

lesson12 移动端布局.note

创建时间：2018/04/01 13:10

更新时间：2018/04/01 13:16

移动端开发有如下的特点

跑在手机端的web页面（H5页面）；

跨平台； IOS Android

基于webview；

告别IE拥抱webkit；

更高的适配和性能要求。

常见移动web适配方法：

PC端：

外部容器固定宽度960px/1000px居中

盒子模型，定高、定宽

display:inline-block； 盒子横向排列

移动web(移动端)：

定高、宽度百分比

flex弹性盒子

Media Query媒体查询

移动web开发与适配

Media Query媒体查询

```
@media 媒体类型 and (媒体特性){  
    /*css样式*/  
}
```

媒体类型:screen, print...

媒体特性:max-width, max-height...

移动web独有的viewport

通俗的讲，移动设备上的viewport就是设备的屏幕上能用来显示我们的网页的那一块区域，在具体一点，就是浏览器上(也可能是一个app中的webview)用来显示网页的那部分区域，但viewport又不局限于浏览器可视区域的大小，它可能比浏览器的可视区域要大，也可能比浏览器的可视区域要小。

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

viewport是用来干什么的

viewport的默认值，一般来说是大于手机屏幕的。这样就可以做到当我们在浏览桌面端网页的时候，可以让桌面端网页正常显示（我们普通页面设计的时候，一般页面的主区域是以960px来做的

所以980px这个值，可以做到桌面端网页的正常显示）。但是，其实我们手机的屏幕宽度是没有960px的，因此浏览器会出现横向滚动条。同时，即使是基于980的viewport，我们在移动端浏览我们

的桌面页面的体验其实也并不好，所以，一般的，我们会专门给浏览器设计一个移动端的页面。

对viewport的控制

如今可以绝大部分浏览器里（即主流的安卓浏览器和ios），都支持对viewport的一个控制了。一般的，我们会这么写。

viewport默认有6个属性

width: 设置viewport的宽度（即之前所提及到的，浏览器的宽度详），这里可以为一个整数，又或者是字符串"width=device"

initial-scale: 页面初始的缩放值，为数字，可以是小数

minimum-scale: 允许用户的最小缩放值，为数字，可以是小数

maximum-scale: 允许用户的最大缩放值，为数字，可以是小数

height: 设置viewport的高度（我们一般而言并不能用到）

user-scalable: 是否允许用户进行缩放, 'no' 为不允许, 'yes' 为允许

我们把这个标签是在head里面, 像这样<meta name="viewport" content="initial-scale=1">, 这样就可以做到对viewport的控制了

相对单位em: 描述相对于应用在当前元素的字体尺寸, 所以它也是相对长度单位。一般浏览器字体大小默认为16px, 则2em == 32px;

相对单位rem原理与简介

rem: font size of the root element 相对长度单位。相对于根元素(即html元素)font-size计算值的倍数

字体单位: 值根据html根元素大小而定, 同样可以作为宽度、高度单位

适配原理: 将px替换成rem, 动态的修改html的font-size适配

兼容性: ios6以上和android2.1以上, 基本覆盖所有流行的手机系统

单位em 与 单位rem的区别:

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(大部分浏览器默认html根元素字体大小为16px, 但是自己也可以设定一个值);

使用em为元素设置字体大小时, 相对的是当前这个元素的字体的尺寸(例如一个文档中p元素文字大小为10px, 行高2em就是20px;但是另外一个元素span字体大小3em却不一定是30px)。

不同移动端手机适配rem, 动态的修改font-size

1. 使用媒体查询对不同的手机屏幕设置不同的font-size(单独的样式文件作为工具, 做一次就好)

.....不同手机不同分辨率不同品牌根据实际情况都需要做适配, 这样做比较麻烦

2. 使用js的方法(二阶段讲)

拓展: (根据情况设计)

max-device-width和max-width的区别【重要】

1) max-device-width是设备的宽度，例如，真实的设备屏幕宽度。

2) max-width是目标显示区域的宽度，例如，浏览器宽度。

3) 如果使用max-device-width，那么在PC浏览器上浏览网页时，缩小或放大浏览器时是不执行CSS

的，因为“PC设备”没有变化。但如果使用max-width，缩小或放大浏览器时是执行CSS的，因为“显

示区域”即浏览器大小发生了变化。

4) 如果使用max-device-width，那么当手机由竖变横时，CSS是不执行的，因为“手机宽度”并没

有变化。如果使用max-width，那么当手机由竖变横时，CSS是执行的，因为“显示区域”发生了变化。

5) 通常，面向“移动设备”用户使用max-device-width；面向“PC设备”用户使用max-width。

```
/*====移动设备 ==== */
@media screen and (max-device-width: 480px) {
    /* CSS 代码 */
}
/*==== PC ==== */
@media screen and (max-width: 1024px) {
    /* CSS 代码 */
}
```

