21Bec7140
Pasupula viswanadhaData_Visualization

Importing Libraries

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

Loading Dataset Car_Crashes

In [2]:
```
df = pd.read_csv('car_crashes.cs
```

In [3]:
```
df
```

Out[3]:

	total	speeding	alcohol	not_distrac
50	17.4	7.308	5.568	14.0

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
 RangeIndex: 51 entries, 0 to 50
 Data columns (total 8 columns):
 #   Column          Non-Null Count  Dtype
```

In [4]:

	total	speeding	alcohol	not_distrac
0	18.8	7.332	5.640	18.0
1	18.1	7.421	4.525	16.2
2	18.6	6.510	5.208	15.6
3	22.4	4.032	5.824	21.0
4	12.0	4.200	3.360	10.9
5	13.6	5.032	3.808	10.7
6	10.8	4.968	3.888	9.3
7	16.2	6.156	4.860	14.0
8	5.9	2.006	1.593	5.9
9	17.9	3.759	5.191	16.4
10	15.6	2.964	3.900	14.8
11	17.5	9.450	7.175	14.3
12	15.3	5.508	4.437	13.0
13	12.8	4.608	4.352	12.0
14	14.5	3.625	4.205	13.7
15	15.7	2.669	3.925	15.2
16	17.8	4.806	4.272	13.7
17	21.4	4.066	4.922	16.6
18	20.5	7.175	6.765	14.9
19	15.1	5.738	4.530	13.1
20	12.5	4.250	4.000	8.8
21	8.2	1.886	2.870	7.1
22	14.1	3.384	3.948	13.3
23	9.6	2.208	2.784	8.4
24	17.6	2.640	5.456	1.7
25	16.1	6.923	5.474	14.8
26	21.4	8.346	9.416	17.9
27	14.9	1.937	5.215	13.8
28	14.7	5.439	4.704	13.9
29	11.6	4.060	3.480	10.0
30	11.2	1.792	3.136	9.6
31	18.4	3.496	4.968	12.3
32	12.3	3.936	3.567	10.8
33	16.8	6.552	5.208	15.7
34	23.9	5.497	10.038	23.6
35	14.1	3.948	4.794	13.9
36	19.9	6.368	5.771	18.3
37	12.8	4.224	3.328	8.5
38	18.2	9.100	5.642	17.4
39	11.1	3.774	4.218	10.2
40	23.9	9.082	9.799	22.9
41	19.4	6.014	6.402	19.0
42	19.5	4.095	5.655	15.9
43	19.4	7.760	7.372	17.6
44	11.3	4.859	1.808	9.9
45	13.6	4.080	4.080	13.0
46	12.7	2.413	3.429	11.0
47	10.6	4.452	3.498	8.6
48	23.8	8.092	6.664	23.0
49	13.8	4.968	4.554	5.3

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
 RangeIndex: 51 entries, 0 to 50Data columns (total 8 columns):#   Column          Non-Null Count  Dtype
 0   total           51 non-null     float64
 1   speeding        51 non-null     float64
 2   alcohol         51 non-null     float64
 3   not_distracted  51 non-null     float64
 4   no_previous     51 non-null     float64
 5   ins_premium     51 non-null     float64
 6   ins_losses      51 non-null     float64
 7   abbrev          51 non-null     object
dtypes: float64(7), object(1)
memory usage: 3.3+ KB
```

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

Out[5]:

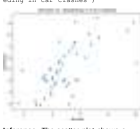
	total	speeding	alcohol	not_distracted
0	18.8	7.332	5.640	18.0
1	18.1	7.421	4.525	16.2
2	18.6	6.510	5.208	15.6
3	22.4	4.032	5.824	21.0
4	12.0	4.200	3.360	10.9

Data Visualization with Inference

- Scatter Plot

In [6]:
```
sns.scatterplot(x="alcohol", y=
plt.title("Alcohol vs. Speeding
```

Out[6]:
```
Text(0.5, 1.0, 'Alcohol vs. Speeding in Car Crashes')
```



Inference: The scatter plot shows a positive correlation between alcohol consumption and speeding involvement in car crashes, stating that higher alcohol consumption tend to have higher speeding involvement.

- Line Plot

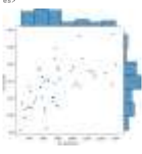
In [7]:
```
plt.figure(figsize=(12, 6))
sns.lineplot(x="abbrev", y="alc
plt.title('Alcohol Consumption
plt.xticks(rotation=90)
plt.show()
```

Inference: The line plot shows the alcohol consumption of each state (abbrev). It shows that state (abbrev) "ND" has the highest alcohol consumption among the observed states.

- Joint Plot

In [8]:
```
plt.figure(figsize=(12, 8))
sns.jointplot(x='ins_premium', y
```

Out[8]:
```
<seaborn.axisgrid.JointGrid at 0x25df2fee4d0>
<Figure size 1200x800 with 0 Axes>
```



Inference: The joint plot displays the bivariate relationship between insurance premium and losses.The lower insurance premiums is associated with lower insurance losses.

- Bar Plot

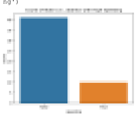
In [9]:
```
plt.figure(figsize=(12, 6))
sns.barplot(x='abbrev',  y='speed
plt.title('Average Speeding in E
plt.xticks(rotation=90)
plt.show()
```

Inference: state (abbrev) "NJ" has the lowest speeding, while state "HI" has the highest average speeding among the state (abbrev).

- Count Plot

In [10]:
```
sns.countplot(x=df['speeding'] >
plt.title('Count of State (i.e.,
```

Out[10]:
```
Text(0.5, 1.0, 'Count of State (i.e., abbrev) with High Speeding')
```



Inference: The count plot shows that a significant number of states (abbrev) have low speeding rates (speeding < 7). This states that a substantial portion of the states (abbrev) has below-average speeding behavior.

- Dist Plot

In [17]:
```
sns.distplot(df['speeding'])
plt.title('Distribution of Speed
```

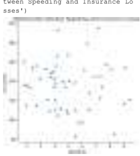
Out[17]:
```
Text(0.5, 1.0, 'Distribution of Speeding Rates')
```

Inference: This displot provides a visual representation of the distribution of speeding rates across the dataset. It states that the distribution is right-skewed, indicating that a majority of the observed data points have lower speeding rates (speeding < 7) , while a smaller number of data points have higher speeding rates.

- Rel Plot

In [12]:
```
sns.relplot(x='speeding', y='ins
plt.title('Relationship between
```

Out[12]:
```
Text(0.5, 1.0, 'Relationship between Speeding and Insurance Losses')
```



Inference :- There is a positive correlation between speeding and insurance losses. States (abbrev) with higher average speeding tend to have higher insurance losses.

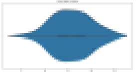
Box Plot

In [13]:
```
plt.figure(figsize=(12, 6))
sns.boxplot(x='abbrev',  y='ins_p
plt.title('Box Plot of Insurance
plt.xticks(rotation=90)
plt.show()
```


Inference :- The box plot shows the distribution of insurance premiums by state. It highlights variations in ins_premium amounts across different states, with some states having higher ins_premiums.

- Violin Plot

In [14]:
```
plt.figure(figsize=(12, 6))
sns.violinplot(x=df["total"])
plt.title('Violin Plot of Total'
plt.xlabel('Total')
plt.show()
```



Inference :- The white dot in the center of the violin represents the median value i.e., 15.6.The violin appears to be roughly symmetrical, indicating that the data distribution is somewhat balanced.

- Pair Plot

In [15]:
```
sns.pairplot(df[['total', 'speed
plt.suptitle('Pair Plots')
plt.show()
```

Inference :- This pair plot displays pairwise scatter plots for selected columns (total,speeding, alcohol, not_distracted,no_previous,ins_premium ins_losses). It allows for the visualization of relationships between these variables.