DOM和 CSS操作

学习要点：

1.DOM简介

2.设置元素及内容

3.元素属性操作

4.元素样式操作

5.CSS方法

DOM在是一种文档对象模型。方便开发者对HTML结构元素内容进行展示和修改。在 JavaScript中，DOM不但内容庞大繁杂，而且我们开发的过程中需要考虑更多的兼容性、扩展性。在 jQuery中，已经将最常用的 DOM操作方法进行了有效封装，并且不需要考虑浏览器的兼容性。

一．DOM简介

由于课程是基于 JavaScript基础上完成的，这里我们不去详细的了解 DOM到底是什么。

只需要知道几个基本概念：

1.D表示的是页面文档 Document、O表示对象，即一组含有独立特性的数据集合、M

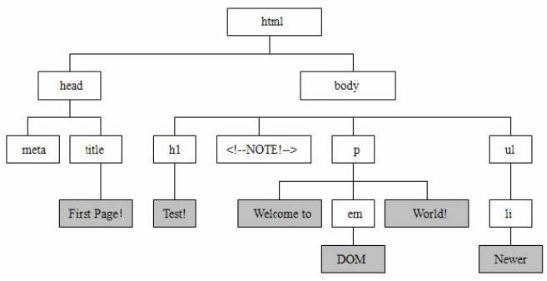
表示模型，即页面上的元素节点和文本节点。

2.DOM有三种形式，标准 DOM、HTML DOM、CSS DOM，大部分都进行了一系列的

封装，在 jQuery中并不需要深刻理解它。

3.树形结构用来表示 DOM，就非常的贴切，大部分操作都是元素节点操作，还有少部

分是文本节点操作。



二．设置元素及内容

我们通过前面所学习的各种选择器、过滤器来得到我们想要操作的元素。这个时候，我

们就可以对这些元素进行 DOM的操作。那么，最常用的操作就是对元素内容的获取和修改。



在常规的 DOM元素中，我们可以使用 html()和 text()方法获取内部的数据。html()方法

可以获取或设置 html内容，text()可以获取或设置文本内容。

$('#box').html();//获取 html内容

$('#box').text();//获取文本内容，会自动清理 html标签

$('#box').html('<em>wxx</em>');//设置 html内容

$('#box').text('<em>wxx</em>');//设置文本内容，会自动转义 html标签

注意：当我们使用 html()或 text()设置元素里的内容时，会清空原来的数据。而我们期

望能够追加数据的话，需要先获取原本的数据。

$('#box').html($('#box').html() + '<em>wxx</em>');//追加数据

如果元素是表单的话，jQuery提供val()方法进行获取或设置内部的文本数据。

$('input').val();//获取表单内容

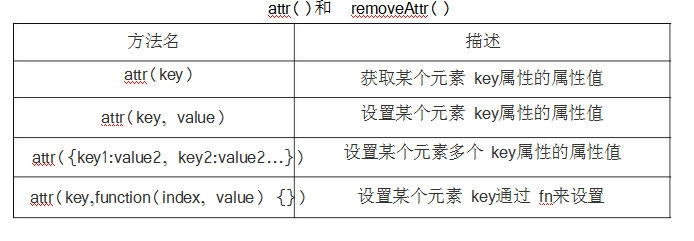
$('input').val('www.li.cc');//设置表单内容

如果想设置多个选项的选定状态，比如下拉列表、单选复选框等等，可以通过数组传递操作。

$("input").val(["check1","check2", "radio1" ]); //value值是这些的将被选定

三．元素属性操作

除了对元素内容进行设置和获取，通过 jQuery也可以对元素本身的属性进行操作，包括获取属性的属性值、设置属性的属性值，并且可以删除掉属性。



后2个作为作业去实现，给某一个元素添加多个自定义属性

attr({key1:value2, key2:value2...})

attr(key,function(index, value) {})

$('div').attr('title');//获取属性的属性值

$('div').attr('title', '我是域名');//设置属性及属性值

注意：attr()方法里的 function() {}，可以不传参数。可以只传一个参数 index，表示当前

元素的索引(从 0开始)。也可以传递两个参数 index、value，第二个参数表示属性原本的值。

注意：jQuery中很多方法都可以使用 function() {}来返回出字符串，比如 html()、text()、

val()和上一章刚学过的 is()、filter()方法。而如果又涉及到多个元素集合的话，还可以传递

index参数来获取索引值，并且可以使用第二个参数 value(并不是所有方法都适合，有兴趣

可以自己逐个尝试)。

删除指定的属性，这个方法就不可以使用匿名函数，传递 index和 value均无效。

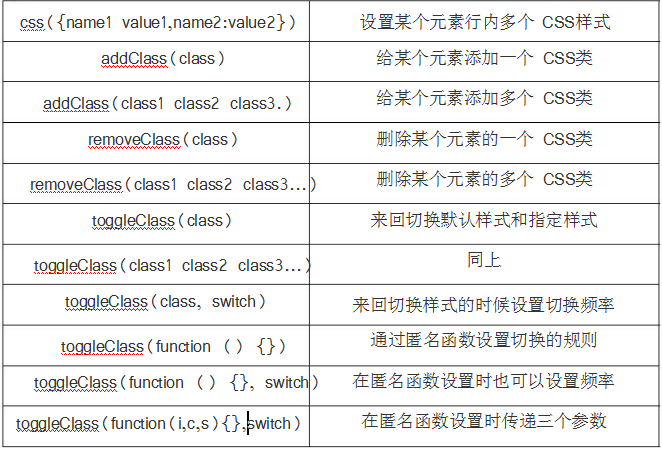
$('div').removeAttr('title');//删除指定的属性

四．元素样式操作

元素样式操作包括了直接设置 CSS样式、增加 CSS类别、类别切换、删除类别这几种

操作方法。而在整个 jQuery使用频率上来看，CSS样式的操作也是极高的，所以需要重点

掌握。



$('div').css('color');//获取元素行内 CSS样式的颜色

$('div').css('color', 'red');//设置元素行内 CSS样式颜色为红色

在 CSS获取上，我们也可以获取多个 CSS样式，而获取到的是一个对象数组，如果用

传统方式进行解析需要使用 for in遍历。

var box = $('div').css(['color', 'height', 'width']); //得到多个 CSS样式的数组对象

for (var i in box) {

alert(i + ':' + box[i]);//逐个遍历出来

}

jQuery提供了一个遍历工具专门来处理这种对象数组，$.each()方法，这个方法可以轻

松的遍历对象数组。

$.each(box, function (attr, value) {//遍历 JavaScript原生态的对象数组

alert(attr + ':' + value);

});

使用$.each()可以遍历原生的 JavaScript对象数组，如果是 jQuery对象的数组怎么使

用.each()方法呢？

$('div').each(function (index, element) {

alert(index + ':' + element);//index为索引，element为元素 DOM

});

在需要设置多个样式的时候，我们可以传递多个 CSS样式的键值对即可。

$('div').css({'background-color' : '#ccc','color' : 'red','font-size' : '20px'});

如果想设置某个元素的 CSS样式的值，但这个值需要计算我们可以传递一个匿名函数。

$('div').css('width', function (index, value) {

return (parseInt(value) - 500) + 'px';

});

除了行内 CSS设置，我们也可以直接给元素添加 CSS类，可以添加单个或多个，并且

也可以删除它。

除了行内 CSS设置，我们也可以直接给元素添加 CSS类，可以添加单个或多个，并且

也可以删除它。

$('div').addClass('red');//添加一个 CSS类

$('div').addClass('red bg');//添加多个 CSS类

$('div').removeClass('bg');//删除一个 CSS类

$('div').removeClass('red bg');//删除多个 CSS类

我们还可以结合事件来实现 CSS类的样式切换功能。

$('div').click(function () {//当点击后触发

$(this).toggleClass('red size');//单个样式多个样式均可

});

.toggleClass()方法的第二个参数可以传入一个布尔值，true表示执行切换到 class类，false

表示执行回默认 class类(默认的是空 class)，运用这个特性，我们可以设置切换的频率。

var count = 0;

$('div').click(function () {//每点击两次切换一次 red

$(this).toggleClass('red', count++ % 3 == 0);

});

默认的 CSS类切换只能是无样式和指定样式之间的切换，如果想实现样式 1和样式 2

之间的切换还必须自己写一些逻辑。

$('div').click(function () {

$(this).toggleClass('red size');//一开始切换到样式 2

if ($(this).hasClass('red')) {//判断样式 2存在后

$(this).removeClass('blue');//删除样式 1

} else {

$(this).toggleClass('blue');//添加样式 1，这里也可以addClass

}

});

上面的方法较为繁琐，.toggleClass()方法提供了传递匿名函数的方式，来设置你所需要

切换的规则。

$('div').click(function () {

$(this).toggleClass(function () {//传递匿名函数，返回要切换的样式

return $(this).hasClass('red') ? 'blue' : 'red size';

});

});

注意：上面虽然一句话实现了这个功能，但还是有一些小缺陷，因为原来的 class类没

有被删除，只不过被替代了而已。所以，需要改写一下。

$('div').click(function () {

$(this).toggleClass(function () {

if ($(this).hasClass('red')) {

$(this).removeClass('red');

return 'green';

} else {

$(this).removeClass('green');

return 'red';

}

});

});

我们也可以在传递匿名函数的模式下，增加第二个频率参数。

var count = 0;

$('div').click(function () {

$(this).toggleClass(function () {

return $(this).hasClass('red') ? 'blue' : 'red size';

},count++ % 3 == 0);//增加了频率

});

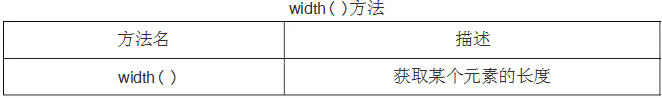
对于.toggleClass()传入匿名函数的方法，还可以传递 index索引、class类两个参数以及

频率布尔值，可以得到当前的索引、class类名和频率布尔值。

五．CSS方法

jQuery不但提供了 CSS的核心操作方法，比如.css()、.addClass()等。还封装了一些特殊

功能的 CSS操作方法，我们分别来了解一下。





$('div').width();//获取元素的长度，返回的类型为 number

$('div').width(500);//设置元素长度，直接传数值，默认加 px

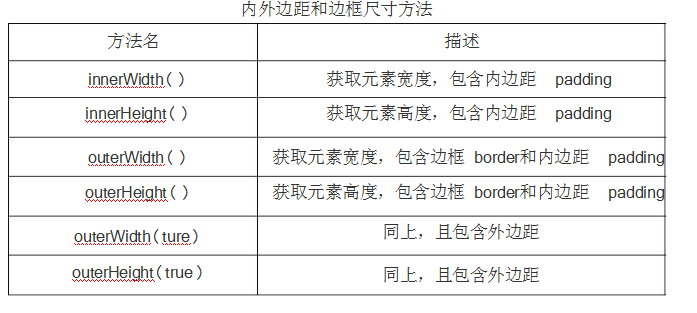
$('div').width('500pt');//同上，设置了 pt单位

$('div').width(function (index, value) {//index是索引，value是原本值

return value - 500;//无须调整类型，直接计算

});

作业：自己去完成height()方法的操作，方法与width类似



alert($('div').width());//不包含

alert($('div').innerWidth());//包含内边距 padding

alert($('div').outerWidth());//包含内边距 padding+边框 border

alert($('div').outerWidth(true));//包含内边距 padding+边框 border+外边距 margin



$('strong').offset().left;//相对于视口的偏移

$('strong').position().left;//相对于父元素的偏移

$(window).scrollTop();//获取当前滚动条的位置

$(window).scrollTop(300);//设置当前滚动条的位置