

Let's now take a few minutes to look back on all the concepts that we just learned throughout this section. So first of all, we learned that components are the building blocks of any user interface in React. Now, each component is a self-contained piece of the user interface, which includes its own data, its own JavaScript logic and its own appearance. Now, in practical terms, we write this appearance using a declarative syntax that's called JSX. And it's this block of JSX that ultimately gets returned from each component. And this is what's going to describe exactly what the user will see on the screen when they use the application. And any piece of JSX can contain some markup basically in the form of HTML. It can contain some CSS, which in this section we wrote using the style prop. So we use the style prop and then passed an object in there which contained some CSS code. And also usually the JSX contains some JavaScript inside curly braces, which I like to call entering the JavaScript mode. So this one is basically like writing JavaScript right inside of HTML. So this is basically everything that a component contains or can contain and it is how we write it. So again, using JSX is now a complete application is usually composed out of many different components which are then organized into a component tree like this one. So in a component tree like this, the components that are at the top have included or used the components that are below them, which makes them their parent component. So the components at the top, for example, here, the app component is the parent component of header menu and footer. And so these three. So header menu and footer are the child components of app. And at the same time, of course, the menu is the parent component of all the pizza components and the footer is the parent component of order. So obviously a component can be a parent and a child at the same time. Now, in order to share data between components, parent components can pass data into a direct child component using props. So for each value that we want to pass down, we simply define one prop which is short for property. And so using props, we can configure components as we wish. For example, in this application we rendered many different pizzas by creating one pizza component and then passing in different pizza objects into that component. So rendering it multiple times with a different prop. Now it's very important to understand that props can only be passed down the tree, so only from parents to children, but never the other way around. Now, something that we do all the time in React applications is to render multiple components of the same type by looping over an array. And so this is what we call creating a list. So in the app that we just built, we created a list of pizzas by looping over the pizza array using the JavaScript map method. And so in React, there's nothing special to create lists. All we need to know is the map method that's available on all JavaScript arrays. So our JavaScript knowledge is more than enough to loop over an array in order to create lists of components of the same type. And finally, we also learned about conditional rendering. So that's another thing that we do all the time in order to render components only when certain conditions are met. And just like rendering lists, we can conditionally render components by using common JavaScript tools that we already know, for example, the and operator, the ternary operator, and also using multiple returns and that's it. So these are the more practical things that we just learned and that we applied to the projects that we have been building. We also learned some more theoretical stuff, like the difference between imperative and declarative approaches. We learned about separation of concerns. We learned exactly about why we cannot mutate props, which is another important thing that we learned. And yeah, we also learned a bunch more theoretical stuff that I'm not going to go into right now. So you can always rewatch those lectures if you're interested in here. I just wanted to summarize the practical aspects of everything that we just did in this section, So I hope that this was helpful. And now to round off this section, there is one final coding challenge waiting for you where we will just finish the developer profile card that we started initially. So hopefully I see you there very soon.