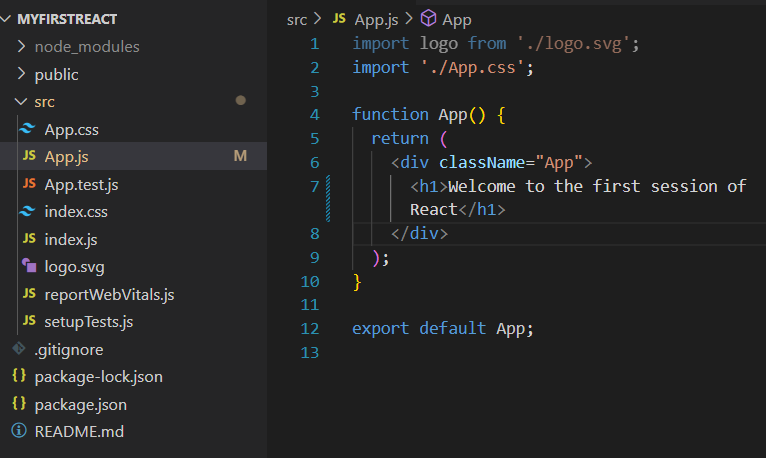
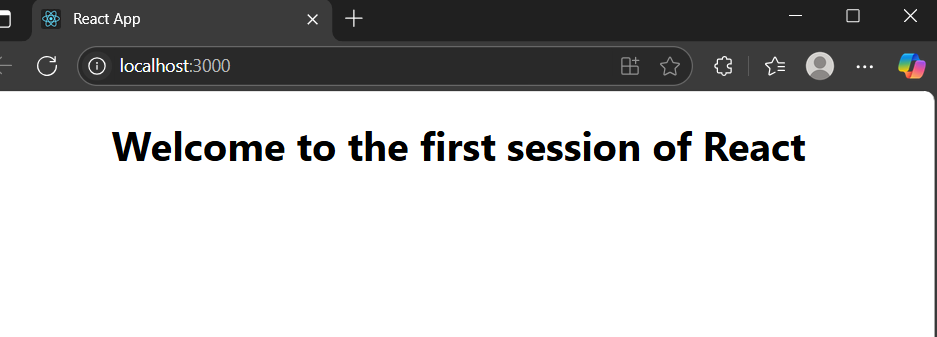
**HandsOn 1**

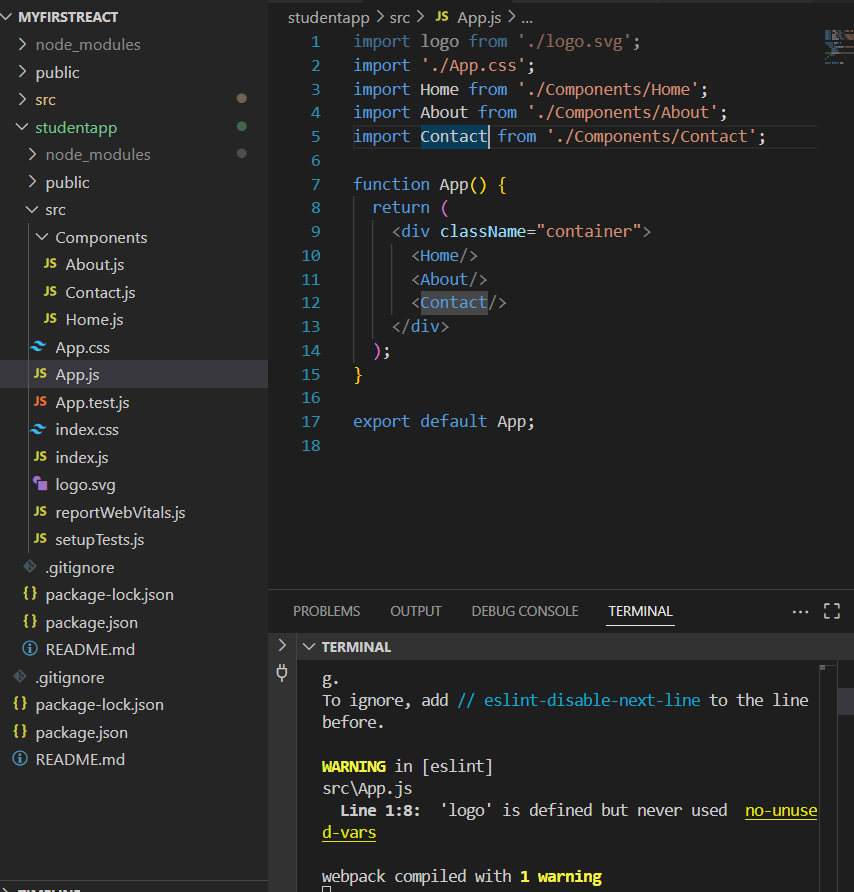
**Create a new React Application with the name “myfirstreact”, Run the application to print “welcome to the first session of React” as heading of that page.**

****

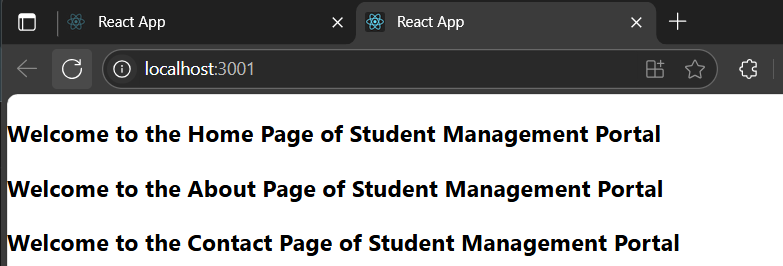


**EXERCISE 2**

**Create a react app for Student Management Portal named StudentApp and create a component named Home which will display the Message “Welcome to the Home page of Student Management Portal”. Create another component named About and display the Message “Welcome to the About page of the Student Management Portal”. Create a third component named Contact and display the Message “Welcome to the Contact page of the Student Management Portal”. Call all the three components.**

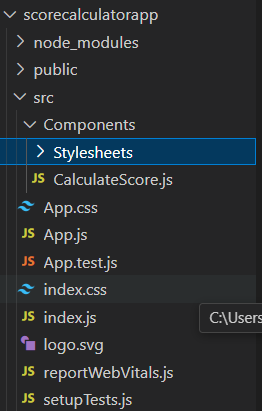
****

Output:



**Exercise 3**

**Create a react app for Student Management Portal named scorecalculatorapp and create a function component named “CalculateScore” which will accept Name, School, Total and goal in order to calculate the average score of a student and display the same.**

****

**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;

**CalculateScore.js**

import './Stylesheets/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 style={{ color: 'brown' }}>Student Details:</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>

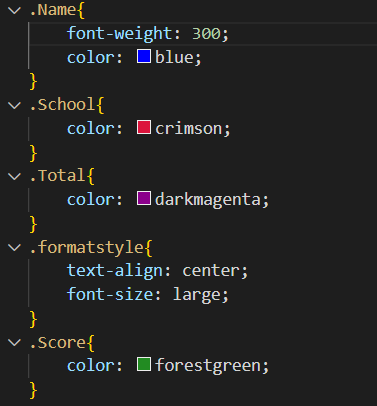
      <span>{calcScore(total, goal)}</span>

    </div>

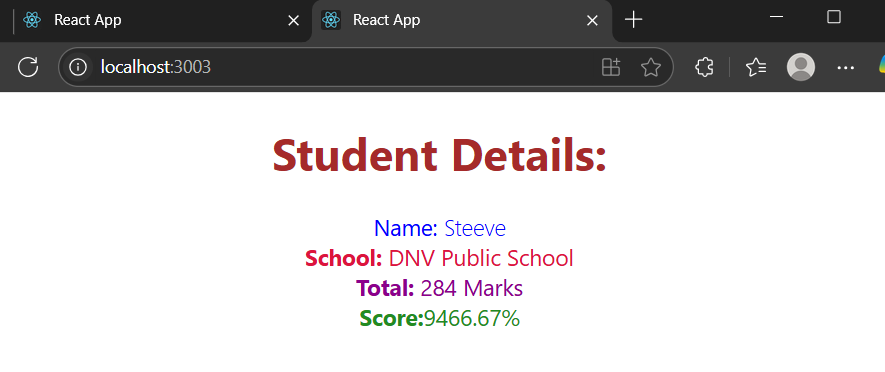
  </div>

);

Stylesheet.css:



**Output:**

****

**Exercise 4:**

* **Implement componentDidMount() hook**
* **Implementing componentDidCatch() life cycle hook.**

1. **Create a new react application using *create-react-app* tool with the name as “blogapp”**
2. **Open the application using VS Code**
3. **Create a new file named as Post.js in src folder with following properties**

****

***Figure 2: Post class***

1. **Create a new class based component named as Posts inside Posts.js file**

****

***Figure 3: Posts Component***

1. **Initialize the component with a list of Post in state of the component using the constructor**
2. **Create a new method in component with the name as loadPosts() which will be responsible for using Fetch API and assign it to the component state created earlier. To get the posts use the url (<https://jsonplaceholder.typicode.com/posts>)**

****

***Figure 4: loadPosts() method***

1. **Implement the componentDidMount() hook to make calls to loadPosts() which will fetch the posts**

****

***Figure 5: componentDidMount() hook***

1. **Implement the render() which will display the title and post of posts in html page using heading and paragraphs respectively.**

****

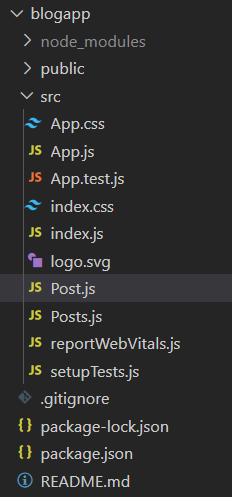
***Figure 6: render() method***

1. **Define a componentDidCatch() method which will be responsible for displaying any error happing in the component as alert messages.**

****

***Figure 7: componentDidCatch() hook***

1. **Add the Posts component to App component.**
2. **Build and Run the application using *npm start* command.**

****

**App.js:**

import React from 'react';

import './App.css';

import Posts from './Posts';

function App() {

  return (

    <div className="App">

      <Posts />

    </div>

  );

}

export default App;

**Post.js**

import React from 'react';

class Post extends React.Component {

  render() {

    return (

      <div className="post">

        <h2>{this.props.title}</h2>

        <p>{this.props.body}</p>

      </div>

    );

  }

}

export default Post;

**Posts.js**

import React from 'react';

import Post from './Post';

class Posts extends React.Component {

  constructor(props) {

    super(props);

    this.state = {

      posts: [],

    };

  }

  // 6. loadPosts using Fetch API

  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));

  }

  // 7. Call loadPosts inside componentDidMount

  componentDidMount() {

    this.loadPosts();

  }

  // 9. Error handling with componentDidCatch

  componentDidCatch(error, info) {

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

    console.error('Error Info:', info);

  }

  // 8. Render posts as heading & paragraph

  render() {

    return (

      <div>

        <h1>Blog Posts</h1>

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

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

        ))}

      </div>

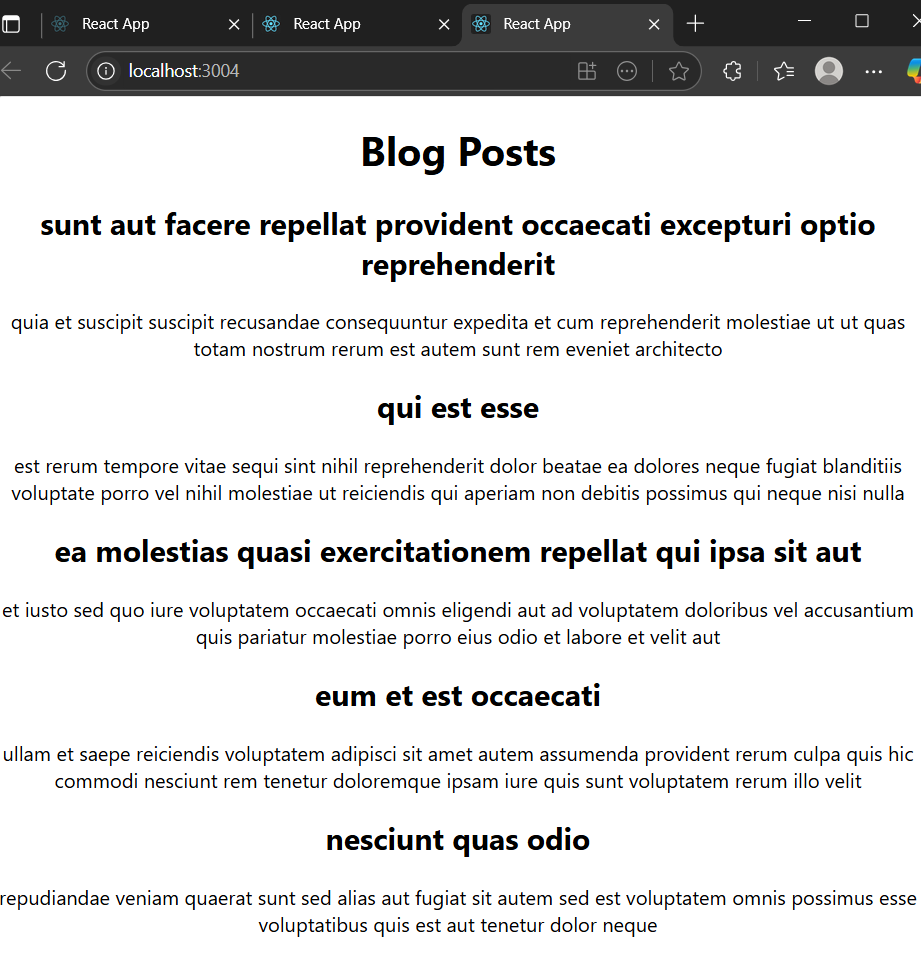
    );

  }

}

export default Posts;

**OUTPUT:**

****

**Exercise 5**

**My Academy team at Cognizant want to create a dashboard containing the details of ongoing and completed cohorts. A react application is created which displays the detail of the cohorts using react component. You are assigned the task of styling these react components.**

**Download and build the attached react application.**

****

1. **Unzip the react application in a folder**
2. **Open command prompt and switch to the react application folder**
3. **Restore the node packages using the following commands**

****

***Figure 1: Restore packages***

1. **Open the application using VS Code**
2. **Create a new CSS Module in a file called “CohortDetails.module.css”**
3. **Define a css class with the name as “box” with following properties**

***Width = 300px;***

***Display = inline block;***

***Overall 10px margin***

***Top and bottom padding as 10px***

***Left and right padding as 20px***

***1 px border in black color***

***A border radius of 10px***

1. **Define a css style for html <dt> element using tag selector. Set the font weight to 500.**
2. **Open the cohort details component and import the CSS Module**
3. **Apply the box class to the container div**
4. **Define the style for <h3> element to use “green” color font when cohort status is “ongoing” and “blue” color in all other scenarios.**
5. **Final result should look similar to the below image**

****

**App.js**

import React from 'react';

import CohortDetails from './components/CohortDetails';

function App() {

// Sample cohort data (from your screenshot)

const cohorts = [

{

id: 'INTADMDF10',

type: '.NET FSD',

startDate: '22-Feb-2022',

status: 'Scheduled',

coach: 'Aathma',

trainer: 'Jojo Jose',

},

{

id: 'ADM21JF014',

type: 'Java FSD',

startDate: '10-Sep-2021',

status: 'Ongoing',

coach: 'Apoorv',

trainer: 'Elisa Smith',

},

{

id: 'CDBJF21025',

type: 'Java FSD',

startDate: '24-Dec-2021',

status: 'Ongoing',

coach: 'Aathma',

trainer: 'John Doe',

},

];

return (

<div>

<h2>Cohorts Details</h2>

{cohorts.map((cohort) => (

<CohortDetails key={cohort.id} cohort={cohort} />

))}

</div>

);

}

export default App;

**CohortDetails.js**

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

function CohortDetails({ cohort }) {

return (

<div className={styles.box}>

<h3 className={cohort.status === 'Ongoing' ? styles.green : styles.blue}>

{cohort.id} - {cohort.type}

</h3>

<dl>

<dt>Started On</dt>

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

<dt>Current Status</dt>

<dd>{cohort.status}</dd>

<dt>Coach</dt>

<dd>{cohort.coach}</dd>

<dt>Trainer</dt>

<dd>{cohort.trainer}</dd>

</dl>

</div>

);

}

export default CohortDetails;

**CohortDetails.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;

}

.green {

color: green;

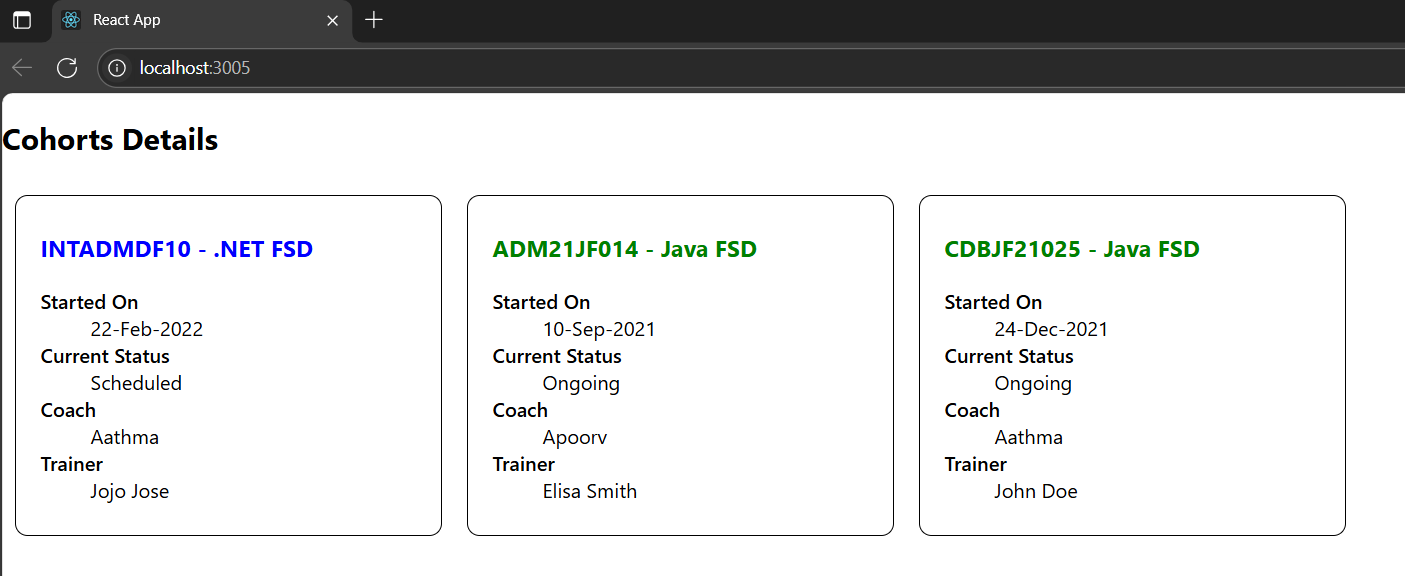
}

.blue {

color: blue;

}

**OUTPUT:**

****