**LUO JIN 2020-05-12 (TUES)**

[1，移动端基础 3](#_Toc40656621)

[1.1 浏览器现状 3](#_Toc40656622)

[1.2 手机屏幕现状 3](#_Toc40656623)

[1.3 常见尺寸 https://material.io/devices/ 3](#_Toc40656624)

[1.4 调式方法 chorme DevTools (谷歌浏览器) 模拟手机调试 F12—>移动端/手机调试 3](#_Toc40656625)

[1.5 总结：webkit内核兼容，主要针对手机端，分辨率和尺寸大小，用谷歌浏览器模拟手机调式 3](#_Toc40656626)

[2. 视口 3](#_Toc40656627)

[2.2 视觉视口 visual viewport 3](#_Toc40656628)

[2.3 理想视口 ideal viewport 3](#_Toc40656629)

[2.4 总结 4](#_Toc40656630)

[2.5 meta视口标签 4](#_Toc40656631)

[2.6 标准viewport 设置 4](#_Toc40656632)

[3. 二倍图 4](#_Toc40656633)

[3.1 物理像素/物理像素比 5](#_Toc40656634)

[3.2 多倍图 5](#_Toc40656635)

[3.3 背景缩放 background-size 5](#_Toc40656636)

[3.4 多倍图切图神器： 5](#_Toc40656637)

[4. 移动端开发选择 6](#_Toc40656638)

[4.1 移动端主流方案 6](#_Toc40656639)

[1） 单独制作移动端页面（主流） 6](#_Toc40656640)

[2）响应式页面兼容移动端（其次） 6](#_Toc40656641)

[4.2 总结：还是单独制作是最好的，如果公司没有要求的话 6](#_Toc40656642)

[5. 移动端技术解决方案 6](#_Toc40656643)

[5.1 移动端浏览器 6](#_Toc40656644)

[5.2 CSS3初始化 normalize.css 6](#_Toc40656645)

[5.3 CSS3盒子模型 box-sizing 6](#_Toc40656646)

[1) 传统的盒子模型： 6](#_Toc40656647)

[2）CSS3盒子模型: 6](#_Toc40656648)

[3）移动端： 6](#_Toc40656649)

[5.4 特殊样式 7](#_Toc40656650)

[6. 移动端常见布局 7](#_Toc40656651)

[6.1 单独制作移动端页面（主流） 7](#_Toc40656652)

[流式布局（**百分比布局**） 7](#_Toc40656653)

[Flex弹性布局（强烈推荐） 7](#_Toc40656654)

[7. 京东移动端首页 8](#_Toc40656655)

[7.1 准备工作 8](#_Toc40656656)

[1) 技术选型： 8](#_Toc40656657)

[2）搭建文件夹相关结构 8](#_Toc40656658)

[3）设置视口标签以继引入初始化样式 8](#_Toc40656659)

[4） 流式布局 8](#_Toc40656660)

[5) 常用的初始化样式 8](#_Toc40656661)

[总结：在320-640px之间按照100%显示，可以按照屏幕进行缩放 9](#_Toc40656662)

[7.2 内容部分 9](#_Toc40656663)

[7.3 图片格式 9](#_Toc40656664)

Keyword: **流式布局**

开发现状 移动端基础

标准的viewport 视口 二倍图

Web调式方法 移动端调式

常见的布局方式 移动端常见布局 移动端技术解决方案

描述流式布局

京东首页案例

# 1，移动端基础

## 1.1 浏览器现状

PC端浏览器 ： 360， 谷歌，火狐，QQ，百度，IE, 搜狗

移动端浏览器： UC, QQ, 欧鹏，百度， 360，搜狗，谷歌

**总结：兼容移动端主流浏览器，处理Webkit内核浏览器即可**。国内的UC,QQ等都是根据Webkit修改过来的内核，国内暂无自主研发的内核，国外的手机操作系统都是基于Android修改开发。

## 1.2 手机屏幕现状

碎片化严重，480x800,….

## 1.3 常见尺寸 https://material.io/devices/

## 1.4 调式方法 chorme DevTools (谷歌浏览器) 模拟手机调试 F12—>移动端/手机调试

搭建本地web服务器，手机和服务器一个局域网内，通过手机访问浏览器

使用外网服务器，直接IP或者域名访问

## 1.5 总结：webkit内核兼容，主要针对手机端，分辨率和尺寸大小，用谷歌浏览器模拟手机调式

# 2. 视口

viewport 就是浏览器显示页面内容的屏幕区域，视口可以分为布局视口，视觉视口和理想视口

2.1 布局视口 layout viewport

一般移动设备的浏览器默认的一个布局视口，用于解决早期的PC端也i按在手机上显示的问题

IOS，Android基本都将这个视口分辨率设置为980px, 所以PC上页面大多都能在手机上呈现，只不过元素看上去都很小，一般默认可以手动缩放

不适合页面开发

## 2.2 视觉视口 visual viewport

用户能看到的网站的区域

可以通过缩放去操作视觉视口，但不会影响我们的布局视口，布局视口仍保持原来宽度

## 2.3 理想视口 ideal viewport

在移动端最想的浏览和阅读宽度设定

手动写meta视口标签通知浏览器操作

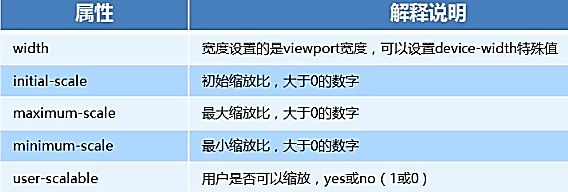
Meta标签主要目的，布局视口的宽度应该与理想视口的宽度一样，简单理解设备有多宽，我们布局视口就有多宽

乔布斯

## 2.4 总结

## 2.5 meta视口标签

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



## 2.6 标准viewport 设置

视口宽度和设备宽度保持一致

视口默认缩放比例1.0

不允许用户自行缩放

最大允许的缩放比例1.0

最小允许许缩放比例1.0

# 3. 二倍图

物理像素/物理像素比

**物理像素点**：屏幕显示的最小颗粒，事物理真是存在的。这是厂商在出场时就设置好的。比如苹果6/7/8 是750 x 1334

**PC端，我们开发的时候1px等于1个物理像素的，但是移动端就不尽相同**

Iphone8: 375px（开发时候的像素） = 750px(物理像素) 🡺 1px = 2个物理像素

物理像素比/屏幕像素比：一个px能显示的物理像素点的个数

物理像素=分辨率

**iPhone8 的物理像素是750px**

**iPhone8中 1个开发像素 = 2个物理像素**

## 3.1 物理像素/物理像素比

PC端和早前的手机屏幕/普通手机屏幕：1CSS像素 = 1 物理像素

Retina(视网膜屏)是一种显示技术，可以将把更多的物理先你告诉压缩在一块屏幕里边，从而达到更高的分辨率，并提高屏幕显示的细腻程度。

## 3.2 多倍图

1）对于一张50\*50px的图片,在手机Retina屏中打开，按照刚才的物理像素比会放大倍数，这样会造成图片模糊

2）在标准在viewport设置中，使用倍图来提高图片质量，解决在高清设备中的模糊问题

3）通常使用二倍图，因为iPone6/7/8的影响，但是现在海逊在3倍图4倍图的情况，这个看公司需求

4）背景图片注意缩放问题

6）假如：我们需要一个50x50px(CSS像素)的图片，直接放进ipone8会放大2倍100x100,就会模糊

7）解决：放一个100x100 图片，手动缩小为50\*50（css像素）

8）**2倍图**：我们准备的图片比我们的实际需要大2倍

## 3.3 背景缩放 background-size

**background-size**属性规定背景图像的尺寸: **width height**

先有背景才可进行设置

单位：**长度/百分比/cover/contain**

**Cover**： 覆盖，拉伸至完全把整个盒子盖住，会有一部分看不见

**Contain**：图像扩大至填满盒子，宽或高等比例拉伸，宽或高到了就不再拉伸了，可能有部分空白区域

1）我们有50\*50的盒子需要一个背景图片，但是根据分析这个图片还是要准备2倍100\*100

2）我们需要把这个图片缩放一般也就是50\*50， background-size

3.4 多倍图切图神器： cutterman

# 4. 移动端开发选择

## 4.1 移动端主流方案

### 1） 单独制作移动端页面（主流）

淘宝，京东，苏宁易购

根据设备进行判断 ,通常域名前加上m(mobile)可以打开移动端

### 2）响应式页面兼容移动端（其次）

三星电子官网：samsung

通过改变屏幕判断浏览器屏幕宽度来改变大小，以适应不同终端

缺点：制作麻烦，许哟啊花很大精力去调式兼容性问题

## 4.2 总结：还是单独制作是最好的，如果公司没有要求的话

# 5. 移动端技术解决方案

## 5.1 移动端浏览器

基本以Webkit为主：QQ, 百度，safari， UC, 🡪H5/CSS3,

同时我们浏览器的如果要兼容老版本的话我们只需要考虑添加-webkit-就可以了

## 5.2 CSS3初始化 normalize.css

1) <http://necolas.github.io/normalize.css>

2) npm i normalize.css

## 5.3 CSS3盒子模型 box-sizing

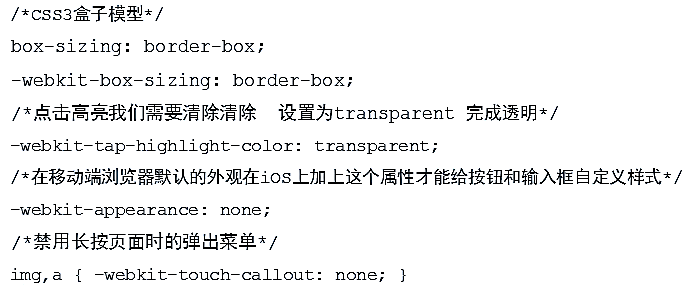
1) 传统的盒子模型：盒子宽 = CSS中设置的width+border+padding 🡪**box-sizing: content-box;从内容去开始计算的**

2）CSS3盒子模型: 盒子宽 = CSS中设置的宽度width里边包含了border+padding，也就是说，我们的**CSS3中的盒子模型，padding和border不会撑大盒子** 🡪 **box-sizing: border-box**;(这句话让盒子变成CSS3盒子模型) 一般浏览器不支持在PC端； 从边框开始算的，边框已经是盒子的一部分了

3）移动端：用CSS3即可

PC: 看情况而定，一般用传统的盒子模型

## 5.4 特殊样式



# 6. 移动端常见布局

不需要定义1000px的布局之类的

移动端技术选型

## 6.1 单独制作移动端页面（主流）

### 流式布局（**百分比布局**）

1. **百分比布局**，非固定像素布局
2. 通过盒子的宽度设置成百分比来根据屏幕的宽度来进行伸缩，不受固定像素的显示，内容两侧填充
3. 流式布局是移动web开发使用的比较常见的一种布局方式
4. 会随着屏幕大小改变而改变，设置最大值最小值在合理范围之类，max-width; min-width, 太大太小内容不适应 margin： auto： 让他居中显示

### Flex弹性布局（强烈推荐）

1. Less+rem+媒体查询布局
2. 混合布局

6.2 响应式页面兼容移动端（其次）

1. 媒体查询
2. Bootstarp

# 7. 京东移动端首页

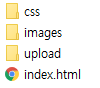
## 7.1 准备工作

### 1) 技术选型：

方案：采取单独制作页面方案

技术：布局采用流式布局

### 2）搭建文件夹相关结构



Upload: 新产品图片文件夹

### 3）设置视口标签以继引入初始化样式

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

    <link rel="stylesheet" href="css/normalize.css">

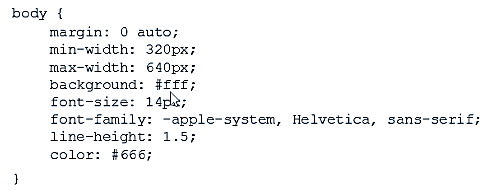
<link rel="stylesheet" href="css/index.css">

### 4） 流式布局

Width：百分比

Body🡪 width: 100%, 不可以无限制放大缩小，内容会跑出去。**max-width: 640px**:, **min-width:320px;**

### 5) 常用的初始化样式



## 总结：在320-640px之间按照100%显示，可以按照屏幕进行缩放

## 7.2 内容部分

1. 页面所有的宽度100%，最大最小有限制
2. 划分盒子
3. 导航搜索栏固定定位

## 7.3 图片格式

DPG 格式：京东自主研发，压缩，节省50%的浏览器流量

Webp格式：谷歌开发 加快加载速度