

JQuery Fundamentals

Mike Korcynski
Steve McDonald

Evaluations: http://bit.ly/nercomp_web

Why use a library?

The biggest benefit of a library is two-fold: it should remove the user from dealing with browser issues, and it should provide the user with an *API* that will simplify their development. It must do both for the library to succeed.

John Resig, JQuery creator

What is JQuery?

1. A javascript library
2. Abstraction layer that makes client side javascript easier
3. Offers cross browser support for more browsers than you can reasonably support by yourself.
4. JQuery is free, open source software.
5. Large developer community, lots of plugins, used in 43% of top 10k visited sites

Getting Started

My favorite set of javascript docs:

<https://developer.mozilla.org/en/javascript>

JQuery specific docs:

http://docs.jquery.com/Main_Page

If you want to play along:

<http://jqueryconsole.com/>

Including JQuery

In your code, use a CDN

...

```
<head>
```

```
  <script type="text/javascript"  
src="http://code.jquery.com/jquery-1.6.1.min.js"></script>  
</head>
```

....

or Download it.

Loading JQuery

\$ vs jquery

```
// Use jquery via jquery(...)
  jquery(document).ready(function(){
    jquery("div").hide();
  });
```

vs.

```
// Use jquery via jquery(...)
  $(document).ready(function(){
    $("div").hide();
  });
```

Avoid conflict in existing apps or when mixing with other JS libraries, release the \$ with:

```
jquery.noConflict()
```

Selecting in the DOM

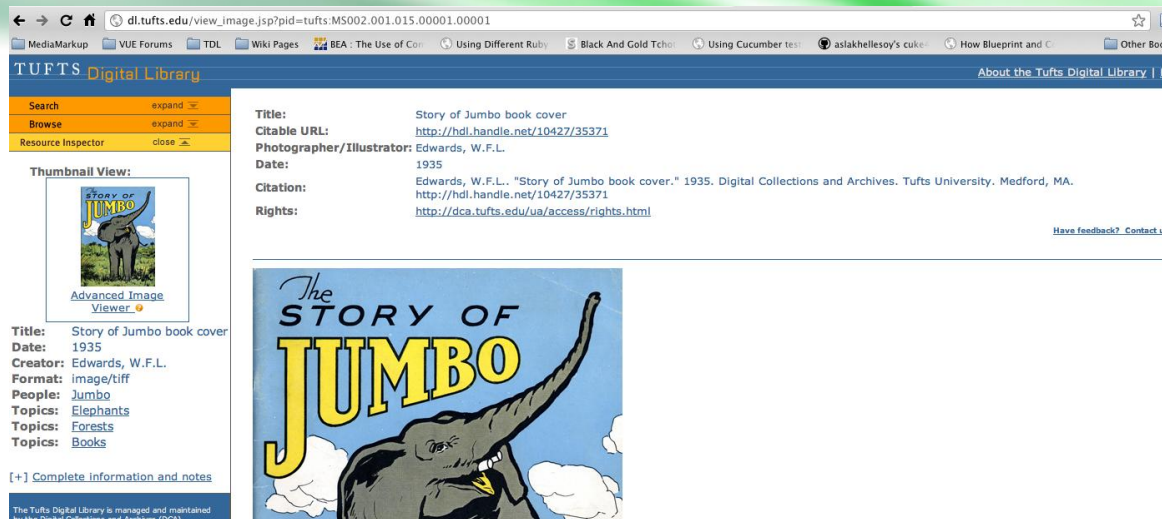
```
<html>
```

```
...
```

```
<body>
```

```
...
```

```
<td id="photographer">  
Edwards, W.F.L.</td>
```



```
<td id="citation" class="textBlue">Edwards, W.F.L.. "Story of  
Jumbo book cover." 1935. Digital Collections and Archives. Tufts  
University. Medford, MA.  
http://hdl.handle.net/10427/35371</td>...
```

JQuery Selectors

Find all the `<td>` `$('td')` will return an array of elements

Find the citation `$('#citation')`

Non matches return an empty array

Selectors (continued)

`$('#p:first')` will return the first paragraph element in the DOM as an array

`$('#p div')` will return an array of all divs under paragraphs

`$('#p > div')` will return an array of divs directly under paragraphs

`$('#div.odd')` find all divs of class odd

`$('#div.odd, div.even')` multiple matching

Manipulating the DOM

Common ways:

`.css()`

`$('#p#someId').css("font-size")` returns a string "16px"

`$('#p#someId').css("font-size","48px");`

`.html() + .text()`

`$('#p#title').text()` returns "Story of Jumbo"

passing HTML to `.text()` results in escaping.

`$('#p#title').html()` returns all the html in the `<p>`

Timing Issues

```
$(document).ready(function(){  
  $("div").hide();  
});
```

- `$(document).ready()` can be comforting
- There are times it can slow down page load unnecessarily, give thought to what you put in there.
- you can use this to fire events very early compared to `onLoad`

JSON

- javascript object notation
- essentially a map of data
- lightweight, good for interchange
- much easier to parse than XML

```
<body>
  <div id="images">

</div>
<script>
$.getJSON("http://api.flickr.com/services/feeds/photos_public.gne?jsoncallback=?",
{
  tags: "cat",
  tagmode: "any",
  format: "json"
},
function(data) {
  $.each(data.items, function(i,item){
    $("<img/>").attr("src", item.media.m).appendTo("#images");
    if ( i == 3 ) return false;
  });
});</script>

</body>
```

Events

```
$(document).ready(function() {  
  $("p#title").click(function() {  
    $(this).hide();  
  });  
});
```

Other events:

```
.hover()  
.keypress(function(event) {  
  if (event.which == ? )  
  });
```

Transitions

.hide()
.show()
.toggle()

```
$(".btn-slide").click(function(){  
$("#panel").slideToggle("slow");  
$(this).toggleClass("active"); return false; });
```

<http://www.webdesignerwall.com/demo/jquery/simple-slide-panel.html>

.fadeIn()
.fadeOut()
.fadeToggle()

.slideUp()
.slideDown()
.slideToggle()

Loading w/ AJAX

`.load()`

Simpler than `getJSON`, can't do as much with it.

```
$('#result').load('ajax/test.html');
```

can also be used with callbacks:

```
$('#result').load('ajax/test.html', function() {  
    alert('Load was performed.');
```

jQuery + SOLR

```
$(document).ready(function() {
$.getJSON('querySolr' , function(events) {
var temp = "";
solrDocs = events.response.docs;
var i=0;
for (i =0; i < solrDocs.length; i++)
{
link = str = '#'+jQuery.trim(solrDocs[i].EventId);
temp = temp + "<li><a
href=\""+link+\">"+solrDocs[i].EventTitle+"</a></li>";
}
$('#eventList').html(temp);
$('ul').listview('refresh');
//insert event details pages.
```


JQuery + SOLR

```
for (i=0; i < solrDocs.length; i++)
{
temp = "";
temp = temp + "<div class=\"ui-page\" data-role=\"page\"
id=\""+jQuery.trim(solrDocs[i].EventId)+"\">";
temp = temp + "<div data-role=\"header\">";
temp = temp + "<h1>"+solrDocs[i].EventTitle+"</h1></div>";
temp = temp + "<div data-role=\"content\">";
temp = temp + "<p>"+solrDocs[i].EventDescription+"</p>";
temp = temp + "</div>";
//temp = temp + "<div data-role=\"footer\">";
//temp = temp + "<h4>Page Footer</h4>";
//temp = temp + "</div></div>";
$('body').append(temp);
// $('ul').listview('refresh')
// $("#"+jQuery.trim(solrDocs[i].EventId)).page();
```

jQuery Mobile

- currently in Alpha 4
- progressively enhances, light, vast support of mobile browsers
- the basics are as easy as just writing HTML

Back Lists with Forms

Text Input:

Textarea:

Search Input:

Flip switch: Off

Slider: 0

Choose as many snacks as you'd like:

- Cheetos
- Doritos
- Fritos
- Sun Chips

```
▼ <div data-role="content" class="ui-content" role="main">
  ▼ <form>
    ▼ <ul data-role="listview" class="ui-listview">
      ▼ <li data-role="fieldcontain" class="ui-li ui-li-static ui-body-c ui-field-contain ui-body ui-br">
        <label for="name" class="ui-input-text">Text Input:</label>
        <input type="text" name="name" id="name" value class="ui-input-text ui-body-null ui-corner-all ui-shadow-inset ui-body-c">
      </li>
      ▼ <li data-role="fieldcontain" class="ui-li ui-li-static ui-body-c ui-field-contain ui-body ui-br">
        <label for="textarea" class="ui-input-text">Textarea:</label>
        <textarea cols="40" rows="8" name="textarea" id="textarea" class="ui-input-text ui-body-null ui-corner-all ui-shadow-inset ui-body-c"></textarea>
      </li>
      ▼ <li data-role="fieldcontain" class="ui-li ui-li-static ui-body-c ui-field-contain ui-body ui-br">
        <label for="search" class="ui-input-text">Search Input:</label>
        ▶ <div class="ui-input-search ui-shadow-inset ui-btn-corner-all ui-btn-shadow ui-icon-searchfield ui-body-c">...</div>
      </li>
      ▶ <li data-role="fieldcontain" class="ui-li ui-li-static ui-body-c ui-field-contain ui-body ui-br">...</li>
    </ul>
  </form>
</div>
```

<http://www.jquerymobile.com/>

Dicussion + Experimentation

Evaluations: http://bit.ly/nercomp_web