Task5

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import folium
from folium.plugins import HeatMap

df = pd.read_csv("C:\\Users\\hp\\Downloads\\archive\\
US_Accidents_March23.csv")

df.head()
```

```
    ID   Source  Severity           Start_Time             End_Time  \
0  A-1  Source2         3  2016-02-08 05:46:00  2016-02-08 11:00:00
1  A-2  Source2         2  2016-02-08 06:07:59  2016-02-08 06:37:59
2  A-3  Source2         2  2016-02-08 06:49:27  2016-02-08 07:19:27
3  A-4  Source2         3  2016-02-08 07:23:34  2016-02-08 07:53:34
4  A-5  Source2         2  2016-02-08 07:39:07  2016-02-08 08:09:07

   Start_Lat  Start_Lng  End_Lat  End_Lng  Distance(mi)  ...
Roundabout  \
0  39.865147 -84.058723      NaN      NaN          0.01  ...
False
1  39.928059 -82.831184      NaN      NaN          0.01  ...
False
2  39.063148 -84.032608      NaN      NaN          0.01  ...
False
3  39.747753 -84.205582      NaN      NaN          0.01  ...
False
4  39.627781 -84.188354      NaN      NaN          0.01  ...
False

  Station   Stop Traffic_Calming Traffic_Signal Turning_Loop
Sunrise_Sunset  \
0   False  False           False          False        False
Night
1   False  False           False          False        False
Night
2   False  False           False           True        False
Night
3   False  False           False          False        False
Night
4   False  False           False           True        False
Day

  Civil_Twilight Nautical_Twilight Astronomical_Twilight
0          Night             Night                 Night
```

```
1          Night              Night                    Day
2          Night                Day                    Day
3            Day                Day                    Day
4            Day                Day                    Day

[5 rows x 46 columns]
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7728394 entries, 0 to 7728393
Data columns (total 46 columns):
 #   Column                 Dtype
---  ------                 -----
 0   ID                     object
 1   Source                 object
 2   Severity               int64
 3   Start_Time             object
 4   End_Time               object
 5   Start_Lat              float64
 6   Start_Lng              float64
 7   End_Lat                float64
 8   End_Lng                float64
 9   Distance(mi)           float64
 10  Description            object
 11  Street                 object
 12  City                   object
 13  County                 object
 14  State                  object
 15  Zipcode                object
 16  Country                object
 17  Timezone               object
 18  Airport_Code           object
 19  Weather_Timestamp      object
 20  Temperature(F)         float64
 21  Wind_Chill(F)          float64
 22  Humidity(%)            float64
 23  Pressure(in)           float64
 24  Visibility(mi)         float64
 25  Wind_Direction         object
 26  Wind_Speed(mph)        float64
 27  Precipitation(in)      float64
 28  Weather_Condition      object
 29  Amenity                bool
 30  Bump                   bool
 31  Crossing               bool
 32  Give_Way               bool
 33  Junction               bool
 34  No_Exit                bool
 35  Railway                bool
```

```
 36  Roundabout             bool
 37  Station                bool
 38  Stop                   bool
 39  Traffic_Calming        bool
 40  Traffic_Signal         bool
 41  Turning_Loop           bool
 42  Sunrise_Sunset         object
 43  Civil_Twilight         object
 44  Nautical_Twilight      object
 45  Astronomical_Twilight  object
dtypes: bool(13), float64(12), int64(1), object(20)
memory usage: 2.0+ GB
```

```
df.isnull().sum()
```

```
ID                          0
Source                      0
Severity                    0
Start_Time                  0
End_Time                    0
Start_Lat                   0
Start_Lng                   0
End_Lat                3402762
End_Lng                3402762
Distance(mi)                0
Description                 5
Street                  10869
City                      253
County                      0
State                       0
Zipcode                  1915
Country                     0
Timezone                 7808
Airport_Code            22635
Weather_Timestamp      120228
Temperature(F)         163853
Wind_Chill(F)         1999019
Humidity(%)            174144
Pressure(in)           140679
Visibility(mi)         177098
Wind_Direction         175206
Wind_Speed(mph)        571233
Precipitation(in)     2203586
Weather_Condition      173459
Amenity                     0
Bump                        0
Crossing                    0
Give_Way                    0
Junction                    0
No_Exit                     0
```

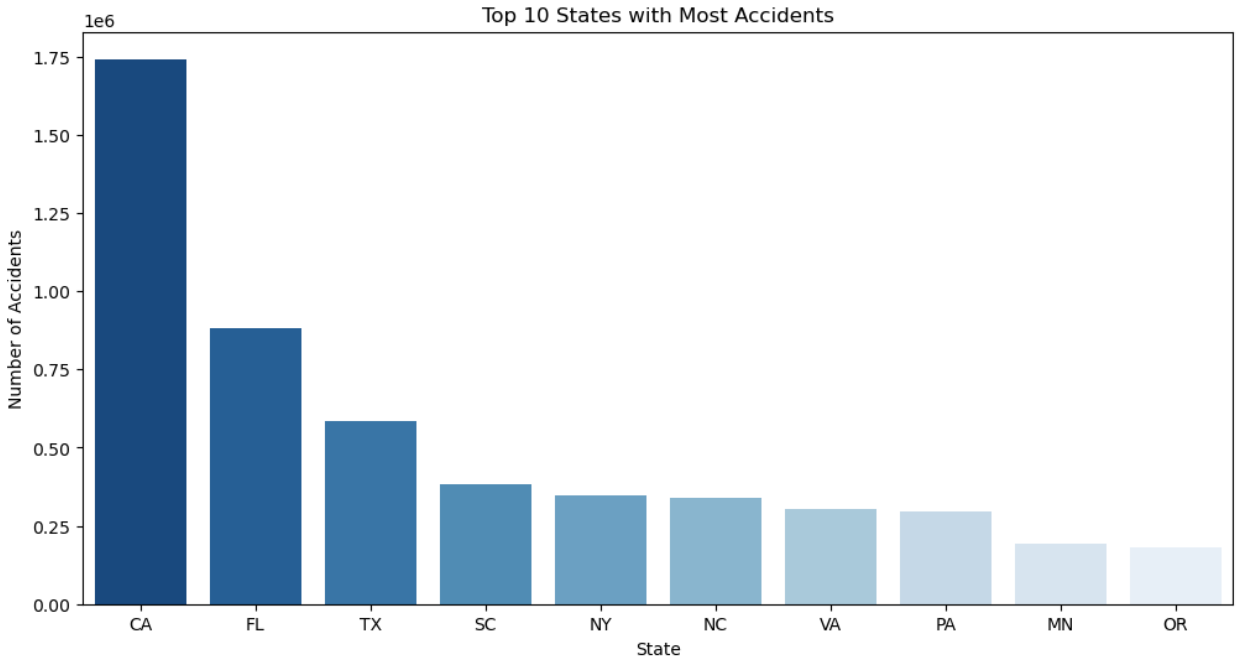
```
Railway                        0
Roundabout                     0
Station                        0
Stop                           0
Traffic_Calming                0
Traffic_Signal                 0
Turning_Loop                   0
Sunrise_Sunset             23246
Civil_Twilight             23246
Nautical_Twilight          23246
Astronomical_Twilight      23246
dtype: int64
```

```
# Fill missing numerical values with median
num_cols = df.select_dtypes(include=['float64', 'int64']).columns
df[num_cols] = df[num_cols].fillna(df[num_cols].median())

# Fill missing categorical values with mode
cat_cols = df.select_dtypes(include=['object']).columns
for col in cat_cols:
    df[col] = df[col].fillna(df[col].mode()[0])

df.isnull().sum()
```

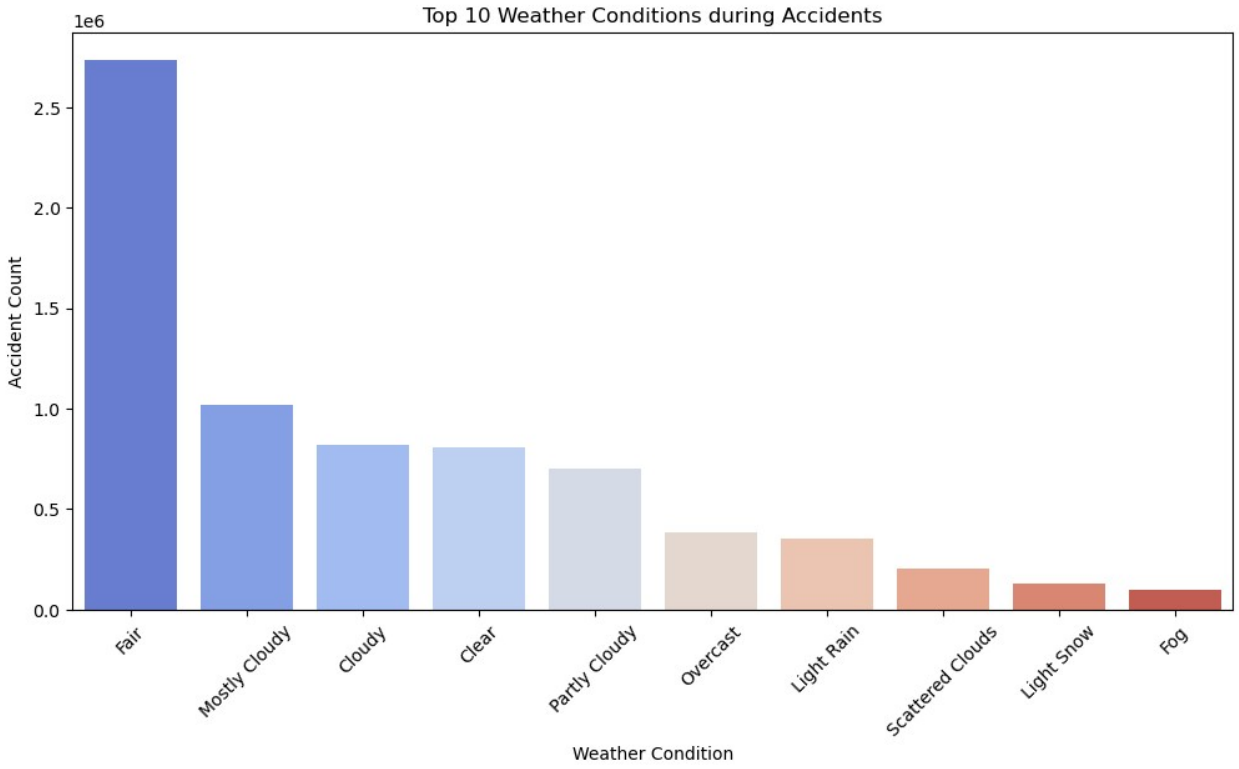
```
ID                       0
Source                   0
Severity                 0
Start_Time               0
End_Time                 0
Start_Lat                0
Start_Lng                0
End_Lat                  0
End_Lng                  0
Distance(mi)             0
Description              0
Street                   0
City                     0
County                   0
State                    0
Zipcode                  0
Country                  0
Timezone                 0
Airport_Code             0
Weather_Timestamp        0
Temperature(F)           0
Wind_Chill(F)            0
Humidity(%)              0
Pressure(in)             0
Visibility(mi)           0
Wind_Direction           0
```

```
Wind_Speed(mph)           0
Precipitation(in)         0
Weather_Condition         0
Amenity                   0
Bump                      0
Crossing                  0
Give_Way                  0
Junction                  0
No_Exit                   0
Railway                   0
Roundabout                0
Station                   0
Stop                      0
Traffic_Calming           0
Traffic_Signal            0
Turning_Loop              0
Sunrise_Sunset            0
Civil_Twilight            0
Nautical_Twilight         0
Astronomical_Twilight     0
dtype: int64
```

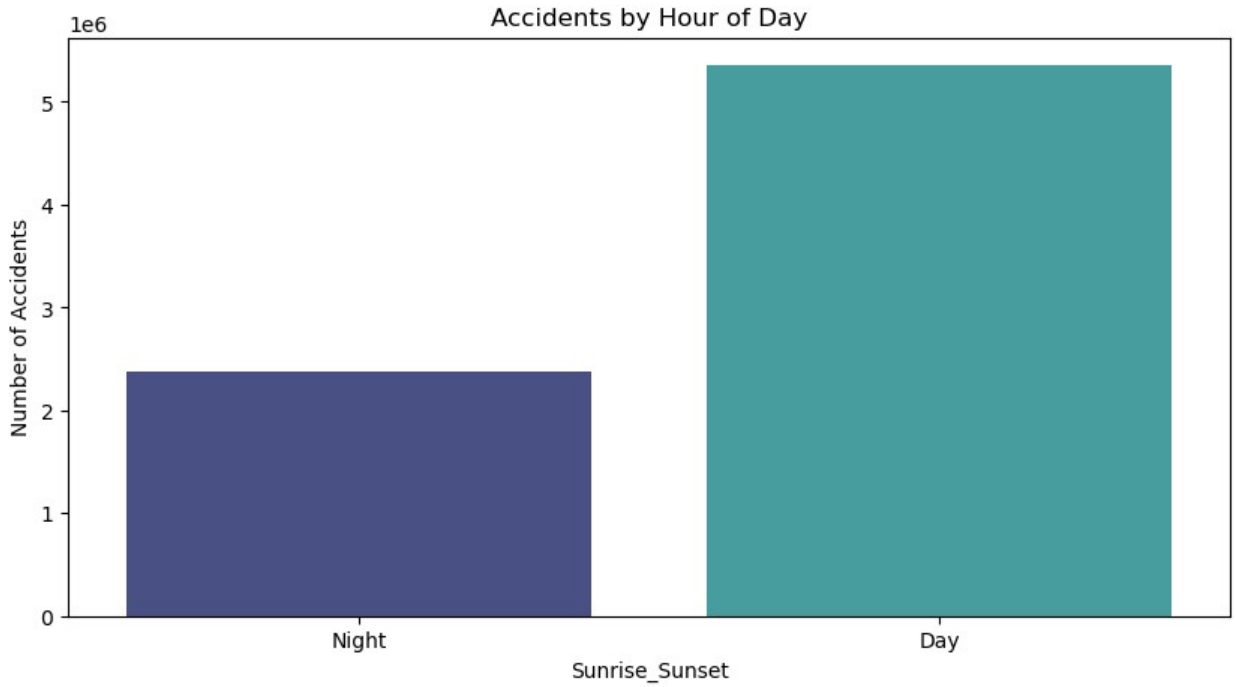
```
plt.figure(figsize=(12,6))
top_states = df['State'].value_counts().head(10)
sns.barplot(x=top_states.index, y=top_states.values,
hue=top_states.index, palette='Blues_r')
plt.title('Top 10 States with Most Accidents')
plt.xlabel('State')
plt.ylabel('Number of Accidents')
plt.show()
```



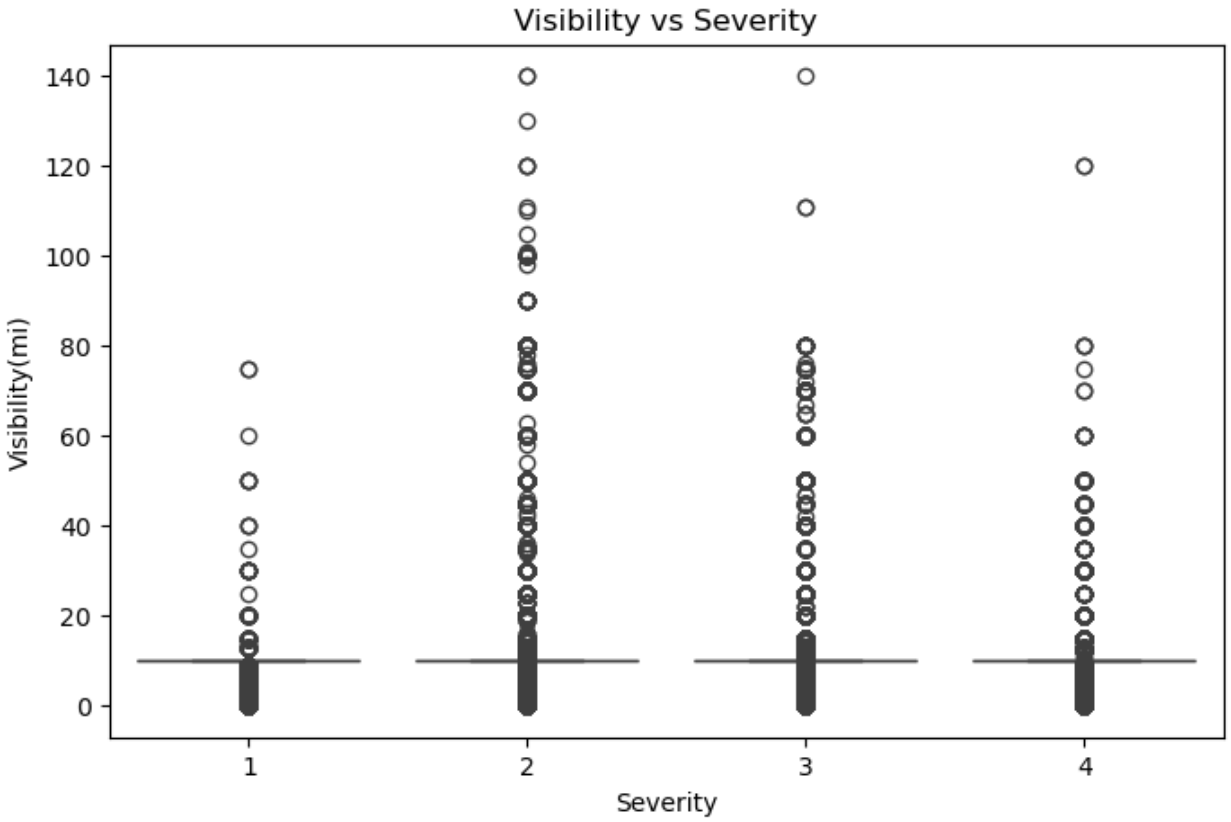

```
plt.figure(figsize=(12,6))
top_weather = df['Weather_Condition'].value_counts().head(10)
sns.barplot(x=top_weather.index, y=top_weather.values,
hue=top_weather.index, palette='coolwarm')
plt.title('Top 10 Weather Conditions during Accidents')
plt.xlabel('Weather Condition')
plt.ylabel('Accident Count')
plt.xticks(rotation=45)
plt.show()
```



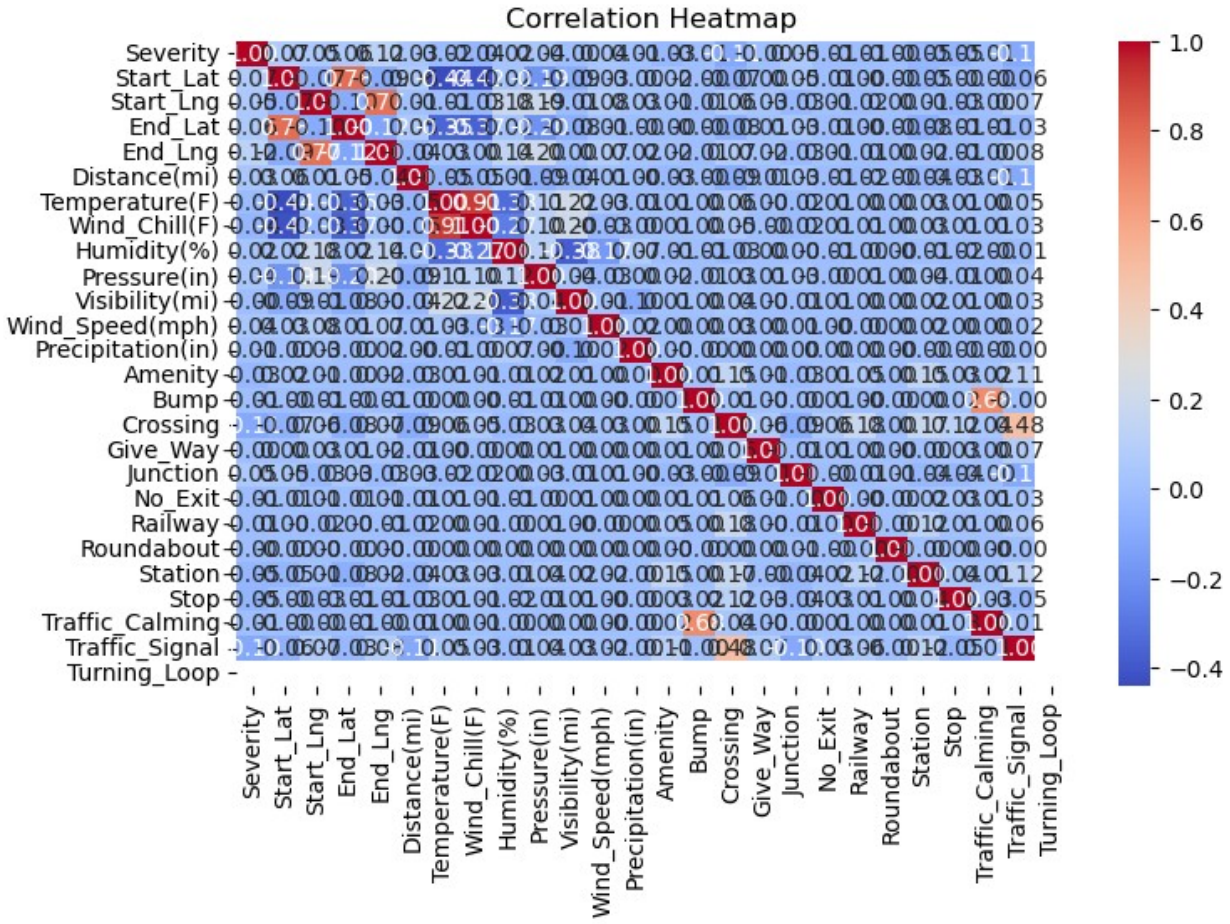

```
plt.figure(figsize=(10,5))
sns.countplot(x='Sunrise_Sunset', data=df, hue='Sunrise_Sunset',
palette='mako')
plt.title('Accidents by Hour of Day')
plt.xlabel('Sunrise_Sunset')
plt.ylabel('Number of Accidents')
plt.show()
```




```
plt.figure(figsize=(8,5))
sns.boxplot(x='Severity', y='Visibility(mi)', data=df)
plt.title('Visibility vs Severity')
plt.show()
```




```
import seaborn as sns
import matplotlib.pyplot as plt

corr = df.corr(numeric_only=True)
plt.figure(figsize=(8, 5))
sns.heatmap(corr, annot=True, cmap='coolwarm', fmt=".2f")
plt.title("Correlation Heatmap")
plt.show()
```


Correlation Heatmap
Severity
Start_Lat
Start_Lng
End_Lat
End_Lng
Distance(mi)
Temperature(F)
Wind_Chill(F)
Humidity(%)
Pressure(in)
Visibility(mi)
Wind_Speed(mph)
Precipitation(in)
Amenity
Bump
Crossing
Give_Way
Junction
No_Exit
Railway
Roundabout
Station
Stop
Traffic_Calming
Traffic_Signal
Turning_Loop
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4