* Overview
* Options
* Events
* Methods
* Theming

jQuery UI Button

## Overview

Button enhances standard form elements like button, input of type submit or reset or anchors to themable buttons with appropiate mouseover and active styles.

In addition to basic push buttons, radio buttons and checkboxes (inputs of type radio and checkbox) can be converted to buttons: Their associated label is styled to appear as the button, while the underlying input is updated on click.

In order to group radio buttons, Button also provides an additional widget-method, called Buttonset. Its used by selecting a container element (which contains the radio buttons) and calling buttonset(). Buttonset will also provide visual grouping, and therefore should be used whenever you have a group of buttons. It works by selecting all descendents and applying button() to them. You can enable and disable a buttonset, which will enable and disable all contained buttons. Destroying a buttonset also calls the button's destroy method.

When using an input of type button, submit or reset, support is limited to plain text labels with no icons.

### Dependencies

* UI Core
* UI Widget

### Example

* Demo
* View Source

A simple jQuery UI Button.

$("button").button();

<!DOCTYPE html>  
<html>  
<head>  
 <link href="http://ajax.googleapis.com/ajax/libs/jqueryui/1.8/themes/base/jquery-ui.css" rel="stylesheet" type="text/css"/>  
 <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.5/jquery.min.js"></script>  
 <script src="http://ajax.googleapis.com/ajax/libs/jqueryui/1.8/jquery-ui.min.js"></script>  
   
 <script>  
 $(document).ready(function() {  
 $("button").button();  
 });  
 </script>  
</head>  
<body style="font-size:62.5%;">  
   
<button>Button label</button>  
  
</body>  
</html>

* Demo
* View Source

A simple jQuery UI Button.

$("#radio").buttonset();

<!DOCTYPE html>  
<html>  
<head>  
 <link href="http://ajax.googleapis.com/ajax/libs/jqueryui/1.8/themes/base/jquery-ui.css" rel="stylesheet" type="text/css"/>  
 <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.5/jquery.min.js"></script>  
 <script src="http://ajax.googleapis.com/ajax/libs/jqueryui/1.8/jquery-ui.min.js"></script>  
   
 <script>  
 $(document).ready(function() {  
 $("#radio").buttonset();  
 });  
 </script>  
</head>  
<body style="font-size:62.5%;">  
   
<div id="radio">  
 <input type="radio" id="radio1" name="radio" /><label for="radio1">Choice 1</label>  
 <input type="radio" id="radio2" name="radio" checked="checked" /><label for="radio2">Choice 2</label>  
 <input type="radio" id="radio3" name="radio" /><label for="radio3">Choice 3</label>  
</div>  
  
</body>  
</html>

## Options

#### **disabled**Type: Boolean Default: false Disables (true) or enables (false) the button. Can be set when initialising (first creating) the button. Code examplesInitialize a button with the disabled option specified. $( ".selector" ).button({ disabled: true });Get or set the disabled option, after init. //getter var disabled = $( ".selector" ).button( "option", "disabled" ); //setter $( ".selector" ).button( "option", "disabled", true );

#### **text**Type: Boolean Default: true Whether to show any text - when set to false (display no text), icons (see icons option) must be enabled, otherwise it'll be ignored. Code examplesInitialize a button with the text option specified. $( ".selector" ).button({ text: false });Get or set the text option, after init. //getter var text = $( ".selector" ).button( "option", "text" ); //setter $( ".selector" ).button( "option", "text", false );

#### **icons**Type: Options Default: { primary: null, secondary: null } Icons to display, with or without text (see text option). The primary icon is displayed by default on the left of the label text, the secondary by default is on the right. Value for the primary and secondary properties must be a classname (String), eg. "ui-icon-gear". For using only one icon: icons: {primary:'ui-icon-locked'}. For using two icons: icons: {primary:'ui-icon-gear',secondary:'ui-icon-triangle-1-s'} Code examplesInitialize a button with the icons option specified. $( ".selector" ).button({ icons: {primary:'ui-icon-gear',secondary:'ui-icon-triangle-1-s'} });Get or set the icons option, after init. //getter var icons = $( ".selector" ).button( "option", "icons" ); //setter $( ".selector" ).button( "option", "icons", {primary:'ui-icon-gear',secondary:'ui-icon-triangle-1-s'} );

#### **label**Type: String Default: HTML content of the button, or value attribute Text to show on the button. When not specified (null), the element's html content is used, or its value attribute when it's an input element of type submit or reset; or the html content of the associated label element if its an input of type radio or checkbox Code examplesInitialize a button with the label option specified. $( ".selector" ).button({ label: "custom label" });Get or set the label option, after init. //getter var label = $( ".selector" ).button( "option", "label" ); //setter $( ".selector" ).button( "option", "label", "custom label" );

## Events

#### **create**Type: buttoncreate This event is triggered when button is created. Code examplesSupply a callback function to handle the create event as an init option. $( ".selector" ).button({ create: function(event, ui) { ... } });Bind to the create event by type: buttoncreate. $( ".selector" ).bind( "buttoncreate", function(event, ui) { ... });

## There are no events for this plugin.

Methods

### **destroy**Signature: .button( "destroy" ) Remove the button functionality completely. This will return the element back to its pre-init state.

### **disable**Signature: .button( "disable" ) Disable the button.

### **enable**Signature: .button( "enable" ) Enable the button.

### **option**Signature: .button( "option" , optionName , [value] ) Get or set any button option. If no value is specified, will act as a getter.

### **option**Signature: .button( "option" , options ) Set multiple button options at once by providing an options object.

### **widget**Signature: .button( "widget" ) Returns the .ui-button element.

### **refresh**Signature: .button( "refresh" ) Refreshes the visual state of the button. Useful for updating button state after the native element's checked or disabled state is changed programatically.

## Theming

The jQuery UI Button plugin uses the jQuery UI CSS Framework to style its look and feel, including colors and background textures. We recommend using the ThemeRoller tool to create and download custom themes that are easy to build and maintain.

If a deeper level of customization is needed, there are widget-specific classes referenced within the jquery.ui.button.css stylesheet that can be modified. These classes are highlighed in bold below.

### Sample markup with jQuery UI CSS Framework classes

<button class="**ui-button ui-button-text-only** ui-widget ui-state-default ui-corner-all">

   <span class="**ui-button-text**">Button Label</span>

</button>

**Note: This is a sample of markup generated by the button plugin, not markup you should use to create a button. The only markup needed for that is <button>Button Label</button>.**