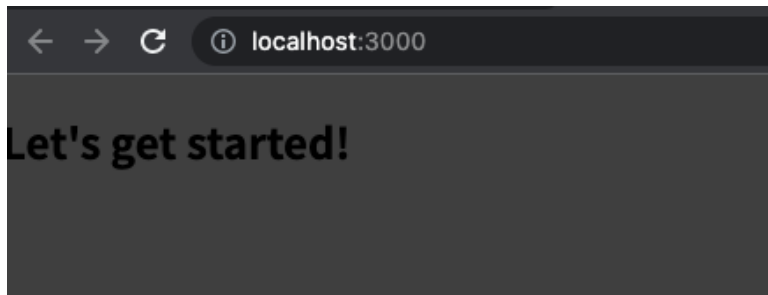
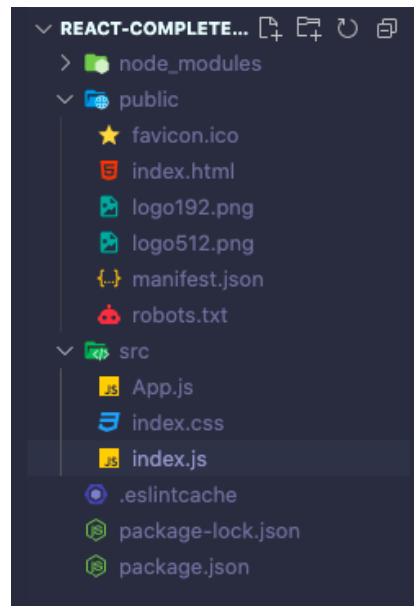


1. Into to components and basic working

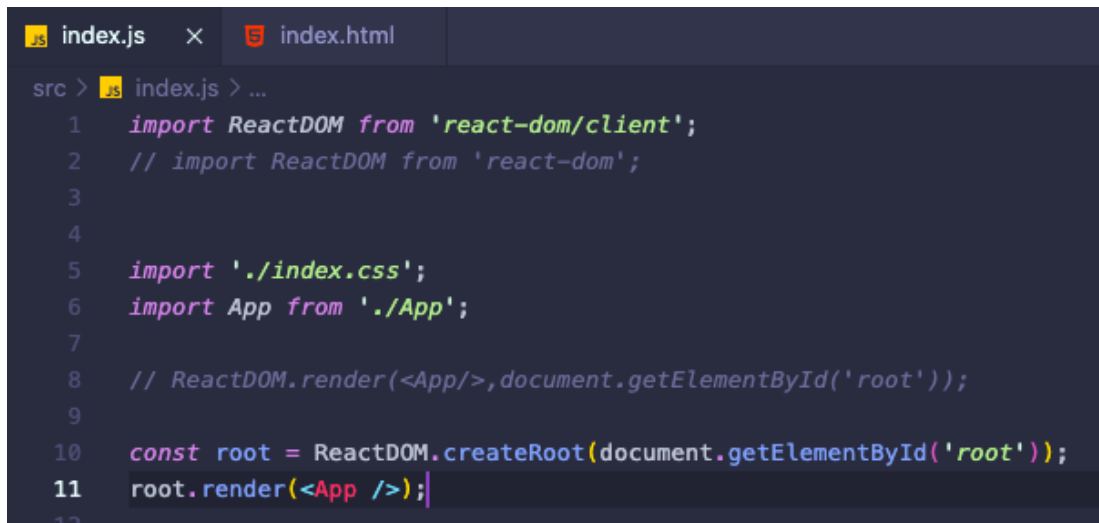


- This is our first page
- Now let us explore our file structure and code



- - This is our file structure
 - Here in src, we are having some js file and in public we have one index.html file and package.json we have some dependencies and packages for react app and in node modules we have the dependencies files which are installed using npm install

- Here index.js is the file which is executed first



```

src > index.js > ...
1  import ReactDOM from 'react-dom/client';
2  // import ReactDOM from 'react-dom';
3
4
5  import './index.css';
6  import App from './App';
7
8  // ReactDOM.render(<App/>, document.getElementById('root'));
9
10 const root = ReactDOM.createRoot(document.getElementById('root'));
11 root.render(<App />);
12

```

- NOTE : in newer version we use react-dom/client not just react-dom
- Here we are importing ReactDOM from react-dom package, we are directly importing it because it is the default export of 'react-dom' package, if it is not an default export then we would be using import {ReactDOM} from
- Then we are importing index.css, here we should mention the extension
- Then we are importing App from ./App where it refers to App.js file. Here we don't need to specify the extension because it is a js file, only for other fiels we need to specify the extension
- In older version we just do

```

■ ReactDOM.render(<App/>, document.getElementById('root'));

```

- But in newer version

```

■ const root =
  ReactDOM.createRoot(document.getElementById('root'));
■ root.render(<App />);

```

- Here createroot just specifies which is the root of our page
 - Now where is this dom element with id root?
 - It is there in public/index.html
 - This is the only file which is loaded in our react app
 - React app is just a single page application, the contents of the root is just dynamically changed

```
index.js index.html x
public > index.html > html > body > div#root
28 </head>
29 <body>
30 <noscript>You need to enable JavaScript to run this app.</noscript>
31 <div id="root"></div>
32 <!--
33   This HTML file is a template.
34   If you open it directly in the browser, you will see an empty page.
35
36   You can add webfonts, meta tags, or analytics to this file.
37   The build step will place the bundled scripts into the <body> tag.
38
39   To begin the development, run `npm start` or `yarn start`.
40   To create a production bundle, use `npm run build` or `yarn build`.
41 -->
42 </body>
43 </html>
```

- Now the content inside root is replaced by <App/> which is a custom component.

```
index.js index.html App.js x
src > App.js > App
1 function App() {
2   return (
3     <div>
4       <h2>Let's get started!</h2>
5     </div>
6   );
7 }
8
9 export default App;
10
```

-