

Enhancing your website using JavaScript

PART ONE

Objective

Implement a light/dark mode switcher. When the light mode is selected, all pages appear in a light theme, and when the dark mode is selected, the screen will appear in a dark theme.

Instructions

- You can make use of a button that appears on all pages (in a place that makes sense), and the user can click on the button to change to light mode or dark mode.
- The button can be a standard toggle button or a regular button that its text changes based on the current state of the theme.

PART TWO

Objective

Enhance your portfolio project by adding a blog section using JavaScript to dynamically load and display blog posts from a JSON file.

Instructions

1. Create a new HTML file for your blog page (e.g., `blog.html`).
2. Create a JSON file to store your blog posts (e.g., `posts.json`).
3. Use and **complete** the following JavaScript code to fetch and display the blog posts. Save this code in a file named `scripts.js`.

```
document.addEventListener('DOMContentLoaded', function() {

    const blogList = document.getElementById('blog-list');

    fetch('../data/posts.json')

    .then(response => response.json())

    .then(posts => {

        posts.forEach(post => {

            const postElement = document.createElement('div');

            // manipulate postElement to show the content of the blog post with the
            specific style defined for it

            //add postElement as a child to blog list

        });

    })

    .catch(error => console.error('Error loading blog posts:', error));

});
```

PART TWO

Objective

To avoid repeating the same header and footer on every page of your website, use JavaScript to insert them dynamically.

Instructions

1. Create separate HTML files for your header and footer content.
2. Write JavaScript code to load these files into your main pages.
3. Make sure your solution works when served from a local web server (not just by opening the files directly).

PART FOUR

Objective

Document your process by adding blog posts that explain your implementation of Parts One and Two.

Instructions:

- Add at least two blog posts in your `posts.json` file.
- Each blog post should explain your process for implementing the light/dark mode switcher (Part One) and the dynamic blog section (Part Two).

Submission

Submit the link to your GitHub repository in a pdf document and the published website containing your updated portfolio project with the new blog section and theme switcher. Make sure to submit your code alongside the pdf document as well.

Grading Criteria

- Correct implementation of the light/dark mode switcher.
- Proper fetching and displaying of blog posts.
- Adequate styling of the blog page.
- Clear and detailed blog posts explaining the implementation process.
- Code quality, readability, and organization.