Creating Well-Structured Output for API Clients Using Postman to Get Weather Report. (Write-up)

In this project, the goal is to create a well-structured output for an API client using Postman to obtain a detailed weather report. The OpenWeatherMap API is utilized for this purpose, and the specific endpoint URL is https://samples.openweathermap.org/data/2.5/weather?q=London,uk&appid=b6907d289e10d714a6e88b30761fae22.

To ensure a systematic approach and maintain version control, the source code is tracked on a GitHub repository. The repository link will be shared for reference. It's essential to document the files that are ignored during the final push to GitHub, ensuring that sensitive information or unnecessary files are not exposed.

The step-by-step process involved in completing this task includes:

1. **Project Setup:**

   - Initialize a new project and create a dedicated folder for the source code.

   - Set up a GitHub repository and link it to the local project.

2. **API Endpoint Exploration:**

   - Understand the OpenWeatherMap API documentation to comprehend the available endpoints and their functionalities.

   - Identify the specific endpoint for retrieving weather data based on location.

3. **Postman Configuration:**

   - Install Postman if not already installed.

   - Create a new request in Postman and configure it with the OpenWeatherMap API endpoint.

   - Provide the necessary parameters such as the city (e.g., London) and the API key.

4. **Response Parsing:**

   - Examine the API response and identify relevant data fields.

   - Create a structured output format that includes key weather information such as temperature, humidity, and conditions.

5. **Documentation:**

   - Document the entire process, including the purpose of the project, API details, and steps to replicate the setup.

   - Include a section on ignored files in the documentation to highlight files not pushed to the GitHub repository.

6. **GitHub Repository Management:**

   - Commit the changes to the local repository.

   - Push the code to the GitHub repository, excluding ignored files.

7. **Submission:**

   - Share the GitHub repository link as part of the project submission.

   - Ensure that the documentation is clear and provides comprehensive insights into the project.

By following this structured approach, the project aims to not only deliver a functional weather report API client but also maintain a well-documented and version-controlled codebase for future reference.