选择器

基本选择器

- ID选择器
  - ID选择器#id就是利用DOM元素的id属性值来筛选匹配的元素，并以jQuery包装集的形式返回给对象。
  - 使用公式：$("#id")
  - 示例：$("#box")　//获取id属性值为box的元素
- 元素选择器
  - 元素选择器是根据元素名称匹配相应的元素。元素选择器指向的是DOM元素的标记名，也就是说元素选择器是根据元素的标记名选择的。
  - 使用公式：$("element")
  - 示例：$("div")　//获取所有div元素
- 类名选择器
  - 类名选择器是通过元素拥有的CSS类的名称查找匹配的DOM元素。在一个页面中，一个元素可以有多个CSS类，一个CSS类又可以匹配多个元素，如果有元素中有一个匹配的类的名称就可以被类名选择器选取到。简单地说类名选择器就是以元素具有的CSS类名称查找匹配的元素。
  - 使用公式：$(".class")
  - 示例：$(".box")　//获取class属性值为box的所有元素
- 复合选择器
  - 使用公式：$("select.selector2,.....,selectorN")
  - 复合选择器将多个选择器（可以是ID选择器、元素选择或是类名选择器）组合在一起，两个选择器之间以逗号“,”分隔，只要符合其中的任何一个筛选条件就会被匹配，返回的是一个集合形式的jQuery包装集，利用jQuery索引器可以取得集合中的jQuery对象。
  - 注意：多种匹配条件的选择器并不是匹配同时满足这几个选择器的匹配条件的元素，而是将每个选择器匹配的元素合并后一起返回。
  - selector1：一个有效的选择器，可以是ID选择器、元素选择器或是类名选择器等。
  - selector2：另一个有效的选择器，可以是ID选择器、元素选择器或是类名选择器等。
  - selectorN：（可选择）任意多个选择器，可以是ID选择器、元素选择器或是类名选择器等。
  - 示例：$("div,#btn")　//要查询文档中的全部的 元素和id属性为btn的元素
- 通配符选择器
  - $("*")　//取得页面上所有的DOM元素集合的jQuery包装集

层次选择器

- ancestor descendant选择器
  - ancestor descendant选择器中的ancestor代表祖先，descendant代表子孙，用于在给定的祖先元素下匹配所有的后代元素。
  - 使用公式：$("ancestor descendant")
  - ancestor是指任何有效的选择器。
  - descendant是用以匹配元素的选择器，并且它是ancestor所指定元素的后代元素。
  - 示例：$("ul li")　//匹配ul元素下的全部li元素
- parent > child选择器
  - parent > child选择器中的parent代表父元素，child代表子元素，用于在给定的父元素下匹配所有的子元素。使用该选择器只能选择父元素的直接子元素。
  - 使用公式：$("parent > child")
  - parent是指任何有效的选择器。
  - child是用以匹配元素的选择器，并且它是parent元素的子元素。
  - 示例：$("form > input")　//匹配表单中所有的子元素input
- prev + next选择器
  - prev + next选择器用于匹配所有紧接在prev元素后的next元素。其中，prev和next是两个相同级别的元素。
  - 使用公式：$("prev + next")
  - prev是指任何有效的选择器。
  - next是一个有效选择器并紧接着prev选择器。
  - 示例：$("div + img")　//匹配 标记后的<img>标记
- prev ~ siblings选择器
  - prev ~ siblings选择器用于匹配prev元素之后的所有siblings元素。其中，prev和siblings是两个相同辈元素。
  - 使用公式：$("prev ~ siblings")
  - prev是指任何有效的选择器。
  - siblings是一个有效选择器并紧接着prev选择器。
  - 示例：$("div ~ ul")　//匹配div元素后面的同辈所有ul元素

过滤选择器

- 简单过滤器
  - 简单过滤器是指以冒号开头，通常用于实现简单过滤效果的过滤器。
  - :first　说明：匹配找到的第一个元素，它是与选择器结合使用的　示例：$("tr:first")　//匹配表格的第一行
  - :last　说明：匹配找到的最后一个元素，它是与选择器结合使用的　示例：$("tr:last")　//匹配表格的最后一行
  - :even　说明：匹配所有索引值为偶数的元素，索引值从0开始计数　示例：$("tr:even")　//匹配索引值为偶数的行
  - :odd　说明：匹配所有索引值为奇数的元素，索引从0开始计数　示例：$("tr:odd")　//匹配索引值为奇数的行
  - :eq(index)　说明：匹配一个给定索引值的元素　示例：$("div:eq(1)")　//匹配第二个div元素
  - :gt(index)　说明：匹配所有大于给定索引值的元素　示例：$("span:gt(0)")　//匹配索引大于0的span元素（注：大于0，而不包括0）
  - :lt(index)　说明：匹配所有小于给定索引值的元素　示例：$("div:lt(2)")　//匹配索引小于2的div元素（注：小于2，而不包括2）
  - :header　说明：匹配如 h1, h2, h3.....之类的标题元素　示例：$(":header")　//匹配全部的标题元素
  - :not(selector)　说明：去除所有与给定选择器匹配的元素　示例：$("input:not(:checked)")　//匹配没有被选中的input元素
  - :animated　说明：匹配所有正在执行动画效果的元素　示例：$("div:animated ")　//匹配正在执行的动画的div元素
- 内容过滤器
  - 内容过滤器就是通过DOM元素包含的文本内容以及是否含有匹配的元素进行筛选。
  - :contains(text)　说明：匹配包含给定文本的元素　示例：$("li:contains('word')")　//匹配含有“word”文本内容的li元素
  - :empty　说明：匹配所有不包含子元素或者文本的空元素　示例：$("td:empty")　//匹配不包含子元素或者文本的单元格
  - :has(selector)　说明：匹配含有选择器所匹配元素的元素　示例：$("td:has(p)")　//匹配表格的单元格中含有<p>标记的单元格
  - :parent　说明：匹配含有子元素或者文本的元素　示例：$("td:parent")　//匹配不为空的单元格，即在该单元格中还包括子元素或者文本
- 可见性过滤器
  - 元素的可见状态有两种，分别是隐藏状态和显示状态。可见性过滤器就是利用元素的可见状态匹配元素的。
  - :visible　说明：匹配所有可见元素
  - :hidden　说明：匹配所有不可见元素　注意：在应用hidden过滤器时，display属性是none以及input元素的type属性为hidden的元素都会被匹配到
- 表单对象的属性过滤器
  - 表单对象的属性过滤器通过表单元素的状态属性（例如选中、不可用等状态）匹配元素
  - :checked　说明：匹配所有选中的被选中元素　示例：$("input:checked")　//匹配所有被选中的input元素
  - :disabled　说明：匹配所有不可用元素　示例：$("input:disabled")　//匹配所有不可用的input元素
  - :enabled　说明：匹配所有可用的元素　示例：$("input:enabled ")　//匹配所有可用的input元素
  - :selected　说明：匹配所有选中的option元素　示例：$("select option:selected")　//匹配所有被选中的选项元素
- 子元素过滤器
  - 子元素选择器就是筛选给定某个元素的子元素，具体的过滤条件由选择器的种类而定。
  - :first-child　说明：匹配所有给定元素的第一个子元素　示例：$("ul li:first-child")　//匹配ul元素中的第一个子元素li
  - :last-child　说明：匹配所有给定元素的最后一个子元素　示例：$("ul li:last-child")　//匹配ul元素中的最后一个子元素li
  - :only-child　说明：如果某个元素是它父元素中唯一的子元素，那么将会被匹配。如果父元素中含有其他元素，则不会被匹配　示例：$("ul li:only-child")　//匹配只含有一个li元素的ul元素中的li
  - :nth-child(index/even/odd/equation)　说明：匹配每个父元素下的第index个子或奇偶元素，index从1开始，而不是从0开始　示例：
    - $("ul li:nth-child(even)")　//匹配ul中索引值为偶数的元素
    - $("ul li:nth-child(3)")　//匹配ul中第3个li元素

属性选择器

- 属性选择器就是通过元素的属性作为过滤条件进行筛选对象。
- [attribute]　说明：匹配包含给定属性的元素　示例：$("div[name]")　//匹配含有name属性的div元素
- [attribute=value]　说明：匹配属性值为value的元素　示例：$("div[name='test']")　//匹配name属性是test的div元素
- [attribute!=value]　说明：匹配属性值不等于value的元素　示例:$("div[name!='test']")　//匹配name属性不是test的div元素
- [attribute*=value]　说明：匹配属性值含有value的元素　示例：$("div[name*='test']")　//匹配name属性中含有test值的div元素
- [attribute^=value]　说明：匹配属性值以value开始的元素　示例：$("div[name^='test']")　//匹配name属性以test开头的div元素
- [attribute$=value]　说明：匹配属性值是以value结束的元素　示例：$("div[name$='test']")　//匹配name属性以test结尾的div元素
- [selector1][selector2][selectorN]　说明：复合属性选择器，需要同时满足多个条件时使用　示例：$("div[id][name^='test']")　//匹配具有id属性并且name属性是以test开头的div元素

表单选择器

- 表单选择器是匹配经常在表单内出现的元素。但是匹配的元素不一定在表单中
- :input　说明：匹配所有的input元素　示例：
  - $(":input")　//匹配所有的input元素
  - $("form :input")　//匹配<form>标记中的所有input元素，需要注意，在form和:之间有一个空格
- :button　说明：匹配所有的普通按钮，即type="button"的input元素　示例：$(":button")　//匹配所有的普通按钮
- :checkbox　说明：匹配所有的复选框　示例：$(":checkbox")　//匹配所有的复选框
- :file　说明：匹配所有的文件域　示例：$(":file")　//匹配所有的文件域
- :hidden　说明：匹配所有的不可见元素，或者type为hidden的元素　示例：$(":hidden")　//匹配所有的隐藏域
- :image　说明：匹配所有的图像域　示例：$(":image")　//匹配所有的图像域
- :password　说明：匹配所有的密码域　示例：$(":password")　//匹配所有的密码域
- :radio　说明：匹配所有的单选按钮　示例：$(":radio")　//匹配所有的单选按钮
- :reset　说明：匹配所有的重置按钮，即type=" reset "的input元素　示例：$(":reset")　//匹配所有的重置按钮
- :submit　说明：匹配所有的提交按钮，即type=" submit "的input元素　示例：$(":reset")　//匹配所有的提交按钮
- :text　说明：匹配所有的单行文本框　示例：$(":text")　//匹配所有的单行文本框

选择器中的注意事项

- 选择器中含有特殊符号的注意事项
  - 含有“.”、“#”、“(”或“)”等特殊字符:
    - 根据W3C规定，属性值中是不能包含这些特殊字符的,但在实际项目应中偶尔也会遇到这种表达式中含有“#”和“]”等特殊字符的情况。这时，如果按照普通方式去处理的话就会出现错误。解决这类错误的方法是使用转义符号将其转义。

```
<div id="mr#soft">sunck good</div>
<div id="mrbook(1)">sunck nice</div>
如果按照普通方式来获取，例如：
$("#mr#soft");
$("#mrbook(1)");
这样是不能正确获取到元素的，正确的写法如下：
$("#mr\\#soft");
$("#mrbook\\(1\\)");
```

  - 属性选择器的@符号问题:
    - 在jQuery升级版本过程中，jQuery在1.3.1版本中彻底放弃了1.1.0版本遗留下的@符号，假如我们使用1.3.1以上的版本，那么不需要在属性前添加@符号。

```
$("div[@name='mingri']");
正确的写法是将@符号去掉，即改为如下形式：
$("div[name='mingri']");
```

- 选择器中含有空格的注意事项
  - 在实际应用当中，选择器中含有空格也是不容忽视的，多一个空格或者少一个空格也会得到截然不同的结果。

```
<div class="name">
    <div style="display: none;">小明</div>
    <div style="display: none;">小王</div>
    <div style="display: none;">小张</div>
    <div style="display: none;" class="name">小甲</div>
</div>
<div style="display: none;" class="name">小明</div>
<div style="display: none;" class="name">小小明</div>
使用如下的jQuery选择器分别获取它们。
<script type="text/javascript">
    var $i_a = $("  name  :hidden");     // 带空格的jQuery选择器
    var $i_b = $("  name:hidden");       // 不带空格的jQuery选择器
    var len_a = $i_a.length;
    var len_b = $i_b.length;
    alert("$(' name  :hidden') = "+len_a);          // 输出4
    alert("$(' name:hidden') = "+len_b);            // 输出3
</script>
以上代码会出现不同的结果，是因为后代选择器和过滤选择器的不同。
```