

What are Nested Elements?

Nested elements in React mean putting one element **inside another**, just like how HTML tags are nested. In JSX, you can nest elements using **JSX syntax**, just like HTML — but with the full power of JavaScript.

◆ Basic Example:

jsx

CopyEdit

```
const element = ( <div> <h1>Hello World</h1> <p>This is a paragraph.</p> </div> );
```

- The `<h1>` and `<p>` tags are **nested inside** the `<div>` .
- This is totally valid JSX — in fact, it's how we build real UIs!

Nesting Custom Components

You can also nest **components** inside each other:

jsx

CopyEdit

```
function Header() { return <h1>Welcome to My Site</h1>; } function Footer() { return <p>© 2025 React Dev</p>; } function App() { return ( <div> <Header /> <p>This is the main content.</p> <Footer /> </div> ); }
```

App is the parent, and it nests `<Header />` , `<p>` , and `<Footer />` inside.

Nesting Rules & Best Practices

1. Wrap multiple elements in a parent (usually a `<div>` or fragment `<> </>`).

jsx

CopyEdit

```
return ( <> <h2>Title</h2> <p>Description</p> </> );
```

2. **Use indentation** for readability.
3. Nest components logically — keep your layout and structure organized.

Why Nesting Matters in React?

- UIs are built like trees — components **inside components**.
- Helps you build **reusable, modular UIs**.
- Makes code **more organized** and **easier to maintain**.