

useState in React

useState is a React Hook that allows you to add state management to functional components. State in React is used to store and manage data that can change over the lifecycle of a component

```
const [state, setState] = useState(initialValue);
```

state: The current state value.

setState: A function used to update the state.

initialValue: The initial value of the state.

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

function Counter() {
  // Declare state variable `count` with initial value 0
  const [count, setCount] = useState(0);

  return (
    <div>
      <h1>Count: {count}</h1>
      <button onClick={() => setCount(count + 1)}>Increment</button>
      <button onClick={() => setCount(count - 1)}>Decrement</button>
    </div>
  );
}
```

Explanation

1. **useState(0)** initializes the **count** state with **0**.
2. **setCount** is a function to update the **count** state.
3. When the button is clicked:
 - **setCount(count + 1)** updates the state to the incremented value.
 - React re-renders the component to reflect the updated state.

Key Points

1. **Initial Value**: **useState** takes an initial value, which can be any data type (number, string, object, array, etc.).

```
const [name, setName] = useState("John");  
const [items, setItems] = useState([]);
```

2. Updating State:

- You can update state with `setState`.
- React ensures the component re-renders after the state changes.

```
setName("Jane"); // Updates state to "Jane"  
setItems([...items, newItem]); // Adds a new item to the array
```

Managing an Object State

If the state is an object, you need to use the spread operator (`...`) to maintain immutability:

```
function Profile() {  
  const [user, setUser] = useState({ name: "John", age: 25 });  
  
  const updateName = () => {  
    setUser({ ...user, name: "Jane" }); // Update name but keep age  
  };  
  
  return (  
    <div>  
      <p>Name: {user.name}</p>  
      <p>Age: {user.age}</p>  
      <button onClick={updateName}>Change Name</button>  
    </div>  
  );  
}
```