**Explain React events**

**React Events**

React events are **synthetic events** that wrap around native browser events to provide a consistent behaviour across different browsers.  
They work similarly to regular DOM events (like click, change, submit), but React handles them internally for better performance and cross-browser compatibility.

Examples of React events:

* onClick — when a user clicks an element
* onChange — when the value of an input changes
* onSubmit — when a form is submitted
* onMouseOver — when the mouse hovers over an element

**Explain about event handlers**

**Event Handlers**

Event handlers are functions that get called when an event occurs. In React, you **pass a function reference** as a prop to the event attribute.

Example:

function handleClick() {

alert('Button clicked!');

}

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

* The function handleClick is called when the button is clicked.
* You do not call the function directly (i.e., no parentheses () when passing the handler).
* Event handlers receive an event object by default.

**Define Synthetic event**

**Synthetic Event**

A Synthetic Event is React’s cross-browser wrapper around the browser’s native event. It has the same interface as native events but works identically across all browsers.

Benefits of Synthetic Events:

* Normalizes events so code behaves consistently across browsers
* React handles event delegation efficiently, improving performance
* Events are pooled and reused for optimization (so event properties are nullified after event callback)

You can access event properties like event.target, event.preventDefault(), etc., just like with native events.

**Identify React event naming convention**

**React Event Naming Convention**

React uses **camelCase** naming for event handlers, which is different from plain HTML where event attributes are lowercase.

| **Native HTML event** | **React event attribute** |
| --- | --- |
| onclick | onClick |
| onchange | onChange |
| onsubmit | onSubmit |
| onmouseover | onMouseOver |

Also, React event attributes are passed JavaScript functions, not strings.

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

**Index.js :-**

import React from 'react';

import ReactDOM from 'react-dom/client';

import './index.css';

import App from './App';

import reportWebVitals from './reportWebVitals';

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

root.render(

  <React.StrictMode>

    <App />

  </React.StrictMode>

);

reportWebVitals();

**App.js :-**

import React, { Component } from "react";

import CurrencyConvertor from "./CurrencyConvertor";

class App extends Component {

  constructor(props) {

    super(props);

    this.state = {

      counter: 0,

    };

    // Bind methods

    this.increment = this.increment.bind(this);

    this.sayHello = this.sayHello.bind(this);

    this.handleIncrementClick = this.handleIncrementClick.bind(this);

    this.decrement = this.decrement.bind(this);

    this.sayWelcome = this.sayWelcome.bind(this);

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

  }

  increment() {

    this.setState((prevState) => ({ counter: prevState.counter + 1 }));

  }

  sayHello() {

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

  }

  handleIncrementClick() {

    this.increment();

    this.sayHello();

  }

  decrement() {

    this.setState((prevState) => ({ counter: prevState.counter - 1 }));

  }

  sayWelcome(message) {

    alert(message);

  }

  handleSyntheticClick() {

    alert("I was clicked");

  }

  render() {

    return (

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

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

        {/\* Increment button calls two methods \*/}

        <button onClick={this.handleIncrementClick}>Increment</button> <br /><br />

        {/\* Decrement button \*/}

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

        {/\* Say Welcome button \*/}

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

        {/\* Synthetic event button \*/}

        <button onClick={this.handleSyntheticClick}>Click me (Synthetic Event)</button>

        <hr />

        {/\* Currency Converter Component \*/}

        <CurrencyConvertor />

      </div>

    );

  }

}

export default App;

**CurrencyConvertor.js :-**

import React, { Component } from "react";

class CurrencyConvertor extends Component {

constructor(props) {

super(props);

this.state = {

amount: "",

converted: null,

};

this.handleInputChange = this.handleInputChange.bind(this);

this.handleSubmit = this.handleSubmit.bind(this);

}

handleInputChange(event) {

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

}

handleSubmit(event) {

event.preventDefault();

const { amount } = this.state;

const inrAmount = parseFloat(amount);

if (isNaN(inrAmount) || inrAmount <= 0) {

alert("Please enter a valid positive number");

this.setState({ converted: null });

return;

}

// Conversion rate: 1 INR = 0.011 Euro (example)

const conversionRate = 0.011;

const euroAmount = inrAmount \* conversionRate;

this.setState({ converted: euroAmount.toFixed(2) });

}

render() {

const { amount, converted } = this.state;

return (

<div>

<h3 style={{color:"green"}}>Currency Converter (INR to Euro)</h3>

<form onSubmit={this.handleSubmit}>

<input

type="number"

placeholder="Enter amount in INR"

value={amount}

onChange={this.handleInputChange}

required

min="0"

/>

<button type="submit" style={{ marginLeft: "10px" }}>

Convert

</button>

</form>

{converted !== null && (

<p>

Converted Amount: <strong>{converted} Euro</strong>

</p>

)}

</div>

);

}

}

export default CurrencyConvertor;





