Implementing API Calls in Blazor Applications

Step 1: Prepare for the Application

You’ll create a new Blazor WebAssembly application that fetches weather data using a public API. Your application will display the weather information and handle errors gracefully.

Instructions:

1. Open Visual Studio Code.
2. Create a new Blazor WebAssembly project: dotnet new blazorwasm -o WeatherApp cd WeatherApp dotnet run
3. Navigate to the URL provided in the terminal to ensure the application is running. Note the port number after “localhost:.”

Step 2: Sign Up for a Weather API and Get an API Key

You’ll need an API key to fetch weather data from a public API like WeatherAPI or OpenWeatherMap.

Instructions:

1. Go to the WeatherAPI website.
2. Create a free account by signing up with your email address.
3. After signing in, navigate to the "API Key" section of your account dashboard.
4. Copy the API key provided in your dashboard. You will use this key in the API request.

Step 3: Configure HTTPClient to Make a GET Request

Set up HTTPClient to fetch weather data from a public API

Instructions:

- Create a new Razor component named WeatherFetch.razor in the Pages folder.
- Inject the HTTPClient service into the component.
- Define a data model that matches the structure of the weather API response.
- Use HTTPClient to fetch data from the API endpoint in the OnInitializedAsync method.

Step 3: Configure HTTPClient to Maкe a GET Request

Set up HTTPClient to fetch weather data from a public API

Instructions:

- Create a new Razor component named WeatherFetch.razor in the Pages folder.
- Inject the HTTPClient service into the component.
- Define a data model that matches the structure of the weather API response.
- Use HTTPClient to fetch data from the API endpoint in the OnInitializedAsync method.
- Use the following API URL. Be sure to replace YOUR_API_KEY with the key you copied in the previous step. https://api.weatherapi.com/v1/current.json?key=YOUR_API_KEY&q=London

Step 4: Implement Error Handling

Ensure the application handles errors during the API call.

Instructions:

1. Wrap the API call in a try-catch block.
2. In the catch block:
    - Log the error message to the console.
    - Display a user-friendly error message in the UI.
3. Use fallback strategies such as displaying default or placeholder data if the API call fails.

Step 5: Test the API Call

Test the application's functionality and error-handling capability.

Instructions:

1. Start your application using: dotnet run
2. Navigate to http://localhost:5000/weatherfetch in your browser. If necessary change 5000 to the correct port from Step 1.
3. Test the app with a valid weather API endpoint.

4. Modify the endpoint in your code to an invalid URL, such as New York, and verify the error handling.