig = px.bar(dbd_Italy, x="Date", y="Total Cases",
              color="Total Cases",
              title="Confirmed cases in Italy")
fig.show()
```

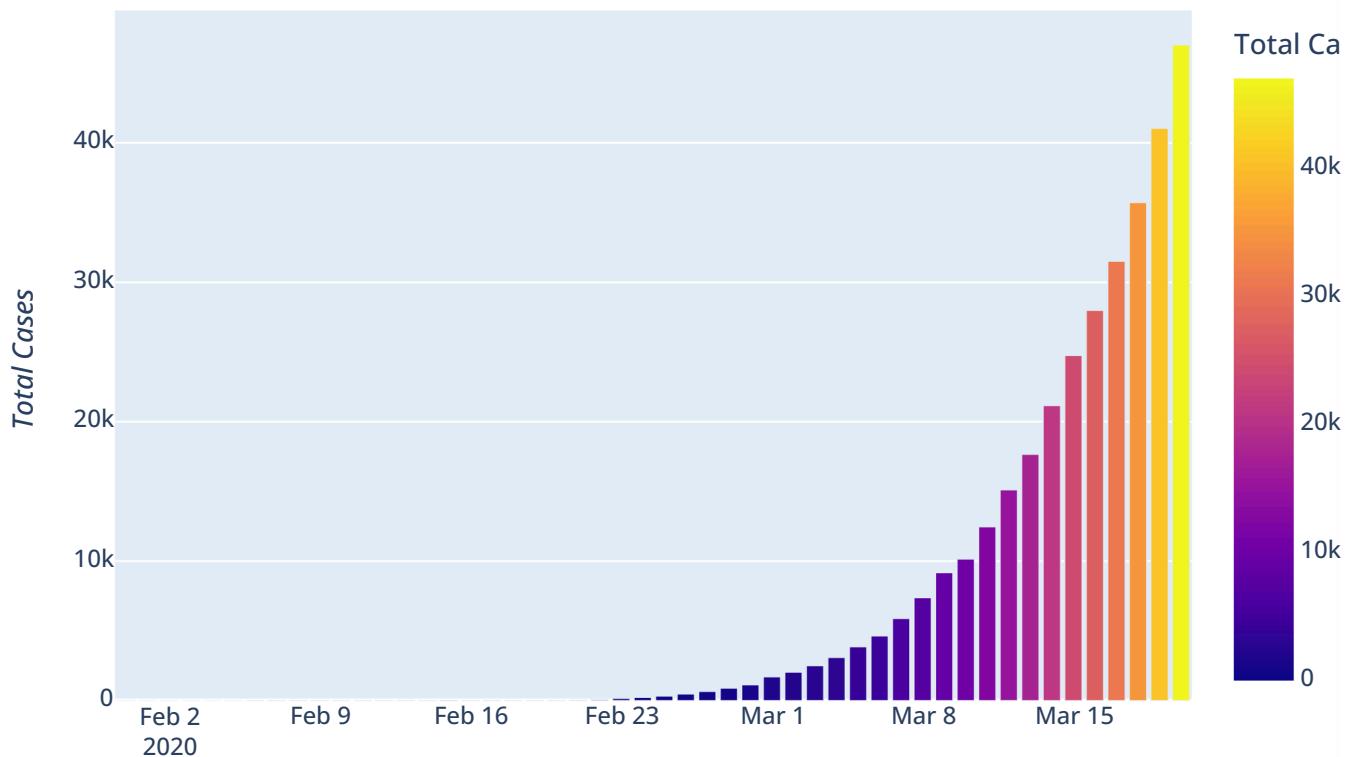
```
fig = px.bar(dbd_korea, x="Date", y="Total Cases",
              color="Total Cases",
              title="Confirmed cases in Korea")
fig.show()
```

```
fig = px.bar(dbd_Wuhan, x="Date", y="Total Cases",
              color="Total Cases",
```

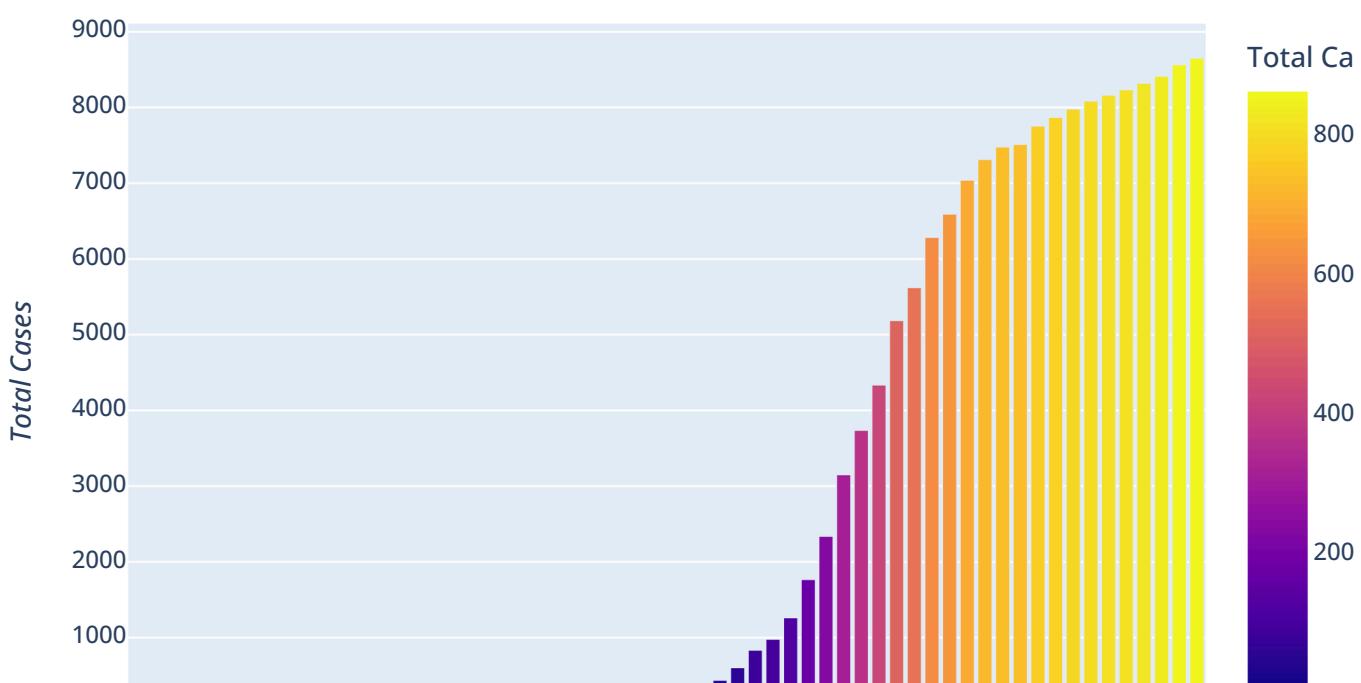
```
title="Confirmed cases in Wuhan")  
fig.show()
```

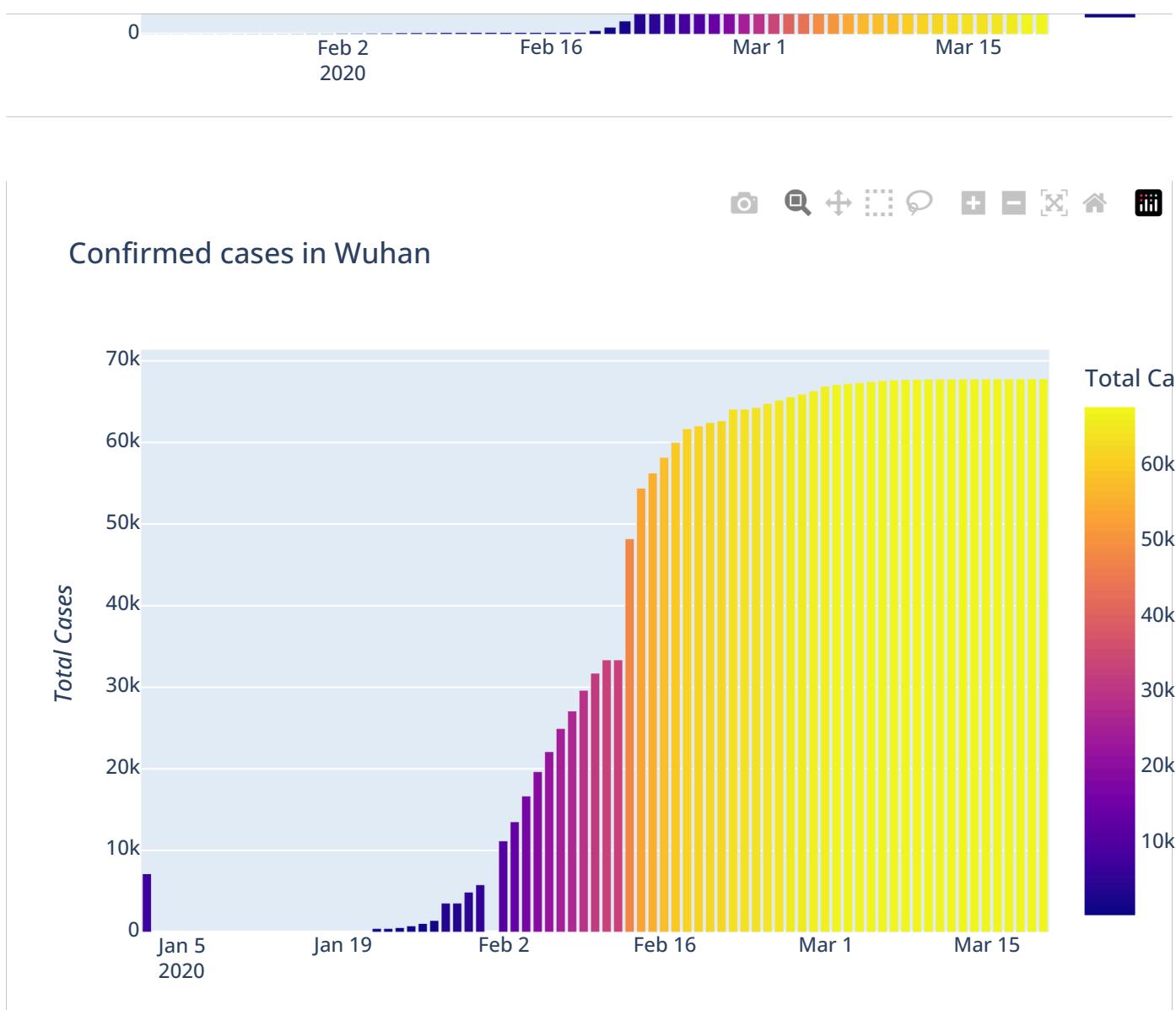


Confirmed cases in Italy



Confirmed cases in Korea

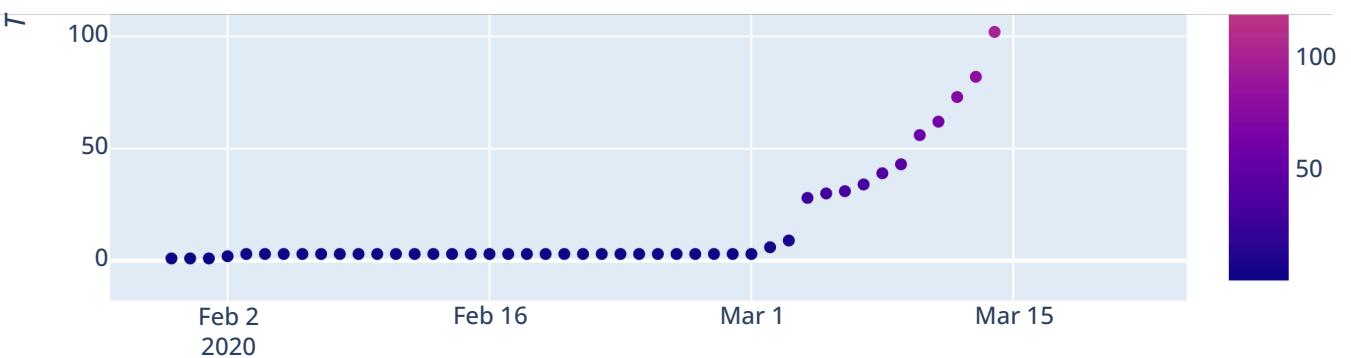




In [35]:

```
fig=px.scatter(dbd_India,x="Date",y="Total Cases",color='Total Cases',title='Confirmed c  
fig.show()
```





In [36]:

```
from plotly.subplots import make_subplots
```

In [41]:

```
fig=make_subplots(
    rows=2,cols=2,
    specs=[[{"secondary_y":True}, {"secondary_y":True}], [{"secondary_y":True}, {"secondary_y":True}],
    subplot_titles=("S.Korea","Italy","India","Wuhan"))

fig.add_trace(go.Bar(x=dbd_Korea['Date'],y=dbd_Korea['Total Cases'],
                     marker=dict(color=dbd_Korea['Total Cases'],coloraxis="coloraxis")),1

fig.add_trace(go.Bar(x=dbd_Italy['Date'],y=dbd_Italy['Total Cases'],
                     marker=dict(color=dbd_Italy['Total Cases'],coloraxis="coloraxis")),1

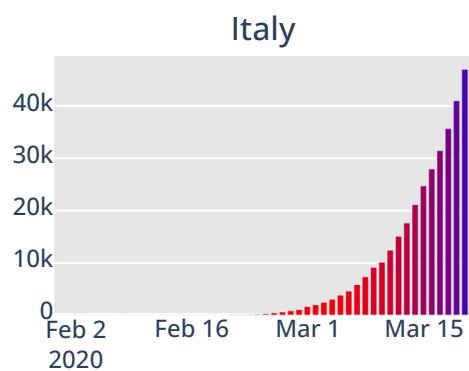
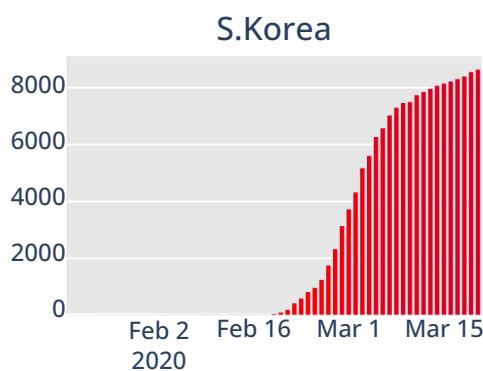
fig.add_trace(go.Bar(x=dbd_India['Date'],y=dbd_India['Total Cases'],
                     marker=dict(color=dbd_India['Total Cases'],coloraxis="coloraxis")),2

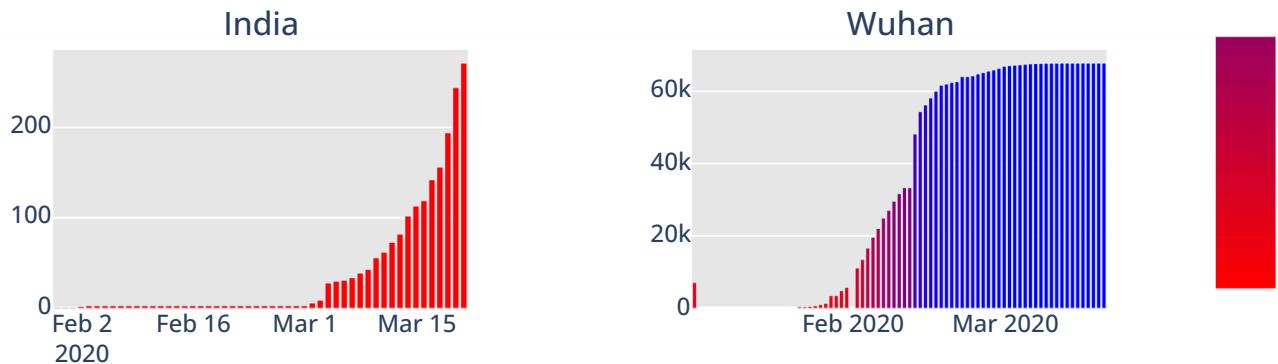
fig.add_trace(go.Bar(x=dbd_Wuhan['Date'],y=dbd_Wuhan['Total Cases'],
                     marker=dict(color=dbd_Wuhan['Total Cases'],coloraxis="coloraxis")),2

fig.update_layout(coloraxis=dict(colorscale='Bluered_r'),showlegend=False,title_text="To
fig.update_layout(plot_bgcolor='rgb(230,230,230)')
```



Total Cases in 4 Countries





In [43]:

```
df=pd.read_csv("C:/Users/Dell Laptop/Downloads/covid_19_data.csv",parse_dates=['Last Upd
```

In [44]:

```
df.rename(columns={'ObservationDate':'Date','Country/Region':'Country'},inplace=True)
```

In [45]:

```
df
```

Out[45]:

SNo	Date	Province/State	Country	Last Update	Confirmed	Deaths	Recovered
0	1 01/22/2020	Anhui	Mainland China	1/22/2020 17:00	1.0	0.0	0.0
1	2 01/22/2020	Beijing	Mainland China	1/22/2020 17:00	14.0	0.0	0.0
2	3 01/22/2020	Chongqing	Mainland China	1/22/2020 17:00	6.0	0.0	0.0
3	4 01/22/2020	Fujian	Mainland China	1/22/2020 17:00	1.0	0.0	0.0
4	5 01/22/2020	Gansu	0.0	0.0	0.0
...	2020-03-14T16:33:03
5885	5886 03/15/2020	Gibraltar	UK	2020-03-15T18:20:19	1.0	0.0	1.0
5886	5887 03/15/2020	Nan	Uzbekistan	2020-03-14T02:33:04	1.0	0.0	0.0
5887	5888 03/15/2020	Diamond Princess cruise ship	Australia	2020-03-10T02:33:04	0.0	0.0	0.0
5888	5889 03/15/2020	West Virginia	US		0.0	0.0	0.0
5889	5890 03/15/2020	Nan	occupied Palestinian territory	2020-03-11T20:53:02	0.0	0.0	0.0

5890 rows × 8 columns

In [46]:

```
df.query('Country=="UK"')
```

Out[56]:

SNo	Date	Province/State	Country	Last Update	Confirmed	Deaths	Recovered	
481	482	01/31/2020	NaN	UK	1/31/2020 23:59	2.0	0.0	0.0
541	542	02/01/2020	NaN	UK	2/1/2020 1:52	2.0	0.0	0.0
611	612	02/02/2020	NaN	UK	2020-02-01T01:52:40	2.0	0.0	0.0
678	679	02/03/2020	NaN	UK	2020-02-01T01:52:40	2.0	0.0	0.0
746	747	02/04/2020	NaN	UK	2020-02-01T01:52:40	2.0	0.0	0.0
817	818	02/05/2020	NaN	UK	2020-02-01T01:52:40	2.0	0.0	0.0
888	889	02/06/2020	NaN	UK	2020-02-01T01:52:40	2.0	0.0	0.0
957	958	02/07/2020	NaN	UK	2020-02-07T18:33:02	3.0	0.0	0.0
1029	1030	02/08/2020	NaN	UK	2020-02-07T18:33:02	3.0	0.0	0.0
1101	1102	02/09/2020	NaN	UK	2020-02-07T18:33:02	3.0	0.0	0.0
1165	1166	02/10/2020	NaN	UK	2020-02-10T19:03:02	8.0	0.0	0.0
1237	1238	02/11/2020	NaN	UK	2020-02-10T19:03:02	8.0	0.0	0.0
1310	1311	02/12/2020	NaN	UK	2020-02-12T19:03:06	9.0	0.0	1.0
1383	1384	02/13/2020	NaN	UK	2020-02-12T19:03:06	9.0	0.0	1.0
1457	1458	02/14/2020	NaN	UK	2020-02-12T19:03:06	9.0	0.0	1.0
1532	1533	02/15/2020	NaN	UK	2020-02-12T19:03:06	9.0	0.0	1.0
1607	1608	02/16/2020	NaN	UK	2020-02-16T16:23:06	9.0	0.0	8.0
1682	1683	02/17/2020	NaN	UK	2020-02-16T16:23:06	9.0	0.0	8.0
1757	1758	02/18/2020	NaN	UK	2020-02-16T16:23:06	9.0	0.0	8.0
1832	1833	02/19/2020	NaN	UK	2020-02-16T16:23:06	9.0	0.0	8.0
1908	1909	02/20/2020	NaN	UK	2020-02-16T16:23:06	9.0	0.0	8.0
1987	1988	02/21/2020	NaN	UK	2020-02-16T16:23:06	9.0	0.0	8.0
2072	2073	02/22/2020	NaN	UK	2020-02-16T16:23:06	9.0	0.0	8.0
2156	2157	02/23/2020	NaN	UK	2020-02-16T16:23:06	9.0	0.0	8.0
2238	2239	02/24/2020	NaN	UK	2020-02-24T08:33:02	13.0	0.0	8.0
2330	2331	02/25/2020	NaN	UK	2020-02-24T08:33:02	13.0	0.0	8.0
2426	2427	02/26/2020	NaN	UK	2020-02-24T08:33:02	13.0	0.0	8.0
2527	2528	02/27/2020	NaN	UK	2020-02-27T13:23:02	15.0	0.0	8.0
2630	2631	02/28/2020	NaN	UK	2020-02-28T20:13:09	20.0	0.0	8.0
2744	2745	02/29/2020	NaN	UK	2020-02-29T18:03:05	23.0	0.0	8.0
2862	2863	03/01/2020	NaN	UK	2020-03-01T23:23:02	36.0	0.0	8.0
2988	2989	03/02/2020	NaN	UK	2020-03-02T20:33:02	40.0	0.0	8.0

SNo	Date	Province/State Country	Last Update	Confirmed	Deaths	Recovered
3125	3126	03/03/2020 NaN	UK 2020-03-03T15:13:26	51.0	0.0	8.0
3272	3273	03/04/2020 NaN	UK 2020-03-04T19:33:03	85.0	0.0	8.0
3428	3429	03/05/2020 NaN	UK 2020-03-05T18:03:03	115.0	1.0	8.0
3597	3598	03/06/2020 NaN	UK 2020-03-06T15:33:03	163.0	2.0	8.0
3794	3795	03/07/2020 NaN	UK 2020-03-07T17:43:05	206.0	2.0	18.0
4017	4018	03/08/2020 NaN	UK 2020-03-08T22:03:10	273.0	3.0	18.0
4271	4272	03/09/2020 NaN	UK 2020-03-09T18:33:03	321.0	4.0	18.0
4537	4538	03/10/2020 NaN	UK 2020-03-10T23:53:02	382.0	6.0	18.0
4743	4744	03/11/2020 UK	UK 2020-03-11T21:33:03	456.0	8.0	18.0
4896	4897	03/11/2020 Channel Islands	UK 2020-03-11T20:53:02	2.0	0.0	0.0
4923	4924	03/11/2020 Gibraltar	UK 2020-03-11T20:53:02	1.0	0.0	1.0
4960	4961	03/12/2020 UK	UK 2020-03-11T21:33:03	456.0	8.0	18.0
5118	5119	03/12/2020 Channel Islands	UK 2020-03-11T20:53:02	2.0	0.0	0.0
5144	5145	03/12/2020 Gibraltar	UK 2020-03-11T20:53:02	1.0	0.0	1.0
5196	5197	03/13/2020 Channel Islands	UK 2020-03-11T20:00:00	2.0	0.0	0.0
5239	5240	03/13/2020 United Kingdom	UK 2020-03-11T20:00:00	798.0	8.0	18.0
5289	5290	03/13/2020 Gibraltar	UK 2020-03-11T20:00:00	1.0	0.0	1.0
5394	5395	03/14/2020 United Kingdom	UK 2020-03-14T14:53:04	1140.0	21.0	18.0
5593	5594	03/14/2020 Channel Islands	UK 2020-03-11T20:53:02	2.0	0.0	0.0
5628	5629	03/14/2020 Gibraltar	UK 2020-03-14T16:33:03	1.0	0.0	1.0
5644	5645	03/15/2020 United Kingdom	UK 2020-03-14T14:53:04	1140.0	21.0	18.0
5833	5834	03/15/2020 Channel Islands	UK 2020-03-15T18:20:19	3.0	0.0	0.0
5885	5886	03/15/2020 Gibraltar	UK 2020-03-14T16:33:03	1.0	0.0	1.0

In [57]:

```
df.groupby('Date').sum()
```

Out[57]:

	SNo	Province/State	Country	Last
	Date			
01/22/2020	741	Anhui Beijing Chongqing Fujian Gansu Guangdong Guang...	Mainland China Mainland China Mainland China Main...	17:00 17:00
01/23/2020	2829	Anhui Beijing Chongqing Fujian Gansu Guangdong Guang...	Mainland China Mainland China Mainland China Main...	17: 17:
01/24/2020	4305	Hubei Guangdong Zhejiang Beijing Chongqing Hunan Gu...	Mainland China Mainland	17:

SNo	Province/State	Country	Last
Date			
		ChinaMainland	17:
		ChinaMain...	17:00
		Mainland	
01/25/2020	6490 HubeiGuangdongZhejiangChongqingHunanBeijingAnh...	ChinaMainland	17:
		ChinaMain...	17:
		Mainland	
01/26/2020	9071 HubeiGuangdongZhejiangHenanChongqingHunanBeiji...	ChinaMainland	16:
		ChinaMain...	16:
		Mainland	
01/27/2020	12342 HubeiGuangdongZhejiangHenanChongqingHunanBeiji...	ChinaMainland	23:
		ChinaMain...	23:
		Mainland	
01/28/2020	15262 HubeiGuangdongZhejiangHenanHunanChongqingJiang...	ChinaMainland	23:
		ChinaMain...	23:
		Mainland	
01/29/2020	18711 HubeiZhejiangGuangdongHunanHenanAnhuiChongqing...	ChinaMainland	19:
		ChinaMain...	19:
		Mainland	
01/30/2020	23345 HubeiZhejiangGuangdongHenanHunanAnhuiChongqing...	ChinaMainland	16:
		ChinaMain...	16:
		Mainland	
01/31/2020	28675 HubeiZhejiangGuangdongHenanHunanJiangxiAnhuiCh...	ChinaMainland	23:59
		ChinaMain...	23:59
		Mainland	
02/01/2020	35309 HubeiZhejiangGuangdongHenanHunanAnhuiJiangxiCh...	ChinaMainland	11:5
		ChinaMain...	10:5
		Mainland	
02/02/2020	39798 HubeiZhejiangGuangdongHenanHunanAnhuiJiangxiCh...	Mainland	02T23:43
		ChinaMainland	
		ChinaMain...	
		Mainland	
02/03/2020	44982 HubeiGuangdongZhejiangHenanHunanAnhuiJiangxiCh...	ChinaMainland	03T23:23
		ChinaMain...	
		Mainland	
02/04/2020	51135 HubeiZhejiangGuangdongHenanHunanAnhuiJiangxiCh...	ChinaMainland	04T23:43
		ChinaMain...	
		Mainland	

SNo	Province/State	Country	Last
Date			
02/05/2020	56871 Hubei Guangdong Zhejiang Henan Hunan Jiangxi Anhui Ch...	Mainland China Mainland China Mainland China Mainland China Main...	05T23:13 05T13:23
02/06/2020	61912 Hubei Guangdong Zhejiang Henan Hunan Jiangxi Anhui Ch...	Mainland China Mainland China Mainland China Main...	06T23:23 06T12:43
02/07/2020	67932 Hubei Guangdong Zhejiang Henan Hunan Anhui Jiangxi Ch...	Mainland China Mainland China Mainland China Main...	07T23:43 07T10:13
02/08/2020	73116 Hubei Guangdong Zhejiang Henan Hunan Anhui Jiangxi Ji...	Mainland China Mainland China Mainland China Main...	08T23:33 08T11:53
02/09/2020	78300 Hubei Guangdong Zhejiang Henan Hunan Anhui Jiangxi Ch...	Mainland China Mainland China Mainland China Main...	09T23:33 09T15:03
02/10/2020	83484 Hubei Guangdong Zhejiang Henan Hunan Anhui Jiangxi Ji...	Mainland China Mainland China Mainland China Main...	10T23:33 10T13:43
02/11/2020	89936 Hubei Guangdong Zhejiang Henan Hunan Anhui Jiangxi Ji...	Mainland China Mainland China Mainland China Main...	11T23:33 11T14:03
02/12/2020	95265 Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Ji...	Mainland China Mainland China Mainland China Main...	12T14:13 12T12:23
02/13/2020	102009 Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Ji...	Mainland China Mainland China Mainland China Main...	13T14:13 13T13:33
02/14/2020	108975 Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Ji...	Mainland China Mainland China Mainland China Main...	14T23:33 14T12:53

	SNo	Province/State	Country	Last
Date				
02/15/2020	114600	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Ji...	Mainland China Mainland China Mainland China Mainland China Main...	15T23:13 15T14:03
02/16/2020	120225	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Ji...	Mainland China Mainland China Mainland China Main...	16T23:53 16T12:03
02/17/2020	125850	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Ji...	Mainland China Mainland China Mainland China Main...	17T23:13 17T12:43
02/18/2020	131475	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Ji...	Mainland China Mainland China Mainland China Main...	18T23:1 18T13:43
02/19/2020	138966	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Ji...	Mainland China Mainland China Mainland China Main...	19T23:23 19T10:23
02/20/2020	144742	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Di...	Mainland China Mainland China Mainland China Main...	20T23:43 20T11:13
02/21/2020	166698	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China Mainland China Mainland China Main...	21T13:03 21T10:13
02/22/2020	173754	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China Mainland China Mainland China Main...	22T23:33 22T10:03
02/23/2020	183005	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China Mainland China Mainland China Main...	23T11:33 23T08:53
02/24/2020	201645	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China Mainland China Mainland China Main...	24T11:13 24T10:23

Date	SNo	Province/State	Country	Last
02/25/2020	219255	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China Mainland China Mainland China Mainland China Main...	25T15:23 25T08:53
02/26/2020	245430	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China Mainland China Mainland China Sout...	26T14:13 26T10:33
02/27/2020	265965	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China South Korea Mainland China Mainlan...	27T12:13 27T12:03
02/28/2020	301245	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China South Korea Mainland China Mainlan...	28T00:43 28T08:53
02/29/2020	328321	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China South Korea Mainland China Mainlan...	29T12:13 29T18:13
03/01/2020	360125	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China South Korea Italy Mainland China Ma...	01T10:13 01T23:43
03/02/2020	424974	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China South Korea Italy Iran Mainland Chi...	02T15:03 02T20:23
03/03/2020	477160	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China South Korea Italy Iran Mainland Chi...	03T11:43 03T09:43
03/04/2020	530480	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China South Korea Italy Iran Mainland Chi...	04T12:53 04T19:23
03/05/2020	602386	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China South Korea Italy Iran Mainland Chi...	05T14:53 05T09:03
03/06/2020	729932	Hubei Guangdong Henan Zhejiang Hunan Anhui Jiangxi Sh...	Mainland China South Korea Iran Italy Mainland Chi...	06T14:23

SNo	Date	Province/State	Country	Last
				06T13:33
03/07/2020	873000	HubeiGuangdongHenanZhejiangHunanAnhuiJiangxiSh...	Mainland ChinaSouth KoreaItalyIranMainland Chi...	07T11:13 07T10:13
03/08/2020	1050600	HubeiGuangdongHenanZhejiangHunanAnhuiJiangxiSh...	Mainland ChinaItalySouth KoreaIranMainland Chi...	08T14:43 08T18:03
03/09/2020	1165213	HubeiGuangdongHenanZhejiangHunanAnhuiJiangxiSh...	Mainland ChinaItalySouth KoreaIranMainland Chi...	09T14:33 09T18:1
03/10/2020	950999	HubeiGuangdongHenanZhejiangHunanAnhuiJiangxiSh...	Mainland ChinaItalyIranSouth KoreaFranceSpainG...	10T15:13 10T17:53
03/11/2020	1042740	HubeiFranceGuangdongHenanZhejiangHunanAnhuiJia...	Mainland ChinaItalyIranSouth KoreaFranceSpainG...	11T10:53 11T21:33
03/12/2020	1099701	HubeiFranceGuangdongHenanZhejiangHunanAnhuiJia...	Mainland ChinaItalyIranSouth KoreaFranceSpainG...	12T09:53 11T21:33
03/13/2020	1211755	HubeiGuangdongHenanZhejiangHunanAnhuiJiangxiSh...	Mainland ChinaMainland ChinaMainland ChinaMain...	13T11:09 13T11:09
03/14/2020	1371492	HubeiFranceGuangdongHenanZhejiangUnited Kingdo...	Mainland ChinaItalyIranSouth KoreaSpainGermany...	14T10:13 14T20:13
03/15/2020	1486467	HubeiFranceGuangdongHenanZhejiangUnited Kingdo...	Mainland ChinaItalyIranSouth KoreaSpainGermany...	15T18:20 14T20:13

In [58]:

```
confirmed=df.groupby('Date').sum()['Confirmed'].reset_index()
death=df.groupby('Date').sum()['Deaths'].reset_index()
rec=df.groupby('Date').sum()['Recovered'].reset_index()
```

In [66]:

```

import plotly.graph_objects as go

fig = go.Figure()

# Confirmed (Orange)
fig.add_trace(go.Scatter(
    x=confirmed['Date'],
    y=confirmed['Confirmed'],
    mode='lines+markers',
    name='Confirmed',
    line=dict(color='orange', width=3, shape='spline'),
    marker=dict(size=6),
    hovertemplate='Date: %{x}<br>Confirmed: %{y}<extra></extra>'
))

# Deaths (Red)
fig.add_trace(go.Scatter(
    x=death['Date'],
    y=death['Deaths'],
    mode='lines+markers',
    name='Deaths',
    line=dict(color='red', width=3, shape='spline'),
    marker=dict(size=6),
    hovertemplate='Date: %{x}<br>Deaths: %{y}<extra></extra>'
))

# Recovered (Green)
fig.add_trace(go.Scatter(
    x=rec['Date'],
    y=rec['Recovered'],
    mode='lines+markers',
    name='Recovered',
    line=dict(color='green', width=3, shape='spline'),
    marker=dict(size=6),
    hovertemplate='Date: %{x}<br>Recovered: %{y}<extra></extra>'
))

# Layout improve
fig.update_layout(
    title='COVID-19 Cases Over Time',
    xaxis_title='Date',
    yaxis_title='Number of Cases',
    template='plotly_white',          # Clean white background
    hovermode='x unified',
    legend=dict(title='Cases'),
    xaxis=dict(showgrid=True, gridcolor='lightgray'),
    yaxis=dict(showgrid=True, gridcolor='lightgray')
)

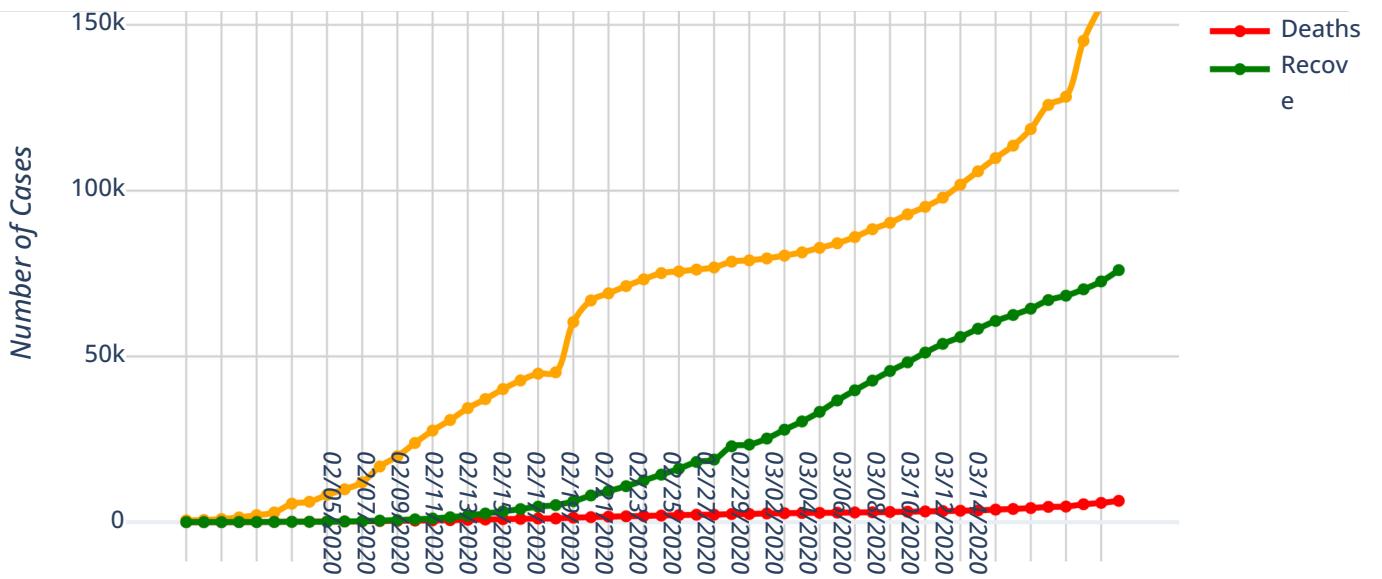
fig.show()

```



COVID-19 Cases Over Time





In [67]:

```
df_confirmed=pd.read_csv(r"C:\Users\DELL LAPTOP\Downloads\time_series_covid_19_confirmed
```

In [68]:

```
df_confirmed.rename(columns={'Country/Region': 'Country'}, inplace=True)
```

In [69]:

```
df_latlong=pd.merge(df, df_confirmed, on=['Country', 'Province/State'])
```

In [71]:

```
df_latlong
```

Out[71]:

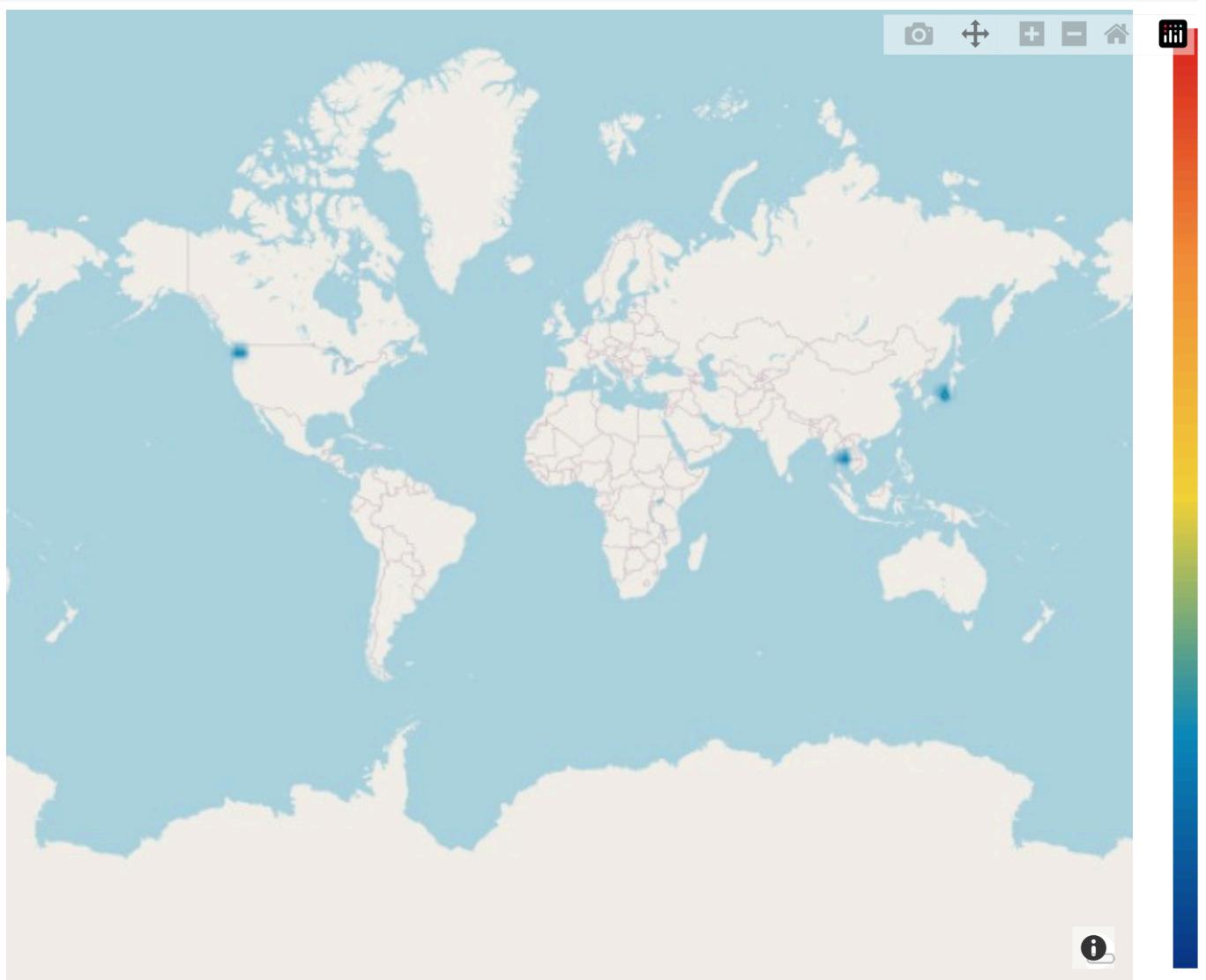
SNo	Date	Province/State	Country	Last Update	Confirmed	Deaths	Recovered	Lat
0	32	01/22/2020	Washington	US	1/22/2020 17:00	1.0	0.0	0.0 47.4009
1	36	01/22/2020	Nan	Japan	1/22/2020 17:00	2.0	0.0	0.0 36.0000
2	37	01/22/2020	Nan	Thailand	1/22/2020 17:00	2.0	0.0	0.0 15.0000
3	70	01/23/2020	Washington	US	1/23/2020 17:00	1.0	0.0	0.0 47.4009
4	74	01/23/2020	Nan	Japan	1/23/2020 17:00	1.0	0.0	0.0 36.0000
...	2020-03-13T14:53:03
3288	5883	03/15/2020	Alaska	US	2020-03-12T09:33:03	1.0	0.0	0.0 61.3707
3289	5884	03/15/2020	North Dakota	US		1.0	0.0	0.0 47.5289

SNo	Date	Province/State	Country	Last Update	Confirmed	Deaths	Recovered	Lat
3290	5885	03/15/2020	Virgin Islands, U.S.	US	2020-03-14T16:15:18	1.0	0.0	0.0 18.3358
3291	5889	03/15/2020	West Virginia	US	2020-03-10T02:33:04	0.0	0.0	0.0 38.4912
3292	5890	03/15/2020	NaN occupied Palestinian territory	NaN	2020-03-11T20:53:02	0.0	0.0	0.0 31.9522

3293 rows × 63 columns

In [72]:

```
fig=px.density_mapbox(df_latlong,lat="Lat",lon="Long",hover_name="Province/State",hover_
fig.update_layout(title='Worldwide Corona Virus Cases')
fig.update_layout(mapbox_style="open-street-map",mapbox_center_lon=0)
fig.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
```



Date=01/22/2020

In []:

