Name: Shivam.R.Prajapati **Div: D15C** **Roll No: 41**

Experiment No: 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.

THEORY:

1] Data Exploration:

Data exploration is the **process of analyzing a dataset** to understand its characteristics, patterns, and relationships between variables before applying any machine learning or statistical techniques. It involves:

- **Summary statistics** (mean, median, mode, standard deviation, etc.)
- **Missing values analysis**
- **Identifying outliers**
- **Checking data distribution**
- **Feature correlations**

2] Data Visualization:

Data visualization is the **graphical representation of data** using charts, graphs, and plots to **help identify trends, patterns, and insights**. It is an essential part of exploratory data analysis (EDA). Common visualization techniques include:

- **Bar charts** (for categorical data)
- **Histograms** (to show data distribution)
- **Box plots** (for detecting outliers)
- **Scatter plots** (to analyze relationships between variables)
- **Heatmaps** (to visualize correlations)

Name: Shivam.R.Prajapati **Div: D15C** **Roll No: 41**

DATASET:

The dataset contains **619,595 records of traffic collisions**, with **18 columns** detailing various attributes such as **the date and time of occurrence, area name, crime code, victim details (age, sex, descent), premise description, and location coordinates**. The dataset includes missing values in some columns, particularly "Victim Age," "Victim Sex," and "MO Codes." The majority of incidents are categorized under "TRAFFIC COLLISION," and the data spans multiple areas, providing a comprehensive overview of traffic-related crimes in different locations in **Los Angeles**.

STEPS:

Step 1: Loading Dataset in Google Colab and then displaying a few instances of it.

```
"/content/Traffic_Collision_Data_from_2010_to_Present (2).csv"
```

```
'/content/Traffic_Collision_Data_from_2010_to_Present (2).csv'
```

```
[ ] import pandas as pd
    df=pd.read_csv('/content/Traffic_Collision_Data_from_2010_to_Present (2).csv')
```

This step **loads a CSV file into a Pandas DataFrame** in Google Colab. The file is stored in /content/, and **pd.read_csv() reads it into df** for data analysis.

```
[ ] 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 3101	49.0	M	B	101.0	STREET	MARTIN LUTHER KING JR	ARLINGTON AV	(34.0108, -118.3182)

This is the output of **df.head()**, which displays the **first five rows of the dataset**. It includes details such as the DR number, date reported, date occurred, time occurred, area information, crime description, victim details, and location coordinates. **This helps in understanding the structure of the dataset.**

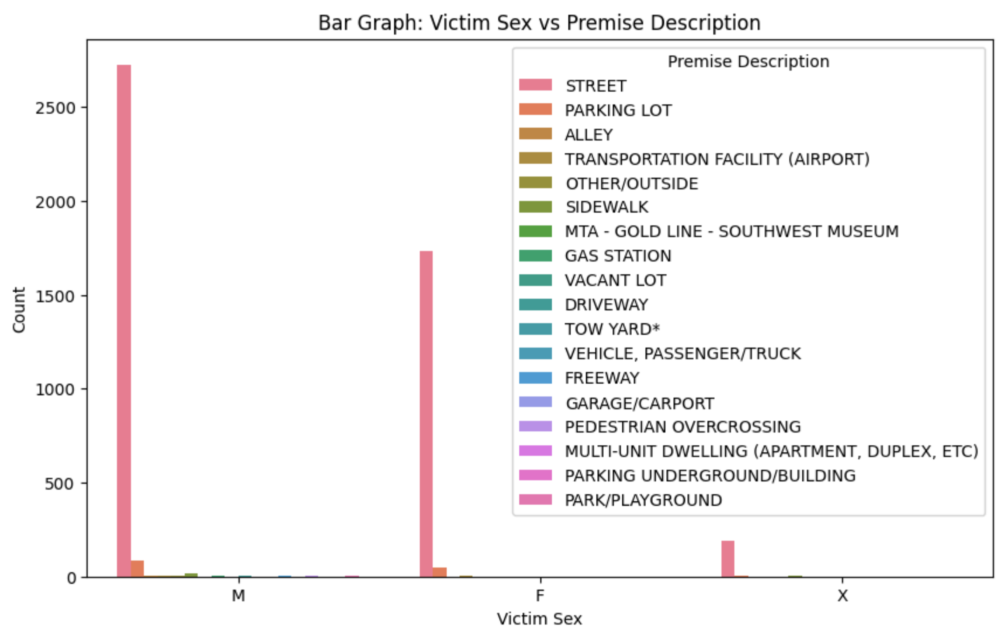
Name: Shivam.R.Prajapati **Div: D15C** **Roll No: 41**

Step 2: Bar Graph Analysis and Contingency table

Code:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
plt.figure(figsize=(10,6))
sns.countplot(data=df, x='Victim Sex', hue='Premise Description')
plt.title('Bar Graph: Victim Sex vs Premise Description')
plt.xlabel('Victim Sex')
plt.ylabel('Count')
plt.show()

contingency_table = pd.crosstab(df['Victim Sex'], df['Premise Description'])
print("Contingency Table:\n", contingency_table)
```

Output:



The bar graph indicates that most incidents occur on **streets**, with a significantly higher number of **male victims** compared to females. Female victims are notably fewer, and cases with an unspecified sex ("X") are minimal. While streets dominate as the primary location, other premises like **parking lots, sidewalks, and multi-unit dwellings** also contribute to a smaller extent. This suggests that public spaces are more prone to such incidents, with males being the most affected group.

Contingency Table:

```
Contingency Table:
 Premise Description  ALLEY  DRIVEWAY  FREEWAY  GARAGE/CARPORT  GAS STATION  \
Victim Sex
F                         1         1        0               1            0
M                         6         2        3               1            4
X                         0         0        0               0            0

Premise Description  MTA - GOLD LINE - SOUTHWEST MUSEUM  \
Victim Sex
F                                                     0
M                                                     1
X                                                     0

Premise Description  MULTI-UNIT DWELLING (APARTMENT, DUPLEX, ETC)  \
Victim Sex
F                                                               1
M                                                               0
X                                                               0

Premise Description  OTHER/OUTSIDE  PARK/PLAYGROUND  PARKING LOT  \
Victim Sex
F                                0                1           47
M                                2                2           85
X                                0                0            6
```

```
Premise Description  PARKING UNDERGROUND/BUILDING  PEDESTRIAN OVERCROSSING  \
Victim Sex
F                                               1                        0
M                                               0                        2
X                                               0                        0

Premise Description  SIDEWALK  STREET  TOW YARD*  \
Victim Sex
F                           0    1732          1
M                          17    2725          0
X                           3     188          0

Premise Description  TRANSPORTATION FACILITY (AIRPORT)  VACANT LOT  \
Victim Sex
F                                                    2           0
M                                                    2           1
X                                                    1           0

Premise Description  VEHICLE, PASSENGER/TRUCK
Victim Sex
F                                           0
M                                           1
X                                           0
```

The contingency table shows the distribution of victim sex across different premises where incidents occurred. Streets have the highest number of cases, with **2,725 male victims**, **1,732 female victims**, and **188 unknown cases**. Parking lots also see significant incidents, while locations like alleys, driveways, and gas stations have fewer cases. Certain places, such as sidewalks, involve only male and unknown victims, with no female cases reported. The presence of "X" (unknown sex) cases is minimal, appearing mainly in streets and parking lots. This table provides a clear numerical insight into crime distribution by location and gender.

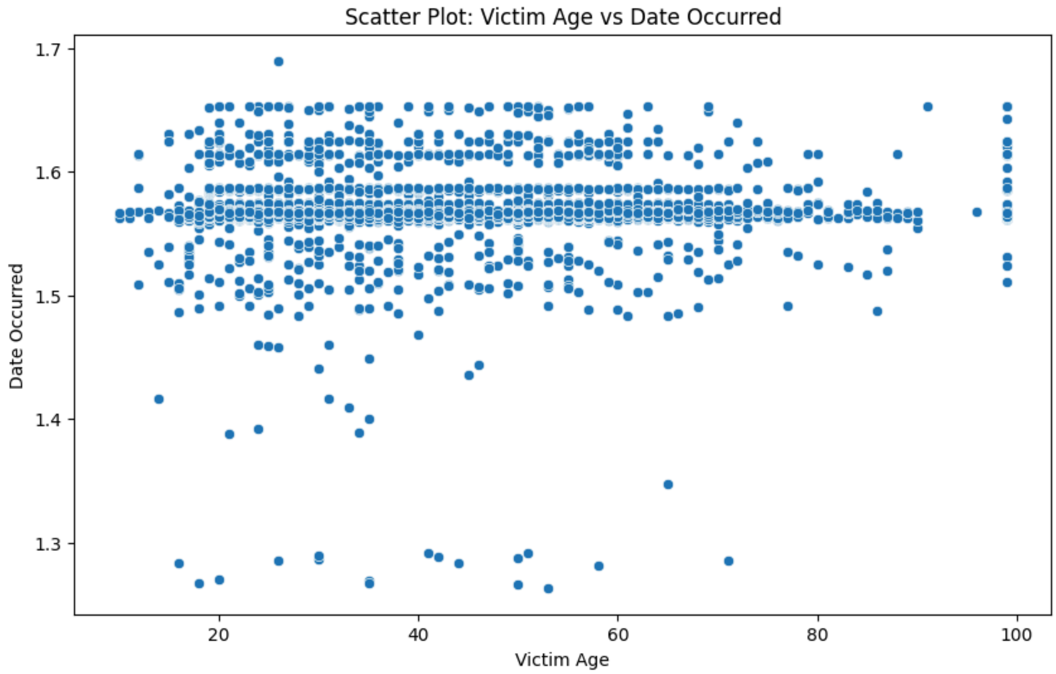
Step 3: Scatter Plot and Analysis

Code:

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

df['Date Occurred'] = pd.to_datetime(df['Date Occurred'], errors='coerce')
df['Date Occurred'] = df['Date Occurred'].map(pd.Timestamp.timestamp)

plt.figure(figsize=(10,6))
sns.scatterplot(data=df, x='Victim Age', y='Date Occurred')
plt.title('Scatter Plot: Victim Age vs Date Occurred')
plt.xlabel('Victim Age')
plt.ylabel('Date Occurred')
plt.show()
```

Name: Shivam.R.Prajapati **Div: D15C** **Roll No: 41**

Output:



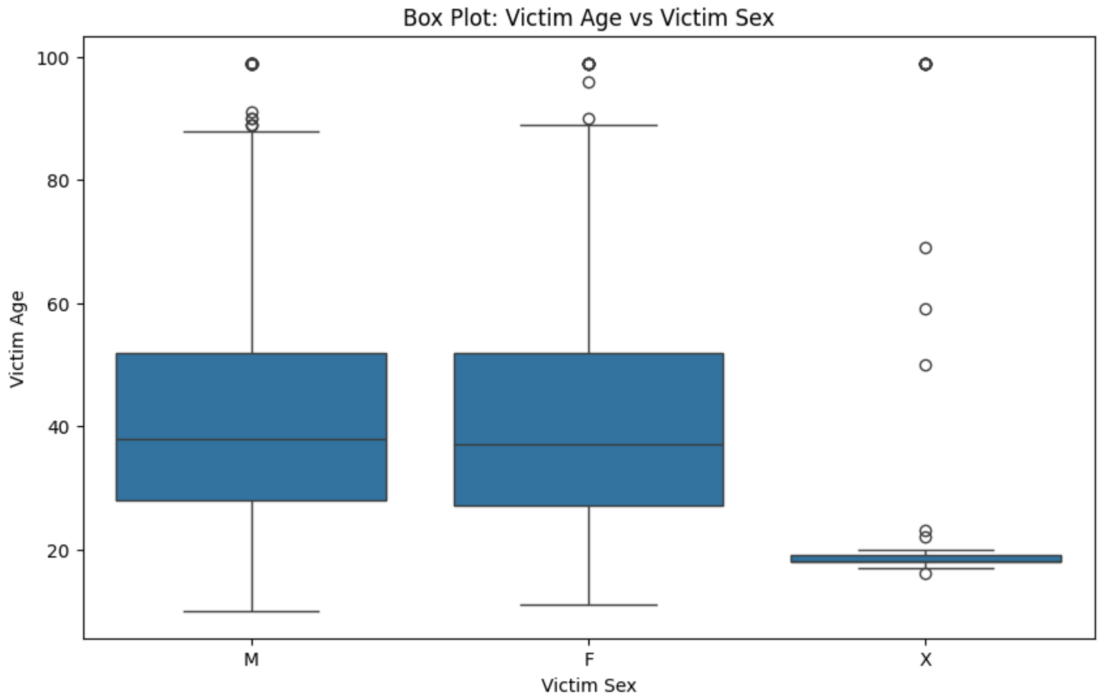
The scatter plot represents the relationship between **victim age** and **date occurred**, showing incidents across different age groups. Most victims fall between **20 to 60 years**, with a dense cluster of points indicating frequent incidents in this range. There are a few cases involving younger and older victims, but they are relatively scattered. The distribution of incidents appears consistent across time, with no clear trend linking age and date. Some outliers exist, suggesting isolated cases outside the common age range. Overall, incidents occur across all ages without a strong correlation to time.

Step 4: Box Plot

Code:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
plt.figure(figsize=(10,6))
sns.boxplot(data=df, x='Victim Sex', y='Victim Age')
plt.title('Box Plot: Victim Age vs Victim Sex')
plt.xlabel('Victim Sex')
plt.ylabel('Victim Age')
plt.show()
```

Output:



The box plot shows the distribution of **victim age** across different **victim sex** categories (M, F, X). The median age for both **male (M) and female (F) victims** appears similar, around **30-40 years**, with a fairly wide interquartile range. Both categories have **outliers above 80 years**, indicating some elderly victims. The **X category**, likely representing unknown or non-binary gender, has a much smaller age range, with most victims concentrated at a lower age. The presence of outliers in all groups suggests occasional incidents involving victims outside the typical age range.

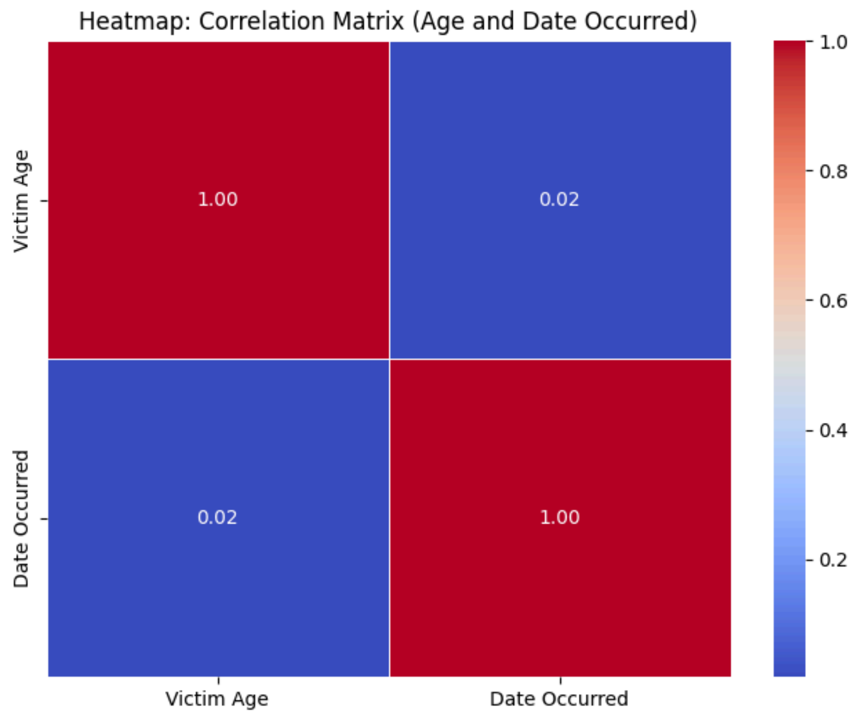
Step 5: Heat Map

Code:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
numerical_columns = df[['Victim Age', 'Date Occurred']]
corr_matrix = numerical_columns.corr()
plt.figure(figsize=(8,6))
```

```
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', fmt='.2f', linewidths=0.5)
plt.title('Heatmap: Correlation Matrix (Age and Date Occurred)')
plt.show()
```

Output:



A **heatmap** is a **data visualization technique** that **represents numerical values using colors to show relationships between variables**. It is commonly used for correlation matrices, where the color intensity indicates the strength and direction of relationships between variables. **Darker or warmer colors (e.g., red) usually indicate stronger positive correlations, while cooler colors (e.g., blue) represent weaker or negative correlations.**

The heatmap shows the correlation matrix between **Victim Age** and **Date Occurred**. The diagonal values (1.00) indicate that each variable is perfectly correlated with itself. The off-diagonal value (0.02) represents the correlation between **Victim Age** and **Date Occurred**, which is very close to zero. This suggests that there is **no significant relationship** between a victim's age and the date of occurrence, meaning that crimes are not influenced by age trends over time.

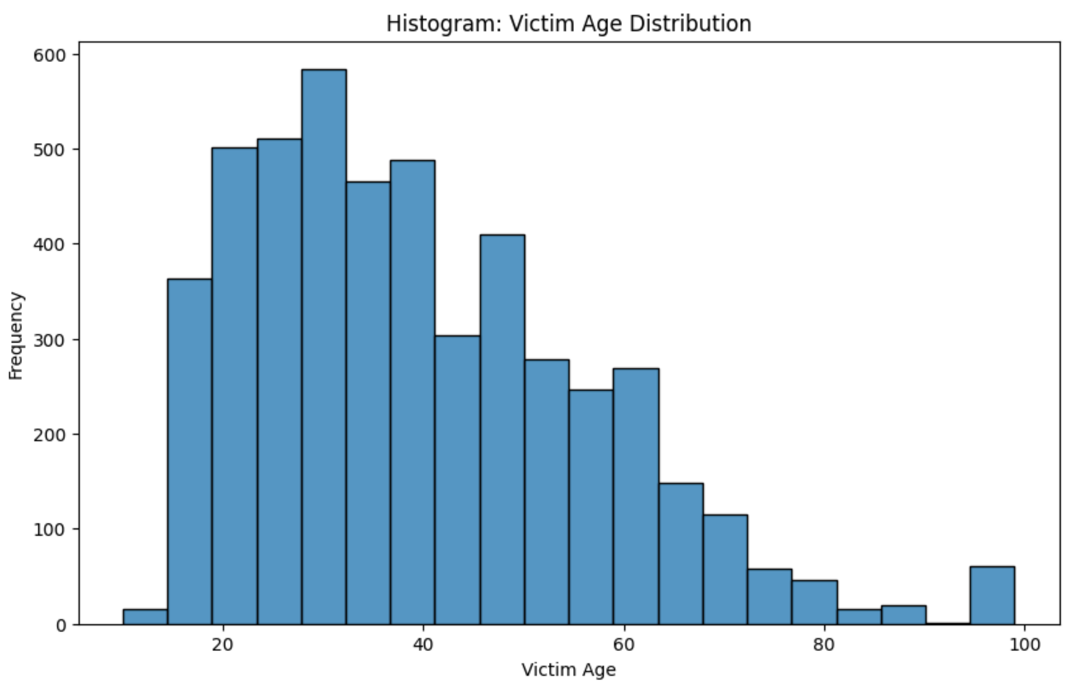
Name: Shivam.R.Prajapati **Div: D15C** **Roll No: 41**

Step 6: Histogram

Code:

```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(10,6))
sns.histplot(df['Victim Age'], kde=False, bins=20)
plt.title('Histogram: Victim Age Distribution')
plt.xlabel('Victim Age')
plt.ylabel('Frequency')
plt.show()
```

Output:



The histogram represents the distribution of victim ages. The x-axis shows **Victim Age**, while the y-axis represents **Frequency** (number of occurrences). The distribution is right-skewed, with most victims falling in the **20–40 age range**, peaking around **25–30 years**. The frequency gradually decreases for older age groups, with fewer victims above **60 years**. There are some outliers around **100 years old**, but they are rare.

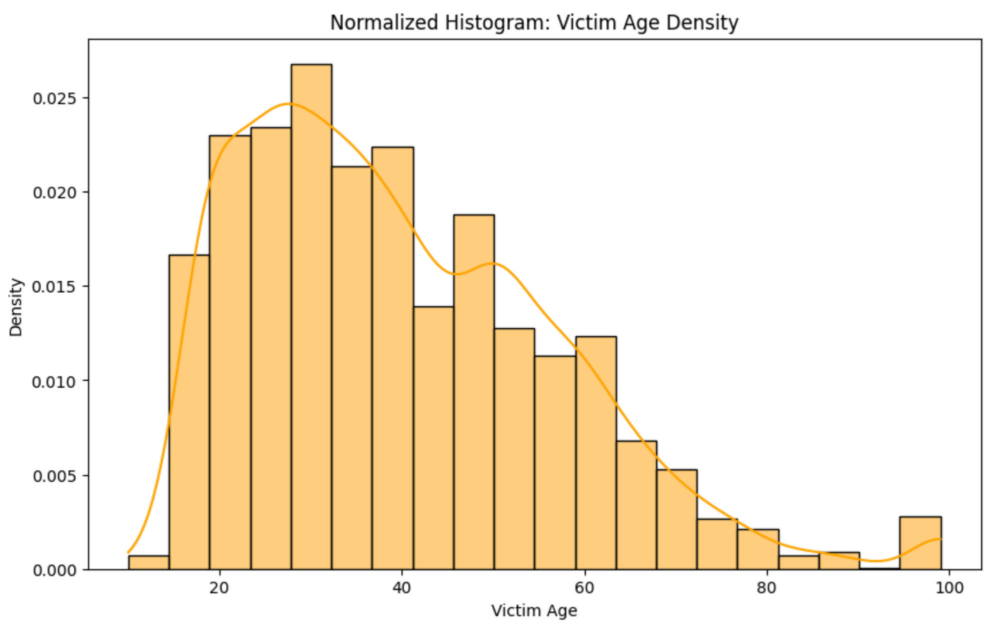
Name: Shivam.R.Prajapati **Div: D15C** **Roll No: 41**

Step 7: Normalized Histogram

Code:

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

plt.figure(figsize=(10,6))
sns.histplot(df['Victim  Age'], kde=True, bins=20, stat='density', color='orange')
plt.title('Normalized Histogram: Victim Age Density')
plt.xlabel('Victim Age')
plt.ylabel('Density')
plt.show()
```

Output:



A **normalized histogram** represents data in terms of probability density instead of raw frequency.

The graph shows the **normalized distribution of victim ages** with a histogram and a smooth **KDE (Kernel Density Estimation) curve** overlaid. The x-axis represents **Victim Age**, while the y-axis represents **Density**. The distribution is **right-skewed**, with a peak around **25–30 years**, indicating that most victims fall within this range. The density decreases gradually for older age groups, with very few victims above **80 years**. A small rise near **100 years** suggests a few outliers. The KDE curve provides a smooth estimate of the underlying distribution.

Name: Shivam.R.Prajapati **Div: D15C** **Roll No: 41**

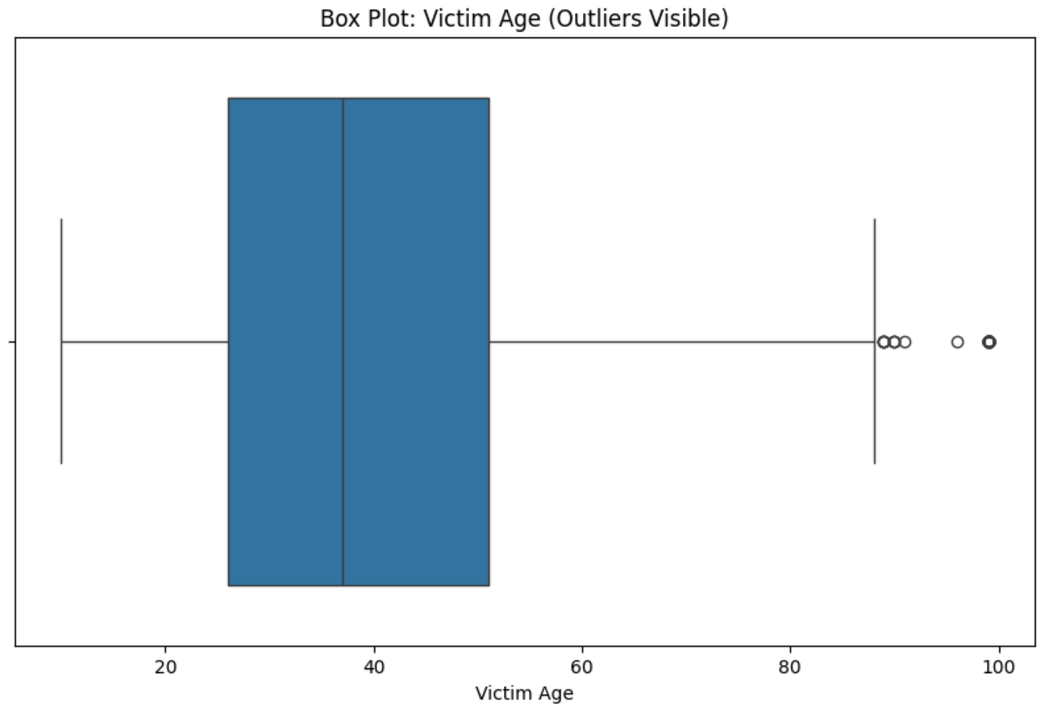
Step 8: Handle outlier using box plot and Inter quartile range

Code:

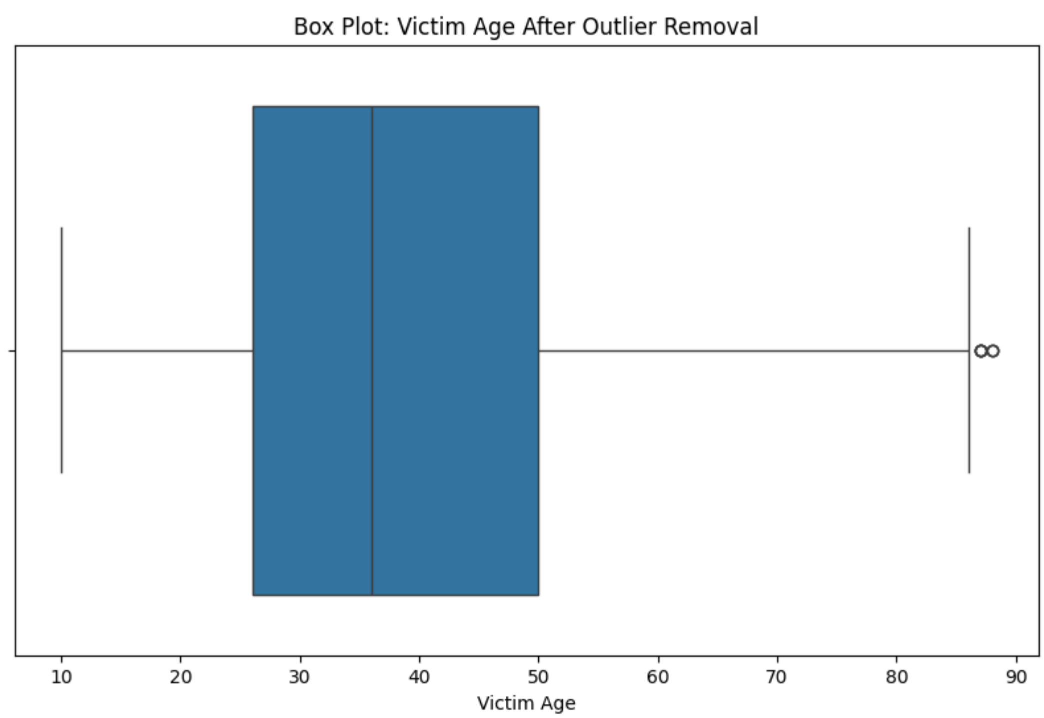
```
plt.figure(figsize=(10,6))

sns.boxplot(data=df, x='Victim Age')

plt.title('Box Plot: Victim Age (Outliers Visible)')

plt.show()

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

df_filtered = df[(df['Victim Age'] >= lower_bound) & (df['Victim Age'] <= upper_bound)]

plt.figure(figsize=(10,6))

sns.boxplot(data=df_filtered, x='Victim Age')

plt.title('Box Plot: Victim Age After Outlier Removal')

plt.show()
```

Output:

1] Before removing Outliers



2] After removing Outliers



The **first box plot** shows significant outliers, mainly on the higher end (ages above 80), indicating extreme values that need handling. The median victim age is around 40 years, with an interquartile range (IQR) from approximately 25 to 60 years. **After applying the IQR method in the second box plot, most extreme outliers have been removed** while maintaining a similar data spread. Though a few mild outliers remain near the upper limit (~80 years), they are closer to the whisker line and can be ignored as they do not significantly impact the analysis.

CONCLUSION:

This experiment helped us explore and visualize patterns in traffic collision data, focusing on Victim Age, Victim Sex, and Accident Location. By using various plots like bar graphs, scatter plots, and heatmaps, we analyzed how factors like gender and age relate to accident locations and timings. The bar graph and contingency table revealed whether certain genders are more involved in accidents in specific places, while the scatter plot and box plot helped identify trends in accident occurrence based on age and time of year.

The heatmap provided insight into the relationship between Victim Age and Date Occurred, showing whether age influences the timing of accidents. These analyses are crucial in identifying patterns and trends that can aid in making data-driven decisions for road safety improvements. Overall, the experiment demonstrated how different factors in the dataset interact and how visualization techniques can uncover valuable insights into traffic collisions.