

移动端布局

移动端现状

- 了解移动端浏览器的内核、屏幕分辨率的问题；

内核

- 国产主流手机浏览器，内核 Webkit。

屏幕分辨率

- 手机分辨率：**碎片化太多**
- Android：480x800, 480x854, 540x960, 720x1280, 1080x1920 等
- iPhone：640x960, 640x1136, 750x1334, 1242x2208 等
- 2K：手机分辨率

谷歌浏览器

- 是**开发过程中的主要模拟手段**；
- 步骤：
 - 1.鼠标右键 检查 或 F12 控制器界面方向；
 - 2.选择 手机模式
 - 3.选择 手机类型及尺寸；调节适当的显示比例；
 - 4.点右键 查看 页面元素；

viewport（视口）

- 视口：PC端的页面直接放入手机屏显示，不友好。内容原因没有设置viewport；

定义

- 视口：浏览器（PC、移动端）显示页面内容的屏幕区域；不同的屏幕的大小，我们看到的区域也是不同的；
- viewport就是为了解决上面问题；



设置前

- HTML默认为980px不是很合适；



- 那么设置HTML宽度多少为合适呢？我们眼睛能看到的，屏幕的窗口多大，设置多大就合适

语法

- 默认：HTML980px;
- 理想：手机屏幕多大，HTML就设置是多大;
- meta标签设置：

```
<meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
```

- **width=device-width**：改变HTML默认的980px 为 屏幕的宽度;
- user-scalable: 是否允许用户缩放屏幕 值：no(0不允许) yes(1允许);
- initial-scale: 初始化缩放比例; 1.0: 不缩放;
- maximum-scale: 用户对页面的最大缩放比例; 值：比例
- minimum-scale: 用户对页面的最小缩放比例; 值：比例

二倍图

物理像素点

- 指计算机显示设备中的最小单位，即一个像素点的大小。每一个像素点可以理解为就是**屏幕上的一个发光点**。每个点可以发一个颜色，就是我们看到的画面。
- 早期的屏幕，物理像素点都比较大，比如玩游戏的超级玛丽的画面的颗粒感很强：随着技术的进步，物理像素点会被做的越来越小；会被做小；



屏幕分辨率

- 屏幕分辨率：物理像素点的个数来衡量，表示屏幕水平和垂直方向的物理像素点的个数，物理像素点有多少个；
- iPhone3和iPhone4是同一个屏幕尺寸下，比较分辨率：



- Retina（视网膜屏幕）是一种显示技术，可以将把更多的物理像素点压缩至一块屏幕里；从而达到更高的分辨率，并提高屏幕显示的细腻程度。
- 物理像素点：越做越小；（iPhone4坑的宽度是iPhone3物理像素点的宽度一半）



图片分辨率

- 假设：**有200*200分辨率**的图片，展示在宽度分别是320（iphone3）、640(iphone4)分辨率的手机上，展示的效果如下：



- 分析：
 - 200*200分辨率的图片：200个颜色发光点；
 - 一个物理像素点发一个颜色：屏幕都需要200个物理像素点；
 - 物理像素点的宽度大小：320（1）；640（0.5）；
 - 所以显示为上图。

- 问题：不同的屏幕下，显示的图片大小是不一样；
 - 宽度：第2个是第1个的0.5倍；
 - 面积：第2个是第1个的1/4倍；
- 目标：显示一样；
- 解决：
 - 物理像素点的宽度大小：i3 (1) ; i4 (0.5) ;
 - 物理像素点的数量：320 (1 x 200=200长度) 、640(0.5 x 400=200长度) ；都是200长度，才能保证显示的大小一样；
- 移动端：
 - 上面这个都是200长度到底如何设置呢？
 - 设置CSS 宽高200px后，会自动保证每个屏幕显示元素大小一样，
 - **会自动算出不同屏幕下，背后需要提供物理像素点需要多少个。**
 - 针对上面，每个手机会自动算出要给图片准备200个物理像素点、400个物理像素点；

二倍图由来

- 开始：设置CSS像素200px宽，320分辨率（200）,640分辨率（400）；这样才显示一样；
- 像针对640分辨率手机屏（iPhone4），要求设计给400*400图，对应我们CSS设置200px,有二倍的关系；
- 命名：
 - xxxxx@2x.png：二倍图
 - xxxxx@3x.png：三倍图

移动端常见布局

- 了解常见布局不同，针对业务需求选择不同的方案；实际开发过程中，都是混合使用，没有哪一种是绝对的单独使用；

单独制作移动端页面

- 流式布局 (百分比布局)
- flex弹性布局 (强烈推荐)
- less+rem+媒体查询布局
- 混合布局

响应式页面兼容移动端

- 媒体查询
- bootstrap

方案

- 单独制作：淘宝、京东、苏宁手机端都是单独制作的，流式、flex、rem布局、专门针对各种手机屏幕进行开发。
- 响应式：三星电子官网。www.samsung.com/cn/；特点：可兼容PC 移动端，一个页面多个端适配显示；制作起来要考虑到兼容性的样式。
- 选择：公司业务需求；

样式初始化-了解

- 国产移动浏览器基本以 webkit 内核为主，因此我们就考虑webkit兼容性问题。
- normalize.css：大家以前自己写的base.css
 - <http://necolas.github.io/normalize.css/>
 - 保护了有价值的默认值；
 - 修复了浏览器的bug；