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

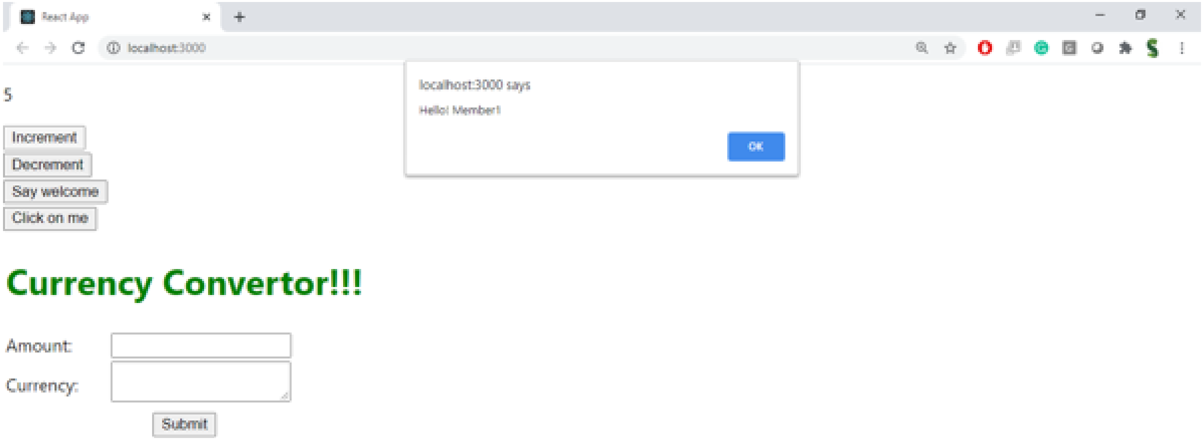
Notes

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.
    a. To increment the value

b. Say Hello followed by a static message.



localhost:3000 says

Hello! Member1

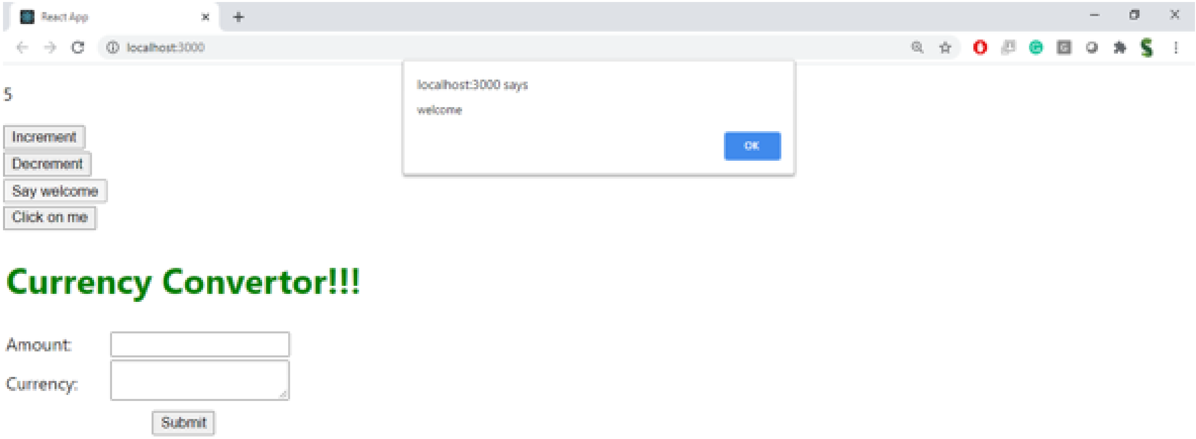
5

Currency Convertor!!!

Amount:

Currency:

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



localhost:3000 says

welcome

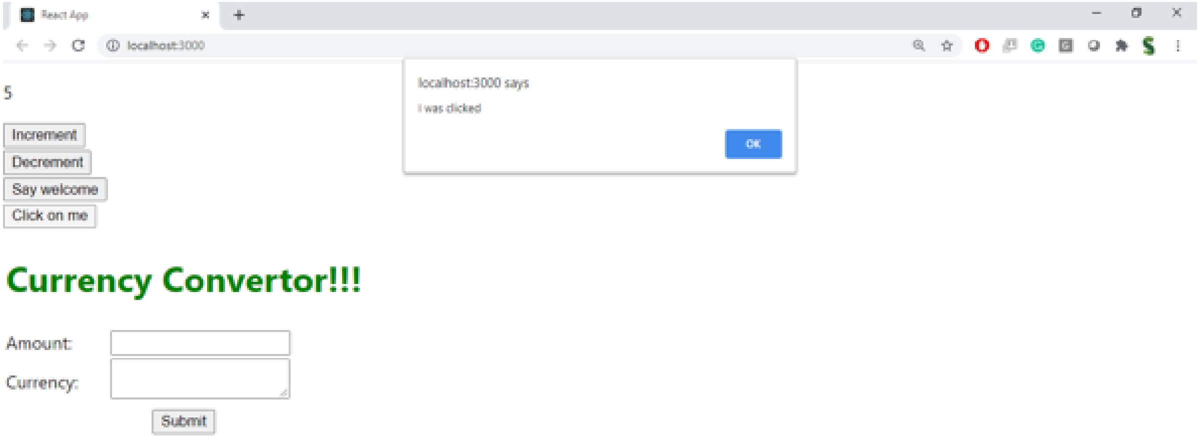
5

Currency Convertor!!!

Amount:

Currency:

3. Create a button which invokes synthetic event “OnPress” which display “I was clicked”



localhost:3000 says

I was clicked

5

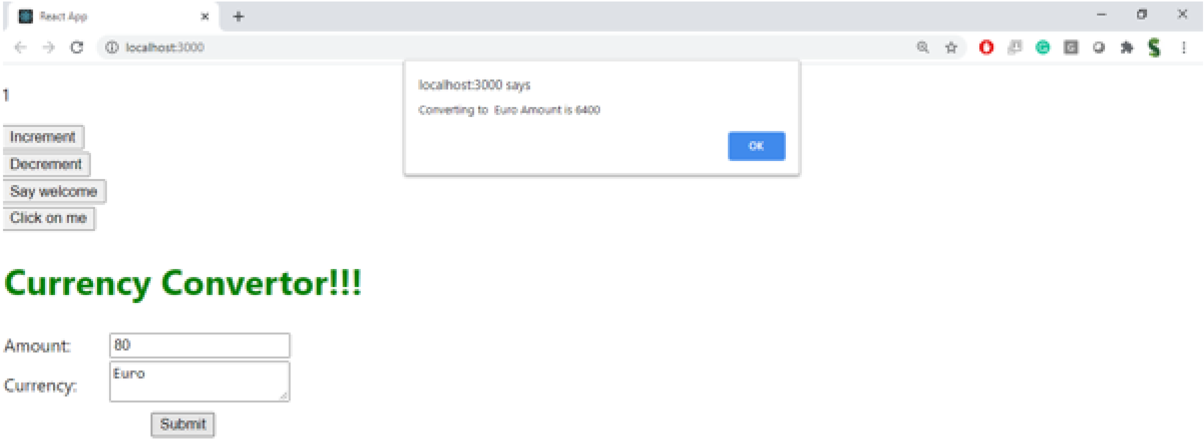
Currency Convertor!!!

Amount:

Currency:

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.



localhost:3000 says

Converting to Euro Amount is 6400

1

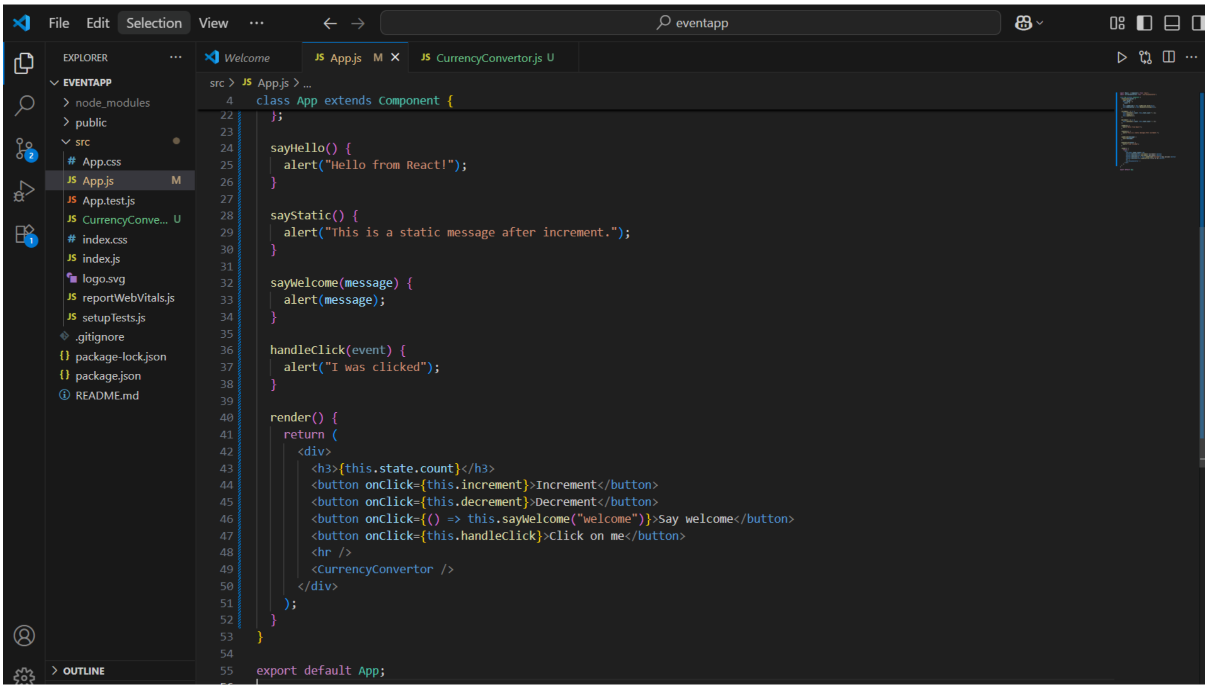
Currency Convertor!!!

Amount: 80

Currency: Euro

Code:

App.js:



```
class App extends Component {
  };

  sayHello() {
    alert("Hello from React!");
  }

  sayStatic() {
    alert("This is a static message after increment.");
  }

  sayWelcome(message) {
    alert(message);
  }

  handleClick(event) {
    alert("I was clicked");
  }

  render() {
    return (
      <div>
        <h3>{this.state.count}</h3>
        <button onClick={this.increment}>Increment</button>
        <button onClick={this.decrement}>Decrement</button>
        <button onClick={() => this.sayWelcome("welcome")}>Say welcome</button>
        <button onClick={this.handleClick}>Click on me</button>
        <hr />
        <CurrencyConvertor />
      </div>
    );
  }
}

export default App;
```

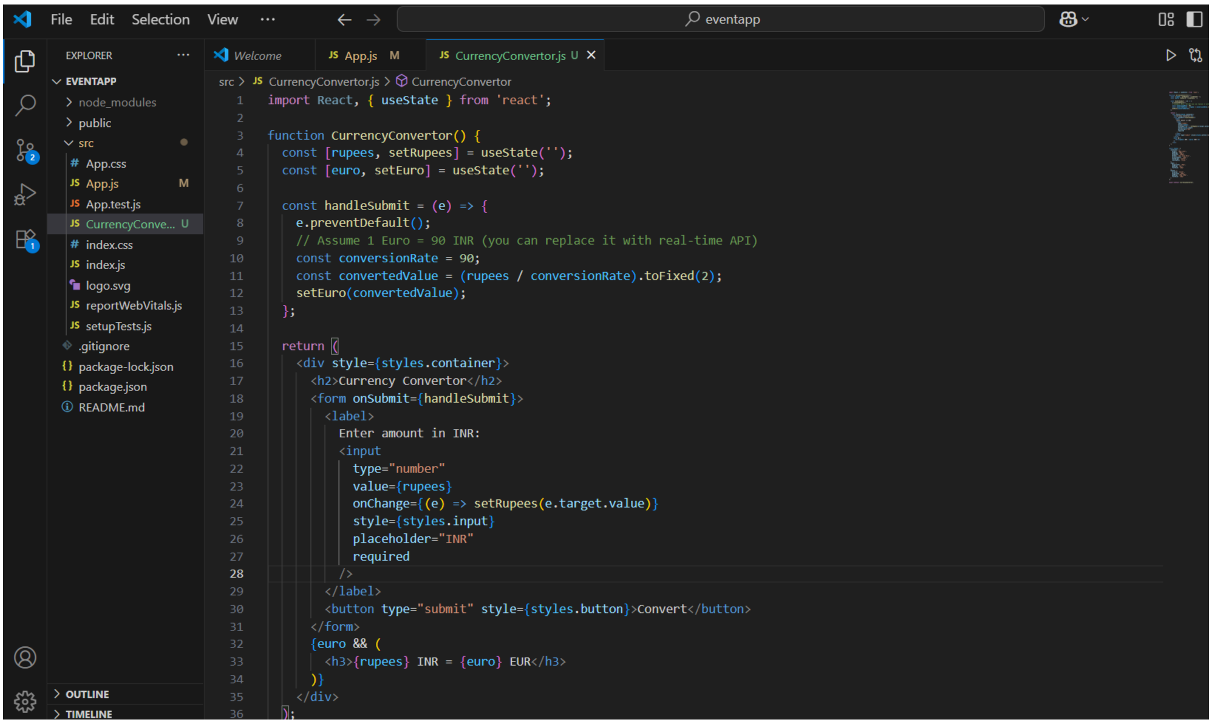
import React, { Component } from 'react';

import CurrencyConvertor from './CurrencyConvertor';

class App extends Component {

```
constructor(props) {
  super(props);
  this.state = {
    count: 5
  };
  this.sayWelcome = this.sayWelcome.bind(this);
  this.handleClick = this.handleClick.bind(this);
}

increment = () => {
  this.setState({ count: this.state.count + 1 });
  this.sayHello();
  this.sayStatic();
};

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

sayHello() {
  alert("Hello from React!");
}

sayStatic() {
  alert("This is a static message after increment.");
}

sayWelcome(message) {
    alert(message);
  }

  handleClick(event) {
    alert("I was clicked");
  }

  render() {
    return (
      <div>
        <h3>{this.state.count}</h3>
        <button onClick={this.increment}>Increment</button>
        <button onClick={this.decrement}>Decrement</button>
        <button onClick={() => this.sayWelcome("welcome")}>Say welcome</button>
        <button onClick={this.handleClick}>Click on me</button>
        <hr />
        <CurrencyConvertor />
      </div>
    );
  }
}
export default App;
```

CurrencyConverter.js:

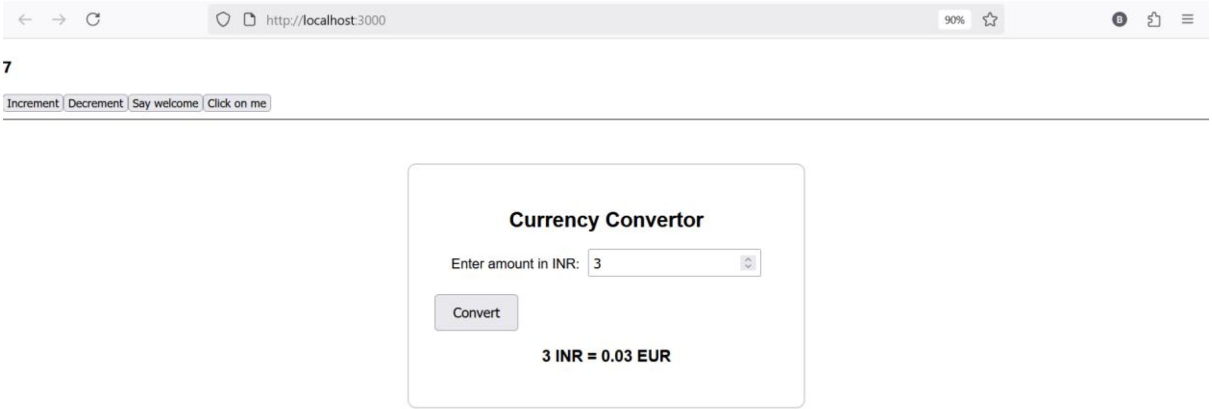


```
import React, { useState } from 'react';

function CurrencyConvertor() {
  const [rupees, setRupees] = useState('');
  const [euro, setEuro] = useState('');

  const handleSubmit = (e) => {
    e.preventDefault();
    // Assume 1 Euro = 90 INR (you can replace it with real-time API)
    const conversionRate = 90;
    const convertedValue = (rupees / conversionRate).toFixed(2);
    setEuro(convertedValue);
  };

  return (
    <div style={styles.container}>
      <h2>Currency Convertor</h2>
      <form onSubmit={handleSubmit}>
        <label>
          Enter amount in INR:
          <input
            type="number"
            value={rupees}
            onChange={(e) => setRupees(e.target.value)}
            style={styles.input}
            placeholder="INR"
            required
          />
        </label>
        <button type="submit" style={styles.button}>Convert</button>
      </form>
      {euro && (
        <h3>{rupees} INR = {euro} EUR</h3>
      )}
    </div>
  );
```

```
import React, { useState } from 'react';


function CurrencyConvertor() {
  const [rupees, setRupees] = useState('');
  const [euro, setEuro] = useState('');


  const handleSubmit = (e) => {
    e.preventDefault();
    // Assume 1 Euro = 90 INR (you can replace it with real-time API)
    const conversionRate = 90;
    const convertedValue = (rupees / conversionRate).toFixed(2);
    setEuro(convertedValue);
  };


  return (
    <div style={styles.container}>
```

```
      <h2>Currency Convertor</h2>
      <form onSubmit={handleSubmit}>
        <label>
          Enter amount in INR:
          <input
            type="number"
            value={rupees}
            onChange={(e) => setRupees(e.target.value)}
            style={styles.input}
            placeholder="INR"
            required
          />
        </label>
        <button type="submit" style={styles.button}>Convert</button>
      </form>
      {euro && (
        <h3>{rupees} INR = {euro} EUR</h3>
      )}
    </div>
  );
}

const styles = {
  container: {
    margin: '50px auto',
    padding: '30px',
    maxWidth: '400px',
    border: '2px solid #ccc',
    borderRadius: '10px',
    textAlign: 'center',
    fontFamily: 'Arial'
  },
  input: {
    marginLeft: '10px',
    padding: '5px',
    fontSize: '16px'
  },
  button: {
    display: 'block',
    marginTop: '20px',
    padding: '10px 20px',
    fontSize: '16px'
  }
};

export default CurrencyConvertor;
```

OUTPUT:



7

Currency Convertor

Enter amount in INR: 3

3 INR = 0.03 EUR