昨日知识点回顾：

01、使用zepto实现左侧点击操作

02、移动端touch事件点透问题+fastclick插件的使用

03、iscroll插件的使用

04、swipe和swiper的插件的使用

05、响应式布局

今日知识点：

01、bootstarp布局容器+栅格系统的使用

02、less环境搭建和语法说明

03、wjs页面结构搭建+wjs页面头部的完成

04、bootstarp导航组件说明及wjs导航的完成

05、轮播图效果的实现

# 1 bootstarp布局容器+栅格系统的使用

## 1.1 01-移动web-bootstrap中的布局容器

1. **container和container-fluid的区别？**
   1. **container (在不同屏幕尺寸container显示不同的一个默认宽度)**

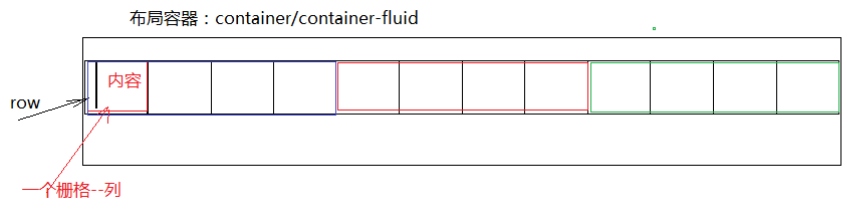
w>1200:1170   
w:992~1200:970  
w:768~992:750  
w<768:100%

**b) container-fluid 实时显示全屏 即宽度100%**

## 1.2 02-移动web-bootstrap中的栅格系统

**1、什么是栅格系统**

**栅格系统**就是来描述当前子元素 在指定屏幕下 占据当前屏幕默认12等分中的n（1-12）等分



1. **如何创建一个栅格系统**
   1. 先添加布局容器container/container-fluid
   2. 在布局容器中添加row
   3. 在row中添加子元素，设置栅格样式
   4. 在子元素中添加具体内容
2. **栅格系统类名的了解**
   1. xs 移动端： w<768 col-xs-3 （表示xs屏及以上屏占3等份） 宽度为33.33%
   2. sm 平板： 768-992 col-sm-3 （表示sm屏及以上屏占3等份） 宽度为33.33%
   3. md 小屏pc: 992-1200 col-md-3 （表示md屏及以上屏占3等份） 宽度为33.33%
   4. lg 大屏pc: w>1200 col-lg-3 （表示lg屏及以上屏占3等份） 宽度为33.33%

## 1.3 03-移动web-bootstrap栅格系统的其它样式说明(上)

1. **offset和push及pull的区别**

**offset:偏移** 偏移的距离是以栅格做为单位 但是这种偏移会直接影响后面的元素， 有可能造成后面的元素换行显示 因为它是通过margin-left属性来实现的

**push(往后-右推)/pull(往前-左拉)** 它是通过定位来实现的，所以不会影响其它元素， 但是有可能和其它元素重叠

## 1.4 04-移动web-bootstrap栅格系统的其它样式说明(下)

1. **栅格嵌套时候container的问题**

1.如果在外层没有再包含container,那么嵌套列的宽度就是参参照当前所在的栅格  
2.如果外层添加了container,那么参照就是核心样式文件所设置的容器宽度

**提示：如果加了container ,container的宽度会参照一下取值**

w>1200:1170   
w:992~1200:970  
w:768~992:750  
w<768:100%

1. **响应式工具的使用**
   1. **hidden-xs/hidden-sm/hidden-md/hidden-lg 分别在不同的屏幕尺寸下会隐藏**

# 2 less环境搭建和语法说明

## 2.1 05-移动web-less环境的搭建

1. **less是干什么的？**

Less 是动态的样式表语言，通过简洁明了的语法定义，使编写 CSS 的工作变得非常简单，本质上，less 包含一套自定义的语法及一个解析器。

1. **less如何搭建？**
   1. **安装node** 
      1. **通过命令行：win键+R > 输入cmd**
      2. **输入：node -v (检测node是否安装成功)**
      3. **输入：npm -v (检测node packag manager是否安装成功)**

**b) 安装less**

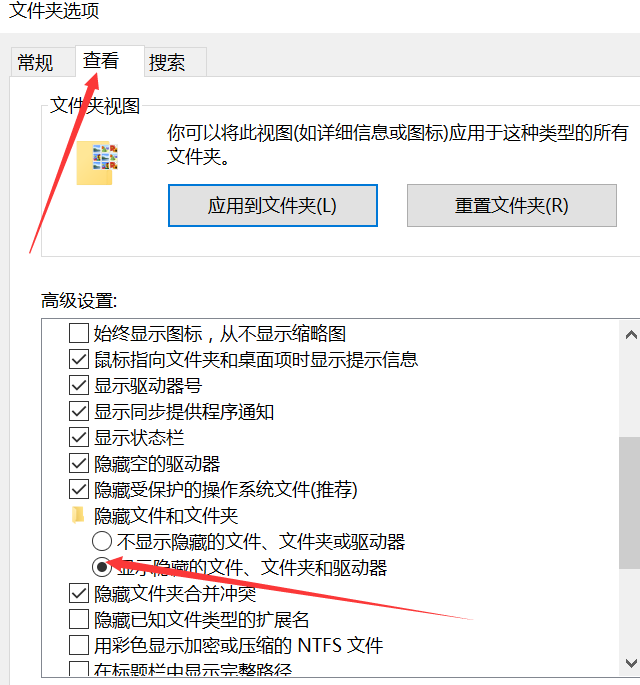
1. **在线安装 直接输入: npm install -g less (检验：lessc -v)**
2. **离线安装**

**i. 复制教学资料npm中的文件**

**ii. C盘 > 用户 > d自己起的电脑名 (如我的叫silence) > AppData > Roaming > npm (再把刚才复制的文件粘贴在npm文件中)**

**Iii. lessc -v检测是否安装成功**

**提示：**如果APPData是隐藏的，C>用户>当前用户目录>查看>选项



**3、webstorm中less文件的解析**

a.新建一个less文件

b在less文件里面建一个less.less

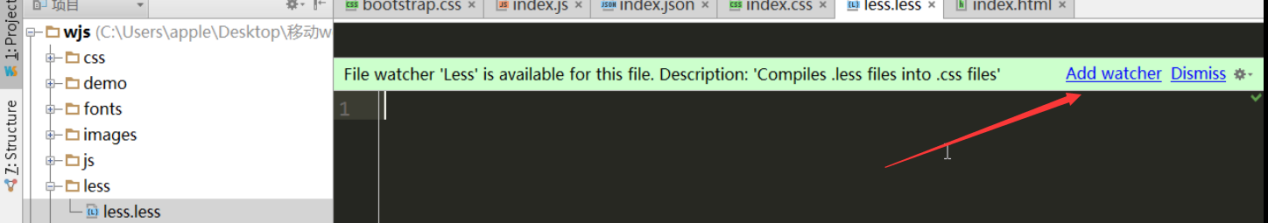
**i.命令行手动解析**less.less

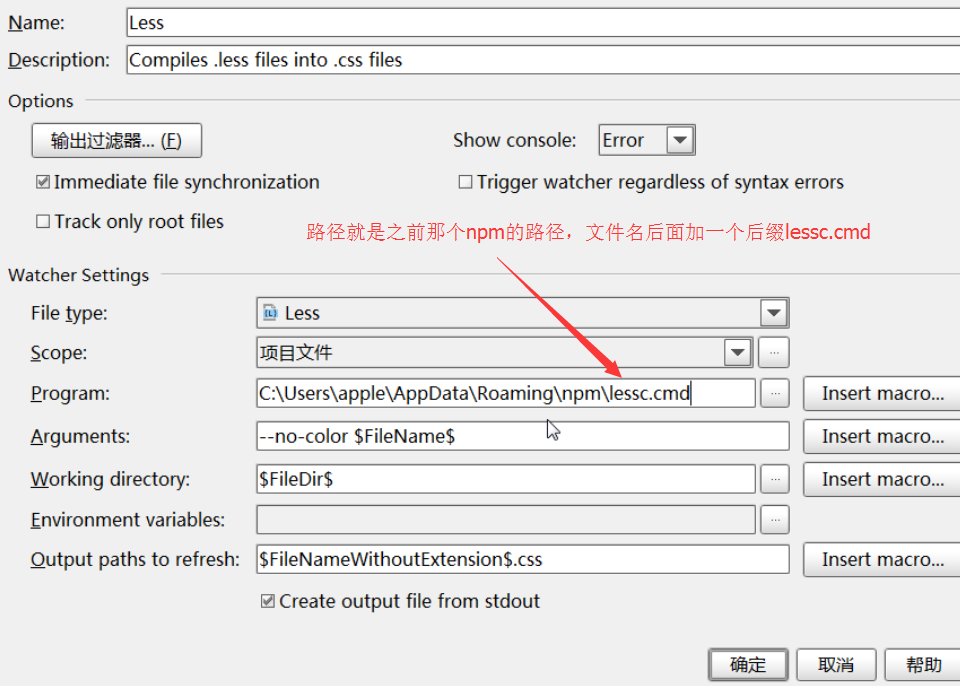
找到刚才创建的less.less >在命令行中输入cd less.less在项目中的路径 按下回车>检测是否正确输入：dir /w 》

在输入预编译less.css :**lessc less.less less.css**

**ii.编译器里面解析** text.less

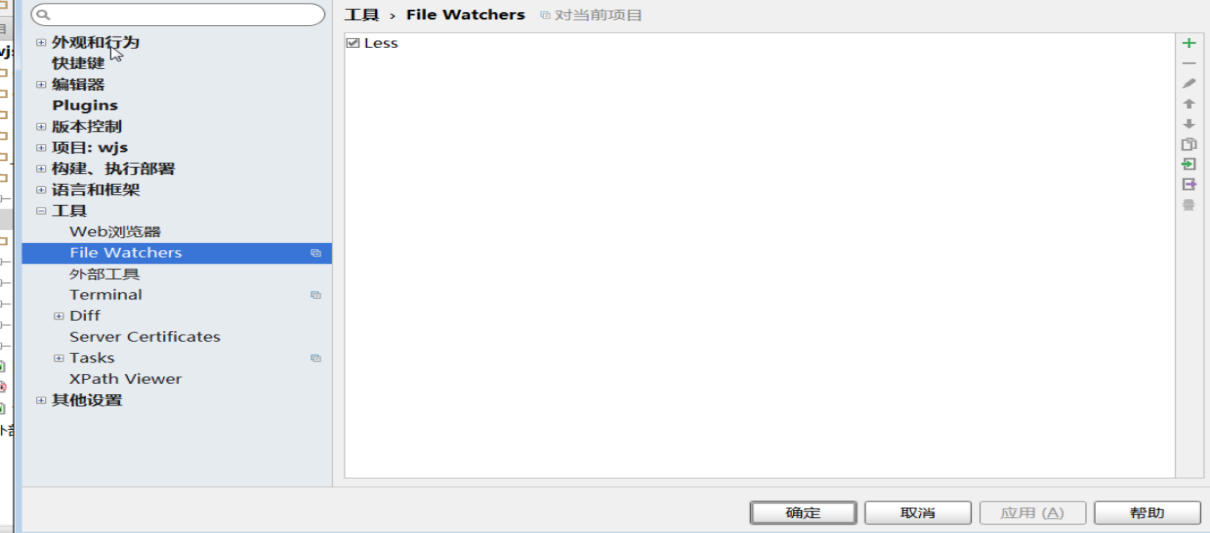
1》.出现如下提示点击add watch





2》检查是否添加好less

Setting>Tools>file watch (如下图)



## 2.2 06-移动web-less语法说明

**1、less.less注释：**

1. less中 // 在less.css中看不到
2. less中 /\* \*/ 在less.css中可以看到
3. **变量**
   1. @变量名：值； 例如：#aa:red; a{color:@aa}
4. **混入 (可以将一个样式引入到另外一个样式中)**
   1. .add{border-radius:20px}

.abc{height:100px ;.add}

* 1. .add(@aa){border-radius:@aa}

.abc{.add(200px)}

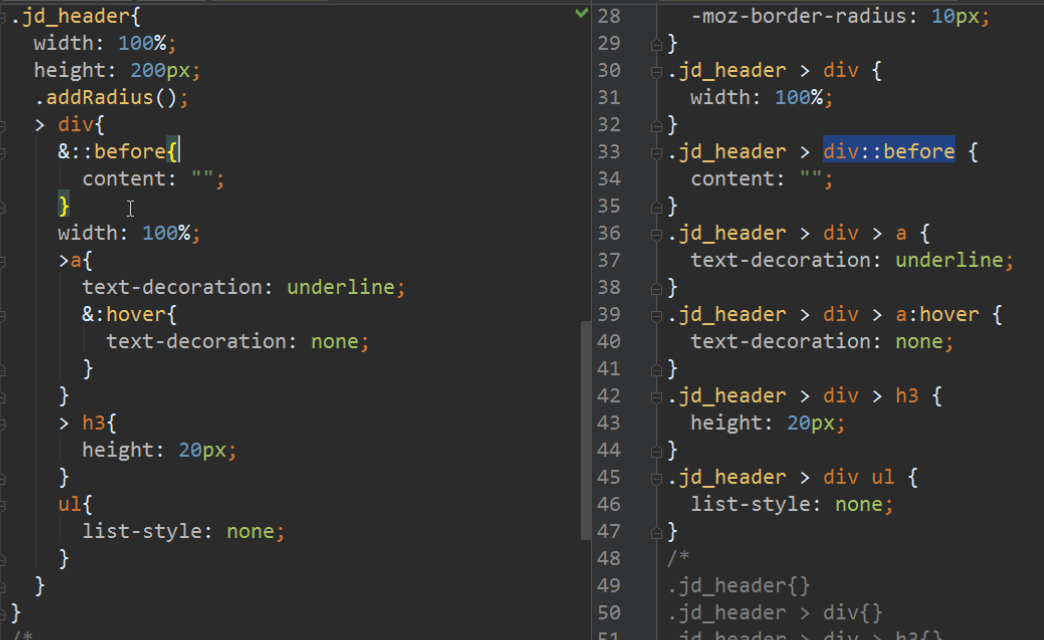
* 1. .add(@aa:10px){border-radius:@aa}

.abc{.add()} 此时不传值就是默认的 .abc{border-radius:10px}

.abc{.add(5px)} 此时传值 .abc{border-radius:5px}



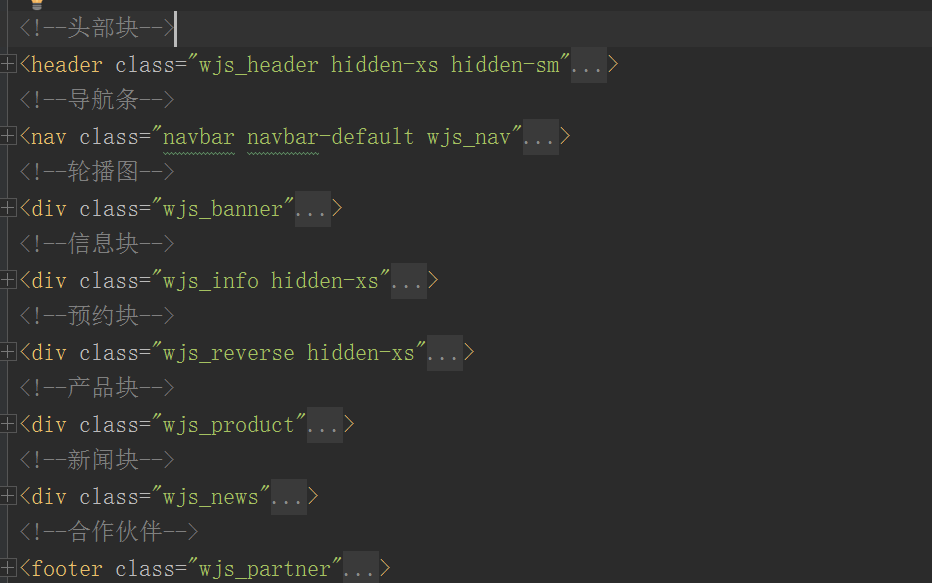
1. **嵌套**



# 3 wjs页面结构搭建+wjs页面头部的完成

## 3.1 07-移动web-wjs页面的整体结构

整体结构是分成了八大快



## 3.2 08-移动web-微金所页面的字体图标样式准备

@font-face {  
 font-family: 'wjs';  
 src: url('../fonts/MiFie-Web-Font.eot'); /\* IE9\*/  
 src: url('../fonts/MiFie-Web-Font.eot') format('embedded-opentype'), /\* IE6-IE8 \*/  
 url('../fonts/MiFie-Web-Font.woff') format('woff'), /\* chrome、firefox \*/  
 url('../fonts/MiFie-Web-Font.ttf') format('truetype'), /\* chrome、firefox、opera、Safari, Android, iOS 4.2+\*/  
 url('../fonts/MiFie-Web-Font.svg') format('svg'); /\* iOS 4.1- \*/  
}  
/\*自定义字体使用样式\*/  
.wjs\_icon{  
 font-family: wjs;  
}

## 3.3 09-移动web-微金所头部块制作(上)

1. 页面的布局：栅格系统
2. 按钮调用框架提供好的 (框架提供位置 : 全局样式>按钮)

## 3.3 10-移动web-微金所头部块制作(中)

**步骤分析：**

1. less来编写css (先安装好)
2. 对容器设置高度和居中+底边框
3. 对前三个列加右边框 nth-of-type(-n+3)
4. 对图片定位,鼠标靠近就显示加边框

## 3.4 11-移动web-微金所头部块制作(下)

1. **如何修改bootstarp预定制样式？**

**找到具体的标签样式元素 在类名前面加上当前块类名 在重新更改它的属性**

# 4 bootstarp导航组件说明及wjs导航的完成

## 4.1 12-移动web-导航条组件的说明

1. 组件 > 导航条 可以找到我们类似的导航栏
2. 结构的选择( 选择需要的内容)

## 4.2 13-移动web-导航条组件的样式修改(上)



**1、导航条样式如何修改？**

* 1. 找到源文件中的样式 navbar-default，复制到当前页面的css文件中(wjs\_nav.css) 但这样会加大代码量
  2. 把原有的navbar-default用wjs\_nav替换，再重新定义新的属性，也就是重置覆盖框架所提供的css样式

## 4.3 14-移动web-导航条组件的样式修改(下)

**修改步骤：**

1. 鼠标靠近给hover,focus加底边框，
2. 给有active类名的加底边框，背景透明
3. Logo更换颜色和大小
4. 移动端就隐藏我要投资等导航栏
5. 更改移动端菜单样式

# 5 轮播图效果的实现

## 5.1 15-移动web-轮播图的效果分析

**1、移动端和pc端的机构**

w<768px-移动端：图片会随着屏幕的缩小自动适应--缩小

img的宽度为100%,通过img标签来实现

w>=768px：

图片做为背景，当屏幕宽度变宽的时候，图片的高度不在改变

1.background-image添加图片

2.添加background-position:center center

3.background-size:cover

## 5.2 16-移动web-轮播图插件的使用说明

**1、data-target和href的两种控制**

data-target="#carousel-example-generic"：当前这个li元素控制的是id为carousel-example-generic这个容器 data-target直接找到目标id容器

href="#carousel-example-generic"：设置控制的是id为carousel-example-generic这个容器 href是锚链接到id上，

## 5.3 17-移动web-移动端的轮播图效果实现

1、移动端图片是img标签图片，需要设置宽度100%，图片可以点击需要包含在a标签中

## 5.4 18-移动web-非移动端的轮播图效果实现

1. 高度不在改变，图片设置成背景图片。
2. 图片也是可以点击跳转链接，需要包含在a标签中
3. 图片居中显示:background-position:center center
4. 图片内容全部显示 banckground-size:cover

## 5.5 19-移动web-使用响应式工具实现轮播图

1. **响应式工具如何使用？**
   1. 非移动端就显示非移动端图片即背景图片

移动端就显示移动端图片即img标签图片

* 1. 背景图在移动端隐藏 hidden-xs, img标签图片在非移动端隐藏 hidden-sm/hidden-md/hidden-lg

1. **利用响应式工具实现会存在哪些问题？**
   1. 资源浪费 (在移动端也会加载非移动端图片，在非移动端也会加载移动端图片)
   2. 加载缓慢 (每次都会把所有的图片都加载一次)，用户体验不好

## 5.6 20-移动web-使用js实现轮播图

1. **如何获取data属性 (jquery的新方法)**

data-small-image=” ” => data(“smallImage”)

data-large-image=” ” => data(“largeImage”)

1. **js如何根据当前屏幕大小来获取对应图片**
   1. 页面结构item写成自定义data属性，其值分别对应移动端和非移动端的图片路径
   2. resize来获取屏幕尺寸的实时切换
   3. 判断当前是移动端就显示移动端图片，非移动端就显示非移动端图片
   4. 遍历再获取item自定义的data属性 ，添加子元素中，同时把data属性传给背景图还是标签图

$(**function**(){  
 **var** items=$(".carousel-inner .item");  
 $(window).on("resize",**function**(){  
 **var** width=$(window).width();  
 **if**(width>=768){/\*说明非移动端\*/  
 /\*为每一个item添加子元素--遍历\*/  
 $(items).each(**function**(index,value){  
 **var** item=$(**this**);/\*当前自定义属性中 存储的图片路径\*/  
 **var** imgSrc=item.data("largeImage");  
 console.log(imgSrc);  
 /\*添加非移动端的子元素\*/  
 item.html($('<a href="javascript:;" class="pcImg"></a>').css("backgroundImage","url('"+imgSrc+"')"));  
 });  
 }  
 **else**{  
 $(items).each(**function**(index,value){  
 **var** item=$(**this**);  
 **var** imgSrc=item.data("smallImage");  
 item.html('<a href="javascript:;" class="mobileImg"><img src="'+imgSrc+'" alt="..."></a>');  
 });  
 }  
 }).trigger("resize");

## 5.7 21-移动web-实现轮播图的滑动操作

**1、如何实现滑动操作？**

1. 移动端滑动事件：touch
2. 获取touchStart ：startX和touchEnd ：endX的坐标
3. 判断endX-startX的差值，

如果>0 表示向右滑，就上一张 carousel.carousel('prev');

如果<0,表示向左滑，就下一张 carousel.carousel('next');

## 5.8 22-移动web-微金所首页轮播图的完成

轮播图图片顶部距离的清除

