Final Project Front End Foundations

Final Project: Personal Website

For the final project in Front End Foundations, you're going to create a personal website. Follow the directions below to ensure you meet the requirements for a passing grade.

Goal

Your goal for this project is to create a website that has the following features:

- Layout your page using Flexbox or Bootstrap.
- A header with your name and your job title (ie. Web Developer, Full Stack Developer, etc..)
- A footer that sticks to the bottom of the screen, commonly referred to as a sticky footer.
- The footer should contain contact information with an email icon and an email address listed next to it.
- An image of yourself so everyone knows who you are.
- Clickable GitHub icon linking to your GitHub.
- Clickable LinkedIn icon linking to your LinkedIn.
- Other **Professional** Social Media (ie. Twitter, Facebook, Instagram, etc..) icons
- Using jQuery, when your icons are hovered over they should transform in size using CSS to show animation.
- List your tech-related skills.
- Dynamically list your GitHub repositories.
- The repositories should be hyperlinks to the actual repository page on your GitHub account.
- Your list of repositories should be styled.
- Style your page using CSS.

GitHub Repositories

Create a dynamic list of your GitHub repositories using an AJAX request. You will need the following API endpoint with your username.

```
https://api.github.com/users/YOURUSERNAME/repos
```

For example, if your username is `superdude` then you would use:

```
https://api.github.com/users/superdude/repos
```

If you navigate to the API with your username you should see a JSON object in the browser. It will look similar to the following code snippet.

```
[
  {
    "id": 69986023,
    "name": "Angular7",
    "full_name": "superdude/Angular7",
    "owner": {
      "login": "superdude",
      "id": 18702556,
      "avatar_url": "https://avatars2.githubusercontent.com/u/18702556?v=4",
      "gravatar_id": "",
      "url": "https://api.github.com/users/superdude",
      "html_url": "https://github.com/superdude",
      "followers_url": "https://api.github.com/users/superdude/followers",
      "following_url":
        "https://api.github.com/users/superdude/following{/other_user}",
      "gists_url":
        "https://api.github.com/users/superdude/gists{/gist_id}",
      "starred_url":
        "https://api.github.com/users/superdude/starred{/owner}{/repo}",
      "subscriptions_url":
        "https://api.github.com/users/superdude/subscriptions",
      "organizations_url": "https://api.github.com/users/superdude/orgs",
      "repos_url": "https://api.github.com/users/superdude/repos",
      "events_url":
        "https://api.github.com/users/superdude/events{/privacy}",
      "received_events_url":
        "https://api.github.com/users/superdude/received_events",
      "type": "User",
      "site_admin": false
    },
    "private": false,
    "html_url": "https://github.com/superdude/Angular7",
    "description": null,
    "fork": false,
    "url": "https://api.github.com/repos/superdude/Angular7",
    "forks_url": "https://api.github.com/repos/superdude/Angular7/forks"
  }
]
```

There are two properties that you will need to use.

First, the `name` property.

```
{"name": "Angular7"}
```

To target the `name` property you will need to use dot notation.

Second, the `html_url` property.

```
{"html_url": "https://github.com/superdude/Angular7",}
```

To target the `html_url` you will need to use dot notation.

> **TIP!** - There are two `html_url` properties. One is part of the nested object for `owner` and the other is part of the primary object.

This list should be generated with a function call, on the click of a button, using an `XMLHttpRequest` function containing a callback function. Each list item needs to be a hyperlink with the `.html_url` property as the `href` and the `.name` property as the displayed text for the hyperlink.

Starter Code:

```
JavaScript
```

```
function loadRepo(url, callback) {
  const gitHubRequest = new XMLHttpRequest();
    // Finish your request here
    // Don't forget the callback()
}

function loadRepoCallback(gitHubRequest) {
  // Callback function needs to parse the JSON
  // Using the DOM get the element ID from your <ul> in your HTML
  // You will need a forEach to loop through the parsed object
  // Using the DOM create new list elements
  // Plug in the .html_url and .name from the parsed object.  To make this a link you'll need
  // Append the new list items to the element you retrieved from HTML
}
```

Stretch Goal: CSS Toggle

Enhance your project in the following way.

The goal is to switch styling with the click of a button.

To do so, you will need to:

- Create an additional CSS file with all new stylings.
- Create a toggle switch.
- Once the switch is toggled the styling should be changed using jQuery.

> **NEED HELP?** - If you find yourself needing additional support, please reach out to a mentor using the live chat, or you can submit a ticket through the help desk (help.exeterlms.com).

Next Lesson >