## RegisterPage.jsx

```
import React, { useState } from 'react';
function RegisterPage({ onRegisterSuccess, onLoginClick }) {
 const [formData, setFormData] = useState({
   password: '',
   gender: '',
   country: '',
   terms: false,
 });
 const [errors, setErrors] = useState({});
 const validateEmail = (email) => /^\S+@\S+\.\S+$/.test(email);
 const validatePassword = (password) => password.length >= 6;
 const handleChange = (e) => {
   const { name, value, type, checked } = e.target;
   setFormData((prev) => ({
     ...prev,
   }));
 const handleSubmit = (e) => {
   e.preventDefault();
   if (!formData.password || !validatePassword(formData.password))
newErrors.password = 'Password must be at least 6 characters';
   if (!formData.gender) newErrors.gender = 'Select a gender';
   if (!formData.country) newErrors.country = 'Select a country';
```

```
setErrors(newErrors);
   if (Object.keys(newErrors).length === 0) {
      const existingUsers = JSON.parse(localStorage.getItem('users')) ||
[];
      const userExists = existingUsers.find((user) => user.email ===
formData.email);
     if (userExists) {
       alert('User with this email already exists.');
     const updatedUsers = [...existingUsers, formData];
      localStorage.setItem('users', JSON.stringify(updatedUsers));
     onRegisterSuccess();
   <form onSubmit={handleSubmit}>
     <h2>Register</h2>
       <label>Name:</label>
       <input name="name" value={formData.name} onChange={handleChange}</pre>
       <label>Email:</label>
       <input name="email" value={formData.email} onChange={handleChange}</pre>
```

```
<label>Password:</label>
       <input name="password" type="password" value={formData.password}</pre>
onChange={handleChange} />
        {errors.password && {errors.password}}
        <label>Gender:</label>
        <input type="radio" name="gender" value="male"</pre>
onChange={handleChange} /> Male
        <input type="radio" name="gender" value="female"</pre>
onChange={handleChange} /> Female
       {errors.gender && {errors.gender}}
        <label>Country:</label>
       <select name="country" value={formData.country}</pre>
onChange={handleChange}>
         <option value="">--Select--</option>
         <option value="India">India</option>
         <option value="USA">USA</option>
         <option value="UK">UK</option>
        {errors.country && {errors.country}}
        <label>Accept Terms:
       <input type="checkbox" name="terms" checked={formData.terms}</pre>
onChange={handleChange} />
       <label>Bio:</label>
       <textarea name="bio" value={formData.bio}</pre>
onChange={handleChange}></textarea>
```

```
js

SetFormData((prev) => ({
    ...prev,
    [name]: type === 'checkbox' ? checked : value
}));

• setFormData is used to update the form state.

• prev is the previous state of formData.

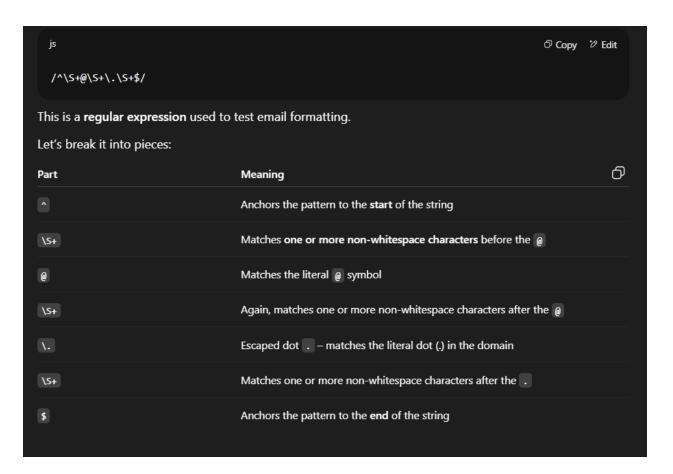
• ...prev keeps all the previous values.

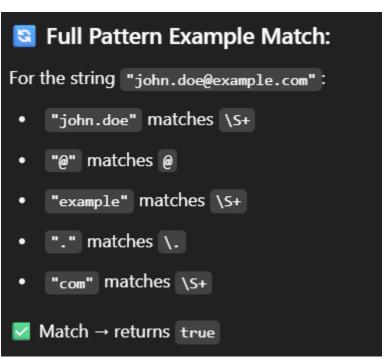
• [name]: This uses computed property names to dynamically set the value of the input field that triggered the event.

• type === 'checkbox' ? checked : value:

• If the input is a checkbox, use checked (a boolean).

• Otherwise, use the input's value.
```





## **Login Page**

```
import React, { useState } from 'react';

function LoginPage({ onRegisterClick, onLoginSuccess }) {
  const [email, setEmail] = useState('');
  const [password, setPassword] = useState('');
  const [error, setError] = useState('');

  const validateEmail = (email) => /^\S+@\S+\.\S+$/.test(email);

  const handleSubmit = (e) => {
    e.preventDefault();

  if (!validateEmail(email)) {
      setError('Invalid email format');
      return;
    }

  if (password.length < 6) {
      setError('Password must be at least 6 characters');
      return;
    }
}</pre>
```

```
const users = JSON.parse(localStorage.getItem('users')) || [];
   const matchedUser = users.find(
     (user) => user.email === email && user.password === password
   );
     setError('');
     onLoginSuccess (matchedUser);
     setError('Invalid email or password');
   <form onSubmit={handleSubmit}>
     <h2>Login</h2>
       <label>Email:</label>
       <input value={email} onChange={(e) => setEmail(e.target.value)} />
       <label>Password:</label>
       <input type="password" value={password} onChange={(e) =>
setPassword(e.target.value) } />
     {error && {error}}
     <button type="submit">Login</button>
       <button type="button" onClick={onRegisterClick}>
         Please register
```

```
);
}
export default LoginPage;
```

## **Dashboard**

## <u>App</u>

```
import React, { useState } from 'react';
import LoginPage from './LoginPage';
import RegisterPage from './RegisterPage';
import Dashboard from './Dashboard';

function App() {
  const [currentView, setCurrentView] = useState('login');
  const [currentUser, setCurrentUser] = useState(null);

  // Called on successful login: set user and show dashboard
```

```
const handleLoginSuccess = (user) => {
  setCurrentUser(user);
  setCurrentView('dashboard');
const handleRegisterSuccess = () => {
  setCurrentView('login');
      <LoginPage
         onRegisterClick={() => setCurrentView('register')}
         onLoginSuccess={handleLoginSuccess}
         onRegisterSuccess={handleRegisterSuccess}
         onLoginClick={() => setCurrentView('login')}
     {currentView === 'dashboard' && currentUser && (
       <Dashboard user={currentUser} onLogout={() => {
         setCurrentUser(null);
        setCurrentView('login');
export default App;
```