**Exercise 3 ScoreCalculatorApp:**

**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 handleCalculate = () => {

    const avg = (parseFloat(total) + parseFloat(goal)) / 2;

    setAverage(avg.toFixed(2));

  };

  return (

    <div className="container">

      <h2>Student Score Calculator</h2>

      <label>Name:</label>

      <input type="text" onChange={e => setName(e.target.value)} />

      <label>School:</label>

      <input type="text" onChange={e => setSchool(e.target.value)} />

      <label>Total Marks:</label>

      <input type="number" onChange={e => setTotal(e.target.value)} />

      <label>Goal Marks:</label>

      <input type="number" onChange={e => setGoal(e.target.value)} />

      <button onClick={handleCalculate}>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;

**src/Stylesheets/mystyle.css:**

.container {

    width: 400px;

    margin: 30px auto;

    padding: 20px;

    border: 2px solid #333;

    border-radius: 8px;

    background-color: #f9f9f9;

  }

  input {

    display: block;

    margin-bottom: 15px;

    width: 100%;

    padding: 8px;

  }

  button {

    padding: 10px;

    width: 100%;

    background-color: #007bff;

    color: white;

    border: none;

    cursor: pointer;

  }

  .result {

    margin-top: 20px;

    background-color: #e7f4ff;

    padding: 10px;

  }

**App.css:**

.App {

  text-align: center;

}

.App-logo {

  height: 40vmin;

  pointer-events: none;

}

@media (prefers-reduced-motion: no-preference) {

  .App-logo {

    animation: App-logo-spin infinite 20s linear;

  }

}

.App-header {

  background-color: #282c34;

  min-height: 100vh;

  display: flex;

  flex-direction: column;

  align-items: center;

  justify-content: center;

  font-size: calc(10px + 2vmin);

  color: white;

}

.App-link {

  color: #61dafb;

}

@keyframes App-logo-spin {

  from {

    transform: rotate(0deg);

  }

  to {

    transform: rotate(360deg);

  }

}

**App.js:**

import React from 'react';

import './App.css';

import CalculateScore from './Components/CalculateScore';

function App() {

  return (

    <div className="App">

      <CalculateScore />

    </div>

  );

}

export default App;

**App.test.js:**

import { render, screen } from '@testing-library/react';

import App from './App';

test('renders learn react link', () => {

  render(<App />);

  const linkElement = screen.getByText(/learn react/i);

  expect(linkElement).toBeInTheDocument();

});

**Index.css:**

body {

  margin: 0;

  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',

    'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',

    sans-serif;

  -webkit-font-smoothing: antialiased;

  -moz-osx-font-smoothing: grayscale;

}

code {

  font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',

    monospace;

}

**Index.js:**

import React from 'react';

import ReactDOM from 'react-dom/client';

import './index.css';

import App from './App';

import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(

  <React.StrictMode>

    <App />

  </React.StrictMode>

);

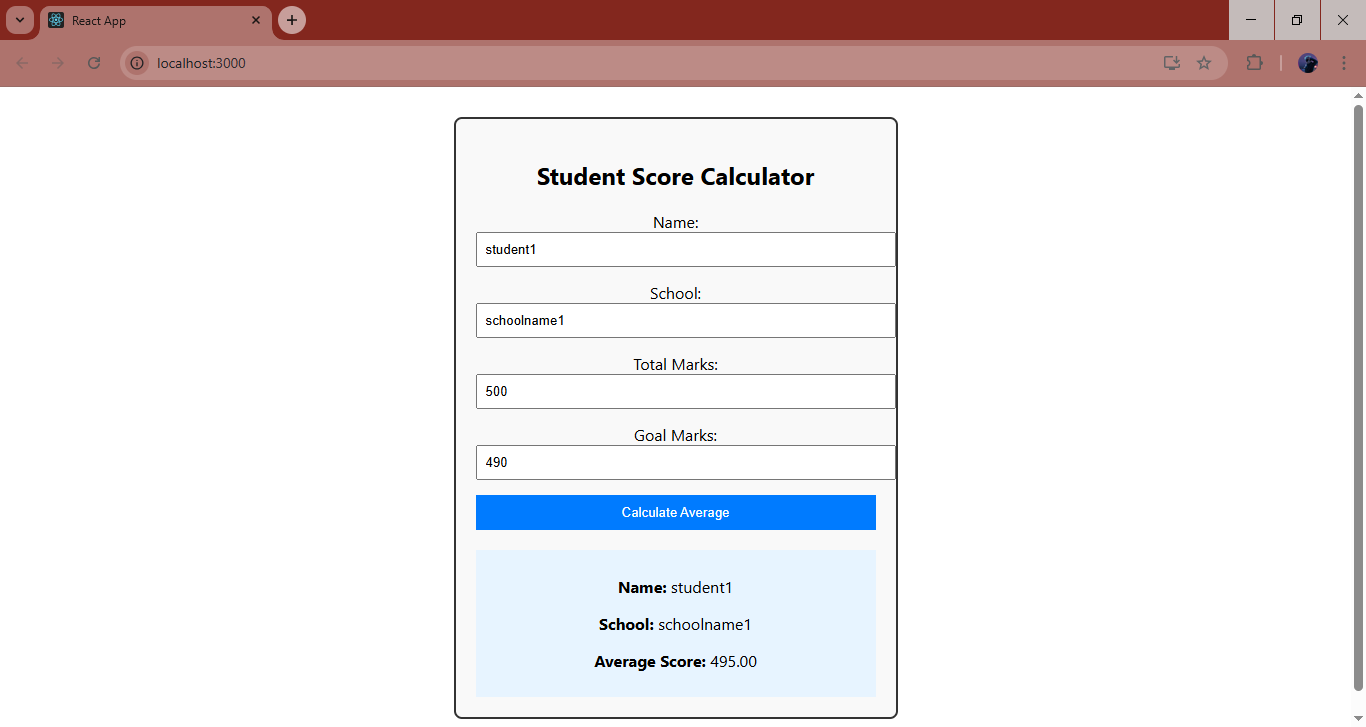
// If you want to start measuring performance in your app, pass a function

// to log results (for example: reportWebVitals(console.log))

// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals

reportWebVitals();

**Output:**

****