**Jquary的选择器：**

Jquary的元素选择器（例如）：

jQuery 使用 CSS 选择器来选取 HTML 元素。

$("p") 选取 <p> 元素。

$("p.intro") 选取所有 class="intro" 的 <p> 元素。

$("p#demo") 选取所有 id="demo" 的 <p> 元素。

## jQuery 属性选择器

jQuery 使用 XPath 表达式来选择带有给定属性的元素。

$("[href]") 选取所有带有 href 属性的元素。

$("[href='#']") 选取所有带有 href 值等于 "#" 的元素。

$("[href!='#']") 选取所有带有 href 值不等于 "#" 的元素。

$("[href$='.jpg']") 选取所有 href 值以 ".jpg" 结尾的元素。

## jQuery CSS 选择器

jQuery CSS 选择器可用于改变 HTML 元素的 CSS 属性。

下面的例子把所有 p 元素的背景颜色更改为红色（例如）：

$("p").css("background-color","red");

|  |  |
| --- | --- |
| **语法** | **描述** |
| $(this) | 当前 HTML 元素 |
| $("p") | 所有 <p> 元素 |
| $("p.intro") | 所有 class="intro" 的 <p> 元素 |
| $(".intro") | 所有 class="intro" 的元素 |
| $("#intro") | id="intro" 的元素 |
| $("ul li:first") | 每个 <ul> 的第一个 <li> 元素 |
| $("[href$='.jpg']") | 所有带有以 ".jpg" 结尾的属性值的 href 属性 |
| $("div#intro .head") | id="intro" 的 <div> 元素中的所有 class="head" 的元素 |

下面是 jQuery 中事件方法的一些例子：

|  |  |
| --- | --- |
| **Event 函数** | **绑定函数至** |
| $(document).ready(function) | 将函数绑定到文档的就绪事件（当文档完成加载时） |
| $(selector).click(function) | 触发或将函数绑定到被选元素的点击事件 |
| $(selector).dblclick(function) | 触发或将函数绑定到被选元素的双击事件 |
| $(selector).focus(function) | 触发或将函数绑定到被选元素的获得焦点事件 |
| $(selector).mouseover(function) | 触发或将函数绑定到被选元素的鼠标悬停事件 |
|  |  |

## jQuery hide() 和 show()

通过 jQuery，您可以使用 hide() 和 show() 方法来隐藏和显示 HTML 元素：

$("#hide").click(function(){

$("p").hide();

});

$("#show").click(function(){

$("p").show();

});

## jQuery fadeIn() 方法

jQuery fadeIn() 用于淡入已隐藏的元素。

### 语法：

$(selector).fadeIn(speed,callback);

可选的 speed 参数规定效果的时长。它可以取以下值："slow"、"fast" 或毫秒。

可选的 callback 参数是 fading 完成后所执行的函数名称。

下面的例子演示了带有不同参数的 fadeIn() 方法：

### 实例

$("button").click(function(){

$("#div1").fadeIn();

$("#div2").fadeIn("slow");

$("#div3").fadeIn(3000);

});

## jQuery fadeOut() 方法

jQuery fadeOut() 方法用于淡出可见元素。

### 语法：

$(selector).fadeOut(speed,callback);

可选的 speed 参数规定效果的时长。它可以取以下值："slow"、"fast" 或毫秒。

可选的 callback 参数是 fading 完成后所执行的函数名称。

下面的例子演示了带有不同参数的 fadeOut() 方法：

### 实例

$("button").click(function(){

$("#div1").fadeOut();

$("#div2").fadeOut("slow");

$("#div3").fadeOut(3000);

});

## jQuery fadeToggle() 方法

jQuery fadeToggle() 方法可以在 fadeIn() 与 fadeOut() 方法之间进行切换。

如果元素已淡出，则 fadeToggle() 会向元素添加淡入效果。

如果元素已淡入，则 fadeToggle() 会向元素添加淡出效果。

### 语法：

$(selector).fadeToggle(speed,callback);

可选的 speed 参数规定效果的时长。它可以取以下值："slow"、"fast" 或毫秒。

可选的 callback 参数是 fading 完成后所执行的函数名称。

下面的例子演示了带有不同参数的 fadeToggle() 方法：

### 实例

$("button").click(function(){

$("#div1").fadeToggle();

$("#div2").fadeToggle("slow");

$("#div3").fadeToggle(3000);

});

## jQuery fadeTo() 方法

jQuery fadeTo() 方法允许渐变为给定的不透明度（值介于 0 与 1 之间）。

### 语法：

$(selector).fadeTo(speed,opacity,callback);

必需的 speed 参数规定效果的时长。它可以取以下值："slow"、"fast" 或毫秒。

fadeTo() 方法中必需的 opacity 参数将淡入淡出效果设置为给定的不透明度（值介于 0 与 1 之间）。

可选的 callback 参数是该函数完成后所执行的函数名称。

下面的例子演示了带有不同参数的 fadeTo() 方法：

### 实例

$("button").click(function(){

$("#div1").fadeTo("slow",0.15);

$("#div2").fadeTo("slow",0.4);

$("#div3").fadeTo("slow",0.7);

});

## jQuery slideDown() 方法

jQuery slideDown() 方法用于向下滑动元素。

### 语法：

$(selector).slideDown(speed,callback);

可选的 speed 参数规定效果的时长。它可以取以下值："slow"、"fast" 或毫秒。

可选的 callback 参数是滑动完成后所执行的函数名称。

下面的例子演示了 slideDown() 方法：

### 实例

$("#flip").click(function(){

$("#panel").slideDown();

});

## jQuery slideUp() 方法

jQuery slideUp() 方法用于向上滑动元素。

### 语法：

$(selector).slideUp(speed,callback);

可选的 speed 参数规定效果的时长。它可以取以下值："slow"、"fast" 或毫秒。

可选的 callback 参数是滑动完成后所执行的函数名称。

下面的例子演示了 slideUp() 方法：

### 实例

$("#flip").click(function(){

$("#panel").slideUp();

});

## jQuery slideToggle() 方法

jQuery slideToggle() 方法可以在 slideDown() 与 slideUp() 方法之间进行切换。

如果元素向下滑动，则 slideToggle() 可向上滑动它们。

如果元素向上滑动，则 slideToggle() 可向下滑动它们。

$(selector).slideToggle(speed,callback);

可选的 speed 参数规定效果的时长。它可以取以下值："slow"、"fast" 或毫秒。

可选的 callback 参数是滑动完成后所执行的函数名称。

下面的例子演示了 slideToggle() 方法：

### 实例

$("#flip").click(function(){

$("#panel").slideToggle();

});

## jQuery 动画 - animate() 方法

jQuery animate() 方法用于创建自定义动画。

### 语法：

$(selector).animate({params},speed,callback);

必需的 params 参数定义形成动画的 CSS 属性。

可选的 speed 参数规定效果的时长。它可以取以下值："slow"、"fast" 或毫秒。

可选的 callback 参数是动画完成后所执行的函数名称。

下面的例子演示 animate() 方法的简单应用；它把 <div> 元素移动到左边，直到 left 属性等于 250 像素为止：

### 实例

$("button").click(function(){

$("div").animate({left:'250px'});

});

默认情况下，所有 HTML 元素的位置都是静态的，并且无法移动。如需对位置进行操作，记得首先把元素的 CSS position 属性设置为 relative、fixed 或 absolute。

## jQuery animate() - 操作多个属性

请注意，生成动画的过程中可同时使用多个属性：

### 实例

$("button").click(function(){

$("div").animate({

left:'250px',

**opacity**:'0.5', //不透明性，值越高越实化

height:'150px',

width:'150px'

});

});

## jQuery animate() - 使用队列功能

默认地，jQuery 提供针对动画的队列功能。

这意味着如果您在彼此之后编写多个 animate() 调用，jQuery 会创建包含这些方法调用的“内部”队列。然后逐一运行这些 animate 调用。

### 实例 1

隐藏，如果您希望在彼此之后执行不同的动画，那么我们要利用队列功能：

$("button").click(function(){

**var div=$("div"); //多次调用animate可定义变量来简化调用方式**

div.animate({height:'300px',opacity:'0.4'},"slow");

div.animate({width:'300px',opacity:'0.8'},"slow");

div.animate({height:'100px',opacity:'0.4'},"slow");

div.animate({width:'100px',opacity:'0.8'},"slow");

});

## jQuery stop() 方法

jQuery stop() 方法用于停止动画或效果，在它们完成之前。

stop() 方法适用于所有 jQuery 效果函数，包括滑动、淡入淡出和自定义动画。

### 语法

$(selector).stop(stopAll,goToEnd);

可选的 stopAll 参数规定是否应该清除动画队列。默认是 false，即仅停止活动的动画，允许任何排入队列的动画向后执行。

可选的 goToEnd 参数规定是否立即完成当前动画。默认是 false。

因此，默认地，stop() 会清除在被选元素上指定的当前动画。

下面的例子演示 stop() 方法，不带参数：

$(document).ready(function(){

$("#start").click(function(){

$("div").animate({left:'100px'},5000);

$("div").animate({fontSize:'3em'},5000);

});

$("#stop").click(function(){

$("div").stop();

//默认参数为false,false.仅停止活动的动画，允许排入队列的动画向后执行。

});

$("#stop2").click(function(){

$("div").stop(true);//true,false停止，不完成当前动画

});

$("#stop3").click(function(){

$("div").stop(true,true);//停止并完成当前动画

});

**Callback 函数在当前动画 100% 完成之后执行。**

### 错误（没有 callback）

$("p").hide(1000);

alert("The paragraph is now hidden");

### 正确（有 callback）

$("p").hide(1000,function(){

alert("The paragraph is now hidden");

});