```python
# -*- coding: utf-8 -*-
"""Road Accident Analysis.ipynb

Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/1DsjCCmJ0spR1dDVWiZmt6-lgXlkpvW7s
"""

# Commented out IPython magic to ensure Python compatibility.
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
# %matplotlib inline

import os
os.getcwd()

df1 = pd.read_csv("Accident_Information.csv")

df1.head()

"""DATA CLEANING

"""

df1.isnull().sum()

cols_to_be_dropped = ['1st_Road_Class','1st_Road_Number','2nd_Road_Class','2nd_Road_Number','LSOA_of_Accident_Location']
df1.drop(columns = cols_to_be_dropped,inplace=True)

df1.shape

df1.columns

#Handling all the missing values
df1.dropna(inplace=True)

#checking again for missing values
df1.isnull().sum()

df1.info()

df1['Date'] = pd.to_datetime(df1['Date'],format='%Y-%m-%d')

df1.head()

df1.dtypes

"""# BASIC EDA

### Casualties Accident Type
"""

df1.groupby('Accident_Severity')['Number_of_Casualties'].sum().plot(kind="pie")
plt.show()

"""### Accidents Day Wise"""

df1.groupby('Day_of_Week')['Number_of_Casualties'].sum().plot(kind="bar")
plt.show()

"""### Accidents Month Wise of all Years Together"""

df1['Month'] = df1['Date'].dt.month

p = df1.groupby('Month')['Number_of_Casualties'].sum().reset_index().sort_values('Number_of_Casualties',ascending=False)
p

"""### Line Plot of Casualties with Years"""

p = df1.groupby('Year')['Number_of_Casualties'].sum().reset_index().sort_values('Number_of_Casualties',ascending=False)

sns.set(rc={'figure.figsize':(12,4)})
sns.lineplot(data=p,x='Year',y='Number_of_Casualties')
plt.title('Number of Accidents by year')
plt.show()

"""### Casualties on the basis of Road Type"""

p = df1.groupby('Road_Type')['Number_of_Casualties'].sum().reset_index()
sns.set(rc={'figure.figsize':(12,7)})
sns.barplot(data=p,x='Number_of_Casualties',y='Road_Type')
plt.show()

"""### Casualties on the basis of Road Surface COnditions"""

df1.groupby('Road_Surface_Conditions')['Number_of_Casualties'].sum()

cx = sns.countplot(data = df1,x='Road_Type',hue='Year')
sns.set(rc={'figure.figsize':(20,10)})
plt.show()
```

```
df1['Speed_limit'].nunique()

"""### Casualties on the basis of Speed Limit"""

df1.groupby('Speed_limit')['Number_of_Casualties'].sum().reset_index().sort_values('Number_of_Casualties',ascending=False)

sns.countplot(data=df1,x='Speed_limit')
sns.set(rc={'figure.figsize':(8,6)})
plt.show()

"""### Casualties on the basis of weather conditions"""

p = df1.groupby('Weather_Conditions')['Number_of_Casualties'].sum().reset_index().sort_values('Number_of_Casualties',ascending=False).head(2)
sns.barplot(data=p,x='Number_of_Casualties',y='Weather_Conditions')
plt.show()

"""### Casualties based on time"""

p = df1.groupby('Time')['Number_of_Casualties'].sum().reset_index().sort_values('Number_of_Casualties',ascending=False).head(5)
p

"""### Accidents happen mostly in evenings"""

df1.groupby('Carriageway_Hazards')['Number_of_Casualties'].sum().reset_index().sort_values('Number_of_Casualties',ascending=False)

sns.countplot(data=df1,x='Urban_or_Rural_Area')
plt.show()

"""### Maximum accidents happened in 2005 So analyzing it"""

df_2005 = df1.loc[df1['Year'] == 2005]

df_2005.groupby('Month')['Number_of_Casualties'].sum().reset_index().sort_values('Number_of_Casualties',ascending=False)

"""### November 2005 had the most accidents

### Correlation between the columns
"""

corr = df1.corr()
sns.heatmap(corr,annot=True,fmt='.1f',linecolor='black')
plt.show()
```