**1.ReactJS**

**Assignment Goals:**

#### **What is SPA**

* **SPA (Single Page Application)** is a web app that dynamically updates a single HTML page without reloading the whole page.

**Benefits:**

* Faster transitions
* Seamless user experience
* Less network usage

#### **What is React?**

* React is a **JavaScript library** by Meta (Facebook) for building fast, interactive UIs.
* Uses **components** and **virtual DOM**.

#### **What is Virtual DOM?**

* A lightweight copy of the actual DOM.
* React uses it to optimize updates without touching real DOM directly.

#### **Features of React:**

* Component-Based
* Virtual DOM
* Reusable code
* One-Way Data Binding
* Fast rendering

1. To create a new React app, Install Nodejs and Npm from the following link: <https://nodejs.org/en/download/>

2. Install Create-react-app by running the following command in the command prompt:

3. To create a React Application with the name of “myfirstreact”, type the following command:

4. Once the App is created, navigate into the folder of myfirstreact by typing the following command:

5. Open the folder of myfirstreact in Visual Studio Code

6. Open the App.js file in Src Folder of myfirstreact

7. Remove the current content of “App.js”

8. Replace it with the following:

**App.js**

import logo from './logo.svg';

import './App.css';

function App() {

return (

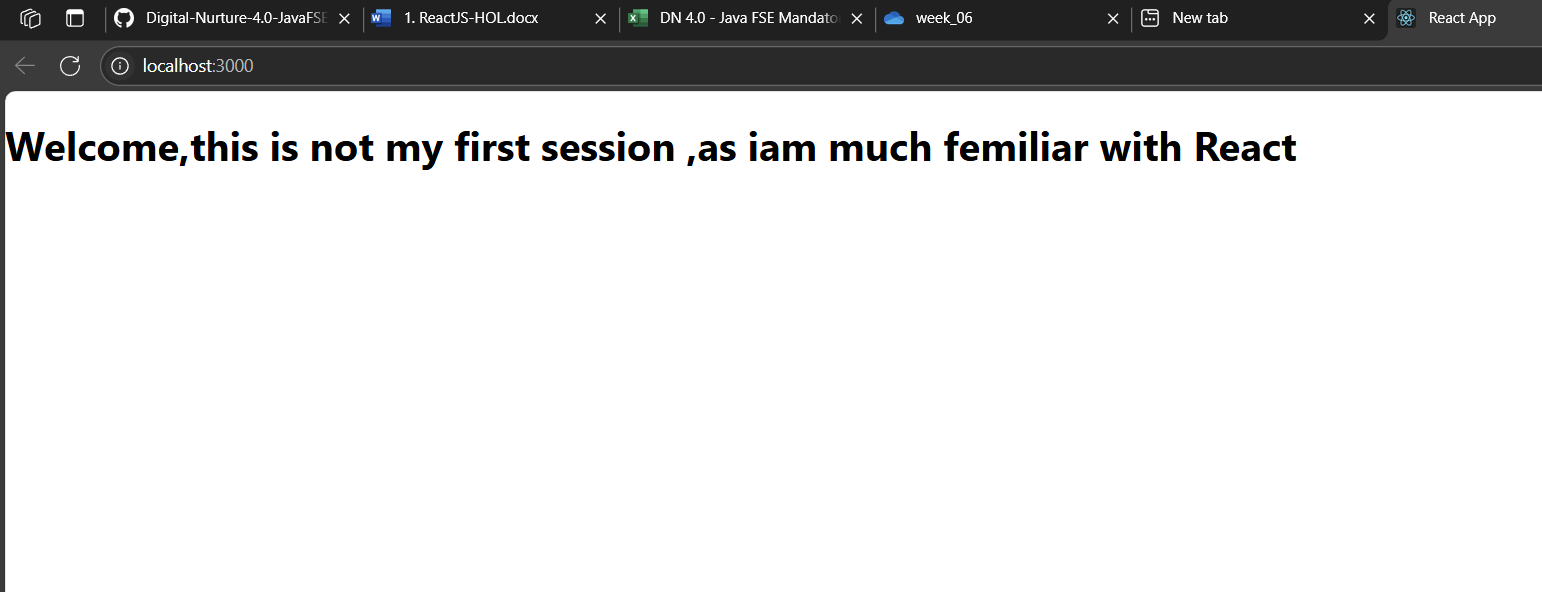
<h1>Welcome,this is not my first session ,as iam much femiliar with React </h1>

);

}

export default App;

Output:



**2.ReactJS**

### **What are React Components?**

* Reusable pieces of UI (like LEGO blocks).
* Components can be **class-based** or **function-based**.

### **Differences: Component vs JavaScript Function**

|  |  |
| --- | --- |
| **React Component** | **JavaScript Function** |
| Returns JSX | Returns plain JS value |
| Can manage state & lifecycle | Cannot (unless React hooks used) |
| Used for UI | Used for logic |

### **Types of Components**

1. **Class Component** – Uses ES6 class, supports lifecycle methods
2. **Function Component** – Simpler, uses Hooks for state/lifecycle

**Home.js**

import React, { Component } from 'react';  
  
class Home extends Component {  
 render() {  
 return (  
 <div>  
 <h2>Welcome to the Home page of Student Management Portal</h2>  
 </div>  
 );  
 }  
}  
  
export default Home;

**About.js**

import React, { Component } from 'react';  
  
class About extends Component {  
 render() {  
 return (  
 <div>  
 <h2>Welcome to the About page of the Student Management Portal</h2>  
 </div>  
 );  
 }  
}  
  
export default About;

**Contact.js**

import React, { Component } from 'react';  
  
class Contact extends Component {  
 render() {  
 return (  
 <div>  
 <h2>Welcome to the Contact page of the Student Management Portal</h2>  
 </div>  
 );  
 }  
}  
  
export default Contact;

**App.js**

import React from 'react';

import './App.css';

import Home from './Components/Home';

import About from './Components/About';

import Contact from './Components/Contact';

function App() {

return (

<div className="App">

<h1>Student Management Portal</h1>

<Home />

<About />

<Contact />

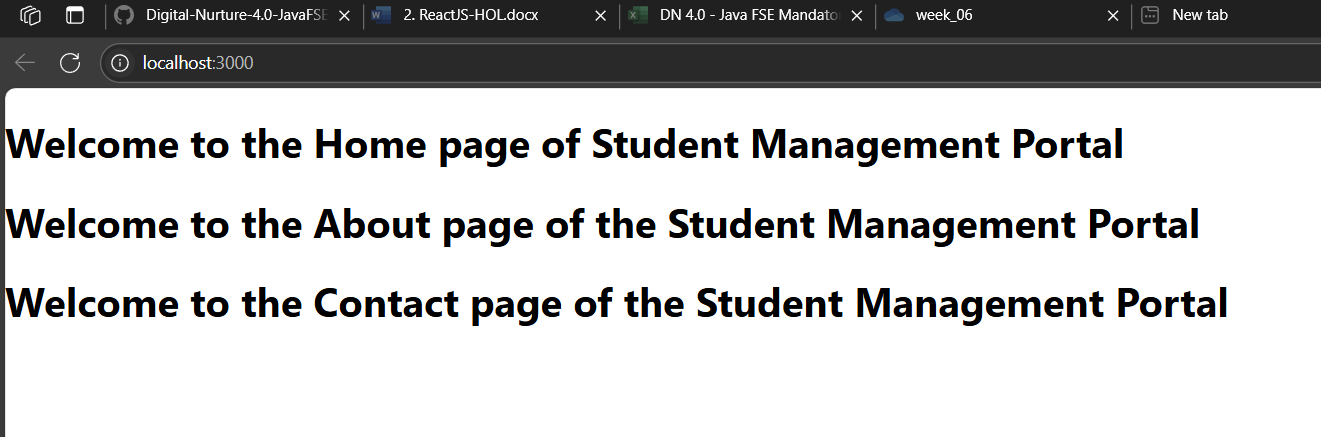
</div>

);

}

export default App;

**Output:**



**3.ReactJS**

**CalculateScore.js**

import React from 'react';

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

export default function CalculateScore(props) {

const { name, school, total, goal } = props;

const average = total / goal;

return (

<div className="score-container">

<h2>Student Score Details</h2>

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

<p><strong>School:</strong> {school}</p>

<p><strong>Total Marks:</strong> {total}</p>

<p><strong>Goal Subjects:</strong> {goal}</p>

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

</div>

)

}

**mystyle.css**

.score-container {

background-color: #f0f8ff;

padding: 20px;

margin: 20px auto;

width: 60%;

border-radius: 10px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

.score-container h2 {

color: #003366;

text-align: center;

}

.score-container p {

font-size: 16px;

margin: 8px 0;

}

.average {

color: green;

font-weight: bold;

}

CalculateScore.js

import React from 'react';

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

export default function CalculateScore(props) {

const { name, school, total, goal } = props;

const average = total / goal;

return (

<div className="score-container">

<h2>Student Score Details</h2>

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

<p><strong>School:</strong> {school}</p>

<p><strong>Total Marks:</strong> {total}</p>

<p><strong>Goal Subjects:</strong> {goal}</p>

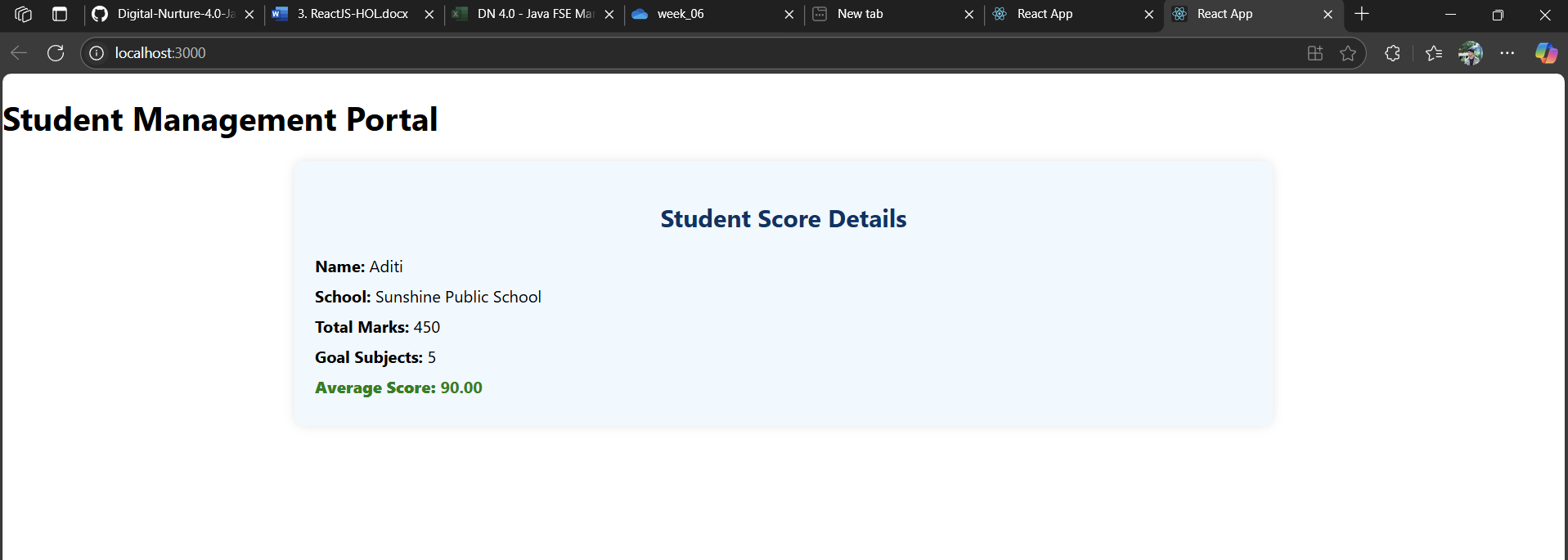
<p className="average"><strong>Average Score:</strong> {average.toFixed(2)}</p>

</div>

)

}

Output:



**4.ReactJS**

**Post.js**

import React from 'react'

export default function Post(props) {

return (

<div style={{ padding: '10px', borderBottom: '1px solid #ccc' }}>

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

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

</div>

)

}

**Posts.js**

import React, { Component } from 'react';

import Post from './Post';

class Posts extends Component {

constructor(props) {

super(props);

this.state = {

posts: [],

hasError: false,

errorMessage: ''

};

}

componentDidMount() {

this.loadPosts();

}

loadPosts = () => {

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

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

.then((data) => {

console.log("Fetched posts:", data);

this.setState({ posts: data });

})

.catch((error) => {

this.setState({ hasError: true, errorMessage: error.message });

});

};

componentDidCatch(error, info) {

this.setState({ hasError: true, errorMessage: error.message });

alert("Something went wrong: " + error.message);

}

render() {

const { posts, hasError, errorMessage } = this.state;

if (hasError) {

return <div>Error Occurred: {errorMessage}</div>;

}

return (

<div>

<h2>Blog Posts</h2>

{Array.isArray(posts) && posts.map((post) => (

<Post key={post.id} title={post.title} body={post.body} />

))}

</div>

);

}

}

export default Posts;

**App.js**

import React from 'react';

import './App.css';

import Posts from './Posts';

function App() {

return (

<div className="App">

<h1>Welcome to BlogApp</h1>

<Posts />

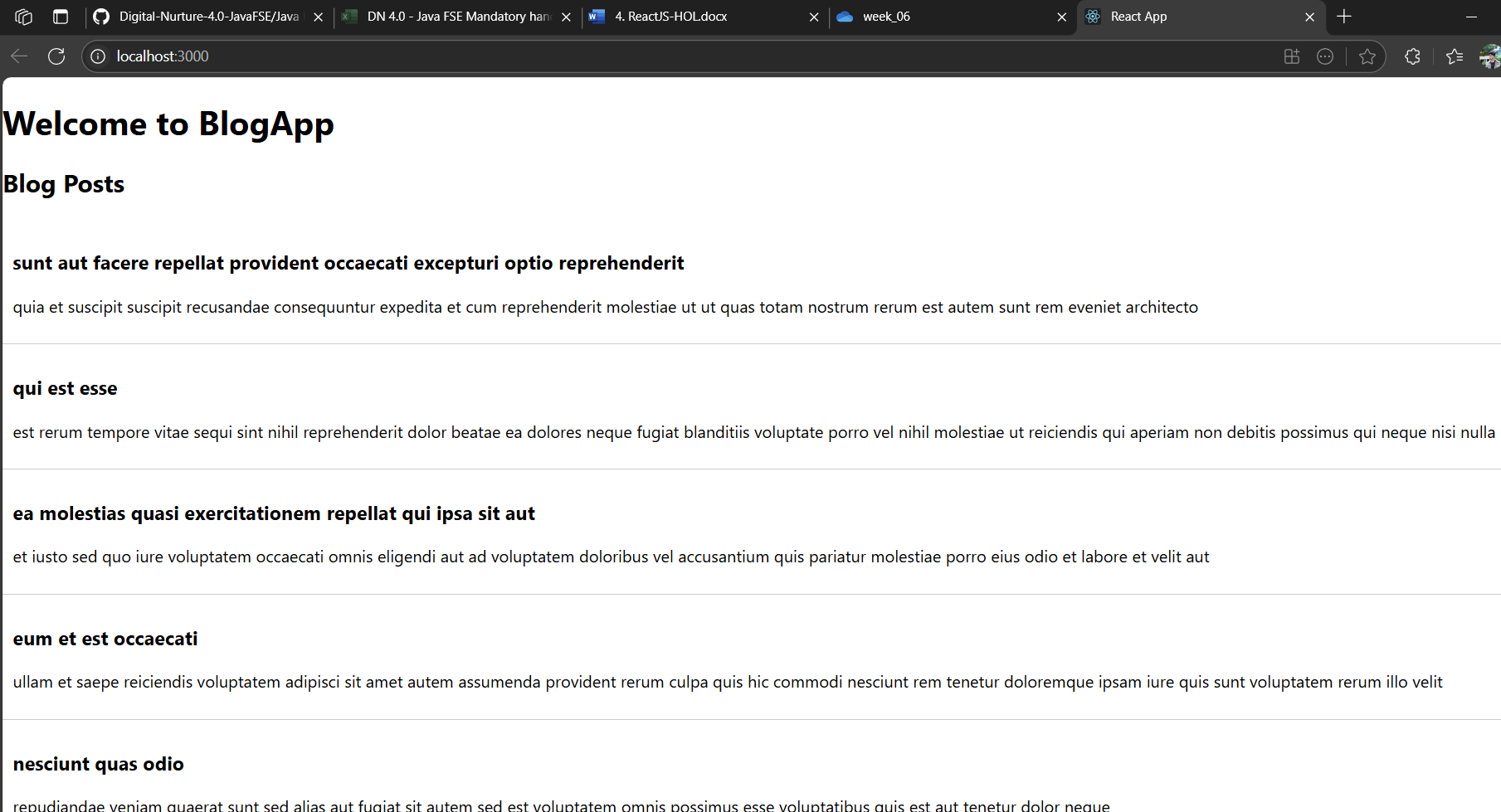
</div>

);

}

export default App;

Output:



**5.ReactJS**

CohortDetails.js

// CohortDetails.js

import React from 'react';

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

const CohortDetails = ({ name, batch, status }) => {

const headingStyle = {

color: status.toLowerCase() === 'ongoing' ? 'green' : 'blue'

};

return (

<div className={styles.box}>

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

<dl>

<dt>Batch</dt>

<dd>{batch}</dd>

<dt>Status</dt>

<dd>{status}</dd>

</dl>

</div>

);

};

export default CohortDetails;

App.js

// App.js

import React from 'react';

import CohortDetails from './Components/CohortDetails';

function App() {

return (

<div>

<h2>My Academy Dashboard</h2>

<CohortDetails name="React Bootcamp" batch="May 2025" status="Ongoing" />

<CohortDetails name="Spring Boot Cohort" batch="April 2025" status="Completed" />

</div>

);

}

export default App;

Module.css

/\* 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; }

Output:

