jQuery是一个非常流行而且实用的JavaScript前端框架，本文并不是介绍jQuery的[特效动画](http://www.webclks.com/archives/tag/%e7%89%b9%e6%95%88%e5%8a%a8%e7%94%bb)，而是分享一些平时积累的12个jQuery实用代码片段，希望对你有所帮助。

1. 导航菜单背景[切换效果](http://www.webclks.com/archives/tag/%e5%88%87%e6%8d%a2%e6%95%88%e6%9e%9c)

在项目的前端页面里，相对于其它的导航菜单，激活的导航菜单需要设置不同的背景。这种效果实现的方式有很多种，下面是使用[JQuery](http://www.webclks.com/archives/tag/jq)实现的一种方式：

<ul id='nav'>

<li>导航一</li>

<li>导航二</li>

<li>导航三</li>

</ul>

//注意：代码需要修饰完善

$('#nav').click(function(e) {

// 要知道siblings的使用

$(e.target).addClass('tclass').siblings('.tclass').removeClass('tclass');;

});

2.反序访问[JQuery对象](http://www.webclks.com/archives/tag/jquery%e5%af%b9%e8%b1%a1)里的元素

在某些场景下，我们可能需要反序访问通过[JQuery](http://www.webclks.com/archives/tag/jq)选择器获取到的页面元素对象，这个怎么实现呢？看下面代码：

//要掌握[JQuery对象](http://www.webclks.com/archives/tag/jquery%e5%af%b9%e8%b1%a1)的get方法 以及数组的reverse方法即可

var arr = $('#nav').find('li').get().reverse();

$.each(arr,function(index,ele){

.... ...

});

3.访问IFrame里的元素

在大多数情况下，IFrame并不是好的解决方案，但由于各种原因，项目中确实用到了IFrame,所以你需要知道怎么去访问IFrame里的元素

var iFrameDOM = $("iframe#someID").contents();

//然后，就可以通过find方法来遍历获取iFrame中的元素了

iFrameDOM.find(".message").slideUp();

4. 管理搜索框的值

现在各大网站都有搜索框，而搜索框通常都有默认值，当输入框获取焦点时，默认值消失。而一旦输入框失去焦点，而输入框里又没有输入新的值，输入框里的值又会恢复成默认值，如果往输入框里输入了新值，则输入框的值为新输入的值。这种特效用[JQuery](http://www.webclks.com/archives/tag/jq)很容易实现：

$("#searchbox")

.focus(function(){$(this).val('')})

.blur(function(){

var $this = $(this);

// '请搜索...'为搜索框默认值

($this.val() === '')? $this.val('请搜索...') : null;

});

5.部分页面加载更新

为了提高web性能，有更新时我们通常不会加载整个页面，而只是仅仅更新部分页面内容，如图片的延迟加载等。页面部分刷新的特效在JQuery中也很容易实现：

setInterval(function() { //每隔5秒钟刷新页面内容

//获取的内容将增加到 id为content的元素后

$("#content").load(url);

}, 5000);

6.采用data方法来缓存数据

在项目中，为了避免多次重复的向服务器请求数据，通常会将获取的数据缓存起来以便后续使用。通过JQuery可以很优雅的实现该功能：

var cache = {};

$.data(cache,'key','value'); //缓存数据

//获取数据

$.data(cache,'key');

7.采配置JQuery与其它库的兼容性

如果在项目中使用JQuery，$ 是最常用的变量名，但JQuery并不是唯一一个使用$作为变量名的库，为了避免命名冲突，你可以按照下面方式来组织你的代码：

//方法一： 为JQuery重新命名为 $j

var $j = jQuery.noConflict();

$j('#id')....

//方法二： 推荐使用的方式

(function($){

$(document).ready(function(){

//这儿，你可以正常的使用JQuery语法

});

})(jQuery);

8.克隆table header到表格的最下面

为了让table具有更好的可读性，我们可以将表格的header信息克隆一份到表格的底部，这种特效通过JQuery就很容易实现：

var $tfoot = $('<tfoot></tfoot>');

$($('thead').clone(true, true).children().get().reverse()).each(function(){

$tfoot.append($(this));

});

$tfoot.insertAfter('table thead');

9. 根据视窗(viewport)创建一个全屏宽度和高度(width/height)的div

下面代码完全可以让你根据viewport创建一个全屏的div。这对在不同窗口大小下展示modal或对话框时非常有效：

$('#content').css({

'width': $(window).width(),

'height': $(window).height(),

});

// make sure div stays full width/height on resize

$(window).resize(function(){

var $w = $(window);

$('#content').css({

'width': $w.width(),

'height': $w.height(),

});

});

10 测试密码的强度

在某些网站注册时常常会要求设置密码，网站也会根据输入密码的字符特点给出相应的提示，如密码过短、强度差、强度中等、强度强等。这又是怎么实现的呢？看下面代码：

<input type="password" name="pass" id="pass" />

<span id="passstrength"></span>

//下面的正则表达式建议各位收藏哦，项目上有可能会用得着

$('#pass').keyup(function(e) {

//密码为八位及以上并且字母数字特殊字符三项都包括

var strongRegex = new RegExp("^(?=.{8,})(?=.\*[A-Z])(?=.\*[a-z])(?=.\*[0-9])(?=.\*\\W).\*$", "g");

//密码为七位及以上并且字母、数字、特殊字符三项中有两项，强度是中等

var mediumRegex = new RegExp("^(?=.{7,})(((?=.\*[A-Z])(?=.\*[a-z]))|((?=.\*[A-Z])(?=.\*[0-9]))|((?=.\*[a-z])(?=.\*[0-9]))).\*$", "g");

var enoughRegex = new RegExp("(?=.{6,}).\*", "g");

if (false == enoughRegex.test($(this).val())) {

$('#passstrength').html('More Characters');

} else if (strongRegex.test($(this).val())) {

$('#passstrength').className = 'ok';

$('#passstrength').html('Strong!');

} else if (mediumRegex.test($(this).val())) {

$('#passstrength').className = 'alert';

$('#passstrength').html('Medium!');

} else {

$('#passstrength').className = 'error';

$('#passstrength').html('Weak!');

}

return true;

});

11.使用JQuery重绘图片的大小

关于图片大小的重绘，你可以在服务端来实现，也可以通过JQuery在客户端实现。

$(window).bind("load", function() {

// IMAGE RESIZE

$('#product\_cat\_list img').each(function() {

var maxWidth = 120;

var maxHeight = 120;

var ratio = 0;

var width = $(this).width();

var height = $(this).height();

if(width > maxWidth){

ratio = maxWidth / width;

$(this).css("width", maxWidth);

$(this).css("height", height \* ratio);

height = height \* ratio;

}

var width = $(this).width();

var height = $(this).height();

if(height > maxHeight){

ratio = maxHeight / height;

$(this).css("height", maxHeight);

$(this).css("width", width \* ratio);

width = width \* ratio;

}

});

//$("#contentpage img").show();

// IMAGE RESIZE

});

12.滚动时动态加载页面内容

有些网站的网页内容不是一次性加载完毕的，而是在鼠标向下滚动时动态加载的，这是怎么做到的呢？看下面代码：

var loading = false;

$(window).scroll(function(){

if((($(window).scrollTop()+$(window).height())+250)>=$(document).height()){

if(loading == false){

loading = true;

$('#loadingbar').css("display","block");

$.get("load.php?start="+$('#loaded\_max').val(), function(loaded){

$('body').append(loaded);

$('#loaded\_max').val(parseInt($('#loaded\_max').val())+50);

$('#loadingbar').css("display","none");

loading = false;

});

}

}

});

$(document).ready(function() {

$('#loaded\_max').val(50);

});