1. Create user define date npm module by using node.js and access date npm package module to read the year is leap year or not.

Source code :

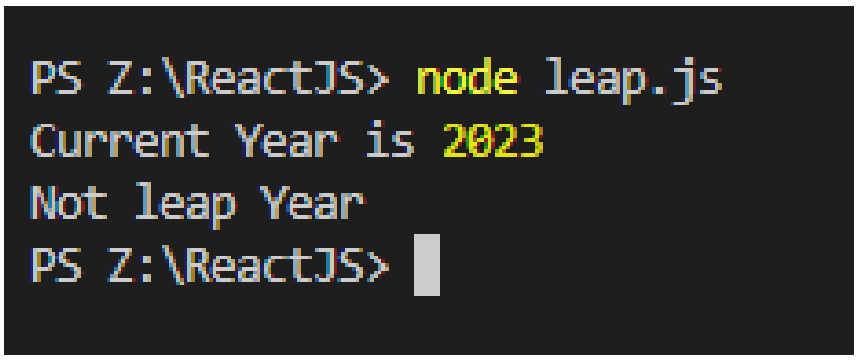
Second.js

```
exports.date=function(){
    return new Date().getFullYear();
}
```

leap.js

```
const year=require('./second');
console.log('Current Year is',year.date());
if(year.date()%400==0 || (year.date()%4==0&& year.date()%100!=0)){
    console.log("Leap Year");
}
else{
    console.log("Not leap Year");
}
```

Output:



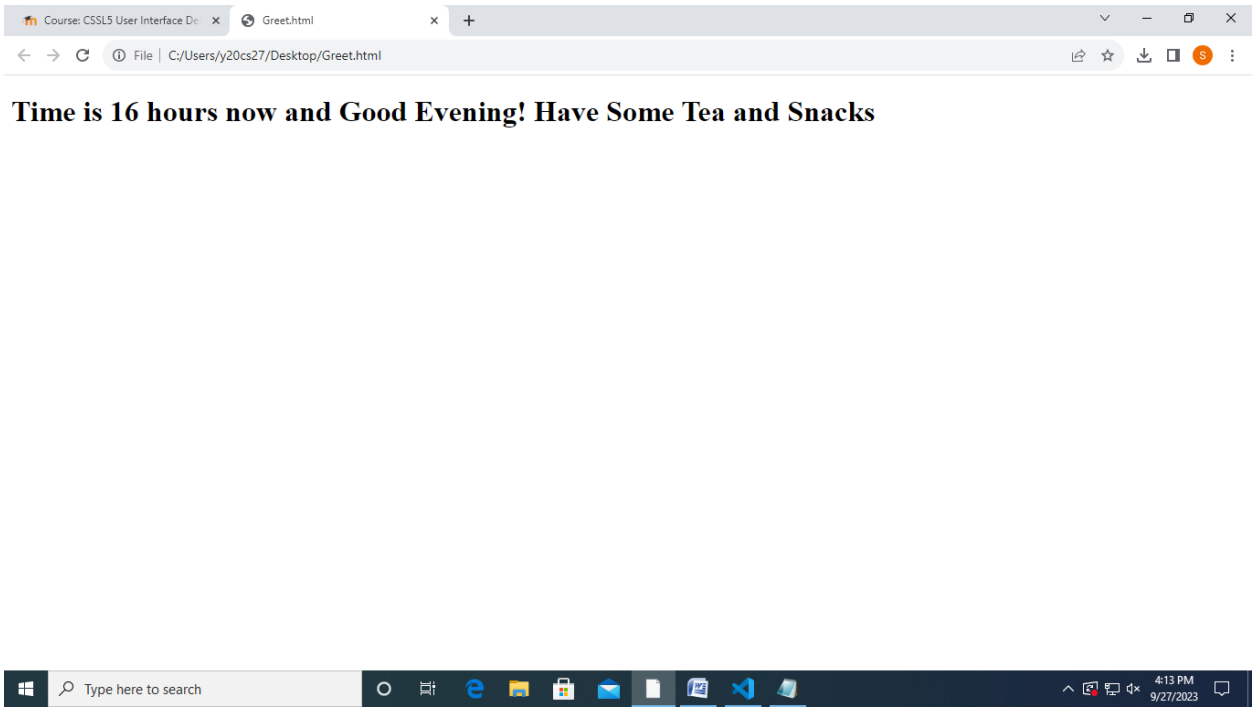
2.Create nested function component by using CDN's (without react environment)

Source Code :

```
<!DOCTYPE html>
<html>
<head>
<script src="https://unpkg.com/react@18/umd/react.development.js" crossorigin></script>
<script src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"
crossorigin></script>
<script src="https://unpkg.com/@babel/standalone/babel.min.js"></script>
</head>
<body>
<div id="root"></div>
<script type="text/babel">
    function Greet() {

  let date=new Date();
  let hours=date.getHours();
  let msg;

  if(hours>5 && hours<12){
  msg='Good Morning! Have a nice Day Ahead';
  }

  else if(hours>=12 && hours<16){
  msg="Good Afternoon! It's Too Hot here,Don't Go Outside";
  }

  else if(hours>=16 && hours<21){
  msg='Good Evening! Have Some Tea and Snacks';
  }

  else if(hours=>21){
  msg='Good Night! Have a Sweet Dreams';
  }
   return <h1 style="color:green"> Time is {hours} hours now and {msg} </h1>;
   }


   function Welcome() {
     return <Greet/>;
   }
```

```
    const container = document.getElementById('root');
    const root = ReactDOM.createRoot(container);
    root.render(<Welcome/>)
</script>
</body>
</html>
```

Output :



3. Create and develop reactJS app folder structure using npx and npm utilities and display the welcome react app message

Source Code :

App.js

```
import rvr from './rvr.jpg';
import './App.css';


function App() {
  return (
<div className="App">
<header className="App-header">
<img src={rvr} className="App-logo" alt="rvr" />
<p>
        Welcome to My First ReactJS Application
</p>
</header>
</div>
  );
}
export default App;
```

App.css

```
.App {
  text-align: center;
}
```

```
.App-logo {
  height: 40vmin;
  pointer-events: none;
}
.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;
}
```

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>
```

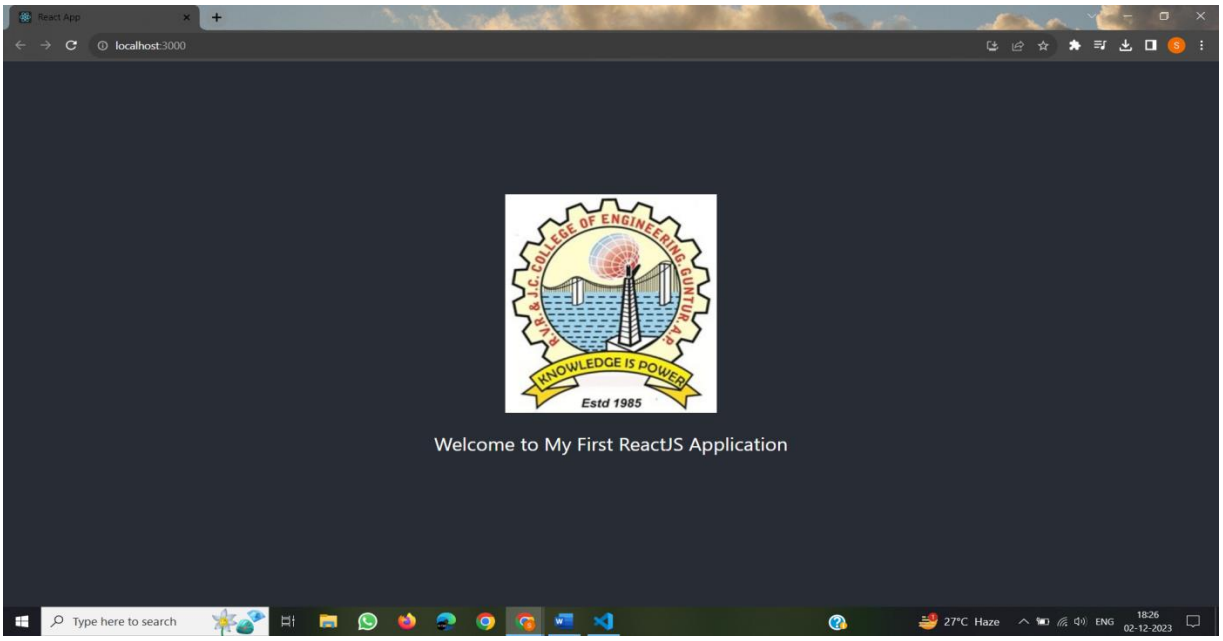
<App />

</React.StrictMode>

);

reportWebVitals();

Output :



4. Create React 2 types of Components (Class Component and Functional Component) and demonstrate each Compoment

Source Code :

ClassComp.js

```
import React from 'react';

class Class1 extends React.Component{

   render(){

      return (<h1 style={{textAlign:"center",backgroundColor:"yellow",color:"green"}}>My First Class Component</h1>);

   }

}


export default Class1;
```

FunctionalComp.js

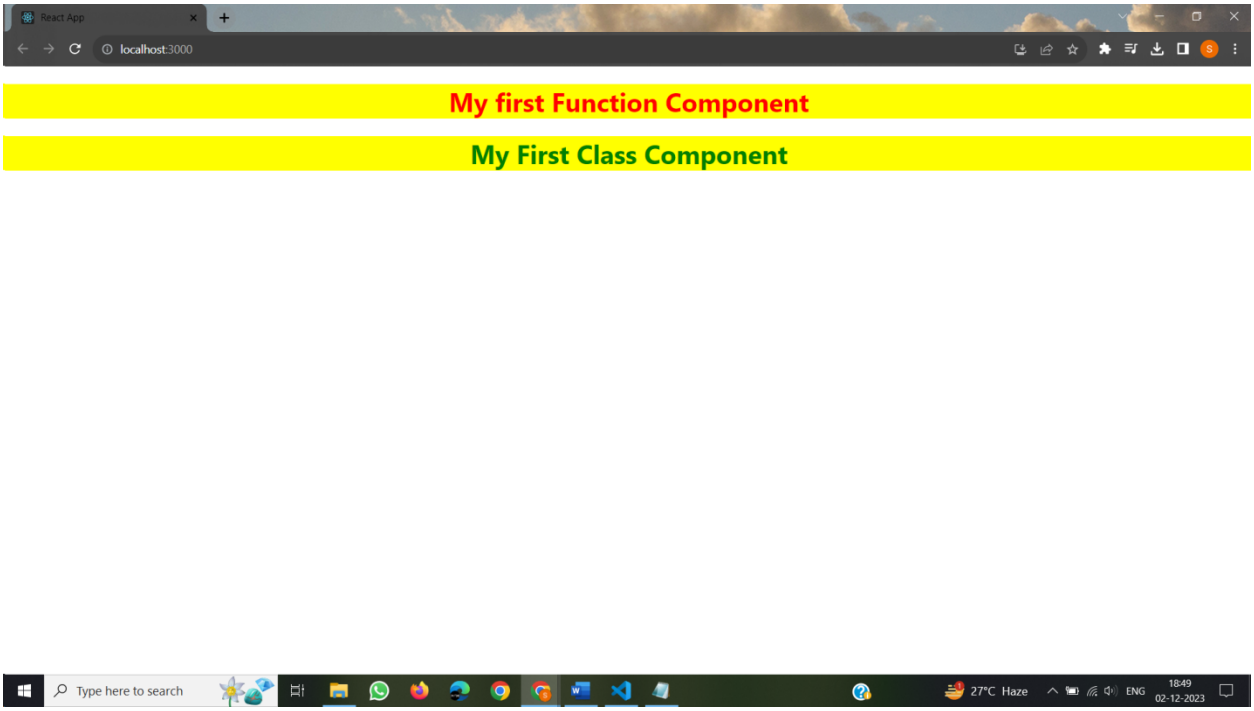
```
function Func1(){

   return (

<h1

      style={{

        color:"red",

        backgroundColor:"yellow",

        textAlign:"center"
```

```
      }}
>My first Function Component</h1>
    );
}
export default Func1;
```

Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import reportWebVitals from './reportWebVitals';
import Func1 from './FunctionalComp.js';
import Class1 from './ClassComp.js'
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
<React.StrictMode>
<Func1/>
<Class1/>
</React.StrictMode>
);
reportWebVitals();
```

Output :


React App
localhost:3000
My first Function Component
My First Class Component
Type here to search
27°C Haze
ENG
18:49
02-12-2023

5. Create React application by using different style components in React

Source Code :

Style.js

```
import React from 'react';
import './style1.css';
const style2={
   fontSize:'50px',
   textAlign: 'center',
   backgroundColor:"yellow",
   color: 'red'
};
class Class2 extends React.Component{
   render(){
      return (
<div>
<p className='style1'>This is the example of external CSS</p>
<p style={style2}>This is the example of Inline CSS</p>
</div>
      );
   }
}
export default Class2;
```

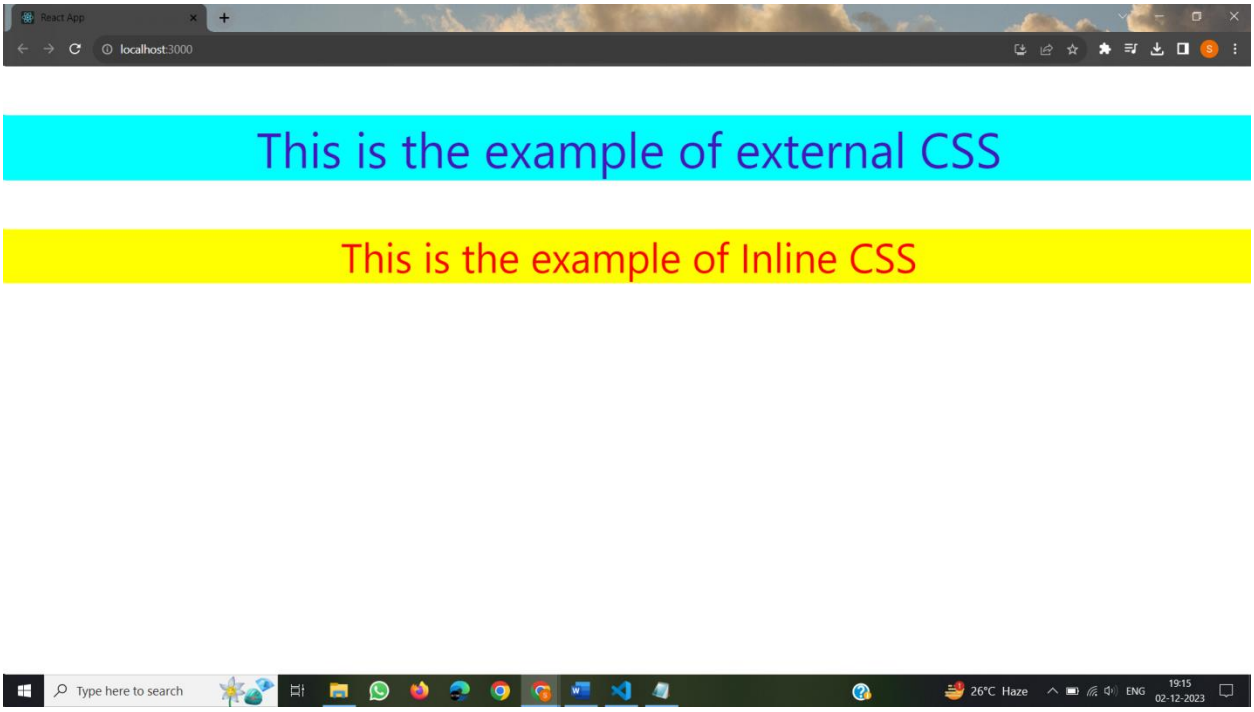
Style1.css

```
.style1
{
   font-size: 60px;
   text-align: center;
   background-color: aqua;
   color: rgb(71, 21, 187);
}
```

Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import reportWebVitals from './reportWebVitals';
import  Class2 from './Class2';
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <Class2/>
  </React.StrictMode>
);
reportWebVitals();
```

Output :


React App
localhost:3000
This is the example of external CSS
This is the example of Inline CSS
Type here to search
26°C Haze
ENG
19:15
02-12-2023

6. Create a simple Counter application using ReactJS which increments and decrements count dynamically on screen as user clicks on the button by using props and states concept

Source Code :

Counter.js

```
import React from "react";
import "./Counter.css";
class Counter1 extends React.Component{
   constructor(){
      super();
      this.state={
         counter:0
      };
   }
   Incr=()=>{
      this.setState({
         counter:this.state.counter+1
      });
   };
   Decr=()=>{
      this.setState({
         counter:this.state.counter-1
      });
   }
```

```
    render(){
        const mystyle={
            textAlign:"center"
        }
        return(
<center>
<div className="main" >
<h1 style={mystyle}>Counter Application : ReactJS</h1>
<h2 style={mystyle}>Counter = {this.state.counter}</h2>
<div style={mystyle}><button className="button" onClick={this.Incr} >Increment</button>
<button className="button" onClick={this.Decr}>Decrement</button>
</div>
</div>
</center>
        )
    }
}
export default Counter1;
```

Counter.css

```
.main{
    border-radius: 8px;
    font-size: 18px;
    background-color: rgba(197, 242, 244, 0.979);
    width: 700px;
```

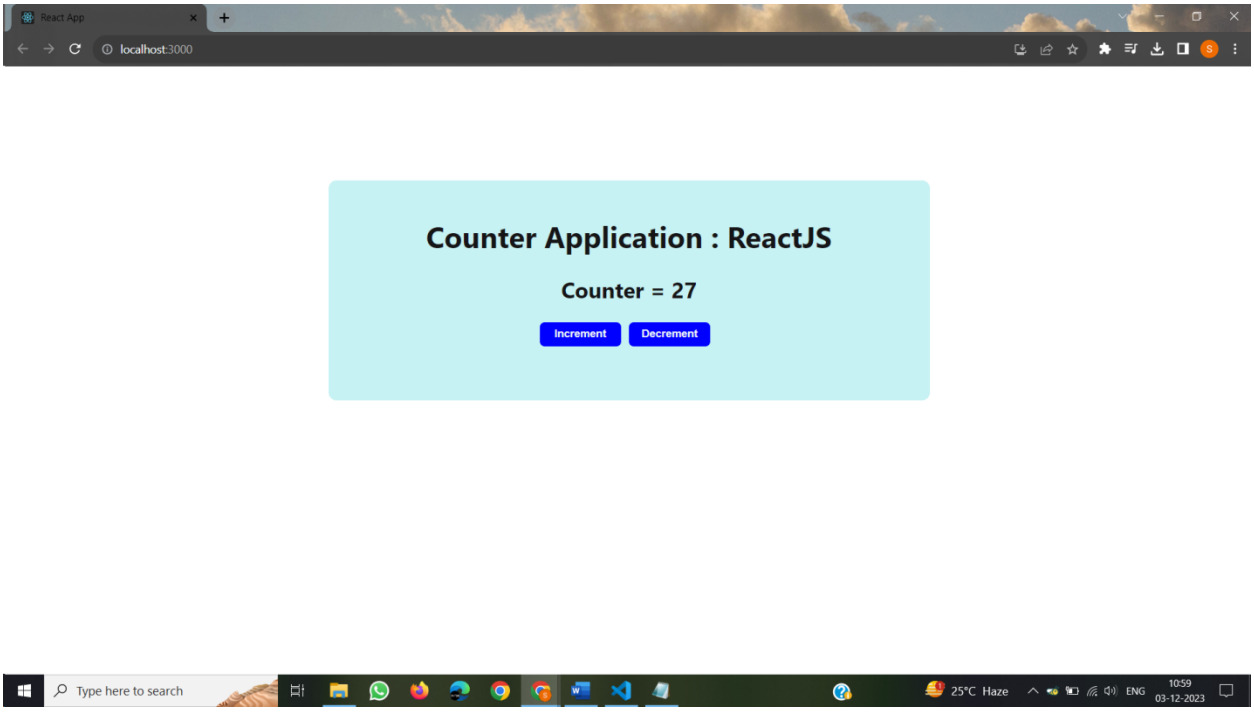
```
    height: 230px;
    border-radius: 10px;
    color: rgb(21, 20, 22);
    margin: 140px;
    padding: 20px;
}
.button{
    color: rgb(238, 247, 247);
    background-color: blue;
    text-align: center;
    margin-right: 10px;
    font-weight: bold;
    width: 100px;
    height: 30px;
    border-radius: 6px;
    border: 2px solid blue;
}
```

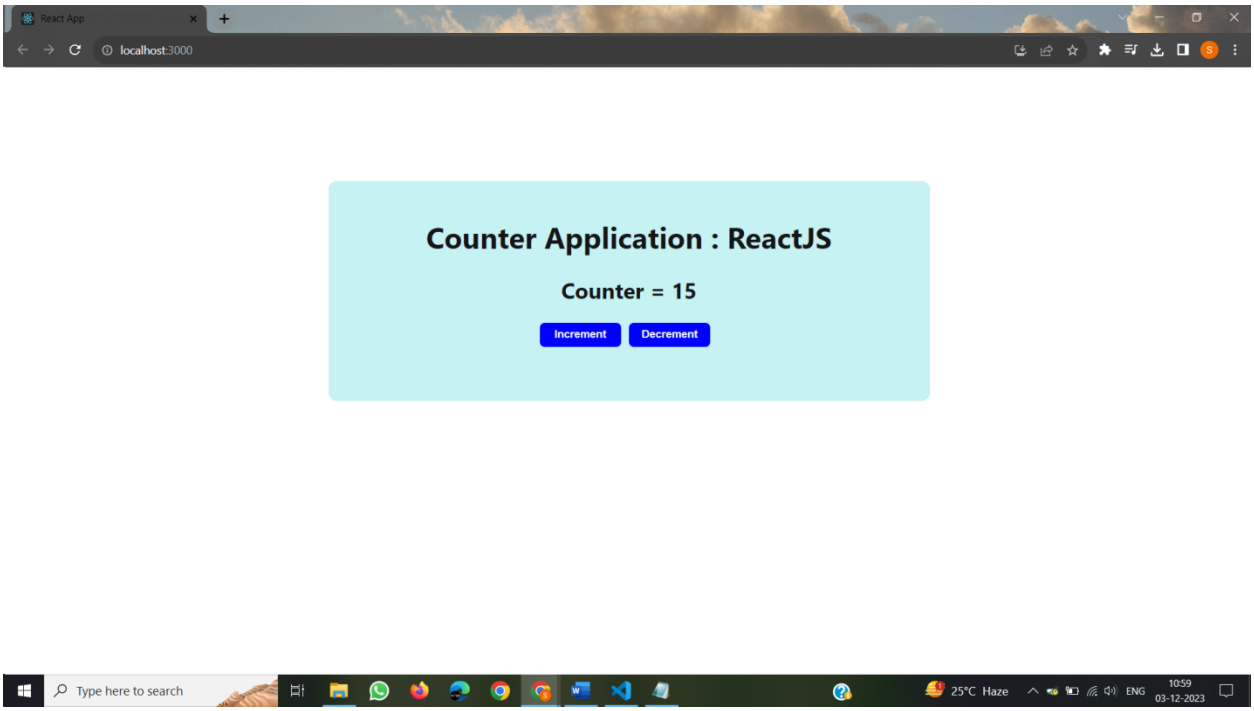
Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import reportWebVitals from './reportWebVitals';
import Counter from './ Counter ';
const root = ReactDOM.createRoot(document.getElementById('root'));
```

```
root.render(
 <React.StrictMode>
  <Counter/>
 </React.StrictMode>
);
reportWebVitals();
```

Output :




React App
localhost:3000
Counter Application : ReactJS
Counter = 15
Increment
Decrement
Type here to search
25°C Haze
ENG
10:59
03-12-2023

7. Create a simple Login Form single page application using ReactJS

Source Code :

Login.js

```
import React, { useState } from "react";
import "./Login.css";

function App6() {
  const [errorMessages, setErrorMessages] = useState({});
  const [isSubmitted, setIsSubmitted] = useState(false);
  const data = [
    {
      username: "chrb",
      password: "abc"
    },
    {
      username: "rishi",
      password: "def"
    },
    {
       username: "RVR",
       password: "def"
     }
  ];
```

```
const errors = {
  uname: "Invalid Username",
  pass: "Invalid Password"
 };
 const handleSubmit = (event) => {
  event.preventDefault();
  var { uname, pass } = document.forms[0];
  const userData = data.find((user) => user.username === uname.value);
  if (userData) {
   if (userData.password !== pass.value) {
    setErrorMessages({ name: "pass", message: errors.pass });
   } else {
    setIsSubmitted(true);
   }
  } else {
   setErrorMessages({ name: "uname", message: errors.uname });
  }
 };
 const renderErrorMessage = (name) =>
  name === errorMessages.name && (
   <div className="error">{errorMessages.message}</div>
  );
  const WebForm = (
  <div className="form">
```

```
   <form onSubmit={handleSubmit}>
     <div className="input-container">
       <label>Username </label>
       <input type="text" name="uname" required />
       {renderErrorMessage("uname")}
     </div>
     <div className="input-container">
       <label>Password </label>
       <input type="password" name="pass" required />
       {renderErrorMessage("pass")}
     </div>
     <div className="button-container">
       <input type="submit" value="LOGIN"/>
     </div>
   </form>
  </div>
);
return (
  <div className="main1">
    <div className="login-form">
      <div className="title">Sign In</div>
      {isSubmitted ? <div>User is successfully logged in</div> : WebForm}
    </div>
  </div>
```

```
  );
}
export default Login;
```

Login.css

```
.main1{
   font-family: sans-serif;
   display: flex;
   align-items: center;
   justify-content: center;
   flex-direction: column;
   height: 100vh;
   font-family:  Georgia, Times, "Times New Roman", serif;
   background-color: lightcoral;
  }
  input[type="text"],
  input[type="password"] {
   height: 30px;
   width: 150px;
   border: 1px solid rgba(0, 0, 0, 0.2);
  }
  input[type="submit"] {
   margin-top: 10px;
   width: 100px;
   font-size: 15px;
```

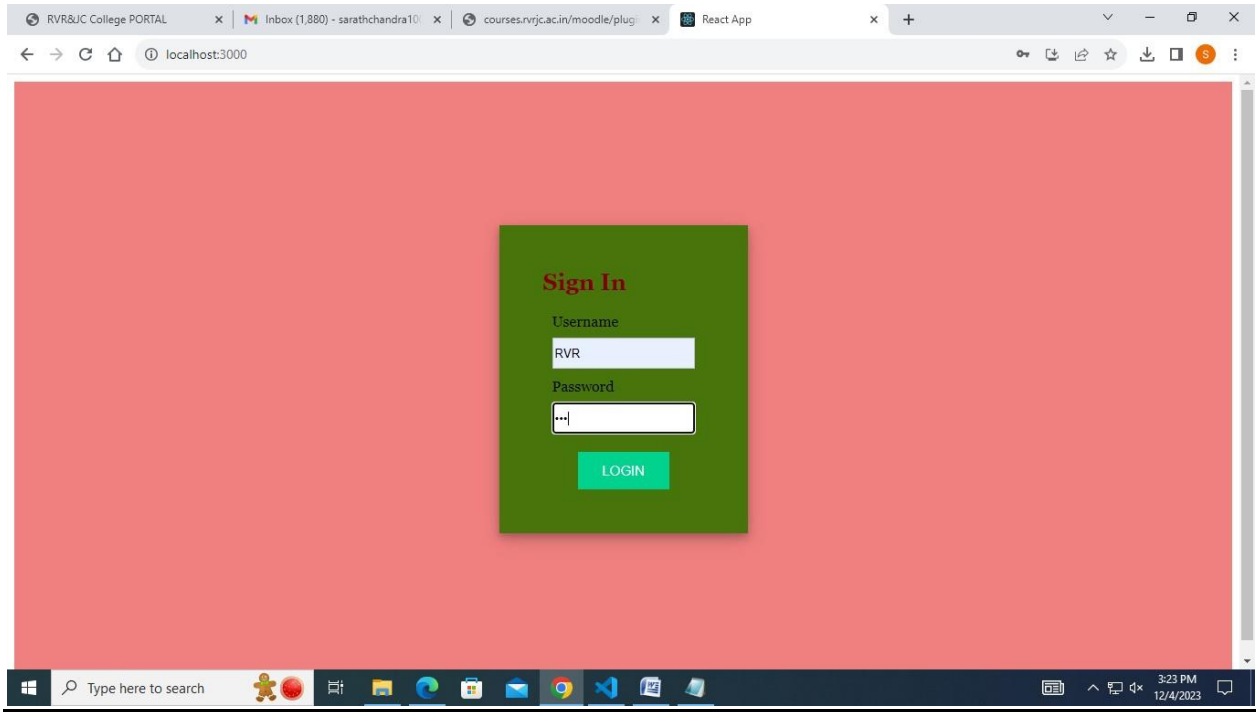
```
  background: #01d28e;
  border: 2px solid #01d28e;
  color: #fff;
  padding: 10px 20px;
}
input[type="submit"]:hover {
  background: #6cf0c2;
}
.button-container {
  display: flex;
  justify-content: center;
}
.login-form {
  background-color: rgb(71, 117, 12);
  padding: 3rem;
  box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2), 0 6px 20px 0 rgba(0, 0, 0, 0.19);
}
.list-container {
  display: flex;
}
.error {
  color: red;
  font-size: 20px;
}
```

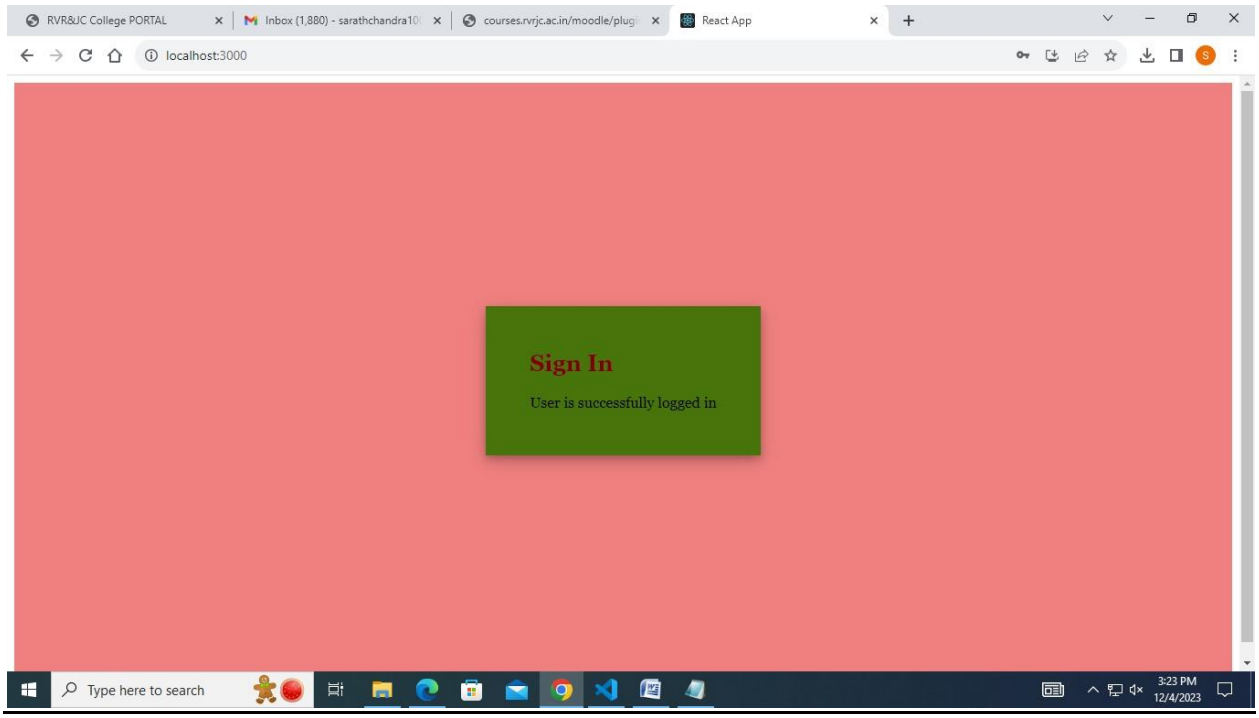
```
 .title {
   font-size: 25px;
   margin-bottom: 20px;
   color:maroon;
   font-weight: bold;
 }
 .input-container {
   display: flex;
   flex-direction: column;
   gap: 8px;
   margin: 10px;
 }
```

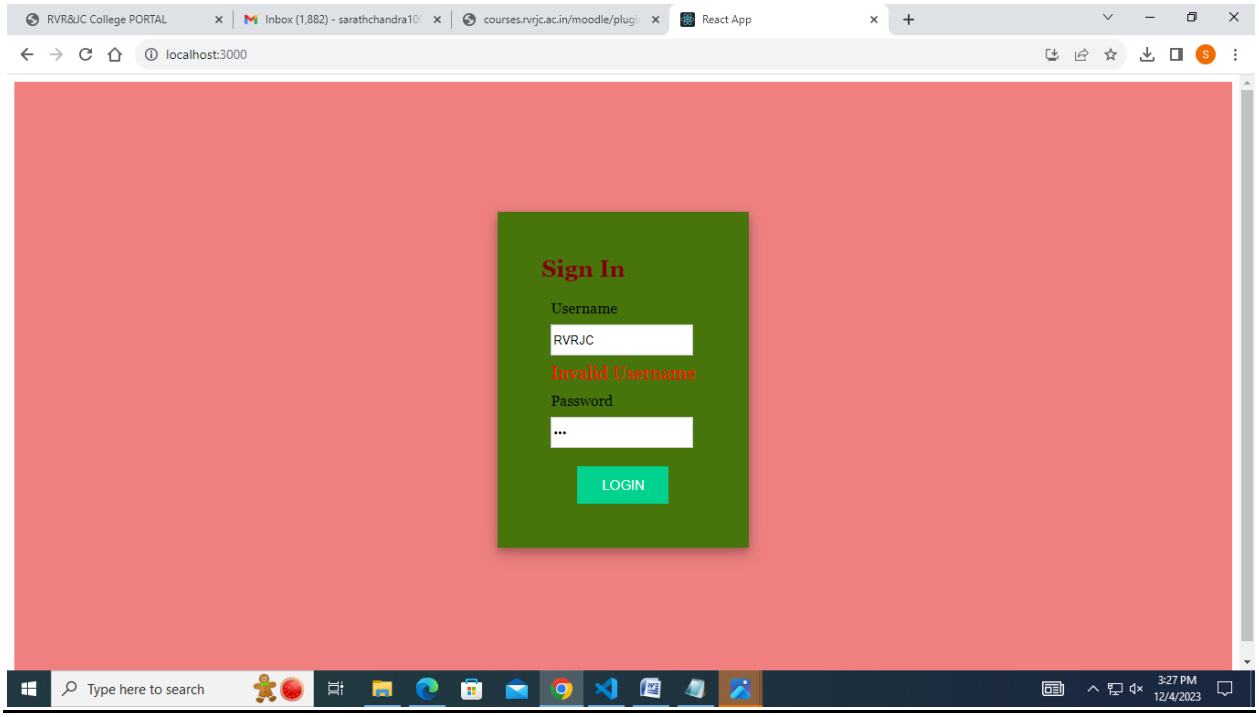
Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import reportWebVitals from './reportWebVitals';
import Login from './Login';
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <Login/>
  </React.StrictMode>
);
reportWebVitals();
```

Output :


RVR&JC College PORTAL
Inbox (1,880) - sarathchandra10
courses.rvrjc.ac.in/moodle/plug
React App
localhost:3000
Sign In
Username
RVR
Password
LOGIN
Type here to search
3:23 PM
12/4/2023


RVR&JC College PORTAL
Inbox (1,880) - sarathchandra10
courses.rvrjc.ac.in/moodle/plug
React App
localhost:3000
Sign In
User is successfully logged in
Type here to search
3:23 PM
12/4/2023


RVR&JC College PORTAL
Inbox (1,882) - sarathchandra10
courses.rvrjc.ac.in/moodle/plug
React App
localhost:3000
Sign In
Username
RVRJC
Invalid Username
Password
LOGIN
Type here to search
3:27 PM
12/4/2023

8. Create a simple Registration form with different input elements using event handling

Source Code :

Registration.js

```
import rvr from './rvr.jpg'
import './Registration.css'
import {useState} from 'react'

function Registration(){
   const [Name,setName]=useState("")
   const [Mobile,setMobile]=useState("")
   const [Age,setAge]=useState("")
   const [Email,setEmail]=useState("")
   const [Password,setPassword]=useState("")
   return (
<center>
<h1>R.V.R & J.C College of Engineering</h1>
<div className='main1'>
<div className='App'>
<header className='App-header'>
<img src={rvr} className='App-logo' alt='rvr'/>
</header>
</div>
<div className='main'>
```

```
<h2>STUDENT REGISTRATION FORM</h2>

<form>

<pre>

<b>Name          : </b><input className='input' placeholder='Name' onChange={(e)=>
setName(e.target.value)}/><br/>

<b>Mobile        : </b><input className='input' placeholder='Mobile' onChange={(e)=>
setMobile(e.target.value)}/><br/>

<b>Age           : </b><input className='input' placeholder='Age' onChange={(e)=>
setAge(e.target.value)}/><br/>

<b>Email         : </b><input className='input' placeholder='Email' onChange={(e)=>
setEmail(e.target.value)}/><br/>

<b>Password      : </b><input className='input' type='password' placeholder='Password'
onChange={(e)=> setPassword(e.target.value)}/><br/><br/>

<b>Confirm Password :  </b><input className='input' type='password' placeholder='Confirm
Password' onChange={(e)=> setPassword(e.target.value)}/><br/><br/>

<b>Address       :  </b>

<textarea rows="4" cols="31"  className='input' placeholder='Address' onChange={(e)=>
setPassword(e.target.value)}/><br/><br/>

<div className='gender'>

<b>Gender    :  </b>

<input type='radio' className='input1' name='gender'/>Male

<input type='radio' className='input1' name='gender'/>Female

<input type='radio' className='input1' name='gender'/>Transgender  <br/>

</div>

</pre>

<button type='submit' className='button'>Login</button>

<button type='submit' className='button'>SignUp</button>
```

```
</form>
</div>
</div>
</center>
   )
}
export default Registration;
```

Registration.css

```
.button{
   color: rgb(238, 247, 247);
   background-color: blue;
   text-align: center;
   margin-right: 10px;
   font-weight: bold;
   width: 100px;
   height: 30px;
   border-radius: 6px;
   border: 2px solid blue;
}
.App-header {
   min-height: 8vh;
   display: flex;
   flex-direction: column;
```

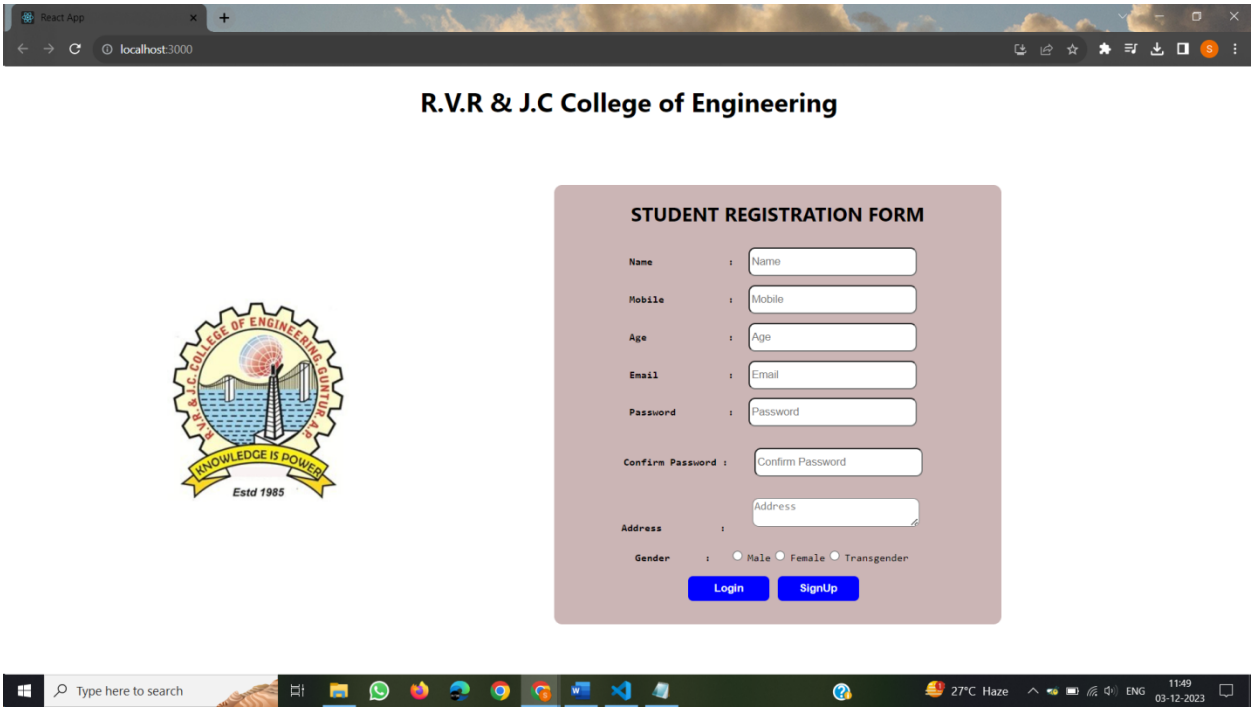
```
    align-items: center;
    justify-content: center;
    font-size: calc(10px + 2vmin);
    color: white;
  }
  .App{
    float: left;
    padding-left: 200px;
    padding-top: 200px;
  }
.input{
    width: 200px;
    height: 30px;
    margin: 1% 2%;
    border-radius: 8px;
}
.main{
    background-color: rgb(203, 181, 181);
    width: 550px;
    height: 540px;
    border-radius: 10px;
    margin: 60px;
    float: right;
}
```

```
.input1{
   margin:1% 1%;
}
.main1{
   padding-right: 250px;
}
```

Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import reportWebVitals from './reportWebVitals';
import Registration from './ Registration ';
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    < Registration />
  </React.StrictMode>
);
reportWebVitals();
```

Output :


React App
localhost:3000
R.V.R & J.C College of Engineering
STUDENT REGISTRATION FORM
Name :
Name
Mobile :
Mobile
Age :
Age
Email :
Email
Password :
Password
Confirm Password :
Confirm Password
Address :
Address
Gender :
Male
Female
Transgender
Login
SignUp
KNOWLEDGE IS POWER
Estd 1985
Type here to search
27°C Haze
ENG
11:49
03-12-2023

9.Build basic arithmatic calculator by using ReactJS.Use Statehook with button events.

Source Code :

Calculator.js

```
import React, { useState } from 'react';
import './Calculator.css';

function Calculator() {
  const [value, setValue] = useState('');
  return (
    <div className="container">
      <div className="calculator">
        <form action="">
          <div className='display'>
            <input type= "text" value={value}/>
            </div>
          <div>
            <input type="button" value="AC" onClick={e => setValue('')}/>
            <input type="button" value="DE" onClick={e => setValue(value.slice(0, -1))}/>
            <input type="button" value="." onClick={e => setValue(value + e.target.value)}/>
            <input type="button" value="/" onClick={e => setValue(value + e.target.value)}/>
          </div>
          <div>
            <input type="button" value="7" onClick={e => setValue(value + e.target.value)}/>
```

```
          <input type="button" value="8" onClick={e => setValue(value + e.target.value)}/>
          <input type="button" value="9" onClick={e => setValue(value + e.target.value)}/>
          <input type="button" value="*" onClick={e => setValue(value + e.target.value)}/>
        </div>
        <div>
          <input type="button" value="4" onClick={e => setValue(value + e.target.value)}/>
          <input type="button" value="5" onClick={e => setValue(value + e.target.value)}/>
          <input type="button" value="6" onClick={e => setValue(value + e.target.value)}/>
          <input type="button" value="+" onClick={e => setValue(value + e.target.value)}/>
        </div>
        <div>
          <input type="button" value="1" onClick={e => setValue(value + e.target.value)}/>
          <input type="button" value="2" onClick={e => setValue(value + e.target.value)}/>
          <input type="button" value="3" onClick={e => setValue(value + e.target.value)}/>
          <input type="button" value="-" onClick={e => setValue(value + e.target.value)}/>
        </div>
        <div>
          <input type="button" value="00" onClick={e => setValue(value + e.target.value)}/>
          <input type="button" value="0" onClick={e => setValue(value + e.target.value)}/>
          <input type="button" value="=" className='equal' onClick={e =>
setValue(eval(value))}/>
        </div>
      </form>
    </div>
```

```
    </div>
  );
}
export default Calculator;
```

Calculator.css

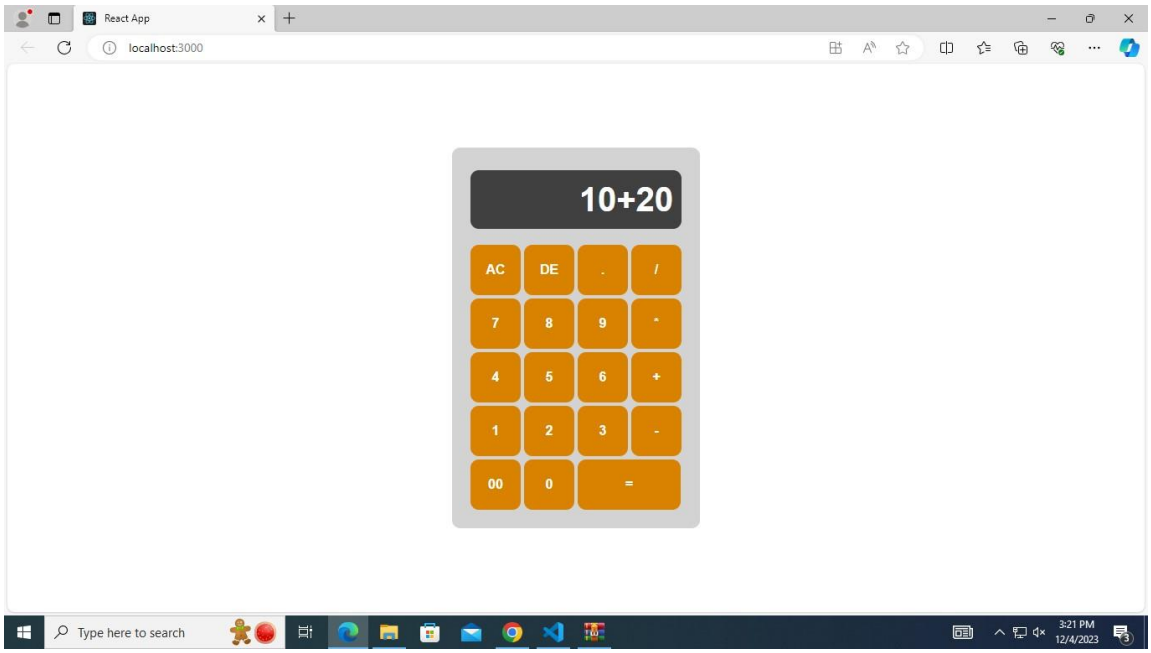
```
.container {
    width: 100%;
    height: 100vh;
    display: flex;
    align-items: center;
    justify-content: center;
    background: linear-gradient(140deg, rgb(255, 255, 255), rgb(255, 255, 255));
  }
  .calculator {
    padding: 20px;
    border-radius: 10px;
    background-color: rgb(211, 211, 211);
  }
  form input {
    outline: 0;
    width: 60px;
    height: 60px;
    font-size: 16px;
```

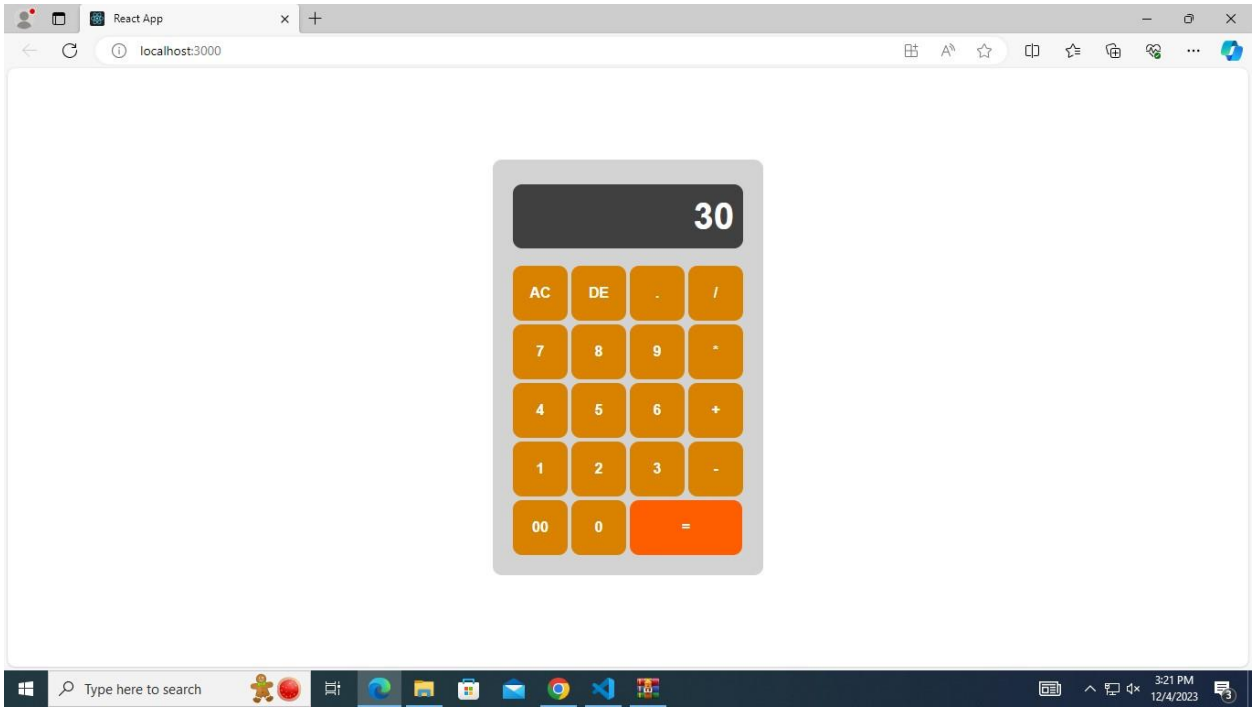
```
  background-color: rgb(216, 130, 0);
  margin: 2px;
  border-radius: 10px;
  color: white;
  font-weight: bolder;
  cursor: pointer;
}
form input[type="button"]:hover {
  background-color: rgb(255, 94, 0);
}
form .display {
  display: flex;
  justify-content: flex-end;
  margin: 5px 0px 15px 0px;
}
form .display input {
  text-align: right;
  font-size: 40px;
  padding: 5px 10px;
  background-color: rgb(64, 64, 64);
}
form input.equal{
  width: 123px;
}
```

Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import reportWebVitals from './reportWebVitals';
import Calculator from './Calculator.js';
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <Calculator/>
  </React.StrictMode>
);
reportWebVitals();
```

Output :




React App
localhost:3000
30
AC
DE
.
/
7
8
9
*
4
5
6
+
1
2
3
-
00
0
=
Type here to search
3:21 PM
12/4/2023

10.Build student name search form filter with basic search functionalities by using array.filter method and filter utilities.

Source Code :

Search.js

```
import React, { useState } from "react";
import logo from './logo.svg';
import "./Search.css";
function Search() {
  const list = [
    "Y20CS021",
    "Y20CS091",
    "Y20CS171",
    "Y20CE021",
    "Y20CE091",
    "Y20CE171",
    "Y20ECE021",
    "Y20ECE091",
    "Y20ECE171",
    "Y20IT021",
    "Y20IT091",
    "Y20IT171",
    "Ratna Babu",
    "Rishi Babu",
```

```
  "Deepak",
  "Jyothi",
  "Rama Raju",
  "Sunil",
  "Sitha",
  "Ramana",
  "Ratna Raju"
];
const [filterList, setFilterList] = useState(list);
const handleSearch = (event) => {
  if (event.target.value === "") {
    setFilterList(list);
    return;
  }
  const filteredValues = list.filter(
    (item) =>
      item.toLowerCase().indexOf(event.target.value.toLowerCase()) !== -1
  );
  setFilterList(filteredValues);
};
return (
  <div className="app11">
    <div className="App">
    <header className="App-header">
```

```
      <img src={logo} className="App-logo" alt="logo" />
    </header>
    <p >STUDENT INFORMATION  SEARCH!</p> </div>
    <div>
      <b>Search: <input  name="query" type="text" onChange={handleSearch} />
    </b></div>
    {filterList &&
      filterList.map((item, index) => (
        <div key={index}>{item}</div> //Display each item
      ))}
  </div>
 );
}
export default Search;
```

Search.css

```
.app11 {
  font-family: sans-serif;
  display: flex;
  align-items: center;
  justify-content: center;
  flex-direction: column;
  gap: 20px;
  height: 100vh;
```

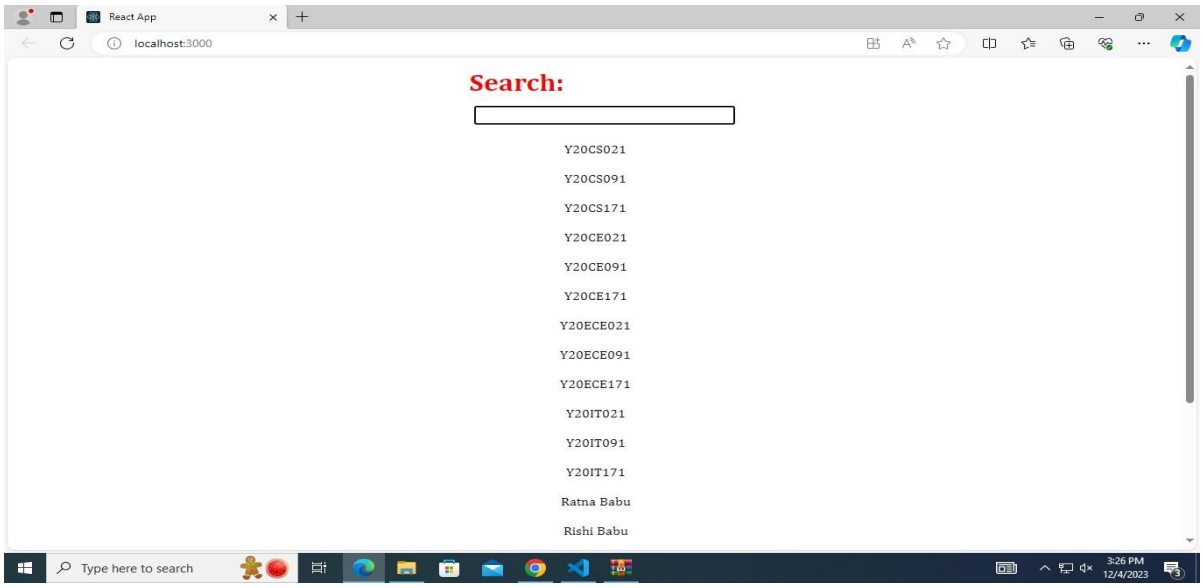
```
  font-family: Cambria, Cochin, Georgia, Times, "Times New Roman", serif;
 }
 b{
  color:red;
  font-size:2pc;
 }
 p{
  color:yellowgreen;
 }
 input {
  padding: 1%;
  width: 100%;
  margin: 1% 2%;
 }
```

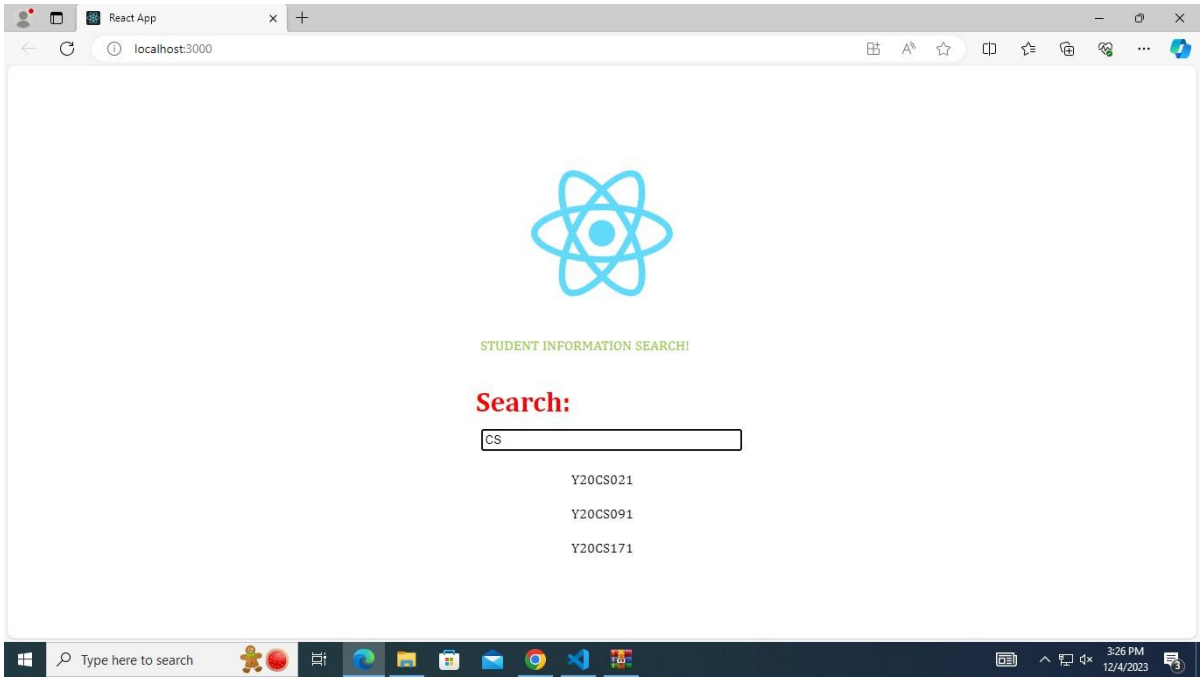
Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import reportWebVitals from './reportWebVitals';
import Search from './Search';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
 <React.StrictMode>
  <Search/>
```

```
 </React.StrictMode>
);
reportWebVitals();
```

Output :





11. create a React JS calendar along with events that are triggered based on user actions.

Source Code :

Calendar.js

```
import React, { useState } from "react";
import Calendar from "react-calendar";
import "react-calendar/dist/Calendar.css";
import "./Calendar.css";
function Calendar () {
  const allMonthValues = [
    "January",
    "February",
    "March",
    "April",
    "May",
    "June",
    "July",
    "August",
    "September",
    "October",
    "November",
    "December"
  ];
  const [selectedDate, setSelectedDate] = useState();
```

```
const [calendarText, setCalendarText] = useState(`No Date is selected`);
const handleDateChange = (value) => {
  setSelectedDate(value);
  setCalendarText(`The selected Date is ${value.toDateString()}`);
};
const handleYearChange = (value) => {
  const yearValue = value.getFullYear();
  setCalendarText(`${yearValue} Year  is selected`);
};
const handleMonthChange = (value) => {
  const monthValue = allMonthValues[value.getMonth()];
  setCalendarText(`${monthValue} Month  is selected`);
};
return (
  <div className="app1">
    <h2 className="calander-details">{calendarText}</h2>
    <Calendar
      onClickMonth={handleMonthChange}
      onClickYear={handleYearChange}
      onChange={handleDateChange}
      value={selectedDate}
    />
  </div>
);
```

```
}
export default Calendar;
```

Calendar.css

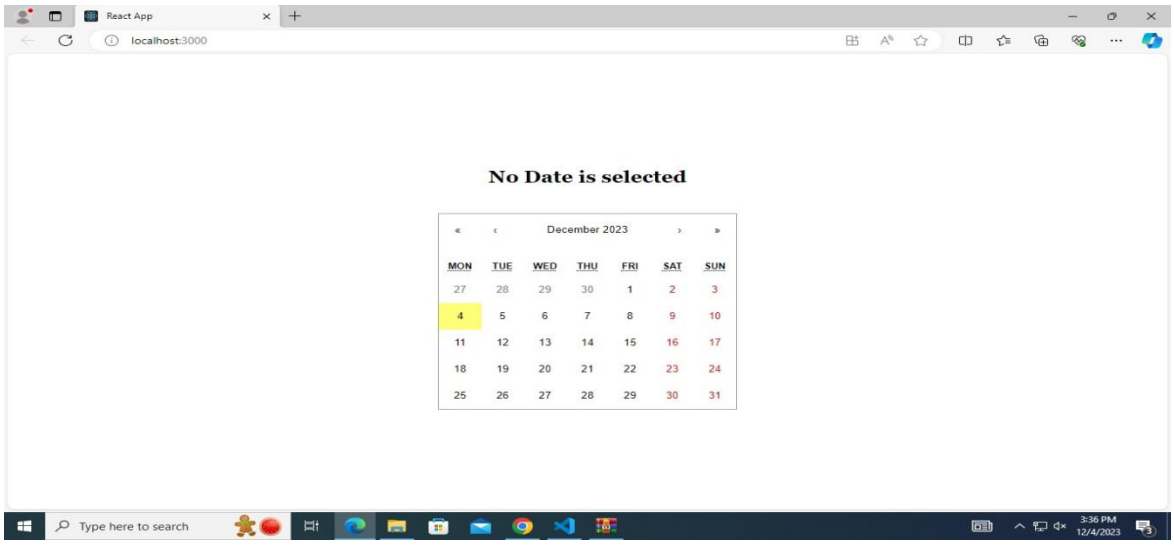
```
.app1 {
   display: flex;
   align-items: center;
   justify-content: center;
   flex-direction: column;
   gap: 20px;
   height: 100vh;
   color:black;
   font-family:Georgia, Times, "Times New Roman", serif;
 }
```

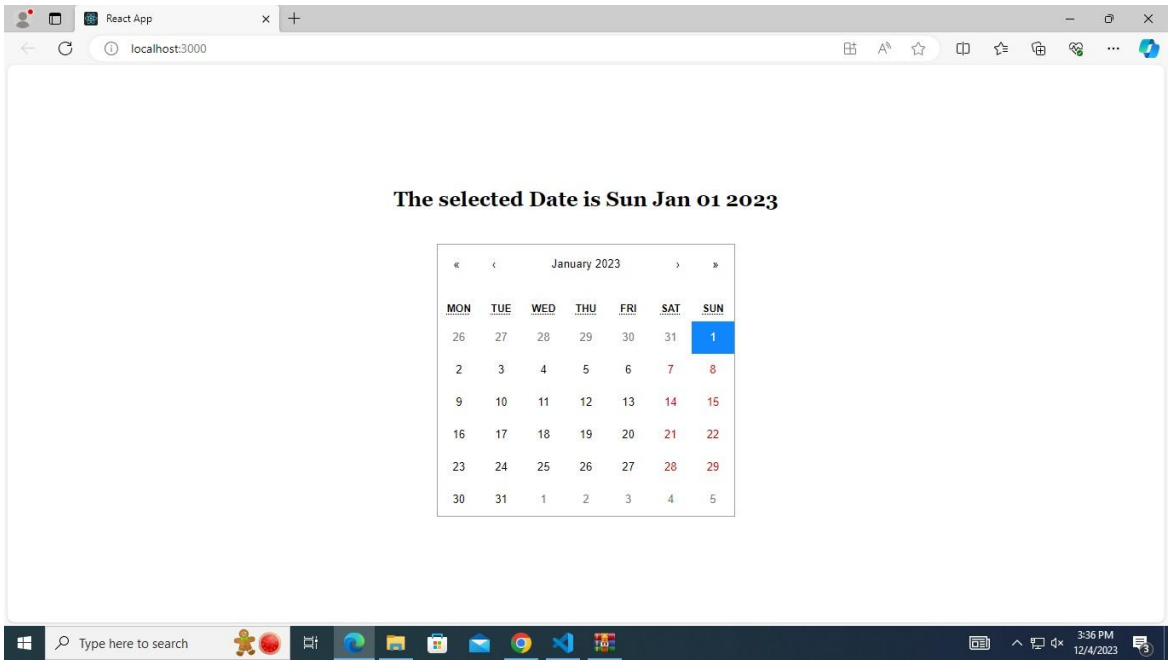
Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import reportWebVitals from './reportWebVitals';
import Calendar from './Calender/App7';
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
```

```
    <Calendar/>
  </React.StrictMode>
);
reportWebVitals();
```

Output :





12.Create a react router application by using React Router utilities

Source Code :

Router1.js

```
import "./Router1.css";
import {
  BrowserRouter as Router,
  Routes,
  Route,
  Navigate,
} from "react-router-dom";
import Home from "./Comp/Home";
import About from "./Comp/About";
import ContactUs from "./Comp/ContactUs";

function Router1() {
  return (
    <>
      <Router>
        <Routes>
          <Route
            exact
            path="/"
            element={<Home />}
```

```
            />
            <Route
              path="/about"
              element={<About />}
            />
            <Route
              path="/contactus"
              element={<ContactUs />}
            />
            <Route
              path="*"
              element={<Navigate to="/" />}
            />
          </Routes>
        </Router>
      </>
    );
}
export default Router1;
```

Router1.css

```
.App {
  text-align: center;
}
```

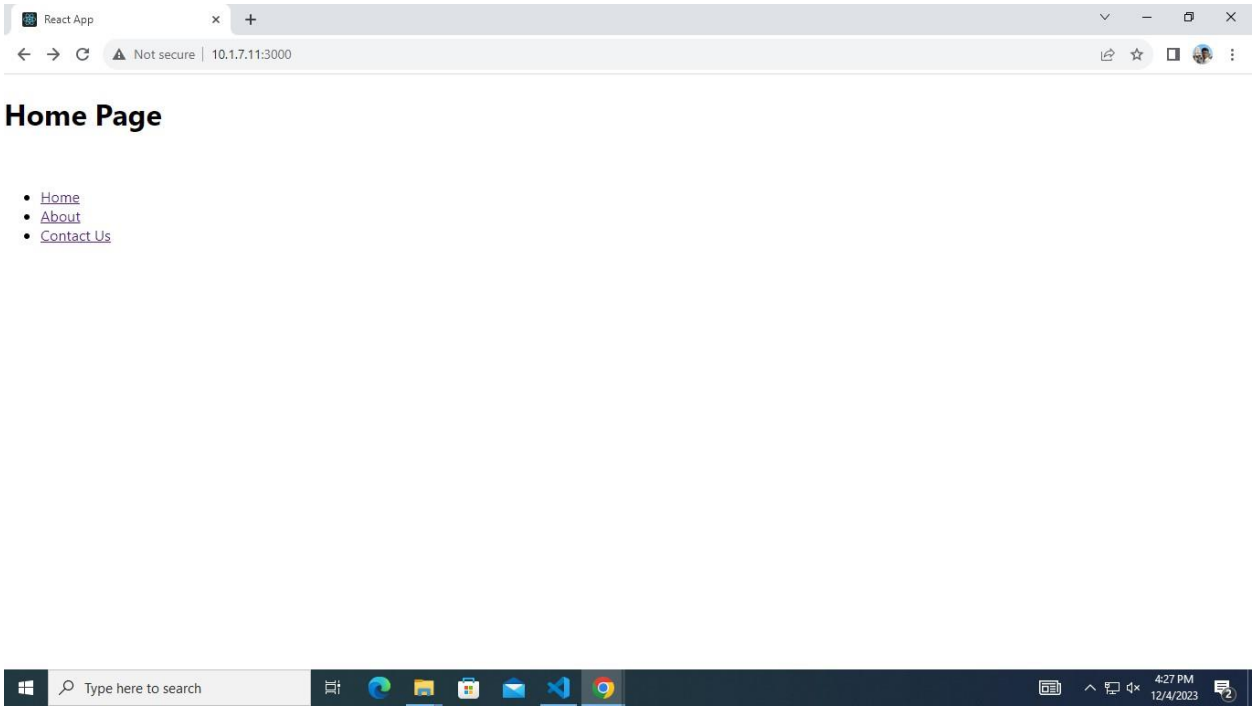
```
.App-logo {
  height: 40vmin;
  pointer-events: none;
}
@media (prefers-reduced-motion: no-preference) {
  .App-logo {
    animation: App-logo infinite 20s linear;
  }
}
.App-header {
  background-color: #060c01;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
  color: white;
}
.App-link {
  color: #61dafb;
}
@keyframes App-logo-spin {
  from {
    transform: rotate(0deg);
```

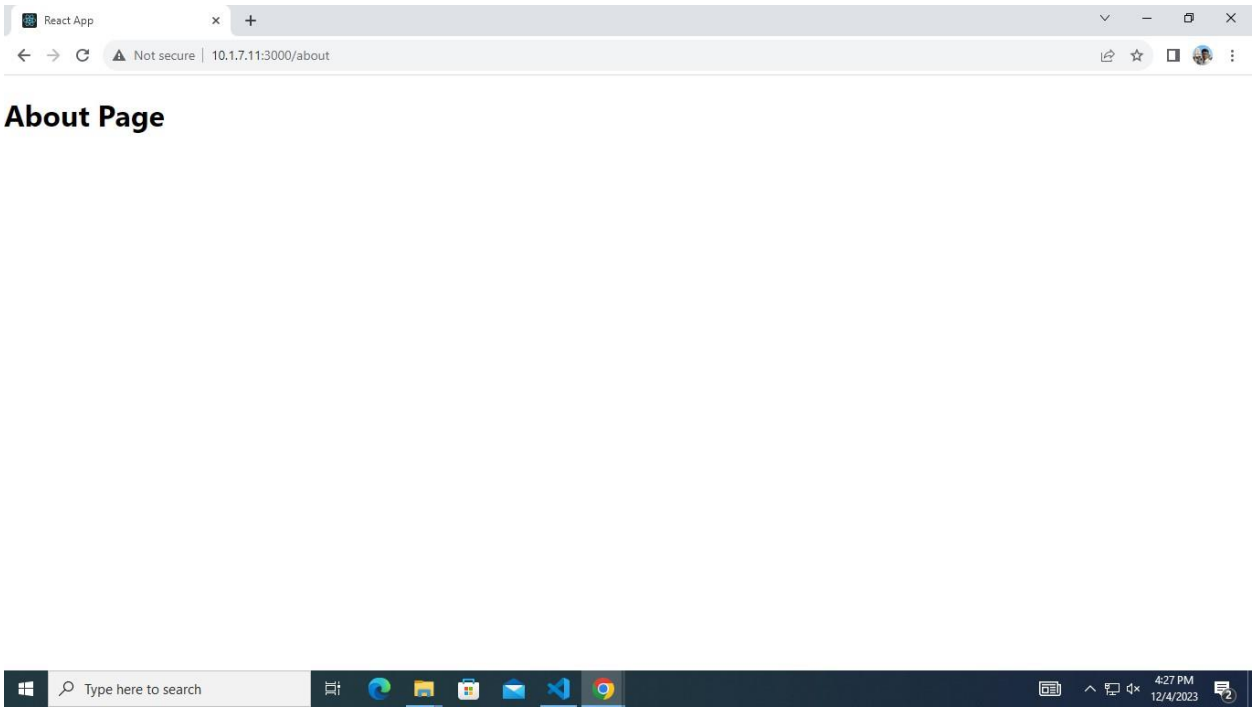
```
  }
  to {
    transform: rotate(360deg);
  }
}
```

Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import reportWebVitals from './reportWebVitals';
import Router1 from './Router1';
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <Router/>
  </React.StrictMode>
);
reportWebVitals();
```

Output :


React App
Not secure | 10.1.7.11:3000
Home Page
Home
About
Contact Us
Type here to search
4:27 PM
12/4/2023


React App
Not secure | 10.1.7.11:3000/about
About Page
Type here to search
4:27 PM
12/4/2023