day11

八.文本相关的样式

1.字体属性

2.文本属性

①文本的颜色

|  |
| --- |
| color:合法的颜色值 |

②水平对齐方式

|  |
| --- |
| 设置内部的文本，行内元素，行内块的水平对齐方式  text-align:  left 默认缺省值  center  right  justify两端对齐 |
| text-align:center和margin:0 auto的区别  margin:0 auto让块级元素水平居中  text-align:center设置内部的文本，行内元素，行内块的水平居中 |

③行高

|  |
| --- |
| 如果设置的行高大于字号大小，文本会在行高的范围内垂直居中显示  line-height:  取值，可以取px为单位的数字，一般定义为容器的高度  无单位的数字，行高为字号的倍数  注意，对于多行文本，行高的垂直居中不适用 |

④文本线条

|  |
| --- |
| text-decoration:  overline 上划线  underline 下划线  line-through 删除线  none 去掉线条 |

⑤首行缩进

|  |
| --- |
| text-indent: 取值px为单位的数字 |

⑥文本的阴影

|  |
| --- |
| text-shadow: h-shadow v-shadow blur color;  h-shadow 水平偏移  v-shadow 垂直偏移  blur 模糊程度  color 颜色 |

练习：01\_ex.html

|  |
| --- |
| 定义div，内容随意  1.设置字体系列  2.设置文本大小  3.加粗并斜体  4.小写字母转成小型大写字母  5.两端对齐  6.有下划线  7.首行缩进60px  8.文本垂直居中  9.有文本阴影 |

九.表格的样式属性

1.表格的常用样式

|  |
| --- |
| table，之前学习的样式基本都可以使用，尺寸，边框，背景，内外边距，文本  但是边框只有最外层的边框，内部没有边框  td/th, 尺寸，边框，背景，内边距，文本有效  外边距无效  vertical-align 设置文本的垂直对齐方式  top/middle(默认)/bottom |
| table是一种特殊的元素表现形式  实际尺寸是根据内容的多少而决定  内容多，尺寸小，以内容为准  内容小，尺寸大，以尺寸为准  某一行，所有列的高度一致  不同行的，同一列宽度一致  导致table的默认渲染方式，需要先全部加载到内存中，之后再一次性渲染到页面，渲染效率低 |

2.表格的特有属性

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1.边框的合并  border-collapse:  separate 分离状态  collapse 合并状态  2.由于td的外边距无效，边框边距  要保证边框是分离状态border-collapse: separate;  border-spacing: 20px;  3.标题的位置  caption-side: top/bottom;  4.设置表格的显示规则   |  |  | | --- | --- | | table自动布局,默认布局方式 | table的固定布局 | | 单元格大小会自动适应内容 | 单元格大小不会再自适应内容 | | 表格复杂的时候，加载速度很慢 | 任何情况下，表格都会快速加载 | | 比较灵活 | 不够灵活 | | 适用于不确定每列大小，并且不复杂的表格 | 适用于，确定每列大小的表格 | | table-layout:auto | table-layout:fixed | |

十.定位

改变元素在页面中的位置

1.定位的分类

①普通文档流

②浮动定位

③相对定位

④绝对定位

⑤固定定位

3.普通流定位

|  |
| --- |
| 默认文档流定位-----------所有元素页面中都占据空间----元素不能发生堆叠  行内元素，行内块共用一行，一行放不下自动换行（从左往右）  块级元素单独成行（从上往下） |

4.浮动定位(重点\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*)

|  |
| --- |
| 让块级元素横向显示  float：  none 默认值，不浮动  left 让元素浮动之后，停靠在父元素的左侧或者其他左浮动元素的后面  right 让元素浮动之后，停靠在父元素的右侧或者其他右浮动元素的后面 |
| 浮动的特点  1.元素一旦浮动，将脱离文档流  2.浮动元素最终，会停靠在父元素的最左/右，或者其他已浮动的元素的后面  3.当父元素横向显示不下所有浮动元素时，显示不下的浮动元素会自动换行  4.浮动元素就是为了解决块级元素横向显示的问题 |
| 元素一旦脱离文档流，会发生4件事  1.页面不占据空间  2.后续元素上前补位  3.元素如果没有宽度，一旦脱离文档流，元素的宽度靠内容撑开  4. 元素一旦脱离文档流，将变为块级元素(可以设置尺寸，上下外边距有效) |

5.浮动的特殊情况

①浮动元素如果没有设置宽度，浮动之后，宽度靠内容撑开

②元素一旦浮动，将变为块级元素(可以设置尺寸，上下外边距有效)

③浮动元素占位问题

|  |
| --- |
| 浮动元素会向着浮动方向，占位，这个占据位置不会让被挤下去的其它浮动元素占据 |

④文本，行内元素，行内块不会被已浮动元素压在下面，而是避开浮动元素显示

6.清除浮动元素带来的影响

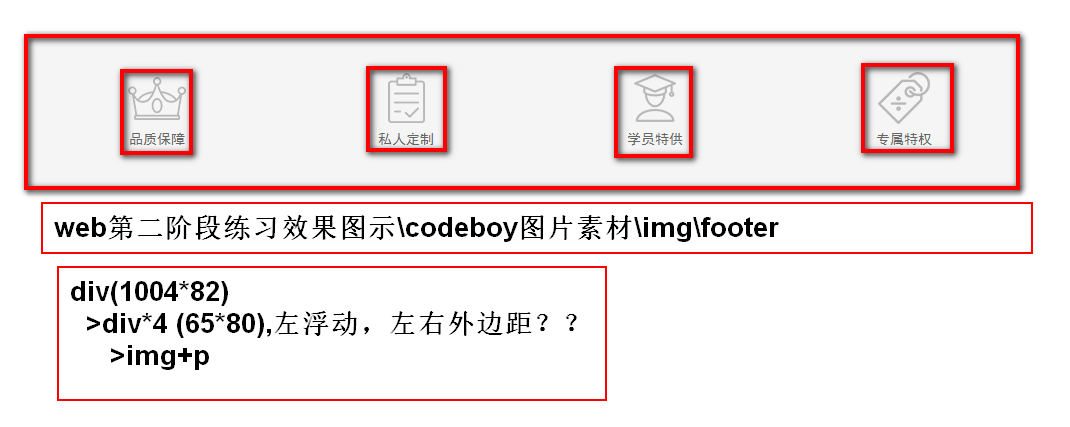
|  |
| --- |
| 浮动元素带来的影响：  一个块级元素之前的元素如果浮动，  这个块级元素会钻到浮动元素的下面去补位  设置 clear:left;  right;  both; |

7.高度坍塌

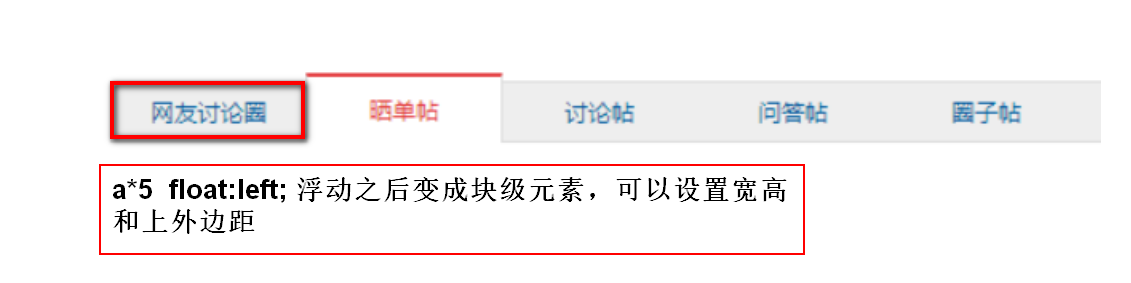
|  |
| --- |
| 高度坍塌发生的原因：  父元素没有定义高度，内部所有元素浮动，父元素在文档流里就认为内部没有元素，找不到高度  解决方案：  1.给父元素定义高度，在子元素数量比较少的情况下，可以使用  子元素的数量不确定,适用  2.父元素也浮动，会影响到父元素的兄弟元素和爷爷元素  3.给父元素overflow:hidden/auto; 但是父元素如果想溢出显示，就不行了  4.在父元素内部追加一个块级元素，不设置高，不设置宽，添加clear:both; |

作业：

1.浮动完成学子品质保障



2.使用浮动完成



3.使用浮动完成学子首页1L上半部分



4.特高题：使用浮动完成学子商城底部

