11. Jquery noConflict() method.

Many JavaScript libraries use $ as a function or variable name, just as jQuery does. In jQuery's case, $ is just an alias for jQuery, so all functionality is available without using $. If you need to use another JavaScript library alongside jQuery, return control of $ back to the other library with a call to $.noConflict(). Old references of $ are saved during jQuery initialization;noConflict() simply restores them.

If for some reason two versions of jQuery are loaded (which is not recommended), calling $.noConflict( true ) from the second version will return the globally scoped jQuery variables to those of the first version.

|  |  |
| --- | --- |
|  | <script src="other\_lib.js"></script>  <script src="jquery.js"></script>  <script>  $.noConflict();  *// Code that uses other library's $ can follow here.*  </script> |

This technique is especially effective in conjunction with the .ready() method's ability to alias the jQuery object, as within callback passed to .ready() you can use $ if you wish without fear of conflicts later:

|  |  |
| --- | --- |
|  | <script src="other\_lib.js"></script>  <script src="jquery.js"></script>  <script>  $.noConflict();  jQuery( document ).ready(**function**( $ ) {  *// Code that uses jQuery's $ can follow here.*  });  *// Code that uses other library's $ can follow here.*  </script> |

If necessary, you can free up the jQuery name as well by passing true as an argument to the method. This is rarely necessary, and if you must do this (for example, if you need to use multiple versions of the jQuery library on the same page), you need to consider that most plug-ins rely on the presence of the jQuery variable and may not operate correctly in this situation.

**Examples:**

Map the original object that was referenced by $ back to $.

|  |  |
| --- | --- |
|  | jQuery.noConflict();  *// Do something with jQuery*  jQuery( "div p" ).hide();  *// Do something with another library's $()*  $( "content" ).style.display = "none"; |

Revert the $ alias and then create and execute a function to provide the $ as a jQuery alias inside the function's scope. Inside the function the original $ object is not available. This works well for most plugins that don't rely on any other library.

|  |  |
| --- | --- |
|  | jQuery.noConflict();  (**function**( $ ) {  $(**function**() {  *// More code using $ as alias to jQuery*  });  })(jQuery);    *// Other code using $ as an alias to the other library* |

Create a different alias instead of jQuery to use in the rest of the script.

|  |  |
| --- | --- |
|  | **var** j = jQuery.noConflict();    *// Do something with jQuery*  j( "div p" ).hide();    *// Do something with another library's $()*  $( "content" ).style.display = "none"; |

Completely move jQuery to a new namespace in another object.

|  |  |
| --- | --- |
|  | **var** dom = {};  dom.query = jQuery.noConflict( true ); |

**Result:**

|  |  |
| --- | --- |
|  | *// Do something with the new jQuery*  dom.query( "div p" ).hide();    *// Do something with another library's $()*  $( "content" ).style.display = "none";    *// Do something with another version of jQuery*  jQuery( "div > p" ).hide(); |

Load two versions of jQuery (not recommended). Then, restore jQuery's globally scoped variables to the first loaded jQuery.

|  |  |
| --- | --- |
|  | **<!doctype html>**  <html lang="en">  <head>  <meta charset="utf-8">  <title>jQuery.noConflict demo</title>  <script src="https://code.jquery.com/jquery-1.10.2.js"></script>  </head>  <body>    <div id="log">  <h3>Before $.noConflict(true)</h3>  </div>  <script src="https://code.jquery.com/jquery-1.6.2.js"></script>    <script>  **var** $log = $( "#log" );    $log.append( "2nd loaded jQuery version ($): " + $.fn.jquery + "<br>" );    *// Restore globally scoped jQuery variables to the first version loaded*  *// (the newer version)*    jq162 = jQuery.noConflict( true );    $log.append( "<h3>After $.noConflict(true)</h3>" );  $log.append( "1st loaded jQuery version ($): " + $.fn.jquery + "<br>" );  $log.append( "2nd loaded jQuery version (jq162): " + jq162.fn.jquery + "<br>" );  </script>    </body>  </html> |