# React HandsOn - 1

App.js

import './App.css';

function App() {

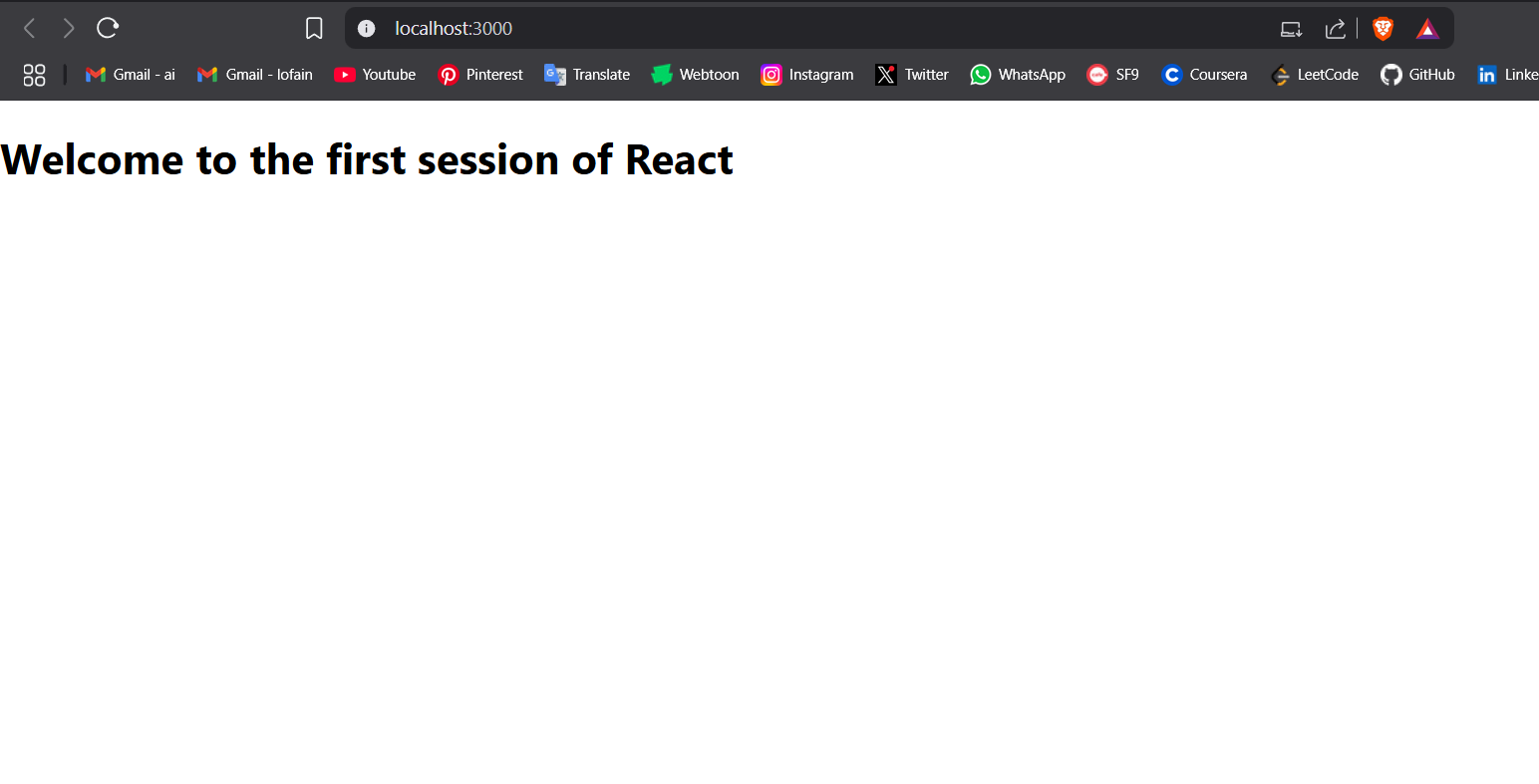
return (

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

);

}

export default App;

Output:  
  


# React HandsOn - 2

components/Home.js

import { Component } from 'react';

class Home extends Component {

render() {

return (

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

);

}

}

export { Home };

components/About.js:

import React, { Component } from 'react';

class About extends Component {

render() {

return (

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

);

}

}

export { About };

components/Contact.js:

import React, { Component } from 'react';

class Contact extends Component {

render() {

return (

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

);

}

}

export { Contact };

App.js:

import './App.css';

import { Home } from './components/Home';

import { About } from './components/About';

import { Contact } from './components/Contact';

function App() {

return (

<div className="container">

<Home />

<About />

<Contact />

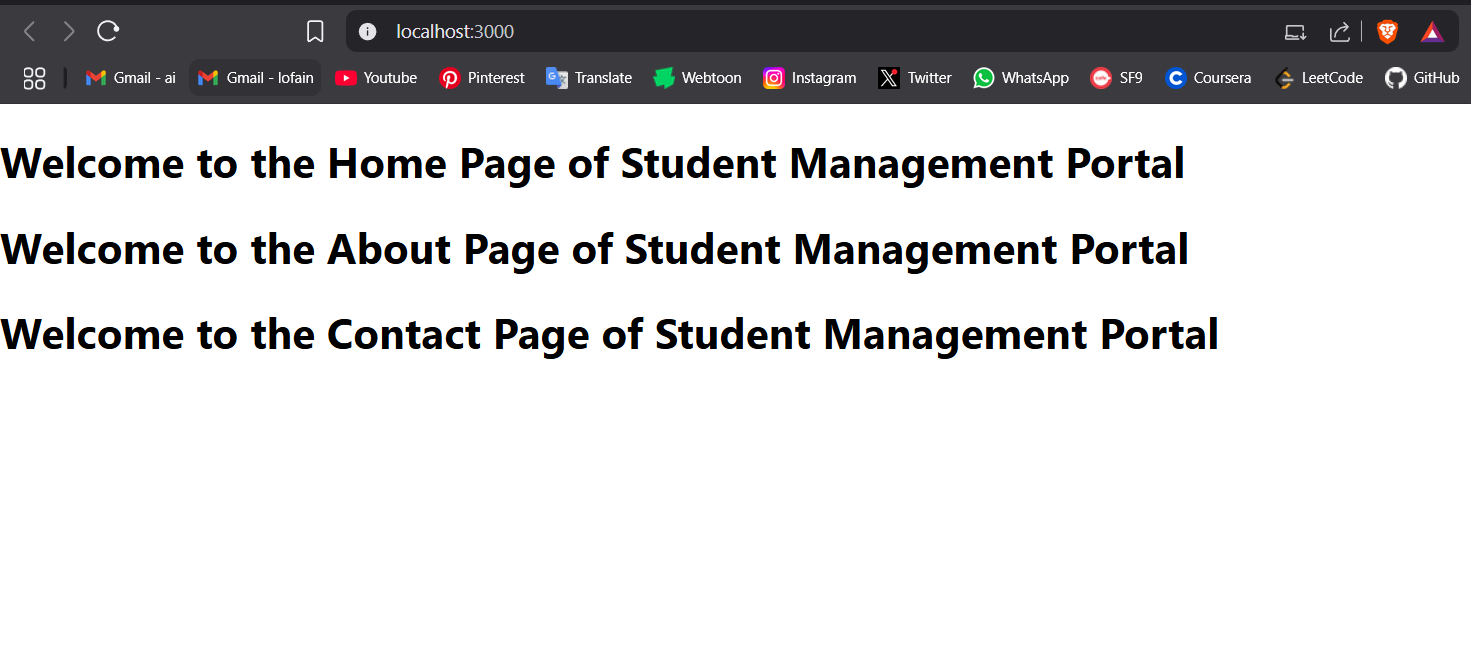
</div>

);

}

export default App;

Output:



# React HandsOn - 3

CalculateScore.js

import '../sylesheets/mystyle.css';

const percentToDecimal = (decimal) => {

return (decimal \* 100).toFixed(2) + '%';

};

const calcScore = (total, goal) => {

return percentToDecimal(total / goal);

};

export const CalculateScore = ({ Name, School, total, goal }) => (

<div className="formatstyle">

<h1>

<font color="Brown">Student Details:</font>

</h1>

<div className="Name">

<b>

<span> Name: </span>

</b>

<span>{Name}</span>

</div>

<div className="School">

<b>

<span> School: </span>

</b>

<span>{School}</span>

</div>

<div className="Total">

<b>

<span> Total:</span>

</b>

<span>{total}</span>

<span>Marks</span>

</div>

<div className="Score">

<b>Score:</b>

<span>{calcScore(total, goal)}</span>

</div>

</div>

);

stylesheets/mystyle.css:

.Name {

font-weight: 300;

color: blue;

}

.School {

color: crimson;

}

.Total {

color: darkmagenta;

}

.formatstyle {

text-align: center;

font-size: large;

}

.Score {

color: forestgreen;

}

App.js:

import CalculateScore from '../src/components/CalculateScore';

function App() {

return (

<div>

<CalculateScore

Name="Steeve"

School="DNV Public School"

total={284}

goal={3}

/>

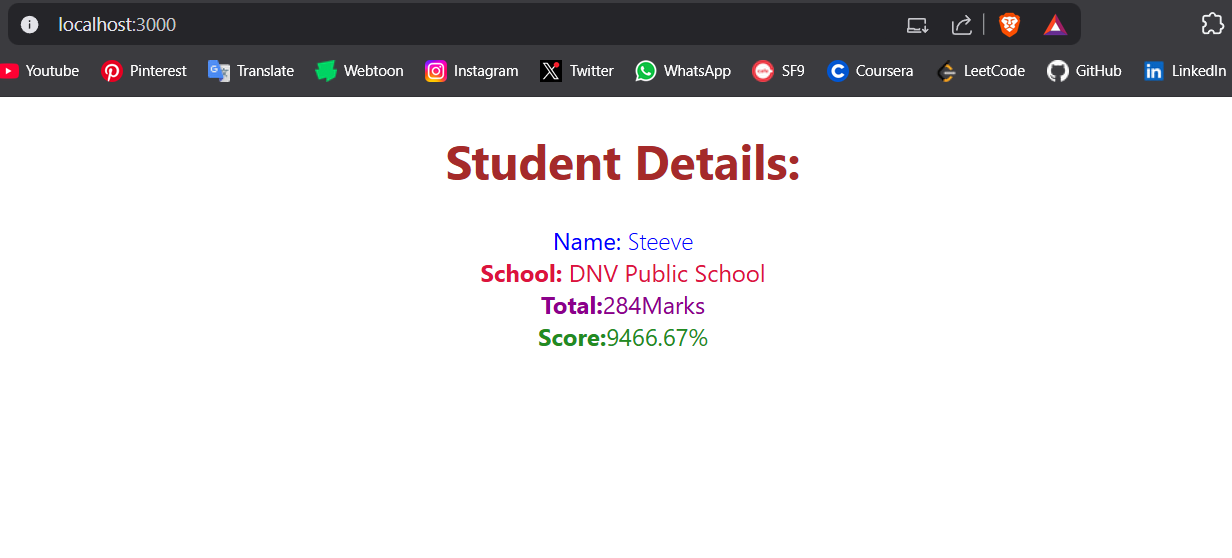
</div>

);

}

export default App;

Output:



# React HandsOn - 4

Post.js:

class Post {

constructor(id, title, body) {

this.id = id;

this.title = title;

this.body = body;

}

}

export default Post;

Posts.js:

import React from 'react';

class Posts extends React.Component {

constructor(props) {

super(props);

this.state = {

posts: []

};

}

loadPosts(limit = 5) {

fetch(`https://jsonplaceholder.typicode.com/posts?\_limit=${limit}`)

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

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

}

componentDidMount() {

this.loadPosts();

}

componentDidCatch(error, info) {

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

}

render() {

return (

<div>

<h1>Posts</h1>

<ul>

{this.state.posts.map(post => (

<li key={post.id}>

<h2>{post.title}</h2>

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

</li>

))}

</ul>

</div>

);

}

}

export default Posts;

App.js:

import Posts from "./Posts.js";

function App() {

return (

<div>

<Posts />

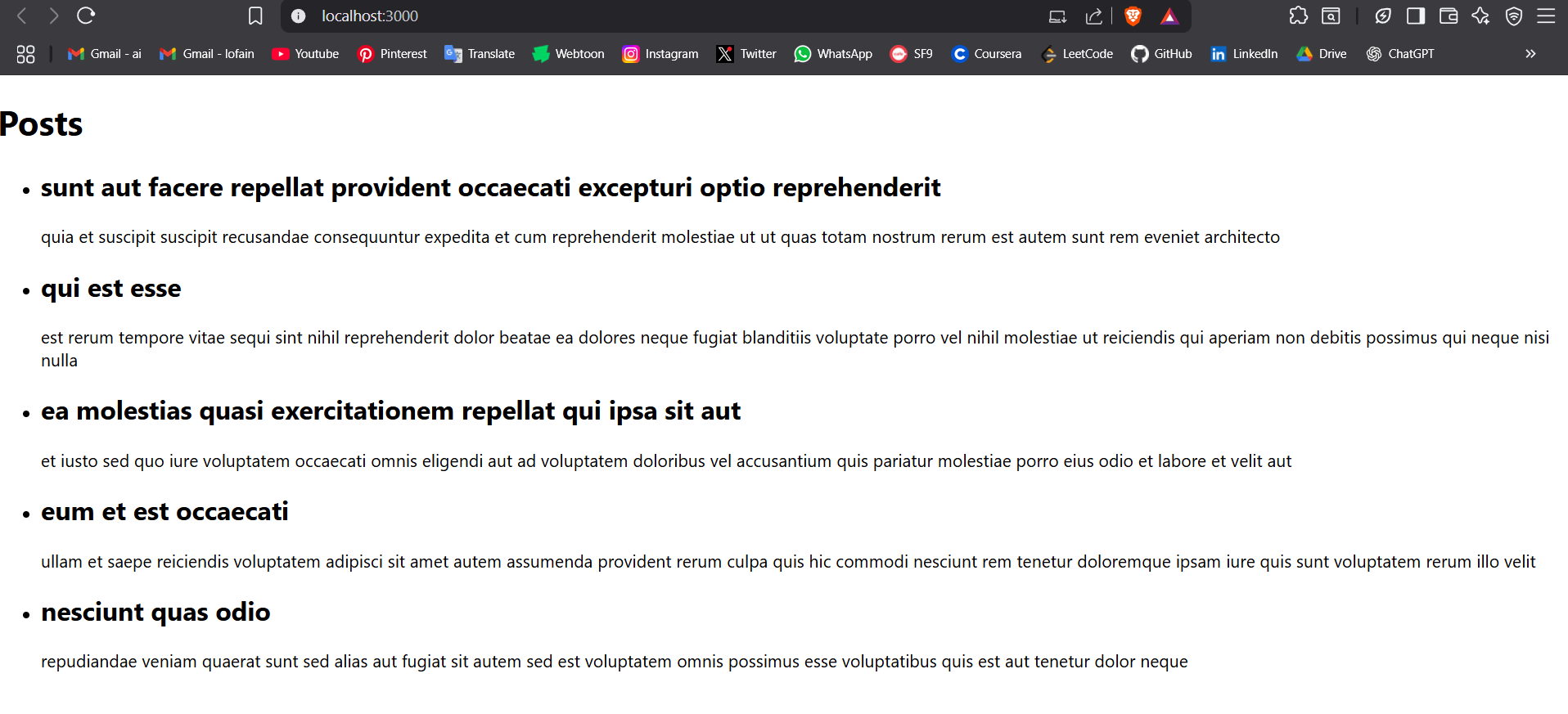
</div>

);

}

export default App;

Output:



# React HandsOn - 5

CohortDetails.js

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

function CohortDetails(props) {

return (

<div className={styles.box}>

<h3 style={{ color: props.cohort.currentStatus === "Ongoing" ? "green" : "blue" }}>

{props.cohort.cohortCode} -

<span>{props.cohort.technology}</span>

</h3>

<dl>

<dt>Started On</dt>

<dd>{props.cohort.startDate}</dd>

<dt>Current Status</dt>

<dd>{props.cohort.currentStatus}</dd>

<dt>Coach</dt>

<dd>{props.cohort.coachName}</dd>

<dt>Trainer</dt>

<dd>{props.cohort.trainerName}</dd>

</dl>

</div>

);

}

export default CohortDetails;

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:

