

Exercise part a: mashup

Topic: Weather Forecast Mashup

Description: Create a weather forecast mashup that combines weather data from multiple sources and presents a comprehensive weather forecast for a specific location. This mashup should include data like current conditions, hourly forecasts, and a 5-day outlook.

Hints to the Solution:

1. Choose two or more weather data sources or APIs (e.g., OpenWeatherMap, Weather.com, or a local weather service).
2. Design a user-friendly web interface using HTML, CSS, and JavaScript to input and display the weather data.
3. Write JavaScript code to fetch data from the selected weather APIs and organize it into a unified forecast.
4. Include features like current conditions, hourly forecasts, and a 5-day outlook.
5. Ensure the mashup provides accurate and up-to-date weather information for the specified location.

Exercise part b: web service

Topic: Weather Forecast Web Service

Description: Develop a weather forecast web service that allows users to retrieve weather forecasts for specific locations by sending requests to your service. The service should provide current weather conditions, hourly forecasts, and daily forecasts.

Hints to the Solution:

1. Choose a language (e.g., Python) to implement your weather forecast web service.
2. Utilize established weather data APIs (e.g., the OpenWeatherMap API) to fetch real-time weather data.
3. Create RESTful API endpoints for retrieving weather information based on parameters like location, time frame (hourly or daily), and forecast details.
4. Structure the API responses in a clear and standardized format (e.g., JSON) that includes current conditions, hourly forecasts, and daily forecasts.
5. Test your web service with different location and timeframe requests to ensure it provides accurate weather information.

Exercise part c. Discuss/compare strenghts and weaknesses of the two solutions