Implementing API Consumption in React

Step 1: Prepare for the Application

You'll create a small application using the Visual Studio Code console to build a simple Blazor application. The app will retrieve and display data from a RESTful API.

Instructions:

1. Open Visual Studio Code and create a new Blazor WebAssembly application
2. Launch the app
3. Ensure the default app runs successfully in your browser.

Step 2: Set Up API Consumption with HttpClient

You'll configure the app to consume data from an external RESTful API using HttpClient.

Instructions:

1. Open the Program.cs file and ensure HttpClient is registered for dependency injection.
2. Create a new Razor component
3. Inject HttpClient into FetchData.razor using the @inject directive to enable HTTP requests.
4. Declare a Posts variable to hold the response fetched from the API
5. Define a data model (Post) for API response handling:
    a. Include properties such as Id, Title, and Body to match the API structure.

Step 3: Make the API Call

You'll fetch data from a RESTful API and bind it to the component.

Instructions:

1. In FetchData.razor, use HttpClient to call the API endpoint (https://jsonplaceholder.typicode.com/posts) using GetFromJsonAsync.
2. Use async Task in the OnInitializedAsync lifecycle method to handle the API request.
3. Store the response data in the <Post> variable created in Step 2.

Step 4: Display Data in a User-Friendly Way

Render the retrieved data in the UI in a table format.

Instructions:

1. Inside FetchData.razor, define an HTML <table> structure.
2. Use a foreach loop to iterate over the API response and dynamically populate rows with Post data.
3. Add headers for Post ID, Title, and Body.

Step 5: Implement Basic Error Handling

Enhance the app to handle API failures gracefully.

Instructions:

1. Wrap the API call in a try-catch block.
2. Log any exceptions and display a fallback message if the API request fails.
3. Use a conditional statement to render "Loading..." or "Error fetching data" messages based on the API call status.