

State Manipulation

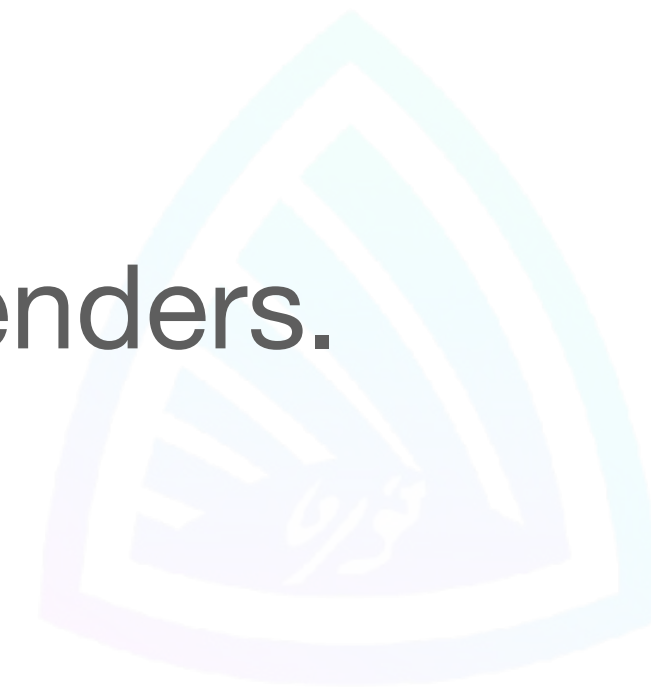
Objective

- What is state
- Creating the state Object
- Using the state Object
- React Hooks
- React useState Hook

What is State

- ❖ React components has a built-in state object.
- ❖ The state object is where you store property values that belongs to the component.
- ❖ When the state object changes, the component re-renders.

أكاديمية طويق
TUWAIQ ACADEMY



Creating the state Object

- ❖ The state object is initialized in the constructor:

```
class Car extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = {brand: "Ford"};  
  }  
  render() {  
    return ( <div> <h1>My Car</h1> </div> );  
  }  
}
```

Creating the state Object

- ❖ The state object can contain as many properties as you like:

```
class Car extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = {  
      brand: "Ford",  
      model: "Mustang",  
      color: "red",  
      year: 1964 };  
  }  
  render() {  
    return ( <div> <h1>My Car</h1> </div> );  
  }  
}
```

Using the state Object

Refer to the state object anywhere in the component by using the `this.state.propertyname` syntax:

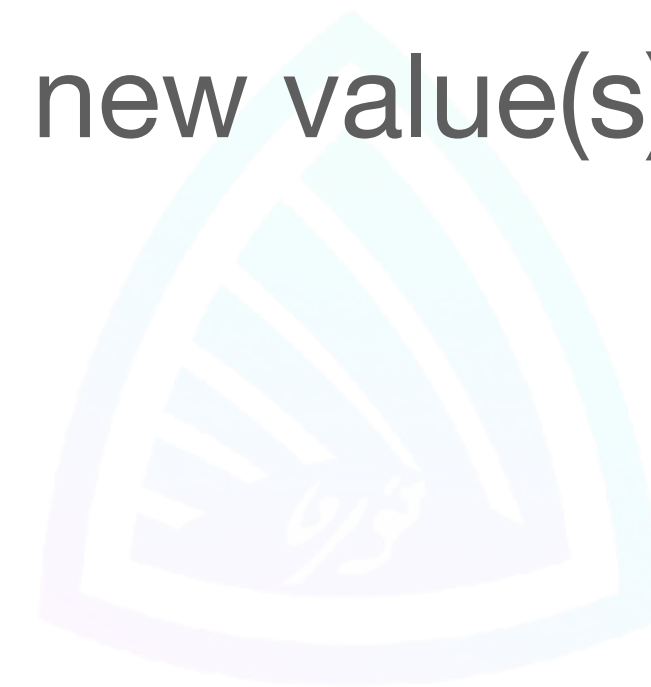
```
class Car extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      brand: "Ford",
      model: "Mustang",
      color: "red",
      year: 1964 };
  }

  render() {
    return (
      <div>
        <h1>My {this.state.brand}</h1>
        <p>It is a {this.state.color} {this.state.model} from {this.state.year}. </p>
      </div> );
  }
}
```

Changing the state Object

- ❖ To change a value in the state object, use the `this.setState()` method.
- ❖ When a value in the state object changes, the component will re-render, meaning that the output will change according to the new value(s).

أكاديمية طويق
TUWAIQ ACADEMY



Example for setState

```
class Car extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      brand: "Ford",
      model: "Mustang",
      color: "red",
      year: 1964 };
    this.changeColor = () => {
      this.setState({color: "blue"});
    }
    render() {
      return (
        <div>
          <h1>My {this.state.brand}</h1>
          <p> It is a {this.state.color} {this.state.model} from {this.state.year}. </p>
          <button type="button" onClick={this.changeColor} >Change color</button>
        </div> );
    }
  }
}
```


React Hooks

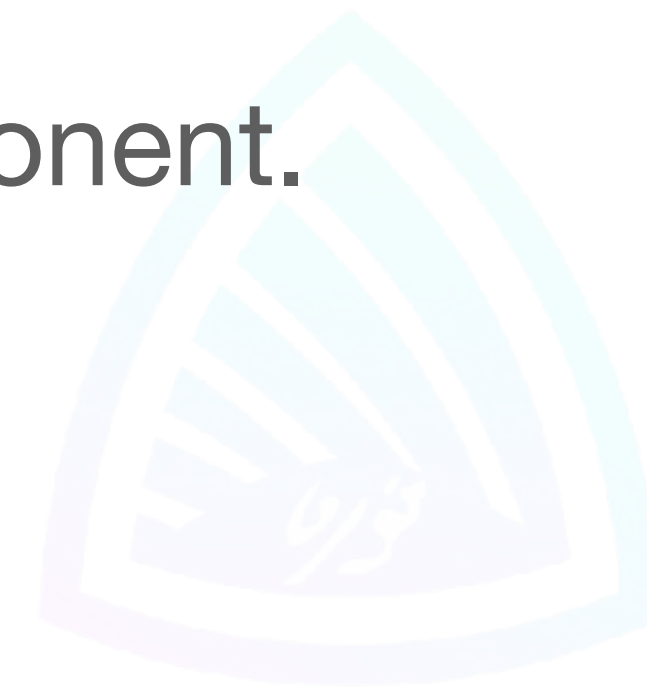
- ❖ Hooks allow function components to have access to state and other React features. Because of this, class components are generally no longer needed.
- ❖ Hooks allow us to "hook" into React features such as **state** and **lifecycle methods**.

Hooks Rules

❖ Hooks Rules:

1. Hooks can only be called inside React function components.
2. Hooks can only be called at the top level of a component.
3. Hooks cannot be conditional

أكاديمية طه
TUWAIQ ACADEMY



React useState Hooks

- ❖ The React `useState` Hook allows us to track state in a function component.
- ❖ State generally refers to data or properties that need to be tracking in an application.

أكاديمية طويق
TUWAIQ ACADEMY



Import useState

- ❖ To use the useState Hook, we first need to import it into our component.

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

TUWAIQ ACADEMY



Initilize useState

- ❖ We initialize our state by calling useState in our function component.
- ❖ useState accepts an initial state and returns two values:
 1. The current state
 2. A function that update the state

```
import { useState } from "react";  
function FavoriteColor() {  
  const [color, setColor] = useState("");  
}
```

Read state

- ❖ We can now include our state anywhere in our component.

```
import { useState } from "react";
import ReactDOM from "react-dom";

function FavoriteColor() {
  const [color, setColor] = useState("red");
  return <h1>My favorite color is {color}!</h1>
}

ReactDOM.render(<FavoriteColor />,
  document.getElementById('root'));
```

Update state

- ❖ To update our state, we use our state updater function.

```
import { useState } from "react";
import ReactDOM from "react-dom";

function FavoriteColor() {
  const [color, setColor] = useState("red"); return ( <
    <h1>My favorite color is {color}!</h1>
    <button type="button" onClick={() => setColor("blue")} >Blue</button>
    </> ) }

ReactDOM.render(<FavoriteColor />, document.getElementById('root'));
```



Resources

- https://www.w3schools.com/react/react_usestate.asp
- https://www.w3schools.com/react/react_state.asp

Summary

- What is state
- Creating the state Object
- Using the state Object
- React Hooks
- React useState Hook