

# 复习 第七章：浏览器兼容性

2017年5月2日 4:10

## 第七章：浏览器兼容性

### 7.1 - css兼容性常见问题


#### 1\_ie不兼容点线

ie浏览器的点线形状与标准浏览器不同；  
如果一定要用点线，就只能用背景；

#### 2\_ie8以下兼容h5新标签

- `<script>`  
    `document.createElement("section");`  
    `document.createElement("header");`  
    `</script>-->`
- `<script src="js/html5shiv.js"></script>`

#### 3\_img会多撑开3px

- 当父元素没有写高，靠img自动撑开时，会默认多撑开3px;在ie6里，只要将父元素的结束标签与img写在同一行即可解决；  
    但是别的浏览器无法用此方法；
- 最佳方法：`vertical-align:top;`  
    也可以{
- `display:block;`  
    }

#### 4\_ie6的默认高度

ie6里元素会有一个默认的最小高度，官方显示是19px;  
即只要给的高度小于19px，都会默认19px;  
最佳解决方法：`overflow:hidden;`  
还有一种是：`font-size:0;`

#### 5\_一个元素浮动另一个元素的margin值等于前元素，ie6会出问题

问题：ie6里，如果第一个元素浮动，第二个元素的margin值等于第一个元素的宽，那么在ie6会出现缝隙；  
解决：不要用这种方法布局，尽量都浮动；

#### 6\_ie6里元素浮动后如果里面有块元素需要由内容撑开，会出问题

ie6里，如果只给父元素浮动，不给父元素里的块元素浮动，会出现掉成两行的现象；  
解决办法：给块浮动即可；

#### 7\_ie6里子元素超出会撑开父元素

ie6里，子元素如果超出了父元素，会默认把父元素撑开；

解决：1.不要让子元素超出父元素；

2.overflow:hidden;仍然生效，会隐藏掉子元素超出的部分；

发现：width和height放到ie6解析，实际上会被解析成最小宽和最小高；

8\_ie6,ie7里包不住定位属性为relative的元素

ie6,ie7里，父元素包不住position:relative;的子元素；

解决：给父元素写position:relative;

9\_ie6的双倍浮向问题

ie6的双倍浮向问题；

如果一个盒子里有几个子元素全体浮动，并且规定了margin值，那么就会在margin值的方向多解析出一倍的margin值；

解决方法：给子元素添加display:inline;

10\_ie6父元素宽高是奇数对绝对定位子元素的影响

ie6里，如果父元素的宽高是奇数，那么子元素依据父元素做绝对定位时，right和bottom的值会有1px的偏差；

解决：尽量不要写奇数宽高，或者尽量用left,top定位；

11\_ie6浮动元素之间有注释，会导致多出来一个字

解决：

1.不要在浮动元素之间写注释和空的内联；

2.不要让盒子的结束标记与文字在同一排；

3.不要让浮动子元素的宽与父元素相差小于3px；

12\_ie6, ie7不识别inline-block

ie6,ie7不支持块元素转为内联块；

用\*display:inline;让ie6,7的块元素转变为内联；

接着用\*zoom:1;触发haslayout,使内联支持宽高；

但注意，虽然此时ie6,7与标准浏览器效果相似，但是不解析空格；

display:inline-block;

\*display:inline;

\*zoom:1;

13\_input在**各个浏览器**距离顶上距离不一致的问题

不同浏览器默认的大小，以及写在按钮里的字都不一样，默认的边框都不同，在写之前重置；

ie6,ie7不认识border:none;只认识border:0;

input框的大小一定要重新规定；

background也要重新规定；重置background:none;

不同的input距离顶上的距离在各个浏览器都不同；

解决方法：float:left;

14\_ie6,ie7,ie8 input被div包住会有空隙

在ie6,ie7,ie8里，如果用一个盒子包住input，input与盒子宽高一致时，会出现间隙；

解决方法：float:left;

对于input标签，写完之后直接浮动，就可以避免很多问题；

15\_ie6里,给按钮添加边框会出现双重边框

在ie6,ie7里，如果给按钮写了边框，在点击时会出现第二层黑色边框，用outline也消不掉；

发现：只要不给按钮本身写边框，那么就不会出现黑色边框；

解决办法：给这个按钮写一个父元素，然后规定父元素宽高与按钮一致，将border加给这个父元素；

#### 16\_ie6的input的背景滚动问题

ie6,对于输入型的input，会产生背景滚动问题；即随着输入内容，背景会滚动；

解决方法：

```
_background-attachment:fixed;
```

#### 17\_ie6里绝对定位元素与浮动元素为兄弟关系时，绝对定位元素可能会消失

在ie6，绝对定位元素与浮动元素是兄弟关系时，绝对定位元素可能会消失；

发现：1.浮动元素只要与父元素的边缘相差超过3px,就不会出现这个bug；

2.这个与绝对定位元素的位置无关，与浮动元素的margin值有关；只要浮动元素的margin值加上浮动元素的宽与父元素的宽差距小于3px，就会出现这个bug；

解决：只要不要让绝对定位元素与浮动元素是兄弟关系，或者不要让二者差距小于3px即不会出现问题；

#### 18\_怎么设置ie6的最小高度

ie6不识别min-height;

为了给ie6写最小高，又不妨碍标准浏览器：

```
min-height:100px;
```

```
_height:100px;
```

ie6不识别min-height;

```
min-height:100px;
```

```
height:auto!important;
```

```
height:100px;
```

min-height和height不是完全一样的属性，不会用先后顺序或者权重去选择用哪个，两条都会同时被用上；但是规定height，就会导致min-height体现不出来；

#### 19\_png24的问题

问题：ie6不支持png24透明；

这一段filter只能给背景图片用；

```
.ie6{
    _filter:progid:DXImageTransform.Microsoft.AlphaImageLoader(src="images/png24.png",
    sizingMethod="scale");
    _display:inline-block;
    width:326px;
    height:326px;
}
body{
    background:#000 url(images/png24.png) no-repeat;
    _background-image:none;
    _filter:progid:DXImageTransform.Microsoft.AlphaImageLoader(src="images/png24.png",
    sizingMethod="scale");
}
```

## 7.2 - 代码调试

### 7.2.1 - 调试工具

### 7.2.2 - 手动调试

## 7.3 - css hack

### 7.3.1 - 常用hack写法

### 7.3.2 - !important

## 7.4 - html条件注释语句

```
<!--[if IE 6]>
    <h2>你的版本过低，请升级</h2>
<![endif]-->
<!--
    ie10,ie11用不了条件注释；
-->
```

## 7.5 - PNG24 兼容性问题

### 7.5.1 - IE6固定定位兼容

来自 <<http://www.miaov.com/index.php/Course/outlineTree/cid/18/sid/2>>

下划线过滤器：即在属性前添加下划线；

这个过滤器只有ie6及以下版本能识别，其他浏览器均会忽略这条属性；

星号过滤器：即在属性前添加星号；

这个过滤器只有ie7及以下版本能识别，其他浏览器均会忽略这条属性；

\9过滤器：即在属性值后面，分号前面添加\9；

这个过滤器只有ie版本能识别，其他浏览器均会忽略这条属性；ie11不能用；

\0过滤器：即在属性值后面，分号前添加\0；

这个过滤器只有ie8以上版本能识别，其他浏览器均会忽略这条属性；

## 1. 为什么会出现浏览器兼容问题？

由于各大主流浏览器由不同的厂家开发，所用的核心架构和代码也很难重合，这就为各种莫名其妙的Bug（代码错误）提供了温床。再加上各大厂商出于自身利益考虑而设置的种种技术壁垒，都让CSS应用起来比想象得要麻

烦。浏览器的兼容问题是我们必须去克服的。

## 2/五大浏览器内核代表作品

\***Trident**: IE、Maxthon(遨游)、腾讯、TheWorld 之窗、360浏览器

代表作品IE, 因为IE捆绑在Windows中, 所以占有极高的市场份额, 又称IE内核或是MSHTML, 此内核只能应用于windows平台, 且是不开源的。

\***Gecko**: 代表作品Mozilla Firefox 是开源的, 它的最大优势是跨平台, 能在Microsoft Windows、Linux和MacOS X等主要操作系统上运行。

\***Webkit**: 代表作品Safari、Chrome, 是一个开源项目。

\***Presto**: 代表作品Opera, Presto是由Opera Software开发的浏览器排版引擎。它也是世界上公认的渲染速度最快的引擎。

\***Blink**: 由Google和Opera Software开发的浏览器排版引擎, 2013年4月发布。

## CSS Bug、CSS Hack和Filter

1) CSS Bug: CSS样式在各浏览器中解析不一致的情况, 或者说CSS样式在浏览器中不能正确显示的问题称为CSS bug.

2) CSS Hack: CSS中, Hack是指一种兼容CSS在不同浏览器中正确显示的技巧方法, 因为它们都属于个人对CSS代码的非官方的修改, 或非官方的补丁。有些人更喜欢使用patch(补丁)来描述这种行为。

3) Filter: 表示过滤器的意思, 它是一种对特定的浏览器或浏览器组显示或隐藏规则或声明的方法。本质上讲, Filter是一种用来过滤不同浏览器的Hack类型。

\*使用Hack带来的一些副作用

降低了CSS代码的可读性，增加了代码的负担。

\*设计CSS Hack和 Filter通常有两种方法

1) 一种是利用浏览器自身的Bug，来隐藏或显示样式或声明；

2) 另一种是利用浏览器对CSS支持的不完善，如对某些规则或语法还没有形成支持，来隐藏或显示样式。