```
#Assignment 8th september
#Take car crashes dataset from seaborn library
#load dataset
#data visualiation
#Inference is must for each and every graph
```

```
#Import necessary libraries
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd
```

```
crashes = sns.load_dataset('car_crashes')
```

```
crashes
```

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	al
0	18.8	7.332	5.640	18.048	15.040	784.55	145.08	
1	18.1	7.421	4.525	16.290	17.014	1053.48	133.93	
2	18.6	6.510	5.208	15.624	17.856	899.47	110.35	
3	22.4	4.032	5.824	21.056	21.280	827.34	142.39	
4	12.0	4.200	3.360	10.920	10.680	878.41	165.63	
5	13.6	5.032	3.808	10.744	12.920	835.50	139.91	
6	10.8	4.968	3.888	9.396	8.856	1068.73	167.02	
7	16.2	6.156	4.860	14.094	16.038	1137.87	151.48	
8	5.9	2.006	1.593	5.900	5.900	1273.89	136.05	
9	17.9	3.759	5.191	16.468	16.826	1160.13	144.18	
10	15.6	2.964	3.900	14.820	14.508	913.15	142.80	
11	17.5	9.450	7.175	14.350	15.225	861.18	120.92	
12	15.3	5.508	4.437	13.005	14.994	641.96	82.75	
13	12.8	4.608	4.352	12.032	12.288	803.11	139.15	
14	14.5	3.625	4.205	13.775	13.775	710.46	108.92	
15	15.7	2.669	3.925	15.229	13.659	649.06	114.47	
16	17.8	4.806	4.272	13.706	15.130	780.45	133.80	
17	21.4	4.066	4.922	16.692	16.264	872.51	137.13	
18	20.5	7.175	6.765	14.965	20.090	1281.55	194.78	
19	15.1	5.738	4.530	13.137	12.684	661.88	96.57	
20	12.5	4.250	4.000	8.875	12.375	1048.78	192.70	
21	8.2	1.886	2.870	7.134	6.560	1011.14	135.63	
22	14.1	3.384	3.948	13.395	10.857	1110.61	152.26	
23	9.6	2.208	2.784	8.448	8.448	777.18	133.35	
24	17.6	2.640	5.456	1.760	17.600	896.07	155.77	
25	16.1	6.923	5.474	14.812	13.524	790.32	144.45	
26	21.4	8.346	9.416	17.976	18.190	816.21	85.15	
27	14.9	1.937	5.215	13.857	13.410	732.28	114.82	
28	14.7	5.439	4.704	13.965	14.553	1029.87	138.71	
29	11.6	4.060	3.480	10.092	9.628	746.54	120.21	

30	11.2	1.792	3.136	9.632	8.736	1301.52	159.85
31	18.4	3.496	4.968	12.328	18.032	869.85	120.75

```
missing_values = crashes.isnull().sum()
```

```
print(missing_values)
```

```
total             0
speeding          0
alcohol           0
not_distracted    0
no_previous       0
ins_premium       0
ins_losses        0
abbrev            0
dtype: int64
```

```
df_cleaned = crashes.dropna()
```

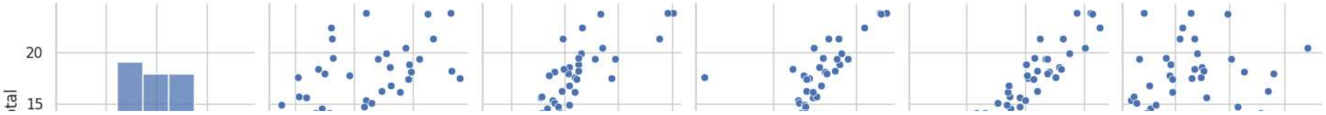
```
df_cleaned = crashes.dropna(axis=1)
```

```
df_cleaned
```

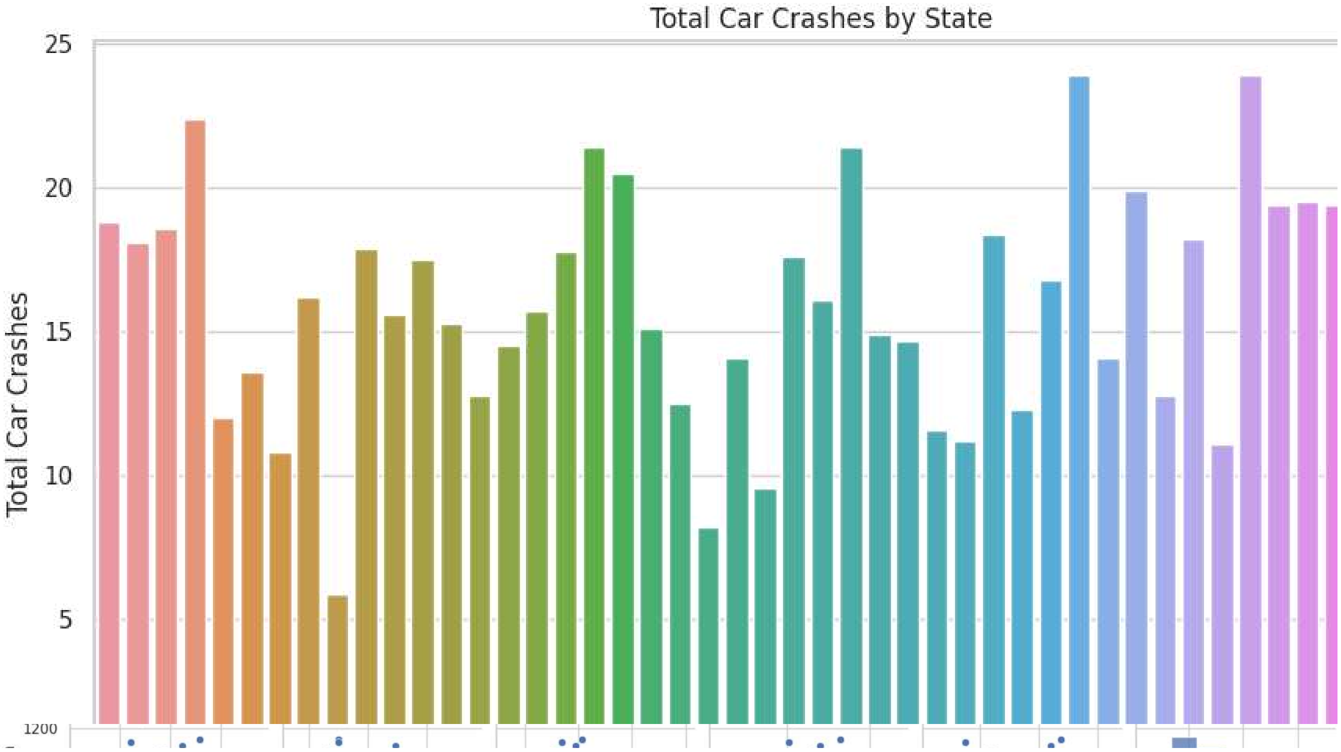
	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbr
0	18.8	7.332	5.640	18.048	15.040	784.55	145.08	
1	18.1	7.421	4.525	16.290	17.014	1053.48	133.93	
2	18.6	6.510	5.208	15.624	17.856	899.47	110.35	
3	22.4	4.032	5.824	21.056	21.280	827.34	142.39	
4	12.0	4.200	3.360	10.920	10.680	878.41	165.63	
5	13.6	5.032	3.808	10.744	12.920	835.50	139.91	
6	10.8	4.968	3.888	9.396	8.856	1068.73	167.02	
7	16.2	6.156	4.860	14.094	16.038	1137.87	151.48	
8	5.9	2.006	1.593	5.900	5.900	1273.89	136.05	
9	17.9	3.759	5.191	16.468	16.826	1160.13	144.18	
10	15.6	2.964	3.900	14.820	14.508	913.15	142.80	
11	17.5	9.450	7.175	14.350	15.225	861.18	120.92	
12	15.3	5.508	4.437	13.005	14.994	641.96	82.75	
13	12.8	4.608	4.352	12.032	12.288	803.11	139.15	
14	14.5	3.625	4.205	13.775	13.775	710.46	108.92	
15	15.7	2.669	3.925	15.229	13.659	649.06	114.47	
16	17.8	4.806	4.272	13.706	15.130	780.45	133.80	
17	21.4	4.066	4.922	16.692	16.264	872.51	137.13	
18	20.5	7.175	6.765	14.965	20.090	1281.55	194.78	
19	15.1	5.738	4.530	13.137	12.684	661.88	96.57	
20	12.5	4.250	4.000	8.875	12.375	1048.78	192.70	
21	8.2	1.886	2.870	7.134	6.560	1011.14	135.63	
22	14.1	3.384	3.948	13.395	10.857	1110.61	152.26	
23	9.6	2.208	2.784	8.448	8.448	777.18	133.35	
24	17.6	2.640	5.456	1.760	17.600	896.07	155.77	
25	16.1	6.923	5.474	14.812	13.524	790.32	144.45	
26	21.4	8.346	9.416	17.976	18.190	816.21	85.15	
27	14.9	1.937	5.215	13.857	13.410	732.28	114.82	
28	14.7	5.439	4.704	13.965	14.553	1029.87	138.71	
29	11.6	4.060	3.480	10.092	9.628	746.54	120.21	

30	11.2	1.792	3.136	9.632	8.736	1301.52	159.85	
31	18.4	3.496	4.968	12.328	18.032	869.85	120.75	N
32	12.3	3.936	3.567	10.824	9.840	1234.31	150.01	I
33	16.8	6.552	5.208	15.792	13.608	708.24	127.82	N
34	23.9	5.497	10.038	23.661	20.554	688.75	109.72	N
35	14.1	3.948	4.794	13.959	11.562	697.73	133.52	C

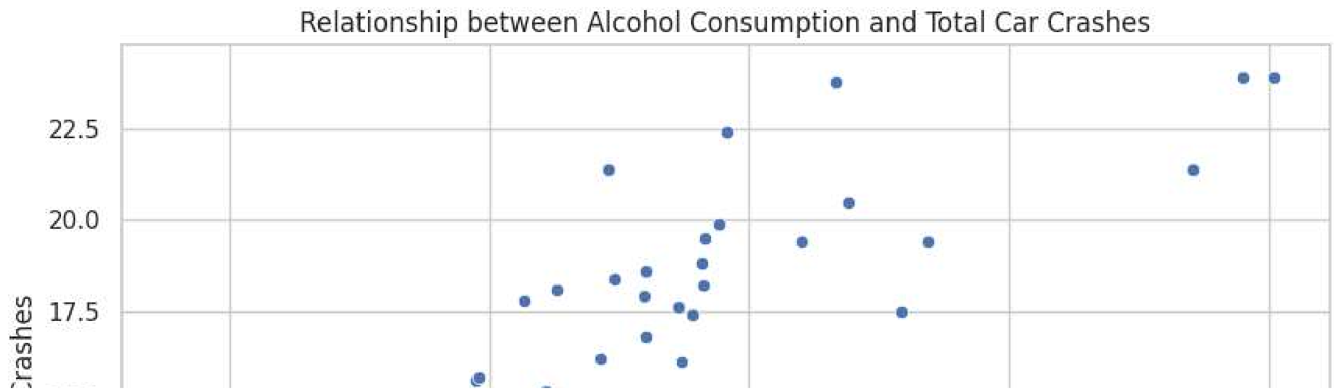
```
crashes = sns.load_dataset('car_crashes')

# Set the style for the plots
sns.set(style="whitegrid")

# Create a pair plot to visualize relationships between numeric variables
sns.pairplot(crashes)
plt.show()
```



```
plt.figure(figsize=(12, 6))
sns.barplot(x='abbrev', y='total', data=crashes)
plt.title('Total Car Crashes by State')
plt.xlabel('State Abbreviation')
plt.ylabel('Total Car Crashes')
plt.show()
```



```
# Create a scatter plot to visualize the relationship between "alcohol" and "total" columns
plt.figure(figsize=(10, 6))
sns.scatterplot(x='alcohol', y='total', data=crashes)
plt.title('Relationship between Alcohol Consumption and Total Car Crashes')
plt.xlabel('Alcohol Consumption (Percentage)')
plt.ylabel('Total Car Crashes')
plt.show()
```



```
# Create a heatmap to visualize the correlation matrix
correlation_matrix = crashes.corr()
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap="coolwarm")
plt.title('Correlation Heatmap')
plt.show()
```