A Closer Look: Components & File Extensions

At this point, you've built a first custom component and you, of course, also worked with the `App` component.

For the moment, both components are stored in the `App.jsx` file (this will change later though).

`.jsx` is a file extension that's not supported by the browser! It's working because you're working in a React project that supports this special extension. Because this extension *"tells"* the underlying build process (which is running behind the scenes when the development server is running) that a file contains JSX code (which is also not supported by browsers).

It's important to understand that it's really just that build process that cares about this extension.

And therefore, you'll also find React projects that **don't use** `.jsx` but instead just `.js` as a file extension. And in those `.js` files, you'll also find JSX code. Because it simply depends on the underlying build process which extension is expected when using this JSX syntax in a file.

Since it doesn't work in the browser either way, there is no hard rule regarding this. Instead, you'll find projects that require `.jsx` (like the project setup we use in this course) and you'll find projects that also support `.js` (with JSX code inside).

I'm emphasizing this here so that you're not confused if you encounter React projects that don't use .jsx files.

In addition, you'll also find projects that require the **file extension as part of file imports** (e.g., `import App from './App.jsx'`) and you'll find other projects that don't require this (i.e., there, you could just use `import App from './App'`).

This, again, has nothing to do with the browser or *"standard JavaScript"* - instead it simply depends on the requirements of the code build process that's part of the project setup you chose.