In [1]:
```
import seaborn as sns # importing the seaborn library
import matplotlib.pyplot as plt # importing the matplotlib.pyplot library
```

In [2]:
```
print(sns.get_dataset_names()) # Finding the inbuilt datasets in seaborn
```

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

In [3]:
```
df = sns.load_dataset('car_crashes') # Loading the dataset into variable
df
```

Out[3]:

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abb
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
32	12.3	3.936	3.567	10.824	9.840	1234.31	150.01
33	16.8	6.552	5.208	15.792	13.608	708.24	127.82
34	23.9	5.497	10.038	23.661	20.554	688.75	109.72
35	14.1	3.948	4.794	13.959	11.562	697.73	133.52
36	19.9	6.368	5.771	18.308	18.706	881.51	178.86
37	12.8	4.224	3.328	8.576	11.520	804.71	104.61
38	18.2	9.100	5.642	17.472	16.016	905.99	153.86
39	11.1	3.774	4.218	10.212	8.769	1148.99	148.58
40	23.9	9.082	9.799	22.944	19.359	858.97	116.29
41	19.4	6.014	6.402	19.012	16.684	669.31	96.87
42	19.5	4.095	5.655	15.990	15.795	767.91	155.57
43	19.4	7.760	7.372	17.654	16.878	1004.75	156.83
44	11.3	4.859	1.808	9.944	10.848	809.38	109.48
45	13.6	4.080	4.080	13.056	12.920	716.20	109.61
46	12.7	2.413	3.429	11.049	11.176	768.95	153.72
47	10.6	4.452	3.498	8.692	9.116	890.03	111.62
48	23.8	8.092	6.664	23.086	20.706	992.61	152.56
49	13.8	4.968	4.554	5.382	11.592	670.31	106.62
50	17.4	7.308	5.568	14.094	15.660	791.14	122.04

Handling Null Values

In [4]:
```
df.isnull().any() # No null values, hence no need of data manipulation
```

Out[4]:
```
total             False
speeding          False
alcohol           False
not_distracted    False
no_previous       False
ins_premium       False
ins_losses        False
abbrev            False
dtype: bool
```

Dataset Demographics/Statistics

In [5]:
```
df.describe() # describing about the df, i.e; metadat of columns with cou
```

Out[5]:

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_
count	51.000000	51.000000	51.000000	51.000000	51.000000	51.000000	51.
mean	15.790196	4.998196	4.886784	13.573176	14.004882	886.957647	134.
std	4.122002	2.017747	1.729133	4.508977	3.764672	178.296285	24.
min	5.900000	1.792000	1.593000	1.760000	5.900000	641.960000	82.
25%	12.750000	3.766500	3.894000	10.478000	11.348000	768.430000	114.
50%	15.600000	4.608000	4.554000	13.857000	13.775000	858.970000	136.
75%	18.500000	6.439000	5.604000	16.140000	16.755000	1007.945000	151.
max	23.900000	9.450000	10.038000	23.661000	21.280000	1301.520000	194.

Univariate

Definition: Univariate data analysis focuses on a single variable or dataset, examining its characteristics and distribution.

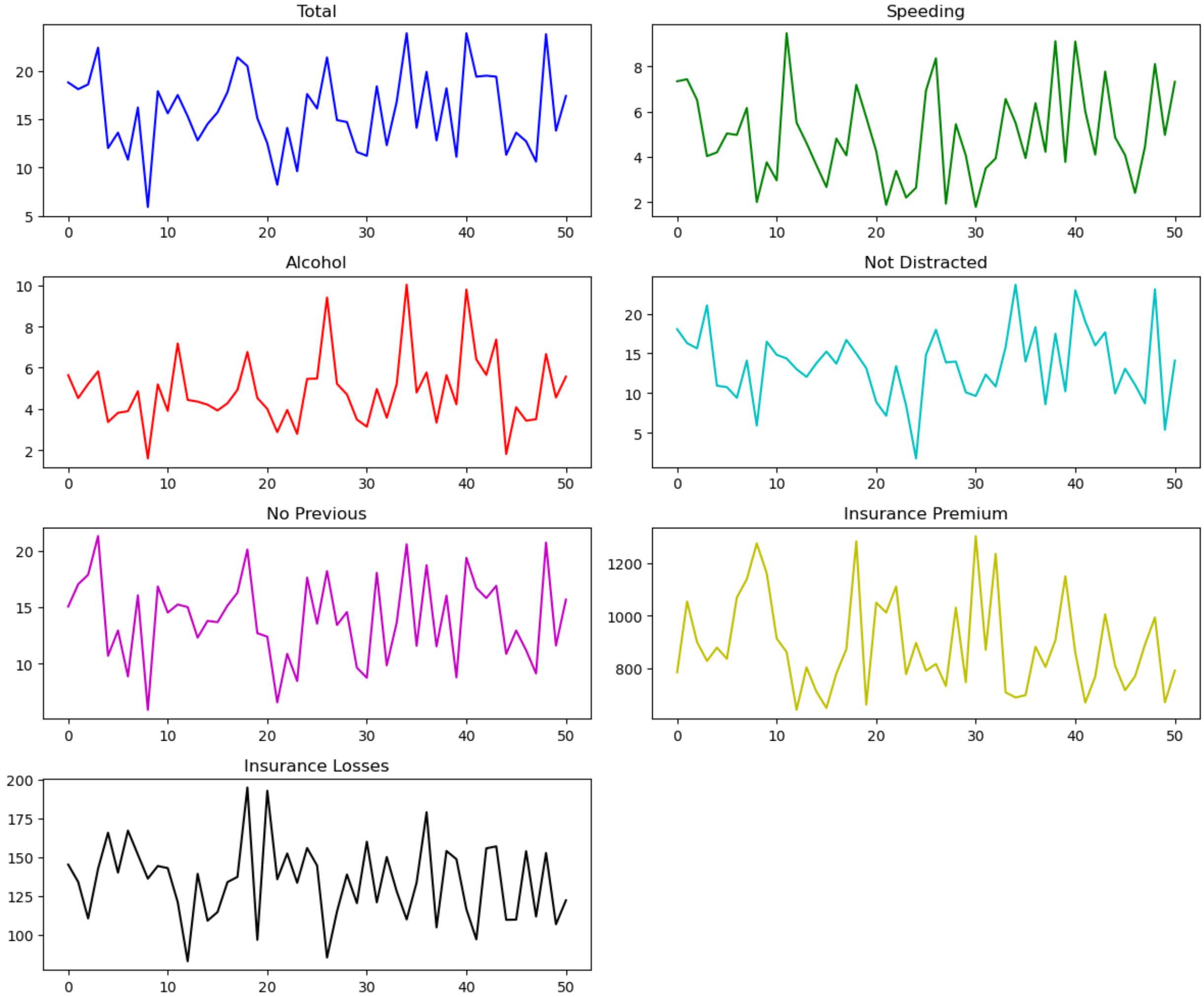
Objective: The primary goal is to describe and summarize the data, understand its central tendency, and identify patterns, outliers, and potential trends within that single variable.

Methods: Common methods include histograms, bar charts, box plots, summary statistics (mean, median, mode), and measures of dispersion (variance, standard deviation)

In [6]:
```
plt.figure(figsize=(12, 10))

plt.subplot(4, 2, 1)
plt.plot(df['total'], 'b')
plt.title('Total')

"""
Total (Blue Line):
The graph shows the trend in total car crashes over the dataset.
Inference: There is a noticeable variation in the total number of car cra
```

```
"""


plt.subplot(4, 2, 2)
plt.plot(df['speeding'], 'g')
plt.title('Speeding')

"""
Speeding (Green Line):
This graph represents the trend in car crashes caused by speeding.
Inference: The number of car crashes due to speeding appears to have some
"""


plt.subplot(4, 2, 3)
plt.plot(df['alcohol'], 'r')
plt.title('Alcohol')

"""
Alcohol (Red Line):
The graph displays the trend in car crashes related to alcohol consumptio
Inference: There is some variation in car crashes involving alcohol, but
"""


plt.subplot(4, 2, 4)
plt.plot(df['not_distracted'], 'c')
plt.title('Not Distracted')

"""
Not Distracted (Cyan Line):
This graph illustrates the trend in car crashes where drivers were not di
Inference: The number of car crashes by non-distracted drivers shows fluc
"""


plt.subplot(4, 2, 5)
plt.plot(df['no_previous'], 'm')
plt.title('No Previous')

"""
No Previous (Magenta Line):
The graph shows the trend in car crashes by drivers with no previous inci
Inference: Car crashes by drivers with no previous incidents appear to ha
"""


plt.subplot(4, 2, 6)
plt.plot(df['ins_premium'], 'y')
plt.title('Insurance Premium')

"""
Insurance Premium (Yellow Line):
This graph represents the trend in insurance premiums.
Inference: The graph doesn't provide clear insights into the trend in ins
"""


plt.subplot(4, 2, 7)
plt.plot(df['ins_losses'], 'k')
plt.title('Insurance Losses')
```

```
"""
Insurance Losses (Black Line):
The graph displays the trend in insurance losses.
Inference: Similar to insurance premiums, insurance losses also appear to
"""

plt.tight_layout() # Used to allocate gaps between the labels and plots
```

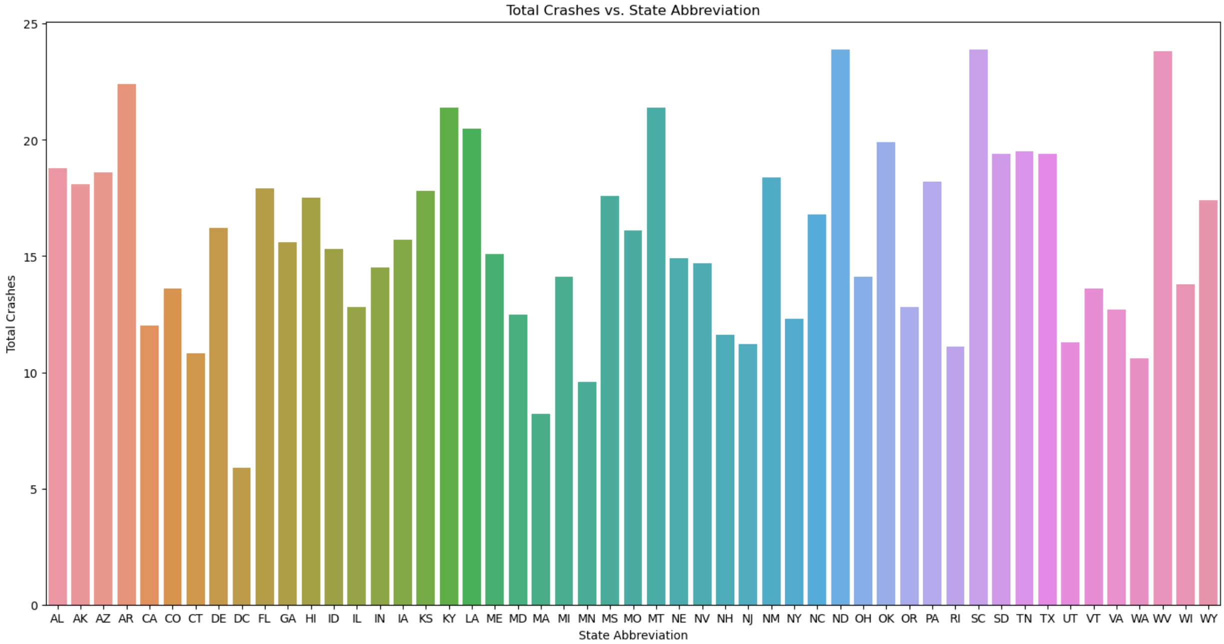


Barplot

In [7]:

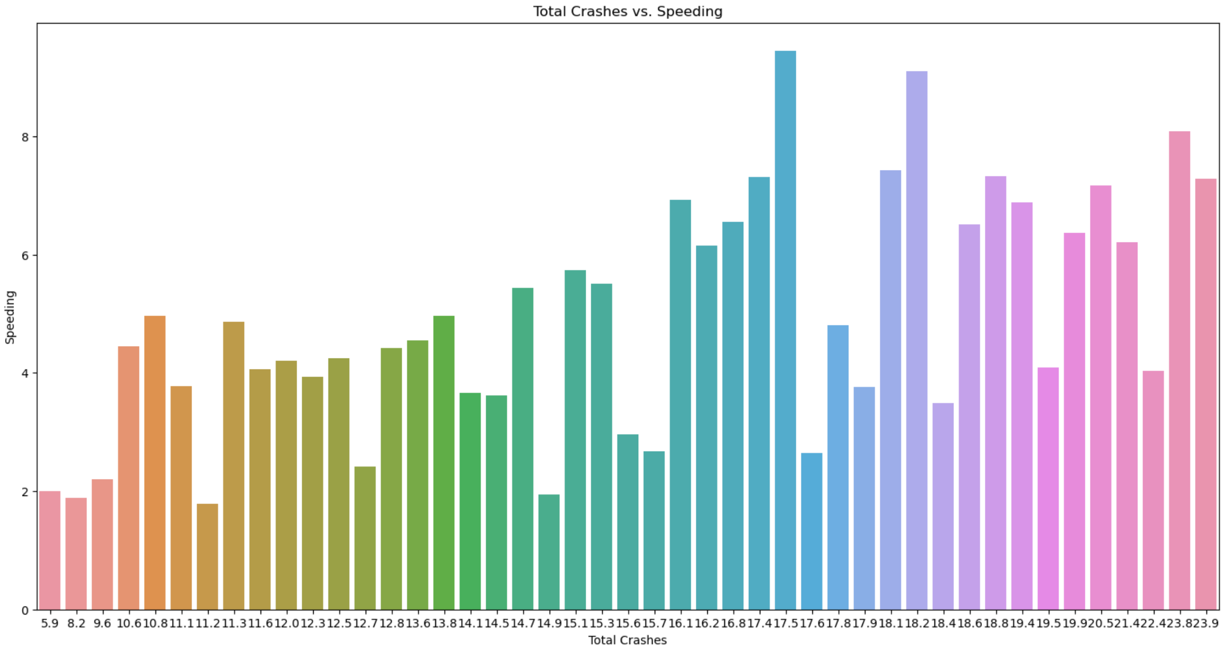
```
plt.figure(figsize=(18, 9))
sns.barplot(data=df,x='abbrev', y='total',errorbar=None)
plt.xlabel('State Abbreviation')
plt.ylabel('Total Crashes')
plt.title('Total Crashes vs. State Abbreviation')

"""
Inference:
State abbreviations are on the x-axis, and the total number of crashes is
The plot provides a clear comparison of car crash counts between states.
For example, states with abbreviations like "DC," "RI," and "NH" have rel
This plot is useful for identifying states with higher or lower crash rat
"""
```

Out[7]:
```
'\nInference:\nState abbreviations are on the x-axis, and the total numbe
r of crashes is on the y-axis.\nThe plot provides a clear comparison of c
ar crash counts between states.\nFor example, states with abbreviations l
ike "DC," "RI," and "NH" have relatively lower total crash counts, while
"TX," "CA," and "FL" have higher crash counts.\nThis plot is useful for i
dentifying states with higher or lower crash rates, which can be valuable
for further analysis or policy considerations.\n'
```



In [8]:
```
plt.figure(figsize=(18, 9))
sns.barplot(data=df,x='total', y='speeding',errorbar=None)
plt.ylabel('Speeding')
plt.xlabel('Total Crashes')
plt.title('Total Crashes vs. Speeding')

"""
Inference:
The total number of crashes is represented on the x-axis, while the numbe
The plot allows us to examine how speeding contributes to the overall num
As the total number of crashes increases, there is a general trend of an
This suggests that as the total number of car crashes goes up, the propor
Analyzing this relationship can help in understanding the impact of speed
"""
```

Out[8]:
```
'\nInference:\nThe total number of crashes is represented on the x-axis,
while the number of crashes involving speeding is on the y-axis.\nThe plo
t allows us to examine how speeding contributes to the overall number of
car crashes.\nAs the total number of crashes increases, there is a genera
l trend of an increase in the number of crashes involving speeding.\nThis
suggests that as the total number of car crashes goes up, the proportion
of crashes involving speeding also tends to increase.\nAnalyzing this rel
ationship can help in understanding the impact of speeding on overall roa
d safety and may inform targeted interventions to reduce speeding-related
accidents.\n'
```



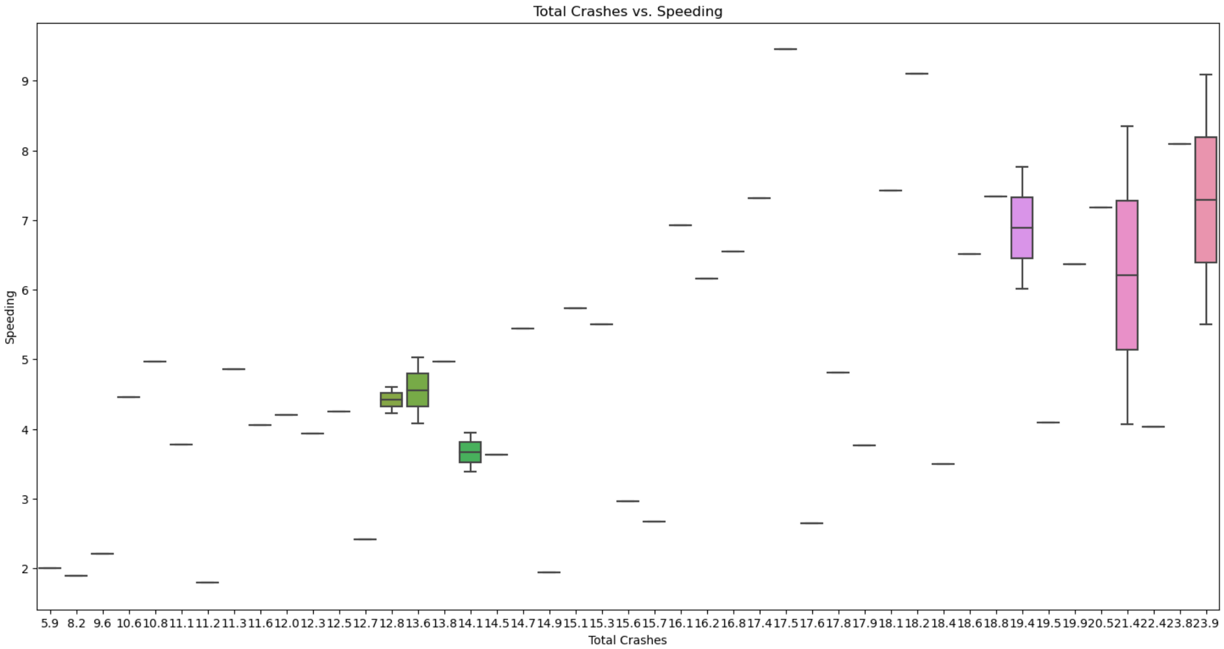
Boxplot

In [9]:

```
plt.figure(figsize=(18,9))
sns.boxplot(x="total",y="speeding",data=df)
plt.ylabel('Speeding')
plt.xlabel('Total Crashes')
plt.title('Total Crashes vs. Speeding')

"""
Inference :
The box plot shows the distribution of speeding-related crashes within di
As the total number of crashes increases, there is increasing variability
This highlights the relationship between total crashes and speeding incid
"""
```

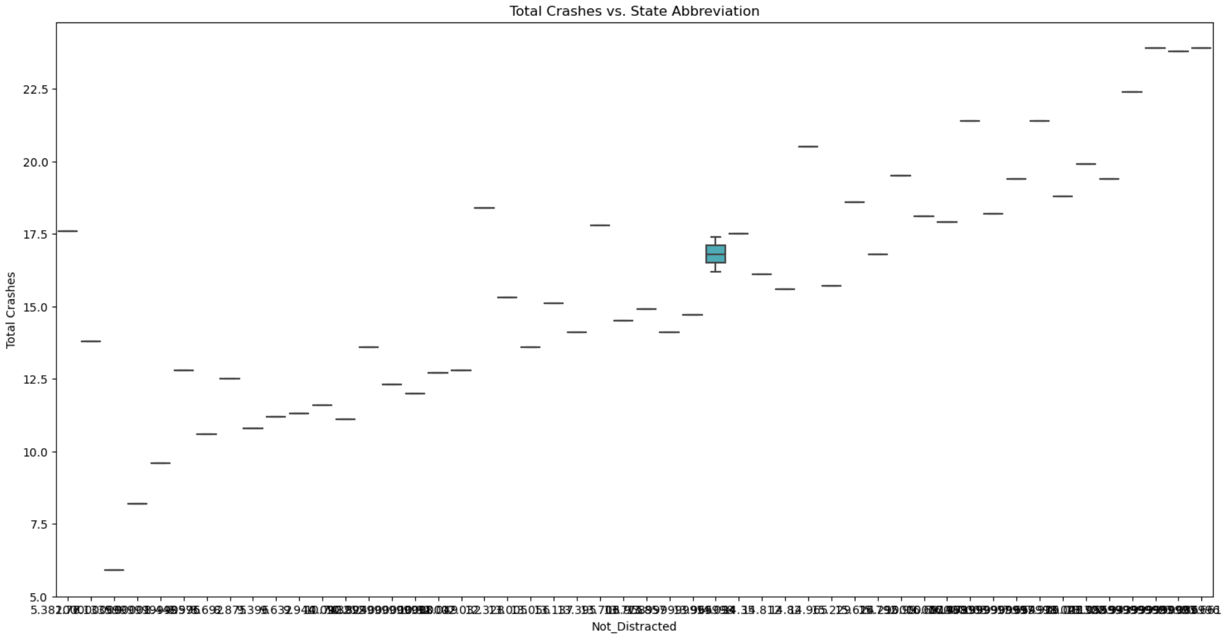
Out[9]:

```
'\nInference :\nThe box plot shows the distribution of speeding-related c
rashes within different total crash categories.\nAs the total number of c
rashes increases, there is increasing variability in the number of crashe
s involving speeding.\nThis highlights the relationship between total cra
shes and speeding incidents, indicating the need for targeted interventio
ns in states or situations with higher variability.\n'
```



In [10]:
```
plt.figure(figsize=(18,9))
sns.boxplot(x="not_distracted",y="total",data=df)
plt.xlabel('Not_Distracted')
plt.ylabel('Total Crashes')
plt.title('Total Crashes vs. State Abbreviation')

"""
Inference :
The box plot illustrates the distribution of total crashes concerning the
It provides insights into how distraction affects the total number of car
The plot shows varying total crash counts based on the distraction status
This suggests that non-distracted drivers may be involved in more crashes
"""
```

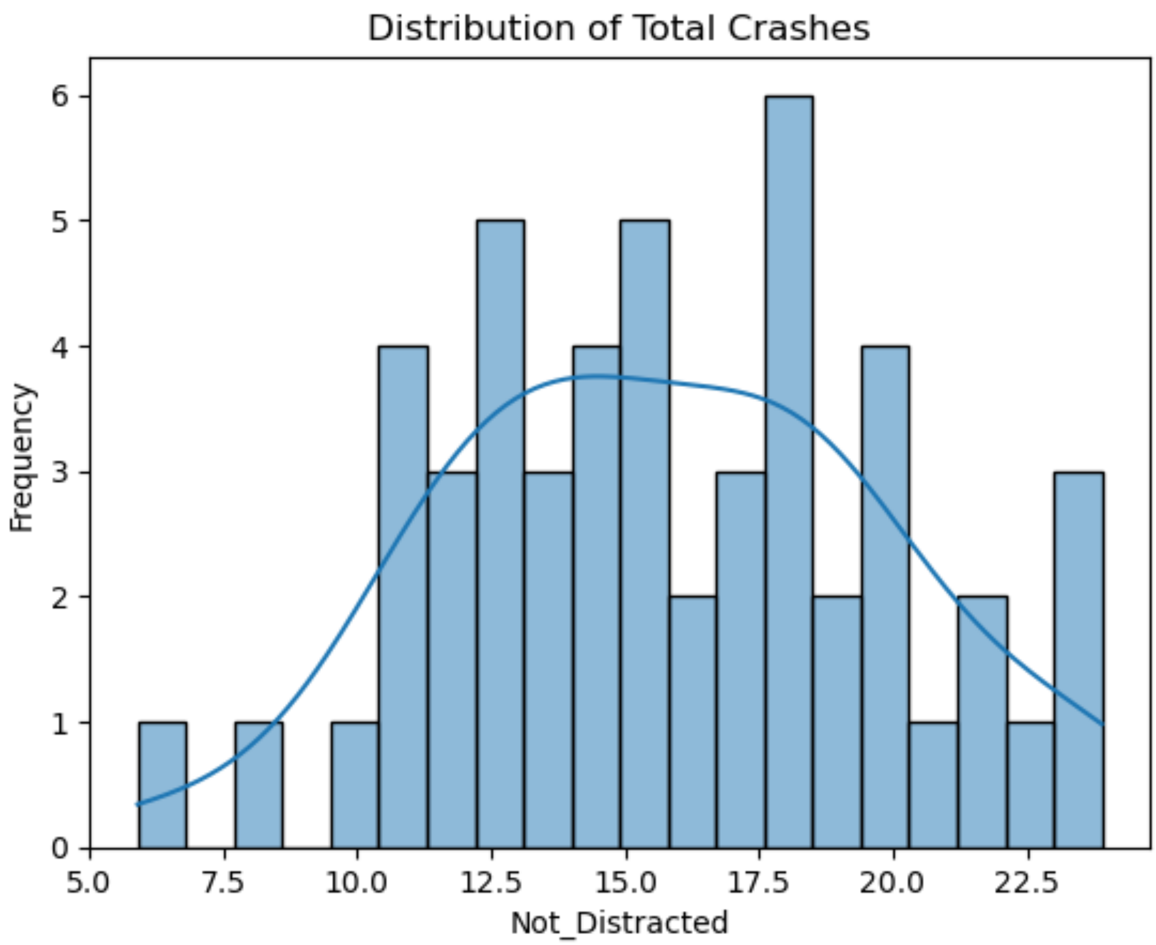
Out[10]:
```
'\nInference :\nThe box plot illustrates the distribution of total crashe
s concerning the distraction status of drivers (Not Distracted).\nIt prov
ides insights into how distraction affects the total number of car crashe
s.\nThe plot shows varying total crash counts based on the distraction st
atus, with potentially higher crashes when drivers are not distracted.\nT
his suggests that non-distracted drivers may be involved in more crashes,
emphasizing the need for examining the causes of distraction and driving
behavior to improve road safety.\n'
```



Histogram

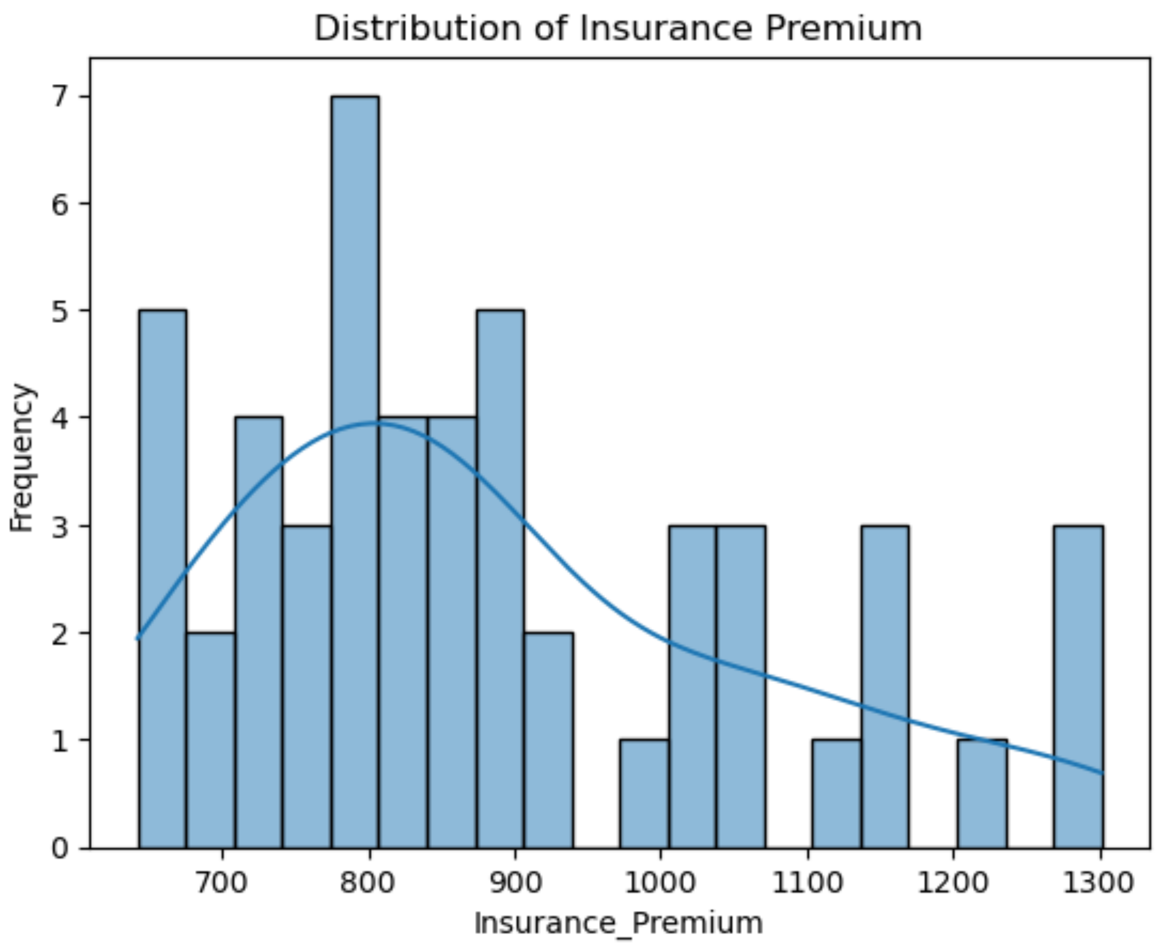
In [11]:
```
sns.histplot(data=df, x='total', bins=20, kde=True)
plt.xlabel('Not_Distracted')
plt.ylabel('Frequency')
plt.title('Distribution of Total Crashes')

"""
Inference :
The histogram displays the distribution of total car crashes.n.
The plot shows that the majority of observations fall within a relatively
There is a right-skewed distribution, indicating that a few instances hav
This visualization helps understand the distribution of total crashes, wh
"""
```

Out[11]:
```
'\nInference :\nThe histogram displays the distribution of total car cras
hes.n.\nThe plot shows that the majority of observations fall within a re
latively low range of total crashes, with a peak in frequency.\nThere is
a right-skewed distribution, indicating that a few instances have signifi
cantly higher crash counts.\nThis visualization helps understand the dist
ribution of total crashes, which can be useful for identifying common cra
sh count ranges and outliers in the dataset.\n'
```



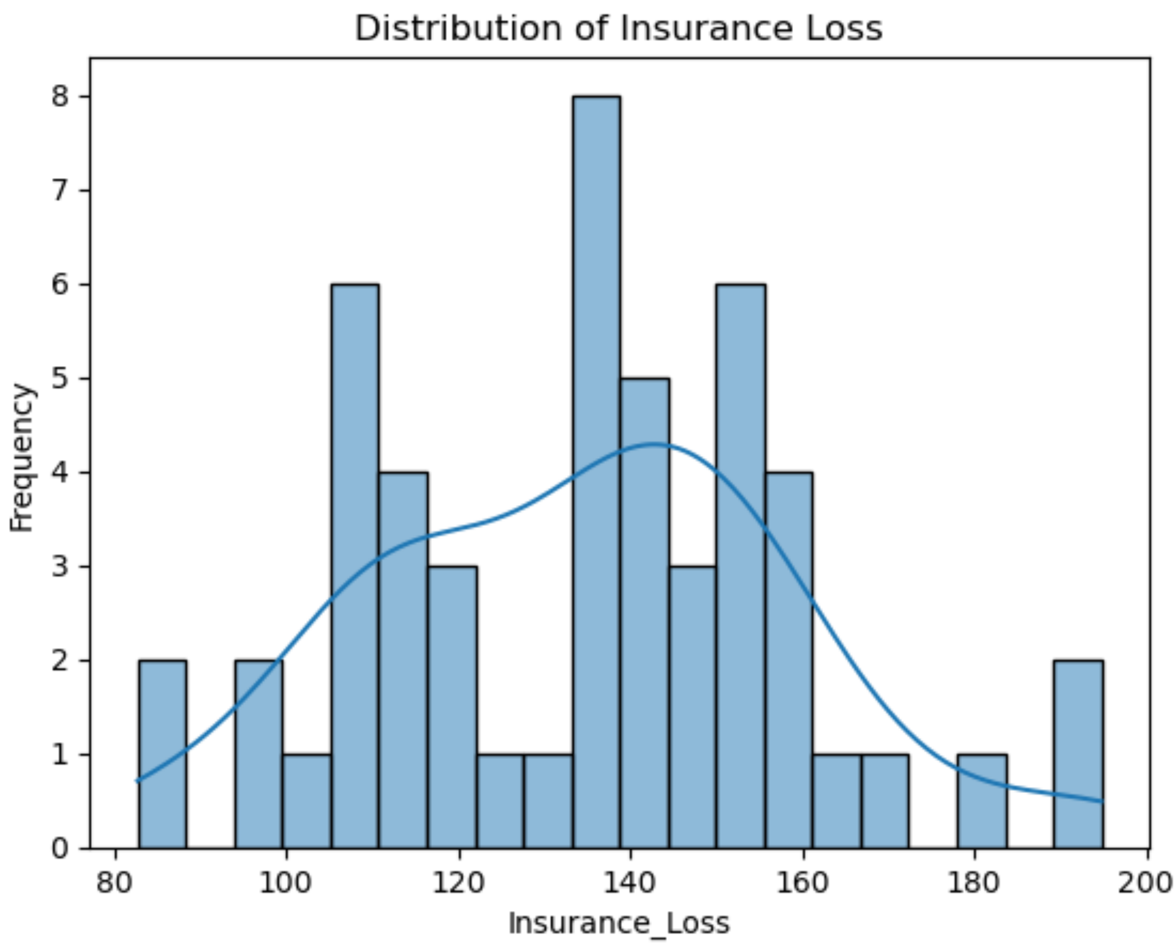
In [12]:
```
sns.histplot(data=df, x='ins_premium', bins=20, kde=True)
plt.xlabel('Insurance_Premium')
plt.ylabel('Frequency')
plt.title('Distribution of Insurance Premium')

"""
Inference :
The histogram depicts the distribution of insurance premiums.
The plot shows that the most common insurance premium ranges have higher
The distribution appears to be right-skewed, suggesting that a few observ
This visualization aids in understanding the distribution of insurance pr
"""
```

Out[12]:
```
'\nInference :\nThe histogram depicts the distribution of insurance premi
ums.\nThe plot shows that the most common insurance premium ranges have h
igher frequencies, forming peaks in the distribution.\nThe distribution a
ppears to be right-skewed, suggesting that a few observations have except
ionally high insurance premiums.\nThis visualization aids in understandin
g the distribution of insurance premiums within the dataset, providing in
sights into common premium ranges and potential outliers.\n'
```



In [13]:
```
sns.histplot(data=df, x='ins_losses', bins=20, kde=True)
plt.xlabel('Insurance_Loss')
plt.ylabel('Frequency')
plt.title('Distribution of Insurance Loss')

"""
Inference :
The histogram represents the distribution of insurance losses.
The plot indicates that the majority of insurance losses fall within spec
The distribution appears right-skewed, indicating that a few instances ha
This visualization helps in understanding the distribution of insurance l
"""
```

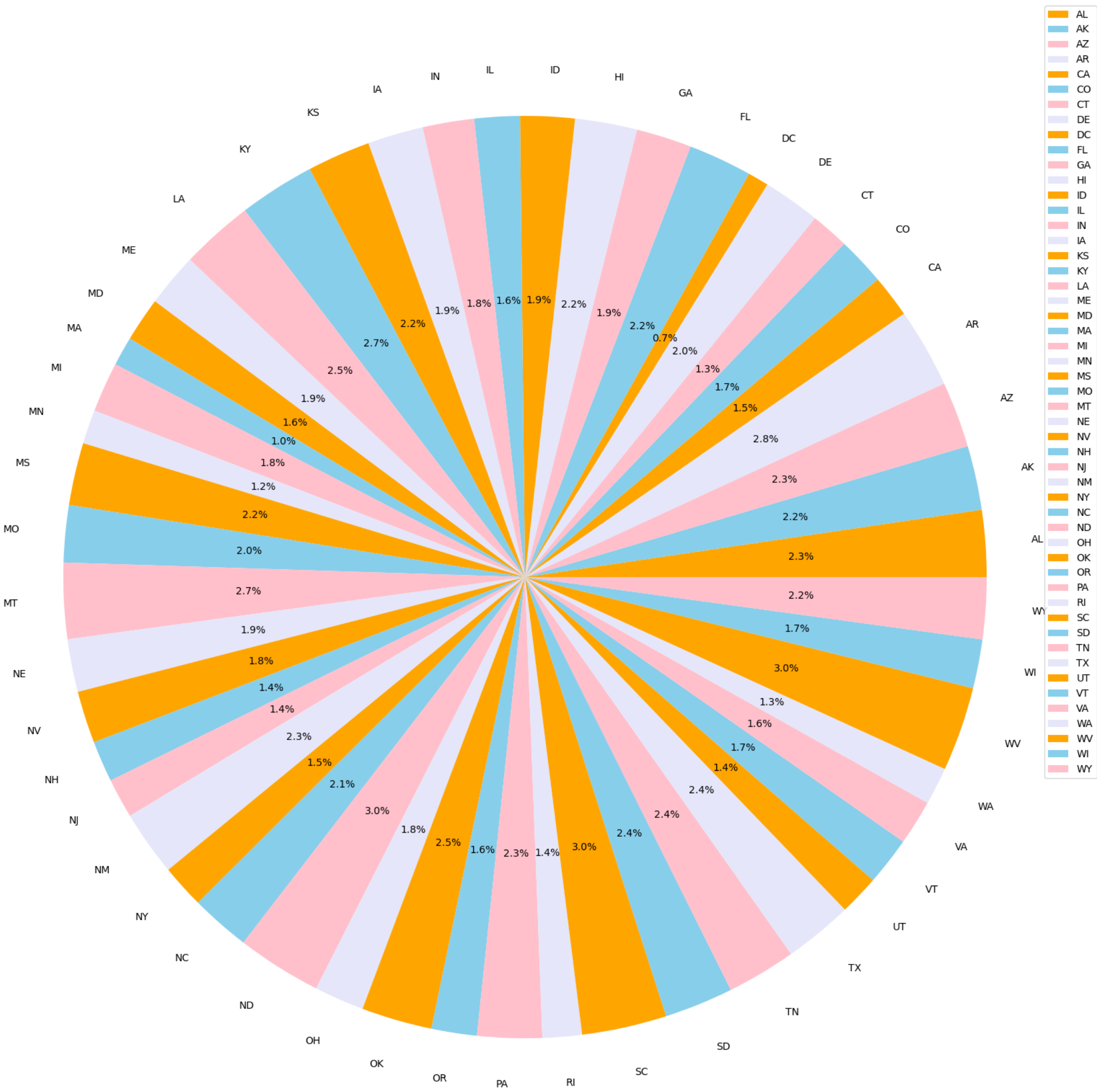
Out[13]:
```
'\nInference :\nThe histogram represents the distribution of insurance lo
sses.\nThe plot indicates that the majority of insurance losses fall with
in specific ranges, with peaks in frequency.\nThe distribution appears ri
ght-skewed, indicating that a few instances have considerably higher insu
rance losses.\nThis visualization helps in understanding the distribution
of insurance losses within the dataset, highlighting common loss ranges a
nd potential outliers.\n'
```



Piechart

In [14]:
```
fig = plt.figure(figsize=(20,20))
axes1 = fig.add_axes([0.1,0.1,0.8,0.8])  # (left,bottom,width,height)
axes1.pie(df['total'],labels=df['abbrev'],autopct='%0.1f%%',colors =['ora
axes1.legend()

"""
Inference :
The pie chart visualizes the distribution of total car crashes across dif
Each slice of the pie represents a state, and the size of the slice corre
The labels on the chart indicate the state abbreviations.
The legend provides a key to identify which state each slice represents.
This pie chart allows for a quick comparison of the contribution of each
"""
```

Out[14]:
```
'\nInference :\nThe pie chart visualizes the distribution of total car cr
ashes across different states, represented by their abbreviations.\nEach
slice of the pie represents a state, and the size of the slice correspond
s to the percentage of total crashes in that state.\nThe labels on the ch
art indicate the state abbreviations.\nThe legend provides a key to ident
ify which state each slice represents.\nThis pie chart allows for a quick
comparison of the contribution of each state to the total number of car c
rashes in the datase\n'
```



Bivariate

Definition: Bivariate data analysis involves the analysis of two variables to explore their relationship and interactions.

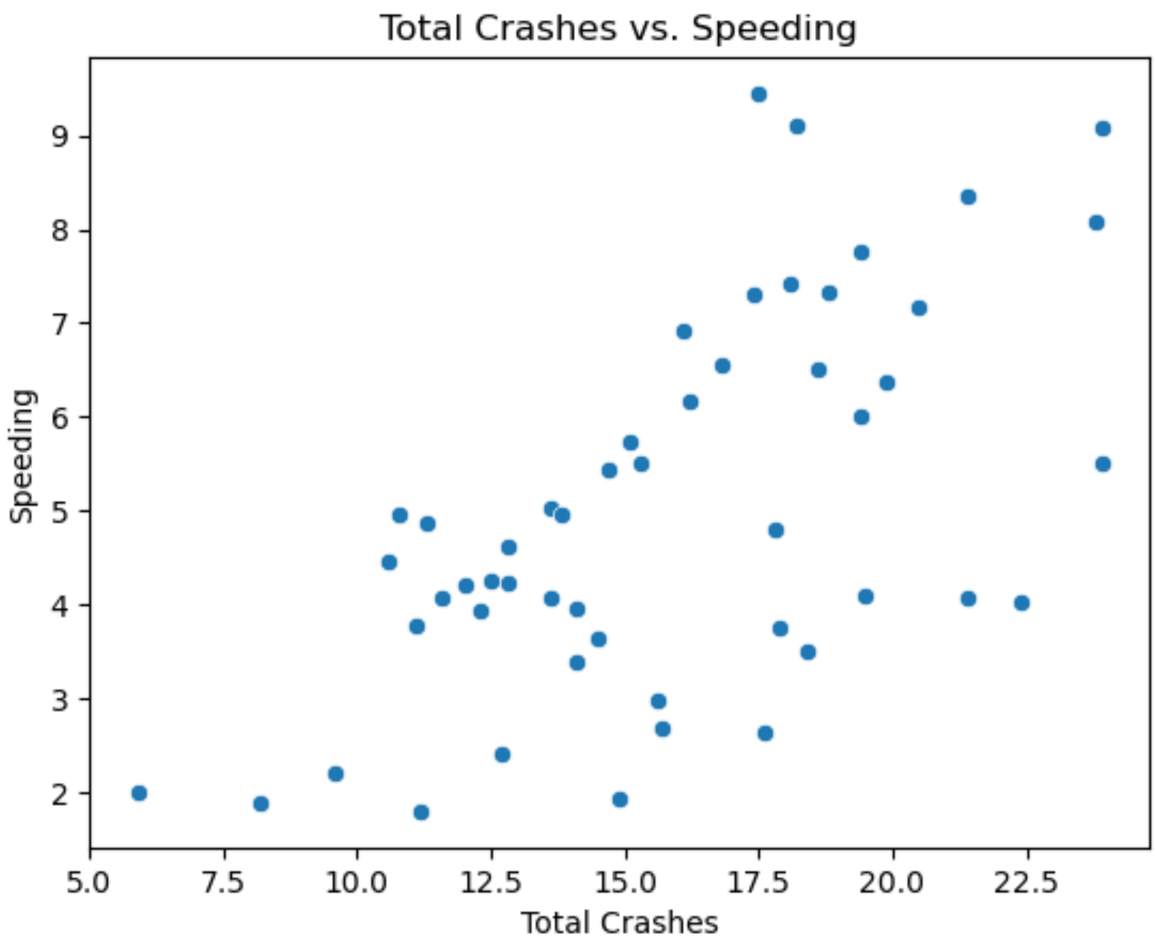
Objective: The primary goal is to understand how two variables are related, whether they exhibit correlation or causation, and to identify patterns or associations between them.

Methods: Common methods include scatter plots, line graphs, correlation coefficients (e.g., Pearson correlation), and hypothesis tests (e.g., t-tests) to determine if relationships are statistically significant.

Scatterplot

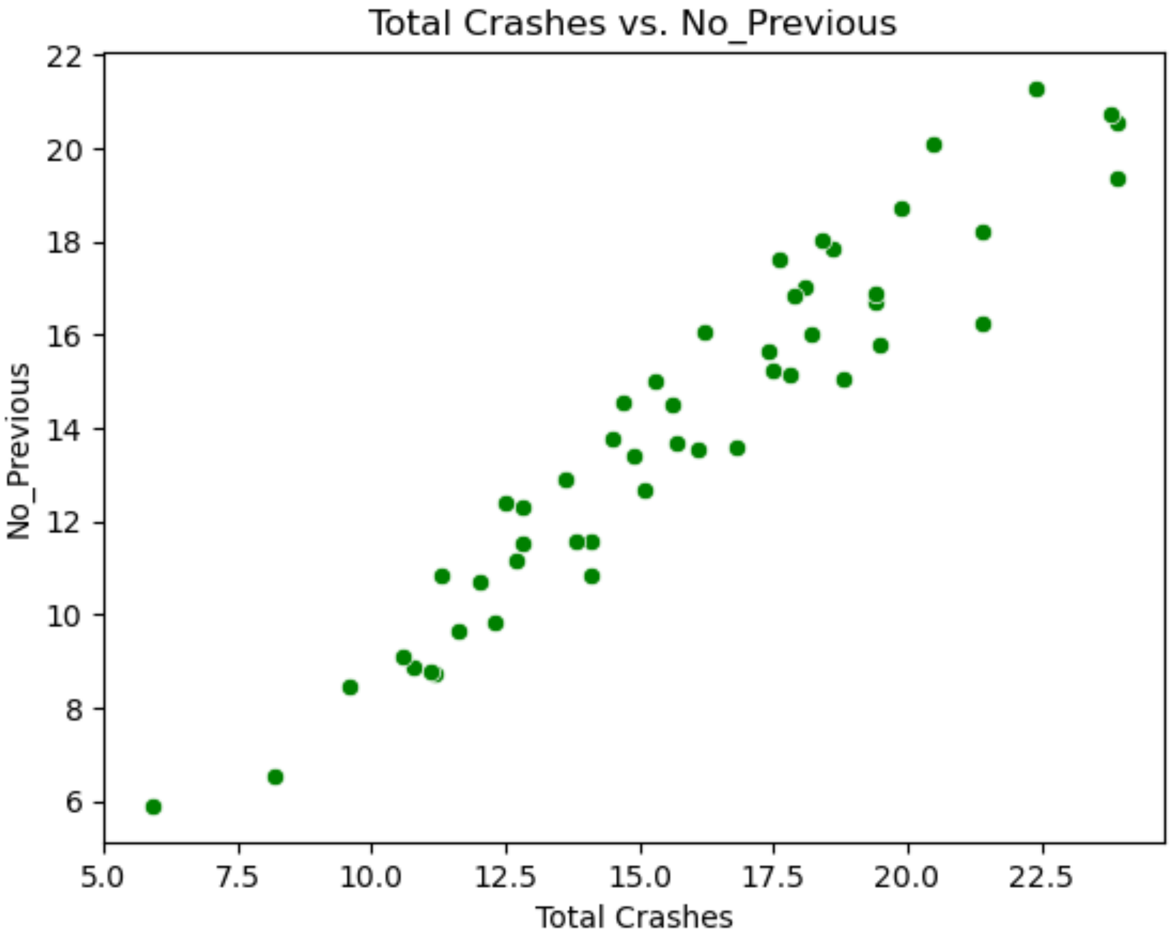
In [15]:
```
sns.scatterplot(x="total",y='speeding',data=df)
plt.ylabel('Speeding')
plt.xlabel('Total Crashes')
plt.title('Total Crashes vs. Speeding')

"""
Inference :
The scatter plot visualizes the relationship between the total number of
There doesn't appear to be a strong linear relationship between total cra
The points are scattered across the plot without a clear trend, suggestin
Further statistical analysis may be needed to quantify the relationship b
"""
```

Out[15]:
```
"\nInference :\nThe scatter plot visualizes the relationship between the
total number of car crashes and the number of crashes involving speeding.
\nThere doesn't appear to be a strong linear relationship between total c
rashes and speeding incidents based on this scatter plot.\nThe points are
scattered across the plot without a clear trend, suggesting that total cr
ashes and speeding may not be strongly correlated.\nFurther statistical a
nalysis may be needed to quantify the relationship between these variable
s accurately.\n"
```



In [16]:
```
sns.scatterplot(x="total",y='no_previous',data=df,c='g')
plt.ylabel('No_Previous')
plt.xlabel('Total Crashes')
plt.title('Total Crashes vs. No_Previous')

"""
Inference :
The scatter plot illustrates the relationship between the total number of
Similar to previous scatter plots, there isn't a distinct linear relation
The points are scattered without a clear trend, suggesting that total cra
"""
```

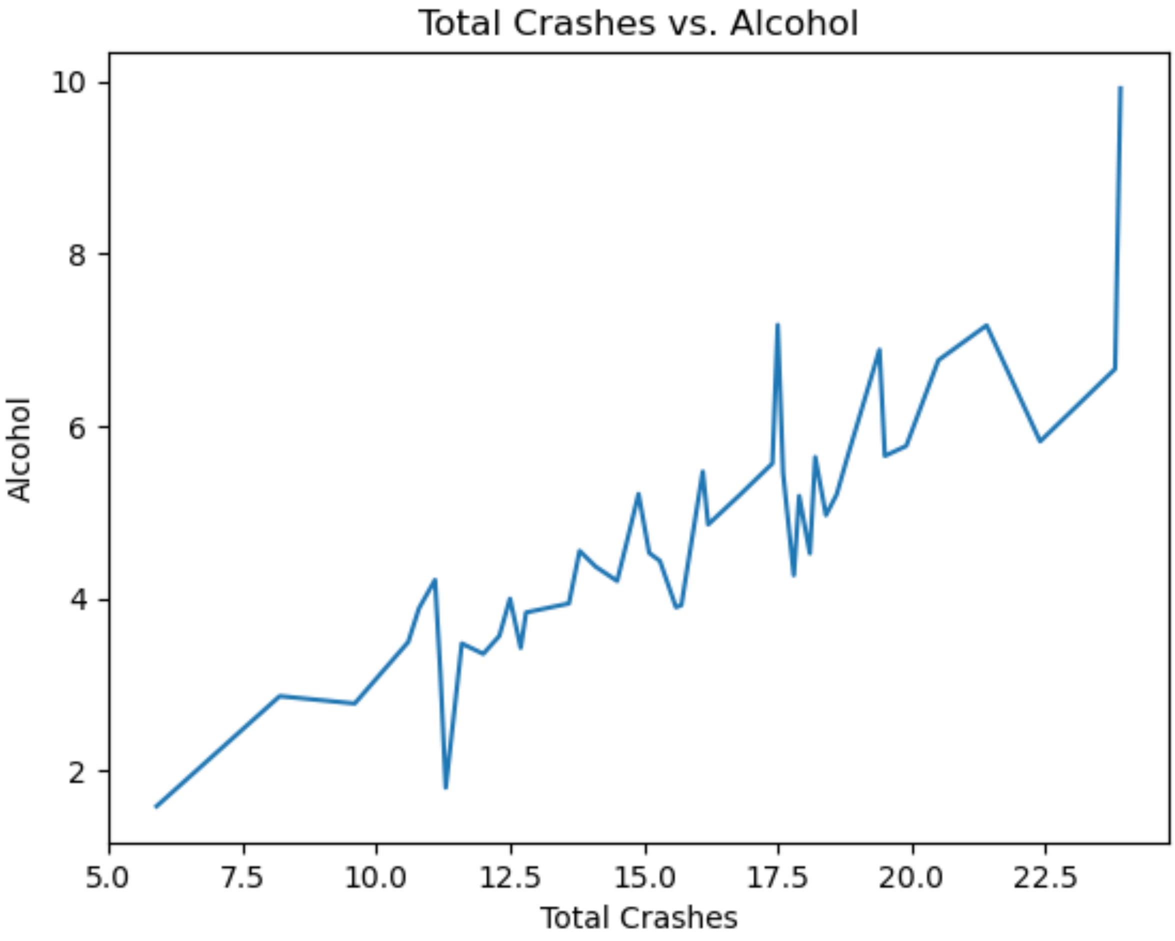
Out[16]:
```
"\nInference :\nThe scatter plot illustrates the relationship between the
total number of car crashes and crashes involving drivers with no previou
s incidents.\nSimilar to previous scatter plots, there isn't a distinct l
inear relationship between total crashes and crashes involving drivers wi
th no previous incidents.\nThe points are scattered without a clear trend
, suggesting that total crashes may not directly correlate with the absen
ce of previous incidents in drivers. Further analysis may be needed.\n"
```



Lineplot

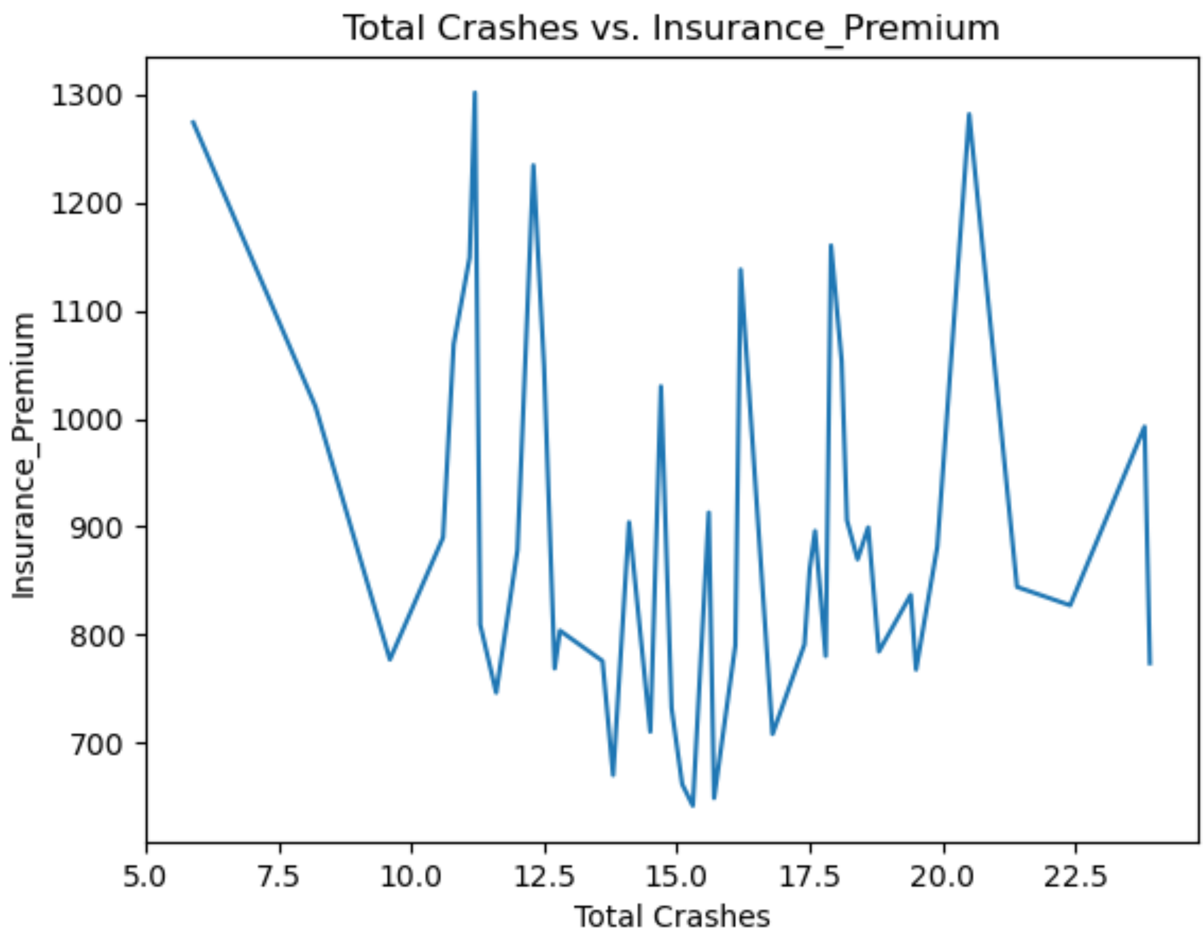
In [17]:
```
sns.lineplot(x="total",y="alcohol",data=df,errorbar=None)
plt.ylabel('Alcohol')
plt.xlabel('Total Crashes')
plt.title('Total Crashes vs. Alcohol')
"""
Inference :
The line plot shows the association between total car crashes and crashes
It visualizes how alcohol-related crashes fluctuate concerning the total
There isn't a clear linear relationship; the points on the line are scatt
This suggests that the total number of crashes may not have a straightfor
"""
```

Out[17]:
```
"\nInference :\nThe line plot shows the association between total car cra
shes and crashes involving alcohol.\nIt visualizes how alcohol-related cr
ashes fluctuate concerning the total number of crashes.\nThere isn't a cl
ear linear relationship; the points on the line are scattered without a d
istinct pattern.\nThis suggests that the total number of crashes may not
have a straightforward correlation with alcohol-related incidents, warran
ting further analysis.\n"
```



In [18]:
```
sns.lineplot(x="total",y="ins_premium",data=df,errorbar=None)
plt.ylabel('Insurance_Premium')
plt.xlabel('Total Crashes')
plt.title('Total Crashes vs. Insurance_Premium')

"""
Inference :
The line plot represents the relationship between total car crashes and i
It visualizes how insurance premiums vary in relation to the total number
The plot does not show a clear linear trend; points on the line are scatt
This suggests that the total number of crashes may not have a straightfor
"""
```

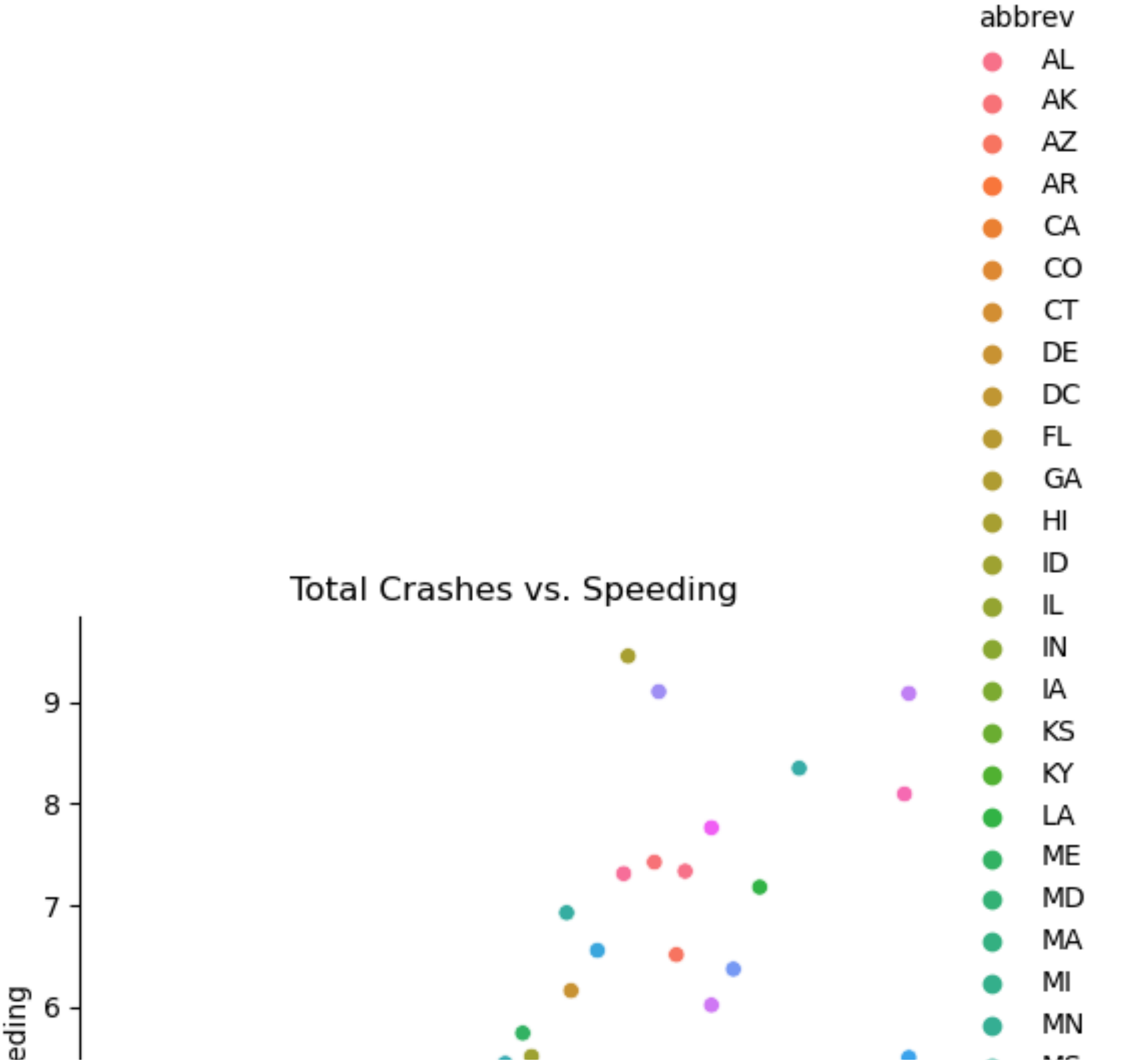
Out[18]:
```
'\nInference :\nThe line plot represents the relationship between total c
ar crashes and insurance premiums.\nIt visualizes how insurance premiums
vary in relation to the total number of crashes.\nThe plot does not show
a clear linear trend; points on the line are scattered without a clear pa
ttern.\nThis suggests that the total number of crashes may not have a str
aightforward correlation with insurance premiums, necessitating further i
nvestigation.\n'
```

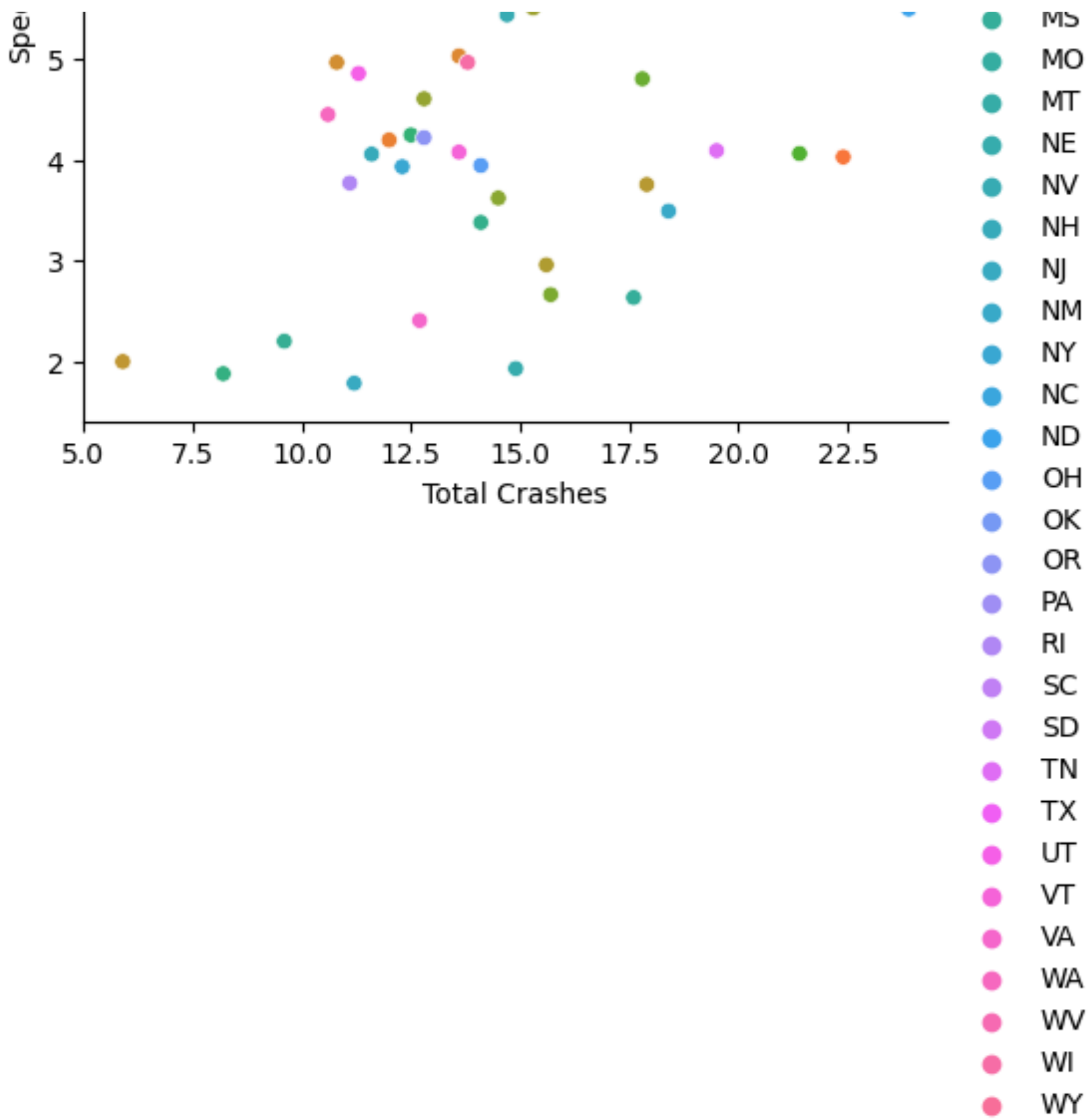


Replot

In [19]:
```
sns.relplot(x="total",y="speeding",data=df,hue="abbrev")
plt.ylabel('Speeding')
plt.xlabel('Total Crashes')
plt.title('Total Crashes vs. Speeding')

"""
Inference :
The relational plot ("relplot") displays the relationship between total c
Each point represents a data point in the dataset, with different states
The plot allows for a quick visual assessment of how speeding-related cra
There is no clear linear trend; points are scattered without a distinct p
"""
```

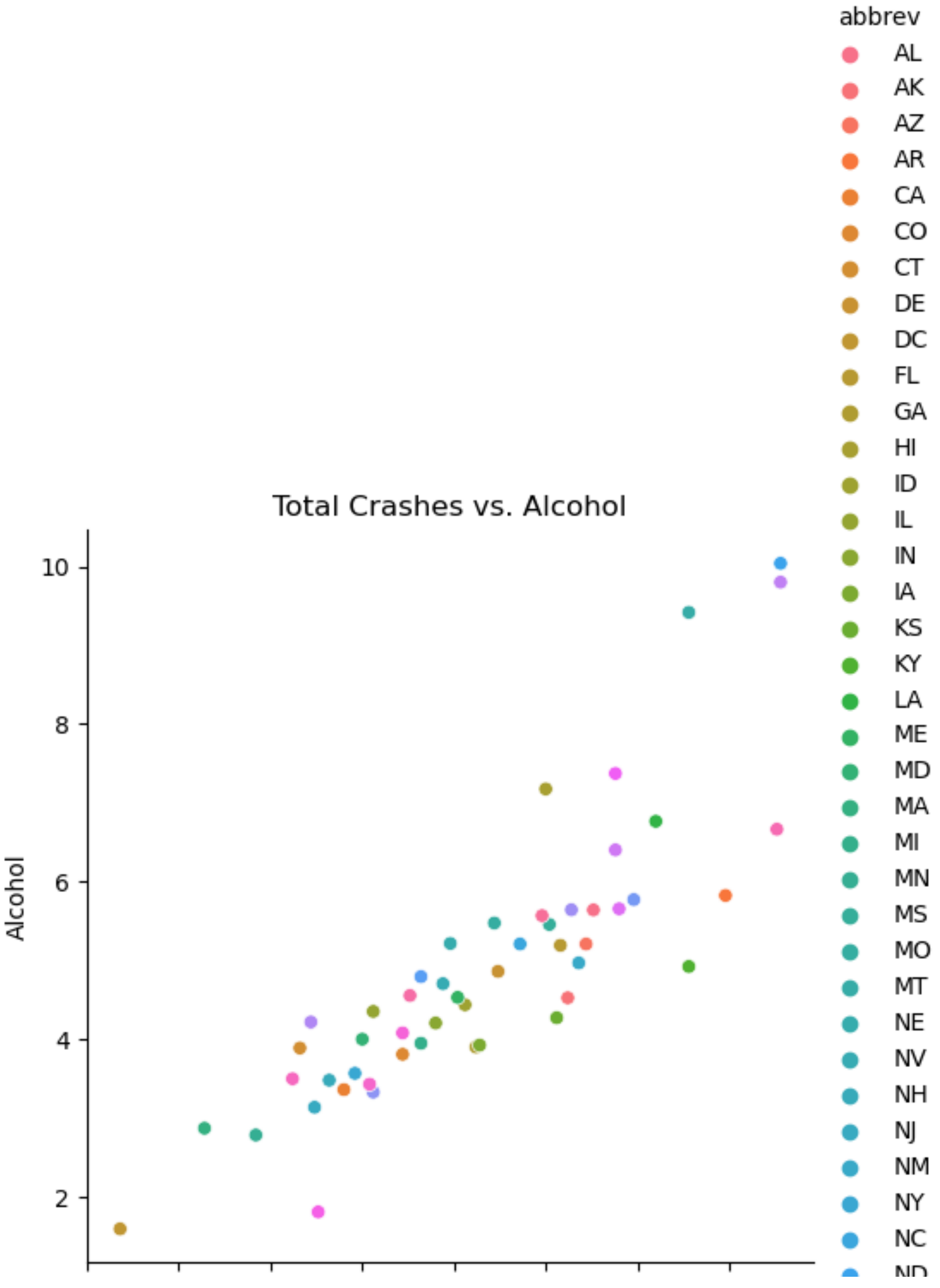
Out[19]:
```
'\nInference :\nThe relational plot ("relplot") displays the relationship
between total car crashes and crashes involving speeding.\nEach point rep
resents a data point in the dataset, with different states distinguished
by colors (hue).\nThe plot allows for a quick visual assessment of how sp
eeding-related crashes vary concerning the total number of crashes in dif
ferent states.\nThere is no clear linear trend; points are scattered with
out a distinct pattern, indicating that the relationship between total cr
ashes and speeding incidents may not be straightforward and may vary by s
tate. Further analysis may be required to explore state-specific trends.\
n'
```

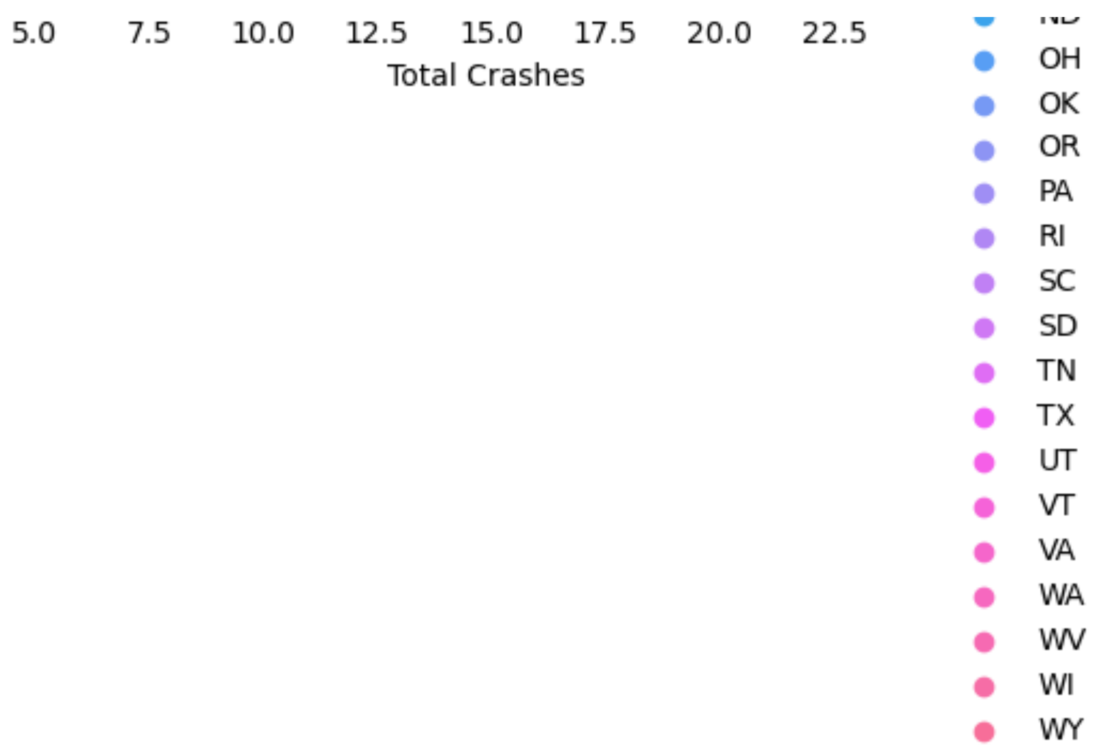




In [20]:
```
sns.relplot(x="total",y="alcohol",data=df,hue="abbrev")
plt.ylabel('Alcohol')
plt.xlabel('Total Crashes')
plt.title('Total Crashes vs. Alcohol')

"""
Inference :
The relational plot ("relplot") illustrates the relationship between tota
Each point on the plot represents a data point in the dataset, and differ
The plot provides a visual comparison of how alcohol-related crashes vary
There isn't a clear linear trend in the relationship; points are scattere
"""
```

Out[20]:
```
'\nInference :\nThe relational plot ("relplot") illustrates the relations
hip between total car crashes and crashes involving alcohol.\nEach point
on the plot represents a data point in the dataset, and different states
are color-coded for comparison (hue).\nThe plot provides a visual compari
son of how alcohol-related crashes vary with the total number of crashes
in different states.\nThere isn\'t a clear linear trend in the relationsh
ip; points are scattered without a distinct pattern, suggesting that the
association between total crashes and alcohol-related incidents may diffe
r by state. Further state-specific analysis may be needed to explore this
further.\n'
```





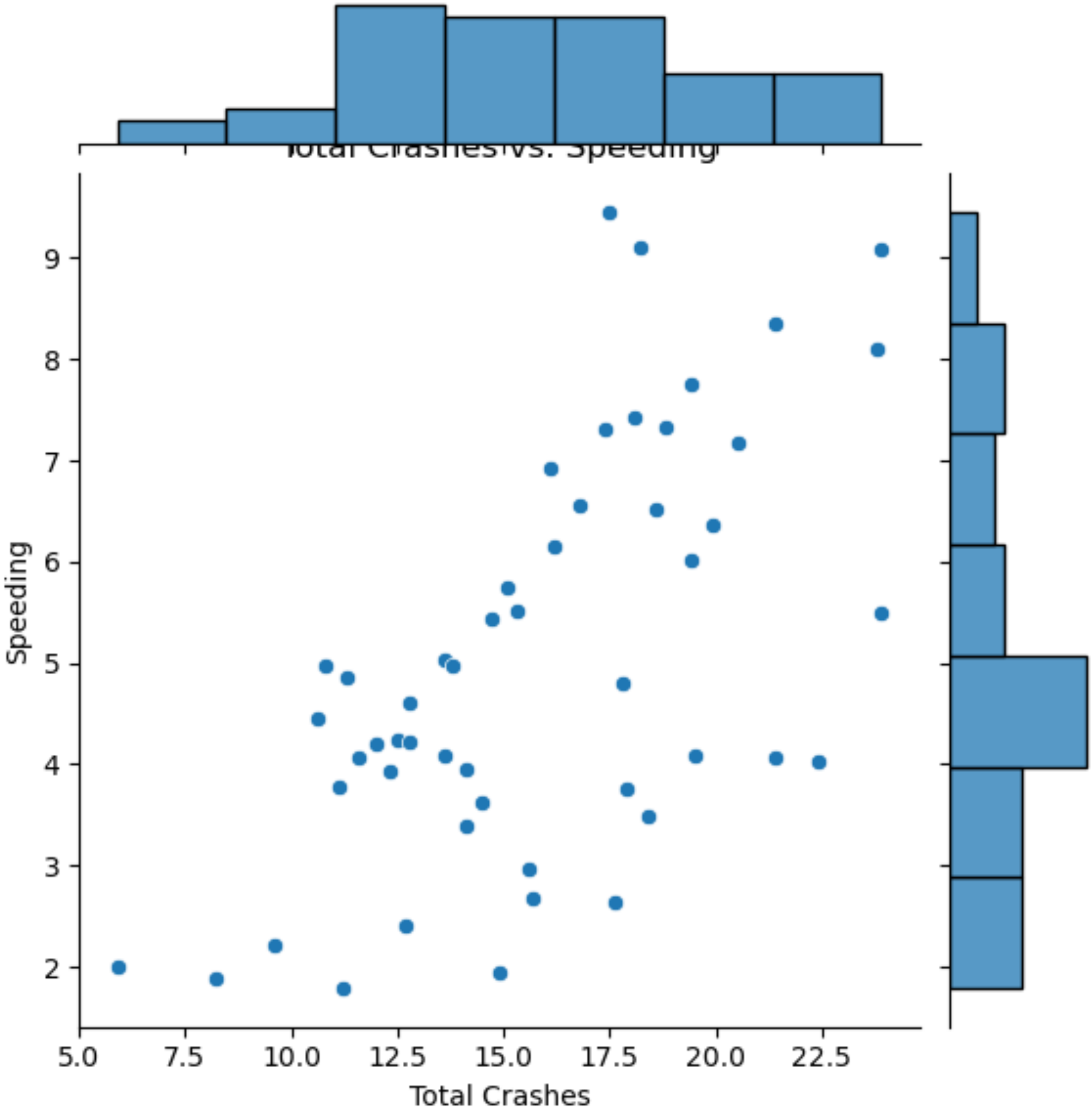
Jointplot

In [21]:

```
sns.jointplot(x="total",y="speeding",data=df)
plt.ylabel('Speeding')
plt.xlabel('Total Crashes')
plt.title('Total Crashes vs. Speeding')

"""
Inference :
The joint plot displays the relationship between total car crashes and cr
It combines a scatter plot and histograms to visualize the distribution a
The scatter plot shows that there isn't a strong linear relationship betw
The histograms on the top and right sides provide additional information
"""
```

Out[21]:

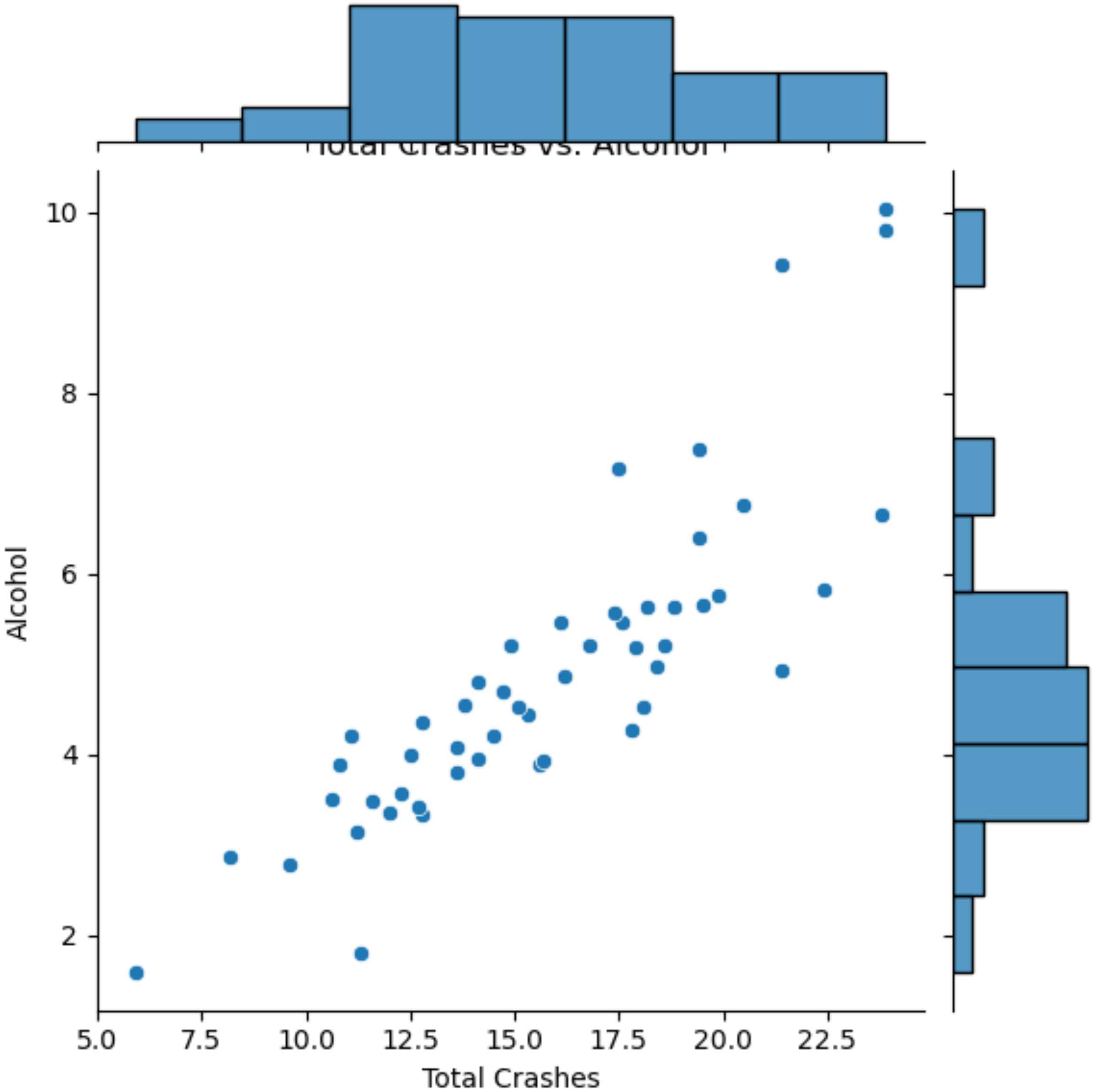
```
"\nInference :\nThe joint plot displays the relationship between total ca
r crashes and crashes involving speeding.\nIt combines a scatter plot and
histograms to visualize the distribution and correlation between the two
variables.\nThe scatter plot shows that there isn't a strong linear relat
ionship between total crashes and speeding incidents.\nThe histograms on
the top and right sides provide additional information about the distribu
tions of both variables.\n"
```



In [22]:
```
sns.jointplot(x="total",y="alcohol",data=df)
plt.ylabel('Alcohol')
plt.xlabel('Total Crashes')
plt.title('Total Crashes vs. Alcohol')

"""
Inference :
The joint plot visualizes the relationship between total car crashes and
It combines a scatter plot and histograms to provide insights into the di
The scatter plot shows that there isn't a strong linear relationship betw
The histograms on the top and right sides offer additional information ab
"""
```

Out[22]:

```
"\nInference :\nThe joint plot visualizes the relationship between total car crashes and crashes involving alcohol.\nIt combines a scatter plot and histograms to provide insights into the distribution and correlation between the two variables.\nThe scatter plot shows that there isn't a strong linear relationship between total crashes and alcohol-related incidents.\nThe histograms on the top and right sides offer additional information about the distributions of both variables.\n"
```



Multivariate

Definition: Multivariate data analysis deals with the examination of three or more variables simultaneously, often in complex datasets.

Objective: The primary goal is to uncover intricate relationships, dependencies, and patterns involving multiple variables. It aims to explore how these variables collectively impact the outcome or phenomenon under study.

Methods: Common methods include multiple regression analysis, principal component analysis (PCA), factor analysis, cluster analysis, and machine learning techniques like decision trees, random forests, and neural networks. These methods enable the exploration of complex interactions and dependencies among multiple variables.

In [23]:
```
corr=df.corr() # Finding the co relation between all the fields in the da
corr
```

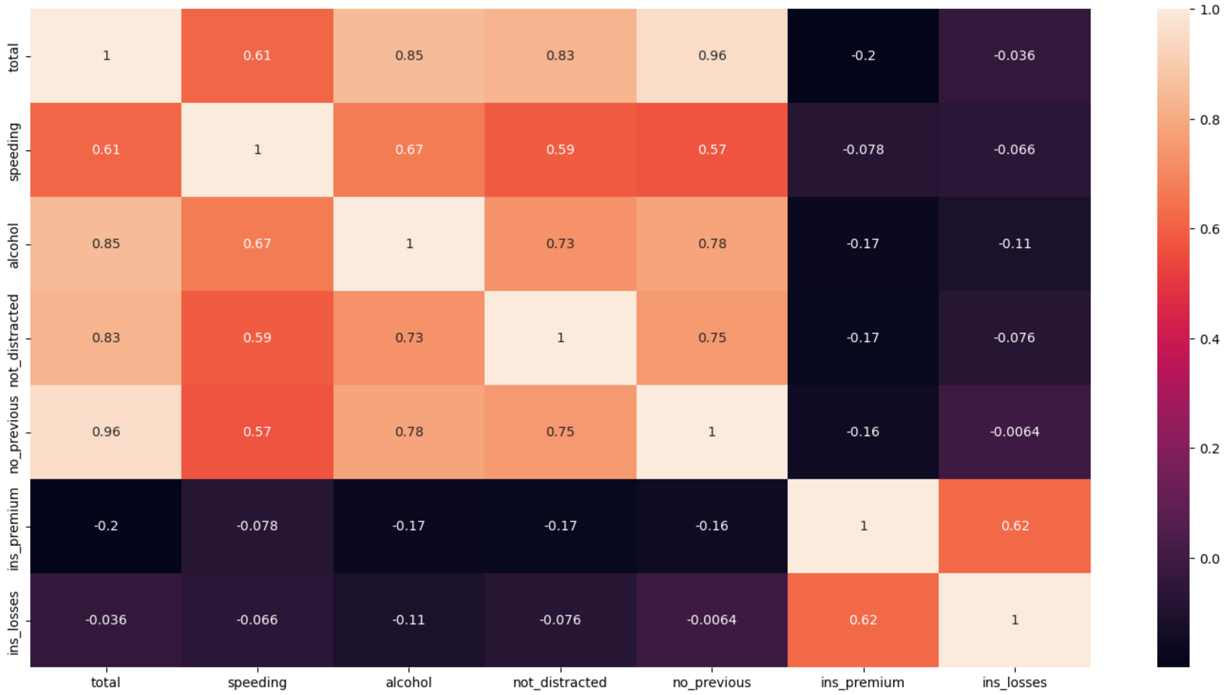
```
/var/folders/ts/3tpz42vj2b9d7t29wtqg41ch0000gn/T/ipykernel_38606/28736413
9.py:1: FutureWarning: The default value of numeric_only in DataFrame.cor
r is deprecated. In a future version, it will default to False. Select on
ly valid columns or specify the value of numeric_only to silence this war
ning.
  corr=df.corr() # Finding the co relation between all the fields in the
dataset and storing it in the variable 'corr'.
```

Out[23]:

	total	speeding	alcohol	not_distracted	no_previous	ins_premiu
total	1.000000	0.611548	0.852613	0.827560	0.956179	-0.19970
speeding	0.611548	1.000000	0.669719	0.588010	0.571976	-0.07767
alcohol	0.852613	0.669719	1.000000	0.732816	0.783520	-0.17061
not_distracted	0.827560	0.588010	0.732816	1.000000	0.747307	-0.17485
no_previous	0.956179	0.571976	0.783520	0.747307	1.000000	-0.15689
ins_premium	-0.199702	-0.077675	-0.170612	-0.174856	-0.156895	1.00000
ins_losses	-0.036011	-0.065928	-0.112547	-0.075970	-0.006359	0.62311

In [24]:
```
plt.subplots(figsize=(18,9))
sns.heatmap(corr,annot=True)

"""
Inference :
The heatmap visualizes the correlation between different variables in the
Darker colors indicate stronger positive correlations, while lighter colo
The heatmap allows for a quick assessment of which variables are strongly
For example, if two variables have a dark-colored cell, it indicates a st
This visualization is valuable for identifying potential relationships an
"""
```

Out[24]:
```
'\nInference :\nThe heatmap visualizes the correlation between different variables in the dataset.\nDarker colors indicate stronger positive correlations, while lighter colors represent weaker or negative correlations.\nThe heatmap allows for a quick assessment of which variables are strongly correlated and which are not.\nFor example, if two variables have a dark-colored cell, it indicates a strong positive correlation between them.\nThis visualization is valuable for identifying potential relationships and dependencies within the dataset.\n'
```



In [ ]: