**WEEK – 07**

**ReactJS-HOL**

**Superset ID: 6262264**

**EXERCISE 12:**

**Objectives:**

**1. Explain about Conditional Rendering in React**

Conditional rendering in React refers to the process of displaying UI elements based on certain conditions (like if, else, or switch logic). Just like JavaScript, React allows us to use conditions inside JSX to determine what gets rendered to the DOM.

**Common techniques used:**

* Using if-else inside a function
* Ternary operator (condition ? true : false)
* Logical && operator

**Example:**

function Greeting(props) {

const isLoggedIn = props.isLoggedIn;

return (

<div>

{isLoggedIn ? <h1>Welcome back!</h1> : <h1>Please sign in.</h1>}

</div>

);

}

**2. Define Element Variables**

Element variables are used to store JSX elements in a variable. This is helpful when we want to conditionally assign different elements to be rendered later.

**Usage:**

* Makes code cleaner.
* Helps apply conditional logic outside of the JSX return statement.

**Example:**

function LoginControl(props) {

let button;

if (props.isLoggedIn) {

button = <LogoutButton />;

} else {

button = <LoginButton />;

}

return (

<div>

{button}

</div>

);

}

Here, the button variable is an **element variable** holding JSX.

**3. Explain How to Prevent Components from Rendering**

In React, we can prevent a component from rendering by returning null from its render method or from a function component. This is useful when a component shouldn't appear based on a condition.

**Example:**

function WarningBanner(props) {

if (!props.warn) {

return null; // Component won't render

}

return <div className="warning">Warning!</div>;

}

In this case, if warn is false, WarningBanner will not render anything to the DOM.

**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.**

***App.js:***

import React, { useState } from "react";

import LoginButton from "./LoginButton";

import LogoutButton from "./LogoutButton";

import GuestPage from "./GuestPage";

function App() {

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

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

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

  return (

    <div style={styles.container}>

      <div style={styles.box}>

        <Greeting isLoggedIn={isLoggedIn} />

        {isLoggedIn ? (

          <LogoutButton onClick={handleLogoutClick} />

        ) : (

          <LoginButton onClick={handleLoginClick} />

        )}

      </div>

    </div>

  );

}

const styles = {

  container: {

    height: "100vh",

    display: "flex",

    justifyContent: "center",

    alignItems: "center",

  },

  box: {

    textAlign: "center",

  },

};

export default App;

***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;

***GuestPage.js:***

function GuestPage(props) {

  const isLoggedIn = props.isLoggedIn;

  if (isLoggedIn) {

    return <h1>Welcome back</h1>;

  }

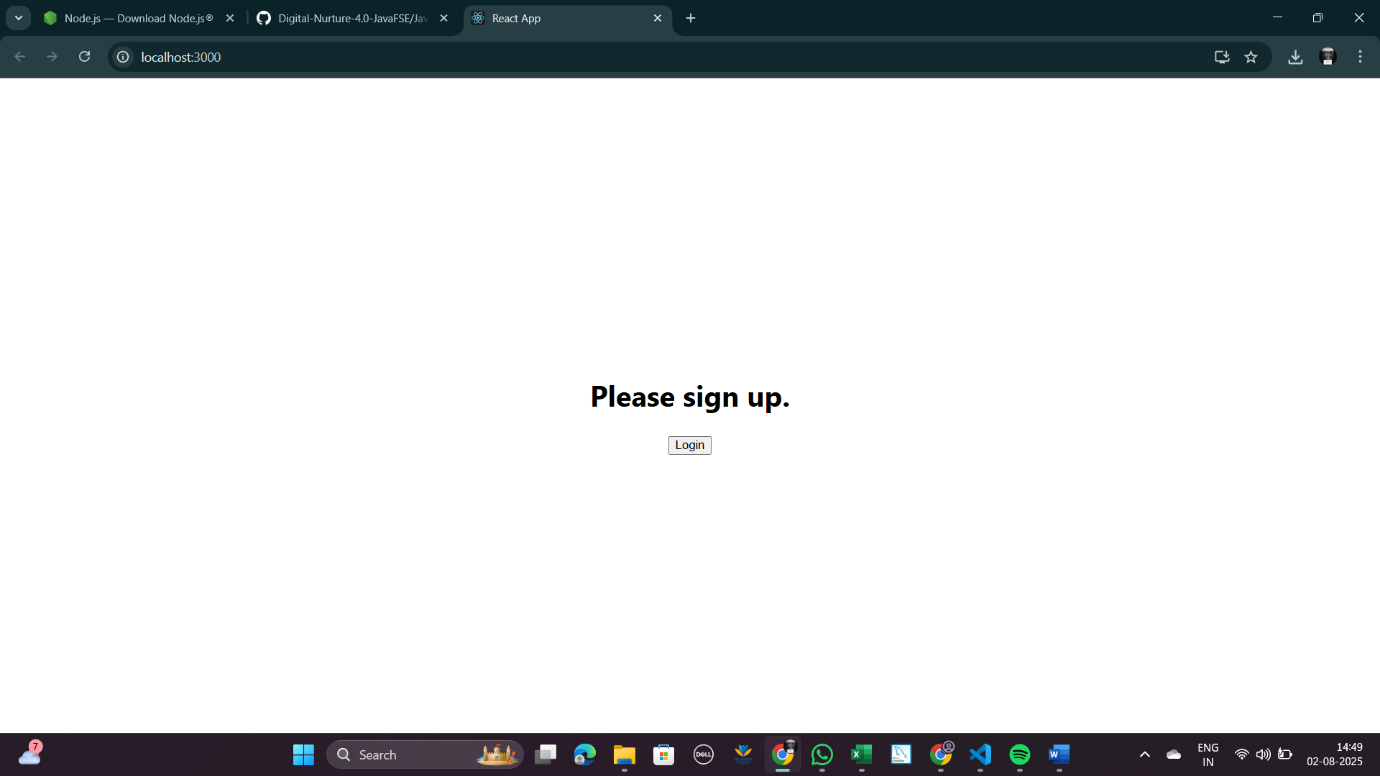
  return <h1>Please sign up.</h1>;

}

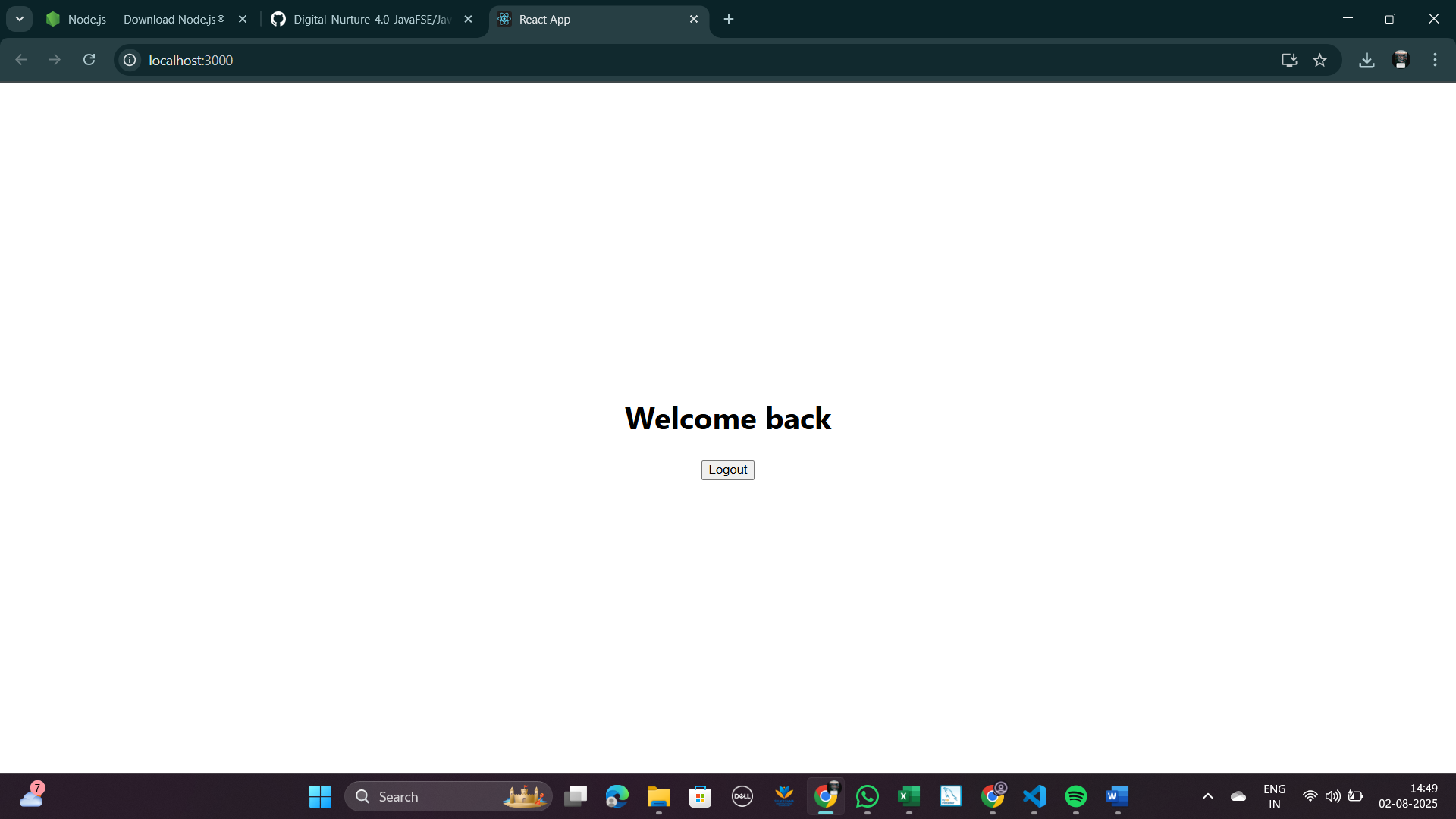
export default GuestPage;

**OUTPUT:**

**LOGIN PAGE**



**USER PAGE:**



**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**