**@media与@media screen的区别**

在网站自适应设计中，@media与@media screen是必须用到的css代码，可能大家对此并不陌生。但是大家又知不知道@media与@media screen的区别在哪呢？在自适应设计中你是用@media还是用@media screen？你知不知道两者产生的结果有什么不同呢？这些问题我想很多人都不清楚吧。本文便通过实例给你说说@media与@media screen的区别，主要是告诉你它们产生的效果区别在哪。

**@media与@media screen手机网页区别**

好吧，理论方面暂时放在一边，先看看两者产生的手机网页视图有什么不同。



@media与@media screen手机网页

@media与@media screen手机网页

上图看出，没有任何区别。没错，确实没有任何区别，因为@media与@media screen的css在手机设备里都是有效的。

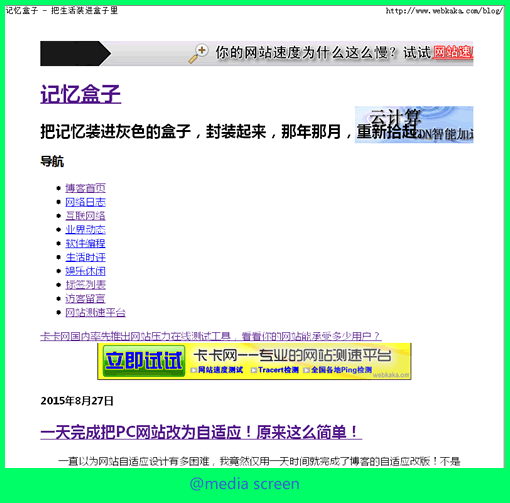
**@media与@media screen打印网页区别**

两者在手机设备上没有区别，在打印设备上是怎样的呢？看看如下截图。



@media打印网页

@media打印网页



@media screen打印网页

@media screen打印网页

很明显，两者的打印效果是不同的。

至此，我们应该明白了@media与@media screen的区别了吧，没错，@media screen的css在打印设备里是无效的，而@media在打印设备里是有效的，这就是它们的区别了。

知道了它们的区别，我们就应该懂得如何使用它们了。如果css需要用在打印设备里，那么就用@media ，否则，就用@media screen。

不过这只是笼统的做法，其实如果把“screen”换为“print”，写为**@media print**，那么该css就可用到打印设备上了，但要注意，**@media print**声明的css只能在打印设备上有效哦。

通过此实例的见解，我们可以举一反三，对@media的用法就能更易接受了。下面的知识扩展可以让你更深入地了解@media的各种用法。

**知识扩展**

**@media only screen and**

only(限定某种设备)  
screen 是媒体类型里的一种  
and 被称为关键字，其他关键字还包括 not(排除某种设备)

\**/\* 常用类型 /*  
类型 解释  
all 所有设备  
braille 盲文  
embossed 盲文打印  
handheld 手持设备  
print 文档打印或打印预览模式  
projection 项目演示，比如幻灯  
screen 彩色电脑屏幕  
speech 演讲  
tty 固定字母间距的网格的媒体，比如电传打字机  
tv 电视

screen一般用的比较多，下面是我自己的尝试，列出常用的设备的尺寸，然后给页面分了几个尺寸的版本。

\**/\* 常用设备 /*设备 屏幕尺寸  
显示器 1280 x 800  
ipad 1024 x 768  
Android 800 x 480  
iPhone 640 x 960

**两种方式**

a：

* <link rel="stylesheet" type="text/css" href="styleB.css" media="screen and (min-width: 600px) and (max-width: 800px)">

意思是当屏幕的宽度大于600小于800时，应用styleB.css

b：

* @media screen and (max-width: 600px) { /*当屏幕尺寸小于600px时，应用下面的CSS样式*/
* .class {
* ​ background: #ccc;
* }
* }

**device-aspect-ratio**

device-aspect-ratio可以用来适配特定屏幕长宽比的设备，这也是一个很有用的属性，比如，我们的页面想要对长宽比为4:3的普通屏幕定义一种样式，然后对于16:9和16:10的宽屏，定义另一种样式，比如自适应宽度和固定宽度：

* @media only screen and (device-aspect-ratio:4/3)

# 关于响应式布局，你必须要知道的

## **前言**

响应式Web设计可以让一个网站同时适配多种设备和多个屏幕，可以让网站的布局和功能随用户的使用环境（屏幕大小、输入方式、设备/浏览器能力）而变化。本文主要介绍一些响应式布局容易忽略但又很重要的知识点。

## **视口**

移动前端中常说的 viewport （视口）就是浏览器中用于呈现网页的区域。视口通常并不等于屏幕大小，特别是可以缩放浏览器窗口的情况下。手机端与PC端视口存在差异，电脑端的视口宽度等于分辨率，而移动端的视口宽度跟分辨率没有关系,宽度默认值是设备厂家指定的。iOS, Android基本都将这个视口分辨率设置为 980px。  
1.为什么手机端视口要设为980px?



乔布斯设想：苹果手机如果在市场上火爆了，但是各个网站还没有来得及制作手机端网页，那么用户不得不用手机访问电脑版的网页，如何用小屏幕访问大屏幕的页面也同样可读呢？乔帮主就想着为手机固定一个视口宽度，让手机的视口宽度等于世界上绝大多数PC网页的版心宽度，就是980px。这样，用手机访问电脑版网页的时候，旁边刚好没有留白。不过页面缩放后文字会变得非常小，用户需要手动放大缩小才能看清楚，体验非常差。

2.约束视口  
为了解决前面的问题，可以在网页的中添加下面这行代码：

<meta name="viewport"   
content="width=device-width,   
    initial-scale=1.0,  
    maximum-scale=1.0,  
    user-scalable=0" />

width=device-width 视口为设备宽度（就是人设置的一个宽度）//不设置的话默认为980px  
initial-scale=1.0 初始化的视口大小是1.0倍  
maximum-scale=1.0 最大的倍数是1.0倍  
user-scalable=0 不允许缩放视口

这个视口的标签告诉浏览器怎么渲染网页。在这里，标签想表达的意思是：按照设备的宽度（device-width）来渲染网页内容。事实上，在支持这个标签的设备上给你看一看效果，你就明白了。



不错呀！用户体验大大改善！！！  
此时如果用document.documentElement.clientWidth来测试浏览器屏幕宽度，你会发现当前视口宽度等于手机屏幕的宽度，约数后的视口宽度都是在320~480之间（手机竖直使用的时候）。  
这个视口的尺寸，是手机厂商设置的，能够保证我们的文字比如16px，在自己的这个视口下清晰、大小刚刚合适。所以大屏幕的手机的约束视口 > 小屏幕手机的约束视口。这就能够保证我们的网页可以用px写字号、写行高。  
需要注意的是：约束之后的视口宽度，不是自己的分辨率！！每个手机的分辨率，都要比自己的视口宽度大得多得多！  
最最重要的一句话：前端开发工程师，丝毫不关心手机的分辨率，我们只关心视口。

## **图片**

人们常说说“一图胜千言”，确实如此。我们网页中关于松饼的文字介绍再多，也没有图片有吸引力。下面我们就在页面上方添加一张松饼的图片（2000像素宽），效果类似引诱用户往下看的大题图。



哇，真是好大一张图，它让整个网页看起来都失衡了，水平方向上图片溢出了。不行，必须解决这个问题。可以用CSS给图片指定固定宽度，但问题是我们想让它能在不同大小的屏幕中自动缩放。比如，我们例子中的iPhone屏幕宽度为320像素，如果我们把图片设置成320像素宽，那么iPhone屏幕旋转后又怎么办呢？这时候320像素变成了480像素。  
解决方案很简单，只要一行CSS代码就可以让图片随容器宽度自动缩放:

img {  
max-width: 100%;  
}

回到手机上，刷新页面，结果比较符合预期了。  
这里声明max-width规则，就是要保证所有图片最大显示为其自身的100%（即最大只可以显示为自身那么大）。此时，如果包含图片的元素（比如包含图片的body或div）比图片固有宽度小，图片会缩放占满最大可用空间。

为什么不用width:100%?  
要实现图片的自动缩放，也可以使用更通用的 width 属性，比如width:100%。然而，这条规则在这里并不适用。因为这条规则会导致它显示得跟它的容器一样宽。在容器比图片宽得多的情况下，图片会被无谓地拉伸。

## **手机浏览器内核**

在移动端，仅有四个独立的浏览器内核，分别为微软的Trident、火狐的Gecko、开源内核Webkit、Opera的Presto。  
目前微软的Trident在移动终端上主要为WP7、8系统内置浏览器。Opera的Presto内核主要为 Opera Mobile、OperaMini、欧朋浏览器以及欧朋HD Beta版。Webkit内核的适用范围则较为广泛，Android原生浏览器、苹果的Safari、谷歌Chrome(Android4.0使用)都是基于Webkit开源内核开发的。

兼容的前缀：  
1 -ms-  
2 -moz-  
3 -o-  
4 -webkit-  
中国用户的浏览器市场份额：

近一年中国用户的浏览器市场份额



UC、Android内置、Chrome、Safari、QQ Browser都是webkit内核，从图上看占了绝大部分的市场份额。  
所以一定要伺候好-webkit-。 有的公司干脆只兼容-webkit-，别的兼容比如-ms-都不写。

## **流式布局**

百分比布局也叫作流式布局、弹性盒布局。手机网页没有版心，都左右撑满。  
百分比能够设置的属性是width、height、padding、margin。其他属性比如border、font-size不能用百分比设置的。

如果用百分比写width，那么指的是父元素width的百分之多少。

如果用百分比写height，那么指的是父元素height的百分之多少。

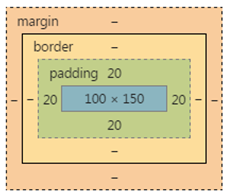
如果用百分比写padding，那么指的是父元素width的百分之多少，无论是水平的padding还是竖直的padding。

如果用百分比写margin，那么指的是父元素width的百分之多少，无论是水平的margin还是竖直的margin。

不能用百分比写border的宽度

接下来我们看一个例子：

div{  
  width:200px;  
  height:300px;  
  padding:10px;  
}  
div p{  
  width:50%;  
  height:50%;  
  padding:10%;     
}  
/\*此时p的真实宽度是多少？\*/



此时p的真实宽度是140px\*190px

## **媒体查询**

1.为什么响应式 Web 设计需要媒体查询  
CSS3媒体查询可以让我们针对特定的设备能力或条件为网页应用特定的CSS样式。如果没有媒体查询，光用CSS是无法大大修改网页外观的。这个模块让我们可以提前编写出适应很多不可预测因素的CSS规则，比如屏幕方向水平或垂直、视口或大或小等等。弹性布局虽然可以让设计适应较多场景，也包括某些尺寸的屏幕，但有时候确实不够用，因为我们还需要对布局进行更细致的调整。媒体查询让这一切成为可能，它就相当于CSS中基本的条件逻辑。

2.媒体查询语法  
我们在媒体查询外面写的第一条规则，是“基本的”样式，它适用于任何设备。在此基础上，我们再为不同视口、不同能力的设备，渐进增加不同的视觉效果和功能。

body {  
    background-color: grey;  
 }   
@media screen and (min-width:1200px){  
    body{  
        background-color: pink;  
    }  
}  
 @media screen and (min-width:700px) and (max-width:1200px){  
    body{  
    background-color: blue;  
    }  
}  
@media screen and (max-width:700px){  
    body{  
    background-color: orange;  
        }  
}

其中@media就表示媒体查询，查询现在看这个网页的设备是什么，以及它的宽度是多少。screen表示看这个网页的设备是显示器，而不是残疾人听力设备、也不是打印机。后面用and符号罗列所有的可能性。  
值得注意：媒体查询只能包裹选择器，不能包裹k:v对儿。  
IE6、7、8不支持媒体查询，也为了防止手机端的某些浏览器不支持媒体查询，所以不要把所有的选择器都放在媒体查询里面。

## **参考**

响应式Web设计：HTML5和CSS3实战/（英）本·弗莱恩（Ben Frain）著；