Hands on 11

Solution:

**1. Create a New React App**

npx create-react-app eventexamplesapp

cd eventexamplesapp

**2. Edit src/App.js**

Replace the contents of src/App.js with the following code:

import React, { useState } from 'react';

import './App.css';

// CurrencyConverter Component

function CurrencyConverter() {

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

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

  const handleSubmit = (e) => {

    e.preventDefault();

    if (amount && currency) {

      // Convert Indian Rupees to Euro (using a simple conversion rate)

      const rupees = parseFloat(amount);

      const euroAmount = rupees \* 0.011; // Approximate conversion rate

      alert(`Converting to ${currency} Amount is ${euroAmount.toFixed(2)}`);

    } else {

      alert('Please enter both amount and currency');

    }

  };

  return (

    <div className="currency-converter">

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

      <form onSubmit={handleSubmit}>

        <div>

          <label>Amount:</label>

          <input

            type="number"

            value={amount}

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

            placeholder="Enter amount in Rupees"

          />

        </div>

        <div>

          <label>Currency:</label>

          <input

            type="text"

            value={currency}

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

            placeholder="Enter currency (e.g., Euro)"

          />

        </div>

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

      </form>

    </div>

  );

}

function App() {

  const [counter, setCounter] = useState(0);

  // Increment function that does multiple things

  const handleIncrement = () => {

    setCounter(prevCounter => prevCounter + 1);

    alert('Hello! Member1');

  };

  // Decrement function

  const handleDecrement = () => {

    setCounter(prevCounter => prevCounter - 1);

  };

  // Say welcome function that takes an argument

  const sayWelcome = (message) => {

    alert(message);

  };

  // Synthetic event handler

  const handleClick = (e) => {

    alert('I was clicked');

  };

  return (

    <div className="App">

      <div className="event-examples">

        <h1>Event Handling Examples</h1>

        {/\* Counter Display \*/}

        <div className="counter-display">

          <h2>{counter}</h2>

        </div>

        {/\* Event Buttons \*/}

        <div className="button-container">

          <button onClick={handleIncrement}>

            Increment

          </button>

          <button onClick={handleDecrement}>

            Decrement

          </button>

          <button onClick={() => sayWelcome('welcome')}>

            Say welcome

          </button>

          <button onClick={handleClick}>

            Click on me

          </button>

        </div>

      </div>

      {/\* Currency Converter Component \*/}

      <CurrencyConverter />

    </div>

  );

}

export default App;

**3. Edit src/App.css**

Replace the contents of src/App.css with the following code for better styling:

.App {

  text-align: center;

  padding: 20px;

  font-family: Arial, sans-serif;

}

.event-examples {

  margin-bottom: 40px;

}

.event-examples h1 {

  color: #333;

  margin-bottom: 30px;

}

.counter-display {

  margin: 20px 0;

}

.counter-display h2 {

  font-size: 48px;

  color: #007bff;

  margin: 0;

  font-weight: bold;

}

.button-container {

  display: flex;

  flex-direction: column;

  align-items: center;

  gap: 10px;

  margin: 20px 0;

}

.button-container button {

  padding: 12px 24px;

  font-size: 16px;

  background-color: #007bff;

  color: white;

  border: none;

  border-radius: 5px;

  cursor: pointer;

  transition: background-color 0.3s;

  min-width: 150px;

}

.button-container button:hover {

  background-color: #0056b3;

}

.currency-converter {

  background-color: #f8f9fa;

  padding: 30px;

  border-radius: 10px;

  max-width: 400px;

  margin: 0 auto;

  box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);

}

.currency-converter h2 {

  margin-bottom: 20px;

  font-size: 24px;

}

.currency-converter form {

  display: flex;

  flex-direction: column;

  gap: 15px;

}

.currency-converter form div {

  display: flex;

  flex-direction: column;

  align-items: flex-start;

}

.currency-converter label {

  font-weight: bold;

  margin-bottom: 5px;

  color: #333;

}

.currency-converter input {

  width: 100%;

  padding: 10px;

  border: 1px solid #ddd;

  border-radius: 5px;

  font-size: 14px;

}

.currency-converter input:focus {

  outline: none;

  border-color: #007bff;

  box-shadow: 0 0 5px rgba(0, 123, 255, 0.3);

}

.currency-converter button {

  padding: 12px 24px;

  font-size: 16px;

  background-color: #28a745;

  color: white;

  border: none;

  border-radius: 5px;

  cursor: pointer;

  transition: background-color 0.3s;

  margin-top: 10px;

}

.currency-converter button:hover {

  background-color: #218838;

}

**4. Run the App**

In your terminal, run:

npm start

Open http://localhost:3000 in your browser.

Output:

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.