



Anatomy of a jQuery Command

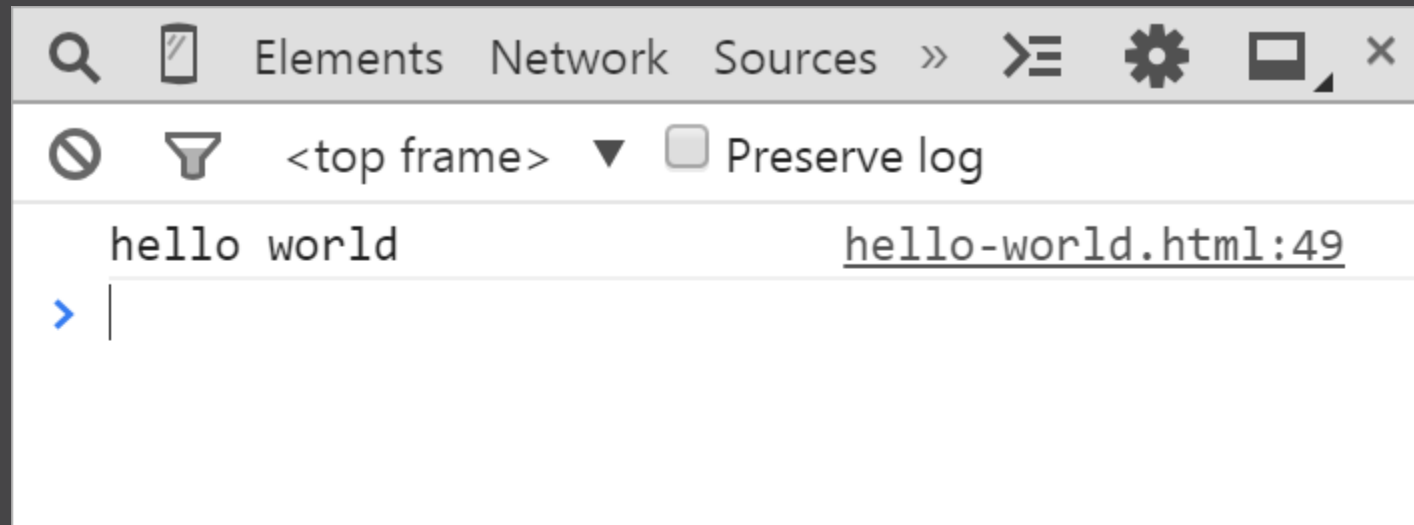


Craig Shoemaker

craigshoemaker.net | [@craigshoemaker](https://twitter.com/craigshoemaker)



```
$(function () {  
    console.log('hello world');  
});
```





```
$(function () {  
    console.log('hello world');  
});
```



```
$(function () {  
    console.log('hello world');  
});
```



```
$(function () {  
    console.log('hello world');  
});
```



```
$(function () {  
    console.log('hello world');  
});
```



```
$(function () {  
    console.log('hello world');  
});
```



```
$(function () {  
    console.log('hello world');  
});
```




```
jQuery(function () {  
    console.log('hello world');  
});
```



```
$(function () {  
    console.log('hello world');  
});
```

```
65
66 var
67     version = "1.11.2",
68
69     // Define a local copy of jQuery
70     jQuery = function( selector, context ) {
71         // The jQuery object is actually just the init constructor 'enhanced'
72         // Need init if jQuery is called (just allow error to be thrown if not included)
73         return new jQuery.fn.init( selector, context );
74     },
75
76     // Support: Android<4.1, IE<9
77     // Make sure we trim BOM and NBSP
78     rtrim = /^[\s\uFEFF\xA0]+|[\s\uFEFF\xA0]+$/g,
79
80     // Matches dashed string for camelizing
81     rmsPrefix = /^-ms-/,
82     rdashAlpha = /-([\da-z])/gi,
83
84     // Used by jQuery.camelCase as callback to replace()
85     fcamelCase = function( all, letter ) {
86         return letter.toUpperCase();
87     };
88
89
```

```
jquery-1.11.2.js* x
10324     window.$ = _$;
10325 }
10326
10327 if ( deep && window.jQuery === jQuery ) {
10328     window.jQuery = _jQuery;
10329 }
10330
10331 return jQuery;
10332 };
10333
10334 // Expose jQuery and $ identifiers, even
10335 // AMD (#7102#comment:10, https://github.com/jquery/jquery/pull/557)
10336 // and CommonJS for browser emulators (#13566)
10337 if ( !typeof noGlobal === 'undefined' ) {
10338     window.jQuery = window.$ = jQuery;
10339 }
10340
10341
10342
10343
10344 return jQuery;
10345
10346 }));
10347 |
```

\$ == jQuery



\$



`$();`



```
$(function () {  
  
});
```




```
$(function () {  
    console.log('hello world');  
});
```



```
$(function () {  
    // page setup  
});
```

jQuery ready

Demo

length/working with sets



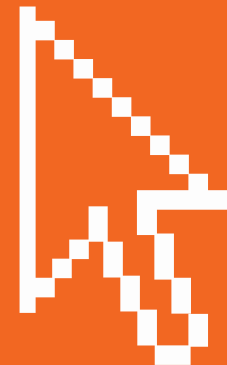
Demo

nomatch



Demo

getter/setter functions



Summary



Significance of \$



Anonymous Functions



Thinking in Sets



Finding Parts of a Page