Task 7: Fetch and Display Random User Data using Async/Await

Objective:

Create a function that fetches random user data from the Random User API using async/await and displays the user's name and email. The function should handle errors gracefully and display an appropriate message if the data cannot be fetched.

Pre-requisites:

- Basic JavaScript (variables, functions)
- Promises
- Async/Await
- Fetch API
- Basic HTML and DOM manipulation

Concepts Covered:

- Async/Await
- Error handling in asynchronous code
- Fetch API
- DOM manipulation

Setup:

Install Node.js:

- Ensure Node.js is installed on your machine. You can download it from nodejs.org.

Tasks:

1. **Define Async Function:**
    - **Task:**
        - Define an async function named `fetchUserData`.
        - Use the Fetch API to get random user data from https://randomuser.me/api/.
        - Extract the user's name and email from the fetched data.
        - Display the user's name and email in the HTML.
        - Implement error handling to display a message if the data cannot be fetched.
    - **Outcome:**
        - Ensure the function correctly fetches and displays user data and handles errors gracefully.

Instructions:

- **Perform the following tasks:**
    - Write the required code in `index.html` and `script.js`.
    - Open `index.html` in a web browser to ensure the code executes without errors and demonstrates the use of async/await for fetching data.

Example Input:

1. Button click to fetch user data.

Expected Output:

1. User's name and email displayed in the HTML.

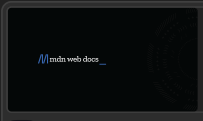
Resources:

- 

  How to use promises - Learn web development | MDN

  Promises are the foundation of asynchronous programming in modern JavaScript. A promi...

  https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Asynchronous/Async_await

- 

  Using the Fetch API - Web APIs | MDN

  The Fetch API provides a JavaScript interface for making HTTP requests and processing th...

  https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch

- 

  Control flow and error handling - JavaScript | MDN

  JavaScript supports a compact set of statements, specifically control flow statements, th...

  https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Control_flow_and_error_handling

Videos:

- 

  Telusko 16:52min 244,574 Views 5,121 Likes

- 

- 

GitHub Instructions:

1. **Open in Visual Studio Code:**
   - After clicking on the "Open in Visual Studio Code" button from the GitHub Classroom confirmation page, Visual Studio Code (VSCode) will open the repository directly.
   - If prompted, select "Open" or "Allow" to open the repository in VSCode.
2. **Open the Terminal in VSCode:**
   - In VSCode, open a terminal by selecting Terminal > New Terminal from the top menu.

3. **Complete the Task:**
    - In VSCode, write your solution in the `index.html` and `script.js` files.
4. **Run and Test Your Code:**
    - Open `index.html` in a web browser to ensure it works correctly.
5. **Commit Your Changes:**
    - In the VSCode terminal, add your changes to git:

```
git add index.html script.js
```

    - Commit your changes with a meaningful message:

```
git commit -m "Completed task 3"
```

6. **Push Your Changes to Your Repository:**
    - Push your changes to your forked repository:

```
git push origin main
```

7. **Create a Pull Request:**
    - Go to your repository on GitHub.
    - Click on the "Pull Requests" tab.
    - Click the "New Pull Request" button.
    - Ensure the base repository is the original template repository and the base branch is `main`.
    - Ensure the head repository is your forked repository and the compare branch is `main`.
    - Click "Create Pull Request".
    - Add a title and description for your pull request and submit it.

Summary of Commands:

```
# Open in Visual Studio Code

# Open terminal in VSCode

# Complete the task by editing index.html and script.js

# Add, commit, and push your changes
git add index.html script.js
git commit -m "Completed task 3"
git push origin main

# Create a pull request on GitHub
```

Need Help?

Task 7 Solutions

Code Hints — index.html: — Ensure your HTML file has a button and a div with the correct ids....

w3o NFThing Last Edited 7/3/2024