FCI MINI PROJECT REPORT

SUBMITTED BY: MANIK DAYA

UID: 2017140008

BE IT

PROFESSOR: PREETI JANI MA'AM

GITHUB LINK:

PROBLEM STATEMENT: The impact of pandemics is disastrous and can be more disastrous in future. So, to help one understand the situation and prepare for the upcoming challenges we must know what is going to come ahead. So, forecasting the number of coronavirus cases over time along with some insightful analysis of death and recovered cases, the analysis of trends in Covid 19 will eventually help to understand the situation and be prepared for the worst.
So, we propose a time series forecasting model for the novel coronavirus to predict the graph of covid for the future days.

Why I chose this Topic:

There are rumors that the third wave is coming in India in august or September, some says that the third wave is coming in the month of December & some says that the impact of second wave was so bad that the third wave is not going to come in India. So I choose this topic to predict the future cases of covid in India and get a idea of how covid is going to increase in future and how much cases are estimated and be prepare ourselves for the third wave if there is any chances.

Dataset Used:

The dataset I have used is COVID_19_INDIA which consists COVID-19 cases at daily level.

The number of new cases is increasing day by day around the world. This dataset has information from the states and union territories of India at daily level.

State level data comes from Ministry of Health & Family Welfare, The Following data comes from "covid19india" and the Information is highly authentic.

Dataset Link: https://www.kaggle.com/sudalairajkumar/covid19-in-india?select=covid_19_india.csv

Model Used: Prophet model

Prophet is a procedure for forecasting time series data based on an additive **model** where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data.

Importing the Required Libraries

FUTURE CASE PREDICTION OF COVID USING PROPHET MODEL

```python
In [2]: #importing all the necessary libraries
        import pandas as pd
        import plotly.express as ply
        import seaborn as sns
        from statsmodels.tsa.seasonal import seasonal_decompose
        from statsmodels import tsa
        import numpy as np
        import statistics
        import plotly.express as px
        import matplotlib.pyplot as plt
        %matplotlib inline
        import seaborn as sns
        import os
        from fbprophet import Prophet
        import warnings
        warnings.filterwarnings('ignore')
        import plotly.graph_objects as go
        pd.set_option('display.max_rows',20000, 'display.max_columns',100)
        import datetime as datetime
```

Loading the Data

IMPORTING THE DATASET OF INDIA

```python
In [3]: #reading the dataset
        df=pd.read_csv('C:/Users/manik/OneDrive/Desktop/fci/covid_19_india.csv')
```

```python
In [4]: df_corona_in_india = pd.read_csv('C:/Users/manik/OneDrive/Desktop/fci/covid_19_india.csv')
```

EXPLORING DATASET

```python
In [5]: df.head(5)
```

Out[5]:

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

```python
In [6]: df.tail(5)
```

Out[6]:

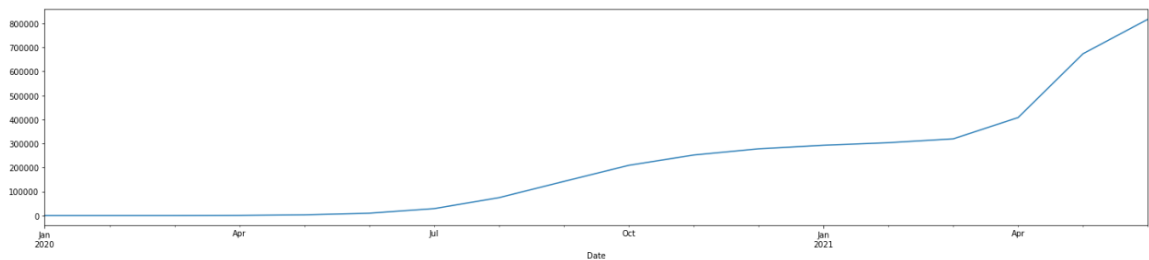|	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cured	Deaths	Confirmed
16521	16522	2021-06-28	8:00 AM	Telangana	-	-	602676	3635	620613
16522	16523	2021-06-28	8:00 AM	Tripura	-	-	60694	672	64863
16523	16524	2021-06-28	8:00 AM	Uttarakhand	-	-	330066	7088	339619
16524	16525	2021-06-28	8:00 AM	Uttar Pradesh	-	-	1679913	22518	1705596
16525	16526	2021-06-28	8:00 AM	West Bengal	-	-	1455453	17612	1494949

Indexing the Data

```
In [7]: import datetime
```

```
In [8]: df['Date']=pd.to_datetime(df['Date'], format='%Y-%m-%d')   #for prohet because prophet model takes datein y-m-d format
```

```
In [9]: df.set_index('Date',inplace=True)   #setting date as unique key
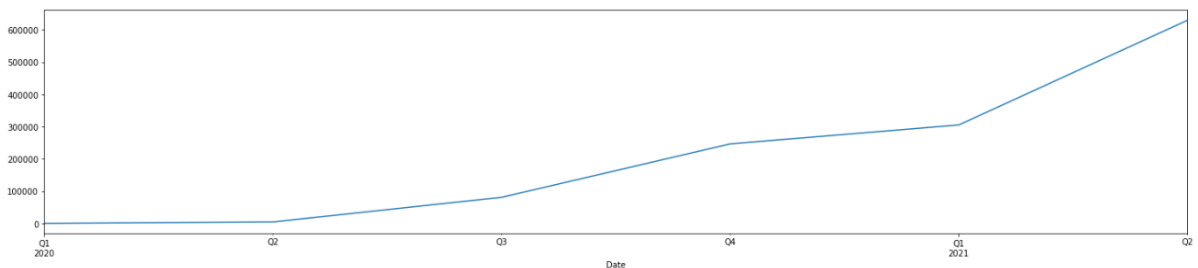```

# Resampling the data

```
In [10]: df['Confirmed'].resample(rule='M').mean().plot(figsize=(25,5))
```

```
Out[10]: <AxesSubplot:xlabel='Date'>
```



```
In [11]: df['Confirmed'].resample(rule='Q').mean().plot(figsize=(25,5))
```

```
Out[11]: <AxesSubplot:xlabel='Date'>
```


Dropping the unnecessary columns

```
In [12]: #Dropping all the unnecessary columns
         df1=pd.DataFrame()
         df1=df.drop(columns=['Sno','Time','ConfirmedIndianNational','State/UnionTerritory','ConfirmedForeignNational','Cured','Deaths'])
```

PLOTTING CONFIRMED CASES

In [13]: df1.plot()

Out[13]: <AxesSubplot:xlabel='Date'>
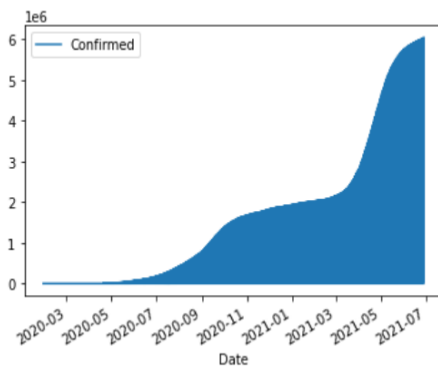
AS WE CAN SEE THAT THE NUMBER OF CONFIRMED CASES KEPT ON INCREASING AND IT IS INCREASING TILL DATE.

DEFINING TOTAL CASES AND ACTIVE CASES:

```
In [16]: Total cases of corona in India
f_corona_in_india['Total Cases'] = df_corona_in_india['Cured'] + df_corona_in_india['Deaths'] + df_corona_in_india['Confirmed']
Active cases of corona in India
f_corona_in_india['Active Cases'] = df_corona_in_india['Total Cases'] - df_corona_in_india['Cured'] - df_corona_in_india['Deaths']
f_corona_in_india.head()
```

Out[16]:

|	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cured	Deaths	Confirmed	Total Cases	Active Cases
0	1	2020-01-30	6:00 PM	Kerala	1	0	0	0	1	1	1
1	2	2020-01-31	6:00 PM	Kerala	1	0	0	0	1	1	1
2	3	2020-02-01	6:00 PM	Kerala	2	0	0	0	2	2	2
3	4	2020-02-02	6:00 PM	Kerala	3	0	0	0	3	3	3
4	5	2020-02-03	6:00 PM	Kerala	3	0	0	0	3	3	3

```
In [17]: df_corona_in_india['Date'] =pd.to_datetime(df_corona_in_india.Date,dayfirst=True)
```

Data Visualization

The following plot gives information about the occurrence total number of cases in India categorized by state wise.
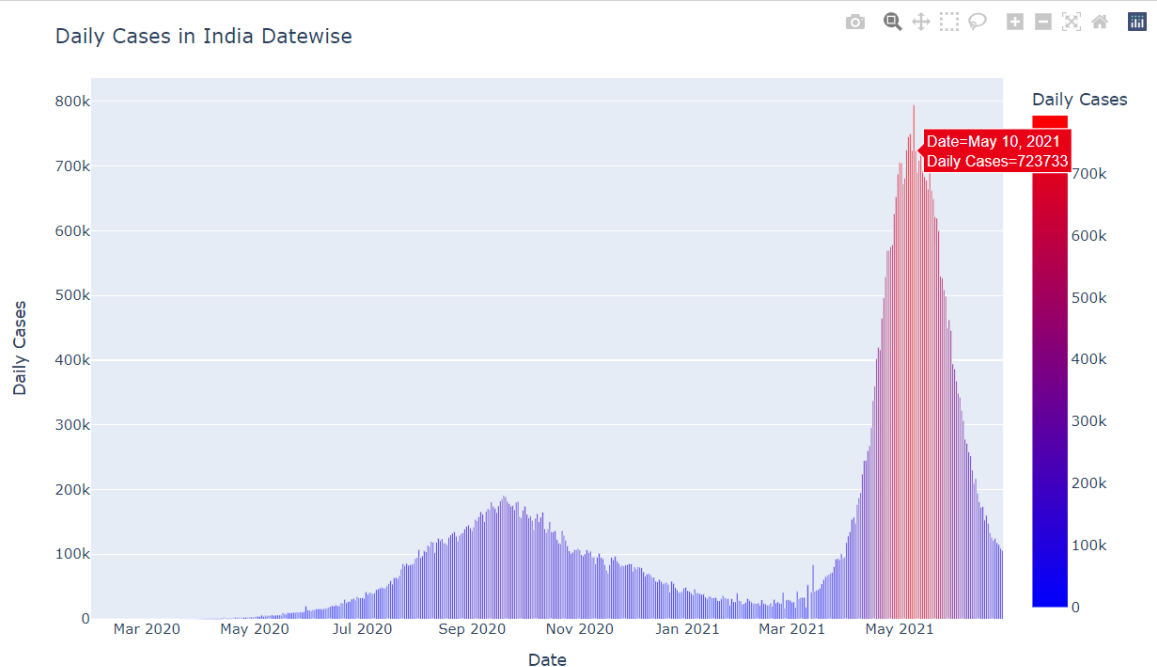
```
In [18]: #Total Cases in India Datewise
corona_data = df_corona_in_india.groupby(['Date'])['Total Cases'].sum().reset_index().sort_values('Total Cases',ascending = True)
fig = px.bar(corona_data, y='Total Cases', x='Date',hover_data =['Total Cases'], color='Total Cases', height=600,color_continuous
fig.update_layout(title='Total Cases in India Datewise')
fig.show()
```

Total Cases in India Datewise

Date=Apr 27, 2021
Total Cases=32390410
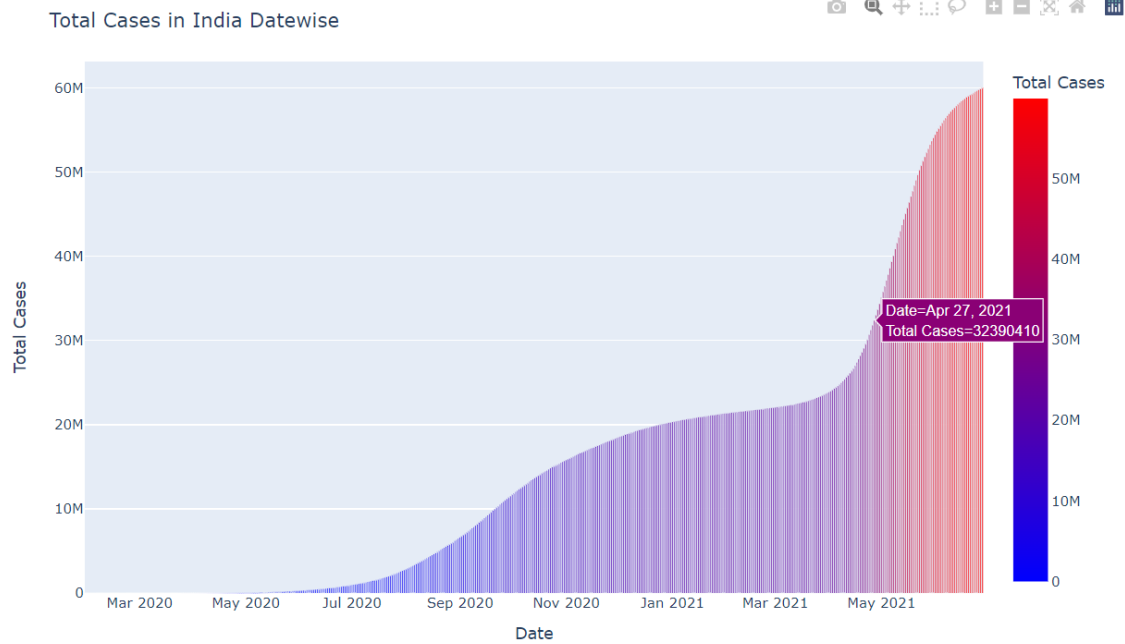
THE ABOVE PLOT CLEARLY SHOWS THAT THE TOTAL CASES IN INDIA TILL JUNE 2021 IS 60 MILLION AND IT IS STILL INCREASING.
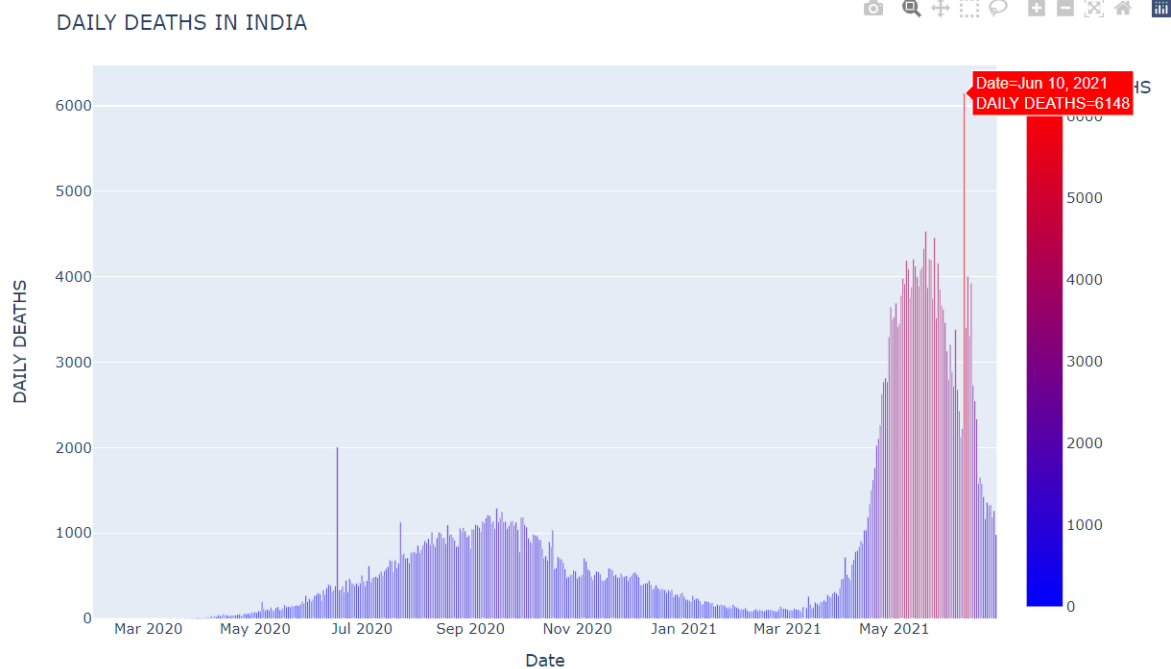
The following plot gives information about the number of cases that were being occurred Daily in India categorized by Date wise.

```python
In [55]:  #Daily Cases in India Datewise
          corona_data = df_corona_in_india.groupby(['Date'])['Total Cases'].sum().reset_index().sort_values('Total Cases',ascending = True)
          corona_data['Daily Cases'] = corona_data['Total Cases'].sub(corona_data['Total Cases'].shift())
          corona_data['Daily Cases'].iloc[0] = corona_data['Total Cases'].iloc[0]
          corona_data['Daily Cases'] = corona_data['Daily Cases'].astype(int)
          fig = px.bar(corona_data, y='Daily Cases', x='Date',hover_data =['Daily Cases'], color='Daily Cases', height=600,color_continuous
          fig.update_layout(
              title='Daily Cases in India Datewise')
          fig.show()
```


Daily Cases in India Datewise

Date=May 10, 2021
Daily Cases=723733

The following plot gives information about the number of deaths that were being occurred Daily in India categorized by Date wise.

```
In [56]: #DAILY DEATHS IN INDIA
corona_data = df_corona_in_india.groupby(['Date'])['Deaths'].sum().reset_index().sort_values('Deaths',ascending = True)
corona_data['DAILY DEATHS'] = corona_data['Deaths'].sub(corona_data['Deaths'].shift())
corona_data['DAILY DEATHS'].iloc[0] = corona_data['Deaths'].iloc[0]
corona_data['DAILY DEATHS'] = corona_data['DAILY DEATHS'].astype(int)
fig = px.bar(corona_data, y='DAILY DEATHS', x='Date',hover_data =['DAILY DEATHS'], color='DAILY DEATHS', height=600,color_continu
fig.update_layout(
    title='DAILY DEATHS IN INDIA')
fig.show()
```


The following plot gives information about the number of recovered cases that were being occurred Daily in India categorized by Date wise.

```
In [57]: #DAILY Recovery IN INDIA
corona_data = df_corona_in_india.groupby(['Date'])['Cured'].sum().reset_index().sort_values('Cured',ascending = True)
corona_data['DAILY RECOVERY'] = corona_data['Cured'].sub(corona_data['Cured'].shift())
corona_data['DAILY RECOVERY'].iloc[0] = corona_data['Cured'].iloc[0]
corona_data['DAILY RECOVERY'] = corona_data['DAILY RECOVERY'].astype(int)
fig = px.bar(corona_data, y='DAILY RECOVERY', x='Date',hover_data =['DAILY RECOVERY'], color='DAILY RECOVERY', height=600,color_c
fig.update_layout(
    title='DAILY RECOVERY IN INDIA')
fig.show()
```

DAILY RECOVERY IN INDIA
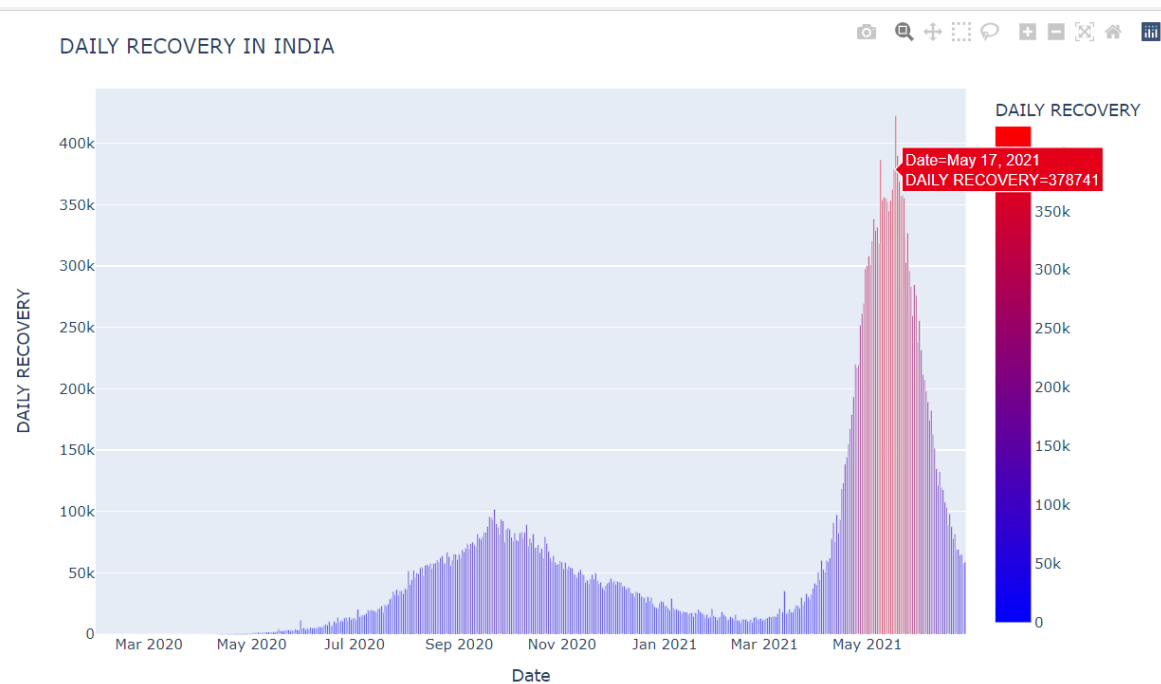
THE ABOVE PLOT SHOWS THAT ON MAY 18 2021 THE NUMBER OF RECOVERED CASES ARE 422436 WHICH IS HIGESHT NUMBER TILL DATE & THE RECOVERY RATE WAS AT PEAK BETWEEN APRIL TO JUNE DURING THE SECOND WAVE.

The following plot gives information about the total cases and active cases and deaths and the number of persons that were cured from coronavirus.

```
In [58]: #Total Cases,Active Cases,Cured,Deaths from Corona Virus in India
corona_data = df_corona_in_india.groupby(['Date'])['Total Cases','Active Cases','Cured','Deaths'].sum().reset_index().sort_values
fig = go.Figure()
fig.add_trace(go.Scatter(x=corona_data['Date'], y=corona_data['Total Cases'],
              mode='lines+markers',name='Total Cases'))
fig.add_trace(go.Scatter(x=corona_data['Date'], y=corona_data['Active Cases'],
              mode='lines+markers',name='Active Cases'))
fig.add_trace(go.Scatter(x=corona_data['Date'], y=corona_data['Cured'],
              mode='lines+markers',name='Cured'))
fig.add_trace(go.Scatter(x=corona_data['Date'], y=corona_data['Deaths'],
              mode='lines+markers',name='Deaths'))
fig.update_layout(title_text='Curve Showing Different Cases from COVID-19 in India',plot_bgcolor='rgb(225,230,255)')
fig.show()
```

Curve Showing Different Cases from COVID-19 in India
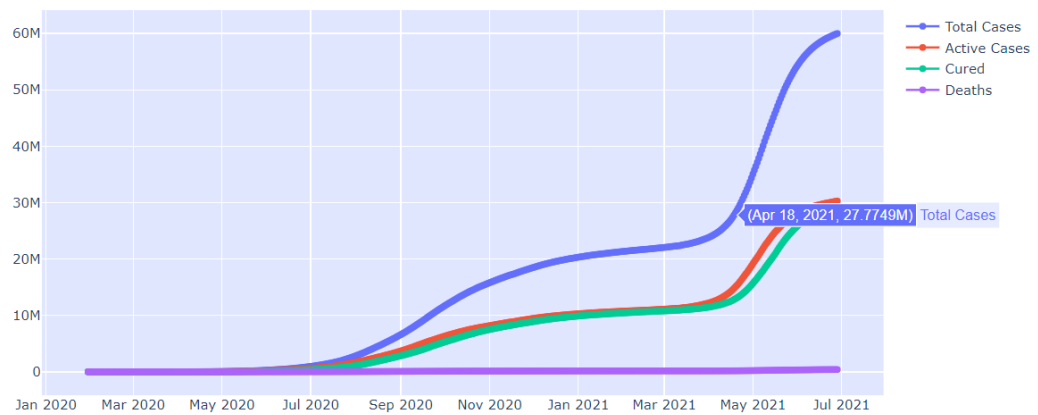
Forecasting the third wave with the help of Facebook prophet model

```python
In [59]: #Forecasting of Total Cases for Next 120 Days
df = df_corona_in_india.groupby('Date')['Total Cases'].sum().reset_index()
# Assigining variables to dates and total cases(Target Class)
df.columns = ['ds','y']
df['ds'] = pd.to_datetime(df['ds'])
# Prophet is a forcasting model made by Facebook
m = Prophet()
# Lets fit the model
m.fit(df)
# Getting the next 30 dates
future = m.make_future_dataframe(periods=120,include_history = False)
forecast = m.predict(future)
#Lets plot on the graph for a easy view and understanding
fig = go.Figure()
# yhat is the predicted value ds is the dates
fig.add_trace(go.Scatter(x=forecast['ds'], y=forecast['yhat'],
                mode='lines+markers',name='Cases',marker_color='Black'))
fig.update_layout(
    title='Forecasting of Total Cases in INDIA for Next 120 Days',xaxis_title="Date",
    yaxis_title="Count")
fig.show()
from fbprophet.diagnostics import cross_validation
# help(cross_validation)
df_cv = cross_validation(m, horizon='30 days', period='15 days', initial='1 days')
print(forecast)
m.plot(forecast)
```

Forecasting of Total Cases in INDIA for Next 120 Days

Out[59]:
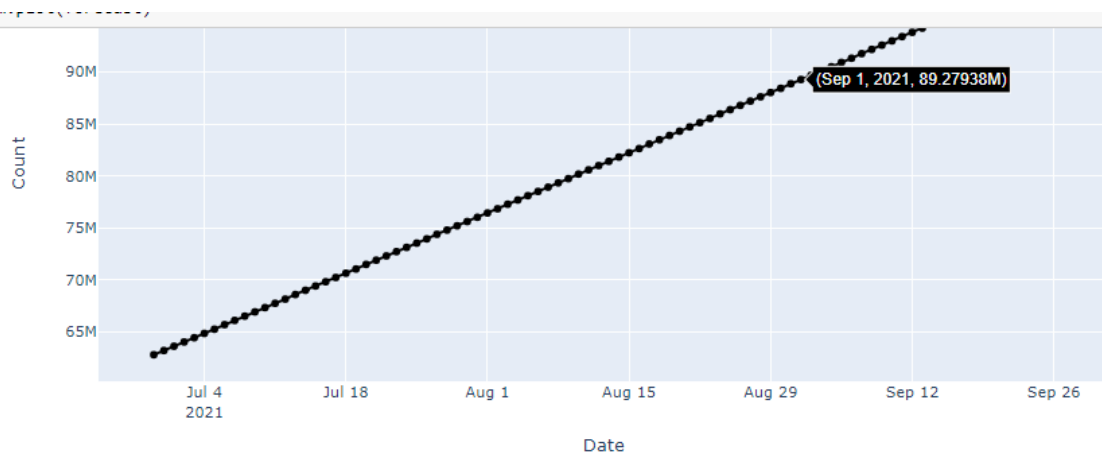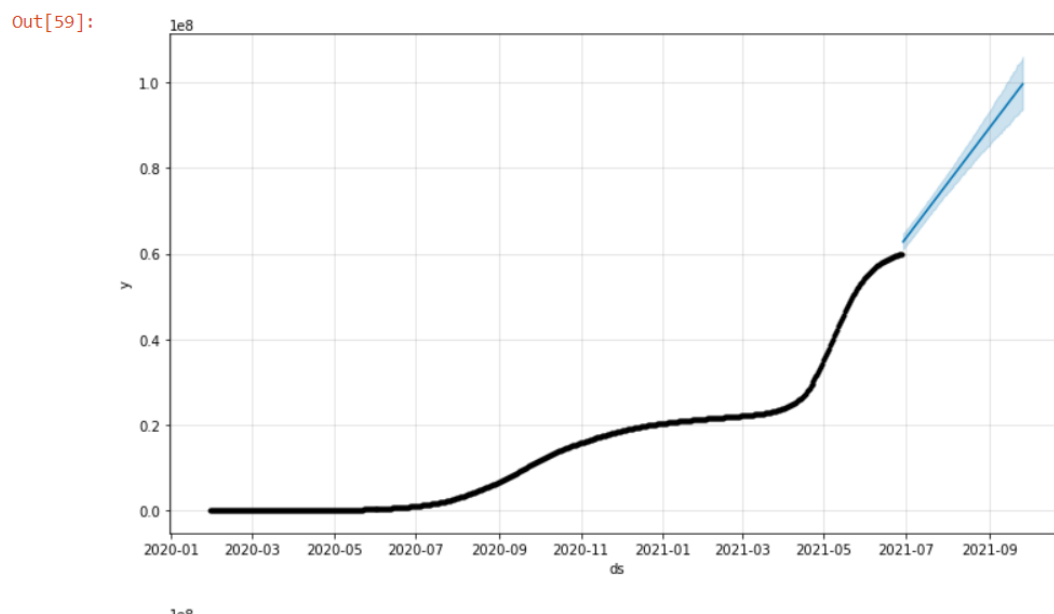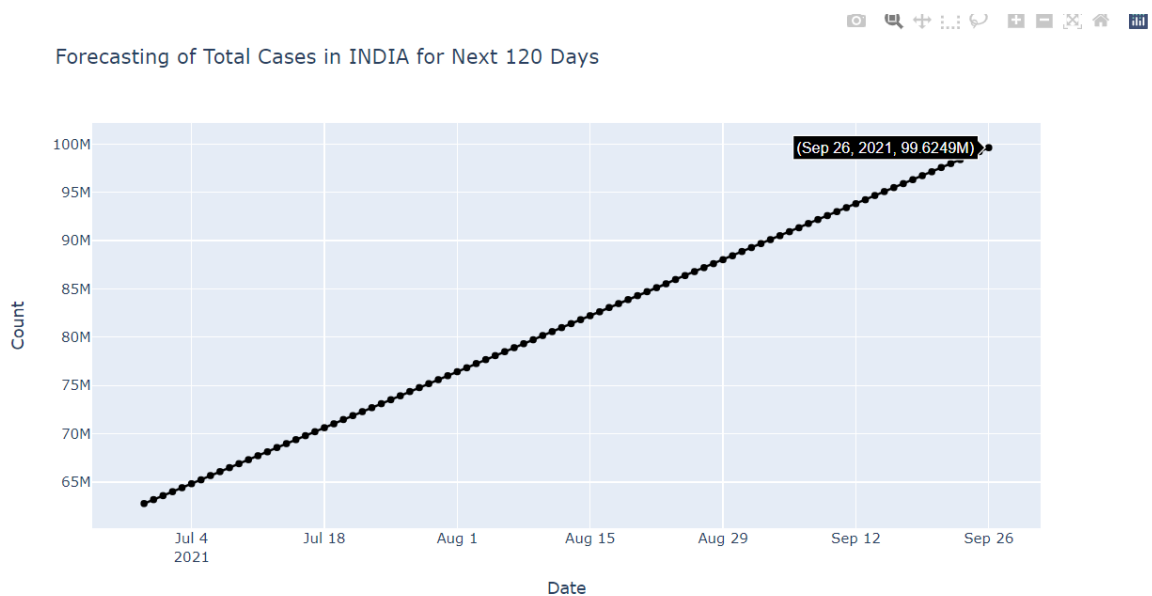
Conclusion

From the above Analysis we were able to state that there will be a huge increase in the number of cases in the coming 4 months by September. The number of people that will be affected in the coming 4 are increasing months so we need to take preventive measures and vaccination in order to prevent the occurrence of third wave in India. The prediction shows that the number of people affected by corona virus can touch the number of 99 million. So looking at the forecasting the chance of third wave in India is high, so we should take preventive measures and start preparing for third wave.