**DAY -2**

**LOGIN FORM**

**Login.js:**

import React, { useState } from 'react';

import { Link,useNavigate } from 'react-router-dom';

import './style.css';

const styles = {

container: {

display: 'flex',

justifyContent: 'center',

alignItems: 'center',

height: '100vh',

fontFamily: 'Arial, sans-serif',

backgroundImage: 'url("https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSGU2DepJn76b\_Pb09BrQEHCC2LrAoVPzG0c2EMmFlEsS8V6W33IjWk8jvKJfGPHrBi9Hw&usqp=CAU")',

backgroundSize: 'cover',

backgroundRepeat: 'no-repeat',

},

form: {

display: 'flex',

color: 'black',

flexDirection: 'column',

width: '400px',

padding: '60px',

paddingtop:'20px',

border: '1px solid #ccc',

borderRadius: '10px',

backgroundColor: 'white',

},

input: {

marginBottom: '10px',

width: '100%',

height: '40%',

padding: '15px',

fontSize: '20px',

},

button: {

padding: '5px',

width: '30%',

fontSize: '18px',

alignItems: 'center',

backgroundColor: 'green',

color: 'white',

border: 'none',

cursor: 'pointer',

},

error: {

color: 'red',

marginBottom: '10px',

},

};

const Login = () => {

const navigate = useNavigate();

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

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

const [error, setError] = useState('');

const handleSubmit = (e) => {

e.preventDefault();

// Add your login logic here

// if (email === 'example@admin.com' && password === 'password') {

// Successful login, navigate to dashboard

setError('');

setEmail('');

setPassword('');

alert('Login successful!');

navigate('/Home')

//} else {

//Invalid credentials, display error message

//setError('Invalid email or password');

//}

};

return (

<div style={styles.container}>

<form style={styles.form} onSubmit={handleSubmit}>

<h2><center>LOGIN</center></h2>

<br></br>

{error && <p style={styles.error}>{error}</p>}

<input

style={styles.input}

type="email"

placeholder="Email Id"

value={email}

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

required

/>

<input

style={styles.input}

type="password"

placeholder="Password"

value={password}

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

required

/>

<br></br>

<center>

<button style={styles.button} type="submit">Login</button></center>

<br></br>

<p><b>Not a Member? </b><Link to="/signup">Sign up</Link></p>

</form>

</div>

);

};

export default Login;

SIGNUP FORM

**Signup.js:**

import React, { useState } from 'react';

import { Link,useNavigate } from 'react-router-dom';

const styles = {

container: {

display: 'flex',

justifyContent: 'center',

alignItems: 'center',

height: '100vh',

fontFamily: 'Arial, sans-serif',

backgroundImage: 'url("https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSGU2DepJn76b\_Pb09BrQEHCC2LrAoVPzG0c2EMmFlEsS8V6W33IjWk8jvKJfGPHrBi9Hw&usqp=CAU")',

backgroundSize: 'cover',

backgroundRepeat: 'no-repeat',

},

form: {

display: 'flex',

flexDirection: 'column',

width: '500px',

padding: '20px',

border: '1px solid #ccc',

borderRadius: '10px',

backgroundColor: 'white',

},

input: {

marginBottom: '10px',

padding: '10px',

fontSize: '16px',

},

button: {

padding: '5px',

width:'20%',

height:'70%',

alignItems: 'center',

fontSize: '16px',

backgroundColor: 'green',

color: 'black',

border: 'none',

cursor: 'pointer',

},

error: {

color: 'red',

marginBottom: '10px',

},

};

const Signup = () => {

const navigate = useNavigate();

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

const [name, setName] = useState('');

const [age, setAge] = useState('');

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

const [state, setState] = useState('');

const [country, setCountry] = useState('');

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

const [confirmPassword, setConfirmPassword] = useState('');

const [error, setError] = useState('');

const handleSubmit = (e) => {

e.preventDefault();

// Add your signup logic here

if (password === confirmPassword) {

// Successful signup, navigate to login page

setError('');

setEmail('');

setName('');

setAge('');

setGender('');

setState('');

setCountry('');

setPassword('');

setConfirmPassword('');

alert('Signup successful!');

navigate('/Home')

} else {

// Passwords do not match, display error message

setError('Passwords do not match');

}

};

return (

<div style={styles.container}>

<form style={styles.form} onSubmit={handleSubmit}>

<h2><center> REGISTER </center></h2>

<br></br>

{error && <p style={styles.error}>{error}</p>}

<input

style={styles.input}

type="text"

placeholder="Name"

value={name}

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

required

/>

<input

style={styles.input}

type="number"

placeholder="Age"

value={age}

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

required

/>

<input

style={styles.input}

type="text"

placeholder="Gender"

value={gender}

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

required

/>

<input

style={styles.input}

type="email"

placeholder="Email"

value={email}

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

required

/>

<input

style={styles.input}

type="password"

placeholder="Password"

value={password}

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

required

/>

<input

style={styles.input}

type="password"

placeholder="Confirm Password"

value={confirmPassword}

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

required

/>

<input

style={styles.input}

type="text"

placeholder="State"

value={state}

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

required

/>

<input

style={styles.input}

type="text"

placeholder="Country"

value={country}

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

required

/>

<br></br>

<center>

<button style={styles.button} type="submit">Signup</button></center>

<br></br>

<center>

<p><b>Already a member?</b> <Link to="/Login">Login</Link></p></center>

</form>

</div>

);

}

export default Signup;

**Style.css:**

.container {

display: 'flex';

align-items: 'center';

justify-content: 'center';

height: '100vh';

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

}

form {

display: 'flex';

flex-direction: 'column';

width: '300px';

padding: '20px';

padding-top: 20%;

padding-bottom: 20%;

padding-left: 45%;

padding-right: 20%;

border: '1px solid #ccc';

border-radius: '4px';

}

input {

margin-Bottom: '10px';

padding: '10px';

font-Size: '20px';

}

button {

padding: '10px';

font-Size: '16px';

background-color: '#3293a8';

color: 'black';

border: 'none';

cursor: 'pointer';

}

.error {

color: 'black';

margin-Bottom: '10px';

}

body{

background-color: white;

background-size: cover;

}

**Front.js:**

import React from 'react';

import { Link} from 'react-router-dom';

import './front.css';

const Front = () =>{

return(

<div className="hero--row">

<div className="hero--wrapper">

<h1> Megaventory Management</h1>

<p>The goal of Megaventory management is to have the right products in the right place at the right time.</p>

<div id="hero--buttons">

<Link to="/login"><button id="finance-btn">Login</button></Link>

<Link to="/signup"><button id="refer-btn">Signup</button></Link>

</div>

</div>

</div>

);

}

export default Front;

**Front.css:**

\* {

box-sizing: border-box;

}

body {

height: 100%;

box-sizing: inherit;

color:black;

font-family: 'Open Sans', sans-serif;

font-size: 18px;

background-color: aliceblue;

}

.hero--wrapper {

margin: 0 auto;

width: 800px;

}

h1 {

font-weight: 100;

font-size: 48px;

text-align: center;

margin: 25px 0 0 0;

}

p {

text-align: center;

}

#hero--buttons {

margin: 0px auto;

text-align: center;

}

#finance-btn {

width: 250px;

height: 50px;

background: #333;

border: 0;

display: inline-block;

margin-right: 10px;

color: #fff;

}

#refer-btn {

width: 250px;

height: 50px;

background: #d8bc5a;

border: 0;

display: inline-block;

color: #fff;

}

#gform\_7, #gform\_63 {

display: none;

}

#gform\_7 {

margin: 0 auto;

width: 300px;

padding: 10px;

}

#gform\_63 {

margin: 0 auto;

width: 300px;

padding: 10px;

}

.i {

margin: 20px auto;

}

**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()

**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;

}