

# Draggable Widget

Categories: [Interactions \(https://api.jqueryui.com/1.8/category/interactions/\)](https://api.jqueryui.com/1.8/category/interactions/)

## Draggable Widget

version added: 1.0

**Description:** *Allow elements to be moved using the mouse.*

### QuickNav

### Examples

#### Options

#### Methods

#### Events

[addClasses](#)

[appendTo](#)

[axis](#)

[cancel](#)

[connectToSortable](#)

[containment](#)

[cursor](#)

[cursorAt](#)

[delay](#)

[disabled](#)

[distance](#)

[grid](#)

[handle](#)

[helper](#)

[iframeFix](#)

[opacity](#)

[refreshPositions](#)

[revert](#)

[revertDuration](#)

[scope](#)

[scroll](#)

[scrollSensitivity](#)

[scrollSpeed](#)

[snap](#)

[snapMode](#)

[snapTolerance](#)

[stack](#)

[zIndex](#)

[destroy](#)

[disable](#)

[enable](#)

[option](#)

[widget](#)

[create](#)

[drag](#)

[start](#)

[stop](#)

Make the selected elements draggable by mouse. If you want not just drag, but drag & drop, see the [jQuery UI Droppable plugin \(/droppable/\)](#), which provides a drop target for draggables.

## Options

### addClasses

**Type:** [Boolean](http://api.jquery.com/Types/#Boolean) (<http://api.jquery.com/Types/#Boolean>)

**Default:** true

If set to `false`, will prevent the `ui-draggable` class from being added. This may be desired as a performance optimization when calling `.draggable()` on hundreds of elements.

### appendTo

**Type:** [jQuery](http://api.jquery.com/Types/#jQuery) (<http://api.jquery.com/Types/#jQuery>) or [Element](http://api.jquery.com/Types/#Element) (<http://api.jquery.com/Types/#Element>) or [Selector](http://api.jquery.com/Types/#Selector) (<http://api.jquery.com/Types/#Selector>) or [String](http://api.jquery.com/Types/#String) (<http://api.jquery.com/Types/#String>)

**Default:** "parent"

Which element the draggable helper should be appended to while dragging.

#### Multiple types supported:

**jQuery:** A jQuery object containing the element to append the helper to.

**Element:** The element to append the helper to.

**Selector:** A selector specifying which element to append the helper to.

**String:** The string "parent" will cause the helper to be a sibling of the draggable.

### axis

**Type:** [String](http://api.jquery.com/Types/#String) (<http://api.jquery.com/Types/#String>)

**Default:** false

Constrains dragging to either the horizontal (x) or vertical (y) axis. Possible values: "x", "y".

### cancel

**Type:** [Selector](http://api.jquery.com/Types/#Selector) (<http://api.jquery.com/Types/#Selector>)

**Default:** "input,textarea,button,select,option"

Prevents dragging from starting on specified elements.

### connectToSortable

**Type:** [Selector \(http://api.jquery.com/Types/#Selector\)](http://api.jquery.com/Types/#Selector)

**Default:** false

Allows the draggable to be dropped onto the specified sortables. If this option is used, a draggable can be dropped onto a sortable list and then becomes part of it. Note: The `helper` option must be set to "clone" in order to work flawlessly. Requires the [jQuery UI Sortable plugin \(/sortable/\)](#) to be included.

---

## containment

**Type:** [Selector \(http://api.jquery.com/Types/#Selector\)](http://api.jquery.com/Types/#Selector) or [Element \(http://api.jquery.com/Types/#Element\)](http://api.jquery.com/Types/#Element) or [String \(http://api.jquery.com/Types/#String\)](http://api.jquery.com/Types/#String) or [Array \(http://api.jquery.com/Types/#Array\)](http://api.jquery.com/Types/#Array)

**Default:** false

Constrains dragging to within the bounds of the specified element or region.

### Multiple types supported:

**Selector:** The draggable element will be contained to the bounding box of the first element found by the selector. If no element is found, no containment will be set.

**Element:** The draggable element will be contained to the bounding box of this element.

**String:** Possible values: "parent", "document", "window".

**Array:** An array defining a bounding box in the form [ x1, y1, x2, y2 ].

---

## cursor

**Type:** [String \(http://api.jquery.com/Types/#String\)](http://api.jquery.com/Types/#String)

**Default:** "auto"

The CSS cursor during the drag operation.

---

## cursorAt

**Type:** [Object \(http://api.jquery.com/Types/#Object\)](http://api.jquery.com/Types/#Object)

**Default:** false

Sets the offset of the dragging helper relative to the mouse cursor. Coordinates can be given as a hash using a combination of one or two keys: { top, left, right, bottom }.

---

## delay

**Type:** [Number \(http://api.jquery.com/Types/#Number\)](http://api.jquery.com/Types/#Number)

**Default:** 0

Time in milliseconds after mousedown until dragging should start. This option can be used to

prevent unwanted drags when clicking on an element.

---

## disabled

**Type:** Boolean (<http://api.jquery.com/Types/#Boolean>)

**Default:** false

Disables the draggable if set to true .

---

## distance

**Type:** Number (<http://api.jquery.com/Types/#Number>)

**Default:** 1

Distance in pixels after mousedown the mouse must move before dragging should start. This option can be used to prevent unwanted drags when clicking on an element.

---

## grid

**Type:** Array (<http://api.jquery.com/Types/#Array>)

**Default:** false

Snaps the dragging helper to a grid, every x and y pixels. The array must be of the form [ x, y ] .

---

## handle

**Type:** Selector (<http://api.jquery.com/Types/#Selector>) or Element (<http://api.jquery.com/Types/#Element>)

**Default:** false

If specified, restricts dragging from starting unless the mousedown occurs on the specified element(s).

---

## helper

**Type:** String (<http://api.jquery.com/Types/#String>) or Function (<http://api.jquery.com/Types/#Function>)()

**Default:** "original"

Allows for a helper element to be used for dragging display.

### Multiple types supported:

**String:** If set to "clone" , then the element will be cloned and the clone will be dragged.

**Function:** A function that will return a DOMElement to use while dragging.

---

## iframeFix

**Type:** Boolean (<http://api.jquery.com/Types/#Boolean>) or Selector (<http://api.jquery.com/Types/#Selector>)

**Default:** false

Prevent iframes from capturing the mousemove events during a drag. Useful in combination with the cursorAt option, or in any case where the mouse cursor may not be over the helper.

### Multiple types supported:

**Boolean:** When set to `true`, transparent overlays will be placed over all iframes on the page.

**Selector:** Any iframes matching the selector will be covered by transparent overlays.

---

## opacity

**Type:** Number (<http://api.jquery.com/Types/#Number>)

**Default:** false

Opacity for the helper while being dragged.

---

## refreshPositions

**Type:** Boolean (<http://api.jquery.com/Types/#Boolean>)

**Default:** false

If set to `true`, all droppable positions are calculated on every mousemove. *Caution: This solves issues on highly dynamic pages, but dramatically decreases performance.*

---

## revert

**Type:** Boolean (<http://api.jquery.com/Types/#Boolean>) or String (<http://api.jquery.com/Types/#String>)

**Default:** false

Whether the element should revert to its start position when dragging stops.

### Multiple types supported:

**Boolean:** If set to `true` the element will always revert.

**String:** If set to `"invalid"`, revert will only occur if the draggable has not been dropped on a droppable. For `"valid"`, it's the other way around.

## revertDuration

**Type:** Number (<http://api.jquery.com/Types/#Number>)

**Default:** 500

The duration of the revert animation, in milliseconds. Ignored if the revert option is `false`.

---

## scope

**Type:** String (<http://api.jquery.com/Types/#String>)

**Default:** "default"

Used to group sets of draggable and droppable items, in addition to droppable's accept ([/droppable/#option-accept](#)) option. A draggable with the same `scope` value as a droppable will be accepted by the droppable.

---

## scroll

**Type:** Boolean (<http://api.jquery.com/Types/#Boolean>)

**Default:** `true`

If set to `true`, container auto-scrolls while dragging.

---

## scrollSensitivity

**Type:** Number (<http://api.jquery.com/Types/#Number>)

**Default:** 20

Distance in pixels from the edge of the viewport after which the viewport should scroll. Distance is relative to pointer, not the draggable. Ignored if the scroll option is `false`.

---

## scrollSpeed

**Type:** Number (<http://api.jquery.com/Types/#Number>)

**Default:** 20

The speed at which the window should scroll once the mouse pointer gets within the scrollSensitivity distance. Ignored if the scroll option is `false`.

---

## snap

**Type:** Boolean (<http://api.jquery.com/Types/#Boolean>) or Selector (<http://api.jquery.com/Types/#Selector>)

**Default:** `false`

Whether the element should snap to other elements.

## Multiple types supported:

**Boolean:** When set to `true`, the element will snap to all other draggable elements.

**Selector:** A selector specifying which elements to snap to.

---

## snapMode

**Type:** String (<http://api.jquery.com/Types/#String>)

**Default:** `"both"`

Determines which edges of snap elements the draggable will snap to. Ignored if the `snap` option is `false`. Possible values: `"inner"`, `"outer"`, `"both"`.

---

## snapTolerance

**Type:** Number (<http://api.jquery.com/Types/#Number>)

**Default:** `20`

The distance in pixels from the snap element edges at which snapping should occur. Ignored if the `snap` option is `false`.

---

## stack

**Type:** Selector (<http://api.jquery.com/Types/#Selector>)

**Default:** `false`

Controls the z-index of the set of elements that match the selector, always brings the currently dragged item to the front. Very useful in things like window managers.

---

## zIndex

**Type:** Number (<http://api.jquery.com/Types/#Number>)

**Default:** `false`

Z-index for the helper while being dragged.

---

## Methods

### destroy()

**Returns:** jQuery (<http://api.jquery.com/Types/#jQuery>) (*plugin only* (<http://learn.jquery.com/jquery-ui/widget-factory/widget-method-invocation/>))

Removes the draggable functionality completely. This will return the element back to its pre-init state.

This method does not accept any arguments.

### Code examples:

Invoke the destroy method:

```
1 $( ".selector" ).draggable( "destroy" );
```

---

### **disable()**

**Returns:** *jQuery* (<http://api.jquery.com/Types/#jQuery>) (plugin only  
(<http://learn.jquery.com/jquery-ui/widget-factory/widget-method-invocation/>))

Disables the draggable.

This method does not accept any arguments.

### Code examples:

Invoke the disable method:

```
1 $( ".selector" ).draggable( "disable" );
```

---

### **enable()**

**Returns:** *jQuery* (<http://api.jquery.com/Types/#jQuery>) (plugin only  
(<http://learn.jquery.com/jquery-ui/widget-factory/widget-method-invocation/>))

Enables the draggable.

This method does not accept any arguments.

### Code examples:

Invoke the enable method:

```
1 $( ".selector" ).draggable( "enable" );
```

---

### **option( optionName )**

**Returns:** *Object* (<http://api.jquery.com/Types/#Object>)

Gets the value currently associated with the specified `optionName`.

**optionName**

Type: *String* (<http://api.jquery.com/Types/#String>)

The name of the option to get.

### Code examples:

Invoke the method:

```
1 $( ".selector" ).draggable( "option" );
```



---

## **option()**

**Returns:** *PlainObject* (<http://api.jquery.com/Types/#PlainObject>)

Gets an object containing key/value pairs representing the current draggable options hash.

This signature does not accept any arguments.

### **Code examples:**

Invoke the method:

```
1 $( ".selector" ).draggable( "option" );
```

---

## **option( optionName, value )**

**Returns:** *jQuery* (<http://api.jquery.com/Types/#jQuery>) (plugin only  
(<http://learn.jquery.com/jquery-ui/widget-factory/widget-method-invocation/>))

Sets the value of the draggable option associated with the specified `optionName`.

### **optionName**

Type: String (<http://api.jquery.com/Types/#String>)

The name of the option to set.

### **value**

Type: Object (<http://api.jquery.com/Types/#Object>)

A value to set for the option.

### **Code examples:**

Invoke the method:

```
1 $( ".selector" ).draggable( "option" );
```

---

## **option( options )**

**Returns:** *jQuery* (<http://api.jquery.com/Types/#jQuery>) (plugin only  
(<http://learn.jquery.com/jquery-ui/widget-factory/widget-method-invocation/>))

Sets one or more options for the draggable.

### **options**

Type: Object (<http://api.jquery.com/Types/#Object>)

A map of option-value pairs to set.

### **Code examples:**

Invoke the method:

```
1 $( ".selector" ).draggable( "option" );
```

## widget()

**Returns:** *jQuery (<http://api.jquery.com/Types/#jQuery>)*

Returns a jQuery object containing the draggable element.

This method does not accept any arguments.

### Code examples:

Invoke the widget method:

```
1 $( ".selector" ).draggable( "widget" );
```

## Events

### create( event, ui )

**Type:** *draggablecreate*

Triggered when the draggable is created.

**event**

Type: [Event](http://api.jquery.com/Types/#Event) (<http://api.jquery.com/Types/#Event>)

**ui**

Type: [Object](http://api.jquery.com/Types/#Object) (<http://api.jquery.com/Types/#Object>)

*Note: The ui object is empty but included for consistency with other events.*

### Code examples:

Initialize the draggable with the create callback specified:

```
1 $( ".selector" ).draggable({  
2   create: function( event, ui ) {}  
3 });
```

Bind an event listener to the draggablecreate event:

```
1 $( ".selector" ).on( "draggablecreate", function( event, ui )
```



### drag( event, ui )

**Type:** *draggabledrag*

Triggered while the mouse is moved during the dragging.

**event**

Type: [Event](http://api.jquery.com/Types/#Event) (<http://api.jquery.com/Types/#Event>)

## ui

Type: [Object \(http://api.jquery.com/Types/#Object\)](http://api.jquery.com/Types/#Object)

### helper

Type: [jQuery \(http://api.jquery.com/Types/#jQuery\)](http://api.jquery.com/Types/#jQuery)

The jQuery object representing the helper that's being dragged.

### position

Type: [Object \(http://api.jquery.com/Types/#Object\)](http://api.jquery.com/Types/#Object)

Current CSS position of the helper as { top, left } object.

### offset

Type: [Object \(http://api.jquery.com/Types/#Object\)](http://api.jquery.com/Types/#Object)

Current offset position of the helper as { top, left } object.

## Code examples:

Initialize the draggable with the drag callback specified:

```
1 $( ".selector" ).draggable({  
2   drag: function( event, ui ) {}  
3 });
```

Bind an event listener to the draggabledrag event:

```
1 $( ".selector" ).on( "draggabledrag", function( event, ui ) {}
```



---

## start( event, ui )

**Type:** *draggablestart*

Triggered when dragging starts.

### event

Type: [Event \(http://api.jquery.com/Types/#Event\)](http://api.jquery.com/Types/#Event)

## ui

Type: [Object \(http://api.jquery.com/Types/#Object\)](http://api.jquery.com/Types/#Object)

### helper

Type: [jQuery \(http://api.jquery.com/Types/#jQuery\)](http://api.jquery.com/Types/#jQuery)

The jQuery object representing the helper that's being dragged.

### position

Type: [Object \(http://api.jquery.com/Types/#Object\)](http://api.jquery.com/Types/#Object)

Current CSS position of the helper as { top, left } object.

### offset

Type: [Object](http://api.jquery.com/Types/#Object) (<http://api.jquery.com/Types/#Object>)

Current offset position of the helper as { top, left } object.


### Code examples:

Initialize the draggable with the start callback specified:

```
1 $( ".selector" ).draggable({  
2   start: function( event, ui ) {}  
3 });
```

Bind an event listener to the draggablestart event:

```
1 $( ".selector" ).on( "draggablestart", function( event, ui ) {
```



---

## stop( event, ui )

**Type:** *draggablestop*

Triggered when dragging stops.

### event

Type: [Event](http://api.jquery.com/Types/#Event) (<http://api.jquery.com/Types/#Event>)

### ui

Type: [Object](http://api.jquery.com/Types/#Object) (<http://api.jquery.com/Types/#Object>)

### helper

Type: [jQuery](http://api.jquery.com/Types/#jQuery) (<http://api.jquery.com/Types/#jQuery>)

The jQuery object representing the helper that's being dragged.

### position

Type: [Object](http://api.jquery.com/Types/#Object) (<http://api.jquery.com/Types/#Object>)

Current CSS position of the helper as { top, left } object.

### offset

Type: [Object](http://api.jquery.com/Types/#Object) (<http://api.jquery.com/Types/#Object>)

Current offset position of the helper as { top, left } object.

### Code examples:

Initialize the draggable with the stop callback specified:

```
1 $( ".selector" ).draggable({  
2   stop: function( event, ui ) {}  
3 });
```

Bind an event listener to the draggablestop event:

```
1 $( ".selector" ).on( "draggablestop", function( event, ui ) {}
```

## Example:

A simple jQuery UI Draggable

```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <meta charset="utf-8">
5   <title>draggable demo</title>
6   <link rel="stylesheet" href="//code.jquery.com/ui/1.8.24/themes/base
7   <style>
8     #draggable {
9       width: 100px;
10      height: 100px;
11      background: #ccc;
12    }
13  </style>
14  <script src="//code.jquery.com/jquery-1.8.2.js"></script>
15  <script src="//code.jquery.com/ui/1.8.24/jquery-ui.js"></script>
16 </head>
17 <body>
18
19 <div id="draggable">Drag me</div>
20
21 <script>
22 $( "#draggable" ).draggable();
23 </script>
24
25 </body>
26 </html>
```

## Demo:

Drag me

