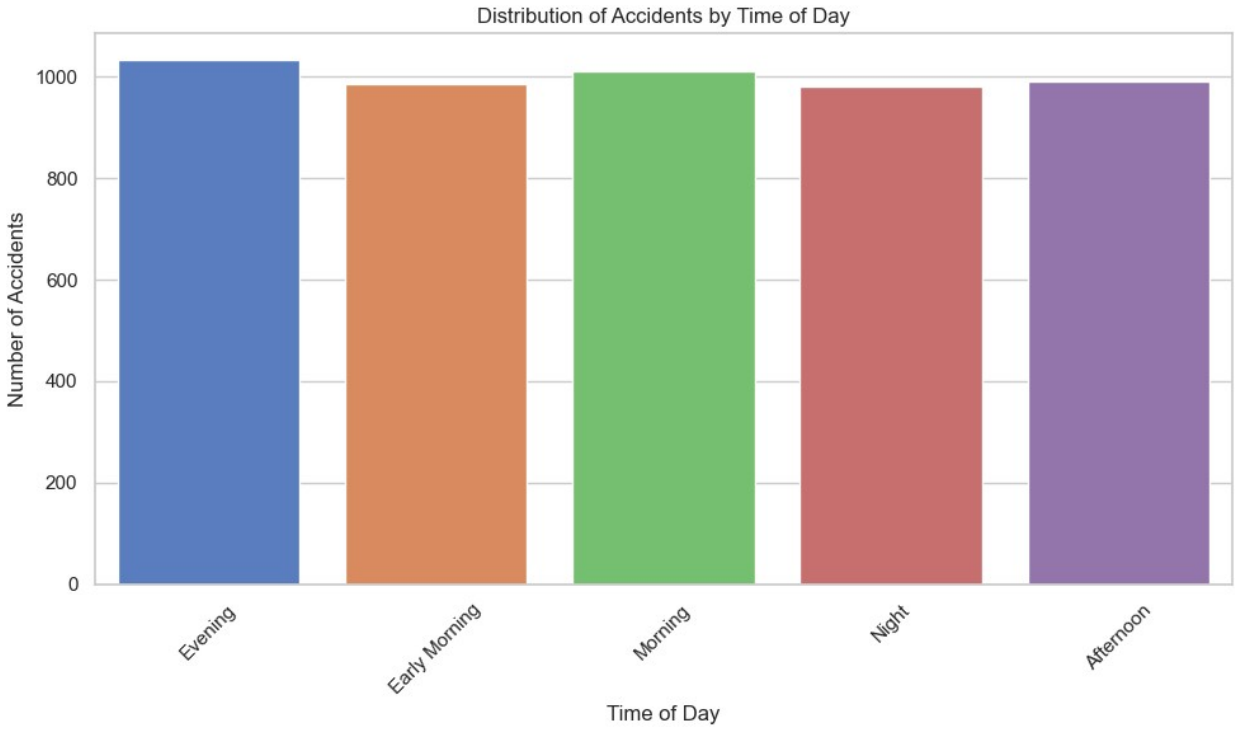
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset
traffic_data = pd.read_csv("traffic_accidents_dataset.csv")
traffic_data.head()
```

```
   Accident_ID    Time_of_Day Weather_Condition      Road_Condition
Severity  \
0            1        Evening             Clear  Under Construction
Medium
1            2  Early Morning             Snowy  Under Construction
High
2            3        Morning             Snowy               Snowy
Low
3            4        Morning             Snowy               Snowy
Medium
4            5        Evening             Foggy                 Dry
Low

   Number_of_Vehicles  Number_of_Casualties Day_of_Week
0                   4                     8     Tuesday
1                   4                     7    Saturday
2                   3                     0   Wednesday
3                   3                     8   Wednesday
4                   3                     9      Friday
```

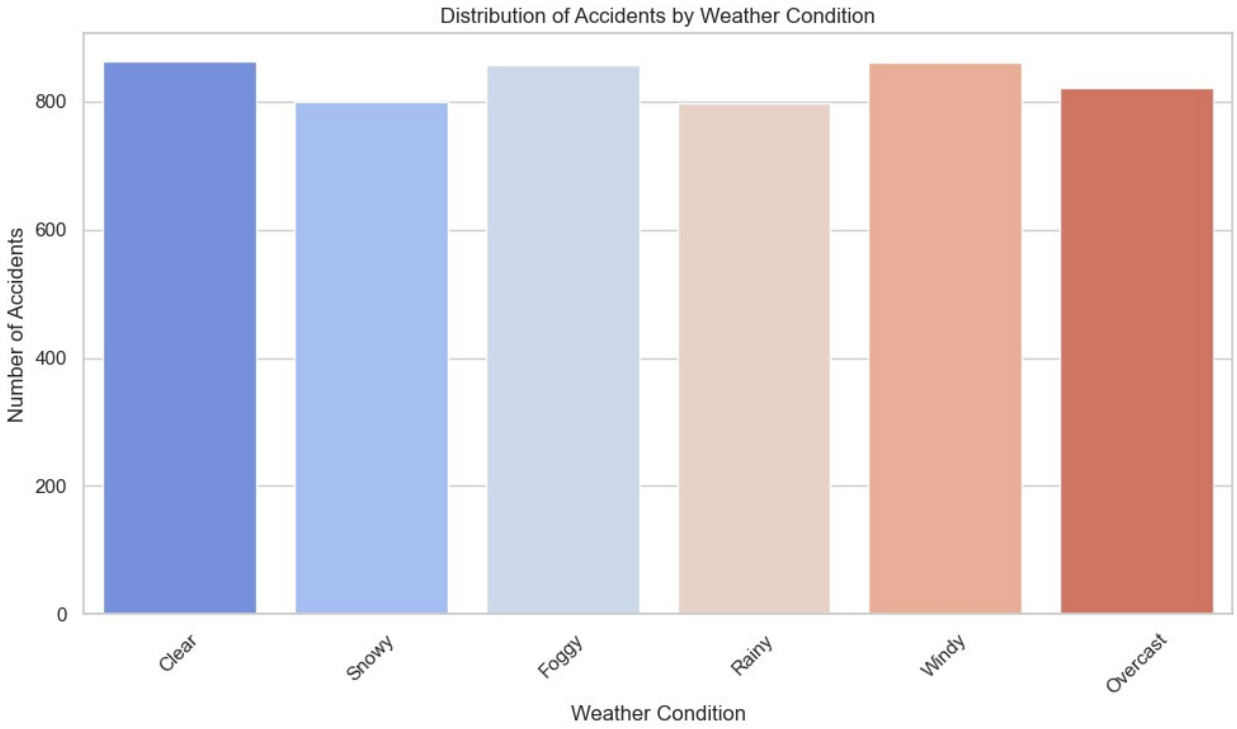
```
# Set plot styles
sns.set(style="whitegrid")

# Exploratory Data Analysis: Distribution of accidents by time of day
plt.figure(figsize=(10, 6))
sns.countplot(data=traffic_data, x="Time_of_Day", palette="muted")
plt.title("Distribution of Accidents by Time of Day")
plt.xlabel("Time of Day")
plt.ylabel("Number of Accidents")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

```
C:\Users\Lenovo\AppData\Local\Temp\ipykernel_11528\26559240.py:6:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.countplot(data=traffic_data, x="Time_of_Day", palette="muted")
```



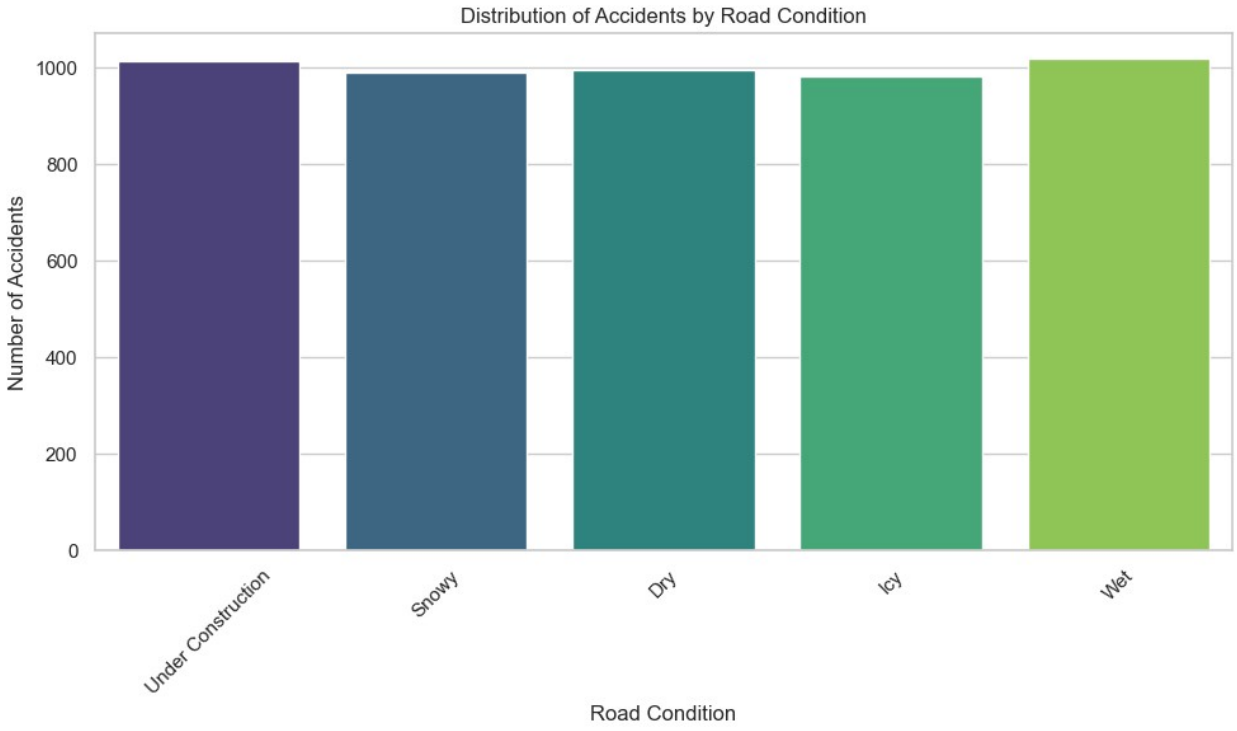

```
# Distribution of accidents by weather condition
plt.figure(figsize=(10, 6))
sns.countplot(data=traffic_data, x="Weather_Condition",
palette="coolwarm")
plt.title("Distribution of Accidents by Weather Condition")
plt.xlabel("Weather Condition")
plt.ylabel("Number of Accidents")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

```
C:\Users\Lenovo\AppData\Local\Temp\ipykernel_11528\2594475895.py:3:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.countplot(data=traffic_data, x="Weather_Condition",
palette="coolwarm")
```

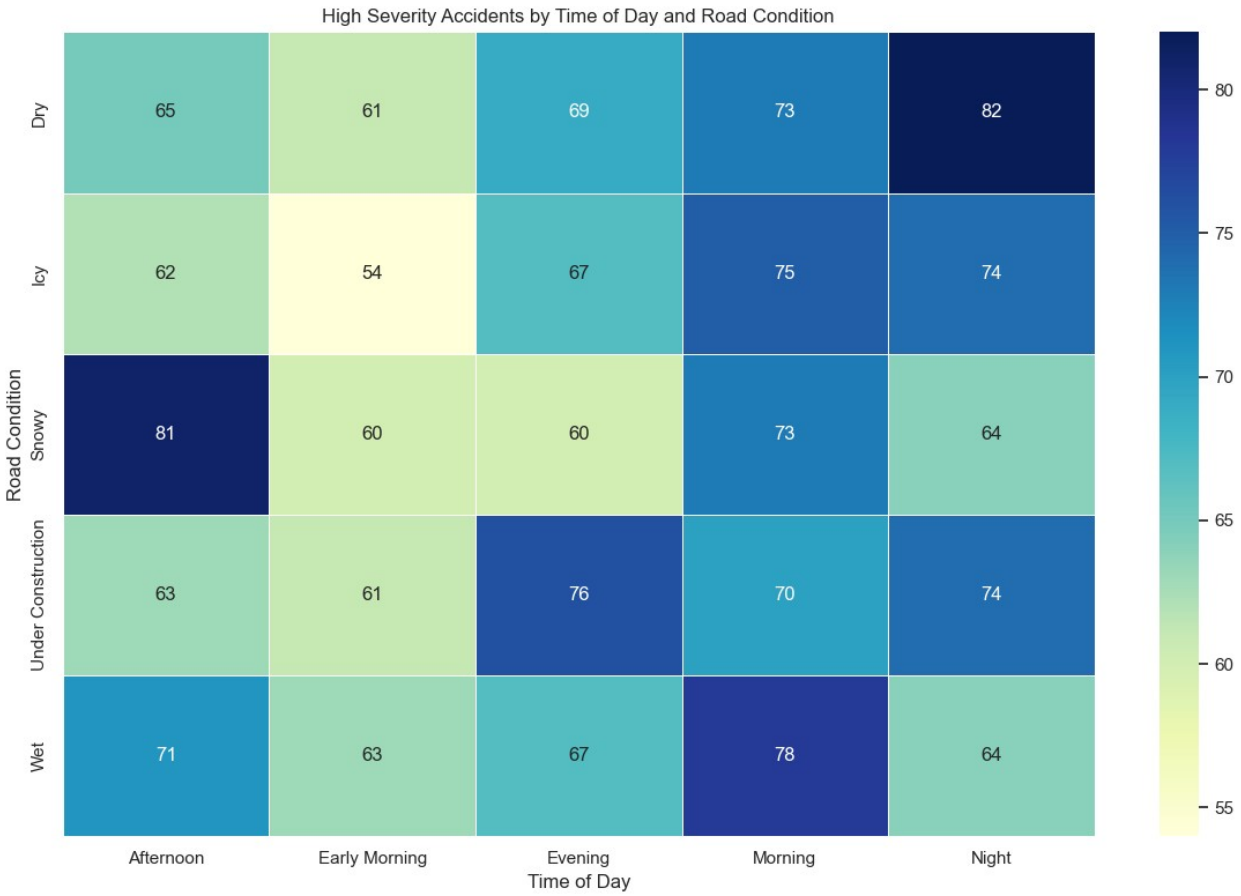



```
# Distribution of accidents by road condition
plt.figure(figsize=(10, 6))
sns.countplot(data=traffic_data, x="Road_Condition",
palette="viridis")
plt.title("Distribution of Accidents by Road Condition")
plt.xlabel("Road Condition")
plt.ylabel("Number of Accidents")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

```
C:\Users\Lenovo\AppData\Local\Temp\ipykernel_11528\3056118063.py:3:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.countplot(data=traffic_data, x="Road_Condition",
palette="viridis")
```

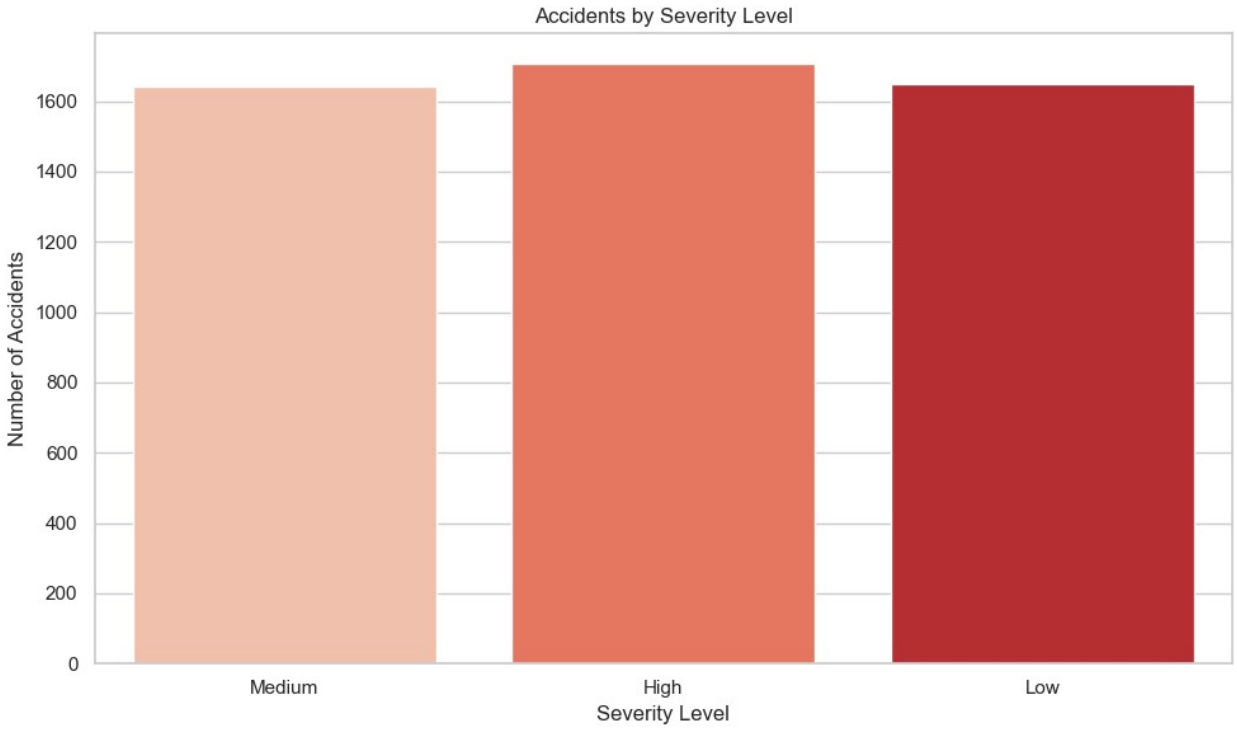



```
# Heatmap of accident severity by time of day and road condition
severity_pivot = traffic_data.pivot_table(
    index="Road_Condition", columns="Time_of_Day", values="Severity",
aggfunc=lambda x: (x == "High").sum()
)

plt.figure(figsize=(12, 8))
sns.heatmap(severity_pivot, annot=True, fmt="d", cmap="YlGnBu",
linewidths=0.5)
plt.title("High Severity Accidents by Time of Day and Road Condition")
plt.xlabel("Time of Day")
plt.ylabel("Road Condition")
plt.tight_layout()
plt.show()
```




```
# Displaying accident hotspots by severity
plt.figure(figsize=(10, 6))
sns.countplot(
    data=traffic_data,
    x="Severity",
    palette="Reds",
)
plt.title("Accidents by Severity Level")
plt.xlabel("Severity Level")
plt.ylabel("Number of Accidents")
plt.tight_layout()
plt.show()
```

```
C:\Users\Lenovo\AppData\Local\Temp\ipykernel_11528\2697175008.py:3:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.countplot(
```




```
Accidents are most frequent during rush hours and on wet or icy roads,
with high-severity cases
often linked to adverse weather and nighttime driving.Prioritizing
safety measures for these hotspots can significantly reduce risks
```