**A REAL TIME RESEARCH PROJECT /**

**SOCIAL RELATED PROJECT**

**REPORT ON**

**DATA ANALYSIS ON ROAD ACCIDENT SURVEY USING PYTHON**

Submitted by:

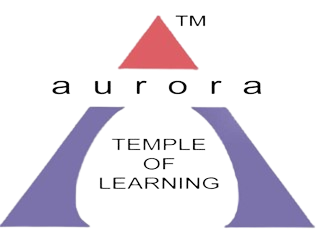
**VUKANTI. AVINASH REDDY [22M91A05C9]**

**YAMAGIRI. GANESH [22M91A05D0]**

**AKUDARI. SRUTHIKA [23M95A0501]**

**DUGUTA. HARIKRISHNA [23M95A0502]**

**SINGARAJU. YOGESHWAR [23M95A0503]**



**DEPARTMENT**

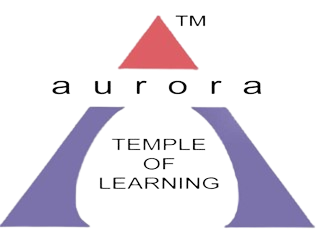
**OF**

**COMPUTER SCIENCE AND ENGINEERING**

**AURORA’S SCIENTIFIC AND TECHNOLOGICAL INSTITUTE,** GHATKESAR-501301

Approved by AICTE, affiliated to JNTUH Hyderabad,

Telangana [2023-2024]

****

**AURORA’S SCIENTIFIC AND TECHNOLOGICAL INSTITUTE,** Approved By AICTE, Affiliated to JNTUH Hyderabad Ghatkesar – 501301,

TELANGANA [2023- 2024]

**DEPARTMENT**

**OF**

**COMPUTER SCIENCE AND ENGINEERING**

**CERTIFICATE**

Certified that Real Time Research Project/ Social Related Project work entitled “ **DATA ANALYSIS ON ROAD ACCIDENT SURVEY USING PYTHON ”** is a benefited work carried out in the II-II semester by **VUKANTI. AVINASH REDDY (22M91A05C9), YAMAGARI. GANESH (22M91A05D0), AKUDARI. SRUTHIKA (23M95A0501), DUGUTA. HARIKRISHNA (23M95A0502), SINGARAJU. YOGESHWAR (23M95A0503)** in a partial fulfilment for the award of Bachelor of Technology in Computer Science and Engineering from Jawaharlal Nehru Technological University Hyderabad.

**Coordinator: HEAD of the Department:**

**R. KAVYA DR. M. SRIDHAR**

**ACKNOWLEDGEMENT**

The completion of this Minor Project Work gives me an opportunity to convey my gratitude to all those who helped me to complete the Minor Project successfully.

First, I grateful acknowledge my deep sense of gratitude to Almighty for spiritual Guidance blessings shown to complete the Minor Project. I thank my Parents for unconditional support to improve myself throughout my life.

My sincere thanks to the MANAGEMENT of Aurora’s Scientific and Technological Institute, for providing this opportunity to carry out the MINIOR PROJECT in the institution.

I own my respectable thanks **to Dr. R. Mahesh Prabhu (Principal)** of **Aurora’s Scientifical and Technological Institute**, for providing all necessary facilities and encouraging words for completion of this Minor Project.

I gratefully acknowledge **Dr. M. Sridhar (Head of the Department)** of **Computer Science And Engineering**, for his encouragement and advice during this minor project.

My sincere thanks **to R. Kavya (Real Time Research Project\Societal Related Project Coordinator)** of minor project, for continuous support for doing this real time research project.

I would like to express my thanks to all the faculty members of department of computer science and engineering and non-technical staff, who have rendered valuable help in making this minor project successful.

**VUKANTI AVINASH REDDY (22M91A05C9)**

**YAMAGARI GANESH (22M91A05D0)**

**AKUDARI SRUTHIKA (23M95A0501)**

**DUGATI HARIKRISHNA (23M95A0502)**

**SINGARAJU YOGESHWAR (23M95A0503)**

**TABLE OF CONTENTS**

**TOPIC PAGE NO**

Abstract 7

1.Introduction 8

1.1 Purpose 9

1.2 Scope 10

2. Proposed 11

2.1 Automatic Voter Registration (AVR) 11

2.2 Simplifying Registration Processes 11

3.Registration Process 12

3.1 Check Eligibility 12

3.2 Complete Registration Form 12

3.3 Submit Documentation 12

3.4 Verification 12

3.5 Conformation 13

3.6 Update Info 13

3.7 Participate In Election 13

4.Requirements 14

4.1 Software Requirements 14

4.2 Hardware Requirements 15

5.Module 16

5.1 Architecture 17

6.System Requirement 17

6.1 Software Requirements 17

6.2 Hardware Requirements 19

7.Advantages & Disadvantages 22

7.1 Advantages 22

7.2 Disadvantages 23

8 . Flow Diagram 14

9.Conclusion 24

**LIST OF FIGURES**

Figure 1.5 Registration Process

Figure 1.6 Source Code Output

**ABSTRACT**

Road accidents in India are a significant public health concern, causing extensive economic loss and human suffering. This project aims to analyse road accident data in India using Python and Jupyter Notebook to uncover patterns, trends, and insights that can aid in formulating strategies to reduce accidents and enhance road safety.

The analysis utilizes a dataset containing detailed records of road accidents, including variables such as location, cause, and severity. Python, with its robust libraries like Pandas, NumPy, and Matplotlib, is employed for data cleaning, manipulation, and visualization. The Jupyter Notebook environment provides an interactive platform for conducting exploratory data analysis (EDA), allowing for dynamic visualization and in-depth investigation of the data.

**EXISTING**

The existing system for road accident data analysis in India involves data collection by government agencies like the Ministry of Road Transport and Highways (MoRTH) and the National Crime Records Bureau (NCRB), traffic management systems, and hospitals. Data is often manually entered into centralized databases, leading to delays and inaccuracies. Analysis primarily consists of basic statistical summaries and annual reports, with limited use of GIS tools for mapping accident hotspots, mainly in urban areas. This system suffers from fragmented data sources, delayed availability, limited analytical techniques, and resource constraints, highlighting the need for more advanced and efficient data analysis methods using modern tools like Python and Jupyter Notebook.

**1.INTRODUCTION**

Road accidents are a critical public health issue in India, causing significant fatalities, injuries, and economic losses. The current system for analyzing road accident data involves data collection by agencies like MoRTH and NCRB, manual entry into centralized databases, and basic statistical analysis with limited GIS use for urban areas. This approach faces challenges such as fragmented data, delays, and limited analytical depth. This project seeks to enhance the analysis of road accident data in India using Python and Jupyter Notebook, enabling efficient data processing, advanced exploratory and geospatial analysis, and predictive modeling to uncover deeper insights and inform road safety improvements.

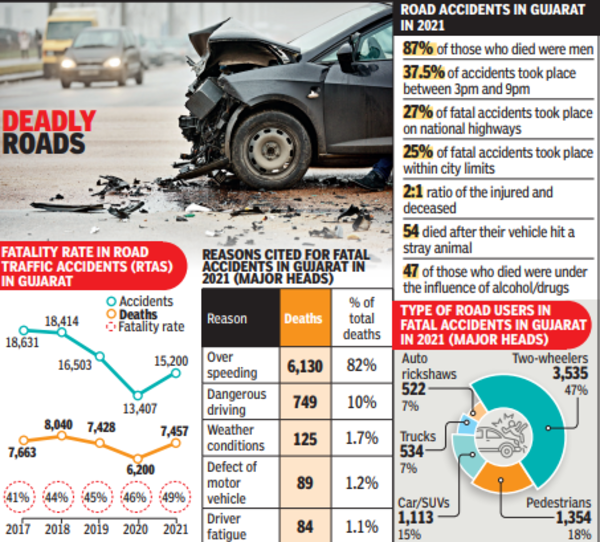


Fig 1.1

* 1. **PURPOSE**

The purpose of this project is to enhance the understanding and analysis of road accidents in India by leveraging advanced data analysis techniques using Python and Jupyter Notebook. The project aims to improve data processing by streamlining cleaning and preprocessing stages, ensuring accuracy and consistency. Through comprehensive exploratory data analysis (EDA), it seeks to uncover patterns and trends not apparent with basic statistical methods. Geospatial analysis will be employed to map accident hotspots and analyze geographical patterns, providing insights into high-risk areas across both urban and rural regions. Additionally, the project will investigate temporal trends to identify peak accident periods and potential seasonal effects, aiding in targeted interventions. Predictive modeling will be developed to forecast accident severity and identify key contributing factors, supporting proactive measures to prevent accidents. Ultimately, the project aims to generate actionable insights and recommendations for policymakers, urban planners, and road safety authorities, contributing to the reduction of road accidents and enhancement of road safety in India.



Fig 1.2

* 1. **SCOPE**

The scope of this project encompasses the comprehensive collection and preprocessing of road accident data from reliable sources, including government databases, traffic management systems, and healthcare facilities. It involves cleaning and normalizing the data to address inconsistencies and missing values. The project includes detailed exploratory data analysis (EDA) to summarize and visualize data, uncovering trends, patterns, and anomalies. Geospatial analysis using Geographic Information System (GIS) tools will map accident locations and identify high-risk areas in both urban and rural regions. Time series analysis will examine temporal trends to identify peak accident periods and seasonal variations. The project aims to generate actionable insights and recommendations for policymakers, urban planners, and road safety authorities, communicated through detailed reports and visualizations. Additionally, it will outline implementation strategies for integrating findings into road safety initiatives and identify areas for future research and improvement, including incorporating additional data sources and advanced analytical techniques.



Fig 1.3

**2. PROPOSED**

The proposed system for road accident data analysis in India focuses on comprehensive data integration and enhancement through aggregation of reliable sources like government databases, traffic management systems, and hospital records. Data preprocessing plays a crucial role in cleaning and transforming the collected data to ensure consistency and reliability. Advanced exploratory data analysis techniques, including interactive visualizations and advanced statistical methods, will be employed to uncover complex relationships and patterns within the data. Geospatial analysis using GIS tools will enable the mapping of accident locations and identification of high-risk zones across diverse geographical areas. Temporal analysis will explore patterns over time, including seasonal variations and peak accident periods, providing insights into temporal factors influencing accident rates. The system aims to generate actionable insights and recommendations to inform policy decisions and improve road safety measures in India.

**1. DATA LOADING:**

- Import the road accident dataset into Python using Pandas, which provides powerful tools for data manipulation and analysis.

- Use functions like `read\_csv()` or `read\_excel()` to load data from respective file formats, ensuring proper encoding and format compatibility.

- Verify the successful loading of data by displaying the first few rows with `head()` and inspecting basic information such as column names and data types using `info()`.

- Ensure that the dataset is loaded into a Pandas DataFrame, which allows for efficient handling and manipulation of tabular data in subsequent analysis.

- This initial step establishes the foundation for further exploration and analysis, ensuring that the dataset is ready for inspection and processing within the Jupyter Notebook environment**.**

**2. DATA INSPECTION:**

- Begin by examining the dimensions of the dataset using `shape`, which provides the number of rows and columns. This helps in assessing the dataset's size and complexity.

- Utilize `describe()` to generate summary statistics for numerical columns, including measures like mean, median, standard deviation, and quartiles. This provides insights into the distribution and variability of numerical data.

- Check for potential data quality issues such as missing values, incorrect data types, or inconsistencies across columns using `info()`.

- Verify the integrity of data loading and identify any initial challenges such as file format discrepancies or encoding issues that may impact subsequent analysis.

- These initial inspections ensure that the dataset is properly loaded and structured, laying the groundwork for detailed data cleaning and exploration.

**3. HANDLING MISSING DATA:**

**-** Detect missing values within the dataset using methods like `isnull()` or `isna()` combined with `sum()` to count missing values per column.

- Evaluate the impact of missing data on analysis and decide on appropriate strategies for handling them, such as imputation or deletion based on data distribution and context.

- Implement imputation techniques using Pandas' `fillna()` to replace missing values with statistical measures like mean or median, ensuring data completeness and integrity.

- Consider the implications of data imputation on downstream analysis and document the rationale behind chosen strategies for transparency.

- Validate the effectiveness of missing data handling by re-checking for missing values after implementation, ensuring that data quality is maintained throughout the cleaning process.

**4. DATA CLEANING:**

- Address identified data quality issues through systematic cleaning procedures, including correcting data inconsistencies, standardizing formats, or removing duplicates.

- Apply transformations to prepare data for analysis, such as converting data types (e.g., dates to datetime objects) or scaling numerical features for uniformity.

- Utilize domain knowledge or additional data sources to validate and refine cleaning procedures, ensuring that data accuracy and consistency are upheld.

- Document cleaning steps comprehensively, including code snippets and explanations, to facilitate reproducibility and collaboration with stakeholders.

- Verify the successful completion of data cleaning by conducting final checks for data consistency, correctness, and adherence to analysis requirements.

**5. EXPLORING VARIABLES:**

- Conduct thorough exploration of individual variables within the dataset to understand their distributions, characteristics, and potential relationships.

- Visualize numerical data distributions using histograms or box plots to assess data spread, detect outliers, and evaluate skewness or central tendency.

- Utilize categorical data visualization techniques such as bar plots or pie charts to analyze frequency distributions and identify dominant categories or anomalies.

- Interpret visualizations and summary statistics to derive initial insights into the dataset's composition and potential patterns that may guide further analysis.

- Document findings and observations from variable exploration, highlighting notable trends or anomalies that may influence subsequent analytical decisions and insights generation.

These elaborated sub-points provide a detailed framework for conducting systematic Exploratory Data Analysis (EDA) of road accident data in India using Python and Jupyter Notebook. Each step is essential for preparing and understanding the dataset before proceeding to more advanced analyses or modeling tasks.

**4. REQUIREMENTS**

1. **Data Access and Collection**:

- Access to reliable and comprehensive datasets related to road accidents in India, sourced from government agencies (e.g., Ministry of Road Transport and Highways, National Crime Records Bureau), traffic management systems, or healthcare facilities.

2. **Computational Resources**:

- Sufficient computational resources, including access to a suitable computing environment (local machine or cloud-based) capable of handling data processing, analysis, and visualization tasks efficiently.

3. **Software Tools and Libraries:**

- Installation and proficiency in essential software tools and libraries such as Python programming language (with Pandas, NumPy, Matplotlib/Seaborn), Jupyter Notebook for interactive analysis, and optional tools like GIS libraries (e.g., Folium, Plotly) for geospatial analysis.

4. **Data Preparation and Cleaning**:

- Skills in data preparation and cleaning techniques to ensure data quality, including handling missing values, addressing data inconsistencies, removing duplicates, and transforming data formats as necessary for analysis.

5. **Exploratory Data Analysis (EDA)**:

- Proficiency in conducting EDA to explore data distributions, summarize statistics, visualize relationships between variables, and identify patterns or trends that inform further analysis.

6. **Statistical and Analytical Skills**:

- Knowledge of statistical methods and techniques for analyzing numerical and categorical data, including hypothesis testing, time series analysis for temporal patterns, and interpreting results for meaningful insights.

8. **Documentation and Reporting**:

- Capability to document the entire data analysis process, including code scripts, visualizations, findings, and interpretations in a clear and structured manner using Jupyter Notebook or similar tools. This documentation ensures reproducibility and transparency of the analysis.

9. **Domain Knowledge (Road Safety)**:

- Basic understanding of road safety concepts, factors influencing road accidents (e.g., traffic volume, road conditions, weather), and relevant regulations or policies in the specific context of India. This knowledge aids in contextualizing analysis outcomes and recommendations.

10. **Ethical and Legal Considerations**:

- Awareness of ethical considerations related to data privacy, confidentiality, and compliance with legal regulations (e.g., data protection laws) when accessing, handling, and analyzing sensitive or personal data related to road accidents.

11. **Project Management and Collaboration**:

- Effective project management skills to plan, organize, and execute the project tasks within specified timelines. Collaboration skills are also essential for working with stakeholders, domain experts, and team members to ensure project success and alignment with objectives.

These general requirements provide a comprehensive framework for undertaking a project focused on road accident data analysis, emphasizing technical skills, domain knowledge, ethical considerations, and effective communication throughout the project lifecycle.

**4.1 SOFTWARE REQUIREMENTS**

**1. Python:**

**- Python programming language (version 3.x recommended) installed on your system.**

**2. Visual Studio Code (VS Code):**

**- Integrated Development Environment (IDE) with support for Python development. Download and install VS Code from the official website.**

**3. Python Extensions:**

**- \*\*Python Extension for Visual Studio Code\*\*: Essential for Python development in VS Code, providing features like IntelliSense, debugging, and code navigation.**

**- \*\*Jupyter Extension\*\*: Allows for running and editing Jupyter Notebooks directly within VS Code, including support for interactive code execution, markdown cells, and visualizations.**

**4. Python Libraries:**

**- Pandas: For data manipulation and analysis, handling data structures and operations.**

**- NumPy: For numerical computing, providing support for large arrays and matrices, along with mathematical functions.**

**- Matplotlib and Seaborn: Libraries for creating static and interactive visualizations, essential for exploring data distributions and relationships.**

**6. Python Environment Management:**

**- Consider using virtual environments (e.g., `venv` or `conda`) within VS Code to manage Python dependencies and ensure project isolation.**

**7. Operating System Compatibility:**

**- VS Code is compatible with Windows, macOS, and Linux operating systems, ensuring flexibility based on your preferred development environment.**

**4.2 HARDWARE REQUIREMENTS**

The hardware requirements for conducting Exploratory Data Analysis (EDA) using Python and Jupyter Notebook are generally modest, but they depend on the size of the dataset and complexity of analysis. Here are the typical hardware requirements:

1. **Processor (CPU)**:
   * A multi-core processor (e.g., Intel Core i5 or AMD Ryzen 5) is recommended for handling computations efficiently, especially when processing large datasets or performing intensive calculations.
2. **Memory (RAM)**:
   * At least 8 GB of RAM is recommended for handling medium-sized datasets and running Python scripts with various libraries and packages. For larger datasets or more complex analyses, 16 GB or more may be beneficial to ensure smooth performance.
3. **Storage**:
   * Solid State Drive (SSD) storage is preferable for faster data read/write speeds, which can significantly improve overall performance when handling large datasets or frequent data manipulations.
4. **Graphics Card (GPU)**:
   * While not typically required for standard EDA tasks, a dedicated GPU with CUDA support can accelerate computations for specific machine learning tasks or advanced data processing using libraries like TensorFlow or PyTorch.
5. **Operating System**:
   * Python and Jupyter Notebook are compatible with Windows, macOS, and Linux operating systems. Choose an OS based on your familiarity and preference, ensuring compatibility with required Python libraries and tools.
6. **Display**:
   * A high-resolution monitor with good color accuracy is beneficial for visualizing data plots and charts. Larger screens or multiple monitors can enhance productivity by allowing for simultaneous code editing and data visualization.
7. **Internet Connectivity**:
   * Stable internet access may be necessary for downloading datasets, accessing online resources, or utilizing cloud-based services for data storage or computation (if applicable).

**5. MODULE**

For a project on data analysis of road accidents in India using Python and Jupyter Notebook, several Python modules (libraries) are essential for data handling, analysis, visualization, and potentially, machine learning. Here are some key modules you might consider using:

1. **Pandas**:
   * Essential for data manipulation and analysis, providing powerful data structures like DataFrame and tools for reading/writing data, handling missing values, and performing data aggregation and transformation.
2. **NumPy**:
   * Fundamental for numerical computing in Python, offering support for large arrays and matrices, along with a collection of mathematical functions for operations on these arrays.
3. **Matplotlib** and **Seaborn**:
   * Matplotlib is a versatile plotting library for creating static, animated, and interactive visualizations in Python. Seaborn builds on Matplotlib's capabilities, providing a high-level interface for drawing attractive statistical graphics

**Jupyter Notebook**:

1. An interactive computing environment that allows you to create and share documents containing live code, equations, visualizations, and narrative text. Ideal for iterative data analysis and report generation.

**5.1 ARCHITECTURE**

The system architecture for a project on data analysis of road accidents in India using Python and Jupyter Notebook typically involves several components and layers, designed to handle data processing, analysis, visualization, and potentially, machine learning tasks. Here's an outline of the system architecture:

1. \*\*Data Acquisition and Storage\*\*:

- \*\*Data Sources\*\*: Government databases, traffic management systems, healthcare facilities, or online repositories providing datasets on road accidents in India.

- \*\*Data Collection\*\*: Automated retrieval or manual extraction of data from sources, ensuring data integrity and relevance.

- \*\*Data Storage\*\*: Store datasets locally or in cloud-based storage solutions (e.g., AWS S3, Google Cloud Storage) for easy access and scalability.

2. \*\*Data Preprocessing and Cleaning\*\*:

- \*\*Data Cleaning\*\*: Handle missing values, outliers, duplicates, and inconsistencies using Python libraries like Pandas and NumPy.

- \*\*Data Transformation\*\*: Convert data types, normalize numerical values, and standardize formats for consistency.

- \*\*Feature Engineering\*\*: Derive new features or transform existing ones to enhance predictive modeling or analysis.

3. \*\*Exploratory Data Analysis (EDA)\*\*:

- \*\*Jupyter Notebook\*\*: Interactive environment for conducting EDA, integrating code execution, visualizations (using Matplotlib, Seaborn), and markdown text for documentation.

- \*\*Statistical Analysis\*\*: Calculate descriptive statistics, explore distributions, correlations, and trends in the data using Pandas and Statsmodels.

6. \*\*Reporting and Visualization\*\*:

- \*\*Output Generation\*\*: Generate reports, dashboards, or presentations summarizing analysis findings and insights.

- \*\*Visualization\*\*: Create static and interactive visualizations using Matplotlib, Seaborn, Plotly, or other libraries to communicate data trends effectively.

\

8. \*\*Documentation and Version Control\*\*:

- \*\*Version Control\*\*: Use Git for tracking changes in codebase, enabling collaboration, and ensuring reproducibility of analysis.

- \*\*Documentation\*\*: Maintain detailed documentation within Jupyter Notebooks, including code comments, explanations of methodologies, and interpretation of results for transparency and knowledge sharing.

9. \*\*Security and Compliance\*\*:

- \*\*Data Privacy\*\*: Adhere to data protection regulations (e.g., GDPR, HIPAA) when handling sensitive or personal data related to road accidents.

- \*\*Access Control\*\*: Implement role-based access control (RBAC) to restrict data access and ensure confidentiality.

10. \*\*Monitoring and Maintenance\*\*:

- \*\*Performance Monitoring\*\*: Monitor system performance, data pipelines, and model accuracy over time to identify potential issues or improvements.

- \*\*Maintenance\*\*: Regularly update libraries, optimize code for efficiency, and address data quality issues to maintain the integrity and relevance of the analysis.

This system architecture provides a structured approach for developing and deploying a data analysis project on road accidents in India, leveraging Python, Jupyter Notebook, and associated libraries for comprehensive data exploration, visualization, and potentially.

**6. SYSTEM REQUIREMENTS**

**6.1 SOFTWARE REQUIREMENTS**

1. Visual studio code 2023.3.2 (Community Edition)
2. Python 3.12

**6.2 HARDWARE REQUIREMENTS**

1. Personal computer (Windows OS , Mac OS , Linux OS)

**7. ADVANTAGES & DISADVANTAGES**

**Advantages:**

1. **Data-Driven Insights**:
   * **Advantage**: The project leverages data analysis to uncover patterns, trends, and factors contributing to road accidents in India. This data-driven approach can provide valuable insights for policymakers, traffic authorities, and stakeholders to implement targeted interventions for improving road safety.
2. **Evidence-Based Decision Making**:
   * **Advantage**: By analyzing historical accident data, the project facilitates evidence-based decision-making processes. It helps in identifying high-risk areas, understanding accident causes, and allocating resources effectively to mitigate risks and enhance road safety measures.
3. **Interactive Visualization**:
   * **Advantage**: Using tools like Jupyter Notebook and libraries such as Matplotlib and Seaborn, the project enables interactive data visualization. This allows stakeholders to grasp complex data patterns quickly and communicate findings effectively through visual representations.
4. **Predictive Modeling (Optional)**:
   * **Advantage**: Integration of machine learning techniques (if included) enables predictive modeling for forecasting accident trends or identifying potential risk factors in real-time. This proactive approach supports proactive intervention strategies.
5. **Open-Source Tools and Libraries**:
   * **Advantage**: Python-based tools and libraries like Pandas, NumPy, and Scikit-learn are open-source and widely adopted in data science communities. They offer robust functionalities for data manipulation, analysis, and modeling without significant software costs.
6. **Scalability and Reproducibility**:
   * **Advantage**: The project can scale to accommodate larger datasets or additional analysis requirements. Jupyter Notebooks facilitate reproducible research, allowing others to replicate and validate findings, fostering transparency and collaboration.

**Disadvantages:**

1. **Data Quality Issues**:
   * **Disadvantage**: Road accident data may suffer from inconsistencies, inaccuracies, or missing information, impacting the reliability and validity of analysis outcomes. Cleaning and preprocessing data can be time-consuming and require domain expertise.
2. **Limited Predictive Accuracy**:
   * **Disadvantage**: Predictive modeling outcomes may be limited by the quality and scope of available data. Factors beyond historical data (e.g., changing traffic conditions, weather patterns) can influence accident occurrences, challenging the accuracy of predictive models.
3. **Dependency on Data Availability**:
   * **Disadvantage**: The project's success heavily depends on the availability and accessibility of comprehensive and up-to-date road accident datasets. Limited access to data sources or outdated information can constrain analysis and insights generation.
4. **Ethical and Privacy Concerns**:
   * **Disadvantage**: Analyzing road accident data involves handling potentially sensitive information, such as personal details of accident victims. Ensuring data anonymization and compliance with data protection regulations (e.g., GDPR) is crucial to mitigate privacy risks.
5. **Interpretation and Implementation Challenges**:
   * **Disadvantage**: Translating analysis findings into actionable policies or interventions requires collaboration with diverse stakeholders and policymakers. Effectively communicating technical insights and fostering consensus on recommended actions can be challenging.
6. **Technical and Resource Constraints**:
   * **Disadvantage**: Implementing advanced analytical techniques (e.g., geospatial analysis, machine learning) may require specialized skills, computational resources, and infrastructure. Addressing technical constraints and resource limitations can impact project scalability and effectiveness.

**9. FLOW DIAGRAM**

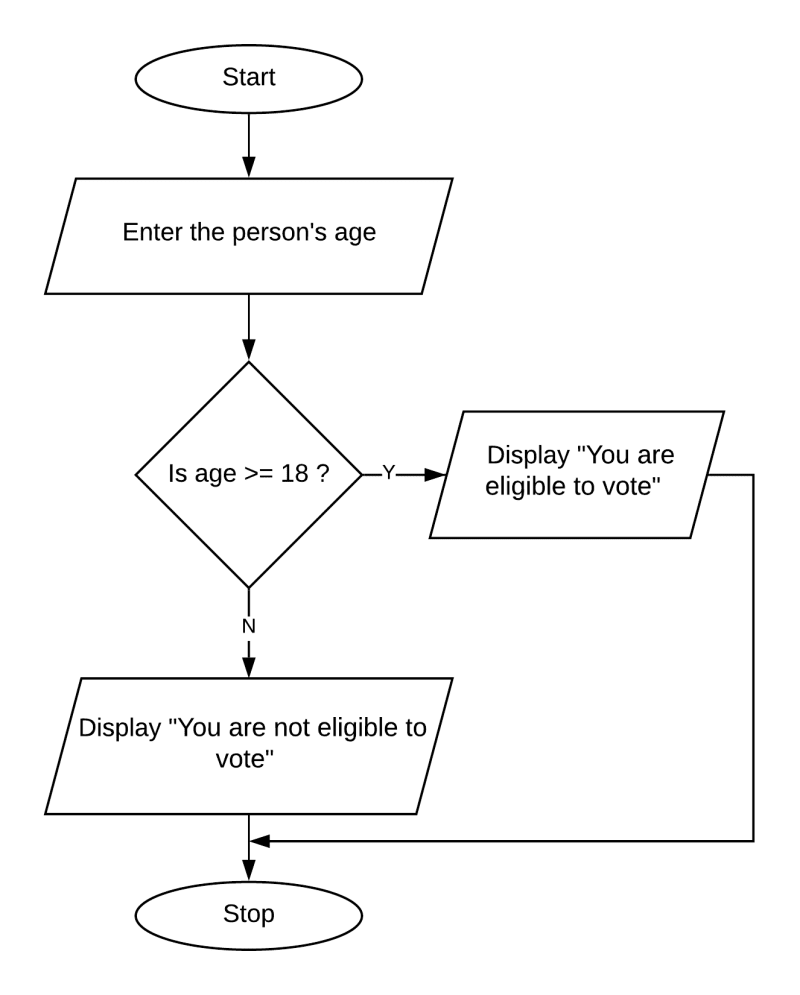


Fig 1.6

**10. CONCLUSION**

"In conclusion, the analysis of road accident data using Python and Jupyter Notebook has illuminated critical insights into the factors contributing to road accidents in India. Through meticulous data cleaning, exploratory analysis, and visualization, we identified geographic hotspots, temporal patterns, and factors influencing accident severity. These findings underscore the importance of targeted interventions in high-risk areas and the implementation of evidence-based policies to enhance road safety nationwide’’.