In [2]:
```
# Importing Seaborn and matplotlib packages
import seaborn as sns
import matplotlib.pyplot as plt
```

In [4]:
```
# Loading the car crashes dataset
df = sns.load_dataset('car_crashes')
```

In [5]:
```
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
```

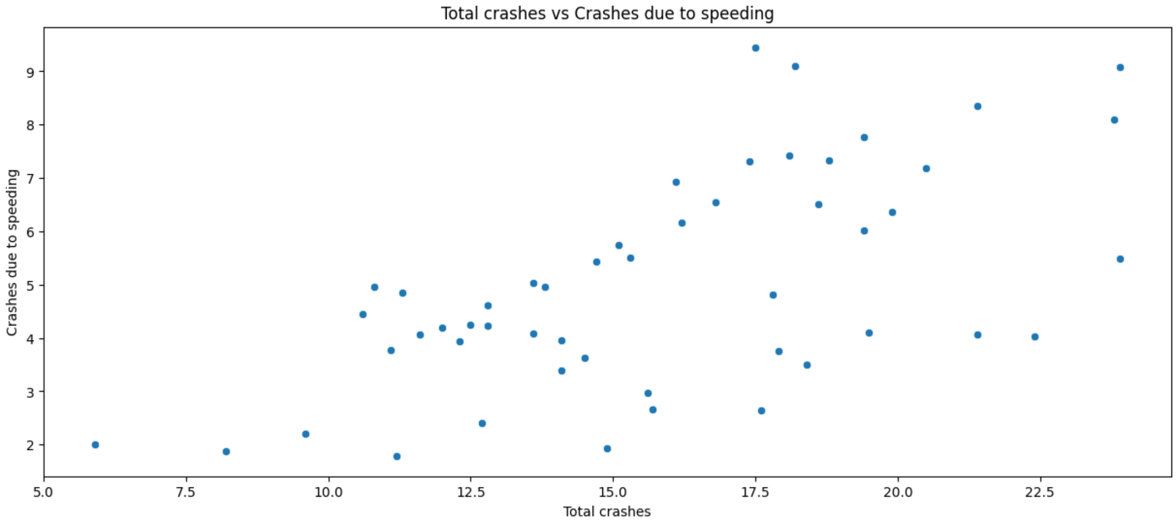
In [100]:
```
df.head()
```

Out[100]:

	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

In [13]:

```
# Scatter plot
plt.figure(figsize=(15, 6))
sns.scatterplot(x='total', y='speeding', data=df)
plt.xlabel('Total crashes')
plt.ylabel('Crashes due to speeding')
plt.title('Total crashes vs Crashes due to speeding')
plt.show()
```

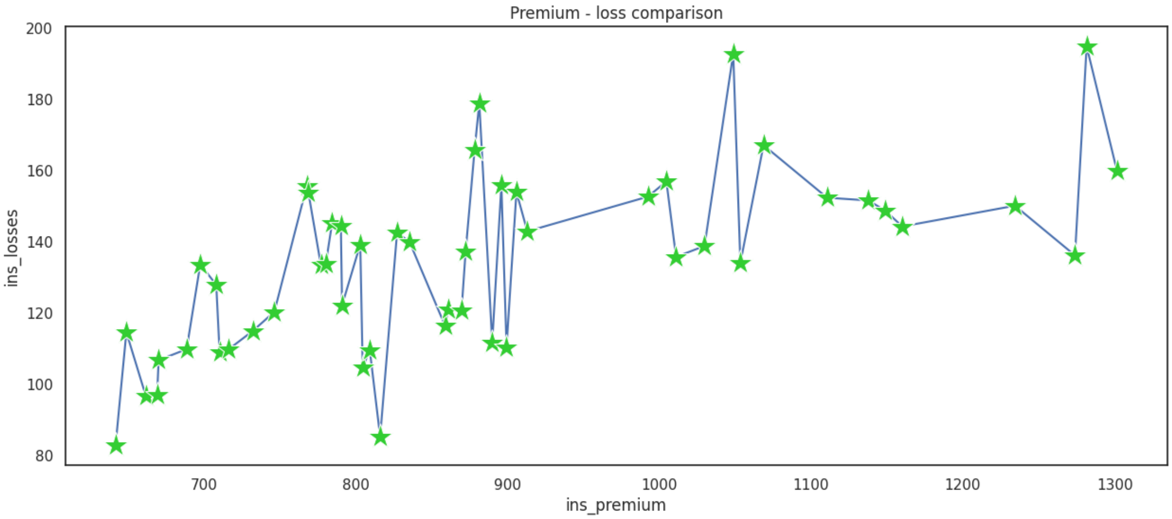


In [ ]:

```
Inference -
     The scatterplot compares Total car crashes with the crashes due to speedi
     The plot has a positive correlation and when there are more number of cra
```

In [25]:

```
# Line plot
plt.figure(figsize=(15, 6))
sns.lineplot(x='ins_premium', y ='ins_losses', data=df, marker='*', markerface
plt.xlabel('ins_premium')
plt.ylabel('ins_losses')
plt.title('Premium - loss comparison')
plt.show()
```

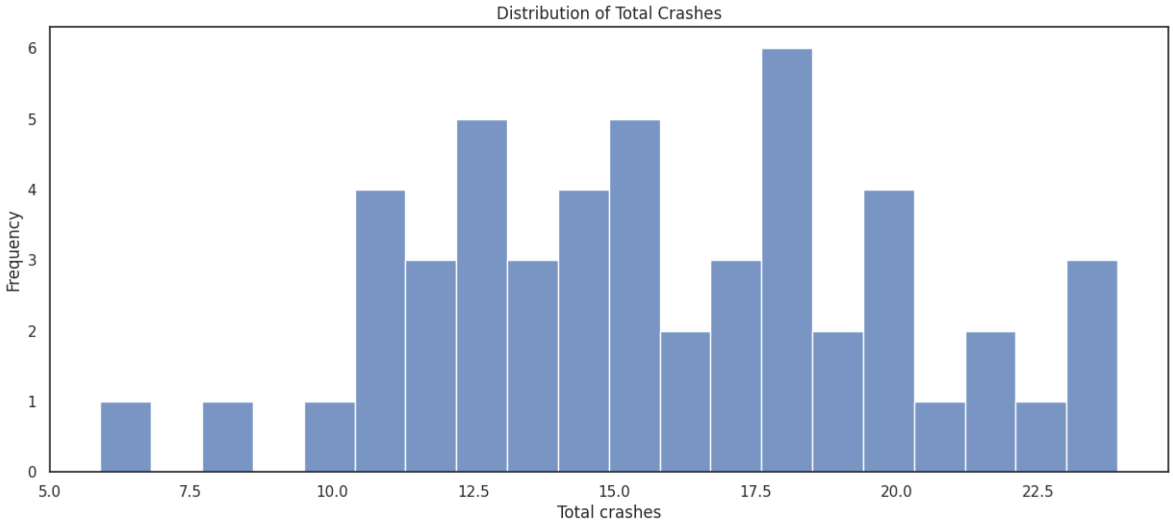


In [ ]:

```
Inference-
  The above lineplot compares the insurance premium and insurance loss over th
  From the plot it is clear that the maximum loss occurred for a insurance pre
```

In [36]:

```
# Histplot
plt.figure(figsize=(15, 6))
sns.histplot(df["total"], bins=20)
plt.title("Distribution of Total Crashes")
plt.xlabel("Total crashes")
plt.ylabel("Frequency")
plt.show()
```

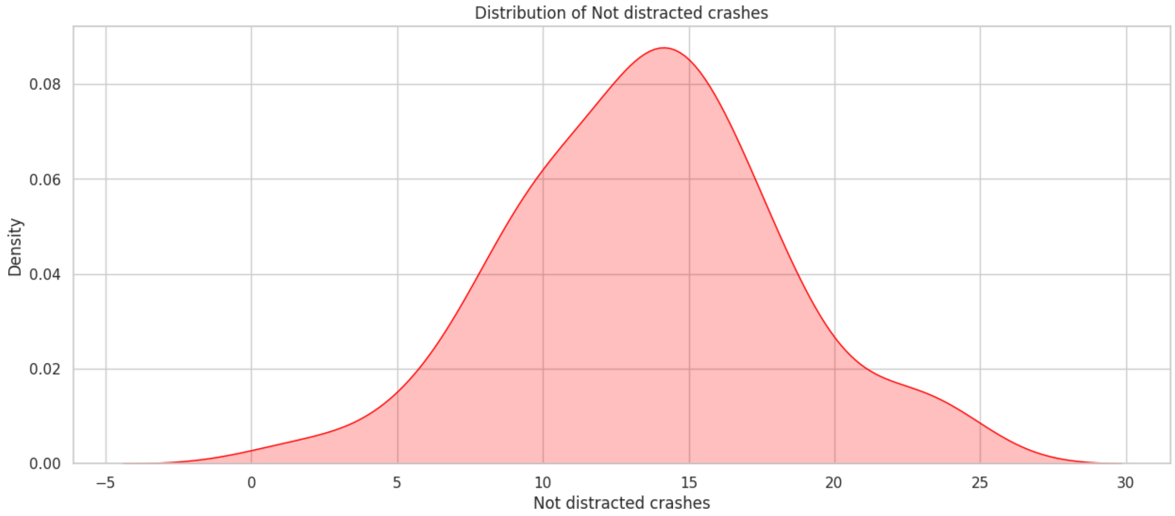


In [ ]:

```
Inference -
  The histogram for the distribution of the total number of crashes shows a ri
  with most states having a lower number of crashes, clustered around 5 to 10
```

In [99]:

```
#Kernel density plot
plt.figure(figsize=(15, 6))
sns.kdeplot(df["not_distracted"], fill = True,color='red',cbar=True)
plt.title("Distribution of Not distracted crashes")
plt.xlabel("Not distracted crashes")
plt.ylabel("Density")
plt.grid()
plt.show()
```

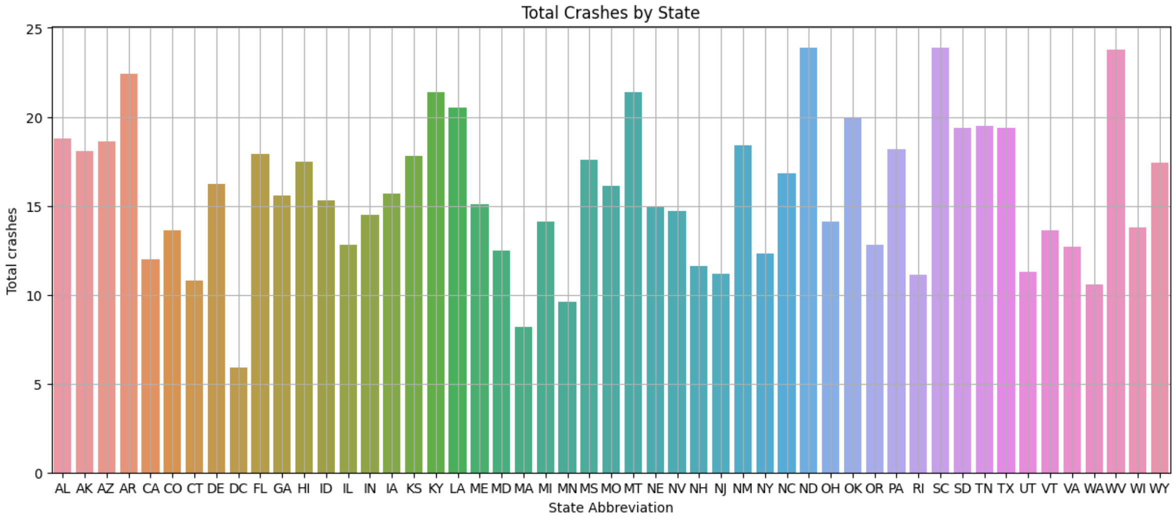


In [ ]:

```
Inference-
  The above kernel density plot shows the density levels of notdistracted cras
  The maximum density 0.09 is obtained for 13 not distracted crashes and there
```

In [12]:

```
#Barplot
plt.figure(figsize=(15, 6))
sns.barplot(x="abbrev", y="total", data=df, orient='v')
plt.title("Total Crashes by State")
plt.xlabel("State Abbreviation")
plt.ylabel("Total crashes")
plt.grid()
plt.show()
```



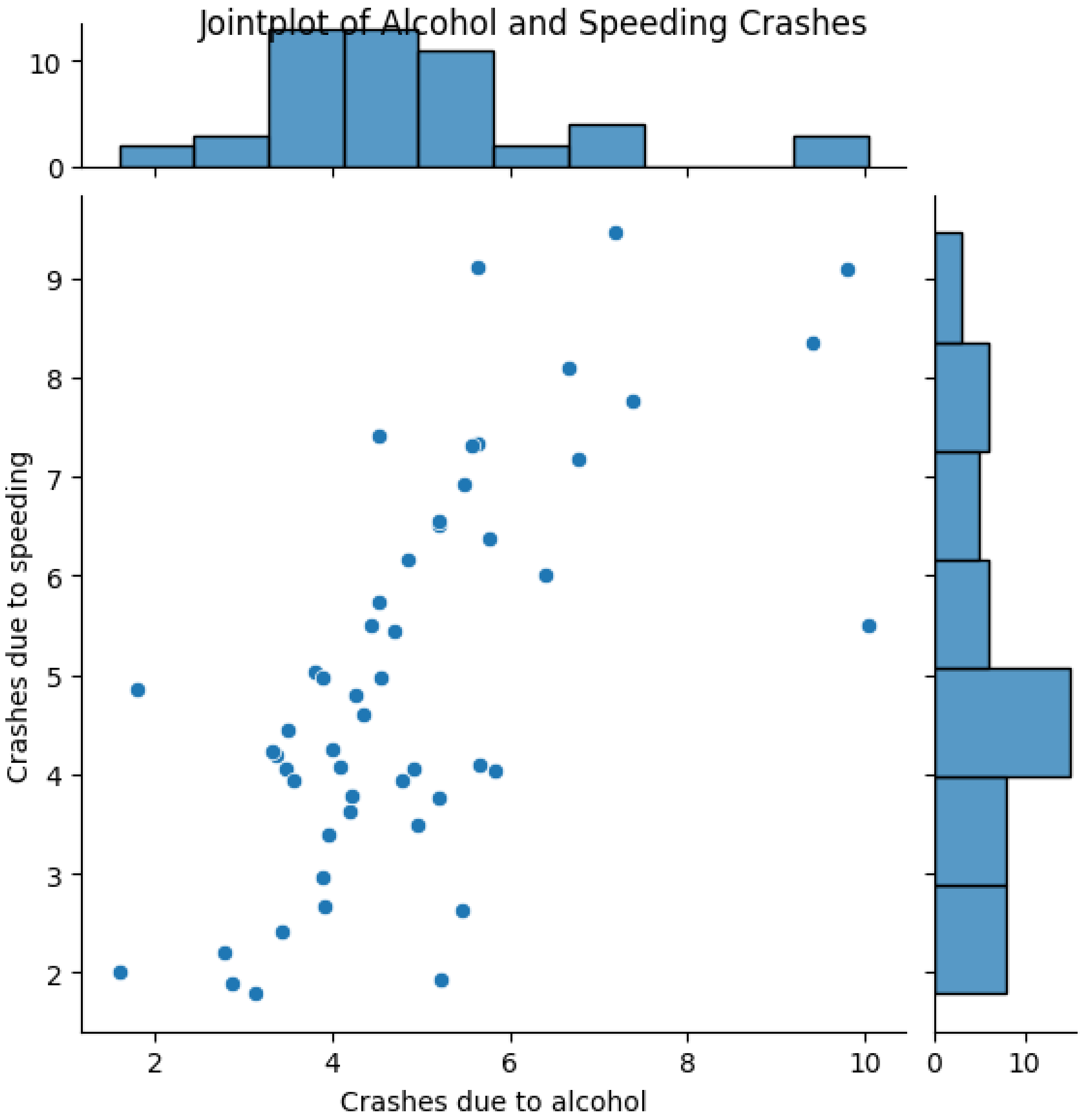
In [ ]:

```
Inference-
 The barplot shows the total number of accidents by state,
 North Dakota, South Carolina & West Virginia have the highest number of accic
```

In [51]:

```
#Jointplot
plt.figure(figsize=(15, 15))
sns.jointplot(x="alcohol", y="speeding", data=df, kind="scatter",marginal_tick
plt.suptitle("Jointplot of Alcohol and Speeding Crashes")
plt.xlabel("Crashes due to alcohol")
plt.ylabel("Crashes due to speeding")
plt.show()
```

```
<Figure size 1500x1500 with 0 Axes>
```

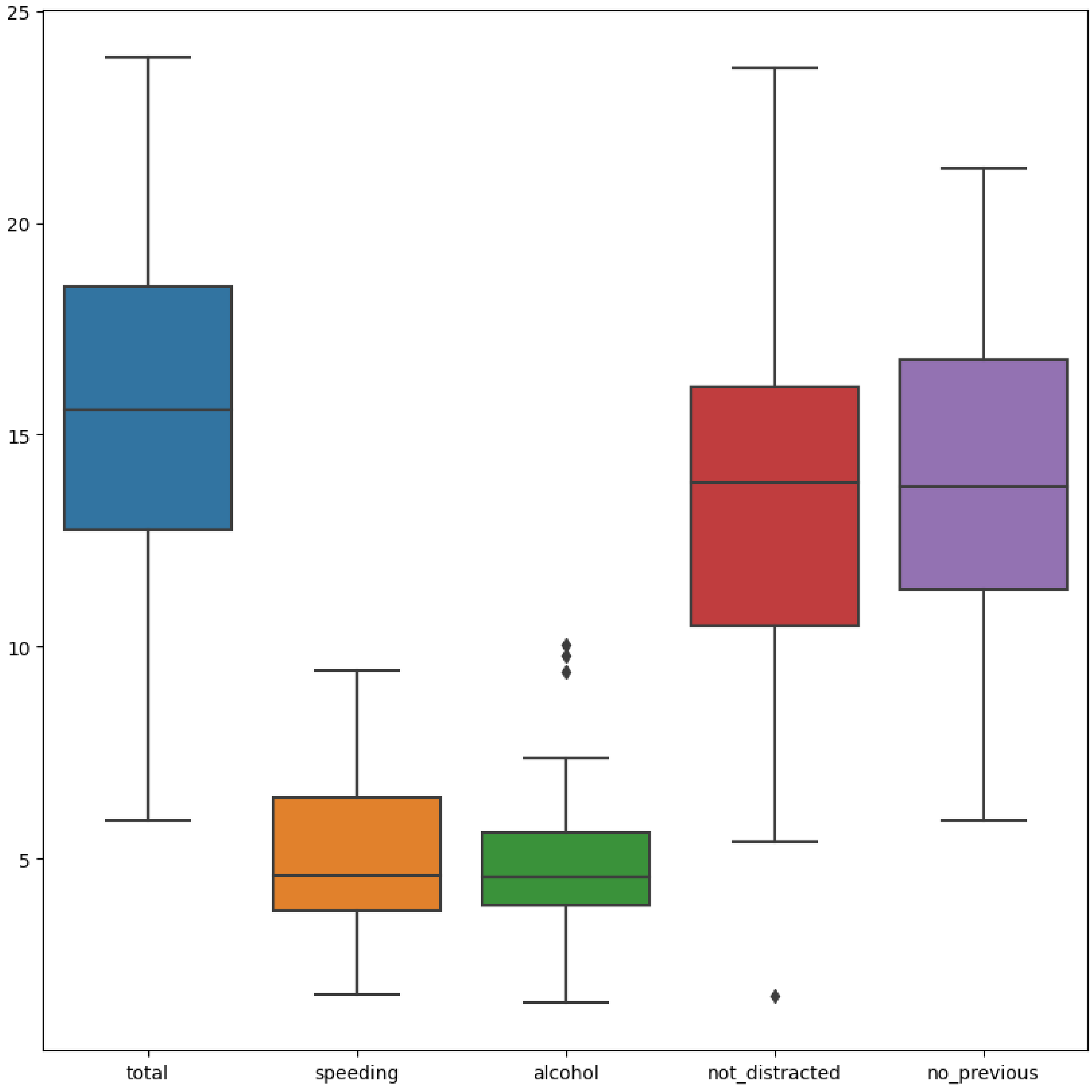


In [ ]:

```
Inference-
 The jointplot visually represents the relationship between alcohol and speedi
 It shows that as alcohol-related accidents increase, speeding-related accider
```

In [76]:

```
#Boxplot
plt.figure(figsize=(10, 10))
sns.boxplot(df.iloc[:,0:5])
plt.show()
```

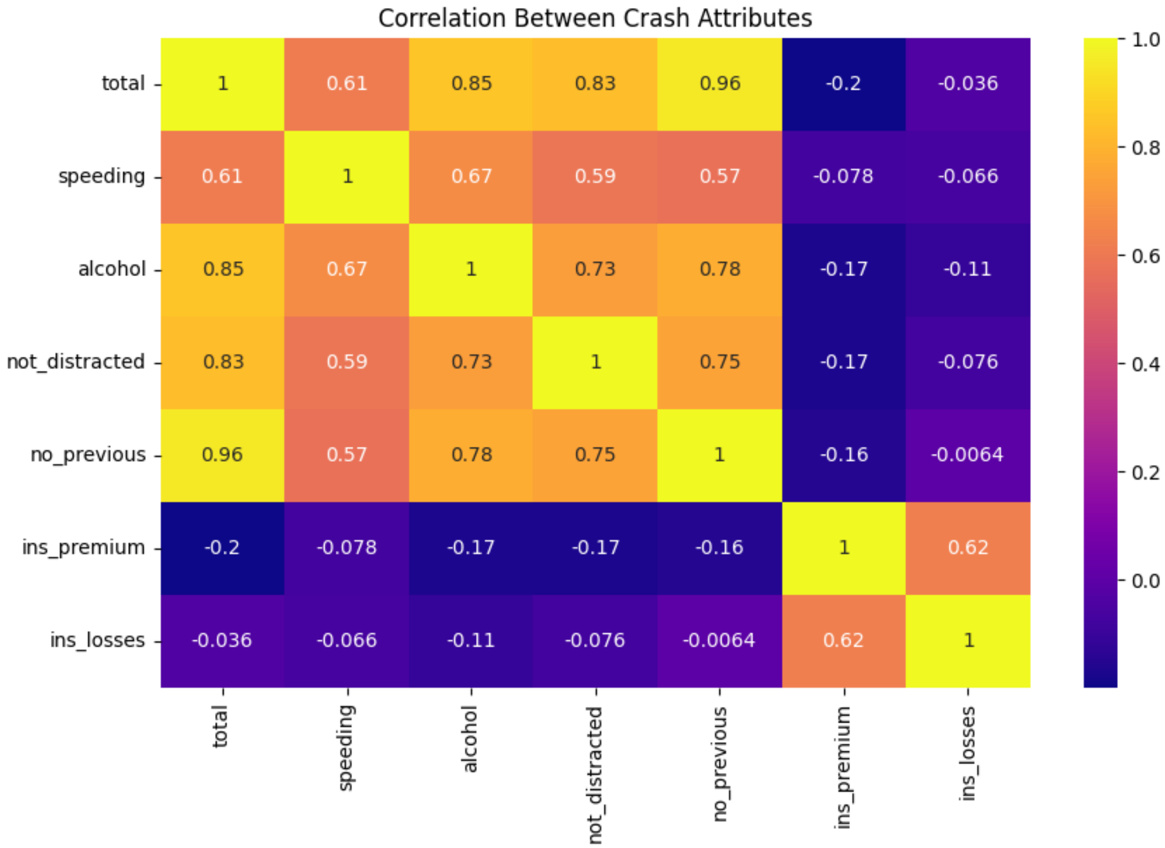


In [ ]:

```
Inference-
  The boxplot of the dataset shows thath there are few outliers for number of
```

In [81]:
```
#Heatmap
correlation_matrix = df.corr()
plt.figure(figsize=(10, 6))
sns.heatmap(correlation_matrix, annot=True, cmap="plasma")
plt.title("Correlation Between Crash Attributes")
plt.show()
```

```
<ipython-input-81-798adf821de5>:2: FutureWarning: The default value of numeri
c_only in DataFrame.corr is deprecated. In a future version, it will default
to False. Select only valid columns or specify the value of numeric_only to s
ilence this warning.
  correlation_matrix = df.corr()
```



In [ ]:
```
Inference-
 The heatmap illustrating the correlation between crashes attributes shows tha
```

In [84]:

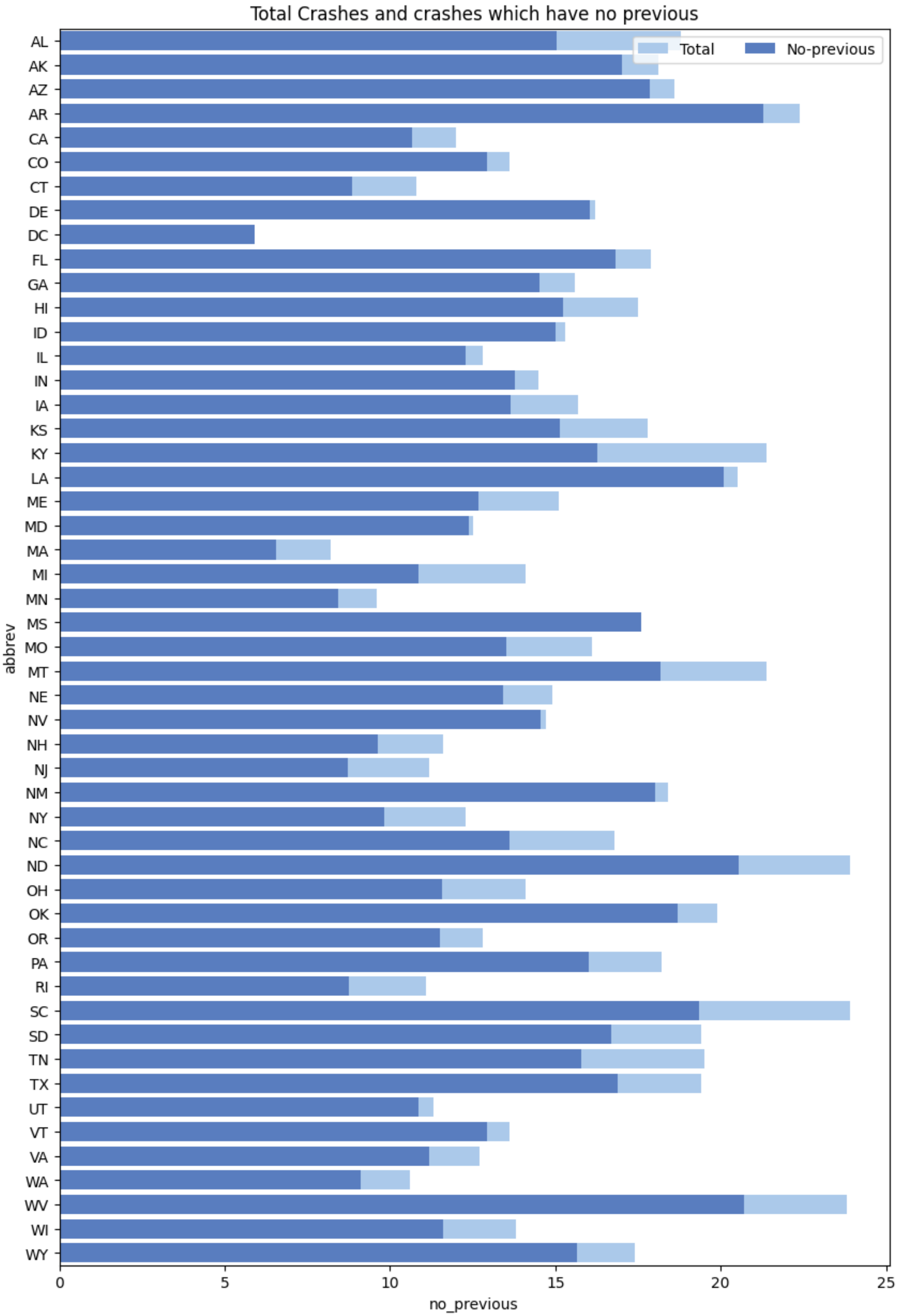
```
# Stacked barchart
f, ax = plt.subplots(figsize=(10, 15))

# Plot the total crashes
sns.set_color_codes("pastel")
sns.barplot(x="total", y="abbrev", data=df ,label="Total", color="b")

# Plot the crashes who are not involved in previous crashes
sns.set_color_codes("muted")
sns.barplot(x="no_previous", y="abbrev", data=df, label="No-previous", color="
plt.title("Total Crashes and crashes which have no previous crashes")
ax.legend(ncol=2, loc="upper right", frameon=True)
```

Out[84]:

```
<matplotlib.legend.Legend at 0x7b4ea26fd2d0>
```

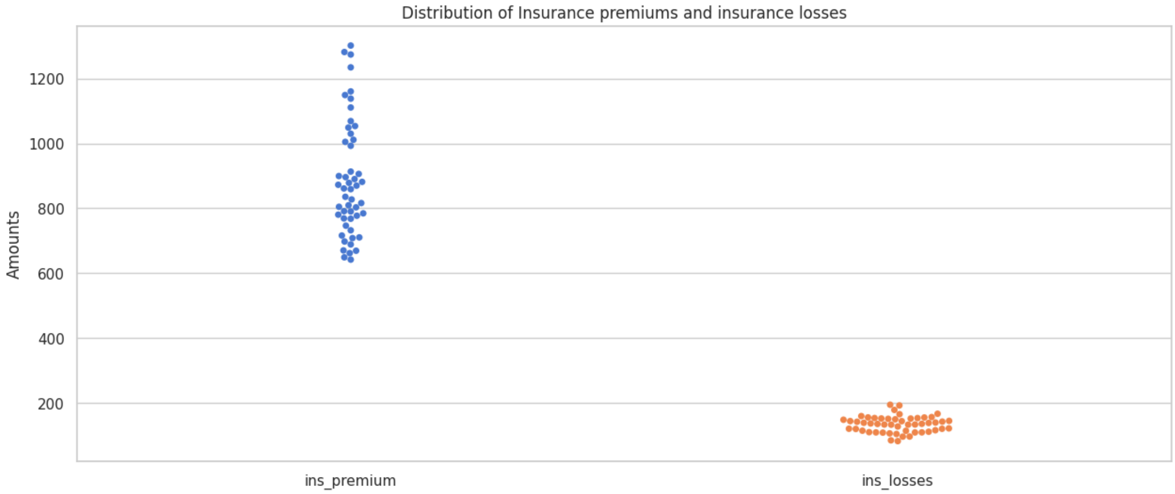

Total Crashes and crashes which have no previous
Total
No-previous
abbrev
AL
AK
AZ
AR
CA
CO
CT
DE
DC
FL
GA
HI
ID
IL
IN
IA
KS
KY
LA
ME
MD
MA
MI
MN
MS
MO
MT
NE
NV
NH
NJ
NM
NY
NC
ND
OH
OK
OR
PA
RI
SC
SD
TN
TX
UT
VT
VA
WA
WV
WI
WY
0
5
10
15
20
25
no_previous

In [ ]:

```
Inference-
  The plot contains the bars of both total number of crashes and crashes with
  It is clear that the total number of crashes is directly proportional to No
  More crashes that occur are having no previous crash history
```

In [97]:

```
# Swarm plot
plt.figure(figsize=(15, 6))
sns.swarmplot(df.iloc[:,5:7])
plt.title("Distribution of Insurance premiums and insurance losses")
plt.ylabel("Amounts")
plt.show()
```



In [ ]:

```
Inference-
  The above swarm plot shows the distribution of ins_premium and ins_losses of
  the range of losses is from 0 to 200 while premium range from 600 to 1200, t
```