React component state

State: -

In react class components, state is built-in object which is use define the dynamic properties of component, the component state can be change over the time, the state properties automatically re-render in UI whenever their values get modified by setState(fun..) function.

setState is again predefine function in react, to update the state of component.

JS Demo.js U

src > demo > JS Demo.js > Demo

```
1
2   import { Component } from "react";
3
4   class Demo extends Component
5   {
6       constructor()
7       {
8           super();
9        this.state={
10          no:0
11       }
12       this.increaseCount=this.increaseCount.bind(this)
13      }
14
15       increaseCount()
16      {
17        this.setState(priviousState=>this.state.no=priviousState.no+1)
18      }
19      render()
20      {
21        return (
22          <div>
23          <h1>{this.state.no}</h1>
24          <button onClick={this.increaseCount}>Increase</button>
25          </div>
26         );
27  }
28  }
29  export default Demo;
```

Q) How to achieve state in React functional components?

useState is a react hoot, which adds the state in React functional components. It is use to re-render the dynamic variable in user interface.

It returns array with two with two values, on 0^{th} index current state and on 1^{st} index, the function to update the value of current state.

The useState hook also accepts initial value of state as a parameter argument.

useState informs to the ReactDOM that the state of variable is changed, so re-render it's updated value on user interface.

Steps to add state in functional component: -

1. **Import `useState()` form react library, which is named export.**

   Ex. import { useState } from "react";

2. **Call the useState() in component only.**
   Ex.

```
function Counter()
{
  useState(0);
}
export default Counter;
```

3. **Store the return value of useState() into array**

Ex.

```
function Counter()
{
 Let [state, setState]=  useState(0);
}
export default Counter;
```

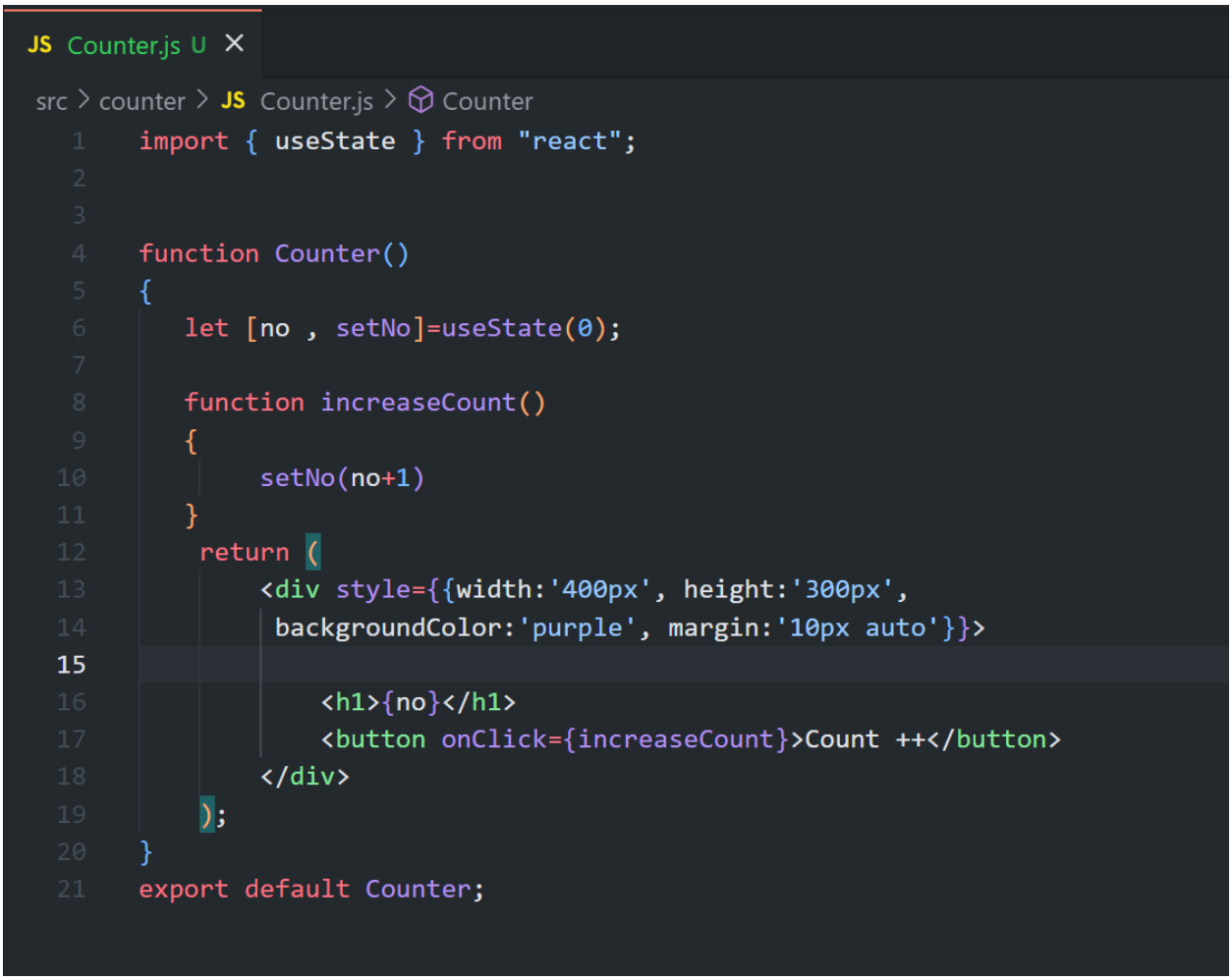
4. **Use `setState()` to update the state.**

Ex.

```
function Counter()
{
 Let [state, setState]=  useState(0);

function  updateState()
{
    setState("new state");
}
export default Counter;
```

Example of useState: -



by Kunal Sir

Some React Hooks: -

Hooks are simple pre-define functions in React.

In react, Any function which is starting with word `use` those functions are called as hooks.

1. useState()
2. useEffect()
3. useContext()
4. useParams()

Rules to use React hooks: -

1. Hooks should be only called inside the components.
2. Hooks should be called only top level of component.
3. Hooks can not be call in event handlers or JSX of component.