

一、初探rem布局

1.1 rem是什么？

rem是CSS3新增的一个相对单位（root em，根em）。这个单位与em有什么区别呢？区别在于使用rem为元素设定字体大小时，仍然是相对大小，但相对的只是HTML根元素。这个单位可谓集相对大小和绝对大小的优点于一身，通过它既可以做到只修改根元素就成比例地调整所有字体大小，又可以避免字体大小逐层复合的连锁反应。目前，除了IE8及更早版本外，所有浏览器均已支持rem。对于不支持它的浏览器，应对方法也很简单，就是多写一个绝对单位的声明。这些浏览器会忽略用rem设定的字体大小。举一个简单的例子：

浏览器默认的 `html font-size=16px`，这样如果我们设置字体的大小为12px，通过计算可知 $12 / 16 = 0.75$ ；因此只需要设置 `font-size= 0.75rem`。

为了计算方便，我们可以设置`html font-size=10px` 或者 `font-size=62.5%`，这样设置12px字体的时候就只需要设置 `font-size=1.2rem`，十进制换算信手拈来。

作者：诗渊

来源：CSDN

原文：<https://blog.csdn.net/u014607184/article/details/79516528>

版权声明：本文为博主原创文章，转载请附上博文链接！

rem布局

原理是，先按定高宽设计出来页面，然后转换为rem单位，配合js查询屏幕大小来改变html的font-size，最终做出所谓的完美自适应。

em适用场景

rem+js是宽度自适应，无法做到高度自适应，所以那些对高度要求很高的rem+js无法实现。

改变浏览器宽度，你会发现，页面所有元素的高宽都等比例缩放，

也就是大屏幕下导航是横的，小屏幕下还是横的只不过变小了。。

优点：理想状态是所有屏幕的高宽比和最初的设计高宽比一样，或者相差不多，完美适应。

缺点：碰到重视高度的设计，或者重视元素间间距的设计，那就玩不开了。