**JAVA FSE WEEK 6**

**Reactjs-HOL**

**Mandatory Hands-on Excercises**

**1 Hands-on : My First React App**

**Objective:**

Display “Welcome to the first session of React” in a React application.

**Steps:**

1. npm install -g create-react-app
2. npx create-react-app myfirstreact
3. cd myfirstreact
4. Modify App.js
5. npm start

**Code :**

App.js

import React from 'react';

function App() {

return (

<div>

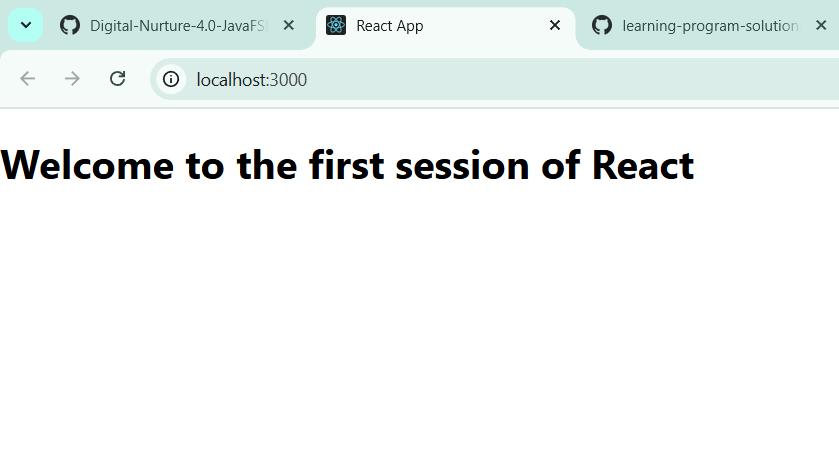
<h1>Welcome to the first session of React</h1>

</div>

);

}

export default App;

**OUTPUT:**

**2 Hands-on : StudentApp**

**Objective**

Create a React application named StudentApp to display three components:

* Home — Welcome to the Home page of Student Management Portal
* About — Welcome to the About page of the Student Management Portal
* Contact — Welcome to the Contact page of the Student Management Portal

**Code Files**

**src/Components/Home.js:**

import React from 'react';

function Home() {

return <h2>Welcome to the Home page of Student Management Portal</h2>;

}

export default Home;

**src/Components/About.js**

import React from 'react';

function About() {

return <h2>Welcome to the About page of the Student Management Portal</h2>;

}

export default About;

**src/Components/Contact.js**

import React from 'react';

function Contact() {

return <h2>Welcome to the Contact page of the Student Management Portal</h2>;

}

export default Contact;

**src/App.js**

import React from 'react';

import Home from './Components/Home';

import About from './Components/About';

import Contact from './Components/Contact';

function App() {

return (

<div>

<Home />

<About />

<Contact />

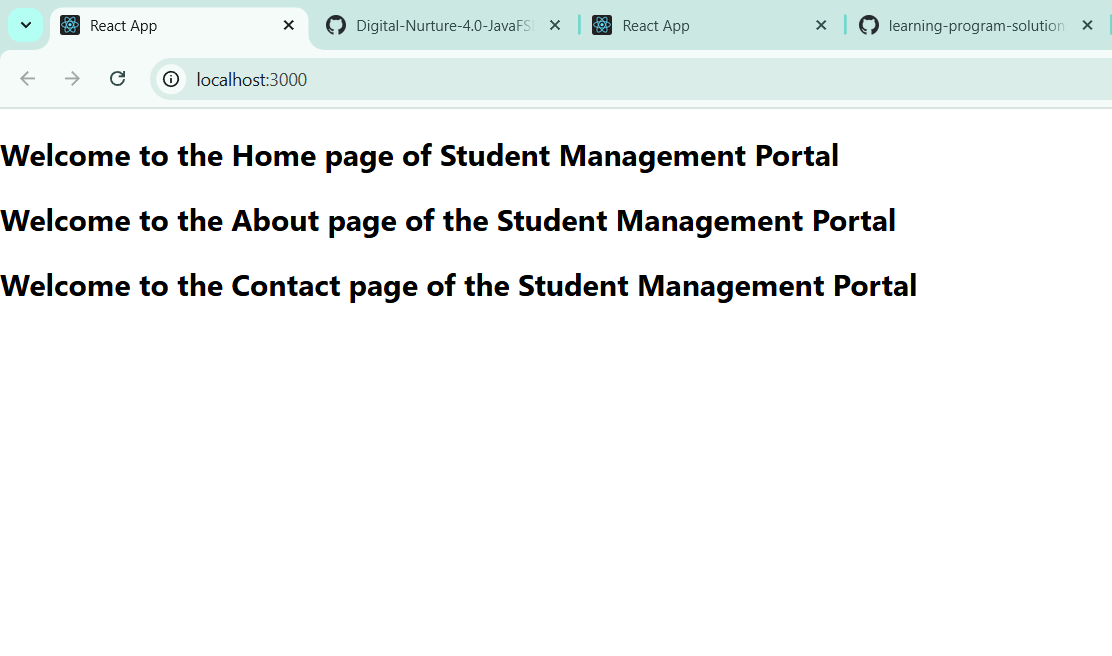
</div>

);

}

export default App;

**OUTPUT:**



**3 Hands-on: scorecalculatorapp**

**🎯 Objective**

Create a React app named scorecalculatorapp that uses a function component named CalculateScore.

**Step 1**: Create React App

npx create-react-app scorecalculatorapp

cd scorecalculatorapp

**Step 2:** Create Folders and Files

**src/Components/CalculateScore.js**

import React from 'react';

import '../Stylesheets/mystyle.css';

function CalculateScore(props) {

const average = props.Total / props.Goal;

return (

<div className="score-box">

<h2>Score Calculator</h2>

<p><strong>Name:</strong> {props.Name}</p>

<p><strong>School:</strong> {props.School}</p>

<p><strong>Total Marks:</strong> {props.Total}</p>

<p><strong>Goal:</strong> {props.Goal}</p>

<p><strong>Average Score:</strong> {average.toFixed(2)}</p>

</div>

);

}

export default CalculateScore;

**src/Stylesheets/mystyle.css:**

.score-box {

width: 350px;

padding: 20px;

border: 2px solid #333;

border-radius: 10px;

background-color: #f0f8ff;

margin: 30px auto;

text-align: left;

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

}

**App.js:**

import React from 'react';

import CalculateScore from './Components/CalculateScore';

function App() {

return (

<div>

<CalculateScore

Name="Shivani Vuyyala"

School="MVSR College"

Total={480}

Goal={6}

/>

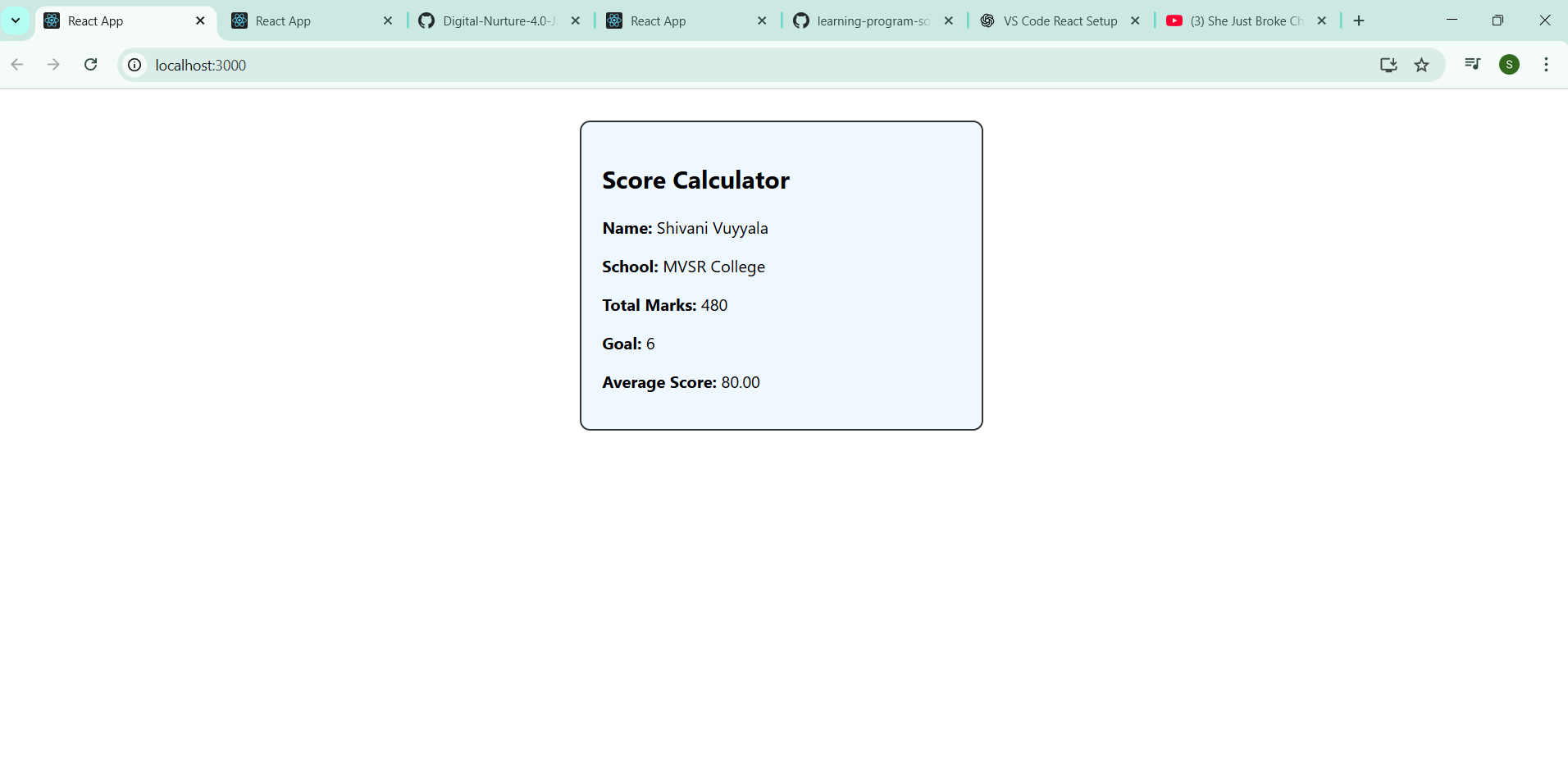
</div>

);

}

export default App;

**OUTPUT:**



**4 Hands-on: blogapp**

**Objective:**

Create a React app named blogapp to:

* Use class components
* Implement componentDidMount() to fetch data from API
* Handle errors with componentDidCatch()
* Render posts from API using state

API:  
 <https://jsonplaceholder.typicode.com/posts>

**Step 1:** Create React App

npx create-react-app blogapp

cd blogapp

**Step 2:** Create Files

**Posts.js**

import React, { Component } from 'react';

class Posts extends Component {

constructor(props) {

super(props);

this.state = {

posts: [],

hasError: false

};

}

loadPosts = () => {

fetch('https://jsonplaceholder.typicode.com/posts')

.then(response => response.json())

.then(data => this.setState({ posts: data }))

.catch(error => {

console.error("Error fetching posts:", error);

this.setState({ hasError: true });

});

}

componentDidMount() {

this.loadPosts();

}

componentDidCatch(error, info) {

alert("Something went wrong!");

console.error("Caught error:", error, info);

}

render() {

if (this.state.hasError) {

return <h2>Oops! Failed to load posts.</h2>;

}

return (

<div>

<h1>Blog Posts</h1>

{this.state.posts.slice(0, 5).map(post => (

<div key={post.id} style={{ border: "1px solid #ccc", margin: "10px", padding: "10px" }}>

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

<p>{post.body}</p>

</div>

))}

</div>

);

}

}

export default Posts;

**Step 3:** Add Posts Component to App.js

**App.js**

import React from 'react';

import Posts from './Posts';

function App() {

return (

<div className="App">

<Posts />

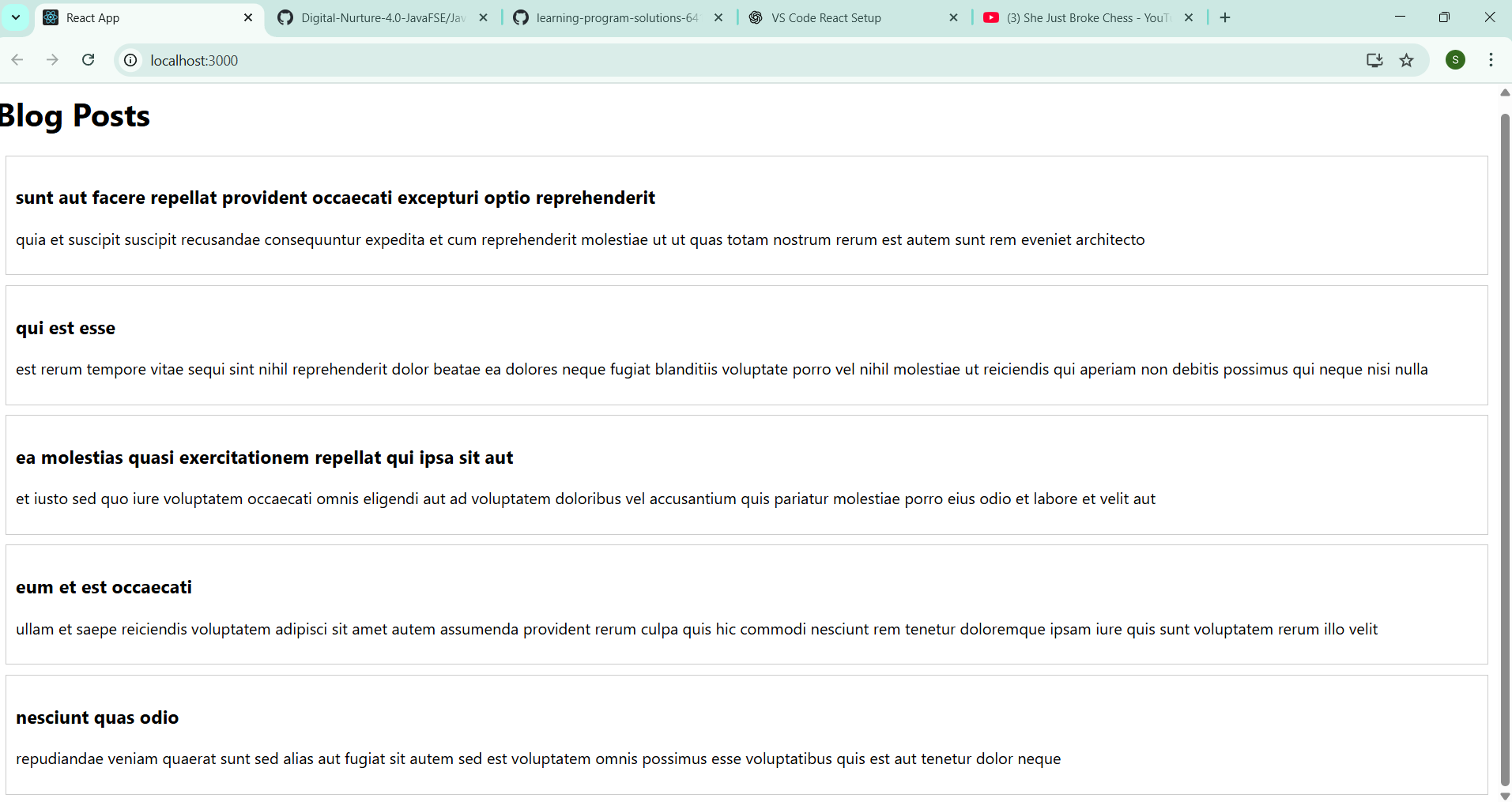
</div>

);

}

export default App;

**OUTPUT:**

****

**5th Hands-on: Styling Cohort Dashboard**

**Objective**

You are tasked with styling a React component called CohortDetails using:

* CSS Module (CohortDetails.module.css)
* Inline styles (conditional styling of <h3> based on status)

**Step 1:** Set Up React App

npx create-react-app cohort-dashboard

cd cohort-dashboard

**Step 2:** Create Component & CSS Module

**Src/Components/CohortDetails.js:**

import React from 'react';

import styles from './CohortDetails.module.css';

function CohortDetails({ name, status, startDate, endDate }) {

const headingStyle = {

color: status === 'ongoing' ? 'green' : 'blue',

};

return (

<div className={styles.box}>

<h3 style={headingStyle}>{name}</h3>

<dl>

<dt>Status:</dt>

<dd>{status}</dd>

<dt>Start Date:</dt>

<dd>{startDate}</dd>

<dt>End Date:</dt>

<dd>{endDate}</dd>

</dl>

</div>

);

}

export default CohortDetails;

**Src/Components/** **CohortDetails.module.css:**

.box {

width: 300px;

display: inline-block;

margin: 10px;

padding: 10px 20px;

border: 1px solid black;

border-radius: 10px;

}

dt {

font-weight: 500;

}

**App.js:**

import React from 'react';

import CohortDetails from './Components/CohortDetails';

function App() {

return (

<div className="App">

<CohortDetails

name="AI & ML - Cohort 1"

status="ongoing"

startDate="01-Jan-2025"

endDate="30-Jun-2025"

/>

<CohortDetails

name="Web Dev - Cohort 2"

status="completed"

startDate="01-Feb-2025"

endDate="31-Jul-2025"

/>

</div>

);

}

export default App;

**OUTPUT:**

