Name: MohammedSaif Shaikh

15 FEB 2023

MID-TERM PROJECT

OSEMN PROCESS FOR WORKING OVER COVID DATASET ACQUIRED FROM INDIAN HEALTH MINISTRY

Problem statement: - The problem of rise in covid cases in India, has an impact on the increase in the large number of deaths, which affects the entire nation, so a good starting point would be to compare the death rates of each state.

OSEMN PROCESS: -

Obtain: -

This data is actually collected from the Indian ministry of health and welfare but I have downloaded it from Kaggle. These data contain information about dates, states, times, confirmed cases, deaths, and the number of people being cured along with the vaccination given. These data are used to monitor the overall covid situation of the entire nation. It shows the number of confirmed cases and deaths in different cities through which we can get an idea about which regions are highly prone and needs more attention.

Scrub: -

Before processing these data, we need to scrub this data. It was a challenging part for me as the datasets were large (over 2 GB) to be able to load in the system, along with that it was unstructured such as discrepancies as well as split across multiple files. So firstly, the actually obtained data was in CSV format and converted into JSON using the script. For this, I need to consider smaller segments then I converted

both CSV files into JSON. After conversion to JSON, I edited both datasets in a standardized format (JSON) and for this, I used Notepad++ as the files were unable to get opened in the visual studio due to memory issues. After that, I converted JSON back to CSV using the script and successfully loaded the datasets. I dropped some of the columns from both of the datasets as they were of no use for analysis.

```
In [1]:
import pandas as pd
import numpy
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
from plotly.subplots import make_subplots
from datetime import datetime

In [2]:
covid_df = pd.read_csv("C:/1st Term/Data Science/Mid term project/covid_19_india.csv")

In [4]:
covid_df.head(20)
```

Out[4]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cured	Deaths	Confirmed
0	1	8/11/2021	6:00 PM	Kerala	1	0	0	0	1
1	2	8/11/2021	6:00 PM	Kerala	1	0	0	0	1
2	3	8/11/2021	6:00 PM	Kerala	2	0	0	0	2
3	4	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
4	5	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
5	6	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
6	7	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
7	8	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
8	9	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
9	10	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
10	11	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
11	12	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
12	13	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
13	14	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
14	15	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
15	16	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
16	17	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
17	18	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
18	19	8/11/2021	6:00 PM	Kerala	3	0	0	0	3
19	20	8/11/2021	6:00 PM	Kerala	3	0	0	0	3

After dropping some of the columns which were having nulls values and are of no use for analysis.

```
In [8]:
covid_df.drop(["Sno","Time","ConfirmedIndianNational","ConfirmedForeignNational"], inplace=True, axis=1)

In [12]:
covid_df.tail(15)
```

Out[12]:

	Date	State/UnionTerritory	Cured	Deaths	Confirmed
18095	2/13/2020	Manipur	96776	1664	105424
18096	2/12/2020	Meghalaya	64157	1185	69769
18097	2/11/2020	Mizoram	33722	171	46320
18098	2/10/2020	Nagaland	26852	585	28811
18099	2/9/2020	Odisha	972710	6565	988997
18100	2/8/2020	Puducherry	119115	1800	121766
18101	2/7/2020	Punjab	582791	16322	599573
18102	2/6/2020	Rajasthan	944700	8954	953851
18103	2/5/2020	Sikkim	25095	356	28018
18104	2/4/2020	Tamil Nadu	2524400	34367	2579130
18105	2/3/2020	Telangana	638410	3831	650353
18106	2/2/2020	Tripura	77811	773	80660
18107	2/1/2020	Uttarakhand	334650	7368	342462
18108	1/31/2020	Uttar Pradesh	1685492	22775	1708812
18109	1/30/2020	West Bengal	1506532	18252	1534999

```
In [9]: vaccine_df=pd.read_csv("C:/1st Term/Data Science/Mid term project/covid_vaccine_statewise.csv")

In [10]: vaccine_df.head(15)
```

Out[10]:

	Updated On	State	Total Doses Administered	Sessions	Sites	First Dose Administered	Second Dose Administered	Male (Doses Administered)	Female (Doses Administered)	Transgender (Doses Administered)	...	18-44 Years (Doses Administered)	45-6 Admin
0	16/01/2021	India	48276.0	3455.0	2957.0	48276.0	0.0	NaN	NaN	NaN	...	NaN	
1	17/01/2021	India	58604.0	8532.0	4954.0	58604.0	0.0	NaN	NaN	NaN	...	NaN	
2	18/01/2021	India	99449.0	13611.0	6583.0	99449.0	0.0	NaN	NaN	NaN	...	NaN	
3	19/01/2021	India	195525.0	17855.0	7951.0	195525.0	0.0	NaN	NaN	NaN	...	NaN	
4	20/01/2021	India	251280.0	25472.0	10504.0	251280.0	0.0	NaN	NaN	NaN	...	NaN	
5	21/01/2021	India	365965.0	32226.0	12600.0	365965.0	0.0	NaN	NaN	NaN	...	NaN	
6	22/01/2021	India	549381.0	36988.0	14115.0	549381.0	0.0	NaN	NaN	NaN	...	NaN	
7	23/01/2021	India	759008.0	43076.0	15605.0	759008.0	0.0	NaN	NaN	NaN	...	NaN	
8	24/01/2021	India	835058.0	49851.0	18111.0	835058.0	0.0	NaN	NaN	NaN	...	NaN	
9	25/01/2021	India	1277104.0	55151.0	19682.0	1277104.0	0.0	NaN	NaN	NaN	...	NaN	
10	26/01/2021	India	1293784.0	60821.0	21467.0	1293784.0	0.0	NaN	NaN	NaN	...	NaN	
11	27/01/2021	India	1726490.0	69495.0	23737.0	1726490.0	0.0	NaN	NaN	NaN	...	NaN	
12	28/01/2021	India	2295491.0	78523.0	25610.0	2295491.0	0.0	NaN	NaN	NaN	...	NaN	
13	29/01/2021	India	2814803.0	83664.0	26219.0	2814803.0	0.0	NaN	NaN	NaN	...	NaN	
14	30/01/2021	India	3067736.0	87822.0	26643.0	3067736.0	0.0	NaN	NaN	NaN	...	NaN	

15 rows × 24 columns

After dropping the columns where “State” = India

```
In [75]: vaccine.head(100)
```

Out[75]:

	Date	State	Total Doses Administered	Sessions	Sites	First Dose Administered	Second Dose Administered	Male (Doses Administered)	Female (Doses Administered)	Transgender (Doses Administered)	...	18-44 Years (Doses Administered)	4 Adm
212	16/01/2021	Andaman and Nicobar Islands	23.0	2.0	2.0	23.0	0.0	12.0	11.0	0.0	...	NaN	
213	17/01/2021	Andaman and Nicobar Islands	23.0	2.0	2.0	23.0	0.0	12.0	11.0	0.0	...	NaN	
214	18/01/2021	Andaman and Nicobar Islands	42.0	9.0	2.0	42.0	0.0	29.0	13.0	0.0	...	NaN	
215	19/01/2021	Andaman and Nicobar Islands	89.0	12.0	2.0	89.0	0.0	53.0	36.0	0.0	...	NaN	
216	20/01/2021	Andaman and Nicobar Islands	124.0	16.0	3.0	124.0	0.0	67.0	57.0	0.0	...	NaN	
...	...	...	...	...	...	...	...	...	...	...	...	...	
307	21/04/2021	Andaman and Nicobar Islands	89072.0	4100.0	40.0	80901.0	8171.0	43818.0	37076.0	7.0	...	NaN	
308	22/04/2021	Andaman and Nicobar Islands	91833.0	4200.0	42.0	83233.0	8600.0	45145.0	38081.0	7.0	...	NaN	
309	23/04/2021	Andaman and Nicobar Islands	94166.0	4200.0	42.0	85201.0	8965.0	46215.0	38978.0	8.0	...	NaN	
310	24/04/2021	Andaman and Nicobar Islands	96093.0	3900.0	37.0	86796.0	9297.0	47109.0	39679.0	8.0	...	NaN	
311	25/04/2021	Andaman and Nicobar Islands	97392.0	3500.0	35.0	87865.0	9527.0	47711.0	40146.0	8.0	...	NaN	

100 rows × 24 columns

Explore: -

In this stage, we will explore this data. After successfully loading the datasets, I performed some analysis on the dataset. From the dataset, by using confirmed cases, cured, and death cases columns we can find active cases, recovery, and mortality rates.

Model: -

In this step, we need to reduce the dimensionality of our data and must select only those data from which we can easily predict the results. In this we use regression and prediction to predict future values, classification to identify, and clustering to group values. In clustering, we would group those data that are having same characteristics in common. For instance, we can create a cluster of the most active case states into one cluster so that we can have an idea of immediate action that can be taken in that area first to control the cases just like that we can group out the states with the most deaths. We can create a bar graph of each state with their active cases, deaths, and confirmed cases which can help us to make a prediction such as how much vaccination supply is needed.

Interpreting: -

This is the final stage as well as the most problematic stage as in this stage we need to present our analysis and prediction to a non-technical person in such a way that they can get an idea about our findings and this presentation should answer all the questions which the stakeholder has. Here we can use graphical figures mostly through which we can make the other entity understand our analysis easily.

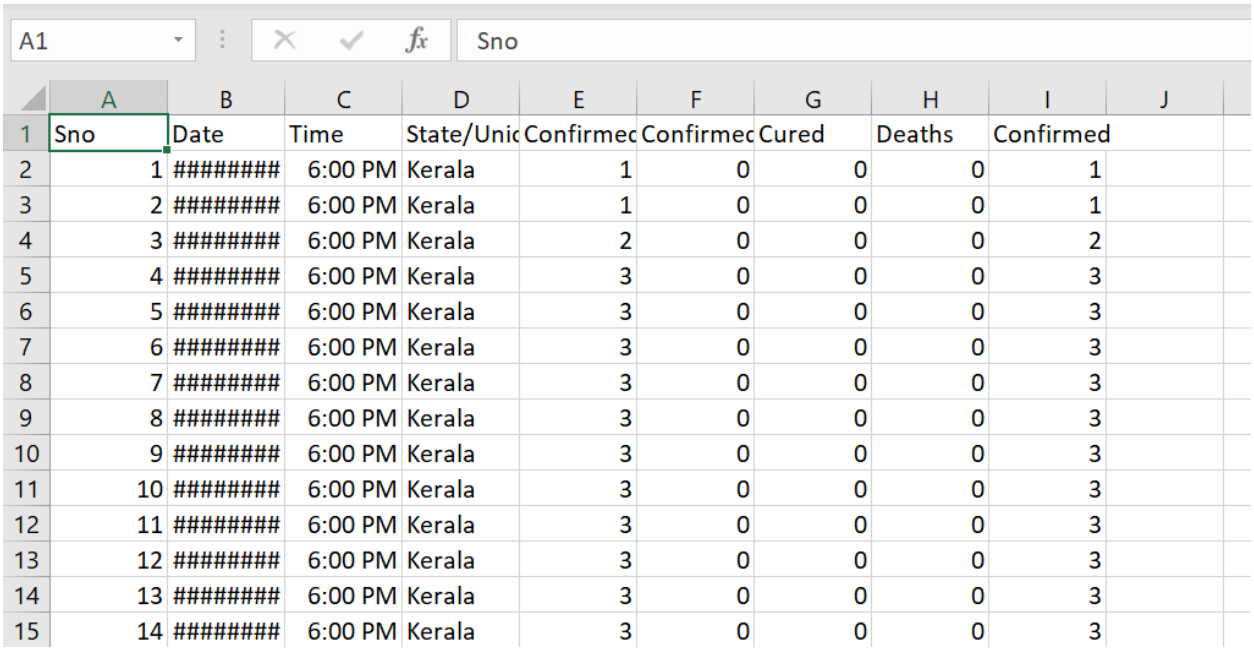
Overview of Dataset: -

The dataset is being taken from Kaggle. There are two datasets namely covid 19 (over 2 GB) and vaccination (over 2 GB). However, there are some columns that are of no use and have replicas, so I had to drop those columns just to make them clean and ready for analysis. The dataset is a challenging one as the data is not in a

standardized format, contains punctuation and is split across multiple files, and is a 4–5-point dataset.

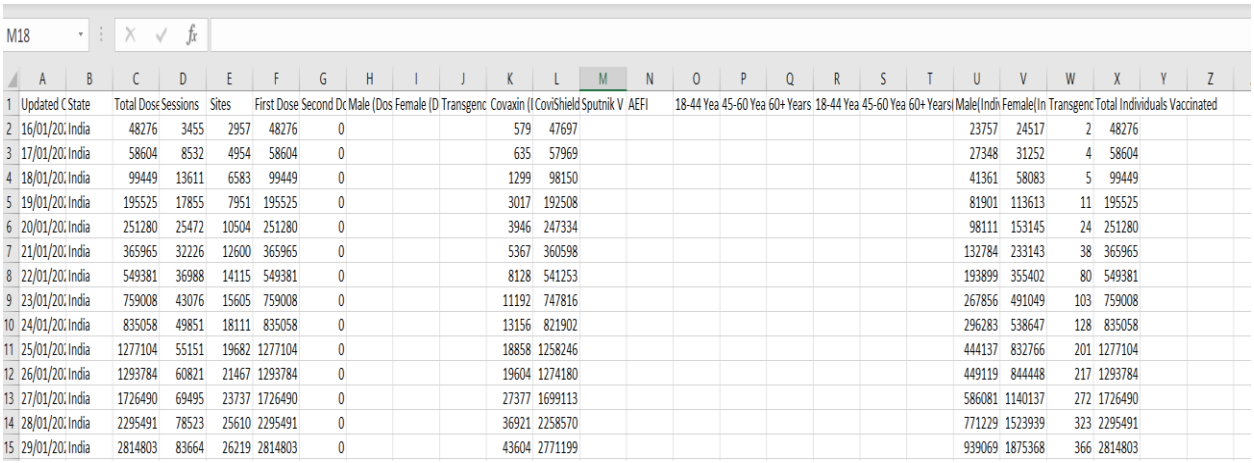
Initial Data: -

Covid 19



Sno	Date	Time	State/Unic	Confirmec	Confirmec	Cured	Deaths	Confirmed
1	########	6:00 PM	Kerala	1	0	0	0	1
2	########	6:00 PM	Kerala	1	0	0	0	1
3	########	6:00 PM	Kerala	2	0	0	0	2
4	########	6:00 PM	Kerala	3	0	0	0	3
5	########	6:00 PM	Kerala	3	0	0	0	3
6	########	6:00 PM	Kerala	3	0	0	0	3
7	########	6:00 PM	Kerala	3	0	0	0	3
8	########	6:00 PM	Kerala	3	0	0	0	3
9	########	6:00 PM	Kerala	3	0	0	0	3
10	########	6:00 PM	Kerala	3	0	0	0	3
11	########	6:00 PM	Kerala	3	0	0	0	3
12	########	6:00 PM	Kerala	3	0	0	0	3
13	########	6:00 PM	Kerala	3	0	0	0	3
14	########	6:00 PM	Kerala	3	0	0	0	3
15	########	6:00 PM	Kerala	3	0	0	0	3

Vaccination



Updated C	State	Total Dose	Sessions	Sites	First Dose	Second Dc	Male (Dos	Female (D	Transgenc	Covaxin (I	CoviShield	Sputnik V	AEFI	18-44 Yea	45-60 Yea	60+ Years	18-44 Yea	45-60 Yea	60+ Years	Male(Indi	Female(In	Transgenc	Total Individuals Vaccinated
16/01/20	India	48276	3455	2957	48276	0				579	47697									23757	24517	2	48276
17/01/20	India	58604	8532	4954	58604	0				635	57969									27348	31252	4	58604
18/01/20	India	99449	13611	6583	99449	0				1299	98150									41361	58083	5	99449
19/01/20	India	195525	17855	7951	195525	0				3017	192508									81901	113613	11	195525
20/01/20	India	251280	25472	10504	251280	0				3946	247334									98111	153145	24	251280
21/01/20	India	365965	32226	12600	365965	0				5367	360598									132784	233143	38	365965
22/01/20	India	549381	36988	14115	549381	0				8128	541253									193899	355402	80	549381
23/01/20	India	759008	43076	15605	759008	0				11192	747816									267856	491049	103	759008
24/01/20	India	835058	49851	18111	835058	0				13156	821902									296283	538647	128	835058
25/01/20	India	1277104	55151	19682	1277104	0				18858	1258246									444137	832766	201	1277104
26/01/20	India	1293784	60821	21467	1293784	0				19604	1274180									449119	844448	217	1293784
27/01/20	India	1726490	69495	23737	1726490	0				27377	1699113									586081	1140137	272	1726490
28/01/20	India	2295491	78523	25610	2295491	0				36921	2258570									771229	1523939	323	2295491
29/01/20	India	2814803	83664	26219	2814803	0				43604	2771199									939069	1875368	366	2814803

Initial CSV to JSON: -

Covid


```
[
	{
		"Sno":1,
		"Date":"8\/11\/2021",
		"Time":"6:00PM",
		"State\/UnionTerritory":"Kerala",
		"ConfirmedIndianNational":"1",
		"ConfirmedForeignNational":"0",
		"Cured":0,
		"Deaths":0,
		"Confirmed":1
	}
]
```

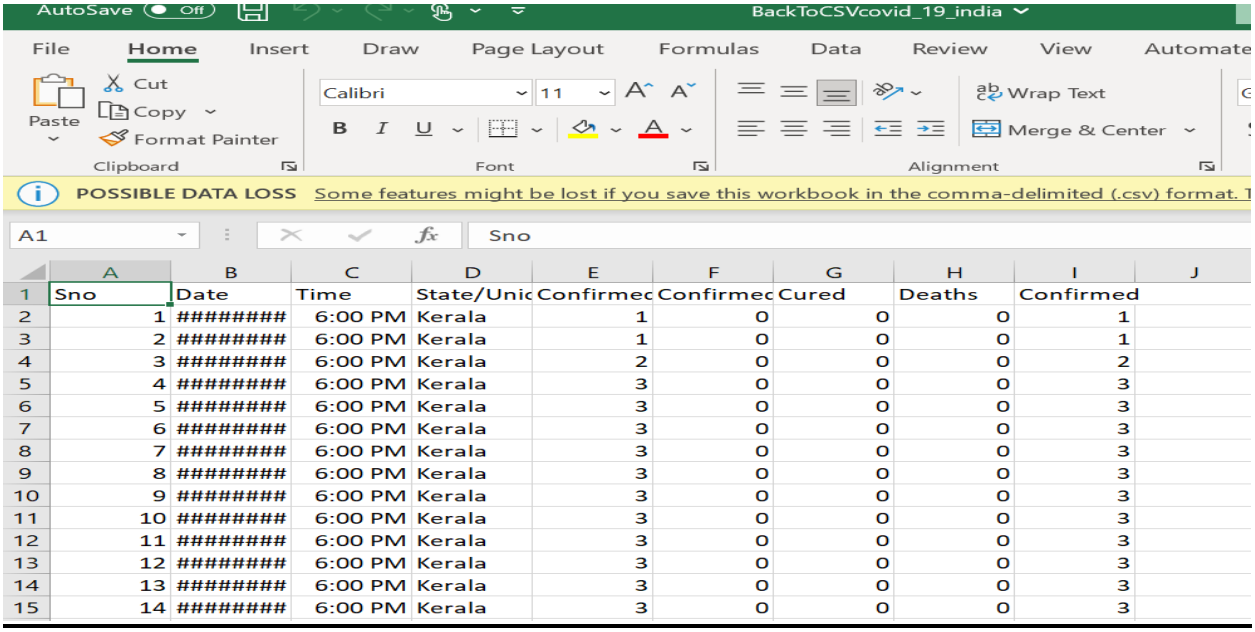

Vaccination


```
[
	{
		"Updated On":"16\/01\/2021",
		"State":"India",
		"Total Doses Administered":48276.0,
		"Sessions":3455.0,
		" Sites ":2957.0,
		"First Dose Administered":48276.0,
		"Second Dose Administered":0.0,
		"Male (Doses Administered)":null,
		"Female (Doses Administered)":null,
```

```
"Transgender (Doses Administered)":null,
" Covaxin (Doses Administered)":579.0,
"CoviShield (Doses Administered)":47697.0,
"Sputnik V (Doses Administered)":null,
"AEFI":null,
"18-44 Years (Doses Administered)":null,
"45-60 Years (Doses Administered)":null,
"60+ Years (Doses Administered)":null,
"18-44 Years(Individuals Vaccinated)":null,
"45-60 Years(Individuals Vaccinated)":null,
"60+ Years(Individuals Vaccinated)":null,
"Male(Individuals Vaccinated)":23757.0,
"Female(Individuals Vaccinated)":24517.0,
"Transgender(Individuals Vaccinated)":2.0,
"Total Individuals Vaccinated":48276.0
}
]
```

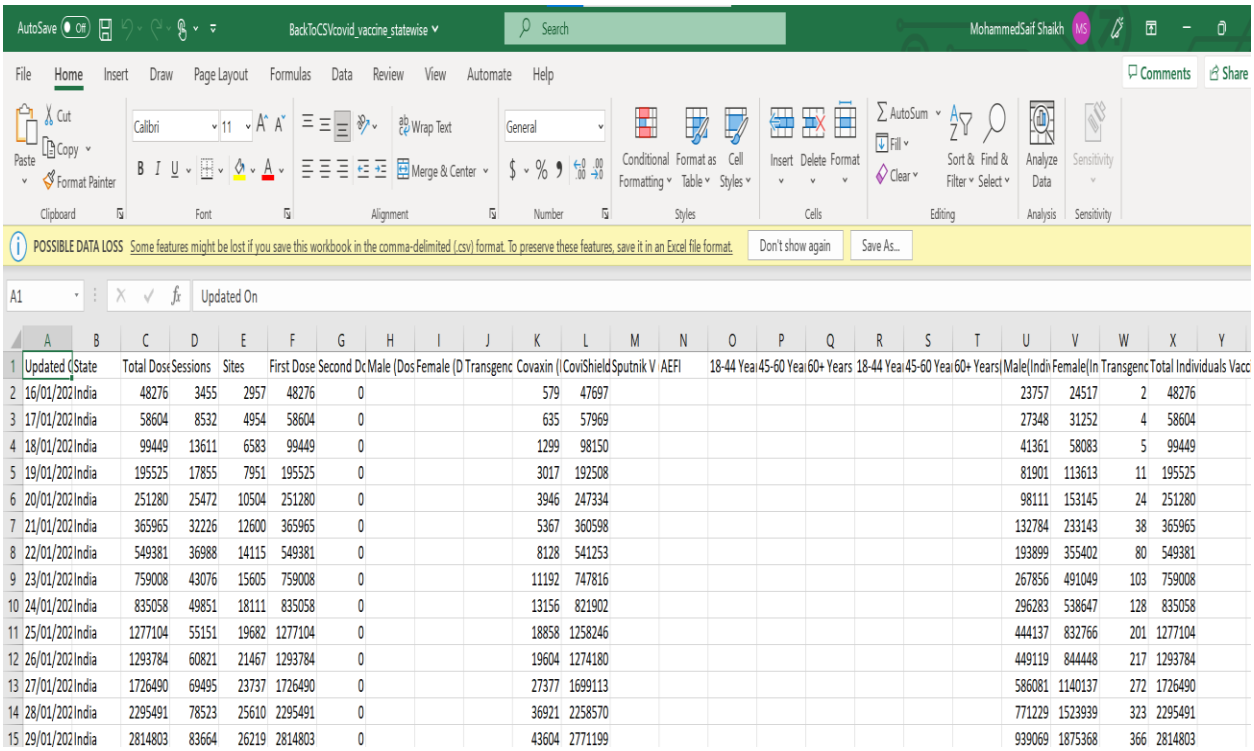
JSON to CSV: -

Covid



Sno	Date	Time	State/Unic	Confirmec	Confirmec	Cured	Deaths	Confirmed
1	#########	6:00 PM	Kerala	1	0	0	0	1
2	#########	6:00 PM	Kerala	1	0	0	0	1
3	#########	6:00 PM	Kerala	2	0	0	0	2
4	#########	6:00 PM	Kerala	3	0	0	0	3
5	#########	6:00 PM	Kerala	3	0	0	0	3
6	#########	6:00 PM	Kerala	3	0	0	0	3
7	#########	6:00 PM	Kerala	3	0	0	0	3
8	#########	6:00 PM	Kerala	3	0	0	0	3
9	#########	6:00 PM	Kerala	3	0	0	0	3
10	#########	6:00 PM	Kerala	3	0	0	0	3
11	#########	6:00 PM	Kerala	3	0	0	0	3
12	#########	6:00 PM	Kerala	3	0	0	0	3
13	#########	6:00 PM	Kerala	3	0	0	0	3
14	#########	6:00 PM	Kerala	3	0	0	0	3
15	#########	6:00 PM	Kerala	3	0	0	0	3

Vaccination



Updated C	State	Total Dose	Sessions	Sites	First Dose	Second Dc	Male (Dos	Female (D	Transgenc	Covaxin (I	CoviShield	Sputnik V	AEFI	18-44 Yea	45-60 Yea	60+ Years	18-44 Yea	45-60 Yea	60+ Years	Male(Indi	Female(In	Transgenc	Total Individuals Vacc
16/01/202	India	48276	3455	2957	48276	0				579	47697									23757	24517	2	48276
17/01/202	India	58604	8532	4954	58604	0				635	57969									27348	31252	4	58604
18/01/202	India	99449	13611	6583	99449	0				1299	98150									41361	58083	5	99449
19/01/202	India	195525	17855	7951	195525	0				3017	192508									81901	113613	11	195525
20/01/202	India	251280	25472	10504	251280	0				3946	247334									98111	153145	24	251280
21/01/202	India	365965	32226	12600	365965	0				5367	360598									132784	233143	38	365965
22/01/202	India	549381	36988	14115	549381	0				8128	541253									193899	355402	80	549381
23/01/202	India	759008	43076	15605	759008	0				11192	747816									267856	491049	103	759008
24/01/202	India	835058	49851	18111	835058	0				13156	821902									296283	538647	128	835058
25/01/202	India	1277104	55151	19682	1277104	0				18858	1258246									444137	832766	201	1277104
26/01/202	India	1293784	60821	21467	1293784	0				19604	1274180									449119	844448	217	1293784
27/01/202	India	1726490	69495	23737	1726490	0				27377	1699113									586081	1140137	272	1726490
28/01/202	India	2295491	78523	25610	2295491	0				36921	2258570									771229	1523939	323	2295491
29/01/202	India	2814803	83664	26219	2814803	0				43604	2771199									939069	1875368	366	2814803

CSV to JSON Script: -

func_data_toJSON.py 1

C: > 1st Term > Data Science > Mid term project > func_data_toJSON.py > ...

```python
import pandas as pd

#Function to convert all CSV files to JSON
def convert_toJSON(filename):
    print('READING CSV FILE')
    csv_df = pd.read_csv(filename)
    print('CONVERTING BACK TO JSON')
    csv_df.to_json('BackToJSON'+filename[:-4]+'.json', orient="records")

#calling convert_toJSON() to convert all created CSV files back to JSON
convert_toJSON('covid_vaccine_statewise.csv')
convert_toJSON('covid_19_india.csv')
print('All FILES DONE')

```

JSON to CSV Script: -

JSON_to_CSV.py 1

C: > 1st Term > Data Science > Mid term project > JSON_to_CSV.py > ...

```python
import pandas as pd


def convert_toCSV(filename):
    print('READING JSON FILE')
    json_df = pd.read_json(filename)
    print('CONVERTING  BACK TO CSV FILE')
    json_df.to_csv('BackToCSV'+filename[:-5]+'.csv', index=False)



convert_toCSV('covid_vaccine_statewise.json')
convert_toCSV('covid_19_india.json')

```

Data Analysis: -

1. **Does Vaccination and our Medical systems have a positive impact on the common people around the country?**

   To answer this question, we can find the Recovery and Mortality Rate for each state and then we can compare them both to find if it is having a positive impact or negative.
   Firstly, I created a new function called “statewise” and for analysis, I took 3 columns namely “Confirmed, Deaths, and Cured” from covid 19 dataset. To find the Recovery Rate, the formula would be:

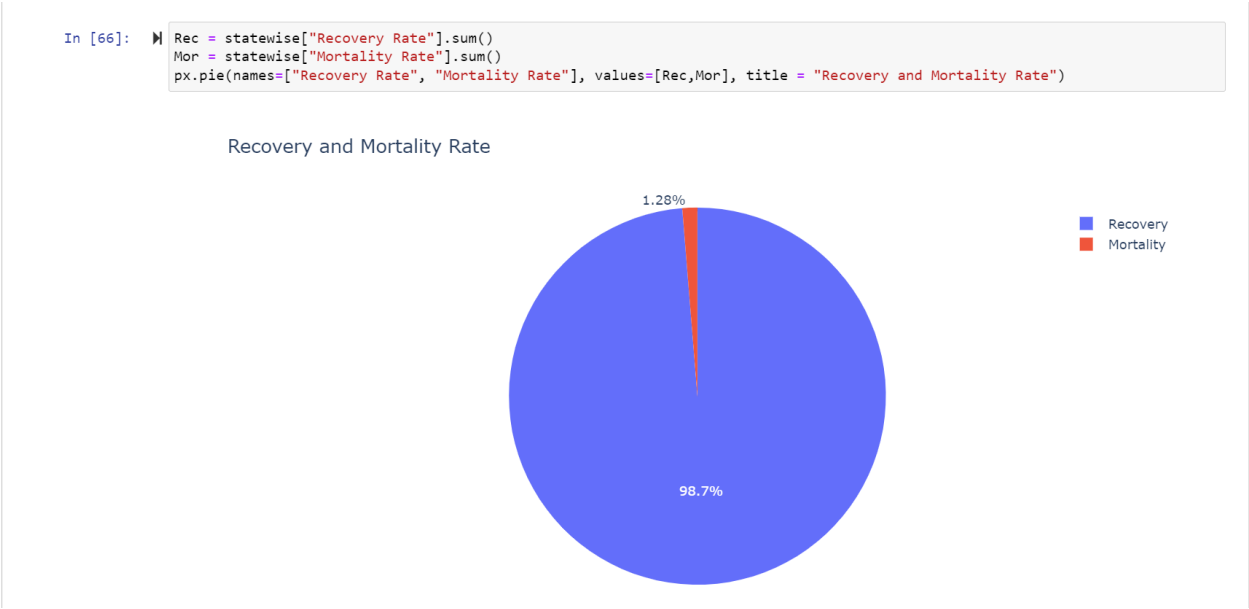
```
In [15]: statewise['Recovery Rate'] = statewise["Cured"]*100/statewise["Confirmed"]
```

For Mortality,

```
In [16]: statewise['Mortality Rate'] = statewise["Deaths"]*100/statewise["Confirmed"]
```

Final Result:

```
In [66]: Rec = statewise["Recovery Rate"].sum()
         Mor = statewise["Mortality Rate"].sum()
         px.pie(names=["Recovery Rate", "Mortality Rate"], values=[Rec,Mor], title = "Recovery and Mortality Rate")
```



From the above analysis, we can predict that the recovery rate is higher which signifies that our vaccination as well as our medical system are having a positive impact on the community.

I also created a pivot table and sort it by “States”. From which we can have a better understanding.

```
In [20]: statewise = statewise.sort_values(by = "Confirmed", ascending = False)

In [22]: statewise.style.background_gradient(cmap = "cubehelix")
```

Out[22]:

State/UnionTerritory	Confirmed	Cured	Deaths	Recovery Rate	Mortality Rate
Maharashtra	6363442	6159676	134201	96.797865	2.108937
Maharashtra***	6229596	6000911	130753	96.329056	2.098900
Kerala	3586693	3396184	18004	94.688450	0.501967
Karnataka	2921049	2861499	36848	97.961349	1.261465
Karanataka	2885238	2821491	36197	97.790581	1.254559
Tamil Nadu	2579130	2524400	34367	97.877967	1.332504
Andhra Pradesh	1985182	1952736	13564	98.365591	0.683262
Uttar Pradesh	1708812	1685492	22775	98.635309	1.332797
West Bengal	1534999	1506532	18252	98.145471	1.189056
Delhi	1436852	1411280	25068	98.220276	1.744647
Chhattisgarh	1003356	988189	13544	98.488373	1.349870
Odisha	988997	972710	6565	98.353180	0.663804
Rajasthan	953851	944700	8954	99.040626	0.938721
Gujarat	825085	814802	10077	98.753704	1.221329
Madhya Pradesh	791980	781330	10514	98.655269	1.327559
Madhya Pradesh***	791656	780735	10506	98.620487	1.327092
Haryana	770114	759790	9652	98.659419	1.253321
Bihar	725279	715352	9646	98.631285	1.329971
Bihar****	715730	701234	9452	97.974655	1.320610
Telangana	650353	638410	3831	98.163613	0.589065
Punjab	599573	582791	16322	97.201008	2.722271

Along with the vaccine dataset, we can also compare which gender has a high percentage of vaccination. For that, I Consider two columns from another Dataset which is “vaccination” and created a pie chart.

```
In [61]: # Male vs Female Vaccination

male = vaccination["Male(Individuals Vaccinated)"].sum()
female = vaccination["Female(Individuals Vaccinated)"].sum()
px.pie(names=["Male", "Female"], values=[male,female], title = "Male and Female Vaccination")
```

From this, we can decide on which particular group we need to focus more on and actively convince them to get vaccinated for the safety of the nation.

2. **How many currently covid active Cases are there in our Country? And which states are highly prone and be given more attention?**

Here, we need to find the active cases first. For that, I used covid dataset and took 3 columns namely " Confirmed, Deaths, and Cured" for analysis. Active cases can be calculated as follows: -

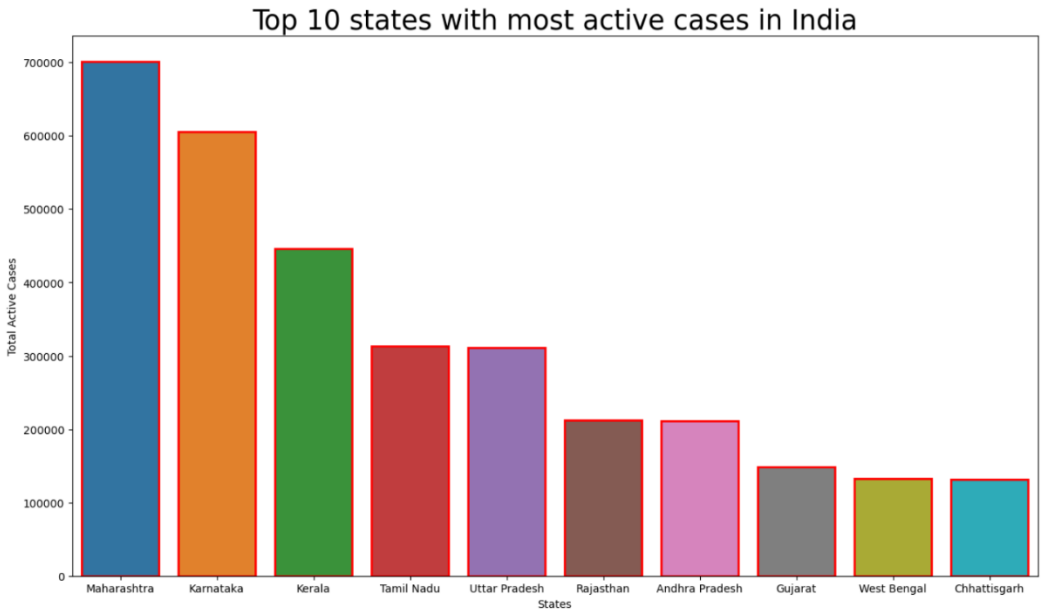
```
In [13]:
# Finding Active Cases
covid_df['Active cases'] = covid_df['Confirmed'] - (covid_df['Cured'] + covid_df['Deaths'])
covid_df.tail()
```

Out[13]:

	Date	State/UnionTerritory	Cured	Deaths	Confirmed	Active cases
18105	2/3/2020	Telangana	638410	3831	650353	8112
18106	2/2/2020	Tripura	77811	773	80660	2076
18107	2/1/2020	Uttarakhand	334650	7368	342462	444
18108	1/31/2020	Uttar Pradesh	1685492	22775	1708812	545
18109	1/30/2020	West Bengal	1506532	18252	1534999	10215

Now to find the states which are having the highest number of active cases, I group them by "States" and created a bar chart which is shown below.

```
In [32]:
# Top 10 active case states

top_10_active_cases = covid_df.groupby(by= 'State/UnionTerritory').max()[['Active cases', 'Date']].sort_values(by = ['Active
fig = plt.figure(figsize=(16,9))
plt.title("Top 10 states with most active cases in India", size = 25)
ax = sns.barplot(data = top_10_active_cases.iloc[:10], y='Active cases', x = 'State/UnionTerritory', linewidth = 2, edgecolor

plt.xlabel("States")
plt.ylabel("Total Active Cases")
plt.show()
```



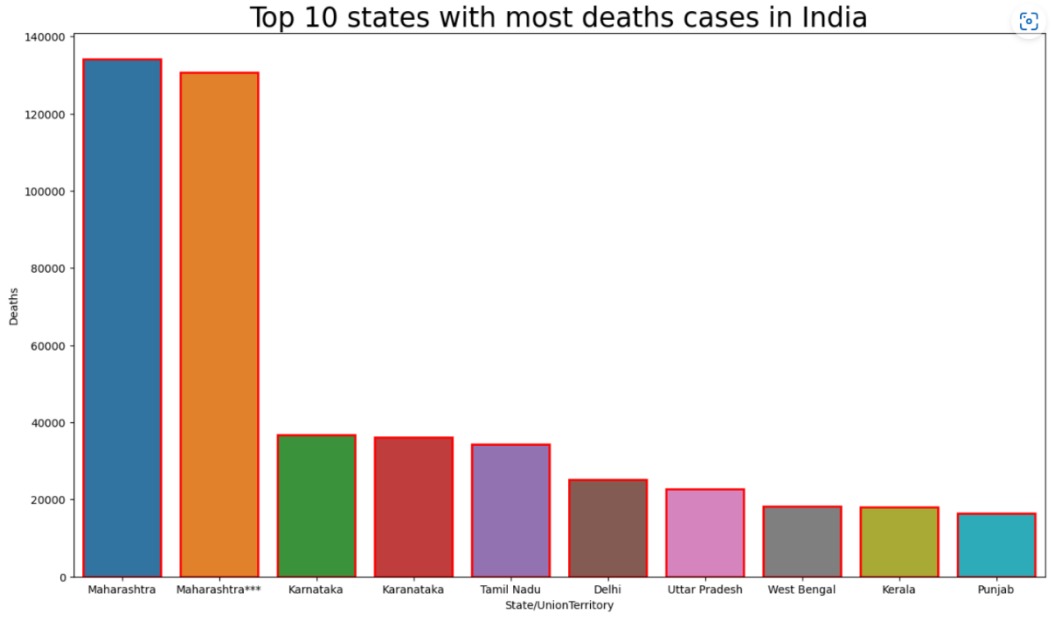
From this, we can analyze that Maharashtra has the highest number of cases which means that it is mandatory for the government to take immediate necessary steps for this state and required continuous monitoring. While West

Bengal and Chhattisgarh can be given less attention as they are having least number of active cases.

In the same way, we can find the top 10 states with the most deaths.

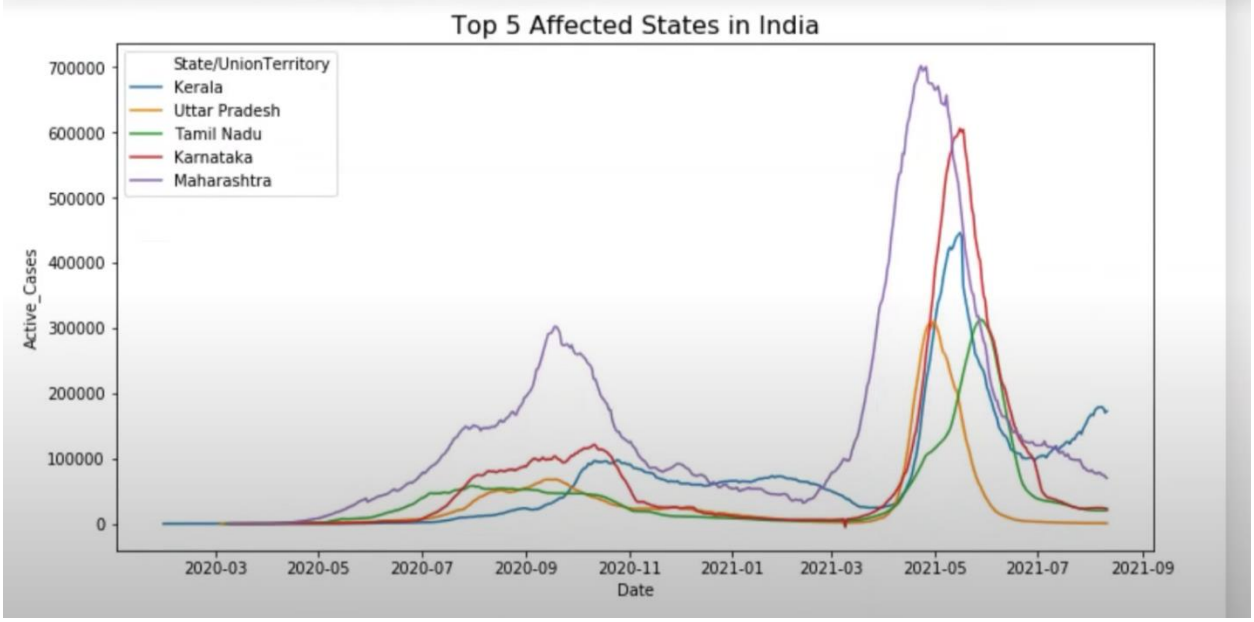
States

In [33]:
```
# Top states with highest deaths

top_10_deaths = covid_df.groupby(by = 'State/UnionTerritory').max()[['Deaths','Date']].sort_values(by =['Deaths'], ascending
fig = plt.figure(figsize=(16,9))
plt.title("Top 10 states with most deaths cases in India", size = 25)
ax = sns.barplot(data = top_10_deaths.iloc[:10], y='Deaths', x = 'State/UnionTerritory', linewidth = 2, edgecolor = 'red')
```



3. **Which States were highly affected during the given period of time?**
For finding this, I consider the two columns namely “Active cases, Date” and group them by “States” where I consider the 5 states and performed an analysis on them, and created a line graph for data visualization.

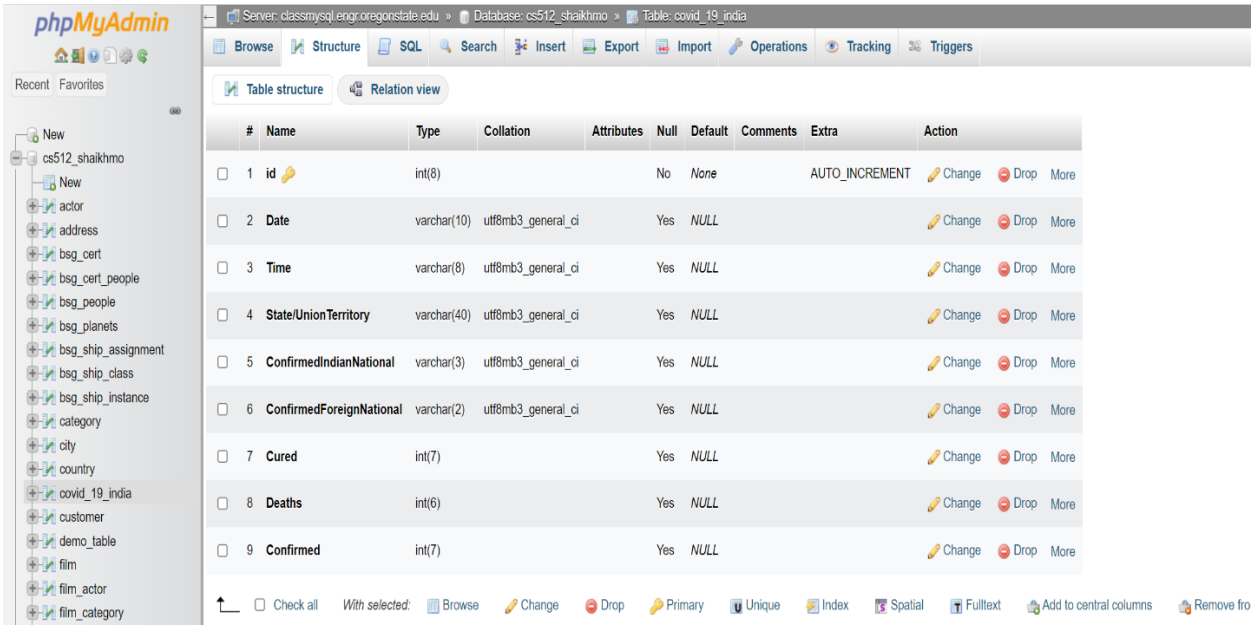
In [78]:
```
# Growth Trends

fig = plt.figure(figsize=(12,6))

ax = sns.lineplot(data = covid_df[covid_df['State/UnionTerritory'].isin(['Maharastra','Karnataka','Kerela','Tamil Nadu','Utta
                                  x ='Date', y ='Active cases'])
ax.set_title("Top 5 Affected states in India", size = 16)
```



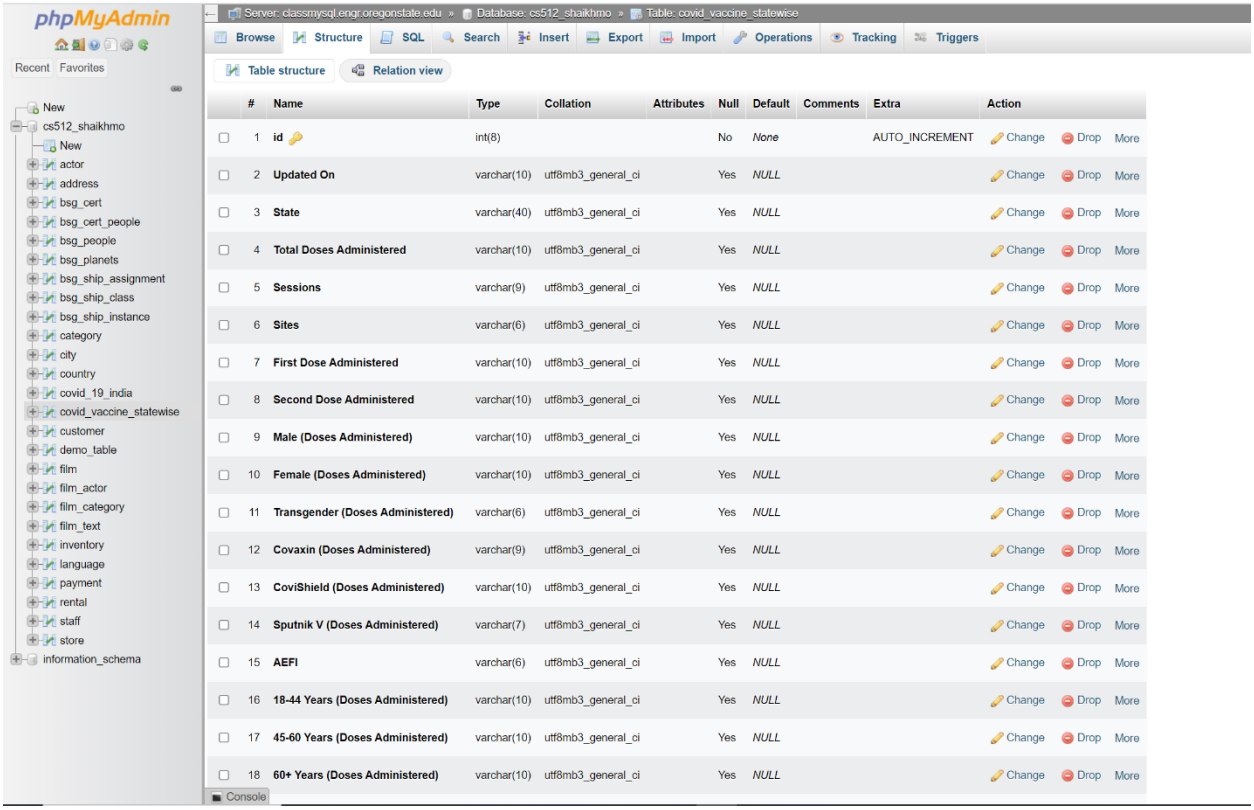
We can predict from the above-performed analysis about the Growth trends. From the line graph, we can see that there were almost 300000 active cases in "Maharashtra" being the state with the highest number of cases at the end of the year 2020 then was having a sudden decrease in the number of cases and a falls down to 50000 in the starting of 2021 and had a sharp increase in cases which almost cross the 700000 mark. From this, we can be directed to the question that what made the number go down during the first two months of 2021 and what factors made it to rose in such a high number suddenly. This analysis will help the professionals in the medical area to shorten their research and can concentrate on finding factors affecting the growth of active cases in a particular given state.

TABLE SCHEMA: -

```
CREATE TABLE covid_19_india (
        Id INT NOT NULL AUTO_INCREMENT,
        Date VARCHAR (10) NULL,
        Time VARCHAR (8) NULL,
        State/Union Territory VARCHAR (40) NULL,
        ConfirmedIndianNational VARCHAR (10) NULL,
        ConfirmedForeignNational VARCHAR (10) NULL,
        Cured INT (7) NULL,
        Deaths INT (6) NULL,
        Confirmed INT (7) NULL,
        PRIMARY KEY(id);
);
```



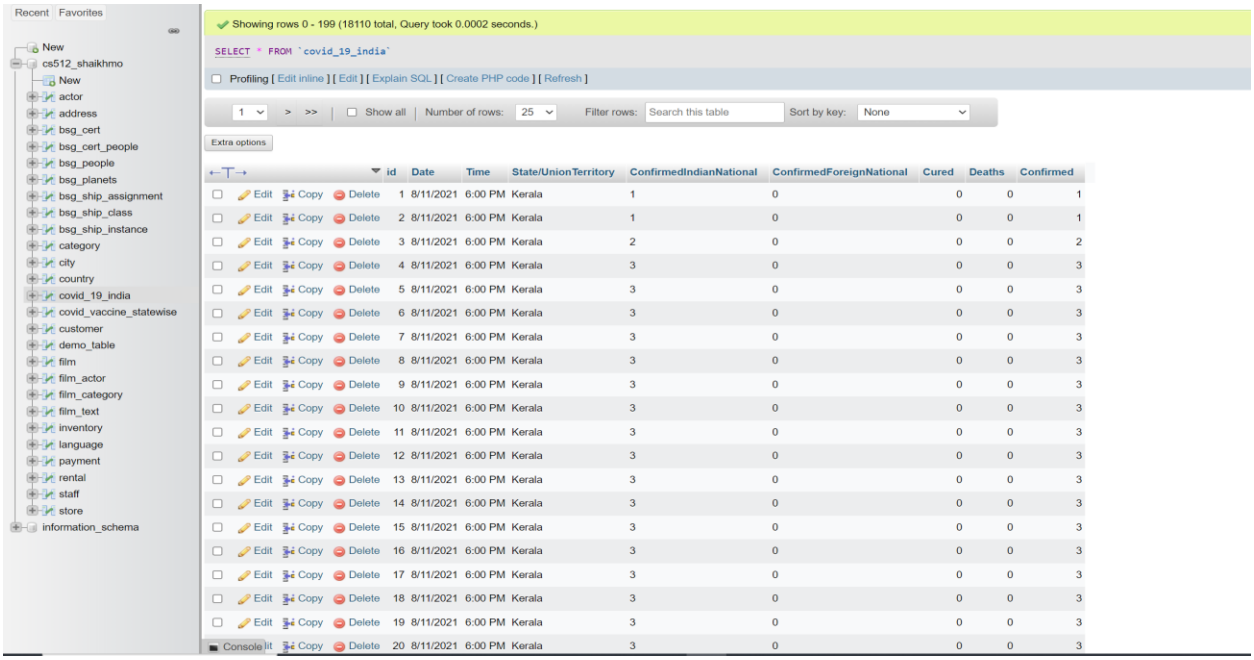
```
CREATE TABLE covid_vaccine_statewise (
        Id INT (8) NOT NULL AUTO_INCREMENT,
        Updated On VARCHAR (10) NULL,
        State VARCHAR (40) NULL,
        Total Doses Administered VARCHAR (10) NULL,
        Sessions VARCHAR (9) NULL,
        Sites VARCHAR (6) NULL,
        First Dose Administered VARCHAR (10) NULL,
        Second Dose Administered VARCHAR (10) NULL,
        Male (Doses Administered) VARCHAR (10) NULL,
        Female (Doses Administered) VARCHAR (10) NULL,
        Transgender (Doses Administered) VARCHAR (6) NULL,
        Covaxin (Doses Administered) VARCHAR (9) NULL,
        CoviShield (Doses Administered) VARCHAR (10) NULL,
        Sputnik V (Doses Administered) VARCHAR (7) NULL,
        AEFI VARCHAR (6) NULL,
        18-44 Years (Doses Administered) VARCHAR (10) NULL,
        45-60 Years (Doses Administered) VARCHAR (10) NULL,
        60+ Years (Doses Administered) VARCHAR (10) NULL,
        18-44 Years (Individuals Vaccinated) VARCHAR (9) NULL,
        45-60 Years (Individuals Vaccinated) VARCHAR (9) NULL,
        60+ Years (Individuals Vaccinated VARCHAR (9) NULL,
        Male (Individuals Vaccinated) VARCHAR (10) NULL,
        Female (Individuals Vaccinated) VARCHAR (10) NULL,
        Transgender (Individuals Vaccinated) VARCHAR (6) NULL,
        Total Individuals Vaccinated VARCHAR (10) NULL,
         PRIMARY KEY (id)
);
```

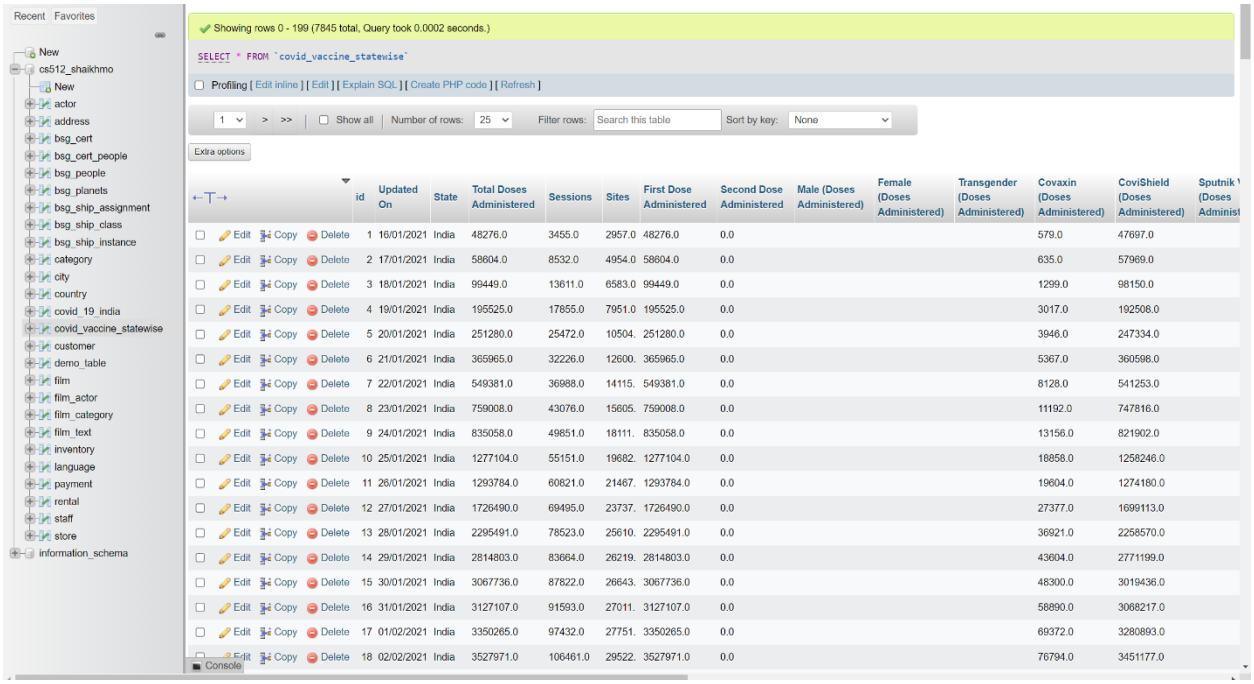


Delimiters used: “;”

Columns separated with “,”

Columns enclosed and escaped with “ ” ”





Process used for loading the data into the database: -

I followed the 2nd procedure from the link provided by a professor on how to load data in ED Discussion where I simply imported my CSV file directly into the database which I used for the SQL queries assignment. After that, for both datasets, I inserted a new row "id" and set it as the primary key. I also drop "Sno" from covid dataset as it was of no use and that way I can use only one primary key.

Statement of Originality: -

I haven't collaborated with anyone; however, I am adding the statement of originality because I have used this dataset before during my undergraduate period for "Data Mining" Classes where I just performed some basics operation on this and haven't done any prior deep analysis as well as I was confident work on this dataset. The Above analysis is solely performed by me for this mid-term project without any collaboration.