**ReactJS-HOL 11**

**Questions & Answers –**

1. **Explain React events**

**Ans -** React events are actions or occurrences that happen in the DOM (like a user clicking a button or typing in a textbox) to which we can respond using JavaScript. React wraps these events into a system known as Synthetic Events, allowing us to write consistent event-handling code across all browsers.

**Example:** <button onClick={handleClick}>Click Me</button>

Here, **onClick** is the event.

1. **Explain about event handlers**

**Ans -** Event handlers are functions that are triggered in response to specific events. In React, these functions are typically defined inside components and passed to JSX elements as props to handle user interactions.

**Example:**

function handleClick() {

alert("Button clicked!");

}

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

Here, **handleClick** is the event handler.

1. **Define Synthetic event**

**Ans -** A Synthetic Event is a cross-browser wrapper around the native browser event in React. It combines the behavior of different browsers into a single, consistent API. Synthetic events work identically across all browsers, making React event handling predictable.

It wraps the native event to provide:

* Consistent properties
* Performance optimization via event pooling

**Example:**

function handleInput(event) {

console.log(event.target.value); // SyntheticEvent

}

1. **Identify React event naming convention**

**Ans -** In React, event names follow camelCase naming convention. Unlike traditional HTML where events are written in lowercase (e.g., onclick), React uses camelCase like onClick, onChange, onSubmit, etc.

Additionally, React expects the event handler to be passed as a function reference, not as a string.

**Example –**

<button onClick={handleClick}>Click Me</button>

<input onChange={handleInput} />

So, the naming convention in React:

* Uses **camelCase** for event names
* Passes a **function reference**, not a string

**Hands-On – (Code)**

*Operations.js –*

import { useState } from 'react';

const Counter = () => {

    const[count, setCount] = useState(0);

    const[amount, setAmount] = useState('');

    const[converted, setConveretd] = useState('');

    const increment = () => {

        setCount(count + 1);

        alert(`Hello: Number${count}`);

    };

    const decrement = () => {

        setCount(count - 1);

    }

    const message = (msg) => {

        alert(msg);

    }

    const onPress = (event) => {

        alert("I was clicked");

        console.log("Synthetic Event: ", event)

    }

    const handleChange = (event) => {

        setAmount(event.target.value);

    }

    const handleSubmit = (event) => {

        event.preventDefault();

        const conversionRate = 80;

        const euro = (parseInt(amount) \* conversionRate);

        setConveretd(euro);

        alert(`Converting to euro amount is ${euro}`);

    }

    return(

        <div>

            <p>{count}</p><br></br>

            <button onClick={increment}>Increment</button>

            <br></br>

            <button onClick={decrement}>Decrement</button>

            <br></br>

            <button onClick={() => message("Welcome")}>Say Welcome</button>

            <br></br>

            <button onClick={onPress}>Click on me</button>

            <br></br>

            <h1 style={{"color": "Green"}}>Currency Converter!!</h1>

            <br></br>

            <form onSubmit={handleSubmit}>

                <label>Amount: </label>

                <input type='number' value={amount} onChange={handleChange} />

                <br></br>

                <label>Currency: </label>

                <input type='textArea' value={converted} placeholder='Euros' readOnly/>

                <br></br>

                <button type="submit">Submit</button>

            </form>

        </div>

    )

};

export default Counter;

*App.js –*

import './App.css';

import Counter from './Operations';

function App() {

  return (

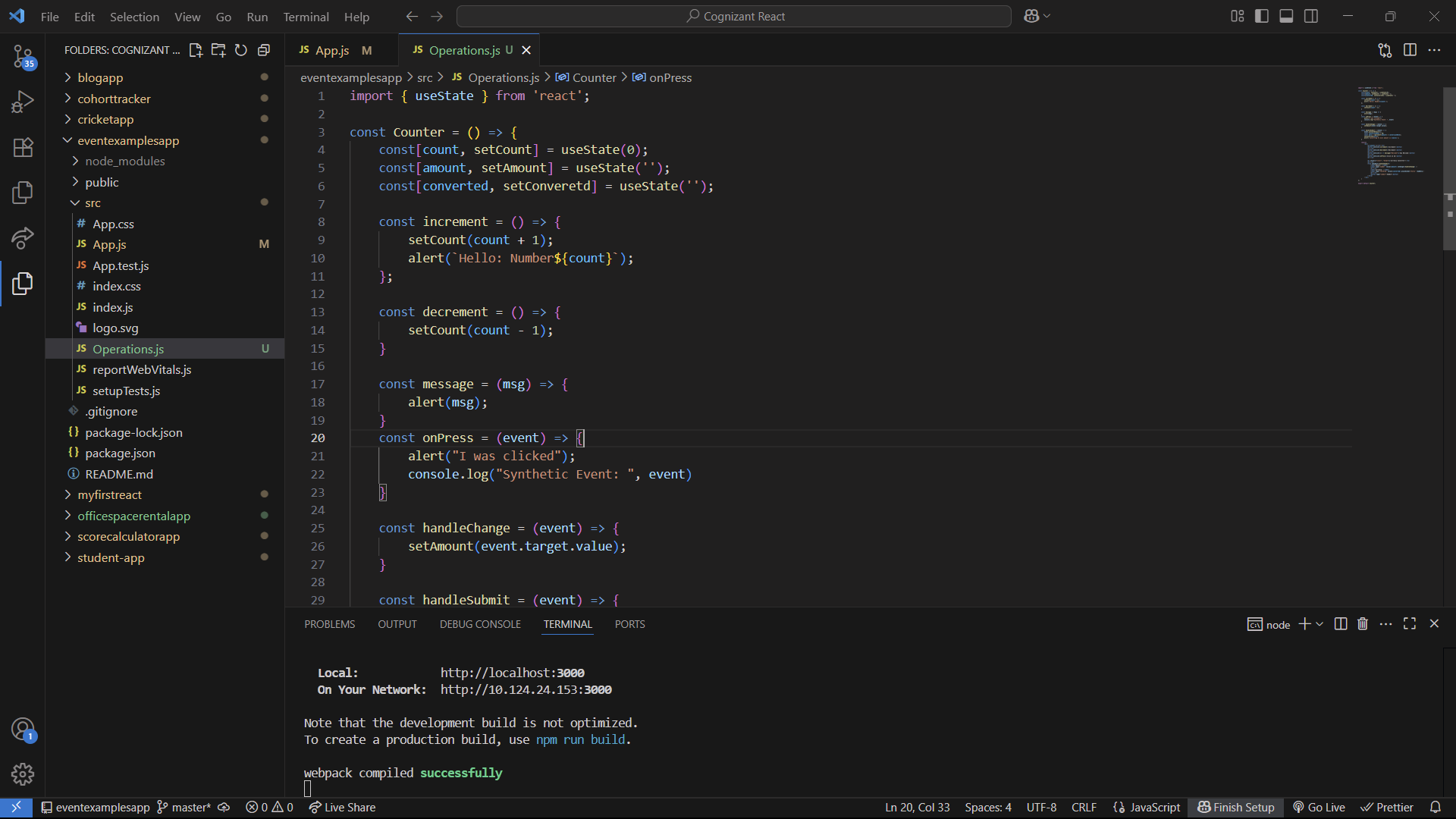
    <Counter />

  );

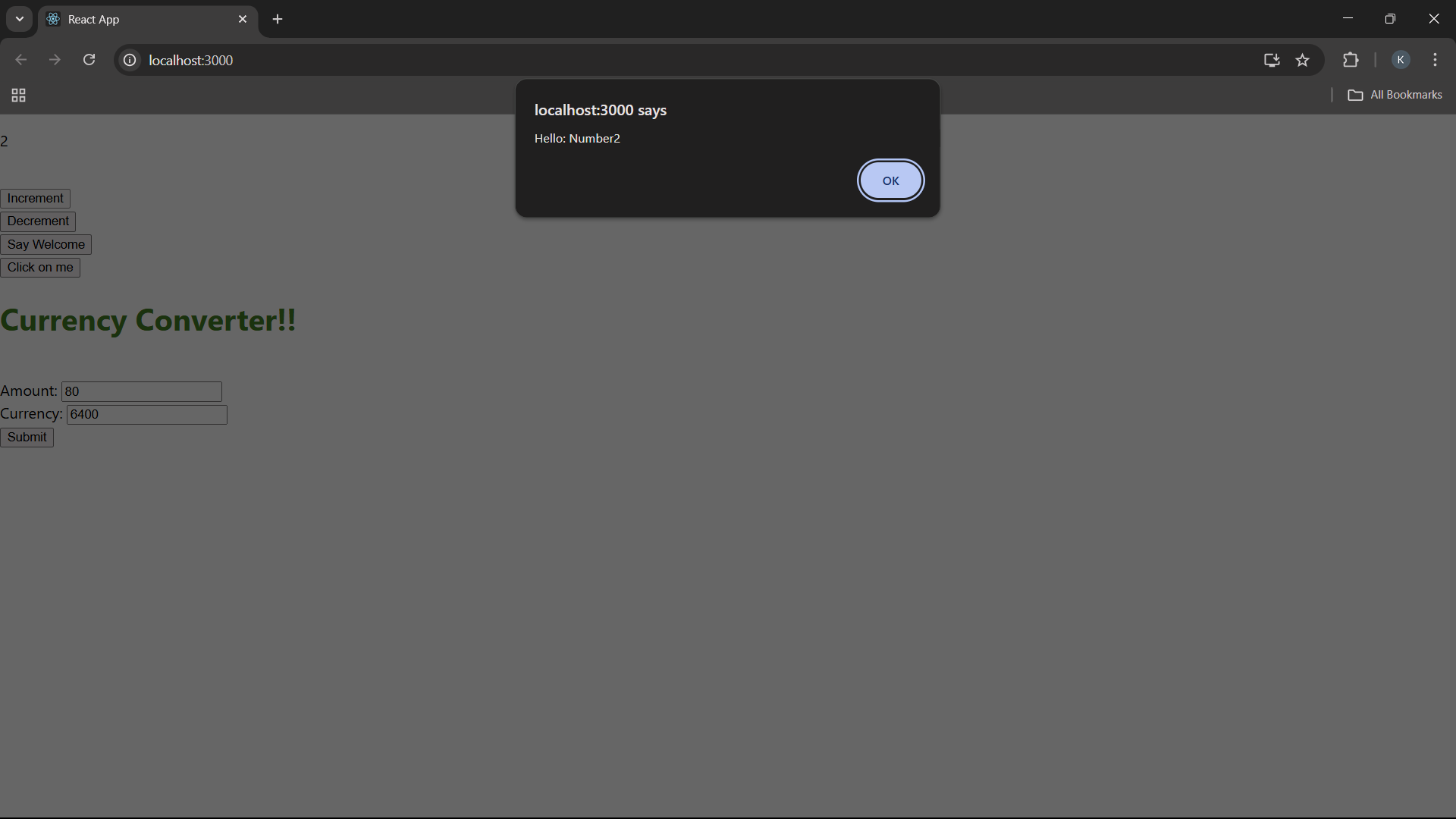
}

export default App;

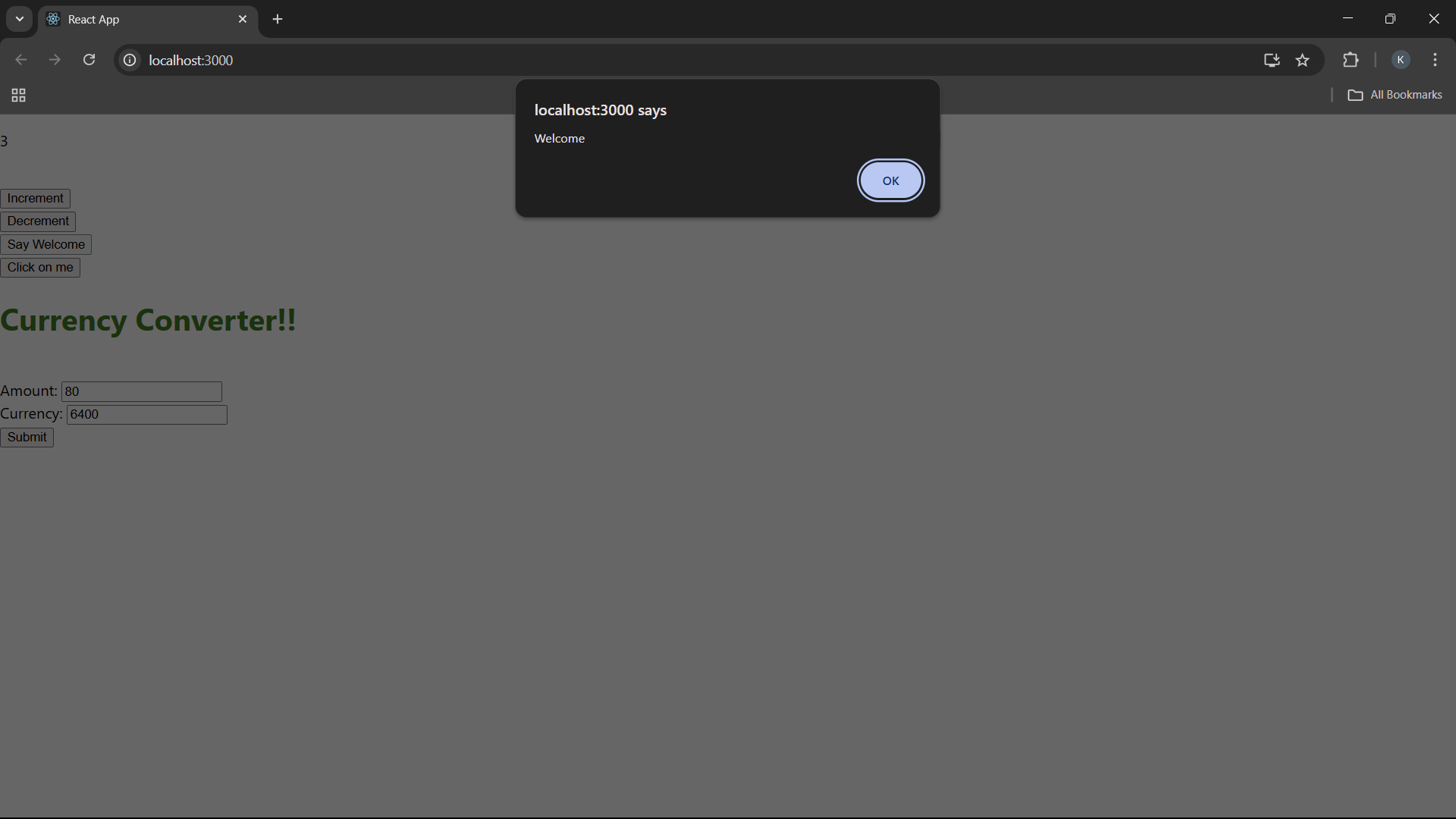
**Output –**



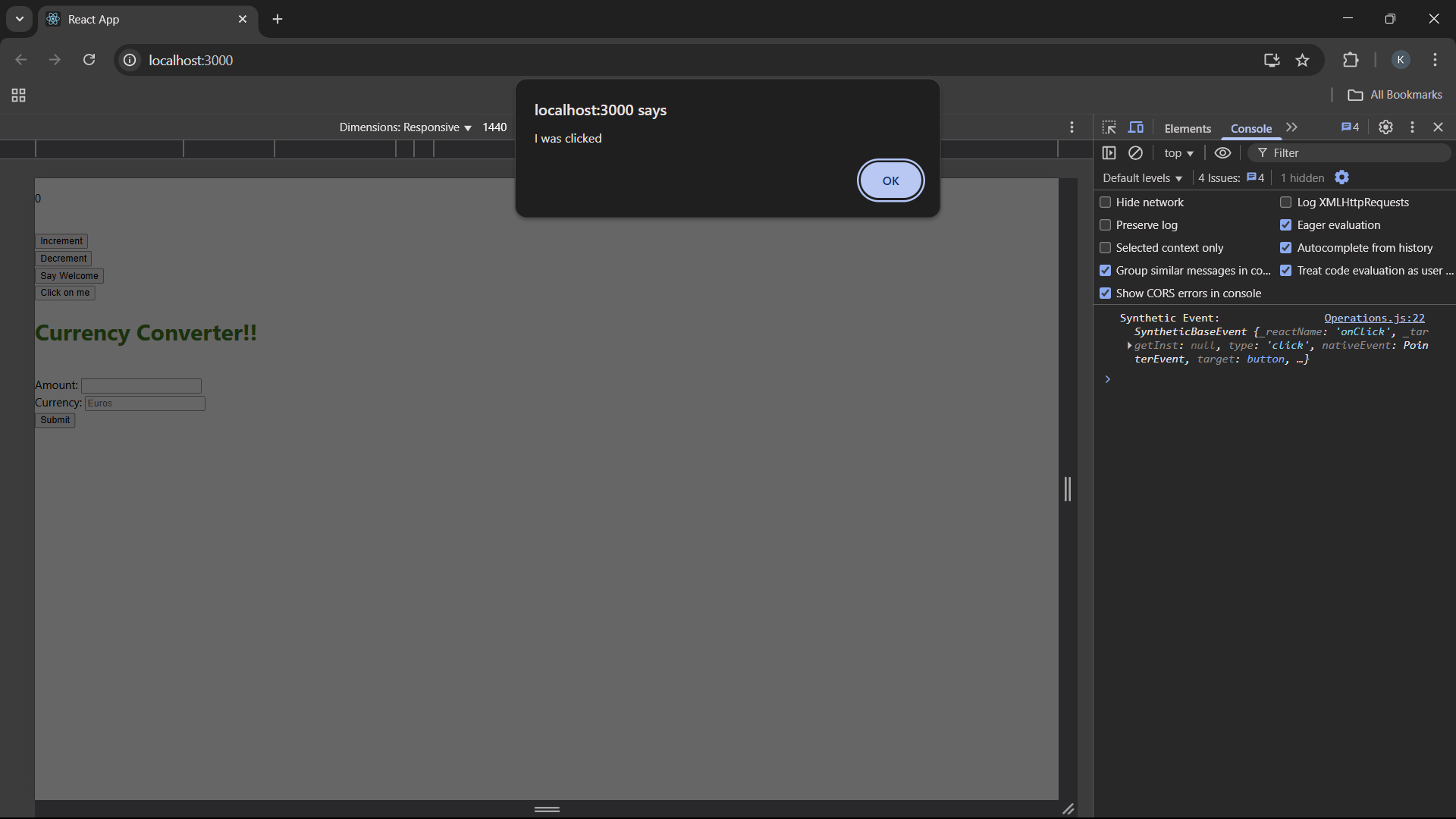
When Increment button is clicked



When Say welcome button is clicked



When click on me button is clicked



When submit button is clicked after entering rupee value in the text box

