



Introduction for Developers

Who am I?

- Jonathan Sharp, Omaha, Nebraska
- Freelance Developer (Out West Media.com)
- jQuery Team member, (web & infrastructure team)
- Co-authored “jQuery Cookbook” (O’Reilly) Q4 2009
- @jdsharp

Who am I?

- Own a horse



Overview

- Who, what, and why of jQuery
- 5 core jQuery concepts
- Overview of jQuery API
- Build a plugin in 6 steps
- jQuery Initiatives
- Questions

Who uses jQuery

- 35% of all sites that use JavaScript, use jQuery
- 1 out 5, of all sites, use jQuery

Who uses jQuery

- 35% of all sites that use JavaScript, use jQuery
- 1 out 5, of all sites, use jQuery

reddit.com

espn.com

ibm.com

stackoverflow.com

kickball.com

boxee.tv

bit.ly

twitpic.com

whitehouse.gov

wikipedia.org

microsoft.com

amazon.com

netflix.com

bing.com

monster.com

tv.com

overstock.com

time.com

capitalone.com

usatoday.com

ning.com

wordpress.com

dell.com

twitter.com

Who uses jQuery

- 35% of all sites that use JavaScript, use jQuery
- 1 out 5, of all sites, use jQuery

reddit.com

espn.com

ibm.com

stackoverflow.com

kickball.com

boxee.tv

bit.ly

twitpic.com

whitehouse.gov

wikipedia.org

microsoft.com

amazon.com

netflix.com

bing.com

monster.com

tv.com

overstock.com

time.com

capitalone.com

usatoday.com

ning.com

wordpress.com

dell.com

twitter.com

What exactly is jQuery

jQuery is a JavaScript Library!

- Interaction with the DOM
(e.g. selecting, creating, traversing, changing etc...)
- JavaScript Events
- Animations
- Ajax interactions

What does that mean?

Add class 'odd' to a table row...

```
var tables = document.getElementsByTagName( 'table' );

for (var t = 0; t < tables.length; t++) {
    var rows = tables[t].getElementsByTagName( "tr" );
    for (var i = 1; i < rows.length; i += 2) {
        if (!(/^|s)odd(s|$)/.test(rows[i].className)) {
            rows[i].className += 'odd';
        }
    }
};
```

...becomes...

<http://jsbin.com/esewu/edit>

Poetry

```
jQuery( 'table tr:nth-child(odd)' ).addClass( 'odd' );
```


Using jQuery we can do this

```
jQuery( 'table tr:nth-child(odd)' ).addClass( 'odd' );
```



jQuery function

Using jQuery we can do this

```
jQuery( 'table tr:nth-child(odd)' ).addClass( 'odd' );
```



jQuery function



jQuery Selector (CSS Expression)

Using jQuery we can do this

```
jQuery( 'table tr:nth-child(odd)' ).addClass( 'odd' );
```



jQuery function



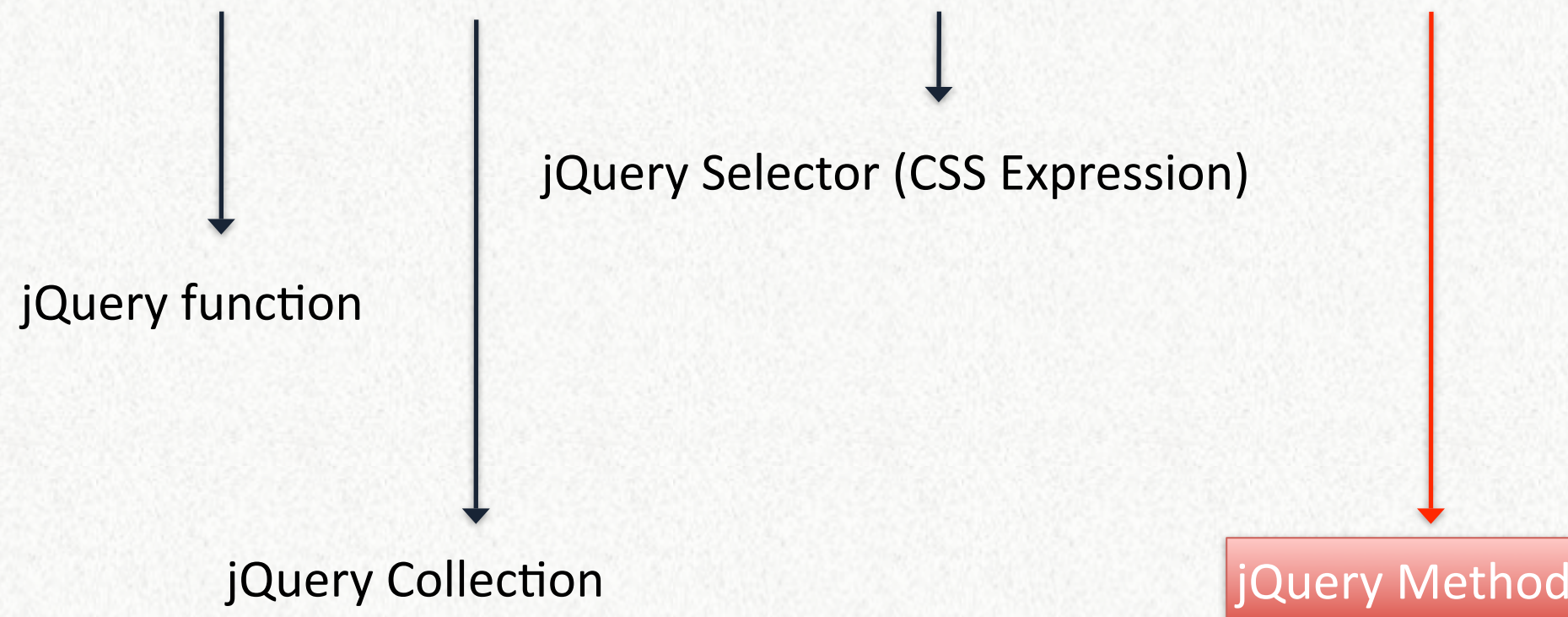
jQuery Selector (CSS Expression)



jQuery Collection

Using jQuery we can do this

```
jQuery( 'table tr:nth-child(odd)' ).addClass( 'odd' );
```



Using jQuery we can do this

```
jQuery( 'table tr:nth-child(odd)' ).addClass( 'odd' );
```

Further Reduced

```
$('table tr:nth-child(odd)').addClass('odd');
```


Further Reduced

```
var jQuery = function(...) { ... };  
  
// $ is a valid variable character in JavaScript  
var $ = jQuery;  
  
$('table tr:nth-child(odd)').addClass('odd');
```

It really is the

“write less, do more”

JavaScript Library!

Why use jQuery

- Simplicity, Speeds up web development
- Avoids common headaches w/ browsers
- Extensive list of plugins
- Large & active community
- Extensive test coverage (50 browsers, 11 platforms)
- API for both coders and designers

Why use jQuery over other solutions

- ~~Simplicity, Speeds up web development~~
- ~~Avoids common headaches w/ browsers~~
- Extensive list of plugins
- Large & active community
- Extensive test coverage (50 browsers, 11 platforms)
- API for both coders and designers

Ok, lets get to it!

Concept 1. Find something, do something

```
<!DOCTYPE html>
<html>
<body>
  <ul>
    <li><a>home</a></li>
    <li><a>about</a></li>
  </ul>
</body>
</html>
```


Concept 1. Find something, do something

```
<!DOCTYPE html>
<html>
<body>
  <ul>
    <li><a>home</a></li>
    <li><a>about</a></li>
  </ul>
  <script src='jquery.js'></script>
  <script>
    $('ul');
  </script>
</body>
</html>
```

Concept 1. Find something, do something

```
<!DOCTYPE html>
<html>
<body>
  <ul id='nav'>
    <li><a>home</a></li>
    <li><a>about</a></li>
  </ul>
<script src='jquery.js'></script>
<script>
  $('ul').attr('id', 'nav');
</script>
</body>
</html>
```

Concept 1. Find something, do something

```
<!DOCTYPE html>
<html>
<body>
  <ul id='nav'>
    <li><a>home</a></li>
    <li><a>about</a></li>
  </ul>
  <script src='jquery.js'></script>
  <script>
    $('#nav li');
  </script>
</body>
</html>
```


Concept 1. Find something, do something

```
<!DOCTYPE html>
<html>
<body>
  <ul id='nav'>
    <li class='navListItem'><a>home</a></li>
    <li class='navListItem'><a>about</a></li>
  </ul>
<script src='jquery.js'></script>
<script>
  $('#nav li').addClass('navListItem');
</script>
</body>
</html>
```

Concept 1. Find something, do something

```
<!DOCTYPE html>
<html>
<body>
    <ul id='nav'>
        <li class='navListItem'><a>home</a></li>
        <li class='navListItem'><a>about</a></li>
    </ul>
<script src='jquery.js'></script>
<script>
    $( '#nav a' );
</script>
</body>
</html>
```

Concept 1. Find something, do something

```
<!DOCTYPE html>
<html>
<body>
  <ul id='nav'>
    <li class='navListItem'><a href='/home'>home</a></li>
    <li class='navListItem'><a href='/about'>about</a></li>
  </ul>
<script src='jquery.js'></script>
<script>
  $( '#nav a' ).each(function() {
    $(this).attr('href', '/' + $(this).text());
  });
</script>
</body>
</html>
```


Concept 2. Create something, do something

```
<!DOCTYPE html>  
<html>  
<body>  
<ul id='nav'>  
</ul>  
</script>  
</body>  
</html>
```

Concept 2. Create something, do something

```
<!DOCTYPE html>
<html>
<body>
<ul id='nav'>
</ul>
<script src='jquery.js'></script>
<script>
    $( '<li>home</li>' );
</script>
</body>
</html>
```

Concept 2. Create something, do something

```
<!DOCTYPE html>
<html>
<body>
<ul id='nav'>
</ul>
<script src='jquery.js'></script>
<script>
    $( '<li>home</li>' ).wrapInner( 'a' );
</script>
</body>
</html>
```


Concept 2. Create something, do something

```
<!DOCTYPE html>
<html>
<body>
<ul id='nav'>
    <li><a>home</a></li>
    <li><a>about</a></li>
</ul>
<script src='jquery.js'></script>
<script>
    $( '<li>home</li>' ).wrapInner( 'a' ).appendTo( '#nav' );
    $( '<li>about</li>' ).wrapInner( 'a' ).appendTo( '#nav' );
</script>
</body>
</html>
```

Concept 3. Chaining & Operating

```
<!DOCTYPE html>
<html>
<body>
  <ul id='nav'>
    <li class='navListItem'><a href='/home'>home</a></li>
    <li class='navListItem'><a href='/about'>about</a></li>
  </ul>
<script src='jquery.js'></script>
<script>
  $('ul').attr('id','nav');
  $('#nav li').addClass('navListItem');
  $('#nav a').each(function(){
    $(this).attr('href','/' + $(this).text());
  });
</script>
</body>
</html>
```

Concept 3. Chaining & Operating

```
<!DOCTYPE html>
<html>
<body>
  <ul id='nav'>
    <li class='navListItem'><a href='/home'>home</a></li>
    <li class='navListItem'><a href='/about'>about</a></li>
  </ul>
<script src='jquery.js'></script>
<script>
  $( 'ul' )
    .attr( 'id', 'nav' )
    .find( 'li' ) // Push stack
    .addClass( 'navListItem' )
    .find( 'a' ) // Push stack
    .each( function() {
      $(this).attr( 'href', '/' + $(this).text() );
    } );
</script>
</body>
</html>
```


Concept 4. Understanding Implicit iteration

```
<!DOCTYPE html>
<html>
<body>
  <ul id='nav'>
    <li class='navListItem'><a href='/home'>home</a></li>
    <li class='navListItem'><a href='/about'>about</a></li>
  </ul>
<script src='jquery.js'></script>
<script>
  $( 'ul' )
    .attr( 'id', 'nav' )
    .find( 'li' )
    .addClass( 'navListItem' )
    .find( 'a' )
    .each( function() {
      $(this).attr( 'href', '/' + $(this).text() );
    } );
</script>
</body>
</html>
```

Concept 4. Understanding Implicit iteration

```
<!DOCTYPE html>
<html>
<body>
  <ul id='nav'>
    <li class='navListItem'><a href='/home'>home</a></li>
    <li class='navListItem'><a href='/about'>about</a></li>
  </ul>
<script src='jquery.js'></script>
<script>
  $( 'ul' )
    .attr( 'id', 'nav' )
    .find( 'li' )
    .addClass( 'navListItem' )
    .find( 'a' )
    .each( function() {
      $(this).attr( 'href', '/' + $(this).text() );
    } );
</script>
</body>
</html>
```

Concept 5. Knowing the jQuery parameter types

- CSS Selectors & custom CSS expressions

e.g. `$('#nav')` and `(':first')`

- HTML

e.g. `$('link')`

- DOM Elements

e.g. `$(document)` or `$(document.createElement('a'))`

- A function (shortcut for jQuery DOM ready event)

e.g. `$(function(){}) = $(document).ready(function(){})`

Review


- Find something, do something
- Create something, do something
- Chaining & Operating
- Understanding Implicit Iteration
- Knowing the jQuery parameter types

jQuery API overview

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

Overview of jQuery API


- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities



```
$()  
  
each()  
size()  
length()  
selector()  
context()  
eq()  
get()  
index()  
  
data()  
removeData()  
queue()  
dequeue()  
  
jQuery.fn.extend()  
jQuery.extend()  
  
jQuery.noConflict()
```


Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities



```
$()  
  
each()  
size()  
length()  
selector()  
context()  
eq()  
get()  
index()  
  
data()  
removeData()  
queue()  
dequeue()  
  
jQuery.fn.extend()  
jQuery.extend()  
  
jQuery.noConflict()
```

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
<!DOCTYPE html>
<html>
<body>

<p>Element Node</p>

<script src="http://ajax.googleapis.com/ajax/libs/
jquery/1.3.2/jquery.min.js" ></script>
<script>
    alert($('p').get(0).nodeType);
</script>

</body>
</html>
```

<http://jsbin.com/aneki/edit#html>

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
<!DOCTYPE html>
<html>
<body>

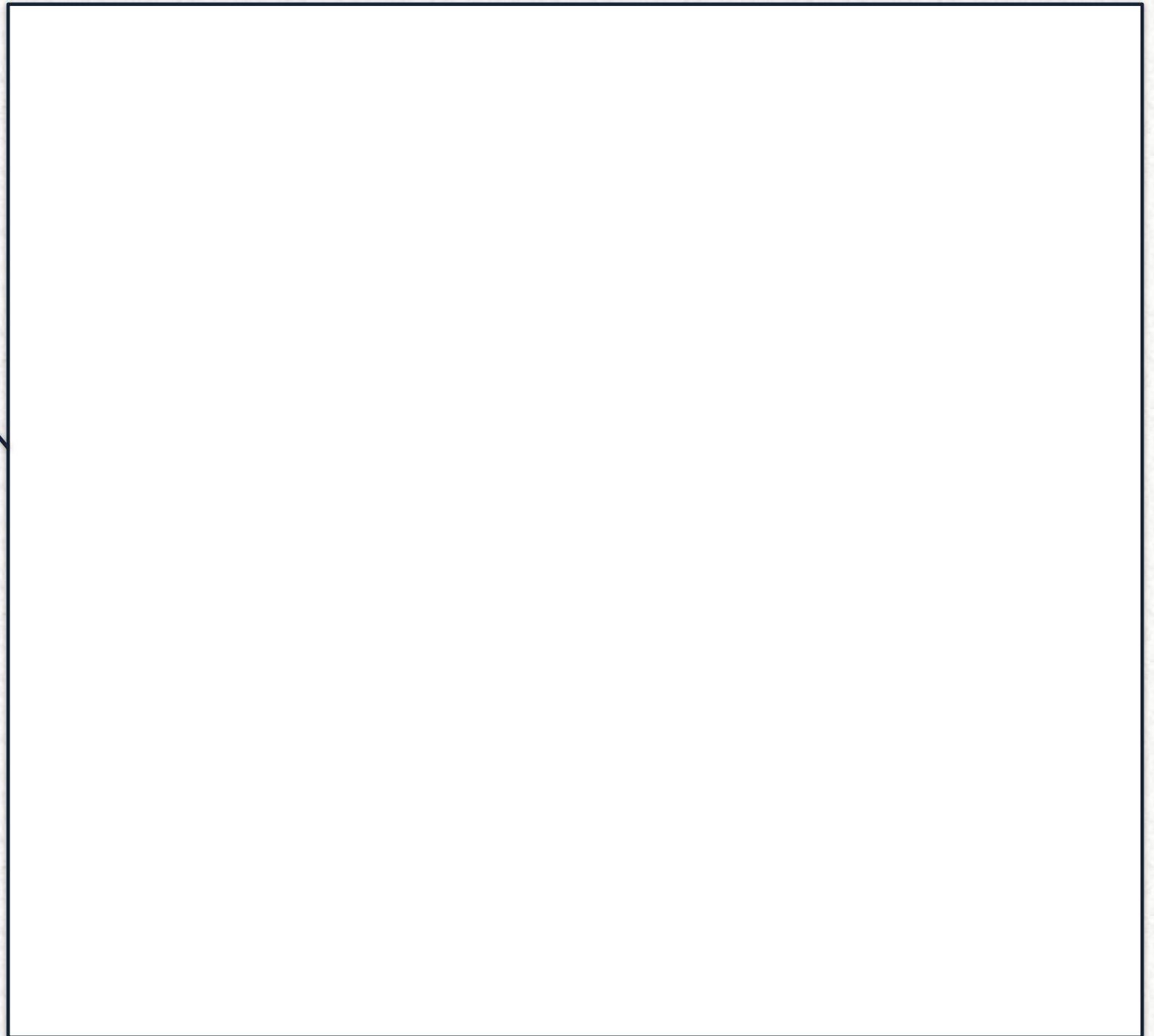
<p>Element Node</p>

<script src="http://ajax.googleapis.com/ajax/libs/
jquery/1.3.2/jquery.min.js" ></script>
<script>
    alert($('p').get(0).nodeType);
    alert($('p')[0].nodeType);
</script>

</body>
</html>
```


Overview of jQuery API

- Core
- **Selectors**
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities



Overview of jQuery API

- Core
- **Selectors**
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities



```
$('#nav li.contact')
```

Overview of jQuery API

- Core
- **Selectors**
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
$('#nav li.contact')
```

```
$( ':visible' )
```


Overview of jQuery API

- Core
- **Selectors**
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
$( '#nav li.contact' )  
  
$( ':visible' )  
  
$( ':radio:enabled:checked' )
```

Overview of jQuery API

- Core
- **Selectors**
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
$( '#nav li.contact' )  
  
$( ':visible' )  
  
$( ':radio:enabled:checked' )  
  
$( 'a[title]' )
```

Overview of jQuery API

- Core
- **Selectors**
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
$( '#nav li.contact' )  
  
$( ':visible' )  
  
$( ':radio:enabled:checked' )  
  
$( 'a[title]' )  
  
$( 'a[title][href*='foo']' )
```


Overview of jQuery API

- Core
- **Selectors**
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
$( '#nav li.contact' )  
  
$( ':visible' )  
  
$( ':radio:enabled:checked' )  
  
$( 'a[title]' )  
  
$( 'a[title][href*='foo']' )  
  
$( 'a:first[href*='foo']' )
```

Overview of jQuery API

- Core
- **Selectors**
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
$('#nav li.contact')  
  
$(':visible')  
  
$(':radio:enabled:checked')  
  
$('a[title]')  
  
$('a[title][href*='foo']')  
  
$('a:first[href*='foo']')  
  
$('#header, #footer')
```

Overview of jQuery API

- Core
- **Selectors**
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
$( '#nav li.contact' )  
  
$( ':visible' )  
  
$( ':radio:enabled:checked' )  
  
$( 'a[title]' )  
  
$( 'a[title][href*='foo']' )  
  
$( 'a:first[href*='foo']' )  
  
$( '#header,#footer' )  
  
$( '#mainContent,#sidebar','#content' )
```


Overview of jQuery API

- Core
- **Selectors**
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
$('#nav li.contact')
```

```
$(':visible')
```

```
$(':radio:enabled:checked')
```

```
$('a[title]')
```

```
$('a[title][href*='foo']')
```

```
$('a:first[href*='foo']')
```


```
$('#header,#footer')
```

```
$('#mainContent,#sidebar','#content')
```

<http://codylindley.com/jqueryselectors/>

Overview of jQuery API


- Core
- Selectors
- **Attributes**
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities



```
attr()  
removeAttr()  
  
addClass()  
hasClass()  
toggleClass()  
removeClass()  
  
val()
```

Overview of jQuery API

- Core
- Selectors
- **Attributes**
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities



```
attr()  
removeAttr()  
  
addClass()  
hasClass()  
toggleClass()  
removeClass()  
  
val()
```


Overview of jQuery API

- Core
- Selectors
- **Attributes**
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
<!DOCTYPE html><html><body>


<input type="text" value="search" />

<script src="jquery.js"></script>
<script>
$('input')
  .focus(function(){
    var v = $(this).val();
    $(this).val( v === this.defaultValue ? '' : v);
  })
  .blur(function(){
    var v = $(this).val();
    $(this).val( v.match(/^\s+$/)?
      this.defaultValue : v);
  });
</script></body></html>
```

<http://jsbin.com/irico/edit#html>

Overview of jQuery API

- Core
- Selectors
- Attributes
- **Traversing**
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities




```
add()  
children()  
closest()  
contents()  
find()  
next()  
nextAll()  
offsetParent()  
parent()  
parents()  
prev()  
prevAll()  
siblings()  
  
andSelf()  
end()
```

```
eq()  
filter()  
is()  
map()  
not()  
slice()
```

Overview of jQuery API

- Core
- Selectors
- Attributes
- **Traversing**
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities



```
add()  
children()  
closest()  
contents()  
find()  
next()  
nextAll()  
offsetParent()  
parent()  
parents()  
prev()  
prevAll()  
siblings()  
  
andSelf()  
end()  
  
eq()  
filter()  
is()  
map()  
not()  
slice()
```


Overview of jQuery API

- Core
- Selectors
- Attributes
- **Traversing**
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
<!DOCTYPE html>
<html>
<body>

<p>Make me bold!</p>

<script src="http://ajax.googleapis.com/ajax/
libs/jquery/1.3.2/jquery.min.js" ></script>
<script>

$( 'p' )
    .contents( )
    .wrap( '<strong></strong>' );

</script>
</body>
</html>
```

<http://jsbin.com/ituza/edit#html>

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- **Manipulation**
- CSS
- Events
- Effects
- Ajax
- Utilities

`html()`
`text()`

`append()`
`appendTo()`
`prepend()`
`prependTo()`

`after()`
`before()`
`insert()`
`insertAfter()`
`insertBefore()`

`wrap()`
`wrapAll()`
`wrapInner()`

`replaceWith()`
`replaceAll()`

`empty()`
`remove()`

`clone()`

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- **Manipulation**
- CSS
- Events
- Effects
- Ajax
- Utilities

`html()`
`text()`

`append()`
`appendTo()`
`prepend()`
`prependTo()`

`after()`
`before()`
`insert()`
`insertAfter()`
`insertBefore()`

`wrap()`
`wrapAll()`
`wrapInner()`

`replaceWith()`
`replaceAll()`

`empty()`
`remove()`

`clone()`

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- **Manipulation**
- CSS
- Events
- Effects
- Ajax
- Utilities

```
<!DOCTYPE html>
<html>
<body>

<p>jQuery</p>

<script src="jquery.js"></script>
<script>

alert($('p').text());

</script>
</body>
</html>
```

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- **CSS**
- Events
- Effects
- Ajax
- Utilities

`css()`

`offset()`

`offsetParent()`

`position()`

`scrollTop()`

`scrollLeft()`

`height()`

`width()`

`innerHeight()`

`innerWidth()`

`outerHeight()`

`outerWidth()`

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- **CSS**
- Events
- Effects
- Ajax
- Utilities

`css()`

`offset()`

`offsetParent()`

`position()`

`scrollTop()`

`scrollLeft()`

`height()`

`width()`

`innerHeight()`

`innerWidth()`

`outerHeight()`

`outerWidth()`

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- **CSS**
- Events
- Effects
- Ajax
- Utilities

```
<!DOCTYPE html>
<html>
<head>
<style>div{background-color:#ccc; width:100px;
margin:0 20px; float:left;}</style>
</head>
<body>

<div></div>
<div></div>
<div></div>

<script src="jquery.js" ></script>
<script>

$( 'div' )
    .height( $(document).height() );

</script>
</body>
</html>
```

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- **Events**
- Effects
- Ajax
- Utilities

`ready()`

`bind()`

`one()`

`trigger()`

`triggerHandler()`

`unbind()`

`live()`

`die()`

`hover()`

`toggle()`

`blur()`

`change()`

`click()`

`dblclick()`

`error()`

`focus()`

`keydown()`

`keypress()`

`keyup()`

`mousedown()`

`mouseenter()`

`mouseleave()`

`mouseout()`

`mouseup()`

`resize()`

`scroll()`

`select()`

`submit()`

`unload()`

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- **Events**
- Effects
- Ajax
- Utilities

`ready()`

`bind()`

`one()`

`trigger()`

`triggerHandler()`

`unbind()`

`live()`

`die()`

`hover()`

`toggle()`

`blur()`

`change()`

`click()`

`dblclick()`

`error()`

`focus()`

`keydown()`

`keypress()`

`keyup()`

`mousedown()`

`mouseenter()`

`mouseleave()`

`mouseout()`

`mouseup()`

`resize()`

`scroll()`

`select()`

`submit()`

`unload()`

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- **Events**
- Effects
- Ajax
- Utilities

```
<!DOCTYPE html>
<html>
<body>

<p>click me</p>
<p>click me</p>

<script src="jquery.js"></script>
<script>

$("p").bind("click", function(){
    $(this).after("<p>click me</p>");
});

</script>
</body>
</html>
```

<http://jsbin.com/ehuki/edit#html>

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- **Events**
- Effects
- Ajax
- Utilities

```
<!DOCTYPE html>
<html>
<body>

<p>click me</p>
<p>click me</p>

<script src="jquery.js"></script>
<script>


$("p").live("click", function(){
    $(this).after("<p>click me</p>");
});

</script>
</body>
</html>
```

<http://jsbin.com/epeha/edit#html>

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- **Effects**
- Ajax
- Utilities



```
show()  
hide()  
toggle()
```


```
slideDown()  
slideUp()  
slideToggle()
```

```
fadeIn()  
fadeOut()  
fadeTo()
```

```
animate()  
stop()
```


Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- **Effects**
- Ajax
- Utilities



```
show()  
hide()  
toggle()  
  
slideDown()  
slideUp()  
slideToggle()  
  
fadeIn()  
fadeOut()  
fadeTo()  
  
animate()  
stop()
```

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- **Effects**
- Ajax
- Utilities

```
<!DOCTYPE html><html><head>
<style>
div{background-color:#bca;width:100px;border:1px
solid green;}
</style>
</head>
<body>
<div id="block">Hello!</div>
<script src="http://ajax.googleapis.com/ajax/
libs/jquery/1.3.2/jquery.min.js" ></script>
<script>

$( "#block" ).animate({
    width: '70%',
    opacity: 0.4,
    marginLeft: '0.6in',
    fontSize: '3em',
    borderWidth: '10px'
}, 1500);

</script></body></html>
```

<http://jsbin.com/evage/edit#html>

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- **Ajax**
- Utilities

jQuery.ajax()
jQuery.get()
jQuerygetJSON()
jQuery.getScript()
jQuery.post()

load()

ajaxCompete()
ajaxError()
ajaxSend()
ajaxStart()
ajaxStop()
ajaxSuccess()

jQuery.ajaxSetup()
serialize()
serializeArray()

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- **Ajax**
- Utilities

jQuery.ajax()
jQuery.get()
jQuerygetJSON()
jQuery.getScript()
jQuery.post()

load()

ajaxComplete()
ajaxError()
ajaxSend()
ajaxStart()
ajaxStop()
ajaxSuccess()

jQuery.ajaxSetup()
serialize()
serializeArray()

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
<!DOCTYPE html><html><body>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/
1.3.2/jquery.min.js" ></script>
<script>
var url = 'http://api.flickr.com/services/feeds/
photos\_public.gne?
tags=jquery&tagmode=all&format=json&jsoncallback=?';

$.getJSON(url, function(data){
    $.each(data.items, function(i,item){
        $("<img/>")
            .attr("src", item.media.m).appendTo("body");
        if ( i == 30 ) return false;
    });
});
</script> </body></html>
```

<http://jsbin.com/erase/edit#html>

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- **Utilities**

\$.support
\$.boxModel

\$.trim()

\$.param()

\$.each(),
\$.extend(),
\$.grep(),
\$.makeArray(),
\$.map(),
\$.inArray(),
\$.merge(),
\$.unique()

\$.isArray(),
\$.isFunction()

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

\$.support
\$.boxModel

\$.trim()

\$.param()

\$.each(),
\$.extend(),
\$.grep(),
\$.makeArray(),
\$.map(),
\$.inArray(),
\$.merge(),
\$.unique()

\$.isArray(),
\$.isFunction()

Overview of jQuery API

- Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

```
$.each(data.items, function(i,item){  
  
    $("<img/>")  
        .attr("src",item.media.m).appendTo("body");  
  
    if ( i == 30 ) return false;  
  
});
```

What happen to the \$ alias and the \$(document).ready() event

```
<!DOCTYPE html> <html>
<head>
<script src="jquery.js"></script>
<script>
$(document).ready(function{
    //jQuery code here
})
</script>
</head>
<body>
</script>
</body>
</html>
```


What happen to the \$ alias and the \$(document).ready() event

```
<!DOCTYPE html> <html>
<head>
<script src="jquery.js"></script>
<script>
$(document).ready(function{
    alert($(document).jquery);
})
</script>
</head>
<body>
</script>
</body>
</html>
```

What happen to the \$ alias and the \$(document).ready() event

```
<!DOCTYPE html> <html>
<head>
<script src="jquery.js"></script>
<script>
$(document).ready(function{
    alert($(document).jquery);
})
</script>
</head>
<body>
</script>
</body>
</html>
```

What happen to the \$ alias and the \$(document).ready() event

```
<!DOCTYPE html>
<html>
<body>
<script src="jquery.js"></script>
<script>
alert($(document).jquery);
</script>
</body>
</html>
```


What happen to the \$ alias and the \$(document).ready() event

```
<!DOCTYPE html>
<html>
<body>
<script src="jquery.js"></script>
<script>
alert($(document).jquery);
(function($){
    alert($.jquery);
})(jQuery);
</script>
</body>
</html>
```

Plugins!

So, what is a plugin?

A plugin is nothing more than a custom jQuery method created to extend the functionality of the jQuery object

```
$( 'ul' ).myPlugin( )
```


Why Create a plugin?

You want to “find something”, and “do something” but the “do something” is not available in jQuery.

Roll your own “do something” with a plugin!

A jQuery plugin in 6 steps

<http://jsbin.com/eheku/edit>

A jQuery plugin in 6 steps

Step 1. create a private scope for \$ alias

```
<!DOCTYPE html><html><body>  
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/  
jquery.min.js"></script>  
<script>  
  (function($) {  
  
  })(jQuery);  
</script></body></html>
```


A jQuery plugin in 6 steps

Step 2. attach plugin to fn alias (aka prototype)

```
<!DOCTYPE html><html><body>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/
jquery.min.js"></script>
<script>
(function($) {
    $.fn.loveNotHate = function() {
        $(this).text($(this).text().replace(/hate/g, 'love'));
    };
})(jQuery);
</script></body></html>
```

A jQuery plugin in 6 steps

Step 2. attach plugin to fn alias (aka prototype)

```
<!DOCTYPE html><html><body>
<p>I hate jQuery!</p>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/
jquery.min.js"></script>
<script>
(function($){
    $.fn.loveNotHate = function(){
        $(this).text($(this).text().replace(/hate/g, 'love'));
    };
})(jQuery);
$('p').loveNotHate();
</script></body></html>
```

A jQuery plugin in 6 steps

Step 3. add implicit iteration

```
<!DOCTYPE html><html><body>
<p>I hate jQuery!</p>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/
jquery.min.js"></script>
<script>
(function($){
    $.fn.loveNotHate = function(){
        this.each(function(){
            $(this).text($(this).text().replace(/hate/g, 'love'));
        });
    };
})(jQuery);
$('p').loveNotHate();
</script></body></html>
```


A jQuery plugin in 6 steps

Step 3. add implicit iteration

```
<!DOCTYPE html><html><body>
<p>I hate jQuery!</p>
<p>I mean really hate jQuery!</p>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/
jquery.min.js"></script>
<script>
(function($){
    $.fn.loveNotHate = function(){
        this.each(function(){
            $(this).text($(this).text().replace(/hate/g, 'love'));
        });
    };
})(jQuery);
$('p').loveNotHate();
</script></body></html>
```

A jQuery plugin in 6 steps

Step 4. enable chaining

```
<!DOCTYPE html><html><body>
<p>I hate jQuery!</p>
<p>I mean really hate jQuery!</p>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/
jquery.min.js"></script>
<script>
(function($){
    $.fn.loveNotHate = function(){
        return this.each(function(){
            $(this).text($(this).text().replace(/hate/g, 'love'));
        });
    };
})(jQuery);
$('p').loveNotHate();
</script></body></html>
```

A jQuery plugin in 6 steps

Step 4. enable chaining

```
<!DOCTYPE html><html><body>
<p>I hate jQuery!</p>
<p>I mean really hate jQuery!</p>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/
jquery.min.js"></script>
<script>
(function($){
    $.fn.loveNotHate = function(){
        return this.each(function(){
            $(this).text($(this).text().replace(/hate/g, 'love'));
        });
    };
})(jQuery);
$('p').loveNotHate().hide();
</script></body></html>
```


A jQuery plugin in 6 steps

Step 5. add default options

```
<!DOCTYPE html><html><body>
<p>I hate jQuery!</p>
<p>I mean really hate jQuery!</p>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/
jquery.min.js"></script>
<script>
(function($) {
    $.fn.loveNotHate = function() {
        return this.each(function() {
            $(this).text($(this).text().replace(/hate/g,
                $.fn.loveNotHate.defaultOptions.text));
        });
    };
    $.fn.loveNotHate.defaultOptions = {text: 'love'};
})(jQuery);
$('p').loveNotHate();
</script></body></html>
```

A jQuery plugin in 6 steps

Step 6. add custom options

```
<!DOCTYPE html><html><body>
<p>I hate jQuery!</p>
<p>I mean really hate jQuery!</p>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/
jquery.min.js"></script>
<script>
(function($){
    $.fn.loveNotHate = function(){
        return this.each(function(){
            $(this).text($(this).text().replace(/hate/g,
                $.fn.loveNotHate.defaultOptions.text));
        });
    };
    $.fn.loveNotHate.defaultOptions = {text:'love'};
})(jQuery);
$('p').loveNotHate({text:'love-love-love'});
</script></body></html>
```


A jQuery plugin in 6 steps

Step 6. add custom options

```
<!DOCTYPE html><html><body>
<p>I hate jQuery!</p>
<p>I mean really hate jQuery!</p>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/
jquery.min.js"></script>
<script>
(function($) {
    $.fn.loveNotHate = function(customOptions) {
        return this.each(function() {
            $(this).text($(this).text().replace(/hate/g,
                $.fn.loveNotHate.defaultOptions.text));
        });
    };
    $.fn.loveNotHate.defaultOptions = {text: 'love'};
})(jQuery);
$('p').loveNotHate({text: 'love-love-love'});
</script></body></html>
```


A jQuery plugin in 6 steps

Step 6. add custom options

```
<!DOCTYPE html><html><body>
<p>I hate jQuery!</p>
<p>I mean really hate jQuery!</p>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/
jquery.min.js"></script>
<script>
(function($) {
    $.fn.loveNotHate = function(customOptions) {
        var options = $.extend({}, $.fn.loveNotHate.defaultOptions,
        customOptions);
        return this.each(function() {
            $(this).text($(this).text().replace(/hate/g,
            $.fn.loveNotHate.defaultOptions.text));
        });
    };
    $.fn.loveNotHate.defaultOptions = {text: 'love'};
})(jQuery);
$('p').loveNotHate({text: 'love-love-love'});
</script></body></html>
```

A jQuery plugin in 6 steps

Step 6. add custom options

```
<!DOCTYPE html><html><body>
<p>I hate jQuery!</p>
<p>I mean really hate jQuery!</p>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/
jquery.min.js"></script>
<script>
(function($) {
    $.fn.loveNotHate = function(customOptions) {
        var options = $.extend({}, $.fn.loveNotHate.defaultOptions,
        customOptions);
        return this.each(function() {
            $(this).text($(this).text().replace(/hate/
            g, options.text));
        });
    };
    $.fn.loveNotHate.defaultOptions = {text: 'love'};
})(jQuery);
$('p').loveNotHate({text: 'love-love-love'});
</script></body></html>
```

Plugins are simple, just follow the steps!

jQuery News

- jQuery.org & Foundation (Software Freedom Law Center)
- Tax-deductible donations
- Four conferences next year (Boston, San Francisco, London, and online)
- Infrastructure Upgrade (MediaTemple Sponsorship)
- New Plugin site (<http://plugins.jquery.com>)
- jQuery Forum (moving away from Google Groups)

?



Jonathan Sharp (@jdsharp)

Special thanks to Cody Lindley