

Experiment 2

Aim: Data Visualization/ Exploratory data Analysis using Matplotlib and Seaborn.
Perform following data visualization and exploration on your selected dataset:-

- Create bar graph, contingency table using any 2 features.
- Plot Scatter plot, box plot, Heatmap using seaborn.
- Create histogram and normalized Histogram.
- Describe what this graph and table indicates.
- Handle outlier using box plot and Inter quartile range.

Performance:

- Prerequisite: Import all the required libraries (pandas for data manipulation, numpy for numerical computations, and data visualization using matplotlib for basic plotting and seaborn for enhanced statistical graphics) and load data into Pandas:

Command: import seaborn as sns

import matplotlib.pyplot as plt

import pandas as pd

import numpy as np

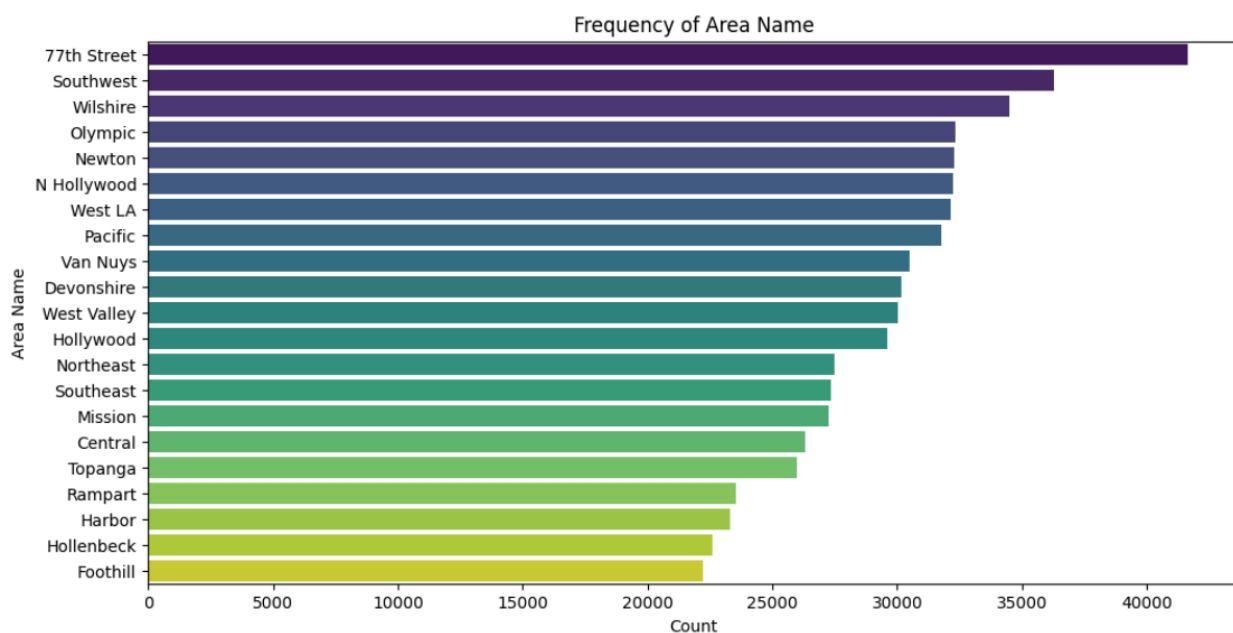
df = pd.read_csv('Traffic_Collision_Data_from_2010_to_Present.csv')

df.head()

	DR Number	Date Reported	Date Occurred	Time Occurred	Area ID	Area Name	Reporting District	Crime Code	Crime Code Description	MO Codes	Victim Age	Victim Sex	Victim Descent	Premise Code	Premise Description	Address	Cross Street	Location
0	190319651	08/24/2019	08/24/2019	450	3	Southwest	356	997	TRAFFIC COLLISION	3036 3004 3026 3101 4003	22.0	M	H	101.0	STREET	JEFFERSON BL	NORMANDIE AV	(34.0255, -118.3002)
1	190319680	08/30/2019	08/30/2019	2320	3	Southwest	355	997	TRAFFIC COLLISION	3037 3006 3028 3030 3039 3101 4003	30.0	F	H	101.0	STREET	JEFFERSON BL	W WESTERN	(34.0256, -118.3089)
2	190413769	08/25/2019	08/25/2019	545	4	Hollenbeck	422	997	TRAFFIC COLLISION	3101 3401 3701 3006 3030	NaN	M	X	101.0	STREET	N BROADWAY	W EASTLAKE AV	(34.0738, -118.2078)
3	190127578	11/20/2019	11/20/2019	350	1	Central	128	997	TRAFFIC COLLISION	0605 3101 3401 3701 3011 3034	21.0	M	H	101.0	STREET	1ST	CENTRAL	(34.0492, -118.2391)
4	190319695	08/30/2019	08/30/2019	2100	3	Southwest	374	997	TRAFFIC COLLISION	0605 4025 3037 3004 3025	49.0	M	B	101.0	STREET	MARTIN LUTHER KING JR	ARLINGTON AV	(34.0108, -118.3182)

- Create bar graph, contingency table using any 2 features:

```
Command: feature_x = "Area Name"
feature_y = "Crime Code Description"
plt.figure(figsize=(12, 6))
sns.countplot(y=df[feature_x], order=df[feature_x].value_counts().index, palette="viridis")
plt.title(f"Frequency of {feature_x}")
plt.xlabel("Count")
plt.ylabel(feature_x)
plt.show()
contingency_table = pd.crosstab(df[feature_x], df[feature_y])
print("Contingency Table:")
print(contingency_table)
```



The above bar graph represents the frequency of vehicle collisions across different areas. The x-axis shows the number of collisions, while the y-axis lists the area names. 77th Street has the highest number of reported collisions, followed by Southwest and Wilshire, while Foothill has the lowest. This suggests that certain areas experience significantly more traffic collisions, which could indicate high traffic density, accident-prone roads, or other contributing factors.

Contingency Table:	
Crime Code Description	TRAFFIC COLLISION
Area Name	
77th Street	41631
Central	26309
Devonshire	30191
Foothill	22215
Harbor	23307
Hollenbeck	22594
Hollywood	29601
Mission	27235
N Hollywood	32259
Newton	32282
Northeast	27508
Olympic	32316
Pacific	31787
Rampart	23541
Southeast	27351
Southwest	36285
Topanga	25979
Van Nuys	30518
West LA	32129
West Valley	30047
Wilshire	34510

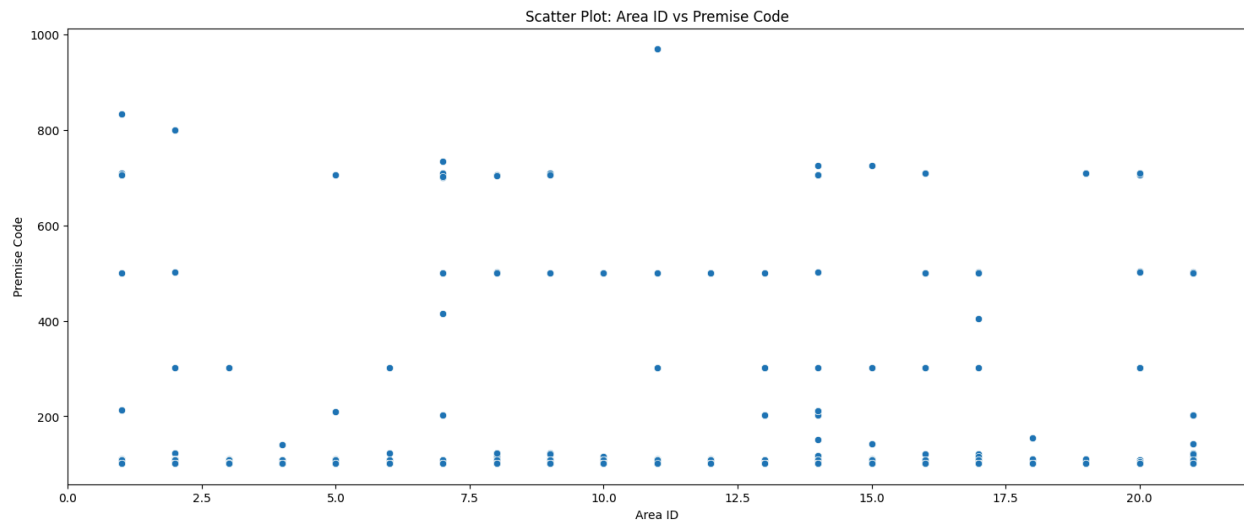
The contingency table displays the number of traffic collisions across different areas. Each row represents an area name, while the corresponding value indicates the number of reported traffic collisions in that area. 77th Street has the highest number of collisions, followed by Wilshire and Southwest, indicating higher traffic incidents in these regions. Conversely, areas like Foothill, Harbor, and Rampart have relatively fewer collisions. This distribution suggests variations in traffic density, road conditions, or reporting frequency across different areas.

- Plot Scatter plot, box plot, Heatmap using seaborn:

1. Scatter plot:-

Command:

```
plt.figure(figsize=(18, 7))
sns.scatterplot(x=df["Area ID"], y=df["Premise Code"])
plt.title("Scatter Plot: Area ID vs Premise Code")
plt.xlabel("Area ID")
plt.ylabel("Premise Code")
plt.show()
```

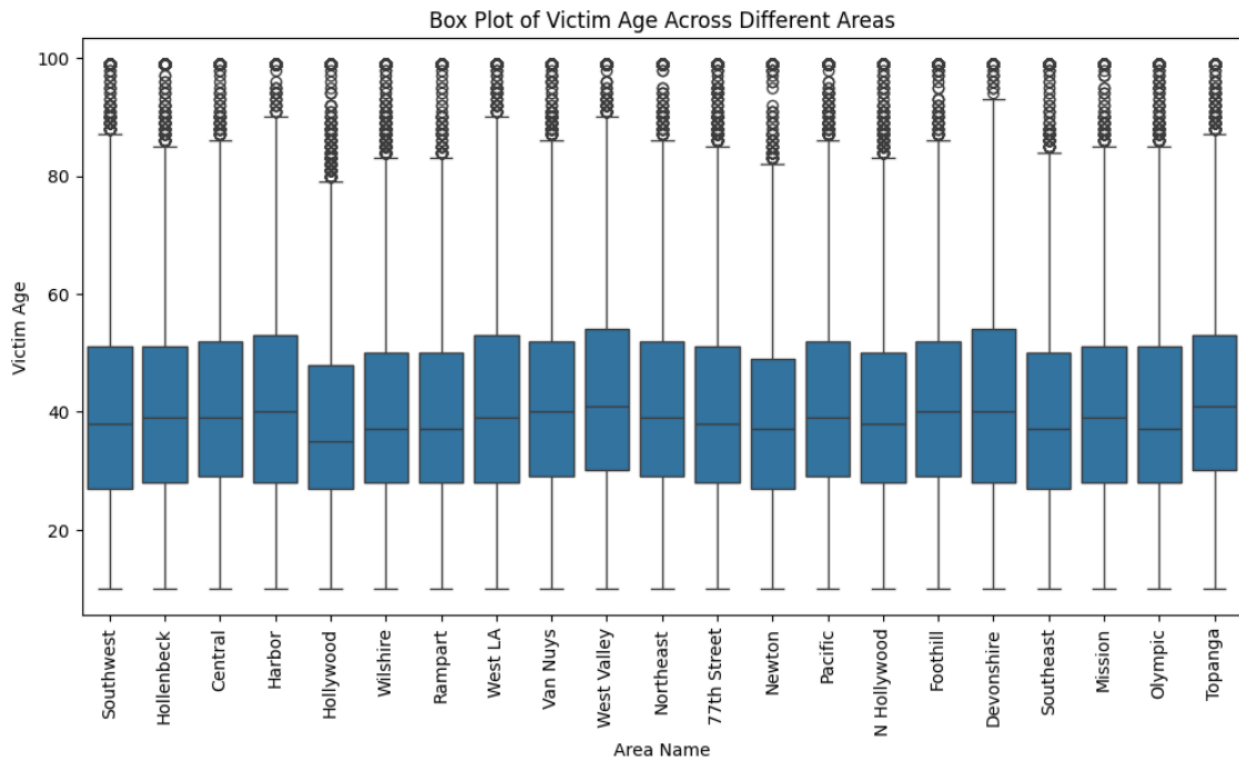


The scatter plot visualizes the relationship between Area ID and Premise Code in vehicle collision data. The x-axis represents different area IDs, while the y-axis represents premise codes, which categorizes the type of location where the collision occurred. The scattered points suggest that collisions happen across various premises in all areas, with some areas showing higher concentrations at specific premise codes. There are a few outliers, indicating locations where collisions are significantly more or less frequent.

2. Box Plot:-

Command:

```
plt.figure(figsize=(12, 6))
sns.boxplot(x=df["Area Name"], y=df["Victim Age"])
plt.xticks(rotation=90)
plt.title("Box Plot of Victim Age Across Different Areas")
plt.xlabel("Area Name")
plt.ylabel("Victim Age")
plt.show()
```

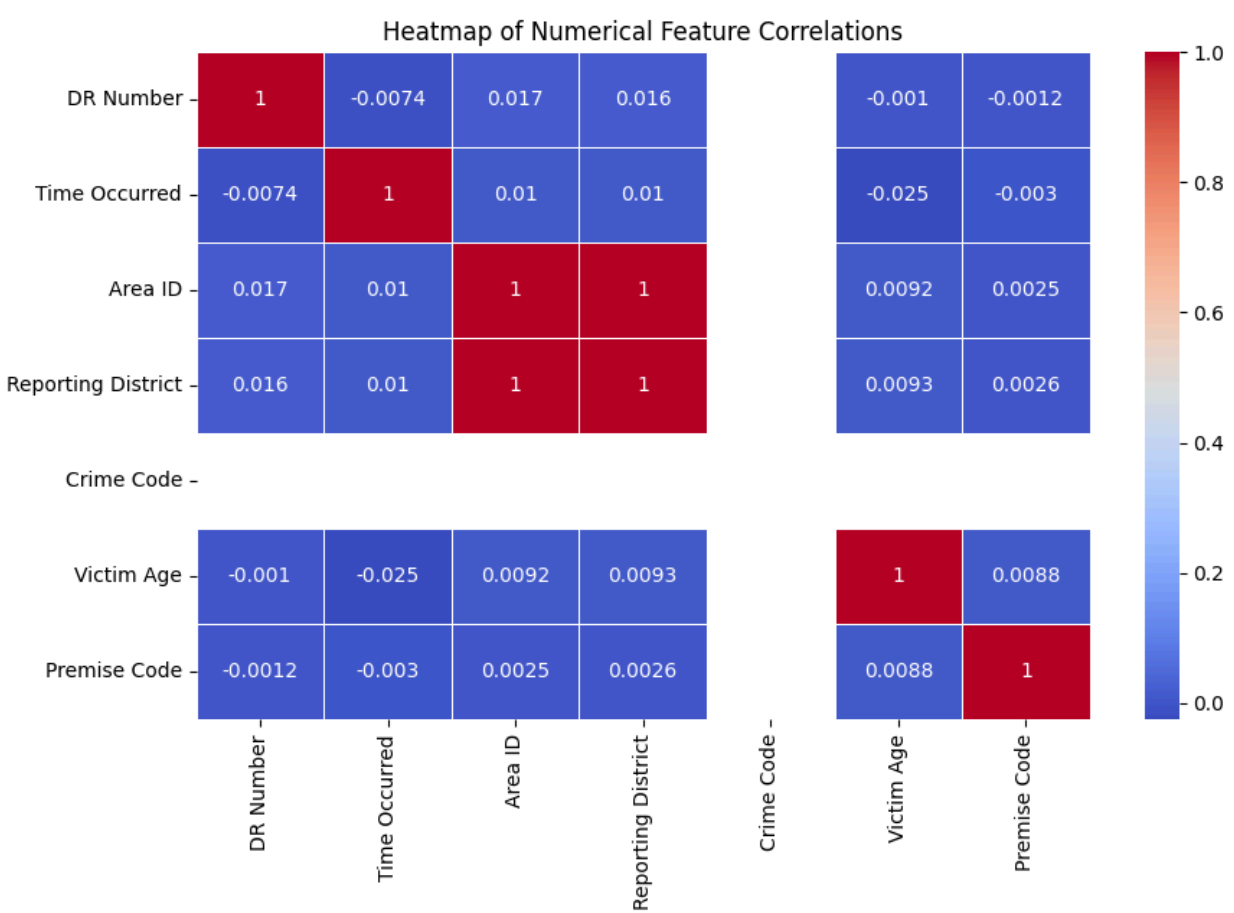


The box plot visualizes the distribution of victim ages across different areas, highlighting variations in age demographics. The median victim age appears to be around 35-45 years in most areas, with interquartile ranges spanning from approximately 25 to 55 years. The whiskers extend towards younger and older victims, with numerous outliers above 80 years, indicating some elderly victims involved in incidents. The overall distribution remains fairly consistent across areas, suggesting similar age patterns in reported cases regardless of location.

3. Heatmap:

Command:

```
plt.figure(figsize=(10, 6))
sns.heatmap(df.select_dtypes(include=np.number).corr(), annot=True,
            cmap="coolwarm", linewidths=0.5)
plt.title("Heatmap of Numerical Feature Correlations")
plt.show()
```



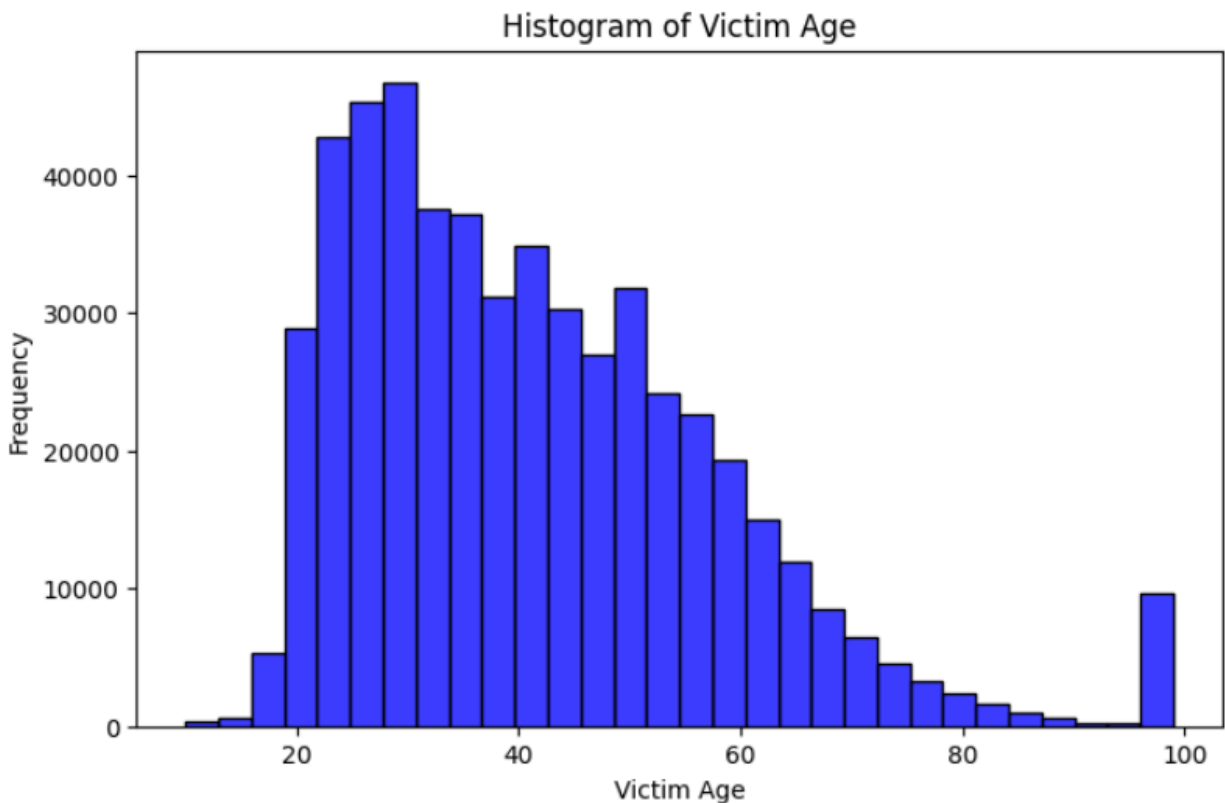
The heatmap visualizes the correlation matrix of numerical features, where values range from -1 to 1. A strong correlation (value = 1) is observed between Area ID and Reporting District, indicating they are closely related. Most other features exhibit weak or near-zero correlations, suggesting minimal linear relationships. Victim Age shows little correlation with Time Occurred and Premise Code, implying that age does not significantly influence when or where incidents occur. Similarly, DR Number and Crime Code have no meaningful correlation with other variables, indicating they function as independent identifiers. Overall, the heatmap suggests that most numerical features are weakly correlated, except for geographical identifiers, which show a strong relationship.

- Create histogram and normalized Histogram:-

1. Histogram:

Command:

```
plt.figure(figsize=(8, 5))  
sns.histplot(df["Victim Age"], bins=30, kde=False, color="blue")  
plt.title("Histogram of Victim Age")  
plt.xlabel("Victim Age")  
plt.ylabel("Frequency")  
plt.show()
```

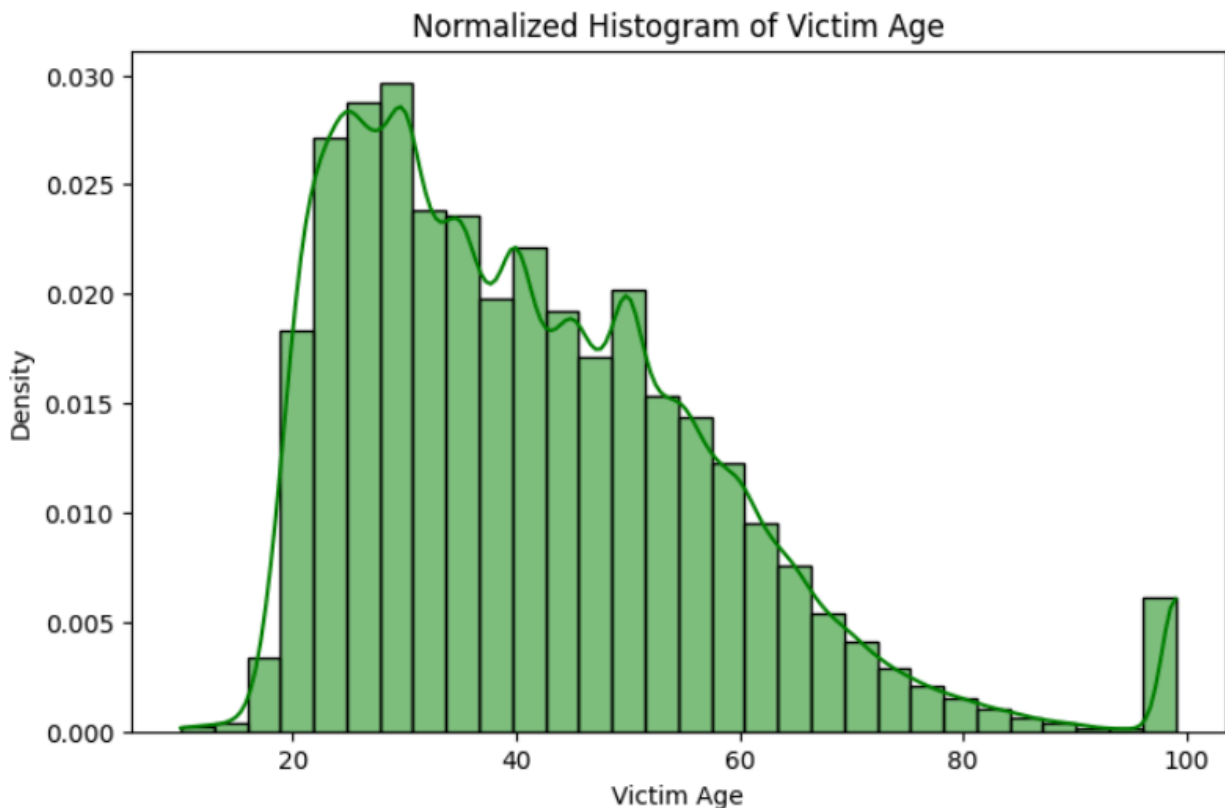


The histogram displays the distribution of Victim Age, showing a right-skewed pattern where most victims fall within the 20 to 40-year-old range, with the highest frequency occurring around the mid-20s to early 30s. The frequency gradually declines as age increases, with fewer incidents reported among victims over 60 years old. A notable spike is observed at 100 years, which may indicate data anomalies or specific reporting issues. Overall, the distribution suggests that younger adults are more frequently involved in incidents, while elderly victims are significantly less common.

2. Normalized Histogram:

Command:

```
plt.figure(figsize=(8, 5))  
sns.histplot(df["Victim Age"], bins=30, kde=True, color="green", stat="density")  
plt.title("Normalized Histogram of Victim Age")  
plt.xlabel("Victim Age")  
plt.ylabel("Density")  
plt.show()
```



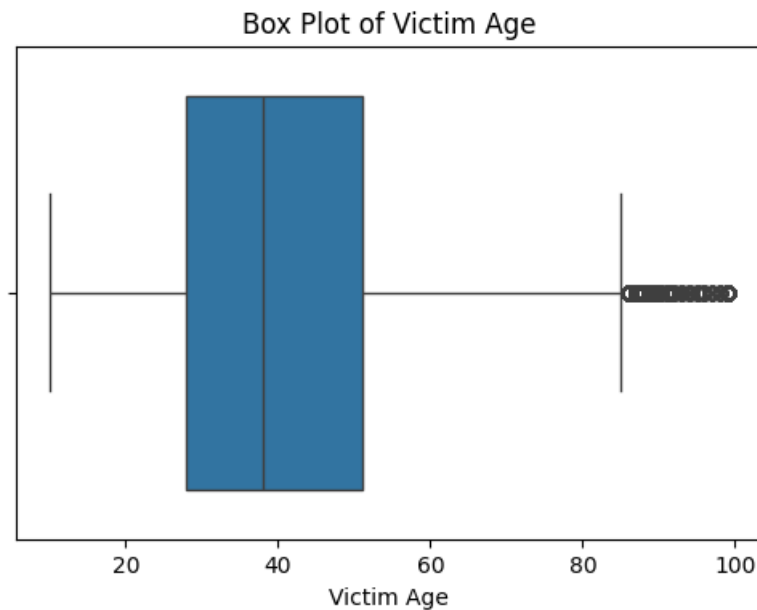
The normalized histogram of Victim Age represents the relative density of victims across different age groups rather than absolute counts, making it easier to compare distributions. The density peaks around the mid-20s to early 30s, indicating that this age range has the highest proportion of victims. As age increases, the density gradually declines, showing fewer cases among older individuals. The kernel density estimate (KDE) curve provides a smooth approximation of the distribution, highlighting fluctuations and reinforcing the overall right-skewed pattern. A small but noticeable spike at 100 years suggests an anomaly or special case in the data.

- Handle outlier using box plot and Inter quartile range:

1. Using box plot:-

Command:

```
plt.figure(figsize=(6, 4))  
sns.boxplot(x=df["Victim Age"])  
plt.title("Box Plot of Victim Age")  
plt.show()
```



The box plot of Victim Age provides a summary of the age distribution, highlighting key statistical measures such as the median, interquartile range (IQR), and outliers. The median age is around the mid-40s, with the interquartile range (IQR) spanning from approximately mid-20s to mid-60s, indicating that the majority of victims fall within this age range. The whiskers extend to the minimum and maximum values within 1.5 times the IQR, while outliers, represented as individual points beyond the whiskers, appear around age 100, suggesting extreme values or possible data anomalies.

To remove outliers using a box plot, one common approach is to filter out values beyond 1.5 times the IQR from both the lower and upper quartiles. This helps in reducing the influence of extreme values on analysis, ensuring a more robust representation of the central data distribution.

2. Using Interquartile range:-

Command:

```
Q1 = df["Victim Age"].quantile(0.25)
```

```
Q3 = df["Victim Age"].quantile(0.75)
```

```
IQR = Q3 - Q1
```

```
lower_bound = Q1 - 1.5 * IQR
```

```
upper_bound = Q3 + 1.5 * IQR
```

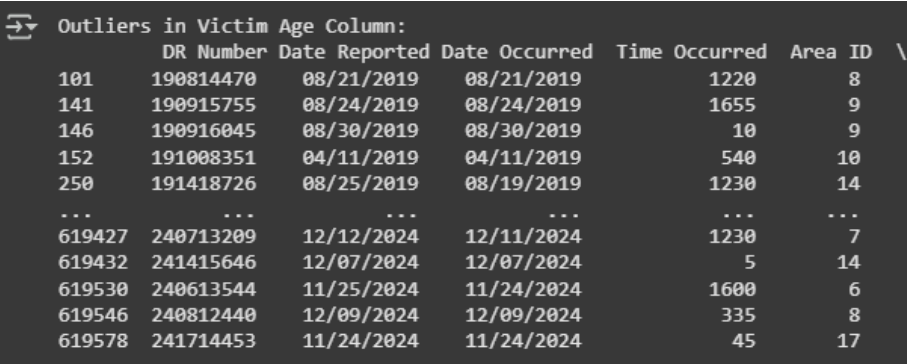
```
outliers = df[(df["Victim Age"] < lower_bound) | (df["Victim Age"] > upper_bound)]
```

```
print("Outliers in Victim Age Column:\n", outliers)
```

```
df_cleaned = df[(df["Victim Age"] >= lower_bound) & (df["Victim Age"] <= upper_bound)]
```

```
print(f"Original dataset size: {df.shape[0]} rows")
```

```
print(f"Dataset size after removing outliers: {df_cleaned.shape[0]} rows")
```



The screenshot shows a Jupyter Notebook interface with a table titled "Outliers in Victim Age Column:". The table has 7 columns: DR Number, Date Reported, Date Occurred, Time Occurred, Area, and ID. It lists several rows of data, including some with ellipses indicating more rows. The rows shown are:

	DR Number	Date Reported	Date Occurred	Time Occurred	Area	ID
101	190814470	08/21/2019	08/21/2019	1220	8	
141	190915755	08/24/2019	08/24/2019	1655	9	
146	190916045	08/30/2019	08/30/2019	10	9	
152	191008351	04/11/2019	04/11/2019	540	10	
250	191418726	08/25/2019	08/19/2019	1230	14	
...
619427	240713209	12/12/2024	12/11/2024	1230	7	
619432	241415646	12/07/2024	12/07/2024	5	14	
619530	240613544	11/25/2024	11/24/2024	1600	6	
619546	240812440	12/09/2024	12/09/2024	335	8	
619578	241714453	11/24/2024	11/24/2024	45	17	

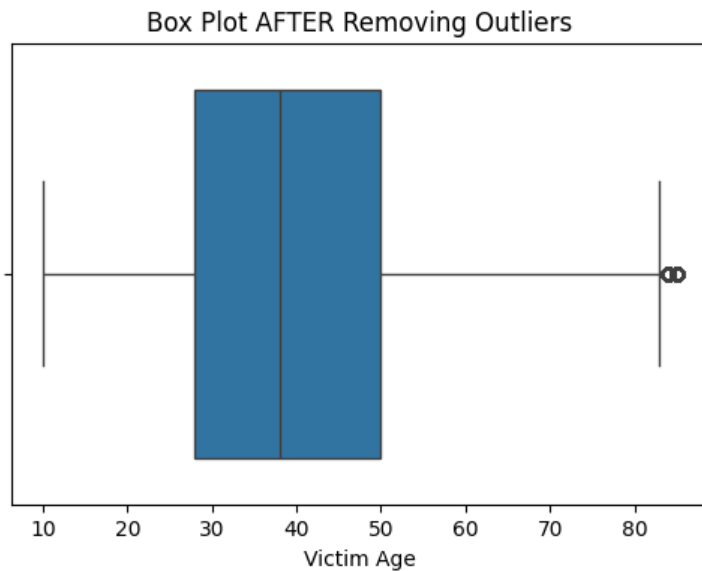
```
[11396 rows x 18 columns]
```

```
Original dataset size: 619595 rows
```

```
Dataset size after removing outliers: 520295 rows
```

The Interquartile Range (IQR) is used to detect outliers by measuring the spread of the middle 50% of the data. First, the first quartile (Q1) and third quartile (Q3) are calculated, representing the 25th and 75th percentiles, respectively. The IQR is then determined as the difference between Q3 and Q1 ($IQR = Q3 - Q1$). To identify outliers, a lower bound is set at $Q1 - 1.5 * IQR$, and an upper bound is set at $Q3 + 1.5 * IQR$. Any values outside this range are considered outliers and can be removed from the dataset. The code first extracts these outliers and then filters the dataset to retain only values within the acceptable range, resulting in a cleaned dataset with reduced extreme values.

Box plot after removing outliers:



The box plot after removing outliers provides a clearer representation of the central distribution of victim ages without extreme values distorting the spread. Compared to the original box plot, the whiskers now extend only to the adjusted lower and upper bounds, ensuring that only values within the $1.5 \times \text{IQR}$ range are included. While a few mild outliers may still be present, the overall data distribution appears more compact and balanced. This refinement helps in more accurate analysis by reducing the influence of extreme values while preserving the essential characteristics of the dataset.

Conclusion:

1. In this experiment, we learned about Data Visualization / Exploratory Data Analysis using Matplotlib and Seaborn.
2. The bar graph showed that 77th Street and Wilshire have the highest number of collisions, likely due to high traffic density, accident-prone roads, or urban congestion.
3. The scatter plot indicated that some areas experience more incidents at specific premises, with a few outliers suggesting unusual collision patterns in certain locations.
4. The contingency table confirmed that certain areas, such as 77th Street and Wilshire, report significantly more collisions, reinforcing the findings from the bar graph analysis.
5. The box plot revealed that most victims are between 25-55 years old, while outliers above 80 suggest that elderly individuals are occasionally involved in incidents.
6. The heatmap showed that most numerical features have weak correlations, except for strong relationships between geographical identifiers like Area ID and Reporting District.
7. The histogram displayed a right-skewed victim age distribution, with the highest frequency in the mid-20s to early 30s, gradually declining for older age groups.

8. Outlier removal using the IQR method helped refine data accuracy by eliminating extreme victim ages that could distort statistical analysis and overall conclusions.