ASSIGNMENT 02

NARLA SWAMY PAWAN KOUSHIK

```python
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
```

```python
df = pd.read_csv('/content/test-new.csv')
df
```

	dvcat	weight	dead	airbag	seatbelt	frontal	sex	ageOFocc	yearacc	yearVeh	abcat	occRole	deploy	caseid
0	40-54	18.895	alive	airbag	belted	0	m	25	2002	2000.0	deploy	pass	1	81:99:1
1	25-39	266.532	alive	none	none	1	f	28	2001	1991.0	unavail	driver	0	76:12:2
2	25-39	51.810	alive	airbag	belted	1	f	36	1999	1994.0	deploy	driver	1	78:53:2
3	10-24	1567.626	alive	airbag	belted	0	f	24	2002	1994.0	nodeploy	driver	0	11:47:1
4	25-39	31.342	alive	none	none	1	f	46	1997	1990.0	unavail	pass	0	12:87:2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
8647	10-24	64.757	alive	airbag	none	0	m	37	2001	1999.0	nodeploy	driver	0	6:71:2
8648	1-9km/h	178.651	alive	airbag	none	1	f	16	2001	2001.0	deploy	pass	1	76:57:2
8649	25-39	482.272	alive	airbag	belted	1	m	16	1997	1992.0	deploy	driver	1	74:65:1
8650	55+	32.467	alive	none	belted	1	m	26	1998	1994.0	unavail	driver	0	12:4:2
8651	40-54	10.800	dead	none	none	0	m	73	1998	1989.0	unavail	driver	0	4:32:1

8652 rows × 14 columns

```python
df = pd.read_csv('/content/train-new.csv')
df
```

	dvcat	weight	dead	airbag	seatbelt	frontal	sex	ageOFocc	yearacc	yearVeh	abcat	occRole	deploy	caseid	injSeverity
0	55+	53.342	dead	airbag	belted	1	f	48	2002	1997	deploy	driver	1	45:150:1	3
1	25-39	154.960	alive	none	none	1	m	26	2001	1968	unavail	driver	0	76:40:1	3
2	55+	38.994	alive	none	none	1	f	51	2002	1994	unavail	driver	0	11:184:1	3
3	25-39	168.568	alive	airbag	belted	1	m	27	1998	1996	deploy	pass	1	9:17:1	3
4	10-24	27.751	alive	airbag	belted	0	m	26	2002	1997	nodeploy	pass	0	2:2:2	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
17560	10-24	131.491	alive	airbag	belted	0	m	28	2002	1997	deploy	driver	1	2:106:1	0
17561	10-24	281.693	alive	none	belted	1	f	39	1998	1987	unavail	driver	0	12:118:1	1
17562	10-24	419.418	alive	airbag	belted	1	f	29	1997	1988	deploy	driver	1	9:37:2	3
17563	40-54	9.815	alive	none	none	1	f	22	2000	1990	unavail	driver	0	49:112:2	3
17564	10-24	1395.602	alive	none	belted	1	m	19	2002	1987	unavail	pass	0	43:159:2	0

17565 rows × 15 columns

```python
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

```python
sns.get_dataset_names()
```

```
['anagrams',
 'anscombe',
 'attention',
 'brain_networks',
 'car_crashes',
 'diamonds',
 'dots',
 'dowjones',
 'exercise',
 'flights',
 'fmri',
 'geyser',
 'glue',
 'healthexp',
 'iris',
 'mpg',
 'penguins',
 'planets',
 'seaice',
 'taxis',
 'tips',
 'titanic']
```

```python
Car_crashes = sns.load_dataset("car_crashes")
```

```python
df=sns.load_dataset("car_crashes")
df
```

	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
5	13.6	5.032	3.808	10.744	12.920	835.50	139.91	CO
6	10.8	4.968	3.888	9.396	8.856	1068.73	167.02	CT
7	16.2	6.156	4.860	14.094	16.038	1137.87	151.48	DE
8	5.9	2.006	1.593	5.900	5.900	1273.89	136.05	DC
9	17.9	3.759	5.191	16.468	16.826	1160.13	144.18	FL
10	15.6	2.964	3.900	14.820	14.508	913.15	142.80	GA
11	17.5	9.450	7.175	14.350	15.225	861.18	120.92	HI
12	15.3	5.508	4.437	13.005	14.994	641.96	82.75	ID
13	12.8	4.608	4.352	12.032	12.288	803.11	139.15	IL
14	14.5	3.625	4.205	13.775	13.775	710.46	108.92	IN
15	15.7	2.669	3.925	15.229	13.659	649.06	114.47	IA
16	17.8	4.806	4.272	13.706	15.130	780.45	133.80	KS
17	21.4	4.066	4.922	16.692	16.264	872.51	137.13	KY
18	20.5	7.175	6.765	14.965	20.090	1281.55	194.78	LA
19	15.1	5.738	4.530	13.137	12.684	661.88	96.57	ME
20	12.5	4.250	4.000	8.875	12.375	1048.78	192.70	MD
21	8.2	1.886	2.870	7.134	6.560	1011.14	135.63	MA
22	14.1	3.384	3.948	13.395	10.857	1110.61	152.26	MI
23	9.6	2.208	2.784	8.448	8.448	777.18	133.35	MN
24	17.6	2.640	5.456	1.760	17.600	896.07	155.77	MS
25	16.1	6.923	5.474	14.812	13.524	790.32	144.45	MO
26	21.4	8.346	9.416	17.976	18.190	816.21	85.15	MT
27	14.9	1.937	5.215	13.857	13.410	732.28	114.82	NE
28	14.7	5.439	4.704	13.965	14.553	1029.87	138.71	NV
29	11.6	4.060	3.480	10.092	9.628	746.54	120.21	NH
30	11.2	1.792	3.136	9.632	8.736	1301.52	159.85	NJ
31	18.4	3.496	4.968	12.328	18.032	869.85	120.75	NM
32	12.3	3.936	3.567	10.824	9.840	1234.31	150.01	NY
33	16.8	6.552	5.208	15.792	13.608	708.24	127.82	NC
34	23.9	5.497	10.038	23.661	20.554	688.75	109.72	ND
35	14.1	3.948	4.794	13.959	11.562	697.73	133.52	OH
36	19.9	6.368	5.771	18.308	18.706	881.51	178.86	OK
37	12.8	4.224	3.328	8.576	11.520	804.71	104.61	OR
38	18.2	9.100	5.642	17.472	16.016	905.99	153.86	PA
39	11.1	3.774	4.218	10.212	8.769	1148.99	148.58	RI
40	23.9	9.082	9.799	22.944	19.359	858.97	116.29	SC
41	19.4	6.014	6.402	19.012	16.684	669.31	96.87	SD
42	19.5	4.095	5.655	15.990	15.795	767.91	155.57	TN
43	19.4	7.760	7.372	17.654	16.878	1004.75	156.83	TX
44	11.3	4.859	1.808	9.944	10.848	809.38	109.48	UT
45	13.6	4.080	4.080	13.056	12.920	716.20	109.61	VT
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

Perform data visualization

```
df.head()
```

total speeding alcohol not_distracted no_previous ins_premium ins_losses abbrev

```
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-13-7d5195e2bf4d>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future versio
  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
ins_losses	-0.036011	-0.065928	-0.112547	-0.075970	-0.006359	0.623116	1.000000

```
plt.subplots(figsize=(20,15))
sns.heatmap(corr,annot=True)
```

<Axes: >
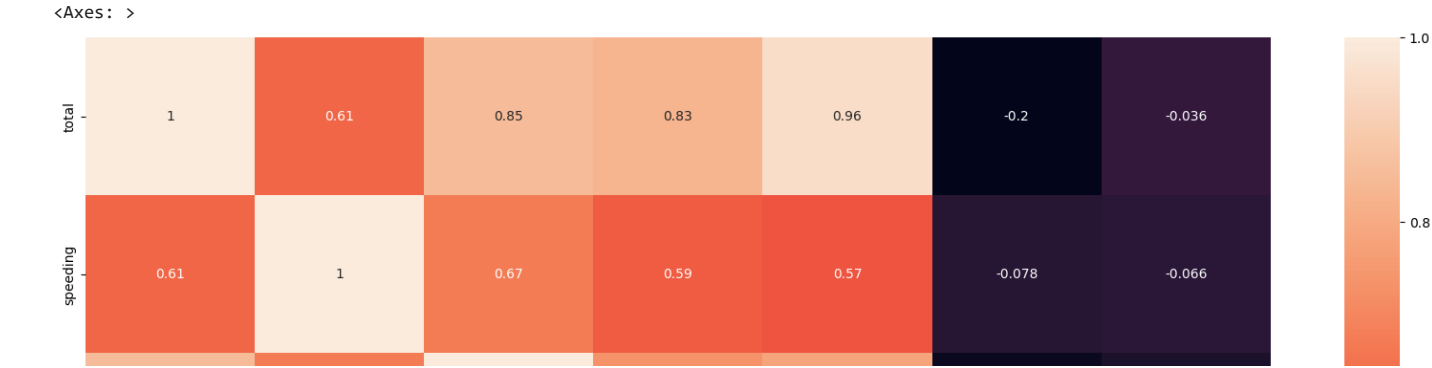

```
sns.scatterplot(x="speeding",y="total",data=df)
```
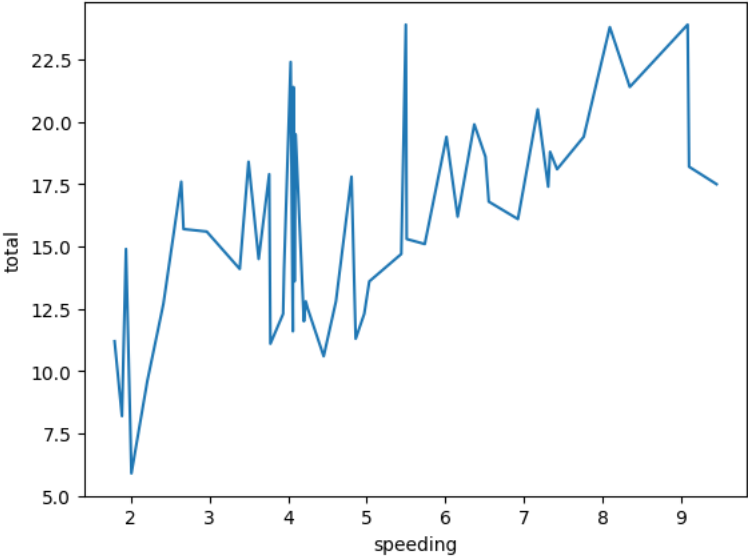
<Axes: xlabel='speeding', ylabel='total'>
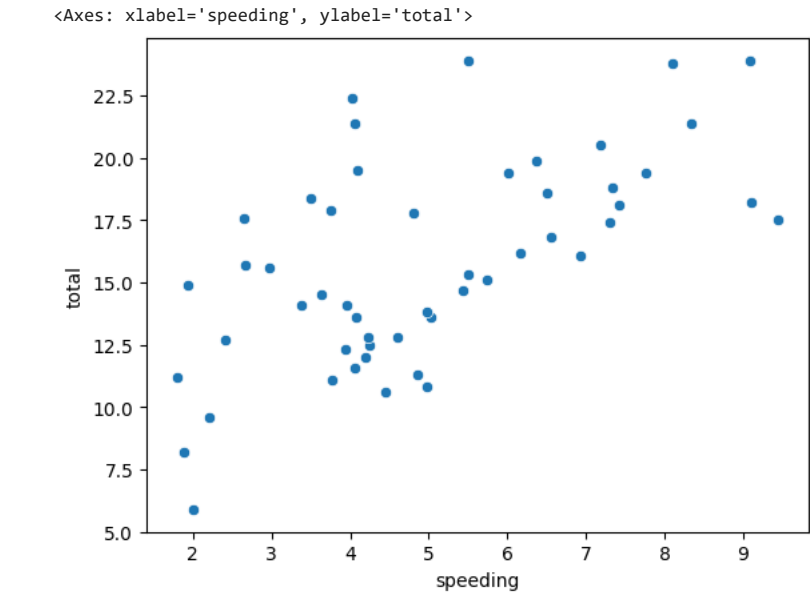

```
sns.scatterplot(x="abbrev",y="total",data=df)
```

<Axes: xlabel='abbrev', ylabel='total'>
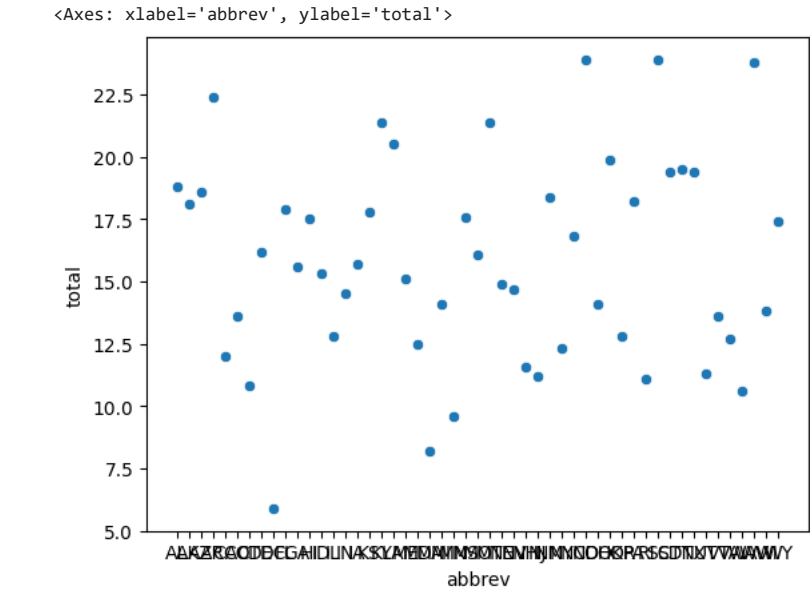

```
sns.lineplot(x="alcohol",y="total",data=df,ci=None)
```

```
<ipython-input-17-10da8a41c1e4>:1: FutureWarning:
```

```python
sns.lineplot(x="speeding",y="total",data=df,errorbar=None)
```
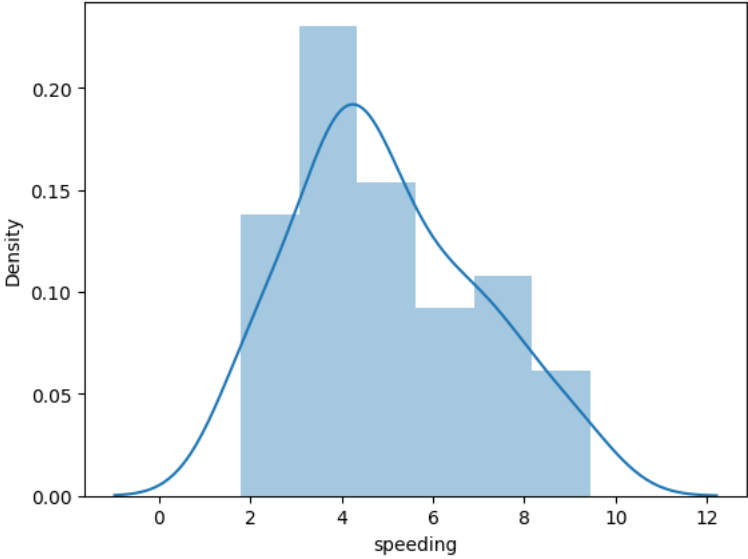
```
<Axes: xlabel='speeding', ylabel='total'>
```



```python
sns.distplot(df["speeding"])
```

```
<ipython-input-19-8ecb7fd34a3c>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751
```

```
  sns.distplot(df["speeding"])
<Axes: xlabel='speeding', ylabel='Density'>
```



```python
sns.relplot(x="total",y="speeding",data=df)
```

```
<seaborn.axisgrid.FacetGrid at 0x79c863e28970>
```
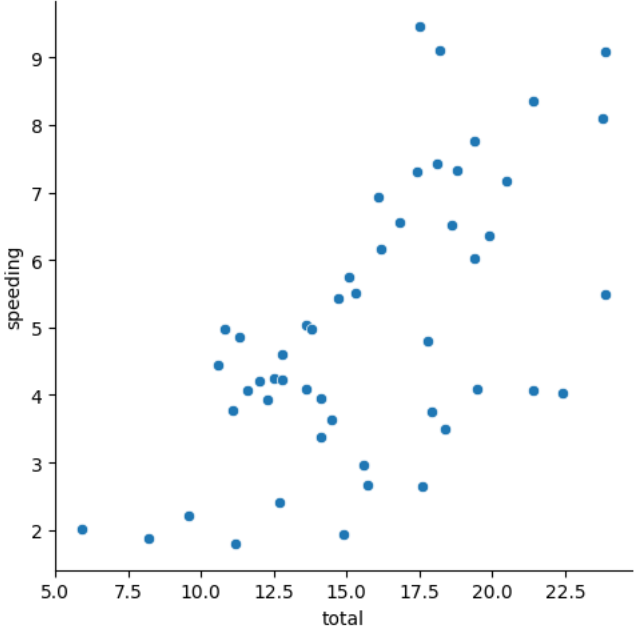


```python
sns.relplot(x="total",y="speeding",data=df,hue="not_distracted")
```

<seaborn.axisgrid.FacetGrid at 0x79c864c857b0>
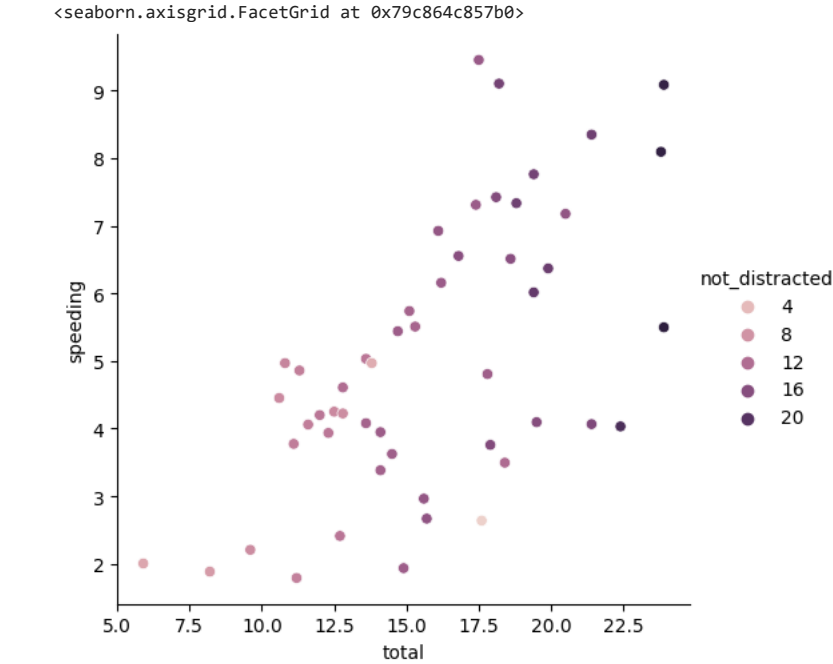


```python
df["ins_losses"].value_counts()
```

```
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
```

```python
sns.barplot(x="speeding",y="total",data=df)
```

```
<Axes: xlabel='speeding', ylabel='total'>
```



```
sns.boxplot(x="speeding",y="total",data=df)
```

```
<Axes: xlabel='speeding', ylabel='total'>
```
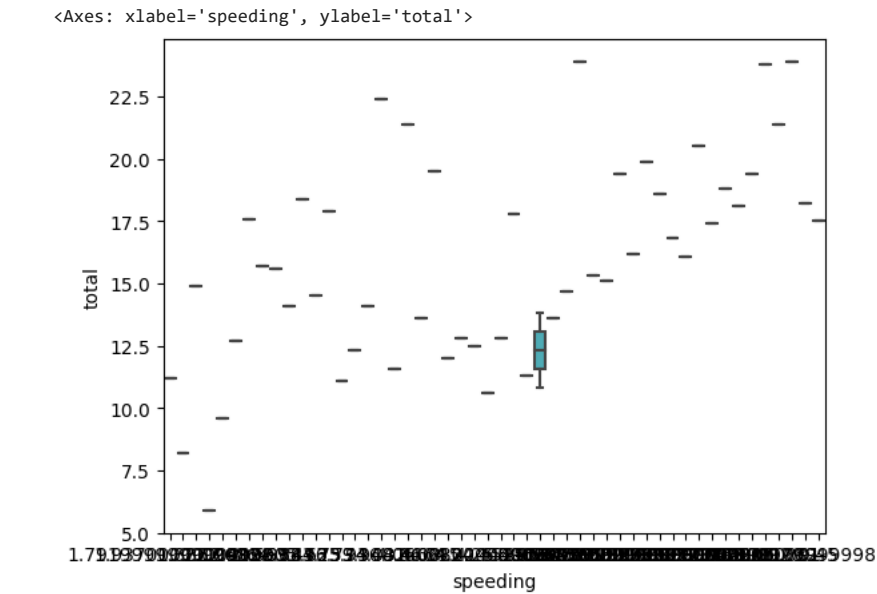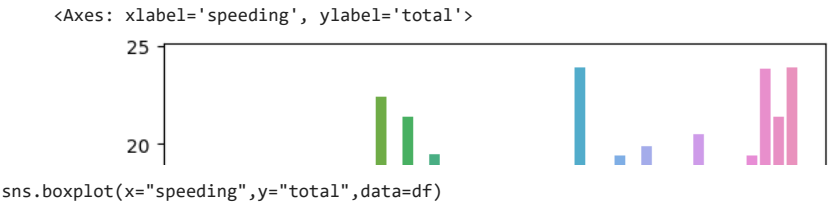


```
sns.heatmap(corr,annot=True,cmap="YlGnBu")
```

```
<Axes: >
```
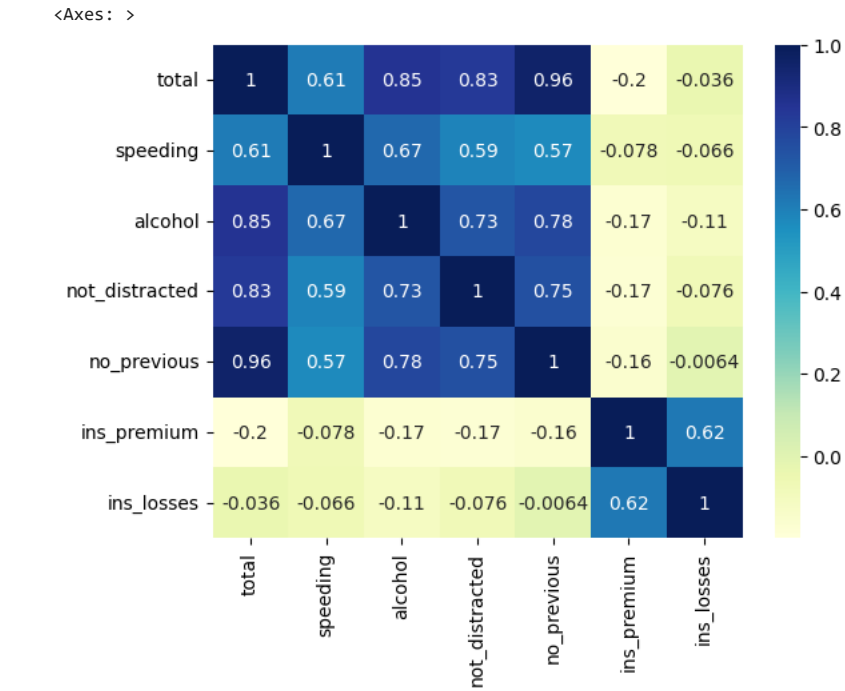


```
car_crashes = sns.load_dataset("car_crashes")
plt.figure(figsize=(10, 6))
sns.barplot(x='abbrev', y='total', data=car_crashes, palette='coolwarm')
plt.xlabel('Abbreviation')
plt.ylabel('Total Car Crashes')
plt.title('Total Car Crashes by State Abbreviation')
plt.xticks(rotation=90)
plt.show()
```

Total Car Crashes by State Abbreviation
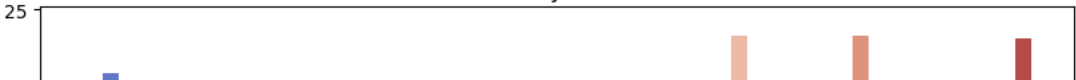
```
selected_columns = ['total', 'speeding', 'alcohol', 'not_distracted']
pair_grid = sns.pairplot(car_crashes[selected_columns])
plt.suptitle('Pair Grid Plot of Selected Car Crash Variables', y=1.02)
plt.show()
```


Pair Grid Plot of Selected Car Crash Variables