

CSS :focus-within伪类选择器及纯CSS下拉等应用举例

这篇文章发布于 2018年01月21日, 星期日, 23:56, 归类于 CSS相关。 阅读 12837 次, 今日 10 次 11 条评论

by zhangxinxu from <https://www.zhangxinxu.com/wordpress/?p=7327>

本文可全文转载, 但需得到原作者书面许可, 同时保留原作者和出处, 摘要引流则随意。

一、了解CSS :focus-within伪类选择器

CSS `:focus-within` 伪类选择器和IE8就开始支持的 `:focus` 可以说是近亲, 区别在于 `:focus` 表示当前元素处于focus状态时候干嘛干嘛, 而 `:focus-within` 表示当前元素或者当前元素的子元素处于focus状态时候干嘛干嘛。

举个例子:

```
form:focus {  
  outline: solid;  
}
```

表示仅仅当 `<form>` 处于focus状态时候, `<form>` 元素 `outline` 出现。

```
form:focus-within {  
  outline: solid;  
}
```

表示 `<form>` 元素自身, 或者 `<form>` 内部的子元素处于focus状态时候, `<form>` 元素 `outline` 出现。换句话说, 子元素 `focus`, 可以让任意父级元素样式发生变化。

这个新特性的支持, 未来势必会带来更加丰富的交互形式, 以及特定场景简化我们实现。

兼容性

兼容性如下截图:

Edge *	Firefox	Chrome	Safari	iOS Safari *	Opera Mini *	Chrome Android
		49		9.3		
		61		10.2		
15		62	10.1	10.3		
16	57	63	11	11.2	all	62
17	58	64	TP			
	59	65				
	60	66				

还是挺喜人的。

二、CSS :focus-within伪类选择器实际应用举例

1. 表单输入勿扰模式

当我们表单输入，选择或者进行交互时候，页面上表单以为其他内容全部都不可见。

CSS代码如下：

```
form {
  outline: 2000px solid hsla(0,0%,100%,0);
  transition: outline .2s;
  position: relative;
  z-index: 1;
}
form:focus-within {
  outline: 2000px solid hsla(0,0%,100%,1);
}
```

效果如下截屏gif：

:focus-within伪类选择器与表单输入勿扰模式

完成您的购物订单

邮箱*	<input type="text"/>
手机号*	<input type="text"/>
您的款式*	<input type="radio"/> 男款 <input type="radio"/> 女款
购买数量*	<input type="text" value="1"/>
所在城市*	<input type="text" value="请选择"/>
给商家留言	<div>少于200字符</div> <div></div>

大屏浏览器体验更有感觉，您可以狠狠的点击这里：[:focus-within伪类选择器与表单输入勿扰模式demo](#)

本案例是真实的勿扰模式效果可以放心大胆在实际项目中使用，因为这是一个体验增强的交互，就算浏览器不支持，对原本功能也不会有任何影响。

2. 带计数文本域的focus高亮

对于带计数的组件化的多行文本域，计数的数值通常是设计在文本域的右下角，这样能够适应各种复杂的场景。如下截图：



然而这种设计对我们的布局实现带来的挑战。

我们通常想到的方法是，计数元素浮在下面的 `textarea` 元素上。然而这种设计存在有致命的不足，那就是输入框的内容有可能和我们的计数重叠，以及出现的滚动条和技术文本重叠，如下截图所示：

您可以狠狠地点击这里：[带计数文本域自身:focus高亮重叠问题demo](#)

因此，我们通常做法就是：边框使用父级 `<div>` 元素模拟，文本域元素和技术元素上下依次排列（非重叠），文本域原本的边框去除。

假设HTML如下：

```
<div class="textarea-x">
  <textarea></textarea>
  <label class="textarea-count">0/250</label>
</div>
```

则核心CSS这样：

```
/* 父级div元素模拟边框 */
.textarea-x {
  border: 1px solid #d0d0d5;
  border-radius: 4px;
  background-color: #fff;
}
/* 文本域原本的边框去除 */
.textarea-x > textarea {
  border: 0;
  background: none;
}
```

然而上面的实现有个非常严重的不足，那就是 `<textarea>` 元素 `:focus` 时候，边框无法高亮，因为CSS中没有父选择器！

因此，实际开发的时候，我们会使用相邻兄弟选择器，以及新建一个兄弟元素来模拟边框。

HTML结构如下：

```
<div class="textarea-x">
  <textarea></textarea>
  <div class="textarea"></div>
  <label class="textarea-count">0/250</label>
</div>
```

原理如下图所示：



对应的核心CSS代码如下：

```
.textarea-x {
  position: relative;
  z-index: 0;
}
.textarea-x > textarea {
  border: 0;
  background: none;
  outline: 0;
  resize: none;
```

```
position: relative;
}
.textarea-x > .textarea {
  position: absolute;
  border: 1px solid #d0d0d5;
  border-radius: 4px;
  background-color: #fff;
  top: 0; bottom: 0; left: 0; right: 0;
  z-index: -1;
}
.textarea-x > :focus + .textarea {
  border-color: #00a5e0;
}
```

由于 `.textarea` 元素和原生的 `<textarea>` 元素是相邻兄弟关系，因此我们可以借助相邻兄弟选择器，让 `<textarea>` 元素focus时候后面用来模拟边框的 `.textarea` 元素高亮。也就是这么一段CSS代码：

```
.textarea-x > :focus + .textarea {
  border-color: #00a5e0;
}
```

这种实现兼容IE7+浏览器。下图为实现后的效果截图：



当然涉及到具体代码，还是有很多细节需要注意的，您可以狠狠地点击这里：[利用相邻兄弟选择器模拟带计数文本域: focus效果demo](#)

但是，现在有了CSS `:focus-within` 伪类选择器，我们的事情就简单多了。

还是开始提到的父级 `<div>` 元素模拟的做法，然后，配合这么一句CSS就可以了：

```
.textarea-x:focus-within {
  border-color: #00a5e0;
}
```

就是这么简单。

您可以狠狠地点击这里：[:focus-within伪类与带计数文本域高亮demo](#)

focus时候效果如下截图：



三、:focus-within更多思路、更棒案例和结束语

`:focus-within` 伪类原本设计的作用是原生表单元素 `focus` 时候，祖先 `<form>` 元素可以也有状态变化。

但是在我看来，`:focus-within` 功能之强悍，远远不是仅仅和祖先 `<form>` 元素玩过家家这么简单。

理论上，只要页面上任意一个元素 `focus`，通过 `:focus-within` 就能对页面上任意的元素进行样式控制。

例如：

```
html:focus-within xxx {  
  /*xxx跑得了吗? 跑不了*/  
}
```

我想到的有：

1. 相邻选择器，加号，或者弯弯都只能选择后面的元素，但是，有了 `:focus-within`，我们就可以对前面元素进行控制，虽然只能在 `:focus` 行为场景下。例如请重复输入密码时候，让上面的输入密码框也高亮。
2. 基于纯CSS和focus行为实现下拉列表的交互成为了可能。只要把按钮和下拉元素放在一个容器中，就可以了。

您可以狠狠地点击这里：[:focus-within伪类与纯CSS下拉列表demo](#)

点击“我的消息”，就会出现下拉列表；此时点击下拉列表里面的条目，下拉元素也一直显示（因为列表是可focus的 `<a>` 元素）。只有点击页面空白元素或其他位置，下拉列表才会隐藏。完全符合实际项目开发的下拉列表交互需求。

交互效果如下GIF截屏所示：



核心代码如下：

```
<div class="details">  
  <a href="javascript:" class="summary">我的消息</a>  
  <div class="box">  
    <a href="javascript:">我的回答<sup>12</sup></a>  
    <a href="javascript:">我的私信</a>  
    <a href="javascript:">未评价订单<sup>2</sup></a>  
    <a href="javascript:">我的关注</a>  
  </div>  
</div>
```

```
.box {  
  display: none;  
}  
.details:focus-within .summary {  
  background-color: #fff;  
}  
.details:focus-within .box {  
  display: block;  
}
```

抛砖引玉，集思广益。

补充于2018-06-27

IE Edge目前最新版还没支持 `:focus-within`，而且，测试表明，当IE支持选择器和 `:focus-within` 一起使用的时候，会自动忽略整条语句，例如：

```
/* IE浏览器忽略所有伪类，CSS语句无效 */  
.example:hover,  
.example:active,
```

```
.example:focus-within {
    color: red;
}
```

需要这样书写：

```
/* IE浏览器可识别 */
.example:hover,
.example:active {
    color: red;
}
/* IE不可识别 */
.example:focus-within {
    color: red;
}
```

这样现象也会我们区分IE浏览器和Chrome/Firefox提供了思路，例如，我们希望某一个元素IE浏览器下是红色，其他浏览器下是蓝色，可以这样：

```
.browser {
    color: red;
}
.browser,
.xxxx:focus-within {
    color: blue;
}
```

感谢阅读，欢迎交流！

《CSS世界》签名版独家发售，包邮，可指定寄语，[点击显示购买码](#)

（本篇完） // 想要打赏？[点击这里](#)。有话要说？[点击这里](#)。



« [借助HTML5 details,summary无JS实现各种交互效果](#)

[CSS改变插入光标颜色caret-color简介及其它变色方法](#) »

猜你喜欢

- [如何在CSS中实现父选择器效果？](#)
- [CSS :placeholder-shown伪类实现Material Design占位符交互效果](#)
- [js下拉菜单实现与可访问性问题的一些思考](#)
- [jQuery powerFloat万能浮动层下拉层插件](#)
- [jQuery之使用slideToggle实现垂直下拉菜单](#)
- [HTML5 datalist在实际项目中应用的可行性研究](#)
- [从height:100%不支持聊聊CSS中的“死循环”](#)
- [CSS3 box-shadow兼容loading效果兼IE10+ CSS Hack介绍](#)
- [折腾：2颗星星+纯CSS实现星星评分交互效果](#)

- 小tip: CSS动态实现文本框清除按钮的隐藏与显示
- CSS3模拟window7炫酷界面效果展示

分享到:         1

标签: :focus-within, css3, focus, 下拉列表, 下拉菜单, 伪类, 兄弟选择器, 文本域, 父选择器, 表单, 选择器

发表评论（目前11条评论）

<input type="text"/>	名称 (必须)
<input type="text"/>	邮件地址(不会被公开) (必须)
<input type="text"/>	网站
<div></div>	
<div>提交评论</div>	

- zzz**说道:
2018年03月15日 21:18

又学到一点东西!

[回复](#)
- 长江长江我是黄河**说道:
2018年02月1日 11:11

```
form:focus {  
outline: solid;  
}
```

这句怎么理解，单独的空的form有focus事件吗，还是需要加“tabindex”? 求解释！

[回复](#)
- 6K**说道:
2018年01月26日 11:47

鑫哥V5

[回复](#)
- 刘东奇**说道:
2018年01月23日 16:58

6666果然神器，之前确实被不能方便的控制父元素样式困扰过，现在有了部分解决办法

[回复](#)

5.

qq说道:

2018年01月23日 09:30

CSS世界 出个kindle 版本吧。

回复


6.

崔某说道:

2018年01月22日 12:23

鑫哥的这篇文章给了我很多启发，我发现: hover, :active 等选择器也有同样的效果，具体效果见demo: <http://47.97.201.13/testTen.html>，纯css实现三级联动菜单效果
域名备案太难，只好用公网地址裸奔QAQ

回复

崔某说道:

2018年01月22日 12:39

demo地址: <http://47.97.201.13/testTen.html>

回复


7.

奔跑的小蜗牛说道:

2018年01月22日 11:16

鑫哥 最后面的那个实例 我的消息 貌似触发不了啊

回复


8.

依韵说道:

2018年01月22日 08:32

文本域元素和技术元素上下依次排列。应该是“计数”

回复


9.

ferrinweb说道:

2018年01月22日 01:34

二-1节错字：“页面上表单以为其他内容全部都不可见”→“页面上表单以外其他内容全部都不可见”。

回复

长江长江我是黄河说道:

2018年02月1日 10:22

细心的网游

回复



最新文章

- » 常见的CSS图形绘制合集
- » 粉丝群第1期CSS小测点评与答疑
- » 分享三个纯CSS实现26个英文字母的案例
- » 小tips: 纯CSS实现打字动画效果
- » CSS/CSS3 box-decoration-break属性简介
- » CSS :placeholder-shown伪类实现Material Design占位符交互效果
- » 从天猫某活动视频不必要的3次请求说起
- » CSS vector-effect与SVG stroke描边缩放
- » CSS ::backdrop伪元素是干嘛用的?
- » 周知: CSS -webkit-伪元素选择器不再导致整行无效

今日热门

- » 常见的CSS图形绘制合集 ⁽¹⁷⁹⁾
- » 粉丝群第1期CSS小测点评与答疑 ⁽¹¹³⁾
- » 未来必热：SVG Sprite技术介绍 ⁽¹¹¹⁾
- » HTML5终极备忘大全（图片版+文字版） ⁽⁸⁵⁾
- » 让所有浏览器支持HTML5 video视频标签 ⁽⁸³⁾
- » Selectivizr-让IE6~8支持CSS3伪类和属性选择器 ⁽⁸⁰⁾
- » CSS3下的147个颜色名称及对应颜色值 ⁽⁷⁸⁾
- » 小tips: 纯CSS实现打字动画效果 ⁽⁷³⁾
- » 写给自己看的display: flex布局教程 ⁽⁷⁰⁾
- » 分享三个纯CSS实现26个英文字母的案例 ⁽⁷⁰⁾

今年热议

- » 《CSS世界》女主角诚寻靠谱一起奋斗之人 ⁽⁷⁶⁾
- » 不借助Echarts等图形框架原生JS快速实现折线图效果 ⁽⁶⁴⁾
- » 看，for..in和for..of在那里吵架！ ⁽⁶⁰⁾
- » 是时候好好安利下LuLu UI框架了！ ⁽⁴⁷⁾
- » 原来浏览器原生支持JS Base64编码解码 ⁽³⁵⁾
- » 妙法攻略：渐变虚框及边框滚动动画的纯CSS实现 ⁽³³⁾
- » 炫酷H5中序列图片视频化播放的高性能实现 ⁽³¹⁾
- » CSS scroll-behavior和JS scrollToView让页面滚动平滑 ⁽³⁰⁾
- » windows系统下批量删除OS X系统.DS_Store文件 ⁽²⁶⁾
- » 写给自己看的display: flex布局教程 ⁽²⁶⁾

猜你喜欢

- 如何在CSS中实现父选择器效果？
- CSS :placeholder-shown伪类实现Material Design占位符交互效果
- js下拉菜单实现与可访问性问题的一些思考
- jQuery powerFloat万能浮动层下拉层插件
- jQuery之使用slideToggle实现垂直下拉菜单
- HTML5 datalist在实际项目中应用的可行性研究
- 从height:100%不支持聊聊CSS中的“死循环”
- CSS3 box-shadow兼容loading效果兼IE10+ CSS Hack介绍
- 折腾：2颗星星+纯CSS实现星星评分交互效果
- 小tip: CSS动态实现文本框清除按钮的隐藏与显示
- CSS3模拟window7炫酷界面效果展示