A

PROJECT REPORT

On

COVID -19 DATA ALALYSIS

Submitted in partial fulfilment

of

the requirement for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

by

Archit (17015001006)

Under the Supervision of

Ms. Jyoti

(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INTERNATIONAL INSTITUTE OF TECHNOLOGY & MANAGEMENT
AFFILIATED TO
DEENBANDHU CHHOTU RAM UNIVERSITY OF SCIENCE AND TECHNOLOGYMURTHAL, SONIPAT - 131039, HARYANA, INDIA

TABLE OF CONTENTS

DECLARATION

I hereby declare that this submission is our own work and that to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree of the university or other institute of higher learning, except where due acknowledgement has been made in the text.

Signature:

Archit (17015001006)

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B.Tech (INTERNSHIP) during B. Tech. Fourth Year. Special gratitude to **Ms. JYOTI** Department of Computer Science & Engineering, International Institute of Technology Murthal, for her constant support and guidance throughout the course of our work. Her sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavours have seen light of the day.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the Report.

ABSTRACT

The aim of the project is to provide data analysis of covid-19 (a pandemic started in December 2019). Through plotting of data, various cases have been studied like most affected INDIAN states due to this pandemic. Study of data gathered is combined to show the growth of cases and recovery graph. In this project, the predictions on various cases has been done and finally, the accuracy of the algorithm has been determined. Comparison graphs has also been plotted to analyse how much INDIA is getting affected/recover day by day.

1. Introduction

On 31st December 2019, in the city of Wuhan (CHINA), a cluster of cases of pneumonia of unknown cause was reported to World Health organisation. In January 2020, a previously unknown new virus was identified, subsequently named 2019 novel corona virus. WHO has declared the COVID-19 as a pandemic. A pandemic is defined as disease spread over a wide range of geographical area and that has affected high proportion of the population.

The pandemic has already taken grip over peoples' life. Since the start of the pandemic, some states are facing problem of ever-increasing cases. Through the data analysis of cases one can analyse how states all over INDIA are doing in terms of controlling the pandemic. Analysing data leads to adapt the prevention model of the states that are doing great in terms of lowering the graph. Predictions are made with the dataset available to the individual/country/organisations, thus helping them to decide how far they are able to control the pandemic or up to how much extent they should guide preventive measures.

Through this project, a step towards helping people to understand the spread and predict the cases of Indian states is done.

SOFTWARE SPECIFICATIONS

- **Python Idle3.6.5**

It is a python supervisor wherein we will really execute the calculation to separate informational indexes, perform controls, and anticipate the normal outcomes with exactness.

• **NumPy**

Numpy is basically a module or you can say a library that is available in python for scientific computing now it contains a lot of things it contains a powerful dimensional array object then tools for integrating with C C++ it is also very useful in linear algebra Fourier transform and random number capabilities now let me tell you guys numpy can also be used as an efficient multi-dimensional container for data for generic data now let me tell you what exactly is multidimensional array now over here this picture actually depicts multi-dimensional array so we have various elements that are stored in their respective memory locations so we have one two threes in their own memory locations now why is it two dimensional it is two dimensional because it has rows as well as columns so you can see we have three columns and we have four rows available so that is the reason why it becomes a two dimensional array so if I would have had only one row then I would have said that it is a one dimensional array but since it contains rows as well as columns that is it is represented in a matrix form that is why we call this as a two dimensional array so I hope we are clear with what exactly two dimensional arrays

• **Matplotlib**

Matplotlib is a very useful library for the Python programming language and its statistical discipline extension NumPy. It gives an critique settled API to implanting plots into submissions developing totally beneficial.

It is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002.

One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

• **Pandas**

It is likewise a Python library for information control and examination. Specifically, it offers information structures and activities for controlling numerical table and time arrangement information.

Pandas makes it simple to do many of the time consuming, repetitive tasks associated with working with data, including:

- Data cleansing
- Data fill
- Data normalization
- Merges and joins
- Data visualization
- Statistical analysis
- Data inspection
- Loading and saving data

*PLOTLY

Plotly's Python graphing library makes interactive, publication-quality graphs. Examples of how to make line plots, scatter plots, area charts, bar charts, error bars, box plots, histograms, heatmaps, subplots, multiple-axes, polar charts, and bubble charts.

*TABLEAU

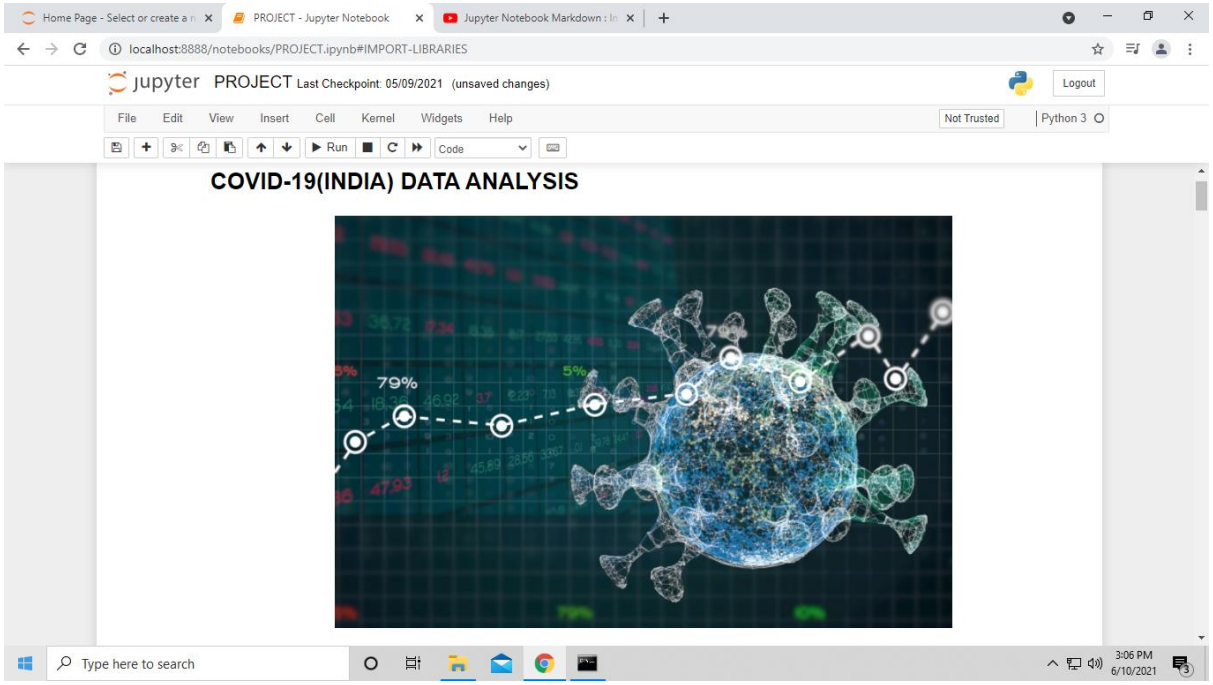
Tableau is a powerful and fastest growing data visualization tool used in the Business Intelligence Industry. It helps in simplifying raw data in a very easily understandable format. Tableau helps create the data that can be understood by professionals at any level in an organization. It also allows non-technical users to create customized dashboards.

Data analysis is very fast with Tableau tool and the visualizations created are in the form of dashboards and worksheets.

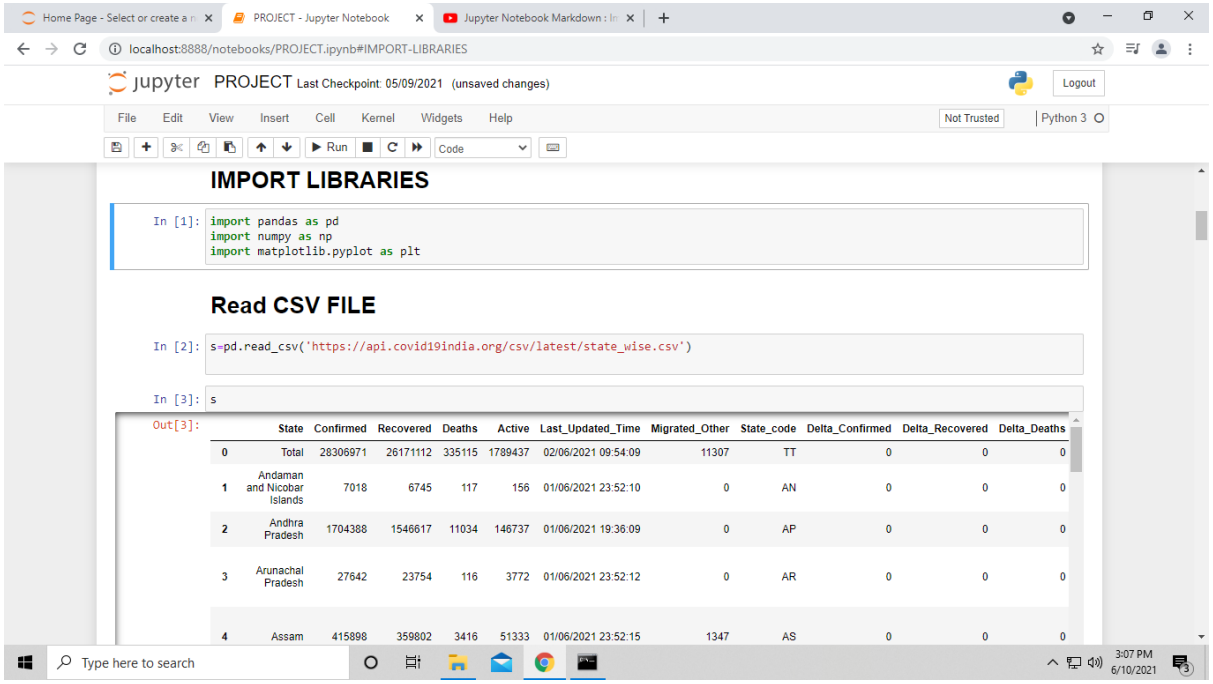
The best features of Tableau software are

- Data Blending
- Real time analysis
- Collaboration of data

PROJECT


Home Page - Select or create a n
PROJECT - Jupyter Notebook
Jupyter Notebook Markdown : In
localhost:8888/notebooks/PROJECT.ipynb#IMPORT-LIBRARIES
jupyter PROJECT Last Checkpoint: 05/09/2021 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
COVID-19(INDIA) DATA ANALYSIS
79%
5%
Type here to search
3:06 PM
6/10/2021

*IMPORTING LIBRARIES AND FETCHING DATASET IN FORM OF CSV FILE THROUGH PANDAS



IMPORT LIBRARIES

```
In [1]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
```

Read CSV FILE

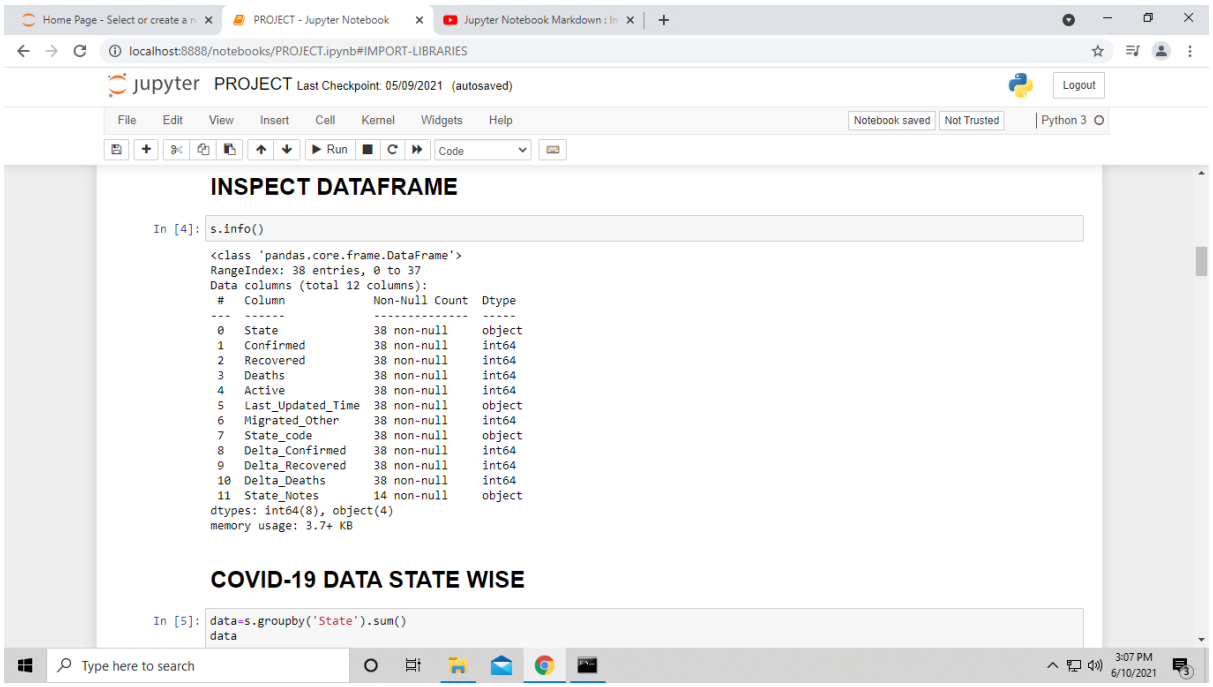
```
In [2]: s=pd.read_csv('https://api.covid19india.org/csv/latest/state_wise.csv')
```

```
In [3]: s
```

Out[3]:

	State	Confirmed	Recovered	Deaths	Active	Last_Updated_Time	Migrated_Other	State_code	Delta_Confirmed	Delta_Recovered	Delta_Deaths
0	Total	28306971	26171112	335115	1789437	02/06/2021 09:54:09	11307	TT	0	0	0
1	Andaman and Nicobar Islands	7018	6745	117	156	01/06/2021 23:52:10	0	AN	0	0	0
2	Andhra Pradesh	1704388	1546617	11034	146737	01/06/2021 19:36:09	0	AP	0	0	0
3	Arunachal Pradesh	27642	23754	116	3772	01/06/2021 23:52:12	0	AR	0	0	0
4	Assam	415898	359802	3416	51333	01/06/2021 23:52:15	1347	AS	0	0	0

*INSPECTING THE DATAFRAME



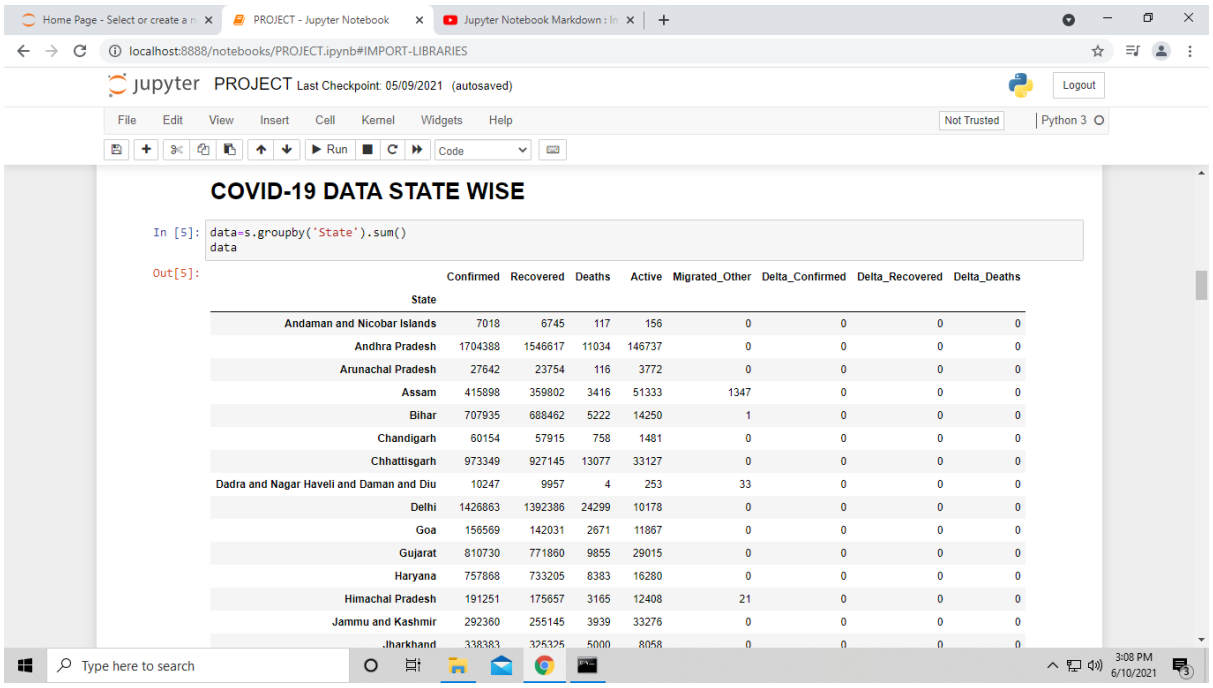
INSPECT DATAFRAME

```
In [4]: s.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 38 entries, 0 to 37
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   State             38 non-null     object
 1   Confirmed         38 non-null     int64
 2   Recovered         38 non-null     int64
 3   Deaths            38 non-null     int64
 4   Active            38 non-null     int64
 5   Last_Updated_Time 38 non-null     object
 6   Migrated_Other    38 non-null     int64
 7   State_code        38 non-null     object
 8   Delta_Confirmed   38 non-null     int64
 9   Delta_Recovered   38 non-null     int64
 10  Delta_Deaths      38 non-null     int64
 11  State_Notes       14 non-null     object
dtypes: int64(8), object(4)
memory usage: 3.7+ KB
```

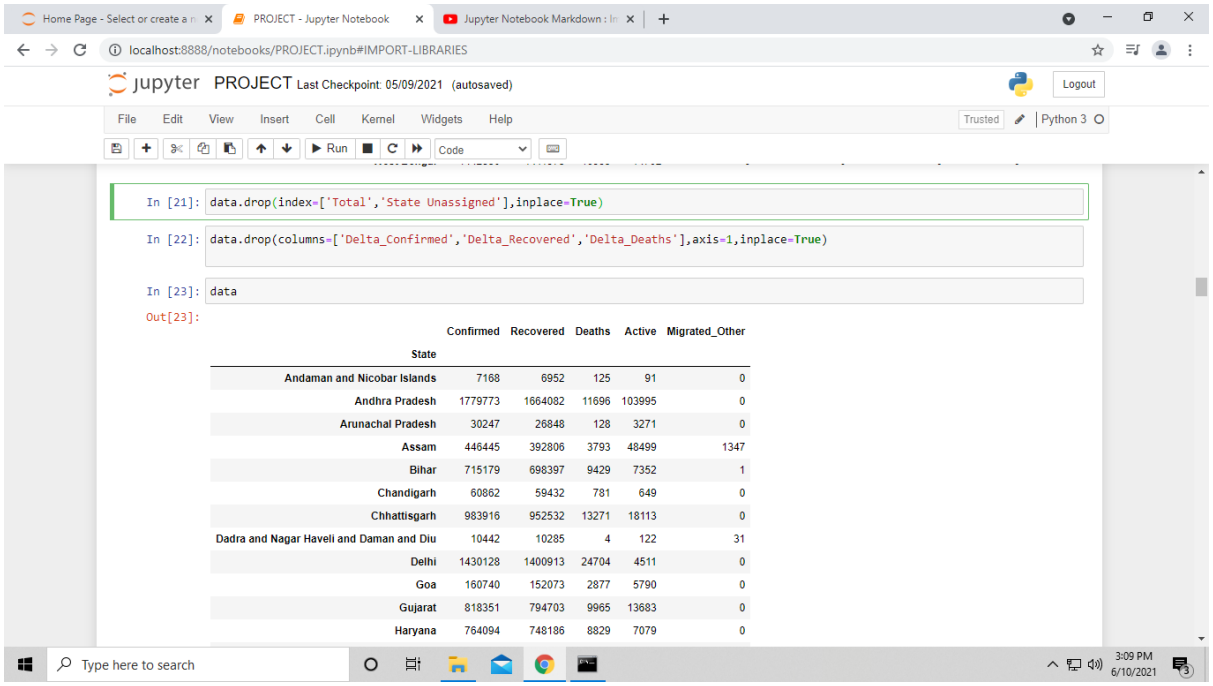
COVID-19 DATA STATE WISE

```
In [5]: data=s.groupby('State').sum()
        data
```



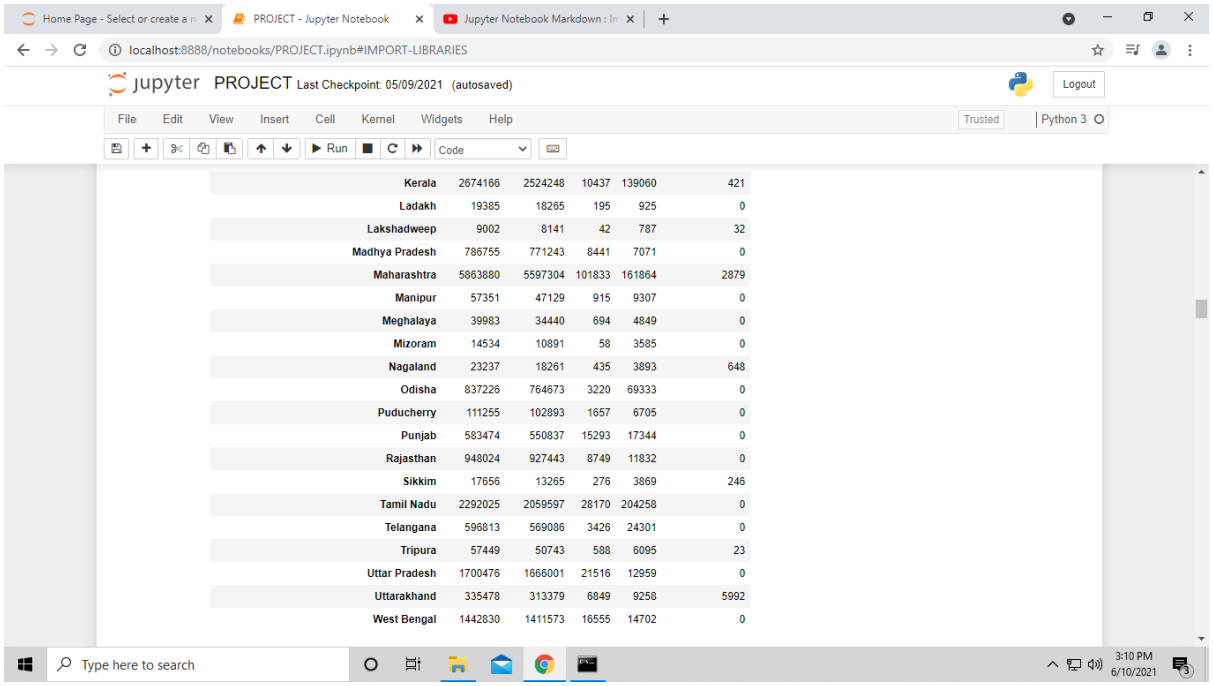
State	Confirmed	Recovered	Deaths	Active	Migrated_Other	Delta_Confirmed	Delta_Recovered	Delta_Deaths
Andaman and Nicobar Islands	7018	6745	117	156	0	0	0	0
Andhra Pradesh	1704388	1546617	11034	146737	0	0	0	0
Arunachal Pradesh	27642	23754	116	3772	0	0	0	0
Assam	415898	359802	3416	51333	1347	0	0	0
Bihar	707935	688462	5222	14250	1	0	0	0
Chandigarh	60154	57915	758	1481	0	0	0	0
Chhattisgarh	973349	927145	13077	33127	0	0	0	0
Dadra and Nagar Haveli and Daman and Diu	10247	9957	4	253	33	0	0	0
Delhi	1426863	1392386	24299	10178	0	0	0	0
Goa	156569	142031	2671	11867	0	0	0	0
Gujarat	810730	771860	9855	29015	0	0	0	0
Haryana	757868	733205	8383	16280	0	0	0	0
Himachal Pradesh	191251	175657	3165	12408	21	0	0	0
Jammu and Kashmir	292360	255145	3939	33276	0	0	0	0
Jharkhand	338383	325325	5000	8058	0	0	0	0

*DATA CLEANING



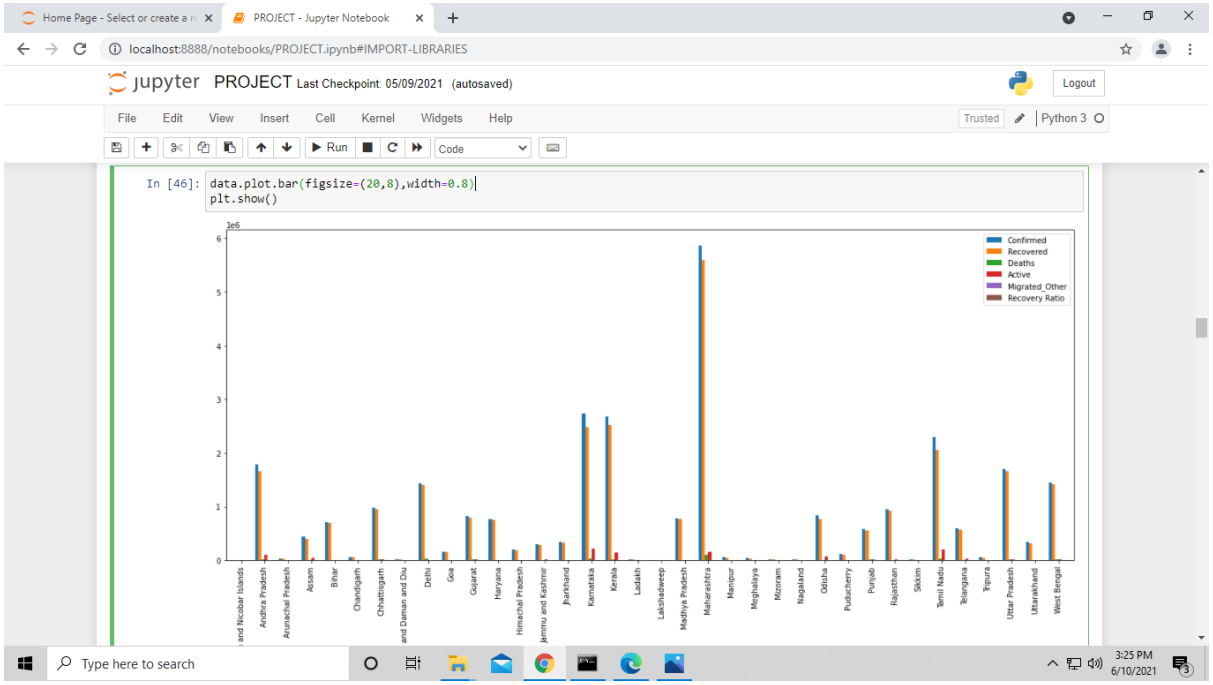
```
In [21]: data.drop(index=['Total','State Unassigned'],inplace=True)

In [22]: data.drop(columns=['Delta_Confirmed','Delta_Recovered','Delta_Deaths'],axis=1,inplace=True)

In [23]: data

Out[23]:
```

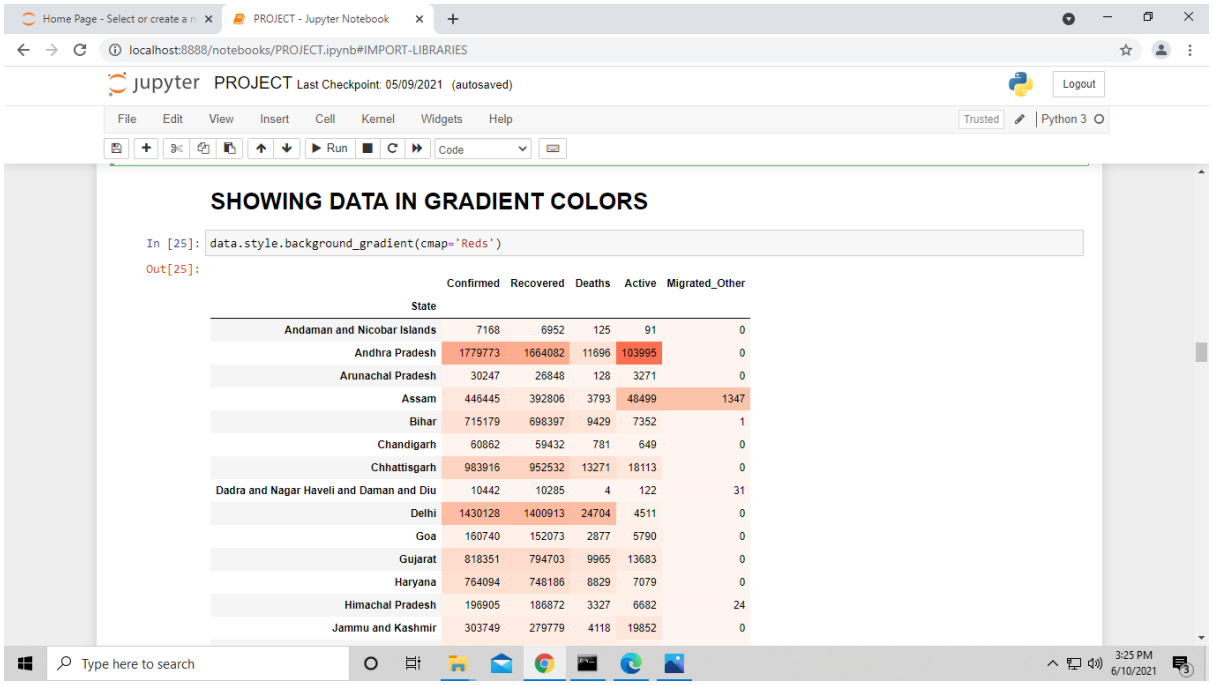
State	Confirmed	Recovered	Deaths	Active	Migrated_Other
Andaman and Nicobar Islands	7168	6952	125	91	0
Andhra Pradesh	1779773	1664082	11696	103995	0
Arunachal Pradesh	30247	26848	128	3271	0
Assam	446445	392806	3793	48499	1347
Bihar	715179	698397	9429	7352	1
Chandigarh	60862	59432	781	649	0
Chhattisgarh	983916	952532	13271	18113	0
Dadra and Nagar Haveli and Daman and Diu	10442	10285	4	122	31
Delhi	1430128	1400913	24704	4511	0
Goa	160740	152073	2877	5790	0
Gujarat	818351	794703	9965	13683	0
Haryana	764094	748186	8829	7079	0



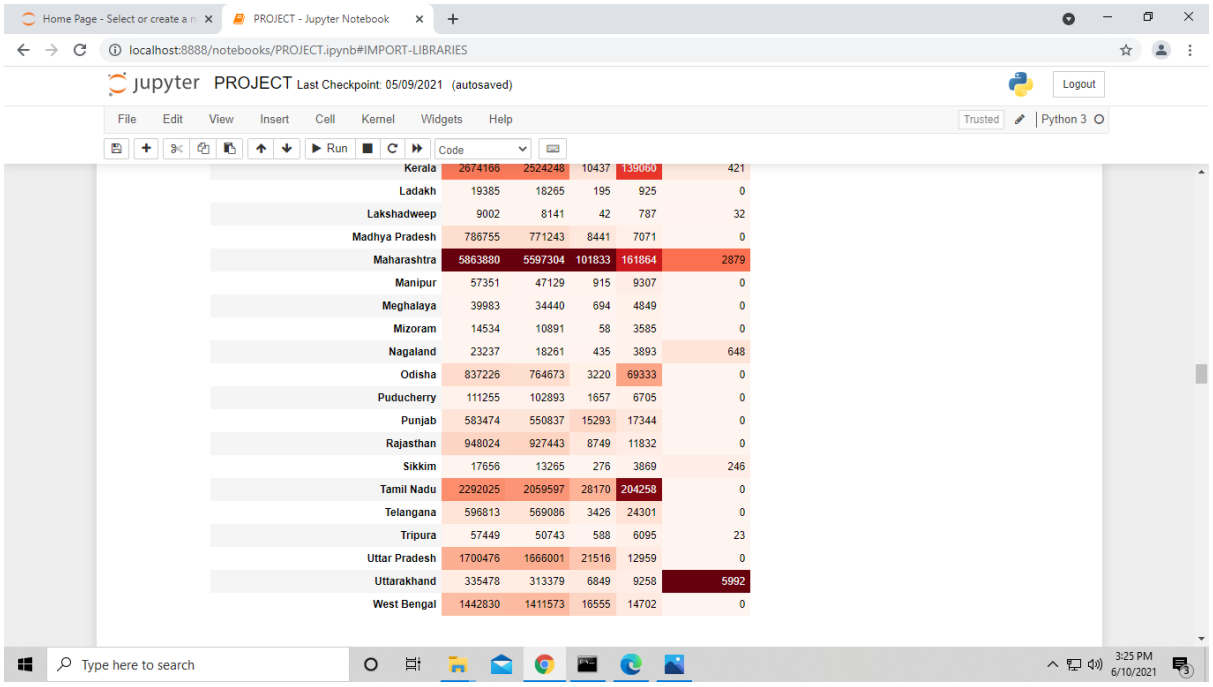
Kerala	2674166	2524248	10437	139060	421
Ladakh	19385	18265	195	925	0
Lakshadweep	9002	8141	42	787	32
Madhya Pradesh	786755	771243	8441	7071	0
Maharashtra	5863880	5597304	101833	161864	2879
Manipur	57351	47129	915	9307	0
Meghalaya	39983	34440	694	4849	0
Mizoram	14534	10891	58	3585	0
Nagaland	23237	18261	435	3893	648
Odisha	837226	764673	3220	69333	0
Puducherry	111255	102893	1657	6705	0
Punjab	583474	550837	15293	17344	0
Rajasthan	948024	927443	8749	11832	0
Sikkim	17656	13265	276	3869	246
Tamil Nadu	2292025	2059597	28170	204258	0
Telangana	596813	569086	3426	24301	0
Tripura	57449	50743	588	6095	23
Uttar Pradesh	1700476	1666001	21516	12959	0
Uttarakhand	335478	313379	6849	9258	5992
West Bengal	1442830	1411573	16555	14702	0

*SHOWING DATA IN FORM OF BAR CHARTS



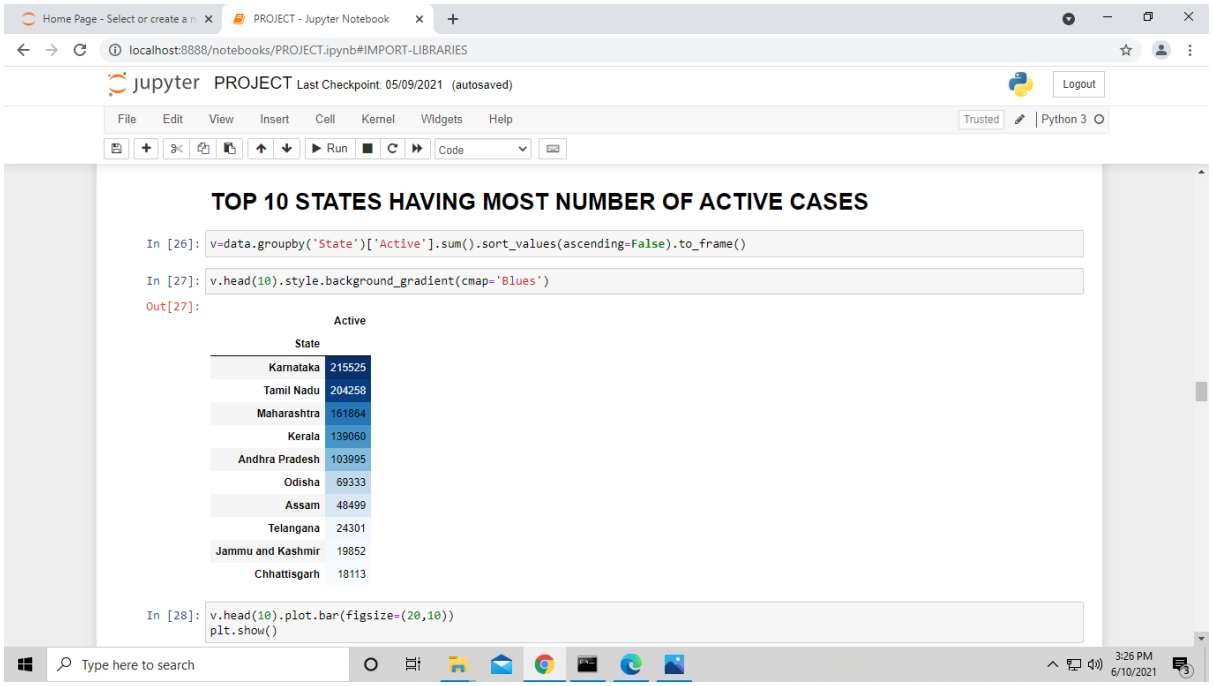


State	Confirmed	Recovered	Deaths	Active	Migrated_Other
Andaman and Nicobar Islands	7168	6952	125	91	0
Andhra Pradesh	1779773	1664082	11696	103995	0
Arunachal Pradesh	30247	26848	128	3271	0
Assam	446445	392806	3793	48499	1347
Bihar	715179	698397	9429	7352	1
Chandigarh	60862	59432	781	649	0
Chhattisgarh	983916	952532	13271	18113	0
Dadra and Nagar Haveli and Daman and Diu	10442	10285	4	122	31
Delhi	1430128	1400913	24704	4511	0
Goa	160740	152073	2877	5790	0
Gujarat	818351	794703	9965	13683	0
Haryana	764094	748186	8829	7079	0
Himachal Pradesh	196905	186872	3327	6682	24
Jammu and Kashmir	303749	279779	4118	19852	0

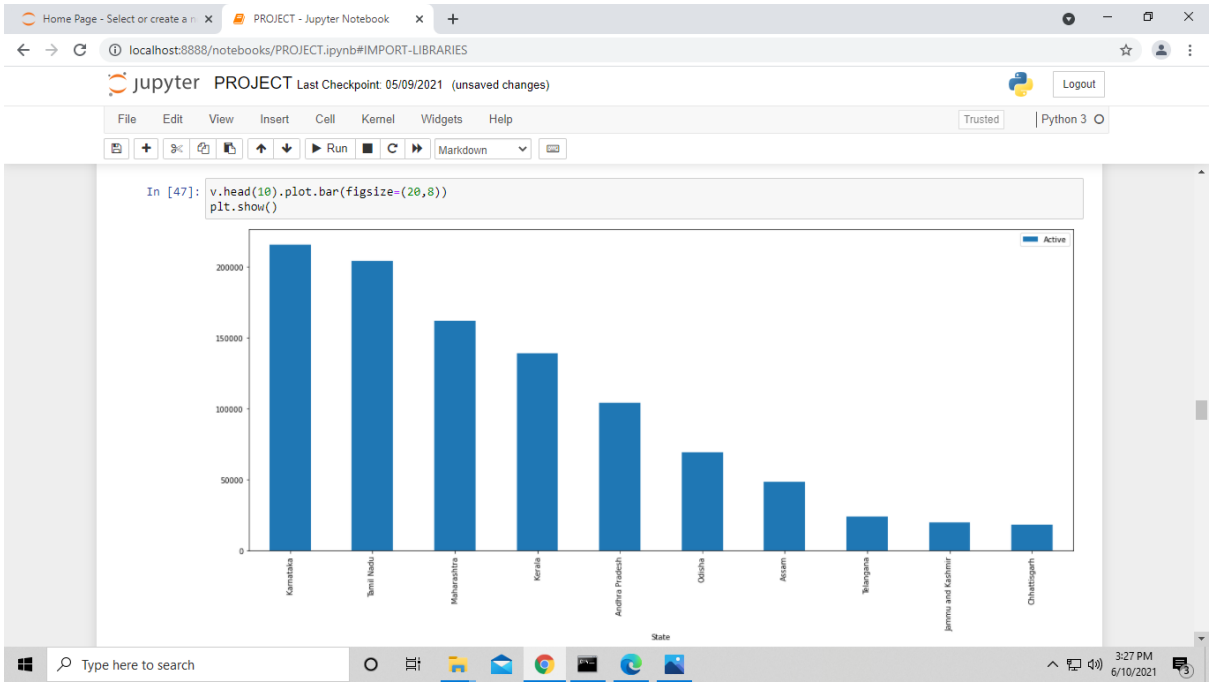


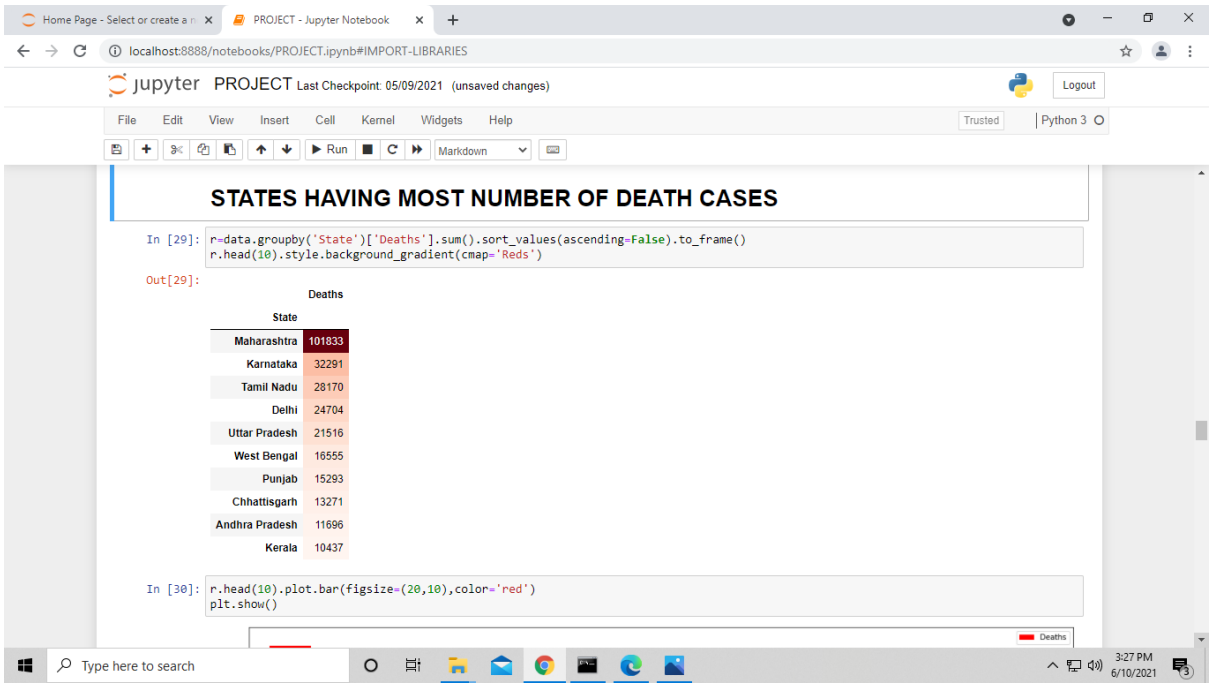
State	Confirmed	Recovered	Deaths	Active	Migrated_Other
Kerala	2674166	2524248	10437	139060	421
Ladakh	19385	18265	195	925	0
Lakshadweep	9002	8141	42	787	32
Madhya Pradesh	786755	771243	8441	7071	0
Maharashtra	5863880	5597304	101833	161864	2879
Manipur	57351	47129	915	9307	0
Meghalaya	39983	34440	694	4849	0
Mizoram	14534	10891	58	3585	0
Nagaland	23237	18261	435	3893	648
Odisha	837226	764673	3220	69333	0
Puducherry	111255	102893	1657	6705	0
Punjab	583474	550837	15293	17344	0
Rajasthan	948024	927443	8749	11832	0
Sikkim	17656	13265	276	3869	246
Tamil Nadu	2292025	2059597	28170	204258	0
Telangana	596813	569086	3426	24301	0
Tripura	57449	50743	588	6095	23
Uttar Pradesh	1700476	1666001	21516	12959	0
Uttarakhand	335478	313379	6849	9258	5992
West Bengal	1442830	1411573	16555	14702	0

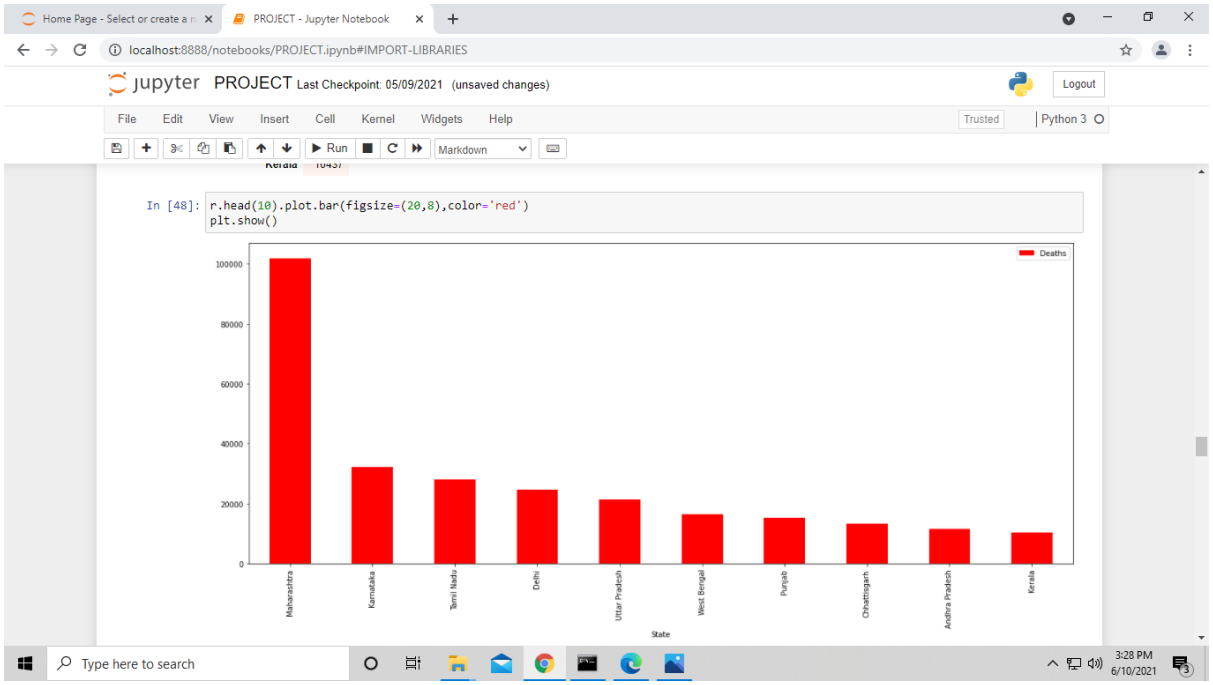
*LISTING TOP 10 STATES HAVING MOST NUMBER OF ACTIVE CASES

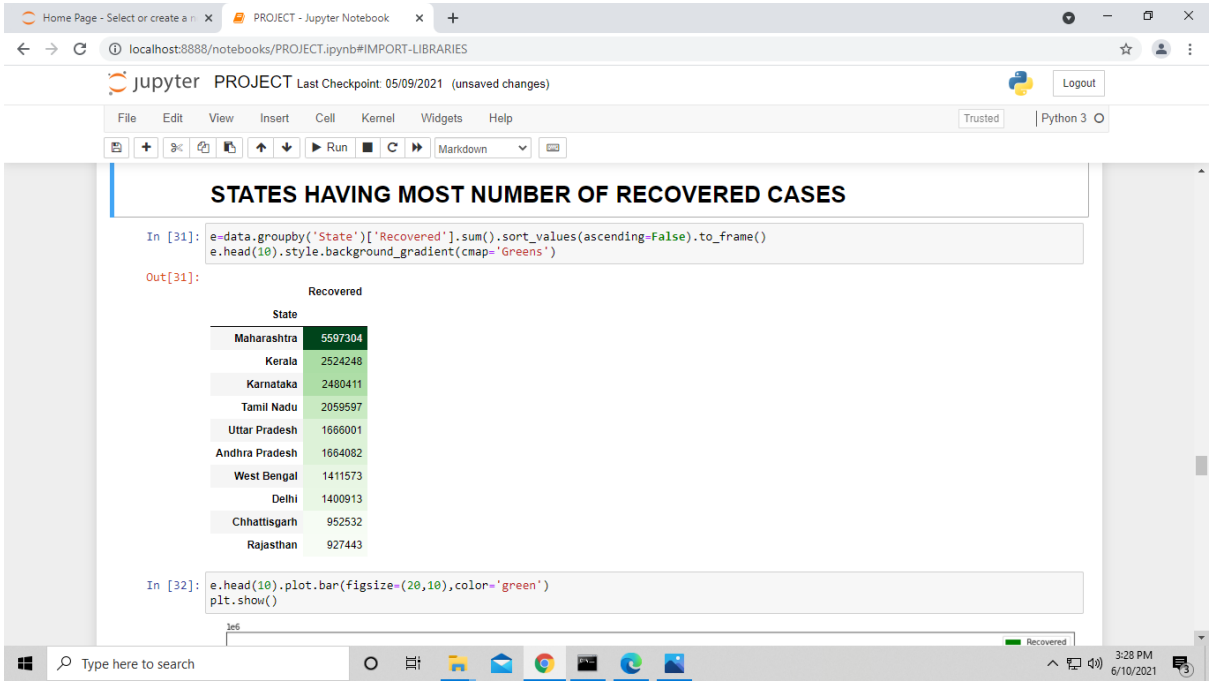


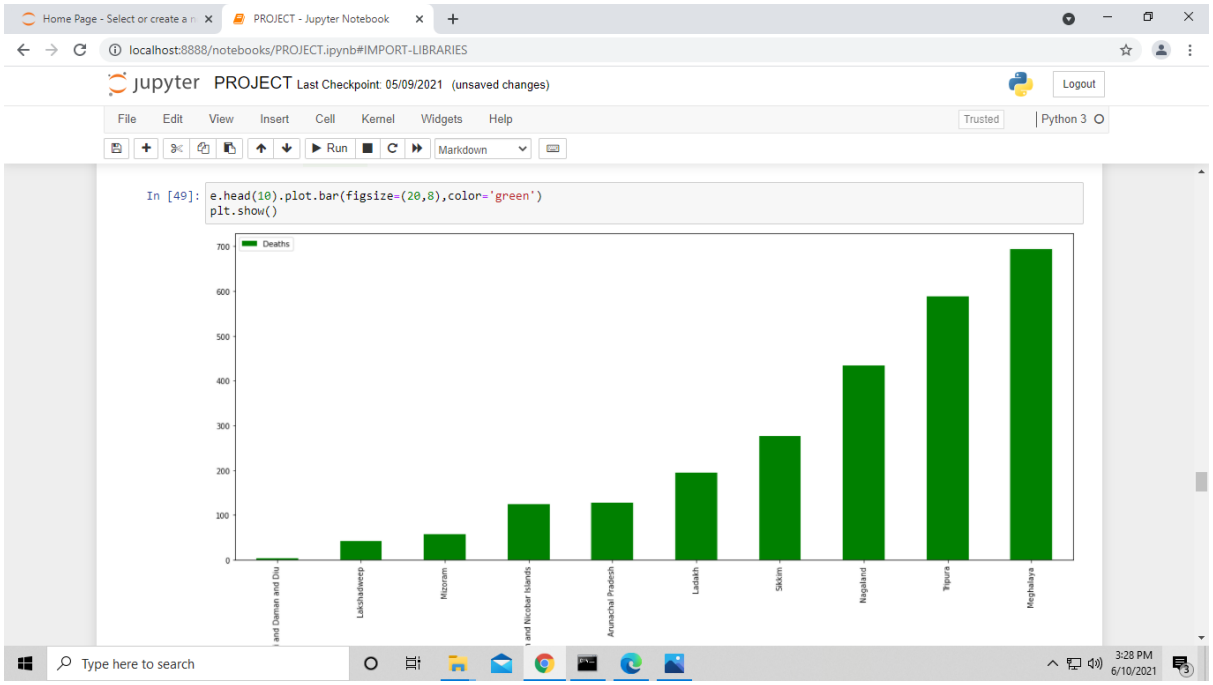
*PLOTTING DATA IN FORM OF BAR CHARTS

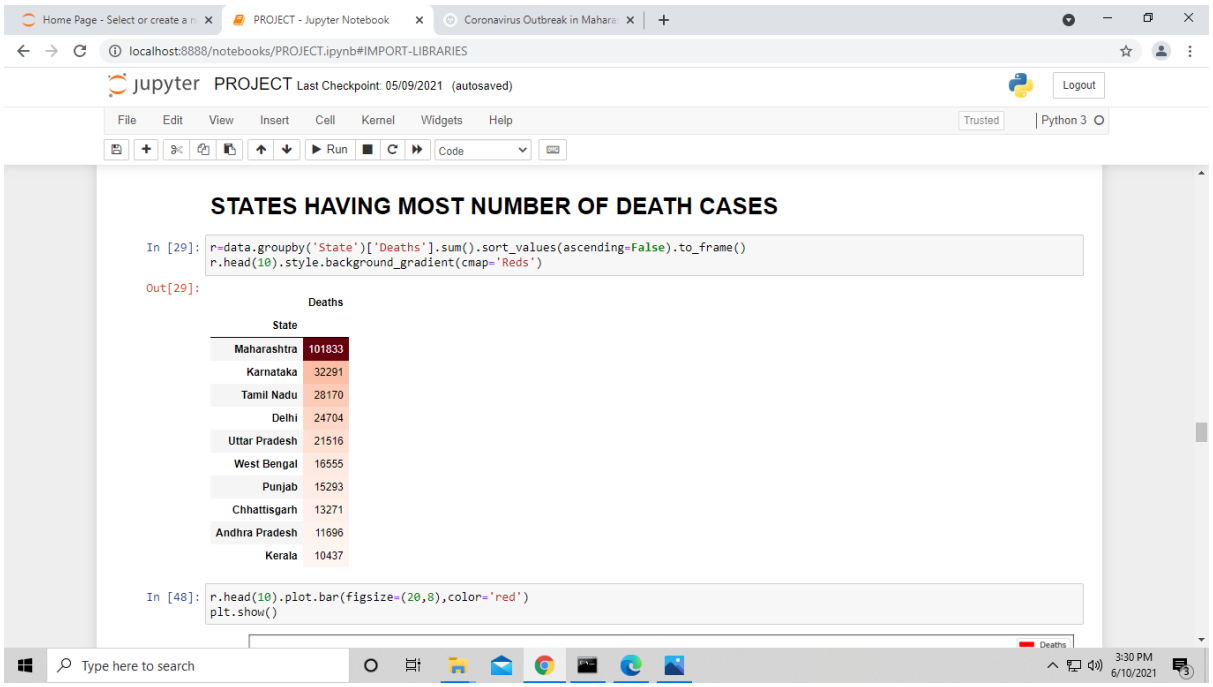


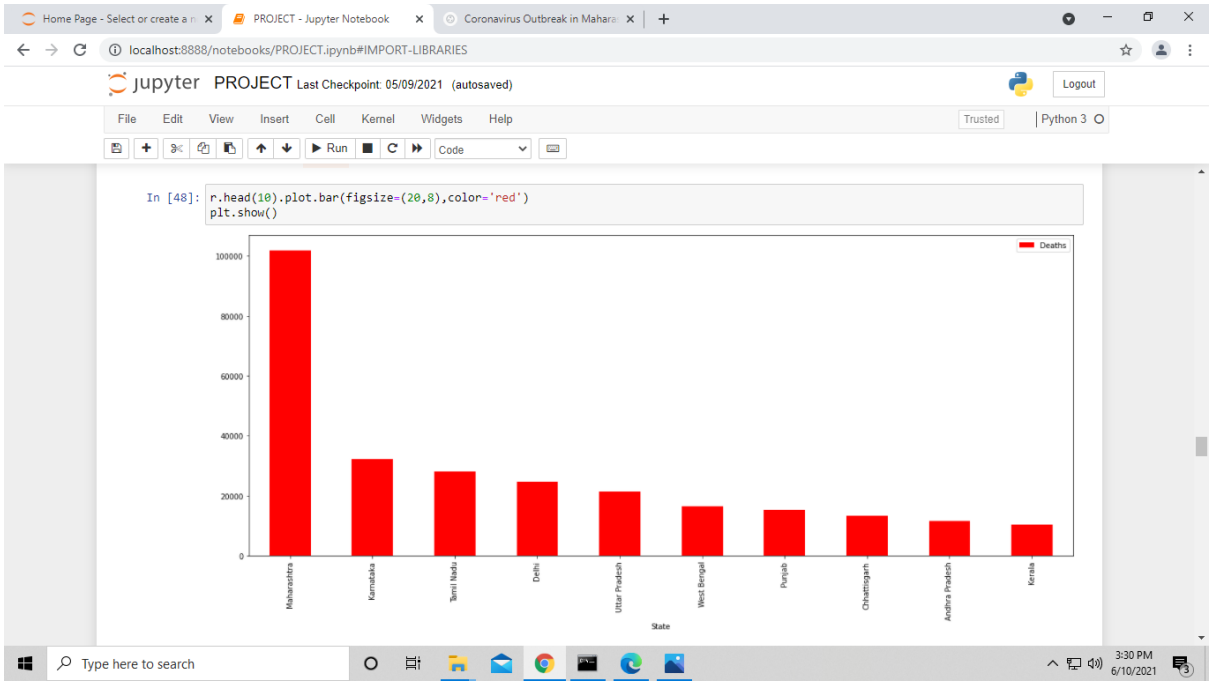

Home Page - Select or create a n
PROJECT - Jupyter Notebook
localhost:8888/notebooks/PROJECT.ipynb#IMPORT-LIBRARIES
jupyter PROJECT Last Checkpoint: 05/09/2021 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Markdown
STATES HAVING MOST NUMBER OF DEATH CASES
In [29]:
r=data.groupby('State')['Deaths'].sum().sort_values(ascending=False).to_frame()
r.head(10).style.background_gradient(cmap='Reds')
Out[29]:
Deaths
State
Maharashtra 101833
Karnataka 32291
Tamil Nadu 28170
Delhi 24704
Uttar Pradesh 21516
West Bengal 16555
Punjab 15293
Chhattisgarh 13271
Andhra Pradesh 11696
Kerala 10437
In [30]:
r.head(10).plot.bar(figsize=(20,10),color='red')
plt.show()
Deaths
Type here to search
3:27 PM
6/10/2021

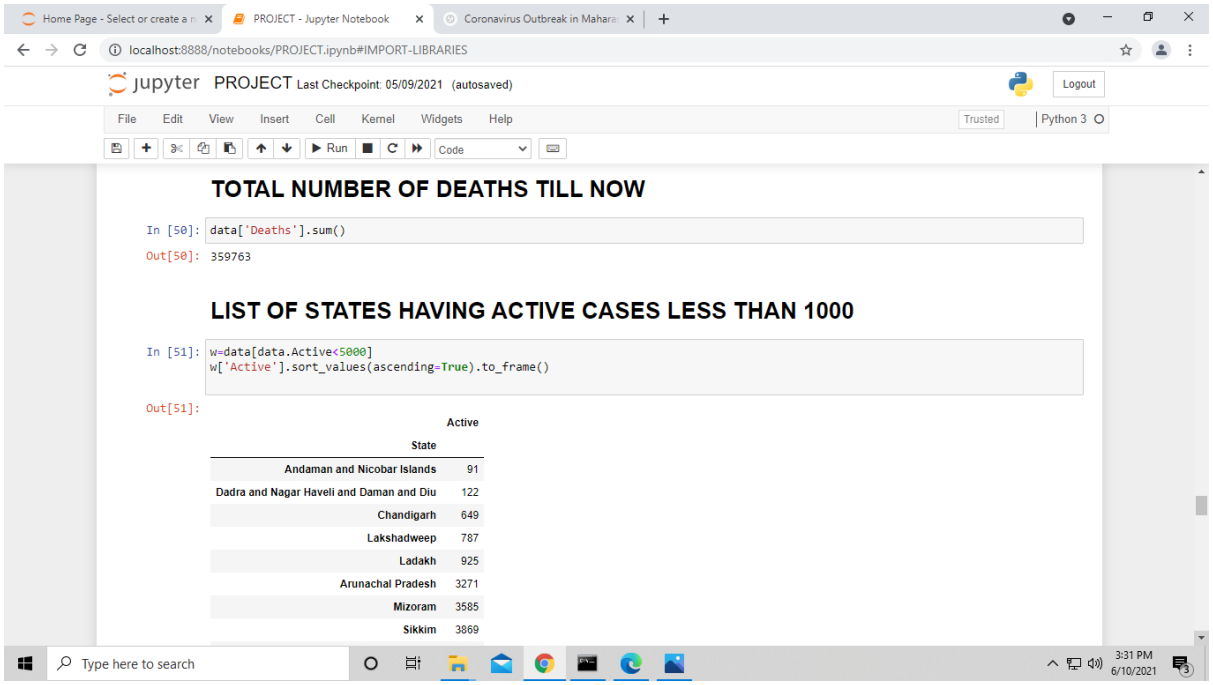

Home Page - Select or create a n
PROJECT - Jupyter Notebook
localhost:8888/notebooks/PROJECT.ipynb#IMPORT-LIBRARIES
jupyter PROJECT Last Checkpoint: 05/09/2021 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Markdown
In [48]:
r.head(10).plot.bar(figsize=(20,8),color='red')
plt.show()
Deaths
100000
80000
60000
40000
20000
0
Maharashtra
Karnataka
Tamil Nadu
Delhi
Uttar Pradesh
West Bengal
Punjab
Chhattisgarh
Andhra Pradesh
Kerala
State
Type here to search
3:28 PM
6/10/2021

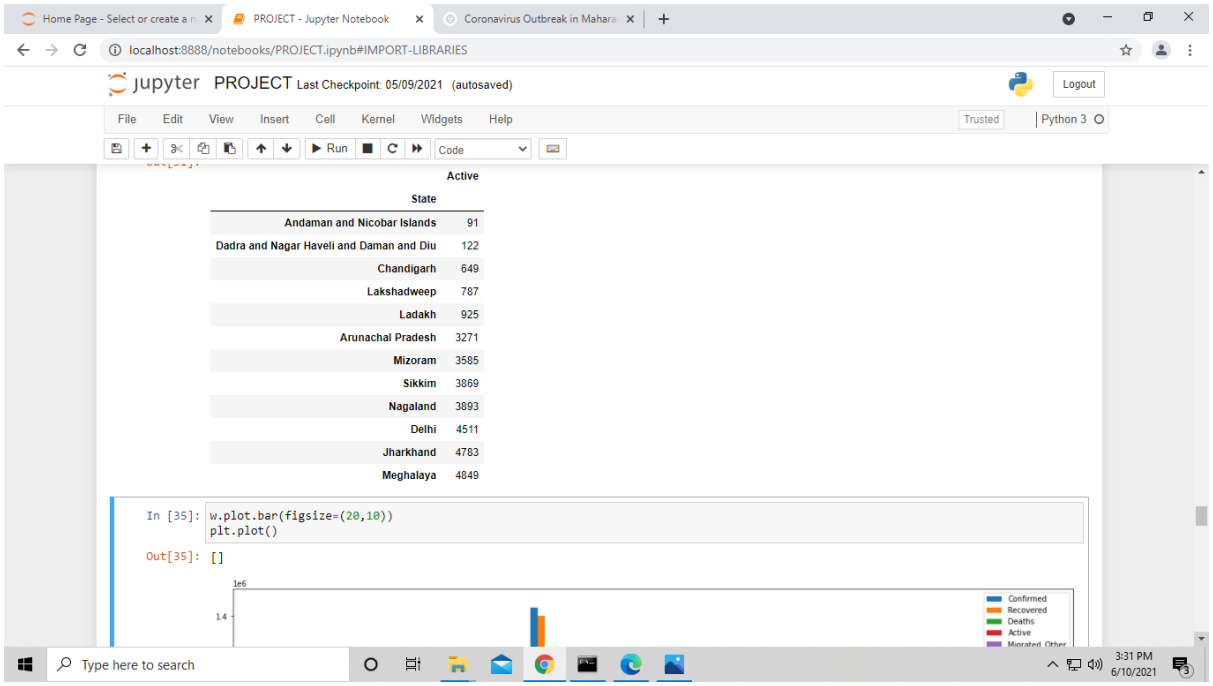

Home Page - Select or create a n
PROJECT - Jupyter Notebook
localhost:8888/notebooks/PROJECT.ipynb#IMPORT-LIBRARIES
jupyter PROJECT Last Checkpoint: 05/09/2021 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Markdown
STATES HAVING MOST NUMBER OF RECOVERED CASES
In [31]:
e=data.groupby('State')['Recovered'].sum().sort_values(ascending=False).to_frame()
e.head(10).style.background_gradient(cmap='Greens')
Out[31]:
Recovered
State
Maharashtra 5597304
Kerala 2524248
Karnataka 2480411
Tamil Nadu 2059597
Uttar Pradesh 1666001
Andhra Pradesh 1664082
West Bengal 1411573
Delhi 1400913
Chhattisgarh 952532
Rajasthan 927443
In [32]:
e.head(10).plot.bar(figsize=(20,10),color='green')
plt.show()
1e6
Recovered
Type here to search
3:28 PM
6/10/2021

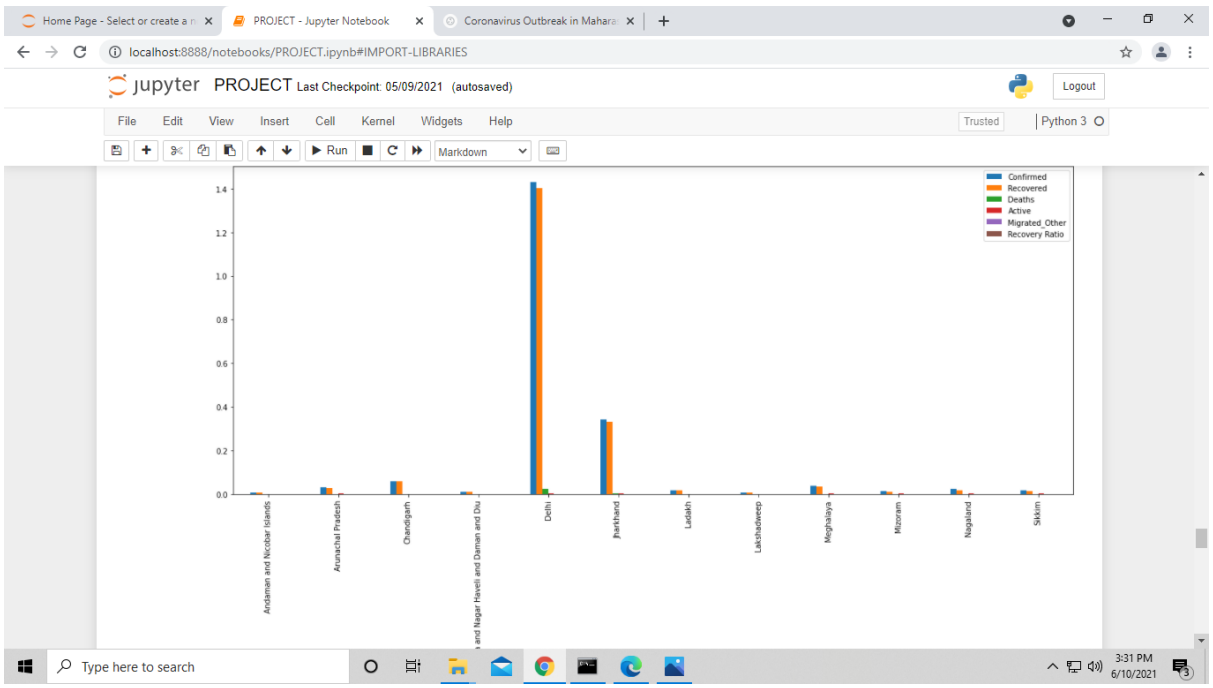

Home Page - Select or create a n
PROJECT - Jupyter Notebook
localhost:8888/notebooks/PROJECT.ipynb#IMPORT-LIBRARIES
jupyter PROJECT Last Checkpoint: 05/09/2021 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Markdown
In [49]:
e.head(10).plot.bar(figsize=(20,8),color='green')
plt.show()
Deaths
700
600
500
400
300
200
100
0
Lakshadweep
Nizoram
Sikkim
Nagaland
Tripura
Meghalaya
Type here to search
3:28 PM
6/10/2021

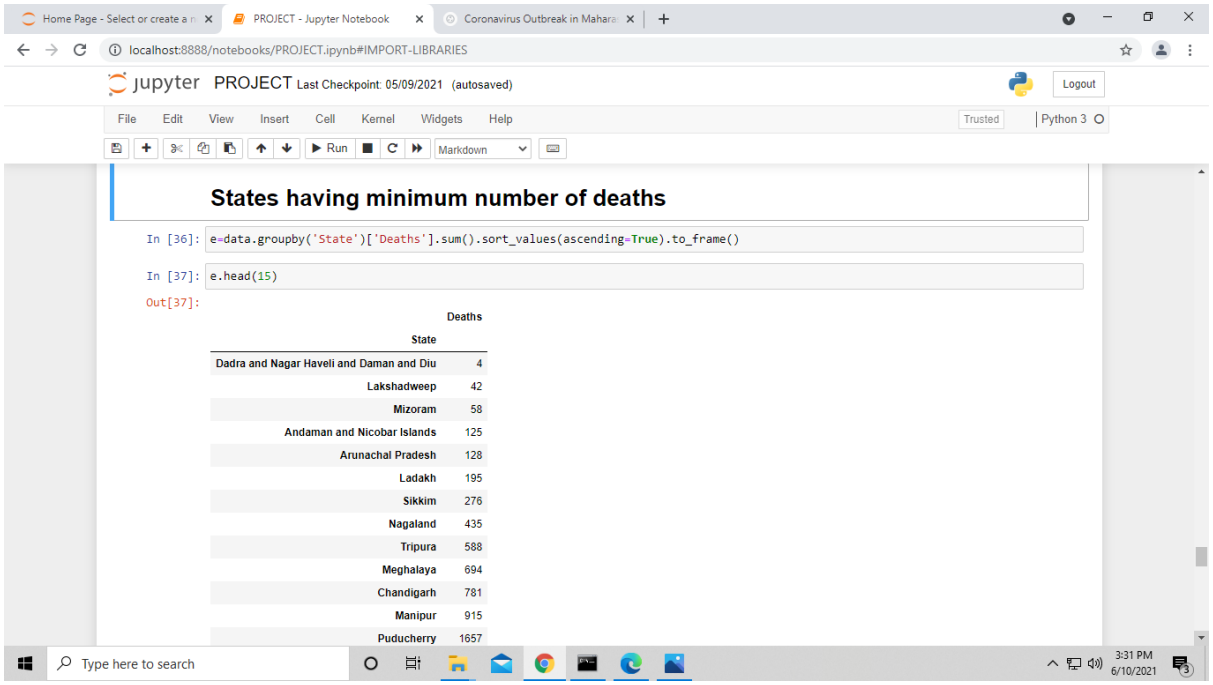

Home Page - Select or create a n
PROJECT - Jupyter Notebook
Coronavirus Outbreak in Mahara
localhost:8888/notebooks/PROJECT.ipynb#IMPORT-LIBRARIES
jupyter PROJECT Last Checkpoint: 05/09/2021 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
STATES HAVING MOST NUMBER OF DEATH CASES
In [29]:
r=data.groupby('State')['Deaths'].sum().sort_values(ascending=False).to_frame()
r.head(10).style.background_gradient(cmap='Reds')
Out[29]:
Deaths
State
Maharashtra 101833
Karnataka 32291
Tamil Nadu 28170
Delhi 24704
Uttar Pradesh 21516
West Bengal 16555
Punjab 15293
Chhattisgarh 13271
Andhra Pradesh 11696
Kerala 10437
In [48]:
r.head(10).plot.bar(figsize=(20,8),color='red')
plt.show()
Deaths
Type here to search
3:30 PM
6/10/2021


Home Page - Select or create a n
PROJECT - Jupyter Notebook
Coronavirus Outbreak in Mahara
localhost:8888/notebooks/PROJECT.ipynb#IMPORT-LIBRARIES
jupyter PROJECT Last Checkpoint: 05/09/2021 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [48]:
r.head(10).plot.bar(figsize=(20,8),color='red')
plt.show()
Deaths
100000
80000
60000
40000
20000
0
Maharashtra
Karnataka
Tamil Nadu
Delhi
Uttar Pradesh
West Bengal
Punjab
Chhattisgarh
Andhra Pradesh
Kerala
State
Type here to search
3:30 PM
6/10/2021

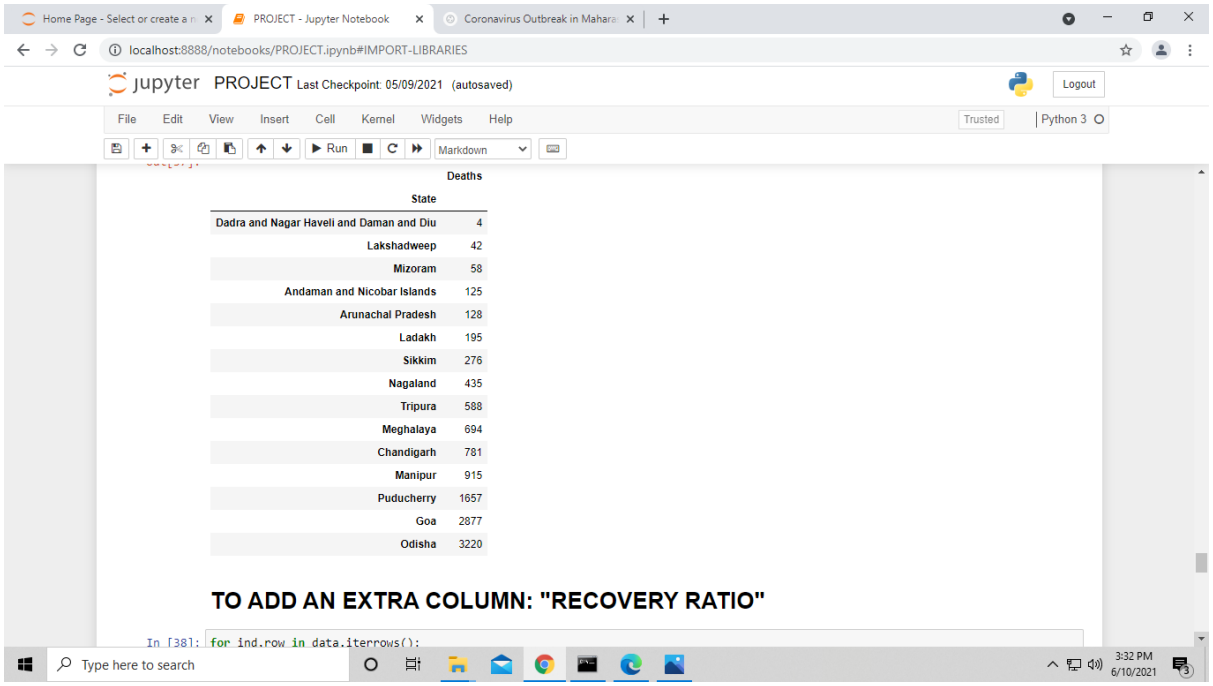

Home Page - Select or create a n
PROJECT - Jupyter Notebook
Coronavirus Outbreak in Mahara
localhost:8888/notebooks/PROJECT.ipynb#IMPORT-LIBRARIES
jupyter PROJECT Last Checkpoint: 05/09/2021 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
TOTAL NUMBER OF DEATHS TILL NOW
In [50]:
data['Deaths'].sum()
Out[50]:
359763
LIST OF STATES HAVING ACTIVE CASES LESS THAN 1000
In [51]:
w=data[data.Active<5000]
w['Active'].sort_values(ascending=True).to_frame()
Out[51]:
Active
State
Andaman and Nicobar Islands 91
Dadra and Nagar Haveli and Daman and Diu 122
Chandigarh 649
Lakshadweep 787
Ladakh 925
Arunachal Pradesh 3271
Mizoram 3585
Sikkim 3869
Type here to search
3:31 PM
6/10/2021

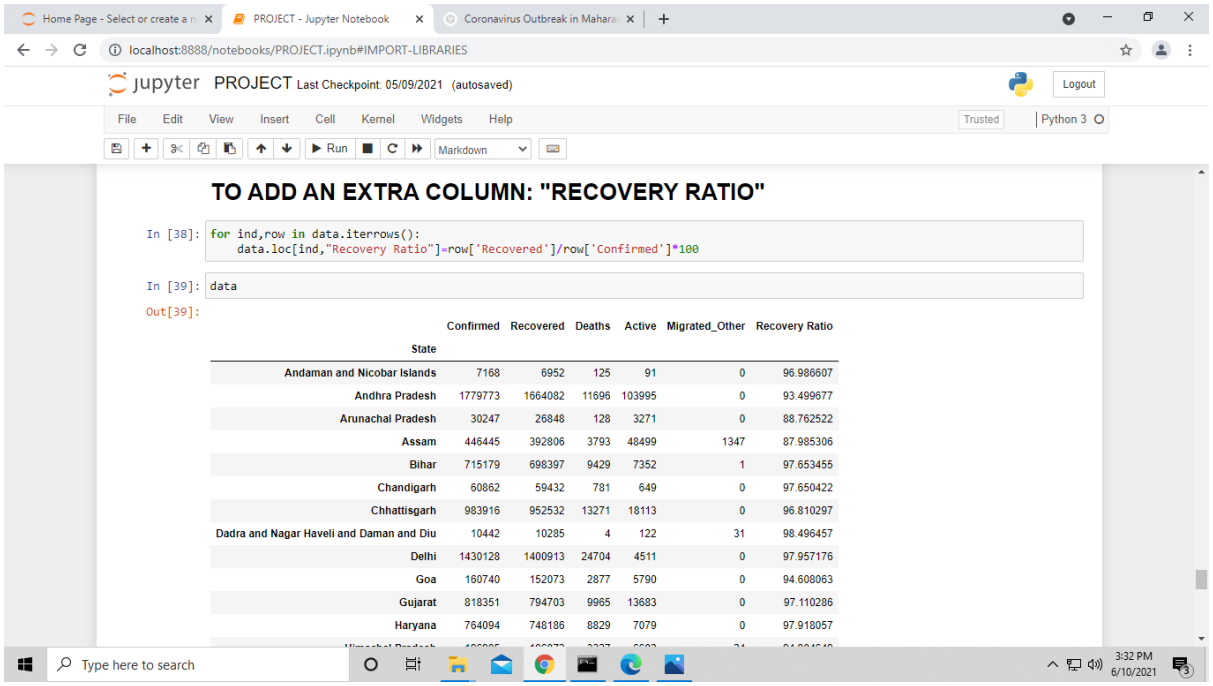

Home Page - Select or create a n
PROJECT - Jupyter Notebook
Coronavirus Outbreak in Mahara
localhost:8888/notebooks/PROJECT.ipynb#IMPORT-LIBRARIES
jupyter PROJECT Last Checkpoint: 05/09/2021 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
Active
State
Andaman and Nicobar Islands 91
Dadra and Nagar Haveli and Daman and Diu 122
Chandigarh 649
Lakshadweep 787
Ladakh 925
Arunachal Pradesh 3271
Mizoram 3585
Sikkim 3869
Nagaland 3893
Delhi 4511
Jharkhand 4783
Meghalaya 4849
In [35]:
w.plot.bar(figsize=(20,10))
plt.plot()
Out[35]: []
1e6
1.4
Confirmed
Recovered
Deaths
Active
Migrated_Other
Type here to search
3:31 PM
6/10/2021


Home Page - Select or create a n
PROJECT - Jupyter Notebook
Coronavirus Outbreak in Mahara
localhost:8888/notebooks/PROJECT.ipynb#IMPORT-LIBRARIES
jupyter PROJECT Last Checkpoint: 05/09/2021 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Markdown
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Confirmed
Recovered
Deaths
Active
Migrated_Other
Recovery Ratio
Andaman and Nicobar Islands
Arunachal Pradesh
Chandigarh
Delhi
Jharkhand
Ladakh
Lakshadweep
Meghalaya
Mizoram
Nagaland
Sikkim
Type here to search
3:31 PM
6/10/2021

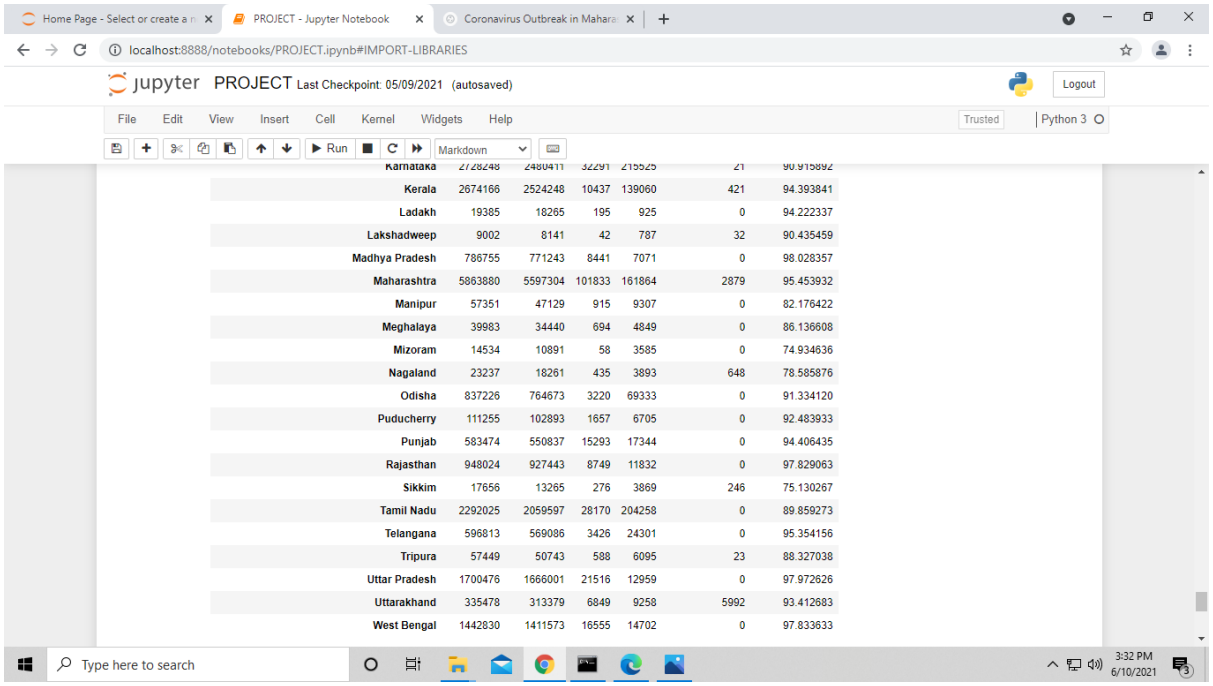


*ADDING AN EXTRA COLUMN :RECOVERY RATIO



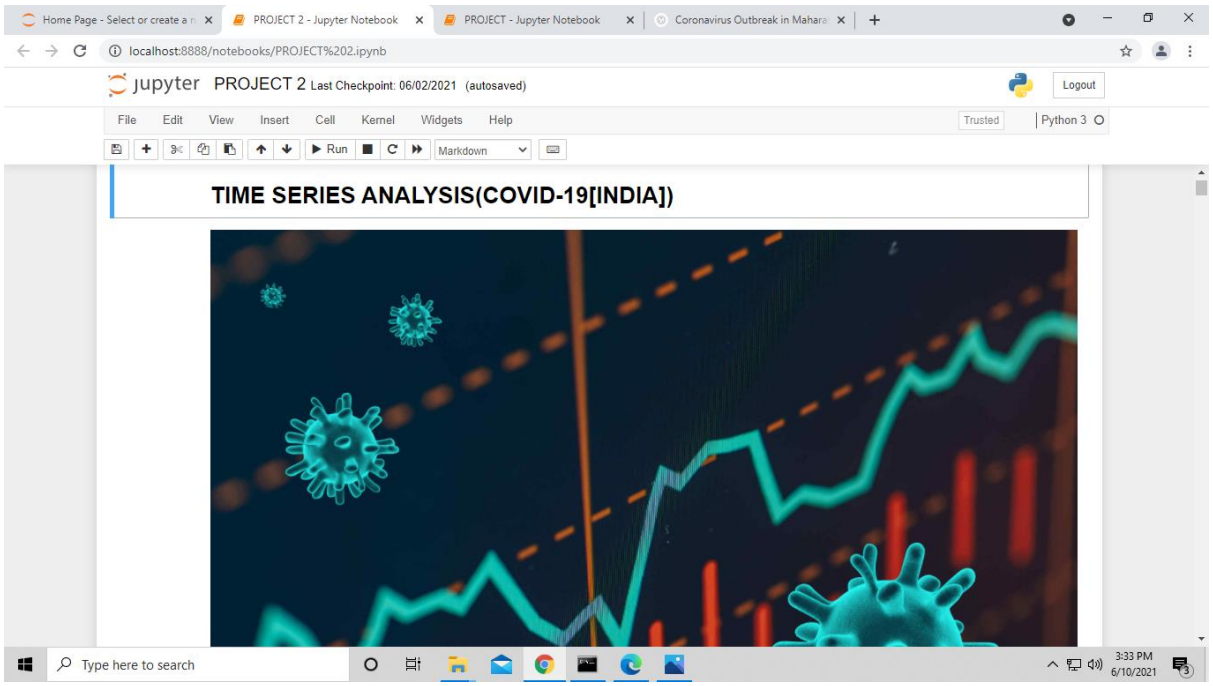


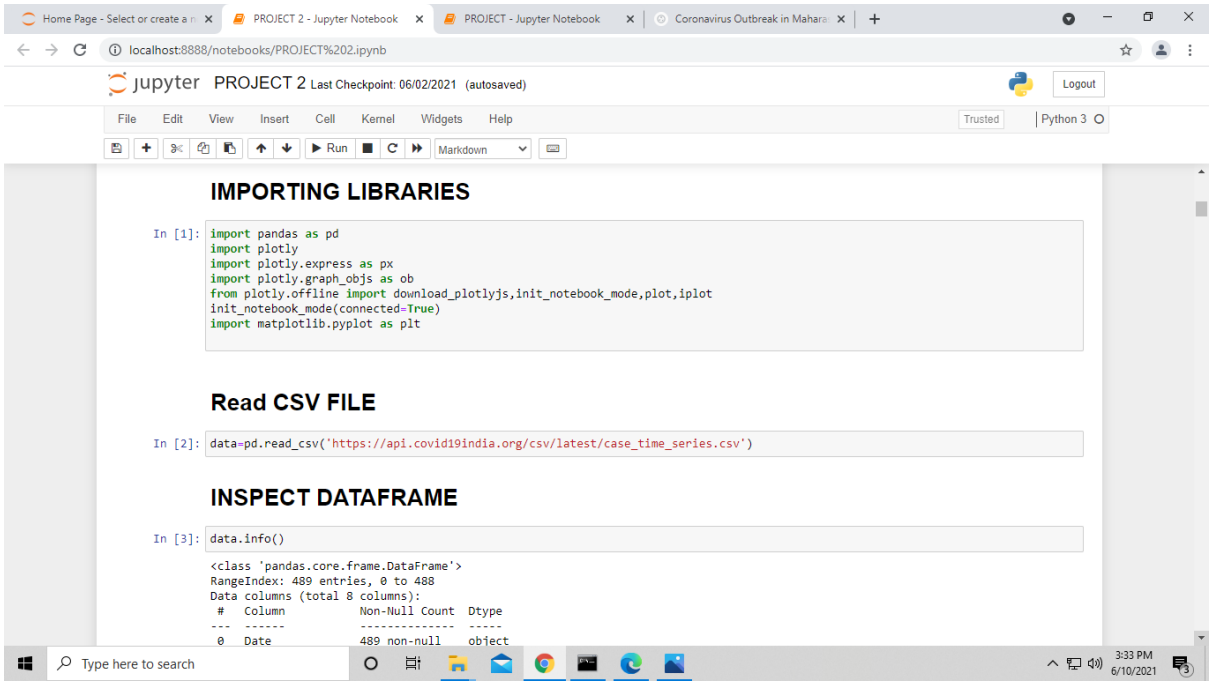
State	Confirmed	Recovered	Deaths	Active	Migrated_Other	Recovery Ratio
Andaman and Nicobar Islands	7168	6952	125	91	0	96.986607
Andhra Pradesh	1779773	1664082	11696	103995	0	93.499677
Arunachal Pradesh	30247	26848	128	3271	0	88.762522
Assam	446445	392806	3793	48499	1347	87.985306
Bihar	715179	698397	9429	7352	1	97.653455
Chandigarh	60862	59432	781	649	0	97.650422
Chhattisgarh	983916	952532	13271	18113	0	96.810297
Dadra and Nagar Haveli and Daman and Diu	10442	10285	4	122	31	98.496457
Delhi	1430128	1400913	24704	4511	0	97.957176
Goa	160740	152073	2877	5790	0	94.608063
Gujarat	818351	794703	9965	13683	0	97.110286
Haryana	764094	748186	8829	7079	0	97.918057

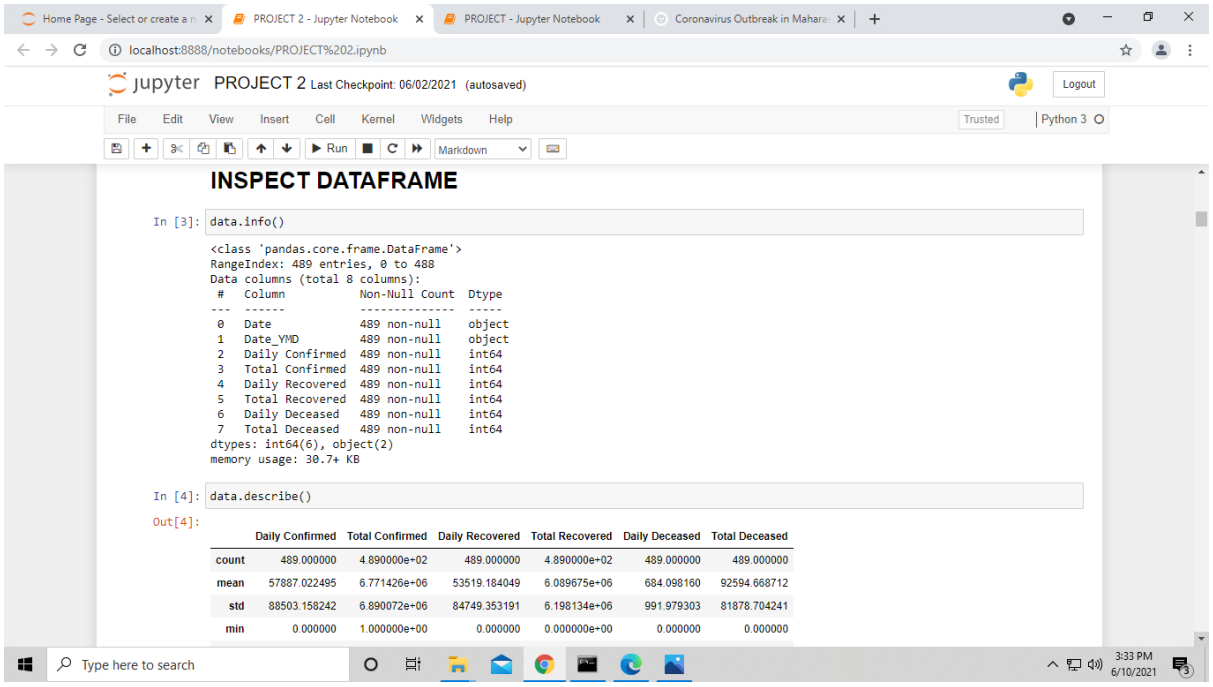


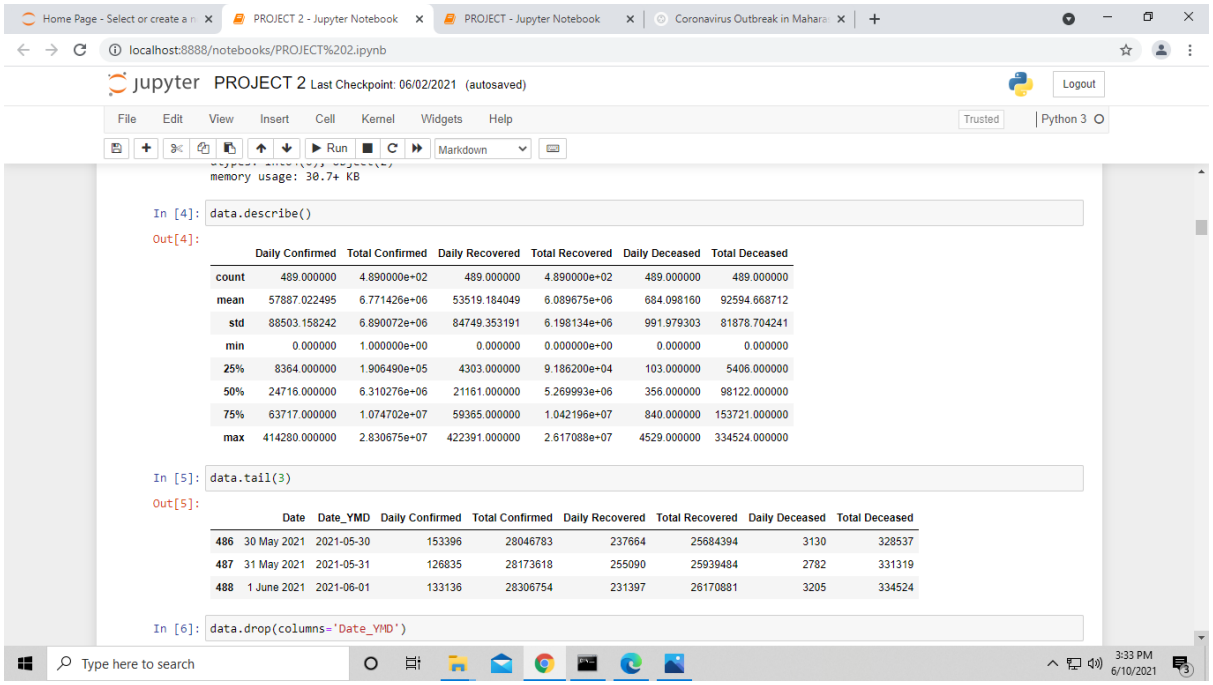
Karnataka	2728248	2480411	32291	215525	21	90.915892
Kerala	2674166	2524248	10437	139060	421	94.393841
Ladakh	19385	18265	195	925	0	94.222337
Lakshadweep	9002	8141	42	787	32	90.435459
Madhya Pradesh	786755	771243	8441	7071	0	98.028357
Maharashtra	5863880	5597304	101833	161864	2879	95.453932
Manipur	57351	47129	915	9307	0	82.176422
Meghalaya	39983	34440	694	4849	0	86.136608
Mizoram	14534	10891	58	3585	0	74.934636
Nagaland	23237	18261	435	3893	648	78.585876
Odisha	837226	764673	3220	69333	0	91.334120
Puducherry	111255	102893	1657	6705	0	92.483933
Punjab	583474	550837	15293	17344	0	94.406435
Rajasthan	948024	927443	8749	11832	0	97.829063
Sikkim	17656	13265	276	3869	246	75.130267
Tamil Nadu	2292025	2059597	28170	204258	0	89.859273
Telangana	596813	569086	3426	24301	0	95.354156
Tripura	57449	50743	588	6095	23	88.327038
Uttar Pradesh	1700476	1666001	21516	12959	0	97.972626
Uttarakhand	335478	313379	6849	9258	5992	93.412683
West Bengal	1442830	1411573	16555	14702	0	97.833633

*FETCHING ANOTHER DATASET TO ANALYSE TIME SERIES FOR COVID-19 INDIA

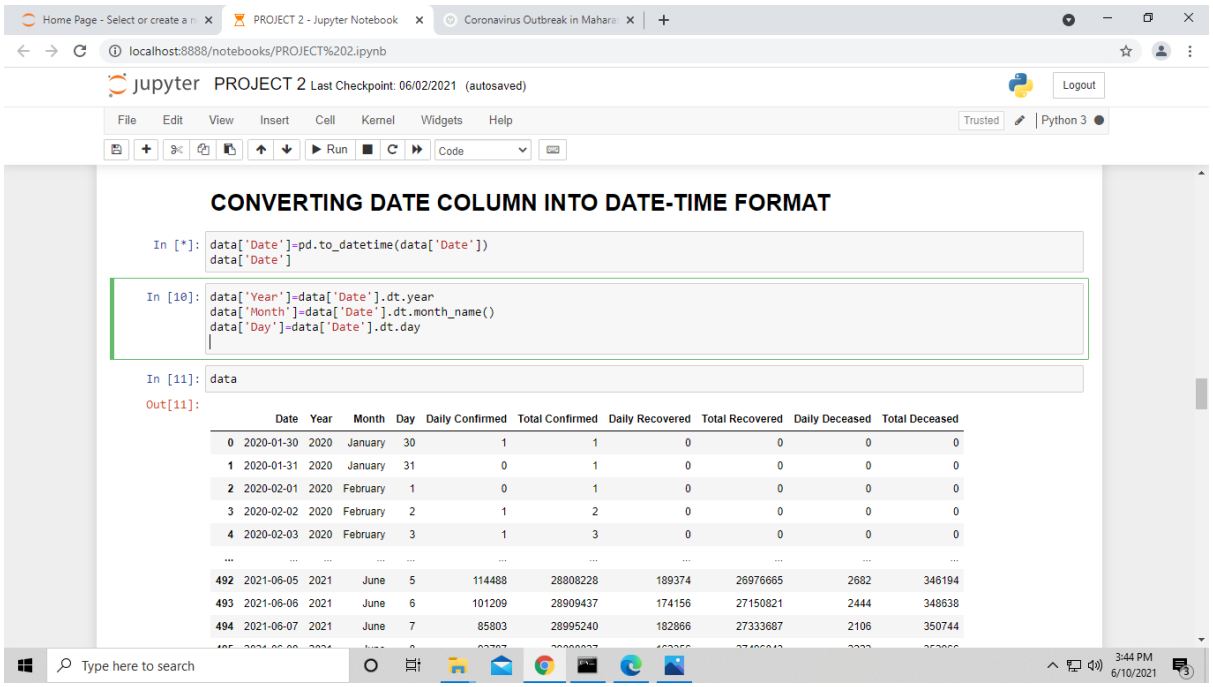



Home Page - Select or create a n
PROJECT 2 - Jupyter Notebook
PROJECT - Jupyter Notebook
Coronavirus Outbreak in Mahara
localhost:8888/notebooks/PROJECT%202.ipynb
jupyter PROJECT 2 Last Checkpoint: 06/02/2021 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Markdown
IMPORTING LIBRARIES
In [1]:
import pandas as pd
import plotly
import plotly.express as px
import plotly.graph_objs as ob
from plotly.offline import download_plotlyjs,init_notebook_mode,plot,iplot
init_notebook_mode(connected=True)
import matplotlib.pyplot as plt
Read CSV FILE
In [2]:
data=pd.read_csv('https://api.covid19india.org/csv/latest/case_time_series.csv')
INSPECT DATAFRAME
In [3]:
data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 489 entries, 0 to 488
Data columns (total 8 columns):
Column Non-Null Count Dtype
--- ------ -------------- -----
0 Date 489 non-null object
Type here to search
3:33 PM
6/10/2021

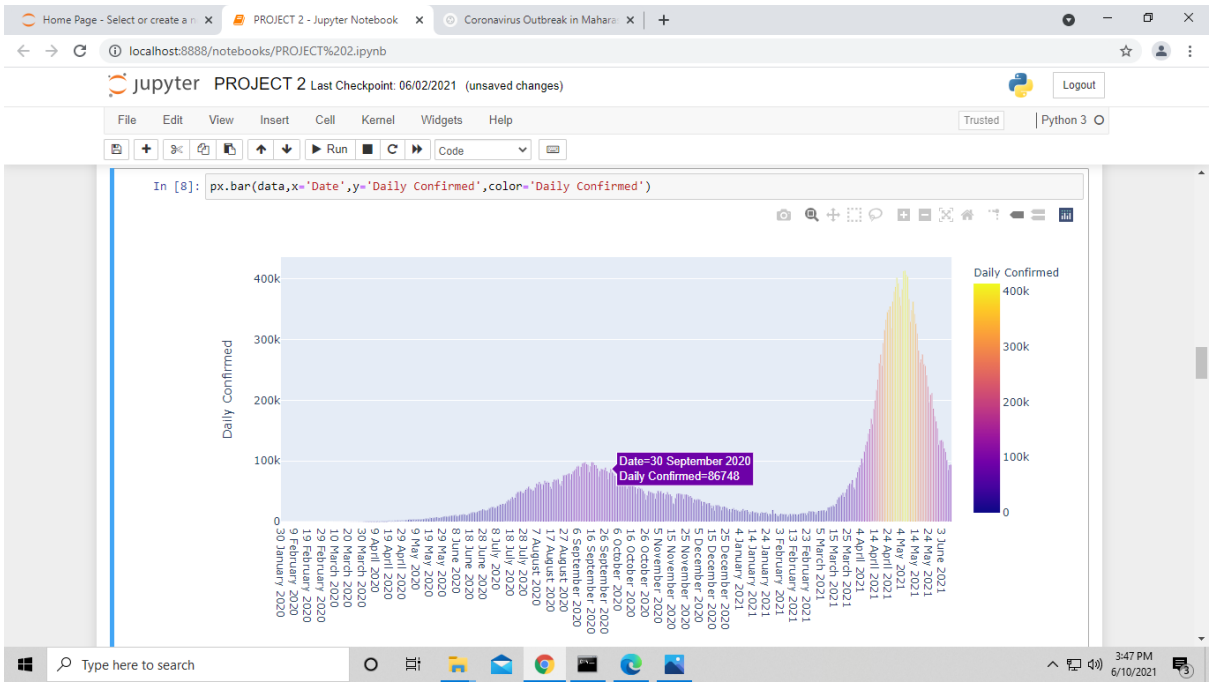

Home Page - Select or create a n
PROJECT 2 - Jupyter Notebook
PROJECT - Jupyter Notebook
Coronavirus Outbreak in Mahara
localhost:8888/notebooks/PROJECT%202.ipynb
jupyter PROJECT 2 Last Checkpoint: 06/02/2021 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Markdown
INSPECT DATAFRAME
In [3]:
data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 489 entries, 0 to 488
Data columns (total 8 columns):
Column Non-Null Count Dtype
--- ------ -------------- -----
0 Date 489 non-null object
1 Date_YMD 489 non-null object
2 Daily Confirmed 489 non-null int64
3 Total Confirmed 489 non-null int64
4 Daily Recovered 489 non-null int64
5 Total Recovered 489 non-null int64
6 Daily Deceased 489 non-null int64
7 Total Deceased 489 non-null int64
dtypes: int64(6), object(2)
memory usage: 30.7+ KB
In [4]:
data.describe()
Out[4]:
Daily Confirmed Total Confirmed Daily Recovered Total Recovered Daily Deceased Total Deceased
count 489.000000 4.890000e+02 489.000000 4.890000e+02 489.000000 489.000000
mean 57887.022495 6.771426e+06 53519.184049 6.089675e+06 684.098160 92594.668712
std 88503.158242 6.890072e+06 84749.353191 6.198134e+06 991.979303 81878.704241
min 0.000000 1.000000e+00 0.000000 0.000000e+00 0.000000 0.000000
Type here to search
3:33 PM
6/10/2021

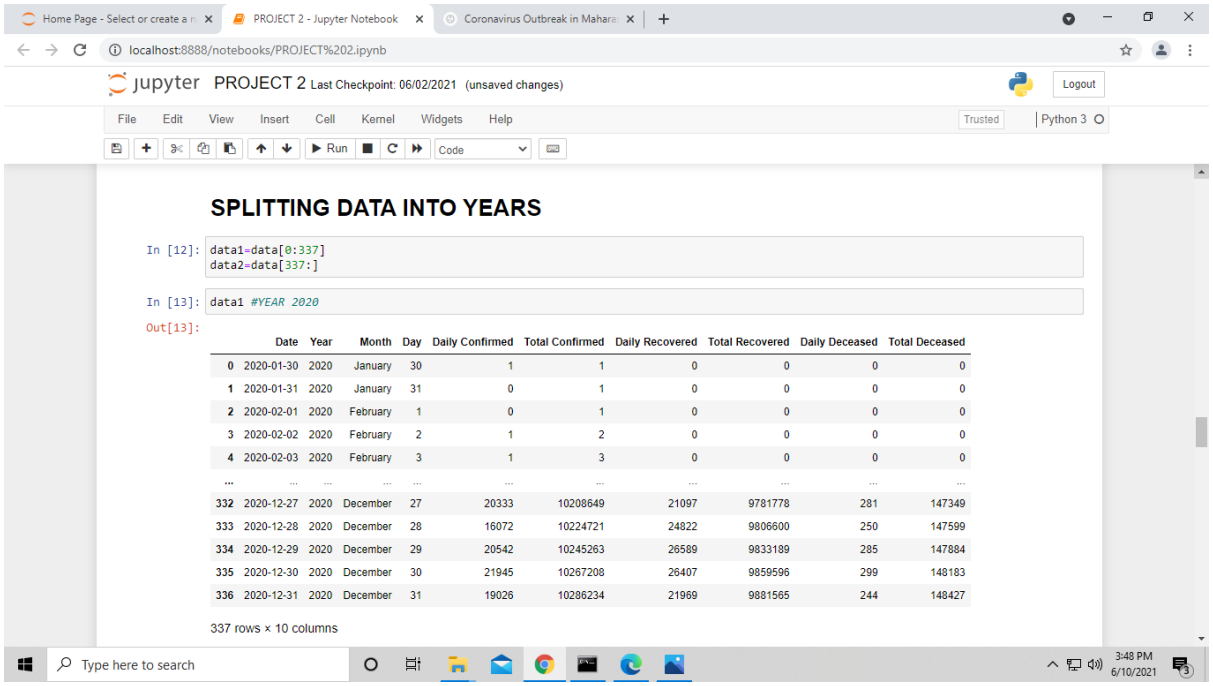


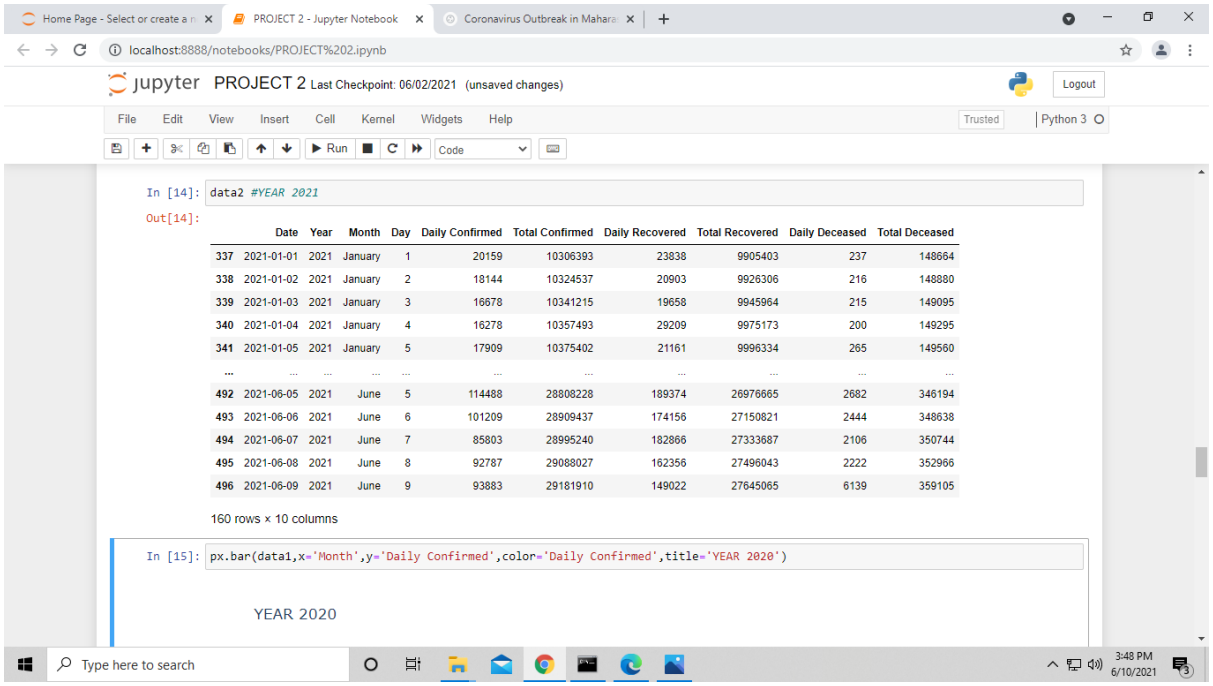
8) CONVERTING DATE COLUMN INTO DATE-TIME FORMAT

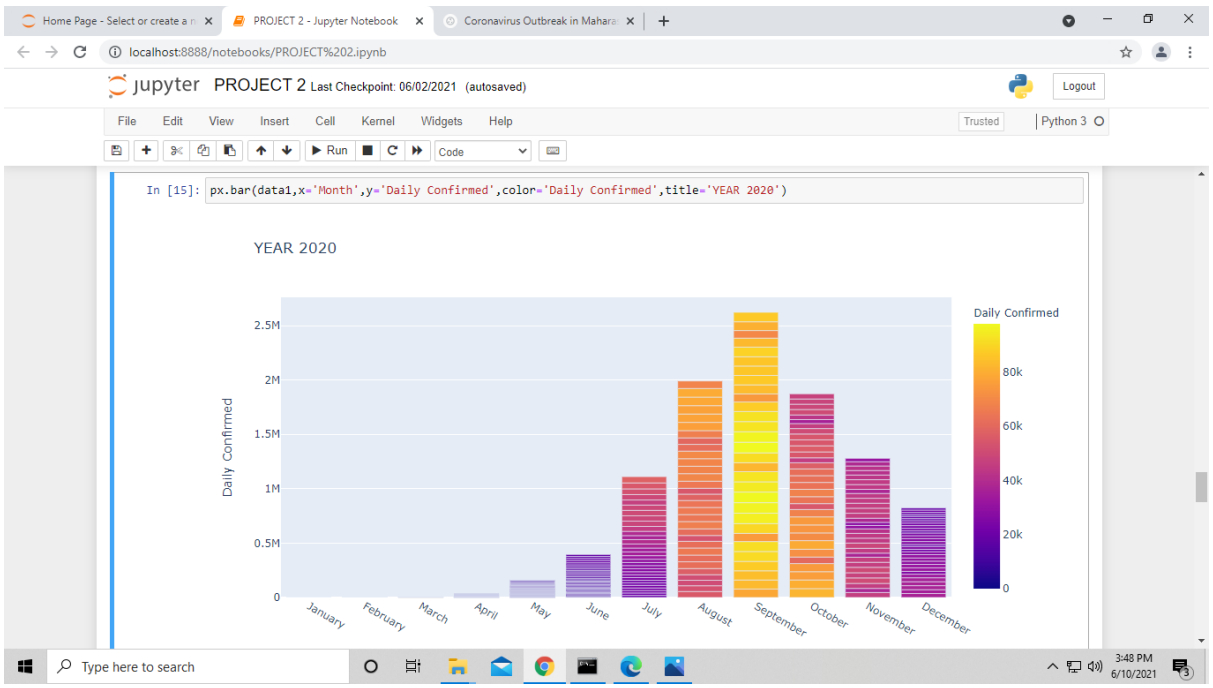


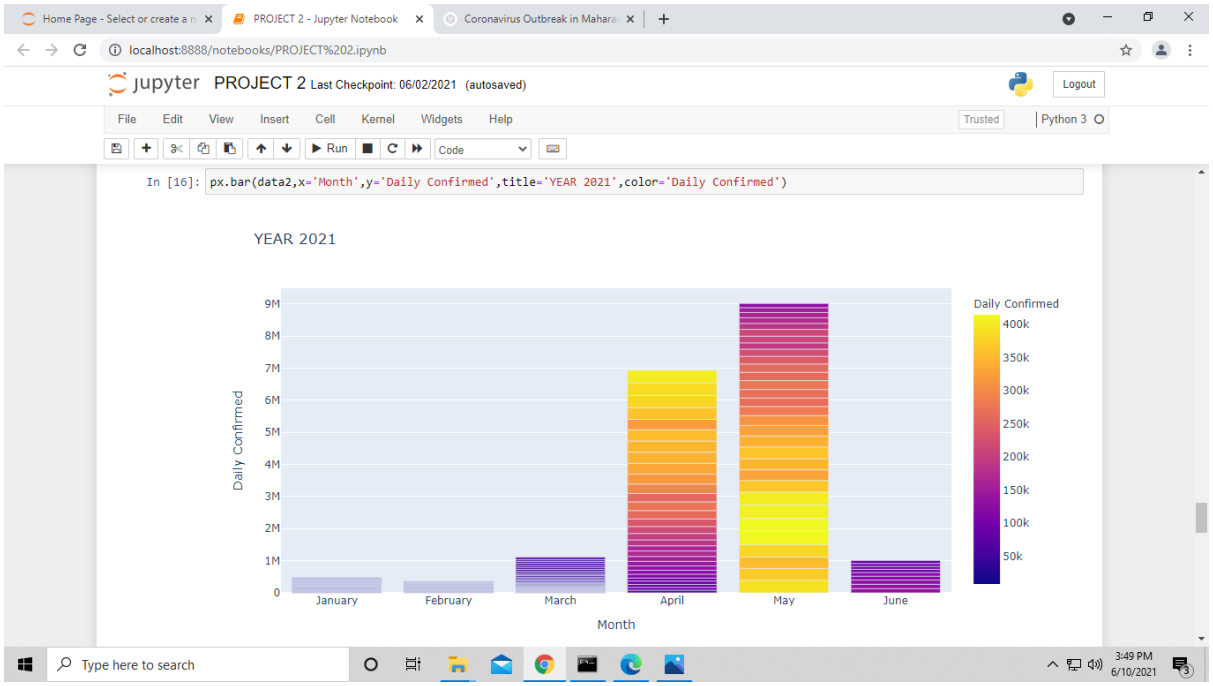
*PLOTTING BAR CHART OF DAY BY DAY CONFIRMED CASES USING PLOTLY.EXPRESS



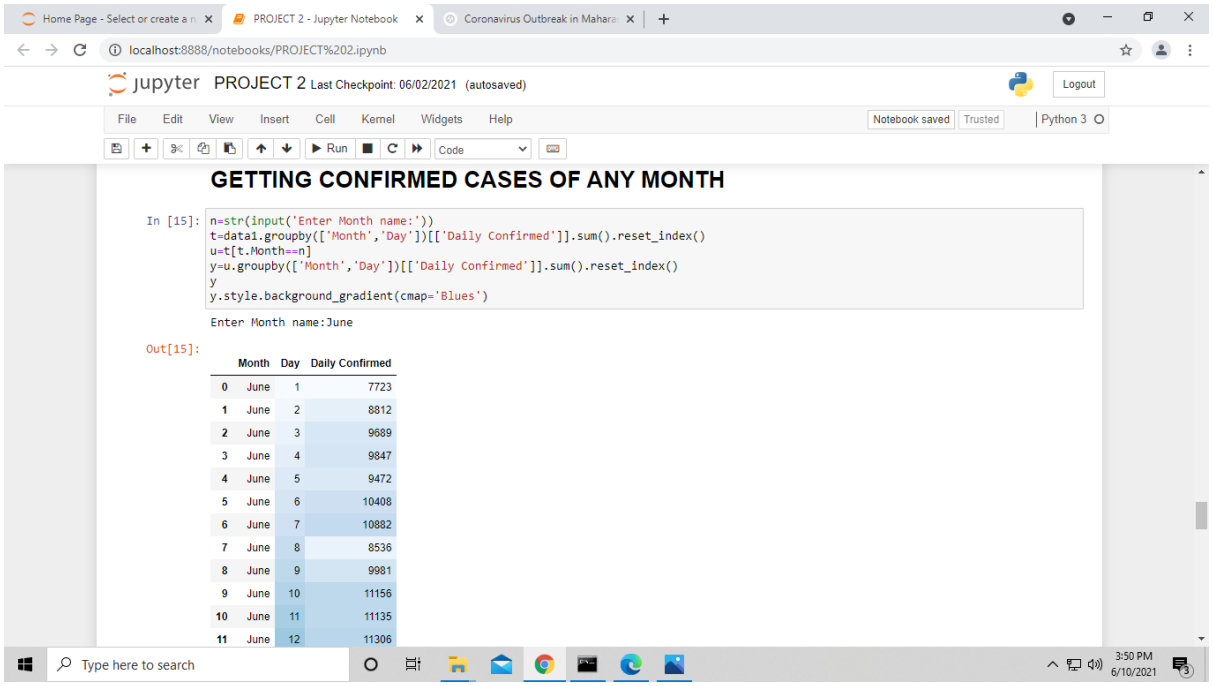


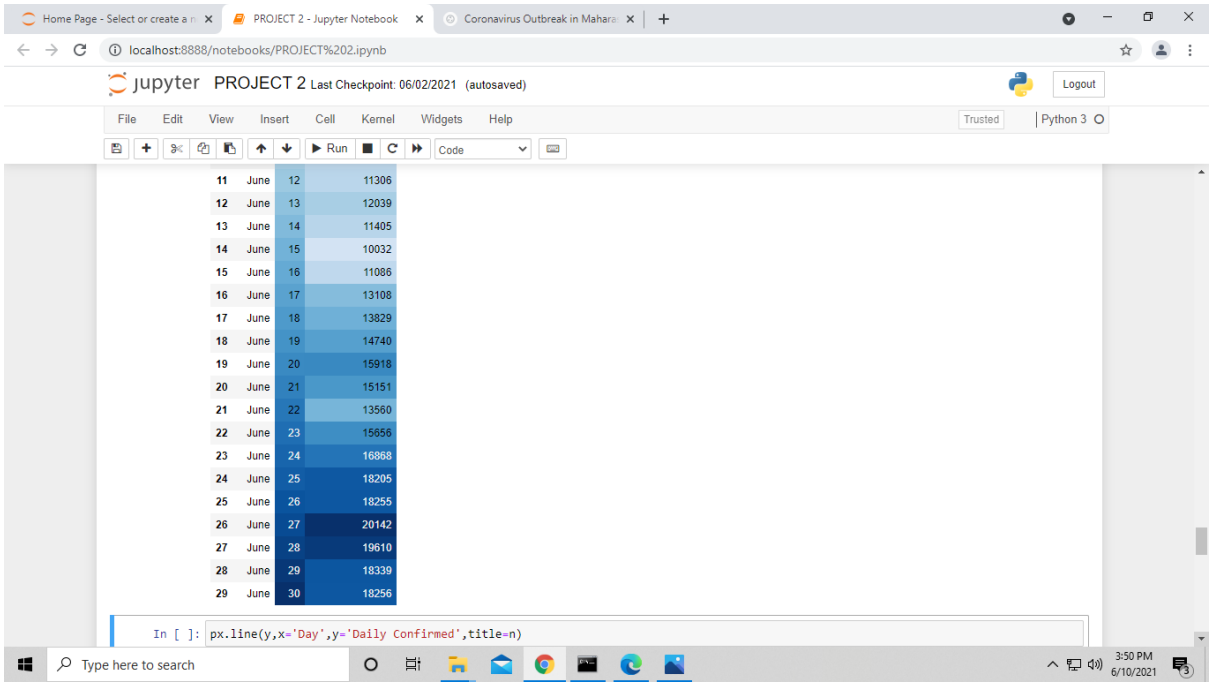

Home Page - Select or create a n
PROJECT 2 - Jupyter Notebook
Coronavirus Outbreak in Mahara
localhost:8888/notebooks/PROJECT%202.ipynb
jupyter PROJECT 2 Last Checkpoint: 06/02/2021 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [14]: data2 #YEAR 2021
Out[14]:
Date Year Month Day Daily Confirmed Total Confirmed Daily Recovered Total Recovered Daily Deceased Total Deceased
337 2021-01-01 2021 January 1 20159 10306393 23838 9905403 237 148664
338 2021-01-02 2021 January 2 18144 10324537 20903 9926306 216 148880
339 2021-01-03 2021 January 3 16678 10341215 19658 9945964 215 149095
340 2021-01-04 2021 January 4 16278 10357493 29209 9975173 200 149295
341 2021-01-05 2021 January 5 17909 10375402 21161 9996334 265 149560
492 2021-06-05 2021 June 5 114488 28808228 189374 26976665 2682 346194
493 2021-06-06 2021 June 6 101209 28909437 174156 27150821 2444 348638
494 2021-06-07 2021 June 7 85803 28995240 182866 27333687 2106 350744
495 2021-06-08 2021 June 8 92787 29088027 162356 27496043 2222 352966
496 2021-06-09 2021 June 9 93883 29181910 149022 27645065 6139 359105
160 rows × 10 columns
In [15]: px.bar(data1,x='Month',y='Daily Confirmed',color='Daily Confirmed',title='YEAR 2020')
YEAR 2020
Type here to search
3:48 PM
6/10/2021


Home Page - Select or create a n
PROJECT 2 - Jupyter Notebook
Coronavirus Outbreak in Mahara
localhost:8888/notebooks/PROJECT%202.ipynb
jupyter PROJECT 2 Last Checkpoint: 06/02/2021 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [15]: px.bar(data1,x='Month',y='Daily Confirmed',color='Daily Confirmed',title='YEAR 2020')
YEAR 2020
Daily Confirmed
2.5M
2M
1.5M
1M
0.5M
0
80k
60k
40k
20k
0
January
February
March
April
May
June
July
August
September
October
November
December
Type here to search
3:48 PM
6/10/2021

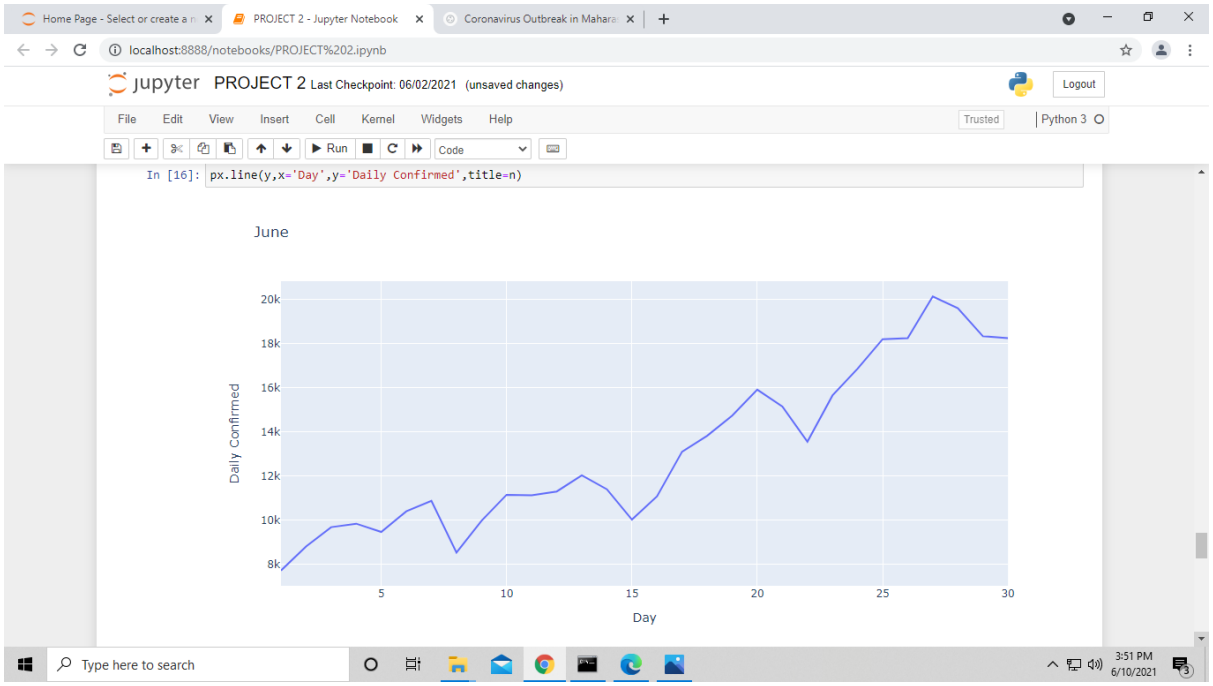


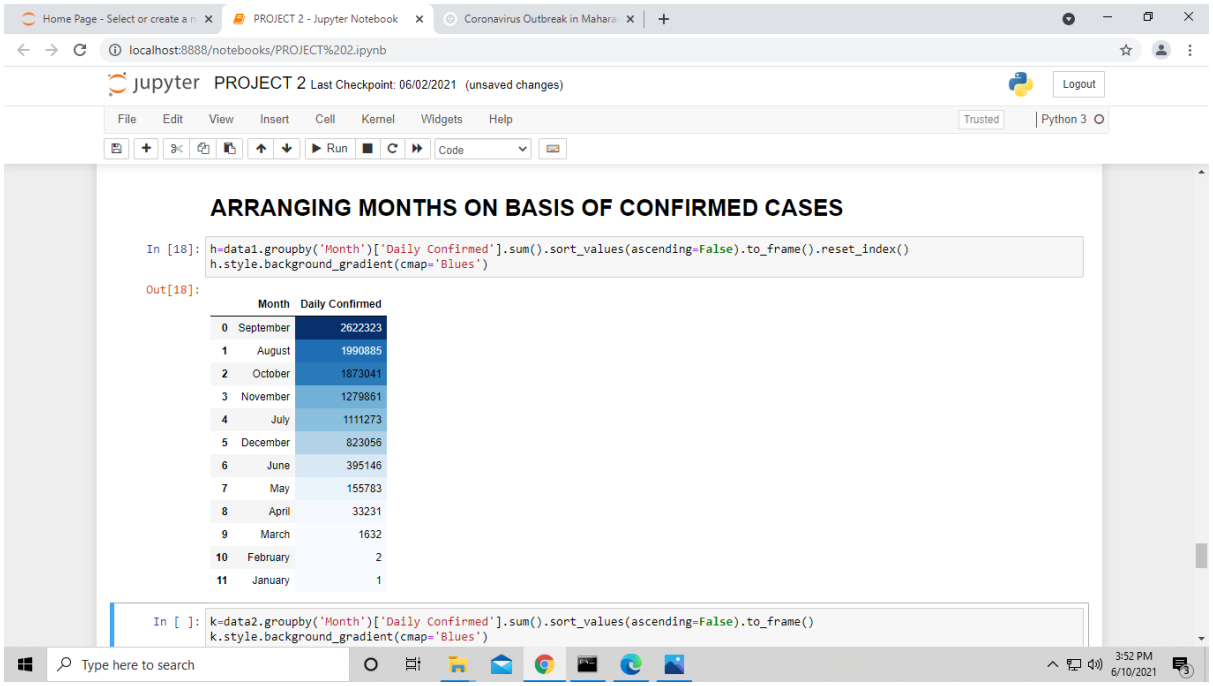
*SHOWING CONFIRMED CASES OF ANY MONTH

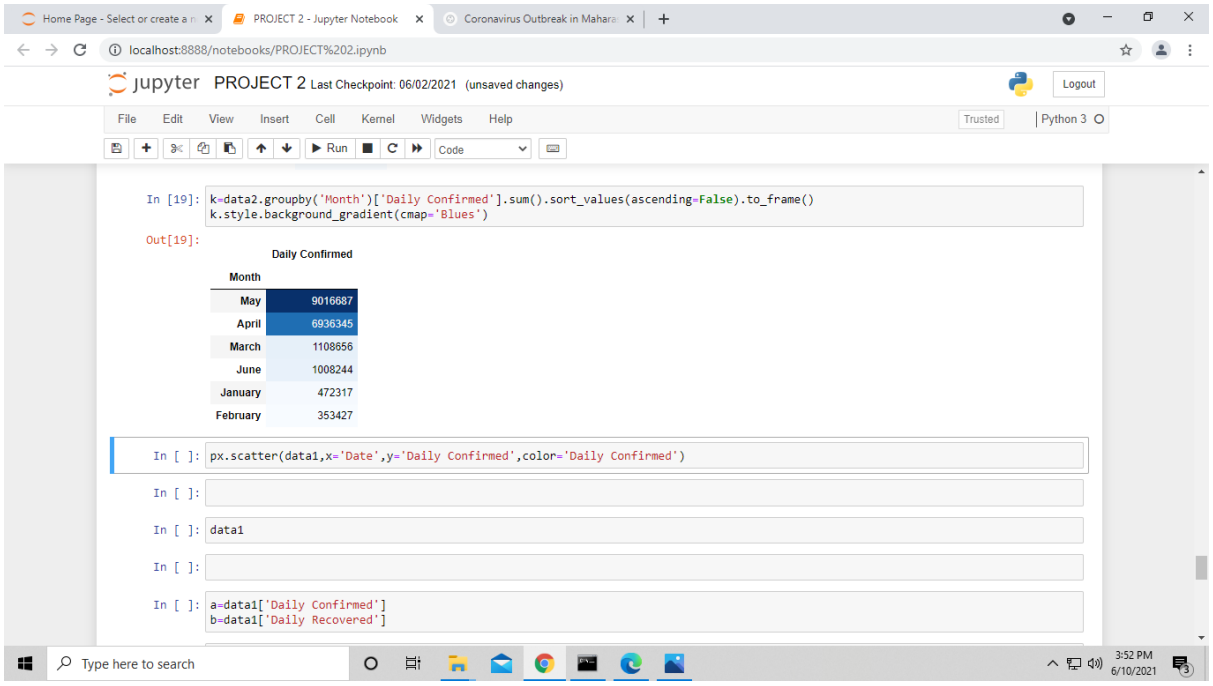




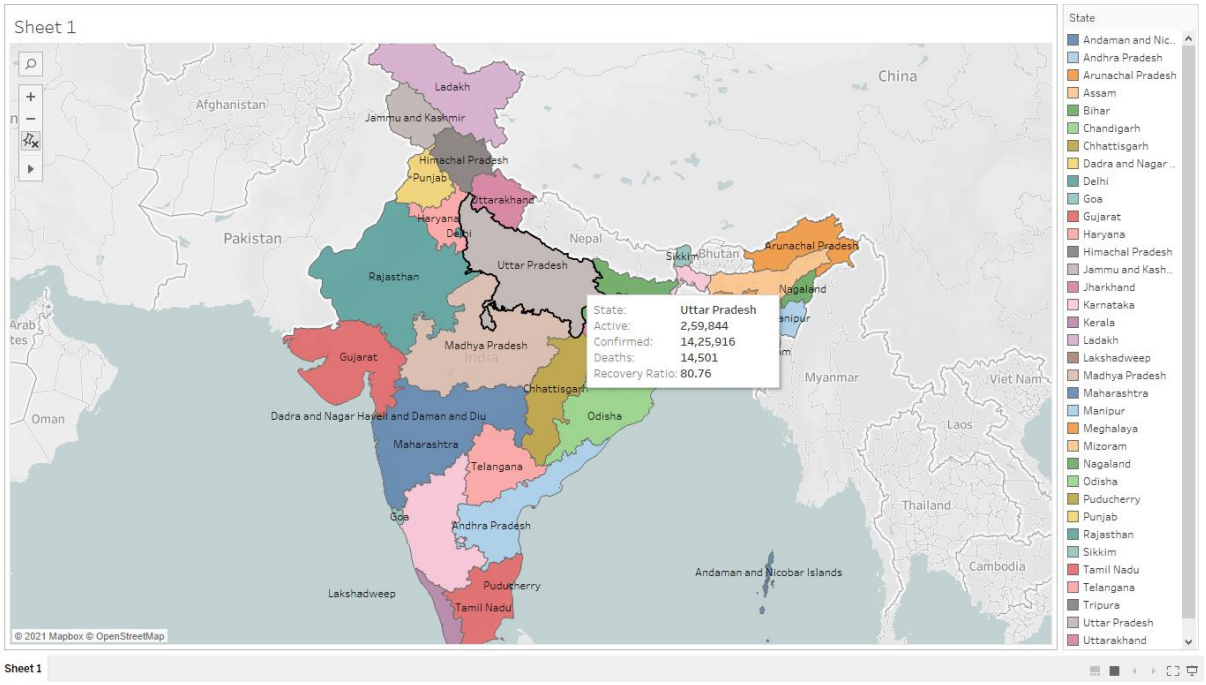
*PLOTTING A LINE PLOT OF DATA

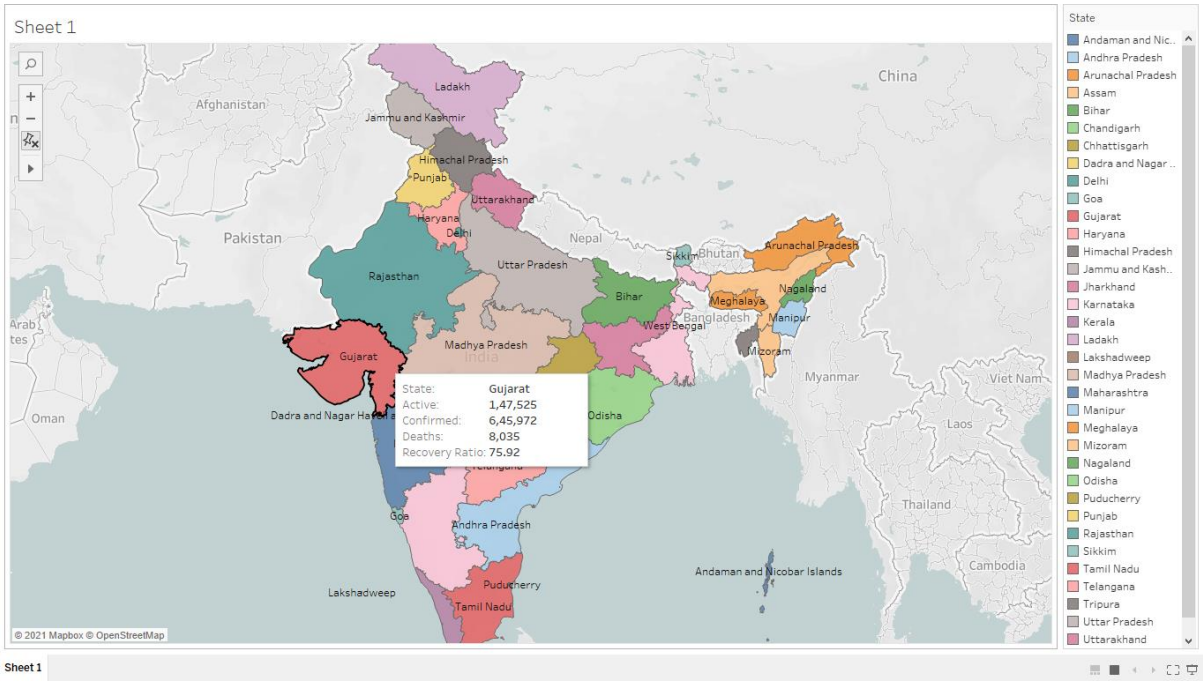



Home Page - Select or create a n
PROJECT 2 - Jupyter Notebook
Coronavirus Outbreak in Mahara
localhost:8888/notebooks/PROJECT%202.ipynb
jupyter PROJECT 2 Last Checkpoint: 06/02/2021 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
ARRANGING MONTHS ON BASIS OF CONFIRMED CASES
In [18]:
h=data1.groupby('Month')['Daily Confirmed'].sum().sort_values(ascending=False).to_frame().reset_index()
h.style.background_gradient(cmap='Blues')
Out[18]:
Month Daily Confirmed
0 September 2622323
1 August 1990885
2 October 1873041
3 November 1279861
4 July 1111273
5 December 823056
6 June 395146
7 May 155783
8 April 33231
9 March 1632
10 February 2
11 January 1
In [ ]:
k=data2.groupby('Month')['Daily Confirmed'].sum().sort_values(ascending=False).to_frame()
k.style.background_gradient(cmap='Blues')
Type here to search
3:52 PM
6/10/2021


Home Page - Select or create a n
PROJECT 2 - Jupyter Notebook
Coronavirus Outbreak in Mahara
localhost:8888/notebooks/PROJECT%202.ipynb
jupyter PROJECT 2 Last Checkpoint: 06/02/2021 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [19]:
k=data2.groupby('Month')['Daily Confirmed'].sum().sort_values(ascending=False).to_frame()
k.style.background_gradient(cmap='Blues')
Out[19]:
Daily Confirmed
Month
May 9016687
April 6936345
March 1108656
June 1008244
January 472317
February 353427
In [ ]:
px.scatter(data1,x='Date',y='Daily Confirmed',color='Daily Confirmed')
In [ ]:
In [ ]:
data1
In [ ]:
In [ ]:
a=data1['Daily Confirmed']
b=data1['Daily Recovered']
Type here to search
3:52 PM
6/10/2021

*VISUALIZING THE STATE WISE DATA IN MAP FORMAT THROUGH TABLEAU




Sheet 1
China
Afghanistan
Pakistan
Nepal
Bhutan
Sikkim
Bangladesh
Myanmar
Viet Nam
Laos
Thailand
Cambodia
Oman
Ladakh
Jammu and Kashmir
Himachal Pradesh
Punjab
Uttarakhand
Haryana
Delhi
Rajasthan
Uttar Pradesh
Bihar
Arunachal Pradesh
Nagaland
Meghalaya
Manipur
Mizoram
West Bengal
Madhya Pradesh
Gujarat
Odisha
Goa
Andhra Pradesh
Puducherry
Tamil Nadu
Lakshadweep
Andaman and Nicobar Islands
State: Gujarat
Active: 1,47,525
Confirmed: 6,45,972
Deaths: 8,035
Recovery Ratio: 75.92
© 2021 Mapbox © OpenStreetMap
State
Andaman and Nic..
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chandigarh
Chhattisgarh
Dadra and Nagar ..
Delhi
Goa
Gujarat
Haryana
Himachal Pradesh
Jammu and Kash..
Jharkhand
Karnataka
Kerala
Ladakh
Lakshadweep
Madhya Pradesh
Maharashtra
Manipur
Meghalaya
Mizoram
Nagaland
Odisha
Puducherry
Punjab
Rajasthan
Sikkim
Tamil Nadu
Telangana
Tripura
Uttar Pradesh
Uttarakhand
Sheet 1

CONCLUSIONS

Through this project, the analysis on COVID-19 data has been performed successfully. The analysis on this pandemic spread has been done and compared between different Indian states. The analysis of confirmed cases, active cases, recovered cases and deaths are done separately to give a clear look on how the virus is spreading, which states are getting affected mostly and how different states are recovering. A separate analysis on time series cases of INDIA has also been done.

Also it is here to remind you that the pandemic is still on, and you need to keep yourself and others around you safe from the coronavirus, by wearing a mask. “Wear a mask. Save lives. Wear a face cover. Wash your hands. Keep a safe distance.”

There should be proper restrictions imposed in certain districts or lockdown where still cases have been increasing.

The Indian Government should encourage more and more people to get vaccinated.

References

• https://pythonprogramming.net

• https://pypi.org/project/pandas/

• https://matplotlib.org

•https://www.google.co.in/amp/s/www.geeksforgeeks.org/numpy-in-python-set1introduction/amp/

*https://api.covid19india.org/