REACT TRAINING DAY 10

K Shankar Sravanth

221910304024

CSE IV year

**Mounting:**

componentDidMount() executes the code within render function first and executes the code in function next.

1. Executes constructor
2. Executes render()
3. Executes componentDidMount()

**Updating:**

1. componentDidUpdate()
2. shouldComponentUpdate(prestate, props, snapshot)

Code:

class App extends Component{

    constructor(){

        super();

        console.warn("constructor")

    }

    componentDidMount(){

        console.warn("DidMount")

    }

    render()

    {

        console.warn("render")

        return(

            <div>

                <h1>Example of componentDidMount</h1>

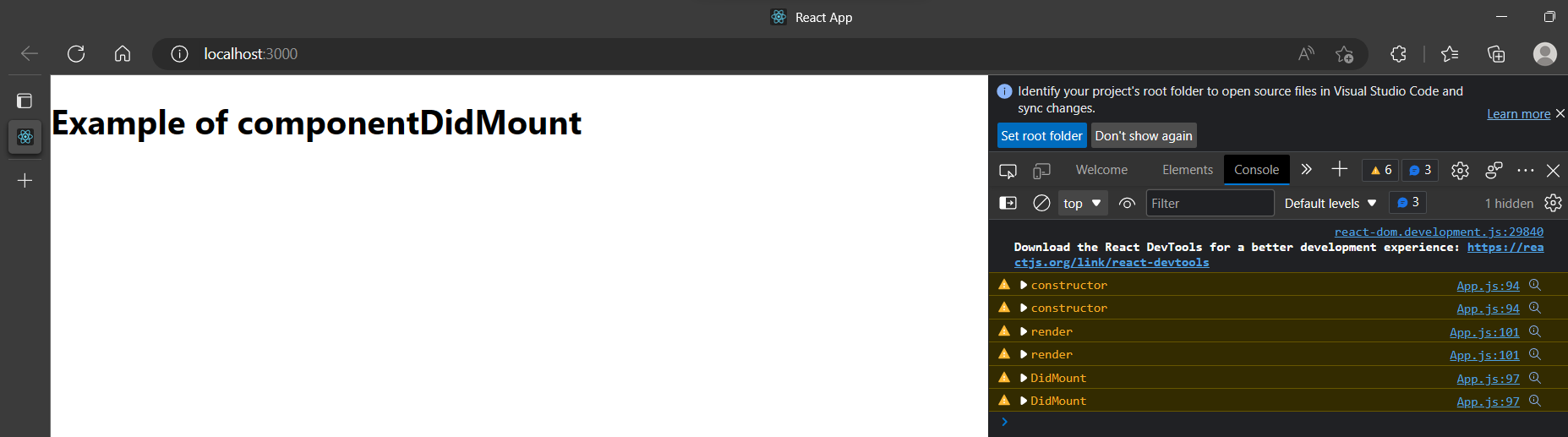
            </div>

        )

    }

}

export default App;



**componentDidUpdate:**

class App extends React.Component{

    constructor()

    {

    super();

    console.warn("Constructor");

    this.state={

    count:0

    }

    }

    componentDidUpdate(preProps,preState,snapshot)

    {

    console.warn("componentDidUpdate",preState.count,this.state.count)

      alert("data is already same")

}

    render()

    {

    return(

    <div>

    <h1>componentDidUpdate{this.state.count}</h1>

    <button onClick={()=>{this.setState({count:1})}}>Update Value</button>

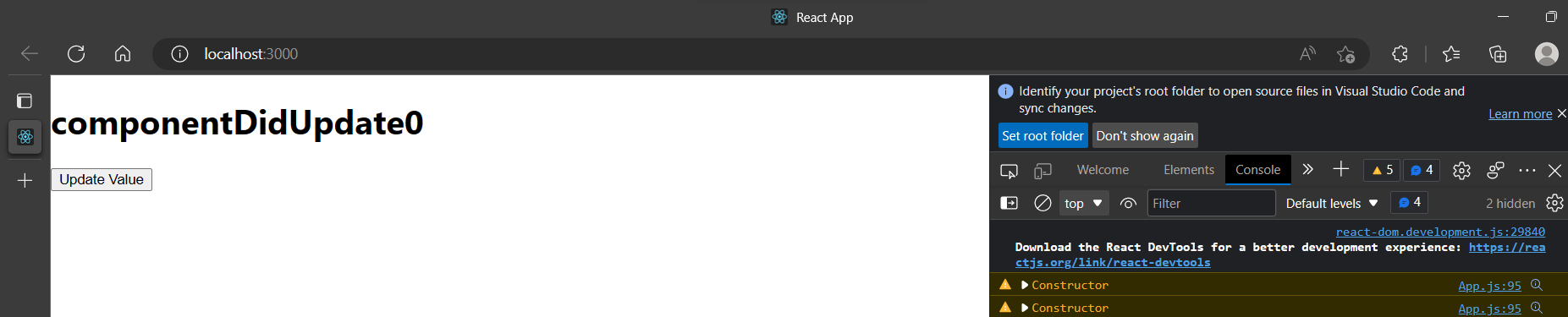
    </div>

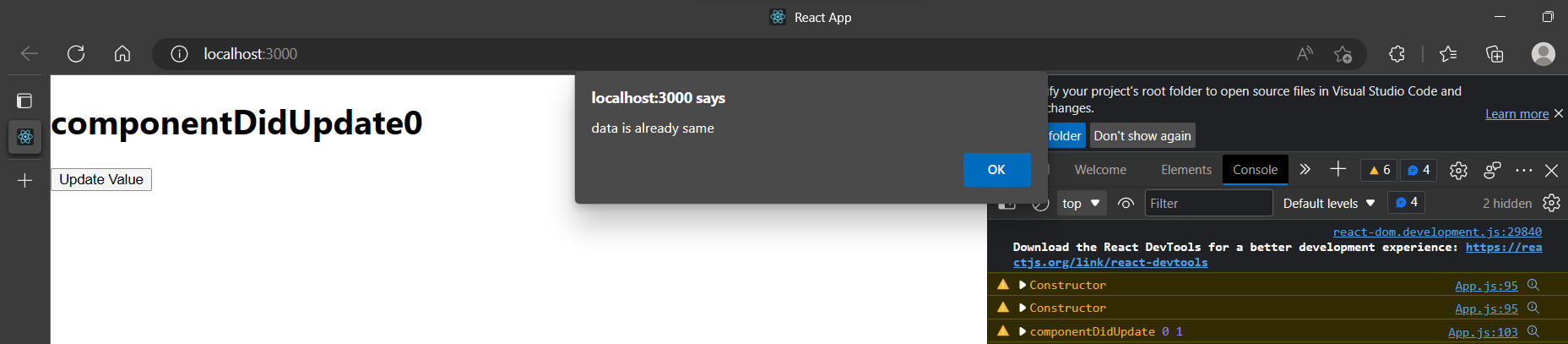
    );

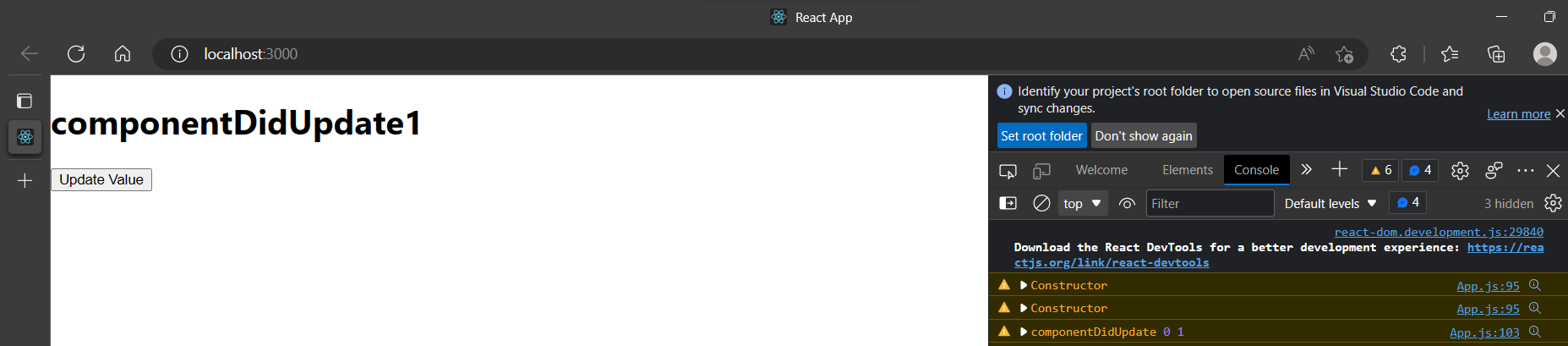
    }

    }

export default App;







**shouldComponentUpdate:**

code:

class App extends React.Component

{

constructor()

{

super();

this.state={

count:0 }

}

shouldComponentUpdate()

{

console.warn("shouldComponentUpdate",this.state.count);

return true;

}

render()

{

return(

<div className="App">

<h1>shouldComponentUpdate{this.state.count}</h1>

<button onClick={()=>{this.setState({count:this.state.count+1})}}>Update Value</button>

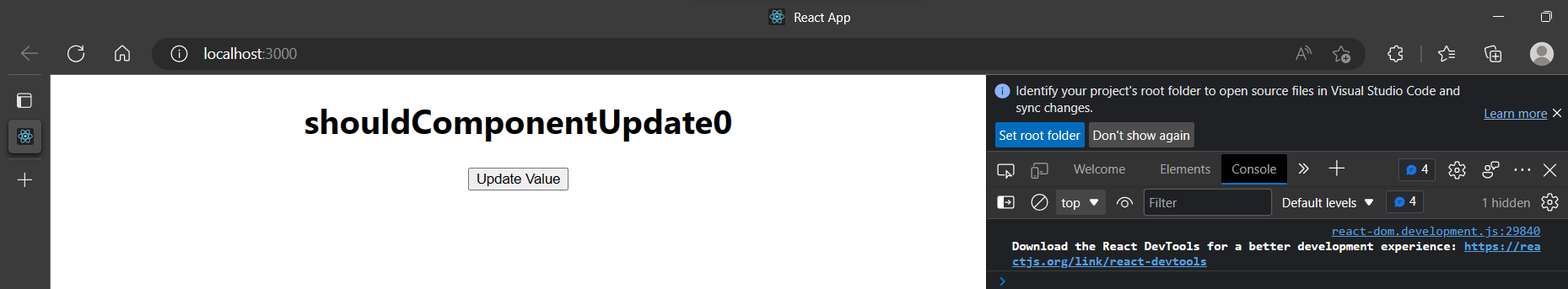
</div>

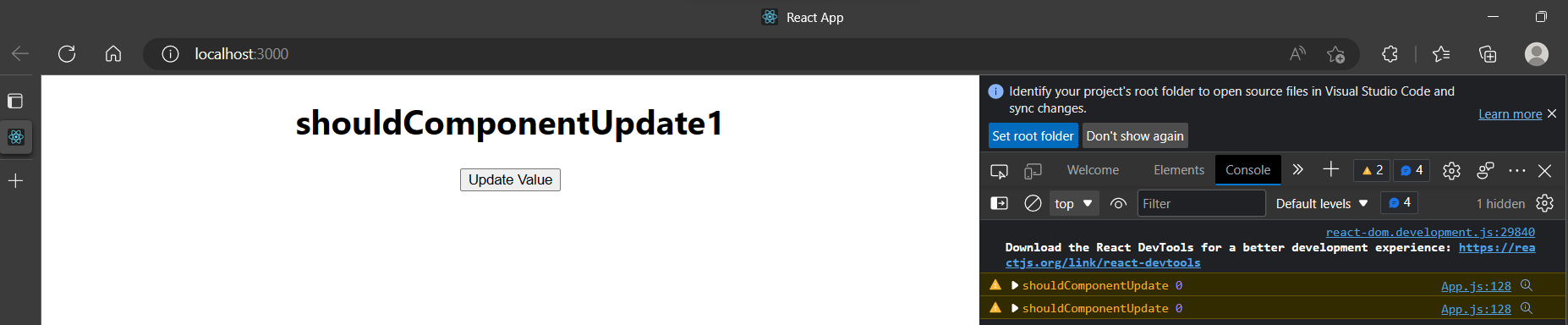
);

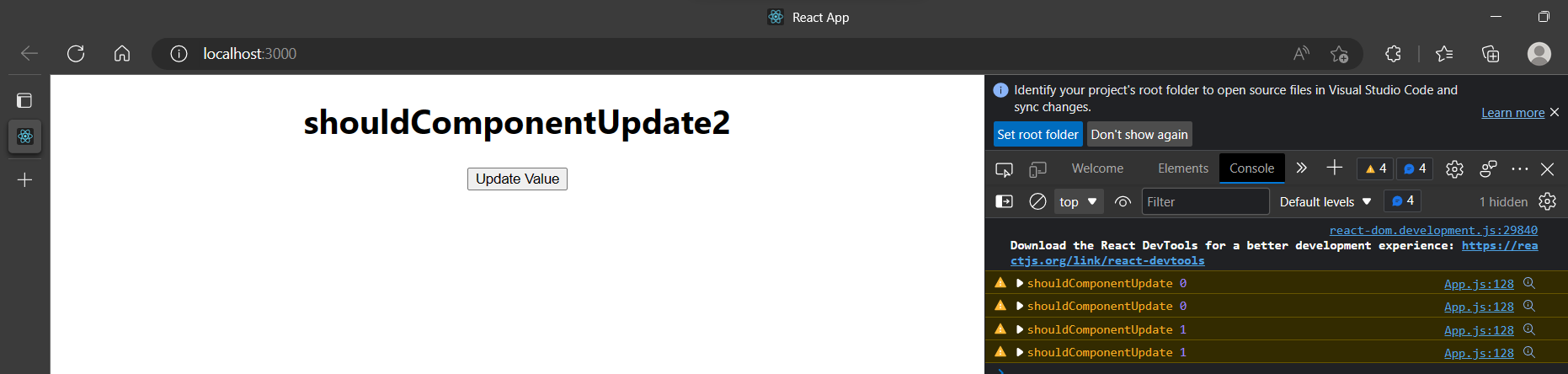
}

}

export default App;







**Create a class component, create a state called increase and assign initial value as 5 and create a button with your name. so whenever you click the button, the value of increase should jump to 10 units. Consider the method shouldcomponentUpdate(), in that, it has to display “hello” and the present value of increase.**

class App extends Component{

    constructor()

    {

        super();

        this.state={

            increase:5

        }

    }

    shouldComponentUpdate()

    {

            console.warn("hello",this.state.increase);

            return true;

        }

    render()

        {

            return(

                <div>

                    <h1>shouldComponentUpdate{this.state.increase}</h1>

                    <button onClick={()=>this.setState({increase:this.state.increase+5})}>Sravanth</button>

                </div>

            )

        }

    }

export default App;

