Atitt 自适应布局

关于响应式布局有很多可以讲的东西，我写几个常见的方面，也当是自己的学习总结。

## 响应式布局

弹性盒（flex）布局是目前最流行的响应式布局方式。弹性盒是一维的，也就是说，它排布元素只能一行一行或一列一列地排，需要嵌套弹性盒或合理利用换行才能实现二维布局。具体的用法这里就不谈了。学习flex相关属性的时候需要注意各种属性的初始值。

## 视区相关单位

视区（viewport）即网页的可视区域。定义容器宽高可以考虑使用vw、vh、vmin、vmax等根据屏幕尺寸计算的单位长度。1vw表示屏幕宽度的1%，1vh表示屏幕高度的1%，1vmin和1vmax则分别表示中1vw和1vh的最小值和最大值（即min(1vw, 1vh)和max(1vw, 1vh)）。

## [视区meta标签](https://www.zhihu.com/search?q=%E8%A7%86%E5%8C%BAmeta%E6%A0%87%E7%AD%BE&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra={"sourceType":"answer","sourceId":1779511490}" \t "https://www.zhihu.com/question/_blank)

在HTML中还有一个名为viewport的<meta>标签，最早在iOS Safari上实现[[2]](https://www.zhihu.com/question/387788930" \l "ref_2)。iPhone初代发布时，一大卖点就是它能提供完整的网页浏览体验（而不是WAP版网页）[[3]](https://www.zhihu.com/question/387788930" \l "ref_3)。iOS Safari的默认viewport宽度为980px[[4]](https://www.zhihu.com/question/387788930" \l "ref_4)，而当时主流的电脑屏幕尺寸还是1024x768，因此大部分网页都能在iPhone上完整显示。也就是说，980px宽的网页会等比缩放到320px再显示。但如果开发者专门为iPhone写了一个网页，宽度只有320px，在iPhone上反而会变得很小。因此苹果定义了viewport meta标签，像这样：

<meta name="viewport" content="width=device-width, initial-scale=1">

其中，width表示viewport的宽度，initial-scale表示初始的缩放倍数[[4]](https://www.zhihu.com/question/387788930" \l "ref_4)。device-width表示宽度为设备的宽度。这样设置以后，网页就会以320px宽显示在设备上。

随着响应式布局的普及，现在几乎所有网页都会带上这个标签。

## CSS像素

从iPhone 4开始，苹果开始使用视网膜屏幕，即高分辨率屏幕。iPhone 4比iPhone 3GS在长宽上的像素各翻了一倍，来到960x640。按照原来的定义，device-width应该是640px。但这样一来，为以前的iPhone设计的网页在iPhone 4上只有正常的1/4。 苹果采用的策略是，使用逻辑像素层，一个点（point）对应设备上的4个[物理像素](https://www.zhihu.com/search?q=%E7%89%A9%E7%90%86%E5%83%8F%E7%B4%A0&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra={"sourceType":"answer","sourceId":1779511490}" \t "_blank)

## 媒介查询

媒介查询（media query）是为不同媒介应用不同样式的一套机制[[10]](" \l "ref_10)。比如说，如果你想为印刷媒介和屏幕媒介应用不同的样式表，可以这样引入样式表：

<link rel="stylesheet" type="text/css" media="screen" href="screen.css"><link rel="stylesheet" type="text/css" media="print" href="print.css">

针对响应式布局，有一些很有用的媒介特性描述符，比如说width、aspect-ratio、resolution、orientation等。如果我希望一组图片在PC上横向排列，在手机上纵向排列，可以这样写：

## 自适应布局与CSS框架 @mediaquery

所谓的自适应布局（adaptive layout），即根据不同的分辨率提供不同的布局样式[[12]](" \l "ref_12)。这在弹性盒等响应式布局流行起来之前曾是被广泛使用的布局方式。

比如我希望一篇文章在iPhone和电脑上都能完整显示，可以这样写：

.article {

width: 980px;}

@media (max-width: 320px) {

.article {

width: 320px;

}}

设备类型少的时候，这样写或许没什么问题，但设备类型一多这就是噩梦了：

作者：卡罗  
现在有了[视区单位](https://www.zhihu.com/search?q=%E8%A7%86%E5%8C%BA%E5%8D%95%E4%BD%8D&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra={"sourceType":"answer","sourceId":1779511490}" \t "_blank)和弹性盒，已经不需要这种繁琐的写法。

虽然我们不再需要为不同的设备写死不同的宽度，不过根据不同的宽度判断设备类型也不失为一种办法。事实上很多CSS框架也在继续沿用这样的方式，比如[Bulma](https://link.zhihu.com/?target=https://bulma.io/documentation/overview/responsiveness/" \t "_blank)、[Bootstrap](https://link.zhihu.com/?target=https://getbootstrap.com/docs/5.0/layout/breakpoints/" \t "_blank)等。以Bulma为例，它定义了5个断点[[13]](" \l "ref_13)：

## **最后**

需要注意，虽然简单的网页想同时适配手机端和PC端并不难，但对于本来就有复杂布局的网页来说，同时适配手机端和PC端几乎不可能做到十全十美，可以说是吃力不讨好，所以这类网站通常都先判断用户的设备类型，再分别提供手机端或PC端的网页。