# 目录

[目录 2](#_Toc9487)

[一、晨早测验题目讲解 3](#_Toc31827)

[二、css基础选择器 8](#_Toc7021)

[2.1 标签选择器 8](#_Toc27163)

[2.2 id选择器 8](#_Toc27153)

[2.3 类选择器 9](#_Toc5732)

[三、css高级选择器 11](#_Toc31803)

[3.1 后代选择器 11](#_Toc31732)

[3.2 交集选择器 12](#_Toc21502)

[3.3 并集选择器（分组选择器） 13](#_Toc17774)

[3.4 通配符\* 13](#_Toc18148)

[四、一些CSS3选择器 14](#_Toc14814)

[4.1 兼容问题介绍 14](#_Toc20683)

[4.2 儿子选择器> 15](#_Toc31399)

[4.3 序选择器 15](#_Toc13978)

[4.4 下一个兄弟选择器 16](#_Toc1498)

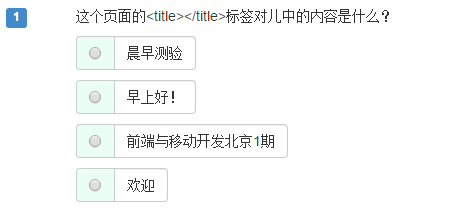
[五、CSS的继承性和层叠性 17](#_Toc16324)

[5.1 继承性 17](#_Toc18489)

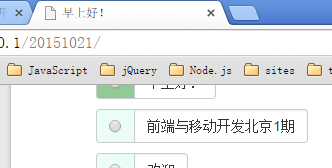
[5.2 层叠性 18](#_Toc17582)

# 一、例题讲解

第1题 考察title标签

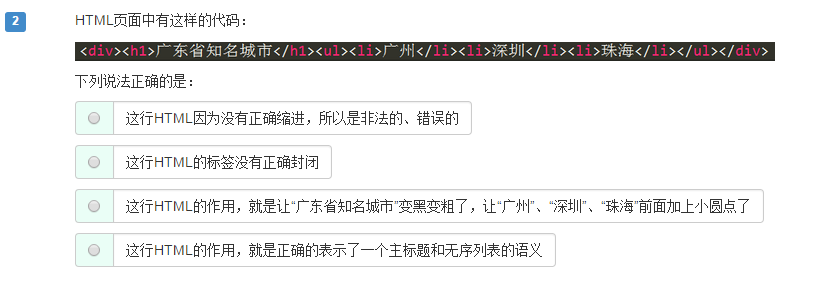


<title>出现在<head>里面，表示页面的标题。直观上，我们可以在浏览器的标题栏（标签栏）中看见。



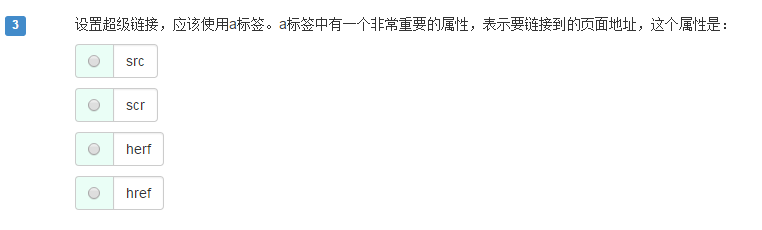
B正确。

第2题 考察的就是HTML的本质



HTML只依靠标签对儿表达语义，和是否缩进、换行没有任何关系！只要是标签有正确的嵌套关系，正确的父子关系，那么就是一个合法的HTML结构，不一定要缩进。百度的首页，为了减小文件体积，所以都是没有缩进的。**提到HTML的作用，只能从语义方面从想，绝对不能想样式。**所以C错误的。D正确。

第3题 考察常用属性



|  |
| --- |
| 1. <img src="1.jpg" /> 2. <a href="1.html">点击我</a> |

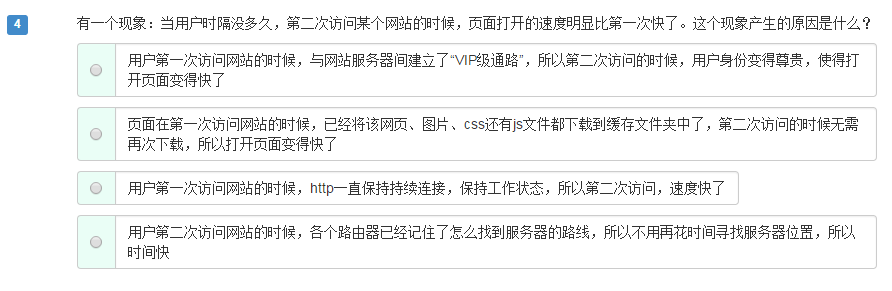
img是**im**a**g**e“图像”；

src是**s**ou**rc**e“资源”；

a是**a**nchor“锚”；

href是**h**ypertext **ref**erence“超文本地址”

第4题 考察HTTP的原理



服务器上有一些文件，html、图片、css、js文件，通过HTTP请求传输到了用户的电脑里面。所以，第二次访问的时候，这些图片就不用传输了（Ajax课上，你将知道这是HTTP 304状态），所以页面变得快。

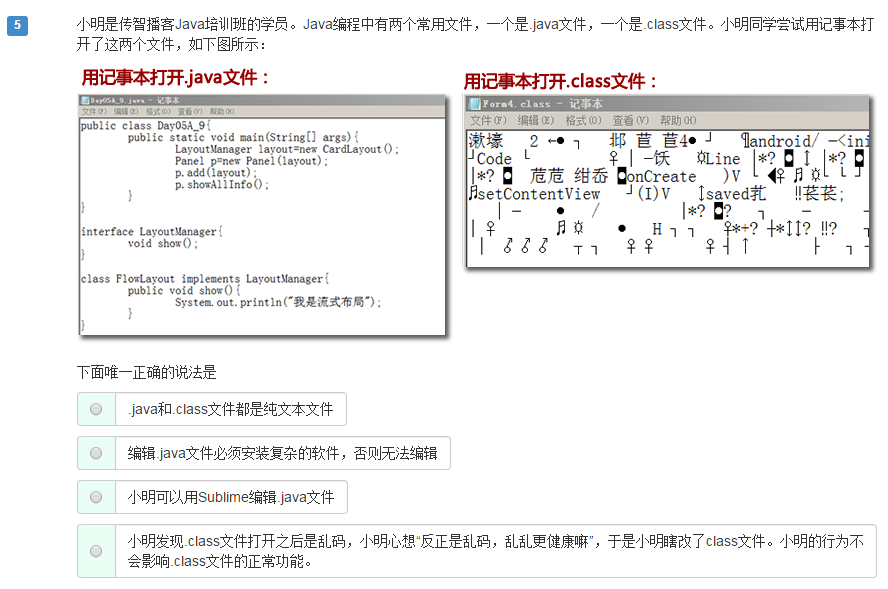
A，错误，没有所谓的vip通路。

B，正确

C，错误。因为HTTP不是一个持久连接的协议，传完就拉倒了，就关闭连接了，所以没有一个持续的通路。

D，错误。每次访问，都是不同的寻址过程，不会“记路”。

第5题 纯文本

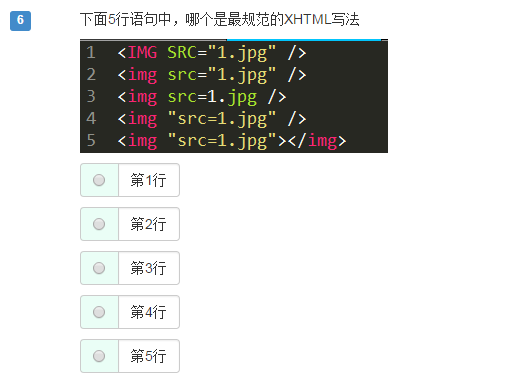


用记事本打开，不是乱码，是可读的，那么一定是纯本文文件。只有文本，没有样式，没有语义。

所以，.java文件是纯文本的, .class文件不是纯文本的。

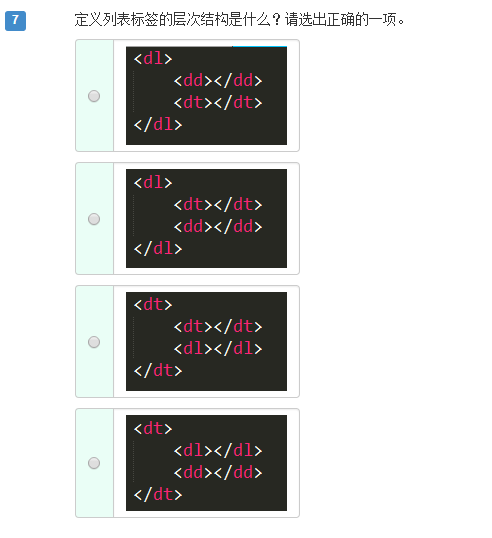
所有的纯文本文件都能用记事本、notepad++、editplus、sublime编辑。

第6题 考察XHTML



所有的标签名、属性都要小写，必须使用引号，必须封闭。答案是B

第7题 考察定义列表



一定要记住每个标签标示什么，就是英语原意是什么？比如

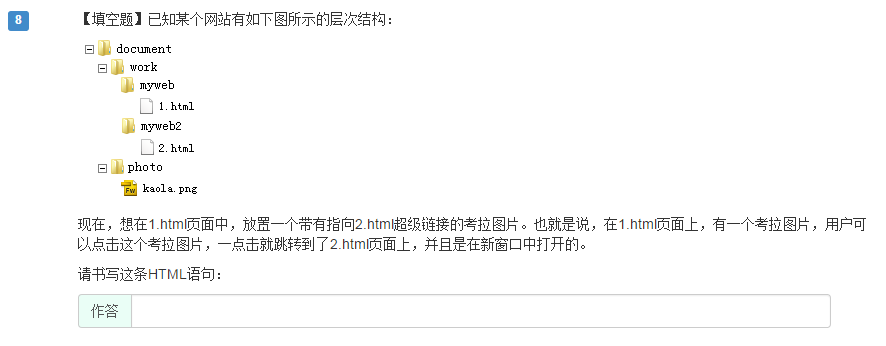
dl 就是definition list， 定义列表；

dt 就是definition title，定义标题；

dd就是definition description，定义描述

第8题 考察相对路径

此题正确：傅宝乐、王丹宇、董家波、张敏、贾久兴、白丽哲、周龙、李婉莹、王博、王佳琦、王飞飞、颜庭光、邵正清、于亚静



1.html中，有一个能够点击的图片。所以骨架：

|  |
| --- |
| 1. <a href=""><img src="" /></a> |

href里面是相对路径，要从1.html出发找到2.html；

src里面也是相对路径，要从1.html出发找到kaola.png。

标准答案：

|  |
| --- |
| 1. <a href="**../myweb2/2.html**" target="\_blank"><img src="**../../photo/kaola.png**" /></a> |

我们一直在用的是相对路径，就是从自己出发找到别人。用相对路径的好处很明显，就是站点文件夹可以拷着走。

**相对路径的好处：站点不管拷贝到哪里，文件和图片的相对路径关系都是不变的。**

**相对路径使用有一个前提，就是网页文件和你的图片，必须在一个服务器上。**

比如，你现在想插入一个新浪网上的图片，那么就不能用相对路径。就要用绝对路径。

**绝对路径非常简单，所有以http://开头的路径，就是绝对路径。**

|  |
| --- |
| 1. <img src="**http://**i1.sinaimg.cn/dy/deco/2013/0329/logo/LOGO\_1x.png" alt="" /> 2. <a href="http://www.sohu.com">点击我跳转到搜狐</a> |

老师，我的网页在C盘，图片却在D盘，能不能插入呢？

答案： **用相对路径不能，用绝对路径也不能。**

注意，可以使用file://来插入，但是这种方法，没有任何意义！因为服务器上没有所谓c盘、d盘。

下面的方法是行的，但是没有任何工程上的意义，这是因为服务器没有盘符，linux系统没有盘符，

|  |
| --- |
| 1. <img src="file://C:\Users\Danny\Pictures\明星\1.jpg" alt="" /> |

总结一下：

我们现在无论是在a标签、img标签，如果要用路径。只有两种路径能用，就是相对路径和绝对路径。

相对路径，就是../ image/ 这种路径。从自己出发，找到别人；

绝对路径，就是http://开头的路径。

绝对不允许使用file://开头的东西，这个是完全错误的！！

# 二、css基础选择器

html负责结构，css负责样式，js负责行为。

css写在head标签里面，容器style标签。

先写选择器，然后写大括号，大括号里面是样式。

|  |
| --- |
| 1. <style type="text/css"> 2. body{ 3. background-color: pink; 4. } 5. </style> |

常见属性：

|  |
| --- |
| 1. h1{ 2. color:blue; 3. font-size: 60px; 4. font-weight: normal; 5. text-decoration: underline; 6. font-style: italic; 7. } |

## 2.1 标签选择器

就是用标签名来当做选择器。

1） 所有标签都能够当做选择器，比如body、h1、dl、ul、span等等

2） 不管这个标签藏的多深，都能够被选择上。

3） 选择的是所有的，而不是某一个。所以是共性，而不是特性。

比如网易，希望页面上所有的超级链接都没有下划线：

|  |
| --- |
| 1. a{ 2. /\*去掉下划线：\*/ 3. text-decoration: none; 4. } |

## 2.2 id选择器

#表示选择id

|  |
| --- |
| 1. #lj1{ 2. font-size: 60px; 3. font-weight: bold; 4. color:black; 5. } |

1）任何的标签都可以有id，id的命名要以字母开头，可以有数字、下划线。大小写严格区别，也就是说mm和MM是两个不同的id。

2）同一个页面内id不能重复，即使不一样的标签，也不能是相同的id。

也就是说，如果有一个p的id叫做haha，这个页面内，其他所有的元素的id都不能叫做haha。

## 2.3 类选择器

**.就是类的符号**。类的英语叫做class。

所谓的类，就是class属性，class属性和id非常相似，**任何的标签都可以携带class属性**。

**class属性可以重复，比如，页面上可能有很多标签都有teshu这个类**：

|  |
| --- |
| 1. <h3>我是一个h3啊</h3> 2. <h3 **class="teshu"**>我是一个h3啊</h3> 3. <h3>我是一个h3啊</h3> 4. <p>我是一个段落啊</p> 5. <p **class="teshu"**>我是一个段落啊</p> 6. <p **class="teshu"**>我是一个段落啊</p> |

css里面用.来表示类：

|  |
| --- |
| 1. **.**teshu{ 2. color: red; 3. } |

**同一个标签，可能同时属于多个类，用空格隔开**：

|  |
| --- |
| 1. <h3 class="**teshu zhongyao**">我是一个h3啊</h3> |

这样，这个h3**就同时属于teshu类，也同时属于zhongyao类**。

初学者常见的错误，就是写成了两个class：

|  |
| --- |
| 1. <h3 **class="teshu" class="zhongyao"**>我是一个h3啊</h3> |

所以要总结两条：

1） class可以重复，也就是说，同一个页面上可能有多个标签同时属于某一个类；

2） 同一个标签可以同时携带多个类。

类的使用，能够决定一个人的css水平。

比如，我们现在要做一个页面：

段落1

段落2

段落3

正确的思路，就是用所谓“公共类”的思路，就是我们类就是提供“公共服务”，比如有绿、大、线，一旦携带这个类名，就有相应的样式变化：

|  |
| --- |
| 1. <style type="text/css"> 2. .lv{ 3. color:green; 4. } 5. .da{ 6. font-size: 60px; 7. } 8. .xian{ 9. text-decoration: underline; 10. } 11. </style> |

每个标签，就去选取自己想要的类：

|  |
| --- |
| 1. <p **class="lv da"**>段落1</p> 2. <p **class="lv xian"**>段落2</p> 3. <p **class="da xian"**>段落3</p> |

也就是说：

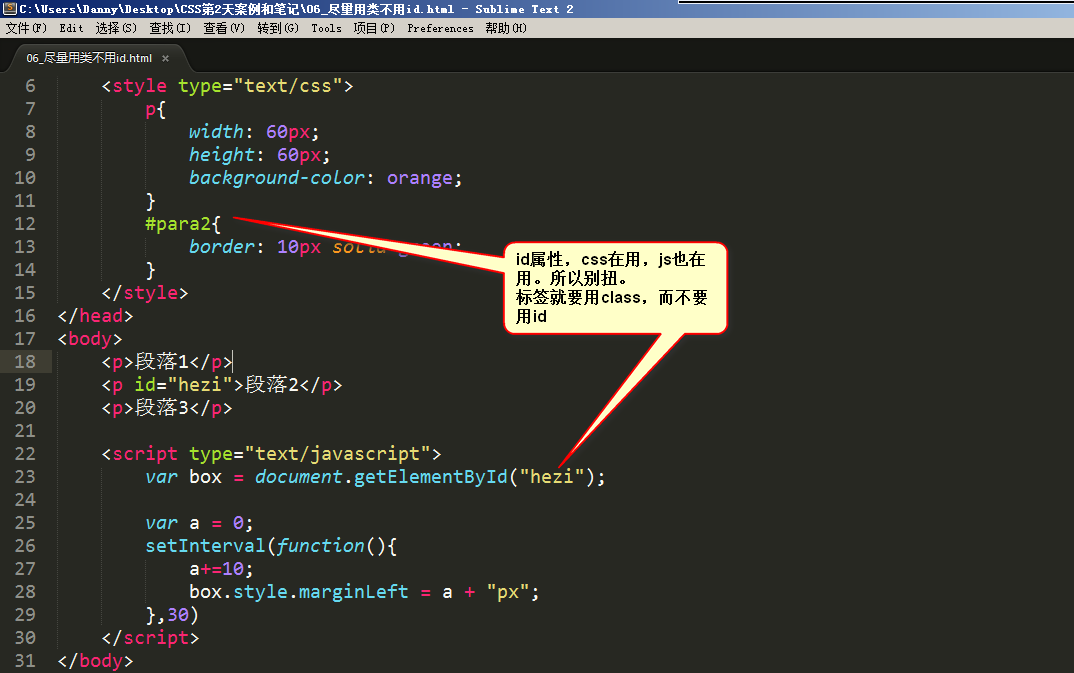
1） 不要去试图用一个类名，把某个标签的所有样式写完。这个标签要多携带几个类，共同造成这个标签的样式。

2） 每一个类要尽可能小，有“公共”的概念，能够让更多的标签使用。

到底用id还是用class？

答案：尽可能的用class，除非极特殊的情况可以用id。

原因：id是js用的。也就是说，js要通过id属性得到标签，所以我们css层面尽量不用id，要不然js就很别扭。另一层面，我们会认为一个有id的元素，有动态效果。



类上样式，id上行为

就是一个标签，可以同时被多种选择器选择，标签选择器、id选择器、类选择器。这些选择器都可以选择上同一个标签，从而影响样式，这就是css的cascading“层叠式”的第一层含义。

# 三、css高级选择器

## 3.1 后代选择器

|  |
| --- |
| 1. <style type="text/css"> 2. **.div1 p**{ 3. color:red; 4. } 5. </style> |

空格就表示后代，.div1 p 就是.div1的后代所有的p。

强调一下，选择的是后代，不一定是儿子。

比如：

|  |
| --- |
| 1. <div class="div1"> 2. <ul> 3. <li> 4. <p>段落</p> 5. <p>段落</p> 6. <p>段落</p> 7. </li> 8. </ul> 9. </div> |

能够被下面的选择器选择上：

|  |
| --- |
| 1. .div1 p{ 2. color:red; 3. } |

所以，看见这个选择器要知道是后代，而不是儿子。选择的是所有.div1“中的”p，就是后代p。

空格可以多次出现。

|  |
| --- |
| 1. .div1 .li2 p{ 2. color:red; 3. } |

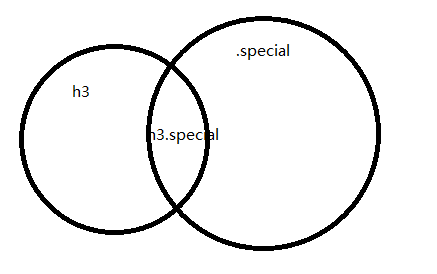
就是.div1里面的后代.li2里面的p。

后代选择器，就是一种平衡：共性、特性的平衡。当要把某一个部分的所有的什么，进行样式改变，就要想到后代选择器。

后代选择器，描述的是祖先结构。



## 3.2 交集选择器



|  |
| --- |
| 1. h3.special{ 2. color:red; 3. } |

选择的元素是同时满足两个条件：必须是h3标签，然后必须是special标签。

交集选择器没有空格。

所以有没有空格

|  |
| --- |
| 1. div.red |

和

|  |
| --- |
| 1. div .red |

不是一个意思。

交集选择器可以连续交（一般不要这么写）

|  |
| --- |
| 1. h3.special.zhongyao{ 2. color:red; 3. } |

交集选择器，我们一般都是以标签名开头，比如div.haha 比如p.special。

## 3.3 并集选择器（分组选择器）

|  |
| --- |
| 1. h3,li{ 2. color:red; 3. } |

用逗号就表示并集。

## 3.4 通配符\*

\*就表示所有元素。

|  |
| --- |
| 1. \*{ 2. color:red; 3. } |

效率不高，如果页面上的标签越多，效率越低，所以页面上不能出现这个选择器。

# 四、一些CSS3选择器

## 4.1 兼容问题介绍

我们现在给大家介绍一下浏览器：

IE： 微软的浏览器，随着操作系统安装的。所以每个windows都有IE浏览器。

windows xp 操作系统安装的IE6

windows vista 操作系统安装的IE7

windows 7 操作系统安装的IE8

windows 8 操作系统安装的IE9

windows10 操作系统安装的edge

浏览器兼容问题，要出，就基本上就是出在IE6、7身上，这两个浏览器是非常低级的浏览器。

http://www1.pconline.com.cn/pcedu/specialtopic/css3-doraemon/

|  |  |  |  |
| --- | --- | --- | --- |
| Chrome浏览器： | Firefox浏览器： | IE8： | IE6： |

浏览器的市场占有率： <http://tongji.baidu.com/data/>

今天：

IE9 5.94%

IE8 21.19%

IE7 4.79%

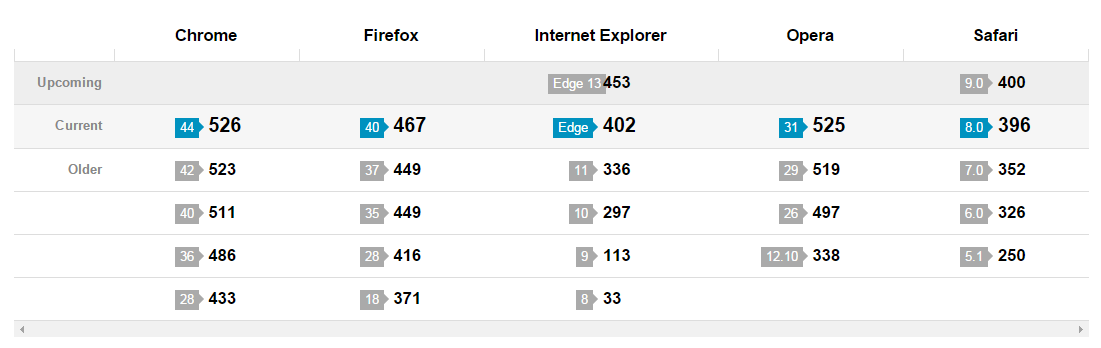
IE6 4.11%

前端与移动学院开发学院对于低级浏览器的教学要求：

学生要知道典型的IE6兼容问题（面试要问），但是做项目我们兼容到IE8。老师不给你解决IE8以下的兼容问题。目的是，培养更高的兴趣和眼光，别天天的跟IE6较劲。

HTML5浏览器打分：

http://html5test.com/results/desktop.html



## 4.2 儿子选择器>

**IE7开始兼容，IE6不兼容。**

|  |
| --- |
| 1. div>p{ 2. color:red; 3. } |

div的儿子p。和div的后代p的截然不同。

能够选择：

|  |
| --- |
| 1. <div> 2. <p>我是div的儿子</p> 3. </div> |

不能选择：

|  |
| --- |
| 1. <div> 2. <ul> 3. <li> 4. <p>我是div的重孙子</p> 5. </li> 6. </ul> 7. </div> |

## 4.3 序选择器

**IE8开始兼容；IE6、7都不兼容**

选择第1个li：

|  |
| --- |
| 1. <style type="text/css"> 2. ul li**:first-child**{ 3. color:red; 4. } 5. </style> |

选择最后一个1i：

|  |
| --- |
| 1. ul li**:last-child**{ 2. color:blue; 3. } |

由于浏览器的更新需要过程，所以现在如果公司还要求兼容IE6、7，那么就要自己写类名：

|  |
| --- |
| 1. <ul> 2. <li **class="first"**>项目</li> 3. <li>项目</li> 4. <li>项目</li> 5. <li>项目</li> 6. <li>项目</li> 7. <li>项目</li> 8. <li>项目</li> 9. <li>项目</li> 10. <li>项目</li> 11. <li **class="last"**>项目</li> 12. </ul> |

用类选择器来选择第一个或者最后一个：

|  |
| --- |
| 1. ul li**.first**{ 2. color:red; 3. } 4. ul li**.last**{ 5. color:blue; 6. } |

## 4.4 下一个兄弟选择器

**IE7开始兼容，IE6不兼容。**

+表示选择下一个兄弟

|  |
| --- |
| 1. <style type="text/css"> 2. h3+p{ 3. color:red; 4. } 5. </style> |

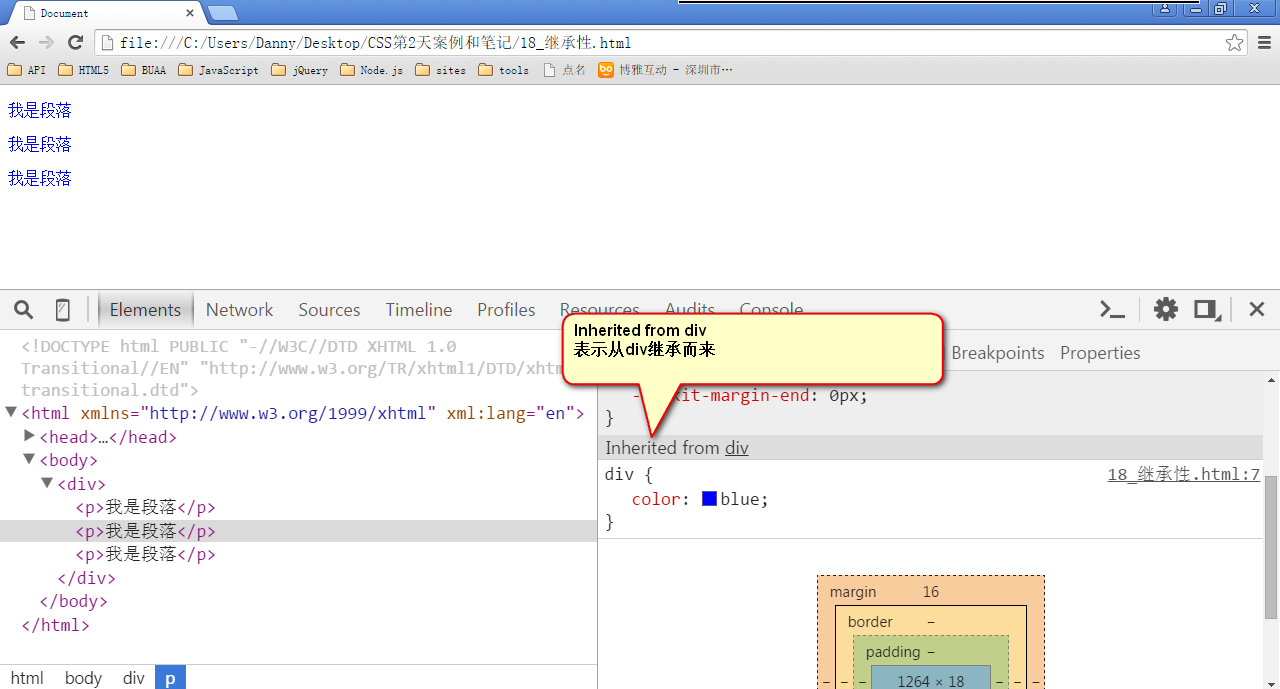
选择上的是h3元素后面紧挨着的第一个兄弟。

|  |
| --- |
| 1. <h3>我是一个标题</h3> 2. **<p>我是一个段落</p>** 3. <p>我是一个段落</p> 4. <p>我是一个段落</p> 5. <h3>我是一个标题</h3> 6. **<p>我是一个段落</p>** 7. <p>我是一个段落</p> 8. <p>我是一个段落</p> 9. <h3>我是一个标题</h3> 10. **<p>我是一个段落</p>** 11. <p>我是一个段落</p> 12. <p>我是一个段落</p> 13. <h3>我是一个标题</h3> |

所有的css3选择器，我们放在HTML5和CSS3课上介绍。

# 五、CSS的继承性和层叠性

## 5.1 继承性



有一些属性，当给自己设置的时候，自己的后代都继承上了，这个就是继承性。

哪些属性能继承？

color、 text-开头的、line-开头的、font-开头的。

**这些关于文字样式的，都能够继承； 所有关于盒子的、定位的、布局的属性都不能继承。**

所以，如果我们的页面的文字，都是灰色，都是14px。那么就可以利用继承性：

|  |
| --- |
| 1. body{ 2. color:gray; 3. font-size:14px; 4. } |

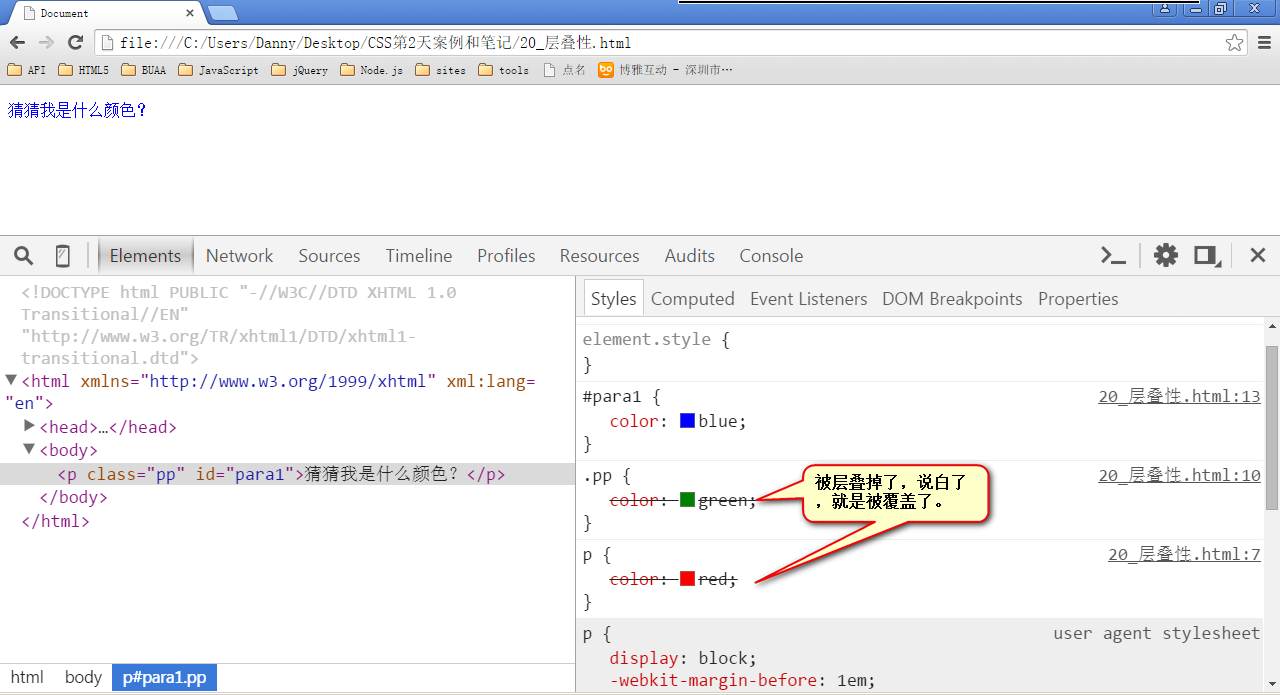
继承性是从自己开始，直到最小的元素。

## 5.2 层叠性

很多公司如果要笔试，那么一定会考层叠性。

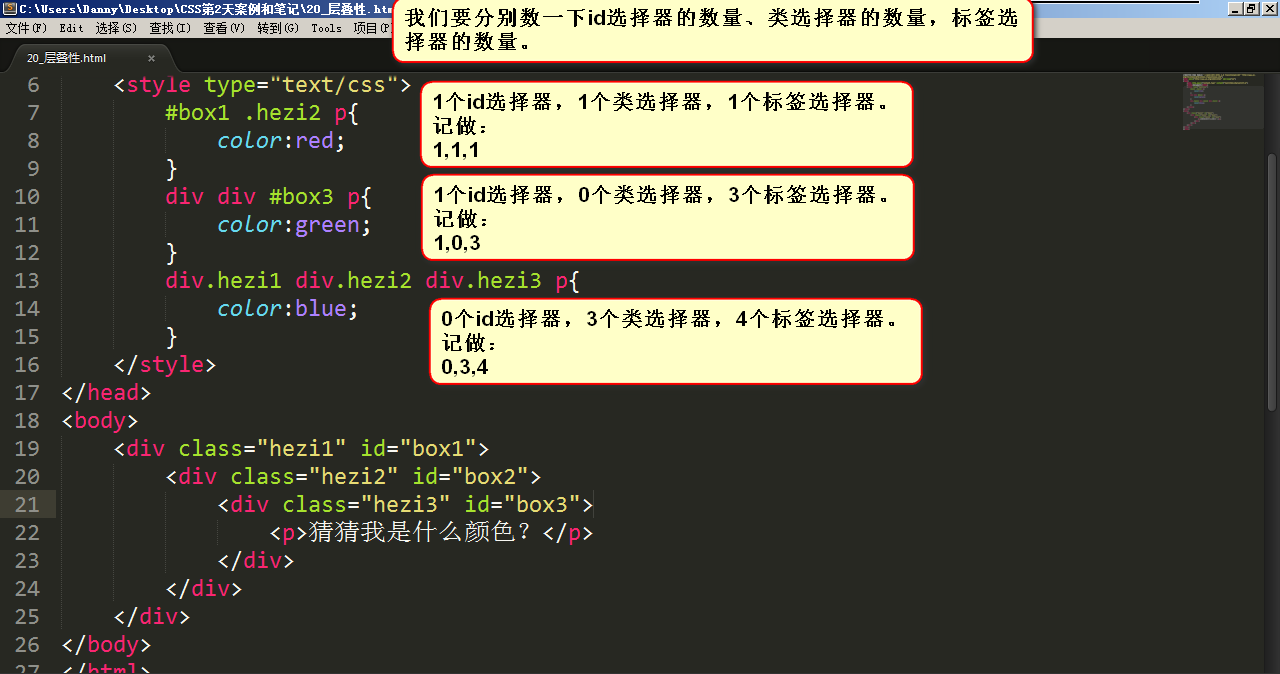
**层叠性：就是css处理冲突的能力。 所有的权重计算，没有任何兼容问题！**

**CSS像艺术家一样优雅，像工程师一样严谨。**

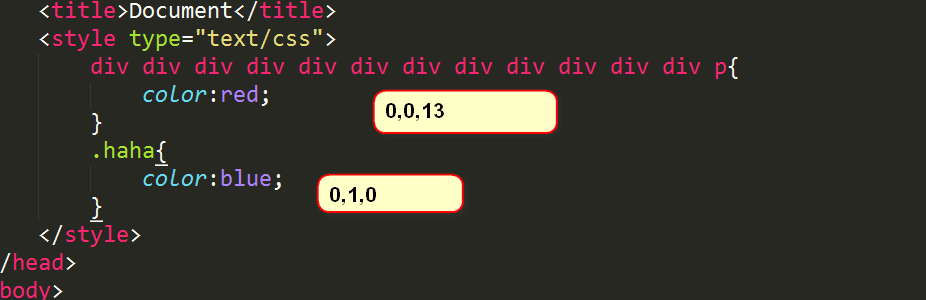


当选择器，选择上了某个元素的时候，那么要这么统计权重：

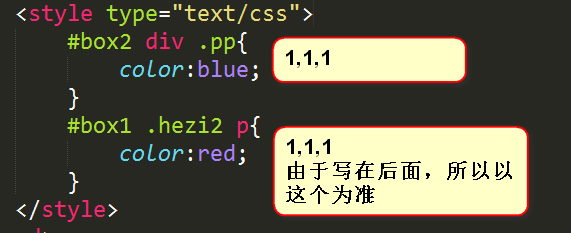
id的数量，类的数量，标签的数量



不进位，实际上能进位（奇淫知识点：255个标签，等于1个类名）但是没有实战意义！



如果权重一样，那么以后出现的为准：



================================我是华丽丽的分界线=================================

如果不能直接选中某个元素，通过继承性影响的话，那么权重是0。



**如果大家都是0，那么有一个就近原则：谁描述的近，听谁的。**



权重问题大总结：

1） 先看有没有选中，如果选中了，那么以（id数，类数，标签数）来计权重。谁大听谁的。如果都一样，听后写的为准。

2） 如果没有选中，那么权重是0。如果大家都是0，就近原则。

题目演示：

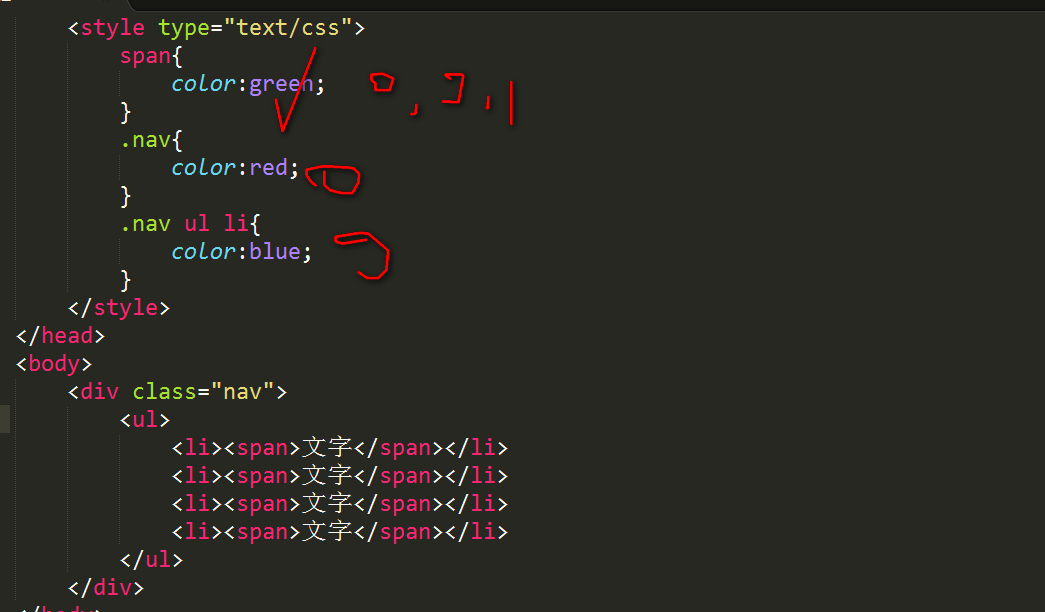
第1题：



第2题：



第3题：



第4题：



css3：盒模型、浮动

css4：定位

css5：背景、表格

css6：切图

复习一下：

下列都是IE6兼容的选择器：

p

#box

.spec

div.box

div .box

div , .box

\*

下列都是IE7开始兼容：

div>p

div+p

下列都是IE8开始兼容：

div p:first-child

div p:last-child

继承性：好的事儿。继承从上到下，哪些能？哪些不能？

层叠性：冲突，多个选择器描述了同一个属性，听谁的？