## **Objectives**

* Explain React events
* Explain about event handlers
* Define Synthetic event
* Identify React event naming convention

In this hands-on lab, you will learn how to:

* Implement Event handling concept in React applications
* Use this keyword
* Use synthetic event

## **Prerequisites**

The following is required to complete this hands-on lab:

* Node.js
* NPM
* Visual Studio Code

## **Notes**

Estimated time to complete this lab: **90 minutes.**

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.



1. Create a button “Say Welcome” which invokes the function which takes “welcome” as an argument.



1. 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.



**Program:**

**App.js:**

import React from 'react';

import Counter from './components/Counter';

import WelcomeButton from './components/WelcomeButton';

import SyntheticEventExample from './components/SyntheticEventExample';

import CurrencyConverter from './components/CurrencyConverter';

function App() {

return (

<div style={{ padding: "20px" }}>

<h1>🎯 React Event Examples</h1>

<Counter />

<hr />

<WelcomeButton />

<hr />

<SyntheticEventExample />

<hr />

<CurrencyConverter />

</div>

);

}

export default App;

**counter.js:**

import React, { Component } from 'react';

class Counter extends Component {

constructor(props) {

super(props);

this.state = {

count: 0

};

}

increment = () => {

this.setState({ count: this.state.count + 1 });

this.sayHello();

}

decrement = () => {

this.setState({ count: this.state.count - 1 });

}

sayHello = () => {

console.log("Hello! This is an increment operation.");

}

render() {

return (

<div>

<h2>Counter</h2>

<p>Count: {this.state.count}</p>

<button onClick={this.increment}>Increment</button>{' '}

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

</div>

);

}

}

export default Counter;

**WelcomeButton.js:**

import React from 'react';

const WelcomeButton = () => {

const sayWelcome = (message) => {

alert(`Say Welcome: ${message}`);

};

return (

<div>

<h2>Welcome Button</h2>

<button onClick={() => sayWelcome("Welcome to React Events!")}>Say Welcome</button>

</div>

);

};

export default WelcomeButton;

**SyntheticEventExample.js:**

import React from 'react';

const SyntheticEventExample = () => {

const handleClick = (event) => {

alert("I was clicked");

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

};

return (

<div>

<h2>Synthetic Event</h2>

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

</div>

);

};

export default SyntheticEventExample;

**CurrencyConverter.js:**

import React, { useState } from 'react';

const CurrencyConverter = () => {

const [rupees, setRupees] = useState('');

const [euro, setEuro] = useState('');

const handleSubmit = (e) => {

e.preventDefault();

const euroValue = parseFloat(rupees) / 90; // Assume ₹90 = 1€

setEuro(euroValue.toFixed(2));

};

return (

<div>

<h2>Currency Converter</h2>

<form onSubmit={handleSubmit}>

<label>Indian Rupees: </label>

<input

type="number"

value={rupees}

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

/>

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

</form>

{euro && <p>Euro: €{euro}</p>}

</div>

);

};

export default CurrencyConverter;

**index.js:**

import React from 'react';

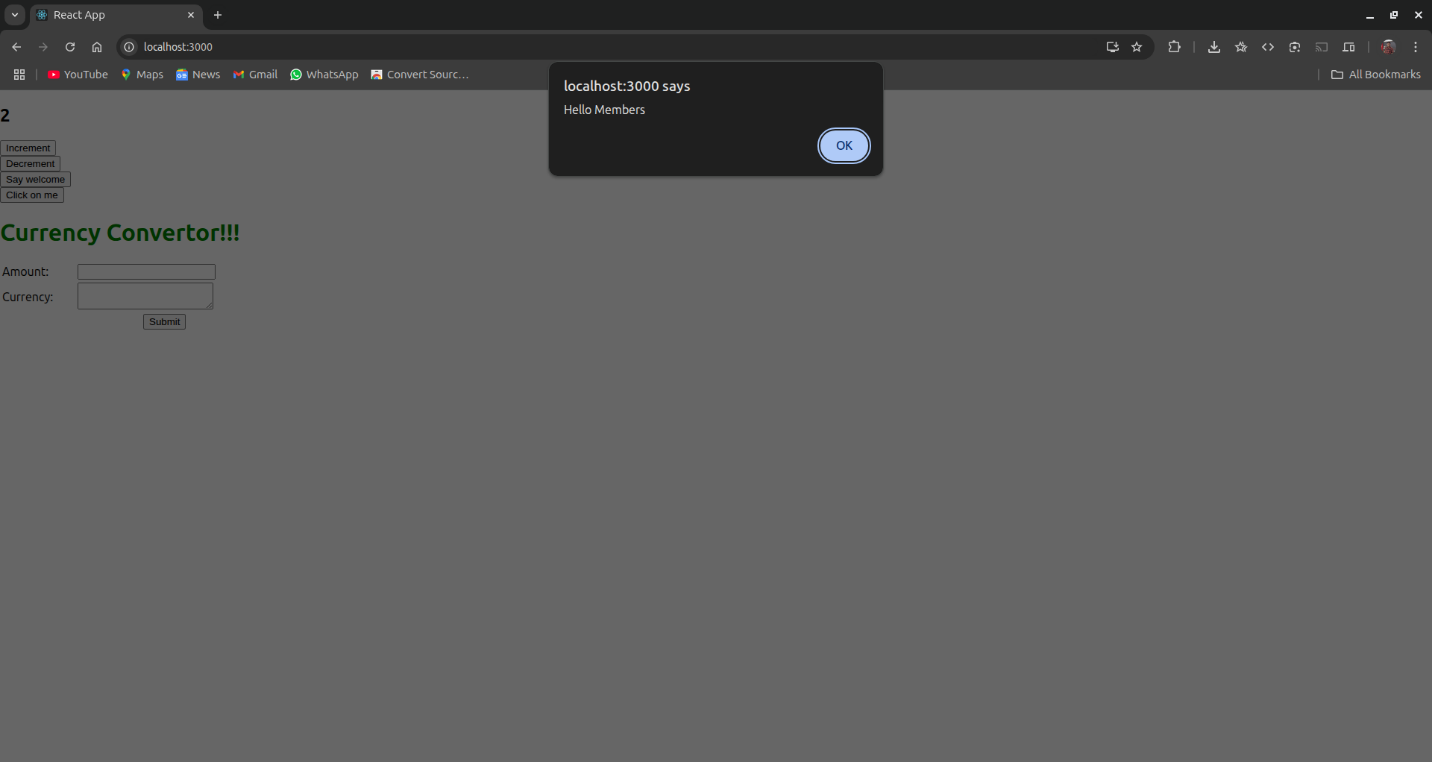
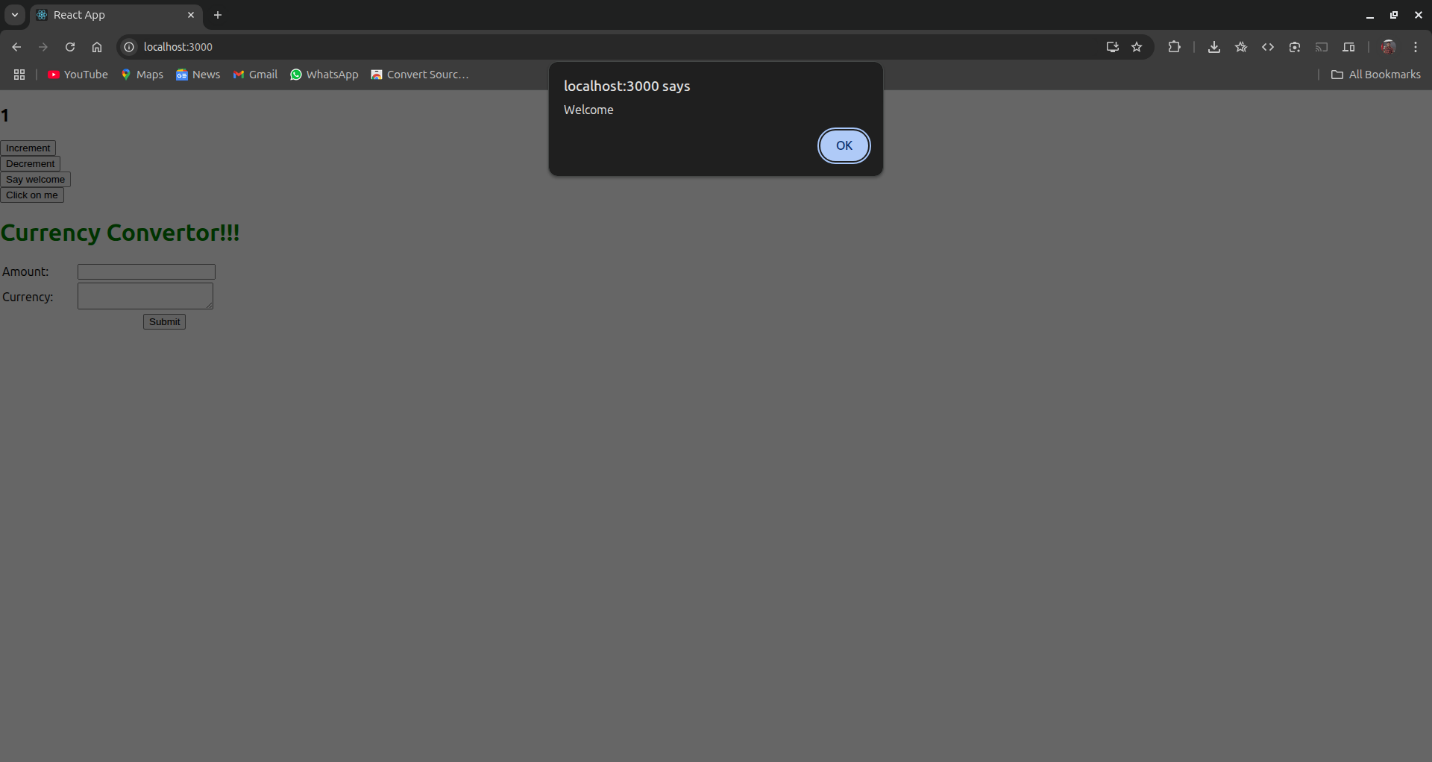
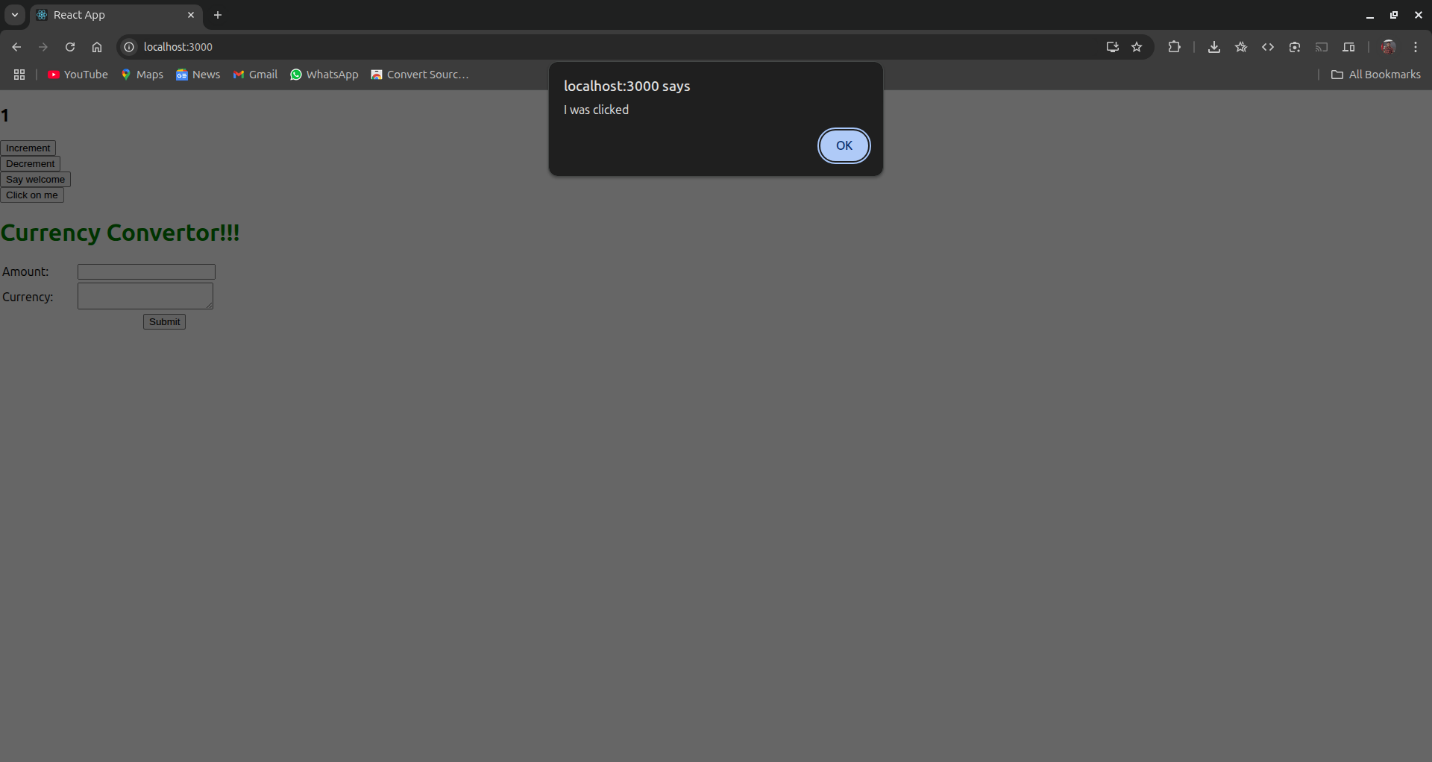
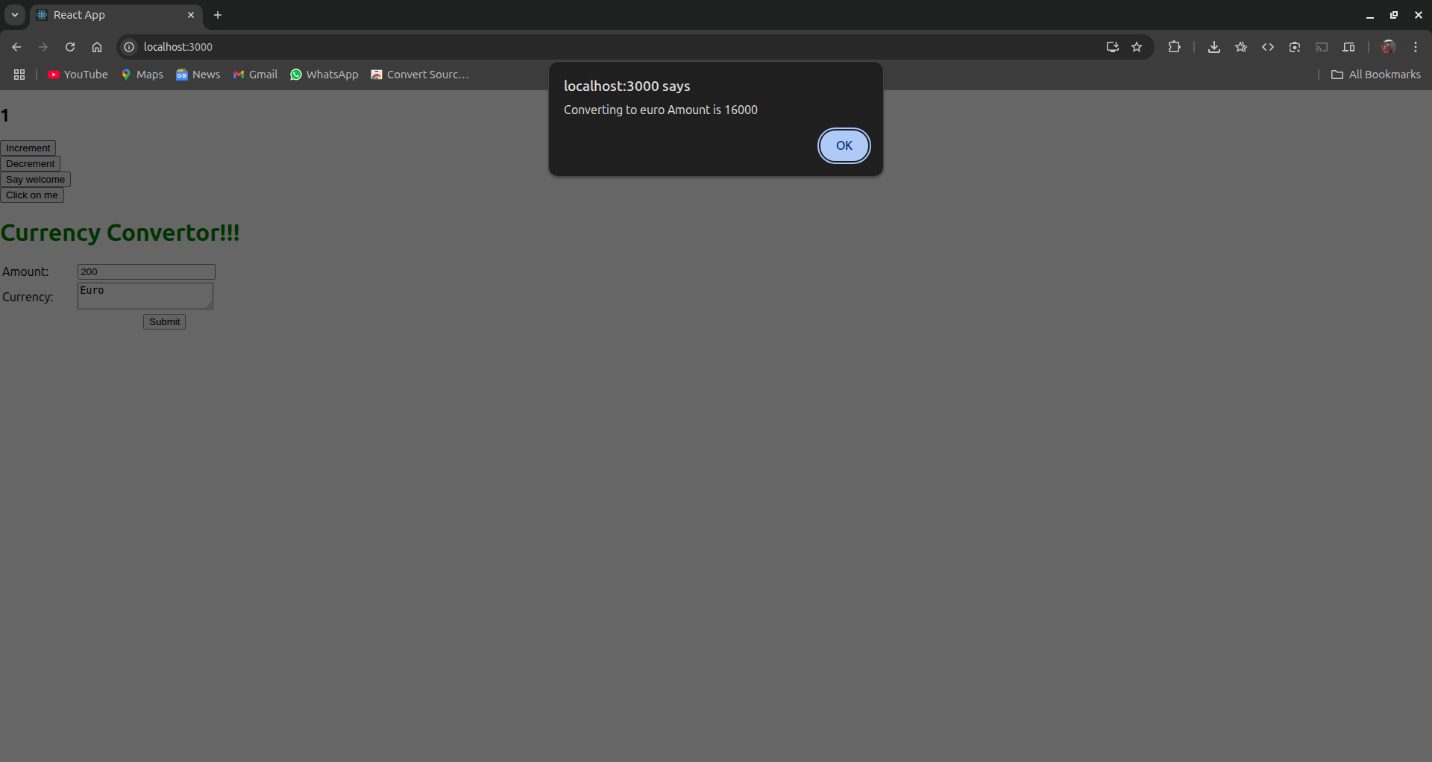
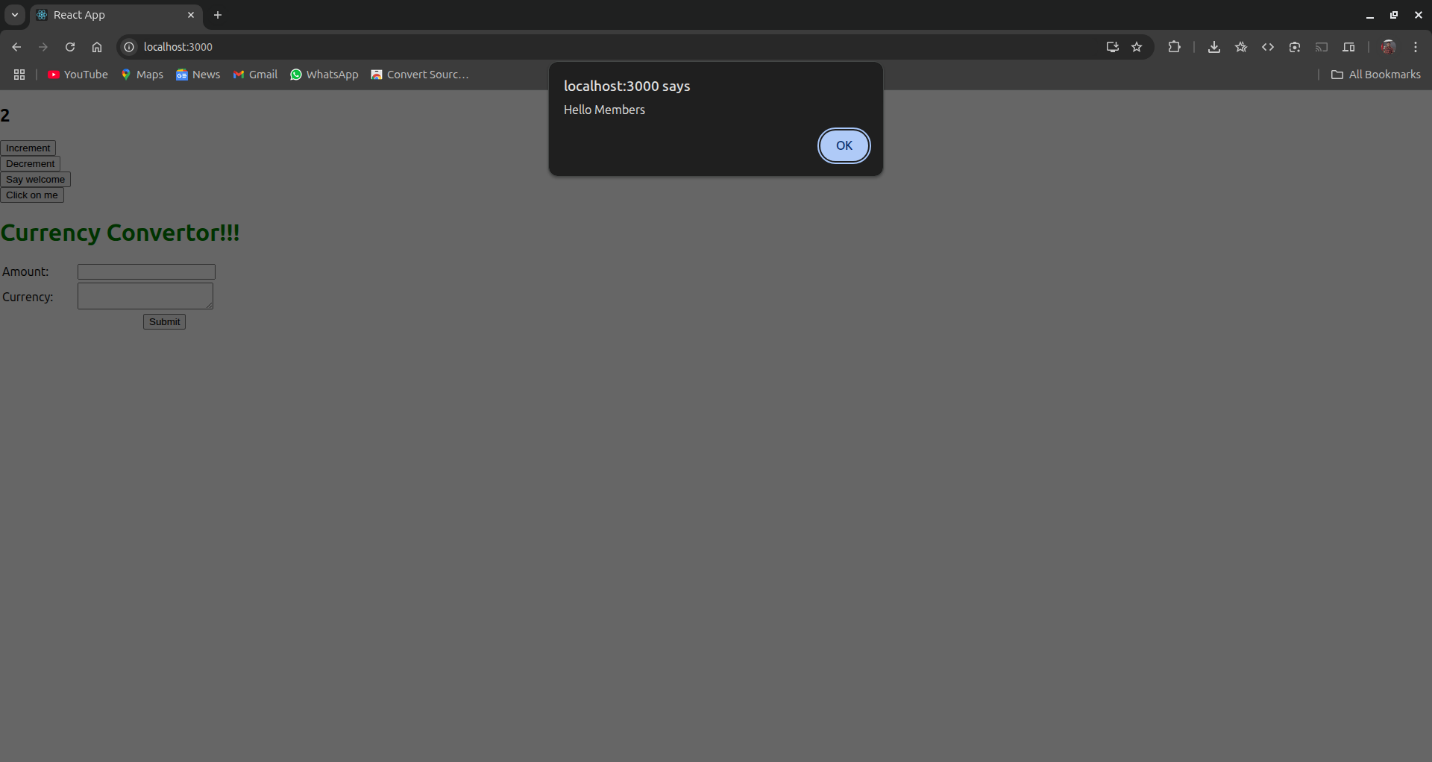
import ReactDOM from 'react-dom/client';

import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<App />);

**Output**

**:**