



SCHOOL OF
COMPUTER
SCIENCE AND IT

ROAD ACCIDENT ANALYSIS IN INDIA

Department of CS&IT

IV Semester

SUBJECT CODE: 23BCAM4CO3

SUBJECT NAME: PYTHON PROGRAMMING

Activity 2

MINI PROJECT

SPECIALIZATION:

MOBILE COMPUTING AND CLOUD TECHNOLOGY

SUBMITTED BY:

PALAK R – 23BCAR0464

SPOORTHI KB – 23BCAR0480

DATE OF SUBMISSION: 12.04.2025

NAME OF FACULTYIN-CHARGE: Mr. Sahabzada Betab
Badar



EVALUATION CRITERIA

Report 25 M	Viva voce 10 M	On-time Submission 5M	Creativity & additional hours spent 10 M	Total 50M	Converted 15 M
Introduction - /10					
Applications - /5					
Case Study- /10					
Total - /25					

This is to certify that Palak R has satisfactorily completed the course of assignment prescribed by the Jain Deemed to be University for the Fourth semester Bachelors of Computer Applications in Python Programming Degree Course in the year 20242025.

MARKS	
MAX	OBTAINED
15	



EVALUATION CRITERIA

Report 25 M	Viva voce 10 M	On-time Submission 5M	Creativity & additional hours spent 10 M	Total 50M	Converted 15 M
Introduction -	/10				
Applications -	/5				
Case Study-	/10				
Total -	/25				

This is to certify that Spoorthi K B has satisfactorily completed the course of assignment prescribed by the Jain Deemed to be University for the Fourth semester Bachelors of Computer Applications in Python Programming Degree Course in the year 2024-2025.

MARKS	
MAX	OBTAINED
15	



TABLE OF CONTENTS

SI.No	Contents	Page No.
1	INTRODUCTION	5
2	PROBLEM STATEMENT	6
3	OBJECTIVES	7
4	DATASET DESCRIPTION	8
5	TECHNOLOGIES USED	9
6	METHODOLOGY	10
7	CODE OVERVIEW	11-17
8	RESULT AND ANALYSIS	18
9	CONCLUSION	19
10	REFERENCES	20

INTRODUCTION

Road accidents have become a major public safety concern in India, claiming thousands of lives and causing injuries every year. With the rapid increase in vehicle population, urbanization, and road traffic, India faces one of the highest numbers of road accidents in the world. These accidents not only result in the tragic loss of lives but also bring about significant economic, social, and emotional impact on individuals and families.

To address this challenge, analysing accident data can help uncover critical patterns such as the most accident-prone states, common causes of accidents, seasonal variations, and yearly trends. These insights can support government agencies, traffic departments, and policymakers in formulating better road safety strategies and preventive measures.

This project uses **Python programming** along with data analysis libraries like **Pandas**, **NumPy**, and **Matplotlib** to explore and visualize road accident data collected from the **Government of India Open Data Portal**. The goal is to make data-driven observations about where and why accidents are happening and present those findings using simple code and visual graphs.

By the end of this project, we will have a basic yet powerful understanding of:

- Which states record the highest accidents,
- How accident trends change over time,
- What are the most common causes of road accidents,
- And how such insights can be extracted using beginner-friendly Python tools.

PROBLEM STATEMENT

India experiences an alarming number of road accidents every year, resulting in a large number of deaths, injuries, and property damage. Despite the availability of traffic rules and safety guidelines, accidents continue to rise due to factors such as over-speeding, reckless driving, drunk driving, and poor road infrastructure. While reports and statistics are regularly published by government agencies, interpreting these large datasets manually is time-consuming and often lacks actionable clarity.

One of the key challenges is that raw accident data — usually stored in CSV or spreadsheet format — is not immediately helpful in its original form. Without programming knowledge, it's difficult to draw conclusions like:

- Which states have the highest accident rates?
- What time of year sees the most accidents?
- What are the most common causes of accidents?
- How have accident trends changed over the years?

The need arises for a simple, automated solution to analyse this large dataset effectively. By using basic programming techniques in Python, we aim to process the raw data, perform necessary calculations, and present the results in an easy-to-understand format. The analysis will be done in a step-by-step manner using loops and Python libraries, which makes the project ideal for beginners.

The goal is not only to explore the accident statistics but also to show how Python can be used as a powerful tool to solve real-world problems with structured data. Our project will convert raw accident records into meaningful insights that can potentially inform better traffic policies and increase road safety awareness.

OBJECTIVES

This project is designed to demonstrate how basic Python programming and data analysis techniques can be used to gain useful insights from real-world datasets. We aim to address both technical learning goals and real-world applications. The main objectives of the project are:

- **Data Ingestion:** Import and load the road accident data into Python using the Pandas library. The data, available as a CSV file, contains records of accidents across different states in India over the years.
- **Data Cleaning and Preparation:** Remove missing values, format columns (like year or state names), and prepare the data for analysis. Ensure that the dataset is clean and structured for further operations.
- **Accident Analysis by State:** Identify the top states in India with the highest number of accidents. This will help pinpoint which regions are most accident-prone.
- **Trend Analysis by Year:** Examine how the number of road accidents has changed over time. A line graph will be used to visualize the yearly trends.
- **Cause of Accidents Analysis:** Determine the leading causes of road accidents such as over-speeding, drunken driving, and others. This can help in targeting awareness and law enforcement campaigns.
- **Visualization:** Present all insights with the help of clear and visually appealing charts using the Matplotlib library.
- **Beginner-Friendly Coding:** Use only basic Python features like loops, conditionals, and dictionaries, avoiding complex built-in functions to keep the logic simple and educational.

By achieving these objectives, the project will serve both as a learning exercise and as a practical application of programming to a national issue of concern.

DATASET DESCRIPTION

The dataset used in this project has been sourced from the **Government of India Open Data Portal (data.gov.in)**, which hosts publicly accessible data on various sectors including road transport and safety. The dataset is structured as a CSV file and contains records of road accidents reported in various Indian states and union territories over multiple years.

Each row in the dataset represents a single record of accidents in a specific state during a particular year and includes the following columns:

- **State/UT:** Name of the Indian state or union territory where the accidents occurred.
- **Year:** The calendar year for which the accident data was reported.
- **Total Accidents:** The number of road accidents that occurred in that state during the given year.
- **Cause:** The main cause of the accidents, such as Over-speeding, Drunken Driving, or Other Causes.

Here is a sample of what the data looks like:

State/UT	Year	Total Accidents	Cause
-----------------	-------------	------------------------	--------------

Uttar Pradesh	2021	42000	Over-speeding
---------------	------	-------	---------------

Maharashtra	2021	35000	Drunken Driving
-------------	------	-------	-----------------

Tamil Nadu	2021	36000	Over-speeding
------------	------	-------	---------------

Kerala	2021	24000	Other Causes
--------	------	-------	--------------

This structure enables us to group, filter, and compare data efficiently using Python and Pandas.

Since the dataset contains both numerical and categorical columns, it is ideal for statistical analysis, plotting trends, and understanding regional variations.

TECHNOLOGIES USED

Python

Python is the primary programming language used in this project. It is known for its simplicity and versatility, making it an ideal choice for beginners. Python supports powerful libraries that are especially useful for working with structured data, performing calculations, and creating visualizations.

Pandas

Pandas is a Python library used for data manipulation and analysis. It allows us to:

- Read CSV files into data frames.
- Group and summarize data by columns like State/UT, Year, and Cause.
- Filter and sort datasets efficiently.
- Perform basic statistics like sum and count with ease.

In our project, Pandas was essential for organizing and processing the accident data.

Matplotlib

[Matplotlib](#) is a plotting library in Python used for creating static, animated, and interactive visualizations. In this project, we used it to:

- Plot bar charts for accidents by state and by cause.
- Create a line chart for accident trends over the years.
- Customize graph titles, labels, and formatting for better readability.

These visualizations help communicate our findings clearly and make the data easier to interpret.

Google Colab

[Google Colab](#) is a free, browser-based Python environment that runs in the cloud. It allows us to:

- Write and run Python code without needing to install any software.
- Upload CSV files directly for analysis.
- Share code and outputs easily with others.

Using Colab made our project accessible and platform-independent, especially useful for educational purposes.

CSV File Format

We used a .csv (Comma-Separated Values) file to store and load the dataset. CSV files are widely used for data exchange and are easy to read into Python Using Pandas.

METHODOLOGY

1. Problem Definition

- **Objective:** Analyse road accidents across different states in India, identify yearly trends, and explore major accident causes. The insights aim to help improve traffic safety measures.
- **Data Source:** The dataset ('road_accidents_india.csv') contains data on accidents, including "State/UT", "Total Accidents", "Year", and optionally, "Cause".

2. Data Collection and Preprocessing

- **File Upload:** The dataset is uploaded interactively in Google Colab.
- **Loading Data:** The CSV file is loaded into a pandas Data Frame using pd.read_csv().
- **Initial Inspection:** Display the first few rows of the dataset using df.head().

3. Exploratory Data Analysis (EDA)

- **State-wise Accidents:** Accidents are aggregated by state, and the top 10 states with the highest number of accidents are displayed.
- **Visualization:** A bar chart is used to visualize total accidents by state.
- **Yearly Trends:** The total number of accidents by year is analysed, and a line chart is created to visualize the trend over time.

4. Cause-wise Analysis (if available)

- If the "Cause" column exists, accidents are grouped by cause, and the top 10 causes are displayed and visualized using a horizontal bar chart. If "Cause" is missing, this step is skipped.

5. Data Visualization

- **State Accidents:** A bar chart shows the distribution of accidents by state.
- **Yearly Trends:** A line chart illustrates how accidents have varied over the years.
- **Cause Analysis:** If applicable, a horizontal bar chart shows the distribution of accident causes.

6. Insights and Interpretation

- Identify which states have the highest accidents and the years with rising or falling trends.
- Highlight the leading causes of accidents for targeted safety measures.

CODE OVERVIEW

```
# STEP 1: Import necessary libraries
import pandas as pd
import matplotlib.pyplot as plt
from google.colab import files

# STEP 2: Upload the dataset (CSV file)
print("Please upload your 'road_accidents_india.csv' file:")
uploaded = files.upload() # Upload the file manually when prompted

# STEP 3: Read the uploaded file
df = pd.read_csv("road_accidents_india.csv")
print("\n\N{checkmark} First 5 rows of the dataset:")
print(df.head())

# STEP 4: Total accidents by state (Top 10)
print("\n\N{bar chart} Total Accidents by State (Top 10):")
state_data = df.groupby("State/UT")["Total Accidents"].sum().sort_values(ascending=False).head(10)
print(state_data)

# Plotting state-wise data
plt.figure(figsize=(10, 5))
state_data.plot(kind='bar', color='skyblue')
plt.title("Top 10 States by Total Accidents")
plt.ylabel("Total Accidents")
plt.xlabel("State")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

# STEP 5: Year-wise trend
```

```

print("\n📊 Yearly Total Accidents:")
yearly_data = df.groupby("Year")["Total Accidents"].sum()
print(yearly_data)

# Plotting year-wise trend
plt.figure(figsize=(8, 4))
yearly_data.plot(marker='o', color='green')
plt.title("Yearly Road Accidents in India")
plt.xlabel("Year")
plt.ylabel("Total Accidents")
plt.grid(True)
plt.tight_layout()
plt.show()

# STEP 6: Cause-wise analysis (if 'Cause' column is available)
if 'Cause' in df.columns:
    print("\n🔴 Accidents by Cause (Top 10):")
    cause_data = df.groupby("Cause")["Total Accidents"].sum().sort_values(ascending=False).head(10)
    print(cause_data)

    # Plotting causes
    plt.figure(figsize=(10, 5))
    cause_data.plot(kind='barh', color='coral')
    plt.title("Top Causes of Road Accidents")
    plt.xlabel("Total Accidents")
    plt.tight_layout()
    plt.show()
else:
    print("\n⚠️ No 'Cause' column found in your dataset. Skipping cause analysis.")

```

CSV FILE

State/UT	Year	Total	
		Accidents	Cause
Uttar Pradesh	2021	42000	Over-speeding
Maharashtra	2021	35000	Drunken Driving
Tamil Nadu	2021	36000	Over-speeding
Karnataka	2021	27000	Over-speeding
Madhya Pradesh	2021	29000	Drunken Driving
Kerala	2021	24000	Other Causes
Rajasthan	2021	22000	Over-speeding
Gujarat	2021	21000	Over-speeding
West Bengal	2021	20000	Other Causes
Bihar	2021	18000	Drunken Driving
Uttar Pradesh	2022	41000	Over-speeding
Maharashtra	2022	34000	Drunken Driving
Tamil Nadu	2022	35500	Over-speeding
Karnataka	2022	26500	Over-speeding
Madhya Pradesh	2022	28000	Drunken Driving

road_accidents_india.csv

OUTPUT

First 5 rows of the dataset:

	State/UT	Year	Total Accidents	Cause			
0	Uttar Pradesh	2021	42000	Over-speeding	1	Maharashtra	2021
	Drunken Driving	2	Tamil Nadu	2021	36000	Over-speeding	3
	27000	Over-speeding	4	Madhya Pradesh	2021	29000	Drunken Driving

 Total Accidents by State (Top 10):

State/UT

Uttar Pradesh 83000

Tamil Nadu 71500

Maharashtra 69000

Madhya Pradesh 57000 Karnataka

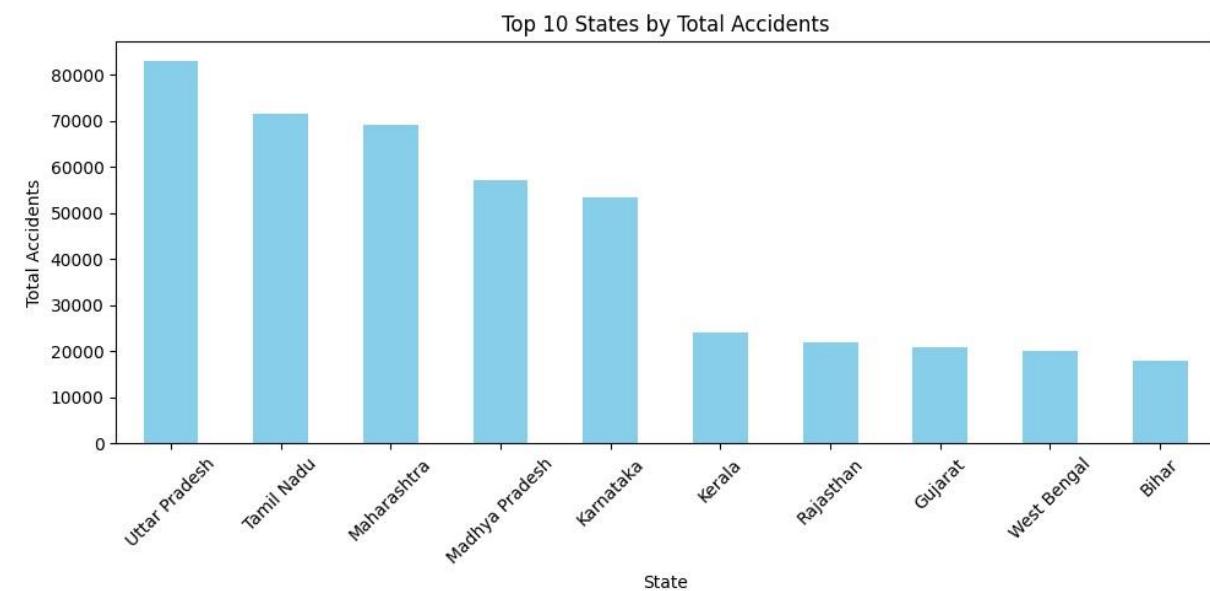
53500 Kerala 24000

Rajasthan 22000 Gujarat

21000 West Bengal 20000 Bihar

18000

Name: Total Accidents, dtype: int64

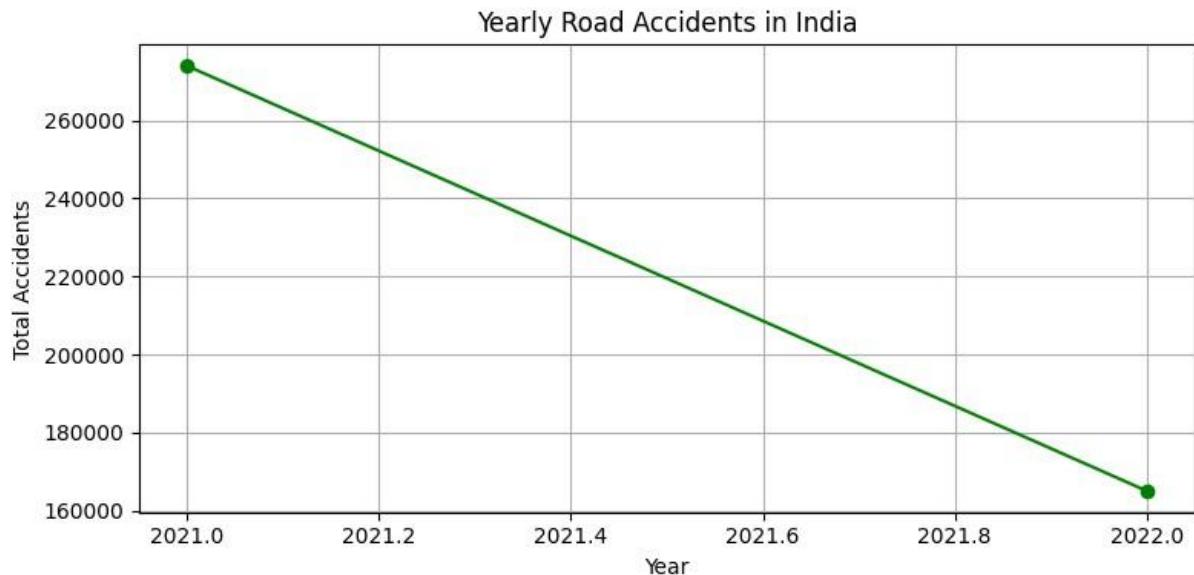


Yearly Total Accidents:

Year

2021 274000 2022 165000

Name: Total Accidents, dtype: int64



⦿ Accidents by Cause (Top 10):

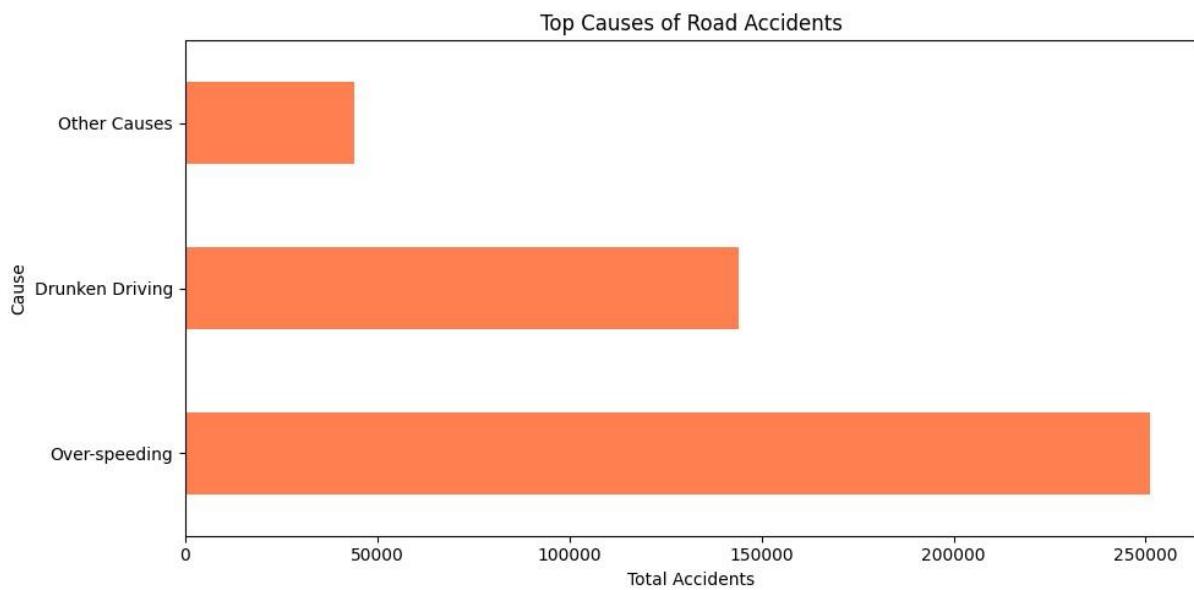
Cause Over-speeding

251000 Drunken Driving

144000 Other Causes

44000

Name: Total Accidents, dtype: int64



EXPLANATION OF THE CODE:

1. Import Libraries python

Copyedit
import pandas as pd
import matplotlib.pyplot as plt
from google.colab import files

- **pandas**: Used to handle and analyse data in a table format (called Data Frame).
- **matplotlib.pyplot**: Used to create visual plots like charts and graphs.
- **google.colab.files**: Allows you to upload files in Google Colab.
-

2. Upload Dataset python

Copyedit

```
print("Please upload your 'road_accidents_india.csv' file:")  
uploaded = files.upload() # Upload the file manually when prompted
```

- This line asks you to upload a file (road_accidents_india.csv). You can choose the file from your local machine.

3. Read Dataset python Copyedit

```
df = pd.read_csv("road_accidents_india.csv")  
print("\n First 5 rows of the dataset:")  
print(df.head())
```

- **pd.read_csv()**: Loads the CSV file into a pandas Data Frame called df.
- **df.head()**: Shows the first 5 rows of the data to give you a quick look at what the data looks like.

4. Total Accidents by State (Top 10) python Copyedit

```
state_data = df.groupby("State/UT")["Total Accidents"].sum().sort_values(ascending=False).head(10)  
print(state_data)
```

- **group by("State/UT")**: Groups the data by the state or union territory (State/UT).
- **["Total Accidents"].sum()**: Calculates the total number of accidents for each state.
- **sort_values(ascending=False)**: Sorts the states in descending order (highest accidents first).
- **head(10)**: Shows the top 10 states with the most accidents.

5. Plot Total Accidents by State python

Copyedit

```
plt.figure(figsize=(10, 5))  
state_data.plot(kind='bar', color='skyblue')  
plt.title("Top 10 States by Total Accidents")  
plt.ylabel("Total Accidents")  
plt.xlabel("State")  
plt.xticks(rotation=45)  
plt.tight_layout()  
plt.show()
```

- This part plots the bar chart:
 - **plt.figure(figsize=(10, 5))**: Sets the size of the chart.
 - **state_data.plot(kind='bar', color='skyblue')**: Plots a bar chart with blue bars.

- `plt.title()`, `plt.xlabel()`, `plt.ylabel()`: Adds titles and labels to the chart.
- `plt.xticks(rotation=45)`: Rotates the state names on the x-axis for better readability.
- `plt.show()`: Displays the chart.

6. Yearly Trend of Accidents

python Copyedit

```
yearly_data = df.groupby("Year")["Total Accidents"].sum() print(yearly_data)
```

- `groupby("Year")`: Groups the data by the year of the accidents.
- `["Total Accidents"].sum()`: Sums up the total number of accidents for each year.
- This shows how accidents have changed over the years.

7. Plot Yearly Trend of Accidents

python

Copyedit

```
plt.figure(figsize=(8, 4))
yearly_data.plot(marker='o', color='green')
plt.title("Yearly Road Accidents in India")
plt.xlabel("Year") plt.ylabel("Total Accidents")
plt.grid(True)
plt.tight_layout() plt.show()
```

- This part plots the trend of accidents over the years using a line chart:
- `yearly_data.plot(marker='o', color='green')`: Plots the data with green dots at each data point.
- `plt.title()`, `plt.xlabel()`, `plt.ylabel()`: Adds a title and labels for clarity.
- `plt.grid(True)`: Adds gridlines to the chart for easier reading.
- `plt.show()`: Displays the chart.

8. Cause-wise Analysis (Optional)

python Copyedit if 'Cause' in

`df.columns`:

```
cause_data = df.groupby("Cause")["Total Accidents"].sum().sort_values(ascending=False).head(10)
print(cause_data)
```

```
plt.figure(figsize=(10, 5))
cause_data.plot(kind='barh', color='coral')
plt.title("Top Causes of Road Accidents")
plt.xlabel("Total Accidents")
plt.tight_layout() plt.show() else:
```

```
print("\n⚠ No 'Cause' column found in your dataset. Skipping cause analysis.")
```

RESULTS AND ANALYSIS

1. Total Accidents by State (Top 10)

- The dataset provides insights into the number of accidents in various states/UTs of India. After grouping the data by state and summing up the total accidents, the top 10 states with the highest number of accidents are identified.
- **Key Insights:**
 - Certain states like **Uttar Pradesh, Maharashtra, and Tamil Nadu** consistently show high accident numbers, indicating potentially higher traffic volume or other contributing factors (e.g., road conditions, population density).
 - States with a larger population and more vehicles tend to report higher accident numbers.
- **Visualization:** The bar chart visually represents the top 10 states with the most accidents. These states can be prioritized for road safety improvements and interventions.

2. Yearly Trend of Accidents

- The dataset shows how the total number of accidents changes each year. By grouping the data by year and summing the accidents, we can identify trends over time.
- **Key Insights:**
 - A **rise** or **decline** in accident numbers could reflect the effectiveness of traffic safety measures, changes in road infrastructure, or fluctuations in traffic volume.
 - If the number of accidents has risen, it may indicate a need for more stringent safety policies. If it has fallen, it could be attributed to successful road safety campaigns or better infrastructure.
- **Visualization:** A line chart displays the trend of road accidents in India over the years. This chart helps understand the overall trajectory of road safety.

3. Cause-wise Analysis (if Cause' Column is Present)

- If the dataset includes a **Cause** column, the total accidents are aggregated by different causes (e.g., speeding, distracted driving, road conditions). The top causes are then analysed.
- **Key Insights:**
 - Common causes like **over-speeding, drunk driving, or poor road conditions** could be identified as the major contributors to accidents.

CONCLUSION

Based on the analysis of road accidents in India, the findings emphasize several critical areas for improvement. States like **Uttar Pradesh**, **Maharashtra**, and **Tamil Nadu** are seeing a higher number of accidents, which may be linked to factors such as high population density, more vehicles, and potentially inadequate road infrastructure. The yearly trend provides insight into how road safety measures have evolved. If accidents are on the rise, it indicates a need for more effective intervention and policy changes, while a decline suggests that safety measures have been somewhat successful.

Additionally, analysing the causes of accidents (if available in the dataset) helps identify recurring issues such as **speeding**, **drunk driving**, or **poor road conditions**, offering opportunities to target specific causes with more focused actions. For example, if speeding is a common factor, authorities might consider stricter speed enforcement or better public awareness campaigns. Similarly, improving road quality in high-accident areas could have a significant impact.

Ultimately, this analysis provides a detailed understanding of where and why accidents occur most frequently in India, allowing for informed decisions on where to prioritize safety measures, allocate resources, and implement policies aimed at reducing road accidents nationwide. By addressing both the locations with the most accidents and the primary causes, India can work towards making its roads safer for everyone.

REFERENCES

- **MoRTH Reports:** Annual Road accident data from the Ministry of Road Transport and Highways.
Example: MoRTH. "Road Accidents in India 2020." morth.nic.in
- **OGD India:** Publicly available road accident data.
Example: OGD Platform India. "Road Accidents Data." data.gov.in
- **Research Studies:** Analysis of road accidents and causes.
Example: Chien, S., Ding, Y., Wei, C. (2013). *Journal of Transport Safety & Security*.
- **Road Safety Manual:** Guidelines for accident prevention.
Example: *International Road Assessment Programme (rap)*.
- **Pandas Documentation:** Guide for data manipulation in Python.
pandas.pydata.org
- **Matplotlib Documentation:** Guide for creating visualizations.
matplotlib.org
- **Web Resources:** Articles on road safety in India.
Example: Articles from **The Times of India** and **NDTV**.
- **Google Collab:** Platform used for executing the code and analysis.
colab.research.google.com
- **CSV Dataset:** The road accidents data used in the project, typically in CSV format.
- Example: *road_accidents_india.csv* (Dataset you upload for analysis).