```
pip install seaborn
```

```
import matplotlib.pyplot as plt
```

```
import seaborn as sns # Import the Seaborn library and alias it as
'sns'
```

```
    'sns'
```

```
print(sns.get_dataset_names())
```

```
    ['anagrams', 'anscombe', 'attention', 'brain_networks', 'car_crashes', 'diamonds', 'dots', 'dowjones', 'exercise', 'flights', 'fmri
```

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

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses
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

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
df.head()
```

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev
0	18.8	7.332	5.640	18.048	15.040	784.55	145.08	AL
1	18.1	7.421	4.525	16.290	17.014	1053.48	133.93	AK
2	18.6	6.510	5.208	15.624	17.856	899.47	110.35	AZ
3	22.4	4.032	5.824	21.056	21.280	827.34	142.39	AR
4	12.0	4.200	3.360	10.920	10.680	878.41	165.63	CA

```
df.tail()
```

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev
46	12.7	2.413	3.429	11.049	11.176	768.95	153.72	VA
47	10.6	4.452	3.498	8.692	9.116	890.03	111.62	WA
48	23.8	8.092	6.664	23.086	20.706	992.61	152.56	WV
49	13.8	4.968	4.554	5.382	11.592	670.31	106.62	WI
50	17.4	7.308	5.568	14.094	15.660	791.14	122.04	WY

```
df.describe()
```

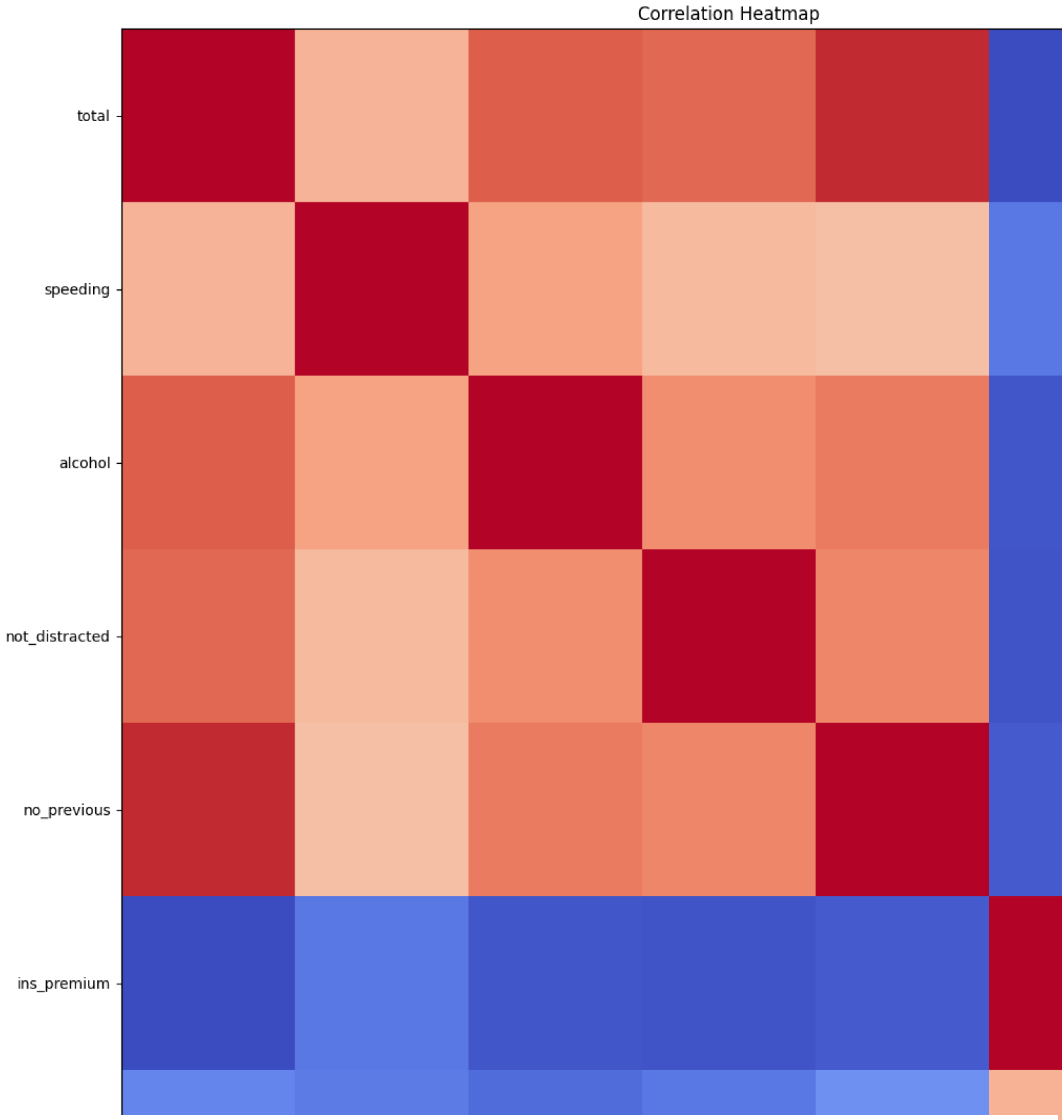
	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses
count	51.000000	51.000000	51.000000	51.000000	51.000000	51.000000	51.000000
mean	15.790196	4.998196	4.886784	13.573176	14.004882	886.957647	134.493137
std	4.122002	2.017747	1.729133	4.508977	3.764672	178.296285	24.835922
min	5.900000	1.792000	1.593000	1.760000	5.900000	641.960000	82.750000
25%	12.750000	3.766500	3.894000	10.478000	11.348000	768.430000	114.645000
50%	15.600000	4.608000	4.554000	13.857000	13.775000	858.970000	136.050000
75%	18.500000	6.439000	5.604000	16.140000	16.755000	1007.945000	151.870000
max	23.900000	9.450000	10.038000	23.661000	21.280000	1301.520000	194.780000

```
corr=df.corr()
corr
```

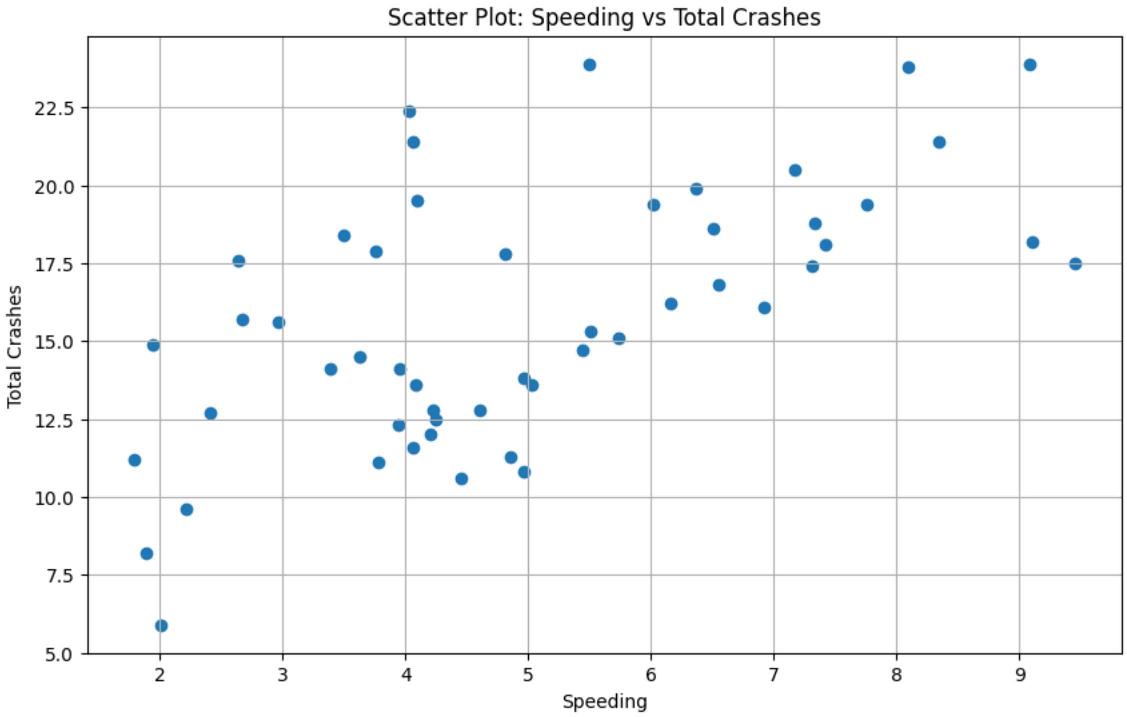
```
<ipython-input-12-7d5195e2bf4d>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future ver
  corr=df.corr()
```

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses
total	1.000000	0.611548	0.852613	0.827560	0.956179	-0.199702	-0.036011
speeding	0.611548	1.000000	0.669719	0.588010	0.571976	-0.077675	-0.065928
alcohol	0.852613	0.669719	1.000000	0.732816	0.783520	-0.170612	-0.112547
not_distracted	0.827560	0.588010	0.732816	1.000000	0.747307	-0.174856	-0.075970
no_previous	0.956179	0.571976	0.783520	0.747307	1.000000	-0.156895	-0.006359
ins_premium	-0.199702	-0.077675	-0.170612	-0.174856	-0.156895	1.000000	0.623116

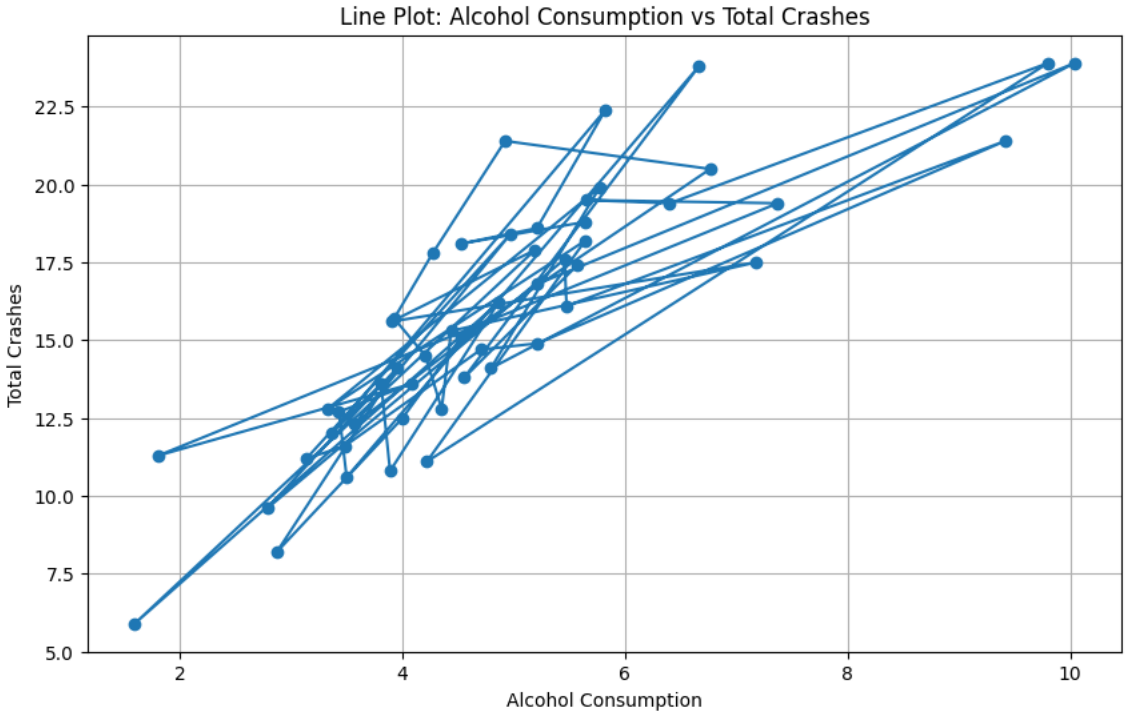
```
plt.figure(figsize=(20, 15))
plt.imshow(corr, cmap='coolwarm', interpolation='nearest')
plt.colorbar()
plt.xticks(range(len(corr.columns)), corr.columns, rotation=90)
plt.yticks(range(len(corr.columns)), corr.columns)
plt.title('Correlation Heatmap')
plt.show()
```



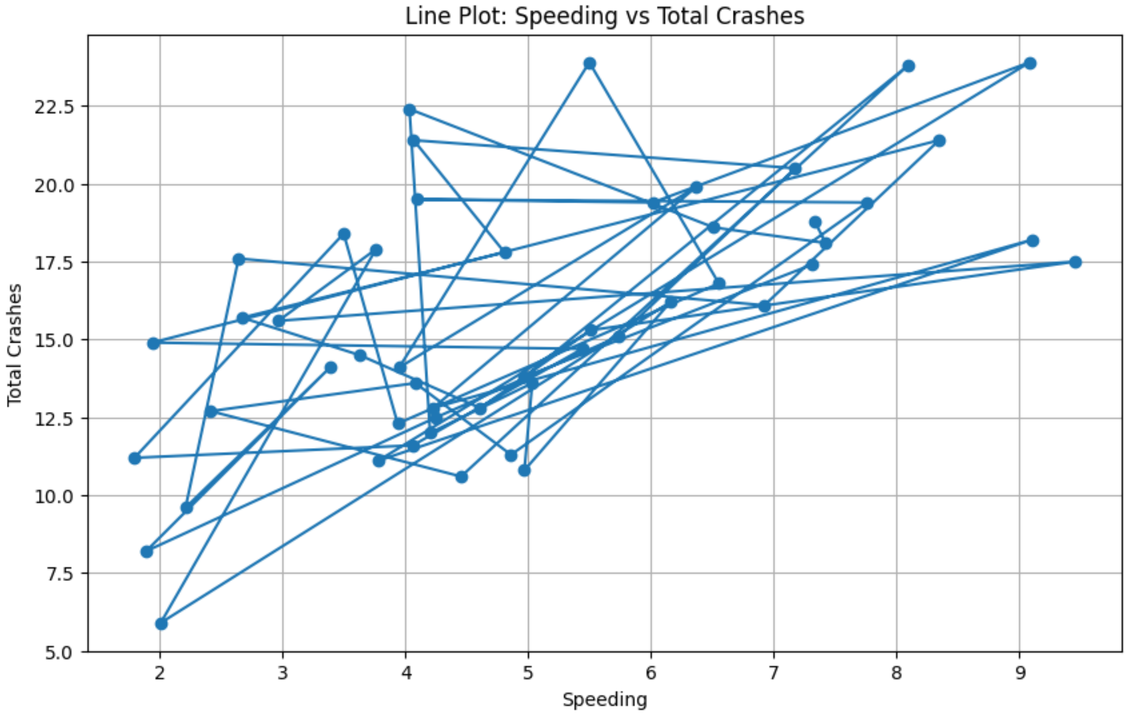
```
plt.figure(figsize=(10, 6))
plt.scatter(df['speeding'], df['total'])
plt.xlabel('Speeding')
plt.ylabel('Total Crashes')
plt.title('Scatter Plot: Speeding vs Total Crashes')
plt.grid(True)
plt.show()
```



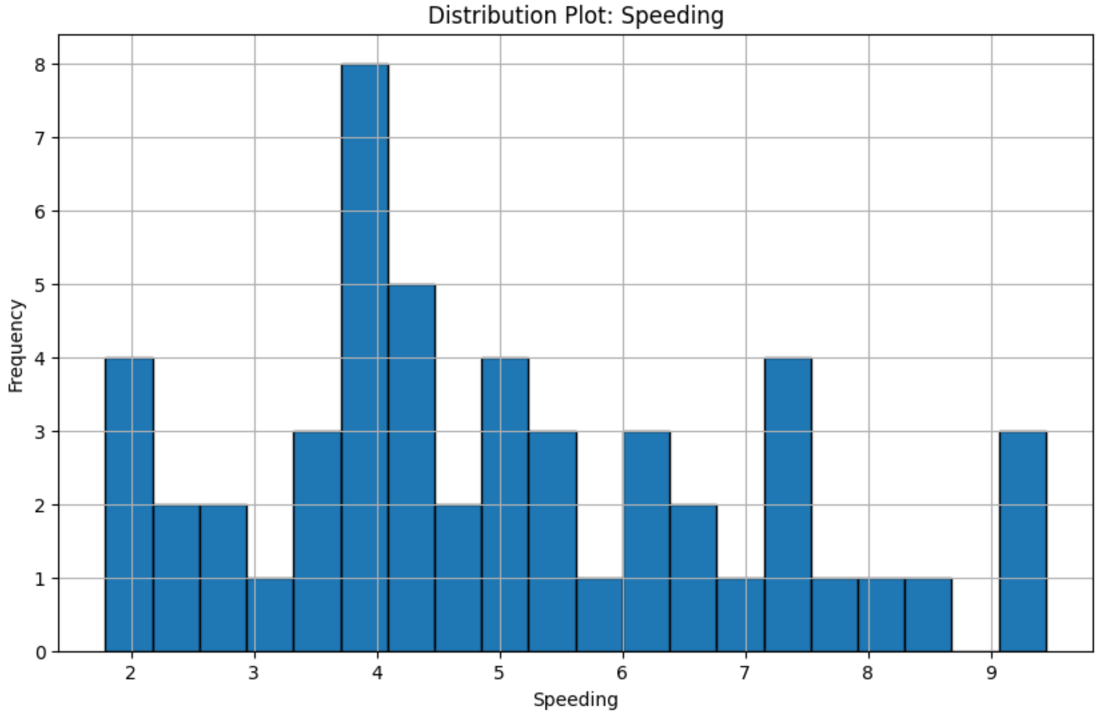
```
plt.figure(figsize=(10, 6))
plt.plot(df['alcohol'], df['total'], marker='o', linestyle='-')
plt.xlabel('Alcohol Consumption')
plt.ylabel('Total Crashes')
plt.title('Line Plot: Alcohol Consumption vs Total Crashes')
plt.grid(True)
plt.show()
```



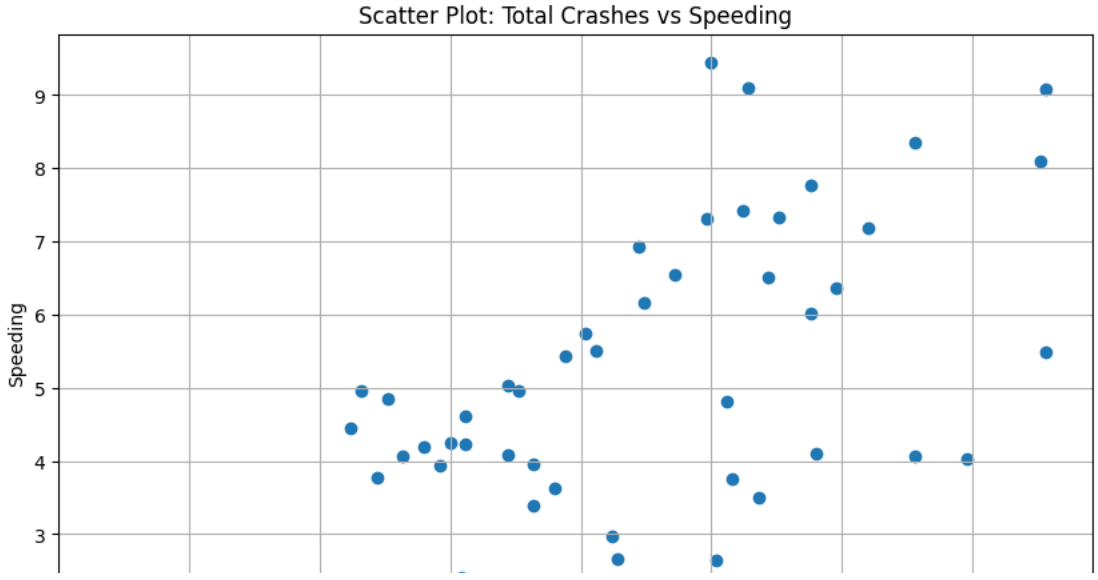
```
plt.figure(figsize=(10, 6))
plt.plot(df['speeding'], df['total'], marker='o', linestyle='-')
plt.xlabel('Speeding')
plt.ylabel('Total Crashes')
plt.title('Line Plot: Speeding vs Total Crashes')
plt.grid(True)
plt.show()
```



```
plt.figure(figsize=(10, 6))
plt.hist(df['speeding'], bins=20, edgecolor='k')
plt.xlabel('Speeding')
plt.ylabel('Frequency')
plt.title('Distribution Plot: Speeding')
plt.grid(True)
plt.show()
```



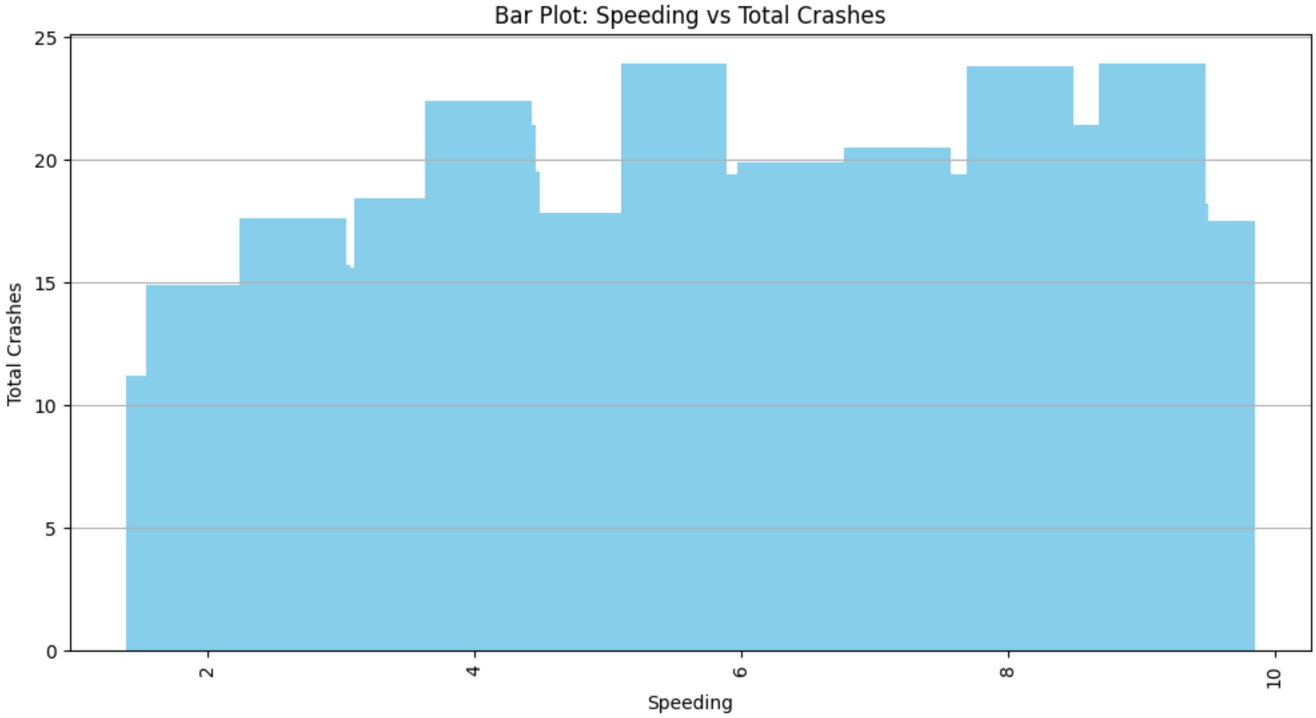
```
plt.figure(figsize=(10, 6))
plt.scatter(df['total'], df['speeding'])
plt.xlabel('Total Crashes')
plt.ylabel('Speeding')
plt.title('Scatter Plot: Total Crashes vs Speeding')
plt.grid(True)
plt.show()
```



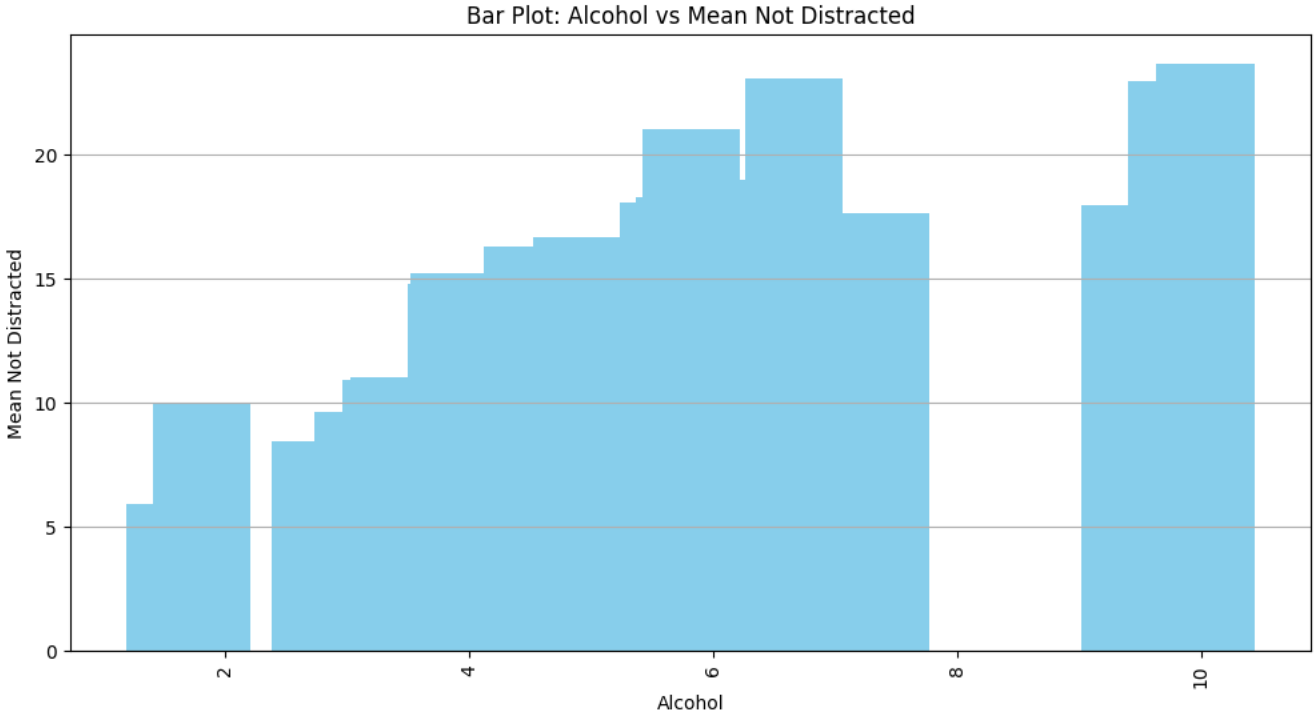
```
df["ins_losses"].value_counts()
```

```
    145.08    1
    153.86    1
    138.71    1
    120.21    1
    159.85    1
    120.75    1
    150.01    1
    127.82    1
    109.72    1
    133.52    1
    178.86    1
    104.61    1
    148.58    1
    85.15     1
    116.29    1
    96.87     1
    155.57    1
    156.83    1
    109.48    1
    109.61    1
    153.72    1
    111.62    1
    152.56    1
    106.62    1
    114.82    1
    144.45    1
    133.93    1
    82.75     1
    110.35    1
    142.39    1
    165.63    1
    139.91    1
    167.02    1
    151.48    1
    136.05    1
    144.18    1
    142.80    1
    120.92    1
    139.15    1
    155.77    1
    108.92    1
    114.47    1
    133.80    1
    137.13    1
    194.78    1
    96.57     1
    192.70    1
    135.63    1
    152.26    1
    133.35    1
    122.04    1
    Name: ins_losses, dtype: int64
```

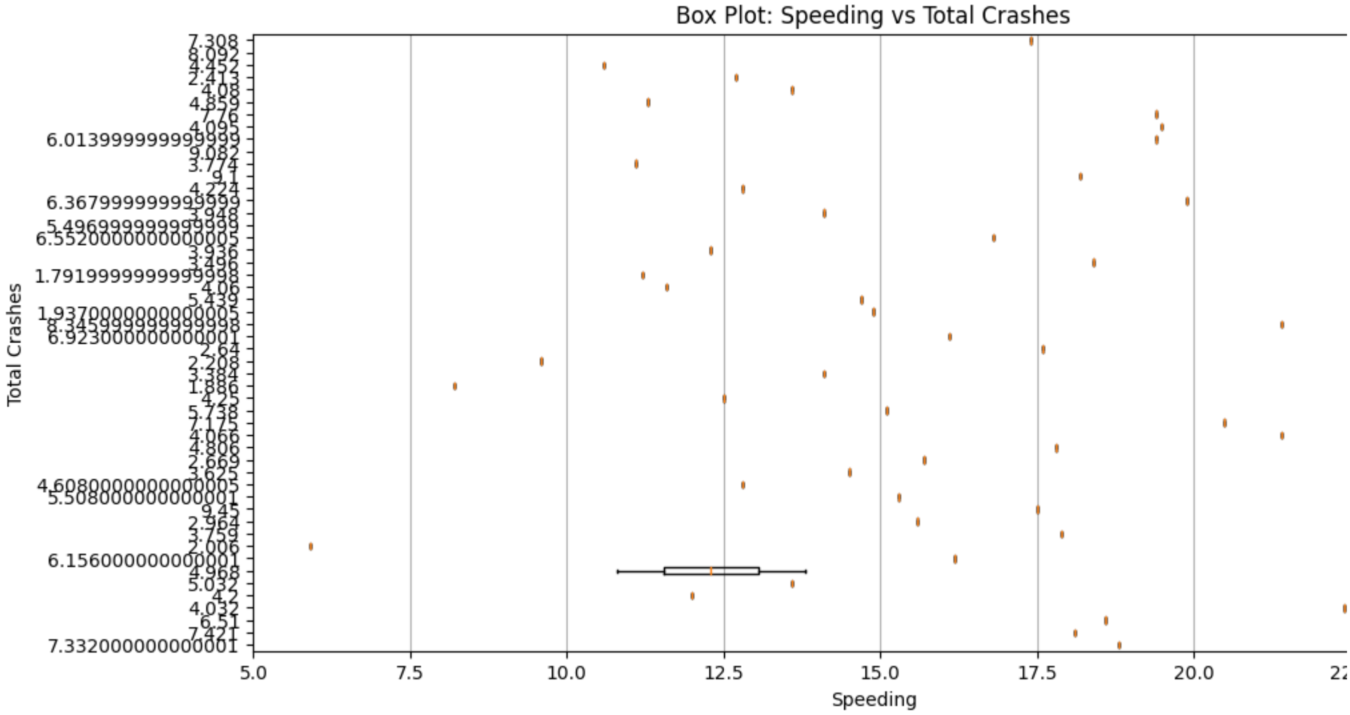
```
# Assuming 'df' is your DataFrame containing the dataset
plt.figure(figsize=(12, 6))
plt.bar(df['speeding'], df['total'], color='skyblue')
plt.xlabel('Speeding')
plt.ylabel('Total Crashes')
plt.title('Bar Plot: Speeding vs Total Crashes')
plt.xticks(rotation=90)  # Rotate x-axis labels if needed
plt.grid(axis='y')
plt.show()
```



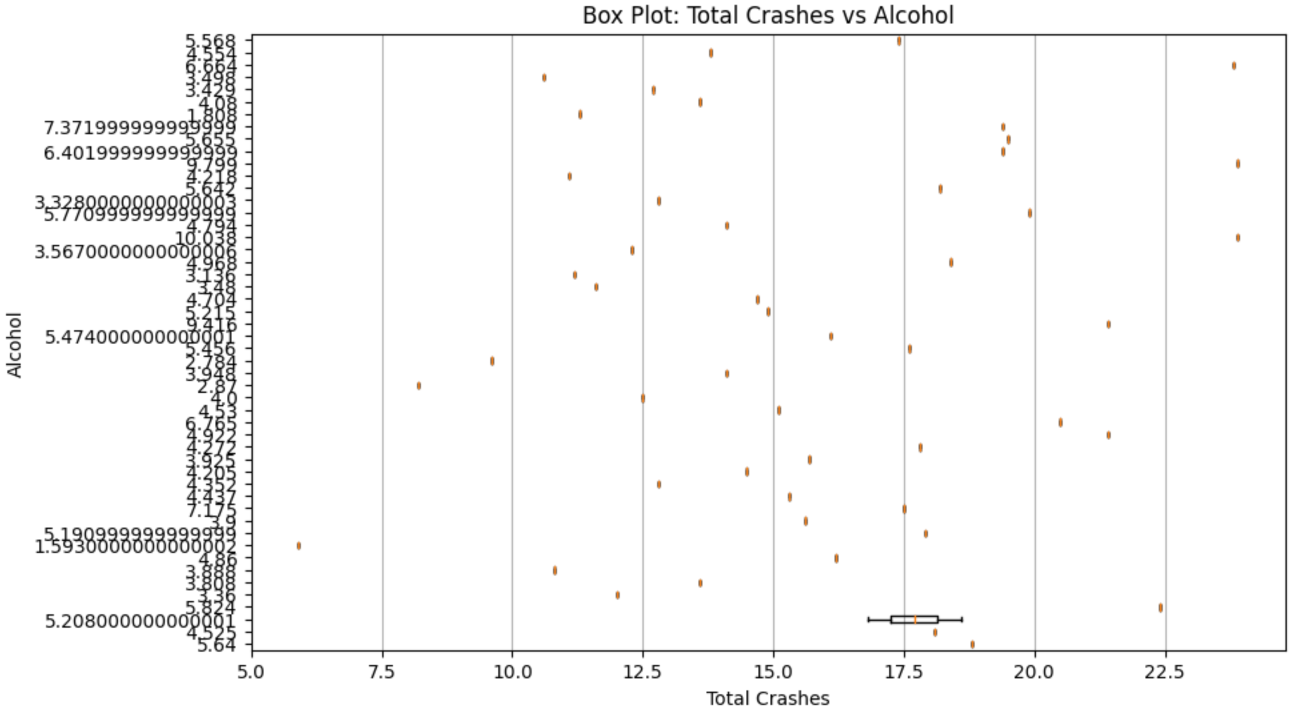
```
# Assuming 'df' is your DataFrame containing the dataset
plt.figure(figsize=(12, 6))
grouped_data = df.groupby('alcohol')['not_distracted'].mean().reset_index()
plt.bar(grouped_data['alcohol'], grouped_data['not_distracted'], color='skyblue')
plt.xlabel('Alcohol')
plt.ylabel('Mean Not Distracted')
plt.title('Bar Plot: Alcohol vs Mean Not Distracted')
plt.xticks(rotation=90)  # Rotate x-axis labels if needed
plt.grid(axis='y')
plt.show()
```



```
plt.figure(figsize=(12, 6))
box_data = [df[df['speeding'] == category]['total'] for category in df['speeding'].unique()]
plt.boxplot(box_data, labels=df['speeding'].unique(), vert=False)
plt.xlabel('Speeding')
plt.ylabel('Total Crashes')
plt.title('Box Plot: Speeding vs Total Crashes')
plt.grid(axis='x')
plt.show()
```



```
plt.figure(figsize=(10, 6))
plt.boxplot([df[df['alcohol'] == category]['total'] for category in df['alcohol'].unique()],
            labels=df['alcohol'].unique(), vert=False)
plt.xlabel('Total Crashes')
plt.ylabel('Alcohol')
plt.title('Box Plot: Total Crashes vs Alcohol')
plt.grid(axis='x')
plt.show()
```

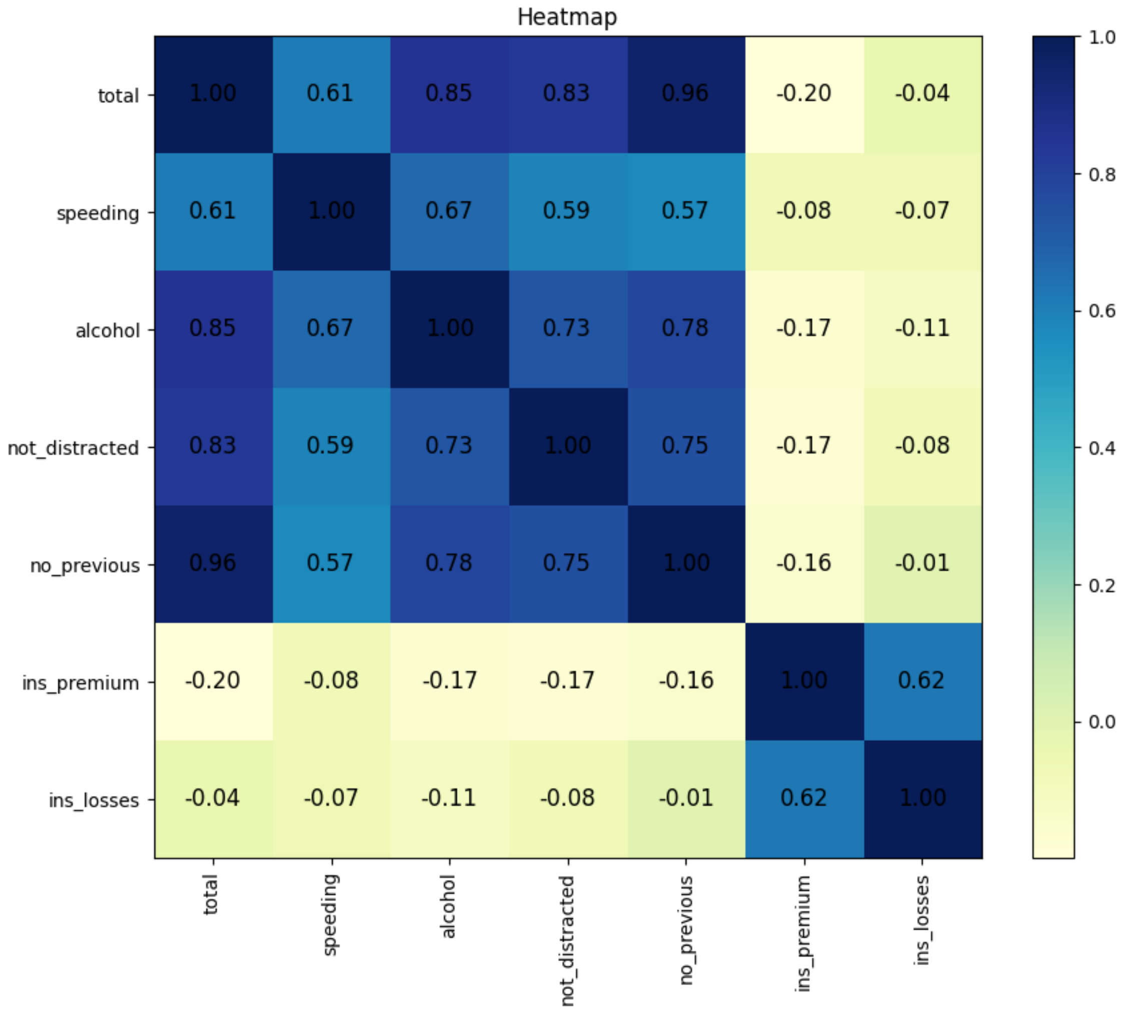


```
# Assuming 'corr' is your correlation matrix
plt.figure(figsize=(10, 8))
heatmap = plt.imshow(corr, cmap='YlGnBu', aspect='auto')

plt.colorbar(heatmap)
plt.title('Heatmap')
plt.xticks(range(len(corr.columns)), corr.columns, rotation=90)
plt.yticks(range(len(corr.columns)), corr.columns)
```

```
# Annotate the heatmap cells with correlation values
for i in range(len(corr.columns)):
    for j in range(len(corr.columns)):
        plt.text(j, i, f'{corr.values[i, j]:.2f}', ha='center', va='center', fontsize=12, color='black')

plt.show()
```

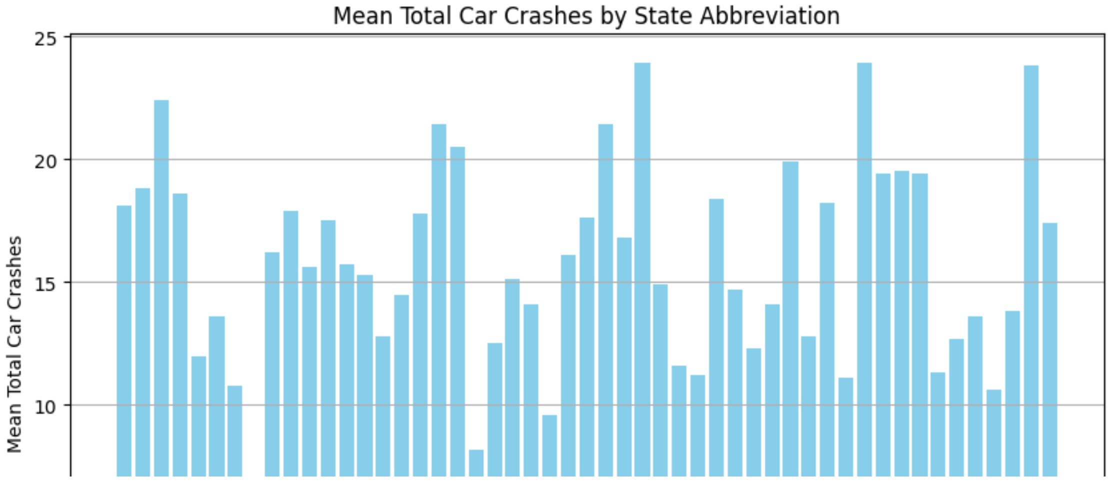


```
import pandas as pd
import matplotlib.pyplot as plt

# Specify the file path in Colab
file_path = '/content/car_crashes.csv'

# Read the CSV file
car_crashes = pd.read_csv(file_path)

plt.figure(figsize=(10, 6))
car_crashes_grouped = car_crashes.groupby('abbrev')['total'].mean().reset_index()
plt.bar(car_crashes_grouped['abbrev'], car_crashes_grouped['total'], color='skyblue')
plt.xlabel('Abbreviation')
plt.ylabel('Mean Total Car Crashes')
plt.title('Mean Total Car Crashes by State Abbreviation')
plt.xticks(rotation=90)
plt.grid(axis='y')
plt.show()
```



```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Specify the file path in Colab
file_path = '/content/car_crashes.csv'

# Read the CSV file
car_crashes = pd.read_csv(file_path)

selected_columns = ['total', 'speeding', 'alcohol', 'not_distracted']
pair_grid = pd.plotting.scatter_matrix(car_crashes[selected_columns], figsize=(12, 8))

# Set individual subplot titles
for ax in pair_grid.ravel():
    ax.set_title('')

# Set a common title for the pair grid plot
plt.suptitle('Pair Grid Plot of Selected Car Crash Variables', y=1.02)
plt.show()
```