

CSS Sticky Footer完美的CSS绝对底部

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <title></title>
6     <style type="text/css">
7       *{margin: 0;padding: 0;}
8       .clearfix:after{content: '';display: block;width: 0;height: 0;font-size: 0;visibility: hidden;overflow: hidden;clear: both;}
9     html,body{
10       width: 100%;
11       height: 100%;
12       background: #00A0E9;
13     }
14     #wrap{
15       height: auto;
16       min-height: 100%;
17     }
18     .main{
19       padding-bottom: 150px;
20     }
21     .footer{
22       position: relative;
23       margin-top: -150px;
24       height: 150px;
25       clear: both;
26       background: #404040;
27     }
28     .header{
29       height: 60px;
30       background: #282828;
31     }
32   </style>
33 </head>
34 <body>
35   <div id="wrap">
36     <div class="main clearfix">
37       <div class="content">
38         <div class="header">
39
40         </div>
41         <div class="aside">
42           <p>165465444</p><p>165465444</p><p>165465444</p><p>165465444</p><p>165465444</p><p>165465444</p><p>165465444</p>
43           <p>165465444</p><p>165465444</p><p>165465444</p><p>165465444</p><p>165465444</p><p>165465444</p><p>165465444</p>
44         </div>
45       </div>
46     </div>
47   </div>
48   <div class="footer">
49
50   </div>
51 </body>
52 </html>
53
```

CSS Sticky Footer: 完美的CSS绝对底部

标签: [cssfoote](#)

CSS的简单在于它易学，CSS的困难在于寻找更好的解决方案。在CSS的世界里，似乎没有完美这种说法。所以，现在介绍的CSS绝对底部，只是目前个人见过的方案中比较完美的吧。

先说我们会使用到这个CSS底部布局解决方案：

当做一个页面时，如果页面内容很少，不足以填充一屏的窗口区域，按普通的布局，就会出现下面图片中的样子(也就是底部内容并没有位于窗口的底部，而留下了大量空白)。



nonstickfooter

对于追求完美的设计师来说，这是不美观的。网上有一些解决方案，但会出现当改变窗口高度时，底部和正文重叠的BUG。尽管没有多少人会有事没事的去改变窗口高度，但设计嘛，追求的就是尽善尽美。



overlapfooter

下面是我找到的一个比较完美的方法，来自国外的设计达人，纯CSS，可以实现：当正文内容很少时，底部位于窗口最下面。当改变窗口高度时，不会出现重叠问题。



cssstickyfooter

甚至，创造该CSS的人还专门成立一个网站介绍这个CSS底部布局方案。不知道他有没有去申请专利:)

代码写法

HTML代码:

```
<div id="wrap">
  <div id="main" class="clearfix">
    <div id="content">
    </div>
    <div id="side">
    </div>
  </div>
</div>
<div id="footer">
</div>
```

说明: 使用这个布局的前提, 就是footer要在总的div容器之外, footer使用一个层, 其它所有内容使用一个总的层。如果确实需要到添加其它同级层, 那这个同级层就必须使用position:absolute进行绝对定位。

CSS代码:

下面是主要的CSS代码, 让你的底部可以位于窗口的最下面:

```
html, body, #wrap {height: 100%;}
body > #wrap {height: auto; min-height: 100%;}
#main {padding-bottom: 150px;} /* 必须使用和footer相同的高度 */
#footer {position: relative;
margin-top: -150px; /* footer高度的负值 */
height: 150px;
clear:both;}
```

说明: 需要注意的就是#main的padding值、footer的高度和负margin值, 需要保持一致。

就是这么简单, 不过还没完。如果你的主体是使用悬浮布局, 还得解决一些浏览器的兼容问题, 这里使用的重点是为了Google Chrome。

对#main部份进行著名的Clearfix Hack:

```
.clearfix:after {content: ".";
display: block;
height: 0;
clear: both;
visibility: hidden;}
.clearfix {display: inline-block;}
/* Hides from IE-mac */
* html .clearfix { height: 1%;}
.clearfix {display: block;}
/* End hide from IE-mac */
```

注: 该方案测试于两栏悬浮布局, 兼容各大主流浏览器, 包括[Google Chrome](#)。

P.S:

- 网络上之前比较知名的footer布局有[Ryan Faits](#)创造的, 不过它的方法在HTML代码中会有一个空的div层。严格来说, 是不符合语义网代码标准的。
- 另外, 还有一篇[Exploring Footers article from A List Apart](#), 但使用了一些JavaScript代码。
- 这样一比较, 上面看似简单的纯CSS, 就显得伟大许多了。

OK, 没有了。如果没看懂, 具体的使用方法和说明可以[到原站查看](#)。