

# jQuery and AJAX for Responsive Web Sites

## Lecture 1 Intro to jQuery

October 2012

22

## Objectives

- First Look at jQuery
- Interaction of HTML, CSS, JavaScript
- Loading jQuery Libraries

October 2012

23

< jQuery Hello World >

October 2012

24

# Why jQuery?

October 2012

25

## AJAX Sadness

```
// Mozilla/Safari
if (window.XMLHttpRequest) {
    self.xmlHttpRequest = new XMLHttpRequest();
}
// IE
else if (window.ActiveXObject) {
    self.xmlHttpRequest = new ActiveXObject("Microsoft.XMLHTTP");
}

self.xmlHttpRequest.open('POST', strURL, true);
self.xmlHttpRequest.setRequestHeader('Content-Type',
    'application/x-www-form-urlencoded');
self.xmlHttpRequest.onreadystatechange = function() {
    if (self.xmlHttpRequest.readyState == 4) {
        updatepage(self.xmlHttpRequest.responseText);
    }
}

self.xmlHttpRequest.send(getquerystring());
```

<http://www.degraeve.com/reference/simple-ajax-example.php>

October 2012

26

## AJAX Happiness

```
$.post( 'http://foo.com/bar.php', { 'param1': 7 }, updatePage );
```

October 2012

27

## Discussion

- Browser variation
- Feature detection vs. user-agents
- Abstractions and software engineering

October 2012

28

## Browser API Standardization

- Event Objects
  - Internet Explorer: srcElement
  - Everyone Else: target
- Similar, but with just enough difference
- Could feature detect everywhere
- Or just use jQuery to give you a standardized interface

October 2012

29

## DOM Manipulation

```
// increase text size on every other list element
$('li:odd').css( 'font-size', '50px' );

// create one hundred new list items
for ( var i = 0; i < 100; ++i ) {
  $('<li></li>').text( "Item " + i ).appendTo( '#coolList' );
}

// grab even elements from one list, append them to another
$('#listA li:even').appendTo( '#listB' );
```

<http://www.degraeve.com/reference/simple-ajax-example.php>

October 2012

30

## DOM Manipulation

```
// increase text size on every other list element
$('li:odd').css( 'font-size', '50px' );

// create one hundred new list items
for ( var i = 0; i < 100; ++i ) {
  $('<li></li>').text( "Item " + i ).appendTo( '#coolList' );
}

// grab even elements from one list, append them to another
$('#listA li:even').appendTo( '#listB' );
```

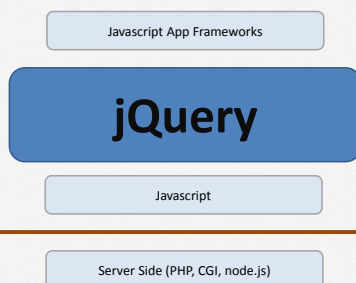
- jQuery supports the selectors even for older browsers
- Selector engine is highly optimized
- Second example demonstrates chaining

<http://www.degraeve.com/reference/simple-ajax-example.php>

October 2012

31

## jQuery in Context



October 2012

32

## Aside on JavaScript


<http://nodejs.org/>

October 2012

33

## Loading jQuery

```
<script type="text/javascript"
  src="//ajax.googleapis.com/ajax/libs/jquery/1.8.2/jquery.min.js">
</script>

<script type="text/javascript">
// Ready handler
$( function() {
  console.log( "hello, jquery!" );
} );
</script>
```

- Google jQuery CDN
- Protocol agnostic script include
- Ready handler

October 2012

34

## Aside on CDNs

- Example: Edgecast
- Users request files from the CDN, which has servers all over the world.
- The CDN originally pulls files from your servers, but after that caches them.
- Google CDN even allows for cross site caching.

.... demo dev console network resources ....

October 2012

35

## Separation of Concerns

# HTML + CSS + JavaScript

October 2012

36

# CSS Zen Garden

[www.csszengarden.com](http://www.csszengarden.com)

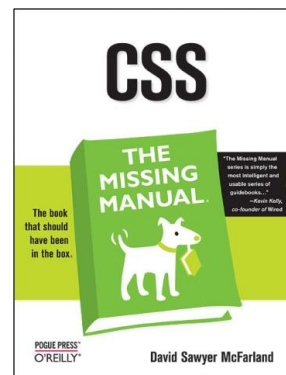
October 2012

37



October 2012

38



October 2012

39

In case you're rusty.....

## Reminder

### Tags

```
<h1></h1>          <h1 />
```

### Attributes

```
<h1 style="color:pink"></h1>
```

### Ids *(unique per dom)*

```
<h1 id="cheese"></h1>  
#cheese { color: pink; }
```

### Classes *(may repeat, can assign multiple to dom element)*

```
<h1 class="smelly cheese"></h1>  
.smelly { color: pink; }  
.cheese { font-size: 50px; }
```

October 2012

40