

# jQuery

## Download

<http://jquery.com/download/>

## API

<http://api.jquery.com/>

Plugins have neat pre-defined solutions like the User Interface or UI plugin.

<http://plugins.jquery.com/tag/ui/>

<http://www.tutorialspoint.com/jquery/jquery-plugins.htm>

We must ensure that it appears after the main jQuery source file, and before our custom JavaScript code.

Load jQuery **from a CDN**, like google or download and **reference directly** after downloading.

```
//<script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js" > </script>
<script type="text/javascript" src="../../SiteAssets/jquery.js"></script>
```

```
// PLUGINS would go here!!
```

```
//<script type="text/javascript" src="http://ftrujillo-lap/NVE/js/common_functions.js"></script>
<script type="text/javascript" src="../../SiteAssets/common_functions.js"></script>
<script type="text/javascript">
```

```
// This is an absolute after the page loads. If you have queries to databases to populate tables,
// and you want to act upon that data, then you MUST use this entry point; otherwise, use ready().
```

```
$(window).load(function() {
    var currentUrl = $.getCurrentUrl();
    var fjt_id = $.getParamForUrl("FJT_ID", currentUrl);
    $.setIDLinkage("MeetingID", fjt_id);
});
```

```
jQuery(document).ready(function($) {
    $.displayUrl(); // use this initially to see if you have connectivity.
    // Then, put your calls here.
})(jQuery);
</script>
```

## HTML Tags

STRUCTURE	=> html head body <b>div span</b>
META	=> title link meta style
TEXT	=> <b>p h1 h2 h3 h4 h5 h6</b>
LINKS	=> <b>a</b> base
IMAGE	=> <b>img</b> area map object param
LISTS	=> <b>ul li ol dl dt dd</b>
FORMS	=> <b>form input textarea select option optgroup button label</b> fieldset legend
TABLE	=> <b>table th tr td tbody thead tfoot col colgroup</b> caption

## Core Attributes

id	=> <p <b>id</b> ="jelly">Wobble wobble wobble.</p>
class	=> <div><span <b>class</b> ="custard">yellow</span></div>
style	=> <p <b>style</b> ="background-color: blue; color: yellow;">Yellow text on a blue background.</p>
title	=> Adds a title to an element.

## Event Attrs <http://api.jquery.com/category/events/>

onclick	href	<b>click()</b>
ondblclick		<b>dblclick()</b>
onmousedown		<b>mousedown()</b>
onmouseup		<b>mouseup()</b>
onmouseover		<b>mouseover()</b>
onmousemove		<b>mousemove()</b>
onmouseout		<b>mouseout()</b>
onkeypress		<b>keypress()</b>
onkeydown		<b>keydown()</b>
onkeyup		<b>keyup()</b>
onfocus		<b>focus()</b>
onblur		<b>blur()</b>
onhover		<b>hover()</b>

// This method will **register a listener** for the focusOnContentBtn when pressed.

```
$.fn.focusOnContentShowTitle = function() {
    var focusOnContentBtn = $("span#fullscreenmodebox");

    // If focusOnContentBtn is clicked, then do the following.
    focusOnContentBtn.click(function() {
        $.displayTitle();
    });
}; // focusOnContentShowTitle
```

To **click an anchor button**, given it has an href or onclick attribute set.

```
$("#a#ctl00_fullscreenmodeBtn").click();
```

## iQuery Selectors <http://api.jquery.com/category/selectors/> <http://api.jquery.com/category/selectors/form-selectors/>

```
$("#div")
$("#div > p")
$("#div#theID")
$("#div:hidden")
$("#div:visible")

$("#theID")

$(".theCLASS")

$("[someAttributeName]")
$("[someAttributeName=100]")
$("#div[someAttributeName]")
$("#div[someAttributeName!=200]")

:first
:last
:even
:odd
:eq(index)
:gt(index)
:lt(index)
:input           (Selects all input fields in the page)
:text           (Selects all text input fields except textarea)
:radio
:checkbox
:checked
:first-child
:last-child


children that are contained in parent)

$("#div, p, span") (Select ALL div, p, and span elements)


```

## Traversing Selected Elements <http://api.jquery.com/category/traversing/>

```
contains()
eq(0)
each()
first()
has()
is()
last()
next()
prev()
parent()
parents()
children()
siblings()
closest("selector or jQuery obj") // Traverse UP the DOM
find("selector or jQuery obj")    // Traverse DOWN the DOM
```

```
$(".someClass").each(function(index,element) {
    // element.doSomething();
});

$(".someClass").each(function(index) {
    // $(this).doSomething();
});

$(".someClass").each(function() {
    // If you do not use index, then you can leave it off
    // $(this).doSomething();
});
```

## CSS Manipulation <http://api.jquery.com/category/css/>

```
CSS("cssAttribute"); // Getter
CSS("cssAttribute", "value"); // Setter

$(".someClass").css( // Setting more than one key/val pair
{
    'background-color' : '#ff00ff',
    'border' : '1px solid #000000'
}
);

height();
width();

$("#theElement").addClass('newClass1 newClass2');
$("#theElement").hasClass('newClass1');
$("#theElement").toggleClass('newClass1 newClass2');
```

## Effects <http://api.jquery.com/category/effects/>

```
show()
hide()
delay()
fadeIn()
fadeOut()

$( "#mybutton" ).click(function() {
    $( "div.first" ).slideUp( 300 ).delay( 800 ).fadeIn( 400 );
    $( "div.second" ).slideUp( 300 ).fadeIn( 400 );
});
```

## **DOM Manipulation**

<http://api.jquery.com/category/attributes/>  
<http://api.jquery.com/category/manipulation/>

```
text()           text('someNewText')  
html()          html('New <b>text</b>')  
val()           val('newValue');  
  
attr('someAttr')  attr('someAttr', 'someNewValue')
```

```
$(".someClass").attr(    // Setting more than one attr  
    {  
        'height' : '100',  
    }  
);
```

```
removeAttr('someAttr')  
prepend('someHtml')  
append('someHtml')
```

### **Insert**

```
$("#theElement").before('someHtml') // Inserts someHtml before #theElement  
$("#theElement").after('someHtml')  // Inserts someHtml after #theElement
```

### **Move**

```
$("#theElement").insertBefore('#anotherElement') // Move #theElement to BEFORE #anotherElement  
$("#theElement").insertAfter('#anotherElement')  // Move #theElement to AFTER #anotherElement
```

### **Wrap**

```
$("#theElement").wrap(<div class="myClass" > </div>) // Wraps #theElement in the given HTML
```

### **Replace**

```
$("#theElement").replaceWith(<div class="myClass" > </div>) // Replaces #theElement with DIV element  
<div class="myClass" > </div>).replaceAll("#theElement") // Replaces #theElement with DIV element
```

### **Prune**

```
$("#theElement").empty() // Remove ALL child elements of the selected element
```

### **Copy**

```
var clone = $('#theElement').clone(); // Clone the selected element
```

## Example code

```
C:\Users\frujillo\Documents\NVE_SSD_DEVELOPMENT\js\common_functions.js (js) - Sublime Text 2
File Edit Selection Find View Goto Tools Project Preferences Help

common_functions.js *
1 // common_functions.js
2 //
3 // http://jshint.com/
4 // http://betterembsw.blogspot.com/2014/06/avoid-high-cyclomatic-complexity.html
5 //
6 // http://jsfiddle.net/
7
8
9 // 1) DOWNLOAD CHROME and jQuery debug plugin.
10 // a. Your page or form MUST have jQuery.min loaded for this to work.
11 // i. Make your paths relative in production. You can have a DEBUG path outside SP2013.
12
13 // // These are relative paths from your page or form when you call from CEWP
14 // <script type="text/javascript" src="../../SiteAssets/jquery.js"></script>
15 // <script type="text/javascript" src="../../SiteAssets/common_functions.js"></script>
16 // <script type="text/javascript" src="http://mylap/NVE/js/common_functions.js"></script>
17
18 // b. Press F12, Elements Tab => Select SEARCH spyglass => find WHERE your item exists.
19
20 // c. If you are modifying SP2013 display, edit, or new forms....
21 // You MUST use $(window).load(function() VS jQuery(document).ready(function()
22 // The data in the form is NOT populated on the ready event.
23 //
24 // executes when complete page is fully loaded, including all frames, objects and images
25
26 // $(window).load(function() {
27 // YOUR STUFF
28 // });
29
30 // 2) Use http://jsfiddle.net to check your jQuery code for syntax errors and to debug offline.
31
32
33 // http://www.sitepoint.com/5-ways-declare-functions-jquery/
34
35 // You can write plain, old javascript function
36 function displayHelloWorldAlert() {
37     alert("Hello World from common_functions.js");
38 }
39
40 function get_MM_DD_YYYY() {
41     var fullDate = new Date();
42     var twoDigitMonth = fullDate.getMonth() + 1 + "";
43     if (twoDigitMonth.length == 1) twoDigitMonth = "0" + twoDigitMonth;
44     var twoDigitDate = fullDate.getDate() + "";
45     if (twoDigitDate.length == 1) twoDigitDate = "0" + twoDigitDate;
46     var currentDate = twoDigitMonth + "/" + twoDigitDate + "/" + fullDate.getFullYear();
47     return(currentDate);
48 }
49
```

```
C:\Users\frujillo\Documents\NVE_SSD_DEVELOPMENT\js\common_functions.js (js) - Sublime Text 2
File Edit Selection Find View Goto Tools Project Preferences Help

common_functions.js x
49
50 // This is for Microsoft m/d/yyyy where month and day are NOT zero padded.
51 function getCurrentDate() {
52     var fullDate = new Date();
53     var month = fullDate.getMonth() + 1 + "";
54     var date = fullDate.getDate() + "";
55     var currentDate = month + "/" + date + "/" + fullDate.getFullYear();
56     return(currentDate);
57 }
58
59 // You may or may not need noConflict();
60 // http://learn.jquery.com/using-jquery-core/avoid-conflicts-other-libraries/
61 // jQuery.noConflict(); // http://api.jquery.com/jQuery.noConflict/
62
63 // You can define plugins like this.
64 // $.fn is alias for jQuery.prototype
65 jQuery.prototype.hello = function () {
66     alert("hello");
67 };
68
69 // Or, you may wish to use the $.fn
70 (function ($) { // Start jQuery namespace
71     $.fn.world = function () {
72         alert("world");
73     }; // watch your semicolons!!
74
75     $.fn.getCurrentUrl = function () {
76         // In plain JavaScript => window.location.href
77         return $(location).attr('href');
78     };
79
80     // var url = "http://www.sitepoint.com/url-parameters-jquery/?city=BOISE";
81     //
82     // var city = decodeURIComponent($.getParamForUrl('city', url));
83     //
84     // http://www.w3schools.com/jsref/jsref_encodeuricomponent.asp
85     // http://www.w3schools.com/jsref/jsref_decodeuricomponent.asp
86     //
87     $.fn.getParamForUrl = function (name, url) {
88         var results = new RegExp('[\\?&]' + name + '=(^&#)+').exec(url);
89         if (results === null) { // Uses === and not ==
90             return null;
91         } else {
92             return results[1] || 0;
93         }
94     };
95
96     // This will get the parameter by name for the CURRENT URI
97 }
```

```
C:\Users\frujillo\Documents\NVE_SSD_DEVELOPMENT\js\common_functions.js (js) - Sublime Text 2
File Edit Selection Find View Goto Tools Project Preferences Help

common_functions.js *
97 // This will get the parameter by name for the CURRENT URL.
98 $.fn.getParam = function (name) {
99     var results = $.getParamForUrl(name, $.getCurrentUrl());
100     return results;
101 };
102
103 // Display the CURRENT URL in an alert.  DEBUGging
104 $.fn.displayUrl = function () {
105     var url = $.getCurrentUrl();
106     alert("URL => " + url);
107 };
108
109
110 // Returns true/false if a SharePoint form.
111 $.fn.isSharePointForm = function (url) {
112     var isForm = false;
113     var results = new RegExp("([A-Z][a-z]+Form).aspx\\?").exec(url);
114
115     if (results !== null) {
116         isForm = true;
117     }
118     return isForm;
119 };
120
121 // Returns true/false if a DISPLAY SharePoint form.
122 $.fn.isSharePointDispForm = function (url) {
123     var isForm = false;
124     var results = new RegExp("(DispForm).aspx\\?").exec(url);
125
126     if (results !== null) {
127         isForm = true;
128     }
129     return isForm;
130 };
131
132 // Returns true/false if a EDIT SharePoint form.
133 $.fn.isSharePointEditForm = function (url) {
134     var isForm = false;
135     var results = new RegExp("(EditForm).aspx\\?").exec(url);
136
137     if (results !== null) {
138         isForm = true;
139     }
140     return isForm;
141 };
142
```

```
C:\Users\ftujillo\Documents\NVE_SSD_DEVELOPMENT\js\common_functions.js (js) - Sublime Text 2
File Edit Selection Find View Goto Tools Project Preferences Help

common_functions.js x
142
143 // Returns true/false if a NEW SharePoint form.
144 $.fn.isSharePointNewForm = function (url) {
145     var isForm = false;
146     var results = new RegExp("(NewForm).aspx\\?").exec(url);
147
148     if (results !== null) {
149         isForm = true;
150     }
151     return isForm;
152 };
153
154
155 // I do not have this play in production. Saving for future use.
156 $.fn.updateDocLinks = function () {
157     $("td.ms-list-addnew a").each(function () {
158         var m_href = $(this).attr('href');
159         var m_pattern = new RegExp("Upload.aspx[\\?]");
160
161         if (m_href !== null) {
162             var m_match = m_pattern.exec(m_href);
163             if (m_match !== null) {
164                 m_href = m_href.replace(m_pattern, "NewLink.aspx?");
165                 var td1 = "<tr><td><a class='ms-heroCommandLink' href='" + m_href + "'>Link To Docum
166                 //alert("HREF => " + m_href);
167                 $(this).closest("tr").after(td1);
168             }
169         }
170     }); // td.ms-list-addnew a
171 }; // updateDocLinks
172
173
174
175
```



```
C:\Users\frujillo\Documents\NVE_SSD_DEVELOPMENT\js\common_functions.js (js) - Sublime Text 2
File Edit Selection Find View Goto Tools Project Preferences Help

common_functions.js x
175
176 // Takes an externally derived ID number => var id = $.getParamForUrl("ID", currentUrl);
177 //
178 // Finds all ADD NEW anchors that have both ONCLICK and HREF attributes.
179 // Then, modifies HREF to have fieldName=ID in the NEW NewItem2() URI.
180 // An finally, removes the old ONCLICK attr.
181 //
182 // $.updateNewAnchorsWithID(id, "FJT_ID"); *** DO NOT use "ID", but prefix it => "SOMETHING_ID"
183 //
184 $.fn.updateNewAnchorsWithID = function (id, fieldName) {
185     var debug = true;
186
187     // Internet explorer. Use F12, select White arrow selector on the Debug Console that just popped up.
188     // then click on an item on the Webpage to find class, id, name, etc...
189     // I tried using a:contains('Add a new item') and a:contains("NewItem") with no success.
190     // Maybe I am too new to jQuery. Anyway, I am still using jQuery and narrowing as I go.
191
192     // First, I get all anchors inside of divs. This list could be large. Don't care.
193     $("#id.ms-list-addnew a").each(function (index) {
194         var m_onclick = $(this).attr('onclick');
195         var m_href = $(this).attr('href');
196
197         // Only those anchors with onclick event are candidates. This narrows down selection somewhat.
198         if (m_onclick !== null && m_href !== null) {
199             var newItem_results = new RegExp("NewItem.*").exec(m_onclick);
200
201             // If the onclick has an event handler with a call to NewItem***, then this is the item we want to modify.
202             if (newItem_results !== null) {
203                 var id_results = new RegExp("[\\?\\&]ID=", "i").exec(m_onclick); // case insensitive ID or id or Id
204
205                 // Verify that we do not see ID=*** or id=*** in the onclick event,
206                 // and if we do not then, let's insert the ID after the aspx?
207                 if (id_results === null) {
208                     var regex_pattern = new RegExp("aspx[\\?]*");
209                     var results = regex_pattern.exec(m_onclick);
210
211                     if (results !== null) {
212                         //m_onclick = m_onclick.replace(regex_pattern, replace_string);
213                         m_href = m_href.replace(regex_pattern, "aspx?" + fieldName + "=" + id + "&");
214
215                         // I had to REMOVE onclick event to get new HREF to be in play.
216                         $(this).removeAttr("onclick");
217                         $(this).removeAttr("href");
218                         var newURL = 'javascript:NewItem2(event, \'' + m_href + '\');';
219                         $(this).attr("href", newURL);
220
221                         // You can view this in IE Console (Press F12), then RUN script
222                         // ...
223                     }
224                 }
225             }
226         }
227     });
228 }
```

```
C:\Users\frujillo\Documents\NVE_SSD_DEVELOPMENT\js\common_functions.js (js) - Sublime Text 2
File Edit Selection Find View Goto Tools Project Preferences Help

common_functions.js x
220
221 // You can view this in IE Console (Press F12), then RUN script
222 if (debug) {
223     console.log(index + ": Added ID=" + id);
224     console.log("href => " + $(this).attr('href'));
225     console.log("\n");
226 }
227 }
228 } // if (id_results === null)
229 } // if (newItem_results !== null)
230 } // if (m_onclick !== null)
231 }); // td.ms-list-addnew a
232 }; // updateNewAnchorsWithID
233
234
235 // You MUST use $(window).load(function() VS $(document).ready(function()
236 // The data in the form is NOT populated on the latter.
237 //
238 // var id = $.getParamForUrl("ID", currentUrl);
239 // $.insertIdFirstRow(id);
240 //
241 $.fn.insertIdFirstRow = function (id) {
242     var td1 = "<td class='ms-label'><h3 class='ms-standardheader'>ID</h3></td>";
243     var td2 = "<td class='ms-formbody'>" + id + "</td>";
244     var tr1 = "<tr>" + td1 + td2 + "</tr>";
245     $('tr#idAttachmentsRow').closest('table').find('tr:first').before(tr1);
246 }; // insertIdFirstRow
247
248
249 // You MUST use $(window).load(function() VS $(document).ready(function()
250 // The data in the form is NOT populated on the latter.
251 //
252 // var id = $.getParamForUrl("ID", currentUrl);
253 // $.insertIdLastRow(id);
254 //
255 $.fn.insertIdLastRow = function (id) {
256     var td1 = "<td class='ms-label'><h3 class='ms-standardheader'>ID</h3></td>";
257     var td2 = "<td class='ms-formbody'>" + id + "</td>";
258     var tr1 = "<tr>" + td1 + td2 + "</tr>";
259     $('tr#idAttachmentsRow').closest('table').find('tr:last').after(tr1);
260 }; // insertIdLastRow
261
262
263 // This is grabbing passed ID value from current URL, then selecting the
264 // proper choice while disabling selection from the user. The value is
265 // left being shown. You may want to .hide() at some point.
266 //
267 //
```

```
C:\Users\frujillo\Documents\NVE_SSD_DEVELOPMENT\js\common_functions.js (js) - Sublime Text 2
File Edit Selection Find View Goto Tools Project Preferences Help

common_functions.js x
267 //if($.isSharePointNewForm(currentUrl)) {
268 // // Extract the FJT_ID passed in param from the Display form.
269 // var fjt_id = $.getParamForUrl("FJT_ID", currentUrl);
270 // $.setIDLinkage("MeetingCalendarID", fjt_id);
271 // }
272
273 $.fn.setIDLinkage = function(fieldTitle, lookupVal) {
274 //Set default value for lookups with less than 20 items
275 var selectorString = "select[title='" + fieldTitle + "']";
276
277 //alert("selectorString => " + selectorString);
278
279 if ($(selectorString).html() !== null) {
280 $(selectorString).val(lookupVal); // SET the new value
281 $(selectorString).attr("disabled");
282 $(selectorString).attr("disabled", "disabled");
283 $(selectorString).closest("tr").hide();
284 //$(selectorString).closest("tr").show();
285 } else {
286 var inputSelector = "input[title='" + fieldTitle + "']";
287 var choices = $(inputSelector).attr("choices");
288 var hiddenInput = $(inputSelector).attr("optHid");
289 var hiddenInputSelector = "input[id='" + hiddenInput + "']";
290
291 // Set the input selection to lookupVal
292 $(hiddenInputSelector).attr("value", lookupVal);
293
294 var choiceArray = choices.split("|");
295 for (var index = 1; index < choiceArray.length; index = index + 2) {
296 if (choiceArray[index] == lookupVal) {
297 $(inputSelector).val(choiceArray[index - 1]);
298 // make sure we can not change the selection.
299 $(inputSelector).attr("disabled");
300 $(inputSelector).attr("disabled", "disabled");
301 $(inputSelector).closest("tr").hide();
302 //$(inputSelector).closest("tr").show();
303 }
304 }
305 }
306
307 }; // setIDLinkage
308
309 // This will disable a choice. (either dropdown or mult-select)
310 //
311 // if($.isSharePointEditForm(currentUrl)) {
312 // $.disableSelectElement("MeetingCalendarID");
313 // }
```

```
C:\Users\frujillo\Documents\NVE_SSD_DEVELOPMENT\js\common_functions.js (js) - Sublime Text 2
File Edit Selection Find View Goto Tools Project Preferences Help

common_functions.js x
315 $.fn.disableSelectElement = function(fieldTitle) {
316     //Set default value for lookups with less than 20 items
317     var selectorString = "select[title='" + fieldTitle + "']";
318     //alert("selectorString => " + selectorString);
319
320     if ($(selectorString).html() !== null) {
321         $(selectorString).attr("disabled");
322         $(selectorString).attr("disabled", "disabled");
323         $(selectorString).closest("tr").hide();
324         //$(selectorString).closest("tr").show();
325     } else {
326         var inputSelector = "input[title='" + fieldTitle + "']";
327         $(inputSelector).attr("disabled");
328         $(inputSelector).attr("disabled", "disabled");
329         $(inputSelector).closest("tr").hide();
330         //$(inputSelector).closest("tr").show();
331     }
332
333 }; // disableSelectElement
334
335 // There was an instance where I did NOT want the Chrome Titles to be a
336 // clickable HREF, but rather just TEXT.
337 // Reads the HREF, splits the path, and ADD the last item as TEXT while
338 // removing the HREF.
339 //
340 $.fn.disableChromeTitles = function() {
341     $("div.ms-webpart-chrome-title a").each(function () {
342         var m_href = decodeURIComponent($(this).attr('href'));
343
344         if(m_href !== null) {
345             var arr_href = m_href.split("/");
346             var title = arr_href[arr_href.length-1];
347             $(this).before(title);
348             $(this).remove();
349         }
350     });
351 }; // disableChromeTitles
352
353 // This method will register a listener for the focusOnContentBtn when pressed.
354 $.fn.focusOnContentShowTitle = function() {
355     var focusOnContentBtn = $("span#fullscreenmodebox");
356
357     // If focusOnContentBtn is clicked, then do the following.
358     focusOnContentBtn.click(function() {
359         $.displayTitle();
360     });
361 }; // focusOnContentShowTitle
```

```
C:\Users\frujillo\Documents\NVE_SSD_DEVELOPMENT\js\common_functions.js (js) - Sublime Text 2
File Edit Selection Find View Goto Tools Project Preferences Help

common_functions.js x
363
364 $.fn.displayTitle = function() {
365     // First, grab the innerHTML that holds the TITLE of sub site.
366     var title = $("span#DeltaPlaceHolderPageTitleInTitleArea").html();
367
368     if( title !== null) {
369         // Sharepoint uses CSS to show and hide the focus on content solution.
370         // Using jQuery .css() method, we can get the style => display and it's value.
371         var style_display = $("div#s4-titlerow").css("display");
372
373         if(style_display !== null) {
374
375             // If the display is "none", then we need to add a row to one of the tables that is still visible.
376             if(style_display === "none") {
377                 var myRow = '<tr id="fjt_title_row"><td><h1 id="fjt_title" class="ms-core-pageTitle">' + title + "</h1></td>";
378
379                 $("table#layoutsTable tr:first").before(myRow);
380                 $("#fjt_title").css("color", "black");
381             }
382             // If the display is "block", then we are displaying NORMAL mode and need to remove the above title.
383             if(style_display === "block") { // If not, then remove my inserted row.
384                 $("#fjt_title_row").remove();
385             }
386         }
387     }
388 };
389
390 $.fn.getSelectedMeeting = function() {
391
392     var selectedMeeting = 'table.ms-listviewtable:contains("Select") img[alt="Selected"]';
393     var meetingDate = $(selectedMeeting).closest("tr").find("td:eq(2)").text();
394     var myRow = '<tr id="fjt_currmeeting_row"><td><h2 id="fjt_currmeeting">' + meetingDate + "</h2></td></tr>";
395
396     $("table#layoutsTable tr:first").before(myRow);
397
398     $("#fjt_currmeeting").css("color", "black");
399     $("#fjt_currmeeting").css("background-color", "#ffbb00");
400     $.displayTitle();
401 }; // getSelectedMeeting
402
403
404 $.fn.hideSharepointLeftNav = function() {
405     $("#sideNavBox").css('display', 'none');
406     $("#contentBox").css('margin-left', '0px');
407 }; // hideSharepointLeftNav
408
409 })(jQuery); // End jQuery namespace
410
```

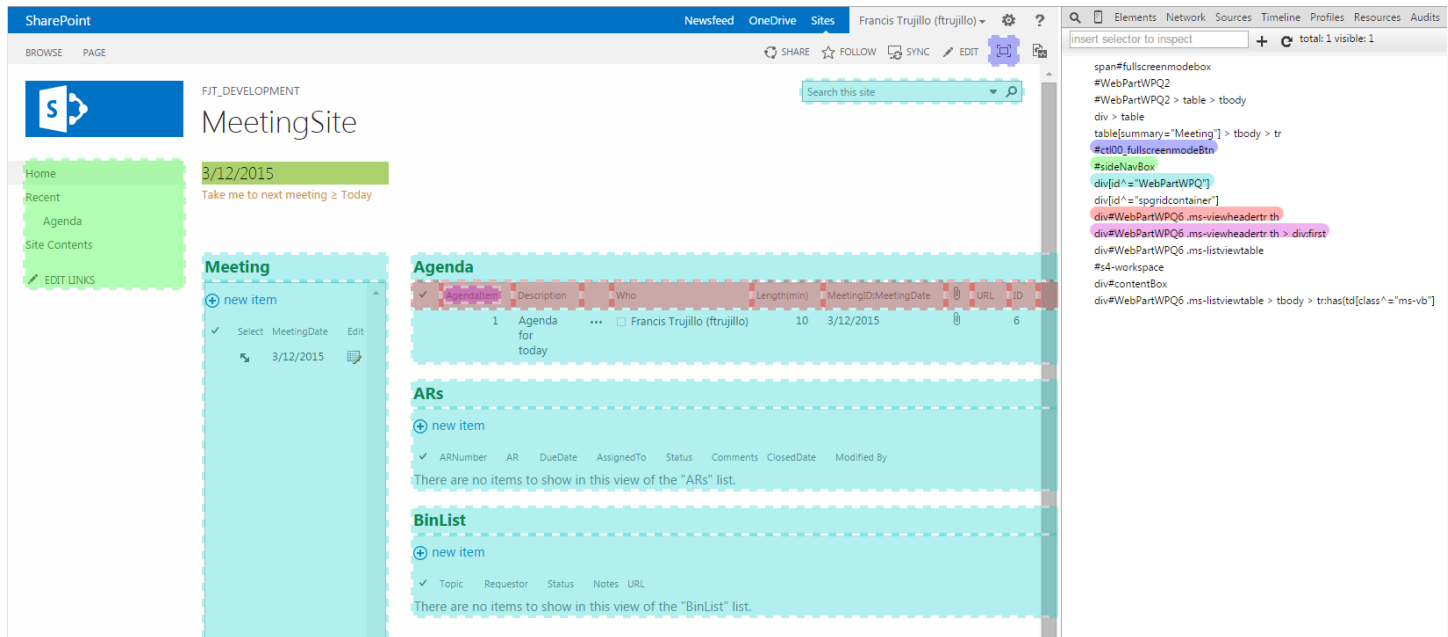
```
C:\Users\frujillo\Documents\NVE_SSD_DEVELOPMENT\js\common_functions.js (js) - Sublime Text 2
File Edit Selection Find View Goto Tools Project Preferences Help

common_functions.js x
452
453 $.fn.meetingScrollBarSelectCurrDate = function() {
454     var viewParam = $.getParam("View");
455     var selectedID = $.getParam("SelectedID");
456
457     // Set up a scroll bar on WebPart. By default, "div"'s will consume all
458     // space unless CCS bounded. The overflow attribute along with height and
459     // width will constrain the WebPart.
460     var container = $("#div#WebPartWPQ2");
461     container.height("475px");
462     container.width("205px");
463     container.css("overflow-x", "hidden");
464     container.css("overflow-y", "scroll");
465
466     // Determine all of the rows in the Sharepoint list. (table)
467     var allRows = $("table[summary='Meeting'] > tbody > tr");
468
469     // If this is FIRST time on page, then click on MeetingDate >= CurrentDate
470     if(viewParam === null && selectedID === null) {
471         var cDate = new Date(getCurrentDate());
472
473         allRows.each(function() {
474             var meetingDate = $(this).find("td:eq(2)").text();
475             var mDate = new Date(meetingDate);
476
477             // We want to select the Next Meeting date since the SP list
478             // is sorted on MeetingDate already.
479             if(+mDate >= +cDate) { // MUST be Date objects to compare.
480                 $(this).find("td:eq(1)").find("a").click();
481                 return(false); // returning false is like a 'break' in C
482             }
483         });
484     } else {
485         // Whenever, we have a selected MeetingID, then Scroll to ID in the list.
486         if(selectedID !== null) {
487             allRows.each(function() {
488                 var meetingID = $(this).find("td:eq(4)").text();
489
490                 if(selectedID === meetingID) {
491                     var scrollTo = $(this);
492                     container.animate({
493                         // http://www.w3schools.com/jsref/prop_element_scrolltop.asp
494                         scrollTop: scrollTo.offset().top - container.offset().top + container.scrollTop()
495                     });
496                     return(false); // returning false is like a 'break' in C
497                 }
498             });
499         }
500     }
501 }; // meetingScrollBarSelectCurrDate
502
```

## Chrome jQuery Debugger plugin

<https://chrome.google.com/webstore/detail/jquery-debugger/dbhnnnpaeobfddmlalhnehgclcmjimi?hl=en>

1. Install plugin in Chrome.
2. Your page MUST have jquery loaded on page.
3. Press **F12**, then on left hand side Select the **jQuery Selector** tab.
4. Type in your selector, then hover over it to see what it selects, in anything.

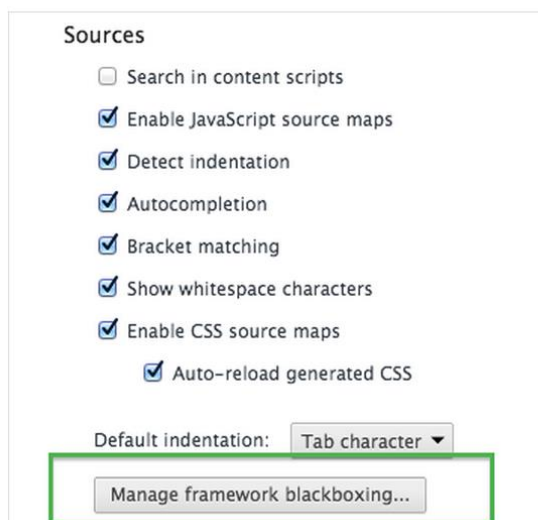


<https://developer.chrome.com/devtools/docs/blackboxing>

1. Using the Settings panel
2. Right-clicking on any script in the Sources panel

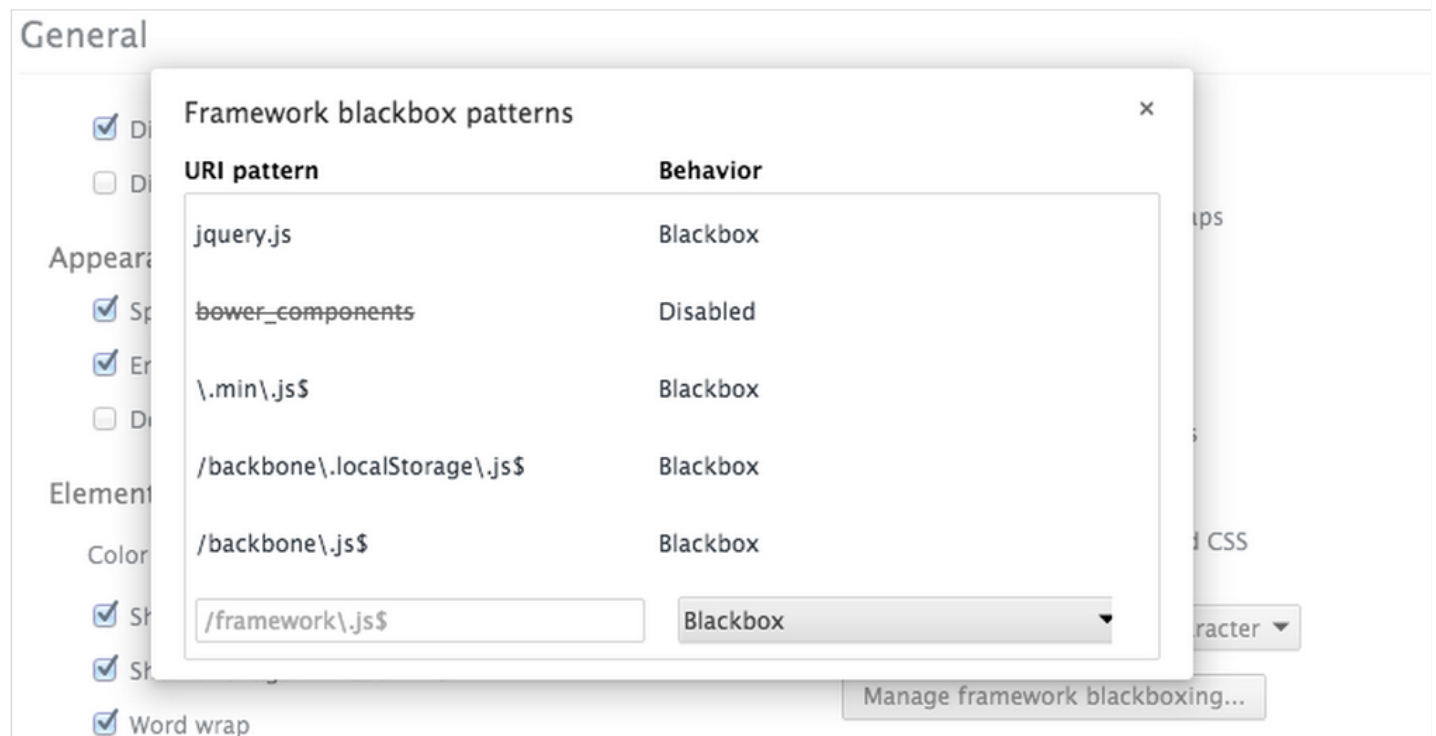
### Settings panel

Use the Settings panel to configure blackboxed scripts. Open the DevTools Settings and under Sources click **Manage framework blackboxing**.



This will open up a dialog where you can input file names of scripts that you would like to add to the blacklist. You can do this a couple of ways:

- enter the name of a file,
- use regular expressions to target:
  - files that contain a specific name `/backbone\.js$`,
  - certain types of files like `\.min\.js$`
- or enter in an entire folder that contains scripts you want to blackbox such as `bower_components`.



## Viewing data attributes in Chrome

Type into the chrome console:

```
console.log($('selector').data());
```

and it will list the `data` in that element