# Jquary HTML

## 获得内容 - text()、html() 以及 val()

三个简单实用的用于 DOM 操作的 jQuery 方法：

* text() - 设置或返回所选元素的文本内容
* html() - 设置或返回所选元素的内容（包括 HTML 标记）
* val() - 设置或返回表单字段的值

下面的例子演示如何通过 jQuery text() 和 html() 方法来获得内容：

### 实例

$("#btn1").click(function(){

alert("Text: " + $("#test").text());

});

$("#btn2").click(function(){

alert("HTML: " + $("#test").html());

});

## 获取属性 - attr()

jQuery attr() 方法用于获取属性值。

下面的例子演示如何获得链接中 href 属性的值：

### 实例

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

alert($("#w3s").attr("href"));

});

下面的例子演示如何通过 jQuery val() 方法获得输入字段的值：

### 实例

$("#btn1").click(function(){

alert("Value: " + $("#test").val());

});

## 设置内容 - text()、html() 以及 val()

我们将使用前一章中的三个相同的方法来设置内容：

* text() - 设置或返回所选元素的文本内容
* html() - 设置或返回所选元素的内容（包括 HTML 标记）
* val() - 设置或返回表单字段的值

下面的例子演示如何通过 text()、html() 以及 val() 方法来设置内容：

### 实例

$("#btn1").click(function(){

$("#test1").text("Hello world!");

});

$("#btn2").click(function(){

$("#test2").html("<b>Hello world!</b>");

});

$("#btn3").click(function(){

$("#test3").val("Dolly Duck");

});

## 添加新的 HTML 内容

我们将学习用于添加新内容的四个 jQuery 方法：

* append() - 在被选元素的结尾插入内容
* prepend() - 在被选元素的开头插入内容
* after() - 在被选元素之后插入内容
* before() - 在被选元素之前插入内容

## jQuery append() 方法

jQuery append() 方法在被选元素的结尾插入内容。

### 实例

$("p").append("Some appended text.");

## 通过 after() 和 before() 方法添加若干新元素

after() 和 before() 方法能够通过参数接收无限数量的新元素。可以通过 text/HTML、jQuery 或者 JavaScript/DOM 来创建新元素。

在下面的例子中，我们创建若干新元素。这些元素可以通过 text/HTML、jQuery 或者 JavaScript/DOM 来创建。然后我们通过 after() 方法把这些新元素插到文本中（对 before() 同样有效）：

<script>

function afterText()

{

var txt1="<b>I </b>"; // 以 HTML 创建元素

var txt2=$("<i></i>").text("love "); // 通过 jQuery 创建元素

var txt3=document.createElement("big"); // 通过 DOM 创建元素

txt3.innerHTML="jQuery!";

$("img").after(txt1,txt2,txt3); // 在 img 之后插入新元素

}

</script>

## 过滤被删除的元素

jQuery remove() 方法也可接受一个参数，允许您对被删元素进行过滤。

该参数可以是任何 jQuery 选择器的语法。

下面的例子删除 class="italic" 的所有 <p> 元素：

### 实例

$("p").remove(".italic");

## jQuery 操作 CSS

jQuery 拥有若干进行 CSS 操作的方法。我们将学习下面这些：

* addClass() - 向被选元素添加一个或多个类
* removeClass() - 从被选元素删除一个或多个类
* toggleClass() - 对被选元素进行添加/删除类的切换操作
* css() - 设置或返回样式属性

## jQuery addClass() 方法

下面的例子展示如何向不同的元素添加 class 属性。当然，在添加类时，您也可以选取多个元素：

### 实例

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

$("h1,h2,p").addClass("blue");

$("div").addClass("important");

});

**多个css()样式设置方法：**

多个css样式设置格式，属性和值中间冒号，不同属性之间用半角,隔开

$("selector").css({"属性1":"值1","属性2":"值2","属性3":"值3"})：例如

<script>

$(document).ready(function(){

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

$("p").css({"background-color":"yellow","font-weight":"bold"});

});

});

</script>

## jQuery innerWidth() 和 innerHeight() 方法

innerWidth() 方法返回元素的宽度（包括内边距）。

innerHeight() 方法返回元素的高度（包括内边距）。

下面的例子返回指定的 <div> 元素的 inner-width/height：

### 实例

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

var txt="";

txt+="Inner width: " + $("#div1").innerWidth() + "</br>";

txt+="Inner height: " + $("#div1").innerHeight();

$("#div1").html(txt);

});

outerWidth(true) 方法返回元素的宽度（包括内边距、边框和外边距）。

outerHeight(true) 方法返回元素的高度（包括内边距、边框和外边距）。

当属性不设置默认为false，则outerHeight和outerWidth就不包括外边距（margin）

### 实例

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

var txt="";

txt+="Outer width (+margin): " + $("#div1").outerWidth(true) + "</br>";

txt+="Outer height (+margin): " + $("#div1").outerHeight(true);

$("#div1").html(txt);

});

## jQuery parent() 方法

parent() 方法返回被选元素的直接父元素。

该方法只会向上一级对 DOM 树进行遍历。

下面的例子返回每个 <span> 元素的的直接父元素：

### 实例

$(document).ready(function(){

$("span").parent();

});

## jQuery parents() 方法

parents() 方法返回被选元素的所有祖先元素，它一路向上直到文档的根元素 (<html>)。

下面的例子返回所有 <span> 元素的所有祖先：

### 实例

$(document).ready(function(){

$("span").parents();

});

## jQuery parentsUntil() 方法

parentsUntil() 方法返回介于两个给定元素之间的所有祖先元素。

下面的例子返回介于 <span> 与 <div> 元素之间的所有祖先元素：

但不包括span和div自身这两个元素。可以直接设置它的css样式。

### 实例

<script>

$(document).ready(function(){

$("span").parentsUntil("div").css({"color":"red","border":"2px solid red"});

});

</script>

## 向下遍历 DOM 树

下面是两个用于向下遍历 DOM 树的 jQuery 方法：

* children()
* find()

也可以使用可选参数来过滤对子元素的搜索。

下面的例子返回类名为 "1" 的所有 <p> 元素，并且它们是 <div> 的直接子元素：

### 实例

$(document).ready(function(){

$("div").children("p.1");

});

下面的例子返回 <div> 的所有后代：

### 实例

$(document).ready(function(){

$("div").find("\*");

});

## jQuery siblings() 方法

siblings() 方法返回被选元素的所有同胞元素。

下面的例子返回 <h2> 的所有同胞元素（不包括自身）：

### 实例

$(document).ready(function(){

$("h2").siblings();

});

## jQuery next() 方法

next() 方法返回被选元素的下一个同胞元素。

**该方法只返回一个元素。**

下面的例子返回 <h2> 的下一个同胞元素：

### 实例

$(document).ready(function(){

$("h2").next();

});

## jQuery nextAll() 方法

nextAll() 方法返回被选元素的所有跟随的同胞元素。

下面的例子返回 <h2> 的所有跟随的同胞元素：

### 实例

$(document).ready(function(){

$("h2").nextAll();

});

## jQuery nextUntil() 方法

nextUntil() 方法返回介于两个给定参数之间的所有跟随的同胞元素。

下面的例子返回介于 <h2> 与 <h6> 元素之间的所有同胞元素：

**注意：之间的元素，不包括h2,h6.**

### 实例

$(document).ready(function(){

$("h2").nextUntil("h6");

});

## jQuery prev(), prevAll() & prevUntil() 方法

prev(), prevAll() 以及 prevUntil() 方法的工作方式与上面的方法类似，只不过方向相反而已：

**它们返回的是前面的同胞元素**

（在 DOM 树中沿着同胞元素向后遍历，而不是向前）。

## 缩写搜索元素的范围

三个最基本的过滤方法是：first(), last() 和 eq()，它们允许您基于其在一组元素中的位置来选择一个特定的元素。

其他过滤方法，比如 filter() 和 not() 允许您选取匹配或不匹配某项指定标准的元素。

## jQuery first() 方法

first() 方法返回被选元素的首个元素。

下面的例子选取首个 <div> 元素内部的第一个 <p> 元素：

<script>

$(document).ready(function(){

$("div p").first().css("background-color","yellow");

});

</script>

## jQuery eq() 方法

**eq() 方法返回被选元素中带有指定索引号的元素。**

索引号从 0 开始，因此首个元素的索引号是 0 而不是 1。下面的例子选取第二个 <p> 元素（索引号 1）：

### 实例

$(document).ready(function(){

$("p").eq(1);

});

## jQuery filter() 方法

**filter() 方法允许您规定一个标准。不匹配这个标准的元素会被从集合中删除，匹配的元素会被返回。**

下面的例子返回带有类名 "intro" 的所有 <p> 元素：

### 实例

$(document).ready(function(){

$("p").filter(".intro");

});

## jQuery not() 方法

not() 方法返回不匹配标准的所有元素。

提示：not() 方法与 filter() 相反。

下面的例子返回不带有类名 "intro" 的所有 <p> 元素：

### 实例

$(document).ready(function(){

$("p").not(".intro");

});

代码示例：

<head>

<script src="/jquery/jquery-1.11.1.min.js">

</script>

<script>

$(document).ready(function(){

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

$("p").filter(".hu").css("background-color","lightblue");

})

})

</script>

</head>

<body>

<div>

<p>hello world</p>

<p class="hu">web前端免费公开课</p>

</div>

<div>

<h4>文化艺术出版社</h4>

<p class="hu">jquary 遍历</p>

</div>

<button id="">资源库</button>

</body>