**返回顶部**

使用JQuery的animate和scrollTop方法可以创建简单地返回顶部的动画：

|  |  |
| --- | --- |
| 1 2 3 4 5 | // Back to top $('a.top').click(function (e) { e.preventDefault(); $(document.body).animate({scrollTop: 0}, 800); }); |

HTML中得有一个按钮：

|  |  |
| --- | --- |
| 1 2 | Back to top |

可以改变scrollTop的值来定位滚动条的位置。

**图片预加载**

如果页面使用了很多不是最初加载便可见的图片，有必要进行预加载：

|  |  |
| --- | --- |
| 1 2 3 4 5 6 | $.preloadImages = function () { for (var i = 0; i < arguments.length; i++) { $('img').attr('src', arguments[i]); } }; $.preloadImages('img/hover-on.png', 'img/hover-off.png'); |

**判断图片是否完全加载**

有时需要图片完全加载才能进行后面的操作，下面的这段脚本可以实现：

|  |  |
| --- | --- |
| 1 2 3 | $('img').load(function () { console.log('image load successful'); }); |

你也可以通过带id或者class的img标签来判断某张特定的图片是否完全加载

**自动修复损坏图片**

如果图片损坏，可以用另外一张进行替换：

|  |  |
| --- | --- |
| 1 2 3 | $('img').on('error', function () { $(this).prop('src', 'img/broken.png'); }); |

**Hover状态的类切换**

当用户的鼠标指针悬浮在一个可以点击的元素上时，你可以通过添加一个class改变它的视觉效果，鼠标指针离开该元素时，就移开刚刚添加的class：

|  |  |
| --- | --- |
| 1 2 3 4 5 | $('.btn').hover(function () { $(this).addClass('hover'); }, function () { $(this).removeClass('hover'); }); |

一个更简单的方式是使用toggleClass：

|  |  |
| --- | --- |
| 1 2 3 | $('.btn').hover(function () { $(this).toggleClass('hover'); }); |

**Note:**CSS may be a faster solution in this case but it's still worthwhile to know this.输入框的不可编辑

有时，你想要表单的提交按钮或者文本框变的不可编辑，直到用户完成某个特定的动作，通过修改input元素的disabled属性来实现：

$('input[type="submit"]').prop('disabled', true);

再次调用prop方法将disabled值改为false，就可以更改元素状态：

$('input[type="submit"]').prop('disabled', false);

**停止链接加载**

若你不想点击链接跳转到另一个页面或者重新加载页面，仅仅只想让它做点别的事情，如触发其它脚本，则需要阻止链接的默认行为:

|  |  |
| --- | --- |
| 1 2 3 | $('a.no-link').click(function (e) { e.preventDefault(); }); |

**Fade/Slide切换**

Slideing 和 fading 是很常用的 JQuery 动画。如果你想要在用户产生点击事件时显示一个元素，可以用fadeIn或者slideDown，若要实现第一次点击显示元素而第二次点击隐藏元素的效果，可以参考下面的脚本：

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 | // Fade $('.btn').click(function () { $('.element').fadeToggle('slow'); }); // Toggle $('.btn').click(function () { $('.element').slideToggle('slow'); }); |

**简单地手风琴**

下面的这段脚本可以简单实现手风琴效果：

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 | // Close all panels $('#accordion').find('.content').hide(); // Accordion $('#accordion').find('.accordion-header').click(function () { var next = $(this).next(); next.slideToggle('fast'); $('.content').not(next).slideUp('fast'); return false; }); |

**让两个Div等高**

有时，需要让两个Div保持等高，而不管两个Div的内容如何：

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 | $('.div').css('min-height', $(.main-div).height()); var $columns = $('.column'); var height = 0; $columns.each(function () { if ($(this).height() > height) { height = $(this).height(); } }); $columns.height(height); |

在上述的示例中，会循环一个元素集合，并将元素的高度设置成元素集合中高度最高的。若要所有 column 保持同样地高度，则可以这样：

|  |  |
| --- | --- |
| 1 2 3 4 | var $rows = $('.same-height-columns'); $rows.each(function () { $(this).find('.column').height($(this).height()); }); |

**在新的Tab/Window打开外链**

在浏览器的新Tab/Window打开外链，并且在同一个Tab/Window打开同源的链接：

|  |  |
| --- | --- |
| 1 2 3 | $('a[href^="http"]').attr('target','\_blank'); $('a[href^="//"]').attr('target','\_blank'); $('a[href^="'+window.location.origin+'"]').attr('target','\_self'); |

Note: window.location.origin doesn't work in IE10. This fix takes care of the issue.

**通过文本查找元素**

利用 JQuery 的contains()选择器可以查找一个包含特定文本的元素，如果文本不存在，元素则隐藏：

|  |  |
| --- | --- |
| 1 2 | var search = $('#search').val(); $('div:not(:contains("'+search+'"))').hide(); |