

FORMS

Do what you gotta do

*“I’m so happy I get to program form validation
for our new product! This is going to be lots of
fun!”*

– NO DEVELOPER EVER

WEB 1.0 FORMS

POST /purchase HTTP /1.1

```
<form method="post" action="/purchase">  
  <!-- inputs go here -->  
  <button type="submit">Submit</button>  
</form>
```

PAGE REFRESH



WEB 1.0 FORMS

- Submitting a form sends a GET or POST request to the server
- Default behavior is to refresh the page when we get the response

FORMS WITH AJAX

AJAX FORMS

- Use JavaScript rather than the default HTML behavior
- Add event listener for the '*submit*' event that fires when a form is submitted
- Must remember to prevent the default behavior!

```
const form = document.getElementById('purchaseForm')
```



```
const form = document.getElementById('purchaseForm')
form.addEventListener('submit', (event) => {
  })
```

```
const form = document.getElementById('purchaseForm')  
  
form.addEventListener('submit', (event) => {  
  event.preventDefault()  
  
  })
```

```
const form = document.getElementById('purchaseForm')

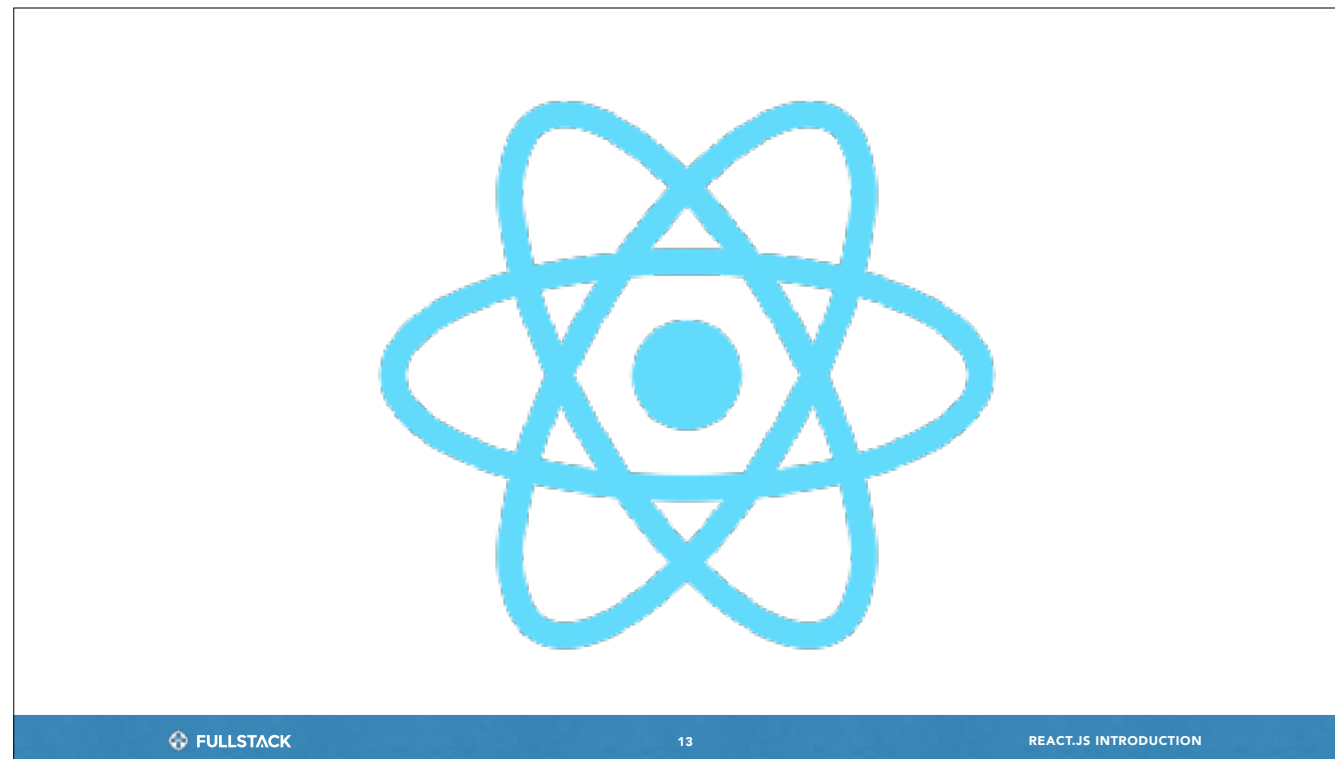
form.addEventListener('submit', (event) => {
  event.preventDefault()
  const firstName = event.target.username.value
  const lastName = event.target.lastName.value
  const address = event.target.address.value

  })
```

```
const form = document.getElementById('purchaseForm')

form.addEventListener('submit', (event) => {
  event.preventDefault()
  const firstName = event.target.username.value
  const lastName = event.target.lastName.value
  const address = event.target.address.value

  axios.post('/api/purchase', {firstName, lastName,
address})
  // etc
})
```



React can make things a little bit easier, because instead of having an imperative API like jQuery where we have to yank our data off the DOM like barbarians, we can just describe how the form looks and behaves as a function of the form's state.

```
class Form extends React.Component {
```

```
render() {  
  <form>  
    <input type='text' name='username' />  
    <button type='submit'>Submit</button>  
  </form>  
}
```

```
class Form extends React.Component {  
  constructor () {  
    super()  
    this.state = {  
      username: ''  
    }  
  }  
  
  render () {  
    <form>  
      <input type='text' name='username' />  
      <button type='submit'>Submit</button>  
    </form>  
  }  
}
```

```
class Form extends React.Component {  
  constructor () {  
    super()  
    this.state = {  
      username: ''  
    }  
  }  
  
  handleChange = (evt) => {  
    this.setState({  
      username: evt.target.value  
    })  
  }  
  
  render () {  
    <form>  
      <input type='text' name='username' onChange={this.handleChange} />  
      <button type='submit'>Submit</button>  
    </form>  
  }  
}
```



```
class Form extends React.Component {  
  constructor () {  
    super()  
    this.state = {  
      username: ''  
    }  
  }  
  
  handleChange = (evt) => {  
    this.setState({  
      username: evt.target.value  
    })  
  }  
  
  handleSubmit = (evt) => {  
    evt.preventDefault()  
    console.log('Submitting: ', JSON.stringify(this.state))  
  }  
  
  render () {  
    <form onSubmit={this.handleSubmit}>  
      <input type="text" name="username" onChange={this.handleChange} />  
      <button type="submit">Submit</button>  
    </form>  
  }  
}
```



LET'S WRITE A FORM

