

## ReactJs Handson 11:

App.js:

```
// src/App.js

import React, { useState } from 'react';
import CurrencyConvertor from './CurrencyConvertor';
import './App.css';
function App() {
  const [count, setCount] = useState(0);

  const increment = () => {
    setCount(prev => prev + 1);
    sayHello();
    staticMessage();
  };

  const decrement = () => {
    setCount(prev => prev - 1);
  };

  const sayHello = () => {
    console.log("Hello!");
  };

  const staticMessage = () => {
    console.log("This is a static message.");
  };

  const sayWelcome = (msg) => {
    alert(msg);
  };

  const onPress = () => {
    alert("I was clicked");
  };

  return (
    <div className="App">
      <h1>React Event Handling Example</h1>

      <h2>Counter: {count}</h2>
      <button onClick={increment}>Increment</button>
      <button onClick={decrement}>Decrement</button>

      <hr />

      <button onClick={() => sayWelcome("Welcome!")}>Say Welcome</button>
    </div>
  );
}
```

```

    <hr />

    <button onClick={onPress}>OnPress</button>

    <hr />

    <CurrencyConvertor />
  </div>
);
}

export default App;

```

Currencyconverter.js:

```

// src/CurrencyConvertor.js

import './App.css';
import React, { useState } from 'react';

function CurrencyConvertor() {
  const [rupees, setRupees] = useState('');
  const [euros, setEuros] = useState(null);

  const handleChange = (e) => {
    setRupees(e.target.value);
  };

  const handleSubmit = () => {
    alert("Converting currency...");
    if (rupees === '') {
      alert("Please enter a valid amount in INR.");
      return;
    }
    const converted = parseFloat(rupees) / 90;
    setEuros(converted.toFixed(2));
  };

  return (
    <div>
      <h2 className="currency">Currency Converter (INR → EUR)</h2>
      <input
        type="number"
        value={rupees}
        onChange={handleChange}
        placeholder="Enter INR"

```

```

    />
    <button onClick={handleSubmit}>Convert</button>
    {euros !== null && <h3>Euro: €{euros}</h3>}
  </div>
);
}

export default CurrencyConvertor;

export default CurrencyConvertor;

```

OUTPUT:

>>npm start



