

REACT

Partha Das Chowdhury



Fragments

can only return one node

```
function App() {  
  return (  
    <p></p>  
    <NavBar/>  
  );  
}
```

JSX expressions must have one parent element.












```
function App() {  
  return (  
    <React.Fragment>  
      <p></p>  
      <NavBar/>  
    </React.Fragment>  
  );  
}
```

or

```
function App() {  
  return (  
    <>  
      <p></p>  
      <NavBar/>  
    </>  
  );  
}
```

Class Based Components

```
class App extends React.Component {  
  render() {  
    return (  
      <p>I am a class based component</p>  
    )  
  }  
}
```

| | |
|---|----------------------------------|
|  cc | Class Component |
|  ccc | Class Component With Constructor |
|  cc1 | consoleClear |
|  cco | consoleCount |
|  createContext | react |
|  caches | |
|  cdc | componentDidCatch |
|  cpc | Class Pure Component |
|  cancelAnimationFrame | |
|  catch | |
|  CSSConditionRule | |

cc Class Component (Simple React Snippets) ×

```
class extends React.Component {  
  render() {  
    return <div></div>;  
  }  
}
```

Add Values

```
class App extends React.Component {  
  state = {name: "Software Tools"};  
  
  render() {  
    return (  
      <p>My Name is {this.state.name} </p>  
    )  
  }  
}  
  
export default App;
```

Passing State

```
class App extends React.Component {  
  state = {name: "Software Tools"};  
  render() {  
    return (  
      <Passingvalues name={this.state.name} />  
    )  
  }  
}
```

```
class Passingvalues extends React.Component {  
  render() {  
    if (this.props.name === "") {  
      return (  
        <p>Hello!</p>  
      )  
    } else {  
      return (  
        <p>Hello, {this.props.name}!</p>  
      )  
    }  
  }  
}
```

While we are on state

```
import React, { useState } from "react";
```

```
const [number, setNumber] = useState();
```

```
return (  
  <>  
  <h1>Choose a number {number}!</h1>  
  <button  
    type="button"  
    onClick={() => setNumber(6)}  
  >6</button>  
  <button  
    type="button"  
    onClick={() => setNumber(7)}  
  >7</button>  
  <button  
    type="button"  
    onClick={() => setNumber(8)}  
  >8</button>  
  <button
```



Choose a number !

6 7 8 9

Choose a number 9!

6 7 8 9

Useful VS Code Extensions




Prettier - Code formatter

✓ Prettier | 30,232,858 | ★★★★★

Code formatter using prettier

[Disable](#) [Uninstall](#) ⚙️

This extension is enabled globally.



Simple React Snippets

✓ Burke Holland | 2,608,437 | ★

Dead simple React snippets you will actually

[Disable](#) [Uninstall](#) ⚙️

This extension is enabled globally.

Happy REACT - ing