1.定位-浮动定位

1.定位

定位，指的是改变元素在页面中默认的位置。

2.定位分类

按照定位的效果，可以分成以下几类：

1.普通流定位（默认显示方式）

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

又称“文档流定位”，页面中元素的默认定位方式。

1.每个元素在页面中都有自己的空间

2.每个元素默认都是从其父元素的左上角开始显示

3.页面中的块级元素都是按照从上到下逐个排列，每个元素独占一行

4.页面中的行内元素以及行内块都是按照从左到右的顺序来排列的

问题：如何实现多个块级元素在一行中显示？

4.浮动

1.什么是浮动&特点

如果将元素浮动起来以后，将具有以下特征：

1.元素一旦浮动起来将不占页面空间，脱离文档流定位， 其它元素将上前补位。

2.浮动元素会停靠在父元素的左边或右边，或其它已经浮动元素的边缘上。

2.浮动语法

属性：float

取值：

left:左浮动，让元素停靠在父元素的左边，或挨着左侧已有的浮动元素

right：右浮动，让元素停靠在父元素的右边，或挨着右侧已有的浮动元素

none:默认值，即无任何浮动效果

3.浮动引发的特殊效果

1.当父元素显示不下所有浮动元素时，最后一个元素将换行，但是，有可能被卡住。

2.元素一旦浮动起来后，宽度将以内容为主（未指定宽度情况下）

3.元素一旦浮动起来后，将变成块级元素

允许修改尺寸，能正常设置垂直外边距

4.文本，行内元素，行内块元素是采用环绕的方式排列的，是不会被浮动元素压在底下的，而是巧妙的避开浮动元素。

4.消除浮动

元素一旦浮动起来之后，就会对后续元素带来一定的影响（后续元素上前补位），如果后续元素不想被影响的话（不想占位），那么就可以通过清除浮动的方式来解决。

属性：clear

取值：

1.left 清除当前元素前面元素左浮动所带来的影响

2.right 清除当前元素前面元素右浮动所带来的影响

3.both 清除当前元素前面元素任何一种浮动所带来的影响

4.none 默认值，不做任何的清除

5.浮动元素对父元素高度带来的影响

元素的高度，都是以未浮动元素的高度为准，浮动元素是不占高度的。

解决父元素高度的问题：

1.直接给父元素设置高度

弊端：不是每次都知道父元素的高度

2.设置父元素也浮动。

弊端：不是任何时候父元素都需要浮动，而且浮动会影响后续元素。

3.为父元素设置overflow

取值：hidden或auto

弊端：如果有内容需要溢出显示的话，也会一同被隐藏。

4.在父元素中，追加一个空标记(在父元素最后一个子元素位置处，块级)，并设置其clear:both;