

M2W2D1

miguel.garrido@ironhack.com



Quick review

1. What are SPA and MPA?
2. What JS Libraries can we mention?
3. What is a Component?
4. What is a Prop?



States in react

States are objects that store a component's dynamic data

They determine how the component behaves.

When the state changes, React re-renders the component to reflect the updated data,

This allow us interactive and dynamic user interfaces.



useState

useState is a React Hook that lets you add a state variable to your component.

Hooks are special functions in React:

```
import { useState } from 'react';
```



useState

Normal variable (or non-state variable)

```
let index = 0;
```

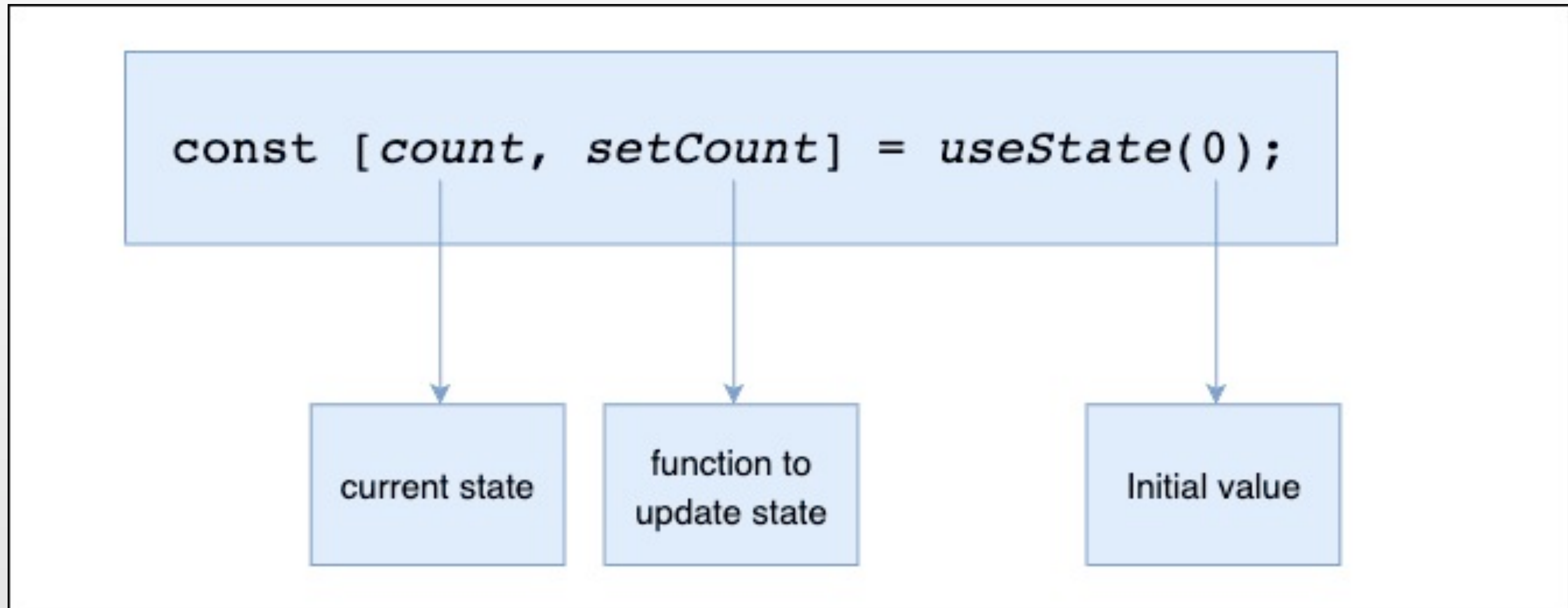
State Variable

```
const [index, setIndex] = useState([]);
```

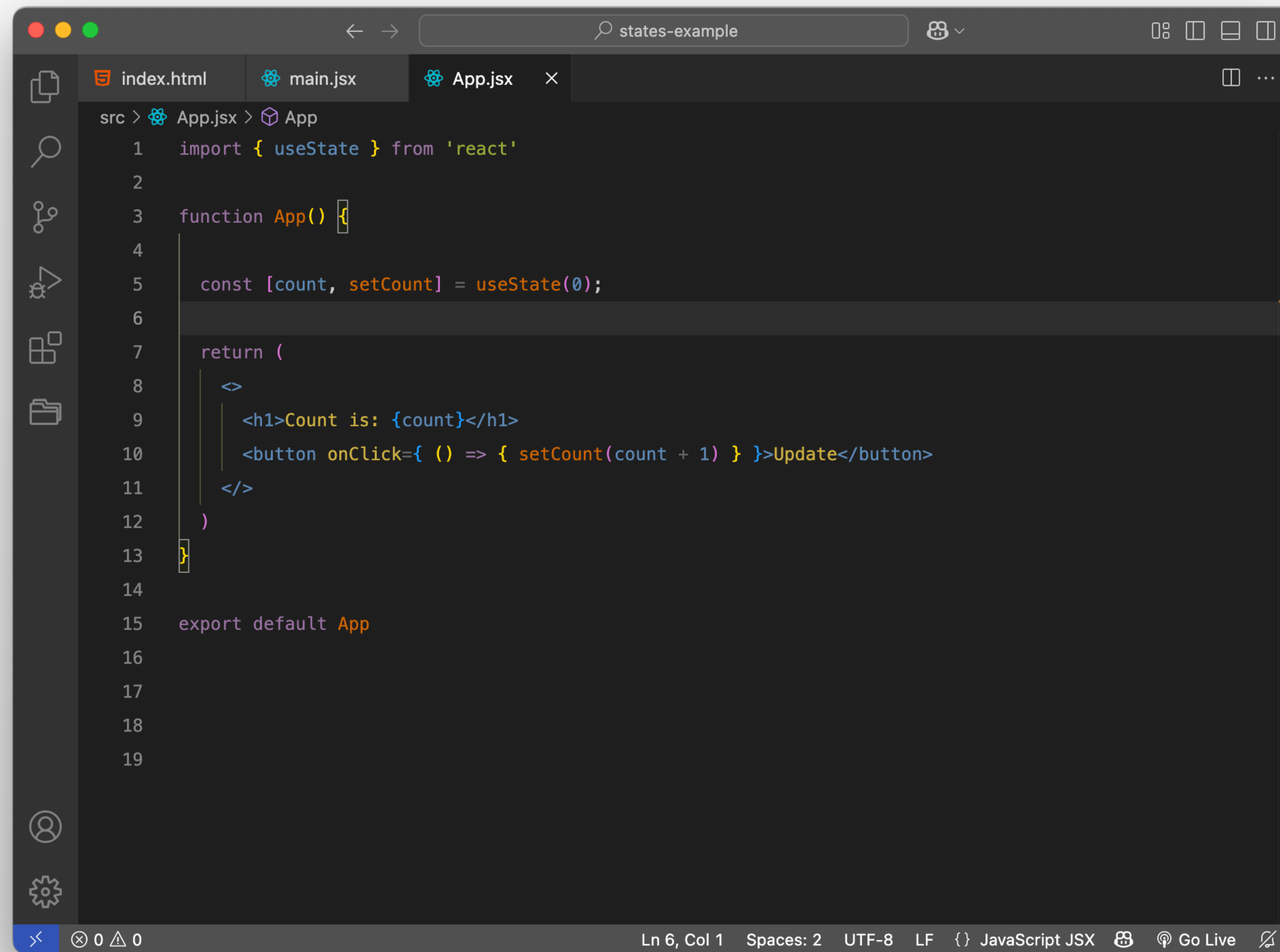
index is a state variable and **setIndex** is the setter function.



useState



useState example #1



The screenshot shows a code editor window with the title 'states-example'. The editor has three tabs: 'index.html', 'main.jsx', and 'App.jsx'. The 'App.jsx' tab is active, showing the following code:

```
1 import { useState } from 'react'
2
3 function App() {
4
5   const [count, setCount] = useState(0);
6
7   return (
8     <>
9       <h1>Count is: {count}</h1>
10      <button onClick={ () => { setCount(count + 1) } }>Update</button>
11    </>
12  )
13 }
14
15 export default App
```

The code is a React functional component named 'App'. It imports 'useState' from 'react'. Inside the component, it uses 'useState(0)' to create a state variable 'count' and a setter function 'setCount'. The component returns a JSX element containing an h1 tag with the text 'Count is: {count}' and a button with the text 'Update'. The button's onClick event handler calls 'setCount(count + 1)'. The component is exported as the default export.



useState example #2

```
1 import { useState } from 'react'
2
3 function App() {
4
5   const [count, setCount] = useState(0);
6   const updateCount = () => {
7     setCount(count + 1);
8   }
9
10  return (
11    <>
12      <h1>Count is: {count}</h1>
13      <button onClick={updateCount}>Update</button>
14    </>
15  )
16 }
17
18 export default App
19
20
21
22
```



useState example #3

```
1 import { useState } from 'react'
2
3 function App() {
4
5     const [userName, setUserName] = useState('');
6     const handleUserName = (event) => {
7         setUserName(event.target.value)
8     }
9
10    return (
11        <>
12            <h1>Username is: {userName}</h1>
13            <input type='text' id='userName' onChange={handleUserName} />
14        </>
15    )
16 }
17
18 export default App
19
20
21
22
```



Further reading

<https://react.dev/learn/state-a-components-memory>

<https://react.dev/reference/react/useState>

[https://developer.mozilla.org/en-US/docs/
Learn web development/Core/Frameworks libraries/
React interactivity events state](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Frameworks_libraries/React_interactivity_events_state)

