**REACT**

**WEEK - 7**

**Handson - 9**

In this hands-on lab, you will learn how to:

* Use map() method of ES6
* Apply arrow functions of ES6
* Implement Destructuring features of ES6

import React from "react";

const ListofPlayers = () => {

  const players = [

    { name: "Virat", score: 95 },

    { name: "Rohit", score: 88 },

    { name: "Rahul", score: 45 },

    { name: "Pant", score: 77 },

    { name: "Bumrah", score: 65 },

    { name: "Shami", score: 90 },

    { name: "Ashwin", score: 58 },

    { name: "Dhawan", score: 73 },

    { name: "Siraj", score: 69 },

    { name: "Hardik", score: 82 },

    { name: "Kuldeep", score: 66 }

  ];

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

  return (

    <div>

      <h2>All Players</h2>

      <ul>

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

          <li key={index}>{player.name} - {player.score}</li>

        ))}

      </ul>

      <h2>Players scoring below 70</h2>

      <ul>

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

          <li key={index}>{player.name} - {player.score}</li>

        ))}

      </ul>

    </div>

  );

};

export default ListofPlayers;

import React from "react";

const IndianPlayers = () => {

  const allPlayers = ["Virat", "Rohit", "Rahul", "Pant", "Bumrah", "Shami"];

  const oddPlayers = allPlayers.filter((\_, index) => index % 2 !== 0);

  const evenPlayers = allPlayers.filter((\_, index) => index % 2 === 0);

  const [t20player1, t20player2] = ["Surya", "Hardik"];

  const [ranjiPlayer1, ranjiPlayer2] = ["Pujara", "Rahane"];

  const mergedPlayers = [...[t20player1, t20player2], ...[ranjiPlayer1, ranjiPlayer2]];

  return (

    <div>

      <h2>Odd Team Players</h2>

      <ul>

        {oddPlayers.map((p, i) => <li key={i}>{p}</li>)}

      </ul>

      <h2>Even Team Players</h2>

      <ul>

        {evenPlayers.map((p, i) => <li key={i}>{p}</li>)}

      </ul>

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

      <ul>

        {mergedPlayers.map((p, i) => <li key={i}>{p}</li>)}

      </ul>

    </div>

  );

};

export default IndianPlayers;

import logo from './logo.svg';

import './App.css';

import React from "react";

import ListofPlayers from "./components/ListofPlayers";

import IndianPlayers from "./components/IndianPlayers";

function App() {

  const flag = true; // Toggle between true or false

  return (

    <div className="App">

      <h1>Welcome to Cricket App</h1>

      {flag ? <ListofPlayers /> : <IndianPlayers />}

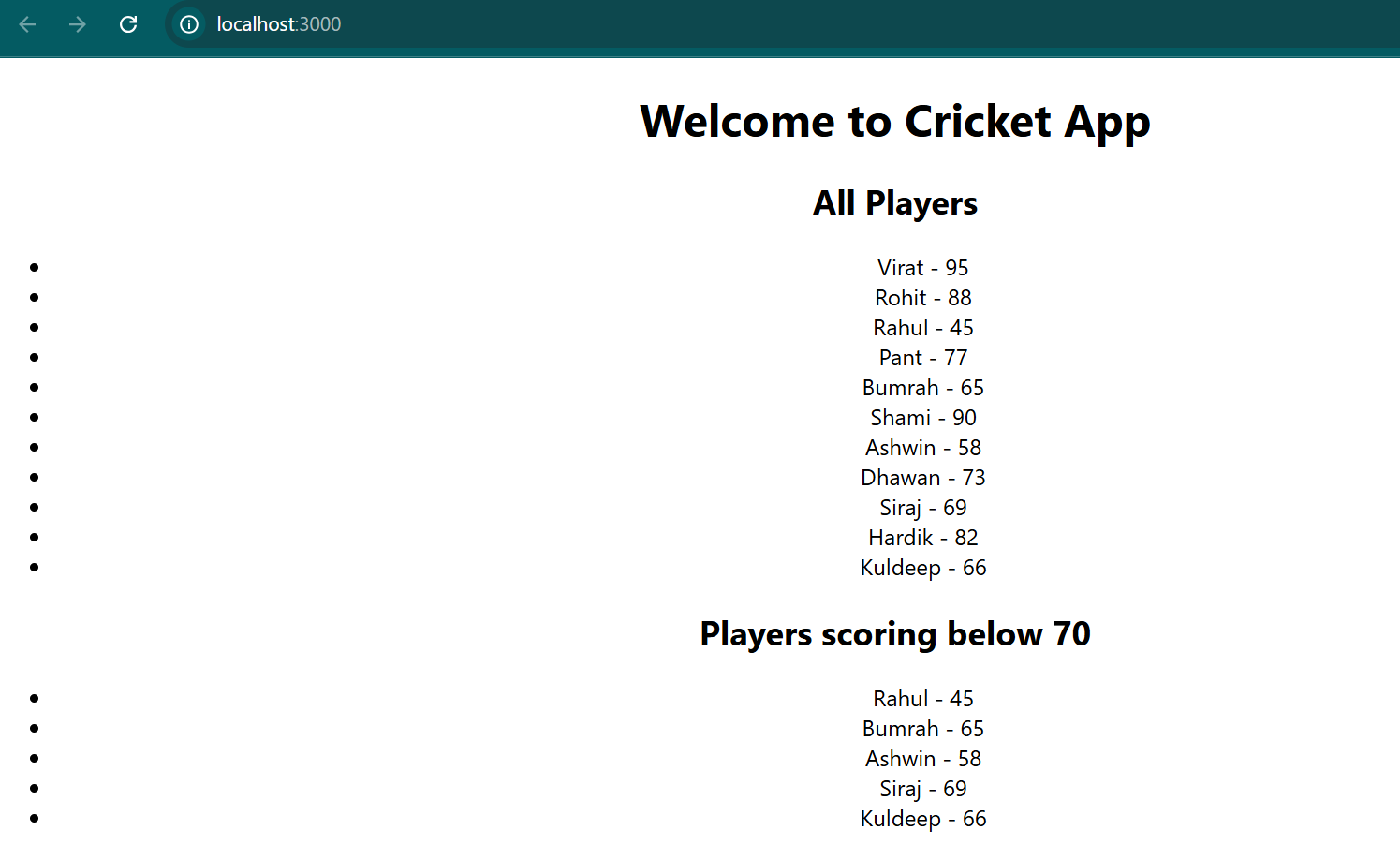
    </div>

  );

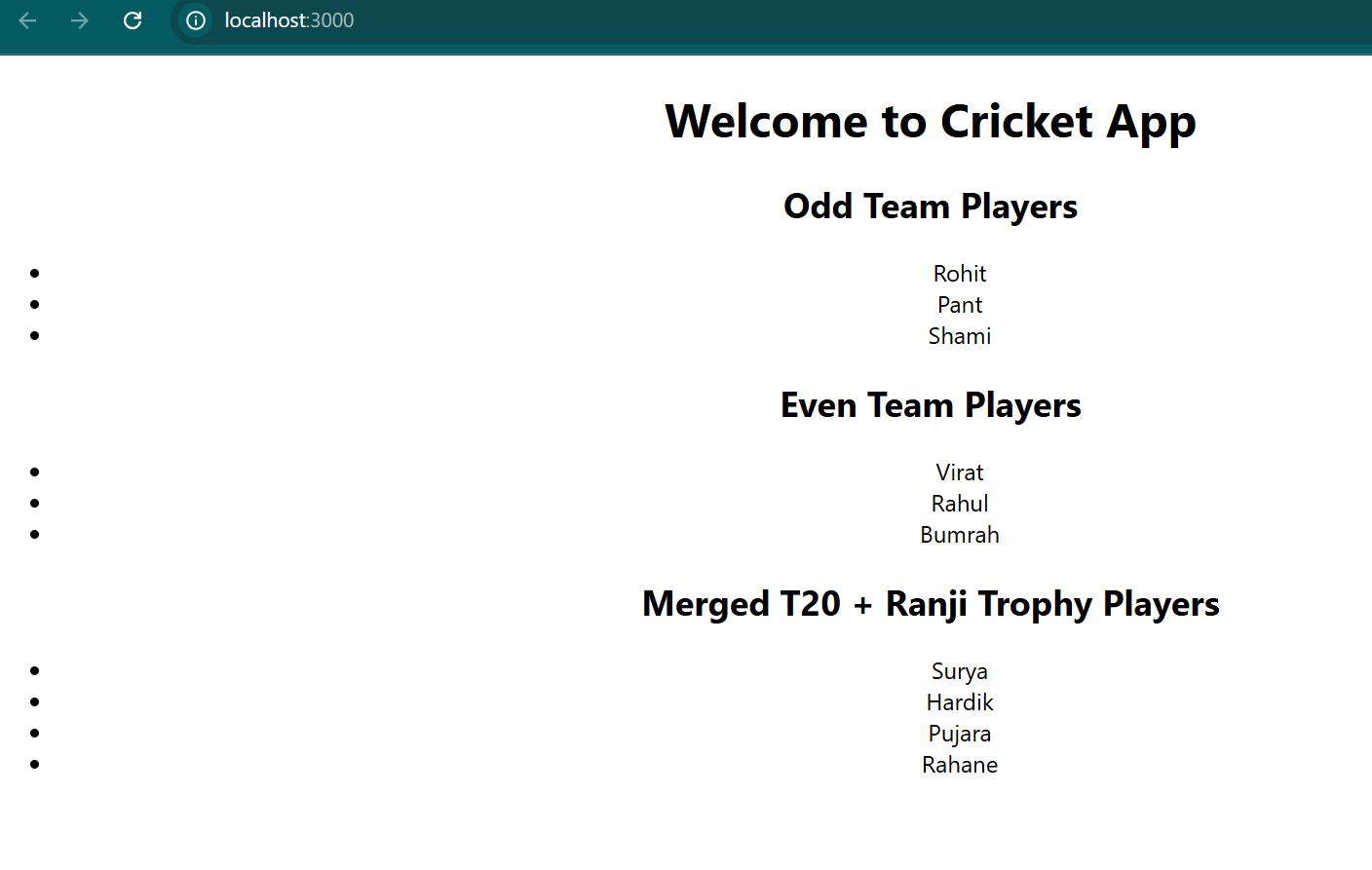
}

export default App;

Flag = true:



When flag = false:



**Handson - 10**

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.

import logo from './logo.svg';

import React from 'react';

import './App.css';

function App() {

  const mainOffice = {

    name: "Cozy Coworking Hub",

    rent: 55000,

    address: "123 Business Bay, Chennai"

  };

  const officeList = [

    { name: "Urban Workloft", rent: 45000, address: "MG Road, Bengaluru" },

    { name: "Sky Tower Office", rent: 78000, address: "Connaught Place, Delhi" },

    { name: "GreenSpace Rentals", rent: 62000, address: "T. Nagar, Chennai" },

    { name: "BudgetBase Offices", rent: 38000, address: "Camp Road, Pune" }

  ];

  return (

    <div className="App">

      <h1>Office Space Rental App</h1>

      <img src="image.jpg" alt="Office Space" width="600" height="300" />

      <h2>Main Office Details</h2>

      <p><strong>Name:</strong> {mainOffice.name}</p>

      <p><strong>Rent:</strong>

        <span style={{ color: mainOffice.rent > 60000 ? 'green' : 'red' }}>

          ₹{mainOffice.rent}

        </span>

      </p>

      <p><strong>Address:</strong> {mainOffice.address}</p>

      <h2>Other Available Offices</h2>

      <ul>

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

          <li key={index} style={{ marginBottom: "15px" }}>

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

            <p><strong>Rent:</strong>

              <span style={{ color: office.rent > 60000 ? 'green' : 'red' }}>

                ₹{office.rent}

              </span>

            </p>

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

          </li>

        ))}

      </ul>

    </div>

  );

}

export default App;

.App-logo {

  height: 40vmin;

  pointer-events: none;

}

@media (prefers-reduced-motion: no-preference) {

  .App-logo {

    animation: App-logo-spin infinite 20s linear;

  }

}

.App-header {

  background-color: #282c34;

  min-height: 100vh;

  display: flex;

  flex-direction: column;

  align-items: center;

  justify-content: center;

  font-size: calc(10px + 2vmin);

  color: white;

}

.App-link {

  color: #61dafb;

}

@keyframes App-logo-spin {

  from {

    transform: rotate(0deg);

  }

  to {

    transform: rotate(360deg);

  }

}

.App {

  padding: 20px;

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

}

h1 {

  color: #333;

}

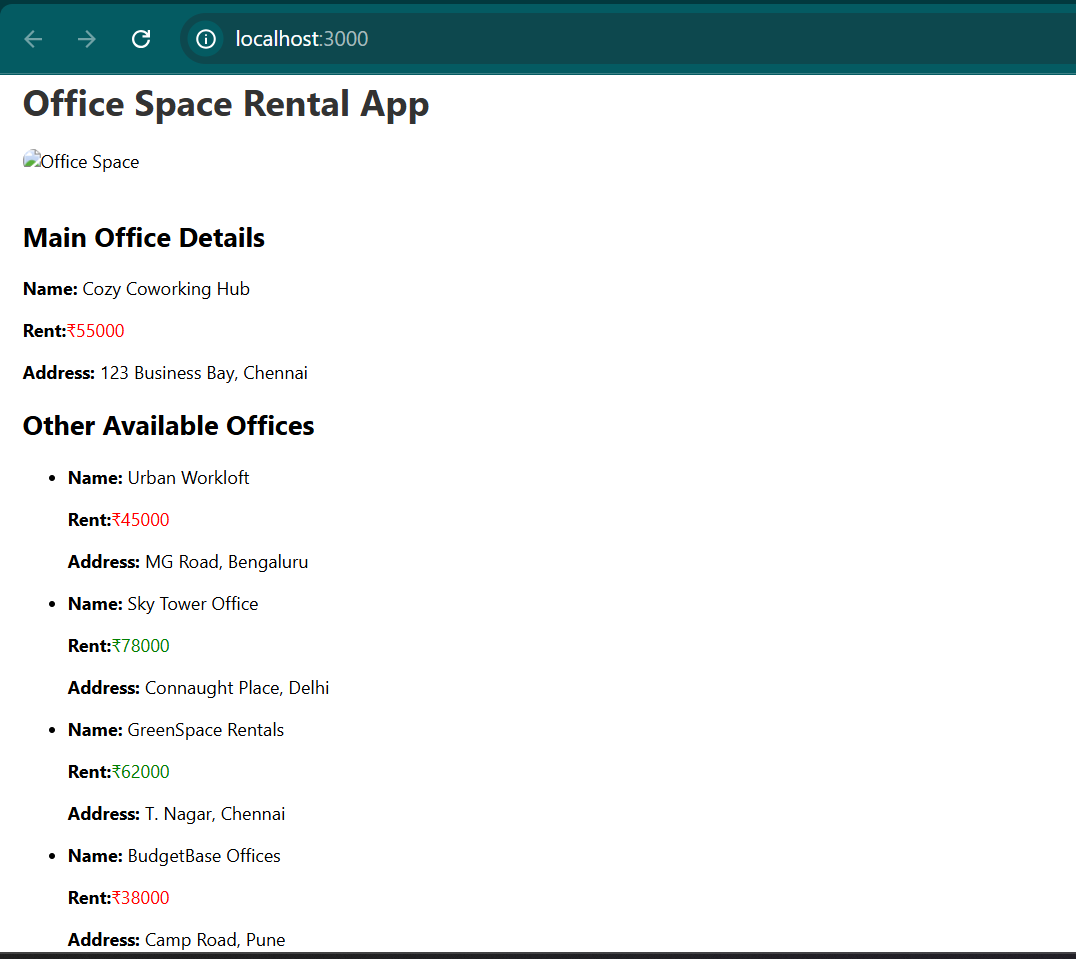
img {

  border-radius: 10px;

  margin-bottom: 20px;

}

OUTPUT:



**Handson - 11**

In this hands-on lab, you will learn how to:

* Implement Event handling concept in React applications
* Use this keyword
* Use synthetic event

import logo from './logo.svg';

import './App.css';

import React, { useState } from 'react';

import CurrencyConverter from './CurrencyConverter';

function App() {

  const [count, setCount] = useState(0);

  const handleIncrement = () => {

    increment();

    sayHello();

  };

  const increment = () => {

    setCount(prev => prev + 1);

  };

  const sayHello = () => {

    console.log("Hello! This is your static message");

  };

  const decrement = () => {

    setCount(prev => prev - 1);

  };

  const sayMessage = (message) => {

    alert(`You said: ${message}`);

  };

  const handleSynthetic = (event) => {

    console.log("I was clicked!");

  };

  return (

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

      <h1> Event Examples App</h1>

      <h2>Counter: {count}</h2>

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

      <button onClick={decrement} style={{ marginLeft: "10px" }}>Decrement</button>

      <br /><br />

      <button onClick={() => sayMessage("Welcome ")}>Say Welcome</button>

      <br /><br />

      <button onClick={handleSynthetic}>Synthetic Event: OnPress</button>

      <br /><br />

      <CurrencyConverter />

    </div>

  );

}

export default App;

import React, { useState } from 'react';

const CurrencyConverter = () => {

  const [rupees, setRupees] = useState('');

  const [euro, setEuro] = useState(null);

  const conversionRate = 0.011;

  const handleSubmit = (e) => {

    e.preventDefault();

    const inr = parseFloat(rupees);

    if (!isNaN(inr)) {

      setEuro((inr \* conversionRate).toFixed(2));

    } else {

      setEuro(null);

      alert("Please enter a valid number!");

    }

  };

  return (

    <div style={{ marginTop: "40px" }}>

      <h2> Currency Converter (INR ➡️ EURO)</h2>

      <form onSubmit={handleSubmit}>

        <input

          type="number"

          placeholder="Enter INR"

          value={rupees}

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

        />

        <button type="submit" style={{ marginLeft: "10px" }}>Convert</button>

      </form>

      {euro !== null && (

        <p> Euro Equivalent: <strong>€{euro}</strong></p>

      )}

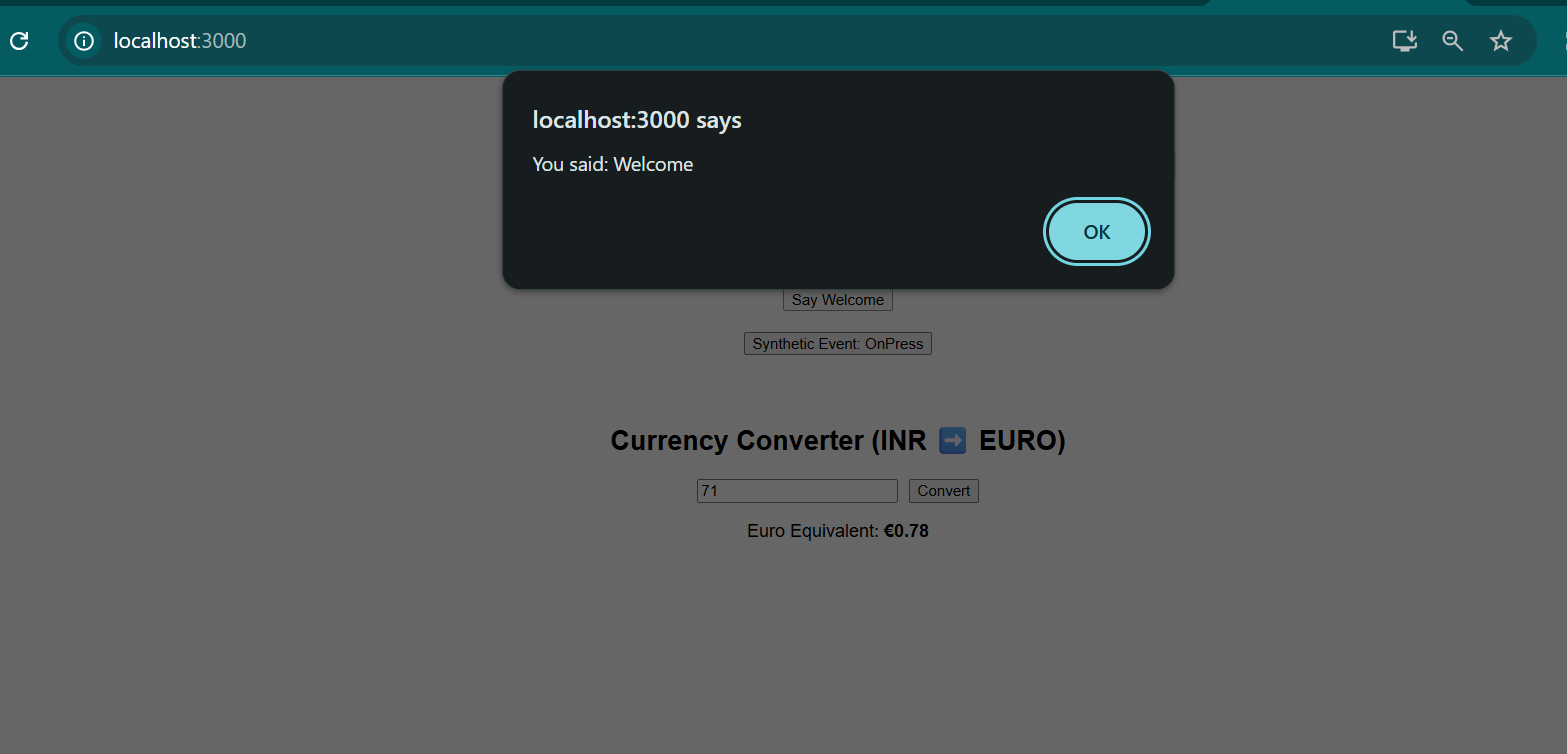
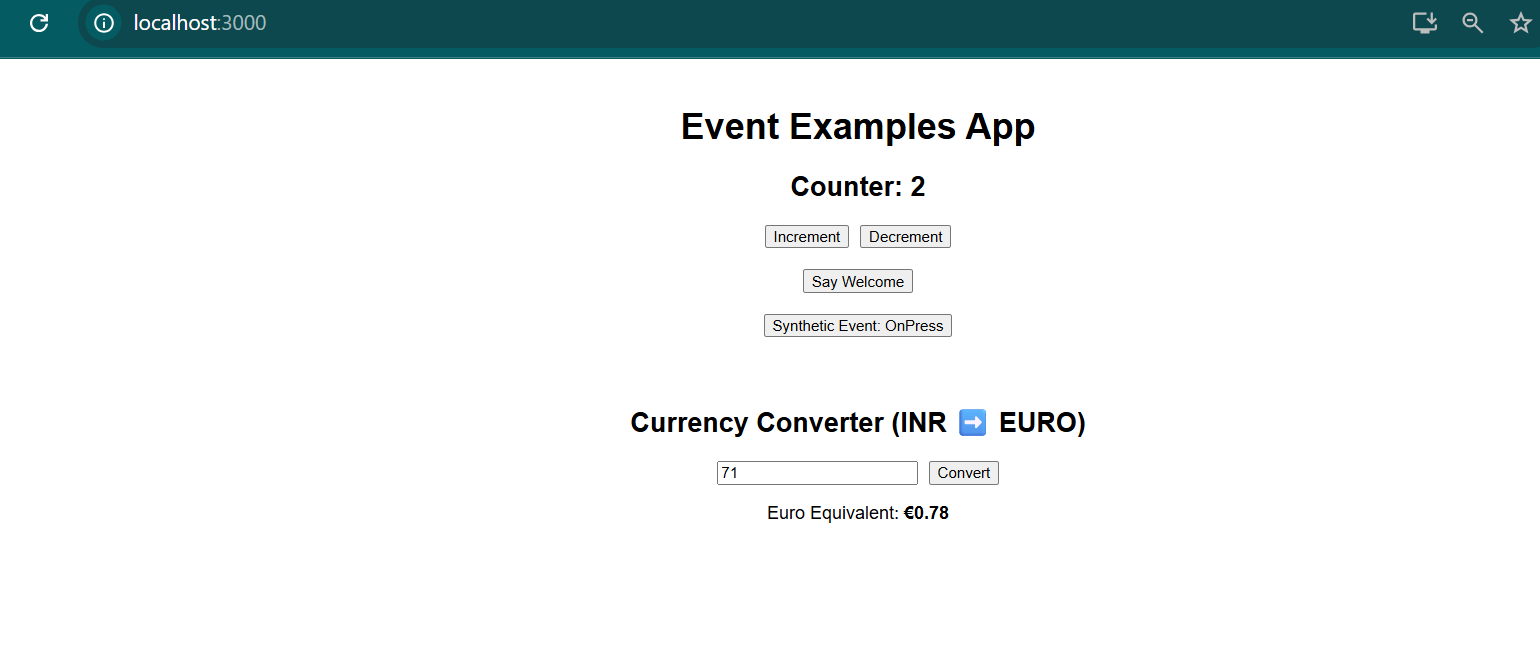
    </div>

  );

};

export default CurrencyConverter;

OUTPUT



**Handson - 12**

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.

import React from 'react';

const GuestPage = () => {

  const flights = [

    { id: 1, from: "Chennai", to: "Delhi", price: 5000 },

    { id: 2, from: "Mumbai", to: "Bangalore", price: 3500 },

    { id: 3, from: "Kolkata", to: "Hyderabad", price: 4000 },

  ];

  return (

    <div>

      <h2>Flight Listings (Guest View)</h2>

      <ul>

        {flights.map((flight) => (

          <li key={flight.id}>

            {flight.from} ➡️ {flight.to} | ₹{flight.price}

          </li>

        ))}

      </ul>

      <p><em>Login to book your flight!</em></p>

    </div>

  );

};

export default GuestPage;

import React from 'react';

const GuestPage = () => {

  const flights = [

    { id: 1, from: "Chennai", to: "Delhi", price: 5000 },

    { id: 2, from: "Mumbai", to: "Bangalore", price: 3500 },

    { id: 3, from: "Kolkata", to: "Hyderabad", price: 4000 },

  ];

  return (

    <div>

      <h2> Flight Listings (Guest View)</h2>

      <ul>

        {flights.map((flight) => (

          <li key={flight.id}>

            {flight.from} ➡️ {flight.to} | ₹{flight.price}

          </li>

        ))}

      </ul>

      <p><em>Login to book your flight!</em></p>

    </div>

  );

};

export default GuestPage;

import logo from './logo.svg';

import './App.css';

// src/App.js

import React, { useState } from 'react';

import GuestPage from './Components/GuestPage';

import UserPage from './Components/UserPage';

function App() {

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

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

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

  return (

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

      <h1>Ticket Booking App</h1>

      {isLoggedIn ? (

        <>

          <button onClick={handleLogout}>Logout</button>

          <UserPage />

        </>

      ) : (

        <>

          <button onClick={handleLogin}>Login</button>

          <GuestPage />

        </>

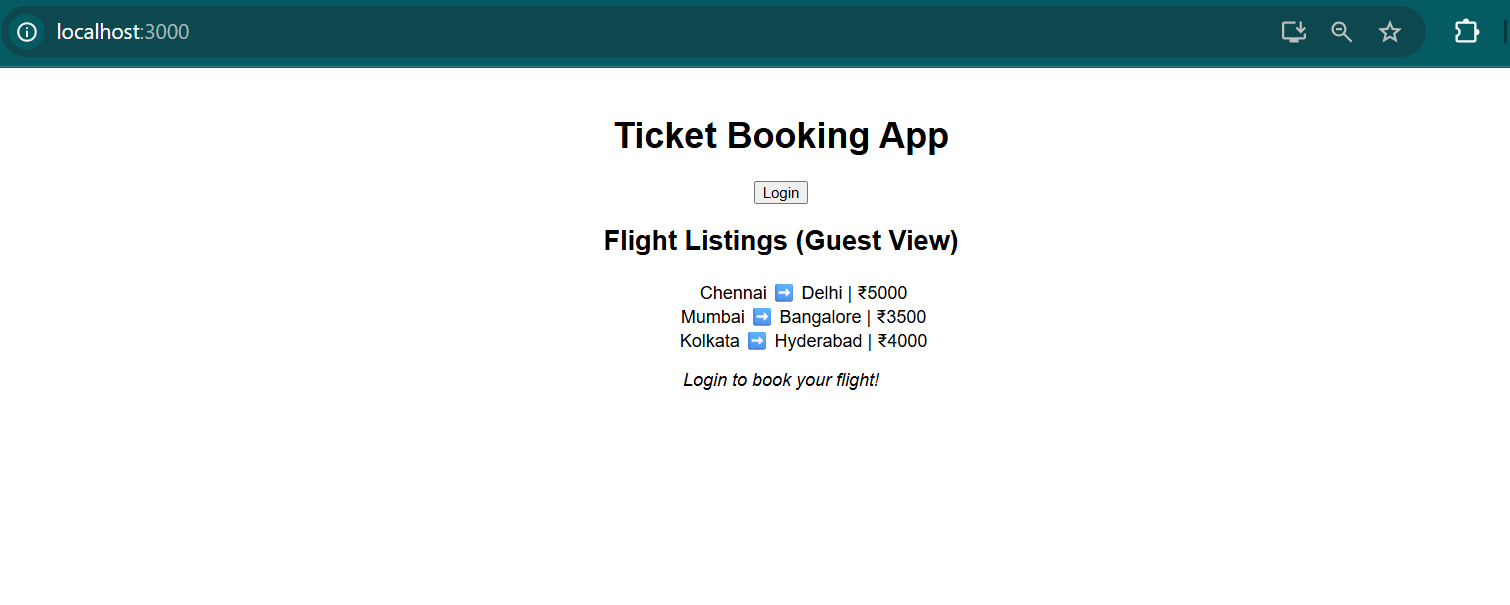
      )}

    </div>

  );

}

export default App;



**Handson - 13**

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.

import React from 'react';

const BookDetails = () => (

  <div>

    <h2>Book Details</h2>

    <p>Title: "React for Beginners"</p>

    <p>Author: Dan Abramov</p>

    <p>Published: 2023</p>

  </div>

);

export default BookDetails;

import React from 'react';

const BlogDetails = () => (

  <div>

    <h2> Blog Details</h2>

    <p>Title: Understanding JSX</p>

    <p>Author: Jane Doe</p>

    <p>Published: July 2025</p>

  </div>

);

export default BlogDetails;

import React from 'react';

const CourseDetails = () => (

  <div>

    <h2> Course Details</h2>

    <p>Course: Full Stack Web Development</p>

    <p>Platform: Udemy</p>

    <p>Duration: 40 Hours</p>

  </div>

);

export default CourseDetails;

import logo from './logo.svg';

import './App.css';

import React, { useState } from 'react';

import BookDetails from './Components/BookDetails';

import BlogDetails from './Components/BlogDetails';

import CourseDetails from './Components/CourseDetails';

function App() {

  const [selected, setSelected] = useState('book');

  // Switch Case Rendering

  const renderUsingSwitch = () => {

    switch (selected) {

      case 'book':

        return <BookDetails />;

      case 'blog':

        return <BlogDetails />;

      case 'course':

        return <CourseDetails />;

      default:

        return <p>Please select an option above.</p>;

    }

  };

  return (

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

      <h1>Blogger App</h1>

      <div>

        <button onClick={() => setSelected('book')}>Book</button>

        <button onClick={() => setSelected('blog')}>Blog</button>

        <button onClick={() => setSelected('course')}>Course</button>

        <button onClick={() => setSelected('none')}>Clear</button>

      </div>

      <hr />

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

      {(() => {

        if (selected === 'book') {

          return <BookDetails />;

        } else if (selected === 'blog') {

          return <BlogDetails />;

        } else if (selected === 'course') {

          return <CourseDetails />;

        } else {

          return <p>Click one of the buttons to display content.</p>;

        }

      })()}

      <hr />

      {/\* 2. Ternary Operator \*/}

      {selected === 'book' ? (

        <BookDetails />

      ) : selected === 'blog' ? (

        <BlogDetails />

      ) : selected === 'course' ? (

        <CourseDetails />

      ) : (

        <p>No section selected via ternary.</p>

      )}

      <hr />

      {/\* 3. Logical AND (&&) \*/}

      {selected === 'book' && <BookDetails />}

      {selected === 'blog' && <BlogDetails />}

      {selected === 'course' && <CourseDetails />}

      {selected === 'none' && <p>No section selected via && logic.</p>}

      <hr />

      {/\* 4. Switch Statement \*/}

      {renderUsingSwitch()}

    </div>

  );

}

export default App;

OUTPUT:

