

# Dialog Widget

Categories: [Widgets \(http://api.jqueryui.com/1.8/category/widgets/\)](http://api.jqueryui.com/1.8/category/widgets/)

## Dialog Widget

version added: 1.0

**Description:** *Open content in an interactive overlay.*

### QuickNav

### Examples

#### Options

[autoOpen](#)  
[buttons](#)  
[closeOnEscape](#)  
[closeText](#)  
[dialogClass](#)  
[disabled](#)  
[draggable](#)  
[height](#)  
[hide](#)  
[maxHeight](#)  
[maxWidth](#)  
[minHeight](#)  
[minWidth](#)  
[modal](#)  
[position](#)  
[resizable](#)  
[show](#)  
[stack](#)  
[title](#)  
[width](#)  
[zIndex](#)

#### Methods

[close](#)  
[destroy](#)  
[disable](#)  
[enable](#)  
[isOpen](#)  
[moveToTop](#)  
[open](#)  
[option](#)  
[widget](#)

#### Events

[beforeClose](#)  
[close](#)  
[create](#)  
[drag](#)  
[dragStart](#)  
[dragStop](#)  
[focus](#)  
[open](#)  
[resize](#)  
[resizeStart](#)  
[resizeStop](#)

A dialog is a floating window that contains a title bar and a content area. The dialog window can be moved, resized and closed with the 'x' icon by default.

If the content length exceeds the maximum height, a scrollbar will automatically appear.

A bottom button bar and semi-transparent modal overlay layer are common options that can be added.

## Additional Notes:

This widget requires some functional CSS, otherwise it won't work. If you build a custom theme, use the widget's specific CSS file as a starting point.

## Options

### autoOpen

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

**Default:** `true`

If set to `true`, the dialog will automatically open upon initialization. If `false`, the dialog will stay hidden until the `open()` method is called.

### buttons

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

**Default:** `{}`

Specifies which buttons should be displayed on the dialog. The context of the callback is the dialog element; if you need access to the button, it is available as the target of the event object.

#### Multiple types supported:

**Object:** The keys are the button labels and the values are the callbacks for when the associated button is clicked.

**Array:** Each element of the array must be an object defining the attributes, properties, and event handlers to set on the button.

### closeOnEscape

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

**Default:** `true`

Specifies whether the dialog should close when it has focus and the user presses the escape (ESC) key.

### closeText

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

**Default:** `"close"`

Specifies the text for the close button. Note that the close text is visibly hidden when using a standard theme.

## dialogClass

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

**Default:** ""

The specified class name(s) will be added to the dialog, for additional theming.

---

## disabled

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

**Default:** false

Disables the dialog if set to true .

---

## draggable

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

**Default:** true

If set to true , the dialog will be draggable by the title bar. Requires the [jQuery UI Draggable widget \(/draggable/\)](#) to be included.

---

## height

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

**Default:** "auto"

The height of the dialog.

### Multiple types supported:

**Number:** The height in pixels.

**String:** The only supported string value is "auto" which will allow the dialog height to adjust based on its content.

---

## hide

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

**Default:** null

If and how to animate the hiding of the dialog.

### Multiple types supported:

**Number:** The dialog will fade out while animating the height and width for the specified duration.

**String:** The dialog will be hidden using the specified jQuery UI effect. See the [list of effects \(/category/effects/\)](#) for possible values.

**Object:** If the value is an object, then `effect`, `duration`, and `easing` properties may be provided. The `effect` property must be the name of a jQuery UI effect. When using a jQuery UI effect that supports additional settings, you may include those settings in the object and they will be passed to the effect. If `duration` or `easing` is omitted, then the default values will be used.

---

## maxHeight

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

**Default:** `false`

The maximum height to which the dialog can be resized, in pixels.

---

## maxWidth

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

**Default:** `false`

The maximum width to which the dialog can be resized, in pixels.

---

## minHeight

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

**Default:** `150`

The minimum height to which the dialog can be resized, in pixels.

---

## minWidth

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

**Default:** `150`

The minimum width to which the dialog can be resized, in pixels.

---

## modal

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

**Default:** `false`

If set to `true`, the dialog will have modal behavior; other items on the page will be disabled, i.e., cannot be interacted with. Modal dialogs create an overlay below the dialog but above other page elements.

---

## position

**Type:** [Object \(http://api.jquery.com/Types/#Object\)](http://api.jquery.com/Types/#Object) 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:** { my: "center", at: "center", of: window }

Specifies where the dialog should be displayed. The dialog will handle collisions such that as much of the dialog is visible as possible.

### Multiple types supported:

**Object:** Identifies the position of the dialog when opened. The `of` option defaults to the window, but you can specify another element to position against. You can refer to the [jQuery UI Position \(/position/\)](#) utility for more details about the various options.

**String:** A string representing the position within the viewport. Possible values: "center", "left", "right", "top", "bottom".

**Array:** An array containing an *x, y* coordinate pair in pixel offset from the top left corner of the viewport or the name of a possible string value.

---

## resizable

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

**Default:** true

If set to `true`, the dialog will be resizable. Requires the [jQuery UI Resizable widget \(/resizable/\)](#) to be included.

---

## show

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

**Default:** null

If and how to animate the showing of the dialog.

### Multiple types supported:

**Number:** The dialog will fade in while animating the height and width for the specified duration.

**String:** The dialog will be shown using the specified jQuery UI effect. See the [list of effects \(/category/effects/\)](#) for possible values.

**Object:** If the value is an object, then `effect`, `duration`, and `easing` properties may be provided. The `effect` property must be the name of a jQuery UI effect. When using a jQuery UI effect that supports additional settings, you may include those settings in the object and they will be passed to the effect. If `duration` or `easing` is omitted, then the default values will be used.

## stack

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

**Default:** `true`

Specifies whether the dialog will stack on top of other dialogs. This will cause the dialog to move to the front of other dialogs when it gains focus.

## title

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

**Default:** `""`

Specifies the title of the dialog. Any valid HTML may be set as the title. The title can also be specified by the `title` attribute on the dialog source element.

## width

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

**Default:** `300`

The width of the dialog, in pixels.

## zIndex

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

**Default:** `1000`

The starting z-index for the dialog.

## Methods

### close()

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

Closes the dialog.

This method does not accept any arguments.

### Code examples:

Invoke the close method:

```
1 $( ".selector" ).dialog( "close" );
```

## destroy()

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

Removes the dialog 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" ).dialog( "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 dialog.

This method does not accept any arguments.

### Code examples:

Invoke the disable method:

```
1 $( ".selector" ).dialog( "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 dialog.

This method does not accept any arguments.

### Code examples:

Invoke the enable method:

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

---

## isOpen()

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

Whether the dialog is currently open.

This method does not accept any arguments.

### Code examples:

Invoke the isOpen method:

```
1 $( ".selector" ).dialog( "isOpen" );
```

---

### moveToTop()

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

Moves the dialog to the top of the dialog stack.

This method does not accept any arguments.

### Code examples:

Invoke the moveToTop method:

```
1 $( ".selector" ).dialog( "moveToTop" );
```

---

### open()

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

Opens the dialog.

This method does not accept any arguments.

### Code examples:

Invoke the open method:

```
1 $( ".selector" ).dialog( "open" );
```

---

### 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" ).dialog( "option" );
```



## **option()**

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

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

This signature does not accept any arguments.

### **Code examples:**

Invoke the method:

```
1 $( ".selector" ).dialog( "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 dialog 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" ).dialog( "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 dialog.

### **options**

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

A map of option-value pairs to set.

### **Code examples:**

Invoke the method:

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

## widget()

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

Returns a jQuery object containing the generated wrapper.

This method does not accept any arguments.

### Code examples:

Invoke the widget method:

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

## Events

### beforeClose( event, ui )

**Type:** *dialogbeforeclose*

Triggered when a dialog is about to close. If canceled, the dialog will not close.

#### 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 dialog with the beforeClose callback specified:

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

Bind an event listener to the dialogbeforeclose event:

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



### close( event, ui )

**Type:** *dialogclose*

Triggered when the dialog is closed.

#### 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 dialog with the close callback specified:

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

Bind an event listener to the dialogclose event:

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

---

## **create( event, ui )**

**Type:** *dialogcreate*

Triggered when the dialog 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 dialog with the create callback specified:

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

Bind an event listener to the dialogcreate event:

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



---

## **drag( event, ui )**

**Type:** *dialogdrag*

Triggered while the dialog is being dragged.

**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)

##### position

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

The current CSS position of the dialog.

##### offset

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

The current offset position of the dialog.

### Code examples:

Initialize the dialog with the drag callback specified:

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

Bind an event listener to the dialogdrag event:

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

---

## dragStart( event, ui )

**Type:** *dialogdragstart*

Triggered when the user starts dragging the dialog.

#### 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)

##### position

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

The current CSS position of the dialog.

##### offset

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

The current offset position of the dialog.

### Code examples:

Initialize the dialog with the dragStart callback specified:

```
2 $( ".selector" ).dialog({
3   dragStart: function( event, ui ) {}
  });
```

Bind an event listener to the dialogdragstart event:

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



---

## dragStop( event, ui )

**Type:** *dialogdragstop*

Triggered after the dialog has been dragged.

### 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)

#### position

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

The current CSS position of the dialog.

#### offset

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

The current offset position of the dialog.


## Code examples:

Initialize the dialog with the dragStop callback specified:

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

Bind an event listener to the dialogdragstop event:

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



---

## focus( event, ui )

**Type:** *dialogfocus*

Triggered when the dialog gains focus.

### 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 dialog with the focus callback specified:

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

Bind an event listener to the dialogfocus event:

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

---

## **open( event, ui )**

**Type:** *dialogopen*

Triggered when the dialog is opened.

**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 dialog with the open callback specified:

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

Bind an event listener to the dialogopen event:

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

---

## **resize( event, ui )**

**Type:** *dialogresize*

Triggered while the dialog is being resized.

**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)

**originalPosition**

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

The CSS position of the dialog prior to being resized.

**position**

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

The current CSS position of the dialog.

**originalSize**

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

The size of the dialog prior to being resized.

**size**

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

The current size of the dialog.

### Code examples:

Initialize the dialog with the resize callback specified:

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

Bind an event listener to the dialogresize event:

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



---

## resizeStart( event, ui )

**Type:** *dialogresizestart*

Triggered when the user starts resizing the dialog.

**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)

**originalPosition**

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

The CSS position of the dialog prior to being resized.

**position**

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

The current CSS position of the dialog.

**originalSize**

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

The size of the dialog prior to being resized.

**size**

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

The current size of the dialog.

**Code examples:**

Initialize the dialog with the `resizeStart` callback specified:

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

Bind an event listener to the `dialogresizestart` event:

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



---

**resizeStop( event, ui )**

**Type:** *dialogresizestop*

Triggered after the dialog has been resized.

**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)

**originalPosition**

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

The CSS position of the dialog prior to being resized.

**position**

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

The current CSS position of the dialog.

**originalSize**

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

The size of the dialog prior to being resized.

**size**

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



The current size of the dialog.

### Code examples:

Initialize the dialog with the `resizeStop` callback specified:

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

Bind an event listener to the `dialogresizestop` event:

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



## Example:

A simple jQuery UI Dialog

```
1 <!doctype html>  
2 <html lang="en">  
3 <head>  
4   <meta charset="utf-8">  
5   <title>dialog demo</title>  
6   <link rel="stylesheet" href="//code.jquery.com/ui/1.8.24/themes/base/j<br>  
7   <script src="//code.jquery.com/jquery-1.8.2.js"></script>  
8   <script src="//code.jquery.com/ui/1.8.24/jquery-ui.js"></script>  
9 </head>  
10 <body>  
11  
12 <button id="opener">open the dialog</button>  
13 <div id="dialog" title="Dialog Title">I'm a dialog</div>  
14  
15 <script>  
16 $( "#dialog" ).dialog({ autoOpen: false });  
17 $( "#opener" ).click(function() {  
18   $( "#dialog" ).dialog( "open" );  
19 });  
20 </script>  
21  
22 </body>  
23 </html>
```



**Demo:**

open the dialog

## BOOKS



### jQuery UI in Action

TJ VanToll

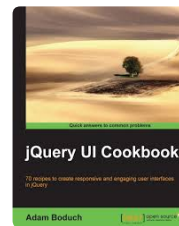
(<http://www.manning.com/tjvntoll/>)



### jQuery UI Themes

Adam Boduch

(<http://www.packtpub.com/web-development/jquery-ui-themes-beginners-guide>)



### jQuery UI Cookbook

Adam Boduch

(<http://www.packtpub.com/web-development/jquery-ui-cookbook>)

Copyright 2015 The jQuery Foundation  
(<https://jquery.org/team/>). jQuery License  
(<https://jquery.org/license/>)

Web hosting by Media Temple (<http://mediatemple.net/>) |

CDN by MaxCDN (<http://www.maxcdn.com/>) |

Powered by WordPress (<http://wordpress.org/>) | Thanks:

Members (<https://jquery.org/members/>), Sponsors  
(<https://jquery.org/sponsors/>)



Learning Center (<http://learn.jquery.com/>)



Forum (<http://forum.jquery.com/using-jquery-ui/>)



API (<http://api.jqueryui.com/>)



Twitter (<https://twitter.com/jqueryui>)



IRC (<http://irc.jquery.org/>)



GitHub (<https://github.com/jquery>)