

Day 53

“Web Development + Security”

Introduction to React:

What Is React?

React is a JavaScript library (not a full framework) used to build interactive user interfaces (UIs) — especially single-page applications (SPAs).

It was developed by Facebook (Meta) and is now open source.

React makes your web app:

- Faster (by using a Virtual DOM)
- Reusable (components)
- Dynamic (state-based UI updates)
- Easier to maintain (modular code)

Why Use React?

Feature	Description
Component-Based	Build UI in small, reusable pieces.
Virtual DOM	Efficiently updates only the parts that change.
Reactive UI	Automatically updates the view when data changes.
SPA Ready	Create single-page apps with smooth navigation.
Ecosystem	Works with tools like Redux, Tailwind, Router, etc.

Core Concepts in React

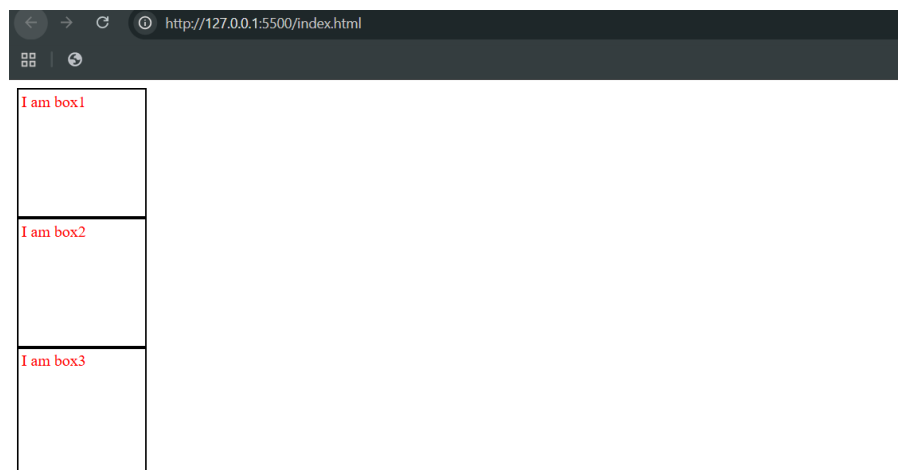
Concept	Description	Example
Component	Reusable block of UI	<code>function Navbar() { return <nav>...</nav>; }</code>
Props	Data passed <i>to</i> components	<code><Card title="Hello" /></code>
State	Data managed <i>inside</i> a component	<code>const [count, setCount] = useState(0)</code>
Event Handling	Handle user actions	<code>onClick={() => setCount(count + 1)}</code>
JSX	JavaScript + HTML syntax	<code><h1>{name}</h1></code>

Let's start from the basics, let's create three boxes using `<div>` and add some text inside it as its content:

Index.html:

```
index.html > html
2  <html lang="en">
3  <head>
7      <style>
8          .box{
9              color: red;
10             height: 121px;
11             width: 121px;
12             border: 2px solid black;
13             padding: 3px;
14         }
15     </style>
16 </head>
17 <body>
18     <div class="box box1">I am box1</div>
19     <div class="box box2">I am box2</div>
20     <div class="box box3">I am box3</div>
21 </body>
22 </html>
```

Output:



Now, suppose we want that text of these boxes should change with 1s, 2s and 3s respectively, then for that we will add `<script>` as shown below:

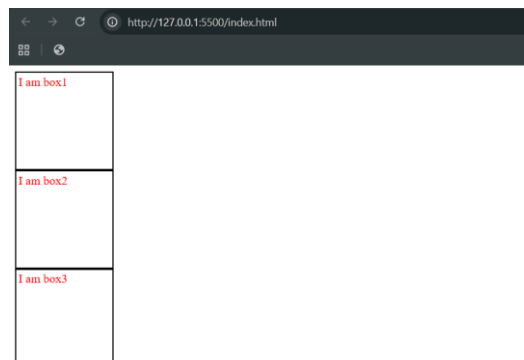
Code snippet:

```
<script>
    setTimeout(()=> {
        document.querySelector(".box1").innerHTML = "Hey"
    }, 1000);
    setTimeout(()=> {
        document.querySelector(".box2").innerHTML = "Heyy"
    }, 2000);
    setTimeout(()=> {
        document.querySelector(".box3").innerHTML = "Heyyy"
    }, 3000);
</script>
```

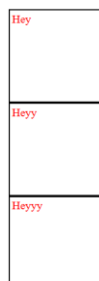
Whole code:

```
index.html > html > body > script
2 <html lang="en">
3 <head>
7 <style>
8   .box{
9     color: red;
10    height: 121px;
11    width: 121px;
12    border: 2px solid black;
13    padding: 3px;
14  }
15 </style>
16 </head>
17 <body>
18   <div class="box box1">I am box1</div>
19   <div class="box box2">I am box2</div>
20   <div class="box box3">I am box3</div>
21 <script>
22   setTimeout(()=> {
23     document.querySelector(".box1").innerHTML = "Hey"
24   }, 1000);
25   setTimeout(()=> {
26     document.querySelector(".box2").innerHTML = "Heyy"
27   }, 2000);
28   setTimeout(()=> {
29     document.querySelector(".box3").innerHTML = "Heyyy"
30   }, 3000);
31 </script>
32 </body>
33 </html>
```

Output: at 0s



Output: after 3s, clearly one by one there, text changed after 1s, 2s and 3s respectively.

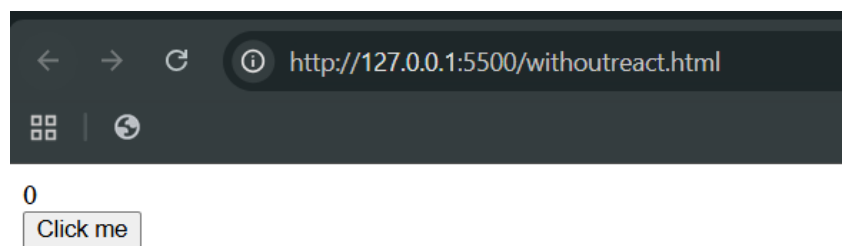


Now, let's create a program where when we click the button the counter will change(increase), without using React:

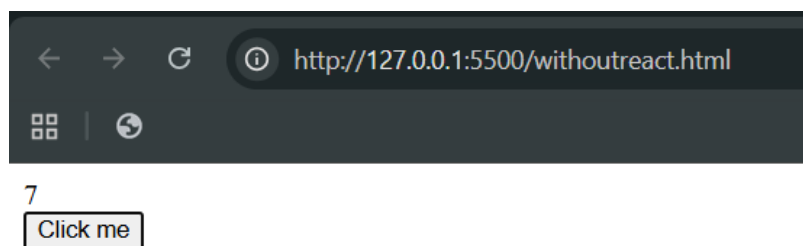
Code:

```
withoutreact.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Document</title>
7  </head>
8  <body>
9    <div id="counter">0</div>
10   <button id="btn">Click me</button>
11   <script>
12     document.getElementById("btn").addEventListener("click", ()=>{
13       let num = parseInt(document.getElementById("counter").innerHTML)
14       document.getElementById("counter").innerHTML = num+1;
15     })
16   </script>
17 </html>
```

Output: at start



Output: after 7 click



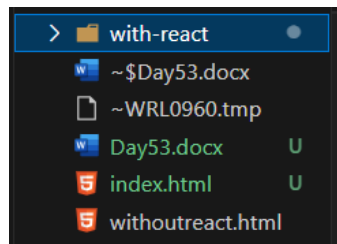
Now, we will create the same using the react:

For this we will first install the react: it will automatically create the folder

Use following code to do so:

```
PS E:\FullStackDevelopment\Day51-60\Day53> npx create-react-app with-react
Need to install the following packages:
create-react-app@5.1.0
Ok to proceed? (y) y
```

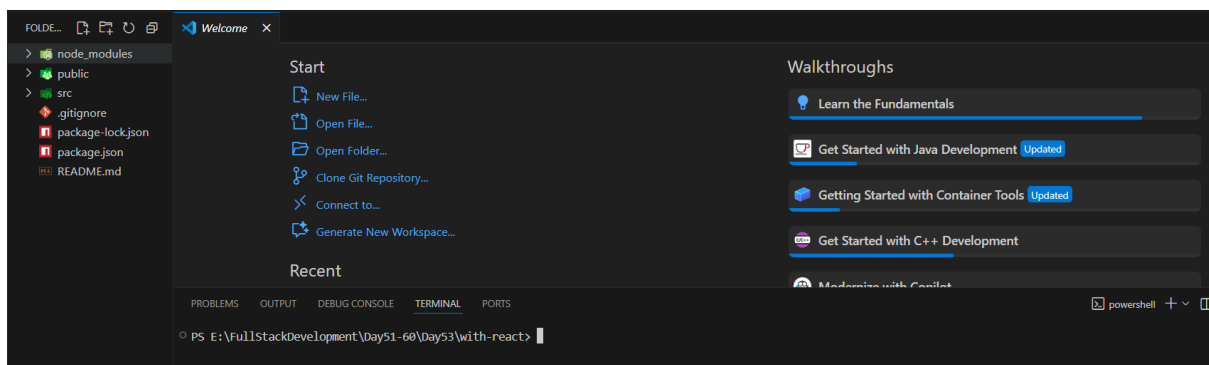
It will create the following folder:



This is the content of this folder:

Name	Date modified	Type	Size
node_modules	20-10-2025 16:41	File folder	
public	20-10-2025 16:41	File folder	
src	20-10-2025 16:41	File folder	
.gitignore	20-10-2025 16:41	Git Ignore Source ...	1 KB
package	20-10-2025 16:41	JSON Source File	1 KB
package-lock	20-10-2025 16:41	JSON Source File	649 KB
README	20-10-2025 16:41	Markdown Source ...	4 KB

Now, switch to this new folder:



Now, we will use npm start to start the react app:

Terminal:

```
PS E:\FullStackDevelopment\Day51-60\Day53\with-react> npm start

> with-react@0.1.0 start
> react-scripts start

(node:31192) [DEP_WEBPACK_DEV_SERVER_ON_AFTER_SETUP_MIDDLEWARE] DeprecationWarning: 'onAfterSetupMiddleware' option is deprecated. Please use the 'setupMiddlewares' option.
(Use 'node --trace-deprecation ...' to show where the warning was created)
(node:31192) [DEP_WEBPACK_DEV_SERVER_ON_BEFORE_SETUP_MIDDLEWARE] DeprecationWarning: 'onBeforeSetupMiddleware' option is deprecated. Please use the 'setupMiddlewares' option.
Starting the development server...
Compiled successfully!

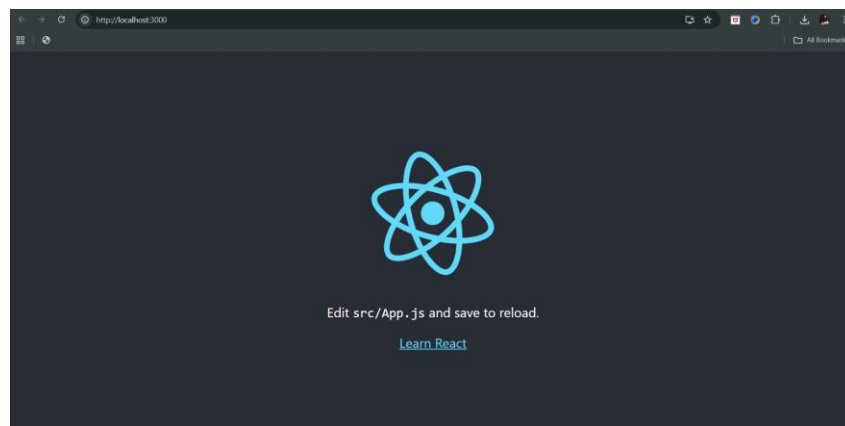
You can now view with-react in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.0.141:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

Output:



Now, in order to make a counter, we will use the App.js of the “src” folder:

Original:

```
node_modules
public
src
  App.css
  JS App.js
  App.test.js
  index.css
  index.js
  logo.svg
  reportWebVitals.js
  setupTests.js
.gitignore
package-lock.json
package.json
README.md

src > JS App.js > ...
1  import logo from './logo.svg';
2  import './App.css';
3
4  function App() {
5    return (
6      <div className="App">
7        <header className="App-header">
8          <img src={logo} className="App-logo" alt="logo" />
9          <p>
10             Edit <code>src/App.js</code> and save to reload.
11          </p>
12          <a
13            className="App-link"
14            href="https://reactjs.org"
15            target="_blank"
16            rel="noopener noreferrer"
17          >
18            Learn React
19          </a>
20        </header>
21      </div>
22    );
23  }
24
25  export default App;
26
```

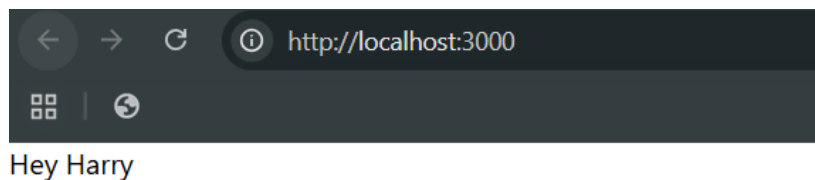
Let's change the content of it from basics:

Updated src/App.js:



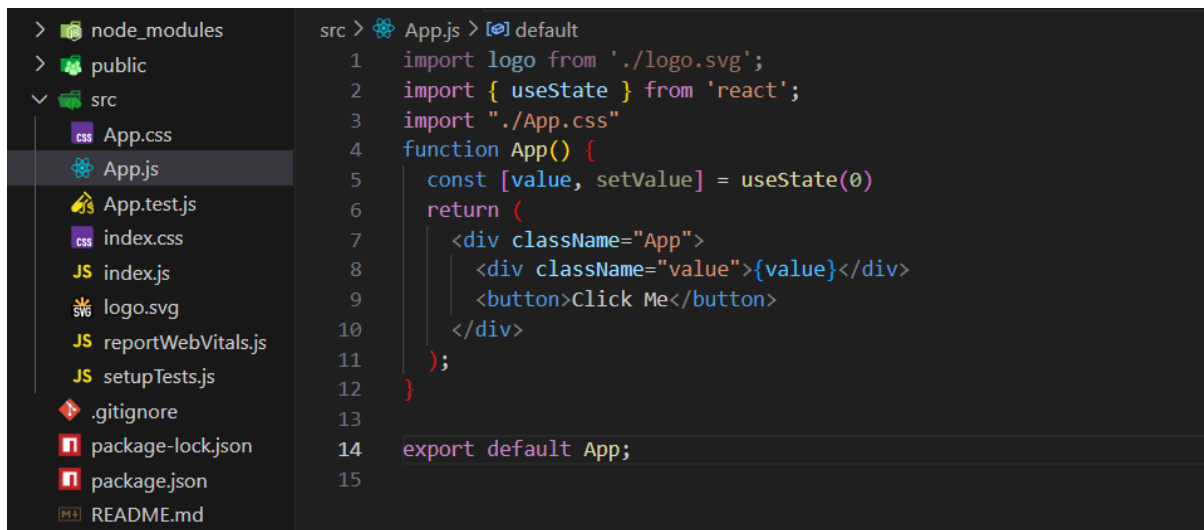
```
src > JS App.js > ...
1  import logo from './logo.svg';
2
3  function App() {
4    return (
5      <div className="App">
6        Hey Harry
7      </div>
8    );
9  }
10
11  export default App;
```

Output:



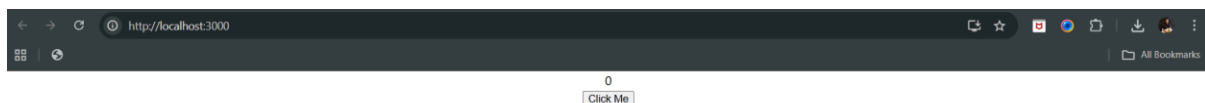
Now, we will add this in the centre, and add the button as well:

App.js:

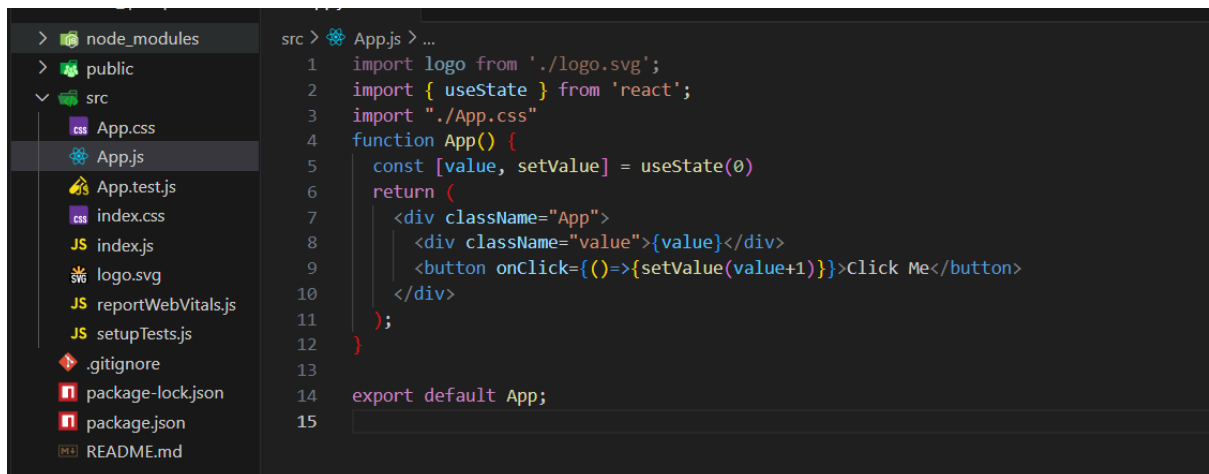


```
src > App.js > default
1  import logo from './logo.svg';
2  import { useState } from 'react';
3  import './App.css'
4  function App() {
5    const [value, setValue] = useState(0)
6    return (
7      <div className="App">
8        <div className="value">{value}</div>
9        <button>Click Me</button>
10      </div>
11    );
12  }
13
14  export default App;
15
```

Output:



Now, we will add the logic where the value will increase by 1 with every click:



```
src > App.js > ...
1  import logo from './logo.svg';
2  import { useState } from 'react';
3  import './App.css'
4  function App() {
5    const [value, setValue] = useState(0)
6    return (
7      <div className="App">
8        <div className="value">{value}</div>
9        <button onClick={() => {setValue(value+1)}}>Click Me</button>
10     </div>
11   );
12 }
13
14 export default App;
15
```

Output: at start



Output: after 1st click



Output: after 2nd click



--The End--