

## Hands-on 16

### 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

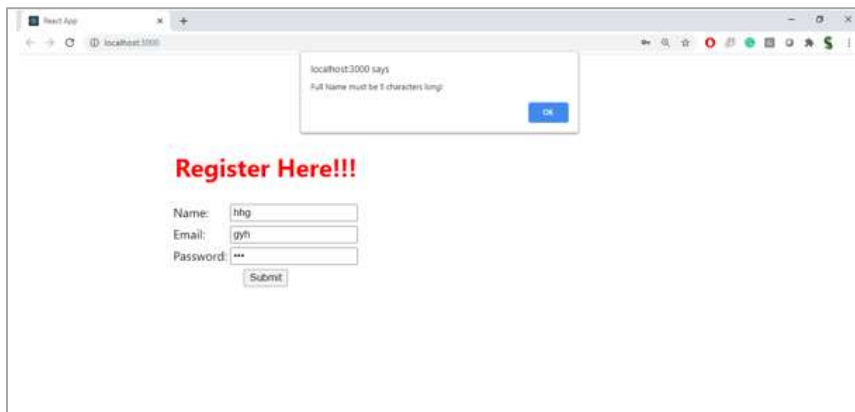
Estimated time to complete this lab: **60 minutes.**

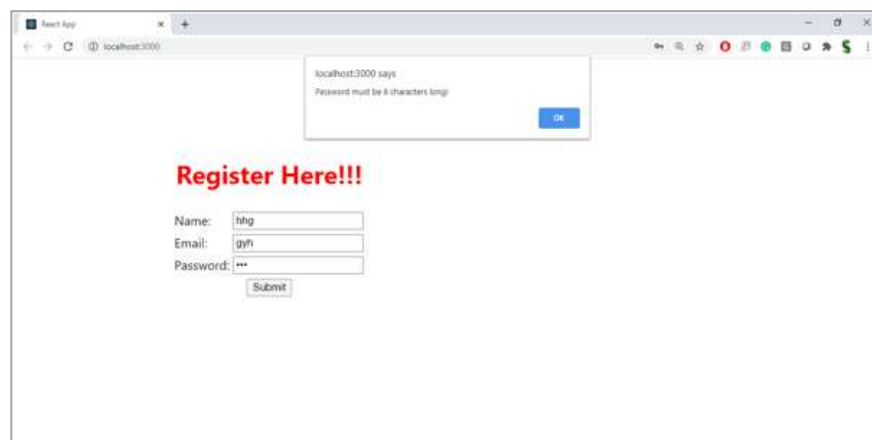
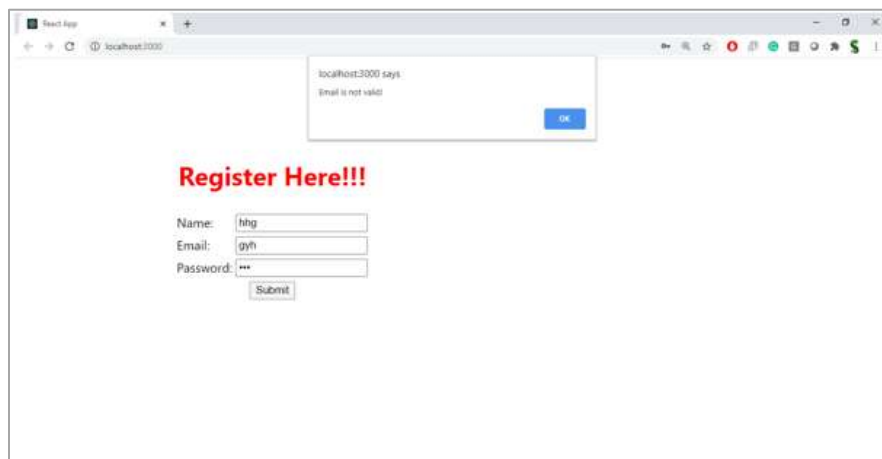
Create a React App named “mailregisterapp” which will have a component named “register.js”. Create a form which accepts the name, email and password and validate the fields as per the following:

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 events submit of a form.

### Output Expected:



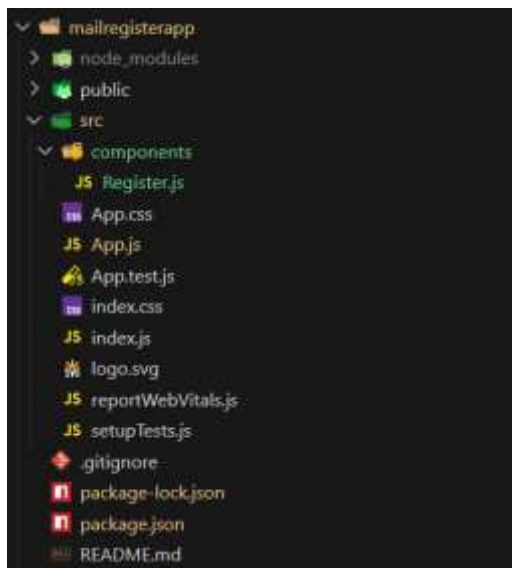


**Hint:**

```
handleSubmit = (event) => {
  event.preventDefault();
  if(validateForm(this.state.errors)) {
    alert('Valid Form')
  }else{
    if(this.state.errors.fullName!="")
    {alert(this.state.errors.fullName)}
    if(this.state.errors.email != "")
    {alert(this.state.errors.email)}
    if(this.state.errors.password != "")
    {alert(this.state.errors.password)}
  }
}
```

```
case 'email':
  const validEmailRegex =
    RegExp(/^([^\s@<>()\[\]\\.,;:\s@'"]+(\.[^<>()\[\]\\.,;:\s@'"]+)+|< >()[\[\],;:\s@'"]{2,})$/i);

  errors.email =
    validEmailRegex.test(value)
      ? ""
      : 'Email is not valid!';
  break;
```



```
import React from 'react';
import Register from './components/Register';
function App() {
  return (
    <div style={{ textAlign: 'center', marginTop: '50px' }}>
      <h1>Mail Register App</h1>
      <Register />
    </div>
  );
}
export default App;
```

## Register.js

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

```
class Register extends Component {  
  constructor(props) {  
    super(props);  
    this.state = {  
      fullName: "",  
      email: "",  
      password: "",  
      errors: {  
        fullName: "",  
        email: "",  
        password: ""  
      }  
    };  
  }  
}
```

```
validateForm = (errors) => {  
  return Object.values(errors).every((val) => val === "");  
};
```

```
handleChange = (event) => {  
  const { name, value } = event.target;  
  const errors = { ...this.state.errors };
```

```
  switch (name) {  
    case 'fullName':  
      errors.fullName = value.length < 5 ? 'Full Name must be at least 5 characters long' : "";  
      break;  
    case 'email':  
      const validEmailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;  
      errors.email = validEmailRegex.test(value) ? "" : 'Email is not valid';  
      break;  
    case 'password':  
      errors.password = value.length < 8 ? 'Password must be at least 8 characters long' : "";  
      break;  
    default:  
      break;  
  }  
  this.setState({ errors, [name]: value });  
};
```

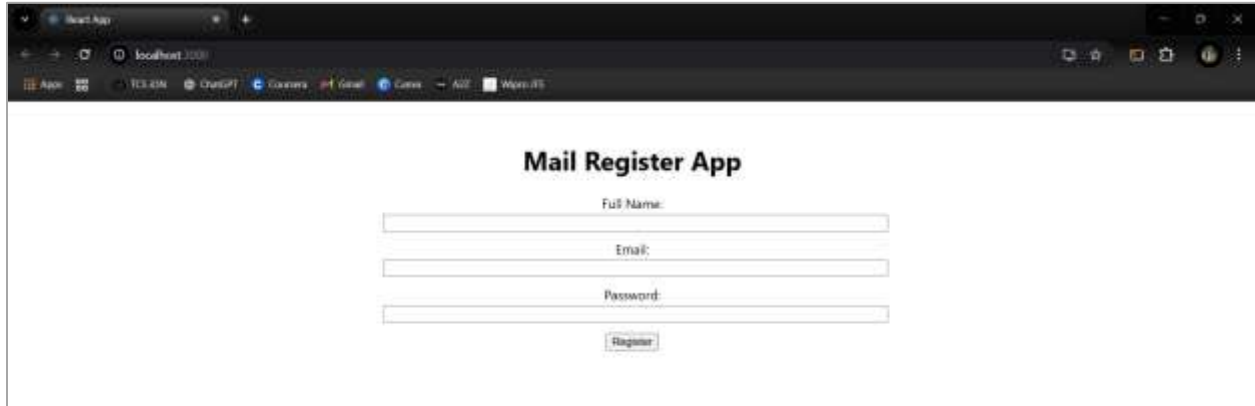
```
handleSubmit = (event) => {  
  event.preventDefault();  
  if (this.validateForm(this.state.errors)) {  
    alert('Valid Form');  
  } else {
```

```

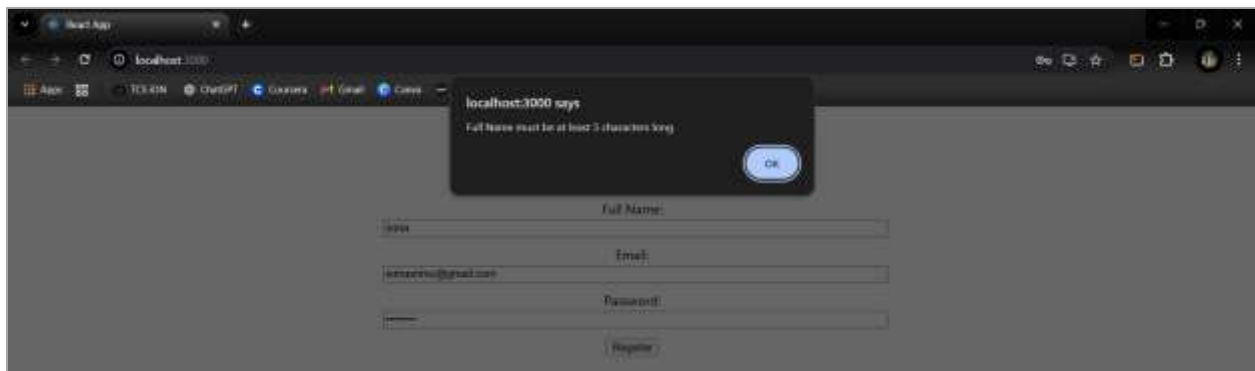
    const { fullName, email, password } = this.state.errors;
    if (fullName !== "") alert(fullName);
    if (email !== "") alert(email);
    if (password !== "") alert(password);
  }
};
render() {
  return (
    <form onSubmit={this.handleSubmit} style={{ width: '40%', margin: '0 auto' }}>
      <div style={{ marginBottom: '10px' }}>
        <label>Full Name:</label><br />
        <input
          type="text"
          name="fullName"
          value={this.state.fullName}
          onChange={this.handleChange}
          required
          style={{ width: '100%' }}
        />
      </div>
      <div style={{ marginBottom: '10px' }}>
        <label>Email:</label><br />
        <input
          type="text"
          name="email"
          value={this.state.email}
          onChange={this.handleChange}
          required
          style={{ width: '100%' }}
        />
      </div>
      <div style={{ marginBottom: '10px' }}>
        <label>Password:</label><br />
        <input
          type="password"
          name="password"
          value={this.state.password}
          onChange={this.handleChange}
          required
          style={{ width: '100%' }}
        />
      </div>
      <button type="submit">Register</button>
    </form>
  );
}
}
export default Register;

```

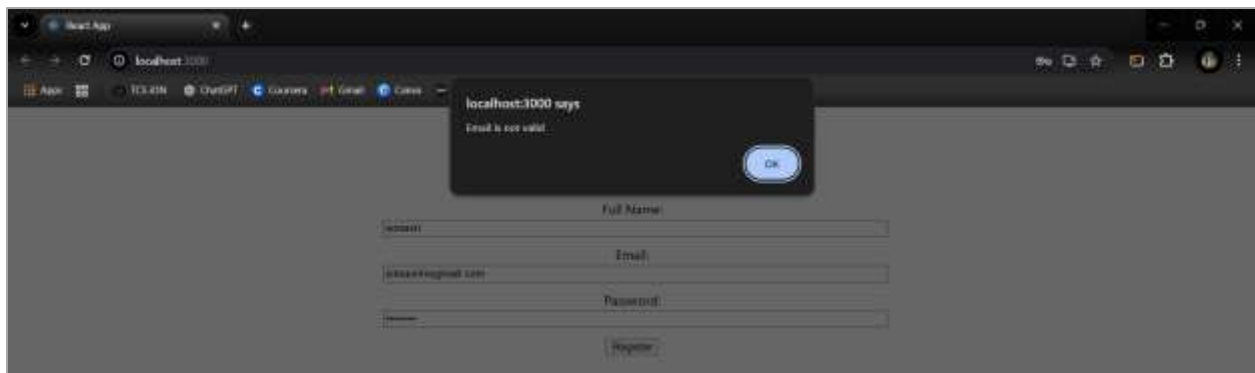
## Output:



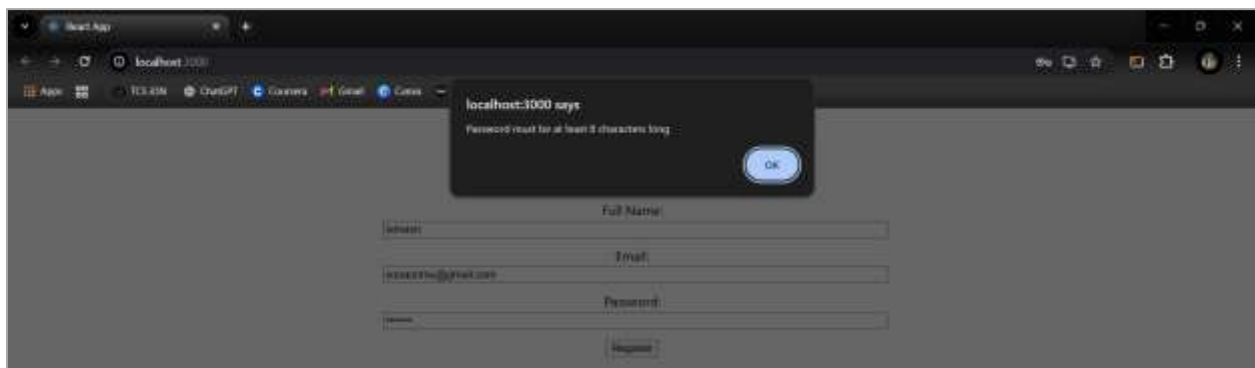
A web browser window titled "Mail App" shows a registration form for "Mail Register App". The form is centered on a white background. It contains three input fields: "Full Name:", "Email:", and "Password:". Below the "Password:" field is a "Register" button. The browser's address bar shows "localhost:3000".



The same web browser window shows the registration form with the "Full Name:" field filled with "john". A dark modal dialog box is displayed in the center, with the text "localhost:3000 says" and "Full Name must be at least 3 characters long". An "OK" button is in the bottom right corner of the dialog. The form fields are dimmed in the background.



The same web browser window shows the registration form with the "Email:" field filled with "john@gmail.com". A dark modal dialog box is displayed in the center, with the text "localhost:3000 says" and "Email is not valid". An "OK" button is in the bottom right corner of the dialog. The form fields are dimmed in the background.



The same web browser window shows the registration form with the "Password:" field filled with "john". A dark modal dialog box is displayed in the center, with the text "localhost:3000 says" and "Password must be at least 8 characters long". An "OK" button is in the bottom right corner of the dialog. The form fields are dimmed in the background.

