

HTML Fundamentals (with JSX)

Introduction to HTML & JSX in Our Application

HTML (Hypertext Markup Language) provides the fundamental structure for web pages. In this application, we use a powerful enhancement called JSX (JavaScript XML) directly within our single `index.html` file. JSX allows us to write HTML-like structures directly within our JavaScript (specifically, within our React components defined in the `<script type="text/babel">` tag).

This approach leverages React's component-based architecture for building dynamic user interfaces while keeping everything contained within a single, easily deployable HTML file. Think of JSX as a template language that gets transformed (by Babel) into regular JavaScript function calls that create HTML elements in the browser.

Why This Approach?

- Simplicity: A single file can be easier to manage and deploy for certain types of projects.
- Performance: Reduces the number of HTTP requests needed to load the initial page structure and logic.
- React Power: Allows us to use React's state management, lifecycle methods, and component reusability within a static HTML context.

Key JSX Syntax Differences:

- `className` instead of `class`
- CamelCase attributes like `onClick`
- Self-closing tags for elements like ``
- JavaScript expressions in `{ }`

Semantic Structure in Our Design:

Using semantic HTML tags helps with accessibility, SEO, and maintainability.

Example:

```
<header className="py-16 px-6">
```

```
  <h2 className="text-3xl">Professional Websites in 3-5 Days</h2>
```

```
</header>
```