

React Personal Portfolio Assignment

Goal

Create a personal portfolio website using React. The portfolio should include various class components such as a header, hero section, cards, footer, contact us form, animations, social media icons, and an about section.

Components to Include

1. Header:

- Navigation bar with links to different sections of the portfolio.
- Make it responsive for different screen sizes.

2. Hero Section:

- Display a welcome message or a tagline.
- Include a background image or color.

3. Cards:

- Create at least three cards to showcase projects or skills.
- Each card should have a title, description, and a link to learn more.

4. Footer:

- Include your name and a copyright statement.
- Optionally, add links to your social media profiles.

5. Contact Us Form:

- Simple form with fields like name, email, message.
- Use class components for form elements.

6. Animations:

- Incorporate animations to make the portfolio visually appealing.

7. Social Media Icons:

- Add icons with links to your social media profiles.
- You can use font awesome or react-icons <https://react-icons.github.io/react-icons/icons/fa/> or similar icon libraries.

8. About Section:

- Provide information about yourself, your skills, and interests.
- Consider adding a timeline or other creative ways to present your journey.

9. Conditional rendering:

- add buttons and when clicked it will render and change the content.
- example 1: click on react shows what you know about react
- example 2: click on javascript shows what you know about javascript
- etc...

10. useState:

- change all of the objects from previous weekend assignment to states.

11. useRef:

- the form you have made refactor it to use useRef and store the data in the local storage

12. events:

- add onClick to elements

Additional JavaScript Without Hooks

Consider incorporating the following JavaScript features without using hooks:

- **Event Handling:**
 - Handle events like button clicks or form submissions.
 - Implement methods within your class components to handle these events.
- **Conditional Rendering:**
 - Show or hide components based on certain conditions.
 - For example, display a message when the form is submitted.
- **Props and PropTypes:**
 - Pass data between components using props.
 - Define PropTypes for your class components.

Submission Guidelines

- Submit the project as a GitHub repository.
- Include a [README.md](#) file with a brief description of your portfolio and instructions on how to run the project locally.
- Deploy your project to Netlify.

Tips

- Explore and experiment with CSS to enhance the visual appeal.
- Test your portfolio on different browsers and devices for responsiveness.

- Use class-based components effectively and make meaningful class method names.

Feel free to reach out for any clarifications or additional guidance. Happy coding!