

HTML accesskey属性与web自定义键盘快捷访问

这篇文章发布于 2017年05月15日, 星期一, 01:58, 归类于 [HTML相关](#)。阅读 11676 次, 今日 3 次 [11 条评论](#)

by zhangxinxu from <http://www.zhangxinxu.com/wordpress/?p=6142>

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。

可能很多小伙伴都不知道, 我们只要在HTML代码上设置几个属性, 就可以让浏览器支持我们自定义的快捷访问方式, 而这个神奇的属性就是本文要介绍的, `accesskey` 属性。

一、HTML accesskey属性基本知识

在HTML4.0.1的时候, HTML `accesskey` 属性请可以作用在以下元素上: `<a>`, `<area>`, `<button>`, `<input>`, `<label>`, `<legend>` 以及 `<textarea>` 元素。然后到了HTML5规范的时候, `accesskey` 属性可以作用在任意的元素上, 变成了“全局属性”。

例如作用在 `<div>` 元素上:

```
<div accesskey="1">自定义快捷访问</div>
```

这里的属性值 `'1'` 对应的就是键盘上的数字 `1`, 如果是 `accesskey="a"` 则 `'a'` 对应的就是键盘上的字母 `a`。

需要注意的是, 虽然说 `accesskey` 属性值和键盘相对应, 但并不是说直接按下这个键, 就能快捷访问。浏览器是通过快捷键组合访问的形式进行页面元素的快速访问的。具体的组合方式参见下表(参考自 [MDN文档](#), 有调整):

Windows
Linux
Mac
Firefox
Alt + Shift + key
Control + Alt + key
Internet Explorer
Alt + key
N/A
Google Chrome
Alt + key
Control + Alt + key
Safari
Alt + key
N/A
Control + Alt + key
Opera
同Google Chrome

//xxx: MDN文档显示Opera浏览器是: `Shift` + `Esc` 打开可访问的快捷键定义列表, 此时可以按下对应的 `key` 进行访问。但是根据自己的测试, `Shift` + `Esc` 打开的是浏览器的任务管理器, 和Chrome浏览器一致。我想, MDN文档对Opera的描述应该是Opera还没有采用Blink内核之前的行为吧~

虽然说IE浏览器和Chrome浏览器的组合键是一样的, 但是其交互行为却和其他浏览器不一样, 举个例子:

```
<a href="" accesskey="1">链接</a>
```

在windows操作系统下, 按下 `Alt` + `1`, IE浏览器只是让 `<a>` 元素获得焦点, 但是其他所有浏览器都是直接触发 `click` 行为。在我看来, IE浏览器的这种行为是不友好的, 对于普通的控件元素而言, 还可以获得焦点后回车访问, 但是, 对于类似 `<div>` 这类元素, `accesskey` 属性几乎是没有任何意义的, 因为根本无法通过键盘触发 `<div>` 点击行为。而Chrome等浏览器就没有这个问题, 如下HTML:

```
<div accesskey="3" onClick="this.style.color='red';">测试</div>
```

此时, 我在windows操作系统下, 按下 `Alt` + `3`, 结果“测试”二字变成了红色, 如下截图:



说明直接触发了绑定的 `click` 点击事件。

隐藏的元素是否可以触发`accesskey`快捷访问?

一个元素, 如果CSS `display` 属性的计算值是 `none`, 是无法通过Tab键进行索引聚焦的。那设置的 `accesskey` 快捷访问是否可以访问呢?

根据我的测试, Chrome浏览器和Firefox浏览器是可以的, 元素即使隐藏, 只要设置了 `accesskey` 快捷访问, 按下对应组合键的时候就能触发 `click` 行为。例如:

```
<a href="" accesskey="1" hidden>链接</a>
```

在windows操作系统Chrome浏览器下, 按下 `Alt` + `1`, 页面会直接刷新。

但是对于IE浏览器, 那就比较惨了, 由于隐藏的元素是不能被 `focus` 聚焦的, 于是 `accesskey` 快捷访问也跟着一起完蛋了, 因为IE的组合键触发的只是 `focus` 行为。

多个元素使用相同的`accesskey`属性值会怎样?

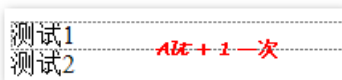
如下测试代码:

```
<div accesskey="1" tabindex="0" onClick="this.style.color='red';">测试1</div>
<div accesskey="1" tabindex="0" onClick="this.style.color='red';">测试2</div>
```

在Chrome浏览器下, 上面一行 `<div>` 的 `accesskey` 属性设置会被忽略, 作用的是后来居上的元素。

在Firefox浏览器下 (版本53), 则是两败俱伤, 两行 `<div>` 进行快捷键匹配的时候都不能触发 `click` 点击事件, 变成了单纯的 `focus` 索引, 行为表现的和IE浏览器一样;

在IE edge下, 原本就不支持 `click` 点击事件, 自然现在也不支持, 使用对应的组合快捷访问的行为也是单纯的依次 `focus` 索引, 如下图示意:



从上面的测试结果可以看出，`accesskey` 属性值相同是一个比较严重的问题，应当极力避免。

二、HTML `accesskey` 自定义快捷键在实际项目中的应用举例

要找一个使用 `accesskey` 自定义快捷键的线上项目很难，其实也很容易。

如果你要想在国内的某个网站上找到对应的应用，很难，说不定反而在一些小众的产品上会应用。说很容易是因为，就算我没去找，我也知道这个网站肯定做了类似的增强处理，那就是社交网站top1 facebook。果不其然，进入主页查看HTML源代码，搜索 `accesskey` 关键字进行匹配，结果有13处匹配，结果如下截图：



比方说：

- 顶部的搜索框设置的是：`accesskey="/"`
- 个人主页链接访问是：`accesskey="2"`
- 首页访问是：`accesskey="1"`
- 新消息是：`accesskey="m"`
- 在 Messenger 中查看所有对话是：`accesskey="4"`
- 所有通知是：`accesskey="5"`

等。

是不是对我们的web网站开发有所启示，简简单单加一个HTML属性，又不要你做任何其他多余的事情，就可以让浏览器支持快捷访问，举手之劳，何乐而不为呢？

几乎所有的网站都有导航，导航都是公用的，是不是可以从 `accesskey="1"` 开始依次设置下；很多网站都有搜索功能，是不是可以像facebook一样，在 `<input>` 输入框HTML上设置 `accesskey="/"`。如果大家都这么设置和处理，久而久之，就会形成约定俗成的行业规范标准，对用户而言其实非常好的一件事情。就好像回车支持表单提交一样的，一旦大家都这么做了，用户就很容易形成习惯，一种更方便快捷高效访问使用网站的习惯，此时如果你的网站不支持，对于多年浸染网络的用户就觉得你这个网站体验做的不好。

理想虽美好，但若想真正实现还比较难，原因就在于：首先组合键访问的方式还是比较啰嗦的，尤其Firefox以及Mac OS X系统下，要同时按下3个键才行，有这么多找键按键的时间，我鼠标早就点过去了；其次，用户如何知道你的网站做了快捷键支持呢？

例如新浪微博网站支持快捷键访问，还是看到微博前端团队一些小伙伴的微博才知道的，如果没有这些同行的微博，估计直到微博倒闭我都不知道做了这样的工作，对于普通用户，那岂不是更无从得知网站支持如此便捷的操作？

因此，要想让 `accesskey` 属性作用发挥得更加彻底，我们还需要做一些额外的其它工作。

三、HTML `accesskey` 自定义快捷键实际应用的进一步增强

花了几个小时，专门为 `accesskey` 属性写了个功能增强辅助JS，以及制作了一个演示页面。

您可以狠狠的点击这里：[accesskey属性实用性增强accesskey.js使用示意demo](#)

该demo页面有两个链接和一个输入框设置了 `accesskey` 属性，分别是：回首页的 `accesskey="1"`，回文章详情页的 `accesskey="2"` 和输入框的 `accesskey="/"`。

下面再来看下这个 `accesskey.js` 做了哪些增强功能呢？

1. 统一IE浏览器和其它浏览器的快捷访问行为，也就是IE浏览器下快捷键也可以直接触发 `click` 行为；

例如，demo页面中，你按下 `Alt + 2` 键，浏览器就直接跳转到文章详情页了，和其他所有浏览器都保持一致了。IE浏览器的原生行为仅仅是 `focus`。

2. 增加单独 `accesskey` 对应按键按下时候的 `focus` 行为，例如 `accesskey="1"`，直接按下键盘数字 `'1'`，就可以让这个元素 `focus`，没有了组合键访问比较啰嗦的问题；

例如，当我们进入demo页面，按下键盘 `2` 的时候，页面表现如下：

此时只要一回车，就能进行链接地址的访问了，比找组合键的方式要方便快捷得多，尤其是Firefox以及Mac OS X系统下的3键齐按的场景。

3. 增加 `accesskey` 生效的前置键按下的提示行为，例如window上 `Alt` 键提示，Mac下的 `Alt` + `control` 组合提示等。有些网站虽然做了 `accesskey` 支持，但是用户根本就不知道哪个键对应哪一个功能，通过这种快捷方式提示，有助于用户即使不看源代码也知道该如何使用快捷键，效果类似下面截图：

4. 增加 `shift + ?` 键的提示行为支持，问号就是帮助的意思，相对于在页面上输入问号。因为快捷前缀的方式记忆成本比较高，不同的操作系统不同的浏览器快捷方式都不一样，而且很有可能跟浏览器本身的快捷方式冲突，因此，这里自定义了一个快捷键提示方式。方便好记兼容性又好，我个人觉得是非常实用的，例如在Firefox浏览器下，假装输入字母 `"?"`，结果就有：

快捷提示以覆盖的形式覆盖在对应的设置了 `accesskey` 属性的元素上，第一次按键出现，再次按键隐藏，按键出现后3秒自动隐藏，任何页面点击行为也会隐藏快捷键提示。

以上就是 `accesskey.js` 对于网站快捷键访问的功能增强支持，具体代码参见demo页面源代码，原生JS书写，兼容IE9+浏览器，直接引入就可以，无需任何初始化。

三、HTML `accesskey`结束语

最近几篇文章都是与键盘访问相关的，本文应该是最后一篇。

很多小伙伴对无障碍访问这种东西都不是很感冒，根据我的观察，业界比较热门，关注度比较高的前端文章，要么是什么资源汇总，要么什么面试技巧，或者就是有Vue字样的，因为和自己的切身利益走得比较近，本文内容注定不温不火，但它有价值吗？有，但只对一小部分的前端。

能参与千万用户数量及产品的前端毕竟是少数，如果产品用户基数很少，则无障碍的东西价值无法能够

得到很好的体现；另外一个也是很重要的，就是，对大部分企业和团队，能够把看得见部分的功能和体验做得及格就已经非常不错了，根本就没有额外的时间和精力关注看不见的体验部分。所谓好刀用在刀刃上，明辨轻重缓急，先把更加重要的部分做好，才是优先考虑的。

但是对于用户基数庞大的产品，例如Facebook亿万级用户，则各类无障碍的东西就显得非常重要了。因为就算只有万分之一的用户习惯使用键盘，那也是上万的用户群体啊！

而往往用户基数庞大的产品，都是福利好，牛人多，不差钱的公司产品，而往往这些公司就是那些关注“资源汇总，面试技巧，或是Vue”小伙伴迫不及待想进入的公司，而这些公司又非常希望能招到一些在无障碍访问有深入研究的人。

所以诸位看出问题来了没有，对你当前的公司而言，键盘无障碍这种东西价值并不大，估计你也没有什么机会使用；但是对于你个人而言，对你未来的发展的好处却是不言而喻，想进入大公司好企业，说不定这方面的知识（深入，非了解皮毛）的加分能力要比你研究面试技巧，学点流行技术要强好几个等级。

大浪淘沙，这些看似不温不火的基础知识，说不定反而才是更经得起历史考验的！

感谢阅读！

《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](#)。



« [HTML tabindex属性与web网页键盘无障碍访问](#)

[使用electron构建跨平台Node.js桌面应用经验分享](#) »

猜你喜欢

- [HTML tabindex属性与web网页键盘无障碍访问](#)
- [实力科普：为什么浮层或弹框一定要有叉叉关闭按钮？](#)
- [是时候好好安利下LuLu UI框架了！](#)
- [借助HTML5 details,summary无JS实现各种交互效果](#)
- [网页制作辅助工具-jQuery标尺参考线插件](#)
- [js写的打方块打飞机小游戏](#)
- [HTML5块状元素链接以及可用性探讨](#)
- [小tip:如何在Dreamweaver中使用emmet\(ZenCoding\)](#)
- [基于VoiceOver的移动网站无障碍访问实战](#)
- [HTML <area><map>标签及在实际开发中的应用](#)
- [IE6下png背景不透明问题的综合拓展](#)

分享到： 1

标签： [accesskey](#), [focus](#), [html](#), [快捷键](#), [无障碍网页应用](#), [键盘](#)

发表评论（目前11条评论）

名称 (必须)

邮件地址(不会被公开) (必须)

网站

提交评论

1. Apparition2018说道：

2018年08月7日 01:13

个人是快捷键爱好者

[回复](#)



2. 杨永全说道：

2017年07月5日 11:26

在国内的盲人网站中，accesskey 是最重要的一个操作能力。他的作用恰恰是为了 focusable ，而不是要 click ，而且是大量重复使用某一个字符。比如 accesskey="x"。

这是因为正好利用 IE 的这个不 click 的特性来快速切换和聚焦。例如在文章列表页面，快速浏览文章标题，这样可以略过发布时间、作者等相对不重要的信息。

当然，在 chrome 等浏览器这个用法就不可以了。

[回复](#)



3. 三爷说道：

2017年06月28日 12:13

基础才是最重要的

[回复](#)



4. 三叔说道：

2017年06月28日 12:12

回归基础

[回复](#)



5. 云叔说道：

2017年06月1日 17:42

accesskey属性的值只能是0-9、A-z么，支不支持keycode

[回复](#)



6. peng说道：

2017年05月18日 11:36

在chrome狂闪。



做过win32程序的都知道，在按钮后面和菜单后面，括号里的字母就是accesskey，已经是约定俗成的事，只是到了WEB这里才打了酱油

[回复](#)

7. **bestRenekton**说道：

2017年05月16日 17:55

冷门英雄助你轻松上分...

[回复](#)



8. **z**说道：

2017年05月16日 10:36

大神呀，方便在网站导航上加上“前端技术-HTML”么~~

[回复](#)



张 鑫旭说道：

2017年05月16日 11:02

可以的，我晚上回去调整下~

[回复](#)



9. **eddiexxx**说道：

2017年05月15日 16:45

总在犹豫要不要学些新东西，感觉不学就脱节了，可是每次看到你的文章，就不那么想了，哈哈...

[回复](#)



10. **zhangolve**说道：

2017年05月15日 08:49

最后说得好，前端说到底，最后拼的还是基础还是老三样。

话说，我这算个沙发吧。哈哈。

[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](#) ⁽¹⁹⁰⁾
- » [未来必热：SVG Sprite技术介绍](#) ⁽¹¹⁹⁾
- » [粉丝群第1期CSS小测点评与答疑](#) ⁽¹¹⁵⁾
- » [HTML5终极备忘大全（图片版+文字版）](#) ⁽⁹³⁾
- » [让所有浏览器支持HTML5 video视频标签](#) ⁽⁸⁶⁾
- » [Selectivizr-让IE6~8支持CSS3伪类和属性选择器](#) ⁽⁸²⁾
- » [CSS3下的147个颜色名称及对应颜色值](#) ⁽⁷⁹⁾
- » [视区相关单位vw, vh..简介以及可实际应用场景](#) ⁽⁷⁶⁾

- » [写给自己看的display: flex布局教程](#) ⁽⁷⁶⁾
- » [小tips: 纯CSS实现打字动画效果](#) ⁽⁷⁶⁾

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 scrollIntoView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

- [HTML tabindex属性与web网页键盘无障碍访问](#)
- [实力科普：为什么浮层或弹框一定要有叉叉关闭按钮？](#)
- [是时候好好安利下LuLu UI框架了！](#)
- [借助HTML5 details,summary无JS实现各种交互效果](#)
- [网页制作辅助工具-jQuery标尺参考线插件](#)
- [js写的打方块打飞机小游戏](#)
- [HTML5块状元素链接以及可用性探讨](#)
- [小tip:如何在Dreamweaver中使用emmet\(ZenCoding\)](#)
- [基于VoiceOver的移动web站无障碍访问实战](#)
- [HTML <area><map>标签及在实际开发中的应用](#)
- [IE6下png背景不透明问题的综合拓展](#)