

小tip: CSS后代选择器可能的错误认识

这篇文章发布于 2012年03月19日, 星期一, 17:15, 归类于 CSS相关。 阅读 59229 次, 今日 8 次 19 条评论

by zhangxinxu from <http://www.zhangxinxu.com>

本文地址: <http://www.zhangxinxu.com/wordpress/?p=2286>

一、关于类选择器的一个问题

假设有下面一个面试题, CSS代码如下:

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

HTML如下:

```
<div class="red"><div class="green"><p>1. 颜色是? </p></div></div>  
<div class="green"><div class="red"><p>2. 颜色是? </p></div></div>
```

问: 第一行和第二行文字颜色分别是?

- ☐ A. 绿 绿 ☐ B. 红 红 ☐ C. 绿 红 ☐ D. 红 绿

这个例子很简单, 我估计基本上都能回答正确: DOM越深的类名权重越高, 因此, 第一行文字红色, 第二行文字绿色。第一行文字绿色, 第二行文字红色! 因此, 上一题答案为C。

您可以狠狠地点击这里: [CSS类名选择器顺序和层级的权重](#)



二、稍作修改: 类选择器→后代选择器

再来一个题目, 如果我们把上例的类选择器改成后代选择器会如何呢?

```
.red p { color: red; }  
.green p { color: green; }
```

HTML维持不变:

```
<div class="red"><div class="green"><p>1. 颜色是? </p></div></div>  
<div class="green"><div class="red"><p>2. 颜色是? </p></div></div>
```

问: 第一行和第二行文字颜色分别是?

☐ A. 绿 绿 ☐ B. 红 红 ☐ C. 绿 红 ☐ D. 红 绿

经验会转化为具象化的认识以帮助我们记忆，然而，这又容易形成固定思维，对一些类似情形造成错误认识。

如果我们按照一开始类选择器的例子来隐射这个类似的例子，您可能就会得到错误的答案——这个例子正确选项不是C。

那正确的选项应该是？

您可以狠狠地点击这里：[CSS后代选择器顺序和层级的权重demo](#)

您可以在各个浏览器下看到，第一行和第二行的文字都是绿色的，如下截图所示：



也就是说正确选项应该是A。

为何？

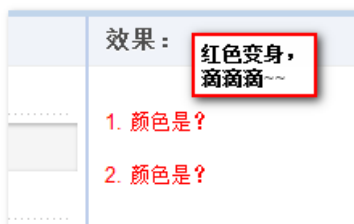
貌似这种后代选择器情形，祖先选择器对应DOM无论多深多浅都是同一级别的，而最终起作用的是选择器声明在CSS样式中的顺序，即优先显示后面的CSS声明。

本例中，`.green p { color: green; }` 在 `.red p { color: red; }` 的后面显示，因此，两行文字都是绿色的。要是我们改成下面：

```
.green p { color: green; }      /* 我在上，酱油命 */  
.red p { color: red; }        /* 我在下，显示我 */
```

那么，两行文字就变身为红色了。

如果您目前手头上的是现代浏览器，可以点击[这里](#)，编辑下CSS代码顺序（FireFox下剪切粘贴有些问题，手动输入），会看到右侧文字果不其然红色了~~



三、再留个问题

改下CSS代码：

```
:not(.green) p { color: red; }
```

```
.green p { color: green; }
```

HTML代码维持不变：

```
<div class="red"><div class="green"><p>1. 颜色是? </p></div></div>
<div class="green"><div class="red"><p>2. 颜色是? </p></div></div>
```

问：第一行和第二行文字颜色分别是？

您可以通过[评论](#)的形式写下您认为的答案~~🤔

《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](#)。



« [WAI-ARIA无障碍网页应用属性完全展示](#)

[介绍：cssrefresh.js-CSS文件自动刷新](#) »

猜你喜欢

- SVG图标颜色文字继承与填充
- MooTools和jQuery库继承实现class切换对比
- 翻译 - CSS继承详解
- CSS项目Minimum Page简单介绍
- 漫谈js自定义事件、DOM/伪DOM自定义事件
- jQuery诞生记-原理与机制
- CSS中height:100%和height:inherit的异同
- 简单了解CSS3的all属性
- 备忘：CSS术语词汇表
- 使用CSS3 ::selection伪元素改变文本选中颜色
- HTML CSS列表元素ul,ol,dl的研究与应用

分享到： 1

标签： [css相关](#), [继承](#), [选择器](#), [颜色](#)

发表评论（目前19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. Evelyn说道:

2017年05月2日 16:10

请教大神个问题:

如今css预处理器sass,less中的选择器支持嵌套形式书写,那么到底是嵌套层级越多越好,还是不要过多为好?

[回复](#)



2. 张家小曼说道:

2014年05月6日 20:13

刚刚试了一下,哪个写在后面就显示哪个颜色
原因嘛,应该是伪类选择器和类选择器权重一样

[回复](#)



3. 一路说道:

2013年02月1日 18:54

1、如果子元素没有指定样式,则直接继承父容器的样式——(赞同5楼赵弟栋所说的)
2、如果子元素指定样式,则对css进行“位”的权重对比,如果权重一样,则进行解析先后顺序对比;
3、:not(.green)(伪选择器)的“位”权重与.green是一样的,所以最后进行解析顺序先后对比了
不知道理解的对否~请指教~~~

[回复](#)



4. DS说道:

2013年01月24日 17:22

额,纠结了

[回复](#)



5. zjx说道:

2012年12月13日 16:42

看到这里,有点困惑。
各种引擎(CSS引擎、JS引擎)是不是按照W3C的规范自行设计渲染、解释程序?
但貌似他们并非都遵循规范,要不也不可能有诸多差异?
既然如此渲染机制是不是都不一样?
渲染机制不同我们怎么可以按照固有的套路理解某种样式规则?
要是真有考官出这种东西,我只能说:您拿台笔记本儿来试试吧,我真不懂。

[回复](#)



6. 美格印象说道:

2012年09月13日 11:13

都是绿哈

[回复](#)



7.

陈清华说道：
2012年07月4日 16:36

都是绿色的，主要看样式谁在谁的后面。对p标签来讲至上而下读取。
[回复](#)


8.

沈大云说道：
2012年07月2日 09:46

是绿红吧 伪类选择器的权重跟class应该是一样重的吧
[回复](#)


9.

王柯说道：
2012年05月16日 00:35

都是红色
[回复](#)


10.

vbyzc说道：
2012年05月7日 16:23

泥玛的这个太难了。
[回复](#)


11.

cyin15288说道：
2012年04月14日 17:50

两行文字都是绿色！
[回复](#)


12.

riophae说道：
2012年03月21日 21:37

都是绿色..
[回复](#)


13.

葵中剑说道：
2012年03月21日 10:34

猜想这篇的灵感来自于 Eric 的 Negative Proximity
[回复](#)

张 鑫旭说道：
2012年03月21日 15:55

@葵中剑 我中枪了，啊~~
[回复](#)




14.

赵弟栋说道：
2012年03月20日 16:02

第一个例子p的样式未指定 样式继承于父级元素div 他的父级元素是指定了的
第二个例子p的样式指定了 两行html和两个css规则都是匹配的 css权重相同的情况下后出现的起作用
第三个 第一个css规则匹配的是类名不是green的元素的后代p 例子中p都是green的后代
[回复](#)



chjund说道:
2012年03月20日 10:02

这个上面讲的很详细，可以参考下：<http://w3help.org/zh-cn/kb/005/>

[回复](#)



2012年03月19日 19:09

两行都是绿的，我觉得: `not(.green) p { color: red; }`可以直接理解成: `p { color: red; }`，所以P应该是一个全局定义，而.green p是后代选择器，这样就出现了两行都是绿色文字！



回复

2012年03月19日 17:21

因此，第一行文字红色，第二行文字绿色！

这算是笔误吗？

[回复](#)

2012年03月19日 17:30

@akasuna 确实是很低级的错误，估计是自己饿傻了～

[回复](#)

- » 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

- » 常见的CSS图形绘制合集 (193)
- » 未来必热：SVG Sprite技术介绍 (120)
- » 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 (80)
- » 视区相关单位vw, vh...简介以及可实际应用场景 (77)
- » 写给自己看的display: flex布局教程 (76)
- » 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

- [SVG图标颜色文字般继承与填充](#)
- [MooTools和jQuery库继承实现class切换对比](#)
- [翻译 - CSS继承详解](#)
- [CSS项目Minimum Page简单介绍](#)
- [漫谈js自定义事件、DOM/伪DOM自定义事件](#)
- [jQuery诞生记-原理与机制](#)
- [CSS中height:100%和height:inherit的异同](#)
- [简单了解CSS3的all属性](#)
- [备忘：CSS术语词汇表](#)
- [使用CSS3 ::selection伪元素改变文本选中颜色](#)
- [HTML CSS列表元素ul,ol,dl的研究与应用](#)