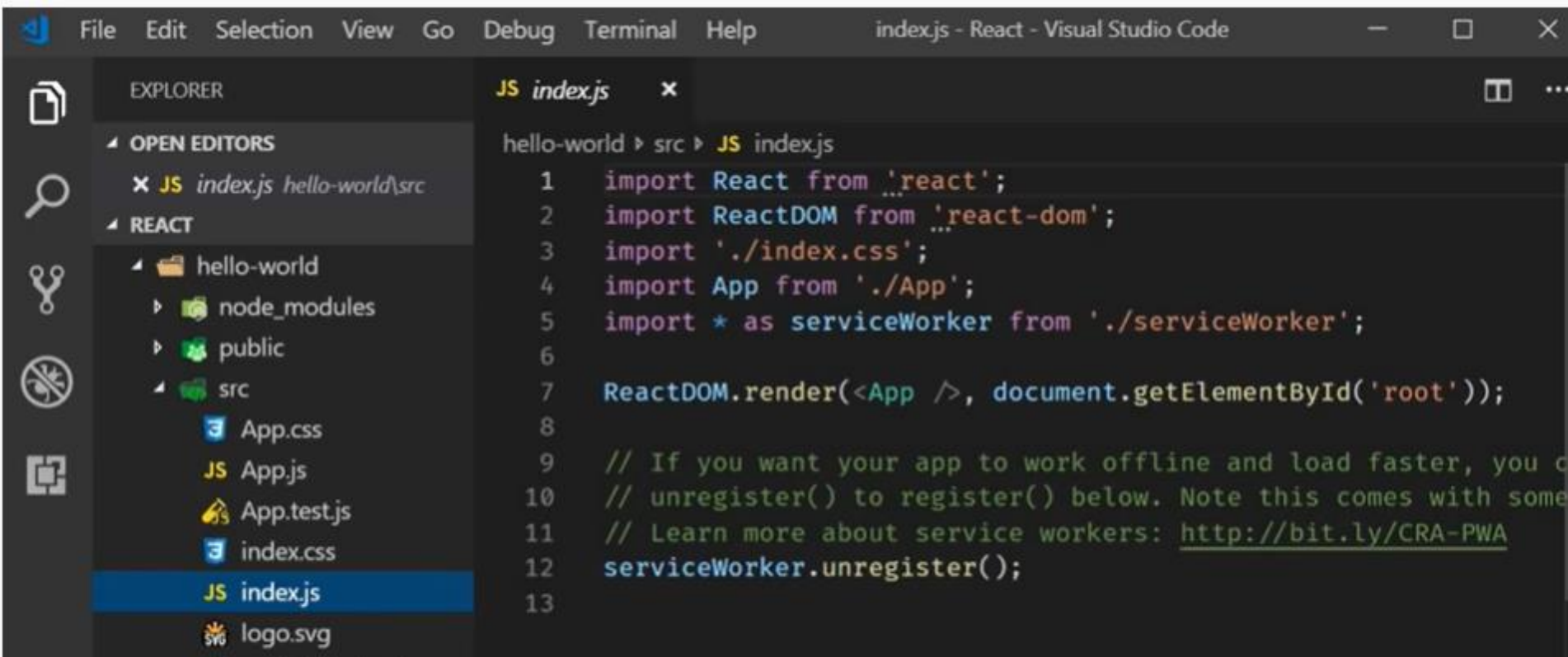


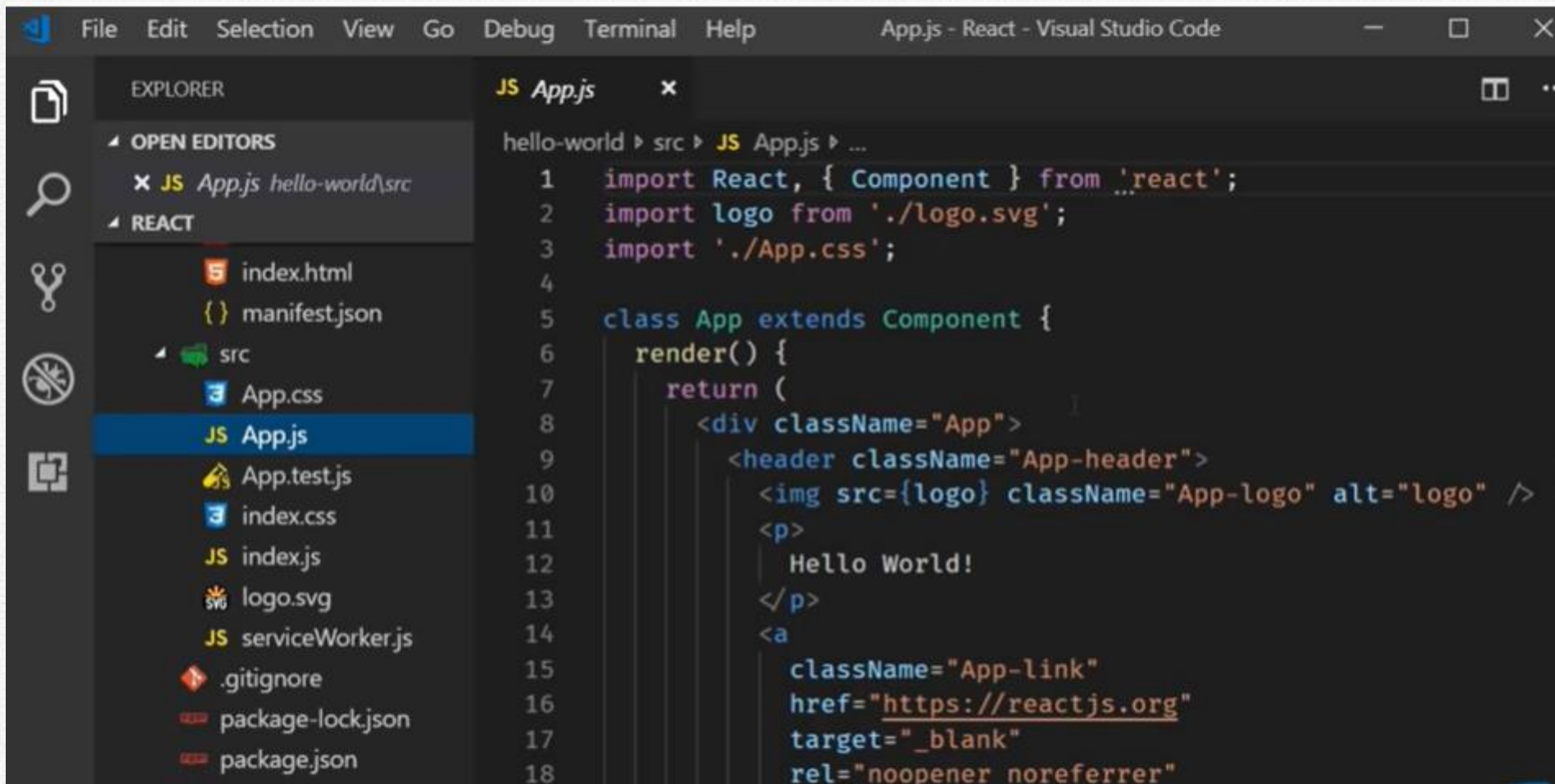
Index.js :--specify the root component which is App component and DOM component control by react, Dom element is element having id root ,root element declare in index.html file App component which is present is App.js file.



The screenshot displays the Visual Studio Code interface with the 'index.js' file open in the editor. The Explorer sidebar on the left shows the project structure, including the 'src' directory with files like 'App.css', 'App.js', 'App.test.js', 'index.css', 'index.js', and 'logo.svg'. The editor window shows the following code:

```
hello-world > 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 * as serviceWorker from './serviceWorker';
6
7  ReactDOM.render(<App />, document.getElementById('root'));
8
9  // If you want your app to work offline and load faster, you c
10 // unregister() to register() below. Note this comes with some
11 // Learn more about service workers: http://bit.ly/CRA-PWA
12 serviceWorker.unregister();
13
```

App.js:--this file contain html and display on browser

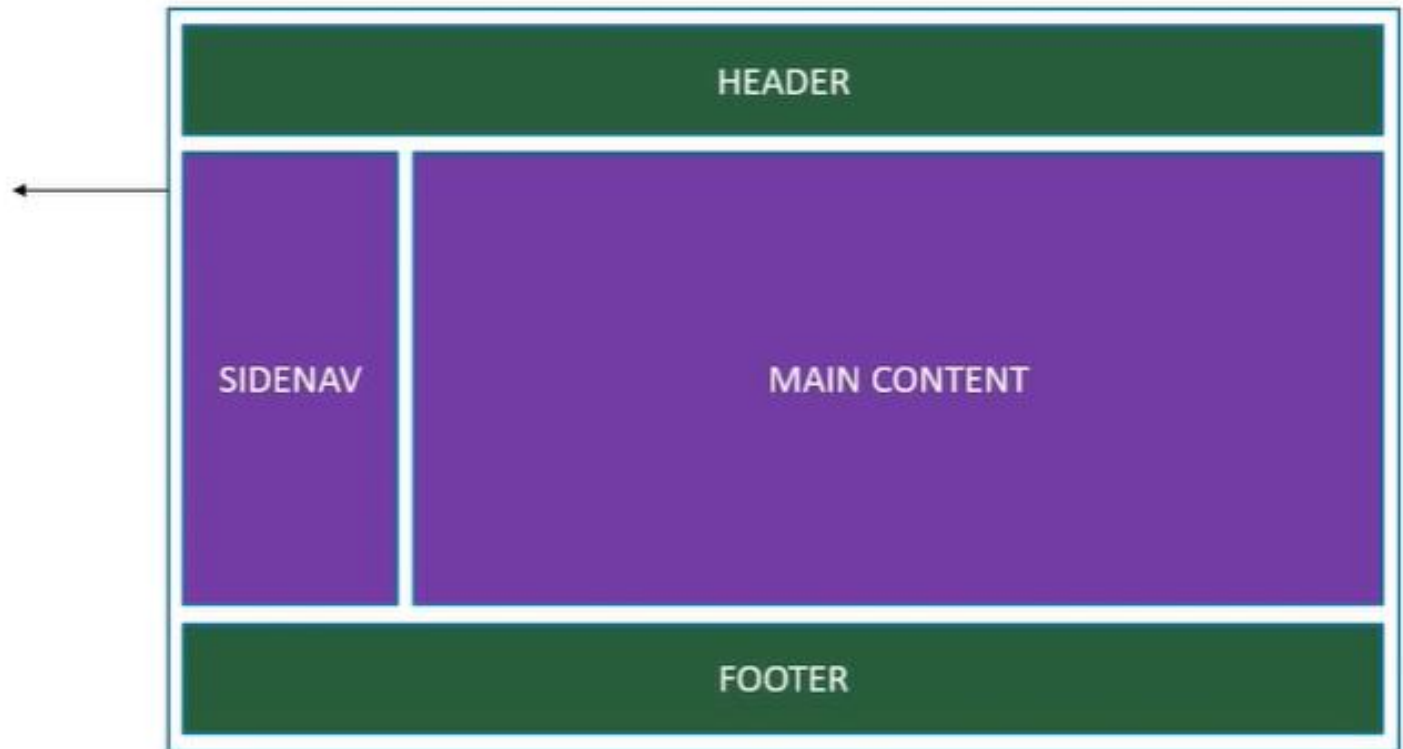


The screenshot shows the Visual Studio Code editor interface. The Explorer sidebar on the left displays the project structure for 'hello-world', including files like index.html, manifest.json, App.css, App.js (selected), App.test.js, index.css, index.js, logo.svg, serviceWorker.js, .gitignore, package-lock.json, and package.json. The main editor area shows the code for App.js, which imports React and Component, and defines a class App that extends Component. The render method returns a JSX element with a header, a logo, and a link to ReactJS.

```
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

Root (App)
Component



Component in Code

JavaScript File

Component Code

App.js

AppComponent

Component Types

Stateless Functional Component

JavaScript Functions

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

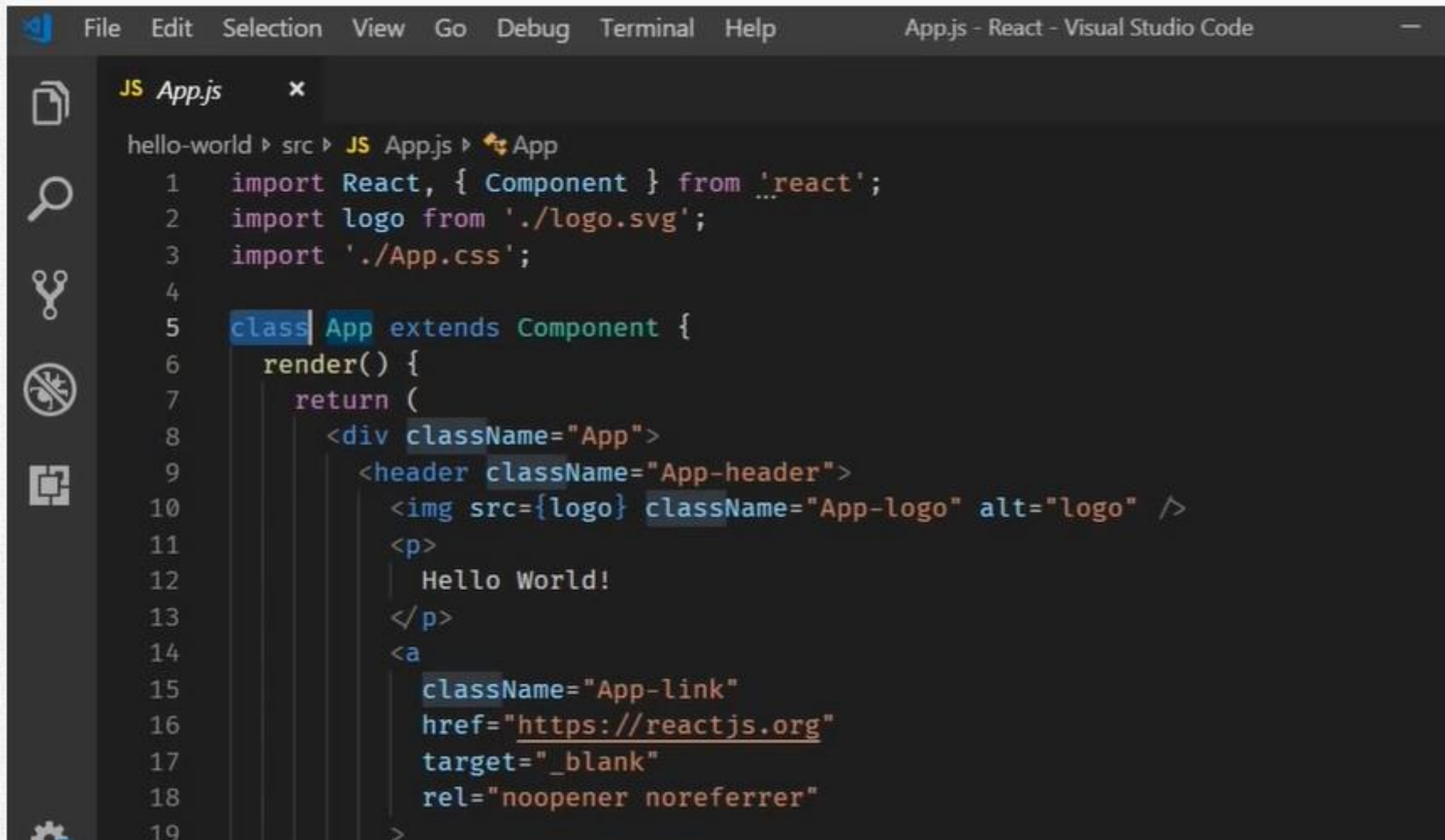
Stateful Class Component

Class extending Component class

Render method returning HTML

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

Class Component:-



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

JS App.js x
hello-world > src > JS App.js > App
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"
19             >
```

Components Summary

Components describe a part of the user interface

They are re-usable and can be nested inside other components

Two Types –

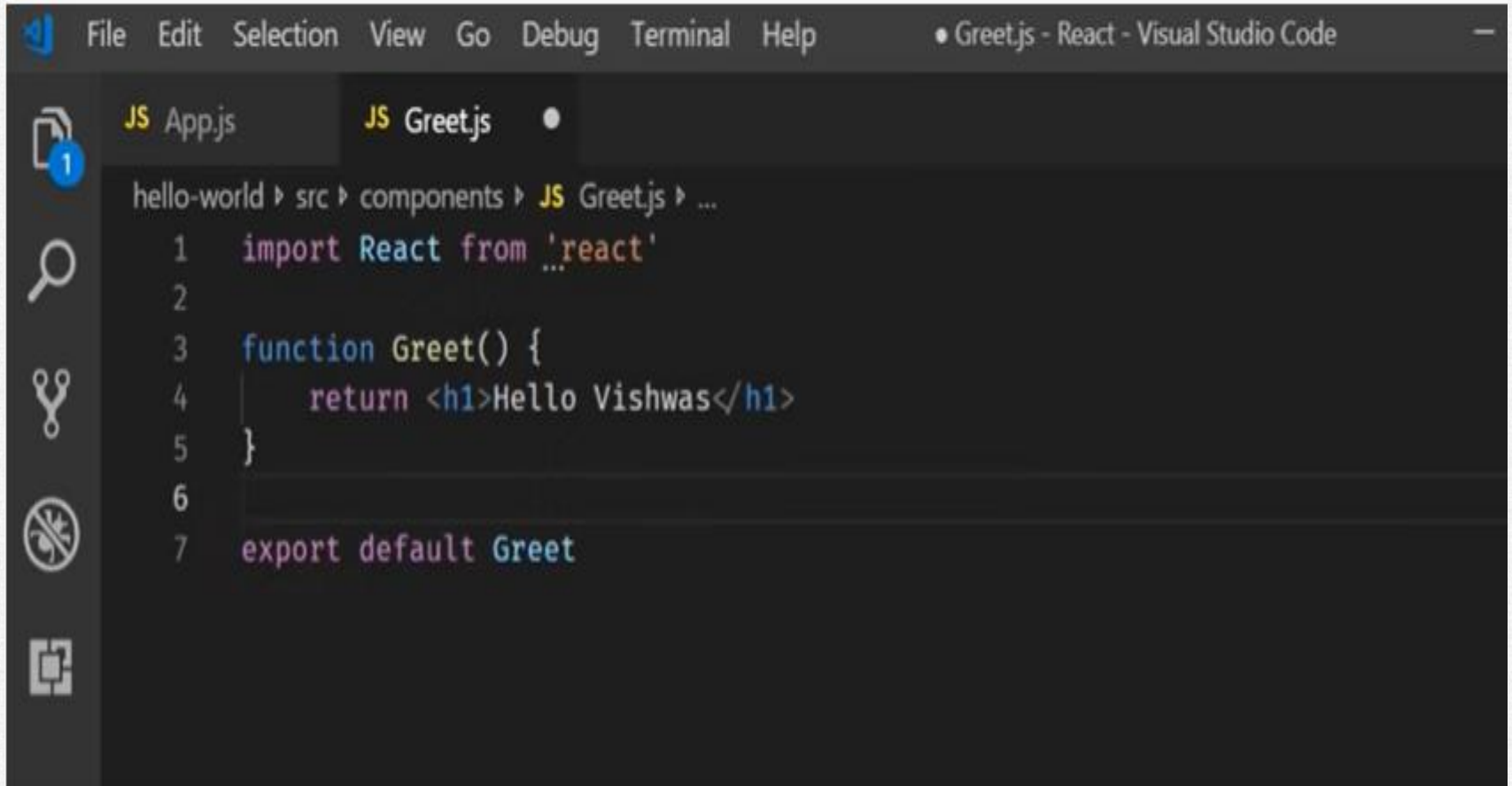
- Stateless Functional Components
- Stateful Class Components

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

Functional Components

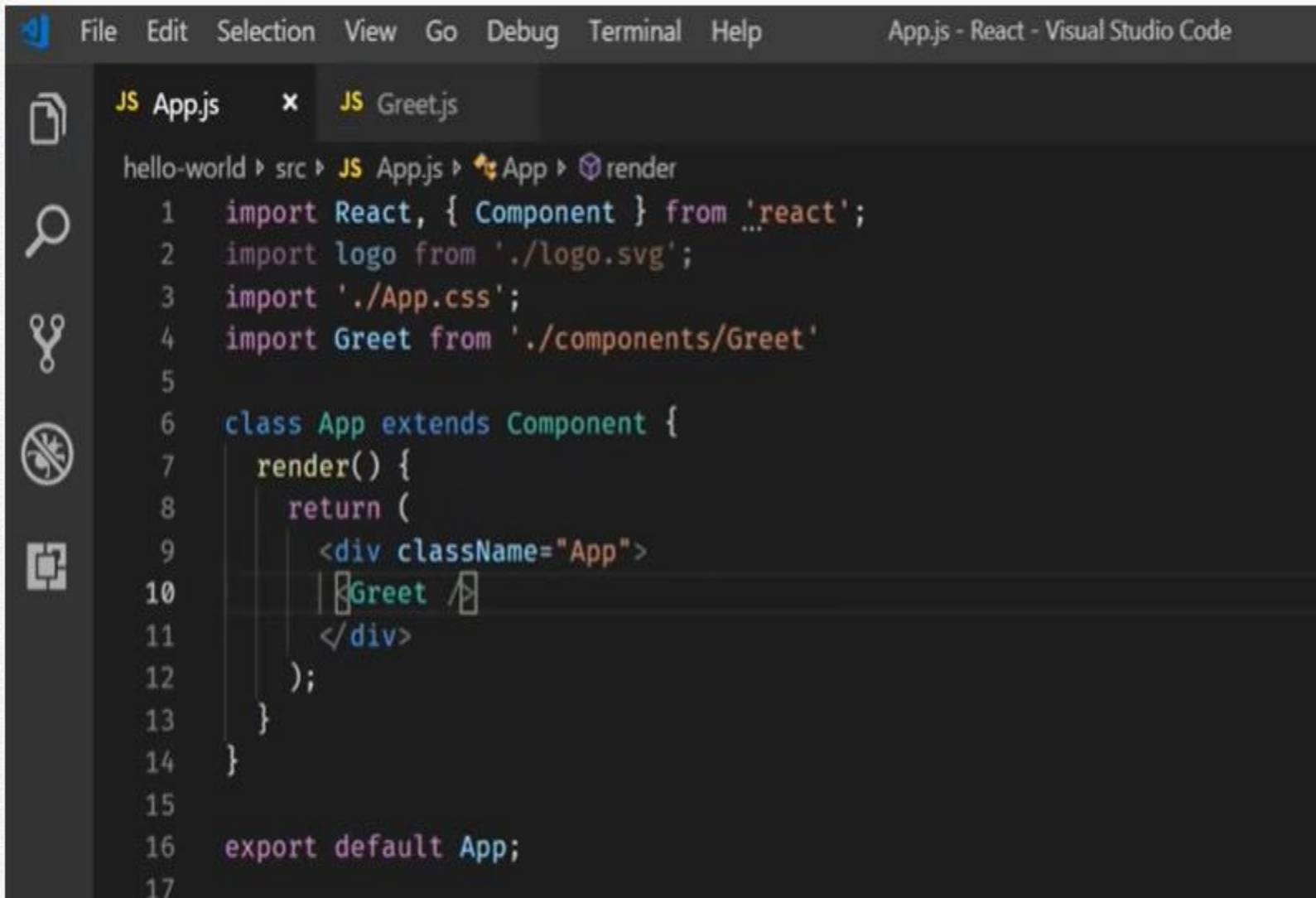


Create a Component folder inside that create a Greet.js File

A screenshot of the Visual Studio Code editor interface. The top menu bar includes File, Edit, Selection, View, Go, Debug, Terminal, and Help. The title bar shows 'Greet.js - React - Visual Studio Code'. The Explorer sidebar on the left shows a file tree with 'JS App.js' and 'JS Greet.js'. The main editor area displays the content of 'Greet.js' with the following code:

```
1 import React from 'react'
2
3 function Greet() {
4   return <h1>Hello Vishwas</h1>
5 }
6
7 export default Greet
```

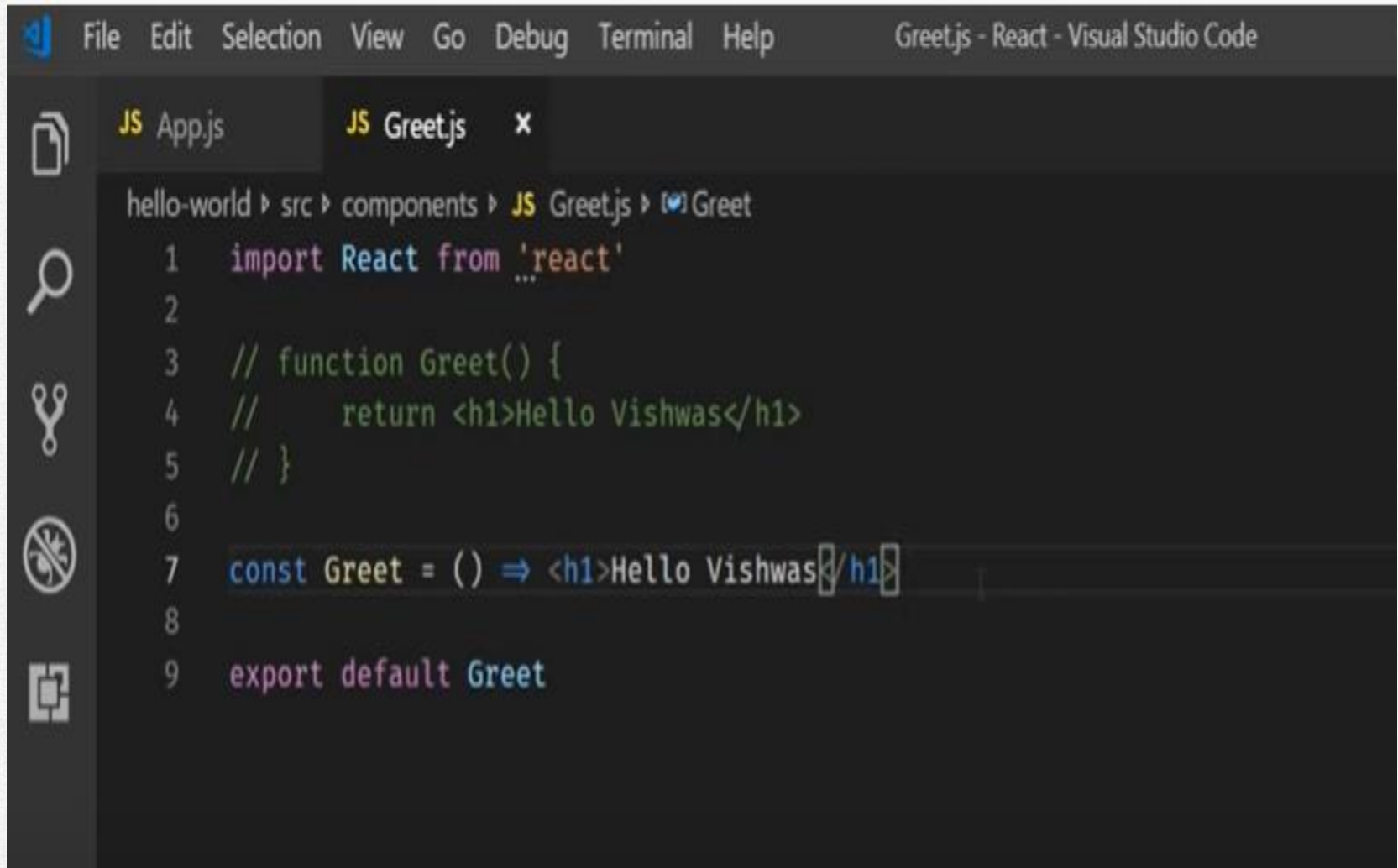
Call this file greet.js inside App.js file



The image shows a screenshot of the Visual Studio Code editor interface. The top menu bar includes File, Edit, Selection, View, Go, Debug, Terminal, and Help. The title bar indicates the active file is 'App.js - React - Visual Studio Code'. The editor has two tabs open: 'JS App.js' and 'JS Greet.js'. The 'JS App.js' tab is active, showing the following code:

```
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 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:-



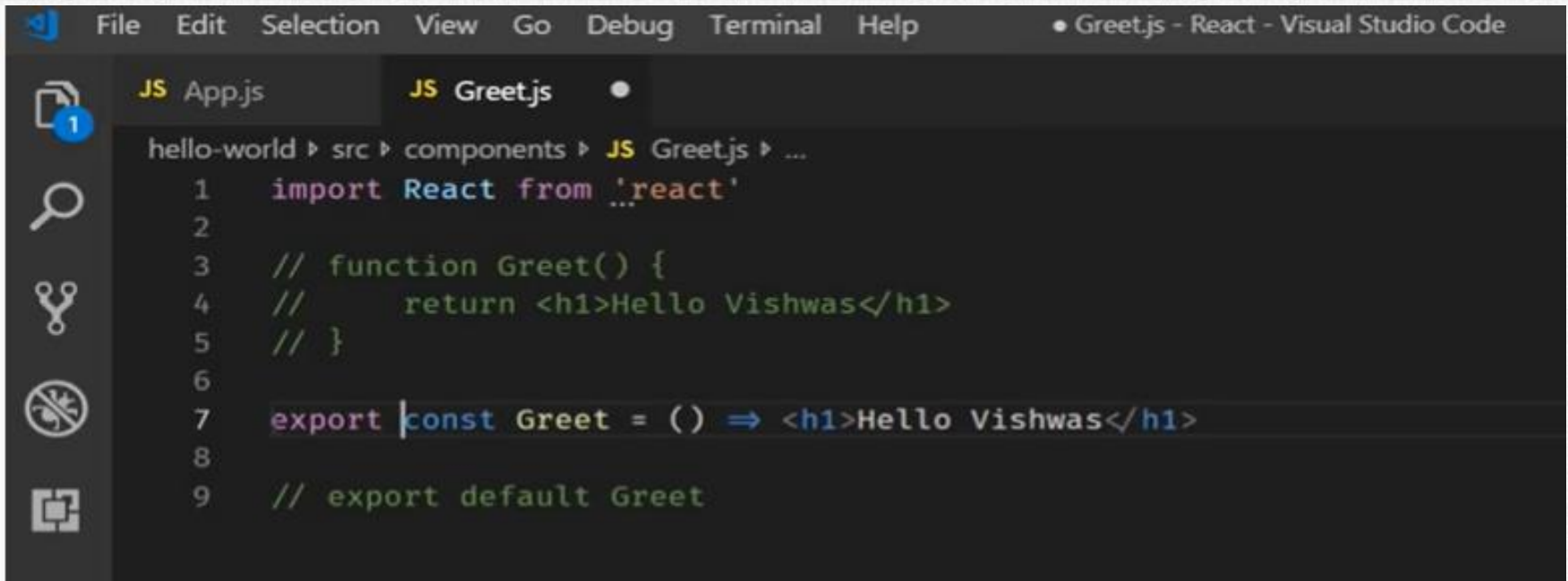
The screenshot shows the Visual Studio Code editor interface. The top menu bar includes File, Edit, Selection, View, Go, Debug, Terminal, and Help. The title bar indicates the file is 'Greet.js - React - Visual Studio Code'. The left sidebar shows the Explorer view with a file tree: 'hello-world' > 'src' > 'components' > 'JS Greet.js' > 'Greet'. The main editor area displays the following code:

```
1  import React from 'react'
2
3  // function Greet() {
4  //    return <h1>Hello Vishwas</h1>
5  // }
6
7  const Greet = () => <h1>Hello Vishwas</h1>
8
9  export default Greet
```


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.



```
File Edit Selection View Go Debug Terminal Help • Greet.js - React - Visual Studio Code

JS App.js JS Greet.js
hello-world ▸ src ▸ components ▸ JS Greet.js ▸ ...
1 import React from 'react'
2
3 // function Greet() {
4 //   return <h1>Hello Vishwas</h1>
5 // }
6
7 export const Greet = () => <h1>Hello Vishwas</h1>
8
9 // export default Greet
```

Thanks