

# Week7- Topics

Introduction to jQuery

jQuery syntax

CSS attribute selectors review

Intro to jQuery Mobile

# jQuery

- is used to select elements on the page
- uses CSS selectors to select elements
- Uses Xpath expressions to select elements

# jQuery Download - <http://jquery.com/download/>

[Plugins](#)[Contribute](#)[Events](#)[Support](#)[jQuery Foundation](#)

## JQUERY.TO();

TORONTO, CANADA | MARCH 15-16TH, 2014

[Download](#)[API Documentation](#)[Blog](#)[Plugins](#)[Browser Support](#)

## Downloading jQuery

Compressed and uncompressed copies of jQuery files are available. The uncompressed file is best used during development or debugging; the compressed file saves bandwidth and improves performance in production. You can also download a [sourcemap file](#) for use when debugging with a compressed file. The map file is *not* required for users to run jQuery, it just improves the developer's debugger experience. As of jQuery 1.11.0/2.1.0 the `// sourceMappingURL` comment is [not included](#) in the compressed file.

To locally download these files, right-click the link and select "Save as..." from the menu.

### jQuery 1.x

The jQuery 1.x line had major changes as of jQuery 1.9.0. We *strongly* recommend that you also use the jQuery Migrate plugin if you are upgrading from pre-1.9 versions of jQuery or need to use plugins that haven't yet been updated. Read the [jQuery 1.9 Upgrade Guide](#) and the [jQuery 1.9 release blog post](#) for more information.

[Download the compressed, production jQuery 1.11.0](#)

[Download the uncompressed, development jQuery 1.11.0](#)



# The jQuery Plugin Registry

Search

## Popular Tags

- [ui](#) (430)
- [jquery](#) (297)
- [form](#) (230)
- [animation](#) (221)
- [input](#) (206)
- [image](#) (166)
- [slider](#) (130)
- [slideshow](#) (115)
- [effect](#) (113)
- [responsive](#) (113)

## ★ New Plugins

### Imapping map embedding plugin

Version 1.0.4

Released 3 hours ago

jQuery plugin which allows a user to embed a custom map to a website, containing list of desired territories and choice of drawing styles. Soon, user will be allowed to define events for mouse click on an area and multiple areas selection.

0

WATCHERS

0

FORKS

### jquery-mutation-summary

Version 0.2.2

Released 18 hours ago

A jQuery wrapper/plugin for mutation-summary,

29

4

## Recent Updates

- [jQuery Boilerplate](#)  
(version 3.3.3)
- [Imapping map embedding plugin](#)  
(version 1.0.4)
- [Image Map Resizer](#)  
(version 0.3.4)
- [jQuery Form - Fancy Web Form Designer](#)  
(version 2.3.24)
- [jQuery Dropdown Menu](#)  
(version 1.3.54)
- [Reset](#)

# jQuery Syntax

`$(selector).action()`

\$ defines statement as being jQuery

The `selector` is used to find (query) HTML elements on the page

The `.action()` is the jQuery action to be performed on the elements

# jQuery \$

`$.ajax()` = is equal to `jquery.ajax()`

uses jQuery to make an Ajax request

# jQuery is:

- a form of JavaScript
- a JavaScript library
- not a W3C standard
- written in JavaScript scripting language

# jQuery is:

to use jQuery you must reference the jQuery library

```
<script type="text/javascript" src="js/jquery-1.6.4.js"></script>
```

- download jQuery library
- put it in js folder
- reference through the `<script>` element



# jQuery Library Link

The browser does not know where jQuery is located

- linking to jQuery library on jQuery.com:

```
<script type="text/javascript" src="http://code.jquery.com/  
jquery-1.6.4.min.js"></script>
```

```
<script type="text/javascript" src="http://code.jquery.com/mobile/  
1.0/jquery.mobile-1.0.min.js"></script></head>
```

- using the `<script>` element to reference a jQuery library held at Google:

```
<script type="text/javascript" src="http://ajax.googleapis.com/ajax/  
libs/jquery/1.7.1/jquery.min.js"></script>
```

# jQuery Link in HTML Document

```
<?xml version="1.0" encoding="UTF-8" ?>
<!doctype html>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<meta charset="utf-8">
<title>Intro to jQuery</title>
</head>
<body>
<a href="http://jquery.com/">jQuery</a>
<!--Link to jQuery script-->
<script src="js/jquery.js"></script>
<script>
</script>
</body>
</html>
```

# jQuery

calls could look like:

```
jQuery("div")
```

uses the \$ dollar sign as a shortcut so that jQuery calls look like:

```
$("#div")
```

# jQuery – Client Scripting Language

Client scripting takes place after the page has been sent to the browser

JavaScript

jQuery

Server scripting takes place before the page is sent to the browser

PHP

# jQuery

can be mixed with other frameworks and scripting languages:

JavaScript

Ajax

JSON

# html() in jQuery

html() is used to set the inner HTML, the value, of HTML elements:

```
$("#div").html("types this inside the div");
```

```
$("#td").html("types this into the table cell");
```

# html() in jQuery

used to retrieve the contents of an element

works with HTML documents, but not work with XML docs

```
$("#div").html()
```

returns the value of the first div

```
$("#td").html()
```

returns the value of the first td

# jQuery Controls CSS Styles

use the `css()` method to set CSS styles using jQuery:

```
$("#div").css("background-color", "orange");
```



# jQuery Controls CSS Styles

to toggle class (e.g., on every click) alternating adding the class, then removing it, adding it, ...etc.

```
$("#div#doToggle").click(function(){  
  $("#p").toggleClass("classSuperBold"); });
```

clicking on the div with id="doToggle" will alternately add and remove the classSuperBold class from all paragraphs

# jQuery Controls CSS Styles

Sets width and height

```
$("#div").height(65);
```

sets the height of all div elements to 65 pixels

```
$("#div").width(480);
```

sets the width of all div elements to 480 pixels

# jQuery hide()

hide() - hides an element

arguments:

slow

fast

without indicating slow or fast argument  
the element will be hidden immediately

# jQuery Syntax

```
$("#a").addClass("orange");
```

adds a CSS class named 'orange' to all <a> elements

```
$("#a").removeClass("orange");
```

removes the CSS class named 'orange' from all <a> elements

```
$("#p").hide();
```

hides all <p> tags on the page

```
$("#p").hide("slow");
```

hides all <p> tags 'slowly' on the page

# jQuery Syntax

`$(this)`

refers to the current HTML element

`$("p")`

p - any HTML element

refers to the paragraph element

`$("a.yellow")`

refers to the `<a>` element with `class="yellow"`

# jQuery Syntax

`$("#yellow")`

Refers to the element with `id="yellow"`

`$("#div#yellow")`

Refers to the `<div>` element with `id="yellow"`

# CSS Attribute Selectors

# CSS Attribute Selectors

allow to create rules that apply to elements that have an attribute with a specific value

SELECTOR	MEANING	EXAMPLE
EXISTENCE	<code>[]</code> Matches a specific attribute (whatever its value)	<code>p[class]</code> Targets any <code>&lt;p&gt;</code> element with an attribute called <code>class</code>
EQUALITY	<code>[=]</code> Matches a specific attribute with a specific value	<code>p[class="dog"]</code> Targets any <code>&lt;p&gt;</code> element with an attribute called <code>class</code> whose value is <code>dog</code>
SPACE	<code>[~=]</code> Matches a specific attribute whose value appears in a space-separated list of words	<code>p[class~="dog"]</code> Targets any <code>&lt;p&gt;</code> element with an attribute called <code>class</code> whose value is a list of space-separated words, one of which is <code>dog</code>



# CSS Attribute Selectors

## PREFIX

[^=]

Matches a specific attribute whose value begins with a specific string

p[attr^="d"]

Targets any <p> element with an attribute whose value begins with the letter "d"

## SUBSTRING

[\*=]

Matches a specific attribute whose value contains a specific substring

p[attr\*"do"]

Targets any <p> element with an attribute whose value contains the letters "do"

## SUFFIX

[\$=]

Matches a specific attribute whose value ends with a specific string

p[attr\$"g"]

Targets any <p> element with an attribute whose value ends with the letter "g"

# CSS Attribute Selectors

Simple attribute selector

selects all elements with the matching value

```
p[class] {color: blue}
```

# CSS Attribute Selectors

## Advanced attribute selector

selects elements with matching substrings in the value of the attribute. ( existing attributes or created custom attributes)

caret (^) - begins with the value "iphone":

```
p[device^="iphone"] {background: green;}
```

dollar sign or string (\$) - ends with the value "iphone":

```
p[device$="iphone"] {background: blue;}
```

star (\*) - contains at least one instance of "lcd":

```
p[device*="lcd"] {background: red;}
```

# jQuery Attribute Selectors

# jQuery Attribute Selectors using [ Xpath ]

```
$("#[style]")
```

refers to all elements with a 'style' attribute

```
$("#[style='text-decoration:underline;']")
```

refers to all elements with style="text-decoration: underline;"

```
$("#[style!='text-decoration:underline;']")
```

refers to all elements with a 'style' attribute NOT equal to style='text-decoration:underline;'

! Exclamation means NOT

# jQuery Attribute Selectors using Xpath

```
$("#[style*='text-decoration:underline;']")
```

refers to all elements with a 'style' attribute that contain text-decoration:underline; within it

\* asterisk is an important syntax part

# jQuery Attribute Selectors using Xpath

```
$("#[src$='.jpg']")
```

refers to all elements with a 'src' attribute that ends with .jpg  
\$ dollar sign specifies suffix

```
$("#[name^='address']")
```

Refers to all elements with a 'name' attribute (think form elements) that *starts with* exactly address

^ - starts exactly with

# jQuery Attribute Selectors using Xpath example

```
<input name="login1" />  
<input name="login2" />  
<input name="prolog" />  
<div name="logistics" />
```

Statement:

```
$("#input[name*='log']").val("has log in it!");
```

Output:

All 3 input elements will have the value set to: "has log in it!"



# CSS Selectors

```
$("div").css("background-color", "#0000ff");
```

sets the bg color of all <div> elements in the document to blue  
to specify an individual <div> we can use #id

# CSS Selectors

```
<head>
```

```
  <style type="text/css">
```

```
    a.mad { font-weight:bold; }
```

```
  </style>
```

```
</head>
```

```
$("a").addClass("mad");
```

adds the CSS class 'mad' to all <a> elements

```
$("a").removeClass("mad");
```

removes the CSS class 'mad' from all <a> elements

# jQuery ready()

remember () happens when the browser first loads  
It replaces the older JavaScript call window.onload()

`$(document).ready()`

is used to prevent code from running until the pages has finished loading

# Ready Event

```
$(document).ready(function(){  
  //add code here  
});
```

This code will execute as soon as the HTML document is loaded in the browser window and ready (for work)

Ready event replaces window.onload JavaScript

# Ready Event

```
$(document).ready(function(){  
    $("li").addClass("mad");  
    $("div#wrapper").css("background-color", "#0000ff");  
});
```

When the page is finished loading in the browser;

- all <li> elements will have the CSS class 'mad' applied to them
- the <div> with id="wrapper" element will have a blue bg color

# Event Functions

```
$("#div#dragon").click(function(event){  
    event.preventDefault();  
    $(this).hide("slow"); });
```

- when the <div> with id="dragon" is clicked, jq prevents any default functionality and hides the <div>
- 'event' is the variable name we used as a parameter:  
function(event) or function(e), and after: e.preventDefault();
- hide() - built-in effect provided by jQuery
- This has to be launched from the ready event

# Event Functions

```
<script type="text/javascript">
    $(document).ready(function(){
        $("#div#wrapper").click(function(e){
            e.preventDefault();
            $(this).hide("slow");
        });    //end of click event
    });    //end of ready event
</script>
```

# Callback

Callback – a function that is passed as an argument to another function and is executed after its parent function has completed

```
$.get("myPage.html", callbackFunction);
```

First parameter is the URL to get

Second parameter is the name of the function to call after getting the URL



# Callback with Parameters

Typical way the two parameters (arguments) are passed into the function:

```
par1 = "cat";
```

```
par2 = "dog";
```

```
someFunction(par1, par2);
```

In jQuery: passing parameters into the function you want to call:

```
$.get("myPage.html", function(){  
    callbackFunction(par1, par2);  
});
```

# .hide() Effect

## .hide()

```
$(this).hide("slow");
```

hides the current element

```
$("#footer").hide();
```

hides the element with id="footer"

```
$("a").hide("fast");
```

hides all the <a> elements

# .animate() Effect

Ex: jQuery\_animate.html  
image id="pic"

Clicking the div will shrink the image over 3 seconds to disappear

Clicking the div again will expand the image over 3 seconds to full size

Opacity and position will remain adjusted, however

# .animate() Effect

Image id="img1"

Clicking the div will shrink the image over  
3 seconds until it disappears

Clicking the div again will expand the image over 3 seconds until it is  
full size again

# .animate() Effect

Ex. jQuery\_animate\_opacity.html

```
$(document).ready(function(){  
    $('#pic2').animate({'opacity' : 1}).hover(function() {  
        //on hover, change opacity to 50%  
        $(this).animate({'opacity' : .5});  
    }, function() {  
        //on exit hover, change opacity back to 100%  
        $(this).animate({'opacity' : 1});  
    });  
});
```

# jQuery \$

`$.ajax()` = is equal to `jquery.ajax()`

uses jQuery to make an Ajax request

# jQuery Mobile

Guest Lecture: Mikael Blomberg, SC5

Examples jQM

Tutorial on vimeo (1h 06 min)

[http://player.vimeo.com/video/53223683?  
title=0&amp;amp;byline=0&amp;amp;portrait=0](http://player.vimeo.com/video/53223683?title=0&amp;amp;byline=0&amp;amp;portrait=0)