[一.复杂选择器 2](#_Toc11427364)

[2.属性选择器 2](#_Toc11427365)

[3.伪类选择器 3](#_Toc11427366)

[4.伪元素选择器 4](#_Toc11427367)

[二.弹性布局（重中之重\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*） 6](#_Toc11427368)

[1.弹性布局是什么 6](#_Toc11427369)

[2.弹性布局的相关概念 6](#_Toc11427370)

[3.语法 7](#_Toc11427371)

[4.容器属性 7](#_Toc11427372)

[5.项目属性，只能设置在某一个项目上，不影响其他项目效果 9](#_Toc11427373)

day13

# 一.复杂选择器

## 2.属性选择器

|  |
| --- |
| id class name type title value style width height alt src href  允许通过元素所附带的属性及其值来匹配页面元素，很精准  属性选择器，在css中使用较少。  但是在后续的js中起到了很大的作用：  1.使用自定义属性精准的找到元素对象  2.根据自定义属性值，给这个对象绑定不同的事件方法  语法  1.[attr]{};attr代表任意属性  允许通过元素所附带的属性，来匹配元素  [class]{ };匹配所有具有class属性的元素  2.elem[attr]{};匹配elem并且具有attr属性的元素  div[id]{};匹配具有id属性的div元素  3.elem[attr1][attr2]{};匹配同时具有attr1和attr2属性的elem元素  span[id][title]{}; 匹配同时具有id属性和title属性的span元素  4.[attr=value]{};匹配具有attr属性，并且值为value  [id=”c1”]{};匹配id属性为c1的元素  5.匹配模糊属性值  [attr^=value]{};匹配有 attr属性，并且以value开头的元素  [attr$=value]{};匹配有 attr属性，并且以value结尾的元素  [attr\*=value]{};匹配有 attr属性，并且值中含有value的元素  [attr~=value]{};匹配有 attr属性，并且值中含有value独立单词的元素 |

## 3.伪类选择器

|  |
| --- |
| 已经学习过的伪类  :link :visited :hover :active :focus |

①目标伪类

|  |
| --- |
| 匹配被激活的锚点  :target{} |

②结构伪类

|  |
| --- |
| 1.selector:first-chi{} 匹配的元素是属于selector父元素的第一个子元素，而且要求第一个子元素也符合selector  2.selector:last-child匹配的元素是属于selector父元素的最后一个子元素，而且要求最后一个子元素也符合 |

③空元素empty

|  |
| --- |
| selector:empty{}  何为空?内部不能有任何东西，包括文本，元素，空格，回车。 |

④:only-child

|  |
| --- |
| 匹配属于其父元素的唯一子元素，匹配的是子元素  a:only-child{color:#f00;font-size:40px;} |

⑤否定伪类

|  |
| --- |
| :not(selector){}把符合selector的元素刨除在外  a:not(:first-child){color:#0f0;} |

## 4.伪元素选择器

①伪元素文本修饰

|  |
| --- |
| 1.:first-letter或者 ::first-letter  匹配元素的首字符  2.:first-line 或者 ::first-line  匹配元素首行文字  注意，首字符和首行冲突，以首字符为准  3.::selection必须是两个冒号  匹配用户选定的内容文本  注意：在selection中，只能修改文本颜色和背景颜色，其它无效  Ex: p:first-letter{color:#f0f;font-size:50px;}  p:first-line{color:#0f0;}  p::selection{color:#ff0;background:#f00;} |

②伪元素选择器，内容生成

|  |
| --- |
| 使用css添加html元素，这个被css添加的元素，称之为伪元素  elem:before 或者 elem::before  在elem的内容区域的最前面（在内容区域中），添加一个假的元素  这个元素，使用content添加内容（文本或者url（图片））  使用display修饰这个元素的显示方式  可以添加所有样式  elem:after 或者 elem::after  在elem的内容区域的最前面（在内容区域中），添加一个假的元素 |
| 使用伪元素的内容添加  1.解决外边距溢出  #parent:before{  content:"";  display:table;  }  在元素的内容区域的最前面，添加一个空table |
| 2.解决高度坍塌  父元素不写高，内部子元素全部浮动  解决：父元素追加一个空的块级元素，此元素清除浮动  #content:after{  content:"";  display:block;  clear:both;  } |

# 二.弹性布局（重中之重\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*）

## 1.弹性布局是什么

|  |
| --- |
| 弹性布局，就是一种布局方式  主要解决某个元素中子元素的布局  为布局提供了很大的灵活性，在bootstrp应用广泛 |

## 2.弹性布局的相关概念

|  |
| --- |
| 1.容器  要发生弹性布局的子元素们的父元素称为容器  在容器中设置display:flex;让子元素们弹性布局  2.项目  要做弹性布局的子元素们，称之为项目  父元素设置了display:flex;子元素们都被称之为弹性项目  3.主轴  项目们排列方向是一根轴  主轴如果是x轴，项目们水平排列  主轴如果是y轴，项目们垂直排列  项目们在主轴上排列顺序，主轴的起点和主轴的终点  4.交叉轴  与主轴方向垂直的方向的一根轴  项目们在交叉轴上的对齐方式，是交叉轴的起点和终点 |

## 3.语法

|  |
| --- |
| 将元素设置为弹性容器之后，他所有的子元素都变为弹性项目  这些弹性项目都可以按照弹性布局的方式排列  默认主轴为x轴，主轴起点在左边，主轴终点在右边  属性display  取值1.flex 将块级元素设置为容器  2.inline-flex 将行内元素设置为容器  注意：  1.元素设置为弹性容器之后，  项目的float/clear失效  容器的vertical-align/text-align失效  2.项目可以设置宽高  3.父元素设置定宽，就没有项目根据尺寸变小的效果了 |

## 4.容器属性

①主轴的方向

|  |
| --- |
| flex-direction:  取值：1.row 默认值，主轴是x轴，主轴起点左端  2.row-reverse 主轴是x轴，主轴起点是右端  3.column 主轴是y轴，主轴起点是顶端  3.column-reverse主轴是y轴，主轴起点是底端  注意：由于交叉轴是与主轴垂直的，所以主轴变化之后，交叉轴也会变 |

②设置项目换行

|  |
| --- |
| flex-wrap:  取值：1.nowrap默认值，空间不够，也不换行，项目自动缩小  2.wrap 空间不够就换行，项目不缩小  3.wrap-reverse;空间不够，换行并反转 |

③flex-direction和 flex-wrap的缩写

|  |
| --- |
| flex-flow:direction wrap; |

④定义项目在主轴上的对齐方式

|  |
| --- |
| justify-content:  取值：1.flex-start默认值，主轴起点对齐  2.flex-end主轴终点对齐  3.center主轴居中对齐  4. space-between两端对齐，两端无空白  5. space-around两端对齐，两端有空白，每个空白距离相等  注意：改变主轴方向，也会改变项目的对齐方式 |

⑤项目在交叉轴的对齐方式

|  |
| --- |
| align-items:  取值：1.flex-start 交叉轴起点对齐  2.flex-end 交叉轴终点对齐  3.center 交叉轴居中对齐  4.baseline 交叉轴基线对齐，就是交叉轴起点  5.stretch 前提，项目不写高，占满交叉轴上所有空间 |

## 5.项目属性，只能设置在某一个项目上，不影响其他项目效果

①项目顺序

|  |
| --- |
| order: 取值：无单位整数  定义项目的排列顺序，值越小越靠近主轴起点，默认值为0 |

②flex-grow

|  |
| --- |
| 定义项目的放大比例  如果容器有足够大的剩余空间，项目将按比例放大  取值：无单位整数，默认值0不放大 |

③flex-shrink

|  |
| --- |
| 定义项目的缩小比例  容器空间不足时，项目该如何缩小  取值：无单位整数，默认值1  如果取值为0，表示不缩小。取值越大，缩的越小。 |

④align-self (align-items：设置所有项目在交叉轴的对齐方式)

|  |
| --- |
| 设置自己在交叉轴的对齐方式  取值：1.flex-start 交叉轴起点对齐  2.flex-end 交叉轴终点对齐  3.center 交叉轴居中对齐  4.baseline 交叉轴基线对齐，就是交叉轴起点  5.stretch 前提，项目不写高，占满交叉轴上所有空间  6.auto 自动，该项目使用容器的align-items的值 |

作业：

1.完成学子商城1L