

jQuery选择的艺术

jQuery简介

jQuery是什么？

- 是一款JavaScript 库



jQuery 是什么？

- 方便地处理HTML、事件、动画等

jQuery 是什么？

- 可以兼容多浏览器



jQuery 是什么？

- 80%以上网站使用

Coverage Totals

Quantcast Top 10k
7,126 of 10,000 **71.3%**

Quantcast Top 100k
73,341 of 100,000 **73.3%**

BuiltWith Top Million
765,718 of 943,546 **81.2%**

Entire Internet
59,985,868 of 363,401,011 **16.5%**



如何安装和选择版本？

下载jQuery

- <http://jquery.com/>



Download jQuery

v3.1.0

[View Source on GitHub →](#)

[How jQuery Works →](#)

压缩和非压缩版本

- Download the compressed, production jQuery
下载压缩版本，用于生产
- Download the uncompressed, development jQuery
下载非压缩版本，用于开发

如何使用？

- `<script src="jquery-3.1.0.js"> </script>`

如何选择版本？

- **V1.x**
- **V2.x**：不兼容IE6-8
- **V3.x**：分为精简版和普通版

使用CDN

- Content Delivery Network 内容分发网络
- <https://code.jquery.com/>

jQuery CDN – Latest Stable Versions

Powered by MaxCDN

jQuery的基本概念

设计哲学

- Write less , do more



结构、表现和行为分离

HTML



结构

Structure

JS



行为

Behavior

CSS



表现

Presentation


```
<td width="100%" height="20" class="f9pt" align="center">  
  <font color="#346F0E">下载</font>  
  <input type="text" name="wd" size="40" maxlength="100"  
    onMouseOver=this.focus() onFocus=this.select()  
    style="margin-bottom:-5px;font-size:16px; height:1.78em;  
      font-family:arial,sans-serif,宋体;padding-top:2px;  
      padding-left:1px">  
</td>
```

test.html

```
<link rel="stylesheet" href="test.css" />
<td class="f9pt">
  <label>下载</label>
  <input type="text" name="wd" id="myInput" maxlength="100" />
</td>
<script src="test.js"></script>
```

test.css ...

test.js ...

\$

- `window.jQuery === window.$`

\$.xxx

- **\$.each(...)**

- **\$.noop**

- **\$.toArray()**

- ...

`$(xxx)`

- `$('#myDiv')`
`$('.myClass')`
- ...
- `$(domObject) → jQuery Object`
- `$(xxx).get(0) → DOM Object`
- `$(xxx)[0] → DOM Object`

链式语法chaining

`$('#divTest').text('Hello, world!')`

`$('#divTest').removeClass('blue')`

`$('#divTest').addClass('bold')`

`$('#divTest').css('color', 'red')`

`$('#divTest')`

`.text('Hello, world!')`

`.removeClass('blue')`

`.addClass('bold')`

`.css('color', 'red')`

`$(document).ready(function(){...})`

- vs window onload

元素的选择

为什么要选择元素？

JavaScript中如何选择元素？

- `document.getElementById()`
- `document.getElementsByName()`
- `document.getElementsByTagName()`
- `document.getElementsByClassName()`

JavaScript中如何选择元素？

- `document.querySelector()`
- `document.querySelectorAll()`

CSS 中如何选择元素？

- **CSS 选择器**

- 基本选择器
- 属性选择器
- 伪类选择器
- 伪元素选择器

| | |
|--------|---|
| * | 通配符选择器，用来选择所有元素，或者某个元素下的所有元素 |
| E | 元素选择器，是很常见的选择器，E代表文档元素，比如div，p等等 |
| .class | 类选择器 |
| #id | id 选择器 |
| E F | 后代选择器, 也被称作包含选择器，就是可以选择某元素的后代元素 |
| E > F | 子元素选择器只能选择某元素的子元素，这里是直接子元素，不包含孙子元素之类的后代元素 |
| E + F | 相邻兄弟选择器，可以选择紧接在另一元素后的元素，而且他们具有一个相同的父元素，也就是，EF两元素具有一个相同的父元素，而且F元素在E元素后面，而且相邻 |
| E ~ F | 通用兄弟元素选择器，他选择某元素后面的所有兄弟元素，和相邻兄弟选择器很像，不过相邻兄弟选择器仅选择与其相邻的后面元素,也就是选中的仅一个元素；而通用兄弟元素选择器，选中的是元素相邻的后面所有兄弟元素 |

| | |
|-------------------|---|
| E[attr] | 只使用属性名，但没有确定任何属性值，就是只要有这个属性就ok |
| E[attr="value"] | 指定了属性名，并指定了该属性的属性值，必须有这个属性，并且属性值为value才行 |
| E[attr*="value"] | 属性值包含value就行 |
| E[attr^="value"] | 选定了属性以value开头的元素，比如把attr写成href，value写成http:// |
| E[attr\$="value"] | 选定了属性以value结尾的所有元素，比如要选择所有png格式的图片，attr写成src，value写成png就行 |
| E[attr~="value"] | 如果属性值包含很多词，并且以空格隔开的话，可以用这种来选择，比如元素的title属性值为hello world，要选中它，就在这里把value写成hello或者world就行，用得比较少 |
| E[attr = "value"] | 选择了属性值是value或者以“value-”开头的值，比如attr写成lang，value写成zh，就包含了zh，zh-cn等等语言的元素，用得比较少 |

| | |
|------------------------|---|
| <code>:link</code> | 选择所有未被访问的链接。 |
| <code>:visited</code> | 选择所有已被访问的链接。 |
| <code>:hover</code> | 选择鼠标指针位于其上的链接。 |
| <code>:active</code> | 选择活动链接。 |
| <code>:focus</code> | 选择获得焦点的 <code>input</code> 元素。 |
| <code>:enabled</code> | 选择每个启用的 <code><input></code> 元素。 |
| <code>:disabled</code> | 选择每个禁用的 <code><input></code> 元素。 |
| <code>:checked</code> | 选择每个被选中的 <code><input></code> 元素。 |

| | |
|----------------------------------|---|
| <code>:first-child</code> | 选择某个元素的第一个子元素 |
| <code>:last-child</code> | 选择某个元素的最后一个子元素 |
| <code>:nth-child()</code> | 选择某个元素的一个或多个特定的子元素, 括号里可以写整数值, 比如1, 2, 也可以写表达式, 注意这里的元素是从1开始, 并不是从0开始 |
| <code>:nth-last-child()</code> | 选择某个元素的一个或多个特定的子元素, 括号里面的写法和nth-child一样, 不过是从这个元素的最后一个子元素开始算起 |
| <code>:nth-of-type()</code> | 类似于:nth-child, 不同的是他多了个type, 也就是只在符合type的子元素中进行计算 |
| <code>:nth-last-of-type()</code> | 和 nth-last-child()类似, 也是从最后一个子元素开始算起 |
| <code>:first-of-type</code> | 选择一个上级元素下的第一个同类子元素 |
| <code>:last-of-type</code> | 选择一个上级元素的最后一个同类子元素 |
| <code>:empty</code> | 选择的元素里面没有任何内容, 这里没有内容指的是一点内容都没有, 哪怕是一个空格也都没有 |
| <code>:not()</code> | 否定选择器, not后面跟着一个选择器, 表示选择和括号里的选择刚好相反 |

| | |
|-----------------------------|------------------------------------|
| <code>::first-line</code> | 用于选取指定选择器的首行 |
| <code>::first-letter</code> | 用于选取指定选择器的首字母 |
| <code>::before</code> | 在被选元素的内容前面插入内容。 |
| <code>::after</code> | 在被选元素的内容后面插入内容。 |
| <code>::selection</code> | 应用于文档中被用户高亮的部分（比如使用鼠标或其他选择设备选中的部分） |

jQuery中如何选择元素？

jQuery选择器简介

- 兼容CSS3选择器标准
- 对选择器语法有更多扩充
- 返回0、1或多个jQuery元素的集合
- 集合内元素顺序和在页面上顺序一致

jQuery中的基本选择器

jQuery中的基本选择器

- ID选择器
- 类选择器
- 元素选择器
- 后代选择器
- 属性选择器

jQuery中的筛选器/过滤器

- 位置筛选器
- 子元素筛选器
- 表单筛选器
- 内容筛选器
- 其他筛选器
- 自定义选择器

位置筛选器

:first

:last

:even

:odd

:eq(n)

:gt(n)

:lt(n)

子元素筛选器

:first-child

:last-child

:first-of-type

:last-of-type

:nth-child()

:nth-last-child()

:nth-of-type()

:nth-last-of-type()

:only-child

:only-of-type

表单筛选器

:checked

:disabled

:enabled

:focus

:button

:checkbox

:file

:image

:input

:password

:radio

:reset

:selected

:submit

:text

内容筛选器

:empty

:contains(text)

:has(selector)

:parent

其他筛选器

:lang(language)

:not(selector)

:root

:target

:hidden

:visible

:header

:animated

jQuery选择器的性能优化

- 尽量使用CSS中有的选择器
- 避免过度约束
- 尽量以ID开头
- 让选择器的右边有更多特征
- 避免使用全局选择器
- 缓存选择器结果