

To get an overview of what React is, you can write React code directly in HTML. But in order to use React in production, you need npm and [Node.js](#) installed.

## React Directly in HTML

The quickest way start learning React is to write React directly in your HTML files. Start by including three scripts, the first two let us write React code in our JavaScripts, and the third, Babel, allows us to write JSX syntax and ES6 in older browsers.

```
<!DOCTYPE html>
<html>
  <head>
    <script src="https://unpkg.com/react@17/umd/react.development.js" crossorigin></script>
    <script src="https://unpkg.com/react-dom@17/umd/react-dom.development.js" crossorigin></script>
    <script src="https://unpkg.com/@babel/standalone/babel.min.js"></script>
  </head>
  <body>

    <div id="mydiv"></div>

    <script type="text/babel">
      function Hello() {
        return <h1>Hello World!</h1>;
      }

      ReactDOM.render(<Hello />, document.getElementById('mydiv'))
    </script>
  </body>
</html>
```

This way of using React can be OK for testing purposes, but for production you will need to set up a **React environment**.

## Setting up a React Environment

If you have npx and Node.js installed, you can create a React application by using `create-react-app`.

If you've previously installed `create-react-app` globally, it is recommended that you uninstall the package to ensure npx always uses the latest version of `create-react-app`.

To uninstall, run this command: `npm uninstall -g create-react-app`.

Run this command to create a React application named `my-react-app`:

```
npx create-react-app my-react-app
```

The `create-react-app` will set up everything you need to run a React application.

## Run the React Application

Now you are ready to run your first *real* React application!

Run this command to move to the `my-react-app` directory:

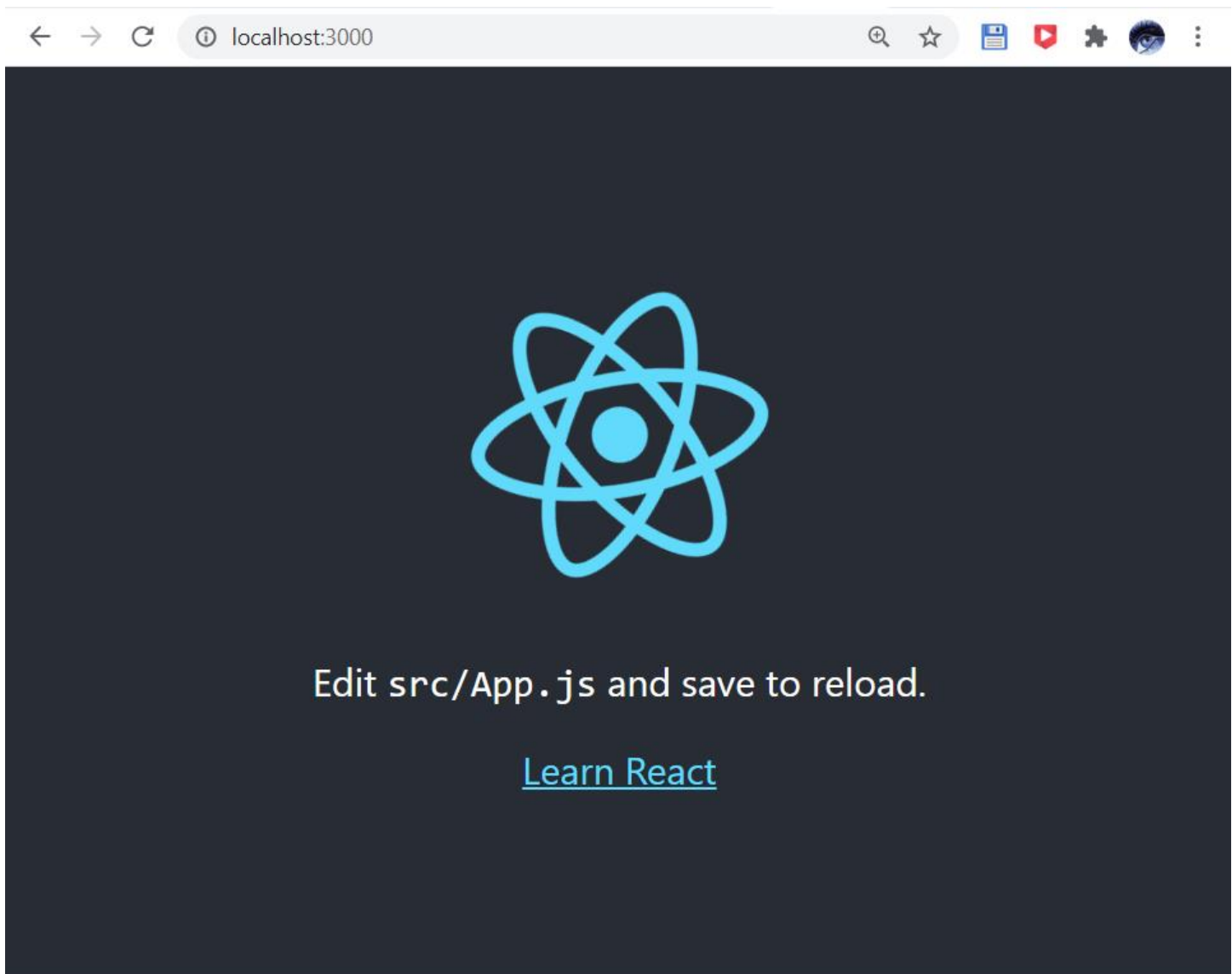
```
cd my-react-app
```

Run this command to run the React application `my-react-app`:

```
npm start
```

A new browser window will pop up with your newly created React App! If not, open your browser and type `localhost:3000` in the address bar.

The result:



Try changing the HTML content and save the file.

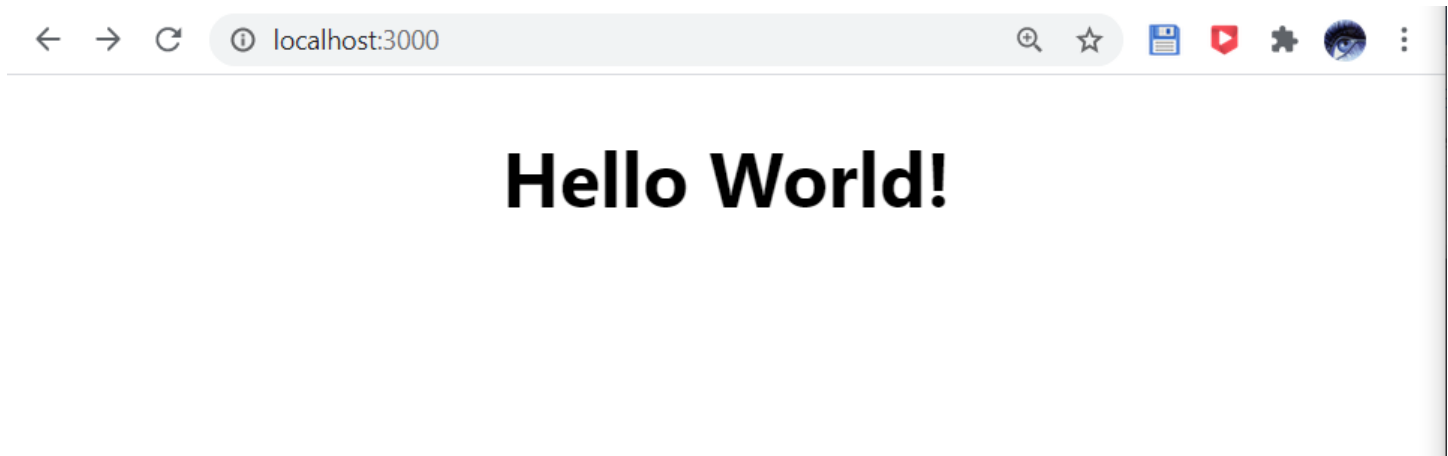
Notice that the changes are visible immediately after you save the file, you do not have to reload the browser!

## Example

Replace all the content inside the `<div className="App">` with a `<h1>` element. See the changes in the browser when you click Save.

```
function App() {  
  return (  
    <div className="App">  
      <h1>Hello World!</h1>  
    </div>  
  );  
}  
  
export default App;
```

Notice that we have removed the imports we do not need (logo.svg and App.css).  
The result:



## What's Next?

Now you have a React Environment on your computer, and you are ready to learn more about React.

In the rest of this tutorial we will use our "Show React" tool to explain the various aspects of React, and how they are displayed in the browser.

If you want to follow the same steps on your computer, start by stripping down the `src` folder to only contain one file: `index.js`. You should also remove any unnecessary lines of code inside the `index.js` file to make them look like the example in the "Show React" tool below:

### Example

Click the "Run Example" button to see the result.

```
index.js:
import React from 'react';
import ReactDOM from 'react-dom';

const myfirstelement = <h1>Hello React!</h1>

ReactDOM.render(myfirstelement, document.getElementById('root'));
```

