**COGNIZANT DIGITAL NURTURE PROGRAM 4.0**

**JAVA FSE DEEP SKILLING**

**SUPERSET ID : 6391159**

**NAME : SATHYA SHREE R**

**CLASS : ECE C**

**REG NO : 727822TUEC207**

**WEEK 7 HANDS ON EXERCISE – REACT**

**Exercise 1 : React Application - CricketApp**

**PROBLEM STATEMENT :**

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

ES6 features:

* Show all players (name & score) using map().
* Filter players with score < 70 using arrow functions.
* Split into odd/even teams using destructuring.
* Merge T20 & Ranji players using spread operator.

**CODE :**

**App.js**

import React from "react";

import ListofPlayers from "./ListofPlayers";

import IndianPlayers from "./IndianPlayers";

function App() {

  // Change the value of flag to see different components

  const flag = false; // set to false to display IndianPlayers component

  return (

    <div className="App">

      <h1>Cricket App</h1>

      {/\* Conditional rendering using flag \*/}

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

    </div>

  );

}

export default App;

**index.js**

import React from "react";

import ReactDOM from "react-dom/client";

import App from "./App";

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

root.render(

  <React.StrictMode>

    <App />

  </React.StrictMode>

);

**IndianPlayers.js**

import React from "react";

const IndianPlayers = () => {

  const teamPlayers = [

    "Virat",

    "Rohit",

    "Rahul",

    "Iyer",

    "Hardik",

    "Jadeja",

    "Pant",

    "Bumrah",

    "Shami",

    "Bhuvi",

    "Kuldeep"

  ];

  const oddTeam = teamPlayers.filter((player, index) => (index + 1) % 2 !== 0);

  const evenTeam = teamPlayers.filter((player, index) => (index + 1) % 2 === 0);

  const T20players = ["Surya", "Samson", "Dinesh Karthik"];

  const RanjiTrophyPlayers = ["Pujara", "Rahane", "Mayank"];

  const mergedPlayers = [...T20players, ...RanjiTrophyPlayers];

  return (

    <div>

      <h2>Odd Team Players</h2>

      <ul>

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

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

        ))}

      </ul>

      <h2>Even Team Players</h2>

      <ul>

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

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

        ))}

      </ul>

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

      <ul>

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

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

        ))}

      </ul>

    </div>

  );

};

export default IndianPlayers;

**ListofPlayers.js**

import React from "react";

const ListofPlayers = () => {

  const players = [

    { name: "Virat Kohli", score: 95 },

    { name: "Rohit Sharma", score: 60 },

    { name: "KL Rahul", score: 75 },

    { name: "Shreyas Iyer", score: 55 },

    { name: "Hardik Pandya", score: 88 },

    { name: "Ravindra Jadeja", score: 65 },

    { name: "Rishabh Pant", score: 72 },

    { name: "Jasprit Bumrah", score: 45 },

    { name: "Mohammed Shami", score: 50 },

    { name: "Bhuvneshwar Kumar", score: 69 },

    { name: "Kuldeep Yadav", score: 80 }

  ];

  const filteredPlayers = 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 with scores below 70</h2>

      <ul>

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

          <li key={index}>

            {player.name} - {player.score}

          </li>

        ))}

      </ul>

    </div>

  );

};

export default ListofPlayers;

**index.html**

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8" />

    <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <title>Cricket App</title>

  </head>

  <body>

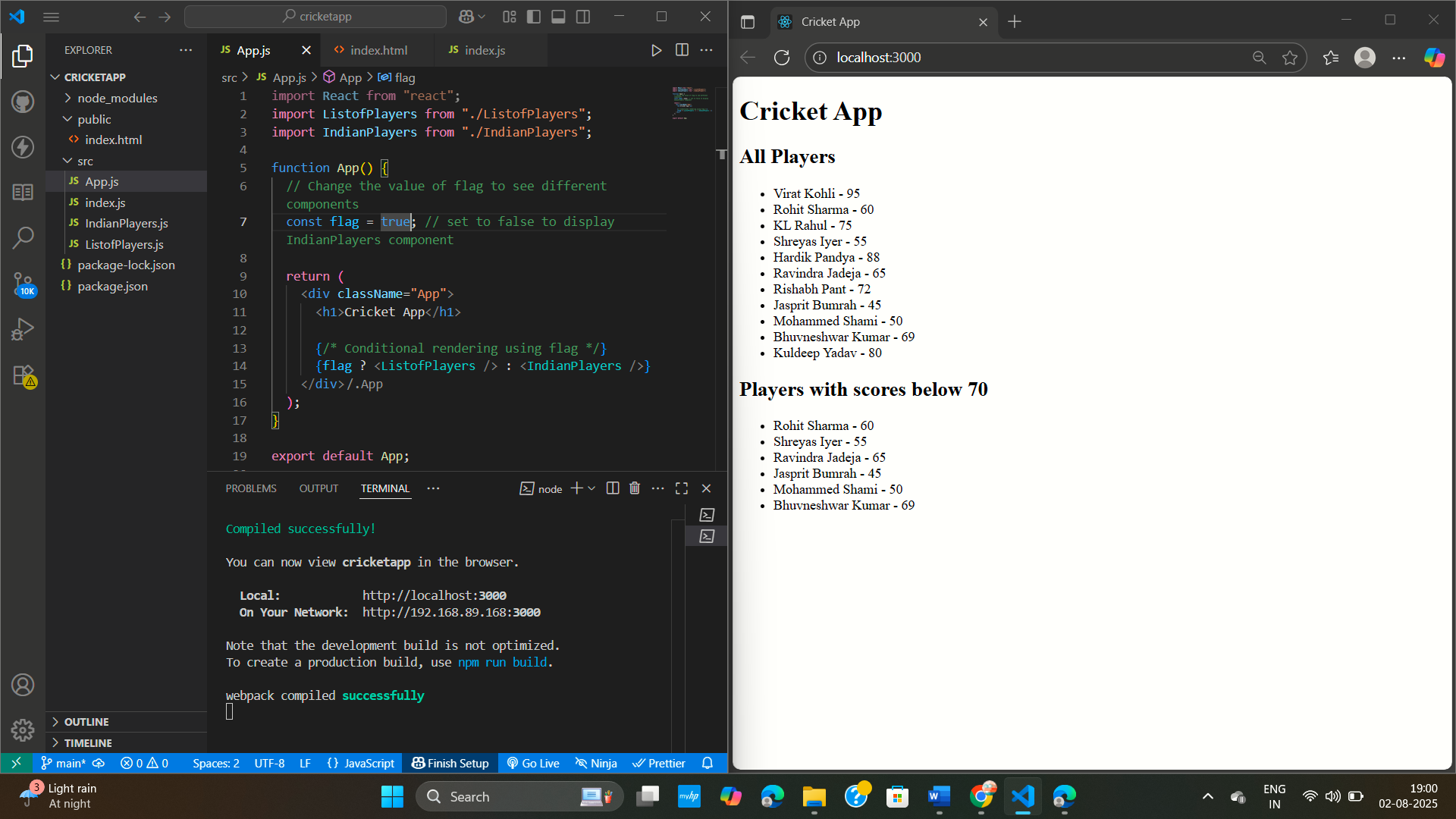
    <noscript>You need to enable JavaScript to run this app.</noscript>

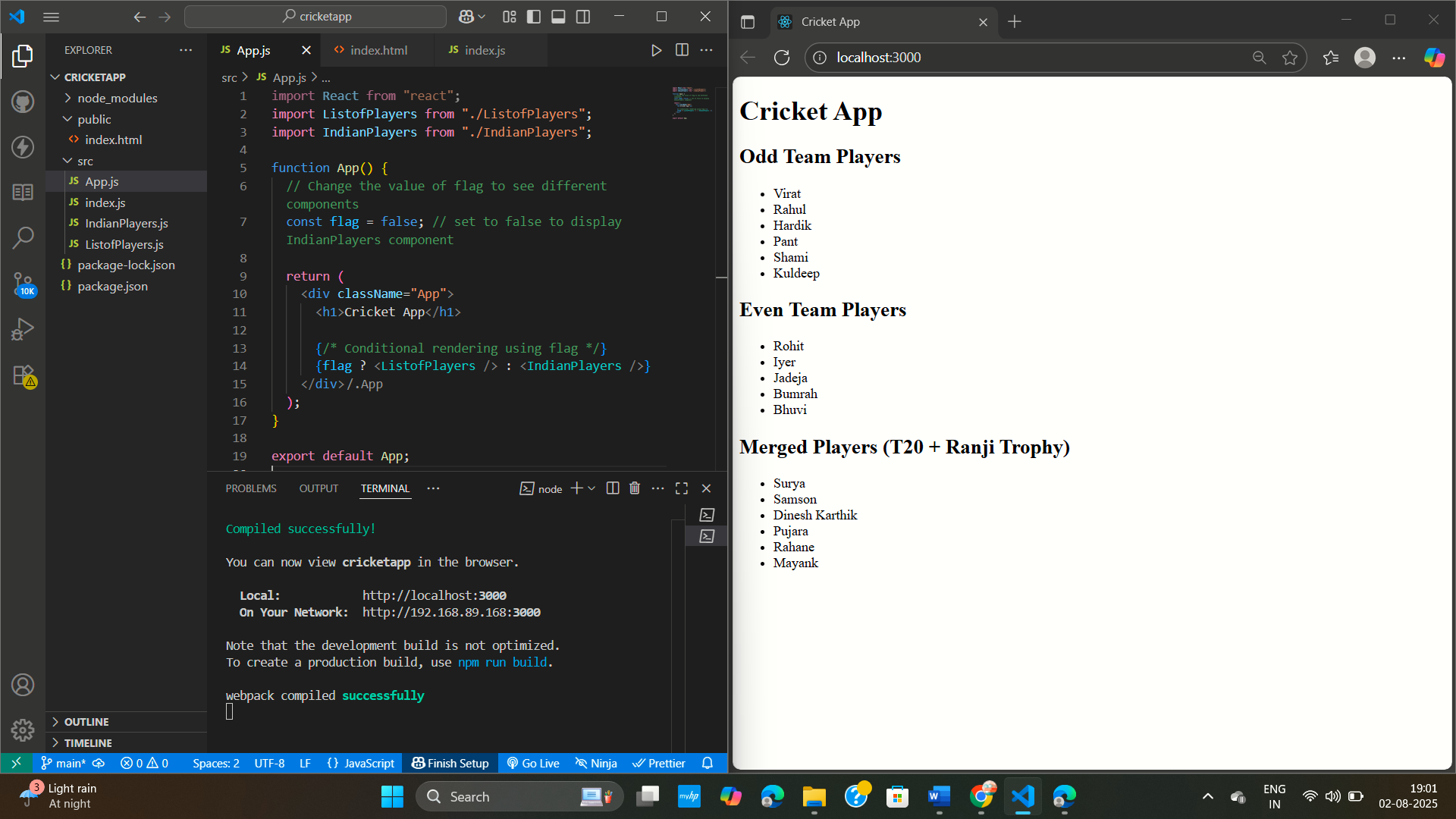
    <div id="root"></div>

  </body>

</html>

**OUTPUT :**





**Exercise 2 : React Application – Office Space Rental**

**PROBLEM STATEMENT :**

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

* Show a heading, office image, name, rent, and address using JSX elements and attributes.
* Loop through multiple office objects using map().
* Apply **inline CSS**: rent < 60000 → red, rent ≥ 60000 → green.

**CODE :**

**App.js**

import React from "react";

function App() {

  // Heading element

  const heading = <h1>Office Space Rental Listings</h1>;

  // List of office objects

  const offices = [

    {

      name: "Tech Park Center",

      rent: 65000,

      address: "456 Tech Street, Hyderabad",

      img: "https://images.unsplash.com/photo-1504384308090-c894fdcc538d?auto=format&fit=crop&w=800&q=80",

    },

    {

      name: "Urban Work Hub",

      rent: 48000,

      address: "789 Downtown Avenue, Mumbai",

      img: "https://images.unsplash.com/photo-1497366216548-37526070297c?auto=format&fit=crop&w=800&q=80",

    },

  ];

  // Render offices with conditional inline style

  return (

    <div style={{ textAlign: "center", padding: "20px" }}>

      {heading}

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

        <div

          key={index}

          style={{

            border: "1px solid #ccc",

            margin: "10px auto",

            padding: "10px",

            width: "300px",

            borderRadius: "10px",

          }}

        >

          <br></br>

          <img src={office.img} alt={office.name} style={{ width: "100%" }} />

          <h2>{office.name}</h2>

          <p style={{ margin: "5px 0" }}>{office.address}</p>

          <p

            style={{

              color: office.rent < 60000 ? "red" : "green",

              fontWeight: "bold",

            }}

          >

            Rent: ₹{office.rent}

          </p>

        </div>

      ))}

    </div>

  );

}

export default App;

**index.js**

import React from "react";

import ReactDOM from "react-dom/client";

import App from "./App";

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

root.render(

  <React.StrictMode>

    <App />

  </React.StrictMode>

);

**index.html**

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <title>Office Space Rental App</title>

  </head>

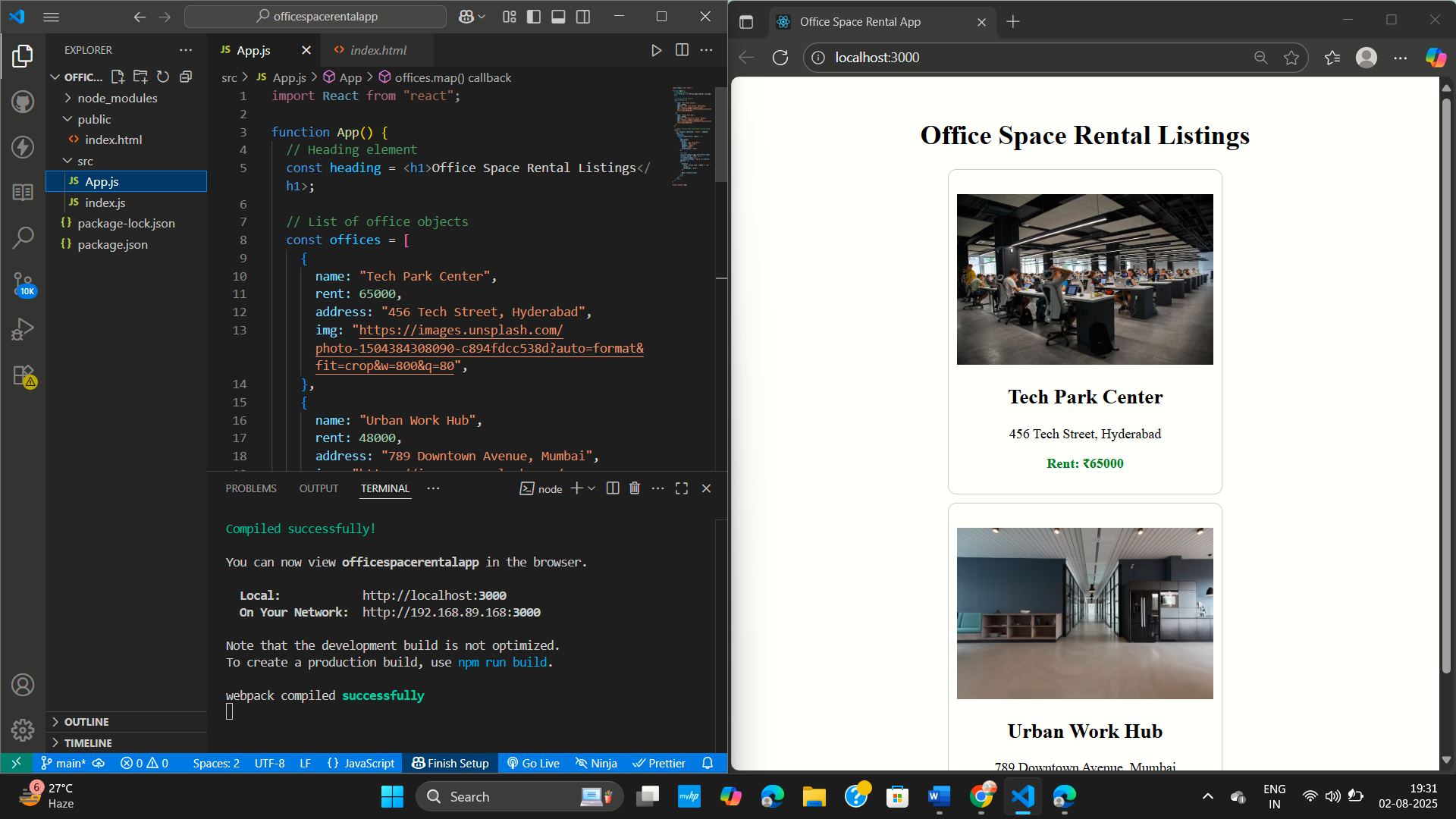
  <body>

    <div id="root"></div>

  </body>

</html>

**OUTPUT :**



**Exercise 3 : Student Management Portal - CalculateScore**

**PROBLEM STATEMENT :**

Create a React application **eventexamplesapp** to demonstrate React event handling using class and functional components. Implement buttons to increment/decrement a counter, display alerts using multiple event handlers, and handle synthetic events. Additionally, build a **Currency Converter** component with an amount input, a dropdown for multiple currency types (Euro, USD, GBP, AUD, CAD), and display the converted amount using event handling and form submission.

**CODE :**

**App.js**

import React, { Component } from "react";

import CurrencyConvertor from "./CurrencyConvertor";

class App extends Component {

  constructor(props) {

    super(props);

    this.state = {

      count: 0,

    };

  }

  increment = () => {

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

  };

  sayHello = () => {

    alert("Hello! This is a static message.");

  };

  handleIncrement = () => {

    this.increment();

    this.sayHello();

  };

  decrement = () => {

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

  };

  sayWelcome = (message) => {

    alert(message);

  };

  handleSyntheticEvent = (e) => {

    alert("I was clicked (Synthetic Event)");

    console.log(e); // Synthetic event object

  };

  render() {

    return (

      <div style={{ textAlign: "center", padding: "20px" }}>

        <h1>React Event Examples</h1>

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

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

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

        <br /><br />

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

        <br /><br />

        <button onClick={this.handleSyntheticEvent}>OnPress</button>

        <CurrencyConvertor />

      </div>

    );

  }

}

export default App;

**CurrencyConvertor.js**

import React, { useState } from "react";

function CurrencyConvertor() {

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

  const [currency, setCurrency] = useState("Euro");

  const handleSubmit = (e) => {

    e.preventDefault();

    // Conversion logic: Example rate (1 Euro = 80 INR)

    if (currency === "Euro") {

      const inrAmount = parseFloat(amount) \* 80;

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

    }

  };

  return (

    <div style={{ marginTop: "30px" }}>

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

      <form onSubmit={handleSubmit}>

        <div>

          <label style={{ marginRight: "10px" }}>Amount: </label>

          <input

            type="number"

            value={amount}

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

            required

          />

        </div>

        <div style={{ marginTop: "10px" }}>

          <label style={{ marginRight: "10px" }}>Currency: </label>

          <select

            value={currency}

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

          >

            <option value="Euro">Euro</option>

          </select>

        </div>

        <div style={{ marginTop: "15px" }}>

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

        </div>

      </form>

    </div>

  );

}

export default CurrencyConvertor;

**index.ja**

import React from "react";

import ReactDOM from "react-dom/client";

import App from "./App";

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

root.render(

  <React.StrictMode>

    <App />

  </React.StrictMode>

);

**index.html**

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <title>Event Examples App</title>

  </head>

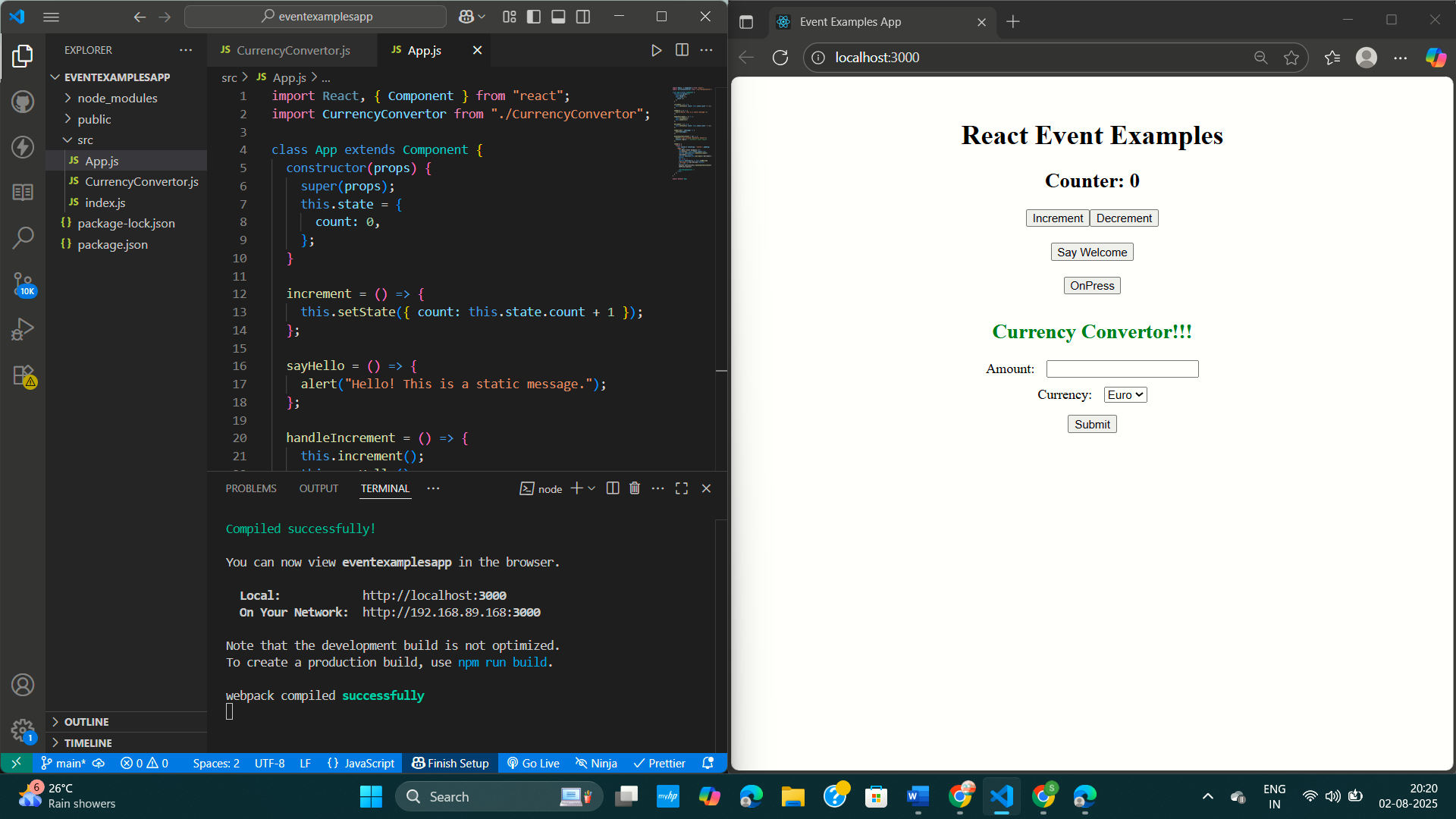
  <body>

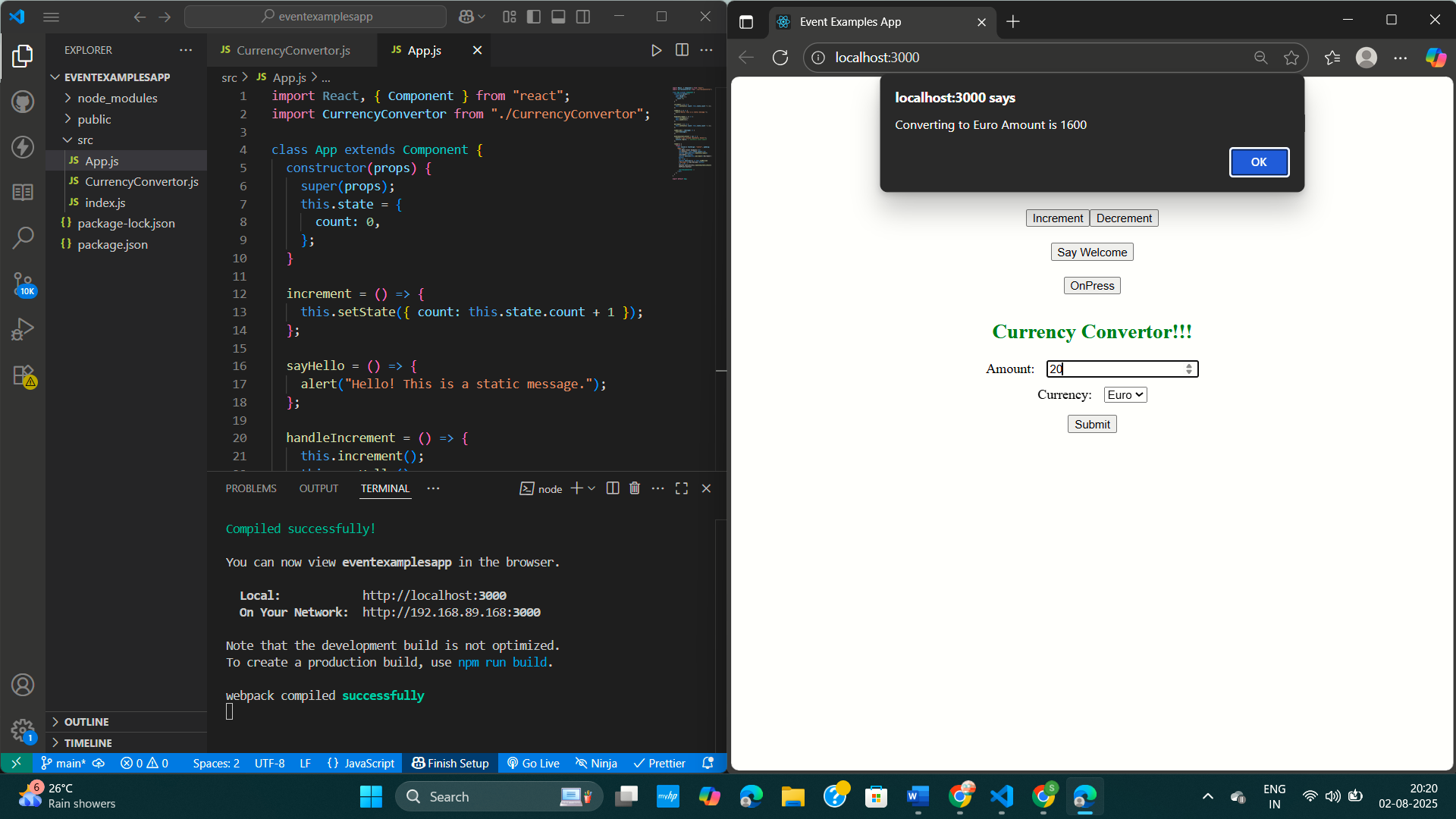
    <div id="root"></div>

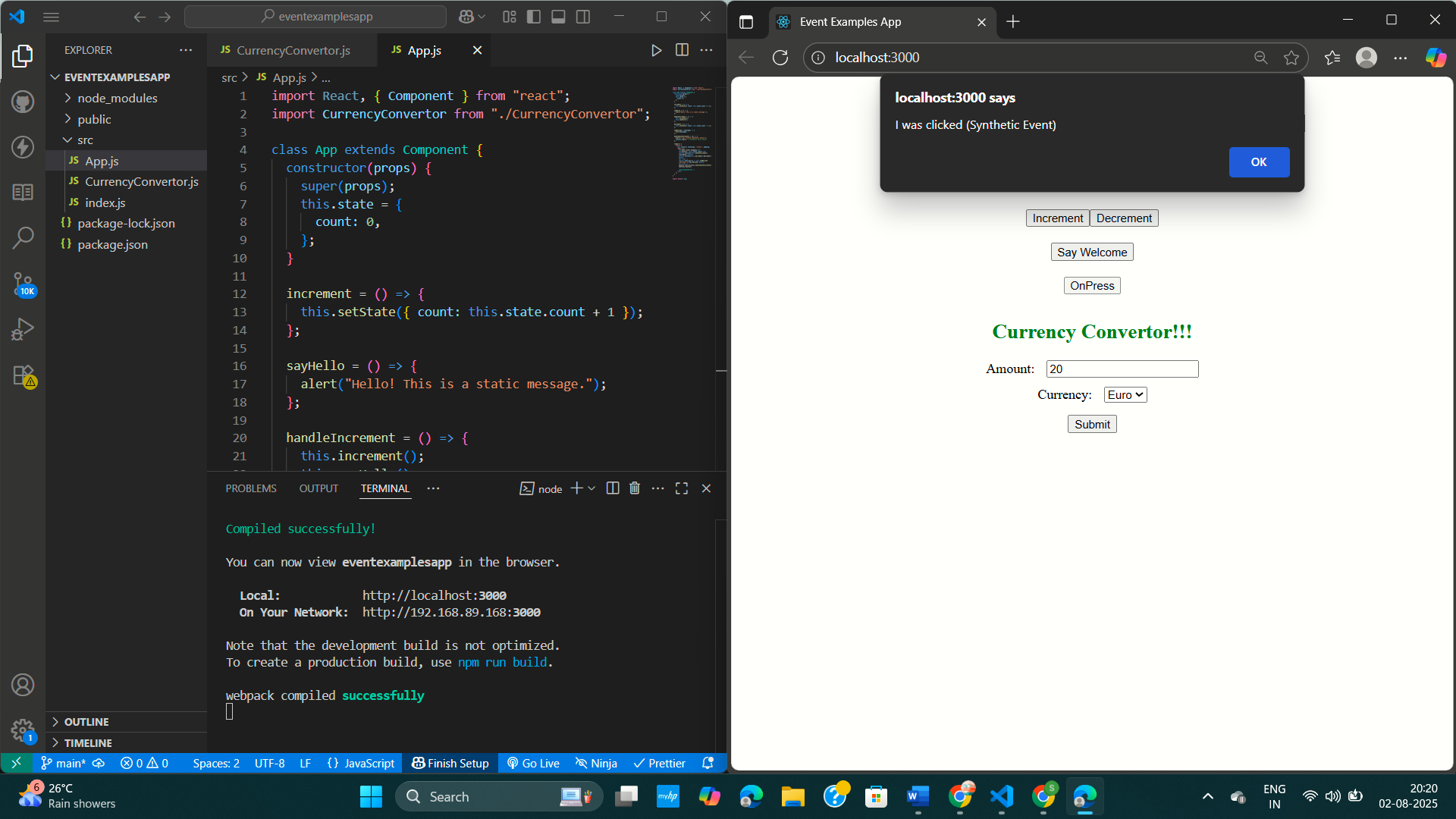
  </body>

</html>

**OUTPUT :**







**Exercise 4 : React Application – TicketBooking App**

**PROBLEM STATEMENT :**

Build a React application **ticketbookingapp** that demonstrates conditional rendering. Show a **Guest Page** with flight details when the user is logged out, and a **User Page** with ticket booking options when the user is logged in. Toggle between these pages using **Login** and **Logout** buttons.

**CODE :**

**App.js**

import React, { useState } from "react";

function GuestPage() {

  return (

    <div style={{ textAlign: "center", padding: "20px" }}>

      <h2>Welcome Guest</h2>

      <ul style={{ listStyleType: "none" }}>

        <li>Flight A - New Delhi to Mumbai  -  ₹5000</li>

        <li>Flight B - Chennai to Bangalore  -  ₹3000</li>

        <li>Flight C - Hyderabad to Kolkata  -  ₹4500</li>

      </ul>

    </div>

  );

}

function UserPage() {

  return (

    <div style={{ textAlign: "center", padding: "20px" }}>

      <h2>Welcome User</h2>

      <p>You can now book your tickets:</p>

      <button>Book Flight A</button>

      <button>Book Flight B</button>

      <button>Book Flight C</button>

    </div>

  );

}

function App() {

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

  const handleLogin = () => {

    setIsLoggedIn(true);

  };

  const handleLogout = () => {

    setIsLoggedIn(false);

  };

  return (

    <div style={{ textAlign: "center" }}>

      <h1>Ticket Booking App</h1>

      {isLoggedIn ? <UserPage /> : <GuestPage />}

      <br />

      {isLoggedIn ? (

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

      ) : (

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

      )}

    </div>

  );

}

export default App;

**index.js**

import React from "react";

import ReactDOM from "react-dom/client";

import App from "./App";

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

root.render(

  <React.StrictMode>

    <App />

  </React.StrictMode>

);

**index.html**

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <title>Ticket Booking App</title>

  </head>

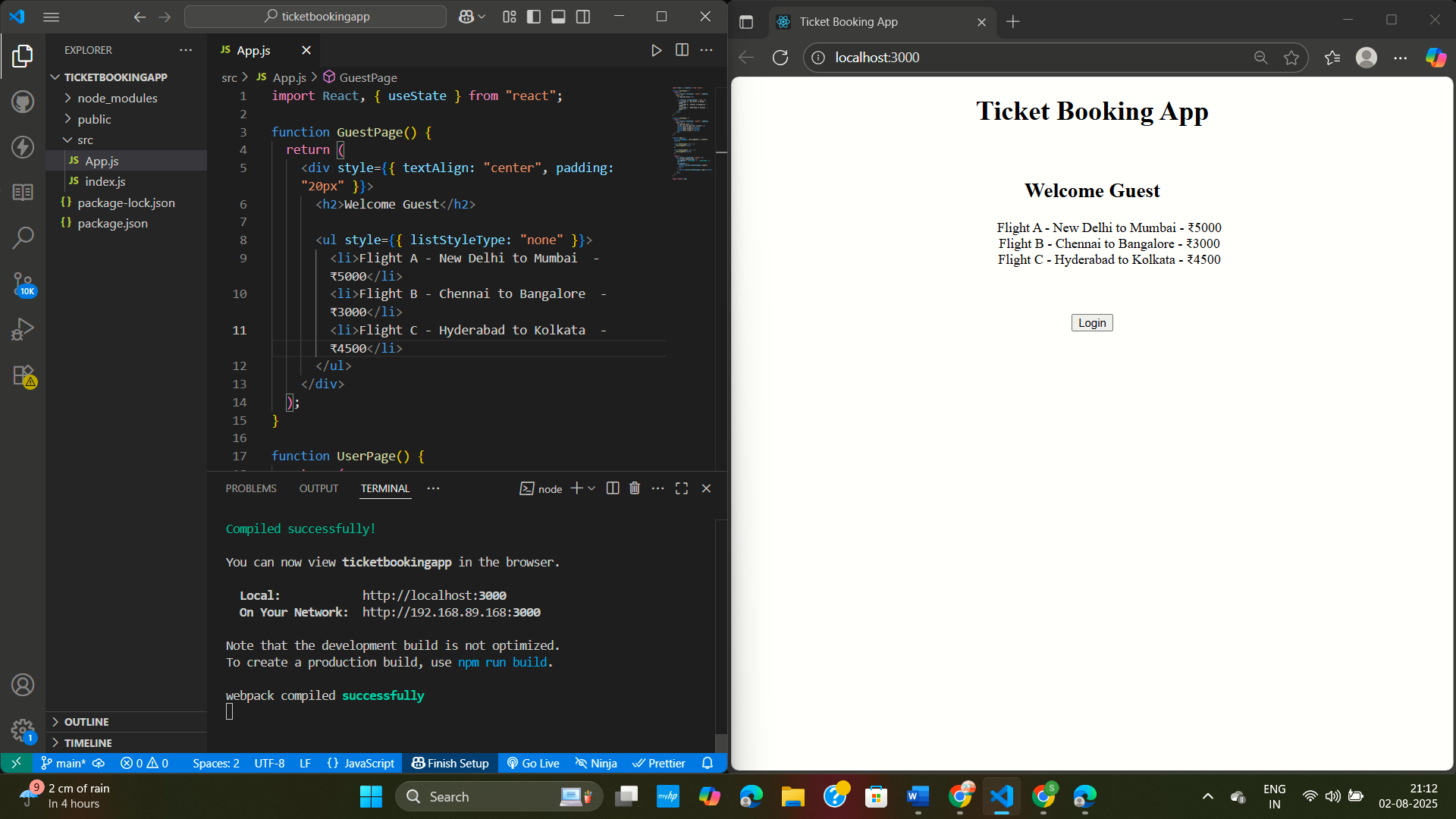
  <body>

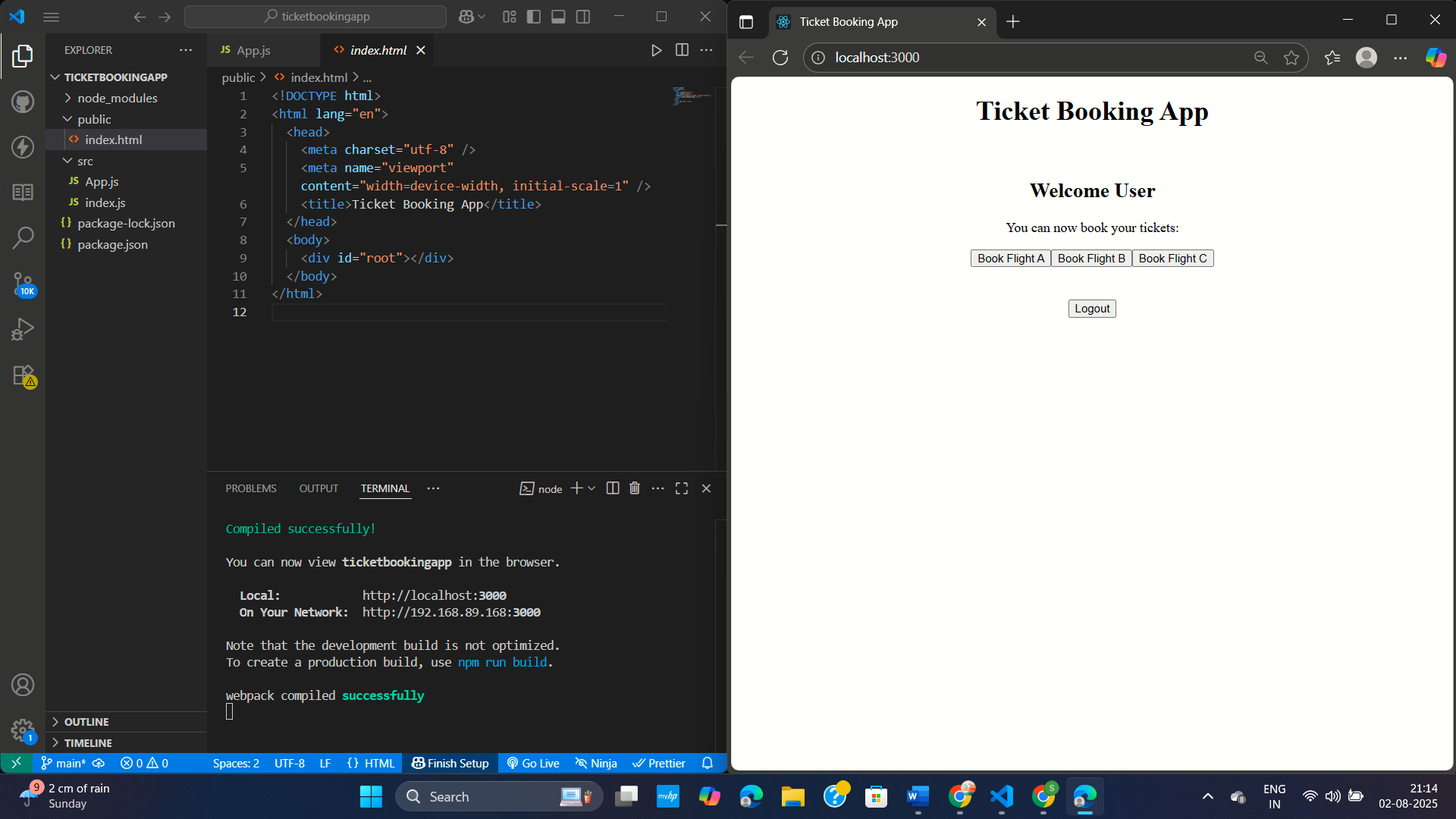
    <div id="root"></div>

  </body>

</html>

**OUTPUT :**





**Exercise 5 : React Application - Bloggerapp**

**PROBLEM STATEMENT :**

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 :**

**App.js**

import React, { useState } from "react";

import BookDetails from "./BookDetails";

import BlogDetails from "./BlogDetails";

import CourseDetails from "./CourseDetails";

function App() {

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

  const [showCourses, setShowCourses] = useState(true);

  // Conditional rendering with if-else

  const renderView = () => {

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

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

    return <CourseDetails />;

  };

  return (

    <div style={{ textAlign: "center" }}>

      <h1>Blogger App</h1>

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

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

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

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

      </div>

      {/\* if-else conditional rendering \*/}

      {renderView()}

      {/\* Ternary conditional rendering \*/}

      <div style={{ marginTop: "20px" }}>

        <button onClick={() => setShowCourses(!showCourses)}>

          Toggle Courses

        </button>

        {showCourses ? <CourseDetails /> : <p>Courses hidden!</p>}

      </div>

      {/\* Logical && conditional rendering \*/}

      <div style={{ marginTop: "20px" }}>

        {showCourses && <p>Courses are visible using && operator!</p>}

      </div>

    </div>

  );

}

export default App;

**BlogDetails.js**

import React from "react";

function BlogDetails() {

  const blogs = [

    { id: 1, title: "Understanding Conditional Rendering", date: "2025-08-01" },

    { id: 2, title: "React Hooks in Depth", date: "2025-08-02" }

  ];

  return (

    <div style={{ padding: "10px" }}>

      <h2>Blog Details</h2>

      <ul>

        {blogs.map((blog) => (

          <li key={blog.id}>

            {blog.title} - {blog.date}

          </li>

        ))}

      </ul>

    </div>

  );

}

export default BlogDetails;

**BookDetails.js**

import React from "react";

function BookDetails() {

  const books = [

    { id: 1, title: "React Basics", author: "John Doe" },

    { id: 2, title: "Learning JavaScript", author: "Jane Smith" },

    { id: 3, title: "Mastering CSS", author: "Mary Johnson" }

  ];

  return (

    <div style={{ padding: "10px" }}>

      <h2>Book Details</h2>

      <ul>

        {books.map((book) => (

          <li key={book.id}>

            {book.title} by {book.author}

          </li>

        ))}

      </ul>

    </div>

  );

}

export default BookDetails;

**CourseDetails.js**

import React from "react";

function CourseDetails() {

  const courses = [

    { id: 1, name: "React Development", duration: "3 Months" },

    { id: 2, name: "Full Stack JS", duration: "6 Months" }

  ];

  return (

    <div style={{ padding: "10px" }}>

      <h2>Course Details</h2>

      <ul>

        {courses.map((course) => (

          <li key={course.id}>

            {course.name} - {course.duration}

          </li>

        ))}

      </ul>

    </div>

  );

}

export default CourseDetails;

**index.js**

import React from "react";

import ReactDOM from "react-dom/client";

import App from "./App";

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

root.render(

  <React.StrictMode>

    <App />

  </React.StrictMode>

);

**index.html**

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <title>Blogger App</title>

  </head>

  <body>

    <div id="root"></div>

  </body>

</html>

**OUTPUT :**

