## 单位比较

**px：**绝对单位，页面按精确像素展示，Windows系统默认是96dpi，Apple系统默认是72dpi。

**em：**是相对长度单位，相对于当前对象内文本的字体尺寸，即em的计算是基于父级元素font-size的，（浏览器默认字体是16px），通常1em=16px，整个页面内1em不是一个固定的值。

<body style="font-size:14px">

<p style="font-size:2em">字体大小是28px(14px\*2)</p>

<div style="font-size:18px">

<p style="font-size:2em">字体大小是36px(18px\*2),而不是上面计算的28px</p>

</div>

</body>

**rem：**相对单位，可理解为”root em”, 相对根节点html的字体大小来计算。

<meta name="viewport" content="width=device-width,initial-scale=1,minimum-scale=1,maximum-scale=1,user-scalable=no" />

//---一定要加上

<body style="font-size:14px">

<p style="font-size:2rem">大小是32px(16px\*2),根据html根元素的font-size大小计算</p>

<div style="font-size:18px">

<p style="font-size:2rem">字体大小是32px(16px\*2),根据html根元素的font-size</p>

</div>

</body>

\*补充默认font-size大小是16px(如果html中没有设置的话)。

## 如何实现响应式变化。

响应头：

<meta name="viewport" content="width=device-width,initial-scale=1,minimum-scale=1,maximum-scale=1,user-scalable=no" />

函数封装：

function getRem(pwidth,prem){

//---pwidth---UI图的宽度

//---prem--自己设置的换算比例

//获取html元素

var html = document.getElementsByTagName("html")[0];

//---屏幕的宽度（兼容处理）

var ScreenWidth = document.body.clientWidth || document.documentElement.clientWidth;

html.style.fontSize = ScreenWidth/pwidth\*prem + "px";

}

调用函数：

window.onload = function(){

//---文档加载完调用getRem()

getRem(1420,100)

/\*720代表设计师给的设计稿的宽度，你的设计稿是多少，就写多少;100代表换算比例，这里写100是

为了以后好算,比如，你测量的一个宽度是100px,就可以写为1rem,以及1px=0.01rem等等\*/

};

window.onresize = function(){

//---窗口或框架被调整大小发生变化

getRem(1420,100)

};