

Creating a basic project

In this lesson, we will start creating a React and TypeScript project by creating the folder structure, the 'package.json' and the HTML file that will host the React app.

WE'LL COVER THE FOLLOWING ^

- Technical requirements
- Creating a folder structure
- Creating package.json
- Adding a root web page
- Wrap up

Technical requirements

We are going to use the following tools in all the lessons where we'll learn how to create a React and TypeScript project manually:

- **Node.js** and **npm**. These can be installed from <https://nodejs.org/en/download/>. **npm** needs to be at least at version 5.2.
- A code editor such as **Visual Studio Code**. Visual Studio Code can be installed from <https://code.visualstudio.com/>

Creating a folder structure

In the folder we want our project located, we are going to create the folder structure for our project. So, let's open our code editor and create the following folders:

- **.vscode**: This is where the project's Visual Studio Code settings file will be located.
- **build**: This is where all the artifacts from the build output will be. This will also hold the HTML page that the React app will be injected into.

- `src`: This will hold our source code.

Note that a `node_modules` folder will also be created automatically as we start to install the project's dependencies in subsequent lessons.

Creating package.json

`package.json` is a file that contains critical information about our project, such as its dependencies and useful commands for starting and building our app.

Let's create this file in the root folder of our project containing the following content:

```
{
  "name": "my-app",
  "version": "0.0.1"
}
```

This file will automatically update with our project dependencies as we install them throughout this lesson.

Adding a root web page

We are going to add the HTML page that is going to host our React app. Let's add a file called `index.html` in the `build` folder with the following content:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
  </head>
  <body>
    <div id="root"></div>
    <script src="bundle.js"></script>
  </body>
</html>
```

Our React app will be injected into the `div` with `id` equal to `"root"`. All the app's JavaScript code will eventually be bundled together into a file called `bundle.js` next to `index.html` in the `build` folder.

Wrap up

wrap up

That's a great start!

In the next lesson, we are going to continue creating the project by adding React and TypeScript.