**REACT**

**Hands-on 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.

**CODE:**

**App.js**

import './App.css';

function App() {

  return (

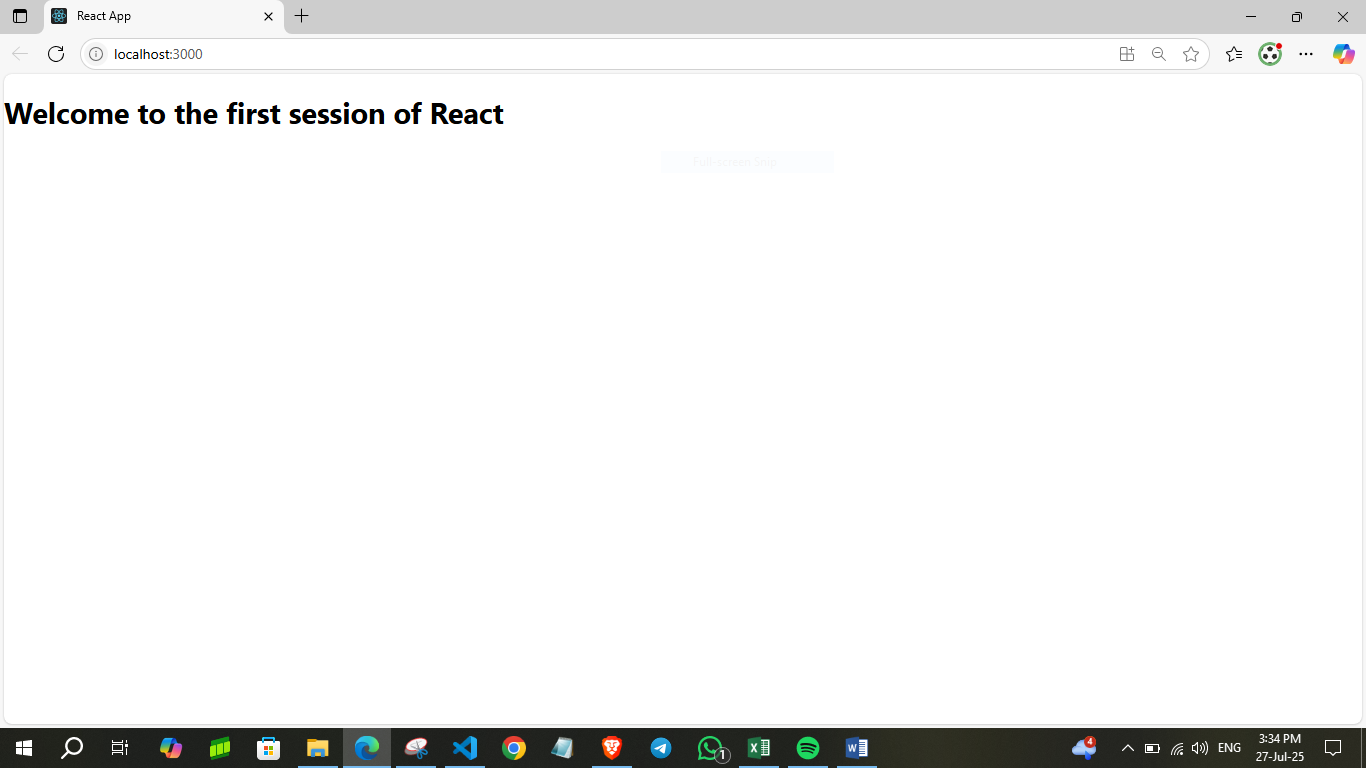
    <h1>Welcome to the first session of React</h1>

  );

}

export default App;

**OUTPUT:**



**Hands-on 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.

**CODE:**

**Home.js**

import React,{Component} from 'react';

class Home extends Component {

    render() {

        return(

            <div>

                <h3>Welcome to the Home Page of Student Management Portal</h3>

            </div>

        )

    }

}

export default Home;

**About.js**

import React,{Component} from 'react';

class About extends Component {

    render() {

        return(

            <div>

                <h3>Welcome to the About Page of Student Management Portal</h3>

            </div>

        )

    }

}

export default About;

**Contact.js**

import React,{Component} from 'react';

class Contact extends Component {

    render() {

        return(

            <div>

                <h3>Welcome to the Contact Page of Student Management Portal</h3>

            </div>

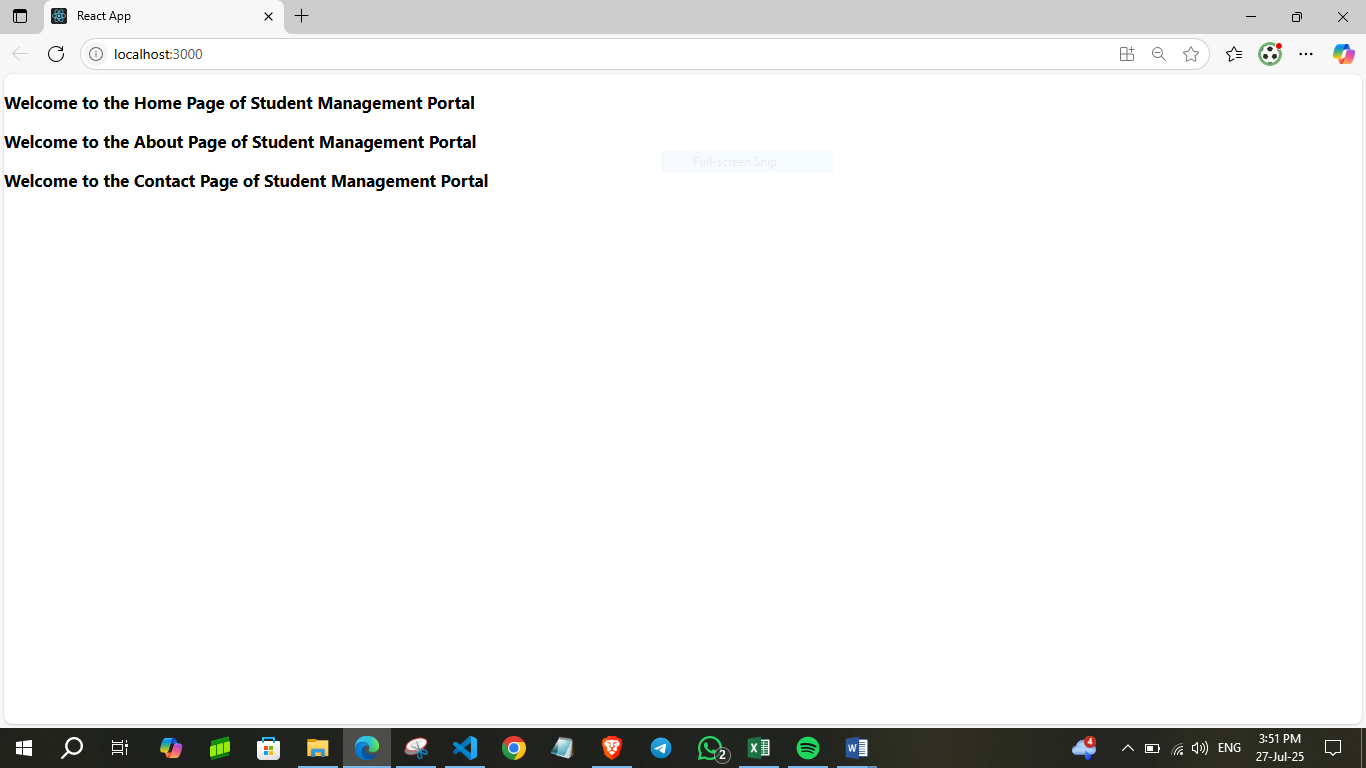
        )

    }

}

export default Contact;

**OUTPUT:**



**Hands-on 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.

**CODE:**

**CalculateScore.js**

import '../Stylesheets/mystyle.css'

const percentToDecimal = (decimal) => {

    return (decimal.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>

);

**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 './App.css';

import { CalculateScore } from './Components/CalculateScore';

function App() {

  return (

    <div>

      <CalculateScore Name={"Steve"}

      School={"DNV Public School"}

      total={284}

      goal={3}>

      </CalculateScore>

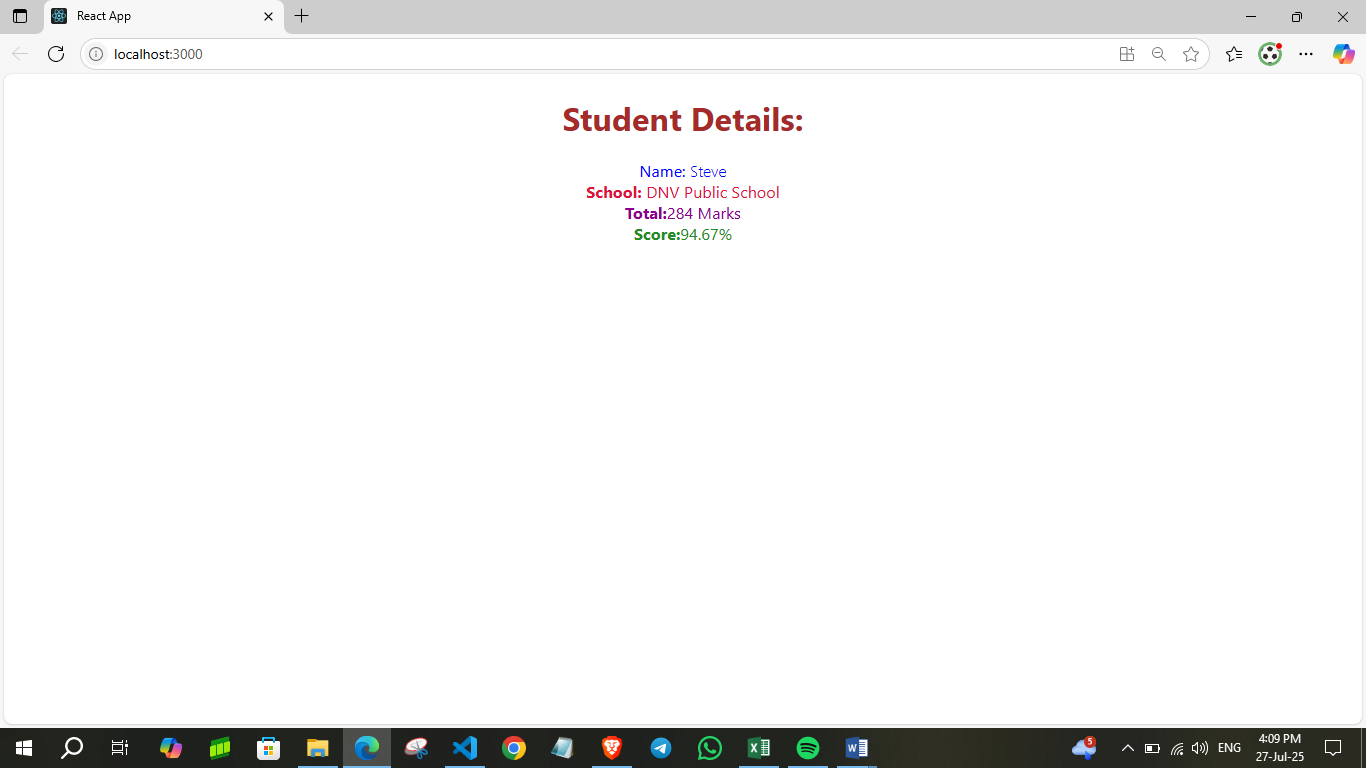
    </div>

  );

}

export default App;

**OUTPUT:**



**Hands-on 4:**

Create a React application that fetches and displays data from an external API. Implement lifecycle methods to manage data loading and error handling. Ensure the fetched content is rendered dynamically and the app runs successfully.

**CODE:**

**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';

import Post from './Post';

class Posts extends React.Component {

  constructor(props) {

    super(props);

    this.state = {

      posts: [],

    };

  }

  loadPosts() {

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

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

      .then(data => {

        const postsList = data.map(

          item => new Post(item.id, item.title, item.body)

        );

        this.setState({ posts: postsList });

      })

      .catch(error => {

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

        alert("Error fetching posts");

      });

  }

  componentDidMount() {

    this.loadPosts();

  }

  componentDidCatch(error, info) {

    console.error("Error caught in component:", error, info);

    alert("An error occurred in the Posts component");

  }

  render() {

    return (

      <div>

        <h1>Blog Posts</h1>

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

          <div key={post.id}>

            <h3>{post.title}</h3>

            <p>{post.body}</p>

            <hr />

          </div>

        ))}

      </div>

    );

  }

}

export default Posts;

**App.js**

import React from 'react';

import './App.css';

import Posts from './Posts';

function App() {

  return (

    <div className="App">

      <Posts />

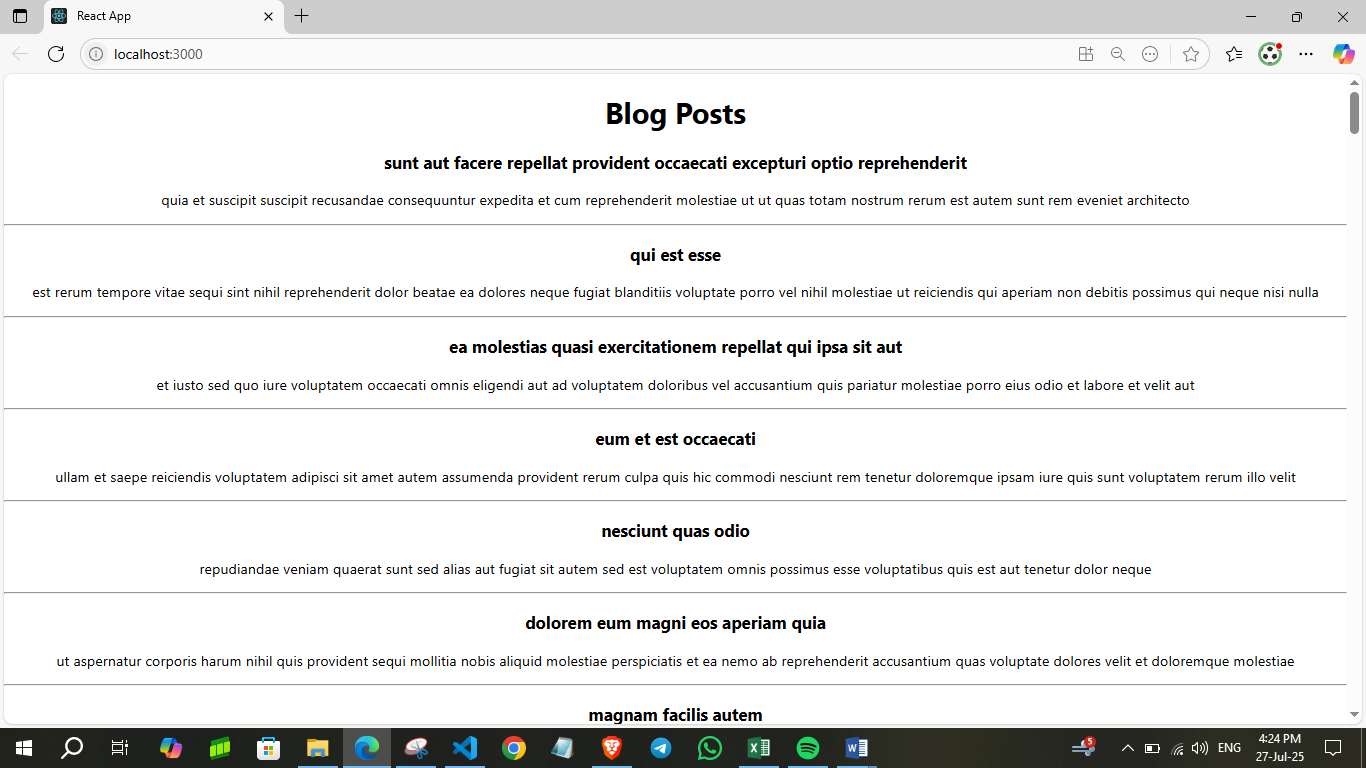
    </div>

  );

}

export default App;

**OUTPUT:**



**Hands-on 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.

**CODE:**

**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:**

