**REACT LAB**

**Experiment No.1**

**Create react app by using npm utilities and develop an app to greet message**

**a)using vs code IDE to view the application**

**1.Test the node.js environment’s pre-requisite commands**

Node –v

Npm –v

**2. After successfully testing node environment commands , create a react project by using CRA (create-react-app) with the following two utilities**

npm init react-app welcome🡪 Now it successfully installs all react project dependencies and project files

We can also use

npx create-react-app welcome

**3.After that open the Visual Studio Code IDE**

In File 🡪 Open Folder 🡪 Load the project into VSCode IDE.

**4.Press ctrl+~ 🡪 It starts terminal**

There Run this 🡪 cd welcome

Now to run the created project ‘welcome’ use the command

* npm start

**5.The react.js folder structure is as follows:**

Y21cs102

|🡪node\_modules (all dependencies)

|🡪Public (Assets )

|🡪src

|🡪app.js

|🡪app.css

|🡪index.js

|🡪app.test.js

|🡪logo.svg

|🡪reportWebVitals.js

|🡪setupTests.js

|------>Few other files including package.json etc

**6. Edit app.js to make changes as follows :**

import './App.css';

function App() {

  return (

    <div className="App">

      <header className="App-header">

        <img src='https:\\collegelogo.org' className="App-logo" alt='logo-RVR' />

        <h1>

          WELCOME TO REACT - Y21CS102

        </h1>

      </header>

    </div>

  );

}

export default App;

**7. Edit app.css to make following changes:**

.App {

  text-align: center;

}

h1{

  color:#282c34;

}

.App-logo {

  height: 25vmin;

  pointer-events: none;

}

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

  .App-logo {

    animation: App-logo-spin infinite 0s linear;

  }

}

.App-header {

  /\* background-color: #282c34; \*/

  background-color:white;

  min-height: 100vh;

  display:flex;

  flex-direction: row;

  align-items: top;

  justify-content: center;

  font-size: calc(10px + 2vmin);

  color: white;

  align-items: start;

}

.App-link {

  color: #61dafb;

}

@keyframes App-logo-spin {

  from {

    transform: rotate(0deg);

  }

  to {

    transform: rotate(360deg);

  }

}

**8. index.js is as follows:**

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>

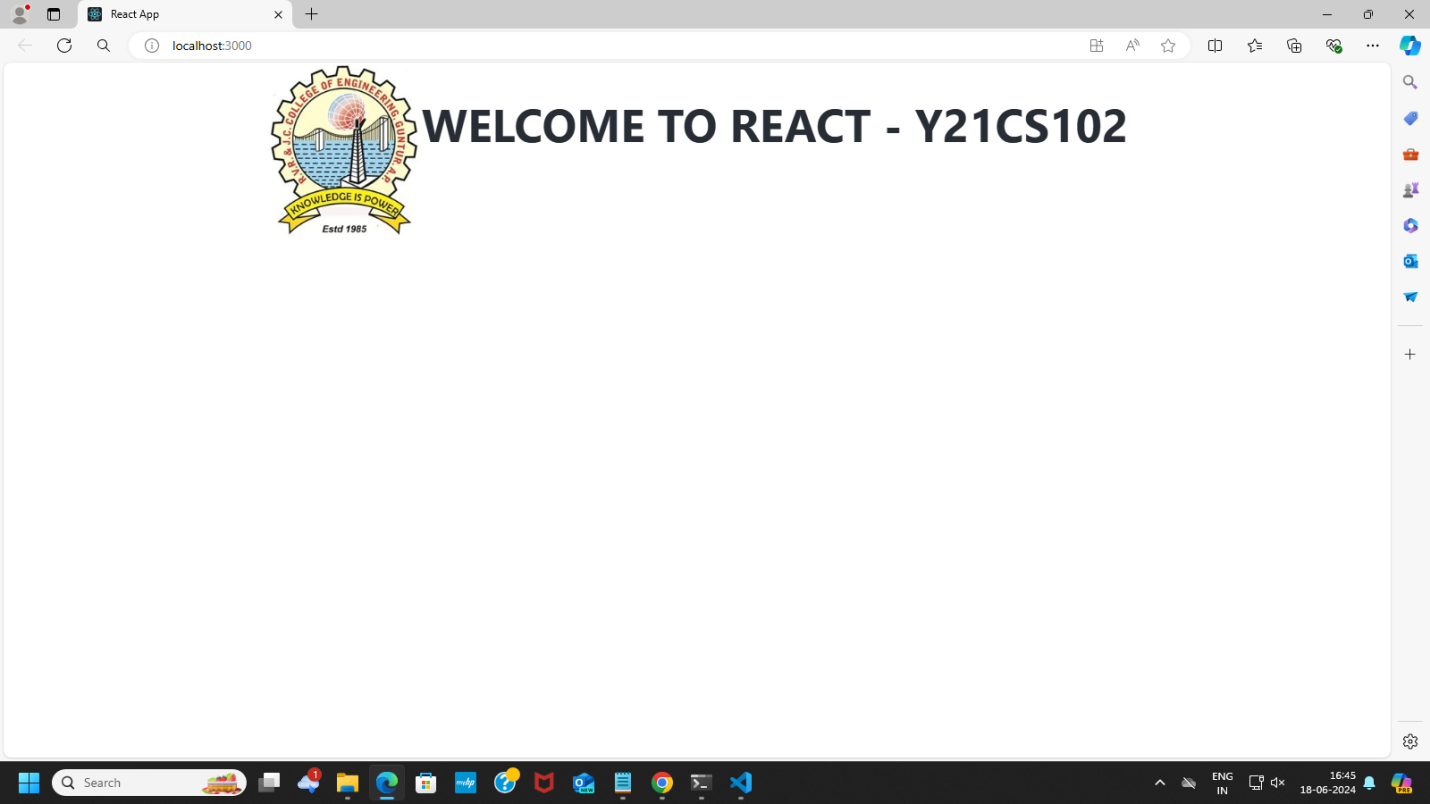
    <App />

  </React.StrictMode>

);

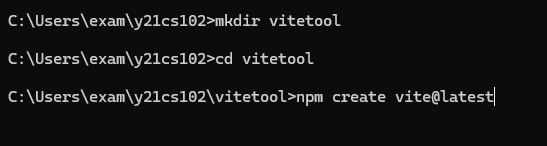
reportWebVitals();

**OUTPUT:**

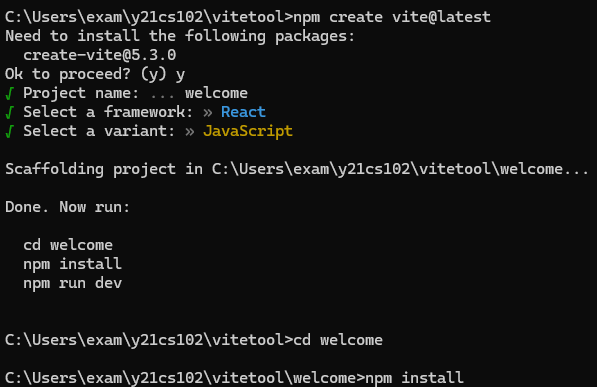
****

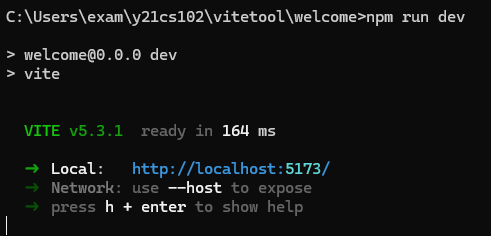
**b)With using vite framework tool to develop react app:**

**1.Create a new folder for clarity , named it vitetool and run the following commands to create a react app using npm create vite@latest**

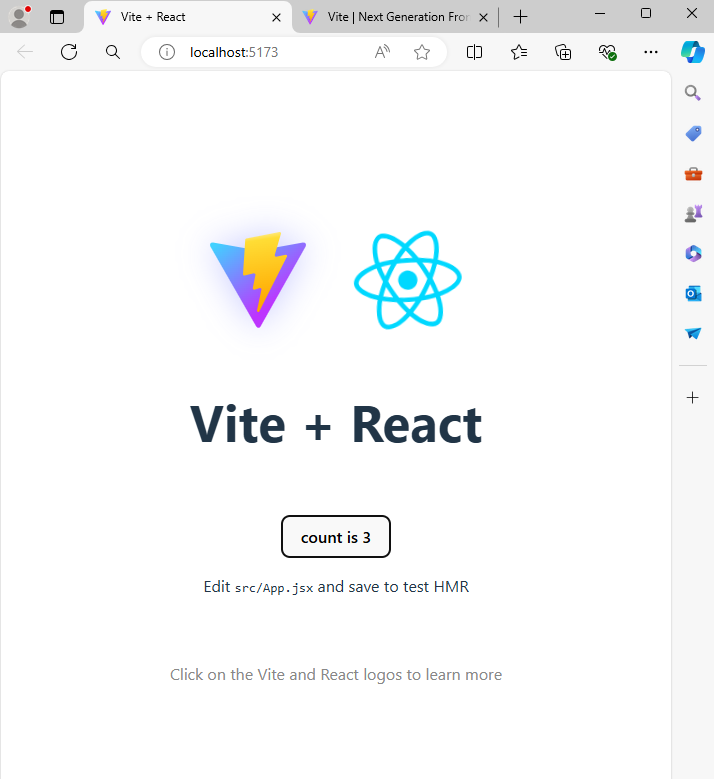
****

**2. Select your preferences as follows and then run the three recommended commands.**

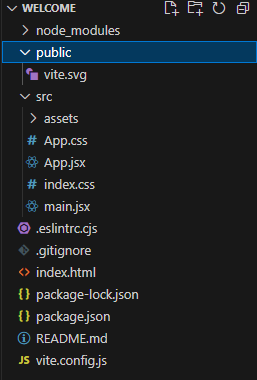
****

****

**The output will be as follows:**

****

**The folder structure is as follows:**

****

**main.jsx**

import React from 'react'

import ReactDOM from 'react-dom/client'

import App from './App.jsx'

import './index.css'

ReactDOM.createRoot(document.getElementById('root')).render(

  <React.StrictMode>

    <App />

  </React.StrictMode>,

)

**App.jsx**

import { useState } from 'react'

import './App.css'

function App() {

  const [count, setCount] = useState(0)

  return (

    <>

      <h1>Welcome to React JS - Y21CS102</h1>

    </>

  )

}

export default App

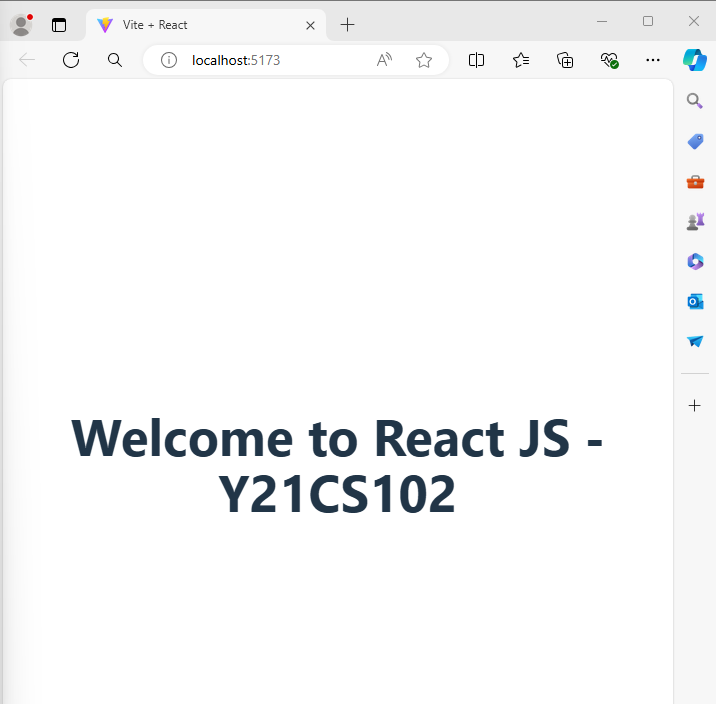
**App.css**

h1{

  text-align:center;

}

**OUTPUT:**

****

**Expt No.2**

**Create a React application using CDNs (Content Delivery Networks) (without react environment) and demonstrate it:**

**a)Using CDNs to test react application by displaying Greet Message using marquee tool using reactDOM properties**

<!DOCTYPE html>

<html>

<head>

    <title>Greet</title>

    <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" crossorigin></script>

</head>

<body>

    <div id="root"></div>

    <script type="text/babel">

        function Greeting() {

            const greet = () => {

                let now = new Date();

                let hrs = now.getHours();

                let name = document.getElementById('a').value;

                let greeting;

                if (hrs >= 5 && hrs < 12) {

                    greeting = "GOOD MORNING " + name;

                } else if (hrs >= 12 && hrs < 18) {

                    greeting = "GOOD AFTERNOON " + name;

                } else {

                    greeting = "GOOD EVENING " + name;

                }

                ReactDOM.render(

                    <div>

                        <marquee><p>{greeting}</p></marquee>

                    </div>,

                    document.getElementById('root')

                );

            };

            return (

                <div id='inputs'>

                    <label>Enter your name : </label>

                    <input type="text" id='a' /><br />

                    <button onClick={greet}>Enable Greet</button>

                </div>

            );

        }

        ReactDOM.render(

            <Greeting />,

            document.getElementById('root')

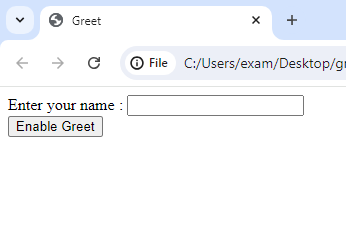
        );

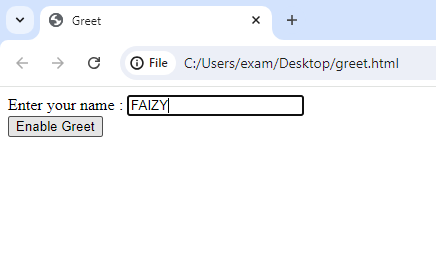
    </script>

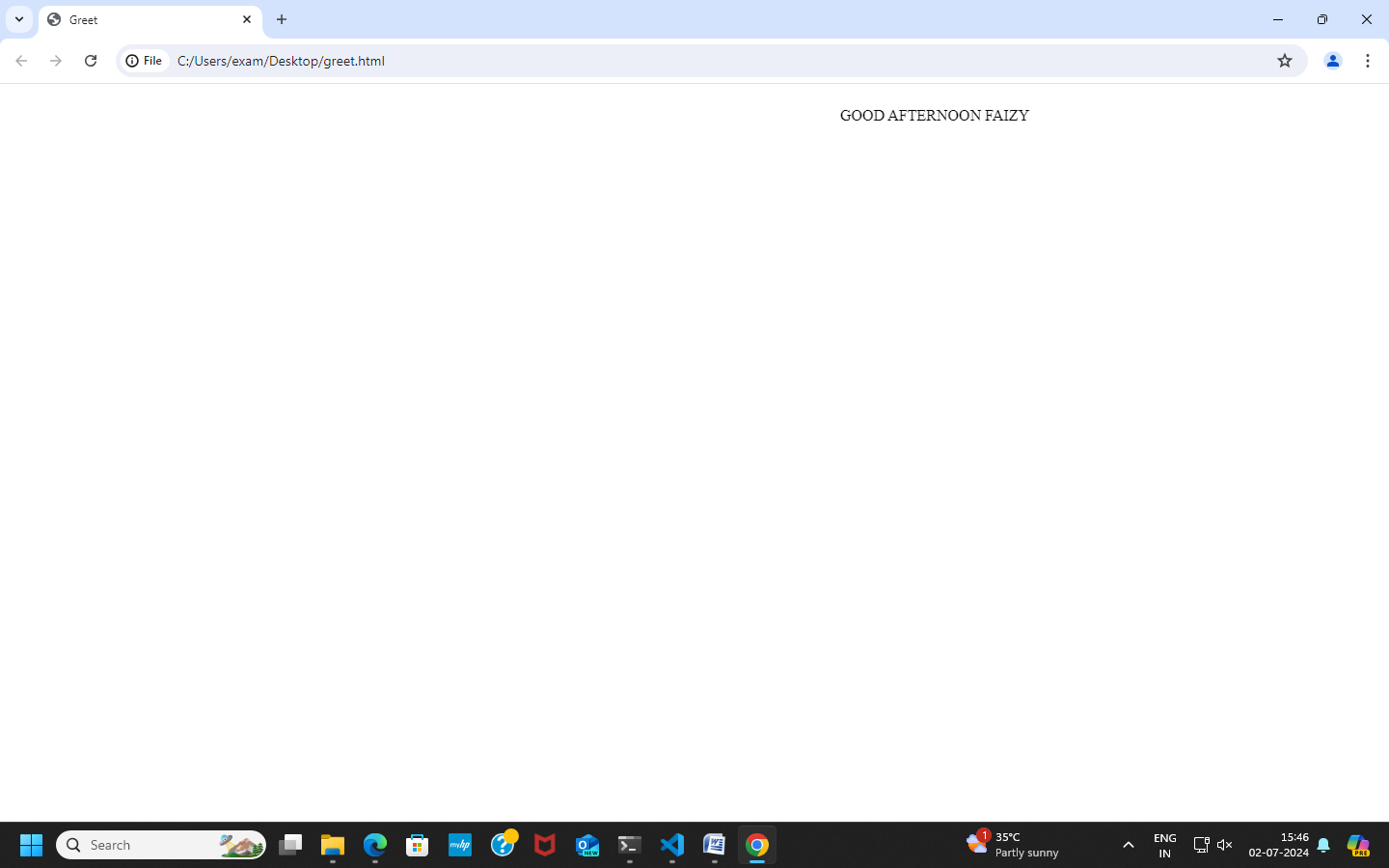
</body>

</html>

**OUTPUT:**

****

****

****

**b)Create maximum of two numbers function component with CDNs**

<!DOCTYPE html>

<html>

<head>

    <title>BiggestOfTwo</title>

    <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" crossorigin></script>

</head>

<body>

    <div id="root"></div>

    <script type="text/babel">

        function Inputs() {

            return (

                <div id='inputs'>

                    <input type="number" id='a' /><br />

                    <input type="number" id='b' /><br />

                    <button onClick={biggest}>Compare</button>

                </div>

            );

        }

        function biggest() {

            let a = document.getElementById('a').value;

            let b = document.getElementById('b').value;

            let lab;

            if (parseInt(a) > parseInt(b)) {

                lab = "num1 is bigger than num2";

            } else if (parseInt(b) > parseInt(a)) {

                lab = "num2 is bigger than num1";

            } else {

                lab = "num1 is equal to num2";

            }

            ReactDOM.render(<p>{lab}</p>, document.getElementById('root'));

        }

        ReactDOM.render(

            <div>

                <Inputs />

            </div>,

            document.getElementById('root')

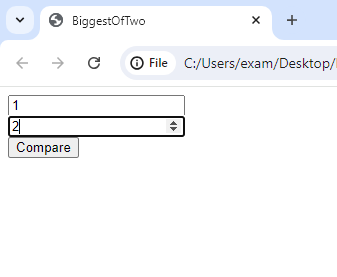
        );

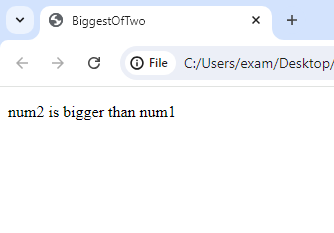
    </script>

</body>

</html>

**OUTPUT:**

****

****

**Expt No.3**

**Create a React application to access child components:**

**a)by using jsx framework**

**App.jsx**

import Child from './Child';

const App = () => {

  return (

    <div>

      <h1>Parent Component</h1>

      <Child message="Hello from child component!" />

    </div>

  );

};

export default App;

**Child.jsx**

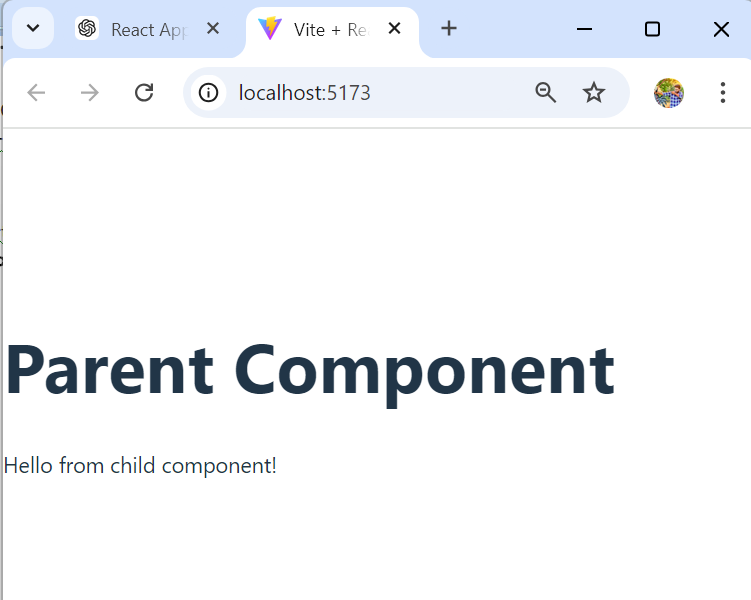
const Child = ({ message }) => {

  return <p>{message}</p>;

};

export default Child;

**OUTPUT:**

****

**b)without using jsx framework**

**App.js**

import React from 'react';

import Child from './Child';

const App = () => {

  return React.createElement(

    'div',

    null,

    React.createElement('h1', null, 'Parent Component'),

    React.createElement(Child, { message: 'Hello from child.js without using jsx framework!' })

  );

};

export default App;

**Child.js**

import React from 'react';

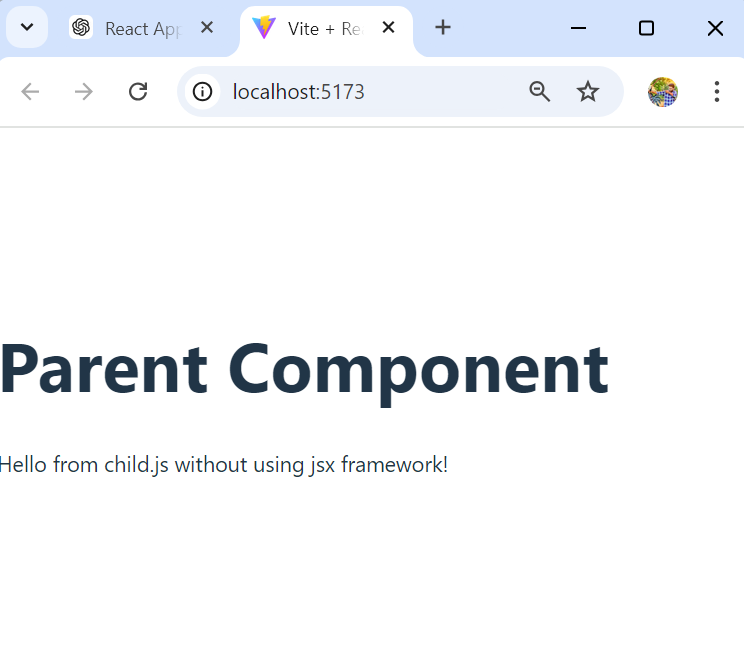
const Child = ({ message }) => {

  return React.createElement('p', null, message);

};

export default Child;

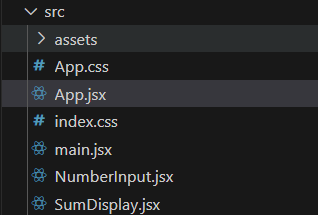
**OUTPUT:**

****

**Expt No.4**

**Create a simple React application by using both the class and function components and demonstrate it.The program should do the following**

* **A class component should take input dynamically.**
* **Function component add and display their sum.**

****

**App.jsx**import React from 'react';

import NumberInput from './NumberInput';

import SumDisplay from './sumDisplay';

class App extends React.Component {

  constructor(props) {

    super(props);

    this.state = {

      num1: 0,

      num2: 0,

    };

  }

  handleNumberChange = (num1, num2) => {

    this.setState({ num1, num2 });

  };

  render() {

    return (

      <div>

        <h1>Sum Calculator</h1>

        <NumberInput onNumberChange={this.handleNumberChange} />

        <SumDisplay num1={this.state.num1} num2={this.state.num2} />

      </div>

    );

  }

}

export default App;

**NumberInput.jsx**

import React from 'react';

class NumberInput extends React.Component {

  constructor(props) {

    super(props);

    this.state = {

      num1: '',

      num2: '',

    };

  }

  handleChange = (event) => {

    this.setState({ [event.target.name]: event.target.value }, () => {

      const { num1, num2 } = this.state;

      this.props.onNumberChange(Number(num1), Number(num2));

    });

  };

  render() {

    return (

      <div>

        <input

          type="number"

          name="num1"

          placeholder="Enter first number"

          value={this.state.num1}

          onChange={this.handleChange}

        />

        <input

          type="number"

          name="num2"

          placeholder="Enter second number"

          value={this.state.num2}

          onChange={this.handleChange}

        />

      </div>

    );

  }

}

export default NumberInput;

**SumDisplay.jsx**  
  
import React from 'react';

const SumDisplay = ({ num1, num2 }) => {

  const sum = num1 + num2;

  return (

    <div>

      <h2>Sum: {sum}</h2>

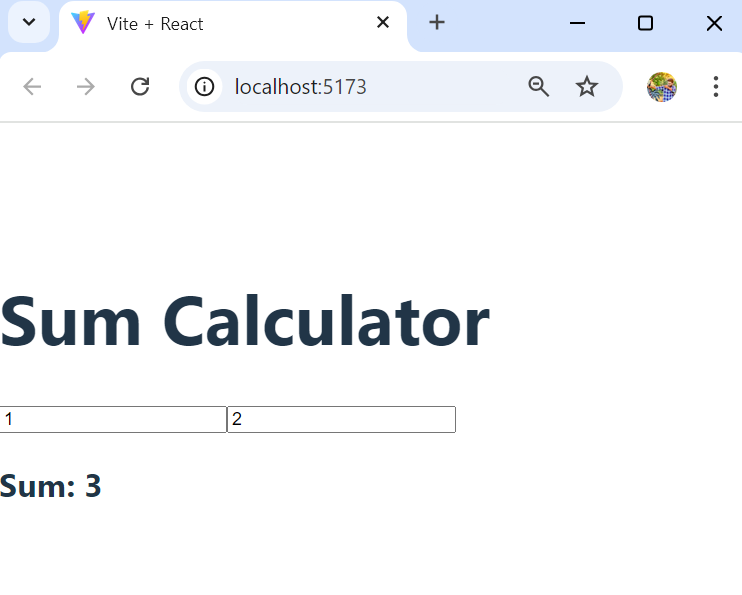
    </div>

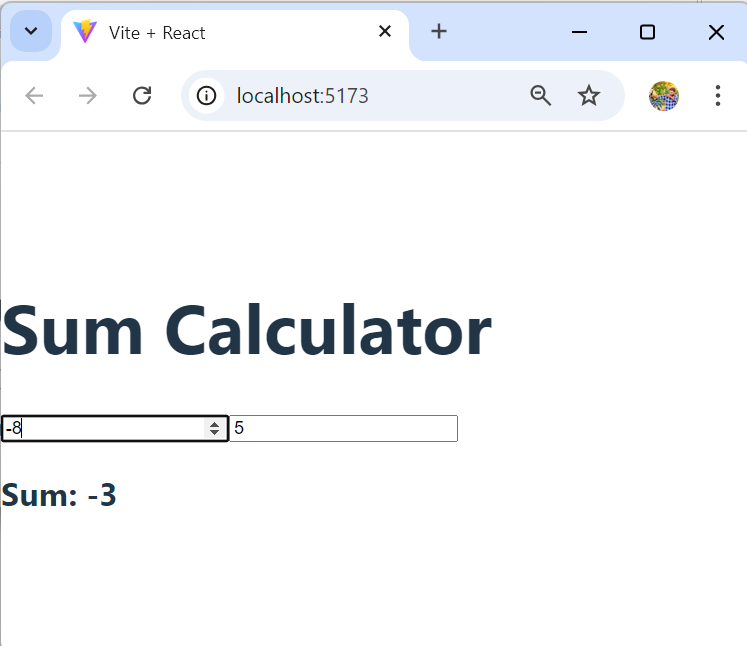
  );

};

export default SumDisplay;

**OUTPUT:**

****

****

**Expt No.5**

**Create a simple counter using react which increments or decrements dynamically as the user clicks on the button, using fundamental react concepts such as state variables, components etc.**

* **Create react state variable to store the counter value**
* **Declare JS functions to increment/decrement the value through setState() method.**
* **Add HTML button with onClick={} to JSX code.**

**App.jsx**

import React, { useState } from 'react';

function App() {

  const [count, setCount] = useState(0);

  const increment = () => {

    setCount(count + 1);

  };

  const decrement = () => {

    setCount(count - 1);

  };

  return (

    <div className = 'x'>

      <div><h1>Counter: {count}</h1></div>

      <button onClick={increment}>Increment</button>

      <button onClick={decrement}>Decrement</button>

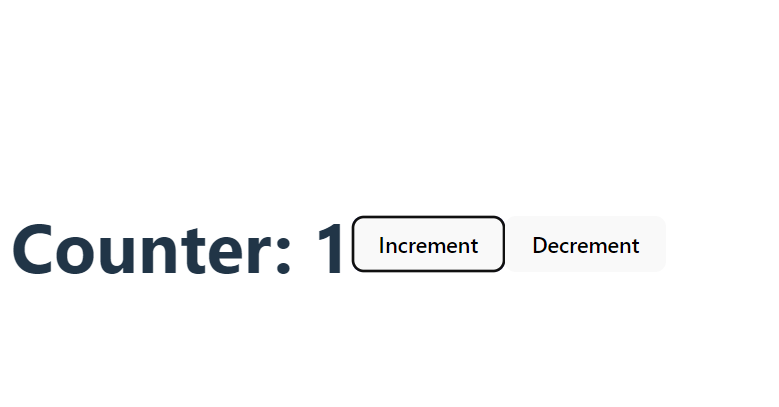
    </div>

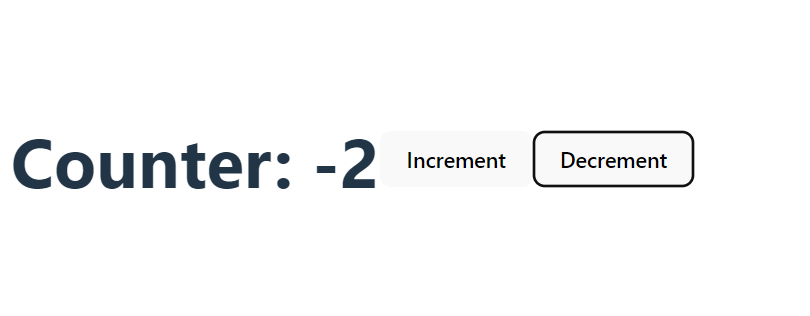
  );

}

export default App;

**Output:**

****

****

**Expt No.6**

**Create a react JS application by using different react styling components:**

**a) React inline styles**

**b)React external styles**

**c)React component styles**

**Incremeter.js**

import React from "react";

import './myStyle.css';

class Incrementer extends React.Component{

    constructor(props){

        super(props);

        this.state={

            counter:0

        };

    }

Incr=()=>{

   this.setState({

    counter:this.state.counter+1

   });

};

Decr=()=>{

    this.setState({

     counter:this.state.counter-1

    });

 };

   render(){

        const mysty1={

        fontSize: '40px',

        color: 'blue'

         }

         const mysty2={

            fontSize: '50px',

            color: 'green'

             }

             const mysty3={

                fontSize: '40px',

                color: 'green'

                 }

        return (

            <div >

        <h1 align="center" style={{ border: '1px solid black', padding: '10px',color:'green',fontSize:'50px' }} >ReactJs::COUNTER APP</h1>

        <p align="center" className={`$ first second third`}> counter={this.state.counter}</p>

        <div align="center">

<button onClick={this.Incr} style={mysty3}>Increment</button>

  <button onClick={this.Decr} style={mysty1}>Decrement</button></div>

        </div>)

   }

}

export default Incrementer;

**App.css**

.first{

    color:red;

}

.second{

    font-size:75px;

}

.third{

    font-weight:bold;

    font-family: Arial, Helvetica, sans-serif;

}

    .button {

        margin: 10px 0;

        padding: 5px 10px;

      }

      .input1 {

        display: block;

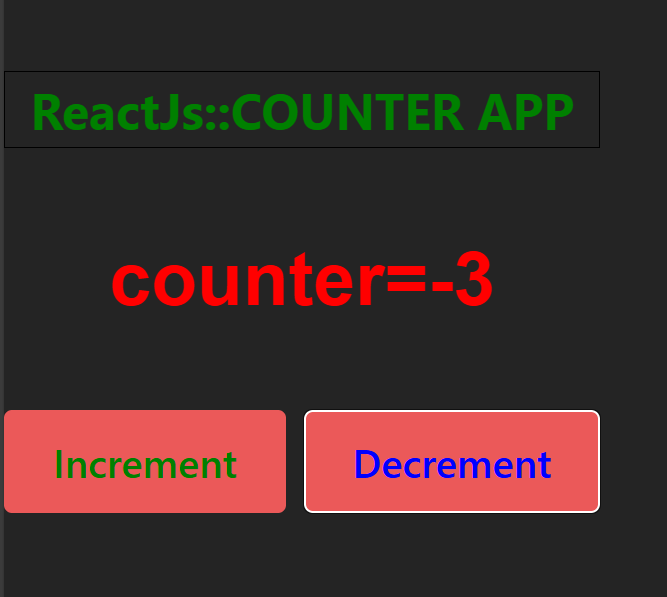
        padding: 5px;

        margin-bottom: 5px;

        width: 120px;border-color: brown;

      }

**OUTPUT:**



**Expt No.7**

**Create a simple login page using reactJS.**

* **Display login page with username,password,login button.**
* **Users can input dynamically on the form**
* **Validate username and password**
* **Display an error message when login fails else show success message**

**App.js**

import { useState } from 'react';

import './App.css';

function App() {

  const [username, setUsername] = useState('');

  const [password, setPassword] = useState('');

  const [message, setMessage] = useState('');

  const [isLoggedIn, setIsLoggedIn] = useState(false);

  function validate() {

    if (username === 'Farhaan' && password === 'Farhaan') {

      setMessage('Login Successful');

      setIsLoggedIn(true);

    } else {

      setMessage('Login Unsuccessful');

    }

  }

  return (

    <div className="x">

      {!isLoggedIn ? (

        <>

          <label>Username: </label>

          <input

            type='text'

            className="fields"

            value={username}

            onChange={(e) => setUsername(e.target.value)}

          /><br />

          <label>Password: </label>

          <input

            type='password'

            className="fields"

            value={password}

            onChange={(e) => setPassword(e.target.value)}

          /><br />

          <button onClick={validate}>Login</button><br />

          <label>{message}</label>

        </>

      ) : (

        <div>{message}</div>

      )}

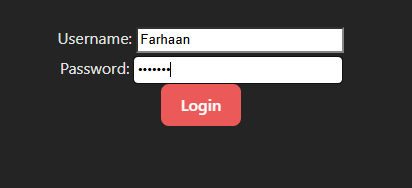
    </div>

  );

}

export default App;

**OUTPUT:**





**Expt No.8**

**Create a simple registration react form with different from input elements using react event handling concepts and also demonstrate by using the following concepts:**

1. **useState & useEffect hook mechanism**
2. **Provide Single Page using ReactDOM**

**App.js**

import './App3.css';

import { useState } from 'react';

function App3() {

  const [firstName, setFirstName] = useState('');

  const [lastName, setLastName] = useState('');

  const [mobile, setMobile] = useState('');

  const [age, setAge] = useState('');

  const [email, setEmail] = useState('');

  const [password, setPassword] = useState('');

  const [gender, setGender] = useState('');

  const [address, setAddress] = useState('');

  const [department, setDepartment] = useState(''); // New state for department

  const [termsAccepted, setTermsAccepted] = useState(false); // New state for terms acceptance

  const [file, setFile] = useState(null); // New state for file upload

  const [dateOfBirth, setDateOfBirth] = useState(''); // New state for date of birth

  const [experienceLevel, setExperienceLevel] = useState(5); // New state for experience level

  const [favoriteColor, setFavoriteColor] = useState('#000000'); // New state for favorite color

  const [isSubmitted, setIsSubmitted] = useState(false);

  const handleSubmit = (event) => {

    event.preventDefault();

    setIsSubmitted(true);

  };

  const WebForm1 = (

    <div align="center">

      <header className="App-header">

        <h1>R.V.R.& J.C.COLLEGE OF ENGINEERING::GUNTUR</h1>

      </header>

      <p align="center" style={{ border: '0px solid black', padding: '10px', color: 'green', fontSize: '40px' }}>

        STUDENT FORM REGISTRATION!

      </p>

      <form onSubmit={handleSubmit}>

        <input placeholder="First Name" onChange={(e) => setFirstName(e.target.value)} /><br/><br/>

        <input placeholder="Last Name" onChange={(e) => setLastName(e.target.value)} /><br/><br/>

        <input placeholder="Mobile Number" onChange={(e) => setMobile(e.target.value)} /><br/><br/>

        <input placeholder="Age" onChange={(e) => setAge(e.target.value)} /><br/><br/>

        <input placeholder="Email" onChange={(e) => setEmail(e.target.value)} /><br/><br/>

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

        <div>

          <label>Gender:</label>

          <label>

            <input type="radio" value="Male" checked={gender === 'Male'} onChange={(e) => setGender(e.target.value)} /> Male

          </label>

          <label>

            <input type="radio" value="Female" checked={gender === 'Female'} onChange={(e) => setGender(e.target.value)} /> Female

          </label>

          <label>

            <input type="radio" value="Other" checked={gender === 'Other'} onChange={(e) => setGender(e.target.value)} /> Other

          </label><br/><br/>

        </div>

        <textarea

          placeholder="Address"

          value={address}

          onChange={(e) => setAddress(e.target.value)}

          rows="4"

          cols="30"

        /><br/><br/>

        <select onChange={(e) => setDepartment(e.target.value)}>

          <option value="">Select Department</option>

          <option value="CSE">Computer Science</option>

          <option value="ECE">Electronics</option>

          <option value="ME">Mechanical</option>

        </select><br/><br/>

        <label>

          <input type="checkbox" checked={termsAccepted} onChange={(e) => setTermsAccepted(e.target.checked)} />

          I agree to the terms and conditions

        </label><br/><br/>

        <input type="file" onChange={(e) => setFile(e.target.files[0])} /><br/><br/>

        <input type="date" onChange={(e) => setDateOfBirth(e.target.value)} /><br/><br/>

        <label>Experience Level:</label>

        <input type="range" min="1" max="10" value={experienceLevel} onChange={(e) => setExperienceLevel(e.target.value)} /><br/><br/>

        <label>Favorite Color:</label>

        <input type="color" value={favoriteColor} onChange={(e) => setFavoriteColor(e.target.value)} /><br/><br/>

        <button type="submit">Submit</button>

      </form>

    </div>

  );

  return (

    <div>{isSubmitted ?

      <div>

        <p>Name: {firstName} {lastName}</p>

        <p>Age: {age}</p>

        <p>Mobile No: {mobile}</p>

        <p>Email: {email}</p>

        <p>Gender: {gender}</p>

        <p>Address: {address}</p>

        <p>Department: {department}</p>

        <p>Terms Accepted: {termsAccepted ? "Yes" : "No"}</p>

        <p>Date of Birth: {dateOfBirth}</p>

        <p>Experience Level: {experienceLevel}</p>

        <p>Favorite Color: <span style={{ backgroundColor: favoriteColor }}> {favoriteColor} </span></p>

        {file && <p>File: {file.name}</p>}

      </div> : WebForm1}

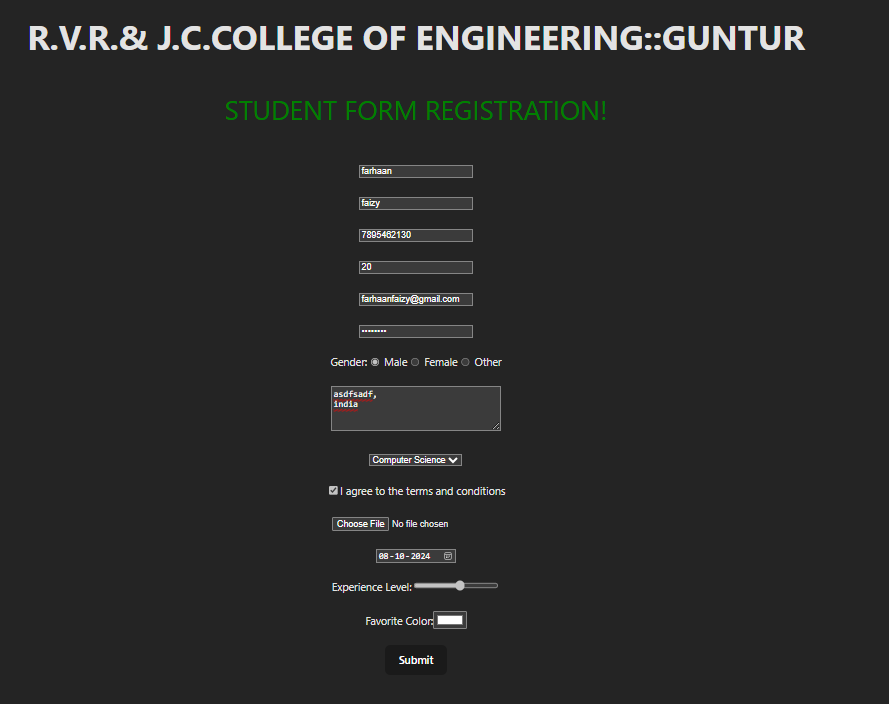
    </div>

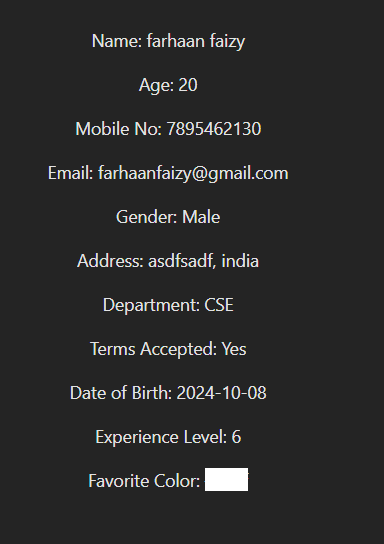
  );

}

export default App3;

**OUTPPUT:**

****

****

**Expt No.9**

**Build a simple arithmetic functional calculator with react Hooke’s & grid single page application.(using lambda react JS expressions).**

**APP.JSX**

import React, { useState } from 'react';

import './Style8.css';

function App() {

  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 App;

**APP.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 {

    border: none;

    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;

    flex: 1;

    font-size: 40px;

    padding: 5px 10px;

    background-color: rgb(64, 64, 64);

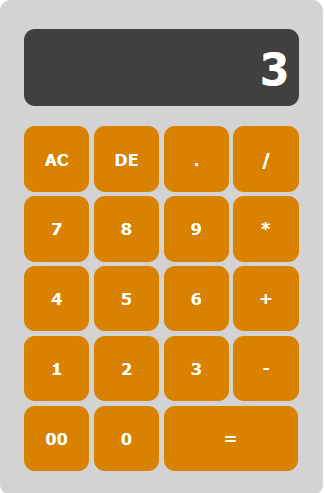
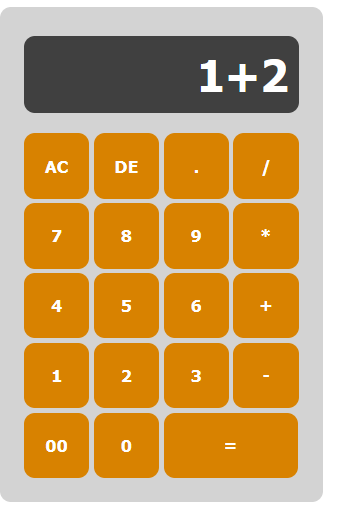
  }

  form input.equal{

    width: 123px;

  }

**OUTPUT:**

****

**Expt No.10**

**Build a student name search react form filter that performs a simple search filter functionality to display a list based on search query entered by the user**

* **Declare state for input query .**
* **Create HTML input text for entering search item & update state in onChange function .**
* **Add Array.filter() method on list of items with search .**

**APP.jsx**

import React, { useState } from "react";

import logo from './logo1.svg';

import "./App3.css";

function App11() {

  const list = [

    "Y21CS021",

    "Y21CS091",

    "Y21CS171",

    "Ratna Babu",

    "Rishi Babu",

  ];

  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 align="center">

      <div >

      <p align="center" style={{ border: '0px solid black', padding: '5px',color:'white',fontSize:'40px' }}>STUDENT INFORMATION  SEARCH!</p> </div>

      <div>

        <b style={{color:'white',fontSize:'20px'}}>Search(Name/Phono/Regd.No):

          <input  name="query" type="text" onChange={handleSearch}

          style={{ border: '5px solid white', padding: '10px',width:'500px',color:'white',fontSize:'40px' }}/>

      </b></div>

      {filterList &&

        filterList.map((item, index) => (

          <div key={index}>{item}</div>

        ))}

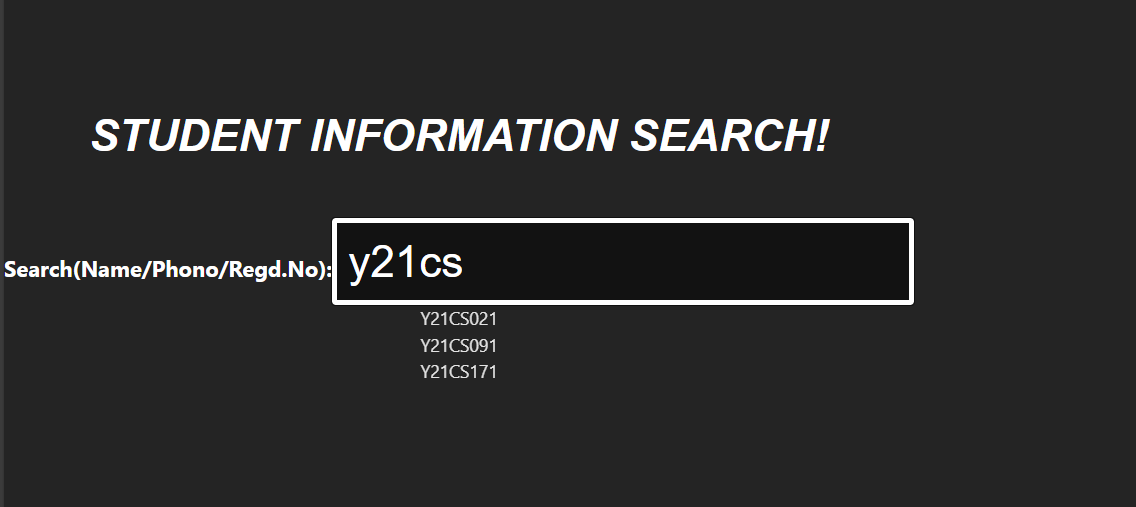
    </div>

  );

}

export default App11;

**OUTPUT:**

****

**Expt No.11**

**Create a ReactJS calendar along with the events that are triggered based on users actions. The following must be included:**

* **onChange() must be triggered each time you change the date on calendar.**
* **onClickYear() is executed when year is selected by user .**
* **onClickMonth() is executed when month is selected by user.**

**APP.jsx**

import React, { useState } from "react";

import Calendar from "react-calendar";

import "react-calendar/dist/Calendar.css";

import "./Style7.css";

function App7() {

  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) => {

    setCalendarText(`${value.getFullYear()} Year is selected`);

  };

  const handleMonthChange = (value) => {

    const month = value.toLocaleString("default", { month: "long" });

    setCalendarText(`${month} Month is selected`);

  };

  return (

    <div className="app1">

      <h2 className="calander-details">{calendarText}</h2>

      <Calendar

        onClickMonth={handleMonthChange}

        onClickYear={handleYearChange}

        onChange={handleDateChange}

        value={selectedDate}

      />

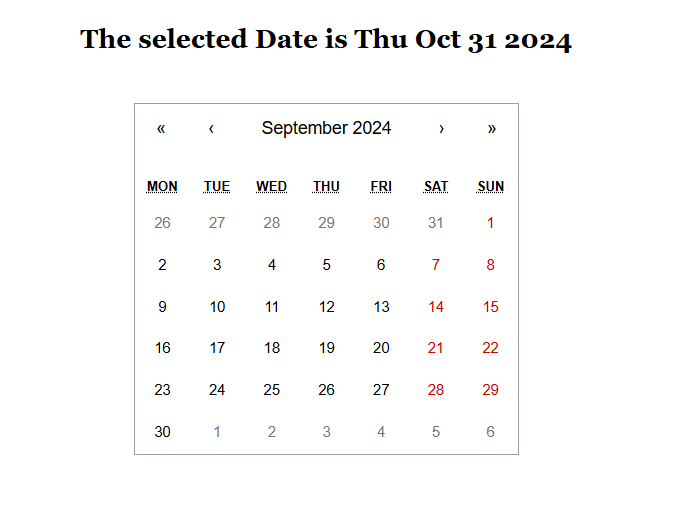
    </div>

  );

}

export default App7;

**OUTPUT:**

****

**Expt No.12**

**Implementing react routing by using <basic browser router> to develop the following functionalities:**

* **Create multiple react Component Apps.**
* **Add router to each component.**
* **Allow users to navigate between different pages.**

**APP.jsx**

// src/App.jsx

import React from 'react';

import { BrowserRouter as Router, Route, Routes, Link } from 'react-router-dom';

import './App.css'

import App1 from './App1';

import App2 from './App2';

import App3 from './App3';

import App4 from './App4';

import App5 from './App5';

function App() {

  return (

    <Router>

      <div className="app-container">

        <nav className="navbar">

          <ul>

            <li><Link to="/app1">App1</Link></li>

            <li><Link to="/app2">App2</Link></li>

            <li><Link to="/app3">App3</Link></li>

            <li><Link to="/app4">App4</Link></li>

            <li><Link to="/app5">App5</Link></li>

          </ul>

        </nav>

        <div className="content">

          <Routes>

             <Route path="/app1" element={<App1 />} />

            <Route path="/app2" element={<App2 />} />

            <Route path="/app3" element={<App3 />} />

            <Route path="/app4" element={<App4 />} />

            <Route path="/app5" element={<App5 />} />

          </Routes>

        </div>

      </div>

    </Router>

  );

}

export default App;

**OUTPUT:**

