**Counter.js:**

import React, { Component } from "react";

class Counter extends Component {

    constructor(props) {

        super(props);

        this.state = {

            count: 0,

        };

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

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

        this.handleIncrease = this.handleIncrease.bind(this);

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

    }

    increment() {

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

    }

    decrement = () => {

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

    };

    sayHello() {

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

    }

    handleIncrease() {

        this.increment();

        this.sayHello();

    }

    sayWelcome(message) {

        alert(`Message: ${message}`);

    }

    handleClick = (e) => {

        alert("I was clicked (Synthetic Event Triggered)");

        console.log("Synthetic Event object:", e);

    };

    render() {

        return (

            <div>

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

                <button onClick={this.handleIncrease}>Increase</button>

                <button onClick={this.decrement}>Decrease</button>

                <br /><br />

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

                <br /><br />

                <button onClick={this.handleClick}>OnPress</button>

            </div>

        );

    }

}

export default Counter;

**CurrencyConverter.js:**

import React, { useState } from "react";

function CurrencyConvertor() {

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

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

    const [converted, setConverted] = useState(null);

    const [error, setError] = useState("");

    const handleSubmit = (e) => {

        e.preventDefault();

        setConverted(null);

        setError("");

        if (currency.trim().toLowerCase() === "euro") {

            const rate = 0.011;

            const result = parseFloat(amount) \* rate;

            setConverted(result.toFixed(2));

        } else {

            setError("Only 'euro' conversion is supported.");

        }

    };

    return (

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

            <h2>Currency Convertor (INR → EURO)</h2>

            <form onSubmit={handleSubmit}>

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

                    <label>

                        Amount :

                    </label>

                    <input

                        type="number"

                        value={amount}

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

                        required

                    />

                </div>

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

                    <label>

                        Currency:

                    </label>

                    <input

                        type="text"

                        value={currency}

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

                        required

                    />

                </div>

                <div style={{ margin: "15px 0" }}>

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

                </div>

            </form>

            {converted && (

                <p>

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

                </p>

            )}

            {error && <p style={{ color: "red" }}>{error}</p>}

        </div>

    );

}

export default CurrencyConvertor;

**App.js:**

import React from "react";

import Counter from "./Components/Counter";

import CurrencyConvertor from "./Components/CurrencyConverter";

function App() {

    return (

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

            <h1>Event Examples App</h1>

            <hr />

            <Counter />

            <hr />

            <CurrencyConvertor />

        </div>

    );

}

export default App;

**Output:**







