**Presentation: Traffic Accident Analysis Using Python**

**Slide 1: Title Slide**

**Title: Traffic Accident Analysis Using Python**

**Subtitle:  
Exploring Patterns, Trends, and Insights in Traffic Accident Data**

**Presented By: Youhem Rouainia**

**Slide 2: Introduction**

**Objective:**

* **Analyse traffic accident data to uncover patterns and trends.**
* **Visualize key insights using Python libraries such as Pandas, Matplotlib, and Seaborn.**

**Dataset Overview:**

* **File: traffic\_accidents.csv**
* **Columns: Includes crash types, weather conditions, lighting conditions, injury severity, and more.**

**Key Questions:**

1. **What are the most common crash types?**
2. **How do weather and lighting conditions affect crashes?**
3. **What is the distribution of injuries across accidents?**
4. **Are there specific times or days with higher crash rates?**

**Slide 3: Data Preprocessing**

**Steps Taken:**

1. **Data Loading:**
   * **Used pandas.read\_csv() to load the dataset.**
   * **Handled encoding issues with unicode\_escape.**
2. **Missing Values:**
   * **Checked for missing values using df.isnull().sum().**
   * **Filled missing categorical values with 'UNKNOWN' using fillna().**
3. **Categorical Conversion:**
   * **Converted relevant columns (e.g., weather\_condition, crash\_type) to categorical data types for efficient analysis.**

**Slide 4: Crash Types Distribution**

**Visualization:**

* **Bar Chart showing the distribution of crash types.**

**Insights:**

* **Example: "Rear-end collisions account for the majority of crashes."**

**Python Code**

**1**

**2**

**3**

**4**

**crash\_type\_counts = df['crash\_type'].value\_counts()**

**sns.barplot(x=crash\_type\_counts.index, y=crash\_type\_counts.values, palette='viridis')**

**plt.title('Crash Types Distribution')**

**plt.show()**

**Slide 5: Most Severe Injuries**

**Visualization:**

* **Pie Chart showing the percentage distribution of injury severity.**

**Insights:**

* **Example: "Fatal injuries make up only 5% of all reported injuries."**

**Python Code**

**1**

**2**

**3**

**4**

**injury\_counts = df['most\_severe\_injury'].value\_counts()**

**injury\_counts.plot.pie(autopct='%1.1f%%', startangle=90, colors=sns.color\_palette("pastel"))**

**plt.title('Most Severe Injuries')**

**plt.show()**

**Slide 6: Crashes by Weather Condition**

**Visualization:**

* **Bar Chart showing the number of crashes under different weather conditions.**

**Insights:**

* **Example: "Clear weather accounts for the highest number of crashes, followed by rainy conditions."**

**Python Code**

**1**

**2**

**3**

**4**

**weather\_counts = df['weather\_condition'].value\_counts()**

**sns.barplot(x=weather\_counts.index, y=weather\_counts.values, palette='coolwarm')**

**plt.title('Crashes by Weather Condition')**

**plt.show()**

**Slide 7: Crashes by Lighting Condition**

**Visualization:**

* **Bar Chart showing the number of crashes under different lighting conditions.**

**Insights:**

* **Example: "Daylight conditions see the highest number of crashes, while nighttime crashes are less frequent."**

**Python Code**

**1**

**2**

**3**

**4**

**lighting\_counts = df['lighting\_condition'].value\_counts()**

**sns.barplot(x=lighting\_counts.index, y=lighting\_counts.values, palette='magma')**

**plt.title('Crashes by Lighting Condition')**

**plt.show()**

**Slide 8: Crashes by Hour of the Day**

**Visualization:**

* **Line Chart showing the number of crashes by hour of the day.**

**Insights:**

* **Example: "Crash frequency peaks during rush hours (7-9 AM and 4-6 PM)."**

**Python Code**

**1**

**2**

**3**

**4**

**hourly\_crashes = df.groupby('crash\_hour').size()**

**hourly\_crashes.plot(kind='line', marker='o', color='blue')**

**plt.title('Crashes by Hour of the Day')**

**plt.show()**

**Slide 9: Crashes by Day of the Week**

**Visualization:**

* **Bar Chart showing the number of crashes by day of the week.**

**Insights:**

* **Example: "Friday has the highest number of crashes, while Sunday sees the lowest."**

**Python Code**

**1**

**2**

**3**

**4**

**weekly\_crashes = df.groupby('crash\_day\_of\_week').size()**

**sns.barplot(x=weekly\_crashes.index, y=weekly\_crashes.values, palette='plasma')**

**plt.title('Crashes by Day of the Week')**

**plt.show()**

**Slide 10: Correlation Matrix**

**Visualization:**

* **Heatmap showing the correlation between different injury metrics.**

**Insights:**

* **Example: "Fatal injuries are strongly correlated with incapacitating injuries."**

**Python Code**

**1**

**2**

**3**

**4**

**correlation\_matrix = df[['injuries\_total', 'injuries\_fatal', 'injuries\_incapacitating']].corr()**

**sns.heatmap(correlation\_matrix, annot=True, cmap='coolwarm', fmt='.2f')**

**plt.title('Correlation Matrix of Injury Metrics')**

**plt.show()**

**Slide 11: Conclusion**

**Summary of Findings:**

1. **Rear-end collisions are the most common crash type.**
2. **Clear weather and daylight conditions contribute to most crashes.**
3. **Fatal injuries are rare but highly correlated with severe injuries.**
4. **Rush hours and Fridays are peak times for crashes.**

This Python script provides a comprehensive overview of traffic accident data through systematic preprocessing and insightful visualizations. It lays the groundwork for deeper analysis and actionable recommendations to improve road safety.

Final Deliverable: The code successfully combines data cleaning, exploratory analysis, and visualization to uncover meaningful patterns in traffic accident data.