## **Objectives**

* Explain React Forms validation
* Identify the differences between React Form and HTML Form
* Explain about controlled components
* Identify various React Form input controls
* Explain how to handle React Forms
* Explain about submitting forms in React

In this hands-on lab, you will learn how to:

* Implement React forms validation
* Use various input controls like textbox, button and textarea

## **Prerequisites**

The following is required to complete this hands-on lab:

* Node.js
* NPM
* Visual Studio Code

## **Notes**

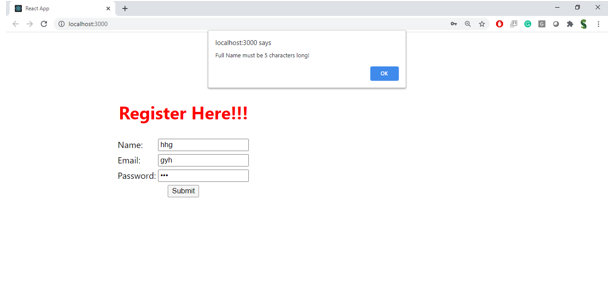
Estimated time to complete this lab: **60 minutes.**

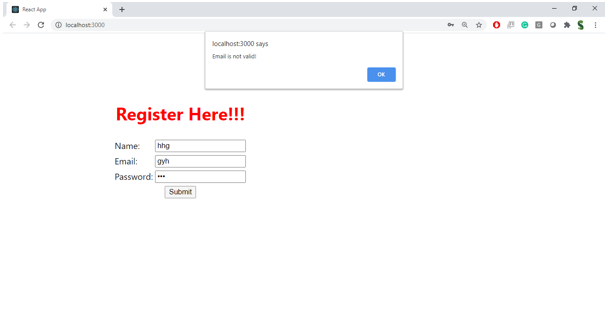
Create a React App named “mailregisterapp” which will have a component named “register.js”. Create a form which accepts the name, email and password and validate the fields as per the following:

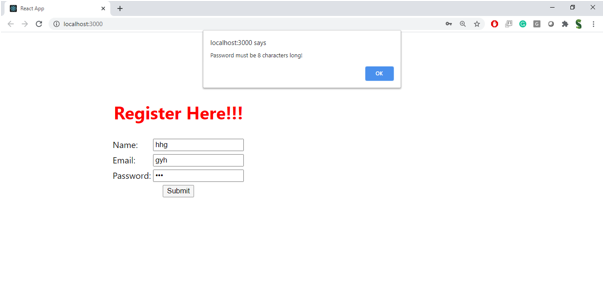
1. Name should have atleast 5 characters
2. Email should have @ and .
3. Password should have atleast 8 characters.

Ensure that validations are implemented through eventhandle and eventsubmit of a form.

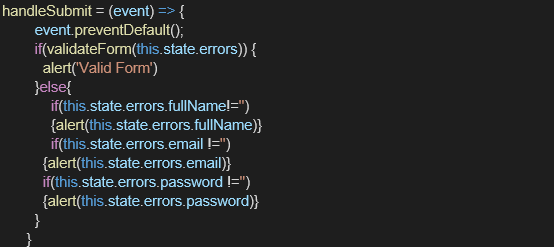
**Output Expected:**

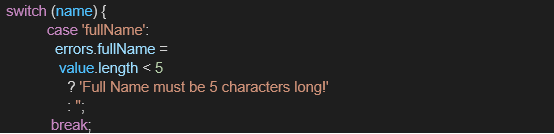


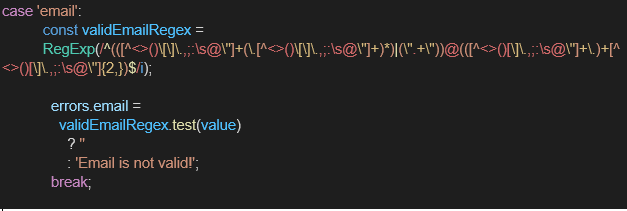




**Hint:**







**Solution :**

**Create the React App**

In the terminal:

npx create-react-app mailregisterapp

cd mailregisterapp

npm start

**Create register.js in src/**

Create a file named register.js inside the src/ folder

// src/register.js

import React, { useState } from 'react';

function Register() {

const [formData, setFormData] = useState({

name: '',

email: '',

password: ''

});

const [errors, setErrors] = useState({});

const handleChange = (e) => {

const { name, value } = e.target;

setFormData({

...formData,

[name]: value

});

// Clear error when user starts typing

setErrors({

...errors,

[name]: ''

});

};

const validate = () => {

let valid = true;

const newErrors = {};

if (formData.name.length < 5) {

newErrors.name = 'Name must be at least 5 characters';

valid = false;

}

if (!formData.email.includes('@') || !formData.email.includes('.')) {

newErrors.email = 'Invalid email address';

valid = false;

}

if (formData.password.length < 8) {

newErrors.password = 'Password must be at least 8 characters';

valid = false;

}

setErrors(newErrors);

return valid;

};

const handleSubmit = (e) => {

e.preventDefault();

if (validate()) {

alert('Registration Successful');

console.log('User Details:', formData);

// Reset form

setFormData({

name: '',

email: '',

password: ''

});

}

};

return (

<div style={{ maxWidth: '400px', margin: '40px auto' }}>

<h2>Mail Register Form</h2>

<form onSubmit={handleSubmit} noValidate>

<div style={{ marginBottom: '15px' }}>

<label>Name:</label><br />

<input

type="text"

name="name"

value={formData.name}

onChange={handleChange}

style={{ width: '100%', padding: '8px' }}

/>

<div style={{ color: 'red' }}>{errors.name}</div>

</div>

<div style={{ marginBottom: '15px' }}>

<label>Email:</label><br />

<input

type="email"

name="email"

value={formData.email}

onChange={handleChange}

style={{ width: '100%', padding: '8px' }}

/>

<div style={{ color: 'red' }}>{errors.email}</div>

</div>

<div style={{ marginBottom: '15px' }}>

<label>Password:</label><br />

<input

type="password"

name="password"

value={formData.password}

onChange={handleChange}

style={{ width: '100%', padding: '8px' }}

/>

<div style={{ color: 'red' }}>{errors.password}</div>

</div>

<button type="submit" style={{ padding: '10px 20px' }}>

Register

</button>

</form>

</div>

);

}

export default Register;

**Update App.js to use Register**

Replace the contents of src/App.js with:

import React from 'react';

import Register from './register';

function App() {

return (

<div className="App">

<Register />

</div>

);

}

export default App;