In [1]:
```
#Covid19 Visualizations
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
import datetime as dt
import numpy as np
```

In [2]:
```
URL_DATASET = r'C:\Users\deshw\OneDrive\Desktop\Covid Outbreak\covid_19_india.csv'
covid = pd.read_csv(URL_DATASET, parse_dates=['Date'], dayfirst=True)
```

In [3]:
```
covid.head()
```

Out[3]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cur
0	1	2020-01-30	6:00 PM	Kerala	1	0	
1	2	2020-01-31	6:00 PM	Kerala	1	0	
2	3	2020-02-01	6:00 PM	Kerala	2	0	
3	4	2020-02-02	6:00 PM	Kerala	3	0	
4	5	2020-02-03	6:00 PM	Kerala	3	0	

In [4]:
```
covid.tail()
```

Out[4]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNationa
15801	15802	2021-06-08	8:00 AM	Telangana	-	
15802	15803	2021-06-08	8:00 AM	Tripura	-	
15803	15804	2021-06-08	8:00 AM	Uttarakhand	-	
15804	15805	2021-06-08	8:00 AM	Uttar Pradesh	-	
15805	15806	2021-06-08	8:00 AM	West Bengal	-	

In [5]:
```
today = covid[covid.Date=='2021-06-08']
```

In [6]:
```
today.shape
```

Out[6]:
```
(36, 9)
```

In [7]:
```
today.head()
```

Out[7]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNationa
15770	15771	2021-06-08	8:00 AM	Andaman and Nicobar Islands	-	
15771	15772	2021-06-08	8:00 AM	Andhra Pradesh	-	
15772	15773	2021-06-08	8:00 AM	Arunachal Pradesh	-	
15773	15774	2021-06-08	8:00 AM	Assam	-	
15774	15775	2021-06-08	8:00 AM	Bihar	-	

In [8]:
```
#Sorting data based on the confirmed cases
max_confirmed_cases = today.sort_values(by='Confirmed',ascending=False)
```

In [9]:
```
max_confirmed_cases.head(10)
```

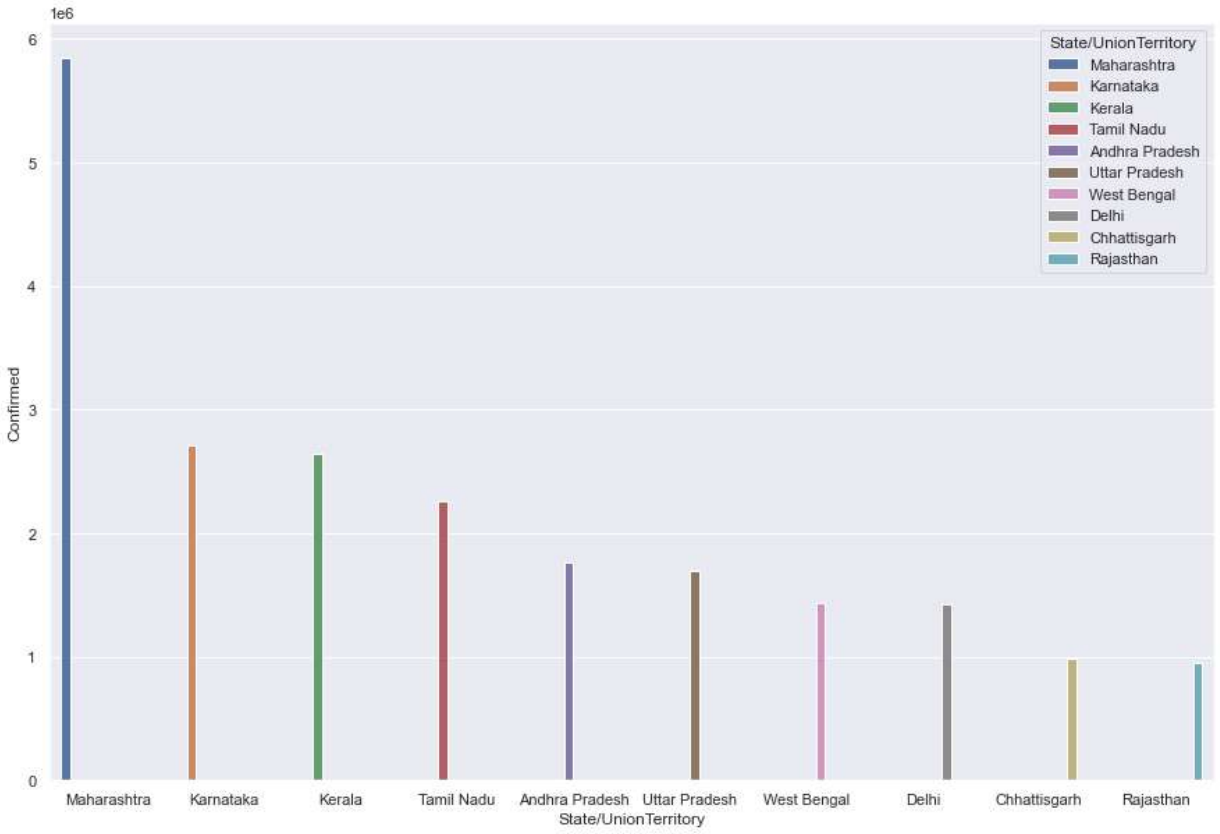
Out[9]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNationa
15790	15791	2021-06-08	8:00 AM	Maharashtra	-	
15785	15786	2021-06-08	8:00 AM	Karnataka	-	
15786	15787	2021-06-08	8:00 AM	Kerala	-	
15800	15801	2021-06-08	8:00 AM	Tamil Nadu	-	
15771	15772	2021-06-08	8:00 AM	Andhra Pradesh	-	
15804	15805	2021-06-08	8:00 AM	Uttar Pradesh	-	
15805	15806	2021-06-08	8:00 AM	West Bengal	-	
15778	15779	2021-06-08	8:00 AM	Delhi	-	
15776	15777	2021-06-08	8:00 AM	Chhattisgarh	-	
15798	15799	2021-06-08	8:00 AM	Rajasthan	-	

In [10]:
```
top_states_confirmed = max_confirmed_cases[0:10]
```

In [11]:
```
sns.set(rc={'figure.figsize':(15,10)})
```

```
sns.barplot(x="State/UnionTerritory", y="Confirmed", data = top_states_confirmed, hu
plt.show()
```



In [12]:

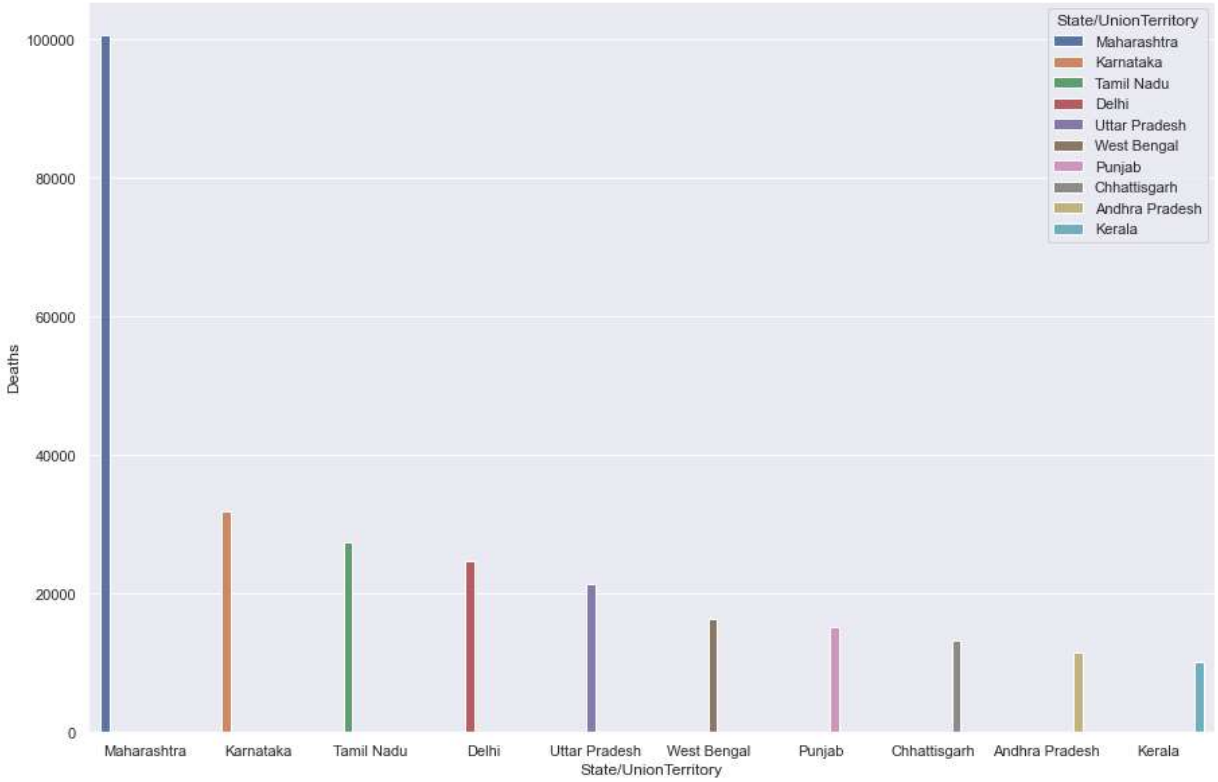
```
#Sorting data based on the deaths
max_death = today.sort_values(by='Deaths',ascending=False)
max_death.head()
```

Out[12]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNationa
15790	15791	2021-06-08	8:00 AM	Maharashtra	-	
15785	15786	2021-06-08	8:00 AM	Karnataka	-	
15800	15801	2021-06-08	8:00 AM	Tamil Nadu	-	
15778	15779	2021-06-08	8:00 AM	Delhi	-	
15804	15805	2021-06-08	8:00 AM	Uttar Pradesh	-	

In [13]:

```
top_states_death = max_death[0:10]
sns.set(rc={'figure.figsize':(15,10)})
sns.barplot(x="State/UnionTerritory", y="Deaths", data = top_states_death, hue="Stat
plt.show()
```

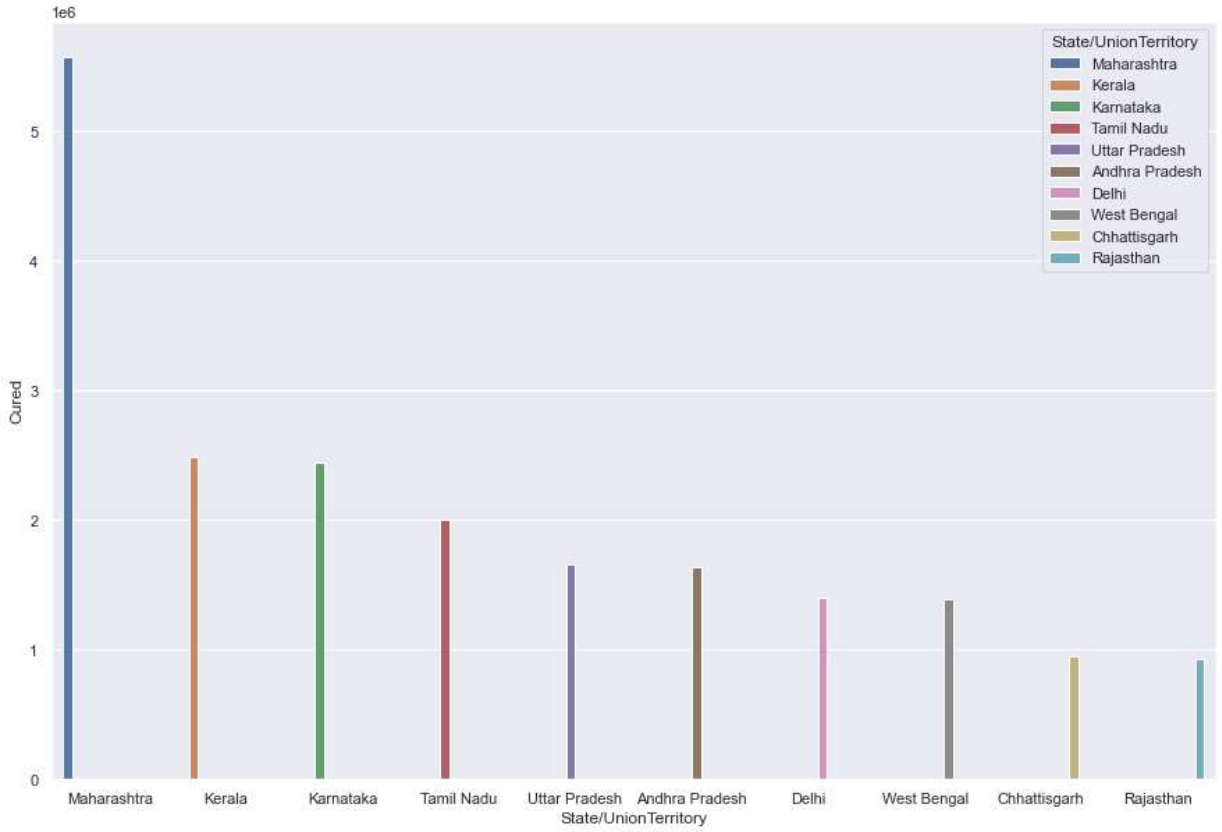


In [14]:
```
#Sorting data based on the cured
max_cured = today.sort_values(by='Cured',ascending=False)
max_cured.head()
```

Out[14]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNationa
15790	15791	2021-06-08	8:00 AM	Maharashtra	-	
15786	15787	2021-06-08	8:00 AM	Kerala	-	
15785	15786	2021-06-08	8:00 AM	Karnataka	-	
15800	15801	2021-06-08	8:00 AM	Tamil Nadu	-	
15804	15805	2021-06-08	8:00 AM	Uttar Pradesh	-	

In [15]:
```
top_states_cured = max_cured[0:10]
sns.set(rc={'figure.figsize':(15,10)})
sns.barplot(x="State/UnionTerritory", y="Cured", data = top_states_cured, hue="State
plt.show()
```



In [16]:
```
updatedCovid = covid.rename(columns = {'State/UnionTerritory': 'State_UnionTerritory
updatedCovid.head()
```

Out[16]:

	Sno	Date	Time	State_UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cur
0	1	2020-01-30	6:00 PM	Kerala	1	0	
1	2	2020-01-31	6:00 PM	Kerala	1	0	
2	3	2020-02-01	6:00 PM	Kerala	2	0	
3	4	2020-02-02	6:00 PM	Kerala	3	0	
4	5	2020-02-03	6:00 PM	Kerala	3	0	

In [17]:
```
#State Wise Analysis
Kerala = updatedCovid[updatedCovid.State_UnionTerritory=='Kerala']
```

In [18]:
```
Kerala.head()
```

Out[18]:

	Sno	Date	Time	State_UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cur
0	1	2020-01-30	6:00 PM	Kerala	1	0	
1	2	2020-01-31	6:00 PM	Kerala	1	0	

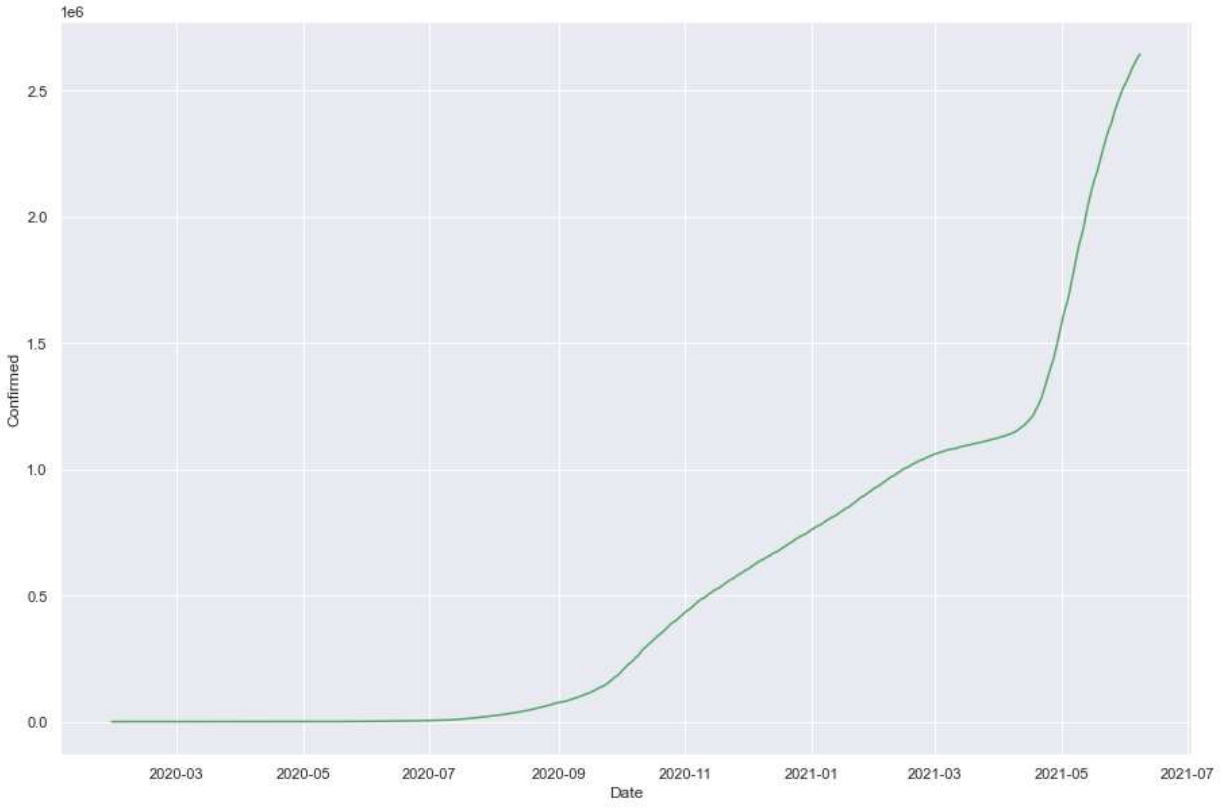
	Sno	Date	Time	State_UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cur
2	3	2020-02-01	6:00 PM	Kerala	2	0	
3	4	2020-02-02	6:00 PM	Kerala	3	0	
4	5	2020-02-03	6:00 PM	Kerala	3	0	

In [19]:
```
Kerala.tail()
```

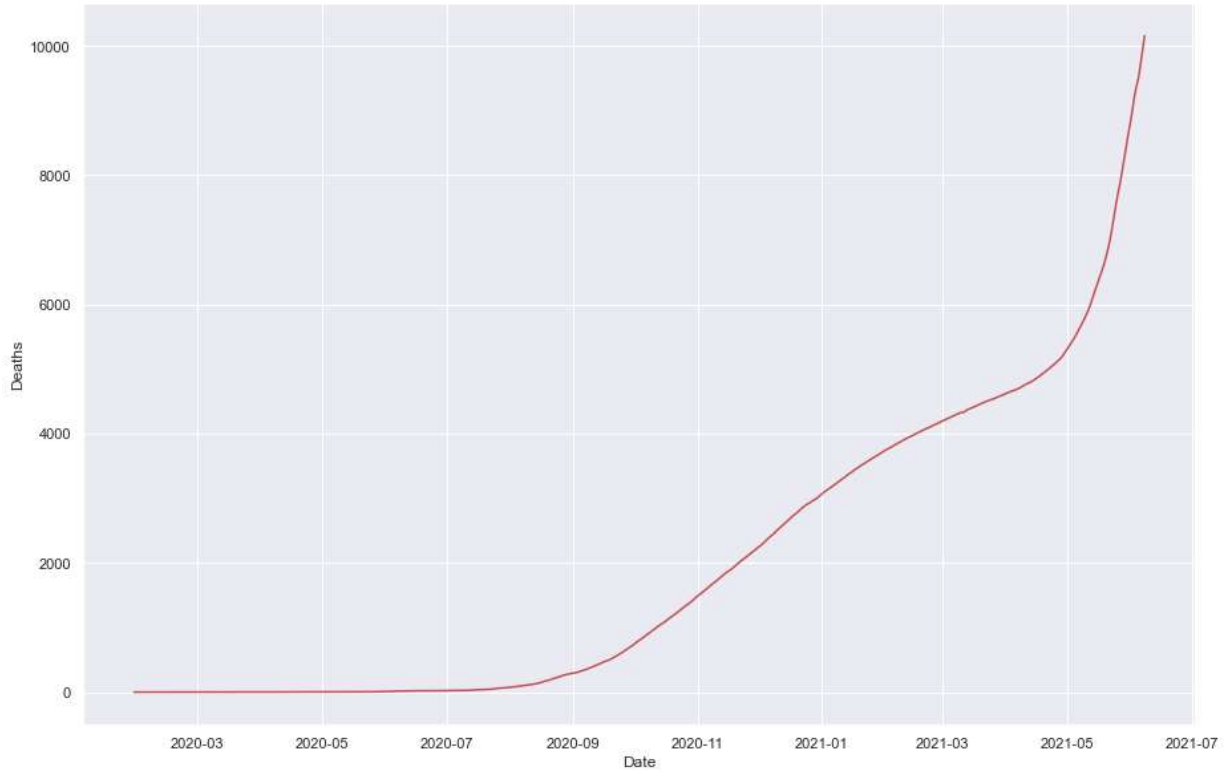
Out[19]:

	Sno	Date	Time	State_UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNationa
15642	15643	2021-06-04	8:00 AM	Kerala	-	
15678	15679	2021-06-05	8:00 AM	Kerala	-	
15714	15715	2021-06-06	8:00 AM	Kerala	-	
15750	15751	2021-06-07	8:00 AM	Kerala	-	
15786	15787	2021-06-08	8:00 AM	Kerala	-	

In [20]:
```
sns.set(rc={'figure.figsize':(15,10)})
sns.lineplot(x="Date", y="Confirmed", data = Kerala, color="g")
plt.show()
```



In [21]:
```
sns.set(rc={'figure.figsize':(15,10)})
sns.lineplot(x="Date", y="Deaths", data = Kerala, color="r")
plt.show()
```



In [22]:
```
#Testing DataSet
TEST_URL_DATASET = r'C:\Users\deshw\OneDrive\Desktop\Covid Outbreak\StatewiseTesting
tests = pd.read_csv(TEST_URL_DATASET)
```

In [23]:
```
tests.head()
```

Out[23]:

	Date	State	TotalSamples	Negative	Positive
0	2020-04-17	Andaman and Nicobar Islands	1403.0	1210	12.0
1	2020-04-24	Andaman and Nicobar Islands	2679.0	NaN	27.0
2	2020-04-27	Andaman and Nicobar Islands	2848.0	NaN	33.0
3	2020-05-01	Andaman and Nicobar Islands	3754.0	NaN	33.0
4	2020-05-16	Andaman and Nicobar Islands	6677.0	NaN	33.0

In [24]:
```
tests.tail()
```

Out[24]:

	Date	State	TotalSamples	Negative	Positive
14093	2021-06-03	West Bengal	12645747.0	NaN	NaN
14094	2021-06-04	West Bengal	12716953.0	NaN	NaN
14095	2021-06-05	West Bengal	12789625.0	NaN	NaN
14096	2021-06-06	West Bengal	12859678.0	NaN	NaN
14097	2021-06-07	West Bengal	12919787.0	NaN	NaN

In [25]:
```
from sklearn.model_selection import train_test_split
```

In [26]:
```
Kerala['Date'] = Kerala['Date'].map(dt.datetime.toordinal)
```

```
<ipython-input-26-875d59e6ee5a>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  Kerala['Date'] = Kerala['Date'].map(dt.datetime.toordinal)
```

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

Out[27]:

	Sno	Date	Time	State_UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cu
0	1	737454	6:00 PM	Kerala	1	0	
1	2	737455	6:00 PM	Kerala	1	0	
2	3	737456	6:00 PM	Kerala	2	0	
3	4	737457	6:00 PM	Kerala	3	0	
4	5	737458	6:00 PM	Kerala	3	0	

In [28]:
```
Kerala.tail()
```

Out[28]:

	Sno	Date	Time	State_UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNatio
15642	15643	737945	8:00 AM	Kerala	-	
15678	15679	737946	8:00 AM	Kerala	-	
15714	15715	737947	8:00 AM	Kerala	-	
15750	15751	737948	8:00 AM	Kerala	-	
15786	15787	737949	8:00 AM	Kerala	-	

In [29]:
```
x=Kerala[['Date']]
y=Kerala[['Confirmed']]
```

In [30]:
```
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.3)
```

In [31]:
```
from sklearn.ensemble import RandomForestRegressor
```

```
rf = RandomForestRegressor()
rf.fit(np.array(x_train).reshape(-1,1),np.array(y_train).reshape(-1,1), sample_weigh
```

```
<ipython-input-31-ebe0f0f54a77>:3: DataConversionWarning: A column-vector y was pass
ed when a 1d array was expected. Please change the shape of y to (n_samples,), for e
xample using ravel().
  rf.fit(np.array(x_train).reshape(-1,1),np.array(y_train).reshape(-1,1), sample_wei
ght=None)
```

Out[31]:
```
RandomForestRegressor()
```

In [32]:
```
rf.predict([[737459]])
```

Out[32]:
```
array([2.99])
```

In [33]:
```
rf.predict([[737950]])
```

Out[33]:
```
array([2608659.92])
```

In [ ]:
```

```