



ReactJS

June 2021





Course Objective

- React Introduction
- Environment Setup
- ES6
- React Render Html
- JSX
- React Components
- Props and State
- Component API
- Component Life cycle
- React Forms, Events Refs, Keys
- Rest API
- React Router

Session Plan

React Introduction

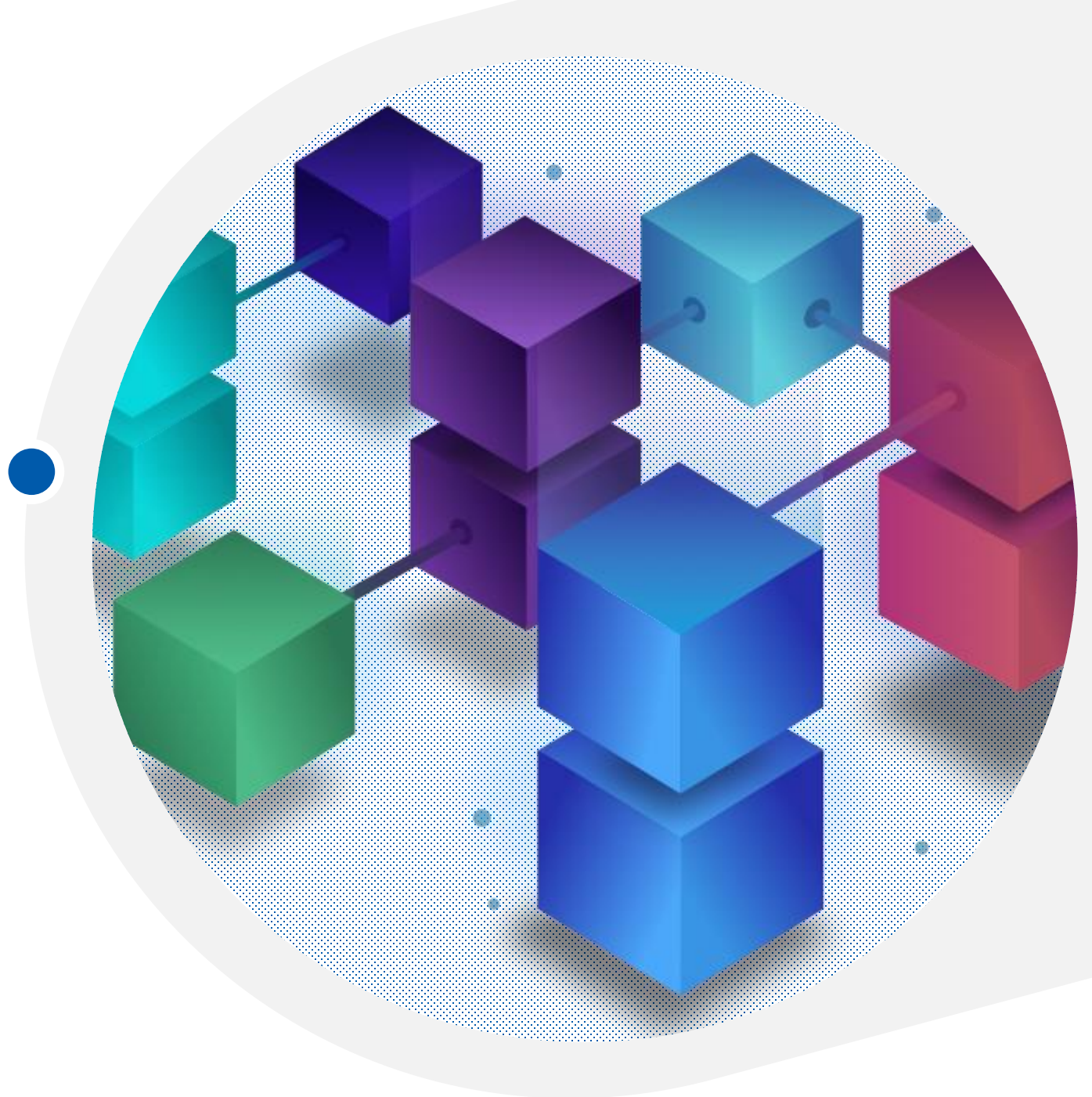
Environment Setup

ES6

React Render Html

JSX

React Components

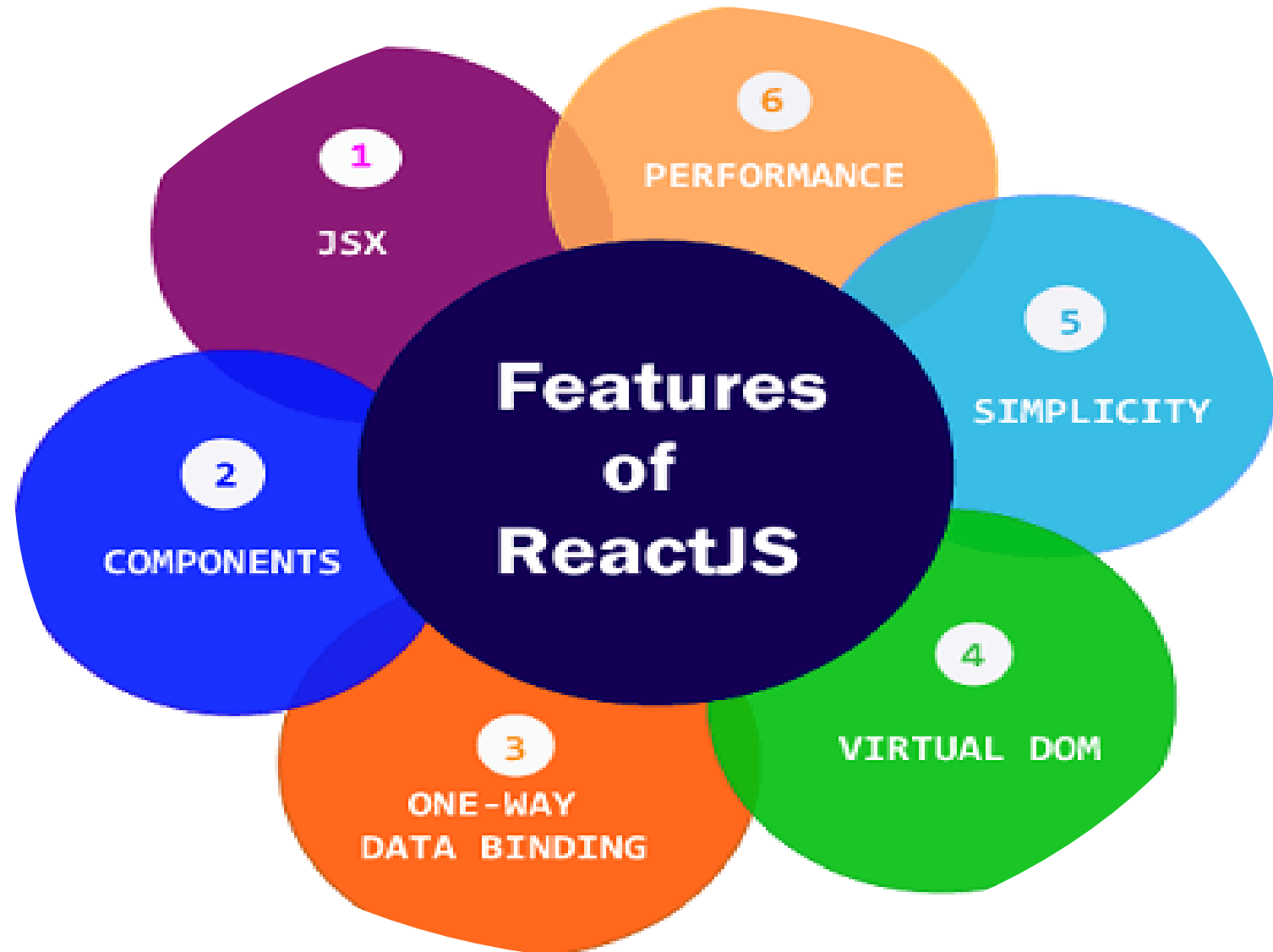


React Introduction



- **React is a JavaScript library created by Facebook to build UI as SPA.**
- **React is a tool for building reusable UI components.**
- **React Components can be reused in Angular and Vue.**
- **React is Declarative.**
- **Seamlessly integrate React into any of your application.**

Features of ReactJS





JSX

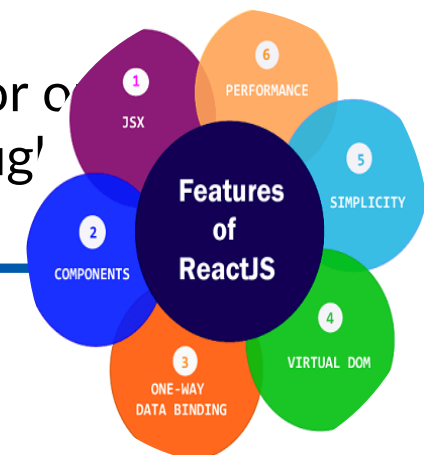
JSX stands for JavaScript XML.

Components

ReactJS components has its own logic and controls. These components can be reusable.

One-way Data Binding

ReactJS is designed in such a manner that follows unidirectional data flow or one-way data binding. The benefits of one-way data binding give you better control through your application.



Virtual DOM

A virtual DOM object is a representation of the original DOM object and works in one-way data binding.

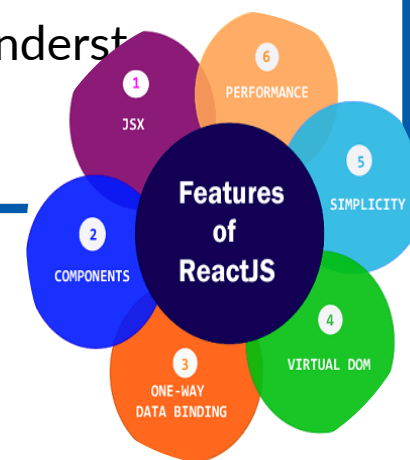
The entire UI is re-rendered in virtual DOM for the modifications in the web application and it checks the difference between the previous DOM representation and new DOM and update the changes.

Simplicity

ReactJS uses JSX file which makes the application simple and to code as well as understand.

Performance

ReactJS is a great performer because of virtual DOM.



- Using the npm command
- Using the create-react-app



Installing ReactJS Using the create-react-app command



Create folder react in Desktop

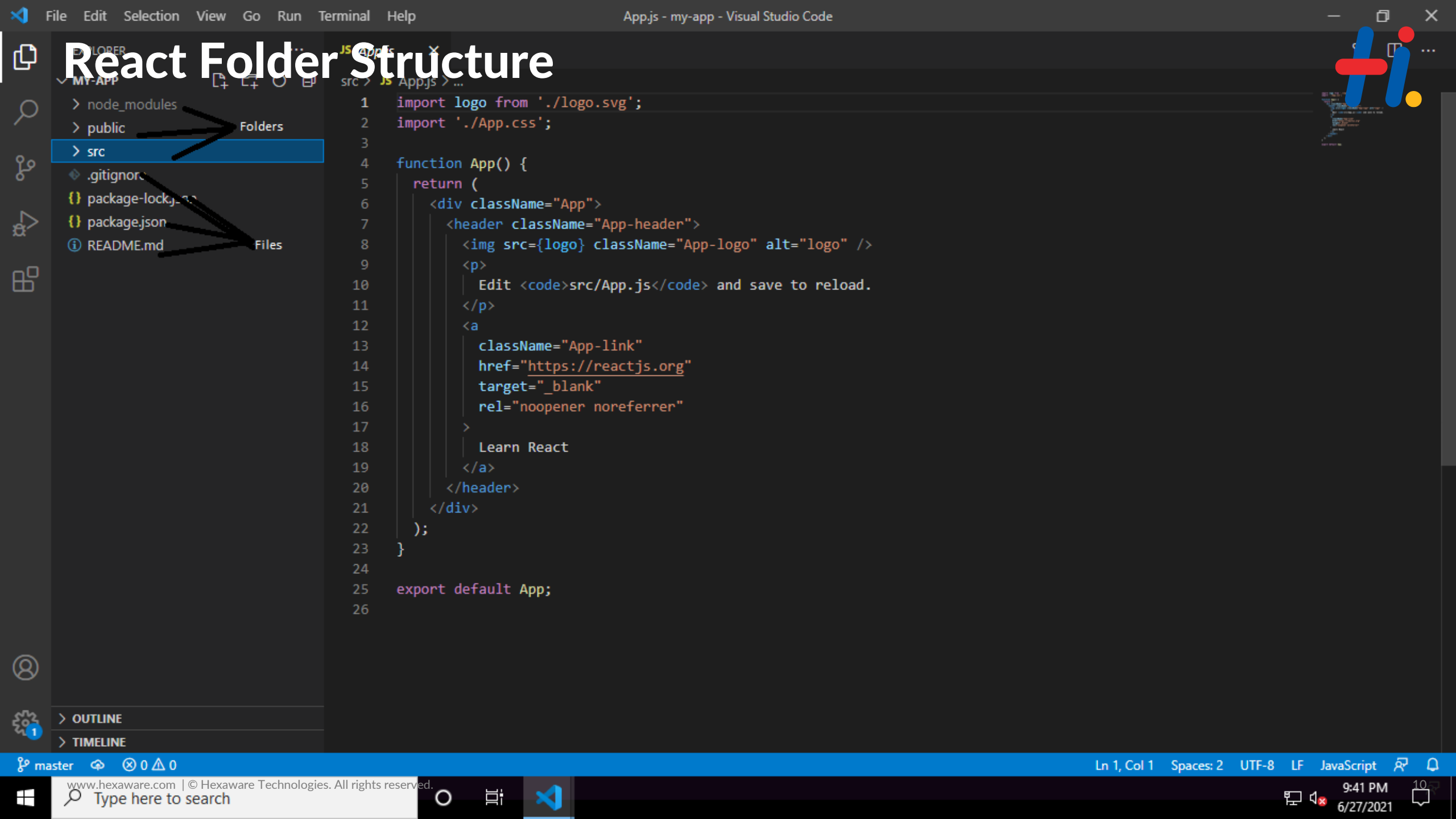
```
C:\Users\username>cd C:\Users\28695\Desktop\react
```

Create the react project using create-react-app

```
C:\Users\Tutorialspoint\Desktop> create-react-app my-app
```

Check the execution of the demo:

```
npm start
```



React Folder Structure

> node_modules

> public

> src

◆ .gitignore

{ } package-lock.json

{ } package.json

{ i } README.md

Folders

Files

```
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;
```

> OUTLINE

> TIMELINE

Ln 1, Col 1 Spaces: 2 UTF-8 LF JavaScript

www.hexaware.com | © Hexaware Technologies. All rights reserved.

Type here to search

9:41 PM
6/27/2021

Package.JSON

```
1 {  
2   "name": "my-app",  
3   "version": "0.1.0",  
4   "private": true,  
5   "dependencies": {  
6     "@testing-library/jest-dom": "^5.14.1",  
7     "@testing-library/react": "^11.2.7",  
8     "@testing-library/user-event": "^12.8.3",  
9     "react": "^17.0.2",  
10    "react-dom": "^17.0.2",  
11    "react-scripts": "4.0.3",  
12    "web-vitals": "^1.1.2"  
13  },  
14  "scripts": {  
15    "start": "react-scripts start",  
16    "build": "react-scripts build",  
17    "test": "react-scripts test",  
18    "eject": "react-scripts eject"  
19  },  
20  "eslintConfig": {  
21    "extends": [  
22      "react-app",  
23      "react-app/jest"  
24    ]  
25  },  
26  "browserslist": {  
27    "production": [  
28      ">0.2%",  
29      "not dead",  
30      "not op_mini all"  
31    ],  
32    "development": [  
33
```

Package.JSON keeps track of all the dependencies and scripts required for the project in JSON format.

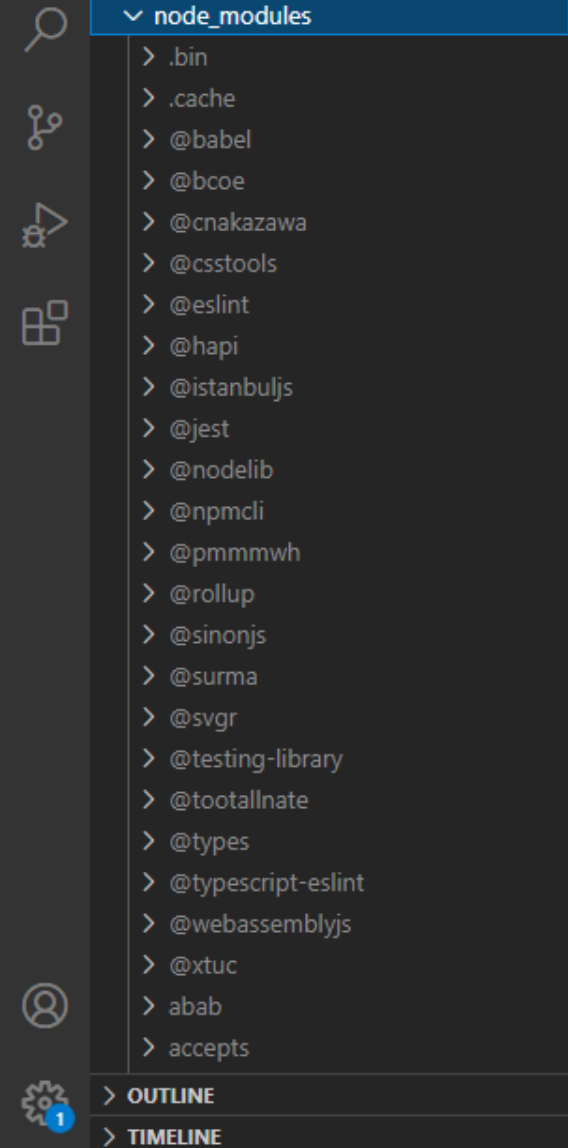
Package-lock.JSON

```
1 {
2   "name": "my-app",
3   "version": "0.1.0",
4   "lockfileVersion": 1,
5   "requires": true,
6   "dependencies": {
7     "@babel/code-frame": {
8       "version": "7.14.5",
9       "resolved": "https://registry.npmjs.org/@babel/code-frame/-/code-frame-7.14.5.tgz",
10      "integrity": "sha512-9V3UjI1/t9Hh0tf5ZD03M5+IhfdKx03q9tJqdM6B7PHF8QJ2LZhHqvtbZnI8Bdf9VJj8y1i2QzFTxp4PzQ==",
11      "requires": {
12        "@babel/highlight": "^7.14.5"
13      }
14    },
15    "@babel/compat-data": {
16      "version": "7.14.7",
17      "resolved": "https://registry.npmjs.org/@babel/compat-data/-/compat-data-7.14.7.tgz",
18      "integrity": "sha512-b13mR0w1j3aVdY683KJ5151kPFH0L1XW1W0L3xLmKsUmm+MgKt5eYf83XocYd7qR2PDuSRDi+UWUd39F0==",
19    },
20    "@babel/core": {
21      "version": "7.12.3",
22      "resolved": "https://registry.npmjs.org/@babel/core/-/core-7.12.3.tgz",
23      "integrity": "sha512-lzJZarS1Bk2gSVrqnKPNp4NFkx5DwB6IQgpfYxXzs8oBgPgruT5G1Uw+ZOXldJbwC5TVNpYeDzF2oUwZbWp4A==",
24      "requires": {
25        "@babel/code-frame": "^7.10.4",
26        "@babel/generator": "^7.12.1",
27        "@babel/helper-module-transforms": "^7.12.1",
28        "@babel/helpers": "^7.12.1",
29        "@babel/parser": "^7.12.3",
30        "@babel/template": "^7.10.4",
31        "@babel/traverse": "^7.12.1",
32        "@babel/types": "^7.12.1",
33        "convert-source-map": "^1.7.0"
34      }
35    }
36  }
37}
```

Package-lock.JSON ensure the consistency of the dependency



Node Modules



node_modules folder is the repository of modules/library using inside your project.

<> index.html X

public > <> index.html > ...

```
1 <!DOCTYPE html>
```

```
2 <html lang="en">
```

```
3 | <head>
```

```
4 <meta charset="utf-8" />
```

```
5 <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
```

```
6 <meta name="viewport" content="width=device-width, initial-scale=1" />
```

```
7 <meta name="theme-color" content="#000000" />
```

8	<meta
---	-------

```
9         name="description"
```

```
10 content="Web site created using create-react-app"
```

11 />

```
12 <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo
```

13 <!--

```
17 <link rel="manifest" href="%PUBLIC_URL%/manifest.json">
```

```
20 It will be replaced with the URL of the `public` folder during the build.
```

21 Only files inside the `public` folder can be referenced from the HTML.

22

```
23 Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC_URL%/favicon.ico" will
```

```
27 <title>React App</title>
```

28 </head>

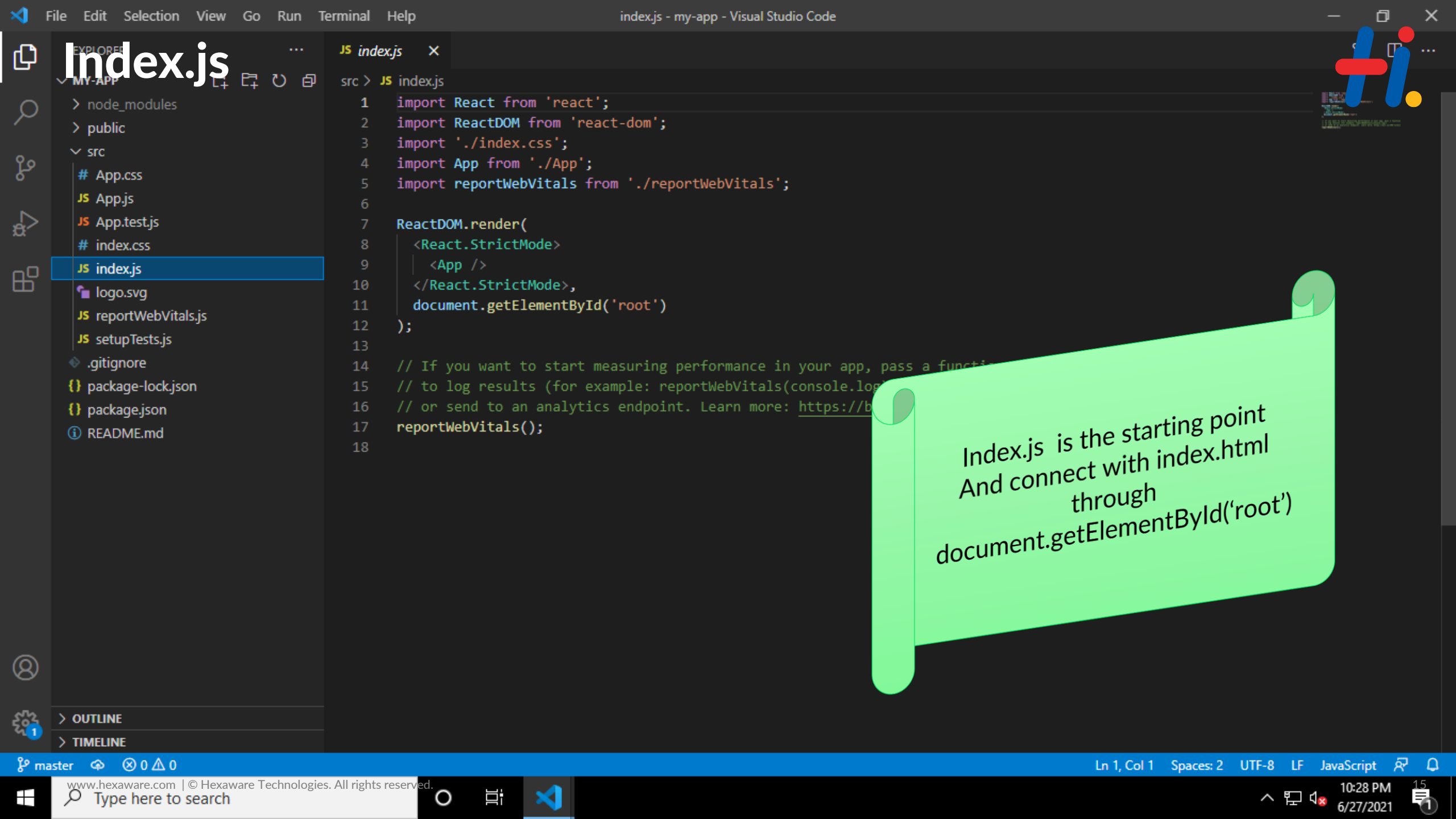
29 <body>

```
30 <noscript>You need to enable JavaScript to run this app.</noscript>
```

```
31 <div id="root"></div>
```

23 This HTML file is a template

Index.html is the only html through
it all views will render and
implementing SPA



Index.js

```
JS index.js
src > JS index.js
1 import React from 'react';
2 import ReactDOM from 'react-dom';
3 import './index.css';
4 import App from './App';
5 import reportWebVitals from './reportWebVitals';
6
7 ReactDOM.render(
8   <React.StrictMode>
9     <App />
10  </React.StrictMode>,
11  document.getElementById('root')
12 );
13
14 // If you want to start measuring performance in your app, pass a function
15 // to log results (for example: reportWebVitals(console.log))
16 // or send to an analytics endpoint. Learn more: https://b
17 reportWebVitals();
18
```

Index.js is the starting point
And connect with index.html
through
document.getElementById('root')

App.js

EXPLORER

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

OUTLINE

TIMELINE

```

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
    
```

App.js is responsible for the html displayed in the browser. App component represent the view in the browser.



React renders HTML to the web page by using a function called ReactDOM.render().

The ReactDOM.render() function takes two arguments, HTML code and an HTML element.

The render function is to display the specified HTML code inside the specified HTML element.

```
ReactDOM.render(<p>Hello</p>, document.getElementById('root'));
```



ES6





- ES6 stands for ECMAScript 6.
- ECMAScript is a standardize JavaScript.
- React uses ES6 as language.
- Main features of ES6 are:
 - Classes
 - Arrow Functions
 - Variables (let, const, var)



ES6 Class

- A class is a type of function, but instead
- of using the keyword function to initiate it,
- we use the keyword class, and the
- properties are assigned inside a
- constructor() method.

```
<!DOCTYPE html>
<html>

<body>

<script>
class Car {
  constructor(name) {
    this.brand = name;
  }
}

mycar = new Car("Ford");

document.write(mycar.brand);
</script>

</body>
</html>
```

Method in Classes



```
class Car {  
    constructor(name) {  
        this.brand = name;  
    }  
  
    present() {  
        return 'I have a ' + this.brand;  
    }  
}  
  
mycar = new Car("Ford");  
mycar.present();
```

Class Inheritance



```
class Car {  
    constructor(name) {  
        this.brand = name;  
    }  
  
    present() {  
        return 'I have a ' + this.brand;  
    }  
}
```


```
class Model extends Car {  
    constructor(name, mod) {  
        super(name);  
        this.model = mod;  
    }  
    show() {  
        return this.present() + ', it is a ' + this.model  
    }  
}  
  
mycar = new Model("Ford", "Mustang");  
mycar.show();
```


Arrow Functions



Arrow functions allow us to write shorter function syntax:

```
<!DOCTYPE html>
<html>
<body>
<h1>Arrow Function</h1>
<p>A demonstration of a simple arrow function.</p>
<p id="demo"></p>
<script>
hello = () => {
  return "Hello World!";
}
document.getElementById("demo").innerHTML = hello();
</script>
</body>
</html>
```



```
hello = function() {
  return "Hello World!";
}
```

Arrow Functions with parameter and return type



```
hello = val => "Hello " + val;
```

Without
parameter

```
hello = (val) => "Hello " + val;
```

With parameter

```
hello = function() {  
  return "Hello World!";  
}
```

With Return
Type

Variables



var

var will be considered as global scope if it declared outside of a function

Var will be considered to function scope if it declared inside the function

If you use var inside of a block like inside a for loop, the variable is still available outside of that block.

var has a function scope, not a block scope.

```
var x = 5.6;
```

Variables



let

let x = 5.6;

let has a block scope.

let is the block scoped version of var and is limited to the block (or expression) it is defined.

The variables declared inside a block like for loop will be available with the block.

const

const x = 5.6;

const is a variable that once it has been created, its value can never change.

const has a block scope.

Rest Components



React Components



Components are independent and reusable bits of code. They serve the same purpose as JavaScript functions, but work in isolation and return HTML via a `render()` function.

Components come in two types:

- Class components

- Function components

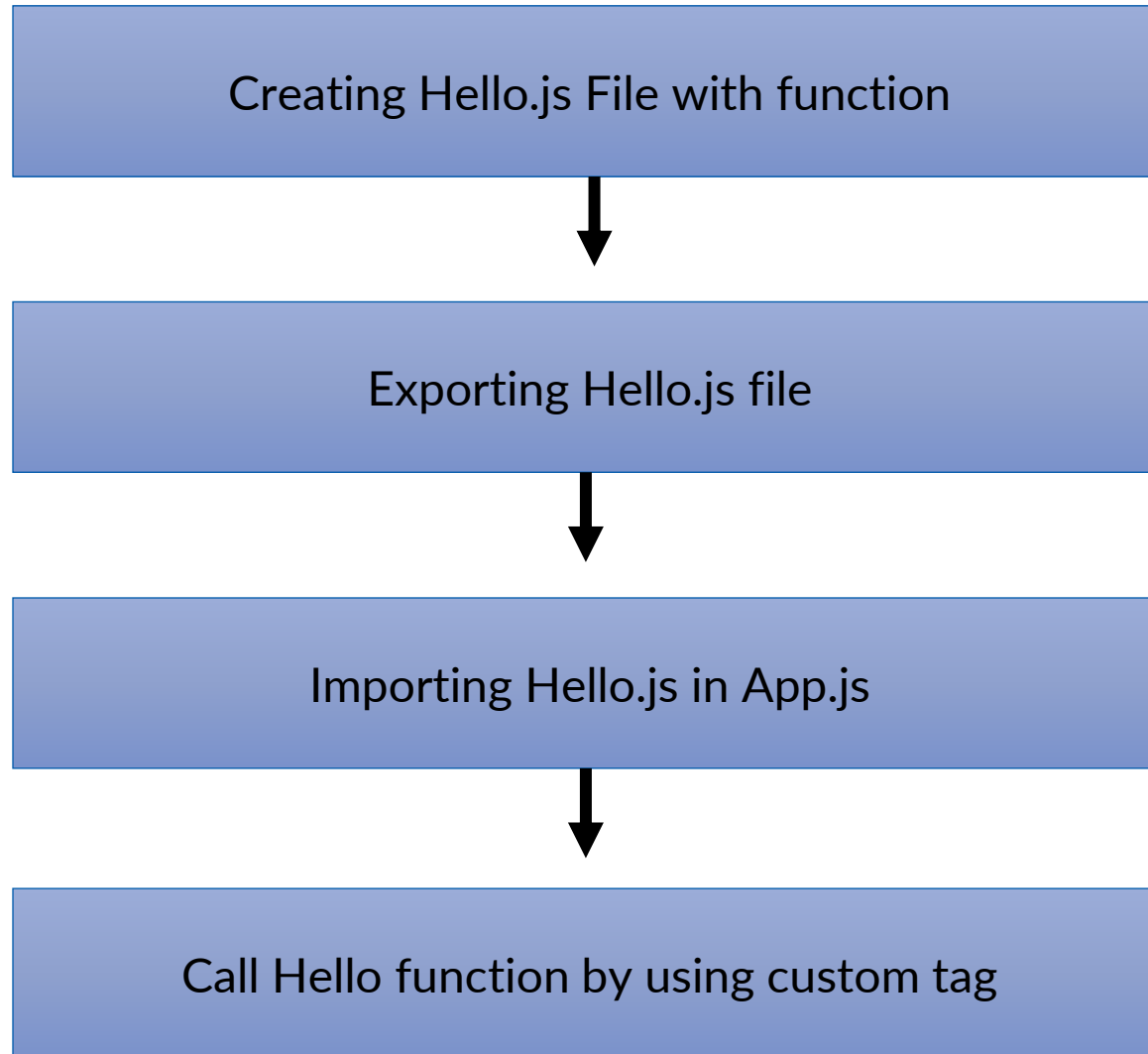
Functional Component

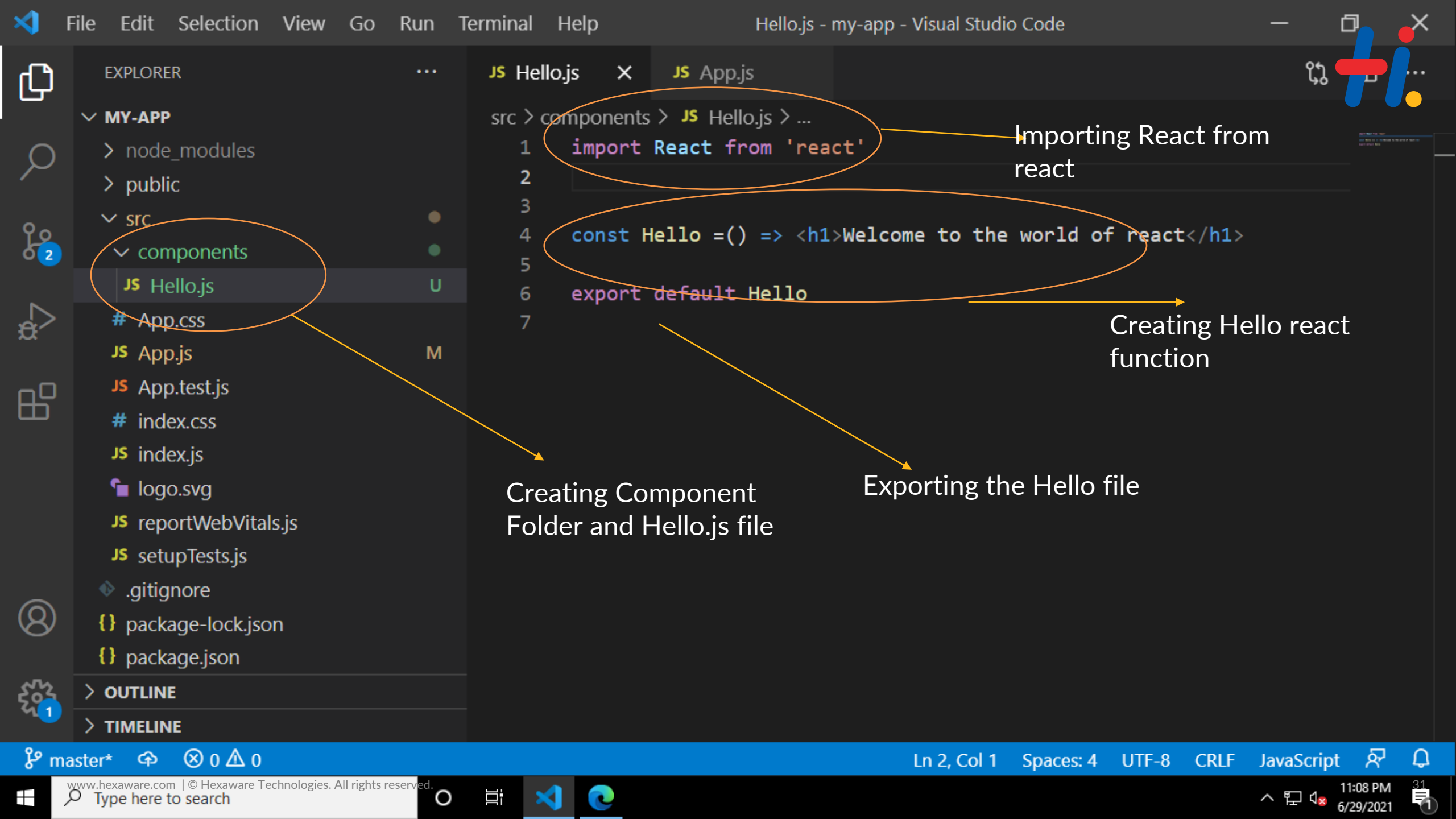


- In React, function components are JavaScript functions
- It receives optionally object of the properties(props) and returns html(JSX) for UI



Steps to creating Functional Component





EXPLORER

MY-APP

> node_modules

> public

> src

> components

JS Hello.js

App.css

JS App.js

JS App.test.js

index.css

JS index.js

logo.svg

JS reportWebVitals.js

JS setupTests.js

.gitignore

{ } package-lock.json

{ } package.json

> OUTLINE

> TIMELINE

JS Hello.js

JS App.js

src > components > JS Hello.js > ...

1 import React from 'react'

2

3

4 const Hello = () => <h1>Welcome to the world of react</h1>

5

6 export default Hello

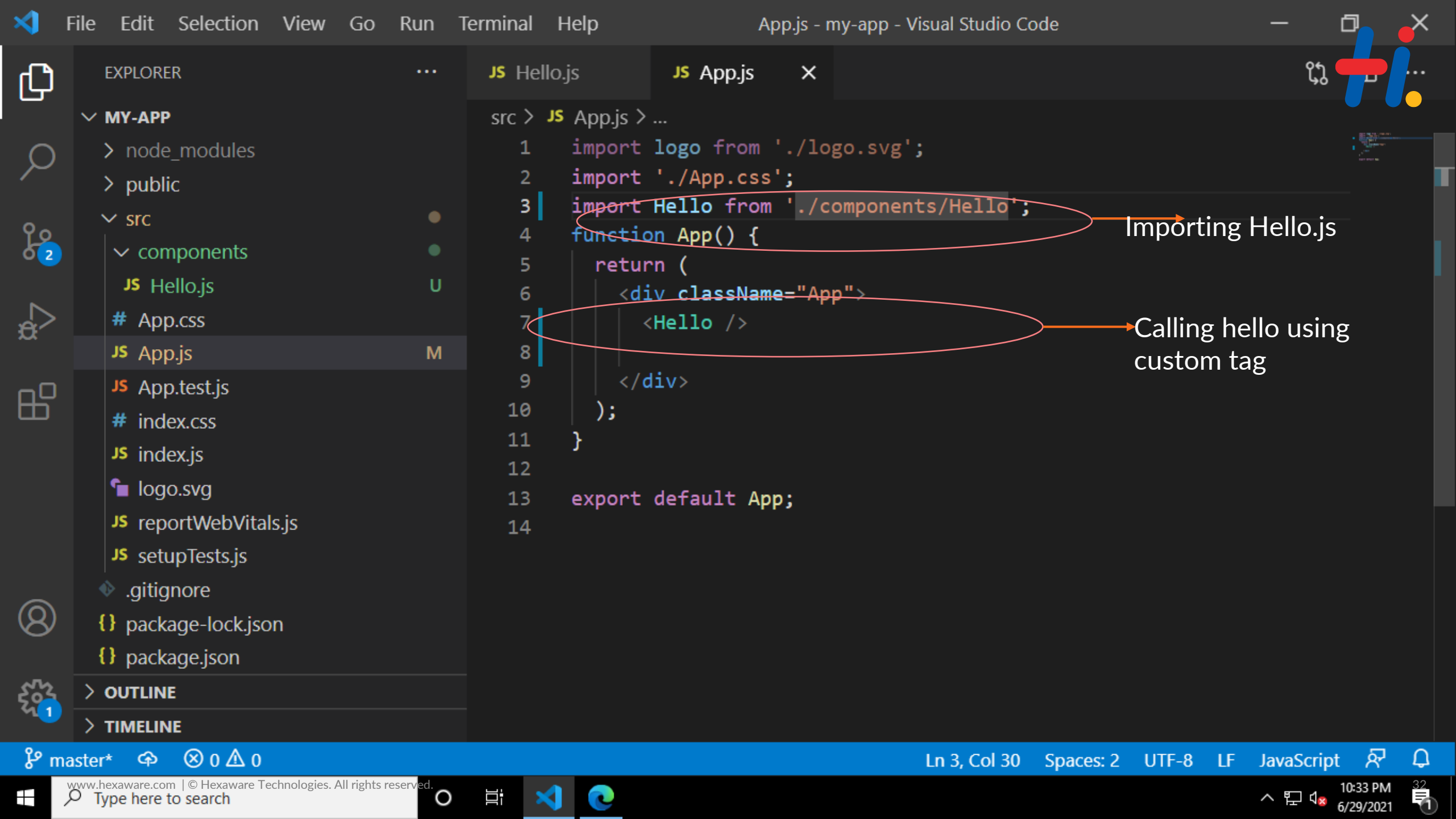
7

Importing React from
react

Creating Hello react
function

Creating Component
Folder and Hello.js file

Exporting the Hello file



EXPLORER

MY-APP

> node_modules

> public

src

components

JS Hello.js

App.css

JS App.js

JS App.test.js

index.css

JS index.js

logo.svg

JS reportWebVitals.js

JS setupTests.js

.gitignore

package-lock.json

package.json

> OUTLINE

> TIMELINE

JS Hello.js

JS App.js

src > JS App.js > ...

1 import logo from './logo.svg';

2 import './App.css';

3 import Hello from './components/Hello';

4 function App() {

5 return (

6 <div className="App">

7 <Hello />

8

9 </div>

10);

11 }

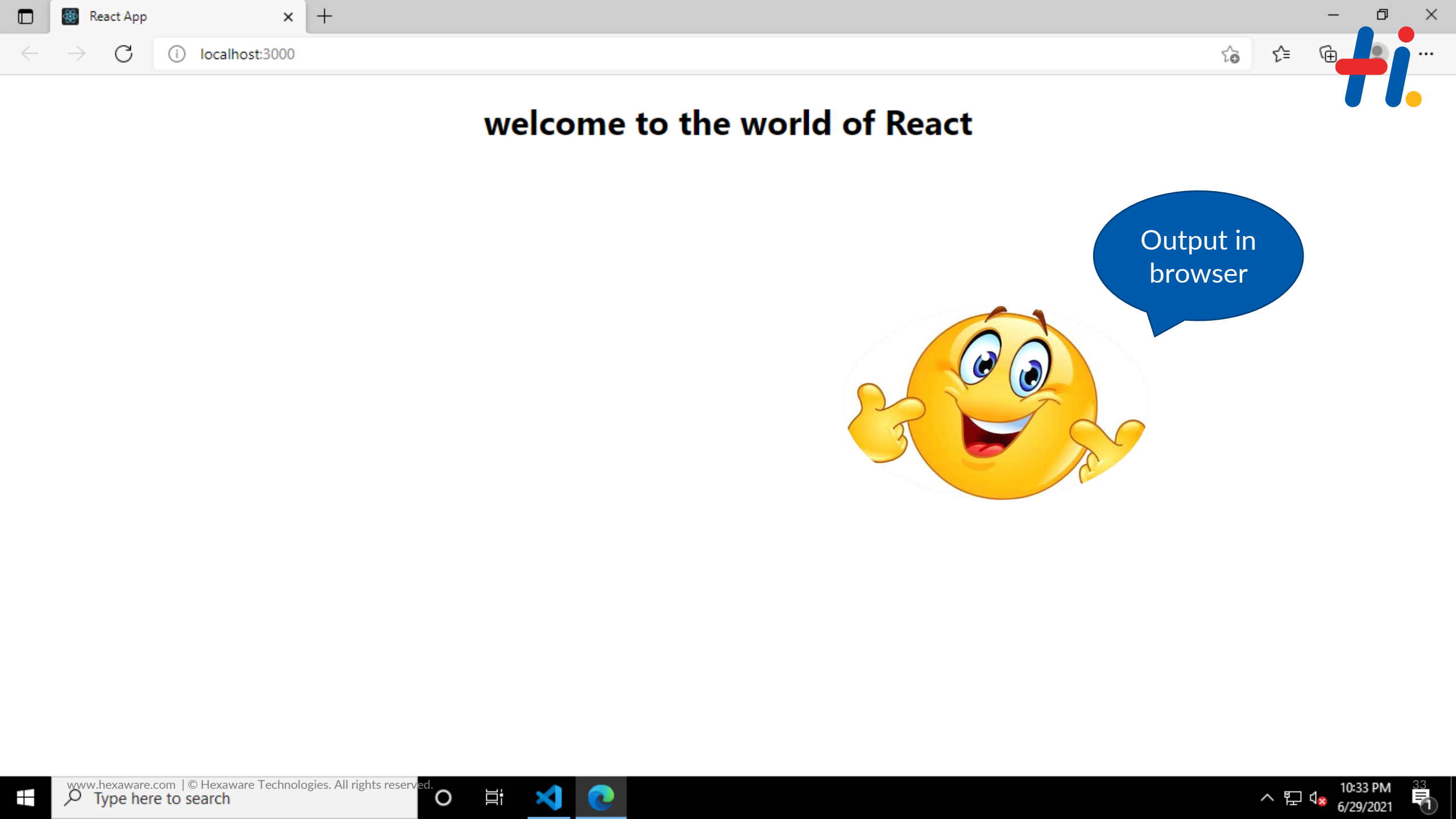
12

13 export default App;

14

Importing Hello.js

Calling hello using custom tag



welcome to the world of React



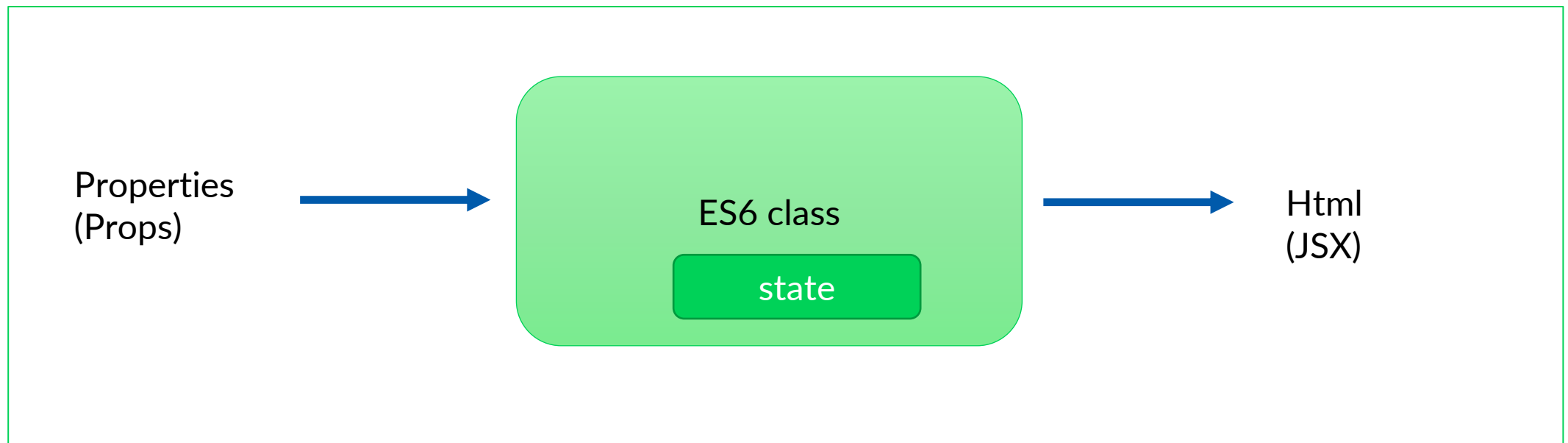
Output in
browser

Class Component

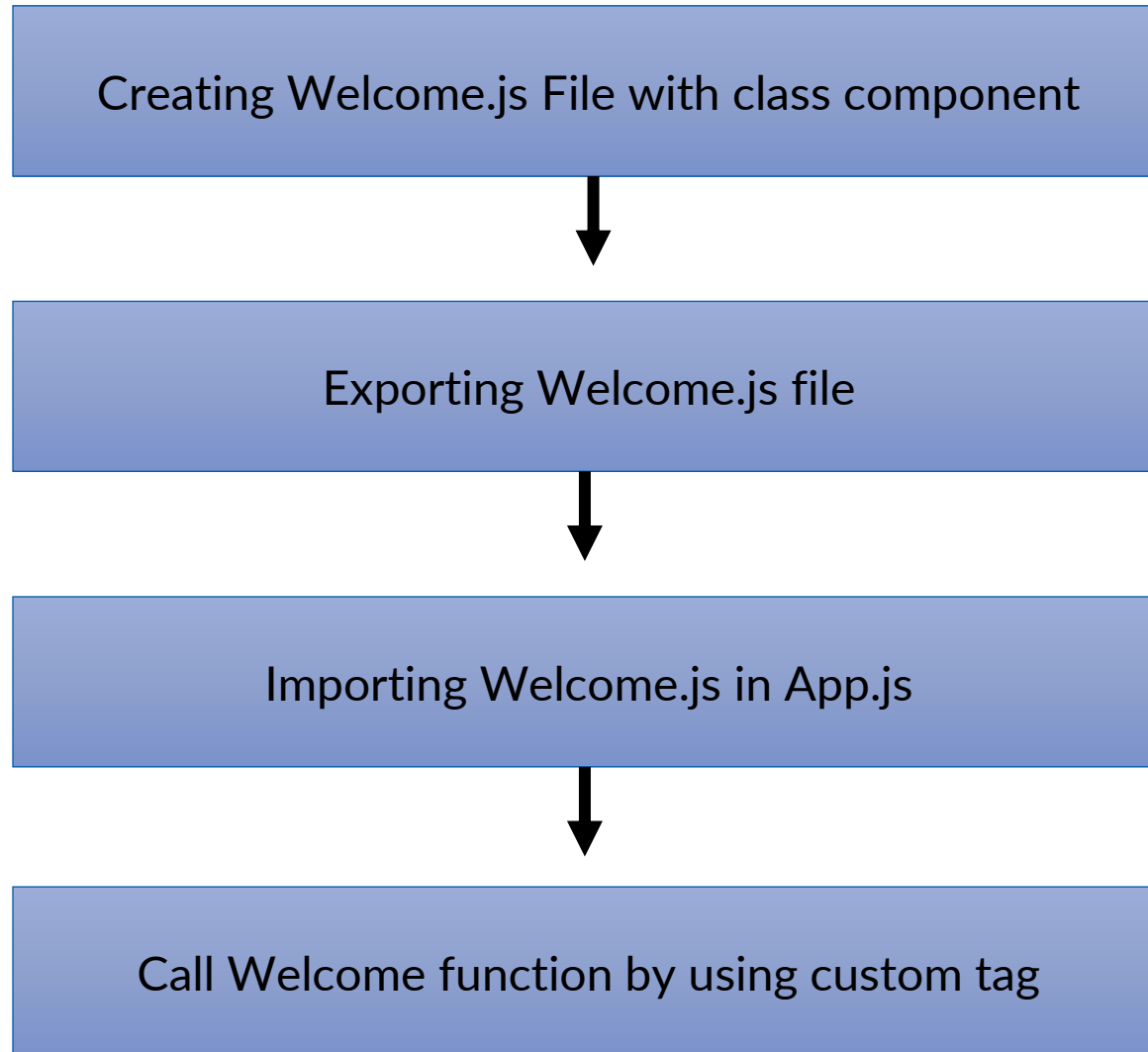


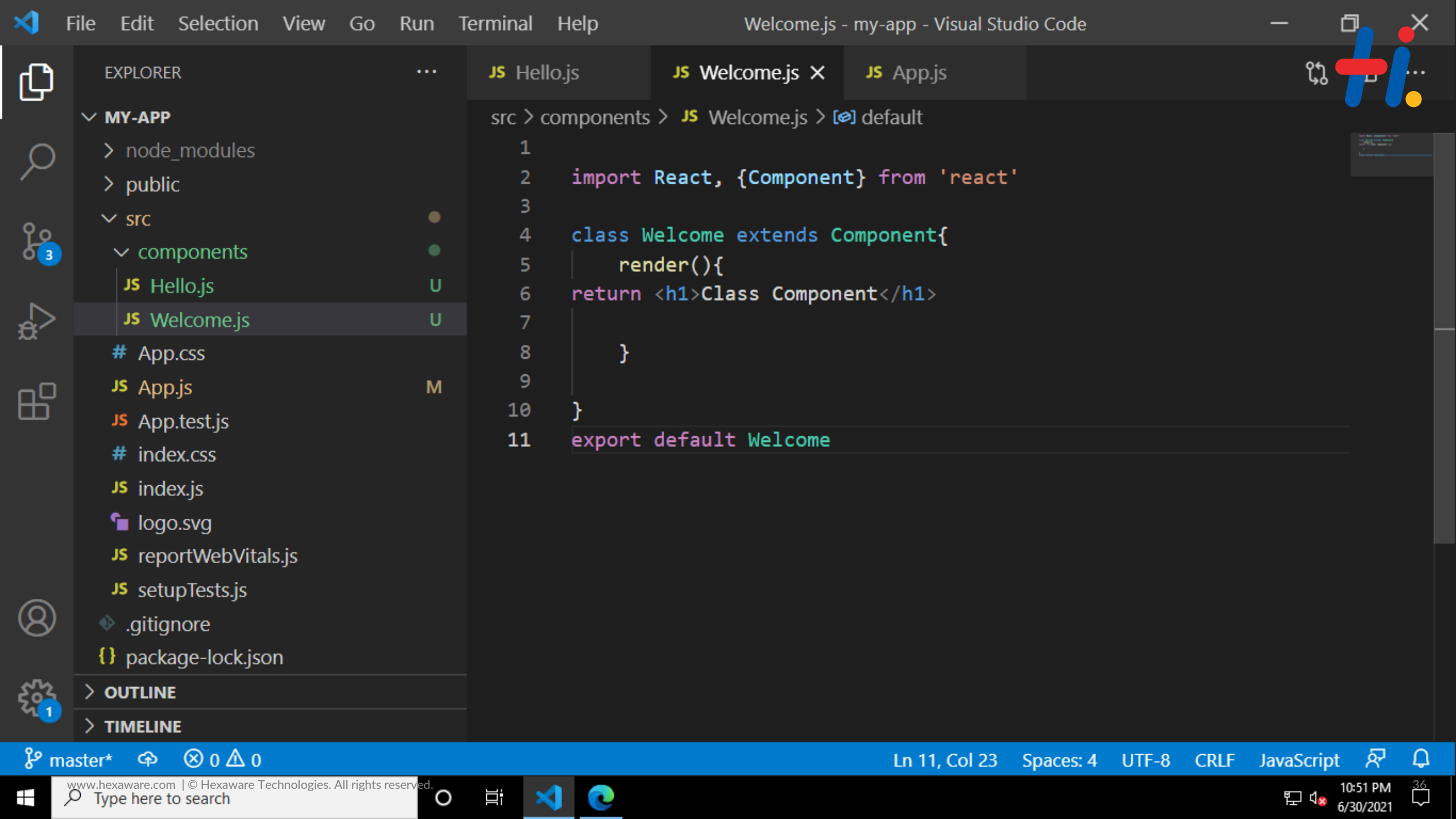
class **components** are basically ES6 classes

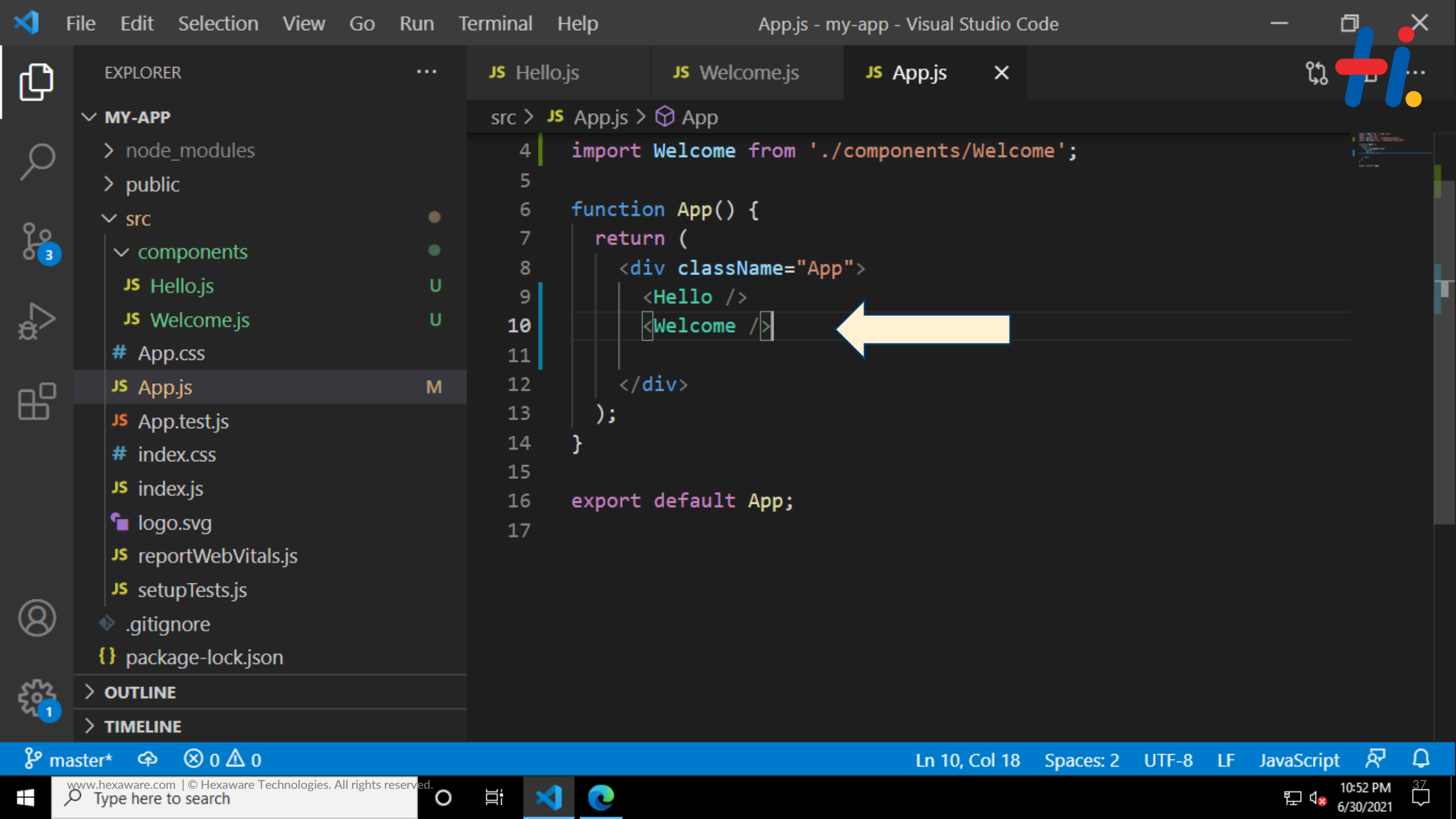
class component will maintain internal state, private to that component



Steps to creating Class Component







Functional Vs Class Component



Functional	Class
Simple Function	Rich and Complex feature
State was not maintained	Maintain their private state
Used to create simple UI	Used to create complex UI
Doesn't have Life cycle hook	Provide Life cycle hook
Stateless/Dump/Presentational	Stateful/Smart/Container



JSX stands for JavaScript XML.

React uses JSX for templating instead of regular JavaScript.

It is faster because it performs optimization while compiling code to JavaScript.

It is also type-safe and most of the errors can be caught during compilation.

It makes it easier and faster to write templates, if you are familiar with HTML.

JSX - Syntax

```
const Hello =() =>{  
  return (  
    <div>  
      <h1> Hello world </h1>  
    </div>  
  ) }  

```

Without JSX - Syntax

```
const Hello =() =>{  
  return React.createElement(  
    'div', null,React.createElement('h1',null, 'Hello World')  
  )  
}
```



Thank you

Innovative Services



Passionate Employees

Delighted Customers

