**WEEK – 07**

**ReactJS-HOL**

**Superset ID: 6262264**

**EXERCISE 11:**

**Objectives:**

**1. Explain React Events**

In React, events are actions that can be handled by JavaScript, such as mouse clicks, key presses, form submissions, or any user interaction. React wraps these browser events into a cross-browser wrapper known as SyntheticEvent to provide a consistent behavior across all browsers.

React events are very similar to DOM events, but with a few syntax differences and improvements for performance and compatibility.

**2. Explain About Event Handlers**

An event handler is a function that is triggered in response to a specific event. In React, we use event handlers to handle user interactions.

**Example:**

function ClickHandler() {

function handleClick() {

alert('Button clicked!');

}

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

}

In this example, handleClick is the event handler, and it's called when the button is clicked.

**3. Define Synthetic Event**

React uses a system called SyntheticEvent, which is a wrapper around the browser’s native event system. It normalizes events so that they have consistent properties across different browsers.

**Benefits of SyntheticEvent:**

* Cross-browser compatibility
* Pooling system for performance optimization
* Same interface as native events

**Example:**

function FormComponent() {

function handleSubmit(event) {

event.preventDefault(); // Using SyntheticEvent

console.log('Form submitted!');

}

return <form onSubmit={handleSubmit}>...</form>;

}

**4. Identify React Event Naming Convention**

React follows camelCase naming for events instead of lowercase (used in HTML). Additionally, React expects event handlers to be passed as functions, not strings.

| **HTML** | **React** |
| --- | --- |
| onclick="..." | onClick={...} |
| onsubmit="..." | onSubmit={...} |
| onchange="..." | onChange={...} |

**Correct Example:**

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

**Incorrect (HTML Style):**

<button onclick="handleClick()">Click</button>

**Create a React Application “eventexamplesapp” to handle various events of the form elements in HTML.**

***App.js:***

import React from 'react';

import Counter from './Components/Counter';

import WelcomeButton from './Components/WelcomeButton';

import SyntheticClick from './Components/SyntheticClick';

import CurrencyConvertor from './Components/CurrencyConvertor';

function App() {

  return (

    <div className="App">

      <Counter />

      <WelcomeButton />

      <SyntheticClick />

      <CurrencyConvertor />

    </div>

  );

}

export default App;

**Counter.js:**

import React, { useState } from 'react';

function Counter() {

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

  const increment = () => {

    setCount(count + 1);

    sayHello();

  };

  const decrement = () => {

    setCount(count - 1);

  };

  const sayHello = () => {

    console.log("Hello! Counter Updated.");

  };

  return (

    <div>

  <h2>Counter Value: {count}</h2>

  <div>

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

  </div>

  <div>

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

  </div>

</div>

  );

}

export default Counter;

***WelcomeButton.js:***

import React, { useState } from 'react';

function WelcomeButton() {

  const [firstClick, setFirstClick] = useState(true);

  const greet = () => {

    if (firstClick) {

      alert('Hello! Member!');

      setFirstClick(false);

    } else {

      alert('Welcome!');

    }

  };

  return (

    <div>

      <button onClick={greet}>Say Welcome</button>

    </div>

  );

}

export default WelcomeButton;

***SyntheticClick.js:***

import React from 'react';

function SyntheticClick() {

  const handleClick = (event) => {

    event.preventDefault();

    alert("I was clicked");

  };

  return (

    <div>

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

    </div>

  );

}

export default SyntheticClick;

***CurrencyConvertor.js:***

import React, { useState } from 'react';

function CurrencyConvertor() {

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

  const [currency, setCurrency] = useState('');

  const handleSubmit = (e) => {

    e.preventDefault();

    let convertedAmount = 0;

    let message = '';

    if (currency.toLowerCase() === 'euro') {

      convertedAmount = amount \* 90;

      message = `Converting to Euro Amount is ${convertedAmount}`;

    } else if (currency.toLowerCase() === 'inr') {

      convertedAmount = (amount / 90).toFixed(2);

      message = `Converting to INR Amount is €${convertedAmount}`;

    } else {

      message = 'Invalid currency. Please enter "euro" or "inr"';

    }

    alert(message);

  };

  return (

    <div>

      <h2 style={{ color: 'green', fontWeight: 'bold' }}>Currency Convertor!!!</h2>

      <form onSubmit={handleSubmit}>

        <label>Amount:</label><br />

        <input

          type="number"

          value={amount}

          onChange={(e) => setAmount(e.target.value)}

        /><br /><br />

        <label>Currency:</label><br />

        <input

          type="text"

          value={currency}

          onChange={(e) => setCurrency(e.target.value)}

        /><br /><br />

        <div style={{ textAlign: 'center' }}>

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

        </div>

      </form>

    </div>

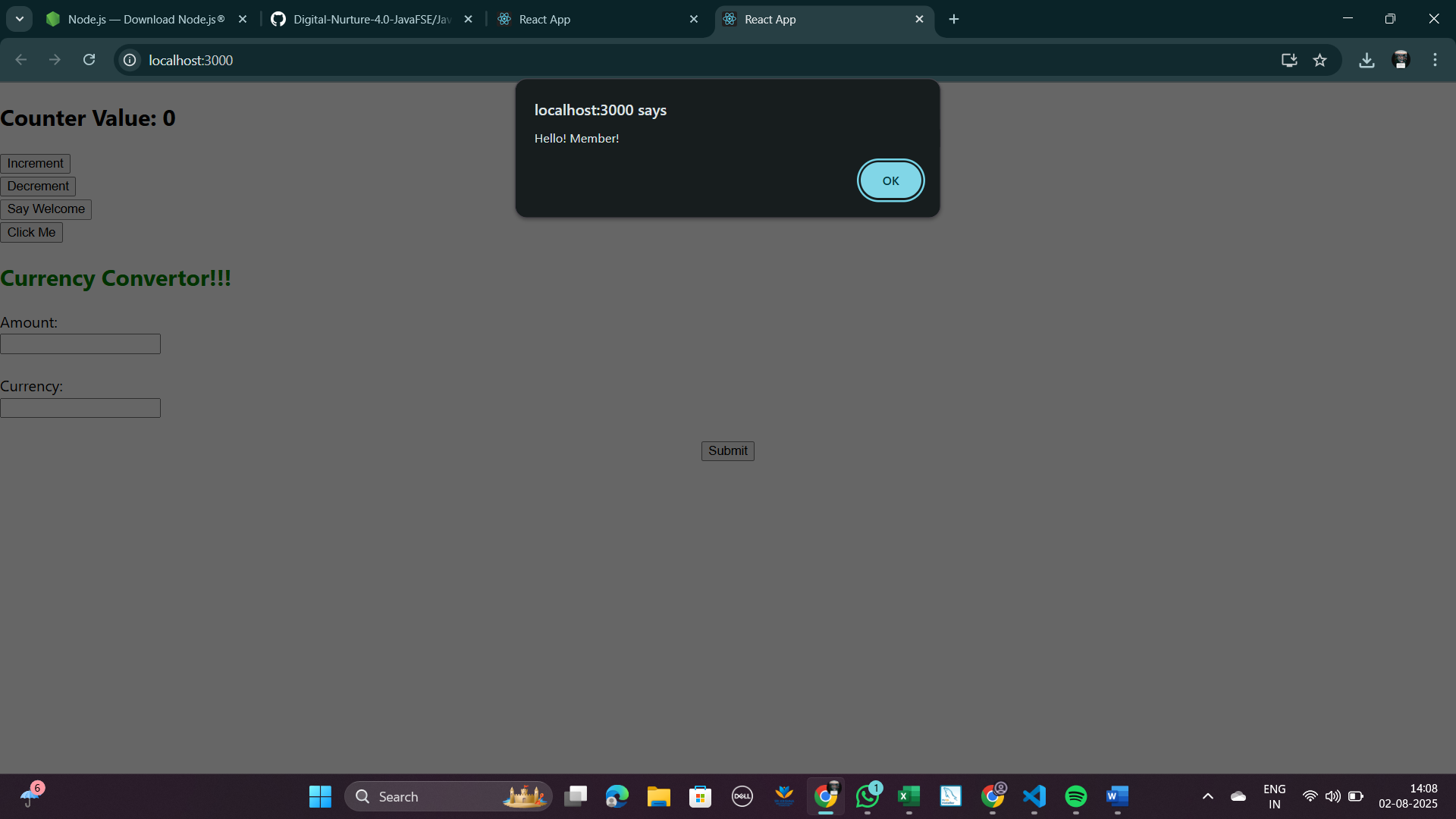
  );

}

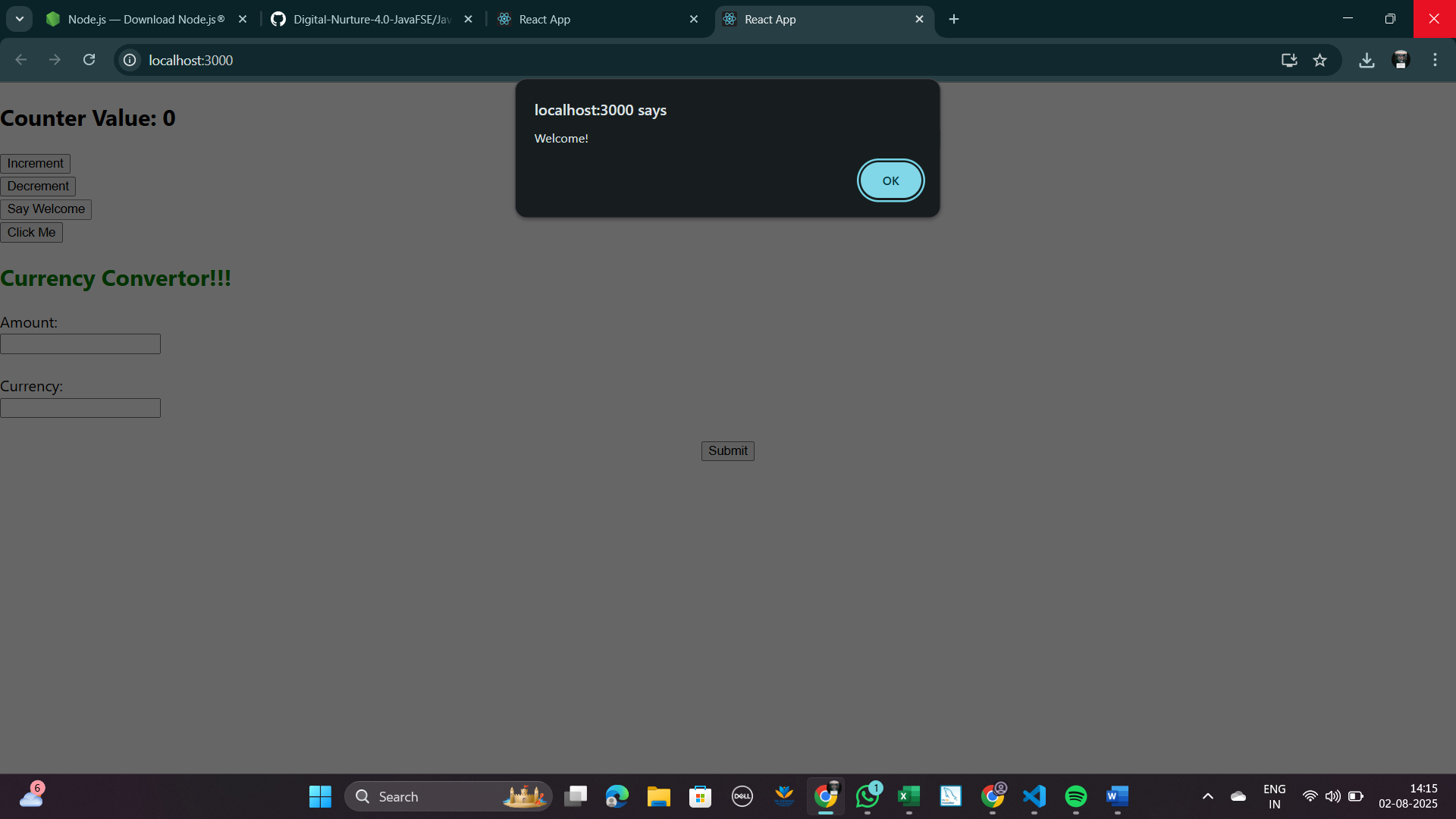
export default CurrencyConvertor;

**OUTPUT:**

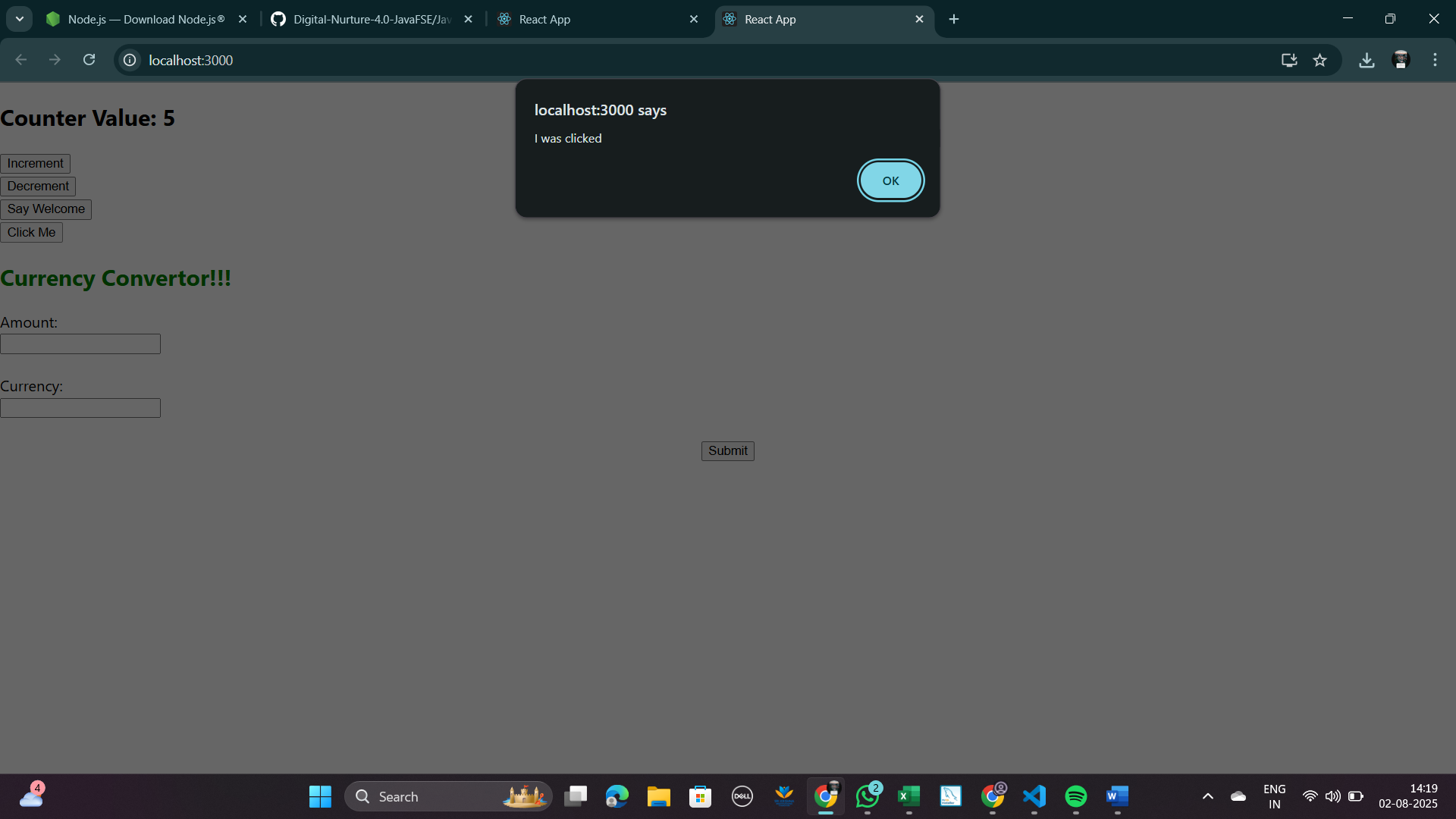
1. **Counter Component Output – Increment and Decrement with Hello Message**



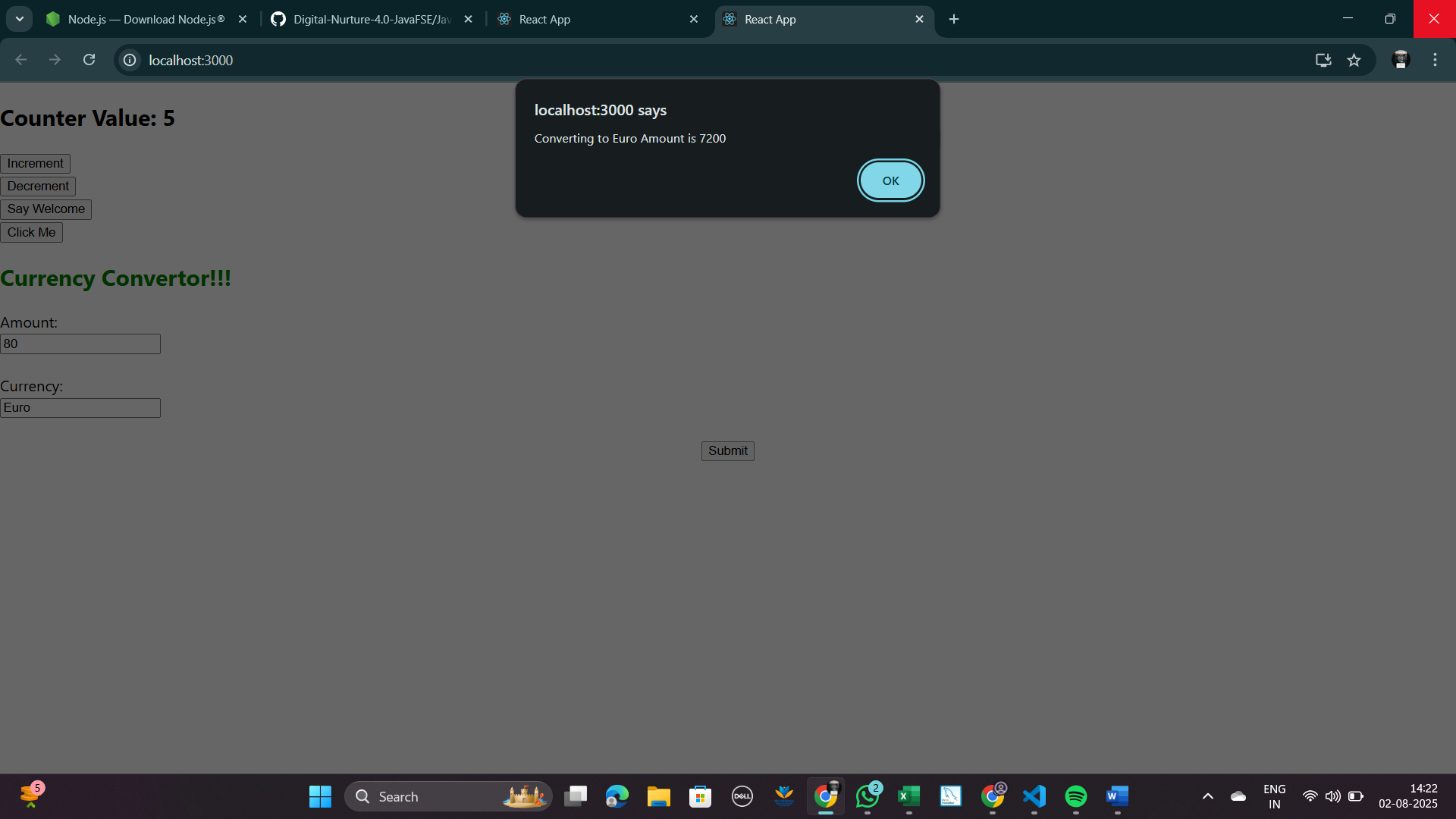
1. **Say Welcome Button Output – First Click and Argument Handling**



1. **Synthetic Event Output – "I was clicked" Alert.**



1. **CurrencyConvertor Output – INR to Euro & Euro to INR Conversion**



**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**