**ReactJS Handson**

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

1. **Create “Increment” button to increase the value of the counter and “Decrement” button to decrease the value of the counter. The “Increase” button should invoke multiple methods.** 
   1. **To increment the value**
   2. **Say Hello followed by a static message.**
2. **Create a button “Say Welcome” which invokes the function which takes “welcome” as an argument.**
3. **Create a button which invokes synthetic event “OnPress” which display “I was clicked”**

**Create a “CurrencyConvertor” component which will convert the Indian Rupees to Euro when the Convert button is clicked.**

**Handle the Click event of the button to invoke the handleSubmit event and handle the conversion of the euro to rupees.**

**App.js**

import React, { Component } from 'react';

import './App.css';

class App extends Component {

  constructor(props) {

    super(props);

    this.state = {

      counter: 0,

      rupees: '',

      euro: ''

    };

    this.handleSayWelcome = this.handleSayWelcome.bind(this);

    this.handleSyntheticClick = this.handleSyntheticClick.bind(this);

    this.handleRupeesChange = this.handleRupeesChange.bind(this);

    this.convertToEuro = this.convertToEuro.bind(this);

  }

  // Method 1 for multiple invocation: Increment counter

  incrementCounter = () => {

    this.setState(prevState => ({

      counter: prevState.counter + 1

    }));

  };

  // Method 2 for multiple invocation: Say Hello

  sayHello = () => {

    alert("Hello!");

  };

  // Method to invoke multiple methods for "Increase" button

  handleIncreaseClick = () => {

    this.incrementCounter();

    this.sayHello();

  };

  // Decrement counter

  decrementCounter = () => {

    this.setState(prevState => ({

      counter: prevState.counter - 1

    }));

  };

  // Method for "Say Welcome" button that takes an argument

  handleSayWelcome(message) {

    alert(`Message: ${message}`);

  }

  // Method for synthetic event "OnPress" (using onClick)

  handleSyntheticClick(event) {

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

    alert("I was clicked!");

  }

  // Currency Converter Handlers

  handleRupeesChange(event) {

    this.setState({ rupees: event.target.value });

  }

  convertToEuro() {

    const { rupees } = this.state;

    const conversionRate = 0.011;

    const euroValue = (parseFloat(rupees) \* conversionRate).toFixed(2);

    this.setState({ euro: euroValue });

  }

  // Function to simulate converting Euro back to Rupees (for handleSubmit concept)

  handleSubmitConversion = (event) => {

    event.preventDefault();

    const { euro } = this.state;

    const conversionRate = 90.91;

    const rupeesValue = (parseFloat(euro) \* conversionRate).toFixed(2);

    alert(`Converting ${euro} Euro back to Rupees: ${rupeesValue} INR`);

  }

  render() {

    return (

      <div className="App">

        <h1>React Event Examples</h1>

        {/\* Counter Section \*/}

        <h2>Counter: {this.state.counter}</h2>

        <button onClick={this.handleIncreaseClick}>Increment & Say Hello</button> {/\*  \*/}

        <button onClick={this.decrementCounter}>Decrement</button> {/\*  \*/}

        <hr />

        {/\* Say Welcome Section \*/}

        <button onClick={() => this.handleSayWelcome("Welcome!")}>Say Welcome</button> {/\*  \*/}

        <hr />

        {/\* Synthetic Event Section \*/}

        <button onClick={this.handleSyntheticClick}>Click me</button> {/\*  \*/}

        <hr />

        {/\* Currency Convertor Section \*/}

        <h2>Currency Convertor (INR to EUR)</h2>

        <div>

          <label>

            Indian Rupees (INR):

            <input

              type="number"

              value={this.state.rupees}

              onChange={this.handleRupeesChange}

            />

          </label>

          <button onClick={this.convertToEuro}>Convert to Euro</button> {/\*  \*/}

          <h3>Euro (EUR): {this.state.euro}</h3>

        </div>

        {/\* Form for handleSubmit demonstration (Euro to Rupees conversion) \*/}

        <h2>Convert Euro back to Rupees</h2>

        <form onSubmit={this.handleSubmitConversion}> {/\*  \*/}

          <label>

            Euro (EUR):

            <input

              type="number"

              value={this.state.euro}

              onChange={(e) => this.setState({ euro: e.target.value })}

            />

          </label>

          <button type="submit">Convert Euro to INR</button>

        </form>

      </div>

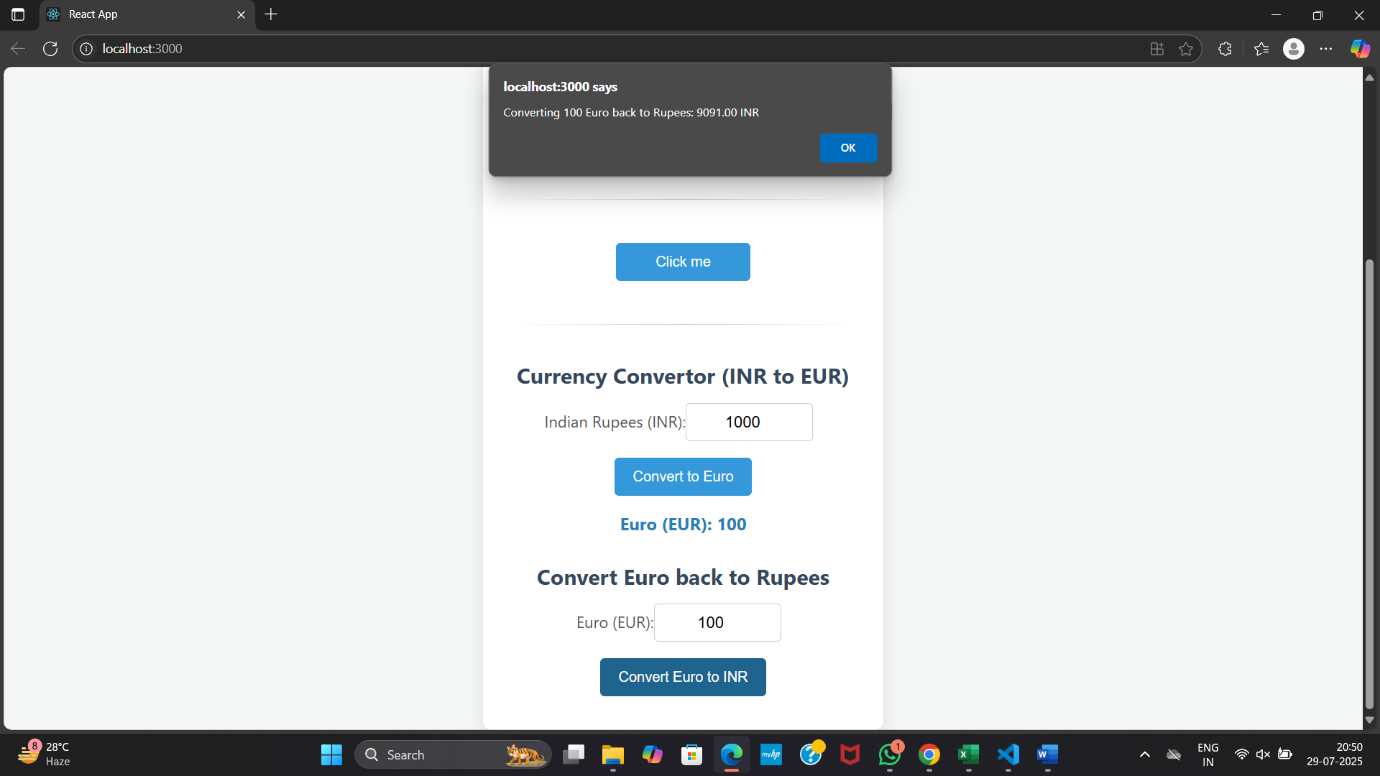
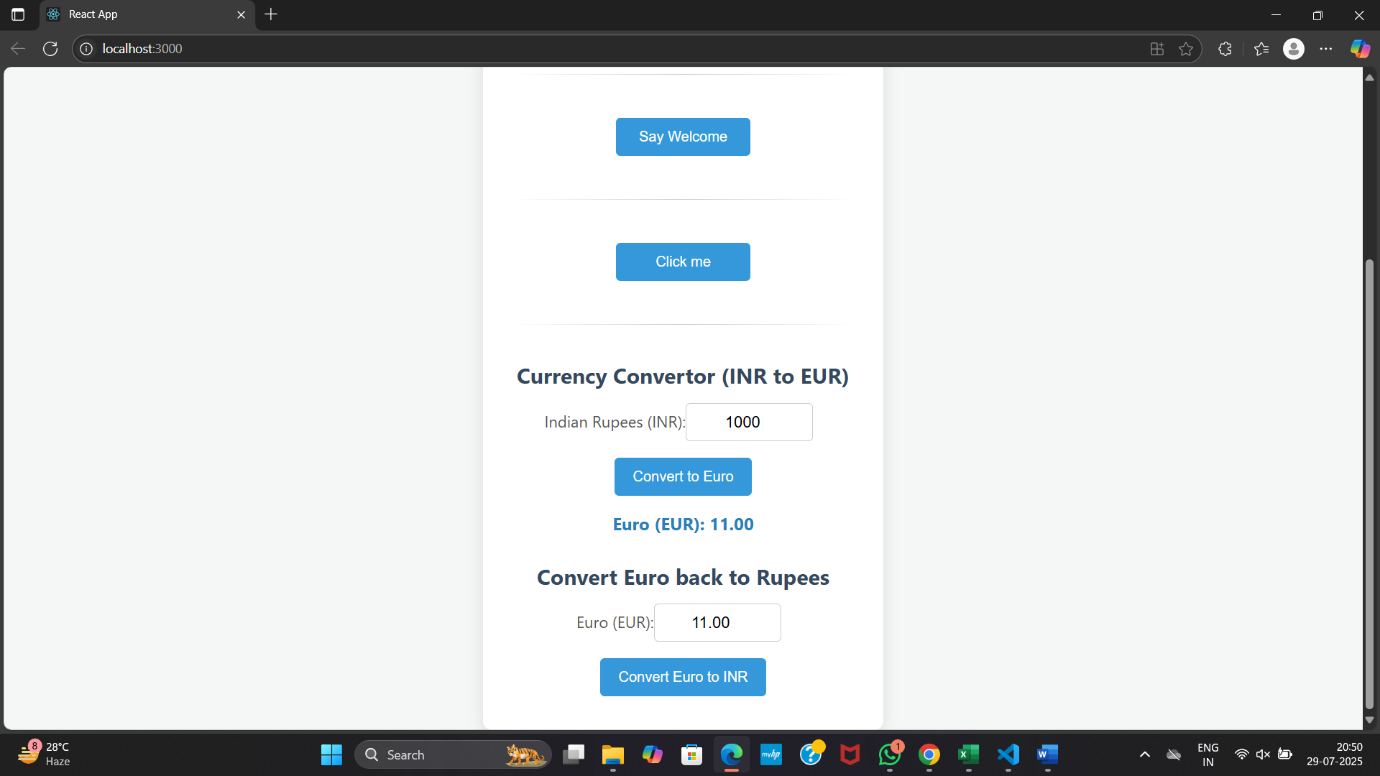
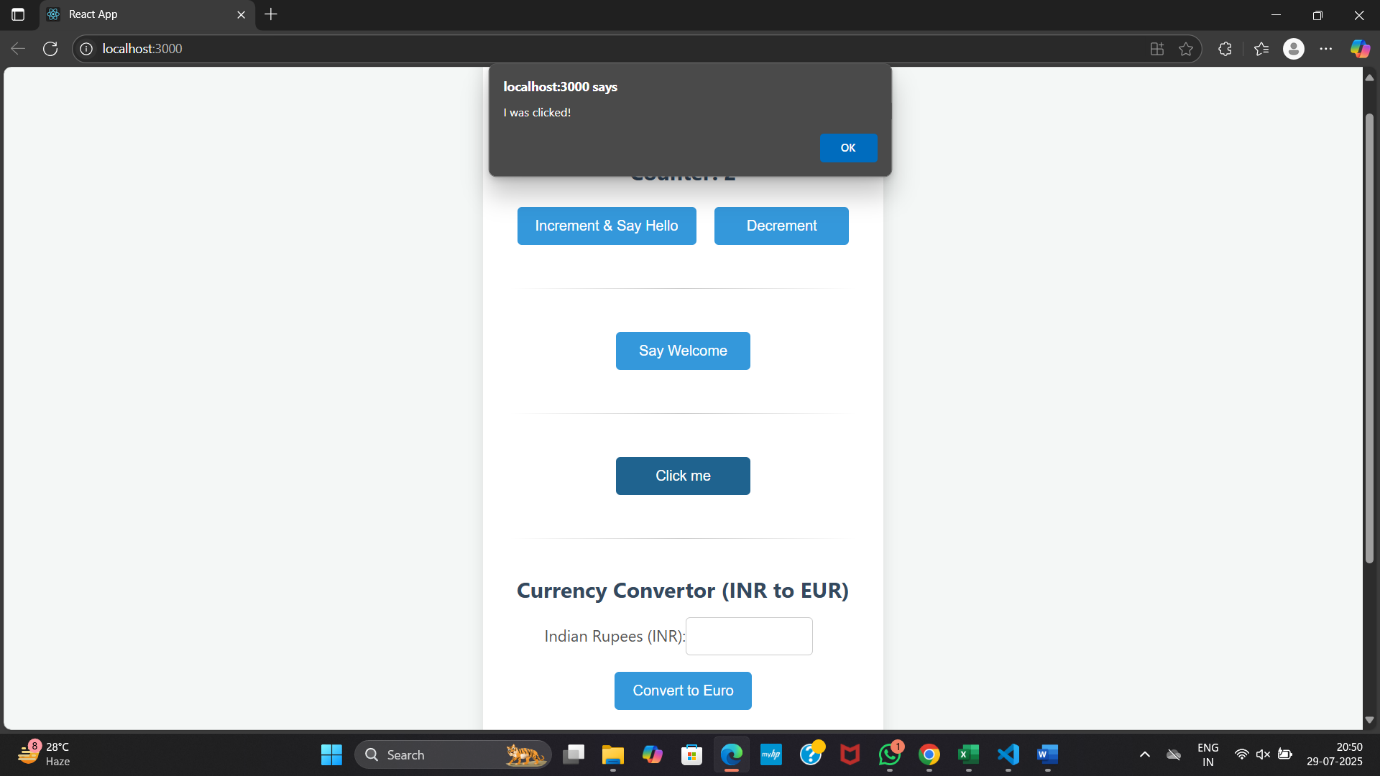
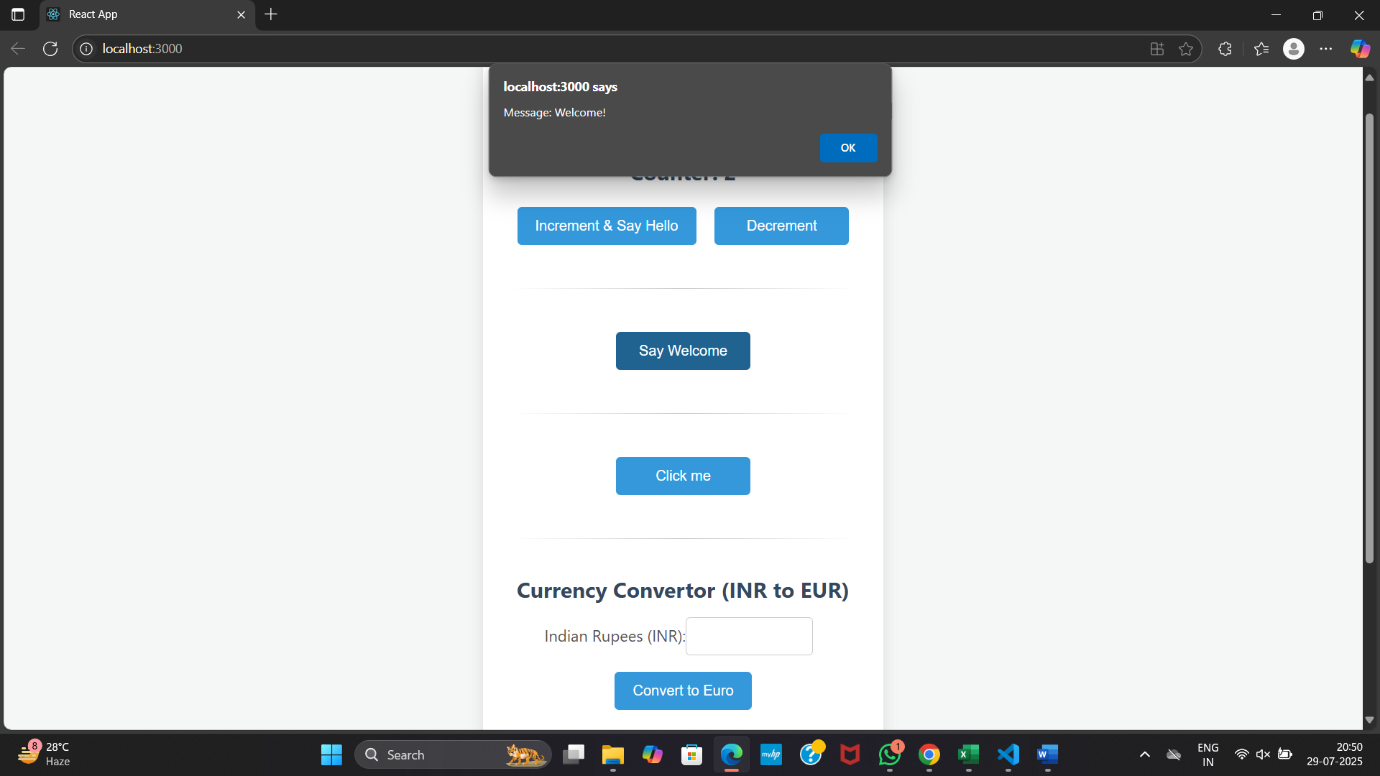
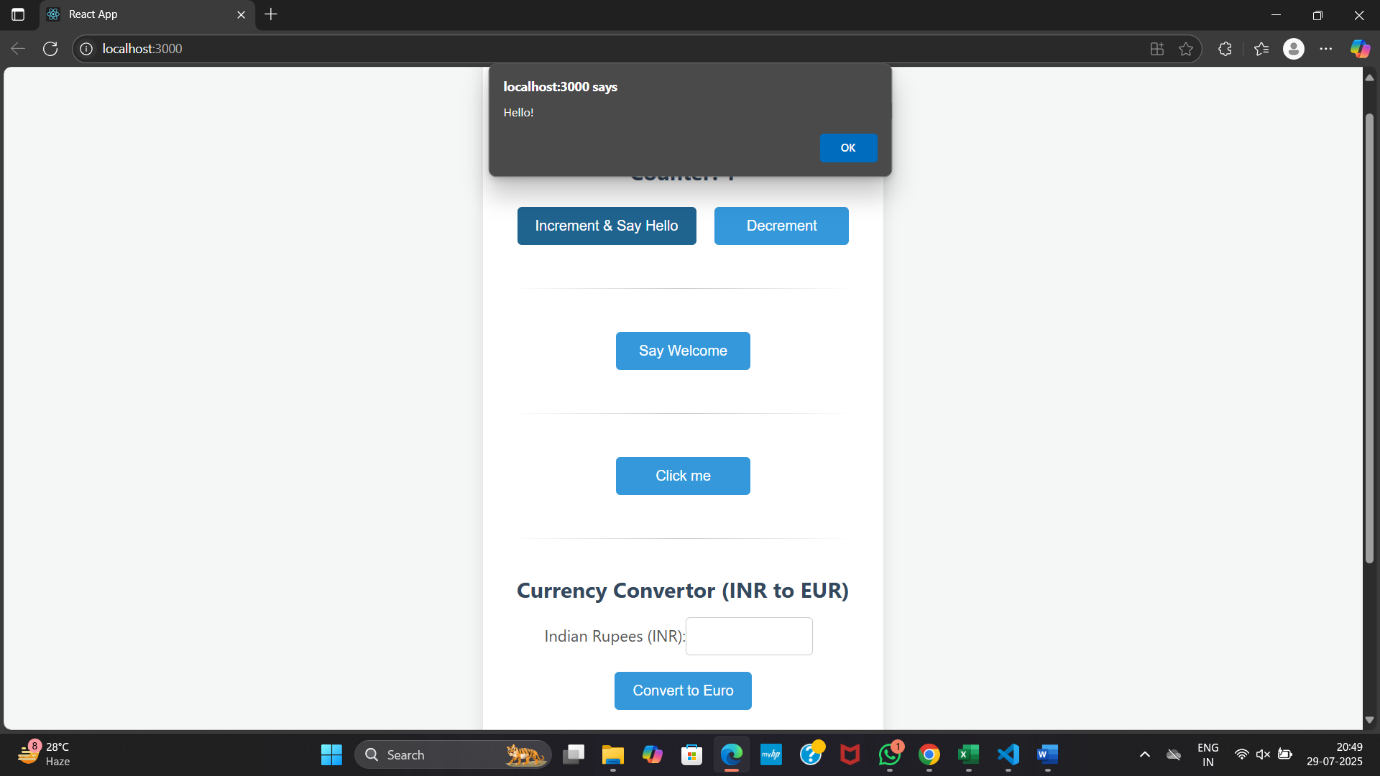
    );

  }

}

export default App;

**OUTPUT**

****