AI Assistant for Natural Language–Driven Data Visualizations

Objective

Build an AI-powered assistant that takes **natural language queries** and generates **appropriate data visualizations** from a given CSV dataset (e.g., dam storage reports).

Background

Government and infrastructure teams often generate reports (like **Pravah** for dams) in tabular formats. Interpreting and extracting insights manually is time-consuming.

Your task is to build an **AI agent** that can:

- Understand plain English queries about the dataset
- Convert them into data operations
- Automatically generate visualizations (tables, bar charts, etc.)

Features to Implement

1. CSV Data Parser

- Load CSV into a pandas DataFrame
- Normalize column names for easier access

2. Natural Language Understanding (NLU)

- Parse and interpret queries like:
    - “Top 5 dams by gross storage”
    - “Show live storage for dams with capacity above 5 TMC”
    - “Bar chart of gross vs live storage for each dam”

- LLMs (like OpenAI/GPT)
- OR rule-based keyword parsing + prompt engineering

3. Query-to-Action Engine

- Convert interpreted query into:
    - Filter conditions (e.g., `gross > 1`)
    - Sorting (e.g., descending by `gross`)
    - Aggregations or groupings
    - Chart types (e.g., bar, pie, table)

4. Visualization Generator

- Use `matplotlib`, `seaborn`, or `plotly` to build the chart
- Save or render the visualization (e.g., PNG, HTML)

5. Agent Response

- Return:
    - A brief explanation of what was done
    - The visualization
    - The filtered/sorted table

Example

Input CSV: Pravah Report (with columns like `Name of Dam`, `Gross Storage (TMC)`, `Live Storage`, etc.)

Query: (the agent should be able to work on the below queries)

"List all dams having gross capacity above 1 TMC in descending order of storage."

“Region-wise comparison bar graph of today’s live storage”

“Region-wise comparison bar graph of today’s live storage vs last year live storage same date”

Agent Output:

- Table view of filtered + sorted dams
- Bar chart: Dam Name vs Gross Storage
- Text: “Found 25 dams with capacity > 1 TMC, sorted by gross storage.”

Deliverables

- Python script or notebook that:
    - Loads CSV
    - Accepts a natural language query
    - Outputs a visualization and data table

- README with setup instructions and examples

- (Optional) Streamlit app for interactive testing