

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

```
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']
```

```
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

df.head(5)

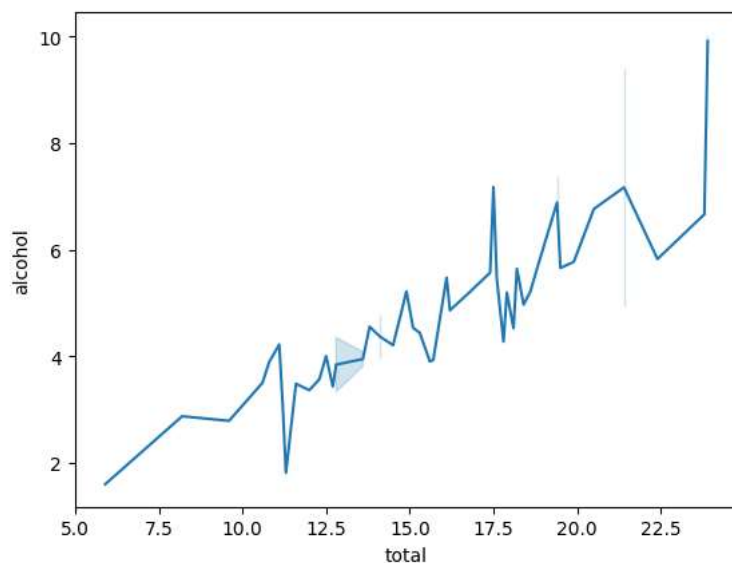
	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev
0	18.8	7.332	5.640	18.048	15.040	784.55	145.08	AL
1	18.1	7.421	4.525	16.290	17.014	1053.48	133.93	AK
2	18.6	6.510	5.208	15.624	17.856	899.47	110.35	AZ
3	22.4	4.032	5.824	21.056	21.280	827.34	142.39	AR
4	12.0	4.200	3.360	10.920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

df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51 entries, 0 to 50
Data 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
```

```
sns.lineplot(x='total',y='alcohol',data=df)
```

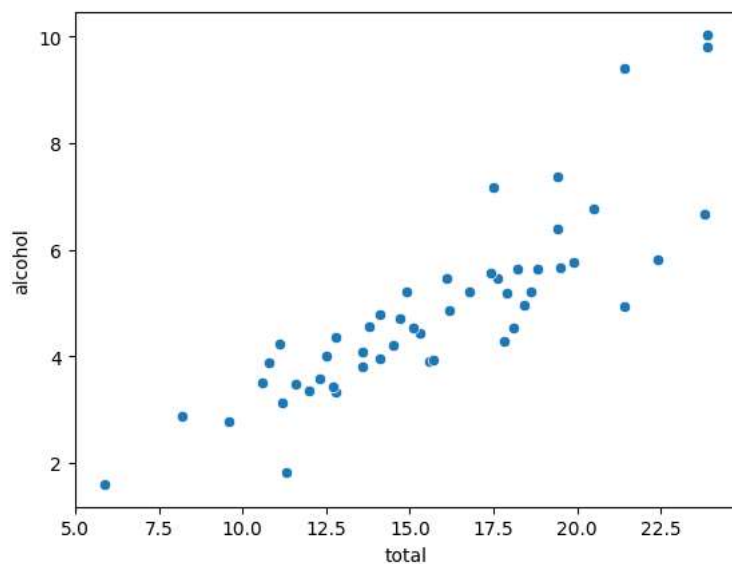
<Axes: xlabel='total', ylabel='alcohol'>



#Inference: By this number of accidents are totally dependent on consumption of alcohol

```
sns.scatterplot(x = 'total',y='alcohol',data=df)
```

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



#Inference: Alcohol consumption has direct impact on number of accidents

```
sns.scatterplot(x='total',y='speeding',data=df)
```

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

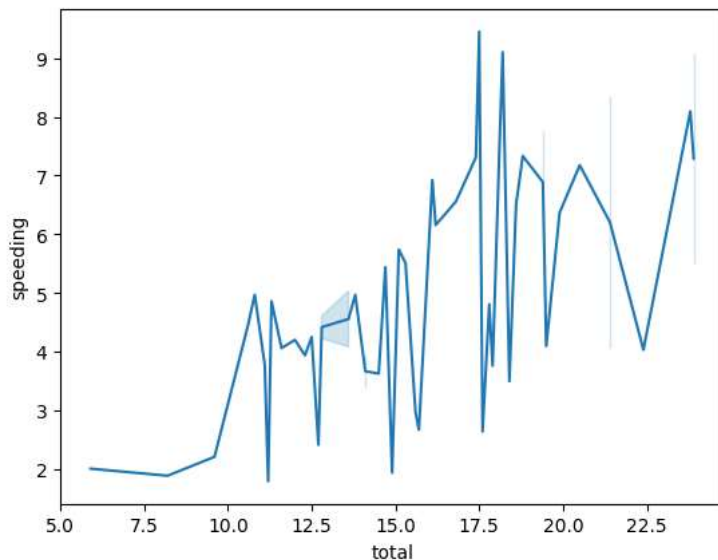


```
#Inference:With increase in speed accidents also increases
```

```
ax |
```

```
sns.lineplot(x='total',y='speeding',data=df)
```

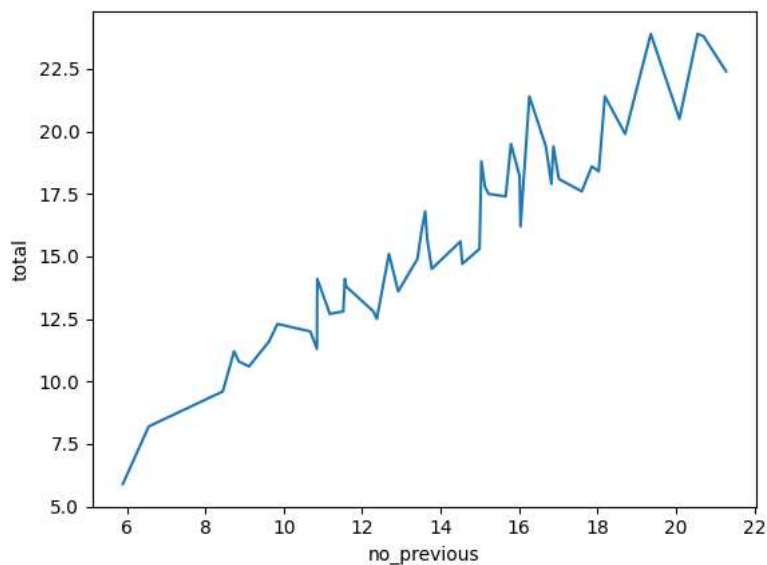
```
<Axes: xlabel='total', ylabel='speeding'>
```



```
#Inferene:If speed increases number of accidents also increases
```

```
sns.lineplot(y='total',x='no_previous',data=df)
```

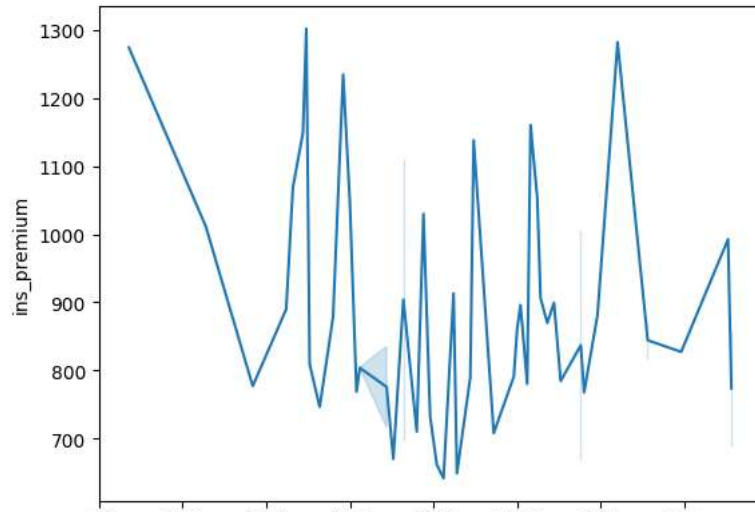
```
<Axes: xlabel='no_previous', ylabel='total'>
```



```
#Inference:Total is directly proportional to the no_previous
```

```
sns.lineplot(x='total',y='ins_premium',data=df)
```

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



```
#Inference:number of accidents has no such impact with ins_premium
```

```
sns.distplot(df['total'])
```

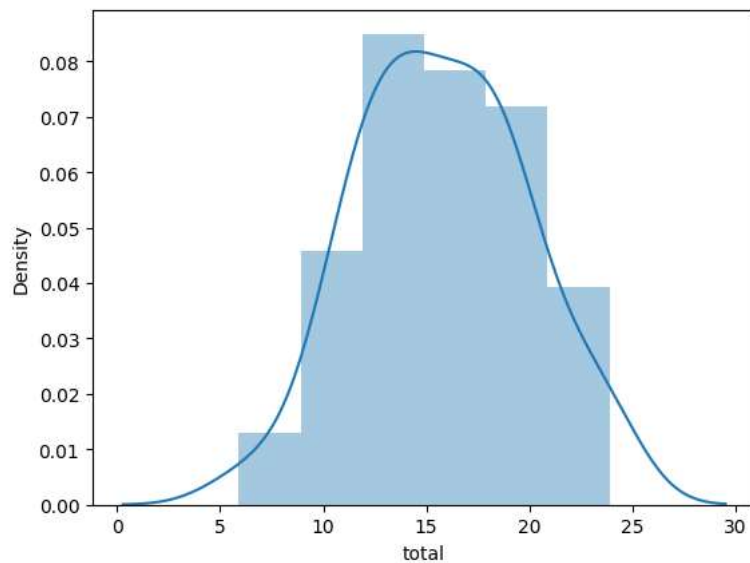
```
<ipython-input-25-2ba73417f012>:1: UserWarning:
```

```
`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['total'])
<Axes: xlabel='total', ylabel='Density'>
```

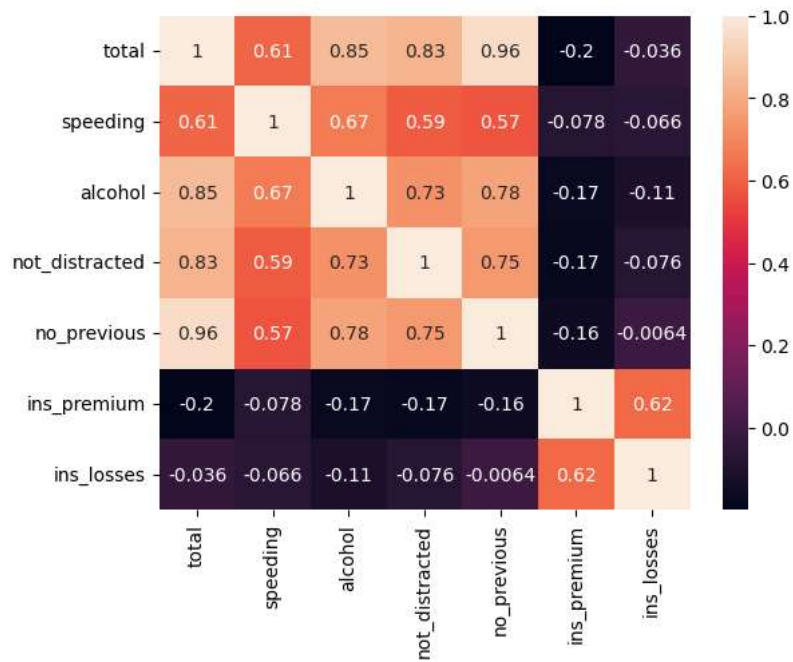


```
corr=df.corr()
```

```
<ipython-input-26-0014364bc22a>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future versior corr=df.corr()
```

```
sns.heatmap(corr,annot=True)
```

<Axes: >



✓ 0s completed at 12:17

