Create a React Application named “cricketapp” with the following components:

**1. ListofPlayers**

Declare an array with 11 players and store details of their names and scores using the map feature of ES6.

Filter the players with scores below 70 using arrow functions of ES6.

**2. IndianPlayers**

Display the Odd Team Player and Even Team players using the Destructuring features of ES6.

Declare two arrays T20players and RanjiTrophy players and merge the two arrays and display them using the Merge feature of ES6.

**Display these two components on the same home page using a simple if-else in the flag variable.**

Absolutely! Here is the **complete, clean, and modular code** for the first question:  
**React Application "cricketapp" with ListofPlayers and IndianPlayers, displayed by a flag toggle, all code split into professional files.**

## src/components/ListOfPlayers.js

## import React from "react";

## const playersData = [

## { name: "Virat Kohli", score: 85 },

## { name: "Rohit Sharma", score: 65 },

## { name: "Jasprit Bumrah", score: 55 },

## { name: "Shubman Gill", score: 75 },

## { name: "Rishabh Pant", score: 45 },

## { name: "KL Rahul", score: 90 },

## { name: "Hardik Pandya", score: 60 },

## { name: "Cheteshwar Pujara", score: 80 },

## { name: "Ravindra Jadeja", score: 50 },

## { name: "Ishant Sharma", score: 30 },

## { name: "Kedar Jadhav", score: 70 },

## ];

## const ListOfPlayers = () => {

## // Show all players

## const allPlayers = playersData.map(({ name, score }, idx) => (

## <li key={idx}>{name} - Score: {score}</li>

## ));

## // Filter players with score below 70

## const below70 = playersData.filter(player => player.score < 70);

## return (

## <section>

## <h2>All Players</h2>

## <ul>{allPlayers}</ul>

## <h2>Players with Score Below 70</h2>

## <ul>

## {below70.map((player, idx) => (

## <li key={idx}>{player.name} - Score: {player.score}</li>

## ))}

## </ul>

## </section>

## );

## };

## export default ListOfPlayers;

## src/components/IndianPlayers.js

import React from "react";

const players = [

"Virat Kohli",

"Rohit Sharma",

"Jasprit Bumrah",

"Shubman Gill",

"Rishabh Pant",

"KL Rahul",

];

// Even: 0,2,4 … Odd: 1,3,5 …

const evenTeam = players.filter((\_, idx) => idx % 2 === 0);

const oddTeam = players.filter((\_, idx) => idx % 2 !== 0);

const T20Players = ["Hardik Pandya", "Ravindra Jadeja"];

const RanjiTrophyPlayers = ["Cheteshwar Pujara", "Mayank Agarwal"];

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

const IndianPlayers = () => (

<section>

<h2>Odd Team Players</h2>

<ul>

{oddTeam.map((name, idx) => <li key={idx}>{name}</li>)}

</ul>

<h2>Even Team Players</h2>

<ul>

{evenTeam.map((name, idx) => <li key={idx}>{name}</li>)}

</ul>

<h2>Merged Players (T20 + Ranji Trophy)</h2>

<ul>

{mergedPlayers.map((name, idx) => <li key={idx}>{name}</li>)}

</ul>

</section>

);

export default IndianPlayers;

## src/App.js

## import React from "react";

## import ReactDOM from "react-dom/client";

## import "./index.css";

## import App from "./App";

## const root = ReactDOM.createRoot(document.getElementById("root"));

## root.render(

## <React.StrictMode>

## <App />

## </React.StrictMode>

## );

## src/index.js

## import React from "react";

## import ReactDOM from "react-dom/client";

## import "./index.css";

## import App from "./App";

## const root = ReactDOM.createRoot(document.getElementById("root"));

## root.render(

## <React.StrictMode>

## <App />

## </React.StrictMode>

## );

## src/index.css

body {

font-family: Arial, sans-serif;

background: #f3f7f9;

color: #232323;

margin: 0;

padding: 0;

}

.main-container {

max-width: 600px;

margin: 36px auto;

padding: 24px;

background: white;

border-radius: 10px;

box-shadow: 0 2px 16px #8fbcf1aa;

}

h1 {

text-align: center;

color: #2d3d6d;

}

button.toggle-btn {

background: #2d3d6d;

color: #fff;

border: none;

margin: 20px 0;

padding: 0.7em 1.4em;

border-radius: 5px;

font-size: 1rem;

cursor: pointer;

}

button.toggle-btn:hover {

background: #1944a3;

}

ul {

list-style-type: none;

padding-left: 0;

}

li {

background: #eaf1fb;

margin: 6px 0;

padding: 8px 12px;

border-radius: 4px;

}

h2 {

color: #247769;

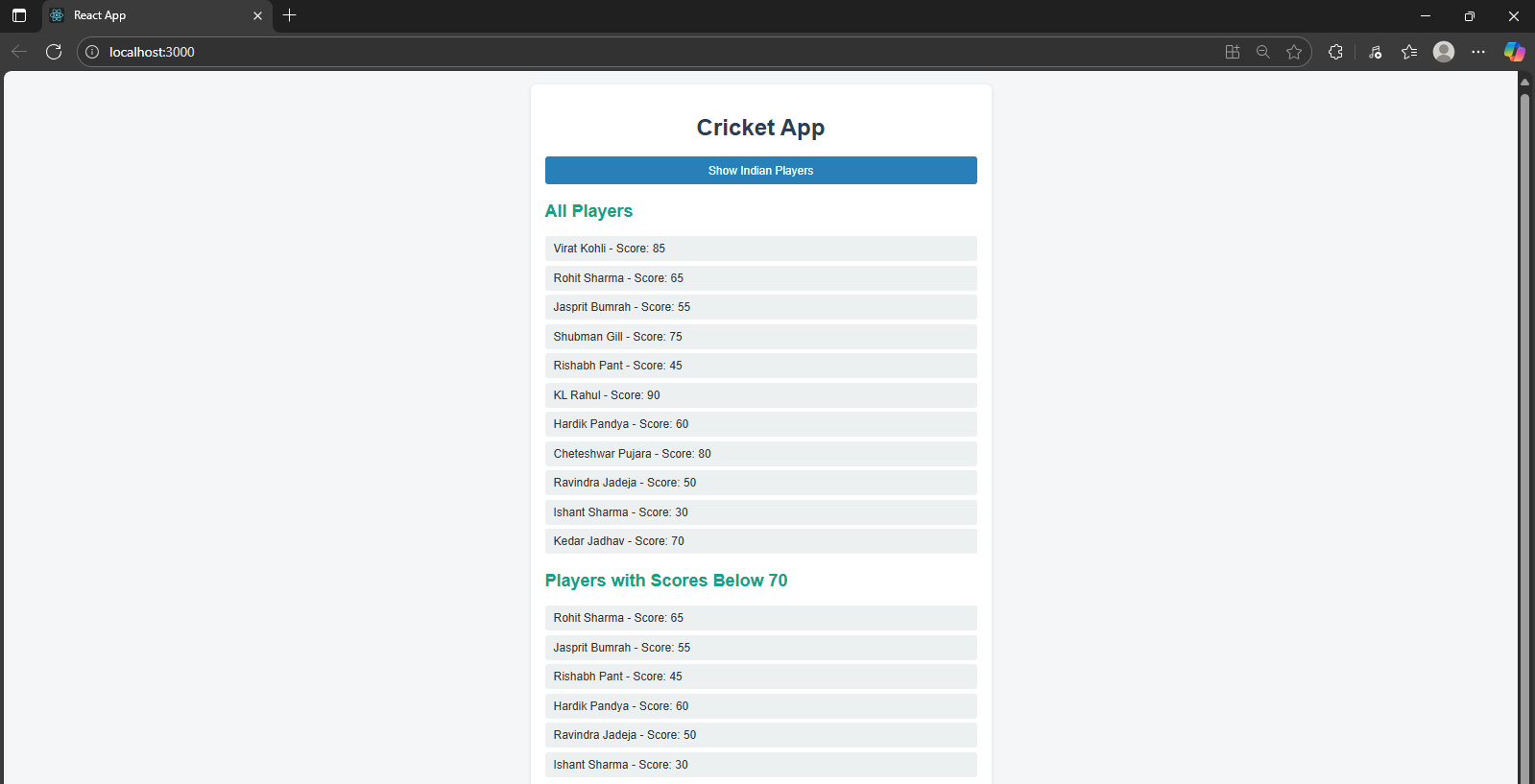
font-size: 1.2rem;

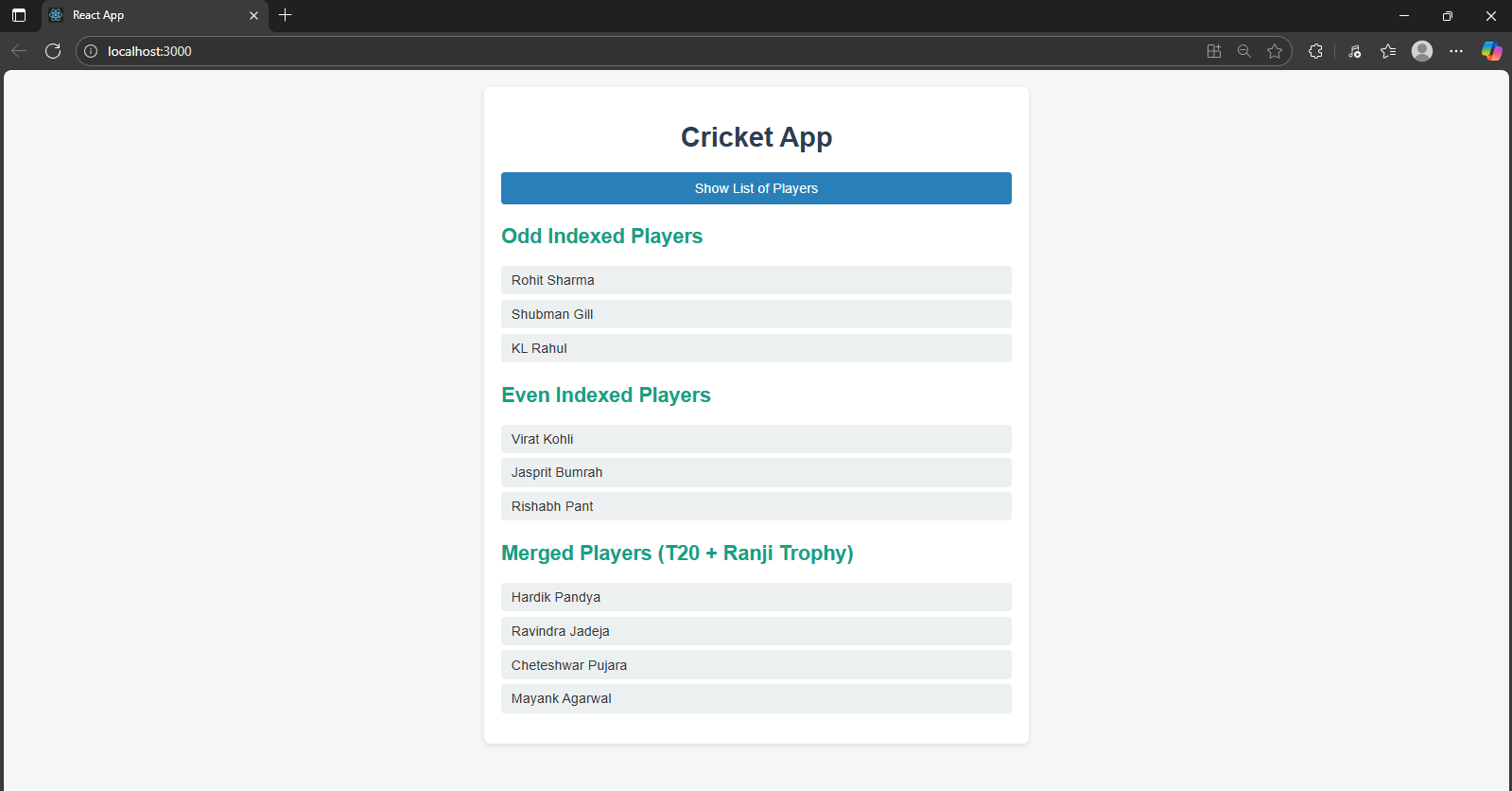
margin-top: 1.5em;

}

🟢

OUTPUT





1. Create a React Application named “officespacerentalapp” which uses React JSX to create elements, attributes and renders DOM to display the page.

Create an element to display the heading of the page.

Attribute to display the image of the office space

Create an object of office to display the details like Name, Rent and Address.

Create a list of Object and loop through the office space item to display more data.

To apply Css, Display the color of the Rent in Red if it’s below 60000 and in Green if it’s above 60000.

## Cmponents/OfficeList.js

## import React, { useState } from "react";

## import "./OfficeList.css";

## // Single featured office object

## const singleOffice = {

## name: "Prestige Tech Park",

## rent: 75\_000,

## address: "Whitefield, Bangalore",

## };

## // List of office objects

## const offices = [

## {

## id: 1,

## name: "Prestige Tech Park",

## rent: 75\_000,

## address: "Whitefield, Bangalore",

## image: "/image.jpg",

## },

## {

## id: 2,

## name: "DLF Cyber City",

## rent: 58\_000,

## address: "Gurugram, Haryana",

## image: "/image1.jpg",

## },

## {

## id: 3,

## name: "MindSpace",

## rent: 65\_000,

## address: "HiTech City, Hyderabad",

## image: "/image2.jpg",

## },

## ];

## const OfficeList = () => {

## const [search, setSearch] = useState("");

## // Filter offices by search text (in name or address)

## const filteredOffices = offices.filter(

## (office) =>

## office.name.toLowerCase().includes(search.toLowerCase()) ||

## office.address.toLowerCase().includes(search.toLowerCase())

## );

## return (

## <section className="section">

## <h2>Featured Office</h2>

## <div className="office-card">

## <img src="/image.jpg" alt={singleOffice.name} className="office-img" />

## <div>

## <p>

## <strong>Name:</strong> {singleOffice.name}

## </p>

## <p>

## <strong>Rent:</strong>{" "}

## <span

## style={{

## color: singleOffice.rent < 60\_000 ? "red" : "green",

## fontWeight: 600,

## }}

## >

## ₹{singleOffice.rent}

## </span>

## </p>

## <p>

## <strong>Address:</strong> {singleOffice.address}

## </p>

## </div>

## </div>

## {/\* Search Bar \*/}

## <div className="search-bar-wrapper">

## <input

## type="text"

## placeholder="Search by office name or city..."

## value={search}

## className="search-bar"

## onChange={(e) => setSearch(e.target.value)}

## />

## </div>

## <h2>Other Office Spaces</h2>

## <div className="office-list">

## {filteredOffices.length > 0 ? (

## filteredOffices.map((office) => (

## <div className="office-card" key={office.id}>

## <img src={office.image} alt={office.name} className="office-img" />

## <div>

## <p>

## <strong>Name:</strong> {office.name}

## </p>

## <p>

## <strong>Rent:</strong>{" "}

## <span

## style={{

## color: office.rent < 60\_000 ? "red" : "green",

## fontWeight: 600,

## }}

## >

## ₹{office.rent}

## </span>

## </p>

## <p>

## <strong>Address:</strong> {office.address}

## </p>

## </div>

## </div>

## ))

## ) : (

## <div style={{ marginLeft: "16px", marginTop: "12px" }}>

## <strong>No office spaces found.</strong>

## </div>

## )}

## </div>

## </section>

## );

## };

## export default OfficeList;

## src/App.js

## import React from "react";

## import OfficeList from "./components/OfficeList";

## const App = () => (

## <div className="main-container">

## <header>

## <h1>Office Space Rental App</h1>

## </header>

## <OfficeList />

## </div>

## );

## export default App;

## src/components/OfficeList.css

## .section {

## margin: 32px auto;

## max-width: 1200px;

## padding: 10px;

## background: #fff;

## border-radius: 16px;

## box-shadow: 0 6px 32px rgba(32,80,114,0.08);

## }

## h2 {

## color: #146b67;

## font-size: 1.7rem;

## font-weight: 700;

## margin-bottom: 22px;

## margin-left: 6px;

## }

## .office-list {

## display: flex;

## flex-wrap: wrap;

## gap: 32px;

## justify-content: center;

## align-items: flex-start;

## }

## .office-card {

## background: linear-gradient(100deg, #f2f4f8 80%, #d1e0ef 100%);

## border-radius: 12px;

## box-shadow: 0 4px 18px #aae0e6a0;

## padding: 22px 20px 18px 20px;

## transition: transform 0.16s, box-shadow 0.18s;

## width: 298px;

## margin-bottom: 20px;

## display: flex;

## gap: 18px;

## align-items: flex-start;

## border: 1.5px solid #e1e9f1;

## position: relative;

## }

## .office-card:hover {

## transform: translateY(-6px) scale(1.02);

## box-shadow: 0 16px 40px #20507233;

## border-color: #87d1e8;

## }

## .office-img {

## width: 90px;

## height: 90px;

## object-fit: cover;

## border-radius: 8px;

## border: 2px solid #7fccda;

## box-shadow: 0 2px 10px #cce8ed77;

## }

## .office-card div {

## flex: 1;

## }

## .office-card p {

## font-size: 1.08rem;

## margin-bottom: 6.5px;

## }

## .office-card strong {

## color: #146b67;

## font-weight: 700;

## margin-right: 4px;

## }

## .office-card span[style\*="color:green"] {

## color: #20946b !important;

## }

## .office-card span[style\*="color:red"] {

## color: #dc3939 !important;

## }

## .search-bar-wrapper {

## margin: 16px 0 30px 0;

## text-align: right;

## }

## .search-bar {

## font-size: 1.05rem;

## padding: 9px 13px;

## border: 1.5px solid #b8d3e5;

## border-radius: 7px;

## background: #f5fafd;

## width: 290px;

## transition: border .18s, box-shadow .18s;

## }

## .search-bar:focus {

## outline: none;

## border-color: #146b67;

## box-shadow: 0 0 6px #80c8e2ac;

## }

## @media (max-width: 900px) {

## .office-list {

## flex-direction: column;

## gap: 0;

## align-items: center;

## }

## .office-card {

## width: calc(100vw - 40px);

## margin-bottom: 24px;

## min-width: 210px;

## max-width: 98vw;

## flex-wrap: wrap;

## }

## .office-img {

## width: 70px;

## height: 70px;

## }

## }

## @media (max-width: 600px) {

## .office-card {

## flex-direction: column;

## align-items: center;

## text-align: center;

## width: 98vw;

## min-width: 0;

## padding: 12px;

## }

## .office-img {

## margin-bottom: 12px;

## width: 80px;

## height: 80px;

## }

## }

## src/index.js

## import React from "react";

## import ReactDOM from "react-dom/client";

## import "./index.css";

## import App from "./App";

## const root = ReactDOM.createRoot(document.getElementById("root"));

## root.render(

## <React.StrictMode>

## <App />

## </React.StrictMode>

## );

## src/index.css

\*,

\*::before,

\*::after {

box-sizing: border-box;

margin: 0;

padding: 0;

}

body {

min-height: 100vh;

background: linear-gradient(120deg, #f5f7fa 0%, #c3cfe2 100%);

font-family: 'Segoe UI', 'Roboto', Arial, sans-serif;

color: #222d3d;

}

.main-container {

max-width: 1200px;

margin: 0 auto;

padding: 16px;

}

header {

padding: 32px 0 18px 0;

}

header h1 {

color: #205072;

font-size: 2.8rem;

font-weight: 800;

text-shadow: 0 2px 12px rgba(32,80,114,0.08);

letter-spacing: 1px;

text-align: center;

margin-bottom: 0.7em;

}

@media (max-width: 600px) {

header h1 {

font-size: 1.5rem;

padding: 20px 0 10px 0;

}

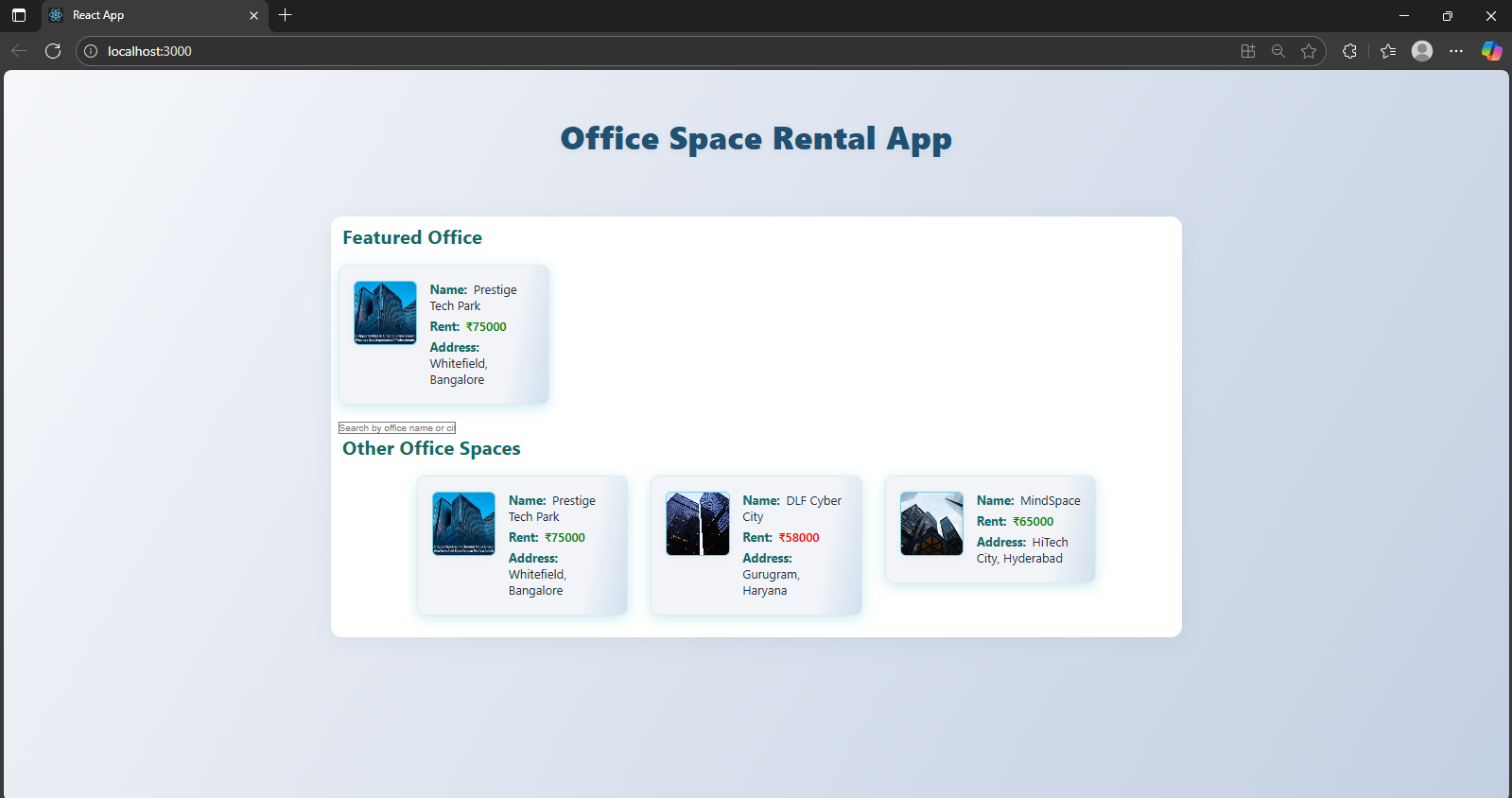
body {

padding: 6px;

}

}

OUTPUT



1. Create a React Application “eventexamplesapp” to handle various events of the form elements in HTML.

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. a. To increment the value b. Say Hello followed by a static message.

 Create a button “Say Welcome” which invokes the function which takes “welcome” as an argument.

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.

## src/App.js

import React, { Component } from "react";

import CurrencyConvertor from "./components/CurrencyConvertor";

class App extends Component {

constructor(props) {

super(props);

this.state = {

counter: 1,

message: "",

syntheticMsg: "",

};

this.handleDecrement = this.handleDecrement.bind(this);

this.handleSynthetic = this.handleSynthetic.bind(this);

}

// --- Event Handlers ---

handleIncrement = () => {

this.incrementValue();

this.sayHello();

};

incrementValue = () => {

this.setState((prevState) => ({

counter: prevState.counter + 1,

}));

};

sayHello = () => {

this.setState({ message: "Hello! Incremented and greeted." });

};

handleDecrement() {

this.setState((prevState) => ({

counter: prevState.counter - 1,

message: "Value decremented.",

}));

}

handleWelcome = (msg) => {

this.setState({ message: `Welcome, ${msg}!` });

};

handleSynthetic(e) {

// SyntheticEvent demo

this.setState({ syntheticMsg: "I was clicked" });

setTimeout(() => this.setState({ syntheticMsg: "" }), 1200);

}

render() {

const { counter, message, syntheticMsg } = this.state;

return (

<div className="main-container">

<h1>Event Examples App</h1>

<div className="event-panel">

<div className="counter-row">

<span className="counter-value">{counter}</span>

<div className="counter-btns">

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

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

</div>

</div>

<div className="action-btns">

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

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

</div>

<div className="panel-message">

{message}

{syntheticMsg && (

<div style={{ marginTop: 5, color: "#17604e", fontWeight: 600 }}>

{syntheticMsg}

</div>

)}

</div>

</div>

<CurrencyConvertor />

</div>

);

}

}

export default App;

## src/components/CurrencyConvertor.js

## import React, { useState } from "react";

## const EURO\_RATE = 80; // 1 Euro = 80 INR (Update as needed)

## function CurrencyConvertor() {

## const [rupees, setRupees] = useState("");

## const [result, setResult] = useState("");

## const handleSubmit = (event) => {

## event.preventDefault();

## if (!rupees || isNaN(rupees) || Number(rupees) <= 0) {

## setResult("Enter a valid Rupees amount.");

## return;

## }

## const euroValue = (parseFloat(rupees) / EURO\_RATE).toFixed(2);

## // Synthetic event and alert demo

## window.alert(`Converting to Euro. Amount is ${euroValue}`);

## setResult(`€${euroValue} (Euro)`);

## };

## return (

## <div className="currency-panel">

## <h2>Currency Convertor!!!</h2>

## <form className="converter-form" onSubmit={handleSubmit}>

## <div>

## <label htmlFor="amount"><b>Amount:</b></label>

## <input

## id="amount"

## type="number"

## value={rupees}

## placeholder="INR"

## min="0"

## onChange={e => setRupees(e.target.value)}

## required

## className="input-txt"

## />

## </div>

## <div>

## <label htmlFor="currency"><b>Currency:</b></label>

## <select id="currency" disabled>

## <option>Euro</option>

## </select>

## </div>

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

## </form>

## {result && (

## <div className="currency-result">{result}</div>

## )}

## </div>

## );

## }

## export default CurrencyConvertor;

## src/index.css

## body {

## background: #fcfcfe;

## color: #20315a;

## font-family: 'Segoe UI', Arial, sans-serif;

## margin: 0;

## }

## .main-container {

## margin: 40px auto;

## max-width: 520px;

## min-width: 300px;

## padding: 30px 22px 24px 22px;

## border-radius: 14px;

## background: #ffffff;

## box-shadow: 0 0 24px #c7dbfa66, 0 2px 16px #85a2ec16;

## }

## /\* Headings \*/

## h1 {

## font-size: 2.1rem;

## font-weight: 800;

## margin-bottom: 28px;

## color: #1e47a3;

## letter-spacing: 2px;

## text-align: center;

## }

## h2 {

## font-size: 1.4rem;

## color: #17813e;

## font-weight: bold;

## margin-bottom: 18px;

## letter-spacing: 1px;

## text-align: left;

## }

## /\* Event panel \*/

## .event-panel {

## background: #f2f8fd;

## border-radius: 8px;

## padding: 22px 14px 18px 14px;

## margin-bottom: 30px;

## box-shadow: 0 1.5px 10px #bdd8f81b;

## }

## .counter-row {

## display: flex;

## align-items: center;

## margin-bottom: 15px;

## }

## .counter-value {

## font-size: 1.5rem;

## font-weight: bold;

## padding: 8px 0;

## margin-right: 16px;

## color: #215db4;

## width: 28px;

## display: inline-block;

## text-align: center;

## }

## .counter-btns {

## display: inline-flex;

## gap: 10px;

## }

## .counter-btns button {

## background: #265bb7;

## color: #fff;

## border: none;

## border-radius: 5px;

## padding: 7px 13px;

## font-size: 1rem;

## font-weight: 500;

## cursor: pointer;

## transition: background .2s;

## }

## .counter-btns button:hover {

## background: #226418;

## }

## .action-btns {

## display: flex;

## gap: 10px;

## margin-top: 6px;

## margin-bottom: 10px;

## }

## .action-btns button {

## background: #f8a459;

## color: #fff;

## border: none;

## border-radius: 5px;

## padding: 6px 14px;

## font-size: 1rem;

## font-weight: 500;

## cursor: pointer;

## transition: background .2s;

## }

## .action-btns button:hover {

## background: #e66c1c;

## }

## .panel-message {

## min-height: 22px;

## color: #187aa5;

## margin-top: 9px;

## font-size: 1.05rem;

## }

## /\* Currency converter \*/

## .currency-panel {

## margin-top: 35px;

## background: #f4fcf3;

## border-radius: 8px;

## box-shadow: 0 1px 8px #46fa6617;

## padding: 23px 14px 25px 16px;

## }

## .converter-form {

## display: flex;

## gap: 14px;

## align-items: center;

## flex-wrap: wrap;

## margin-bottom: 12px;

## }

## .converter-form label {

## margin-right: 5px;

## font-size: 1rem;

## }

## .input-txt {

## font-size: 1rem;

## padding: 6px 9px;

## border: 1.2px solid #a7b6cb;

## border-radius: 5px;

## background: #fff;

## width: 83px;

## }

## select {

## padding: 6px 7px;

## font-size: 1rem;

## border: 1.2px solid #a7b6cb;

## border-radius: 5px;

## background: #fafdfe;

## }

## .converter-form button {

## background: #23c700;

## color: #fff;

## border: none;

## border-radius: 5px;

## padding: 6px 18px;

## font-size: 1rem;

## margin-left: 9px;

## cursor: pointer;

## transition: background .2s;

## }

## .converter-form button:hover {

## background: #1e9000;

## }

## .currency-result {

## margin-top: 13px;

## font-size: 1.15rem;

## font-weight: bold;

## color: #17813e;

## }

## /\* Responsive \*/

## @media (max-width: 600px) {

## .main-container { padding: 9px 2px 12px 2px; }

## .converter-form { flex-direction: column; gap: 2px; align-items: flex-start; }

## }

## src/index.js

import React from "react";

import ReactDOM from "react-dom/client";

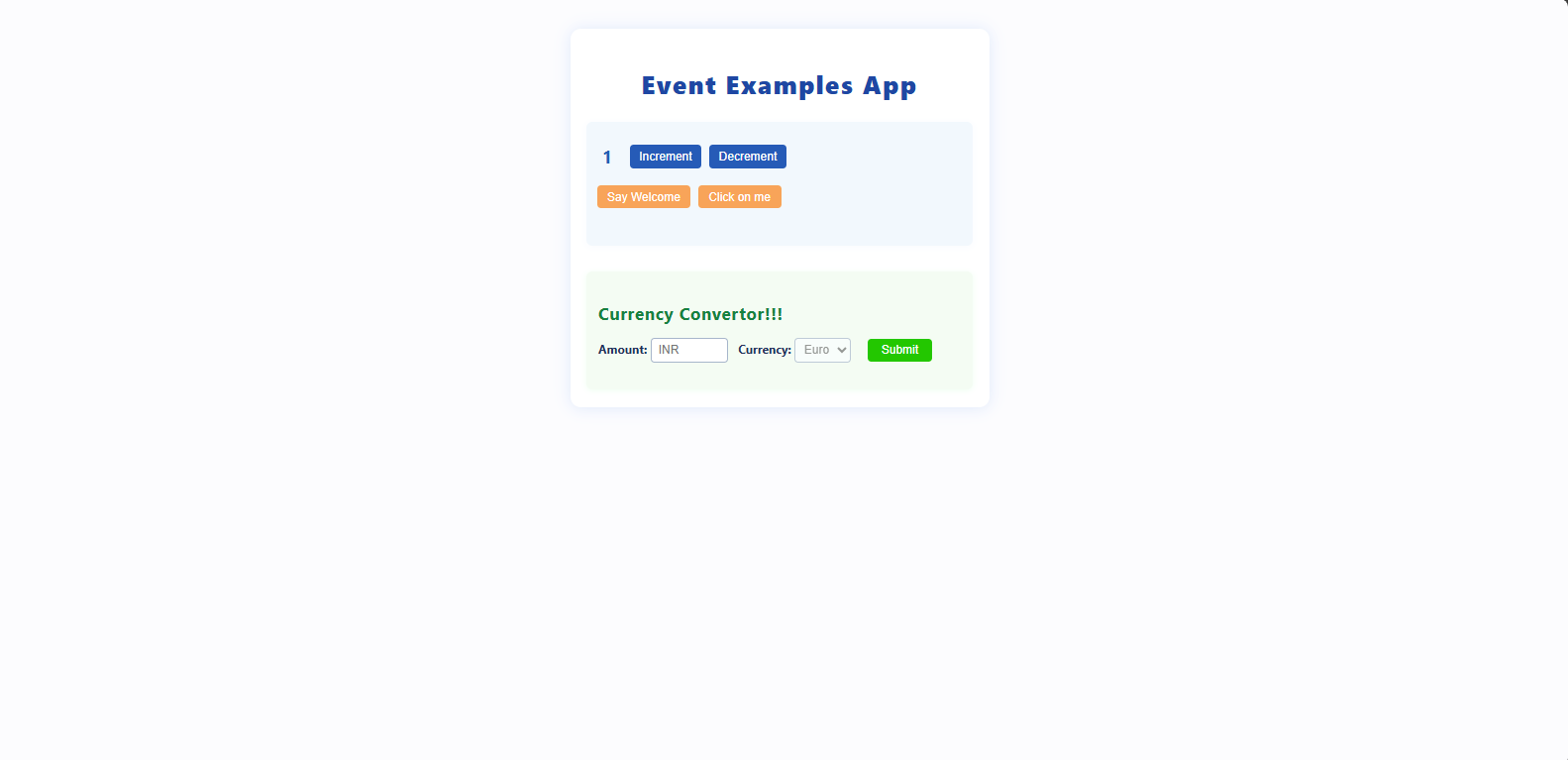
import App from "./App";

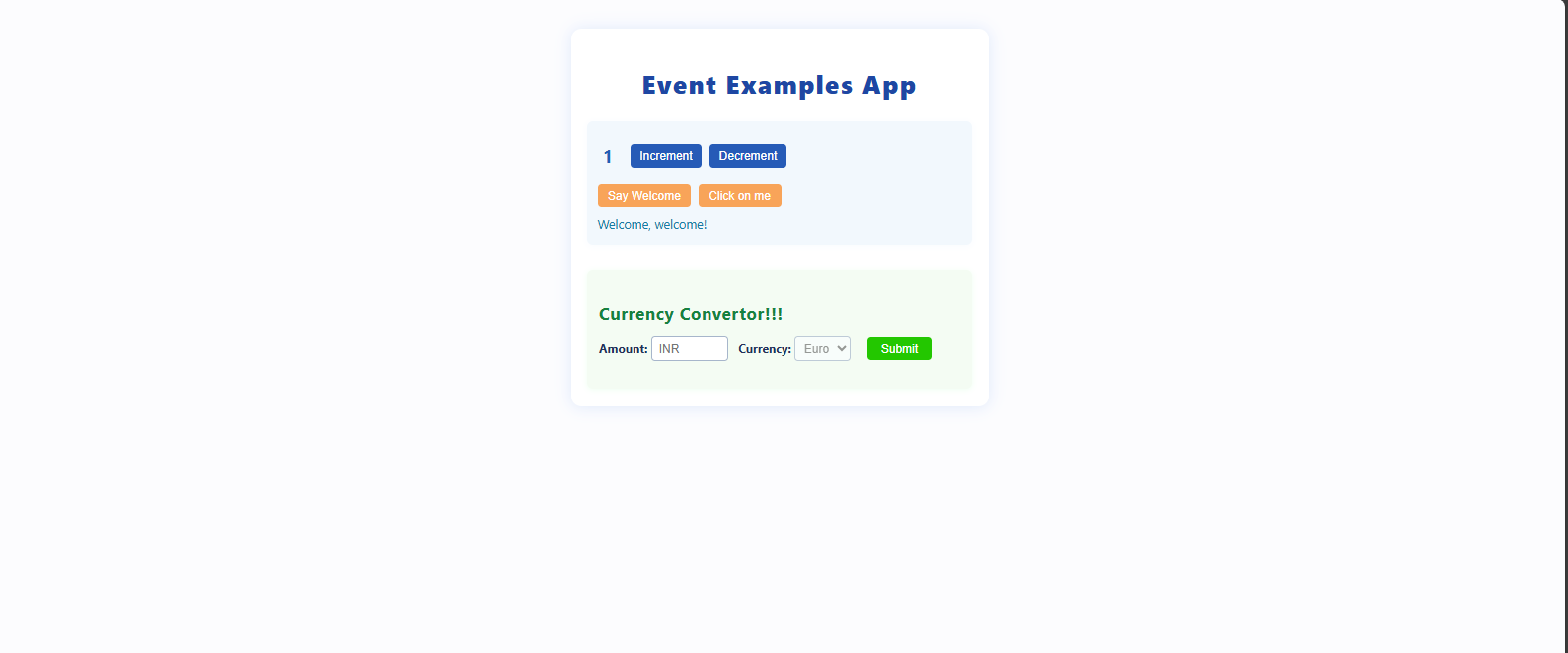
import "./index.css";

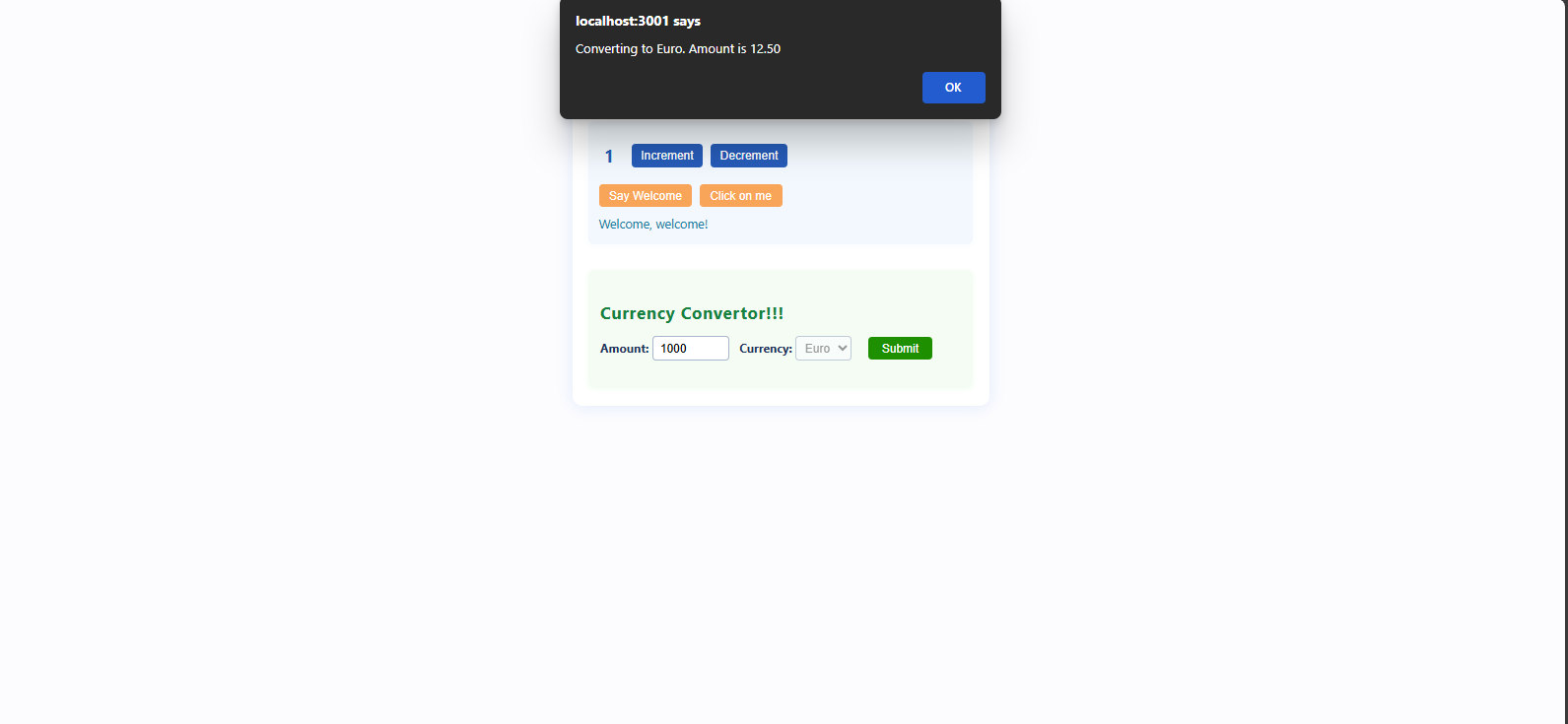
const root = ReactDOM.createRoot(document.getElementById("root"));

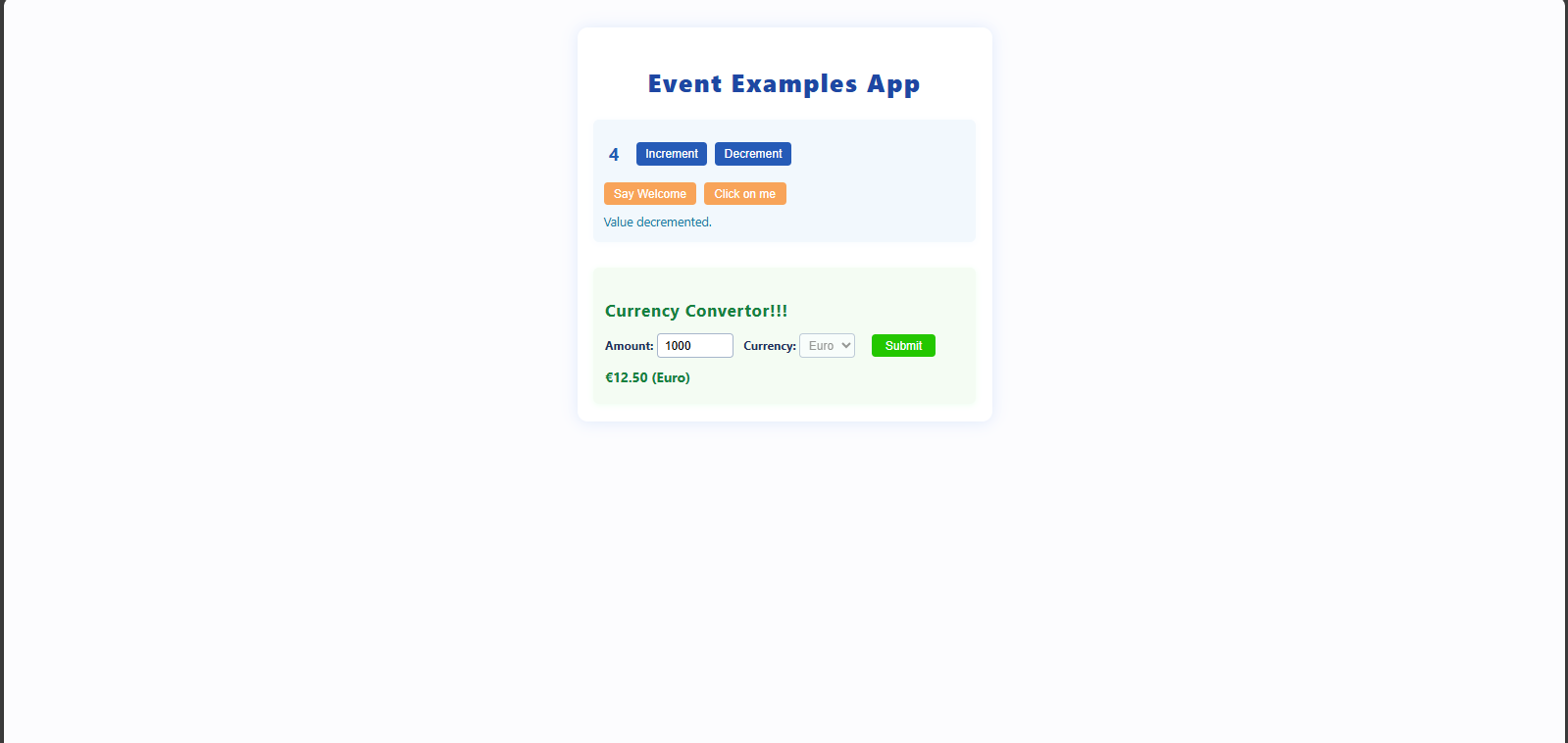
root.render(<App />);

OUTPUT:









1. 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.

## src/components/FlightDetails.js

## import React from "react";

## const flights = [

## { id: 1, airline: "Air India", from: "Delhi", to: "Mumbai", time: "09:00 AM" },

## { id: 2, airline: "IndiGo", from: "Bangalore", to: "Chennai", time: "01:30 PM" },

## { id: 3, airline: "SpiceJet", from: "Hyderabad", to: "Goa", time: "06:45 PM" },

## ];

## const FlightDetails = () => (

## <div>

## <h2>Available Flights</h2>

## <table style={{ width: "100%", borderCollapse: "collapse" }}>

## <thead>

## <tr>

## <th style={thStyle}>Airline</th>

## <th style={thStyle}>From</th>

## <th style={thStyle}>To</th>

## <th style={thStyle}>Time</th>

## </tr>

## </thead>

## <tbody>

## {flights.map(({ id, airline, from, to, time }) => (

## <tr key={id}>

## <td style={tdStyle}>{airline}</td>

## <td style={tdStyle}>{from}</td>

## <td style={tdStyle}>{to}</td>

## <td style={tdStyle}>{time}</td>

## </tr>

## ))}

## </tbody>

## </table>

## </div>

## );

## const thStyle = {

## borderBottom: "2px solid #ccc",

## padding: "8px",

## textAlign: "left",

## backgroundColor: "#f5f5f5",

## };

## const tdStyle = {

## borderBottom: "1px solid #ddd",

## padding: "8px",

## };

## export default FlightDetails;

## src/components/GuestPage.js

## import React from "react";

## import FlightDetails from "./FlightDetails";

## const GuestPage = () => (

## <div>

## <h1>Welcome, Guest!</h1>

## <p>Please log in to book your tickets.</p>

## <FlightDetails />

## </div>

## );

## export default GuestPage;

## 3. src/components/UserPage.js

## import React, { useState } from "react";

## const UserPage = () => {

## const [bookingInfo, setBookingInfo] = useState({

## flight: "",

## name: "",

## seats: 1,

## });

## const [message, setMessage] = useState("");

## const handleChange = (e) => {

## const { name, value } = e.target;

## setBookingInfo((prev) => ({ ...prev, [name]: value }));

## };

## const handleSubmit = (e) => {

## e.preventDefault();

## if (!bookingInfo.flight || !bookingInfo.name) {

## setMessage("Please fill in all fields.");

## return;

## }

## setMessage(`Successfully booked ${bookingInfo.seats} seat(s) on ${bookingInfo.flight} for ${bookingInfo.name}.`);

## setBookingInfo({ flight: "", name: "", seats: 1 });

## };

## return (

## <div>

## <h1>Book Your Ticket</h1>

## <form onSubmit={handleSubmit} style={formStyle}>

## <label>

## Flight:

## <select name="flight" value={bookingInfo.flight} onChange={handleChange} required style={inputStyle}>

## <option value="">--Select Flight--</option>

## <option value="Air India (Delhi to Mumbai)">Air India (Delhi to Mumbai)</option>

## <option value="IndiGo (Bangalore to Chennai)">IndiGo (Bangalore to Chennai)</option>

## <option value="SpiceJet (Hyderabad to Goa)">SpiceJet (Hyderabad to Goa)</option>

## </select>

## </label>

## <label>

## Name:

## <input

## type="text"

## name="name"

## value={bookingInfo.name}

## onChange={handleChange}

## required

## placeholder="Enter your name"

## style={inputStyle}

## />

## </label>

## <label>

## Number of Seats:

## <input

## type="number"

## name="seats"

## min="1"

## value={bookingInfo.seats}

## onChange={handleChange}

## required

## style={inputStyle}

## />

## </label>

## <button type="submit" style={buttonStyle}>Book Ticket</button>

## </form>

## {message && <p style={{ marginTop: 10, color: "green" }}>{message}</p>}

## </div>

## );

## };

## const formStyle = {

## display: "flex",

## flexDirection: "column",

## maxWidth: 400,

## gap: "12px",

## };

## const inputStyle = {

## padding: "8px",

## fontSize: "1rem",

## marginTop: "4px",

## };

## const buttonStyle = {

## padding: "10px",

## backgroundColor: "#007bff",

## border: "none",

## color: "white",

## fontSize: "1rem",

## cursor: "pointer",

## borderRadius: "4px",

## };

## export default UserPage;

## src/components/LoginLogoutButtons.js

## import React from "react";

## const LoginLogoutButtons = ({ loggedIn, onLogin, onLogout }) => (

## <div style={{ marginBottom: 20 }}>

## {loggedIn ? (

## <button onClick={onLogout} style={btnStyle}>

## Logout

## </button>

## ) : (

## <button onClick={onLogin} style={btnStyle}>

## Login

## </button>

## )}

## </div>

## );

## const btnStyle = {

## padding: "10px 20px",

## fontSize: "1.1rem",

## cursor: "pointer",

## borderRadius: "5px",

## border: "none",

## backgroundColor: "#28a745",

## color: "white",

## };

## export default LoginLogoutButtons;

## src/App.js

## import React, { useState } from "react";

## import GuestPage from "./components/GuestPage";

## import UserPage from "./components/UserPage";

## import LoginLogoutButtons from "./components/LoginLogoutButtons";

## const App = () => {

## const [loggedIn, setLoggedIn] = useState(false);

## const handleLogin = () => setLoggedIn(true);

## const handleLogout = () => setLoggedIn(false);

## return (

## <div style={containerStyle}>

## <LoginLogoutButtons loggedIn={loggedIn} onLogin={handleLogin} onLogout={handleLogout} />

## {loggedIn ? <UserPage /> : <GuestPage />}

## </div>

## );

## };

## const containerStyle = {

## maxWidth: 700,

## margin: "40px auto",

## padding: 24,

## fontFamily: "'Segoe UI', Tahoma, Geneva, Verdana, sans-serif",

## boxShadow: "0 4px 15px rgba(0,0,0,0.1)",

## borderRadius: "10px",

## backgroundColor: "#fff",

## };

## export default App;

## 6. src/index.js

## import React from "react";

## import ReactDOM from "react-dom/client";

## import App from "./App";

## import "./index.css";

## const root = ReactDOM.createRoot(document.getElementById("root"));

## root.render(<App />);

## 7. src/index.css

body {

margin: 0;

background-color: #f4f6f8;

color: #333;

}

h1, h2, p, label {

margin: 0;

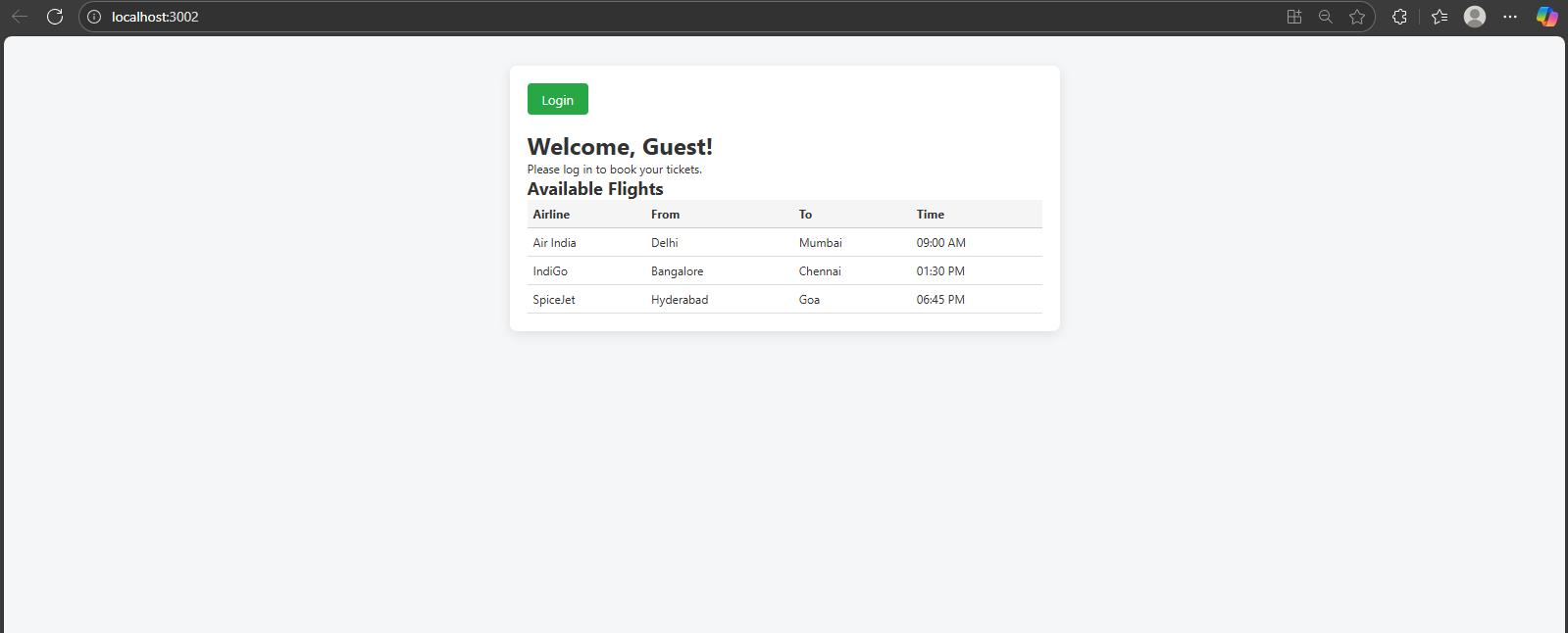
}

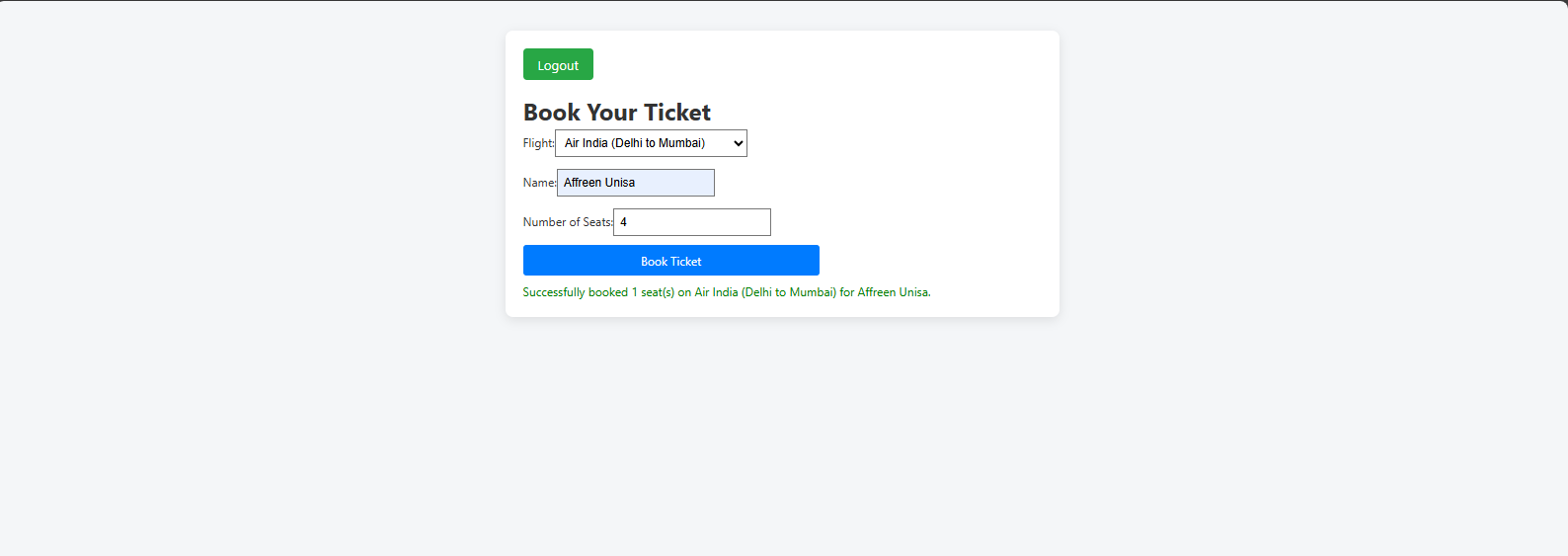
button {

font-family: inherit;

}

OUTPUT:





1. 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.

src/index.css

body {

margin: 0;

background-color: #f2f2f2;

font-family: Arial, sans-serif;

color: #333;

}

src/index.js

import React from "react";

import ReactDOM from "react-dom/client";

import App from "./App";

import "./index.css";

const root = ReactDOM.createRoot(document.getElementById("root"));

root.render(<App />);

src/App.js

import React, { useState } from "react";

import BookDetails from "./components/BookDetails";

import BlogDetails from "./components/BlogDetails";

import CourseDetails from "./components/CourseDetails";

function App() {

const [selectedComponent, setSelectedComponent] = useState("book");

// 1. Using if-else statement

const renderWithIfElse = () => {

if (selectedComponent === "book") {

return <BookDetails />;

} else if (selectedComponent === "blog") {

return <BlogDetails />;

} else if (selectedComponent === "course") {

return <CourseDetails />;

} else {

return <p>Please select a valid option.</p>;

}

};

// 2. Using ternary operator (nested)

const renderWithTernary = () =>

selectedComponent === "book" ? (

<BookDetails />

) : selectedComponent === "blog" ? (

<BlogDetails />

) : selectedComponent === "course" ? (

<CourseDetails />

) : (

<p>Please select a valid option.</p>

);

const renderWithSwitch = () => {

switch (selectedComponent) {

case "book":

return <BookDetails />;

case "blog":

return <BlogDetails />;

case "course":

return <CourseDetails />;

default:

return <p>Please select a valid option.</p>;

}

};

return (

<div className="app-container" style={{ maxWidth: 600, margin: "30px auto", fontFamily: "Arial, sans-serif", padding: 20 }}>

<h1>Blogger App</h1>

<div style={{ marginBottom: 20 }}>

<button onClick={() => setSelectedComponent("book")} style={buttonStyle}>

Book Details

</button>

<button onClick={() => setSelectedComponent("blog")} style={buttonStyle}>

Blog Details

</button>

<button onClick={() => setSelectedComponent("course")} style={buttonStyle}>

Course Details

</button>

</div>

<div style={{ border: "1px solid #ddd", padding: 16, borderRadius: 6 }}>

<h3>Conditional Rendering Examples:</h3>

<div style={sectionStyle}>

<h4>1. If-Else Statement</h4>

{renderWithIfElse()}

</div>

<div style={sectionStyle}>

<h4>2. Ternary Operator</h4>

{renderWithTernary()}

</div>

<div style={sectionStyle}>

<h4>3. Switch Case</h4>

{renderWithSwitch()}

</div>

<div style={sectionStyle}>

<h4>4. Logical AND Operator</h4>

{/\* Logical AND example: show BookDetails only if selected \*/}

{selectedComponent === "book" && <BookDetails />}

{selectedComponent === "blog" && <BlogDetails />}

{selectedComponent === "course" && <CourseDetails />}

</div>

<div style={sectionStyle}>

<h4>5. Inline IIFE (Immediately Invoked Function Expression)</h4>

{(() => {

if (selectedComponent === "book") return <BookDetails />;

if (selectedComponent === "blog") return <BlogDetails />;

if (selectedComponent === "course") return <CourseDetails />;

return <p>Please select a valid option.</p>;

})()}

</div>

</div>

</div>

);

}

const buttonStyle = {

padding: "8px 16px",

marginRight: 10,

background: "#007bff",

border: "none",

borderRadius: 4,

color: "white",

cursor: "pointer",

fontSize: 16,

};

const sectionStyle = {

backgroundColor: "#f9f9f9",

marginTop: 12,

padding: 12,

borderRadius: 4,

border: "1px solid #ccc",

};

export default App;

src/components/CourseDetails.js

import React from "react";

const CourseDetails = () => (

<div>

<h2>Course Details</h2>

<p>Course: Modern React Development</p>

<p>Instructor: John Smith</p>

<p>Duration: 6 weeks</p>

</div>

);

export default CourseDetails;

## src/components/BlogDetails.js

import React from "react";

const BlogDetails = () => (

<div>

<h2>Blog Details</h2>

<p>Title: React Best Practices</p>

<p>Author: Jane Doe</p>

<p>Published: August 2025</p>

</div>

);

export default BlogDetails;

import React from "react";

## src/components/BookDetails.js

const BookDetails = () => (

<div>

<h2>Book Details</h2>

<p>Title: Learning React</p>

<p>Author: Alex Banks & Eve Porcello</p>

<p>Pages: 350</p>

</div>

);

export default BookDetails;

