

标签选择器和id选择器

标签选择器

- ◆ 标签选择器也称元素选择器、类型选择器，它直接使用元素的标签名当做选择器，将选择页面上所有该种标签

选择所有的span

← span {
color: red;
}

选择所有的b

← b {
color: green;
}

标签选择器

- ◆ 标签选择器将选择页面上所有该种标签，无论这个标签所处位置的深浅

```
<p>我们一定<span>不负韶华，只争朝夕</span></p>
<ul>
  <li>小明</li>
  <li>小红</li>
  <li><span>小强</span></li>
</ul>
```

会被选择

会被选择

标签选择器的常见作用

- ◆ 标签选择器“覆盖面”非常大，所以通常用于标签的初始化

```
ul {
  /* 去掉无序列表的小圆点 */
  list-style: none;
}

a {
  /* 去掉超级链接的下划线 */
  text-decoration: none;
}
```

list style none	列表 样式 没有
-----------------------	----------------

text decoration none	文本 装饰 没有
----------------------------	----------------

认识id属性

- ◆ 标签可以有id属性，是这个标签的唯一标识

```
<p id="para1">我是一个段落</p>
```

- ◆ id的名称只能由字母、数字、下划线、短横构成，且不能以数字开头，字母区分大小写，但习惯上一般为小写字母
- ◆ 同一个页面上不能有相同id的标签

id选择器

- ◆ CSS选择器可以使用井号#前缀，选择指定id的标签

```
<p id="para1">我是一个段落</p>
```

选择id为para1的标签

```
#para1 {  
    color: red;  
}
```

class选择器

class类名

- ◆ class属性表示“类名”

<p class="warning">我是段落</p>

- ◆ 类名的命名规范和id的命名规范相同

class选择器

- ◆ 使用点.前缀选择指定class的标签

选择class为warning的标签

```
.warning {  
    color: red;  
}
```

class类名非常灵活

- ◆ 多个标签可以为相同类名

```
<p class="warning">我是段落</p>  
<p class="warning">我是段落</p>  
<ul>  
    <li>我是列表项</li>  
    <li class="warning">我是列表项</li>  
    <li>我是列表项</li>  
    <li class="warning">我是列表项</li>  
</ul>
```


class类名非常灵活

- ◆ 同一个标签可以同时属于多个类，类名用空格隔开

<p class="warning spec">我是段落</p>

同时属于两个类

```
.warning {  
  color: red;  
}  
.spec {  
  /* 文字倾斜 */  
  font-style: italic;  
}
```

我是段落

font	字体
style	样式
italic	斜体

原子类

- ◆ 在做网页项目前，可以将所有的常用字号、文字颜色、行高、外边距、内边距等都设置为单独的类

font	字体
size	尺寸
12px	12像素

```
.fs12 {  
  font-size: 12px;  
}  
.fs14 {  
  font-size: 14px;  
}  
.fs16 {  
  font-size: 16px;  
}
```

```
.color-red {  
  color: red;  
}  
.color-blue {  
  color: blue;  
}  
.color-green {  
  color: green;  
}
```

原子类

- ◆ HTML标签就可以“按需选择”它的类名了，这样可以非常快速的添加一些常见样式

```
<p class="fs18 color-green">我是一个文字</p>
```

我是一个文字

复合选择器

复合选择器

选择器名称	示例	示例的意义
后代选择器	.box .spec	选择类名为box的标签内部的类名为spec的标签
交集选择器	li.spec	选择既是li标签，也属于spec类的标签
并集选择器	ul, ol	选择所有ul和ol标签

后代选择器

- ◆ CSS选择器中，使用空格表示“后代”

选择类名为box的标签
后代的p标签

← .box p {
 color: red;
}

将被选择

```
<div class="box">  
  { <p>我是盒子中的段落</p>  
    <p>我是盒子中的段落</p>  
  </div>  
<p>我是段落</p>  
<p>我是段落</p>
```


后代选择器

◆ “后代” 并不一定是 “儿子”

选择类名为box的标签
后代的p标签

← `.box p {`
`color: red;`
`}`

`<div class="box">`
``

将被选择 `<p>我是盒子中的段落</p>`
`<p>我是盒子中的段落</p>`

``

`</div>`

`<p>我是段落</p>`

`<p>我是段落</p>`

后代选择器

◆ 后代选择器可以有很多空格，隔开好几代

`.box ul li .spec em {`
`color: red;`
`}`

`<div class="box">`

``

``

`<p class="spec">我是段落强调文字</p>`

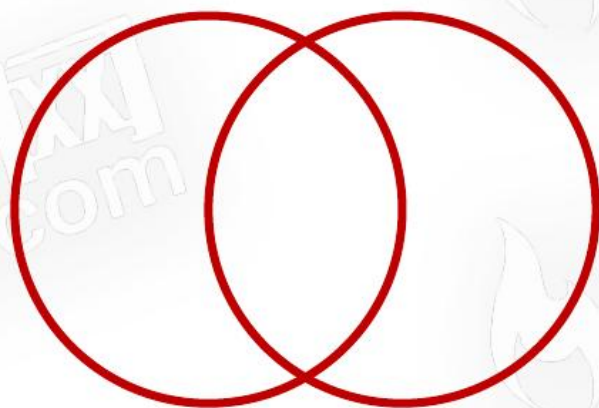
``

``

`</div>`

交集选择器

- ◆ 选择有.spec类的h3标签，此时应该使用交集选择器



```
h3.spec {  
  font-style: italic;  
}
```

并集选择器

- ◆ 并集选择器也叫作分组选择器，逗号表示分组

同时选择ul标签和ol标签

```
ul, ol {  
  list-style: none;  
}
```

复合选择器

- ◆ 选择器可以任何搭配、结合，从而形成复合选择器，我们必须能一目了然的看出选择器代表的含义

```
div.box li p.spec em {  
  color: red;  
}  
  
<div class="box">  
  <ul>  
    <li>  
      <p>我是段落</p>  
    </li>  
    <li>  
      <p class="spec">我是段落<em>强调文字</em></p>  
    </li>  
  </ul>  
</div>
```

伪类

伪类

- ◆ 伪类是添加到选择器的描述性词语，指定要选择的元素的特殊状态，超级链接拥有4个特殊状态

伪类	意义
a:link	没有被访问的超级链接
a:visited	已经被访问过的超级链接
a:hover	正被鼠标悬停的超级链接
a:active	正被激活的超级链接（按下按键但是还没有松开按键）

爱恨准则

- ◆ a标签的伪类书写，要按照“爱恨准则”的顺序，否则会有伪类不生效

LOVE HATE

:link :visited :hover :active



谢谢！

