

# Use State

What is use state ?

→ Class 1 of the most fundamental hooks <sup>not</sup> it is a named export. it is state of components, there can be multiple states. it consists of a item, method, initial value <sup>if the</sup> item.

const [count, setCounter] = useState // initial value of counter

eg. program

const UseStateExample = () => {

const [log, counter]

const handleClick = () => { setCounter(counter + 1); }

}

return (

<Fragment>

<button onClick={handleClick}>counter </button>

<Fragment>

),

};

eg program-2

import React, { Fragment, useState } from "react",

const UseStateHookExample2 = () => {

// State item, method, initial value

~~const [counter, setCounter] = useState(0)~~

const handleOnChange = () => { name1: "", name2: "" }

const [infoData, setInfoData] = useState(initialState);

const handleOnChange = (event) => { const { name, value } = event.target; }

return (

<Fragment>

<input placeholder="Input one" name=

"name1" onChange={handleOnChange} />

<input placeholder="Input two name="name2"

onChange={handleOnChange} />

</Fragment>

)

};



# Use Effect

classmate

Date \_\_\_\_\_

Page \_\_\_\_\_

# Use effect hook will ~~over~~ ~~render~~ ~~only~~ ~~when~~ ~~important~~ ~~state~~ ~~is~~ ~~import~~ ~~use~~ ~~effect~~;

it takes in an effect and also a dependency array

# Use effect will run only once when the dependency array is empty

# We can pass a prop/variable in the

Use effect dependency array and also set conditions on that prop for use effect to render

eg program - 1

useState

```
import React, { useState, useEffect } from 'react';
```

```
const UseEffectHookExample = () => { const [click, clicked] =
```

```
  useState(0);
```

```
  useEffect(() => {
```

```
    console.log('Use Effect hook in action');
```

```
  }, [click]);
```

```
  return (
```

```
    <div>Clicked {click} times</div>
```

```
    <button onClick={click} >Click Me</button>
```

```
  )
```

```
</>
```

```
)
```

```
};
```

export default UseEffectHookExample;

// Output → console log will print again with every next click

// telling use effect to run on certain conditions  
with dependency array

import React, { useState, useEffect } from "react"

const UseEffectHookExample2 = () => {

const [count1, setCounter1] = useState(0);

const [counter2, setCounter2] = useState(0);

useEffect(() => {

console.log("Use Effect will only run - state of count 2 changes");

}, [counter2]);

return (

<>

<button onClick={() => setCounter1(counter1+1)}>

Button 1

</button>

<button onClick={() => setCounter2(counter2+1)}>

Button 2

</button>

<p> Button 1 clicked {count1} times </p>

<p> Button 2 clicked {count2} times </p>

</>

); // console log will only print when Button  
2 is clicked

export default UseEffectHookExample;