

08 CSS选择器（三）之伪类和伪元素选择器

更新时间：2019-06-21 16:15:52



“

勤学如春起之苗，不见其增，日有所长。

——陶潜

”

在前面的两节，我们介绍了基础的选择器以及它们的组合方式。这一节我们来介绍另外一类稍微特殊的两种选择器—伪类选择器和伪元素选择器。在页面中，这两类选择器通常用于标记一些特殊的样式，这些样式是基础选择器和组合选择器解决不了的，有些甚至是 **JS** 也做不到的。下面我们就分别介绍一下这两种特殊的选择器。

伪类选择器

在页面中，有时候同一个元素在不同动作下有不同的样式。比如链接在没有点击的时候有个样式，在鼠标放上去有另外的样式，还有在点击完成以后又会又一个样式。这几种情况下这个链接的标签并没有变化，有变化的只是它的状态，这时候就可以用伪类来实现这个需求。在浏览器中，伪类的出现是为了向某些选择器添加特殊的效果或限制。伪类是在正常的选择器后面加上伪类名称，中间用冒号(:)隔开。比如我们希望一个超链接在鼠标放上去的时候有一个下划线：

```
a{
  text-decoration: none;
}
a:hover{
  text-decoration: underline;
}
```

这样设置一个 **a** 标签样式后，就是鼠标放上去会出现下划线，移走后下划线又会消失。

伪类主要有两方面的用处，一方面是标记一些特殊的状态，也就是一开始说的那个例子那种；另外还有一类伪类是有筛选的功能。下面来列举一下这两种伪类里常用的选择器的功能。

一、标记状态的伪类：

可以标记元素状态的伪类有如下几种：

- **:link**，选取未访问过的超链接元素。如果我们注意过搜索引擎的结果的话，它里面的链接只要点过的就会变色，从而标记哪个链接是访问过的。**:link** 这个属性就是用来标识没访问过的链接。
- **:visited**，选取访问过的超链接元素。和第一条相反，**:visited** 是用来标记哪个链接是已经访问过的，防止重复点击。
- **:hover**，选取鼠标悬停的元素。，这个伪类经常用在 PC 端，当鼠标放在一个元素上时，可以用 **:hover** 来控制鼠标悬停的样式。因为在移动端里没有鼠标的概念，所以移动端里很少用这个伪类。
- **:active**，选取点中的元素。这个伪类的作用在刚才提到过了，当我们希望按钮有操作反馈的时候，可以用它来标记操作反馈的样式。当然这个伪类也是可以通用的，并不是只能用在按钮上。
- **:focus**，选取获得焦点的元素。这个伪类用来标识获得焦点的元素，比如搜索框在聚焦的时候有个比较明显的边框，方便用户知道当前在可输入的状态。

二、筛选功能的伪类：

有些伪类也可以有筛选的功能，可以根据元素的特点或者索引来给特定的元素加上样式。常用的有筛选功能的伪类如下：

- **:empty**，选取没有子元素的元素。比如选择空的 **span**，就可以用 **span:empty** 选择器来选择。这里要注意元素内有空格的话也不能算空，不会被这个伪类选中。
- **:checked**，选取勾选状态的 **input** 元素，只对 **radio** 和 **checkbox** 生效。
- **:disabled**，选取禁用的表单元素。
- **:first-child**，选取当前选择器下第一个元素。
- **:last-child**，和 **first-child** 相反，选取当前选择器下最后一个元素。
- **:nth-child(an+b)**，选取指定位置的元素。这个伪类是有参数的，参数可以支持 **an+b** 的形式，这里 **a** 和 **b** 都是可变的，**n** 从0起。使用这个伪类可以做到选择第几个，或者选择序号符合 **an+b** 的所有元素。比如使用 **li:nth-child(2n+1)**，就可以选中 **li** 元素中序号是2的整数倍加1的所有元素，也就是第1、3、5、7、9、2n+1个 **li** 元素。
- **:nth-last-child(an+b)**，这个伪类和 **nth-child** 相似，只不过在计数的时候，这个伪类是从后往前计数。
- **:only-child**，选取唯一子元素。如果一个元素的父元素只有它一个子元素，这个伪类就会生效。如果一个元素还有兄弟元素，这个伪类就不会对它生效。
- **:only-of-type**，选取唯一的某个类型的元素。如果一个元素的父元素里只有它一个当前类型的元素，这个伪类就会生效。这个伪类允许父元素里有其他元素，只要不和自己一样就可以。

这个项目中会用到具有筛选功能的“**:nth-child(an+b)**”选择器，我们通过使用这个伪类按着网格排布的规律选择出需要加边框的元素。

伪元素选择器

伪元素选择器是用于向某些元素设置特殊效果。伪元素选择器选中的并不是真实的 DOM 元素，所以叫伪元素选择器。伪元素选择器常用的也就下面 5 个：

- **::first-line**，为某个元素的第一行文字使用样式。
- **::first-letter**，为某个元素中的文字的首字母或第一个字使用样式。
- **::before**，在某个元素之前插入一些内容。
- **::after**，在某个元素之后插入一些内容。
- **::selection**，对光标选中的元素添加样式。

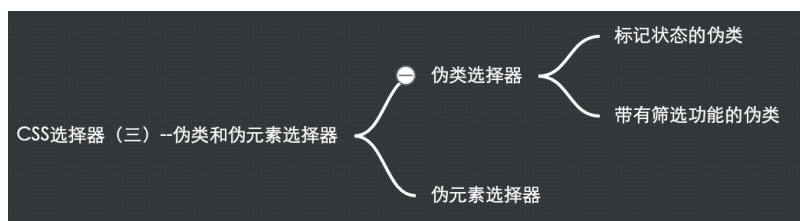
Tips:

1. 伪元素选择器构造的元素是虚拟的，所以不能用 JS 去操作它。
2. 如果同时使用了 **before** 和 **first-letter** 两个伪类，第一个字是要从 **before** 里的内容开始算起的，如果 **before** 里面的内容是一个非文本元素，那 **first-letter** 也会作用在这个非文本元素上，但是不一定能生效。
3. **first-line** 和 **first-letter** 不适用于内联元素，在内联元素中这两个选择器都会失效。
4. 在 CSS3 中，规定了伪类用一个冒号 (:) 表示，伪元素用两个冒号表示 (::)。但除了 **selection**，其余四个伪元素选择器已经在 CSS2 中存在且和伪类用的是一样的单冒号表示的。为了向下兼容，现在的浏览器中伪元素选择器用单冒号和双冒号都可以。在没有兼容问题的情况下，还是建议大家按着新的 CSS3 标准来开发。

小结

这一节讲了伪类选择器和伪元素选择器，这两种选择器都是为了向元素添加特殊效果，但这些效果仅用于显示，不能被 js 操作。选择器这部分里列出来的也都是 CSS3 之前版本定义出来的常用选择器，CSS 选择器相关的文档在 W3C 网站: <https://www.w3.org/TR/selectors/#selectors> 上进行查看。里面还有下一代 CSS 里会出现的内容，有兴趣的同学可以先研究一下。

这一节的内容结构如下：



到这里选择器相关的内容就介绍完了，这部分主要介绍的是各种选择器的概念，一些开发上的技巧，我们会在后面的开发过程中讲解，到时再让同学们慢慢体会。能把选择器运用好，将会省去我们很多的工作量。我们这部分的内容先到这里，下一节开始我们将进入页面渲染机制相关内容的介绍。