day18

boot最重要的3个知识点。媒体查询+栅格布局+scss

五.组件和js插件

13.焦点轮播图

|  |  |  |
| --- | --- | --- |
| 结构  div. carousel 相对定位  <!-- 1.轮播的图片 -->  >div. carousel-inner 相对定位，w-100,溢出隐藏  >div. carousel-item display:none;  . active 显示  >img.w-100  <!-- 2.轮播指示器 -->  >ul. carousel-indicators 绝对定位，偏移量，弹性，x轴，居中对齐  >li.active   |  | | --- | | 需要对li重写样式  .carousel-indicators li{  width: 0.8rem;height: 0.8rem;  border-radius: 50%;  background-color: #fff;  margin-left:0.3rem;  margin-right: 0.3rem;  }  .carousel-indicators .active {  background-color: #0aa1ed;  } |   <!-- 3.左右箭头 -->  <a href="" class="carousel-control-next">  <span class="carousel-control-next-icon"></span>  </a>  <a href="" class="carousel-control-prev">  <span class="carousel-control-prev-icon"></span>  </a>   |  | | --- | | 样式的重写  .carousel-control-prev,.carousel-control-next{  width: 4%;height: 20%;  top:40%;  border-radius: 0.25rem;  background-color: #aaa;  }  .carousel-control-prev{left:5px}  .carousel-control-next{right:5px} |   事件  <!-- 1.轮播的图片 -->  写给最外层div. carousel data-ride="carousel"  <!-- 2.轮播指示器 -->  <li data-target="#parent" data-slide-to="1"></li>  data-target中的id，是父元素div.carousel的id  <!-- 3.左右箭头 -->  a data-slide="prev/next" href="#parent" |

14.分页

|  |
| --- |
| ul.pagination  >li.page-item 第一个和最后一个li中的a添加对应圆角  .active 选中  .disabled 禁用  >a.page-link 块级，内边距，边框，文本设置，相关的伪类样式 |

15. 面包屑导航

|  |
| --- |
| ul. breadcrumb 弹性，x轴，可换行，背景色，圆角  >li.breadcrumb-item |

16. 进度条

|  |
| --- |
| div. progress 进度条的凸(艹皿艹 )  >div. progress-bar 进度条  .progress-bar-striped 带条纹的进度条  .progress-bar-animated 带动画的进度条 |

SCSS

一.动态样式语言

|  |
| --- |
| Boot3使用less生成的  Boot4使用scss生成的 |

1.css有几个缺点

|  |
| --- |
| 语法不够强大，没有变量等合理的样式复用机制，导致难以维护  我们需要动态的样式语言，为css提供新的特性  常见的动态样式语言  1.scss/sass （scss兼容sass）  2.stylus  3.Less |

2.scss是什么

|  |
| --- |
| scss是一款强化css的辅助语言  语法和css十分相似  添加了变量，嵌套，混合，导入，函数等高级功能  这些拓展功能让css更加的强大和优雅 |

二.使用scss

1.安装scss

|  |
| --- |
| 参考大礼包中的word文档  windows 安装 choco install sass  npm install -g sass |

2.scss的使用

|  |
| --- |
| scss运行在服务器，把.scss文件转换成.css文件，再把css发送给前台  ①单文件转换----一次转换一个scss文件  node-sass scss文件路径和文件名 css文件路径和文件名  node-sass scss/01.scss css/01.css  ②多文件转换----一次转换一个文件夹的scss文件  node-sass scss文件夹名称 -o css文件夹名称  node-sass scss -o css  ③单文件监听  node-sass -w scss文件路径和文件名 css文件路径和文件名  ④多文件监听  node-sass -w scss文件夹名称 -o css文件夹名称  node-sass -w scss -o css |

三.scss基础语法

1.变量

|  |
| --- |
| 声明变量  $变量名称: 值；  使用变量 border:$变量名称  变量名的要求，与类选择器规则类似  可以由字母，数字，- ,\_ 不能以数字开头，不能用关键字  变量中可以保存的值：  1.数字  2.数值+单位  3.颜色  4.样式属性  5.其它变量  变量有作用域！！！ |

2.嵌套

|  |
| --- |
| 在选择器中，直接写其它选择器，生成后代关系  #content{  p{color:red}  }  #content p{color:red;} |
| #content,p,.c1{  a{color:#111;}  }  #content a, p a, .c1 a {color: #111; } |
| a{  color:red;  &:hover{color:#111;}  }  伪类的嵌套，需要添加&当做占位符，不然会生成空格 |
| div{  border:{width:1px;color:#111;style:solid}  }  div {  border-width: 1px;  border-color: #111;  border-style: solid; } |
| 练习：  $mybule:#0aa1ed;  $mywidth:130px;  $fontcolor:#fff;  $myheight:40px;  $my\_color:red；  使用a标签，完成样式  当hover的时候，字体变为红色（嵌套） |

3.导入

|  |
| --- |
| @import "其它scss文件"; 加不加文件后缀都正确  使用下划线开头的scss文件，称为局部scss文件  不使用下划线开头的scss文件，称为全局scss文件  在做scss---->css的时候，局部文件不会转换成对应的css文件  导入的时候，不写下划线，不写后缀  \_123.scss  @import "123";  被导入文件的变量，在导入文件中可以直接使用 |

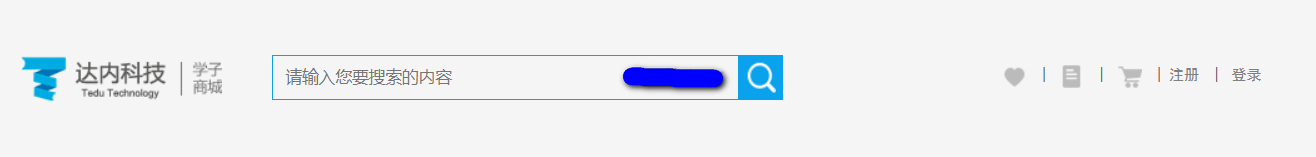
4.混合器

|  |
| --- |
| 把一组样式封装进一个混合器，之后哪里使用，就在哪里调用混合器  定义混合器  @mixin 混合器名称{ 一堆样式 }  调用混合器  选择器{  @include混合器名称；  } |

作业：

1.完成带参数的混合器练习，编写线性渐变

2.尝试使用boot完成



2.尝试使用boot完成

