Weather Data Analysis

Description

Students will work with historical weather data in CSV format. They will use Python to read and process the data, calculate summary statistics, and create visualizations such as temperature trends over time. The project will emphasize data manipulation and visualization.

Steps

1: Dataset Research

Description: Research and select a suitable CSV dataset containing historical weather data. Ensure that the dataset includes important information like dates, temperature, and location.

Result Output: Dataset structure should be explained and provided as a technical document.

2: Data Reading

Description: Write Python code to read the selected CSV dataset using the standard csv module.

Result Output: Display a table in the terminal summarizing the dataset's structure and provide sample data rows. Dataset should be displayed partially to avoid long terminal output. Only show 10 rows of data: 5 top and 5 bottom rows.

3: Data Sanitization

Description: Clean the dataset by handling missing values, removing duplicates rows, and checking for data consistency.

Result Output: Display statistics about the cleaned dataset, including the number of records and any data quality issues.

4: Data Analysis

Description: Use Python to perform data analysis on the weather data, such as calculating summary statistics like average temperatures, maximum and minimum temperatures, and temperature trends over time.

Result Output: Create Matplotlib diagrams (e.g., line charts, scatter plots) to visualize temperature trends and generate tables summarizing key statistics.

5: Report Generation* (Optional)

Description: Generate a report summarizing the weather data analysis. The report can include tables of summary statistics and visualizations of temperature trends.

Result Output: Save the report as a HTML file and display a message in the terminal indicating the report has been generated.

6: Conclusion and Presentation

Description: Provide a conclusion based on the data analysis, highlighting temperature trends and any significant findings. Prepare a short presentation to share the findings with the class.

Result Output: Present the findings to the class and submit the conclusion and presentation materials as needed.

General requirements

- Add comments before all loop operations explaining the purpose of the loop.
- All functions must be annotated and must contain docstrings.
- Note that you need to avoid any errors/exceptions.
- Note that Python code must pass the PEP8 syntax check.
- argparse module from standard library should be used to get command line arguments.