**Week 7: ReactJS HOL**

**Task 1:** Create a React application named **cricketapp** that demonstrates ES6 features by completing the following tasks:

1. Create a component called ListofPlayers that:
   * Declares an array of 11 players with their names and scores.
   * Displays the list using the map() method.
   * Filters and displays players with scores below 70 using arrow functions.
2. Create another component called IndianPlayers that:
   * Uses destructuring to display odd and even team players.
   * Merges two arrays: T20Players and RanjiTrophyPlayers, using the spread operator, and displays the result.
3. Use a flag variable to conditionally render either ListofPlayers or IndianPlayers components on the homepage using a simple if-else.

**Code:  
ListofPlayers.js**// src/components/ListofPlayers.js

import React from 'react';

const ListofPlayers = ({ players }) => {

return (

<ul>

{players.map((item, index) => (

<li key={index}>

Mr. {item.name} <span>{item.score}</span>

</li>

))}

</ul>

);

};

export default ListofPlayers;  
  
**ScoreBelow70.js**

// src/components/ScoreBelow70.js

import React from 'react';

const ScoreBelow70 = ({ players }) => {

const players70 = players.filter(player => player.score <= 70);

return (

<ul>

{players70.map((item, index) => (

<li key={index}>

Mr. {item.name} <span>{item.score}</span>

</li>

))}

</ul>

);

};

export default ScoreBelow70;  
  
**OddPlayers.js**

// src/components/OddPlayers.js

import React from 'react';

export const OddPlayers = ({ first, third, fifth }) => {

return (

<div>

<li>First : {first}</li>

<li>Third : {third}</li>

<li>Fifth : {fifth}</li>

</div>

);

};  
  
**EvenPlayers.js**  
// src/components/EvenPlayers.js

import React from 'react';

export const EvenPlayers = ({ second, fourth, sixth }) => {

return (

<div>

<li>Second : {second}</li>

<li>Fourth : {fourth}</li>

<li>Sixth : {sixth}</li>

</div>

);

};  
  
**ListofIndianPlayers.js**  
// src/components/ListofIndianPlayers.js

import React from 'react';

const ListofIndianPlayers = ({ IndianPlayers }) => {

return (

<ul>

{IndianPlayers.map((player, index) => (

<li key={index}>Mr. {player}</li>

))}

</ul>

);

};

export default ListofIndianPlayers;

**App.js**

// src/App.js

import React from 'react';

import './App.css';

import ListofPlayers from './components/ListofPlayers';

import ScoreBelow70 from './components/ScoreBelow70';

import { OddPlayers } from './components/OddPlayers';

import { EvenPlayers } from './components/EvenPlayers';

import ListofIndianPlayers from './components/ListofIndianPlayers';

function App() {

const flag = true; // if changed to false then we can see other list

const players = [

{ name: 'Jack', score: 50 },

{ name: 'Michael', score: 70 },

{ name: 'John', score: 40 },

{ name: 'Ann', score: 61 },

{ name: 'Elisabeth', score: 61 },

{ name: 'Sachin', score: 95 },

{ name: 'Dhoni', score: 100 },

{ name: 'Virat', score: 84 },

{ name: 'Jadeja', score: 64 },

{ name: 'Raina', score: 75 },

{ name: 'Rohit', score: 80 },

];

const IndianTeam = ['Sachin1', 'Dhoni2', 'Virat3', 'Rohit4', 'Yuvaraj5', 'Raina6'];

const [first, second, third, fourth, fifth, sixth] = IndianTeam;

const T20Players = ['First Player', 'Second Player', 'Third Player'];

const RanjiTrophyPlayers = ['Fourth Player', 'Fifth Player', 'Sixth Player'];

const IndianPlayers = [...T20Players, ...RanjiTrophyPlayers];

return (

<div className="App">

{flag ? (

<div>

<h1>List of Players</h1>

<ListofPlayers players={players} />

<hr />

<h1>List of Players having Scores Less than 70</h1>

<ScoreBelow70 players={players} />

</div>

) : (

<div>

<h1>Indian Team</h1>

<h2>Odd Players</h2>

<OddPlayers first={first} third={third} fifth={fifth} />

<hr />

<h2>Even Players</h2>

<EvenPlayers second={second} fourth={fourth} sixth={sixth} />

<hr />

<h2>List of Indian Players Merged:</h2>

<ListofIndianPlayers IndianPlayers={IndianPlayers} />

</div>

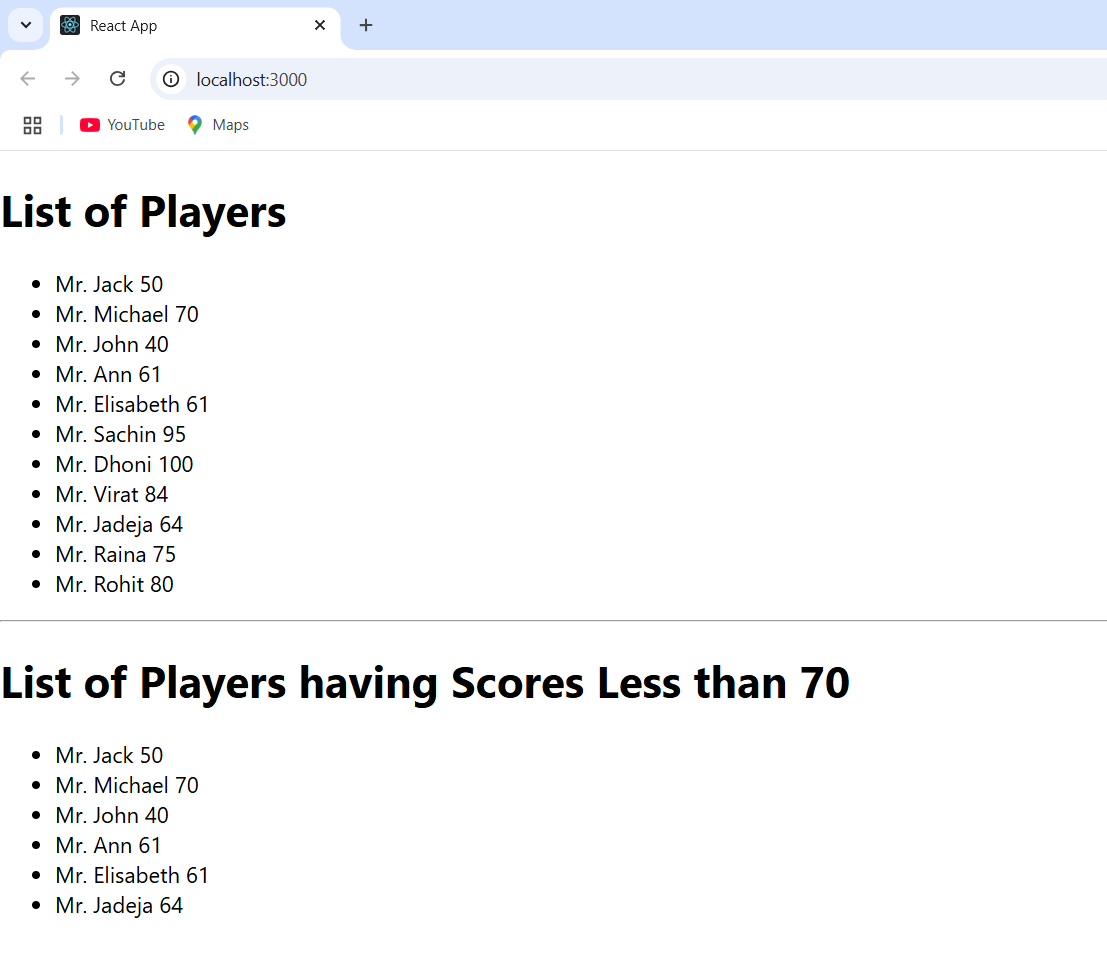
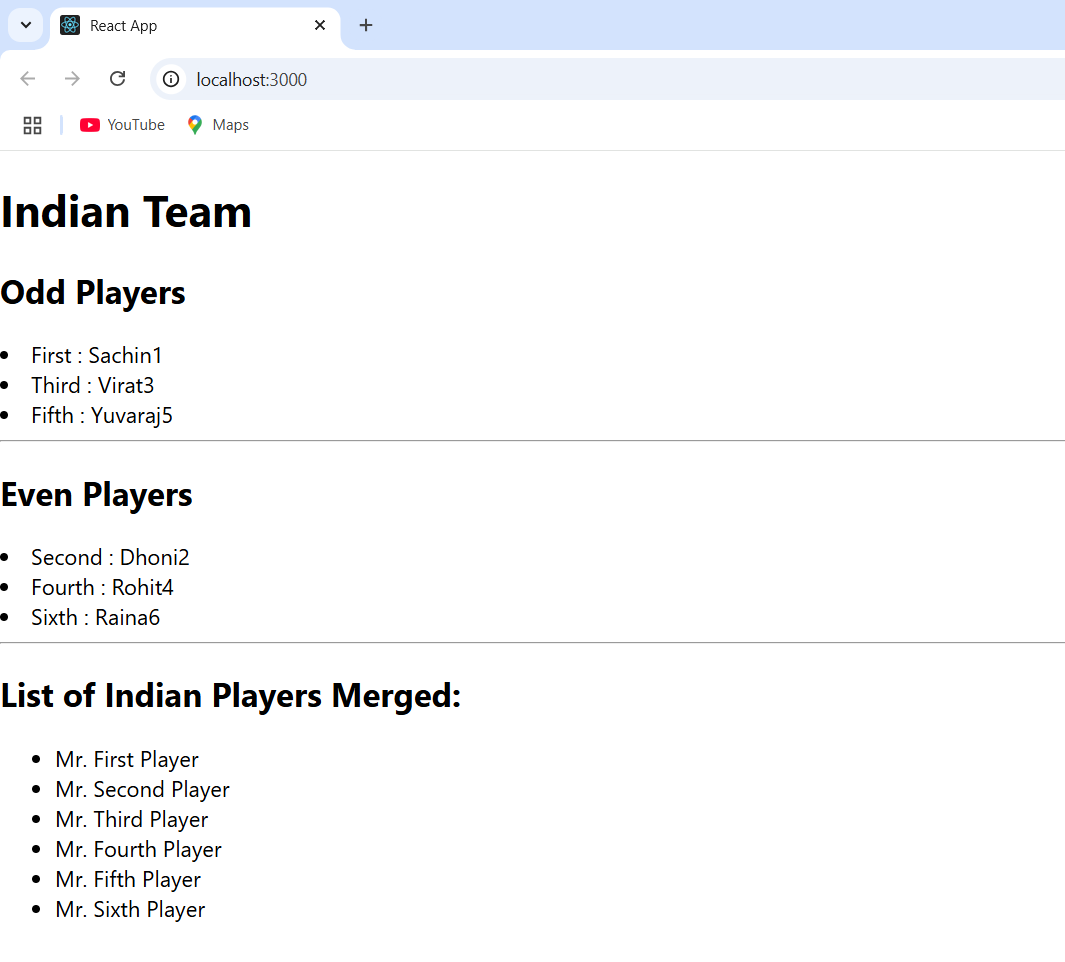
)}

</div>

);

}

export default App;

**Task 2:** Create a React Application named “officespacerentalapp” which uses React JSX to create elements, attributes and renders DOM to display the page. To apply Css, Display the color of the Rent in Red if it’s below 60000 and in Green if it’s above 60000.

**Code**

**App.css**  
.textRed {

color: red;

}

.textGreen {

color: green;

}

**App.js**  
// src/App.js

import './App.css';

function App() {

const heading = "Office Space";

const imageSrc = process.env.PUBLIC\_URL + "/office.webp";

const ItemName = { Name: "DBS", Rent: 50000, Address: "Chennai" };

const offices = [

{ Name: "DBS", Rent: 50000, Address: "Chennai" },

{ Name: "WeWork", Rent: 65000, Address: "Bangalore" },

{ Name: "Regus", Rent: 45000, Address: "Mumbai" },

{ Name: "SmartWorks", Rent: 75000, Address: "Hyderabad" },

];

const getRentColor = (rent) => {

return rent <= 60000 ? 'textRed' : 'textGreen';

};

return (

<div>

<h1>{heading}, at Affordable Range</h1>

<img src={imageSrc} width="25%" height="25%" alt="Office Space" />

<h2>Single Office Example:</h2>

<h3>Name: {ItemName.Name}</h3>

<h3 className={getRentColor(ItemName.Rent)}>Rent: Rs. {ItemName.Rent}</h3>

<h3>Address: {ItemName.Address}</h3>

<hr />

<h2>All Office Listings:</h2>

{offices.map((office, index) => (

<div key={index}>

<h3>Name: {office.Name}</h3>

<h3 className={getRentColor(office.Rent)}>Rent: Rs. {office.Rent}</h3>

<h3>Address: {office.Address}</h3>

<hr />

</div>

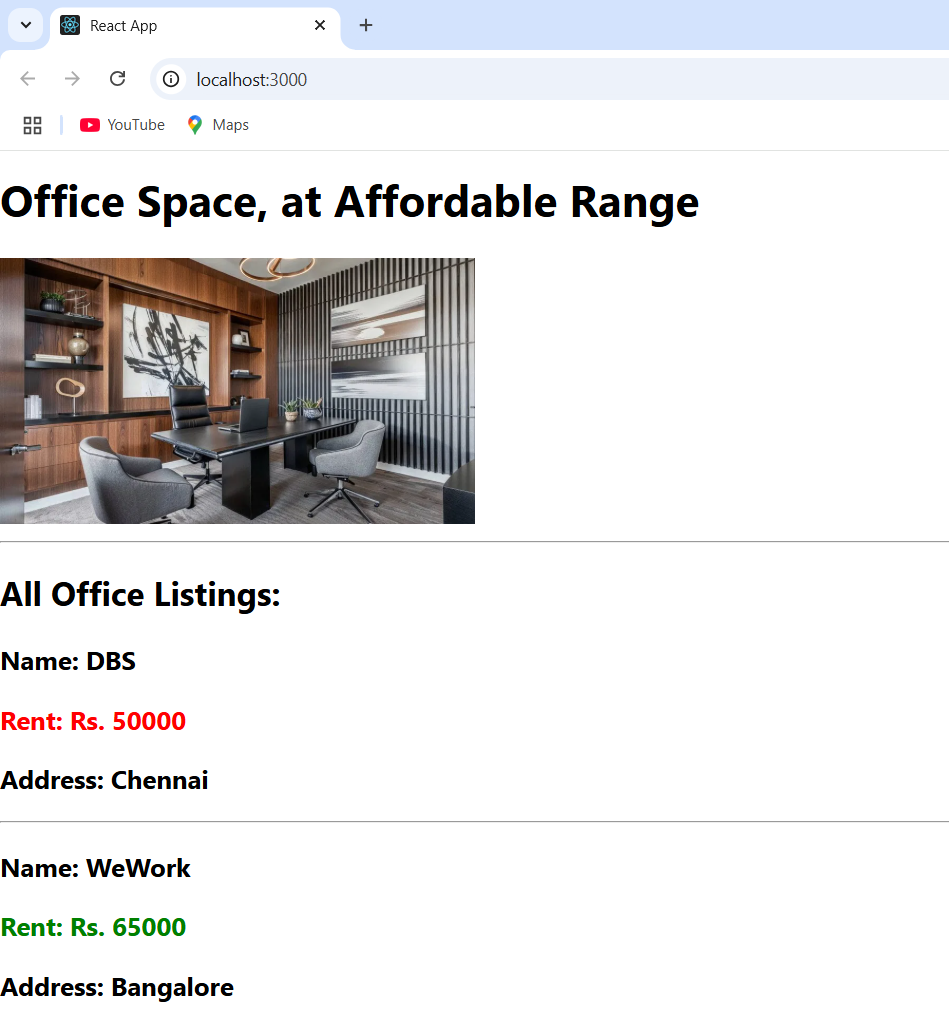
))}

</div>

);

}

export default App;

  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
**Task 3**: Create a React Application “eventexamplesapp” to handle various events of the form elements in HTML.

1. Create “Increment” button to increase the value of the counter and “Decrement” button to decrease the value of the counter. The “Increase” button should invoke multiple methods.
   1. To increment the value
   2. Say Hello followed by a static message.
2. Create a button “Say Welcome” which invokes the function which takes “welcome” as an argument.
3. Create a button which invokes synthetic event “OnPress” which display “I was clicked”

Create a “CurrencyConvertor” component which will convert the Indian Rupees to Euro when the Convert button is clicked.

Handle the Click event of the button to invoke the handleSubmit event and handle the conversion of the euro to rupees.

**Code  
  
App.js**

// src/App.js

import React from 'react';

import './App.css';

import CurrencyConvertor from './components/CurrencyConvertor';

class App extends React.Component {

  constructor(props) {

    super(props);

    this.state = {

      counter: 0,

      message: ''

    };

  }

  increment = () => {

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

  };

  sayHello = () => {

    alert("Hello! Have a great day ");

  };

  incrementAndGreet = () => {

    this.increment();

    this.sayHello();

  };

  decrement = () => {

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

  };

  sayWelcome = (msg) => {

    alert(`Message: ${msg}`);

  };

  handleSyntheticEvent = (e) => {

    e.preventDefault();

       alert('I was clicked');

  };

  render() {

    return (

      <div className="App">

        <h1>React Event Handling Example</h1>

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

        <button onClick={this.incrementAndGreet}>Increment</button>

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

        <hr />

        <button onClick={() => this.sayWelcome("Welcome to React Events!")}>

          Say Welcome

        </button>

        <hr />

        <button onClick={this.handleSyntheticEvent}>Synthetic Event (OnPress)</button>

        <p>{this.state.message}</p>

        <hr />

        <CurrencyConvertor />

      </div>

    );

  }

}

export default App;

**CurrencyConverter.js**

import React, { useState } from 'react';

const CurrencyConvertor = () => {

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

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

  const handleClick = (e) => {

    let convertedAmount = 0;

    if (currency.toLowerCase() === 'euro') {

      convertedAmount = amount \* 80;

    } else if (currency.toLowerCase() === 'usd') {

      convertedAmount = amount \* 74;

    } else {

      convertedAmount = amount;

    }

    alert(`Converting to ${currency} Amount is ${convertedAmount}`);

  };

  return (

    <div style={{ padding: '20px' }}>

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

      <form>

        <div>

          <label>Amount: </label>

          <input

            type="number"

            value={amount}

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

            required

          />

        </div>

        <br />

        <div>

          <label>Currency: </label>

          <input

            type="text"

            value={currency}

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

            placeholder="e.g. Euro or USD"

            required

          />

        </div>

        <br />

        <button type="submit" onClick={handleClick}>

          Submit

        </button>

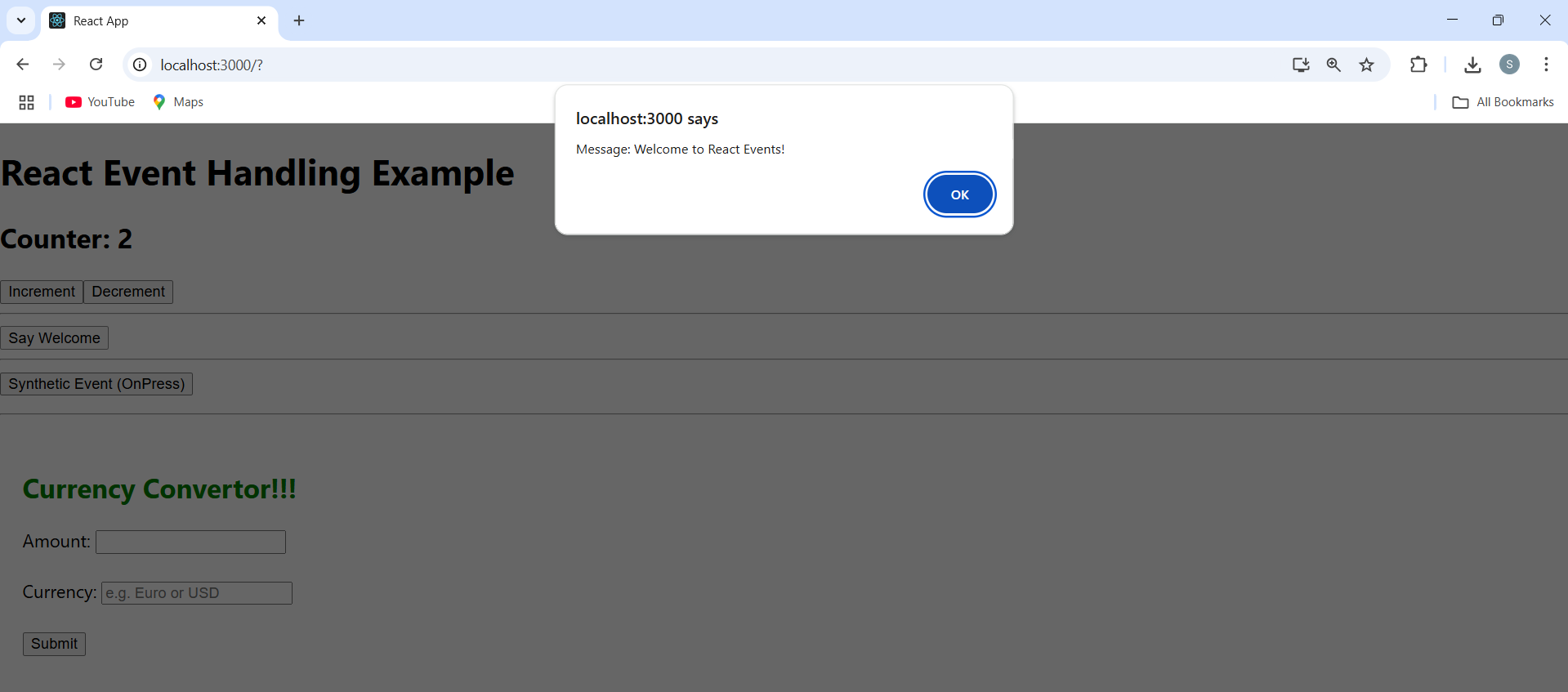
      </form>

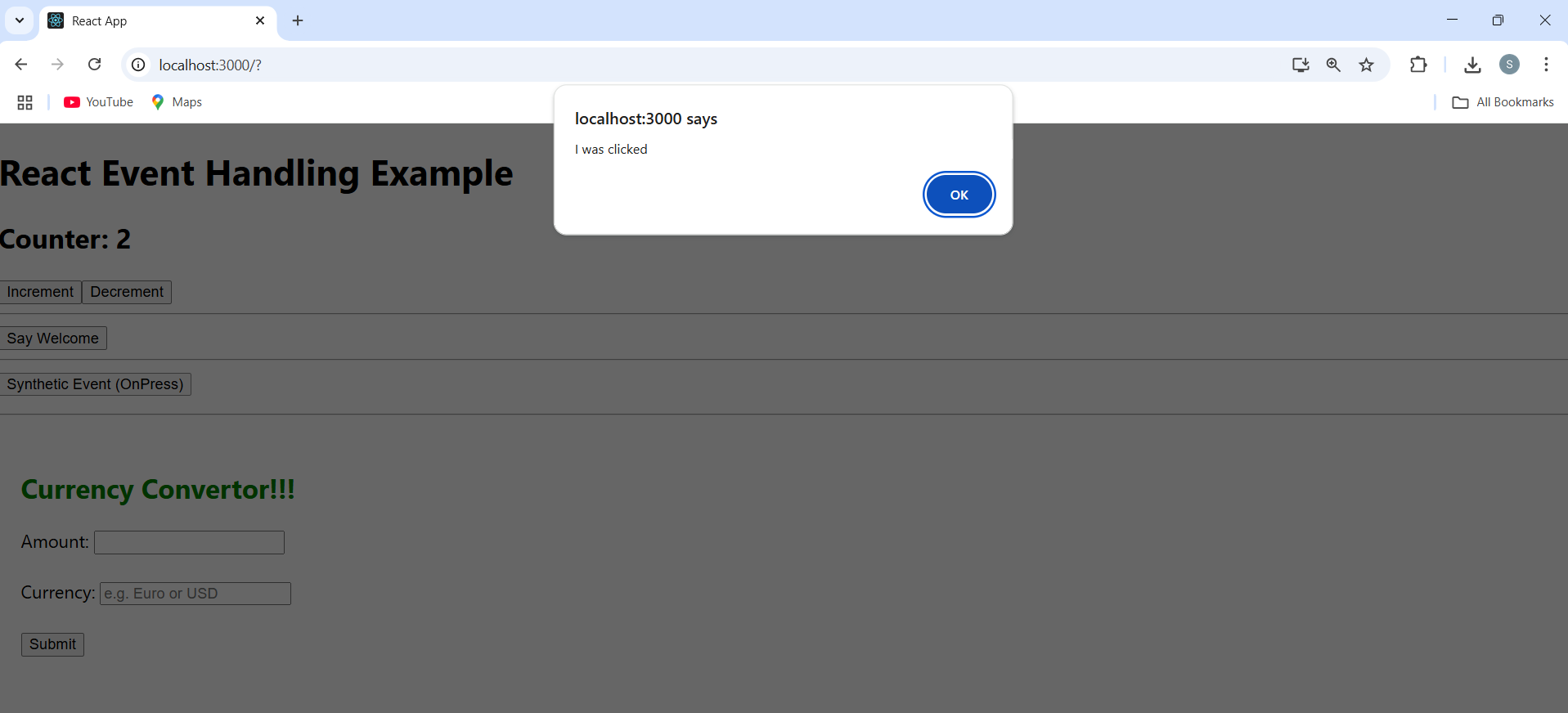
    </div>

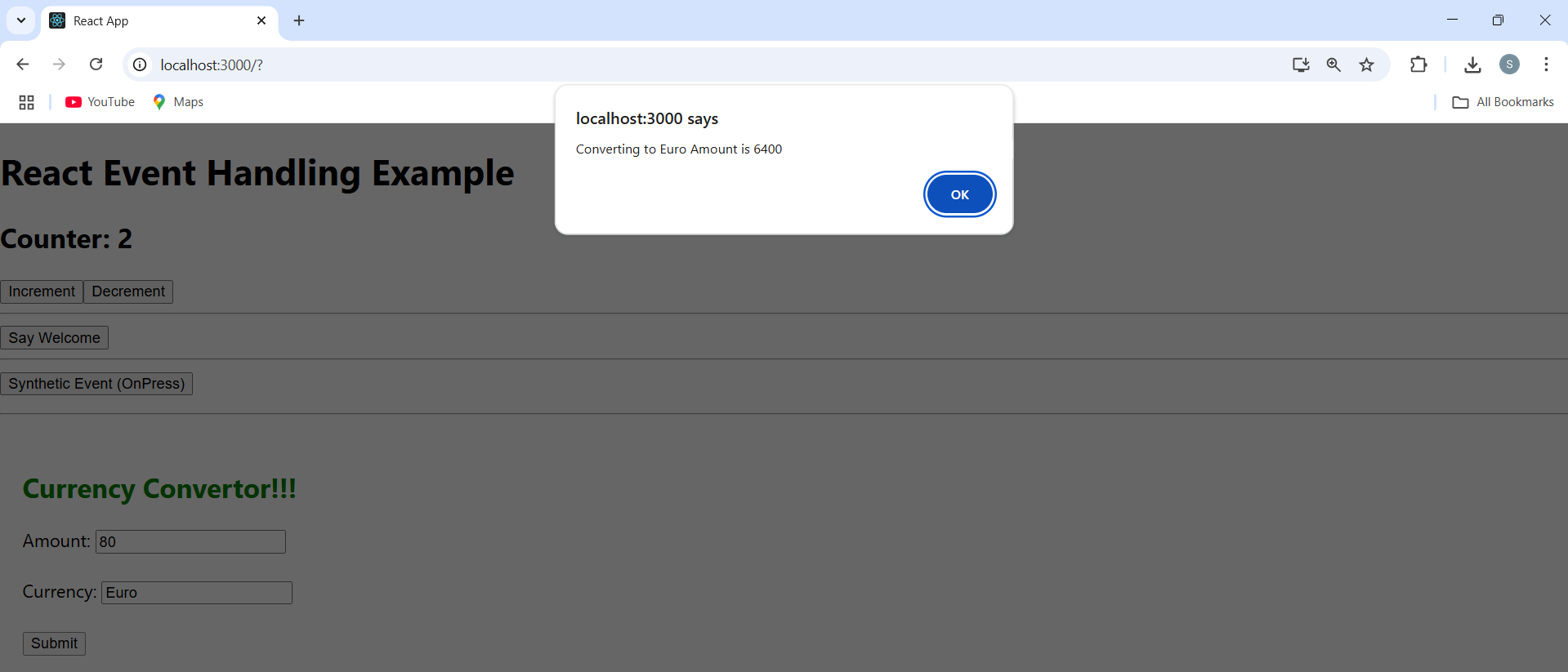
  );

};

export default CurrencyConvertor;







**Task 4:** Create a React Application named “ticketbookingapp” where the guest user can browse the page where the flight details are displayed whereas the logged in user only can book tickets.

The Login and Logout buttons should accordingly display different pages. Once the user is logged in the User page should be displayed. When the user clicks on Logout, the Guest page should be displayed.

**Code**

**GuestGreeting.js**

function GuestGreeting() {

return <h2>Please sign up.</h2>;

}

export default GuestGreeting;

**UserGreeting.js**

function UserGreeting() {

return <h2>Welcome back!</h2>;

}

export default UserGreeting;

**LoginButton.js**

function LoginButton(props) {

return <button onClick={props.onClick}>Login</button>;

}

export default LoginButton;

**LogoutButton.js**

function LogoutButton(props) {

return <button onClick={props.onClick}>Logout</button>;

}

export default LogoutButton;

**Greeting.js**

import GuestGreeting from './GuestGreeting';

import UserGreeting from './UserGreeting';

function Greeting({ isLoggedIn }) {

return isLoggedIn ? <UserGreeting /> : <GuestGreeting />;

}

export default Greeting;

**App.js**

import { useState } from 'react';

import Greeting from './components/Greeting';

import LoginButton from './components/LoginButton';

import LogoutButton from './components/LogoutButton';

function App() {

  const [isLoggedIn, setIsLoggedIn] = useState(false);

  const handleLogin = () => setIsLoggedIn(true);

  const handleLogout = () => setIsLoggedIn(false);

  return (

    <div style={{ padding: '2rem', textAlign: 'center' }}>

      <Greeting isLoggedIn={isLoggedIn} />

      {

        isLoggedIn

          ? <LogoutButton onClick={handleLogout} />

          : <LoginButton onClick={handleLogin} />

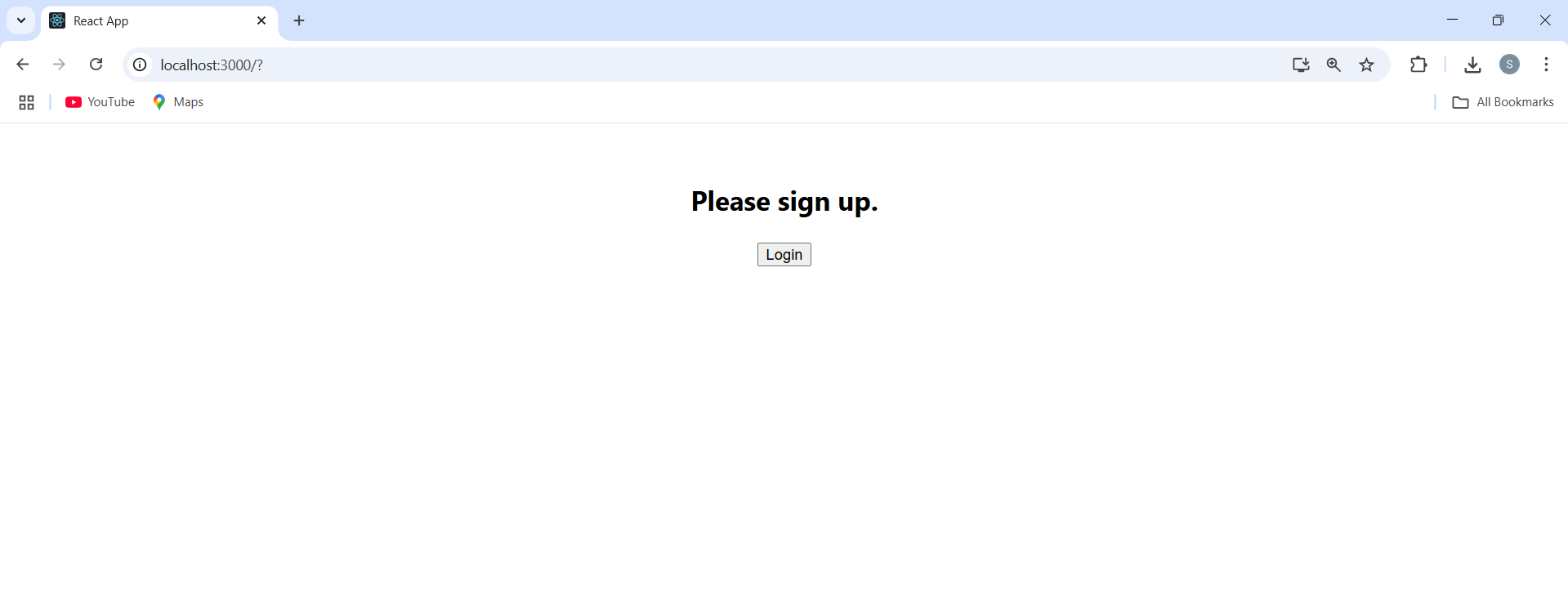
      }

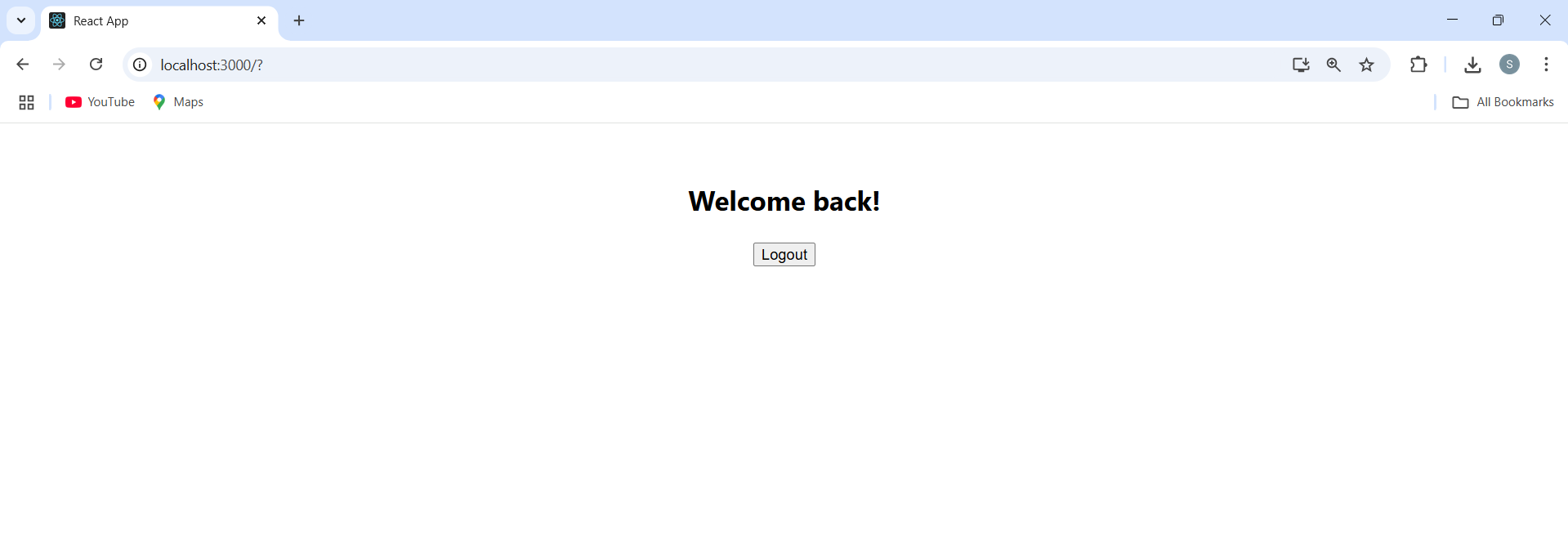
    </div>

  );

}

export default App;



  
**Task 5:** Create a React App named “bloggerapp” in with 3 components.

1. Book Details
2. Blog Details
3. Course Details

Implement this with as many ways possible of Conditional Rendering.

**Code:**  
**BookDetails.js**src/components/BookDetails.js  
const BookDetails = ({ books }) => {

return (

<div>

<h2>Book Details</h2>

{books.map(book => (

<div key={book.id}>

<h3>{book.bname}</h3>

<h4>{book.price}</h4>

</div>

))}

</div>

);

};

export default BookDetails;  
  
**BlogDetails.js**src/components/BlogDetails.js  
const BlogDetails = ({ blogs }) => {

return (

<div>

<h2>Blog Details</h2>

{blogs.map(blog => (

<div key={blog.id}>

<h3>{blog.title}</h3>

<p>{blog.desc}</p>

</div>

))}

</div>

);

};

export default BlogDetails;

**CourseDetails.js**

src/components/CourseDetails.js

const CourseDetails = ({ courses }) => {

return (

<div>

<h2>Course Details</h2>

{courses.map(course => (

<div key={course.id}>

<h3>{course.name}</h3>

<p>{course.duration}</p>

</div>

))}

</div>

);

};

export default CourseDetails;

**App.js**

import { useState } from "react";

import BookDetails from "./components/BookDetails";

import BlogDetails from "./components/BlogDetails";

import CourseDetails from "./components/CourseDetails";

function App() {

const [view, setView] = useState("books");

const books = [

{ id: 1, bname: "React in Action", price: "$30" },

{ id: 2, bname: "Learning JavaScript", price: "$25" },

];

const blogs = [

{ id: 1, title: "React Hooks", desc: "Intro to Hooks." },

{ id: 2, title: "JSX Tips", desc: "Best practices." },

];

const courses = [

{ id: 1, name: "Frontend Dev", duration: "3 months" },

{ id: 2, name: "Backend Dev", duration: "4 months" },

];

return (

<div className="App">

<h1>Blogger App</h1>

<button onClick={() => setView("books")}>Show Books</button>

<button onClick={() => setView("blogs")}>Show Blogs</button>

<button onClick={() => setView("courses")}>Show Courses</button>

{/\* Different Ways of Conditional Rendering \*/}

{/\* 1. Using If-Else \*/}

{(() => {

if (view === "books") return <BookDetails books={books} />;

else if (view === "blogs") return <BlogDetails blogs={blogs} />;

else return <CourseDetails courses={courses} />;

})()}

{/\* 2. Ternary Operator \*/}

{/\* {view === "books" ? <BookDetails books={books} /> : view === "blogs" ? <BlogDetails blogs={blogs} /> : <CourseDetails courses={courses} />} \*/}

{/\* 3. Logical AND (&&) \*/}

{/\* {view === "books" && <BookDetails books={books} />}

{view === "blogs" && <BlogDetails blogs={blogs} />}

{view === "courses" && <CourseDetails courses={courses} />} \*/}

</div>

);

}

export default App;

