等线1.React:

//App.js

import React from 'react';

function App() {

return (

<div>

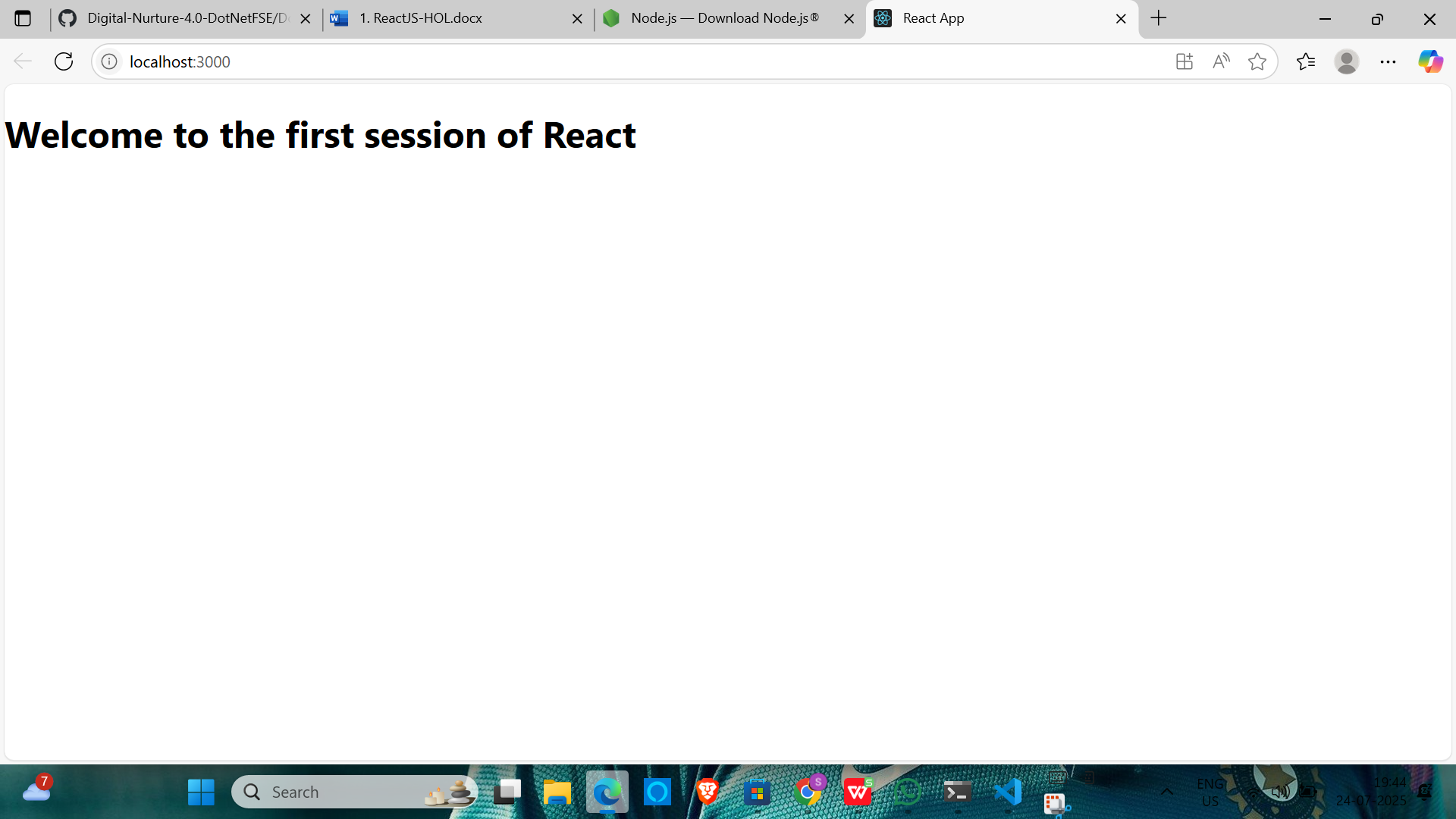
<h1>Welcome to the first session of React</h1>

</div>

);

}

export default App;



2.React:

//About.js

// src/Components/About.js

import React from 'react';

function About() {

return (

<div>

<h1>Welcome to the About page of the Student Management Portal</h1>

</div>

);

}

export default About;

//Contact.js

// src/Components/Contact.js

import React from 'react';

function Contact() {

return (

<div>

<h1>Welcome to the Contact page of the Student Management Portal</h1>

</div>

);

}

export default Contact;

//Home.js

// src/Components/Home.js

import React from 'react';

function Home() {

return (

<div>

<h1>Welcome to the Home page of Student Management Portal</h1>

</div>

);

}

export default Home;

//App.js

// 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 />

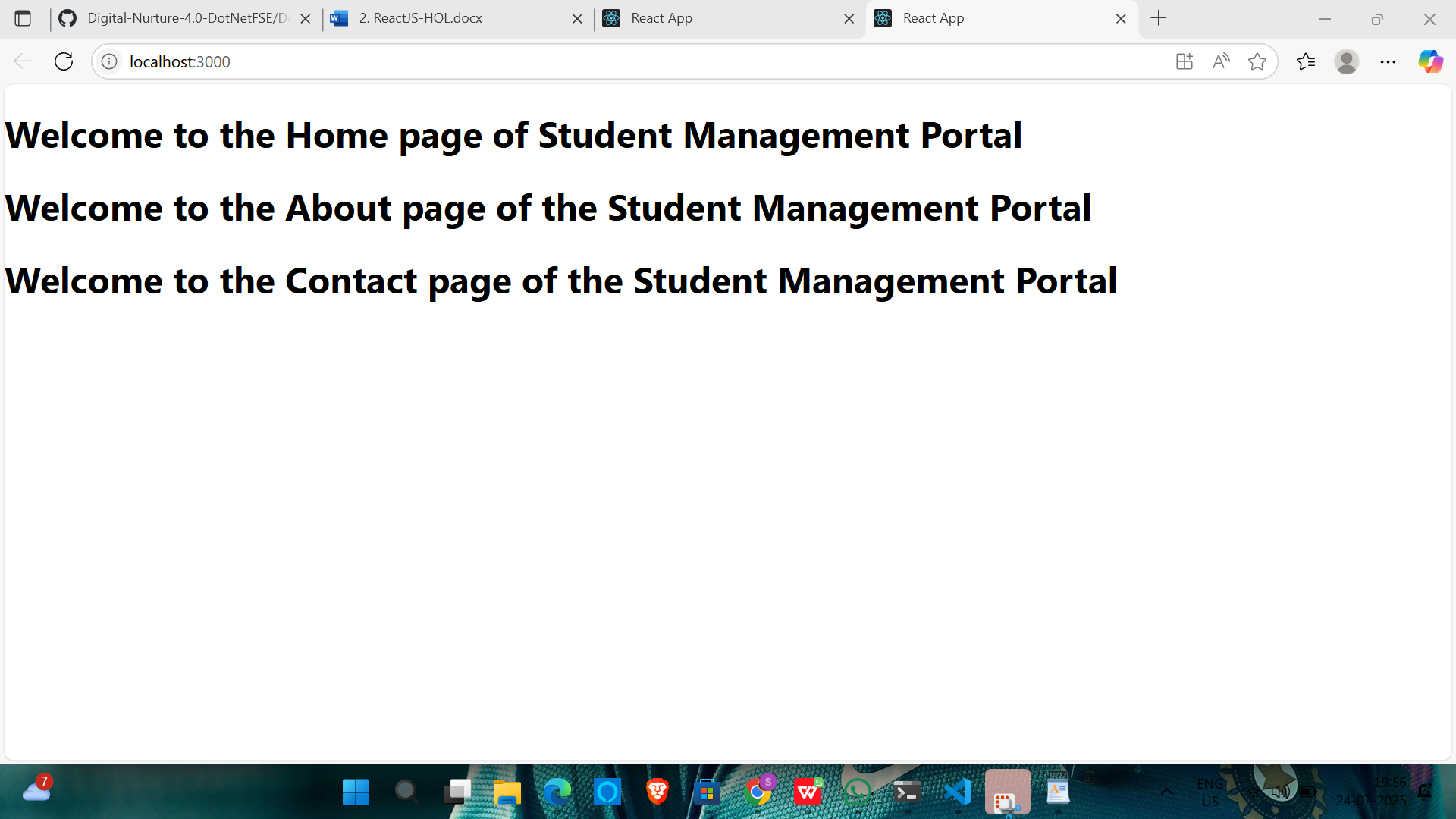
</div>

);

}

export default App;

OUTPUT:



3.React

//App.js

// src/App.js

import React from 'react';

import CalculateScore from './Components/CalculateScore';

function App() {

return (

<div>

<CalculateScore />

</div>

);

}

export default App;

//mystyle.js

/\* src/Stylesheets/mystyle.css \*/

.container {

width: 400px;

margin: 40px auto;

padding: 20px;

border: 2px solid #007acc;

border-radius: 10px;

background-color: #f0f8ff;

font-family: Arial, sans-serif;

}

.container h1 {

text-align: center;

color: #007acc;

}

input {

width: 100%;

margin: 8px 0;

padding: 10px;

font-size: 16px;

}

button {

width: 100%;

padding: 10px;

background-color: #007acc;

color: white;

font-size: 16px;

border: none;

margin-top: 10px;

cursor: pointer;

border-radius: 5px;

}

button:hover {

background-color: #005f99;

}

.result {

margin-top: 20px;

background-color: #e6f7ff;

padding: 10px;

border-radius: 5px;

}

//CalculateScore.js

// src/Components/CalculateScore.js

import React, { useState } from 'react';

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

function CalculateScore() {

const [name, setName] = useState('');

const [school, setSchool] = useState('');

const [total, setTotal] = useState('');

const [goal, setGoal] = useState('');

const [average, setAverage] = useState(null);

const calculateAverage = () => {

if (total && goal) {

const avg = Number(total) / Number(goal);

setAverage(avg.toFixed(2));

} else {

setAverage(null);

}

};

return (

<div className="container">

<h1>Student Score Calculator</h1>

<input

type="text"

placeholder="Enter Name"

value={name}

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

/>

<input

type="text"

placeholder="Enter School"

value={school}

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

/>

<input

type="number"

placeholder="Enter Total Marks"

value={total}

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

/>

<input

type="number"

placeholder="Enter Goal Subjects"

value={goal}

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

/>

<button onClick={calculateAverage}>Calculate Average</button>

{average && (

<div className="result">

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

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

<p><strong>Average Score:</strong> {average}</p>

</div>

)}

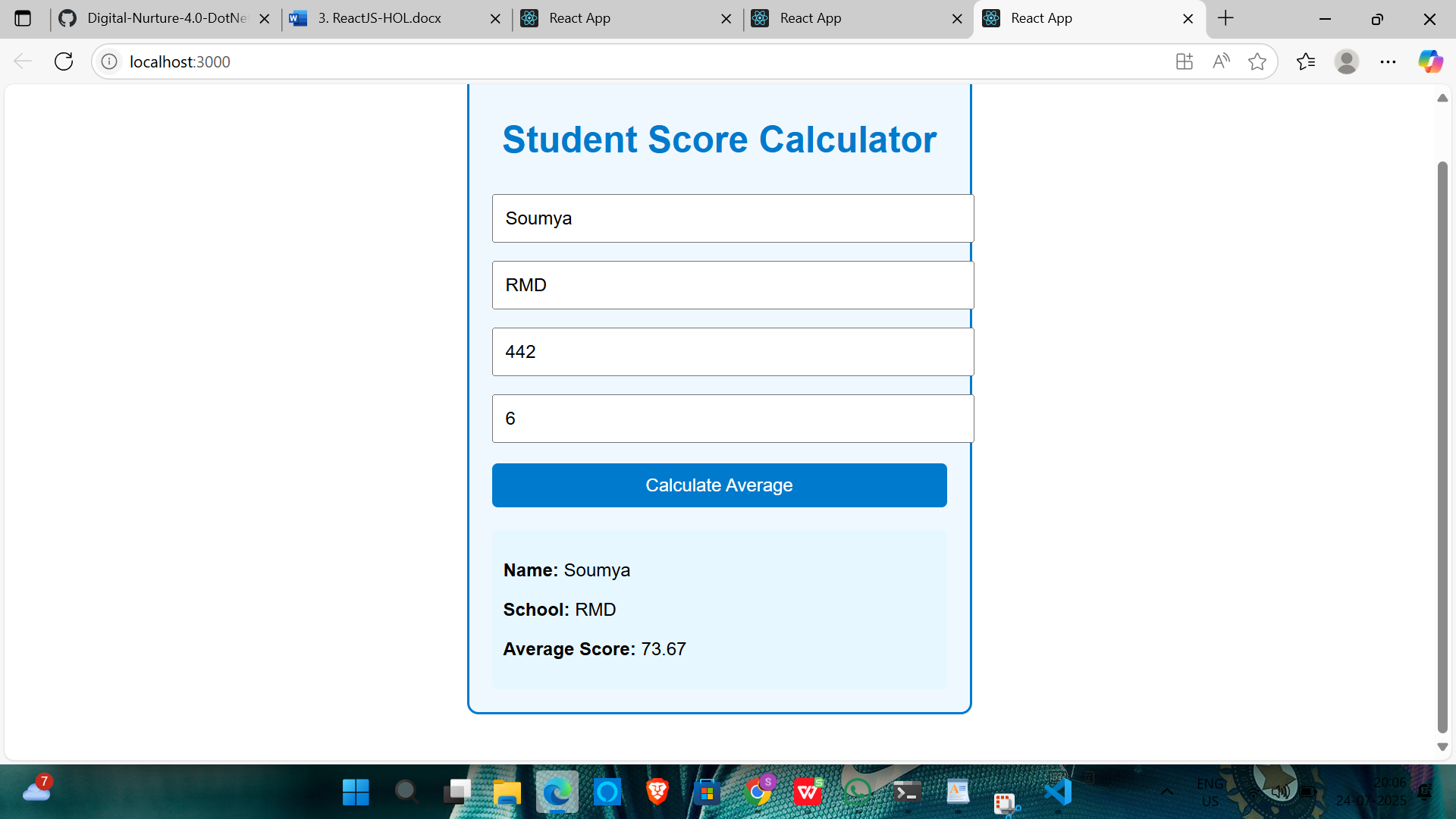
</div>

);

}

export default CalculateScore;

OUTPUT:



4.React

//post.js

// src/Post.js

import React from 'react';

function Post({ title, body }) {

return (

<div className="post">

<h2>{title}</h2>

<p>{body}</p>

<hr />

</div>

);

}

export default Post;

//Posts.js

// src/Posts.js

import React, { Component } from 'react';

import Post from './Post';

class Posts extends Component {

constructor(props) {

super(props);

this.state = {

posts: [],

error: null

};

}

// Step 6: Fetch posts

loadPosts() {

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

.then((response) => {

if (!response.ok) {

throw new Error('Failed to fetch posts');

}

return response.json();

})

.then((data) => {

this.setState({ posts: data });

})

.catch((error) => {

this.setState({ error });

});

}

// Step 7: Lifecycle method

componentDidMount() {

this.loadPosts();

}

// Step 9: Error boundary

componentDidCatch(error, info) {

alert('An error occurred: ' + error.message);

}

// Step 8: Render

render() {

const { posts, error } = this.state;

if (error) {

return <p style={{ color: 'red' }}>Error: {error.message}</p>;

}

return (

<div>

<h1>Blog Posts</h1>

{posts.map((post) => (

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

))}

</div>

);

}

}

export default Posts;

//App.js

// src/App.js

import React from 'react';

import Posts from './Posts';

function App() {

return (

<div className="App">

<Posts />

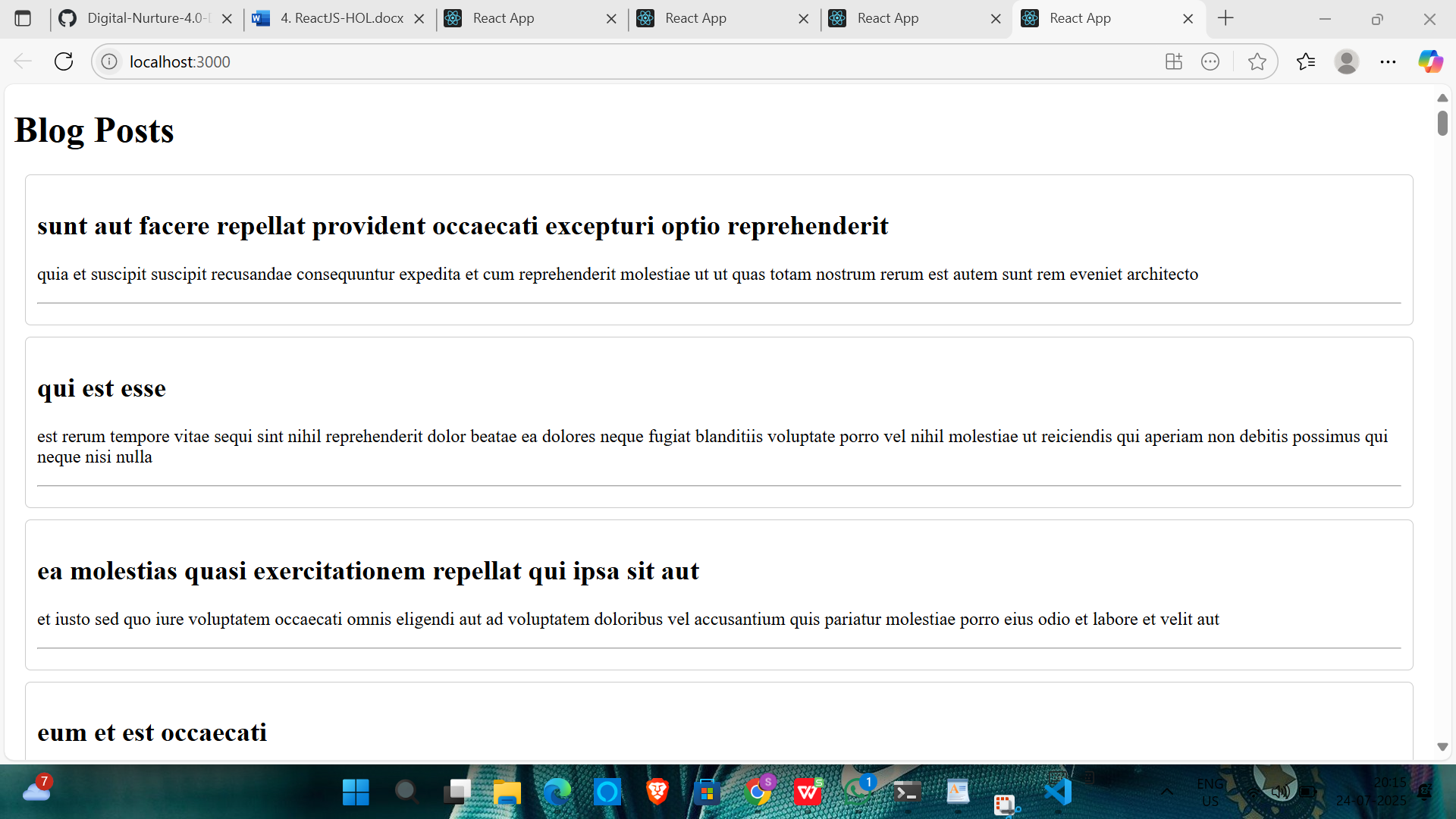
</div>

);

}

export default App;

OUTPUT:



5.React

//CohortDetails.js

import React from 'react';

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

const CohortDetails = ({ cohortName, status, startDate }) => {

return (

<div className={styles.box}>

<h3 style={{ color: status === "ongoing" ? "green" : "blue" }}>

{cohortName}

</h3>

<dl>

<dt>Status</dt>

<dd>{status}</dd>

<dt>Start Date</dt>

<dd>{startDate}</dd>

</dl>

</div>

);

};

export default CohortDetails;

//APP.js

// App.js

import React from 'react';

import CohortDetails from './CohortDetails';

function App() {

const cohorts = [

{ cohortName: "React 101", status: "ongoing", startDate: "2025-07-20" },

{ cohortName: "Java Advanced", status: "completed", startDate: "2025-05-15" },

{ cohortName: "DevOps Bootcamp", status: "ongoing", startDate: "2025-07-10" },

];

return (

<div>

<h2>Cohorts Dashboard</h2>

{cohorts.map((cohort, index) => (

<CohortDetails

key={index}

cohortName={cohort.cohortName}

status={cohort.status}

startDate={cohort.startDate}

/>

))}

</div>

);

}

export default App;

//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:

