Weather Status by Date and Location

Introduction:

The Weather Status project aims to provide users with a convenient way to retrieve weather information based on a specific date and location. It utilizes the OpenCage Geocoding API to obtain the coordinates of a given city and the OpenWeatherMap API to fetch weather data.

Functionality:

The project consists of a command-line application that prompts the user to enter the desired city name and date (in the format YYYY-MM-DD). The application then makes API calls to obtain the latitude and longitude of the city using the OpenCage Geocoding API. These coordinates are subsequently used to fetch weather data for the specified date from the OpenWeatherMap API.

API Integration:

The integration with the OpenCage Geocoding API involves making a GET request to the API endpoint with the provided city name and API key. The API response, in JSON format, is parsed to extract the latitude and longitude of the city.

The OpenWeatherMap API integration is performed by constructing the API URL with the obtained latitude, longitude, API key, and the specified date (converted to Unix timestamp). The application then sends a GET request to the API and processes the JSON response to extract relevant weather information such as temperature, humidity, wind speed, and description.

Dependencies and Libraries:

The project utilizes the **requests** library to handle HTTP requests to the APIs and the **json** library to parse the API responses. The **datetime** library is used to convert the user-provided date to a Unix timestamp.

Output Presentation:

The generated weather status output is presented in a visually appealing format. Decorative Unicode characters are used to create a bordered box around the weather information, enhancing the overall presentation of the data.

Future Enhancements:

Potential improvements to the project could include:

- Adding support for extended forecasts to retrieve weather information for multiple days.
- Incorporating graphical visualizations to present weather trends and patterns.
- Implementing user-friendly features such as autocomplete for city names.
- Enhancing error handling and providing informative error messages to the user.

Overall, the Weather Status project showcases the integration of external APIs and demonstrates the use of Python libraries to fetch and process weather data. It serves as a valuable tool for users to access real-time weather information and make informed decisions based on weather conditions.