Problem Statement:

The goal is to develop an application to analyse vehicle accident data across the US to derive meaningful insights. The application should allow dynamic scaling as new datasets are added, maintaining efficiency and ease of use.

Initial Analysis on Dataset

* Conducted preliminary exploration to understand the structure, unique values, and quality of data
* Datasets analysed:
* Primary\_Person\_use: Contains primary information about individuals involved in accidents.
* Units\_use: Contains details about the vehicles involved in accidents.

Application Details:

Script Name: usva\_app.py

**Approach for Data Handling**

1. **Approach 1 (Chosen)**
   * **Dynamic & Scalable Solution**:
     + All datasets are loaded into a single **data dictionary** (df) with keys as filenames (e.g., damages, charges) and values as corresponding DataFrames.
     + Enables seamless integration of new datasets by placing them in the /data directory.
     + The application automatically identifies and integrates the new datasets into its analysis pipeline.
     + **Advantages**:
       - Minimal code changes needed to accommodate new datasets.
       - Easier to scale for multiple datasets.
2. **Approach 2**
   * **Readable & Explicit Solution**:
     + Each dataset is loaded into a separate DataFrame.
     + Analysis methods explicitly reference individual datasets by name.
     + **Advantages**:
       - Improved readability due to explicit naming.
       - Ideal for limited or fixed datasets.
     + **Disadvantages**:
       - Less flexible and scalable as new datasets require manual updates to the code.

Reason for Choosing Approach 1:

Approach 1 was selected to prioritize dynamic scaling and automation, ensuring the application is future-proof and can adapt to evolving data requirements.

Future Enhancements

* Visualization:
  + Incorporate interactive dashboards using libraries like Plotly or Dash.
* Advanced Analysis:
  + Integrate machine learning models to predict accident severity or high-risk areas.
* Data Quality Checks:
  + Include automated checks for missing or inconsistent data.