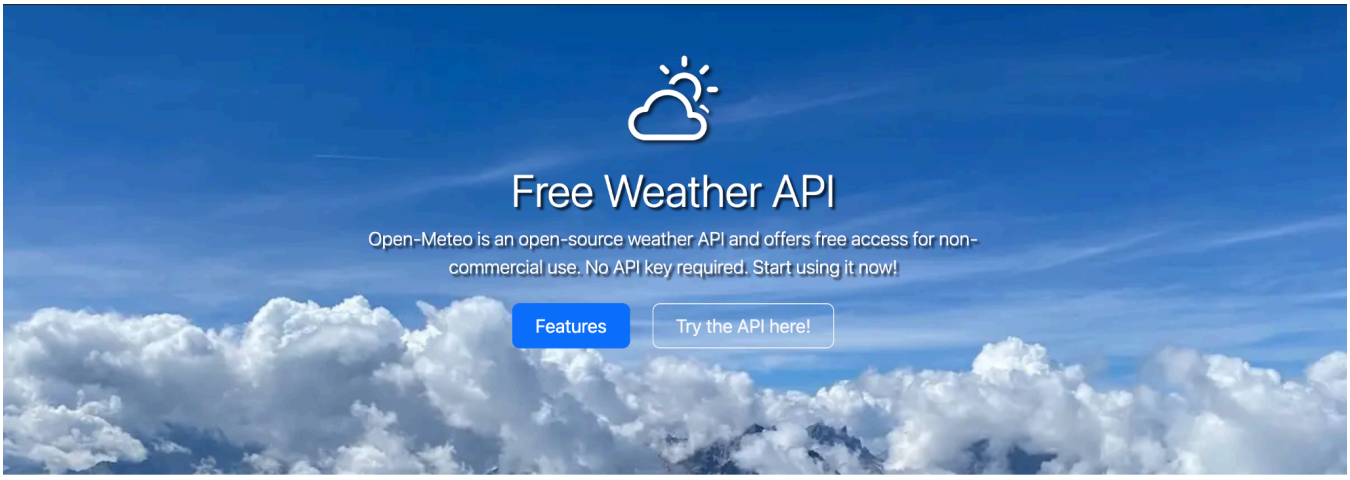

Lab 1: Data Collection and Cleaning using Python

Objective

Let's interact with the weather data and build our first data pipeline! In this lab, you will:

- Connect to an API to retrieve data.
- Store the retrieved data into a CSV file.
- Perform basic data cleaning operations (handling missing values, formatting dates).

Tools Required

- Python
- Git & Github (for version control)
- Basic command-line operations
- Libraries: requests, csv

Mini-Lab 1: Fetching and Cleaning Data from an API

Step 1: Setting Up Github Environment

- Sign in to your GitHub account.
- Navigate to the course's repo: [GitHub Repo URL]
- Fork the repo into your own GitHub account.
- Clone your fork to your local machine:

    - git clone <your-fork-repo-url>
    - cd <repo-name>
5. Pull the latest updates from the main branch:
    a. git checkout main
    b. git pull origin main
6. Create a new branch for Lab 1:
    a. git checkout -b lab1-<studentid>
7. Verify that the relevant files exist in labs/lab1/

Step 2: Setting Up Python Environment

- Create a new virtual environment from your terminal
    - python3 -m venv env
- Enable your environment
    - source env/bin/activate
- Install the requirements from the labs/lab1/requirement.txt file
    - pip3 install -r labs/lab1/requirements.txt
- Execute the following file to test your environment
    - python3 labs/lab1/envtest.py

Part 1 & 2: Connecting to an API and Fetching Data

1. Fetch the last 10 days of data from the weather API here.
2. Write a Python script to:
    - Send an HTTP GET request to fetch data.
    - Parse the JSON response.
    - Save the data into a CSV file.
3. Check that weather_data.csv is created.

Part 3: Cleaning the Dataset

1. Open weather_data.csv in a visual editor like excel and observe missing or inconsistent values.
2. Modify your script according to the rules mentioned in the pipeline file:
3. Verify the cleaned dataset in cleaned_data.csv.

Part 4: Aggregation

3. Load the cleaned_data.csv file and compute the summary statistics mentioned in the pipeline file.
4. Verify the results on your command line.

Submission Instructions

- **Stage and commit changes**:
  - git add weather_data_pipeline.py
  - git commit -m "Completed Lab 1 - <studentid>"
  - git push origin lab1-<studentid>
- **Go to GitHub → Open a Pull Request (PR)** from lab1-<studentid> to main branch. The PR name should be "Lab 1: Student ID Part 1, 2, 3, 4" where part numbers represent the completed parts.
