

移动 Web 第三天

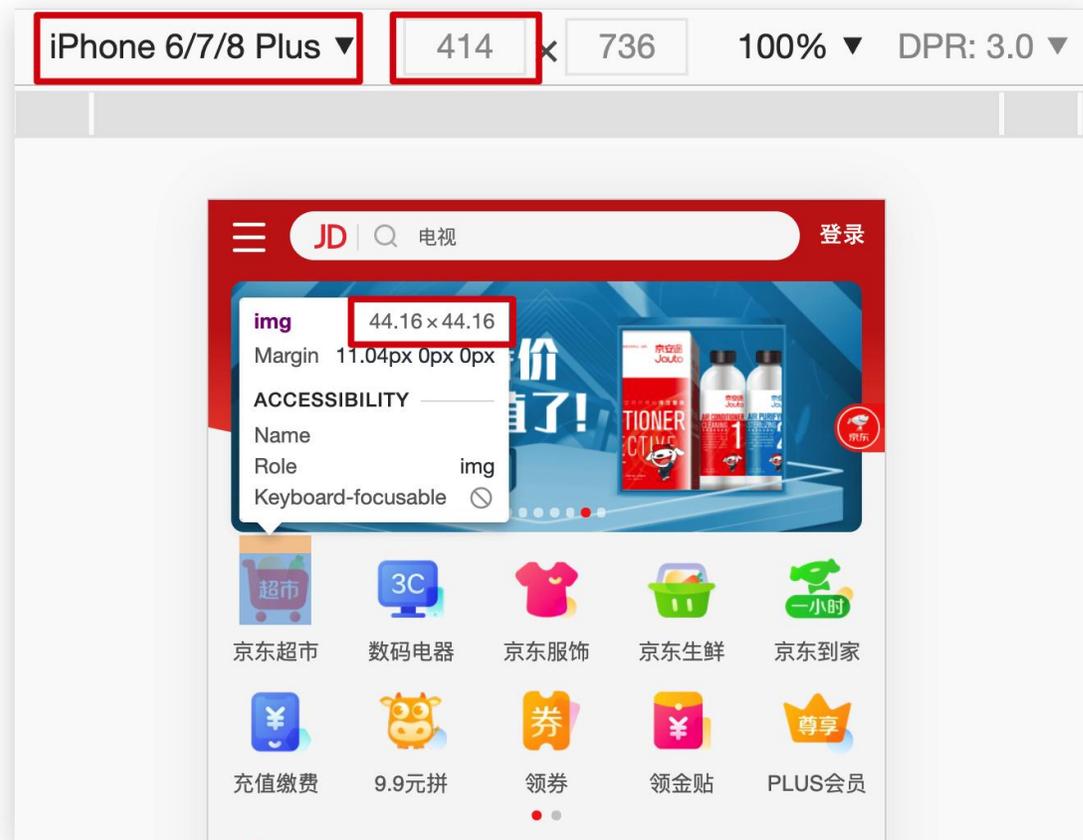
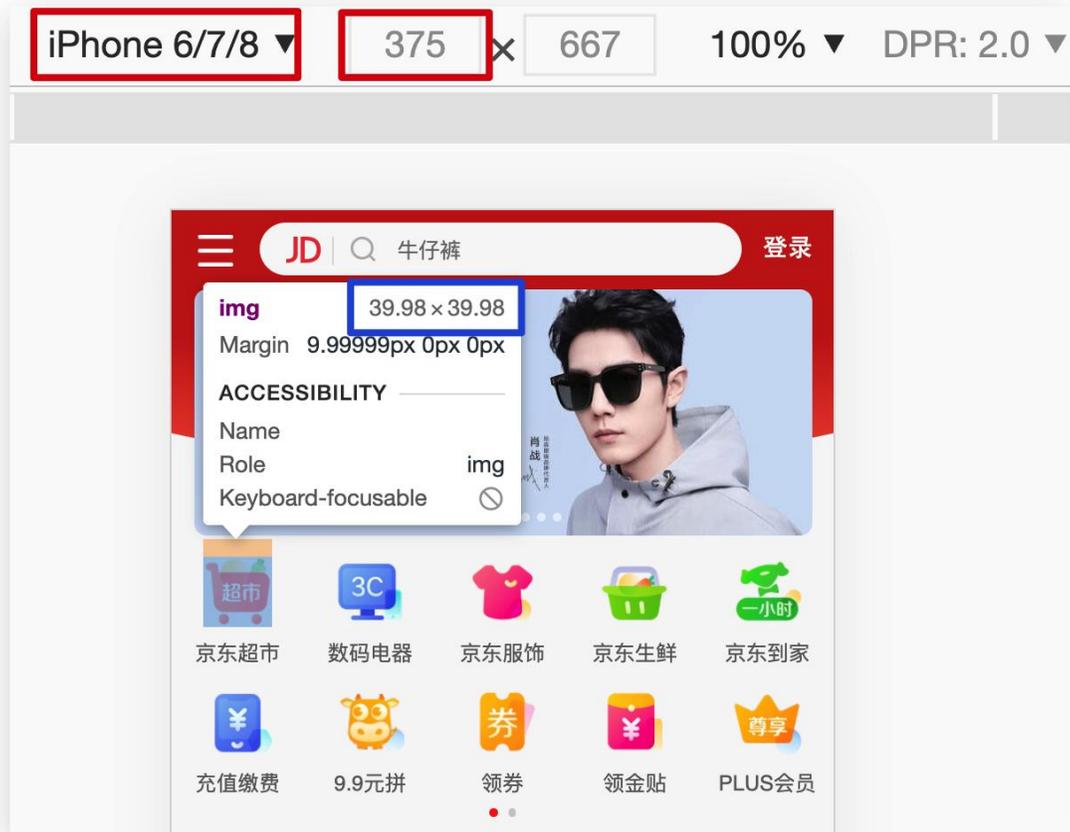
移动适配



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

移动适配



移动适配方案

- rem
- vw

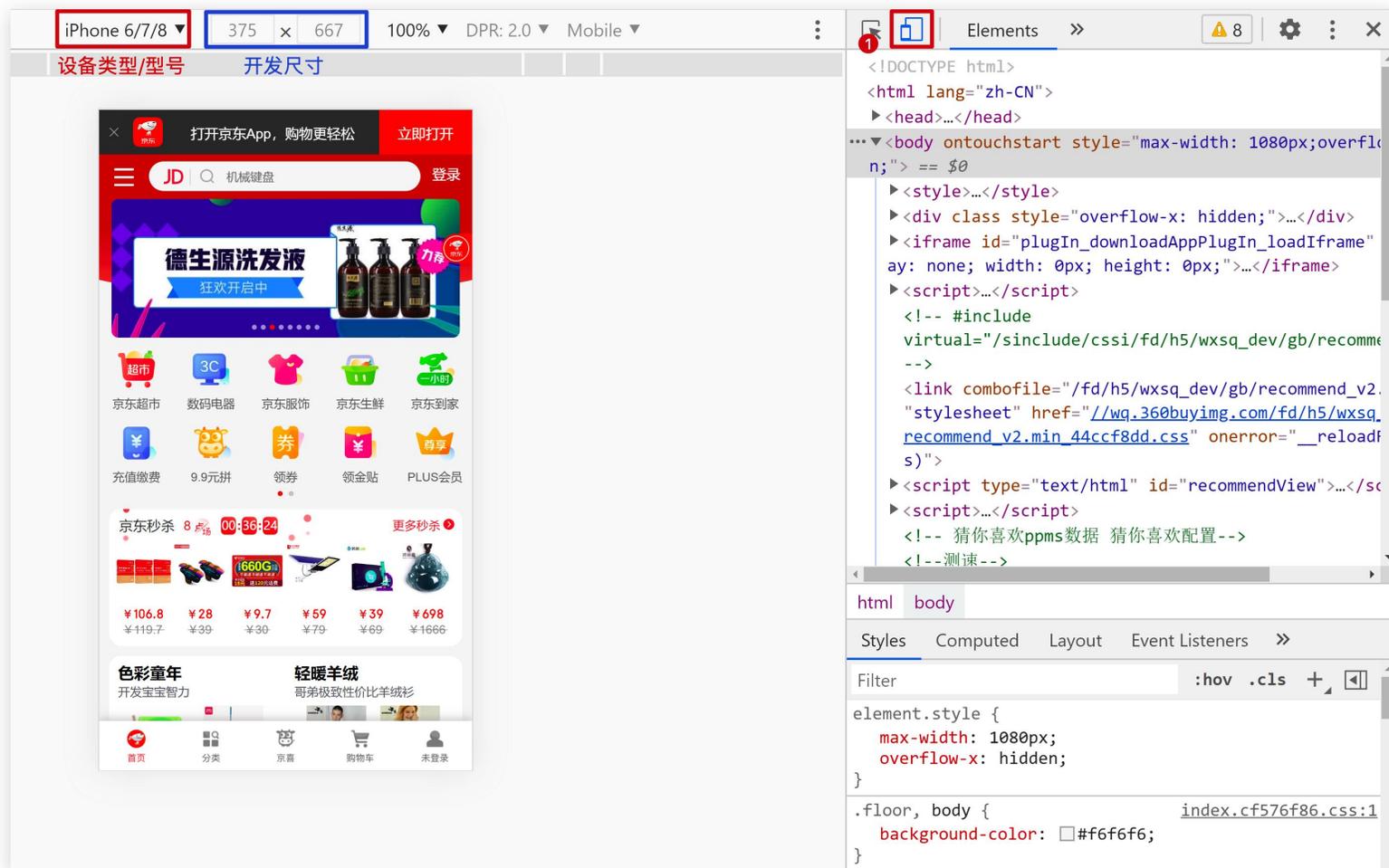
rem 适配方案

less



移动Web基础

谷歌模拟器



屏幕分辨率

- 屏幕分辨率：纵横向上的像素点数，单位是px
- PC 分辨率
 - ❑ 1920 * 1080
 - ❑ 1366 * 768
 - ❑
- 缩放 150%
 - ❑ 1920 / 150%
 - ❑ 1080 / 150%
- 总结
 - ❑ 硬件分辨率 → 物理分辨率（出厂设置）
 - ❑ 缩放调节的分辨率 → 逻辑分辨率（软件/驱动设置）

缩放与布局

更改文本、应用等项目的大小

150% (推荐) ▾

高级缩放设置

显示分辨率

1920 × 1080 (推荐) ▾

显示方向

横向 ▾

屏幕分辨率



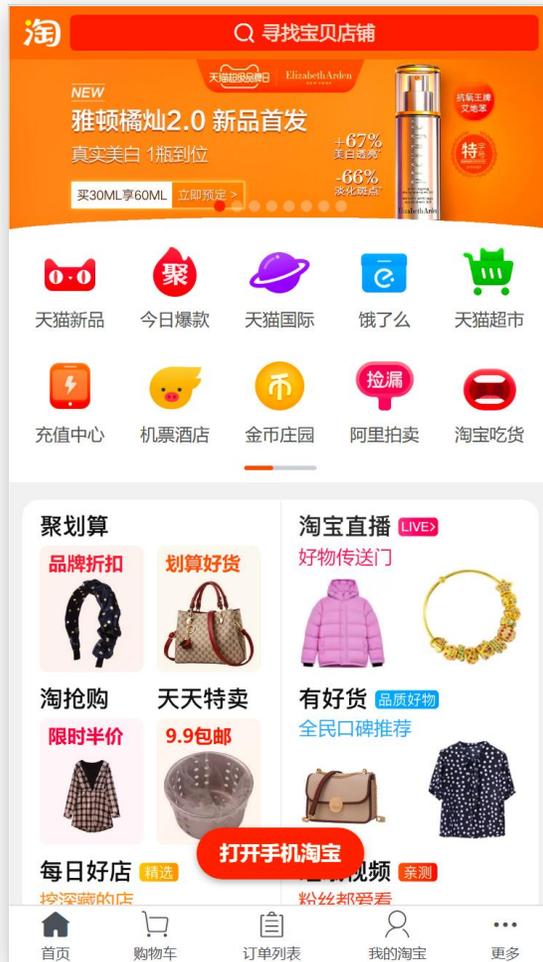
结论：制作网页参考 **逻辑分辨率**

屏幕分辨率

手机型号	物理分辨率	逻辑分辨率	比例关系
iPhone3GS	320 * 480	320 * 480	1 : 1
iPhone4	640 * 960	320 * 480	2 : 1
iPhone5	640 * 1136	320 * 568	2 : 1
iPhone6/7/8	750 * 1334	375 * 667	2 : 1
iPhone6/7/8 Plus	1080 * 1920	414 * 736	2.6 : 1
iPhone11Pro/X/XS	1125 * 2436	375 * 812	3 : 1
iPhone11/XR	828 * 1792	414 * 896	2 : 1
iPhone11 Pro Max/XS Max	1242 * 2688	414 * 896	3 : 1
iPhone12 mini	1080 * 2340	360 * 780	3 : 1
iPhone12/iPhone12 Pro	1170 * 2532	390 * 844	3 : 1
iPhone12 Pro Max	1284 * 2778	428 * 926	3 : 1

视口

- 手机屏幕尺寸不同，网页宽度均为100%
- 网页的宽度和逻辑分辨率尺寸相同



视口

- 视口：显示HTML网页的区域，用来约束HTML尺寸

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <!-- 视口标签 -->
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Document</title>
</head>
<body>

</body>
</html>
```

- `width=device-width`: 视口宽度 = 设备宽度
- `initial-scale=1.0`: 缩放1倍（不缩放）

二倍图

概念：设计稿里面每个元素的尺寸的**倍数**

作用：防止图片在高分辨率屏幕下模糊失真



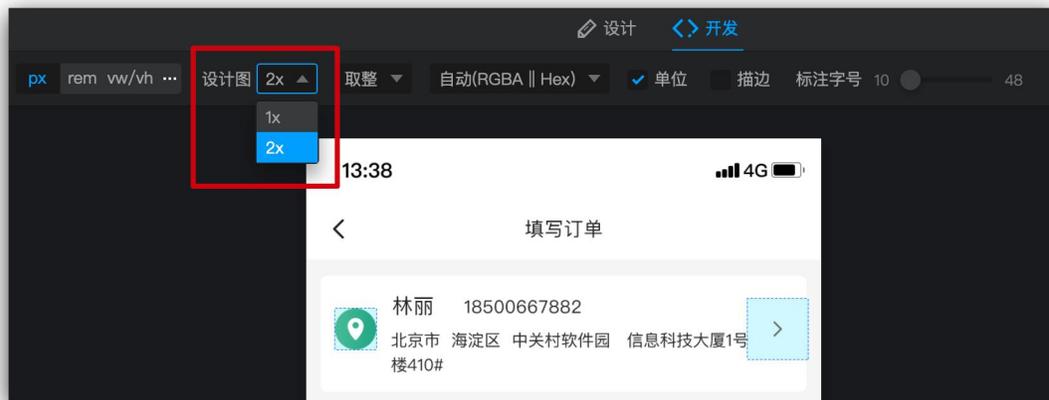
二倍图

作用：防止图片在高分辨率屏幕下模糊失真

手机型号	物理分辨率	逻辑分辨率	比例关系
iPhone3GS	320 * 480	320 * 480	1 : 1
iPhone4	640 * 960	320 * 480	2 : 1
iPhone5	640 * 1136	320 * 568	2 : 1
iPhone6/7/8	750 * 1334	375 * 667	2 : 1
iPhone6/7/8 Plus	1080 * 1920	414 * 736	2.6 : 1
iPhone11Pro/X/XS	1125 * 2436	375 * 812	3 : 1
iPhone11/XR	828 * 1792	414 * 896	2 : 1
iPhone11 Pro Max/XS Max	1242 * 2688	414 * 896	3 : 1
iPhone12 mini	1080 * 2340	360 * 780	3 : 1
iPhone12/iPhone12 Pro	1170 * 2532	390 * 844	3 : 1
iPhone12 Pro Max	1284 * 2778	428 * 926	3 : 1

二倍图

- 现阶段设计稿参考iPhone6/7/8，设备宽度375px产出设计稿。
- 二倍图设计稿尺寸：750px。





适配方案

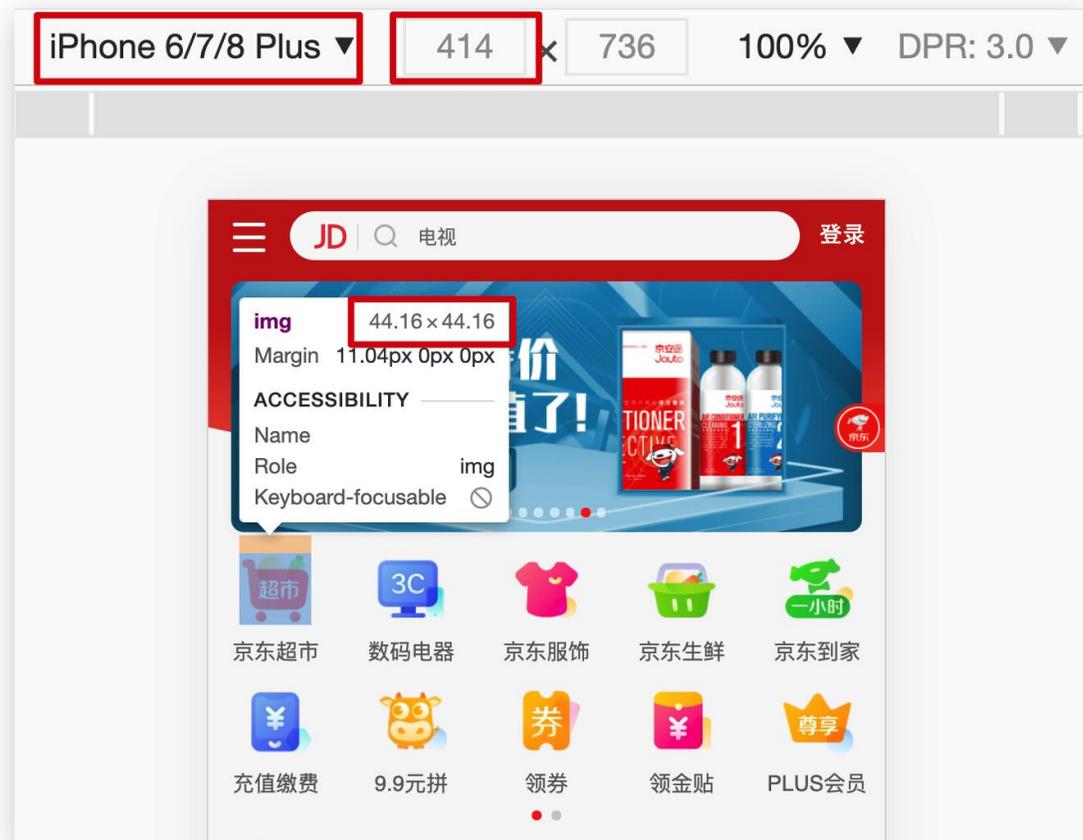
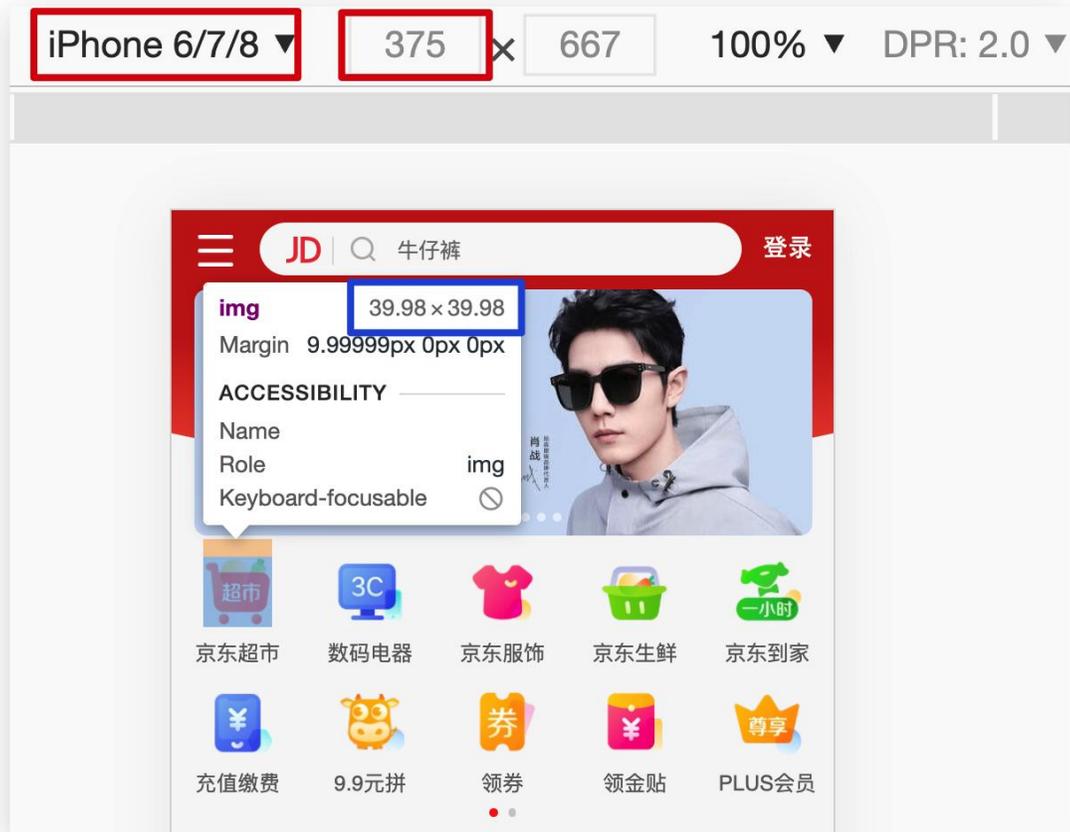
适配方案

- 宽度适配：宽度自适应
 - 百分比布局
 - Flex 布局
- 等比适配：宽高等比缩放
 - rem
 - vw



rem 适配方案

rem



rem

- rem单位，是相对单位
- rem单位是相对于HTML标签的字号计算结果
- $1\text{rem} = 1\text{HTML}$ 字号大小

rem

- 思考
- 1. 手机屏幕大小不同，分辨率不同， 如何设置不同的HTML标签字号？
- 2. 设备宽度不同，HTML标签字号设置多少合适？

媒体查询

思考

- 手机屏幕大小不同，分辨率不同， 如何设置不同的HTML标签字号？
 - 媒体查询
- 媒体查询能够检测视口的宽度，然后编写差异化的 CSS 样式
- 当某个条件成立，执行对应的CSS样式

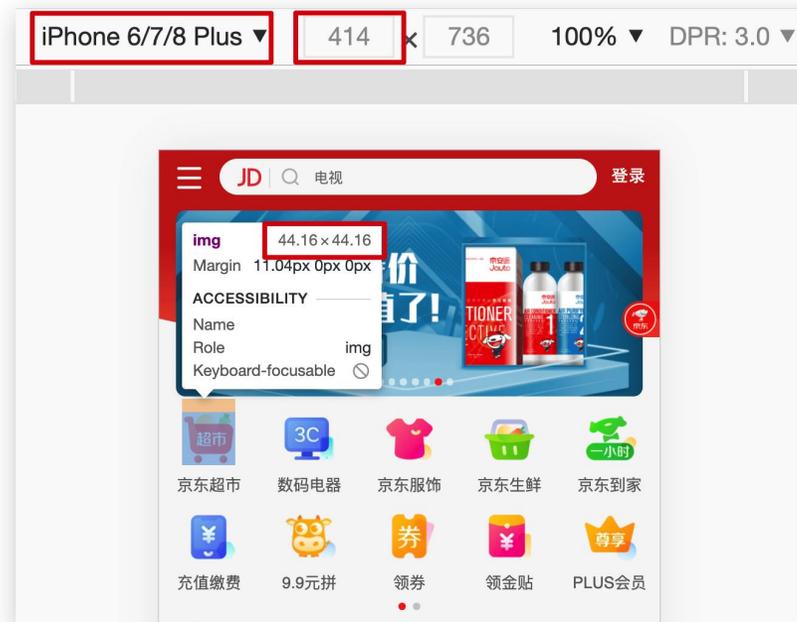
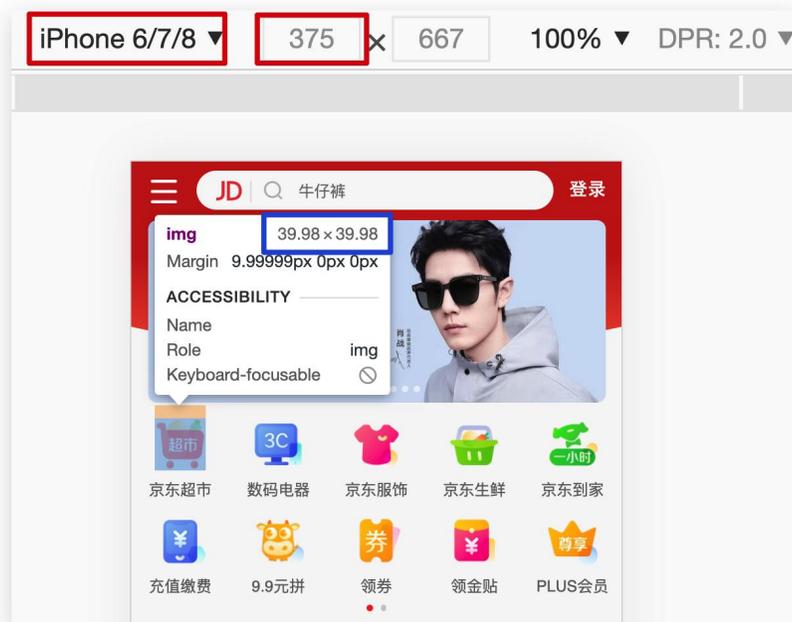
```
@media (媒体特性) {  
    选择器 {  
        CSS属性  
    }  
}
```

```
@media (width:320px) {  
    html {  
        background-color: green;  
    }  
}
```

rem

思考

- 设备宽度不同，HTML标签字号设置多少合适？
 - 设备宽度大，元素尺寸大
 - 设备宽度小，元素尺寸小



rem

目前rem布局方案中，将网页等分成10份，HTML标签的字号为视口宽度的 1/10

```
/* 视口宽度320px, 根字号为32px */
@media (width: 320px) {
  html {
    font-size: 32px;
  }
}

/* 视口宽度375px, 根字号为37.5px */
@media (width: 375px) {
  html {
    font-size: 37.5px;
  }
}

/* 视口宽度414px, 根字号为41.4px */
@media (width: 414px) {
  html {
    font-size: 41.4px;
  }
}
```

rem - flexible.js

- flexible.js 是手淘开发出的一个用来适配移动端的 js 库。
- 核心原理就是根据不同的视口宽度给网页中 html 根节点设置不同的 font-size。

```
<body>
  .....
  <script src="./js/flexible.js"></script>
</body>
```

rem - 移动适配

思考:

- 工作中，书写代码的尺寸要参照设计稿尺寸，设计稿中是px还是rem?
- 如何确定rem的数值?



rem - 移动适配

- 目标：计算68px是多少个rem?(设计稿适配375px视口)
- $N * 37.5 = 68$
- $N = 68 / 37.5$

- rem单位尺寸

1. 确定基准根字号

- 查看设计稿宽度 → 确定参考设备宽度(视口宽度) → 确定基准根字号 (1/10视口宽度)

2. rem单位的尺寸

- rem单位的尺寸 = px单位数值 / 基准根字号



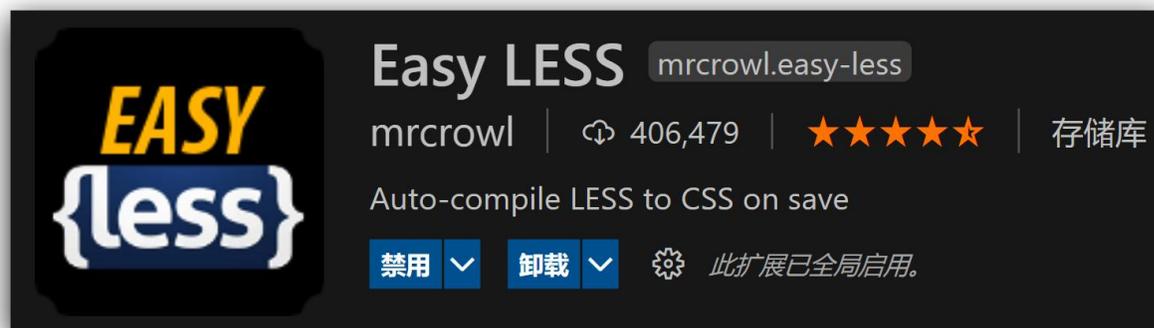


less

- 思考：在px单位转换到rem单位过程中，哪项工作是最麻烦的？
- 答：除法运算。CSS不支持计算写法。
- 解决方案：可以通过Less实现。

less - 简介

- Less是一个CSS预处理器, Less文件后缀是**.less**。扩充了 CSS 语言, 使 CSS 具备一定的逻辑性、计算能力
- 注意: 浏览器不识别 Less 代码, 目前阶段, 网页要引入对应的 CSS 文件
- VS Code 插件: **Easy LESS**, 保存 less文件后自动生成对应的 CSS 文件



less - 注释

注释:

- 单行注释
 - 语法: `// 注释内容`
 - 快捷键: `ctrl + /`
- 块注释
 - 语法: `/* 注释内容 */`
 - 快捷键: `Shift + Alt + A`

less - 运算

运算：

- 加、减、乘直接书写计算表达式
- 除法需要添加 **小括号** 或 **.**

注意：表达式存在多个单位以**第一个单位**为准

```
.box {  
  width: 100 + 50px;  
  height: 5 * 32px;  
  
  width: (100 / 4px);  
  height: 100 ./ 4px;  
}
```

less - 嵌套

作用：快速生成后代选择器

```
.父级选择器 {  
  // 父级样式  
  .子级选择器 {  
    // 子级样式  
  }  
}
```

```
.father {  
  color: red;  
  .son {  
    width: 100px;  
  }  
}
```

提示：用 & 表示当前选择器，不会生成后代选择器，通常配合伪类或伪元素使用

```
.father {  
  color: ■ red;  
  &:hover {  
    color: ■ green;  
  }  
}
```



```
.father {  
  color: ■ red;  
}  
.father:hover {  
  color: ■ green;  
}
```

less - 变量

概念：容器，存储数据

作用：存储数据，方便使用和修改

语法：

- 定义变量：@变量名: 数据;
- 使用变量：CSS属性: @变量名;

```
// 定义变量
@myColor: pink;

// 使用变量
.box {
  color: @myColor;
}

a {
  color: @myColor;
}
```

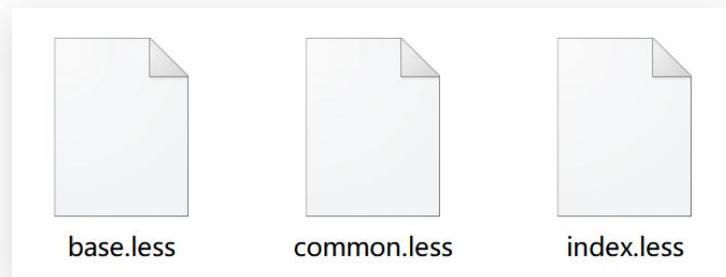
less - 导入

作用：导入 less 公共样式文件

语法：导入：`@import` “文件路径”；

提示：如果是 less 文件可以省略后缀

```
@import './base.less';  
@import './common';
```



less - 导出

写法：在 less 文件的**第一行**添加 `// out: 存储URL`

提示：文件夹名称后面添加 `/`

```
// out: ./index.css  
// out: ./css/
```

less - 禁止导出



写法：在 less 文件第一行添加：`// out: false`

```
1 // out:false
```

案例

项目 - 极速问诊

- ◆ 项目目录
- ◆ 设计稿





传智教育旗下高端IT教育品牌