**class SimpleForm extends React.Component {**

**constructor(props) {**

**super(props);**

**// create a ref to store the DOM element**

**this.nameEl = React.createRef();**

**this.handleSubmit = this.handleSubmit.bind(this);**

**}**

**handleSubmit(e) {**

**e.preventDefault();**

**alert(this.nameEl.current.value);**

**}**

**render() {**

**return (**

**<form onSubmit={this.handleSubmit}>**

**<label>Name:**

**<input type="text" ref={this.nameEl} />**

**</label>**

**<input type="submit" name="Submit" />**

**</form>**

**)**

**}**

**}**

As you can see above, for a class-based component you initialize a new ref in the constructor by calling React.createRef, assigning it to an instance property so it’s available for the lifetime of the component.

In order to associate the ref with an input, it’s passed to the element as the special ref attribute. Once this is done, the input’s underlying DOM node can be accessed via this.nameEl.current.

functional component:

**function SimpleForm(props) {**

**const nameEl = React.useRef(null);**

**const handleSubmit = e => {**

**e.preventDefault();**

**alert(nameEl.current.value);**

**};**

**return (**

**<form onSubmit={handleSubmit}>**

**<label>Name:**

**<input type="text" ref={nameEl} />**

**</label>**

**<input type="submit" name="Submit" />**

**</form>**

**);**

**}**

There’s not a lot of difference here, other than swapping out createRef for the useRef hook.

### Example: login form

**function LoginForm(props) {**

**const nameEl = React.useRef(null);**

**const passwordEl = React.useRef(null);**

**const rememberMeEl = React.useRef(null);**

**const handleSubmit = e => {**

**e.preventDefault();**

**const data = {**

**username: nameEl.current.value,**

**password: passwordEl.current.value,**

**rememberMe: rememberMeEl.current.checked,**

**}**

**// Submit form details to login endpoint etc.**

**// ...**

**};**

**return (**

**<form onSubmit={handleSubmit}>**

**<input type="text" placeholder="username" ref={nameEl} />**

**<input type="password" placeholder="password" ref={passwordEl} />**

**<label>**

**<input type="checkbox" ref={rememberMeEl} />**

**Remember me**

**</label>**

**<button type="submit" className="myButton">Login</button>**

**</form>**

**);**

**}**

import React, { Component } from 'react'

export default class FormDemoComponent extends Component {

  constructor(props)

  {

      super(props);

      this.textInput = React.createRef();

  }

    render() {

        return (

            <div>

                <form>

    <h2> Not Controlled By React </h2>

                  Name:   <input type="text"/>

                    <br/>

              Password:      <input type="password" ref={this.textInput}/>

              <br/>   Address : <input type="text"/>

                </form>

            </div>

        )

    }

}

import React, { Component } from 'react'

export default class FormDemoComponent extends Component {

  constructor(props)

  {

      super(props);

      this.state=

      {

          name : '',

          address:'',

          password:''

      }

      this.txtName = React.createRef();

      this.txtAddress = React.createRef();

      this.txtPassword = React.createRef();

  }

  componentDidMount = ()=> {

      console.log(this.txtName.current.value);

      console.log(this.txtPassword.current.value);

      console.log(this.txtAddress.current.value);

      this.txtName.current.focus();

  }

  handleSubmit = event => {

      event.preventDefault();

      console.log(this.txtName.current.value);

      console.log(this.txtAddress.current.value);

      console.log(this.txtPassword.current.value);

      this.setState({name : this.txtName.current.value});

      this.setState({address : this.txtAddress.current.value});

      this.setState({password : this.txtPassword.current.value});

  }

    render() {

        return (

            <div>

                <form onSubmit={this.handleSubmit}>

    <h2> Not Controlled By React </h2>

                  Name:   <input type="text" ref={this.txtName}/>

                    <br/>

              Password:      <input type="password"  ref={this.txtPassword}/>

              <br/>   Address : <input type="text" ref={this.txtAddress}  />

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

                </form>

            </div>

        )

    }

}