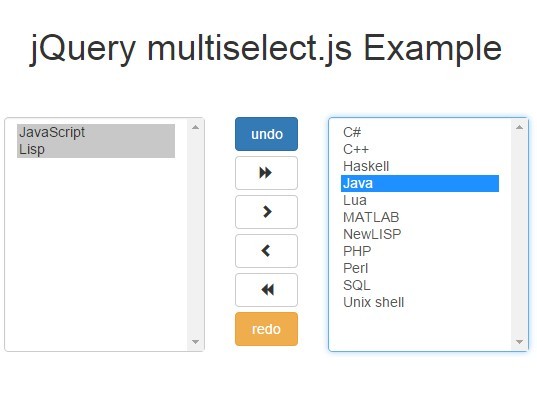
# [Two-side Multi Select Plugin with jQuery - multiselect.js](http://www.jqueryscript.net/form/Two-side-Multi-Select-Plugin-with-jQuery-multiselect-js.html" \o "Read 'Two-side Multi Select Plugin with jQuery - multiselect.js')

**[](http://www.jqueryscript.net/images/Two-side-Multi-Select-Plugin-with-jQuery-multiselect-js.jpg)**

|  |  |
| --- | --- |
|  |  |

http://www.jqueryscript.net/form/Two-side-Multi-Select-Plugin-with-jQuery-multiselect-js.html

multiselect.js is a jQuery plugin that converts a multiple select into two-side select boxes so the users can move items between two boxes by selecting items(s) and clicking on arrow buttons. Compatible with [Bootstrap](http://www.jqueryscript.net/tags.php?/Bootstrap/) 3 framework.

## See also:

* [Fully Customizable jQuery Select Element Plugin - Multiselect](http://www.jqueryscript.net/form/Fully-Customizable-jQuery-Select-Element-Plugin-Multiselect.html)

## How to use it:

1. Load Twitter's Bootstrap's stylesheet in the header.

|  |  |
| --- | --- |
| 1 | <link rel="stylesheet" href="//netdna.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css"> |

2. Load jQuery library, Bootstrap's JS and the jQuery multiselect.js script at the bottom of the html page.

|  |  |  |
| --- | --- | --- |
| 1 | | <script src="//code.jquery.com/jquery-1.11.2.min.js"></script> |
| 2 | <script src="//netdna.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js"></script> | | |

|  |  |
| --- | --- |
| 3 | <script src="js/multiselect.js"></script> |

3. Create a two sided select list as follow:

|  |  |
| --- | --- |
| 01 | <div class="row"> |
| 02 | <div class="col-xs-5"> | |

|  |  |  |
| --- | --- | --- |
| 03 | <select name="from" id="multiselect" class="form-control" size="8" multiple="multiple"> | |
| 04 | <option value="1">C++</option> |

|  |  |
| --- | --- |
| 05 | <option value="2">C#</option> |
| 06 | <option value="3">Haskell</option> | |

|  |  |
| --- | --- |
| 07 | <option value="4">Java</option> |
| 08 | <option value="5">JavaScript</option> | |

|  |  |  |
| --- | --- | --- |
| 09 | <option value="6">Lisp</option> | |
| 10 | <option value="7">Lua</option> |

|  |  |
| --- | --- |
| 11 | <option value="8">MATLAB</option> |
| 12 | <option value="9">NewLISP</option> | |

|  |  |
| --- | --- |
| 13 | <option value="10">PHP</option> |
| 14 | <option value="11">Perl</option> | |

|  |  |
| --- | --- |
| 15 | <option value="12">SQL</option> |
| 16 | <option value="13">Unix shell</option> | |

|  |  |  |
| --- | --- | --- |
| 17 | </select> | |
| 18 | </div> |

|  |  |
| --- | --- |
| 19 | <div class="col-xs-2"> |
| 20 | <button type="button" id="multiselect\_rightAll" class="btn btn-block"><i class="glyphicon glyphicon-forward"></i></button> | |

|  |  |
| --- | --- |
| 21 | <button type="button" id="multiselect\_rightSelected" class="btn btn-block"><i class="glyphicon glyphicon-chevron-right"></i></button> |
| 22 | <button type="button" id="multiselect\_leftSelected" class="btn btn-block"><i class="glyphicon glyphicon-chevron-left"></i></button> |

|  |  |  |
| --- | --- | --- |
| 23 | <button type="button" id="multiselect\_leftAll" class="btn btn-block"><i class="glyphicon glyphicon-backward"></i></button> | |
| 24 | </div> |

|  |  |
| --- | --- |
| 25 | <div class="col-xs-5"> |
| 26 | <select name="to" id="multiselect\_to" class="form-control"size="8" multiple="multiple"> | |

|  |  |  |
| --- | --- | --- |
| 27 | </select> | |
| 28 | </div> |

|  |  |
| --- | --- |
| 29 | </div> |

4. Initialize the plugin with default options.

|  |  |
| --- | --- |
| 1 | $('#multiselect').multiselect(); |

5. Plugin options available.

* right: The select where the options are moved to
* rightSelected: The button that moves selected options from left to right
* rightAll: multiselect\_id\_rightAll The button that moves all options from left to right
* leftSelected: The button that moves selected options from right to left
* leftAll: The button that moves all options from right to left
* undo: The button that triggers the undo action
* redo: The button that triggers the redo action
* moveUp: The button that move options from right side to the top
* moveDown: The button that move options from right side to the bottom
* startUp: remove from left all options that are present in right  Whatever you want to do with $left and $right elements when the multiselect plugin is initialised
* sort: Will sort options when an action happend into right or left elements.
* beforeMoveToRight: Whatever you want to do with $left, $right and option[s] elements before they are moved to right. Keep in mind that this function must return a boolean value. true will let the action to be performed. false will stop the action
* moveToRight: If you want to define your own moveToRight functionality you can do this by defining moveToRight: function(Multiselect, options, event, silent, skipStack) { ... }. options - HTML element containing all selected options. event - the event that initialised the action. silent - boolean that tells if you have to trigger beforeMoveToRight and afterMoveToRight. skipStack - boolean that tells if you have to handle undo/redo stack
* afterMoveToRight: Whatever you want to do with $left, $right and option[s] elements after they are moved to right.
* beforeMoveToLeft: Whatever you want to do with $left, $right and option[s] elements before they are moved to left. Keep in mind that this function must return a boolean value. true will let the action to be performed. false will stop the action
* moveToLeft: If you want to define your own moveToLeft functionality you can do this by defining moveToLeft: function(Multiselect, options, event, silent, skipStack) { ... }.. options - HTML element containing all selected options. event - the event that initialised the action. silent - boolean that tells if you have to trigger beforeMoveToLeft and afterMoveToLeft. skipStack - boolean that tells if you have to handle undo/redo stack
* afterMoveToLeft: Whatever you want to do with $left, $right and option[s] elements after they are moved to left.
* keepRenderingSort: When you want to keep options sorted as they was rendered, keepRenderingSort must be true. When keepRenderingSort is true, sort function will be ignored.
* submitAllLeft: When you don’t want to send all options from the left side to server, submitAllLeft must be false.
* submitAllRight: When you don’t want to send all options from the right side to server, submitAllRight must be false.
* search: When you want to have search fields for left and right elements
* fireSearch: Tell to multiselect when to start applying the search.