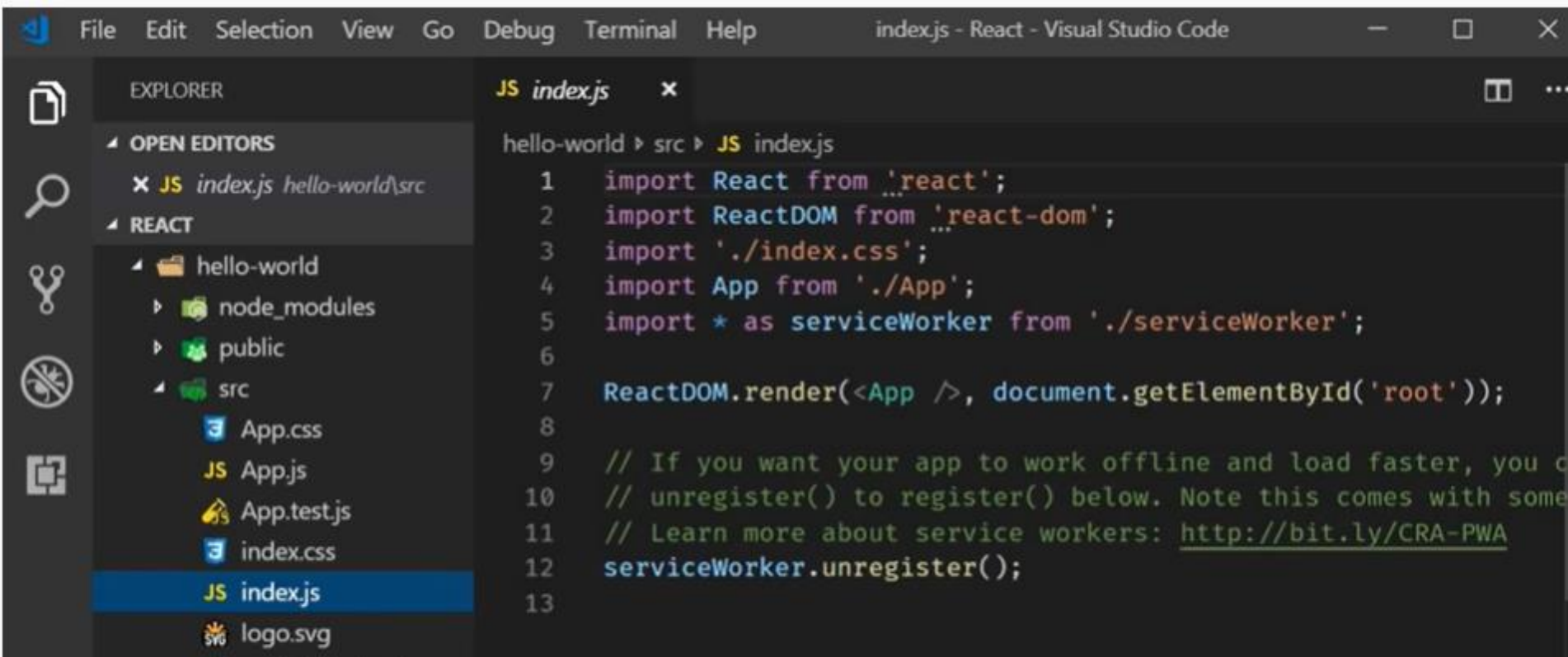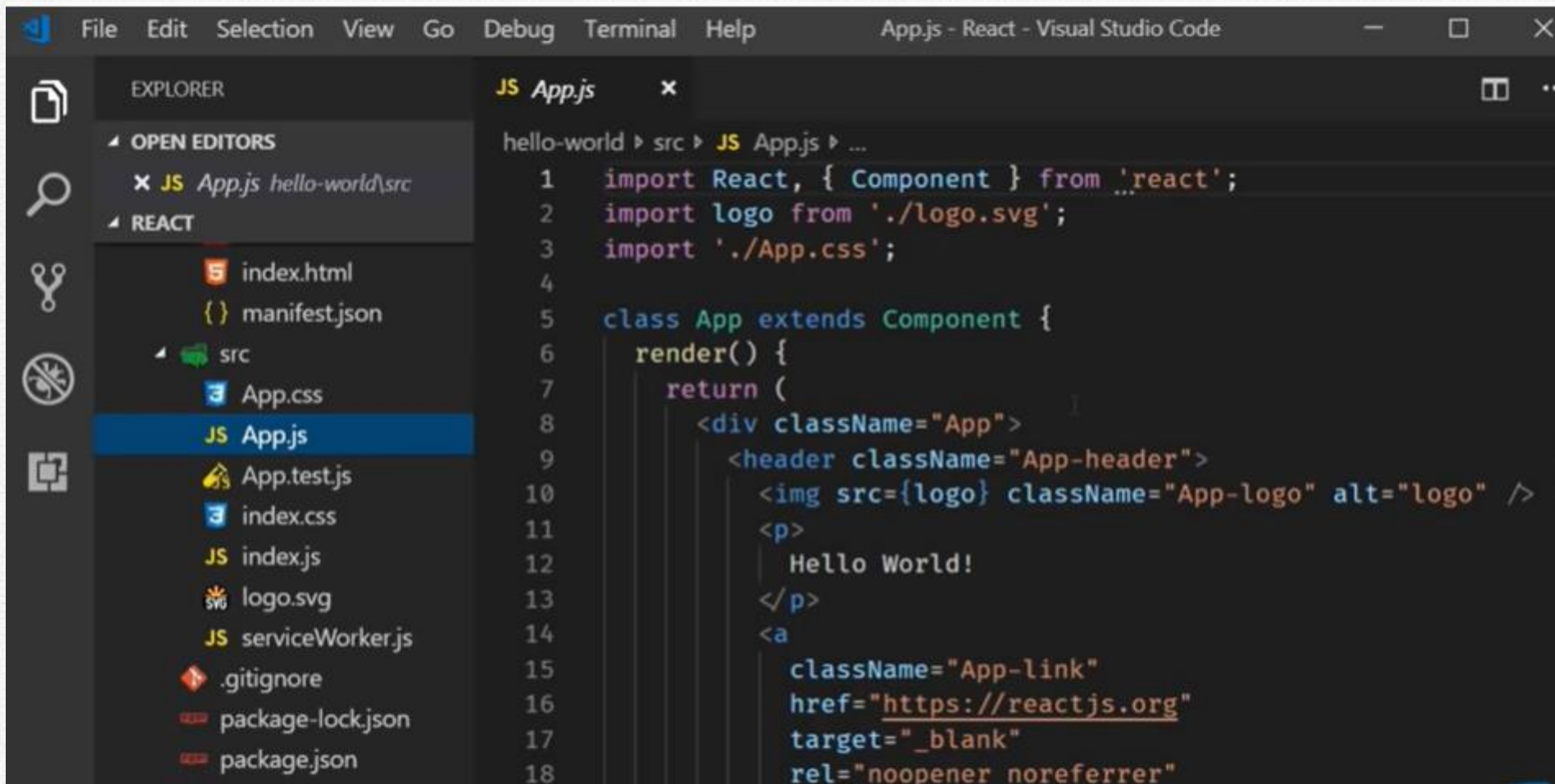Index.js :--specify the root component which is App component and DOM component control by react, Dom element is element having id root ,root element declare in index.html file App component which is present is App.js file.


```
File   Edit   Selection   View   Go   Debug   Terminal   Help          index.js - React - Visual Studio Code          —   □   ×

EXPLORER                          JS  index.js   ×                                                                □  ···

▲ OPEN EDITORS                    hello-world ▸ src ▸ JS index.js
  × JS  index.js  hello-world\src      1    import React from 'react';
▲ REACT                               2    import ReactDOM from 'react-dom';
  ▲ 📁 hello-world                     3    import './index.css';
    ▸ 📦 node_modules                  4    import App from './App';
    ▸ 📁 public                        5    import * as serviceWorker from './serviceWorker';
    ▲ 📁 src                           6
        App.css                        7    ReactDOM.render(<App />, document.getElementById('root'));
        JS App.js                      8
        App.test.js                    9    // If you want your app to work offline and load faster, you c
        index.css                     10    // unregister() to register() below. Note this comes with some
        JS index.js                   11    // Learn more about service workers: http://bit.ly/CRA-PWA
        logo.svg                      12    serviceWorker.unregister();
                                      13
```
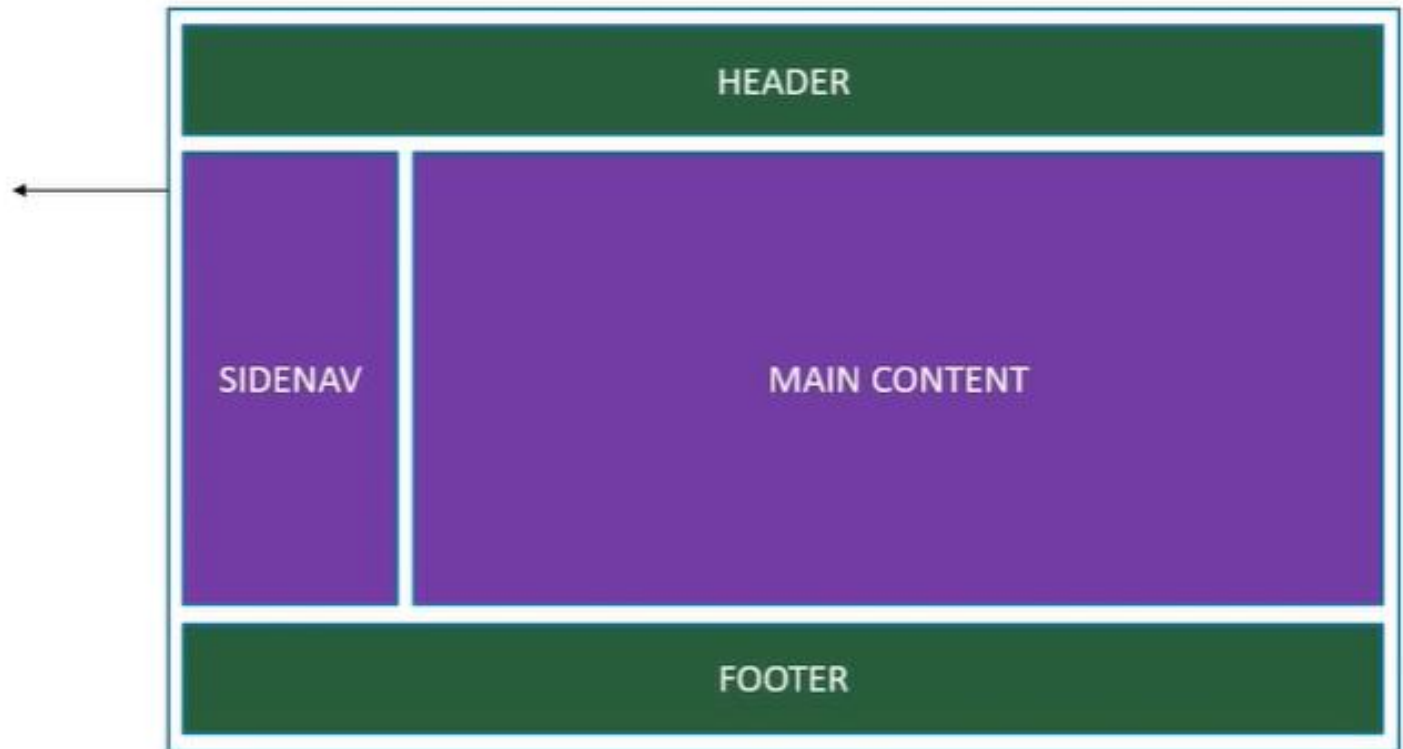
App.js:--this file contain html and display on browser

```
hello-world ▶ src ▶ JS App.js ▶ ...
1    import React, { Component } from 'react';
2    import logo from './logo.svg';
3    import './App.css';
4
5    class App extends Component {
6      render() {
7        return (
8          <div className="App">
9            <header className="App-header">
10             <img src={logo} className="App-logo" alt="logo" />
11             <p>
12               Hello World!
13             </p>
14             <a
15               className="App-link"
16               href="https://reactjs.org"
17               target="_blank"
18               rel="noopener noreferrer"
```

Components

Component in Code

JavaScript File	App.js
Component Code	AppComponent

Component Types

Stateless Functional Component

JavaScript Functions

```
function Welcome(props) {
  return <h1>Hello, {props.name}</h1>;
}
```
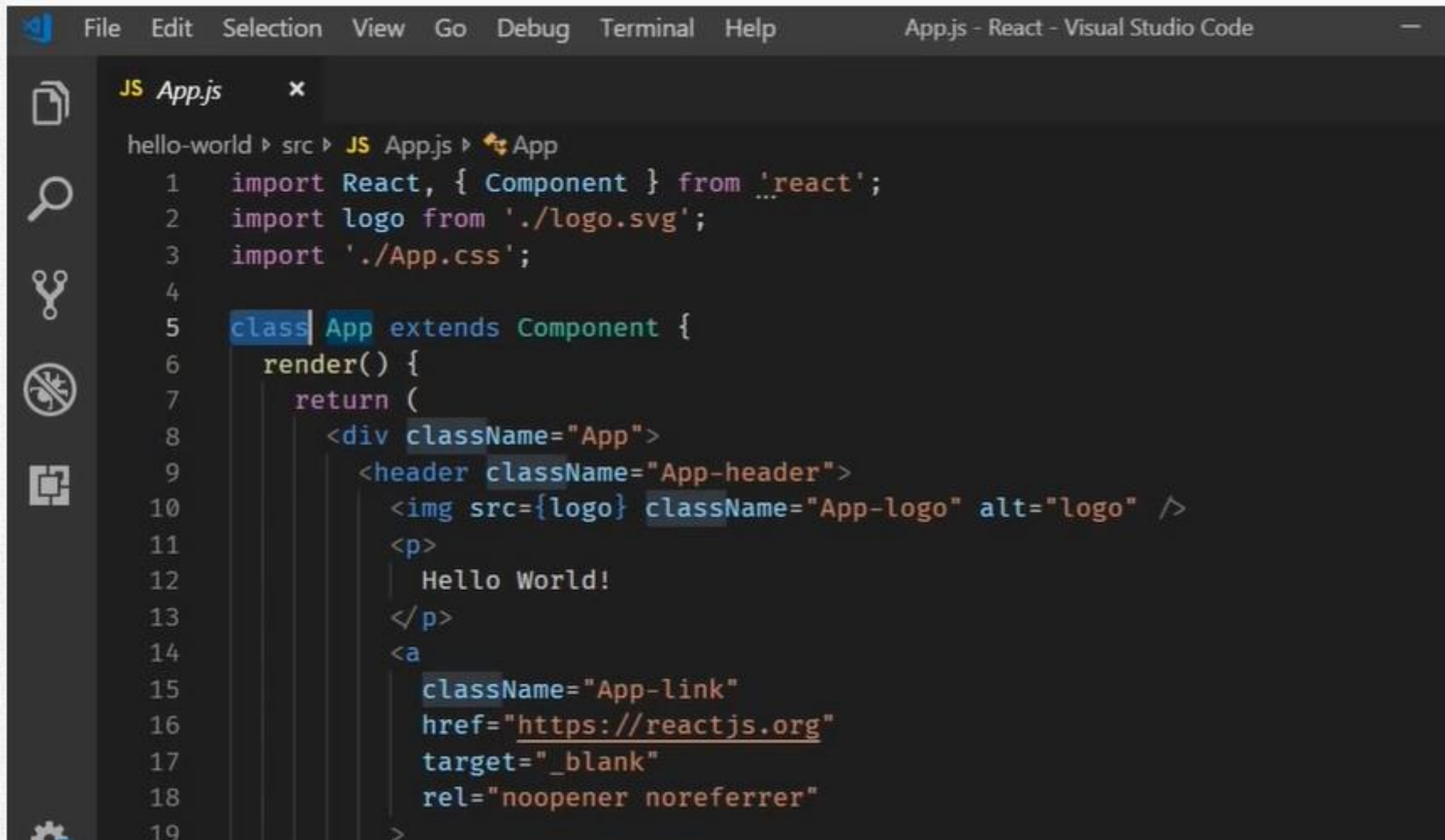
Stateful Class Component

Class extending Component class

Render method returning HTML

```
class Welcome extends React.Component {
  render() {
    return <h1>Hello, {this.props.name}</h1>;
  }
}
```

Class Component:-


```js
import React, { Component } from 'react';
import logo from './logo.svg';
import './App.css';

class App extends Component {
  render() {
    return (
      <div className="App">
        <header className="App-header">
          <img src={logo} className="App-logo" alt="logo" />
          <p>
            Hello World!
          </p>
          <a
            className="App-link"
            href="https://reactjs.org"
            target="_blank"
            rel="noopener noreferrer"
          >
```

Components Summary

Components describe a part of the user interface

They are re-usable and can be nested inside other components
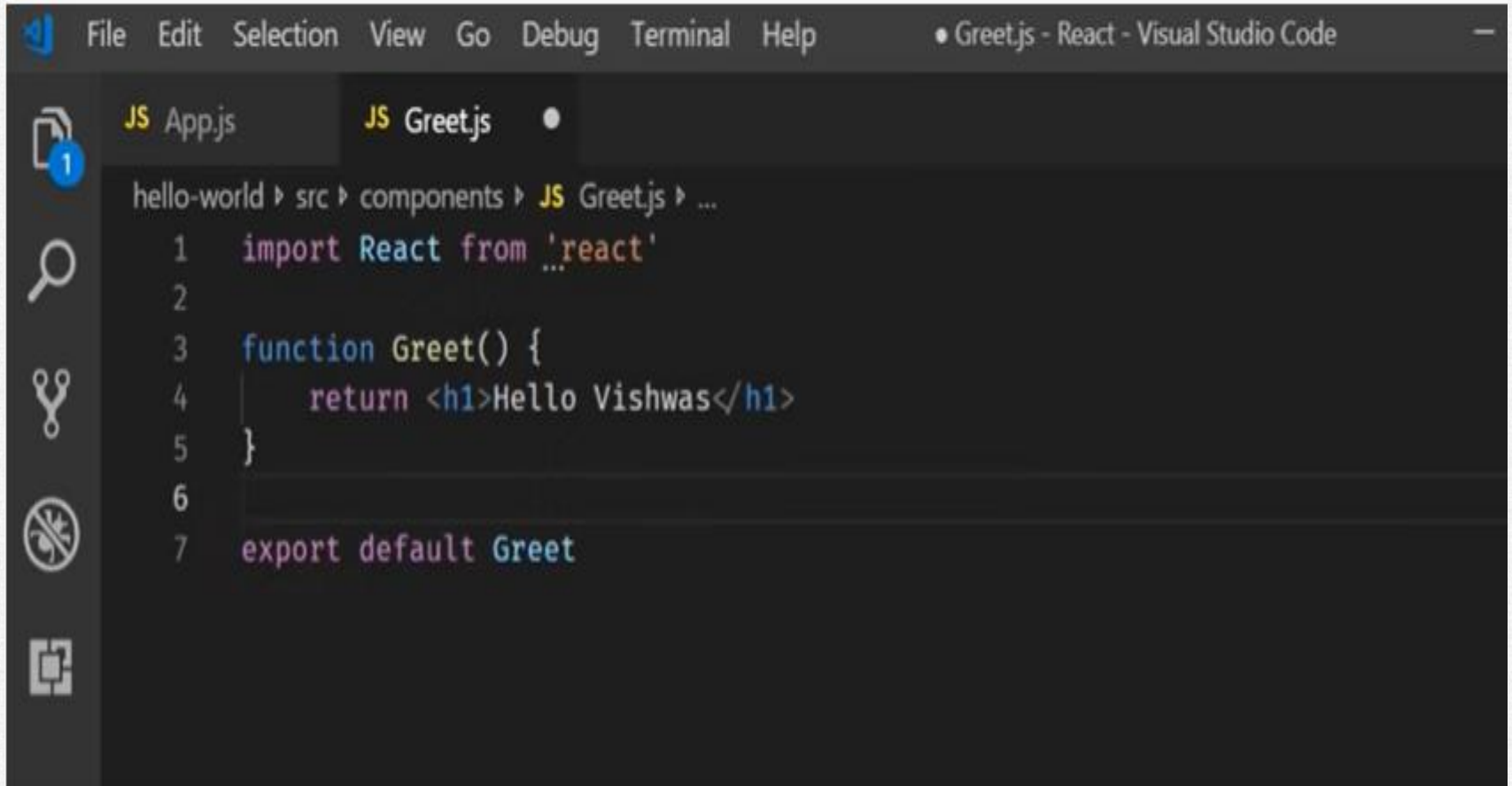
Two Types –
- ◦ Stateless Functional Components
- ◦ Stateful Class Components

Functional component is a JavaScript fun that accept in an input of property and return the html to describe the UI.
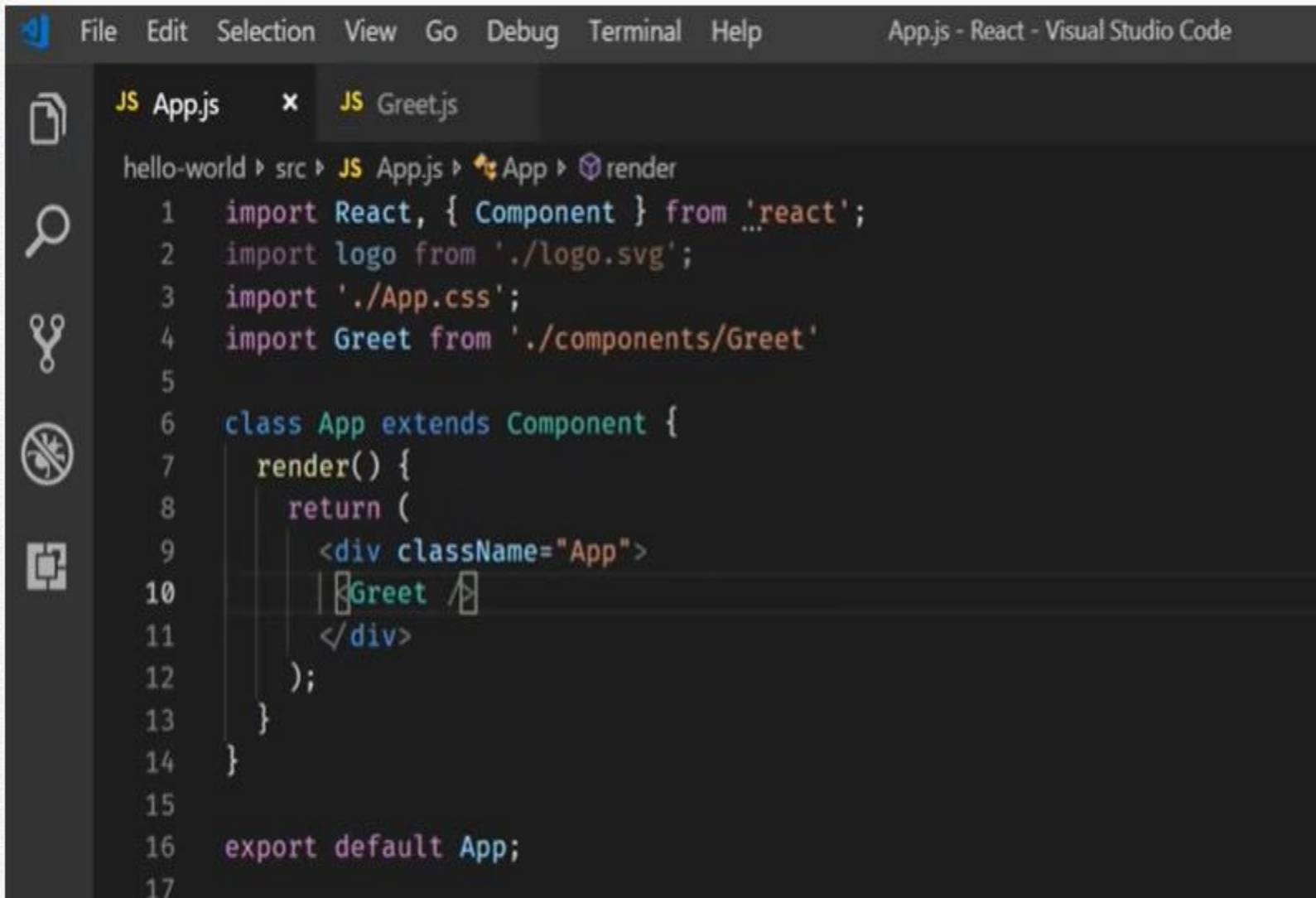
Functional Components

Properties (props) → JavaScript Function → HTML (JSX)

Create a Component folder inside that create a Greet.js File

Call this file greet.js inside App.js file

File Edit Selection View Go Debug Terminal Help   App.js - React - Visual Studio Code

JS App.js   ✕   JS Greet.js

hello-world ▸ src ▸ JS App.js ▸ ⚛ App ▸ ⊗ render

```javascript
1   import React, { Component } from 'react';
2   import logo from './logo.svg';
3   import './App.css';
4   import Greet from './components/Greet'
5
6   class App extends Component {
7     render() {
8       return (
9         <div className="App">
10          <Greet />
11        </div>
12      );
13    }
14  }
15
16  export default App;
17
```
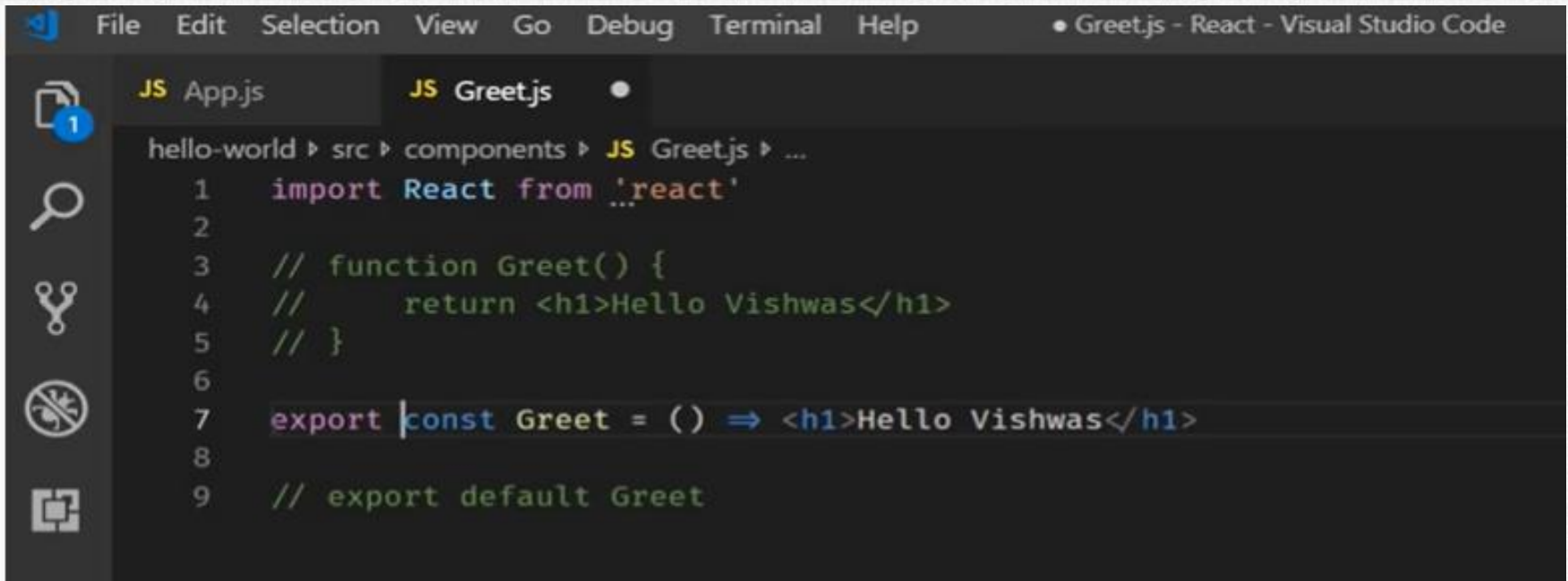
Arrow function syntax:-

If we put export default Greet:--time of import
the component give any name if not use the
export we can access the component the same
name in import.

```
JS App.js        ●      JS Greet.js

hello-world ▸ src ▸ JS App.js ▸ ⬢ App ▸ ⊗ render
  1    import React, { Component } from 'react';
  2    import logo from './logo.svg';
  3    import './App.css';
  4    import MyComponent from './components/Greet'
  5
  6    class App extends Component {
  7      render() {
  8        return (
  9          <div className="App">
 10            <MyComponent />
 11          </div>
 12        );
 13      }
 14    }
 15
 16    export default App;
 17
```

- Import the same name Greet if not its throw the error.

Thanks