11.REACTJS-HOL

11.REACTJS-HOL

import React, { useState } from 'react';

// A reusable, styled section component for better organization.

const Section = ({ title, children }) => (

<div className="bg-white p-6 rounded-lg shadow-md border border-gray-200 mb-8">

<h2 className="text-2xl font-bold text-gray-800 mb-4 border-b pb-2">{title}</h2>

{children}

</div>

);

// A reusable, styled button component.

const Button = ({ onClick, children, className = 'bg-blue-600 hover:bg-blue-700' }) => (

<button

onClick={onClick}

className={`text-white font-bold py-2 px-4 rounded-lg transition duration-300 ease-in-out shadow ${className}`}

>

{children}

</button>

);

// --- 1. Counter Component ---

// Demonstrates basic state updates and calling multiple functions from one event.

function Counter() {

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

// Method to increment the counter value.

const incrementValue = () => {

setCount(prevCount => prevCount + 1);

};

// Method to display a message.

const sayHello = () => {

alert('Hello! The counter will now be increased.');

};

// Requirement: The "Increment" button should invoke multiple methods.

// This handler calls both `sayHello` and `incrementValue`.

const handleIncrementClick = () => {

sayHello();

incrementValue();

};

// Handler for the "Decrement" button.

const handleDecrementClick = () => {

setCount(prevCount => prevCount - 1);

};

return (

<Section title="Counter">

<div className="text-center">

<p className="text-4xl font-bold text-gray-700 mb-4">{count}</p>

<div className="flex justify-center gap-4">

<Button onClick={handleIncrementClick}>Increment</Button>

<Button onClick={handleDecrementClick} className="bg-red-600 hover:bg-red-700">Decrement</Button>

</div>

</div>

</Section>

);

}

// --- 2. Argument Button Component ---

// Demonstrates how to pass an argument to an event handler function.

function ArgumentButton() {

// Requirement: Invoke a function that takes "welcome" as an argument.

const showMessage = (message) => {

alert(message);

};

return (

<Section title="Passing Arguments to Events">

{/\* We use an arrow function in the onClick prop to call `showMessage` with a specific argument. \*/}

<Button onClick={() => showMessage('Welcome to our Event Handling App!')}>

Say Welcome

</Button>

</Section>

);

}

// --- 3. Synthetic Event Component ---

// Demonstrates the use of React's SyntheticEvent object.

function SyntheticEventButton() {

// Requirement: Invoke a synthetic event "OnPress" (we'll use onClick for web).

// The event object `e` passed here is a React SyntheticEvent.

const handleClick = (e) => {

// SyntheticEvent provides a consistent interface across browsers.

console.log('SyntheticEvent Type:', e.type); // e.g., "click"

alert('I was clicked! Check the console for the synthetic event object.');

};

return (

<Section title="Synthetic Event Demo">

<Button onClick={handleClick}>

Click for Synthetic Event

</Button>

</Section>

);

}

// --- 4. Currency Converter Component ---

// Demonstrates handling form events and state management in a component.

function CurrencyConverter() {

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

const [euros, setEuros] = useState(0);

const EURO\_CONVERSION\_RATE = 0.011; // As of late 2023, 1 INR is approx 0.011 EUR

// Handles changes in the input field.

const handleInputChange = (e) => {

setRupees(e.target.value);

};

// Requirement: Handle the Click event to invoke handleSubmit.

// This function prevents the default form submission behavior and performs the conversion.

const handleSubmit = (e) => {

e.preventDefault(); // Prevents page reload on form submission

const rupeesValue = parseFloat(rupees);

if (!isNaN(rupeesValue)) {

setEuros(rupeesValue \* EURO\_CONVERSION\_RATE);

} else {

alert('Please enter a valid number for Rupees.');

}

};

return (

<Section title="Currency Converter">

<form onSubmit={handleSubmit} className="space-y-4">

<div>

<label htmlFor="rupees" className="block text-lg font-medium text-gray-700 mb-1">

Indian Rupees (INR)

</label>

<input

type="number"

id="rupees"

value={rupees}

onChange={handleInputChange}

placeholder="Enter amount in Rupees"

className="w-full p-2 border border-gray-300 rounded-md shadow-sm focus:ring-blue-500 focus:border-blue-500"

/>

</div>

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

{euros > 0 && (

<div className="mt-4 p-4 bg-green-100 border border-green-400 text-green-800 rounded-lg">

<p className="text-xl font-semibold">

Converted Amount: €{euros.toFixed(2)}

</p>

</div>

)}

</form>

</Section>

);

}

// --- Main App Component ---

// This component assembles all the other components into a single page.

export default function App() {

return (

<div className="bg-gray-100 min-h-screen font-sans p-4 sm:p-8">

<div className="max-w-2xl mx-auto">

<h1 className="text-4xl font-extrabold text-center text-gray-900 mb-8">

React Event Examples App

</h1>

<Counter />

<ArgumentButton />

<SyntheticEventButton />

<CurrencyConverter />

<footer className="text-center mt-8 text-gray-500">

<p>Event Handling Assignment</p>

</footer>

</div>

</div>

);

}



