

Post Instruction : Dataset used in this project - '<https://www.kaggle.com/datasets/khanmohammadanas/district-wise-crimes-in-india>'

## STATE/UT wise Crimes in India -

The main Theme about this project is to concentrate on crimes in India between 2001-2012. I picked this dataset from Kaggle. The techniques used for this project Matplotlib,Seaborn,Pandas,Numpy and Basics of python learnt in this course.

This project is part of my Data Analysis with Python: Zero to Pandas course - [www.zerotopandas.com](http://www.zerotopandas.com).

## Downloading the Dataset

I found this data set from Kaggle and according to jovian i have loaded via jovian opendatasets.

Link : <https://www.kaggle.com/datasets/khanmohammadanas/district-wise-crimes-in-india>

We need to install Jovain opendaasets as its not previously installed as shown below:

```
!pip install jovian opendatasets --upgrade --quiet
```

```
|██████████████████████████████████████████████████████████████████████████| 68 kB 5.7 MB/s eta  
0:00:01  
Building wheel for uuid (setup.py) ... done
```

Here we are using pandas, numpy, matplotlib, seaborn and few other libraries. we need to load/import them.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

As we previously installed `opendatasets`, loading the dataset

```
import opendatasets as od
dataset_url = 'https://www.kaggle.com/datasets/khanmohammadanas/district-wise-crimes-in-i
od.download('https://www.kaggle.com/datasets/khanmohammadanas/district-wise-crimes-in-i
```

Please provide your Kaggle credentials to download this dataset. Learn more:

<http://bit.ly/kaggle-creds>

Your Kaggle username: dspnavya

Your Kaggle Key: .....

```
Downloading district-wise-crimes-in-india.zip to ./district-wise-crimes-in-india
```

100% ██████████ | 387k/387k [00:00<00:00, 4.62MB/s]

# Data Preparation and Cleaning

Using pandas we are retrieving the csv file to the data frame given as project\_df.

```
project_df = pd.read_csv('/content/district-wise-crimes-in-india/01_District_wise_crime
```

showing the dataframe as project\_df

```
project_df
```

	STATE/UT	DISTRICT	YEAR	MURDER	ATTEMPT TO MURDER	CULPABLE HOMICIDE NOT AMOUNTING TO MURDER	RAPE	CUSTODIAL RAPE	OTHER RAPE	KIDN/ ABDI
0	ANDHRA PRADESH	ADILABAD	2001	101	60	17	50	0	50	
1	ANDHRA PRADESH	ANANTAPUR	2001	151	125	1	23	0	23	
2	ANDHRA PRADESH	CHITTOOR	2001	101	57	2	27	0	27	
3	ANDHRA PRADESH	CUDDAPAH	2001	80	53	1	20	0	20	
4	ANDHRA PRADESH	EAST GODAVARI	2001	82	67	1	23	0	23	
...	...	...	...	...	...	...	...	...	...	...
9012	LAKSHADWEEP	LAKSHADWEEP	2012	0	0	0	0	0	0	
9013	LAKSHADWEEP	TOTAL	2012	0	0	0	0	0	0	
9014	PUDUCHERRY	KARAIKAL	2012	5	6	2	6	0	6	
9015	PUDUCHERRY	PUDUCHERRY	2012	24	21	10	7	0	7	
9016	PUDUCHERRY	TOTAL	2012	29	27	12	13	0	13	

9017 rows × 33 columns

Here we clean the dataset

- 1. Explore the number of rows & columns, ranges of values etc.
- 2. Handle missing, incorrect and invalid data.
- 3. We drop few columns which are not necessary.

```
#here we drop few columns
project_df = project_df.drop(columns=['DISTRICT', 'CULPABLE HOMICIDE NOT AMOUNTING TO MU
    'OTHER RAPE', 'KIDNAPPING & ABDUCTION', 'KIDNAPPING AND ABDUCTION OF OTHERS',
    'PREPARATION AND ASSEMBLY FOR DACOITY', 'ROBBERY', 'BURGLARY', 'THEFT',
    'AUTO THEFT', 'OTHER THEFT', 'RIOTS', 'CRIMINAL BREACH OF TRUST', 'COUNTERFIETIN
    'DOWRY DEATHS', 'ASSAULT ON WOMEN WITH INTENT TO OUTRAGE HER MODESTY',
```

```
'INSULT TO MODESTY OF WOMEN', 'CRUELTY BY HUSBAND OR HIS RELATIVES',
'OTHER IPC CRIMES', 'TOTAL IPC CRIMES']])
```

```
project_df
```

	STATE/UT	YEAR	MURDER	ATTEMPT TO MURDER	RAPE	KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS	DACOITY	CHEATING	IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES	C/ DE NEGLI
0	ANDHRA PRADESH	2001	101	60	50	30	9	104	0	
1	ANDHRA PRADESH	2001	151	125	23	30	8	65	0	
2	ANDHRA PRADESH	2001	101	57	27	34	4	209	0	
3	ANDHRA PRADESH	2001	80	53	20	20	1	37	0	
4	ANDHRA PRADESH	2001	82	67	23	26	4	220	0	
...	...	...	...	...	...	...	...	...	...	...
9012	LAKSHADWEEP	2012	0	0	0	0	0	0	0	
9013	LAKSHADWEEP	2012	0	0	0	0	0	0	0	
9014	PUDUCHERRY	2012	5	6	6	2	0	15	0	
9015	PUDUCHERRY	2012	24	21	7	14	5	75	0	
9016	PUDUCHERRY	2012	29	27	13	16	5	90	0	

9017 rows × 10 columns

```
project_df.columns
```

```
Index(['STATE/UT', 'YEAR', 'MURDER', 'ATTEMPT TO MURDER', 'RAPE',
      'KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS', 'DACOITY', 'CHEATING',
      'IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES',
      'CAUSING DEATH BY NEGLIGENCE'],
      dtype='object')
```

Using .info() we get datatype of values and get null values.

```
project_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 9017 entries, 0 to 9016
```

```
Data columns (total 10 columns):
```

#	Column	Non-Null Count	Dtype
---	-----	-----	-----
0	STATE/UT	9017 non-null	object
1	YEAR	9017 non-null	int64
2	MURDER	9017 non-null	int64

3	ATTEMPT TO MURDER	9017 non-null	int64
4	RAPE	9017 non-null	int64
5	KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS	9017 non-null	int64
6	DACOITY	9017 non-null	int64
7	CHEATING	9017 non-null	int64
8	IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES	9017 non-null	int64
9	CAUSING DEATH BY NEGLIGENCE	9017 non-null	int64

dtypes: int64(9), object(1)

memory usage: 704.6+ KB

We use .isna and add them to know if there any na values in our dataframe.

```
v = project_df.isna().sum()
v
```

STATE/UT	0
YEAR	0
MURDER	0
ATTEMPT TO MURDER	0
RAPE	0
KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS	0
DACOITY	0
CHEATING	0
IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES	0
CAUSING DEATH BY NEGLIGENCE	0

dtype: int64

We use .describe to get the maximum,minimum and other statistical values in our dataframe.

```
project_df.describe()
```

	YEAR	MURDER	ATTEMPT TO MURDER	RAPE	KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS	DACOITY	CHEATING	IMPORTA OF G F FOR COUNT
count	9017.000000	9017.00000	9017.000000	9017.000000	9017.000000	9017.000000	9017.000000	9017.000000
mean	2006.629034	89.28535	78.069646	53.041366	58.535433	13.177110	170.255517	0.190000
std	3.463623	327.27503	303.446020	190.741450	246.937464	56.778798	743.536365	2.410000
min	2001.000000	0.00000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	2004.000000	18.00000	10.000000	8.000000	6.000000	1.000000	13.000000	0.000000
50%	2007.000000	38.00000	28.000000	20.000000	18.000000	3.000000	37.000000	0.000000
75%	2010.000000	66.00000	56.000000	41.000000	42.000000	9.000000	104.000000	0.000000
max	2012.000000	7601.00000	7964.000000	3425.000000	7910.000000	1319.000000	19646.000000	83.000000

## Exploratory Analysis and Visualization

We have created new sub dataframe as per analysis from project\_df.

###Q. Compare Murder cases vs Attempt to Murder in Andhra Pradesh using line plot?

```
andhra_pradesh = project_df[project_df['STATE/UT']=="ANDHRA PRADESH"]
```

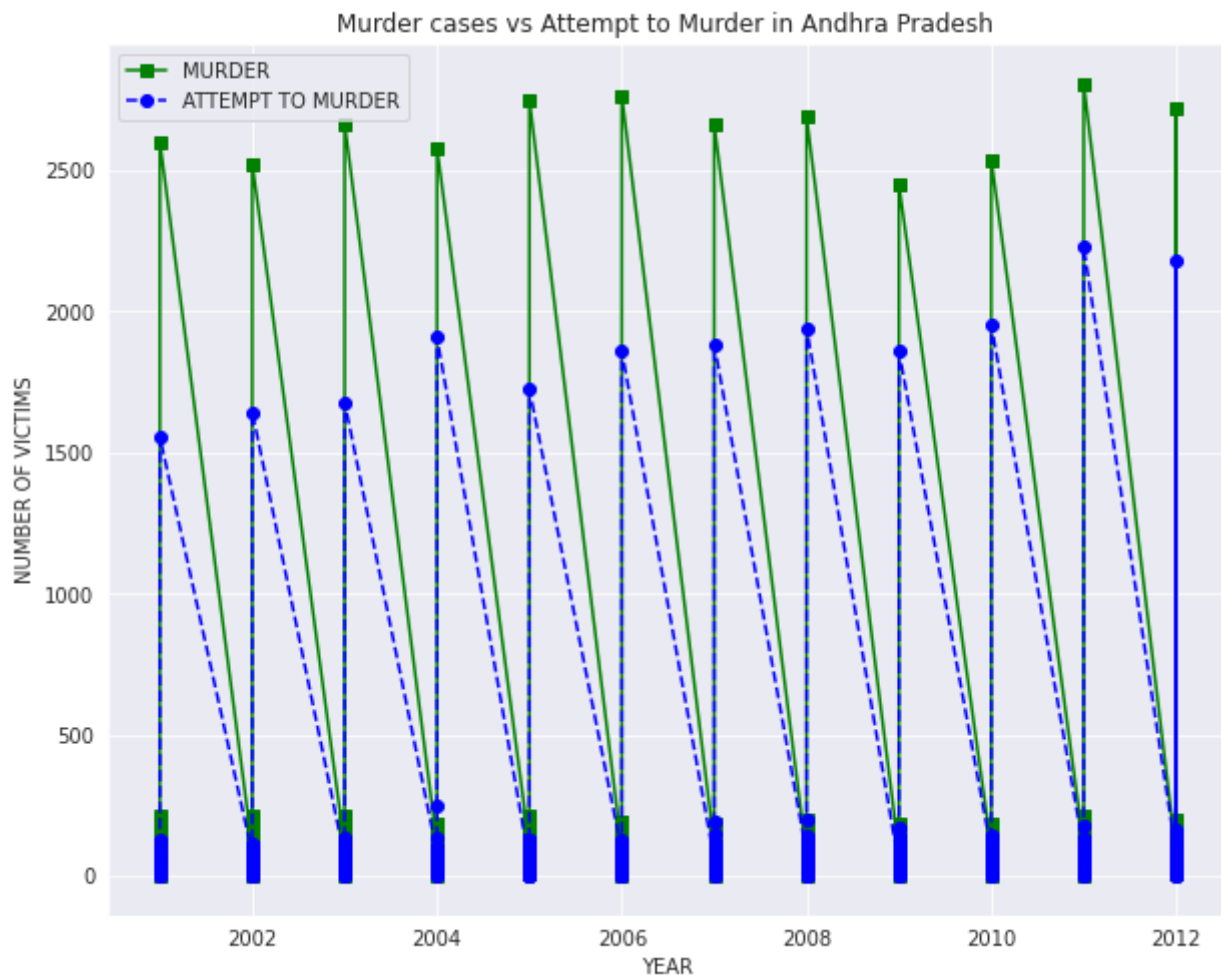
```
andhra_pradesh
```

	STATE/UT	YEAR	MURDER	ATTEMPT TO MURDER	RAPE	KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS	DACOITY	CHEATING	IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES	CAUSING DEATH BY NEGLIGENCE
0	ANDHRA PRADESH	2001	101	60	50	30	9	104	0	18
1	ANDHRA PRADESH	2001	151	125	23	30	8	65	0	27
2	ANDHRA PRADESH	2001	101	57	27	34	4	209	0	40
3	ANDHRA PRADESH	2001	80	53	20	20	1	37	0	23
4	ANDHRA PRADESH	2001	82	67	23	26	4	220	0	43
...	...	...	...	...	...	...	...	...	...	...
8235	ANDHRA PRADESH	2012	44	36	44	20	2	118	0	28
8236	ANDHRA PRADESH	2012	75	47	56	58	3	197	0	30
8237	ANDHRA PRADESH	2012	41	17	30	39	2	281	0	22
8238	ANDHRA PRADESH	2012	91	91	130	59	4	403	0	57
8239	ANDHRA PRADESH	2012	2717	2183	1341	1403	131	11002	0	1392

370 rows × 10 columns

We use line Graph to represent Number of Murders and Attempt's to Murder that took place against particular year.

```
plt.figure(figsize=(10,8))
sns.set_style("darkgrid")
plt.title('Murder cases vs Attempt to Murder in Andhra Pradesh ')
plt.xlabel('YEAR')
plt.ylabel('NUMBER OF VICTIMS')
plt.plot(andhra_pradesh['YEAR'],andhra_pradesh['MURDER'],'s-g');
plt.plot(andhra_pradesh['YEAR'],andhra_pradesh['ATTEMPT TO MURDER'],'o--b')
plt.legend(['MURDER','ATTEMPT TO MURDER']);
```



Insight from Above graph : As we go from 2002-2012 the number of victims in both the cases are increasing in Murder's and Attempts to Murder's. But when we compare between them No of Murder's are higher than attempt's in every year.

We create a new dataframe on the basis of grouping STATE/UT , all the column's (crimes) and aggregating .sum() so that all the values get's added and represent as sub dataframe with just state's and crimes.

```
new_df = project_df.groupby('STATE/UT')[['MURDER', 'ATTEMPT TO MURDER', 'RAPE',
    'KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS', 'DACOITY', 'CHEATING',
    'CAUSING DEATH BY NEGLIGENCE']].sum()
```

new\_df

	MURDER	ATTEMPT TO MURDER	RAPE	KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS	DACOITY	CHEATING	CAUSING DEATH BY NEGLIGENCE
STATE/UT							
A & N ISLANDS	320	130	218	154	30	506	84
ANDHRA PRADESH	63512	44840	26958	29872	3974	197686	276870
ARUNACHAL PRADESH	1682	834	1000	1116	520	1040	1490
ASSAM	30864	11324	32356	45840	8340	24782	60410
BIHAR	82490	81838	26248	38778	22198	61380	88884

	MURDER	ATTEMPT TO MURDER	RAPE	KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS	DACOITY	CHEATING	CAUSING DEATH BY NEGLIGENCE
STATE/UT							
CHANDIGARH	486	602	562	1068	62	4602	196
CHHATTISGARH	24120	15820	23676	5260	2936	13996	50864
D & N HAVELI	216	62	112	188	80	342	296
DAMAN & DIU	136	76	40	30	68	146	622
DELHI UT	12410	10554	12848	31300	762	52256	15006
GOA	974	538	700	328	110	2352	5328
GUJARAT	27550	12224	8498	25836	7684	34132	108770
HARYANA	20942	14868	12820	12258	2750	31150	30394
HIMACHAL PRADESH	2908	1670	3608	3082	90	5306	12618
JAMMU & KASHMIR	12608	20606	5620	17640	376	10926	9048
JHARKHAND	38120	24540	18318	10564	11812	18952	27224
KARNATAKA	39874	37024	10348	10630	5162	92670	10156
KERALA	9466	10654	14894	3796	2728	87516	978
LAKSHADWEEP	6	12	14	2	6	0	0
MADHYA PRADESH	56798	58916	72174	18486	3310	36990	129938
MAHARASHTRA	65534	41178	35972	21706	16350	159804	269226
MANIPUR	3910	7072	774	2114	96	2462	174
MEGHALAYA	3460	1202	2040	528	1596	2030	1364
MIZORAM	806	516	1652	20	90	1486	630
NAGALAND	2076	928	440	142	250	758	744
ODISHA	28906	31304	22860	16582	6762	26740	73014
PUDUCHERRY	714	686	154	242	76	1128	5204
PUNJAB	18986	20794	10918	10770	920	75872	66678
RAJASTHAN	31688	42684	31596	49342	1126	281600	156700
SIKKIM	332	222	390	124	12	700	844
TAMIL NADU	40254	53194	13904	24872	2474	63378	280386
TRIPURA	4044	1362	4116	1750	452	1704	3084
UTTAR PRADESH	130886	121040	38116	95180	9618	167920	227642
UTTARAKHAND	5896	5254	2756	4668	714	11220	13360
WEST BENGAL	42112	29386	41574	43546	5284	61662	75502

```

q = new_df[['MURDER', 'ATTEMPT TO MURDER', 'RAPE',
            'KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS', 'DACOITY', 'CHEATING',
            'CAUSING DEATH BY NEGLIGENCE']].sum()
q.sum()

```

6172868

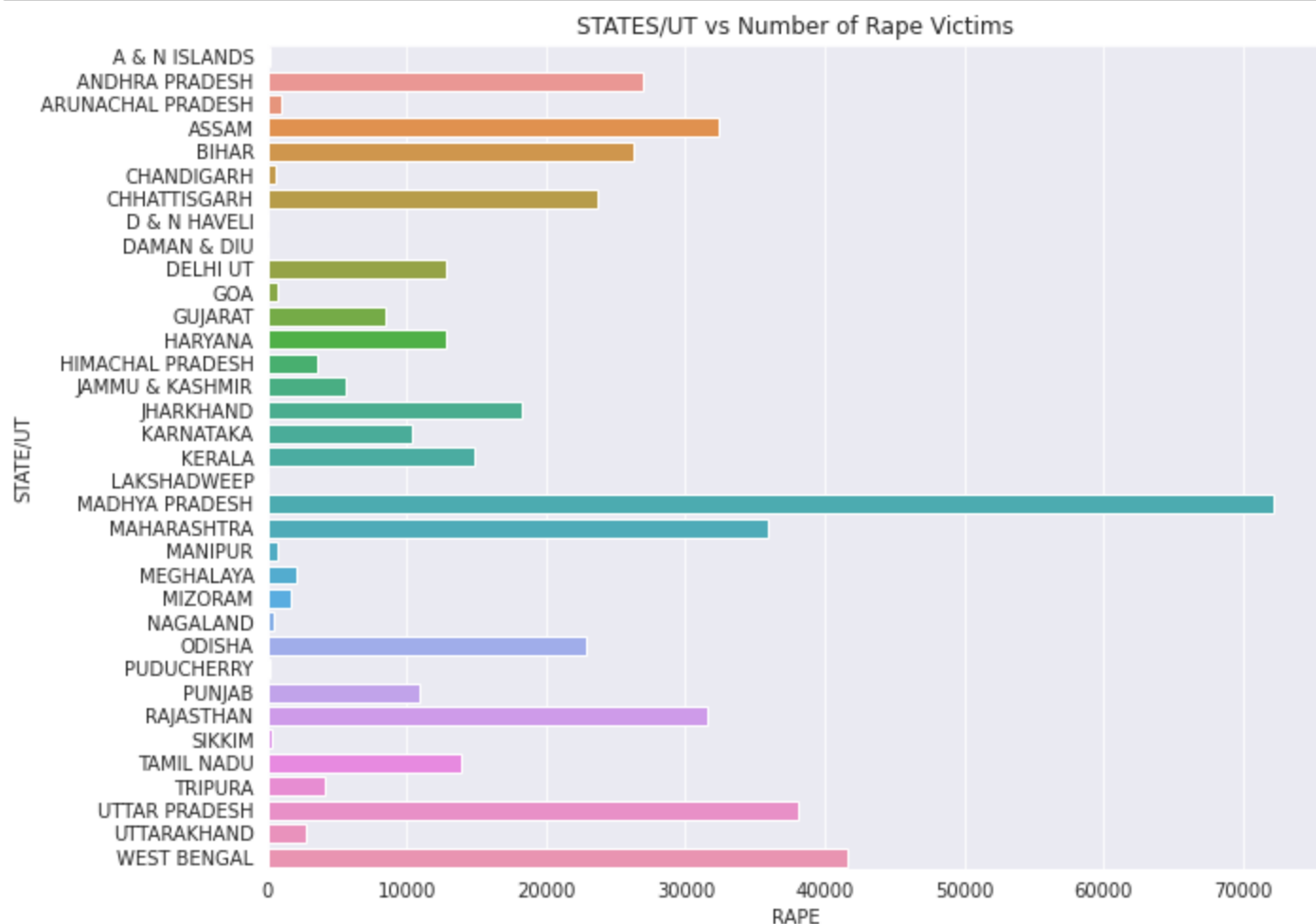
As new\_df is a sub dataframe , In this dataframe we STATE/UT as index and so we use reset\_index(inplace= True) to get an index and column 1 as STATE/UT.

```
new_df.reset_index(inplace=True)
```

###Q. Compare Rape cases in all States/UT between 2001-2012?

We use Barplot in this case to show Number of Rape's cases were taken palce so far in respective STATE/UT.

```
plt.figure(figsize=(10,8))
plt.title('STATES/UT vs Number of Rape Victims')
sns.barplot(x = new_df['RAPE'],y = new_df['STATE/UT']);
```



Insight from Above BAR Graph : Clearly, It depict's that **MADHYA PRADESH** is leading with number of rape victim's, 2nd highest **West Bengal** and so on. The rape's that had happened between 2001-2012.

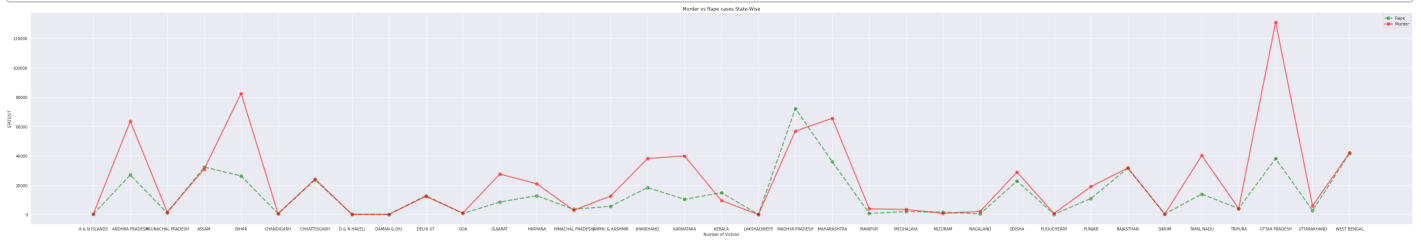
###Q. Compare murders vs rape cases between all the states?

We use LINE GRAPH in this case to contrast Number of Rape cases were taken place vs Murder so far in respective STATE/UT.

```
plt.figure(figsize=(64,10))
plt.title('Murder vs Rape cases State-Wise')
plt.xlabel('Number of Victims')
```



```
plt.ylabel('STATE/UT')
sns.lineplot(y = new_df['RAPE'],x = new_df['STATE/UT'],marker='o', color='g', ls='--',
sns.lineplot(y = new_df['MURDER'],x = new_df['STATE/UT'],marker='o', color='r', ls='-',
plt.legend(['Rape', 'Murder']));
```



Insight from Above Line Graph : When we zoomin it Clearly discloses that the number of murders are greater than number of rape victims most states ,**MADHYA PRADESH** in terms of number of rape victim's and **UTTAR PRADESH** in terms of murder and so on. The rape's and murder's that had happened between 2001-2012.

We create a new dataframe on the basis of grouping Years , all the column's (crimes) and aggregating .sum() so that all the value's get's added and represent as sub dataframe with just years,states and crimes.

```
year_wise = project_df.groupby('YEAR')[[ 'MURDER', 'ATTEMPT TO MURDER', 'RAPE',
      'KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS', 'DACOITY', 'CHEATING',
      'CAUSING DEATH BY NEGLIGENCE', 'IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES']].su
```

As year\_wise is a sub dataframe , In this dataframe YEAR was taken as index by default so we use reset\_index(inplace= True) to get an index and column 1 as YEAR.

```
year_wise.reset_index(inplace = True)
```

```
year_wise
```

	YEAR	MURDER	ATTEMPT TO MURDER	RAPE	KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS	DACOITY	CHEATING	CAUSING DEATH BY NEGLIGENCE	IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES
0	2001	72404	63046	32150	29290	12308	89454	114364	228
1	2002	70580	60760	32746	29012	12202	92542	128088	152
2	2003	65432	51884	31694	26592	10606	94956	121344	92
3	2004	67216	55780	36466	31156	10622	103878	138846	178
4	2005	65438	56062	36718	31500	10282	107250	143396	298
5	2006	64962	54460	38696	34828	9494	116152	157026	134
6	2007	64636	54802	41474	40832	9158	130652	173580	122
7	2008	65532	57196	42934	45878	9060	133158	184372	134
8	2009	64738	58076	42794	51482	9172	145436	197064	96
9	2010	66670	58842	44344	59590	8716	157998	212686	72
10	2011	68610	62770	48412	71130	8570	175312	217780	160
11	2012	68868	70276	49846	76524	8628	188406	215182	118

###As we Know analysis is important , i.e to know the trends in all crimes that were happened in recent year's. This graph would give us the clear idea of the crimes that took palce and drastically changed over the timeline from 2001-2012.

```
fig, axes = plt.subplots(2, 3, figsize=(20, 8))

#1 of 6
sns.lineplot(x=year_wise['YEAR'],y=year_wise['DACOITY'], palette="tab10",marker='o', cc
axes[0,0].set_xlabel('Year')
axes[0,0].set_ylabel('DACOITY')
axes[0,0].set_title('GRAPH 1: DACOITY vs YEAR WISE')

# 2 of 6
axes[0,1].set_title('GRAPH 2: NUMBER of RAPES PER YEAR')
axes[0,1].set_xlabel('Year')
axes[0,1].set_ylabel('Number of Rapes')
sns.kdeplot(x = year_wise['YEAR'],y = year_wise['RAPE'],shade=True,ax=axes[0,1]);

# 3 of 6
axes[0,2].set_title('GRAPH 3: Number of Girls IMPORTATION vs Year wise')
axes[0,2].set_xlabel('Year')
axes[0,2].set_ylabel('Number of Girls')
sns.distplot(year_wise['IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES'],ax=axes[0,2]);

# 4 of 6
axes[1,0].set_title('GRAPH 4: CHEATING Per YEAR')
axes[1,0].set_xlabel('Year')
axes[1,0].set_ylabel('Number of Girls')
sns.barplot(y = year_wise['CHEATING'],x = year_wise['YEAR'],ax=axes[1,0]);

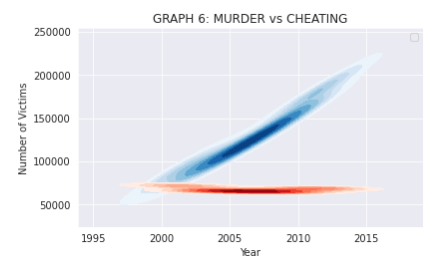
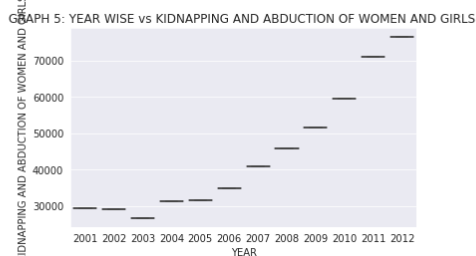
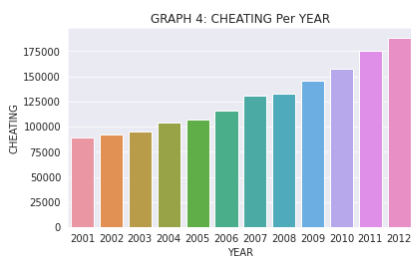
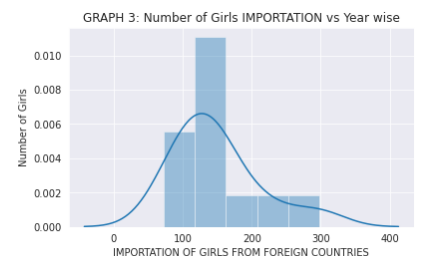
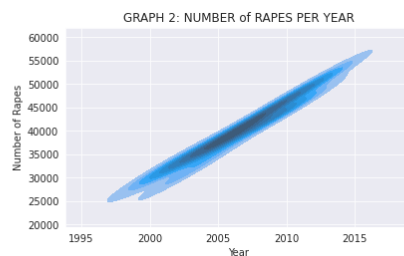
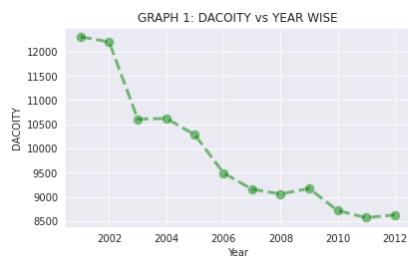
# 5 of 6
axes[1,1].set_title('GRAPH 5: YEAR WISE vs KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS
sns.boxplot(x = year_wise['YEAR'],y = year_wise['KIDNAPPING AND ABDUCTION OF WOMEN AND

# 6 of 6
axes[1,2].set_title('GRAPH 6: MURDER vs CHEATING')
axes[1,2].set_xlabel('Year')
axes[1,2].set_ylabel('Number of Victims')
axes[1,2].legend(['CHEATING', 'MURDER'])
sns.kdeplot(y = year_wise['CHEATING'],x=year_wise['YEAR'],shade=True,cmap='Blues',ax=ax
sns.kdeplot(y = year_wise['MURDER'],x=year_wise['YEAR'],shade=True,cmap='Reds',ax=axes[

plt.tight_layout(pad=2);
```

/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning:  
`distplot` is a deprecated function and will be removed in a future version. Please  
adapt your code to use either `displot` (a figure-level function with similar  
flexibility) or `histplot` (an axes-level function for histograms).

```
warnings.warn(msg, FutureWarning)
```



Insight from Graph 1: It depicts that DACOITY cases are decreasing over the years.

Insight from Graph 2: It depicts that RAPE cases are increasing over the years.

Insight from Graph 3: It depicts that IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES are decreasing over time the year.

Insight from Graph 4: It depicts that number of CHEATING's that took place are increasing over the year.

Insight from Graph 5: It depicts that KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS are increasing over the year.

Insight from Graph 6: It depicts that Murder are increasing over the year whereas cheating on comparatively decreasing over the year.

###Q. Compare deaths CAUSED BY NEGLIGENCE in STATE/UT in all YEAR's?

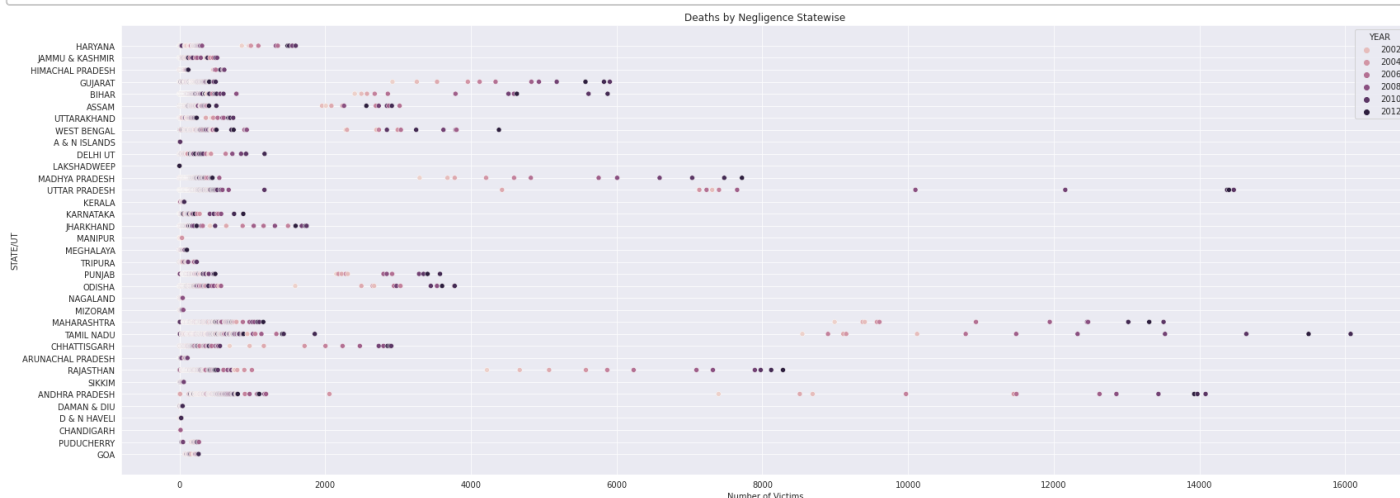
*#casuing negalicance vs state/ut vs year*

```
scat_df = project_df.sort_values('CAUSING DEATH BY NEGLIGENCE')[['YEAR', 'STATE/UT', 'CAUSING DEATH BY NEGLIGENCE']]
scat_df
```

	YEAR	STATE/UT	CAUSING DEATH BY NEGLIGENCE
<b>3824</b>	2006	HARYANA	0
<b>3108</b>	2005	JAMMU & KASHMIR	0
<b>3107</b>	2005	JAMMU & KASHMIR	0
<b>3094</b>	2005	HIMACHAL PRADESH	0
<b>3090</b>	2005	HARYANA	0
...	...	...	...
<b>8938</b>	2012	UTTAR PRADESH	14406
<b>7342</b>	2010	UTTAR PRADESH	14472
<b>7262</b>	2010	TAMIL NADU	14644
<b>8851</b>	2012	TAMIL NADU	15499
<b>8049</b>	2011	TAMIL NADU	16076

9017 rows × 3 columns

```
plt.figure(figsize=(28,10))
plt.xlabel('Number of Victims')
plt.ylabel("STATE/UT")
plt.title('Deaths by Negligence Statewise')
sns.scatterplot(x = scat_df['CAUSING DEATH BY NEGLIGENCE'],y = scat_df['STATE/UT'],hue=
```



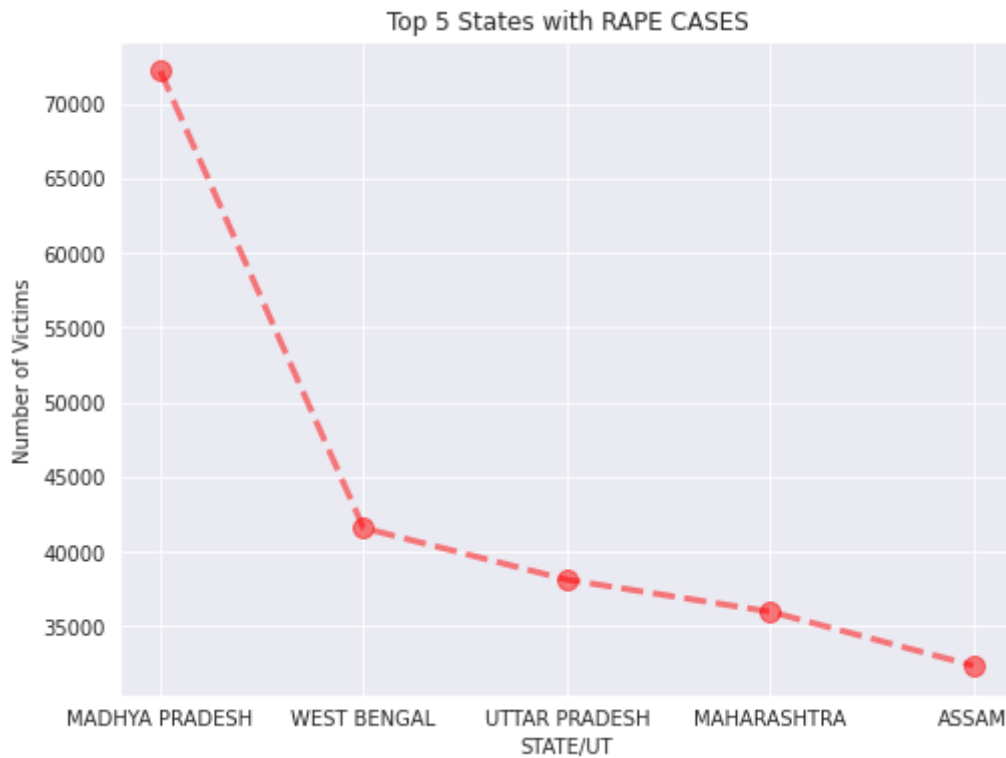
Insight from Above Scatter Plot : Clearly, It depict's that **TAMIL NADU** is leading with number of Deaths cased by negligence, 2nd highest **UTTAR PRADESH** and so on. The CAUSING DEATH BY NEGLIGENCE that happened between 2001-2012.

###Q. TOP 5 state's with highest Rape Cases ?

```
a = new_df.sort_values('RAPE',ascending= False)[['STATE/UT','RAPE']].head(5)
a
```

	STATE/UT	RAPE
19	MADHYA PRADESH	72174
34	WEST BENGAL	41574
32	UTTAR PRADESH	38116
20	MAHARASHTRA	35972
3	ASSAM	32356

```
plt.figure(figsize=(8,6))
plt.xlabel('STATE/UT')
plt.ylabel("Number of Victims")
plt.title('Top 5 States with RAPE CASES')
plt.plot(a['STATE/UT'],a['RAPE'],marker='o', color='r', ls='--', lw=3, ms=10, alpha=.5)
```



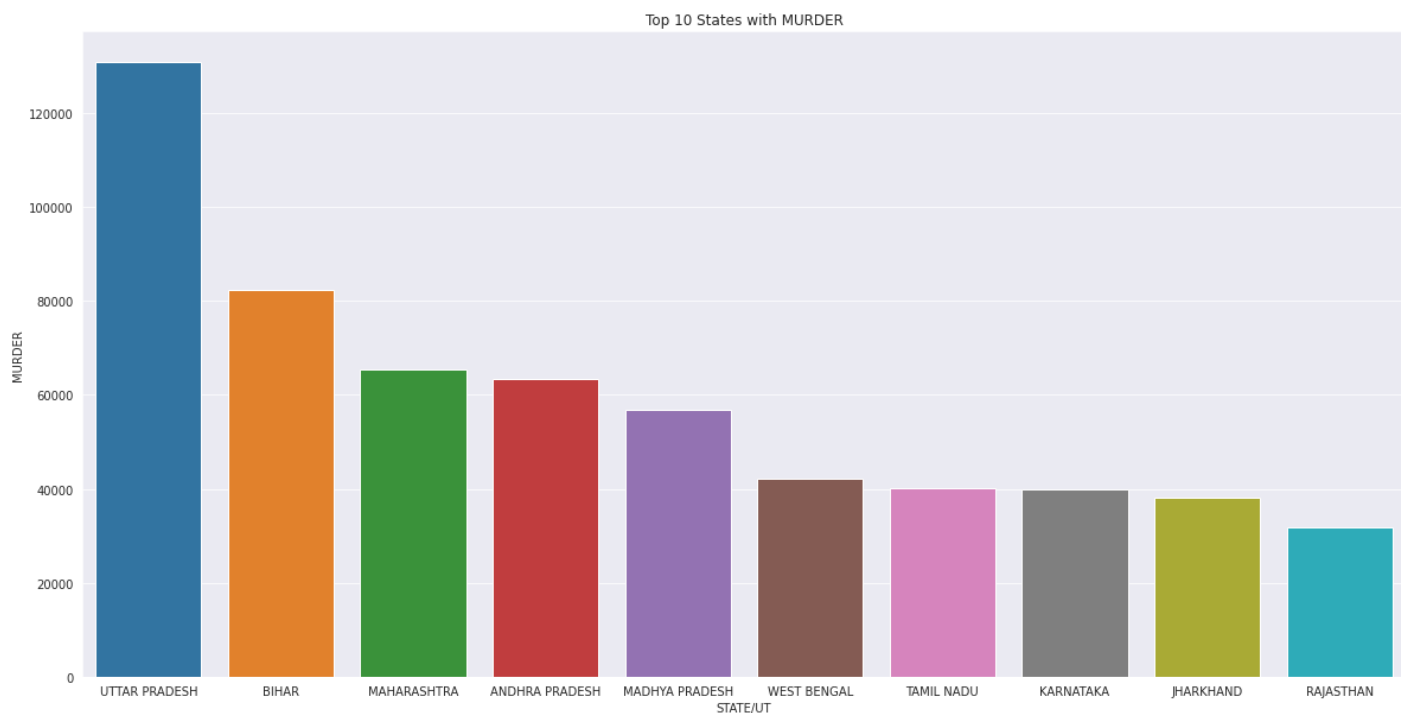
Insight from Above LINE Plot : Madhya Pradesh is having more no of cases when compared to other 4 with more than 70,000 cases.

###Q. TOP 10 state's with Maximum Murder Cases ?

```
b = new_df.sort_values('MURDER', ascending= False)[['STATE/UT', 'MURDER']].head(10)
b
```

	STATE/UT	MURDER
32	UTTAR PRADESH	130886
4	BIHAR	82490
20	MAHARASHTRA	65534
1	ANDHRA PRADESH	63512
19	MADHYA PRADESH	56798
34	WEST BENGAL	42112
30	TAMIL NADU	40254
16	KARNATAKA	39874
15	JHARKHAND	38120
28	RAJASTHAN	31688

```
plt.figure(figsize=(20,10))
plt.xlabel('STATE/UT')
plt.ylabel("Number of Victims")
plt.title('Top 10 States with MURDER')
sns.barplot(x=b['STATE/UT'], y=b['MURDER']);
```



Insight from Above Bar Plot : Uttar Pradesh is having more no of Murders when compared to other STATES/UT with more than 1,20,000 cases.

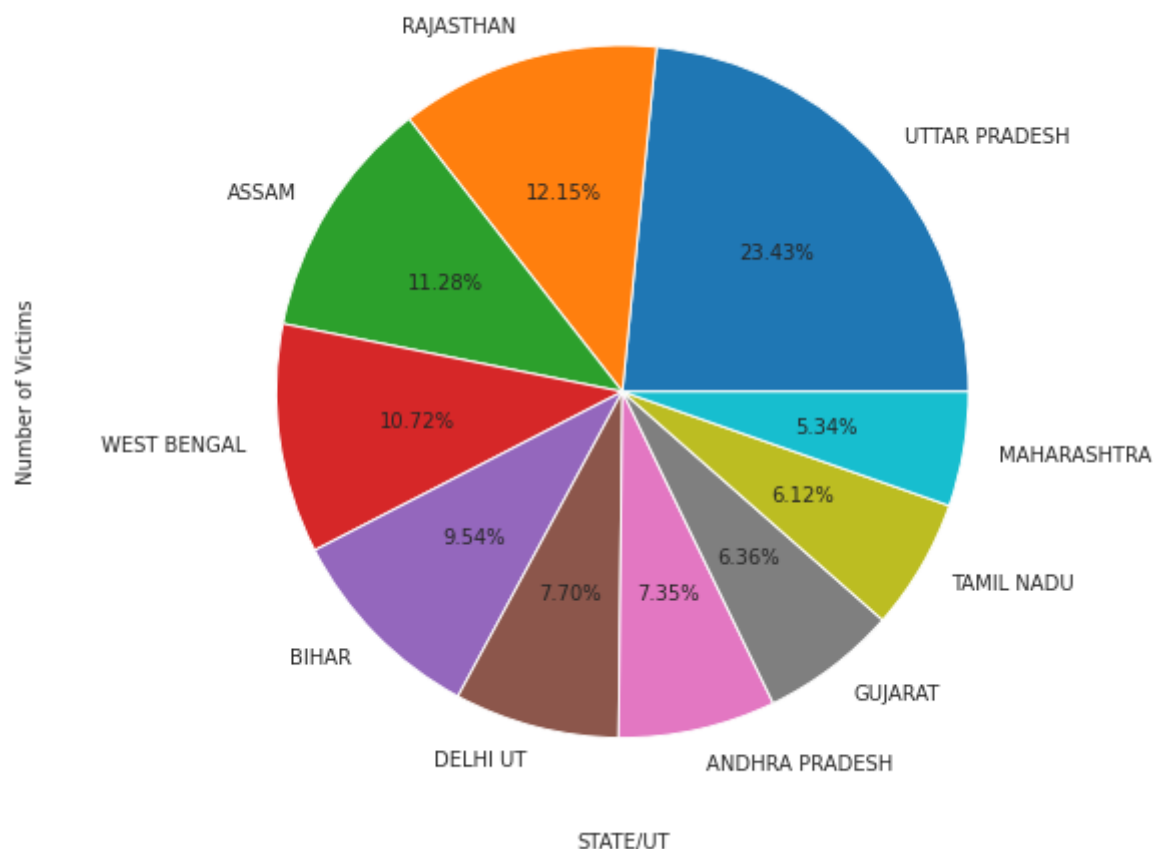
###Top 10 states with KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS.

```
c = new_df.sort_values('KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS', ascending= False)[
c
```

	STATE/UT	KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS
32	UTTAR PRADESH	95180
28	RAJASTHAN	49342
3	ASSAM	45840
34	WEST BENGAL	43546
4	BIHAR	38778
9	DELHI UT	31300
1	ANDHRA PRADESH	29872
11	GUJARAT	25836
30	TAMIL NADU	24872
20	MAHARASHTRA	21706

```
fig = plt.figure(figsize=(8,6))
ax = fig.add_axes([0,0,1,1])
ax.axis('equal')
plt.xlabel('STATE/UT')
plt.ylabel("Number of Victims")
plt.title('KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS')
mylabels = c['STATE/UT']
no_of_Victims = c['KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS']
ax.pie(no_of_Victims, labels = mylabels, autopct='%1.2f%%')
plt.show()
```

## KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS



Insight from Above Pie Plot : Uttar Pradesh is having **KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS** more when compared to other **STATES/UT** with 23.43%.

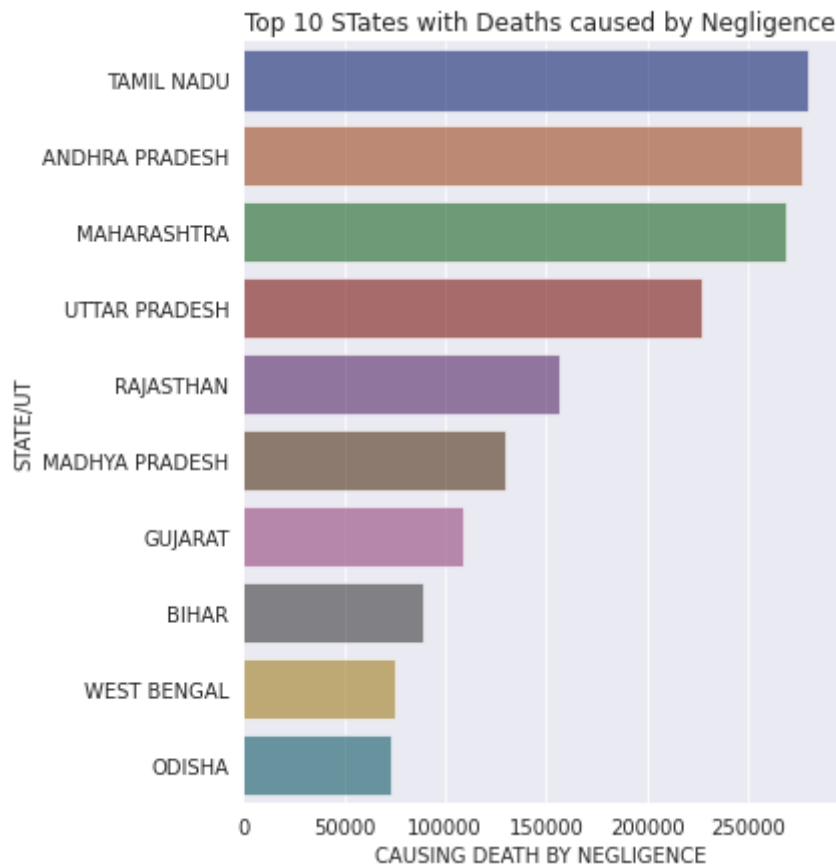
Top 10 STATES/UT CAUSING DEATH BY list top 10 states.

```
d = new_df.sort_values('CAUSING DEATH BY NEGLIGENCE', ascending= False)[['STATE/UT', 'CAUSING DEATH BY NEGLIGENCE']]
```

	STATE/UT	CAUSING DEATH BY NEGLIGENCE
30	TAMIL NADU	280386
1	ANDHRA PRADESH	276870
20	MAHARASHTRA	269226
32	UTTAR PRADESH	227642
28	RAJASTHAN	156700
19	MADHYA PRADESH	129938
11	GUJARAT	108770
4	BIHAR	88884
34	WEST BENGAL	75502
25	ODISHA	73014

Top 10 STATES/UT with Deaths cased by Negligence.

```
g = sns.catplot(
    data=d, kind="bar",
    x="CAUSING DEATH BY NEGLIGENCE", y="STATE/UT",
    ci="sd", palette="dark", alpha=.6, height=6
)
plt.title('Top 10 States with Deaths caused by Negligence');
```



Insight from Above BAR Plot : TAMIL NADU is having **CAUSING DEATH BY NEGLIGENCE** more when compared to other **STATES/UT**.

### ##Inferences And Conclusion

###The main goal of this project is to analyse the crime rates that took place in India from 2001-2012.

###In this project we used various charts, sorting, aggregation and many other Python concepts to get the in-depth analysis of this data set, we have asked few questions and summarized respectively.

1. More than 6.1 Million people were victims for the crimes that took place in many states in India under MURDER, RAPE, CHEATING, IMPORTATION OF GIRLS and more.
2. We found that in time span of 2001 - 2012 murder was increasing and more than attempt to murder's.
3. From State/UT Bar graph we would get solid information of rape's that took place and were huge in number from Madhya Pradesh with more than 70,000 victims.
4. We even got to know that Murder's that took place between all the states were more than rape's that were took place. i.e crime rate of murder cases were more than rape between 2001-2012.
5. We have got many conclusions from that 6 in 1 graph and their insights were given below.
6. We found 5 states which have maximum rapes, murder, kidnapping and abduction of women and girls,
7. We found top 10 states causing death by negligence.



## ##References and Future Work

### #Reference -

1. For knowing more about pandas and it's functions in deatil - <https://pandas.pydata.org>
2. For more ideas on Matplotlib and it's library- <https://matplotlib.org>
3. Used this for dropping a column <https://www.geeksforgeeks.org/python-pandas-dataframe/>
4. Used this for plotting diferent types of graphs <https://seaborn.pydata.org/#>

### ##Future Work-

As this data analysis has given me a berif description on handling of big data. I like to work on cleaning the data and playing around with graphs. This data set had many columns had to cut short into few my further analysis is to get them on screen and work with the same.

```
import jovian
```