**Objectives**

**Explain React events**

React events are JavaScript events (like click, change, submit) that are handled using React's custom event system, not the browser’s default.  
They provide a consistent interface across all browsers using a wrapper called SyntheticEvent.

Events in React are handled similarly to HTML but with slight differences in syntax and behavior.

**Explain Event Handlers**

Event handlers in React are **functions** that are called when a specific event occurs — like clicking a button or submitting a form.

You pass a **function reference** to an event prop like onClick, onChange, etc.

**Example:**

function handleClick() {

alert("Button clicked!");

}

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

Don't use quotes or call the function directly like onClick="handleClick()".

**Define Synthetic Event**

A **SyntheticEvent** is a **wrapper** around the native browser event object, provided by React.

It normalizes events so they work consistently across **all browsers**.

Key features:

* Has the same interface as the browser event
* Works identically in all browsers
* Automatically **pooled for performance** (in older React versions)

Example:

function handleChange(event) {

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

}

**React Event Naming Convention**

React uses **camelCase** for event names, unlike HTML which uses lowercase.

|  |  |
| --- | --- |
| **HTML** | **React** |
| onclick | onClick |
| onchange | onChange |
| onsubmit | onSubmit |
| onkeydown | onKeyDown |

Example:

<input onChange={handleInputChange} />

**Code :**

**--App.js**

import React, { useState } from 'react';

import CurrencyConverter from './CurrencyConverter';

function App() {

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

  const increment = () => {

    setCount(prev => prev + 1);

  };

  const sayHello = () => {

    alert("Hello! This is a static message.");

  };

  const handleIncrement = () => {

    increment();

    sayHello();

  };

  const sayWelcome = (msg) => {

    alert(`localhost:3000 says\n${msg}`);

  };

  const handleClick = () => {

    alert("localhost:3000 says\nI was clicked");

  };

  return (

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

      <div style={{ marginBottom: "10px" }}>

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

      </div>

      <div style={{ marginBottom: "10px" }}>

        <button onClick={() => setCount(prev => prev - 1)}>Decrement</button>

      </div>

      <div style={{ marginBottom: "10px" }}>

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

      </div>

      <div style={{ marginBottom: "10px" }}>

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

      </div>

      <p>Count: {count}</p>

      <h2 style={{ color: "green" }}>Currency Convertor!!!</h2>

      <CurrencyConverter />

    </div>

  );

}

export default App;

**--CurrencyConverter.js**

import React, { useState } from 'react';

function CurrencyConverter() {

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

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

  const handleSubmit = (e) => {

    e.preventDefault();

    const rupees = parseFloat(amount);

    const euroRate = 0.011; // 1 INR = 0.011 EUR (as per requirement)

    if (isNaN(rupees)) {

      alert("Please enter a valid numeric amount in INR.");

      return;

    }

    const euro = (rupees \* euroRate).toFixed(2);

    alert(`localhost:3000 says\nConverting to Euro: Amount is €${euro}`);

  };

  return (

    <form onSubmit={handleSubmit}>

      <div style={{ marginBottom: "10px" }}>

        <label>

          Amount (INR):{" "}

          <input

            type="number"

            value={amount}

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

            required

          />

        </label>

      </div>

      <div style={{ marginBottom: "10px" }}>

        <label>

          Currency:{" "}

          <input

            type="text"

            value={currency}

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

            placeholder="e.g. Euro"

            required

          />

        </label>

      </div>

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

    </form>

  );

}

export default CurrencyConverter;

**Output :**







