First React

Let's start jumping right into React. Since React is just a Javascript library, we can plug it right into our html. First, go to a folder that you want to start creating your React project. Next, within that folder, create an index.html file. Then, copy the following code into your index.html file:

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Hello React</title>

</head>

<body>

<div id="root">

<h1>First React page rendering...</h1>

</div>

     <script crossorigin="" src="https://unpkg.com/react@16/umd/react.development.js"></script>

        <script crossorigin="" src="https://unpkg.com/react-dom@16/umd/react-dom.development.js"></script>

</body>

</html>

copy

If you open index.html in your browser, you should see an h1 tag that says "First React page rendering...". Also, we have included two scripts that we will need for React. We have not set up React in our index.html yet (other than including the external Javascript), so you will be waiting for the rest of time if you want to wait for React to render. Next, let's actually add some custom React scripts to our file. We have already told our user that our React page is rendering, so let's actually render it. After the last external React script, create another script tag to contain our code. We will add the following:

<script>

*const* App = React.createElement("h1", {}, "Our First React page has rendered");

ReactDOM.render(App, document.getElementById("root"));

</script>

copy

What did we do here? We created a new method (using arrow notation) that returns a new React Element. Note that there are three different arguments provided to the React.createElement method. First, we have a string that contains h1. This tells React which element we want to create, in this case it is an h1 html element. Next, we have empty curly braces where we will pass props. We do not need to worry about this for the time being, but we will learn more about them later.

Lastly, we send in the children. This is what we want to create within our h1 tag. This could be a list of many different children, but for now we are just sending in a string.

After we define the App method, we call ReactDOM.render. We are telling React to display our App component within the element that has an id of root.

We have just written our first React project. If you open your index.html, you should briefly see "First React page rendering..." before seeing "Our First React page has rendered". Our browser will first show the html we have written.

Then, after the React scripts load, our browser executes the script that will call the method ReactDOM.render. This method will overwrite what we originally had in the div with id root with an h1 that contains "Our First React page has rendered". If you continue to refresh your page, you can see it quickly change. Congratulations! You have created your first React project.

**IMPORTANT: Do not include the node\_modules folder in your zip upload! The easiest way to do this is to highlight all the other files and folders in your project folder and compress them.**