**11. ReactJS-HOL**

**App.js:**

import React from 'react';

import './App.css';

import CurrencyConvertor from './CurrencyConvertor';

class App extends React.Component {

  constructor(props) {

    super(props);

    this.state = {

      counter: 0,

      message: ''

    };

  }

  increment = () => {

    this.setState({ counter: this.state.counter + 1 });

  };

  handleIncrement = () => {

    this.increment();

  };

  decrement = () => {

    this.setState({ counter: this.state.counter - 1 });

  };

  sayWelcome = (msg) => {

    alert(msg);

  };

  handleClick = (e) => {

    alert('I was clicked');

  };

  render() {

    return (

      <div className="App">

        <h2>{this.state.counter}</h2>

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

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

        <br />

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

        <br />

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

        <br /><br />

        <CurrencyConvertor />

      </div>

    );

  }

}

export default App;

**CurrencyConverter.js:**

import React, { Component } from 'react';

class CurrencyConvertor extends Component {

  constructor(props) {

    super(props);

    this.state = {

      amount: '',

      type: ''

    };

  }

  handleAmountChange = (e) => {

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

  };

  handleTypeChange = (e) => {

    this.setState({ type: e.target.value.toLowerCase() });

  };

  handleSubmit = (e) => {

    e.preventDefault();

    const { amount, type } = this.state;

    const numAmount = parseFloat(amount);

    if (!numAmount || (type !== 'euro' && type !== 'rupee')) {

      alert('Please enter a valid amount and type (euro or rupee)');

      return;

    }

    let result;

    if (type === 'euro') {

      result = numAmount \* 90;

      alert(`€${numAmount} is ₹${result.toFixed(2)}`);

    } else {

      result = numAmount / 90;

      alert(`₹${numAmount} is €${result.toFixed(2)}`);

    }

  };

  render() {

    return (

      <div>

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

        <form onSubmit={this.handleSubmit}>

          <div>

            <label>Amount : </label>

            <input type="number" value={this.state.amount} onChange={this.handleAmountChange} />

          </div>

          <div>

            <label>Currency : </label>

            <input type="text" value={this.state.type} onChange={this.handleTypeChange} />

          </div>

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

        </form>

      </div>

    );

  }

}

export default CurrencyConvertor;

**OUTPUT:**

