# 复习

定位：

相对定位：

相对定位的元素原位置保留，不脱离标准文档流

新位置相对于原位置进行偏移。

应用：作为绝对定位元素参考盒子

绝对定位：

原位置不保留，脱离标准文档流

1不针对祖先元素，有top参与，参考页面

有bottom参与，参考首屏

2针对祖先元素，参考距离最近且有定位的祖先元素。

3水平居中：left: 50%;

|  |
| --- |
| 1. /\*绝对定位参考顶点：边框以内的四个顶点\*/ 2. left: 50%; 3. /\*拉回自身占有宽度一半\*/ 4. margin-left: -81px; |

固定定位：

原位置不保留，脱离标准文档流

相对于浏览器窗口四个顶点进行偏移

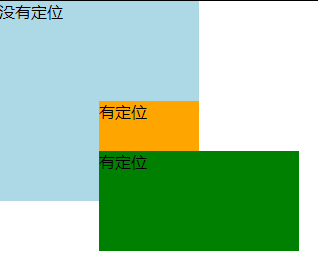
# 定位

## 元素默认压盖顺序

1有定位的元素压盖住没有定位的元素

2都有定位的元素，结构在后面的压盖住前面的

|  |
| --- |
|  |



## 自定义压盖属性

自定义压盖顺序使用z-index属性

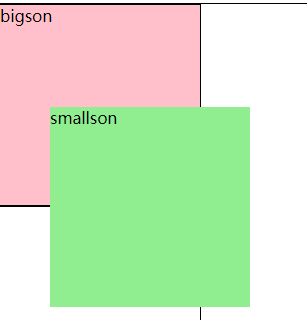
只有定位的元素才可以使用z-index

z-index属性值是数字，数字越大压盖顺序越靠上。

如果z-index数字相同，结构在后面的压盖住前面

对于父子嵌套盒，父盒子z-index小，儿子z-index再大也没用。

|  |
| --- |
| 1. <div class="big">//10 2. <p class="big-s">bigson</p>//10000 3. </div> 4. <div class="sm">//11 5. <p class="sm-s">smallson</p>//1 6. </div> |



## 呼吸轮播图

京东：所有的图片摞在一起

## 滚动轮播图

淘宝：所有图片并排显示

|  |
| --- |
| 1. .carousel { 2. width: 880px; 3. height: 550px; 4. border: 10px solid red; 5. margin: 50px auto; 6. position: relative; 7. overflow: hidden; 8. } 9. .carousel ul.unit { 10. /\*移动的大盒子\*/ 11. /\*定位：位置移动\*/ 12. position: absolute; 13. /\*移动的大盒子 宽度要非常宽\*/ 14. width: 9999px; 15. height: 550px; 16. left: 0px; 17. top: 0; 18. } 19. .carousel ul.unit li { 20. float: left; 21. width: 880px; 22. height: 550px; 23. } |

# css常用单位

em:相对单位，相对于父盒子字号的倍数

rem:相对单位：相对于根元素(html)字号的倍数

Px

像素，是屏幕上显示数据的最基本的点。用于网页设计，直观方便，相对单位

像素是相对于显示器屏幕分辨率而言的。

　　譬如，windows的用户所使用的分辨率一般是96像素/英寸。而MAC的用户所使用的分辨率一般是72像素/英寸。

pt是印刷行业常用单位，1pt = 1/72英寸。绝对单位

Pc：派卡,绝对单位，

In:英寸，绝对单位

Mm:绝对单位，毫米

Cm:绝对单位，厘米

1in = 2.54cm = 25.4 mm = 72pt = 6pc

# 三、hack

hacker黑客

因为浏览器的兼容性问题，我们针对不同的浏览器书写不同html或者css代码，以便达到不同浏览器查看效果相同。

html hack 和css hack

## 3.1 html hack

IE浏览器兼容有很大问题，给我们提供了很多端口，书写hack。

|  |
| --- |
| 1. <!--[if lte IE 9]> 2. 内容部分 3. <![endif]--> |

针对不同的浏览器在同一个html文件里，去写不同的html结构。

IE浏览器有很多兼容性问题，给我们提供了一些接口。

表示Hack符里面的内容只在IE9及以下的浏览器内显示，IE10及以上或者高级浏览器会认为这是注释。

if是如果，lte是less than or equal，小于或者等于，IE浏览器，9代表版本。

lte : less than or equal , 表示小于或等于。

lt : less than , 表示小于。

gt : greater than , 表示大于。

gte:表示大于或者等于

应用：可以针对低版本浏览器书写特定的代码。

针对某一个特定的 IE版本有特殊写法：

只给IE6浏览器加说明 代码，

|  |
| --- |
| 1. <!--[if IE 6]> 2. <h3>我是IE6</h3> 3. <![endif]--> |

## 3.2 css hack

css hack ：值的hack 和选择器hack

### 3.2.1css值的hack

/\* IE 6 \*/

.selector { \_color: blue; }

.selector { -color: blue; }

Hack符：-、\_

在属性名的前面加下划线或横线。

表示这个属性只有IE6认识，其他的都不认识这个属性。

|  |
| --- |
| 1. .box{ 2. width: 200px; 3. height: 200px; 4. background: orange; 5. color:#000; 6. -color:#fff; //只有IE6显示文字颜色为白色。 7. } |



/\* IE 6/7 \*/

Hack符：! $ & \* ( ) = % + @ , . / ` [ ] # ~ ? : < > |

表示这个属性只有IE6/7认识，其他的都不认识这个属性。

/\* IE 8/9 \*/

.selector { color: blue\0/; }

Hack符：\0/。

需要写在属性值的后面。

只在IE8、9里认识。

/\* IE 6/7/8/9/10 \*/

.selector { color: blue\9; }

Hack符：\9.

需要写在属性值的后面，分号前。

### 3.2.2 css选择器hack

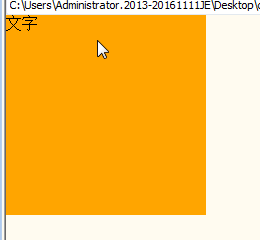
/\* IE 6 and below \*/

\* html .selector {}

这种选择器只在IE6里加载，其他的浏览器认为你的选择器是错的。

高级浏览器认为html已经是根标签，\*不是他的祖先元素。

|  |
| --- |
| 1. \* html .box{ 2. width: 200px; 3. height: 200px; 4. background: orange; 5. color:#000; 6. } |



/\* IE 7 and below \*/

.selector, {}

这种选择器只在IE7及以下版本里加载，其他的浏览器认为你的选择器是错的。

|  |
| --- |
| 1. .box,{ 2. width: 200px; 3. height: 200px; 4. background: orange; 5. color:#000; 6. } |

/\* 除了IE 6 \*/

html > body .selector {}

>子级选择器：只选择儿子级，后代其他级别不选。

IE6不认识子级选择器。

除了IE6都能正常加载。

|  |
| --- |
| 1. html > body .box{ 2. width: 200px; 3. height: 200px; 4. background: orange; 5. color:#000; 6. } |

# 四、IE6的兼容

## 4.1 选择器的兼容

IE6不兼容交集选择器里的类选择器连写。

div.box 可以兼容

div.box.cl 不兼容

解决办法：

使用标签和一个类的交集。

其他的7种选择器完美的兼容：

标签、id、类、交集（div.box）、后代、并集、通配符

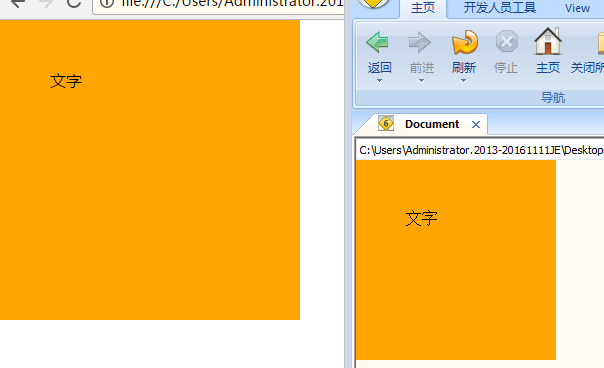
|  |
| --- |
| 1. div.box.cl{ 2. width: 200px; 3. height: 200px; 4. background: orange; 5. } 6. div.box { 7. background-color: red; 8. } 9. //高级浏览器中.box.cl权重比.box高渲染为橘色 10. //IE6中.box 和.box.cl权重相同，css书写后面的层叠掉前面的颜色渲染为红色。 |

## 4.2 盒模型的兼容

### 5.2.1如果不写DTD，IE6里的盒子是内减的，其他浏览器是外扩。

解决方法：

必须写DTD。



### 4.2.2如果盒子高度小于默认字号，不会正常显示。高度会是默认的字号。

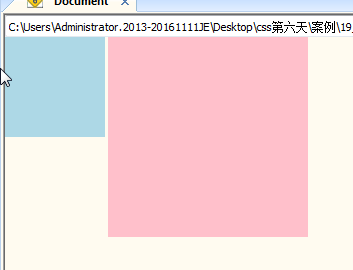
解决方法：

单独给IE6浏览器，强制给个很小的字号。

|  |
| --- |
| 1. .box{ 2. width: 200px; 3. height:3px; 4. background: orange; 5. -font-size:3px; 6. } |

## 4.3 浮动的兼容

### 4.3.1 不浮动的盒子不会钻到底下



情况：一个盒子浮动，一个盒子不浮动，在IE6里，不浮动的盒子不会钻到浮动盒子的下面占领它原来的标准流位置。

解决办法：制作压盖效果用定位。同一级的盒子要浮动都浮动，要不浮动都不浮动。

### 4.3.2 3pxbug

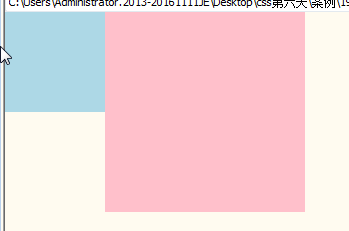
情况：一个浮动，一个不浮动，IE6里两个盒子之间会出现3px的间距。

解决方法：（不允许这么写）。

必须解决：给左边的浮动盒子加一个-3px的右margin。

只给IE6浏览器加，注意Hack符。

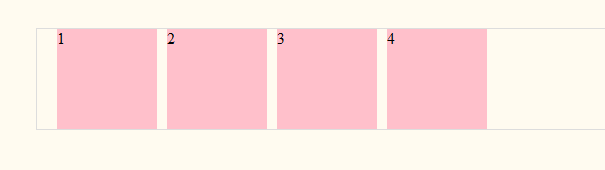
|  |
| --- |
| 1. .box1{ 2. width: 200px; 3. height: 200px; 4. background: pink; 5. -margin-right:-3px; //且只有ie6认识。 6. } |



### 4.3.3 双倍margin问题

情况：一些元素浮动，有一个与浮动方向相同的方向的margin，第一个元素会出现双倍边距的问题。

|  |
| --- |
| 1. .box p{ 2. float: left; 3. margin-left:10px; 4. width: 100px; 5. height: 100px; 6. background: pink; 7. } |



解决办法：①用父亲的padding去挤

|  |
| --- |
| 1. .box{ 2. width: 590px; 3. height: 100px; 4. border:1px solid #ddd; 5. margin:100px auto; 6. padding-left:10px; 7. } 8. .box p.first{ 9. margin-left: 0; 10. } |

②给第一个浮动的元素margin减半(只有IE6添加)

|  |
| --- |
| 1. .box p.first{ 2. -margin-left: 5px; 3. } |

# 五、html5概述

## 5.1 概述

HTML5 是 HTML 标准的最新演进版本。 这个术语代表了两个不同的概念：

① 它是一个新的 HTML 语言版本包含了新的元素，属性和行为

② 同时包含了一系列可以被用来让 Web 站点和应用更加多样化，功能更强大的技术

html5的改变：

● 语义：能够让你更恰当地描述你的内容是什么。

● 连通性：能够让你和服务器之间通过创新的新技术方法进行通信。

● 离线 & 存储：能够让网页在客户端本地存储数据以及更高效地离线运行。

● 多媒体：使 video 和 audio 成为了在所有 Web 中的一等公民。

● 2D/3D 绘图 & 效果：提供了一个更加分化范围的呈现选择。

● 性能 & 集成：提供了非常显著的性能优化和更有效的计算机硬件使用。

● 设备访问 Device Access：能够处理各种输入和输出设备。

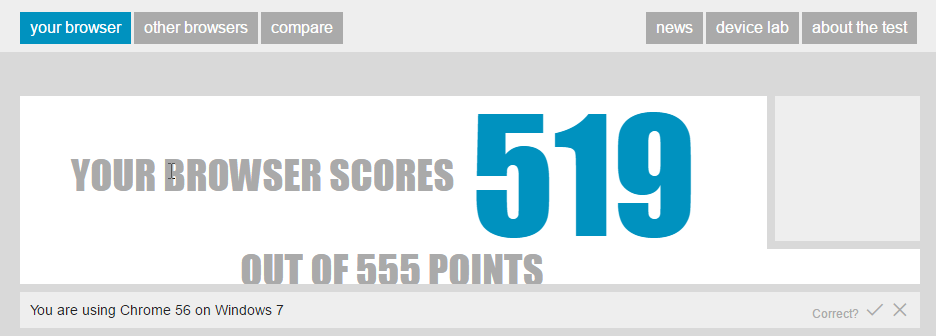
● 样式设计: 让作者们来创作更加复杂的主题吧！

HTML5的改变，涵盖了CSS3和JS的一些新特性，这些东西统称HTML5技术。

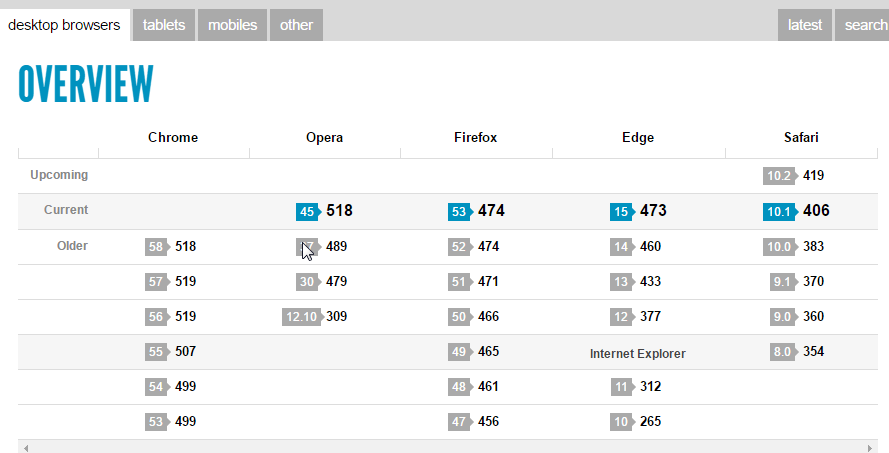
我们之前学习的是HTML4.01（XHTML1.0）。2014年10月29日，万维网联盟泪流满面地宣布，经过几乎8年的艰辛努力，HTML5标准规范终于最终制定完成了，并已公开发布。

## 5.2 兼容性

我的浏览器支持情况：



桌面浏览器支持情况：



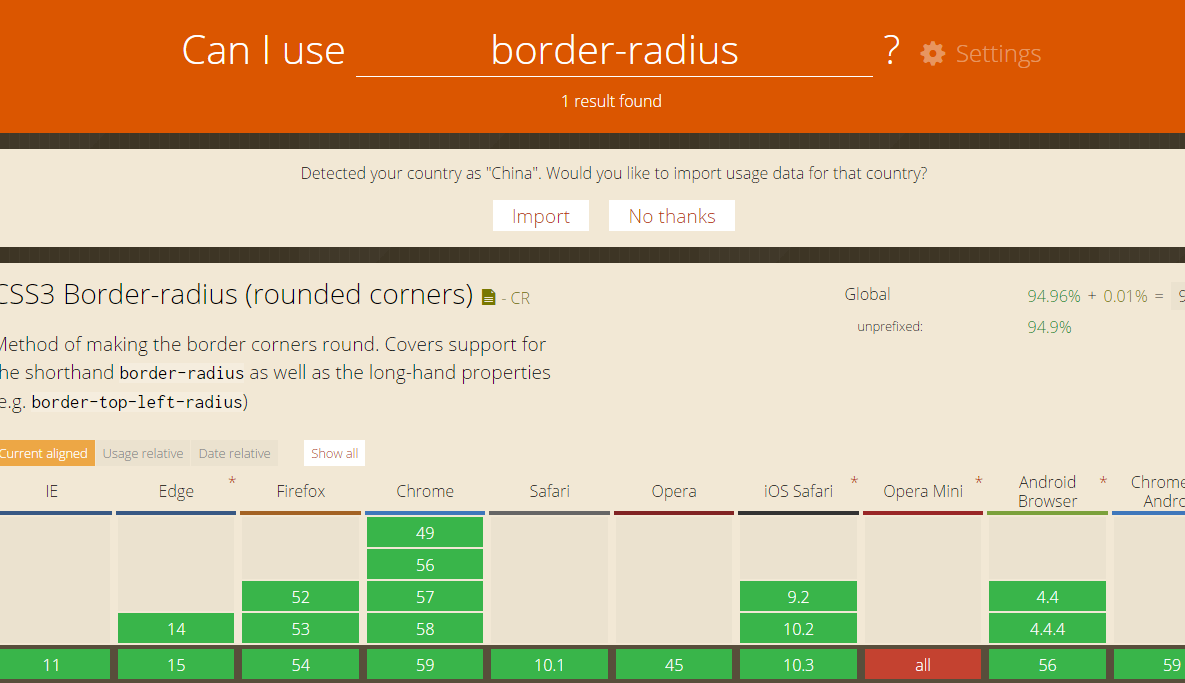
pc使用html5新的属性需要稍微注意，但是移动端使用内核（webkit）支持情况非常，移动端可以大胆使用各种html5.

官方HTML5浏览器性能评分：http://html5test.com/

满分555分，1个特性就是1分，一共555个新特性。

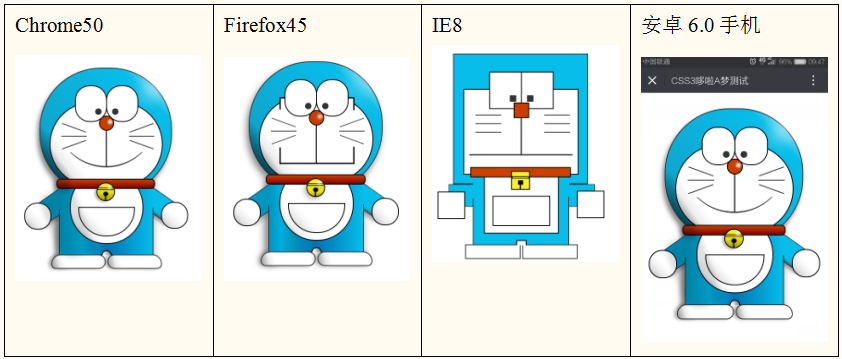
caniuse.com/

可以检索任何一个HTML5新特性，在各个浏览器里面的兼容情况。比如我们检索box-shadow的支持情况：



哆啦A梦测试，这是测试CSS3的：

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



# 六、html5新的骨架和语义化标签

## 6.1 html5骨架

DTD：书写骨架第一步就是书写ＤＴＤ，

|  |
| --- |
| 1. <!DOCTYPE html> |

这种DTD书写不要加5，因为表示书写网页对于不支持html5新特性的浏览器采用标准模式解析。就是对于不认识的属性或者标签，直接静默处理不报错，对于支持的html5新特性直接显示。

还需要设置字符集

|  |
| --- |
| 1. <meta charset=”UTF-8”> |

HTML5骨架：

|  |
| --- |
| 1. <!DOCTYPE html> 2. <html lang="en"> 3. <head> 4. <title>网页标题</title> 5. <meta charset="UTF-8"> 6. </head> 7. <body> 8. 网页主体部分 9. </body> 10. </html> |

## 6.2 新的特性

* 自封闭标签（单标签）不写关闭符号也是合法的。（可以使用）

|  |
| --- |
| 1. <img src="images/0.jpg" alt=""> |

* 所有的type属性不写也是合法的。（可以使用）

|  |
| --- |
| 1. <style></style> 2. <link href=””> 3. <script></script> |

* 属性值不写双引号也是合法的。（别使用）

|  |
| --- |
| 1. <img src=images/12.jpg alt=图片加载失败> |

* 标签大小写都合法。（别使用）

|  |
| --- |
| 1. <div class="box">文字</div> 2. <!-- 标签可以书写大写字母，不许同学们用 --> 3. <Div>文字</Div> 4. <DIV>文字</DIV> |

## 6.3 新的语义化标签

HTML就是负责页面语义的，HTML4.01中，div负责的任务太多，具体的语义没有拆分。

HTML5中，就是把div拆分为了更多的标签：

<section>, <article>, <nav>, <header>, <footer>, <aside> .

这些标签都和div一样，是容器级标签，可以互相嵌套，放什么都行，放自己也行。都是block-level，标准流中是块级元素。

其他的语义化标签：

一组：figure和figcaption。

|  |
| --- |
| 1. <figure> 2. <figcaption>这是标题</figcaption> 3. <img src="images/c1.jpg" alt=""> 4. </figure> |

time标签，时间语义化标签。是行内元素。

|  |
| --- |
| 1. <p>我们上午课上时间是从<time>9点</time>开始</p> |

mark，有强调的语义。

|  |
| --- |
| 1. 同学们早点到校<mark>学习</mark> |

一组：details和summary。有总结含义的标签。

|  |
| --- |
| 1. <details> 2. <summary>我们每一阶段有考试</summary> 3. <p>html考试</p> 4. <p>js基础考试</p> 5. <p>js进阶考试</p> 6. </details> |

# 表单

## 7.1 表单元素可以不书写在form标签内部

|  |
| --- |
| 1. <form action="" id="biaodan"></form> 2. <input type="text" value="用户名" form="biaodan"> |

## 7.2 label标签改变

|  |
| --- |
| 1. <label> 2. 请输入用户名：<input type="text"> 3. </label> |

## 7.2 datalist

自动感应标签

|  |
| --- |
| 1. <p> 2. 你学会技能：<input type="text" list="ganying"> 3. <datalist id="ganying"> 4. <option value="html">网页制作</option> 5. <option value="css">样式设置</option> 6. <option value="javascript">用户交互</option> 7. </datalist> 8. </p> |

## 7.3 placeholder属性

占位符

## 7.4 autofocus

自动获取焦点属性

## 7.5 required

必填项属性

## 7.6 输入控件

input标签更改type属性值表示不同的表单元素

电子邮箱：email

电话：tel

搜索框：search

网址:url

数字：number min:最小值 max:最大值

## 7.7 其他控件

拖拽条：range min:最小值 max:最大值

日期：date

颜色：color

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |