1. ReactJS-HOL

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.

**App.js:**

import React from 'react';

function App() {

  return (

    <div>

      <h1>Hello, welcome to my first React app!</h1>

      <p>This is a basic React application.</p>

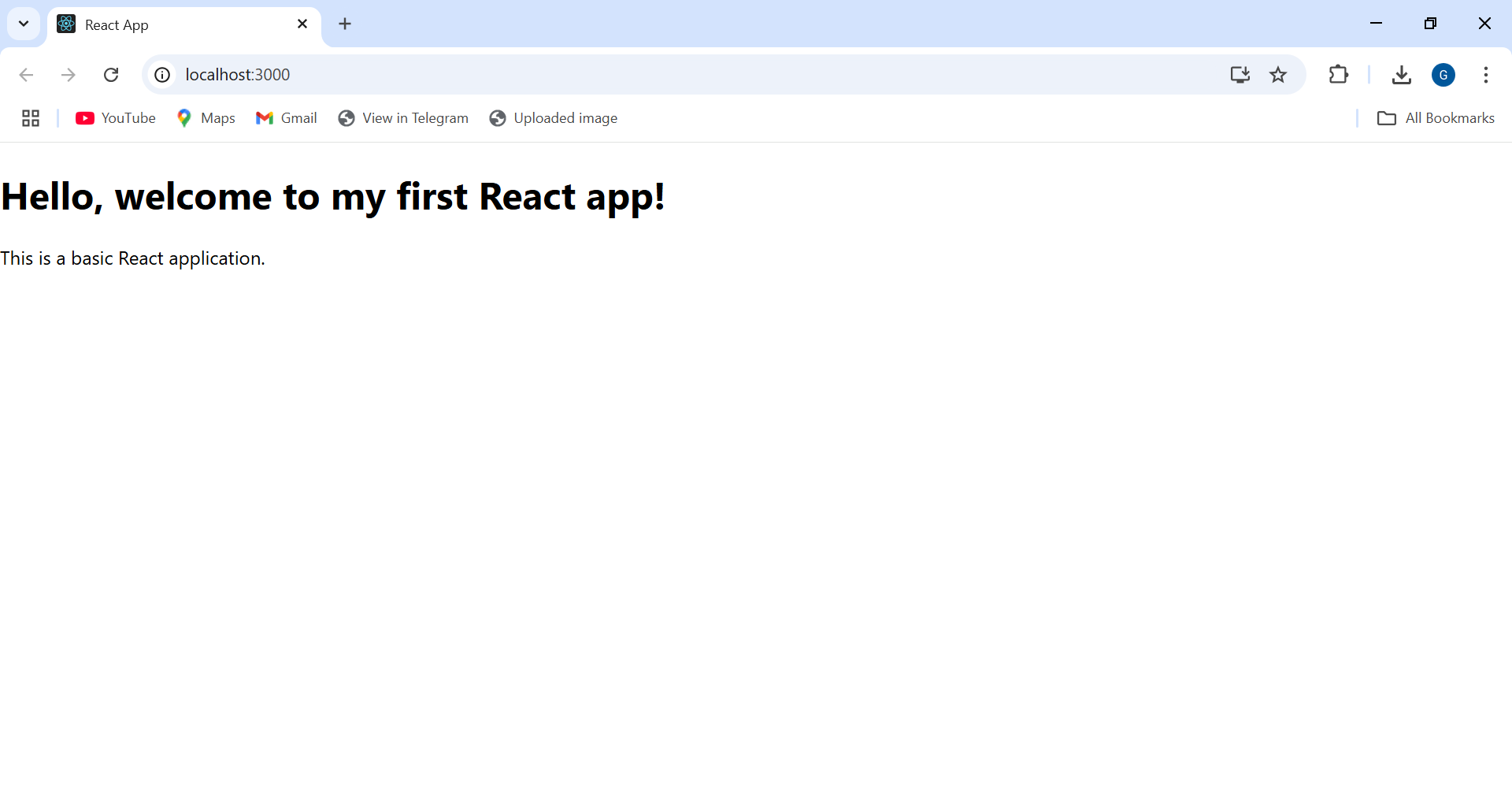
    </div>

  );

}

export default App;

**Output:**



2. ReactJS-HOL

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”.

**About.js:**

import React from 'react';

function About() {

  return (

    <div>

      <h2>Welcome to the About page of Student Management Portal</h2>

    </div>

  );

}

export default About;

**Contact.js:**

import React from 'react';

function Contact() {

  return (

    <div>

      <h2>Welcome to the Contact page of Student Management Portal</h2>

    </div>

  );

}

export default Contact;

**Home.js:**

import React from 'react';

function Home() {

  return (

    <div>

      <h2>Welcome to the Home page of Student Management Portal</h2>

    </div>

  );

}

export default Home;

**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>

      <h1>Welcome to StudentApp</h1>

      <Home />

      <About />

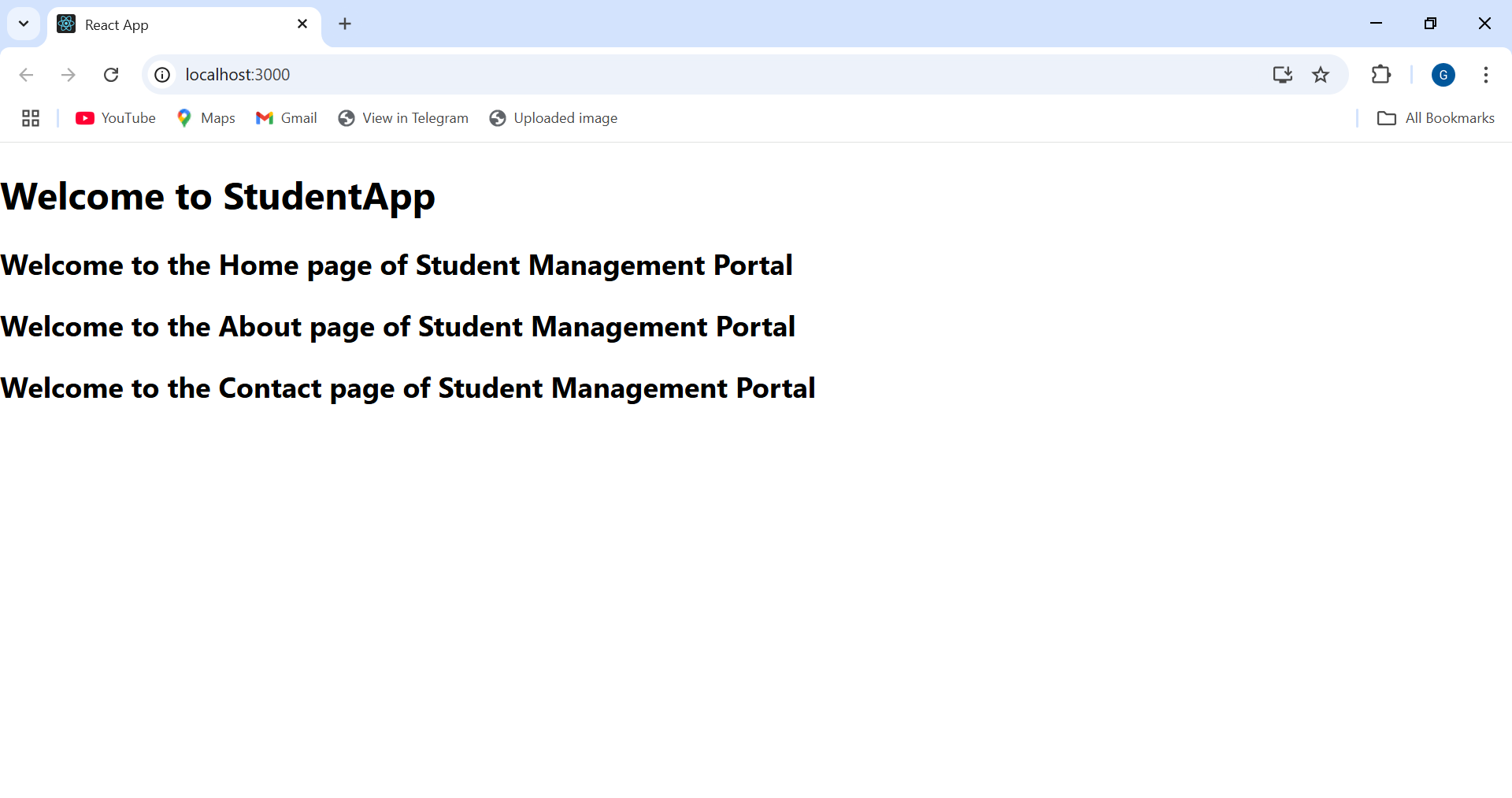
      <Contact />

    </div>

  );

}export default App;

**Output:**



3. ReactJS-HOL

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.

**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;

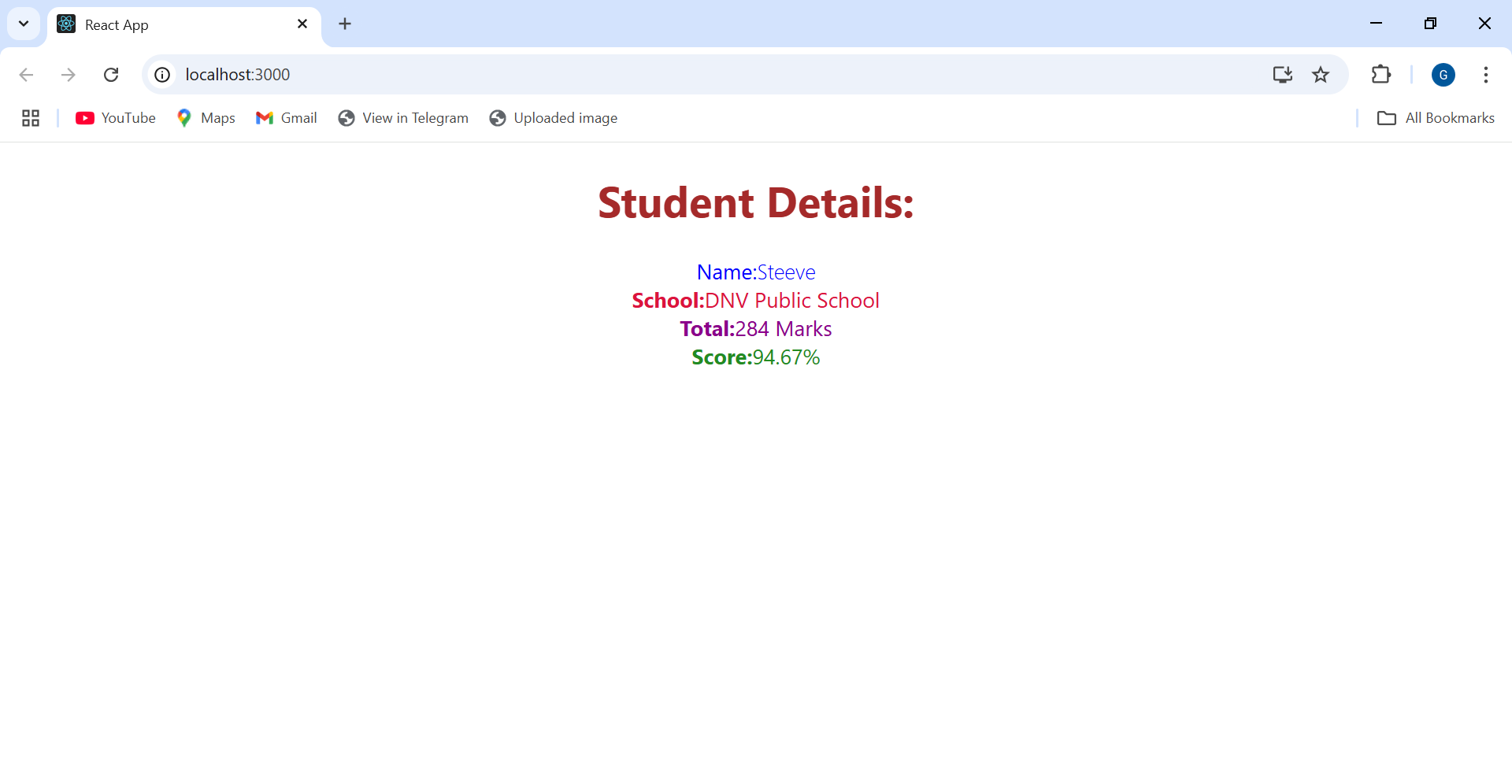
}

.Score {

  color: forestgreen;

}

**Output:**



4. ReactJS-HOL

Create a new react application using *create-react-app* tool with the name as “blogapp”.

**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('Fetch error:', error);

      });

  }

  componentDidMount() {

    this.loadPosts();

  }

  render() {

    return (

      <div>

        <h2>Blog Posts</h2>

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

          <div key={post.id} style={{ border: '1px solid gray', margin: '10px', padding: '10px' }}>

            <h3>{post.title}</h3>

            <p>{post.body}</p>

          </div>

        ))}

      </div>

    );

  }

  componentDidCatch(error, info) {

    alert('An error occurred in the Posts component.');

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

  }

}

export default Posts;

**App.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('Fetch error:', error);

      });

  }

  componentDidMount() {

    this.loadPosts();

  }

  render() {

    return (

      <div>

        <h2>Blog Posts</h2>

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

          <div key={post.id} style={{ border: '1px solid gray', margin: '10px', padding: '10px' }}>

            <h3>{post.title}</h3>

            <p>{post.body}</p>

          </div>

        ))}

      </div>

    );

  }

componentDidCatch(error, info) {

    alert('An error occurred in the Posts component.');

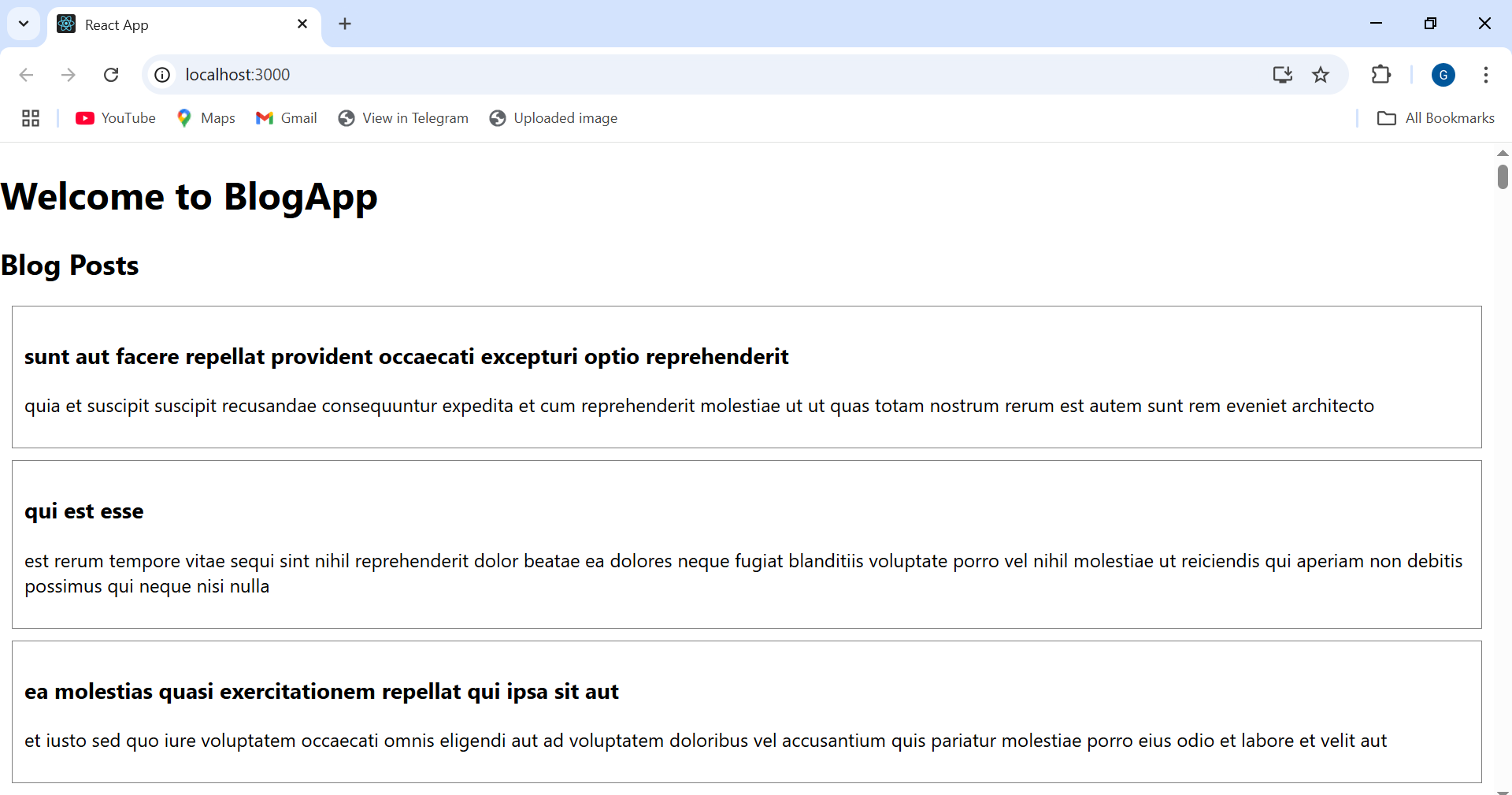
    console.error('Error details:', error, info);

  }

}

export default Posts;

**Output:**



5. ReactJS-HOL

**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;

}

**CohortDetails.js:**

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

function CohortDetails(props) {

    // Get status value in a safe way

    const status = props.cohort.currentStatus?.toLowerCase();

    // Set heading color based on status

    const headingStyle = {

        color: status === 'ongoing' ? 'green' : 'blue'

    };

    return (

        <div className={styles.box}>

            <h3 style={headingStyle}>

                {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;

**Output:**

