In [1]:

```
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib import style

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

import cufflinks as cf

import plotly.offline as pyo
from plotly.offline import init_notebook_mode,plot,iplot
```

In [2]:

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

In [3]:

```
df = pd.read_csv('Covid-cases-in-India.csv')
df
```

Out[3]:

	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	4	0	0	1
6	7	Karnataka	20	0	3	2
7	8	Kerala	131	7	11	0
8	9	Madhya Pradesh	23	0	0	1
9	10	Maharashtra	144	3	15	4
10	11	Odisha	3	0	0	0
11	12	Puducherry	1	0	0	0
12	13	Punjab	29	0	0	1
13	14	Rajasthan	41	2	3	0
14	15	Tamil Nadu	32	3	1	1
15	16	Telengana	34	11	1	0
16	17	Chandigarh	7	0	0	0
17	18	Jammu and Kashmir	18	0	1	1
18	19	Ladakh	13	0	0	0
19	20	Uttar Pradesh	42	1	11	0
20	21	Uttarakhand	4	0	0	0
21	22	West Bengal	11	0	0	1
22	23	Bihar	7	0	0	1
23	24	Mizoram	1	0	0	0
24	25	Goa	6	0	0	0
25	26	Manipur	1	0	0	0

In [4]:

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

In [5]:

```
df
```

Out[5]:

	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
17	Jammu and Kashmir	18	0	1	1
18	Ladakh	13	0	0	0
19	Uttar Pradesh	42	1	11	0
20	Uttarakhand	4	0	0	0
21	West Bengal	11	0	0	1
22	Bihar	7	0	0	1
23	Mizoram	1	0	0	0
24	Goa	6	0	0	0
25	Manipur	1	0	0	0

In [6]:

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

In [7]:

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

Out[7]:

	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
1	Chhattisgarh	6	0	0	0	6	6
2	Delhi	38	1	6	1	39	32
3	Gujarat	43	0	0	3	43	40
4	Haryana	16	14	11	0	30	19
5	Himachal Pradesh	4	0	0	1	4	3
6	Karnataka	20	0	3	2	20	15
7	Kerala	131	7	11	0	138	127
8	Madhya Pradesh	23	0	0	1	23	22
9	Maharashtra	144	3	15	4	147	128
10	Odisha	3	0	0	0	3	3
11	Puducherry	1	0	0	0	1	1
12	Punjab	29	0	0	1	29	28
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
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	Uttarakhand	4	0	0	0	4	4
21	West Bengal	11	0	0	1	11	10
22	Bihar	7	0	0	1	7	6
23	Mizoram	1	0	0	0	1	1
24	Goa	6	0	0	0	6	6
25	Manipur	1	0	0	0	1	1

In [8]:

```
#which state has highest number of active cases
```

In [9]:

```
Total_Active_Cases = df.groupby('Name of State / UT')['Active Cases'].sum().sort_values(ascending=False).to_frame
()
```

In [10]:

```
Total_Active_Cases
```

Out[10]:

	Active Cases
Name of State / UT	
Maharashtra	128
Kerala	127
Telengana	44
Rajasthan	40
Gujarat	40
Tamil Nadu	33
Uttar Pradesh	32
Delhi	32
Punjab	28
Madhya Pradesh	22
Haryana	19
Jammu and Kashmir	16
Karnataka	15
Ladakh	13
Andhra Pradesh	11
West Bengal	10
Chandigarh	7
Goa	6
Chhattisgarh	6
Bihar	6
Uttarakhand	4
Himachal Pradesh	3
Odisha	3
Manipur	1
Mizoram	1
Puducherry	1

In [11]:

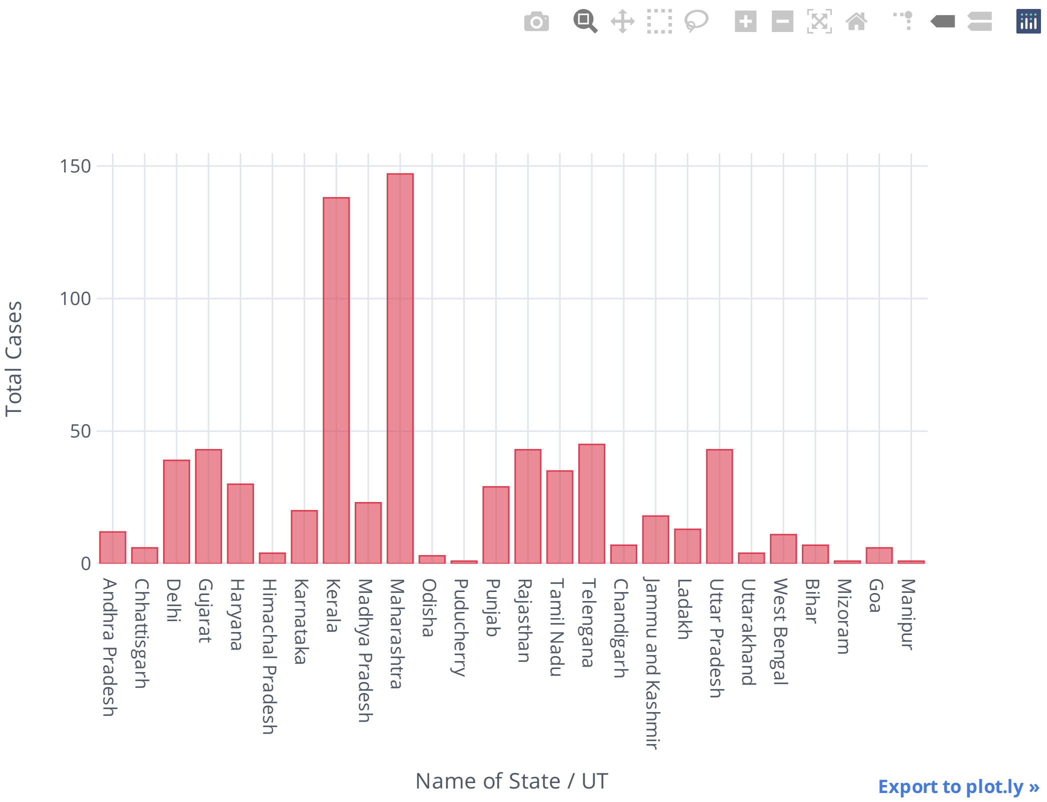
```
Total_Active_Cases.style.background_gradient(cmap='pink_r')
```

Out[11]:

	Active Cases
Name of State / UT	
Maharashtra	128
Kerala	127
Telengana	44
Rajasthan	40
Gujarat	40
Tamil Nadu	33
Uttar Pradesh	32
Delhi	32
Punjab	28
Madhya Pradesh	22
Haryana	19
Jammu and Kashmir	16
Karnataka	15
Ladakh	13
Andhra Pradesh	11
West Bengal	10
Chandigarh	7
Goa	6
Chhattisgarh	6
Bihar	6
Uttarakhand	4
Himachal Pradesh	3
Odisha	3
Manipur	1
Mizoram	1
Puducherry	1

In [12]:

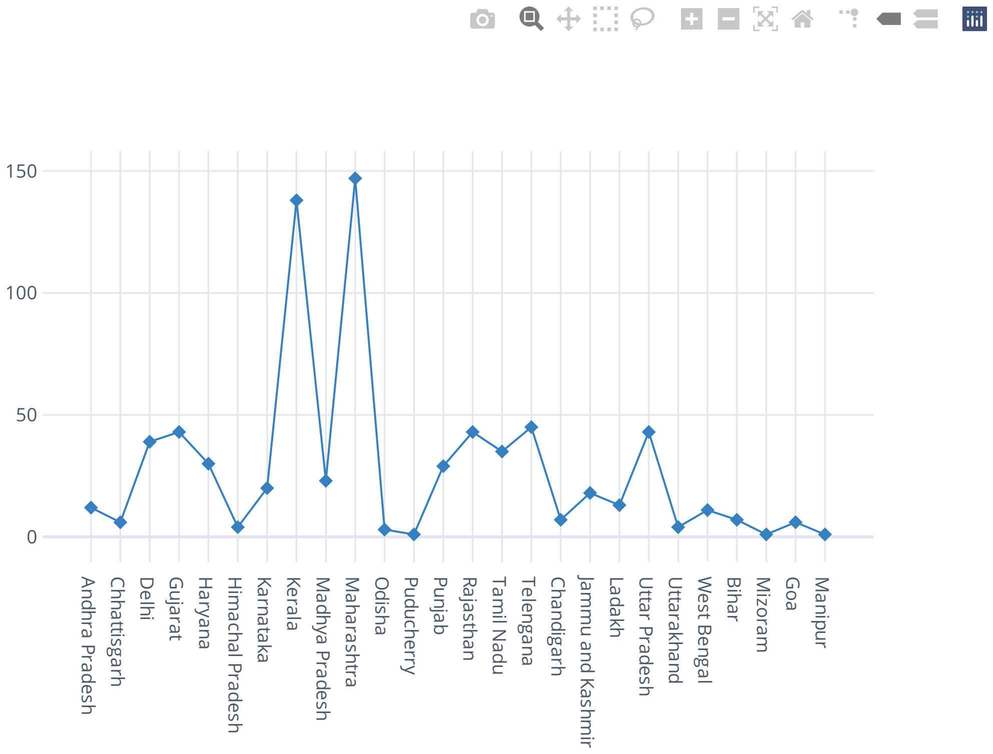
```
df.iplot(kind='bar',x='Name of State / UT',y='Total Cases',xTitle='Name of State / UT',yTitle='Total Cases',color
s='Red')
```



Export to plot.ly »

In [13]:

```
df.iplot(kind='scatter',x='Name of State / UT',y='Total Cases',mode='lines+markers',colors='blue',symbol='diamond
',size=7)
```



Export to plot.ly »

In [14]:

```
pdc_India = pd.read_excel('per_day_cases.xlsx',parse_dates=True,sheet_name='India')
pdc_Italy = pd.read_excel('per_day_cases.xlsx',parse_dates=True,sheet_name='Italy')
pdc_Wuhan = pd.read_excel('per_day_cases.xlsx',parse_dates=True,sheet_name='Wuhan')
pdc_Korea = pd.read_excel('per_day_cases.xlsx',parse_dates=True,sheet_name='Korea')
```

In [15]:

```
pdc_India
```

Out[15]:

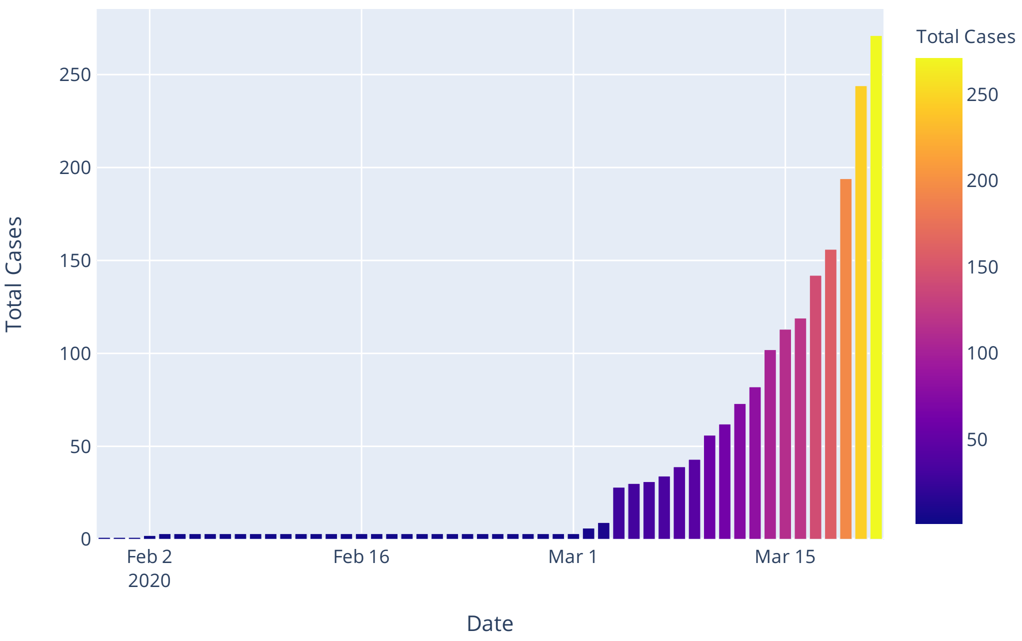
	Date	Total Cases	New Cases	Days after surpassing 100 cases
0	2020-01-30	1	1	NaN
1	2020-01-31	1	0	NaN
2	2020-02-01	1	0	NaN
3	2020-02-02	2	1	NaN
4	2020-02-03	3	1	NaN
5	2020-02-04	3	0	NaN
6	2020-02-05	3	0	NaN
7	2020-02-06	3	0	NaN
8	2020-02-07	3	0	NaN
9	2020-02-08	3	0	NaN
10	2020-02-09	3	0	NaN
11	2020-02-10	3	0	NaN
12	2020-02-11	3	0	NaN
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
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 [16]:

```
fig = px.bar(pdc_India,x='Date',y='Total Cases',title='Confirmed cases in India datewise',color='Total Cases')
fig.show()
```

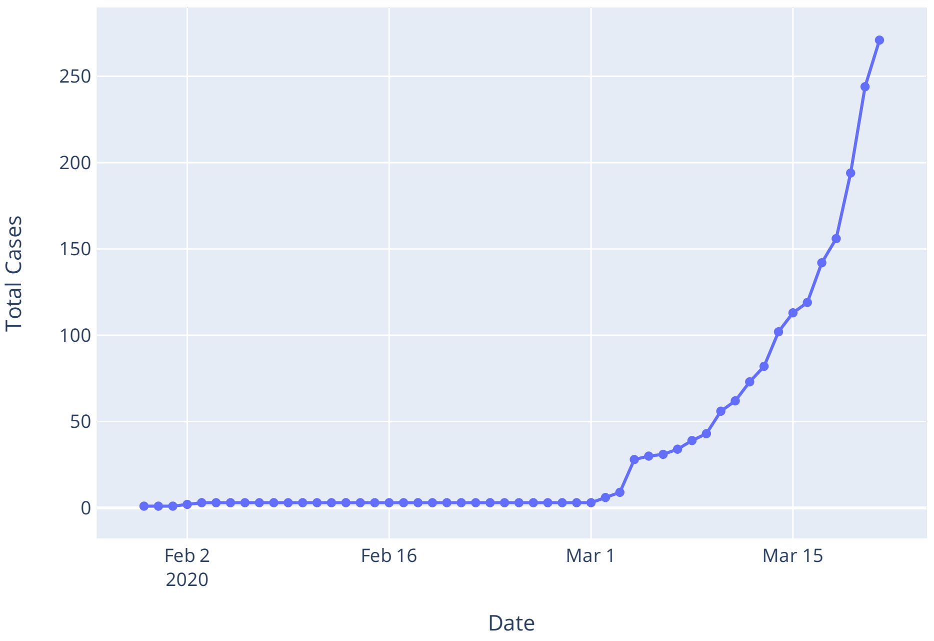
Confirmed cases in India datewise



In [17]:

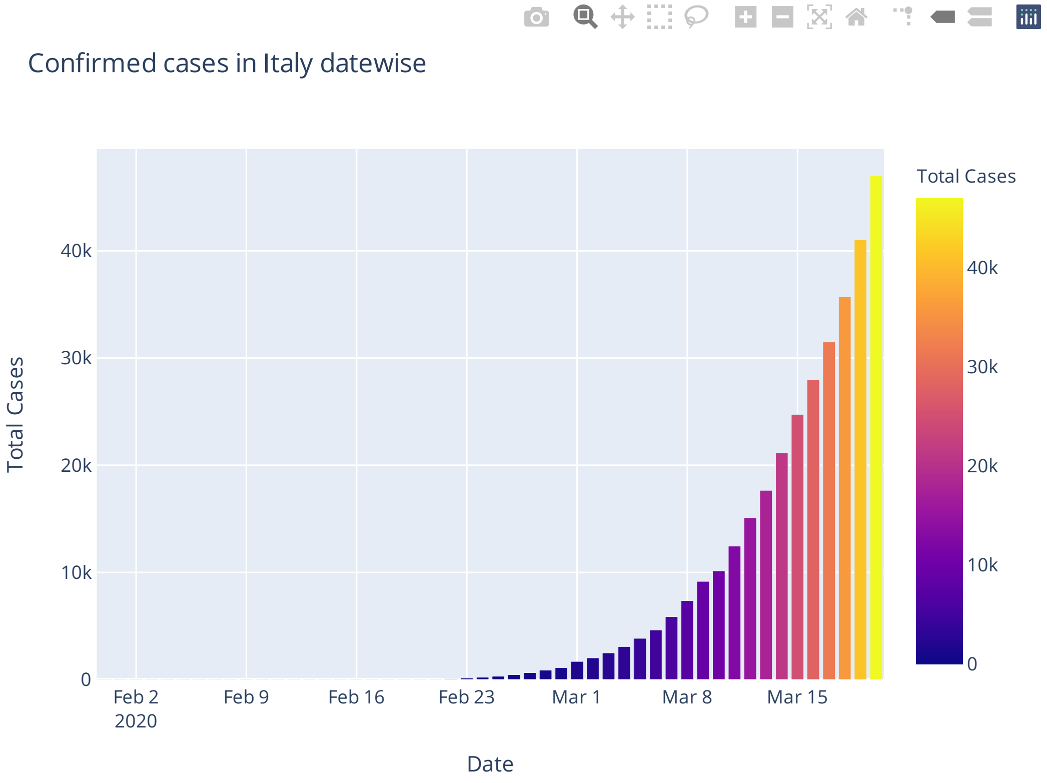
```
fig = go.Figure()
fig.add_trace(go.Scatter(x=pdc_India['Date'],y=pdc_India['Total Cases'],mode='lines+markers'))
fig.update_layout(title='Confirmed cases in India datewise',xaxis=dict(title='Date'),yaxis=dict(title='Total Case
s'))
```

Confirmed cases in India datewise



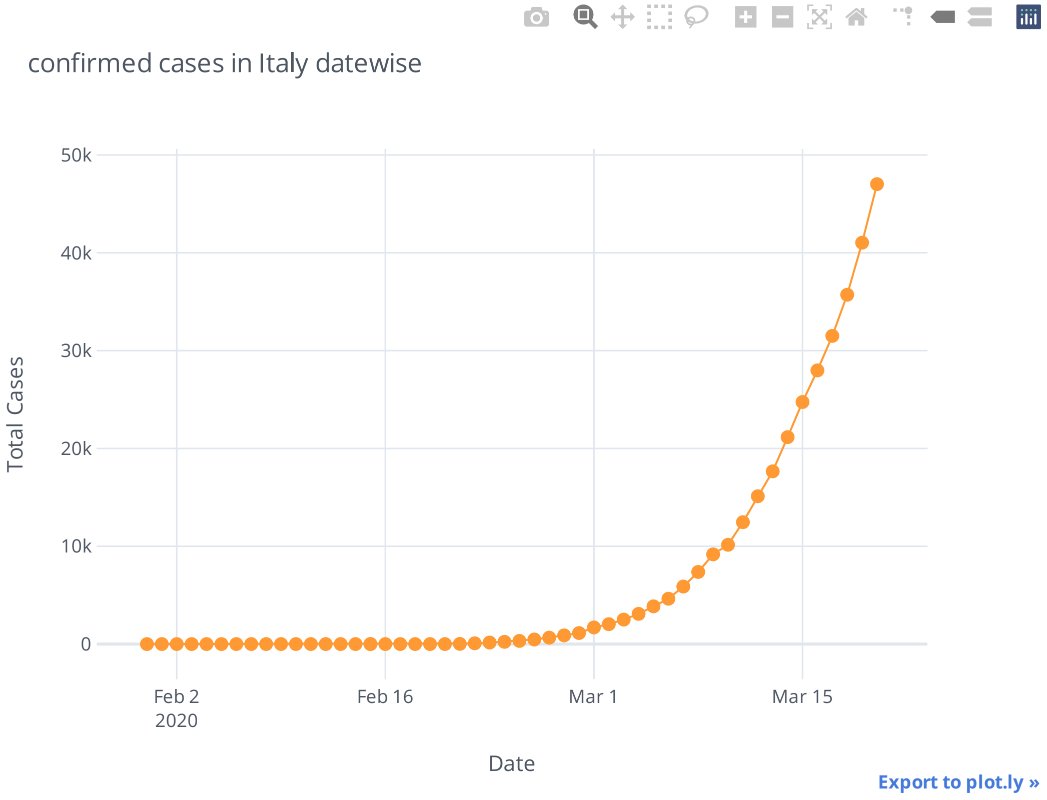
In [18]:

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



In [19]:

```
pdc_Italy.iplot(kind='scatter',x='Date',y='Total Cases',xTitle='Date',yTitle='Total Cases',mode='lines+markers',s
ize=9,title='confirmed cases in Italy datewise')
```



In [20]:

```
from plotly.subplots import make_subplots
style.use('ggplot')
```

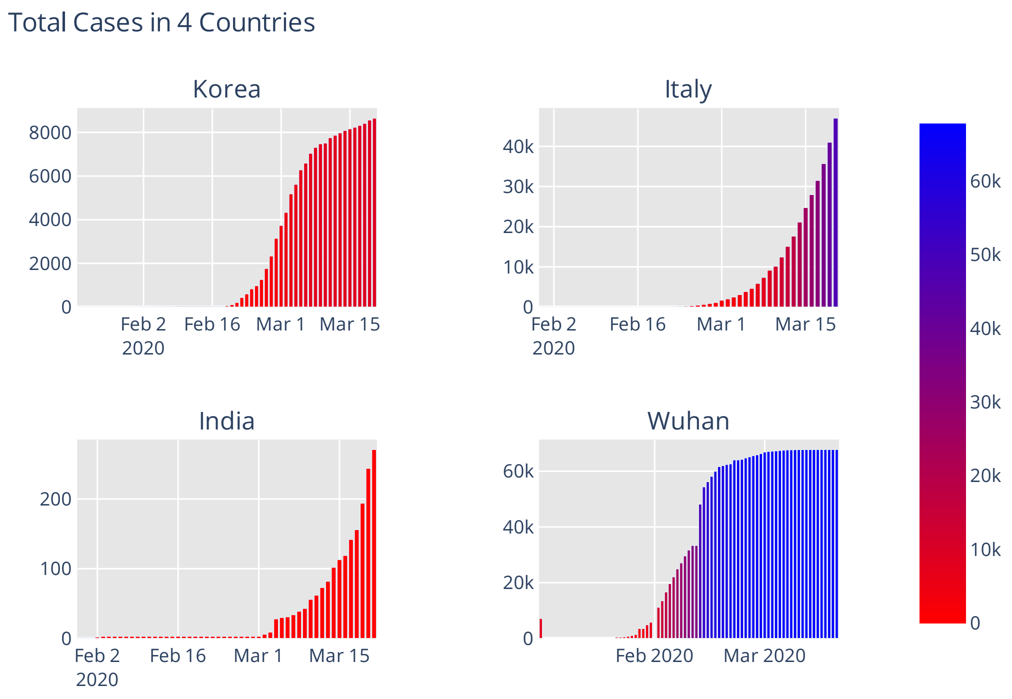
In [21]:

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

In [22]:

```
fig.add_trace(go.Bar(x=pdc_Korea['Date'],y=pdc_Korea['Total Cases'],marker=dict(color=pdc_Korea['Total Cases'],co
loraxis='coloraxis')),1,1)
fig.add_trace(go.Bar(x=pdc_Italy['Date'],y=pdc_Italy['Total Cases'],marker=dict(color=pdc_Italy['Total Cases'],co
loraxis='coloraxis')),1,2)
fig.add_trace(go.Bar(x=pdc_India['Date'],y=pdc_India['Total Cases'],marker=dict(color=pdc_India['Total Cases'],co
loraxis='coloraxis')),2,1)
fig.add_trace(go.Bar(x=pdc_Wuhan['Date'],y=pdc_Wuhan['Total Cases'],marker=dict(color=pdc_Wuhan['Total Cases'],co
loraxis='coloraxis')),2,2)

fig.update_layout(coloraxis=dict(colorscale='Bluered_r'),showlegend=False,title_text='Total Cases in 4 Countries'
)
fig.update_layout(plot_bgcolor='rgb(230,230,230)')
```

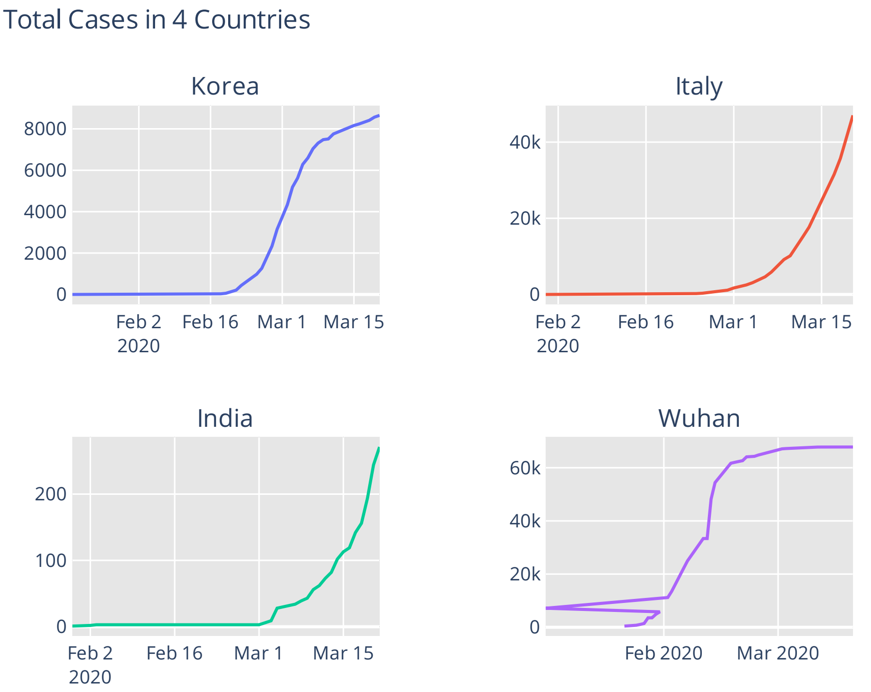


In [23]:

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

fig.add_trace(go.Scatter(x=pdc_Korea['Date'],y=pdc_Korea['Total Cases'],marker=dict(color=pdc_Korea['Total Cases'
],coloraxis='coloraxis')),1,1)
fig.add_trace(go.Scatter(x=pdc_Italy['Date'],y=pdc_Italy['Total Cases'],marker=dict(color=pdc_Italy['Total Cases'
],coloraxis='coloraxis')),1,2)
fig.add_trace(go.Scatter(x=pdc_India['Date'],y=pdc_India['Total Cases'],marker=dict(color=pdc_India['Total Cases'
],coloraxis='coloraxis')),2,1)
fig.add_trace(go.Scatter(x=pdc_Wuhan['Date'],y=pdc_Wuhan['Total Cases'],marker=dict(color=pdc_Wuhan['Total Cases'
],coloraxis='coloraxis')),2,2)

fig.update_layout(coloraxis=dict(colorscale='Bluered_r'),showlegend=False,title_text='Total Cases in 4 Countries'
)
fig.update_layout(plot_bgcolor='rgb(230,230,230)')
```



In [ ]: