**jQuery常用的选择器分类及其用法**

**[一、基本选择器](#_Toc4877)** [3](#_Toc4877)

**[1. id选择器（指定id元素）](#_Toc14204)** [3](#_Toc14204)

**[2. class选择器（遍历css类元素）](#_Toc15883)** [3](#_Toc15883)

**[3. element选择器（遍历html元素）](#_Toc23418)** [3](#_Toc23418)

**[4. \* 选择器（遍历所有元素）](#_Toc2502)** [3](#_Toc2502)

**[5. 并列选择器](#_Toc23885)** [3](#_Toc23885)

**[二、 层次选择器](#_Toc12516)** [4](#_Toc12516)

**[1. parent child（所有后代元素）](#_Toc20296)** [4](#_Toc20296)

**[2. parent > child（直系子元素）](#_Toc23955)** [4](#_Toc23955)

**[3. prev + next（下一个兄弟元素，等同于next()方法）](#_Toc6015)** [4](#_Toc6015)

**[4. prev ~ siblings（prev元素的所有兄弟元素，等同于nextAll()方法）](#_Toc6085)** [5](#_Toc6085)

**[三、 过滤选择器](#_Toc14210)** [5](#_Toc14210)

**[1. 基本过滤选择器](#_Toc13965)** [5](#_Toc13965)

**[1.1 :first和:last（取第一个元素或最后一个元素）](#_Toc7894)** [5](#_Toc7894)

**[1.2 :not（取非元素）](#_Toc29999)** [6](#_Toc29999)

**[1.3 :even和:odd（取偶数索引或奇数索引元素，索引从0开始，even表示偶数，odd表示奇数）](#_Toc17973)** [6](#_Toc17973)

**[1.4 :eq(x) （取指定索引的元素）](#_Toc4575)** [7](#_Toc4575)

**[1.5 :gt(x)和:lt(x)（取大于x索引或小于x索引的元素）](#_Toc483)** [7](#_Toc483)

**[1.6 :header（取H1~H6标题元素）](#_Toc19264)** [7](#_Toc19264)

**[2. 内容过滤选择器](#_Toc11163)** [8](#_Toc11163)

**[2.1 :contains(text)（取包含text文本的元素）](#_Toc31527)** [8](#_Toc31527)

**[2.2 :empty（取不包含子元素或文本为空的元素）](#_Toc2264)** [9](#_Toc2264)

**[2.3 :has(selector)（取选择器匹配的元素）](#_Toc16434)** [9](#_Toc16434)

**[2.4 :parent（取包含子元素或文本的元素）](#_Toc14481)** [10](#_Toc14481)

**[3. 可见性过滤选择器](#_Toc29984)** [10](#_Toc29984)

**[3.1 :hidden（取不可见的元素）](#_Toc2971)** [10](#_Toc2971)

**[3.2 :visible（取可见的元素）](#_Toc27684)** [12](#_Toc27684)

**[4. 属性过滤选择器](#_Toc1355)** [12](#_Toc1355)

**[4.1 [attribute]（取拥有attribute属性的元素）](#_Toc21887)** [13](#_Toc21887)

**[4.2 [attribute = value]和[attribute != value]（取attribute属性值等于value或不等于value的元素）](#_Toc28185)** [13](#_Toc28185)

**[4.3 [attribute ^= value], [attribute $= value]和[attribute \*= value]（attribute属性值以value开始，以value结束，或包含value值）](#_Toc13341)** [14](#_Toc13341)

**[4.4 [selector1][selector2]（复合型属性过滤器，同时满足多个条件）](#_Toc28743)** [15](#_Toc28743)

**[5. 子元素过滤选择器](#_Toc6170)** [15](#_Toc6170)

**[5.1 :first-child和:last-child](#_Toc14045)** [15](#_Toc14045)

**[5.2 :only-child](#_Toc22094)** [17](#_Toc22094)

**[5.3 :nth-child](#_Toc2431)** [18](#_Toc2431)

**[6. 表单对象属性过滤选择器](#_Toc27983)** [20](#_Toc27983)

**[6.1 :enabled和:disabled（取可用或不可用元素）](#_Toc29665)** [20](#_Toc29665)

**[6.2 :checked（取选中的单选框或复选框元素）](#_Toc2967)** [21](#_Toc2967)

**[6.3 :selected（取下拉列表被选中的元素）](#_Toc15843)** [22](#_Toc15843)

**[四、表单选择器](#_Toc1956)** [23](#_Toc1956)

**[1. :input（取input,textarea,select,button元素）](#_Toc24996)** [23](#_Toc24996)

**[2. :text（取单行文本框元素）和:password（取密码框元素）](#_Toc25165)** [23](#_Toc25165)

**[3. :radio（取单选框元素）](#_Toc22278)** [24](#_Toc22278)

**[4. :checkbox（取复选框元素）](#_Toc27819)** [24](#_Toc27819)

**[5. :submit（取提交按钮元素）](#_Toc2172)** [25](#_Toc2172)

**[6. :reset（取重置按钮元素）](#_Toc13883)** [25](#_Toc13883)

**[7. :button（取按钮元素）](#_Toc8984)** [25](#_Toc8984)

**[8. :file（取上传域元素）](#_Toc19753)** [26](#_Toc19753)

**[9. :hidden（取不可见元素）](#_Toc17454)** [26](#_Toc17454)

**一、基本选择器**

**1. id选择器（指定id元素）**

将id="one"的元素背景色设置为黑色。（id选择器返单个元素）

$(document).ready(function () {

$('#one').css('background', '#000');

});

**2. class选择器（遍历css类元素）**

将class="cube"的元素背景色设为黑色

$(document).ready(function () {

$('.cube').css('background', '#000');

});

**3. element选择器（遍历html元素）**

将p元素的文字大小设置为12px

$(document).ready(function () {

$('p').css('font-size', '12px');

});

**4. \* 选择器（遍历所有元素）**

$(document).ready(function () {

// 遍历form下的所有元素，将字体颜色设置为红色

$('form \*').css('color', '#FF0000');

});

**5. 并列选择器**

$(document).ready(function () {

// 将p元素和div元素的margin设为0

$('p, div').css('margin', '0');

});

1. **层次选择器**

**1. parent child（所有后代元素）**

$(document).ready(function () {

// 选取div的所有后代span元素，将字体颜色设为红色

$('div span').css('color', '#FF0000');

});

下面的代码，下一代和下下代都属于后代元素，所以两个span都会变色。

<div>

<span>123</span>

<p>

<span>456</span>

</p>

</div>

**2. parent > child（直系子元素）**

$(document).ready(function () {

// 选取div下的第一代span元素，将字体颜色设为红色

$('div > span').css('color', '#FF0000');

});

下面的代码，只有第一个span会变色，第二个span不属于div的一代子元素，颜色保持不变。

<div>

<span>123</span>

<p>

<span>456</span>

</p>

</div>

**3. prev + next（下一个兄弟元素，等同于next()方法）**

$(document).ready(function () {

// 选取class为item的下一个div兄弟元素

$('.item + div').css('color', '#FF0000');

// 等价代码

//$('.item').next('div').css('color', '#FF0000');

});

下面的代码，只有123和789会变色

<p class="item"></p>

<div>123</div>

<div>456</div>

<span class="item"></span>

<div>789</div>

**4. prev ~ siblings（prev元素的所有兄弟元素，等同于nextAll()方法）**

$(document).ready(function () {

// 选取class为inside之后的所有div兄弟元素

$('.inside ~ div').css('color', '#FF0000');

// 等价代码

//$('.inside').nextAll('div').css('color', '#FF0000');

});

下面的代码，G2和G4会变色

<div class="inside">G1</div>

<div>G2</div>

<span>G3</span>

<div>G4</div>

**三、 过滤选择器**

**1. 基本过滤选择器**

**1.1 :first和:last（取第一个元素或最后一个元素）**

$(document).ready(function () {

$('span:first').css('color', '#FF0000');

$('span:last').css('color', '#FF0000');

});

下面的代码，G1（first元素）和G3（last元素）会变色

<span>G1</span>

<span>G2</span>

<span>G3</span>

**1.2 :not（取非元素）**

$(document).ready(function () {

$('div:not(.wrap)').css('color', '#FF0000');

});

下面的代码，G1会变色

<div>G1</div>

<div class="wrap">G2</div>

但是，请注意下面的代码：

<div>

G1

<div class="wrap">G2</div>

</div>

当G1所在div和G2所在div是父子关系时，G1和G2都会变色。

**1.3 :even和:odd（取偶数索引或奇数索引元素，索引从0开始，even表示偶数，odd表示奇数）**

$(document).ready(function () {

$('tr:even').css('background', '#EEE'); // 偶数行颜色

$('tr:odd').css('background', '#DADADA'); // 奇数行颜色

});

A、C行颜色#EEE（第一行的索引为0），B、D行颜色#DADADA

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030141021.png)

<table width="200" cellpadding="0" cellspacing="0">

<tbody>

<tr><td>A</td></tr>

<tr><td>B</td></tr>

<tr><td>C</td></tr>

<tr><td>D</td></tr>

</tbody>

</table>

**1.4 :eq(x) （取指定索引的元素）**

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030158348.png)

$(document).ready(function () {

$('tr:eq(2)').css('background', '#FF0000');

});

更改第三行的背景色，在上面的代码中C的背景会变色。

**1.5 :gt(x)和:lt(x)（取大于x索引或小于x索引的元素）**

$(document).ready(function () {

$('ul li:gt(2)').css('color', '#FF0000');

$('ul li:lt(2)').css('color', '#0000FF');

});

L4和L5会是红色，L1和L2会是蓝色，L3是默认颜色

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030173723.png)

<ul>

<li>L1</li>

<li>L2</li>

<li>L3</li>

<li>L4</li>

<li>L5</li>

</ul>

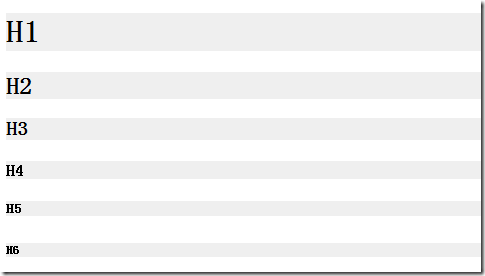
**1.6 :header（取H1~H6标题元素）**

$(document).ready(function () {

$(':header').css('background', '#EFEFEF');

});

下面的代码，H1~H6的背景色都会变

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030191573.png)

<h1>H1</h1>

<h2>H2</h2>

<h3>H3</h3>

<h4>H4</h4>

<h5>H5</h5>

<h6>H6</h6>

**2. 内容过滤选择器**

**2.1 :contains(text)（取包含text文本的元素）**

$(document).ready(function () {

// dd元素中包含"jQuery"文本的会变色

$('dd:contains("jQuery")').css('color', '#FF0000');

});

下面的代码，第一个dd会变色

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030211375.png)

<dl>

<dt>技术</dt>

<dd>jQuery, .NET, CLR</dd>

<dt>SEO</dt>

<dd>关键字排名</dd>

<dt>其他</dt>

<dd></dd>

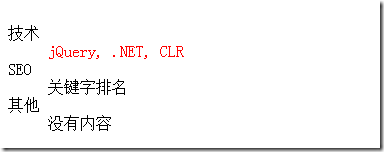
</dl>

**2.2 :empty（取不包含子元素或文本为空的元素）**

$(document).ready(function () {

$('dd:empty').html('没有内容');

});

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030228701.png)

上面第三个dd会显示"没有内容"文本

**2.3 :has(selector)（取选择器匹配的元素）**

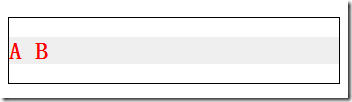
$(document).ready(function () {

// 为包含span元素的div添加边框

$('div:has(span)').css('border', '1px solid #000');

});

即使span不是div的直系子元素，也会生效

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/20120602203024488.png)

<div>

<h2>

A

<span>B</span>

</h2>

</div>

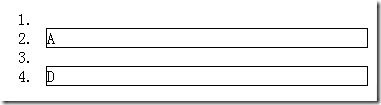
**2.4 :parent（取包含子元素或文本的元素）**

$(document).ready(function () {

$('ol li:parent').css('border', '1px solid #000');

});

下面的代码，A和D所在的li会有边框

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030269451.png)

<ol>

<li></li>

<li>A</li>

<li></li>

<li>D</li>

</ol>

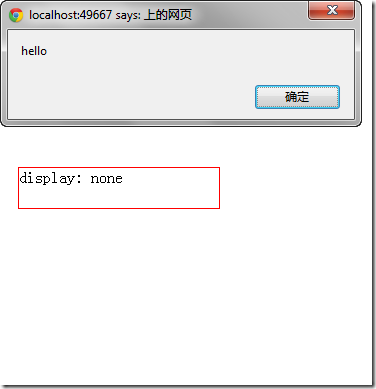
**3. 可见性过滤选择器**

**3.1 :hidden（取不可见的元素）**

jQuery至1.3.2之后的:hidden选择器仅匹配display:none或<input type="hidden" />的元素，而不匹配visibility: hidden或opacity:0的元素。这也意味着hidden只匹配那些“隐藏的”并且不占空间的元素，像visibility:hidden或opactity:0的元素占据了空间，会被排除在外。

参照：<http://www.jquerysdk.com/api/hidden-selector>

下面的代码，先弹出"hello"对话框，然后hid-1会显示，hid-2仍然是不可见的。

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/2012060220302892.png)

<html xmlns="http://www.w3.org/1999/xhtml" >

<head runat="server">

<title></title>

<style type="text/css">

div

{

margin: 10px;

width: 200px;

height: 40px;

border: 1px solid #FF0000;

display:block;

}

.hid-1

{

display: none;

}

.hid-2

{

visibility: hidden;

}

</style>

<script type="text/javascript" src="js/jquery.min.js"></script>

<script type="text/javascript">

$(document).ready(function() {

$('div:hidden').show(500);

alert($('input:hidden').val());

});

</script>

</head>

<body>

<div class="hid-1">display: none</div>

<div class="hid-2">visibility: hidden</div>

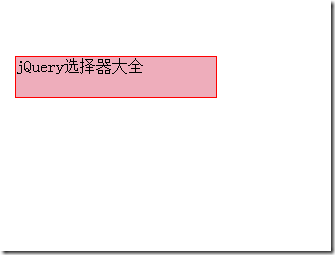
<input type="hidden" value="hello"/>

</body>

</html>

**3.2 :visible（取可见的元素）**

下面的代码，最后一个div会有背景色

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030302369.png)

<script type="text/javascript">

$(document).ready(function() {

$('div:visible').css('background', '#EEADBB');

});

</script>

<div class="hid-1">display: none</div>

<div class="hid-2">visibility: hidden</div>

<input type="hidden" value="hello"/>

<div>

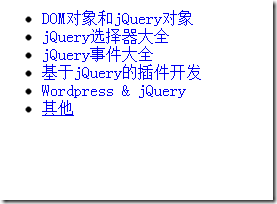
jQuery选择器大全

</div>

**4. 属性过滤选择器**

**4.1 [attribute]（取拥有attribute属性的元素）**

下面的代码，最后一个a标签没有title属性，所以它仍然会带下划线

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030329347.png)

<script type="text/javascript">

$(document).ready(function() {

$('a[title]').css('text-decoration', 'none');

});

</script>

<ul>

<li><a href="#" title="DOM对象和jQuery对象" class="item">DOM对象和jQuery对象</a></li>

<li><a href="#" title="jQuery选择器大全" class="item-selected">jQuery选择器大全</a></li>

<li><a href="#" title="jQuery事件大全" class="item">jQuery事件大全</a></li>

<li><a href="#" title="基于jQuery的插件开发" class="item">基于jQuery的插件开发</a></li>

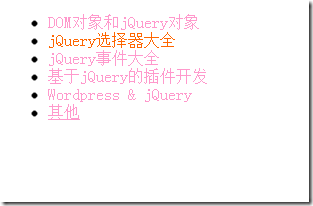
<li><a href="#" title="Wordpress & jQuery" class="item">Wordpress & jQuery</a></li>

<li><a href="#" class="item">其他</a></li>

</ul>

**4.2 [attribute = value]和[attribute != value]（取attribute属性值等于value或不等于value的元素）**

分别为class="item"和class!=item的a标签指定文字颜色

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030343576.png)

<script type="text/javascript">

$(document).ready(function() {

$('a[class=item]').css('color', '#FF99CC');

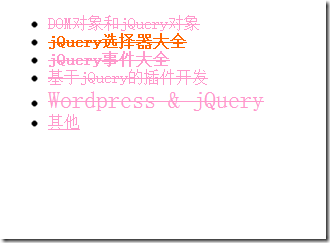
$('a[class!=item]').css('color', '#FF6600');

});

</script>

**4.3 [attribute ^= value], [attribute $= value]和[attribute \*= value]（attribute属性值以value开始，以value结束，或包含value值）**

在属性选择器中，^$符号和正则表达式的开始结束符号表示的含义是一致的，\*模糊匹配，类似于sql中的like '%str%'。

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030364217.png)

<script type="text/javascript">

// 识别大小写，输入字符串时可以输入引号，[title^=jQuery]和[title^="jQuery"]是一样的

$('a[title^=jQuery]').css('font-weight', 'bold');

$('a[title$=jQuery]').css('font-size', '24px');

$('a[title\*=jQuery]').css('text-decoration', 'line-through');

</script>

**4.4 [selector1][selector2]（复合型属性过滤器，同时满足多个条件）**

将title以"jQuery"开始，并且class="item"的a标签隐藏，那么<a href="#" title="jQuery事件大全" class="item">jQuery事件大全</a>会被隐藏

<script type="text/javascript">

$(document).ready(function() {

$('a[title^=jQuery][class=item]').hide();

});

</script>

**5. 子元素过滤选择器**

**5.1 :first-child和:last-child**

:first-child表示第一个子元素，:last-child表示最后一个子元素。

需要大家注意的是，:fisrst和:last返回的都是单个元素，而:first-child和:last-child返回的都是集合元素。举个例子：div:first返回的是整个DOM文档中第一个div元素，而div:first-child是返回所有div元素下的第一个元素合并后的集合。

这里有个问题：如果一个元素没有子元素，:first-child和:last-child会返回null吗？请看下面的代码：

<html xmlns="http://www.w3.org/1999/xhtml" >

<head runat="server">

<title></title>

<script type="text/javascript" src="js/jquery.min.js"></script>

<script type="text/javascript">

$(document).ready(function() {

var len1 = $('div:first-child').length;

var len2 = $('div:last-child').length;

});

</script>

</head>

<body>

<div>

<div>

<div></div>

</div>

</div>

</body>

</html>

也许你觉得这个答案，是不是太简单了？len1 = 2, len2 = 2。但实际确并不是，它们俩都等于3。   
把上面的代码稍微修改一下：

<html xmlns="http://www.w3.org/1999/xhtml" >

<head runat="server">

<title></title>

<script type="text/javascript" src="js/jquery.min.js"></script>

<script type="text/javascript">

$(document).ready(function() {

var len1 = $('div:first-child').length;

var len2 = $('div:last-child').length;

$('div:first-child').each(function() {

alert($(this).html());

});

});

</script>

</head>

<body>

<div>123

<div>456

<div></div>

</div>

</div>

</body>

</html>

结果却是弹出三个alert，只不过最后一个alert里面是空白的。

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030398903.png)

**5.2 :only-child**

当某个元素有且仅有一个子元素时，:only-child才会生效。

<html xmlns="http://www.w3.org/1999/xhtml" >

<head runat="server">

<title></title>

<script type="text/javascript" src="js/jquery.min.js"></script>

<script type="text/javascript">

$(document).ready(function() {

$('div:only-child').css('border', '1px solid #FF0000').css('width','200px');

});

</script>

</head>

<body>

<div>123

<div>456

<div></div>

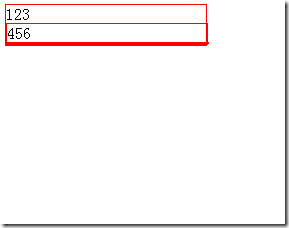
</div>

</div>

</body>

</html>

这里:only-child也是三个元素，从最后一个很粗的红色边框（实际是两个元素的边框重叠了）也可以看出来。

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030431539.png)

**5.3 :nth-child**

看到这个就想起英文单词里的，fourth, fifth, sixth……，nth表示第n个，:nth-child就表示第n个child元素。要注意的是，这儿的n不像eq(x)、gt(x)或lt(x)是从0开始的，它是从1开始的，英文里好像也没有zeroth这样的序号词吧。

**:nth-child有三种用法：**

1) :nth-child(x)，获取第x个子元素   
2) :nth-child(even)和:nth-child(odd)，从1开始，获取第偶数个元素或第奇数个元素   
3) :nth-child(xn+y)，x>=0，y>=0。例如x = 3, y = 0时就是3n，表示取第3n个元素（n>=0）。实际上xn+y是上面两种的通项式。（当x=0,y>=0时，等同于:hth-child(x)；当x=2,y=0时，等同于nth-child(even)；当x=2,y=1时，等同于:nth-child(odd)）

下面的两个例子是针对2)和3)的，1)的例子我就不列举了。

例2：

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030467894.png)

<html xmlns="http://www.w3.org/1999/xhtml" >

<head runat="server">

<title></title>

<style type="text/css">

td {

width: 200px;

height: 32px;

line-height: 32px;

}

</style>

<script type="text/javascript" src="js/jquery.min.js"></script>

<script type="text/javascript">

$(document).ready(function() {

// 偶数行背景红色

$('tr:nth-child(even)').css('background', '#FF0000');

// 奇数行背景蓝色

$('tr:nth-child(odd)').css('background', '#0000FF');

});

</script>

</head>

<body>

<table>

<tr><td>1. NBA 2012季后赛</td></tr>

<tr><td>2. NBA 2011季后赛</td></tr>

<tr><td>3. NBA 2010季后赛</td></tr>

<tr><td>4. NBA 2009季后赛</td></tr>

<tr><td>5. NBA 2008季后赛</td></tr>

<tr><td>6. NBA 2007季后赛</td></tr>

</table>

</body>

</html>

例3（html代码和例2是一样的）：

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030495056.png)

<script type="text/javascript">

$(document).ready(function() {

$('tr:nth-child(3n)').css('background', '#0000FF');

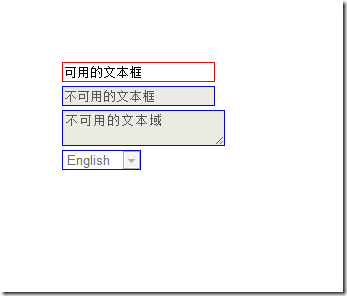
});

</script>

**6. 表单对象属性过滤选择器**

**6.1 :enabled和:disabled（取可用或不可用元素）**

:enabled和:diabled的匹配范围包括input, select, textarea。

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030515697.png)

<script type="text/javascript">

$(document).ready(function() {

$(':enabled').css('border', '1px solid #FF0000');

$(':disabled').css('border', '1px solid #0000FF');

});

</script>

<div>

<input type="text" value="可用的文本框" />

</div>

<div>

<input type="text" disabled="disabled" value="不可用的文本框" />

</div>

<div>

<textarea disabled="disabled">不可用的文本域</textarea>

</div>

<div>

<select disabled="disabled">

<option>English</option>

<option>简体中文</option>

</select>

</div>

**6.2 :checked（取选中的单选框或复选框元素）**

下面的代码，更改边框或背景色仅在IE下有效果，chrome和firefox不会改变，但是alert都会弹出来。

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030537974.png)

<script type="text/javascript">

$(document).ready(function() {

$(':checked').css('background', '#FF0000').each(function() {

alert($(this).val());

});

});

</script>

<div>

<input type="checkbox" checked="checked" value="must"/>必须勾选

</div>

<div>

你现在工作的企业属于：

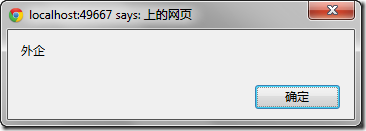
<input type="radio" name="radio" checked="checked" value="外企"/>外企

<input type="radio" name="radio" value="国企"/>国企

<input type="radio" name="radio" value="民企"/>民企

</div>

**6.3 :selected（取下拉列表被选中的元素）**

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030566281.png)

<script type="text/javascript">

$(document).ready(function() {

alert($(':selected').val());

});

</script>

<select>

<option value="外企">外企</option>

<option value="国企">国企</option>

<option value="私企">私企</option>

</select>

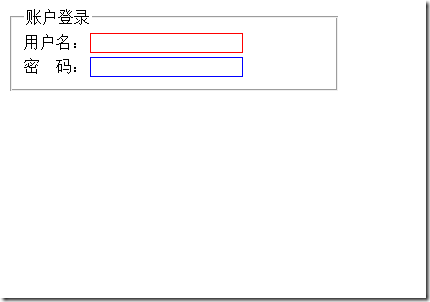
**四、表单选择器**

**1. :input（取input,textarea,select,button元素）**

:input元素这里就不再多说了，前面的一些例子中也已经囊括了。

**2. :text（取单行文本框元素）和:password（取密码框元素）**

这两个选择器分别和属性选择器$('input[type=text]')、$('input[type=password]')等同。

[](http://images.cnblogs.com/cnblogs_com/keepfool/201206/201206022030582985.png)

<script type="text/javascript">

$(document).ready(function() {

$(':text').css('border', '1px solid #FF0000');

$(':password').css('border', '1px solid #0000FF');

// 等效代码

//$('input[type=text]').css('border', '1px solid #FF0000');

//$('input[type=password]').css('border', '1px solid #0000FF');

});

</script>

<fieldset style="width: 300px;">

<legend>账户登录</legend>

<div>

<label>用户名：</label><input type="text"/>

</div>

<div>

<label>密&nbsp;&nbsp;码：</label><input type="password"/>

</div>

</fieldset>

**3. :radio（取单选框元素）**

:radio选择器和属性选择器$('input[type=radio]')等同

<script type="text/javascript">

$(document).ready(function() {

$(':radio').each(function() {

alert($(this).val());

});

// 等效代码

/\*

$('input[type=radio]').each(function() {

alert($(this).val());

});

\*/

});

</script>

你现在工作的企业属于：

<input type="radio" name="radio" checked="checked" value="外企"/>外企

<input type="radio" name="radio" value="国企"/>国企

<input type="radio" name="radio" value="民企"/>民企

**4. :checkbox（取复选框元素）**

:checkbox选择器和属性选择器$('input[type=checkbox]')等同

<script type="text/javascript">

$(document).ready(function() {

$(':checkbox').each(function() {

alert($(this).val());

});

// 等效代码

/\*

$('input[type=checkbox]').each(function() {

alert($(this).val());

});

\*/

});

</script>

您的兴趣爱好：

<input type="checkbox" />游泳

<input type="checkbox" />看书

<input type="checkbox" checked="checked" value="打篮球"/>打篮球

<input type="checkbox" checked="checked" value="电脑游戏"/>电脑游戏

上面的代码，会将所有额checkbox的value输出出来。若你想选择选中项，有三种写法：

$(':checkbox:checked').each(function() {

alert($(this).val());

});

$('input[type=checkbox][checked]').each(function() {

alert($(this).val());

});

$(':checked').each(function() {

alert($(this).val());

});

**5. :submit（取提交按钮元素）**

:submit选择器和属性选择器$('input[type=submit]')等同

**6. :reset（取重置按钮元素）**

:reset选择器和属性选择器$('input[type=reset]')等同

**7. :button（取按钮元素）**

:button选择器和属性选择器$('input[type=button]')等同

**8. :file（取上传域元素）**

:file选择器和属性选择器$('input[type=file]')等同

**9. :hidden（取不可见元素）**

:hidden选择器和属性选择器$('input[type=hidden]')等同