最近研究响应式设计框架的时候，发现网上很多相关的属性介绍，却很少有系统的入门级使用的文章，我自己整理了一篇入门知识，并没有什么高深的理论，也不牵扯到框架。

目前已经越来越多的站点以及wap站点使用响应式设计，因为大屏幕的移动设备越来越普及。而自适应布局已经无法胜任各种浏览需求。响应式设计的目的是尽可能以最好的布局显示您的数据，以实现用户友好的 Web 页面。

css2的时期有一个不是很常用的media type，他拥有比如aural（声音）braille（触摸）print（打印）handheld（移动设备）等等十种媒体类型，（附加媒体类型详细[http://www.w3.org/TR/CSS2/media.html#media-types](http://www.w3.org/TR/CSS2/media.html" \l "media-types" \o "http://www.w3.org/TR/CSS2/media.html#media-types" \t "_blank)）利用@media规则可以在同一样式表里为不同终端使用不同的样式。尴尬的是这个media type并没有被多少终端真正的支持。

现在CSS3有了个更为实用的 media query。而移动终端的浏览器基本已经完全支持了css3.他可以为你获取各种终端的数据。

先举个例子，大家看这个demo。（由于相应区域过大，就不截图了，请大家点击打开这个连接）

<http://www.caihong.cc/demo/media/>

一个普通的自适应显示的三栏网页，当你用不同的终端来查看这个页面的时候，他会根据几种终端来显示不同的样式，在电脑上是三列，在pad上应该也是三列，在大屏手机上是三行，在屏幕小于320的手机上只显示主要内容，隐藏掉了次要元素。（这里关于响应式布局还有个比较好的消息，就是拖动浏览器也可以触发判断条件，测试的时候你不需要去找一堆手机，只要把自己的浏览器窗口缩小到一定尺寸就可以了，这个demo也可以用拖动浏览器大小的方式测试。）

这就是一个最简单的 响应式布局 的页面。我们就从这个例子里认识下media query属性吧。

1. @media screen and (min-width: 320px) and (max-width : 479px)

就从这个条件语句开始介绍，media属性后面跟着的是一个 screen 的媒体类型（上面说过的十种媒体类型之一）。然后用 and 关键字来连接条件（其他关键字还有 not， only，看字面大家能理解，就不多说。），然后括号里就是一个媒体查询语句，稍微懂点css的同学都能看懂，这个条件语句意思是在大于320小于479的分辨率下所激活的样式表。

这个语句，就是响应式布局的基础应用了。在判断终端分辨率大小之后，赋予不同的样式进去，就像我们的例子里

1. @media screen and (max-width : 320px){
2. body{...}
3. }
4. @media screen and (min-width: 800px) and (max-width: 1024px){
5. body{...}
6. }

至于要判断多少种分辨率，完全取决于你产品的需求，常见的分辨率有手机，平板（注意这些终端是存在 横屏 竖屏 区别的,这个下一篇里提），桌面显示器。自己为自己所面对的终端定制样式。

一般大于960的显示器都可以用默认样式而不必在媒体查询里判断了。有一种情况除外，就是高像素比的终端，比如 iphone4以上的retina屏，一个iphone5的小小的屏幕（iphone的屏幕是真小啊），他的分辨率竟然达到了1136\*640，如何让放大了两倍的屏幕显示依然清晰？

在面对这种分辨率精细的终端，我们有另外一个条件查询语句 **device-pixel-ratio**。

比如例子里的

1. @media only screen and (-moz-min-device-pixel-ratio: 2), only screen and (-o-min-device-pixel-ratio: 2/1), only screen and (-webkit-min-device-pixel-ratio: 2), only screen and (min-device-pixel-ratio: 2)

就是判断终端的像素比是2的话，所渲染的样式。iphone4以上像素比是2，高分辨率Andriod设备像素比是1.5，例子里只有像素比为2的查询，1.5的或者其他比例方法一样，前面用的是几种浏览器的私有属性，最后一种是通用属性，

1. @media only screen and (-moz-min-device-pixel-ratio: 2), only screen and (-o-min-device-pixel-ratio: 2/1), only screen and (-webkit-min-device-pixel-ratio: 2), only screen and (min-device-pixel-ratio: 2)

等于

1. @media only screen and (min-device-pixel-ratio: 2)

为了一些版本的兼容性，不得已写的长了。

1. body{
2. font-size:24px;
3. }
4. .box2{
5. background: url(d/20.png) #ccc;
6. background-size:50%;
7. }

在像素比为2的终端里这样写的目的，就是让他显示的更容易识别，一般来说显示一张1px的背景图片，我们要准备一张2px的，然后再background-size:50%这样。1.5像素比同例。

比如上面的demo，如果你用iphone4以上的苹果手机来看，中间的背景图片应该是显示“2.0像素比”。

这里也暴露了响应式一个很大的缺点：需要多做若干背景图（作为内容显示的图片暂时无视，弹性图片与弹性字体，下次单独写一篇介绍博文介绍）。

对于media query的兼容性，我想不是很重要，因为很少有终端自带IE9以下的浏览器。基本都是高级浏览器。如果特殊需要，可以下载一个兼容的JS文件

1. <!--[if lt IE 9]>
2. <scriptsrc="http://css3-mediaqueries-js.googlecode.com/svn/trunk/css3-mediaqueries.js"></script>
3. <![endif]-->

以条件注释的方式加在文件里。

看完以上，我想关于响应式设计入门的知识应该了解了，后面有机会介绍一些好用的框架给大家分享。

现在终端设备越来越多，分辨率也越来越五花八门，但是为了用户体验，再多也要去兼容他呀~