# CSS入门和高级技巧

第6天课堂笔记（本课程共8天）

# 目录

[CSS入门和高级技巧](#_Toc31912)

[目录](#_Toc15354)

[一、排查错误](#_Toc2682)

[二、博雅互动作业](#_Toc20229)

[三、定位](#_Toc12527)

[3.1 相对定位](#_Toc32126)

[3.2 绝对定位](#_Toc32281)

[3.2.1 定位参考点](#_Toc20962)

[3.2.2 祖先盒子当参考点](#_Toc31849)

[3.2.3 绝对定位元素的居中](#_Toc22302)

[3.2.4 应用](#_Toc5202)

[3.3 固定定位](#_Toc8236)

[四、z-index属性](#_Toc28658)

# 一、排查错误

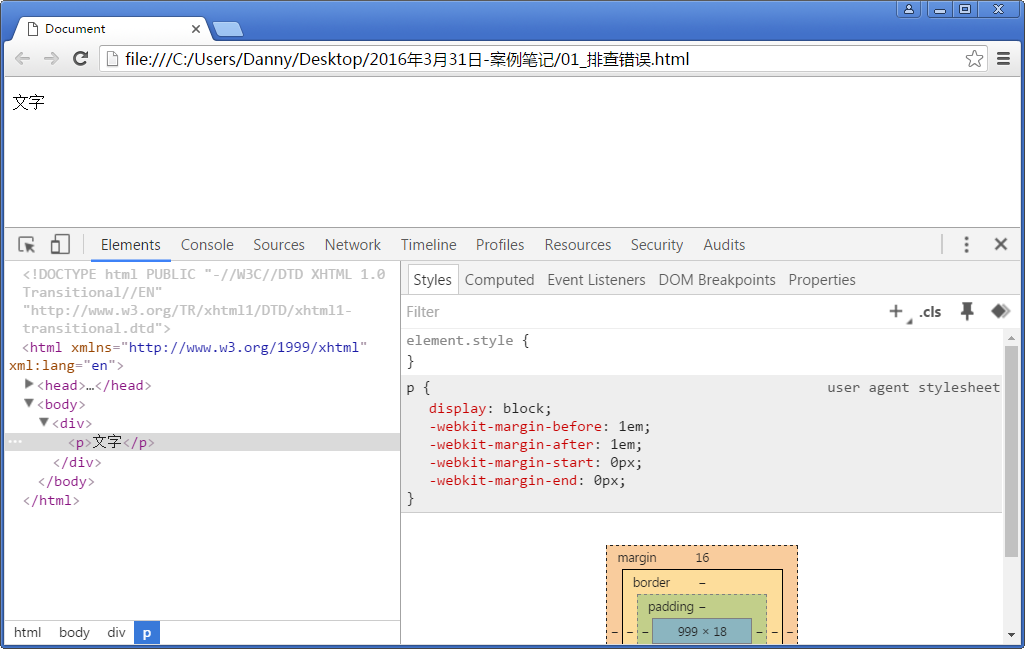
Chrome浏览器的审查功能。

错误1：选择器写错了，压根没有选择上；

如果写了一个错误的选择器，

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

Chrome审查一下，就能看见没有任何选择器选择上这个元素：



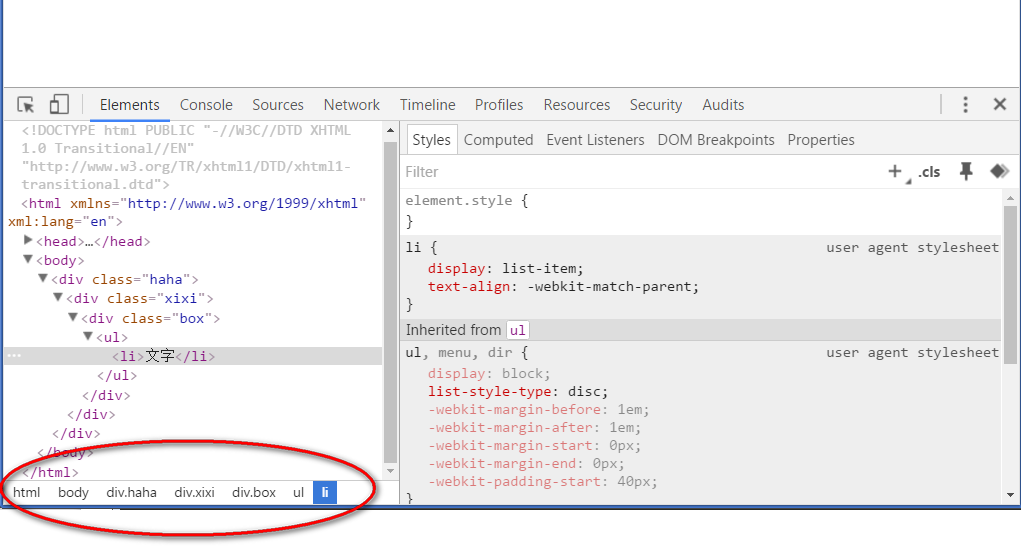
错误的写class：

|  |
| --- |
| 1. <div **calss**="haha"> 2. <p>文字</p> 3. </div> |

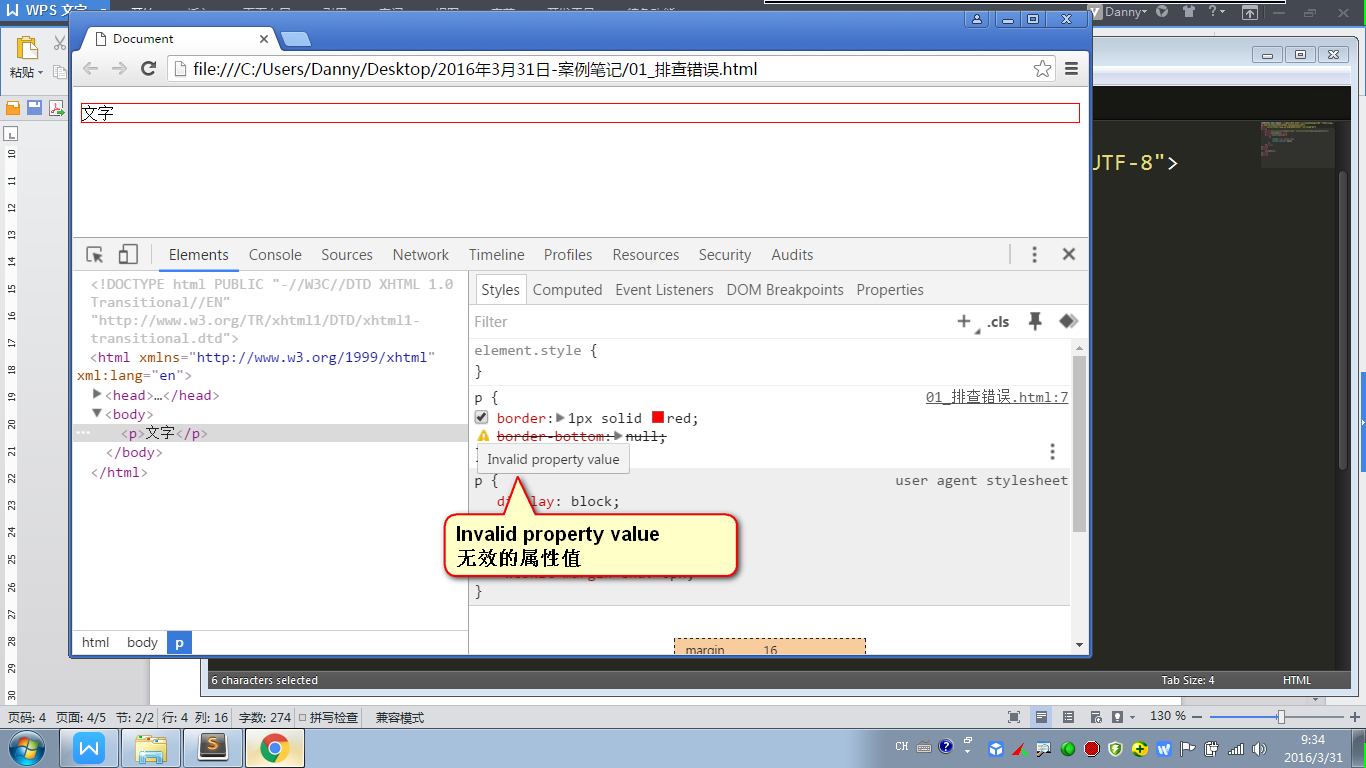
上面一个选择器后面有多余的东西，所以就干扰了下一个选择器：

|  |
| --- |
| 1. body{ 2. background-color: pink; 3. }} 4. div p{ 5. color:red; 6. } |

家谱顺序错误，也是选择不上的。Chrome浏览器底部，是选中的元素的家谱：



错误2：选择上了，但是属性写错了；



# 二、博雅互动作业

版心：就是网页中间居中的部分。

响应式的网站responsive web design，就是网页随着用户屏幕的大小不同，有不同尺寸的版心。

我们现在做的网站，版心都是固定的：1200px、1100px、1000px、980px、970px



padding这个东西，是不能够挤背景图片的，因为padding区域也能有背景，背景图也能渲染在padding里。

到底用<img src=”” />插入， 还是backgrond:url(); ， 要看这个图片是内容的一部分还是杂碎、起装饰作用的图片。

# 三、定位

[position定位？](https://developer.mozilla.org/en-US/docs/Web/CSS/position)

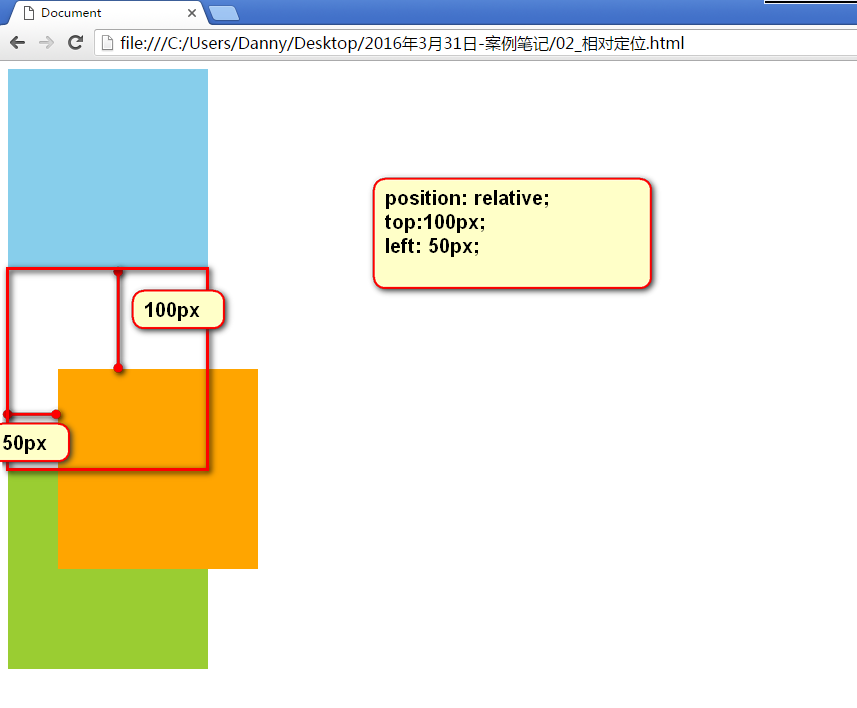
CSS中重要的东西就三个：盒模型、浮动、定位。这三个技术，就是负责网页的布局的。

其余的技术，都是杂碎：background、文字、颜色表示法……

## 3.1 相对定位

|  |
| --- |
| 1. position:relative; |

相对自己进行微调



相对定位就是相对自己原来的位置，进行位置移动。移动的时候需要使用定位值。

top：正数向下移动

bottom： 正数向上移动

left：正数向右移动

right：正数向左移动

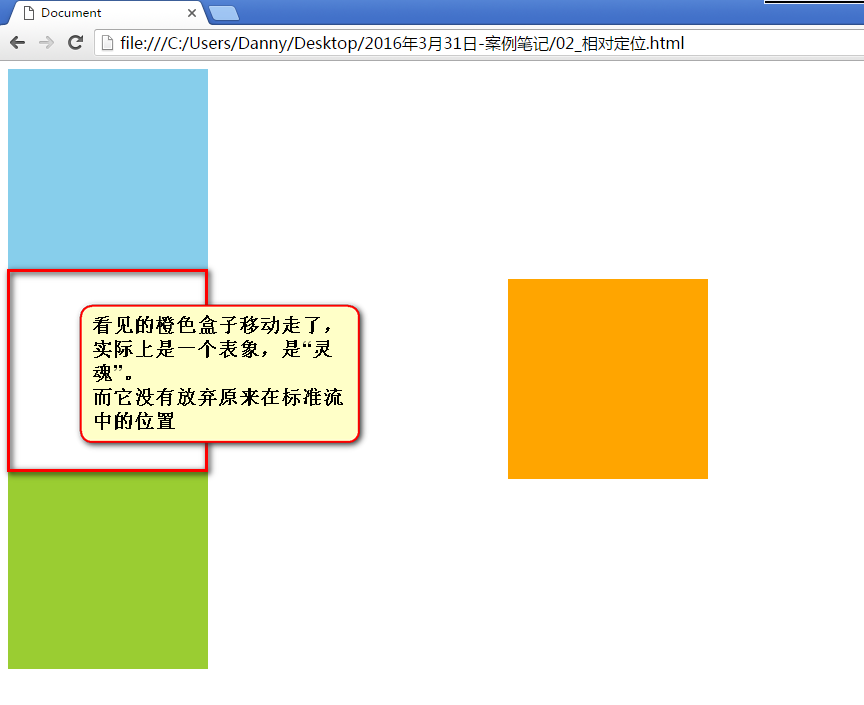
可以任意组合，从top、bottom中选择一个，然后从left、right中选择一个。

|  |
| --- |
| 1. top:10px; 2. left:20px; |

等价于

|  |
| --- |
| 1. bottom:-10px; 2. right:-20px; |

相对定位不脱标，老家留坑，形影分离。



相对定位这个东西，因为在老家留坑，所以用处不大：

1） 微调元素

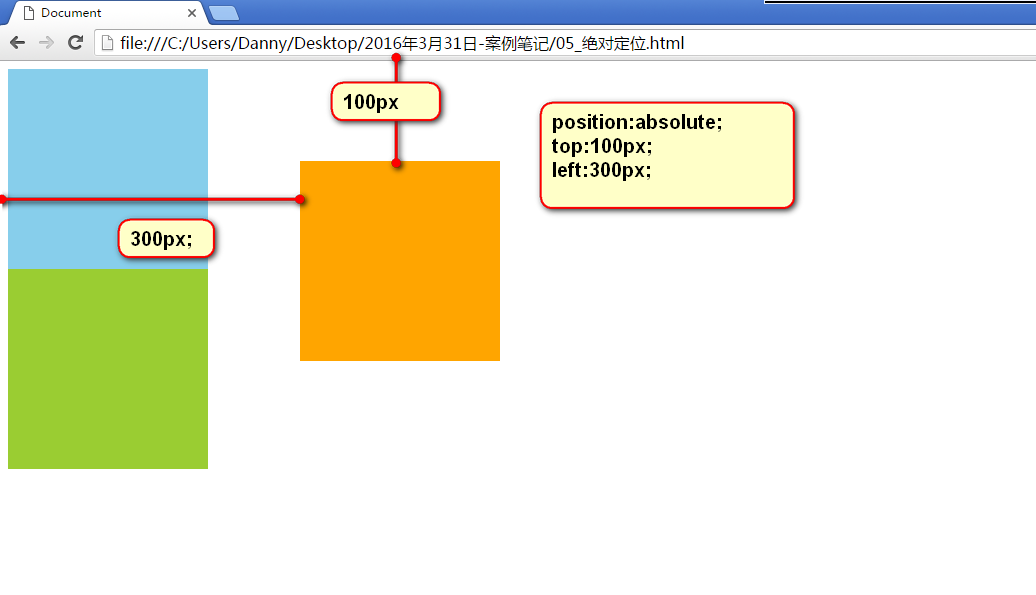
2） 下午我们介绍的“子绝父相”

微调元素：



## 3.2 绝对定位

|  |
| --- |
| 1. position: absolute; |

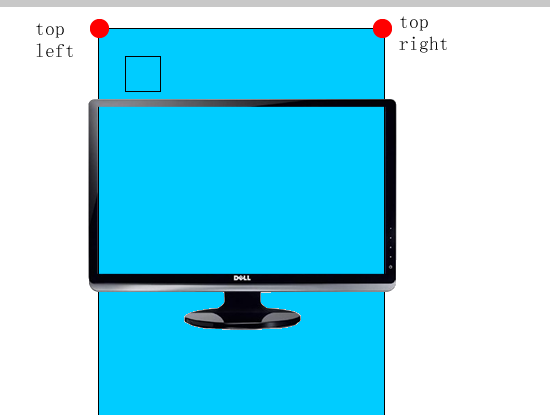


绝对定位的元素，脱离标准文档流。用top、bottom、left、right来进行定位。参考的是页面的左上角。

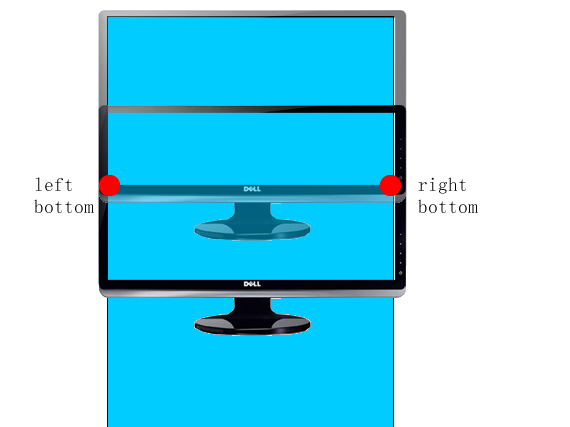
### 3.2.1 定位参考点

我们现在来研究，定位的参考原点到底是哪里？

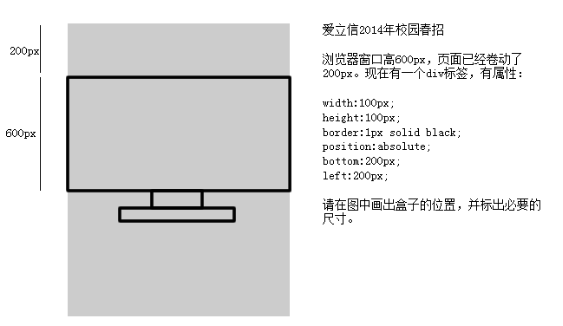
**如果用top值定位，那么参考点就是页面的左上角或者右上角，而不是浏览器窗口的。**



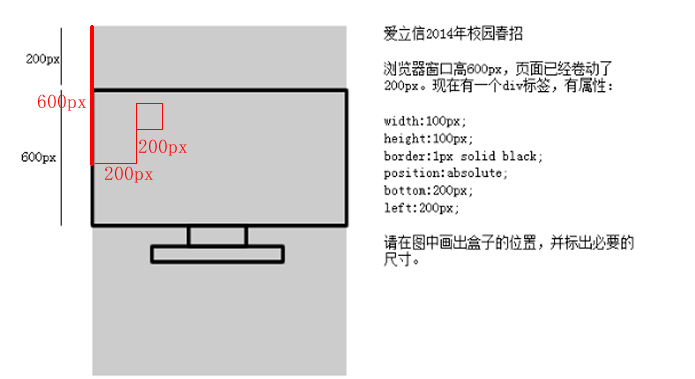
**如果用bottom进行定位，就是浏览器首屏的左下角在页面中的位置。**



面试题：



答案：



绝对定位在老家不留坑，绝对定位的元素就是飘起来了，真实位置也是飘起来了。

所以在页面中特别常用。

### 3.2.2 祖先盒子当参考点

子绝父相：

|  |
| --- |
| 1. <div> → 相对定位 2. <p></p> → 绝对定位 3. </div> |

此时这个p就是以div的边框内侧为参考点进行定位。

一个盒子可以以某一个祖先盒子作为定位参考点，**离自己最近的、已经定位了的祖先盒子。**

要记住，祖先盒子不一定只有相对定位才能成为参考点，而是什么定位都行。比如，相对定位、绝对定位、固定定位。

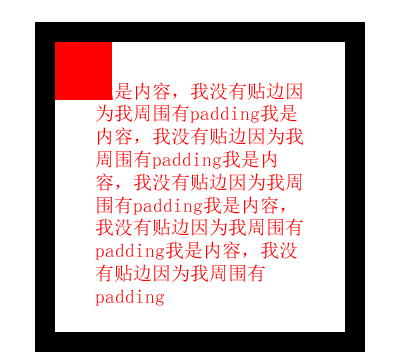
除了“子绝父相”之外，“子绝父绝”也是一个典型的定位模型；“子绝父固”也是一个典型的定位模型。

祖先中离自己最近的、已经定位的元素：

|  |
| --- |
| 1. <div class="box1"> → 没有定位 2. <div class="box2"> → 相对定位 3. <div class="box3"> → 绝对定位 4. <div class="box4"> → 没有定位 5. <p></p> 6. </div> 7. </div> 8. </div> 9. </div> |

所以p以.box3为参考点；.box3以box2为参考点。

绝对定位的元素，无视参考盒子的padding：



红色盒子top:0;left:0; 无视padding的。

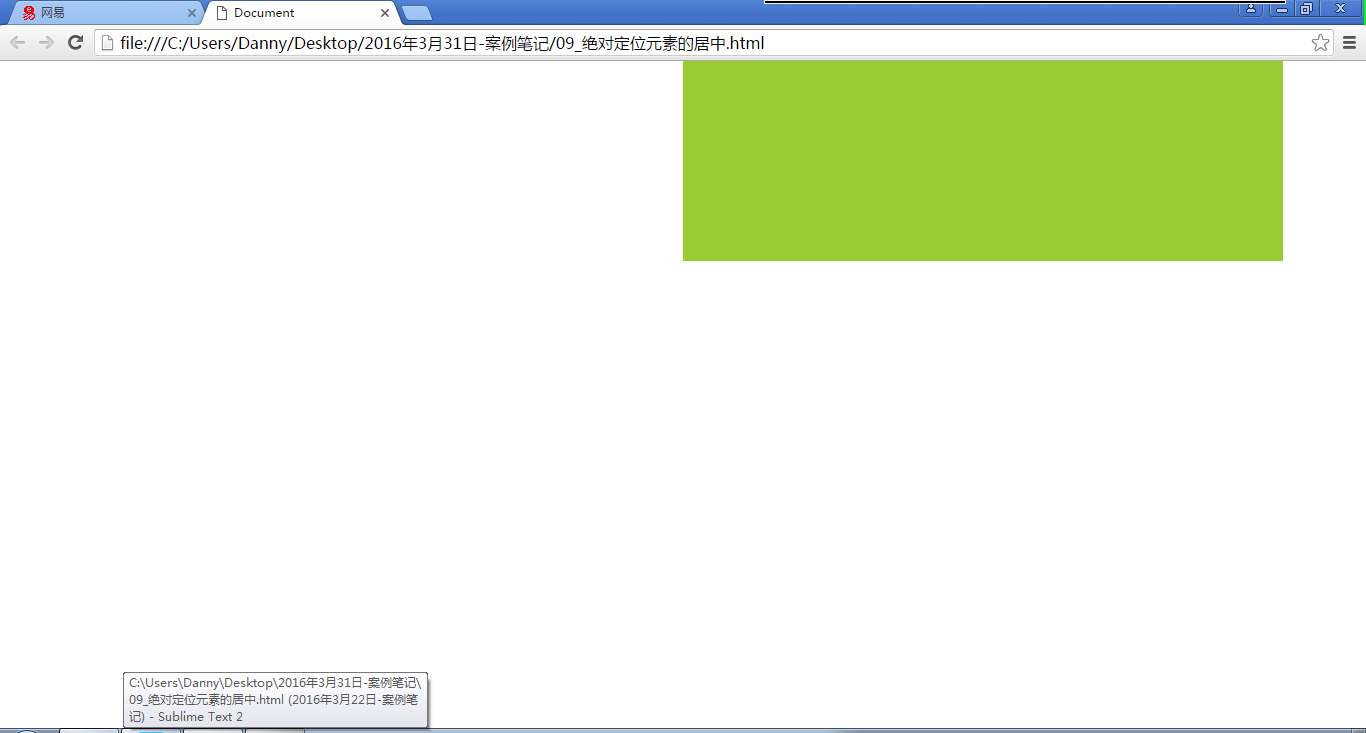
### 3.2.3 绝对定位元素的居中

新说一个知识点，就是绝对定位盒子已经不属于标准文档流了，所以不能使用margin:0 auto；来居中了。那么绝对定位的盒子如何居中？

|  |
| --- |
| 1. position: absolute; 2. **left:50%;** 3. top: 0; |

left:50%;指的是元素的左边线是50%的位置。这点和background-position不一样。

所以要往回拉一半的自己宽度。



|  |
| --- |
| 1. <style type="text/css"> 2. div{ 3. width: 600px; 4. height: 200px; 5. background-color: yellowgreen; 6. position: absolute; 7. **left:50%;** 8. top: 0; 9. **margin-left: -300px;** 10. } 11. </style> |

### 3.2.4 应用

所有的压盖效果，想都别想，90%是绝对定位做的。

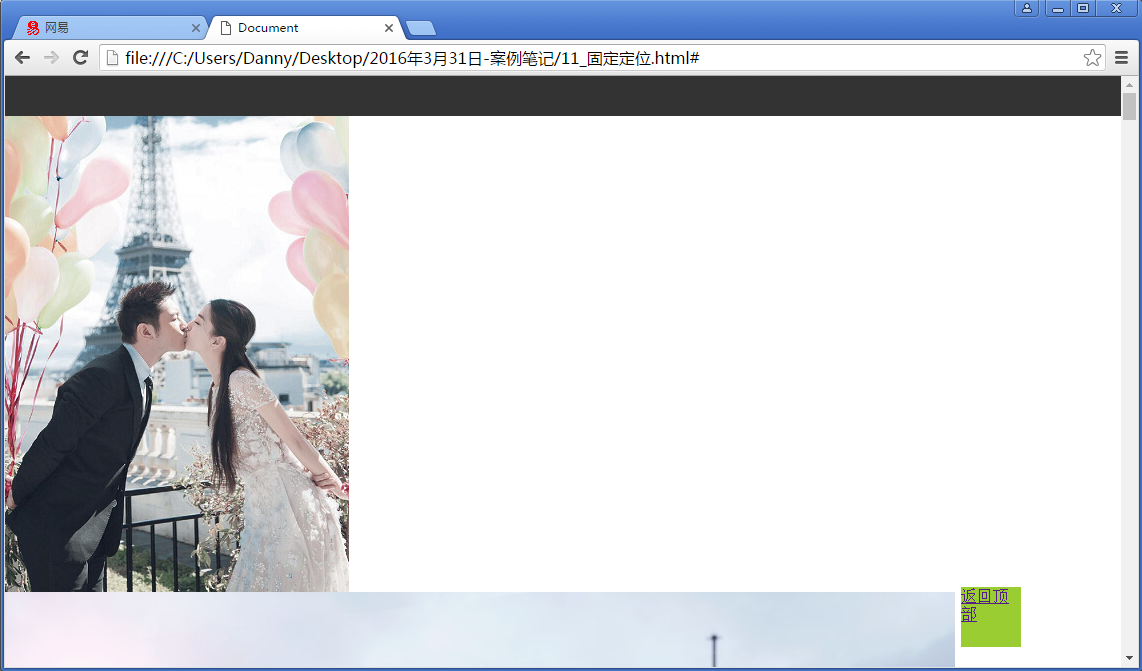


## 3.3 固定定位

|  |
| --- |
| 1. position:fixed; |

固定定位脱标，参考点是浏览器的角。

随着窗口的卷动，固定定位的元素不会在视口中消失。



**固定定位的参考点一定是浏览器，我们无法让一个盒子当做fixed定位的参考点。**

至此我们已经全部学习了脱标的3种方法：

浮动、绝对定位、固定定位

这三种脱标的方法，只能有一种作用在同一个元素身上。不存在一个元素同时浮动、同时绝对定位的情况。

我们给一个父亲overflow:hidden; 这个时候父亲就能认识浮动儿子的高度了，就能被儿子撑高了。

但是，绝对定位、固定定位的儿子，父亲永远不可能被他们撑高！

# 四、z-index属性

[z-index？](https://developer.mozilla.org/en-US/docs/Web/CSS/z-index)

负责设置压盖顺序，谁压盖谁的问题。

先说的默认的压盖顺序：

1） 定位了的能够压住没有定位的；

2） 如果都定位了，那么HTML顺序后面的能够压住前面的；

|  |
| --- |
| 1. <div class="box1">1</div> → 没有定位 2. <div class="box2">2</div> → 绝对定位 3. <div class="box3">3</div> → 相对定位 |

3压住2和1； 2压住1

z-index**没有单位**，**数字大的能够压住数字小的**。

|  |
| --- |
| 1. z-index: 5; |

**只有定位了的元素才能写z-index值**。标准流的元素不能写z-index值，浮动的元素不能写z-index。

从父现象：父亲怂了，儿子再牛逼也没用。

|  |
| --- |
| 1. <div class="linzhiying"> → z-index:9; 2. <p class="kimi"></p> → z-index:100000; 3. </div> 4. <div class="wangjianlin"> → z-index:10; 5. <p class="wangsicong"></p> → z-index:无论是多少肯定能压住kimi因为父亲z-index大 6. </div> |