

# State and Props in React

# What is State?

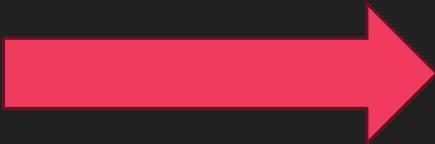
- A state is an object that stores the values of the properties belonging to the component that could change over a period of time
- A state can be modified based on the user action or network changes
- The state object can store multiple properties.
- The state object is initialized in the constructor

# Syntax for State

- `this.setState()` is used to change the value of the state object
- `setState()` function performs a shallow merge between the new and the previous state

```
class color extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = {  
      color: "red",  
      year: 1964  
    };  
  }  
  changeColor = () => {  
    this.setState({color: "blue"});  
  }  
  render() {  
    return ( <div>  
      <p> color is {this.state.color}</p>  
      <button type="button" onClick={this.changeColor} >Change  
color</button>  
    </div>  
  );  
}
```

Output



red

Change color



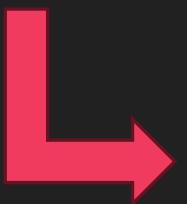
blue

Change color

# What is Props?

- Props stands for properties , allow user to pass argument or data to components.
- Props in component are read-only and cannot be changed

In PropsExample.jsx

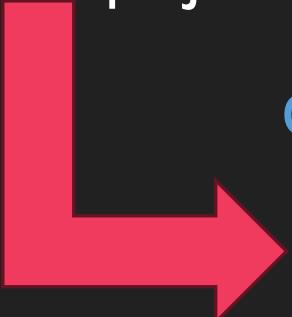


```
import React, { Component } from 'react'

class PropsExample extends Component {
  render() {
    return (
      <div>
        Hi! {this.props.name} And Your Age is
        {this.props.age }
      </div>
    )
  }
}

export default PropsExample
```

In App.js



```
class App extends Component {
  render() {
    return (
      <div className='App'>
        <PropsExample name="Vishwa" age="19"/>
      </div>
    )
  }
}
```

Output

Hi! Vishwa And Your Age is 19

# Difference between State and Props

## State

- State holds the data and can change over time
- State can only be used in class component
- State is generally Updated by event Handler

## Props

- Props are immutable-Once set,Props cannot be changed
- Props can be used in both functional and class component
- Props are set by the parent component for the children components