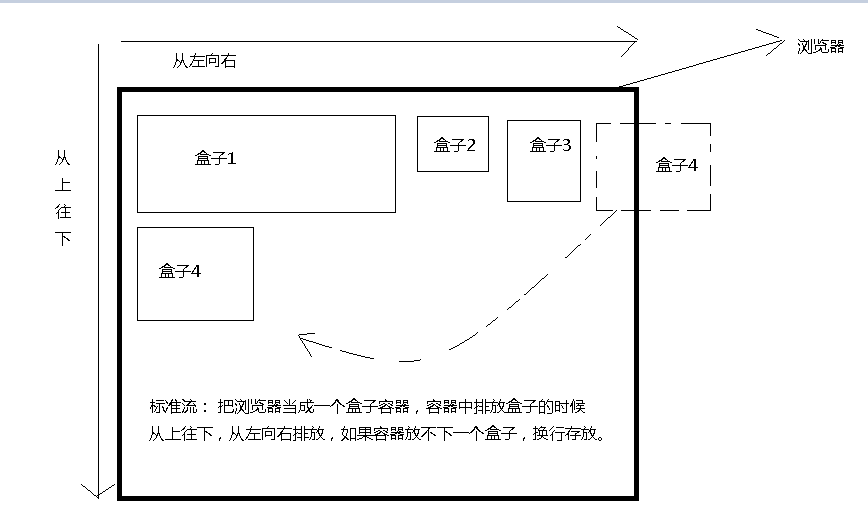
# 浮动

## 标准流

含义:浏览器默认摆放盒子的标准。



## 浮动写法:

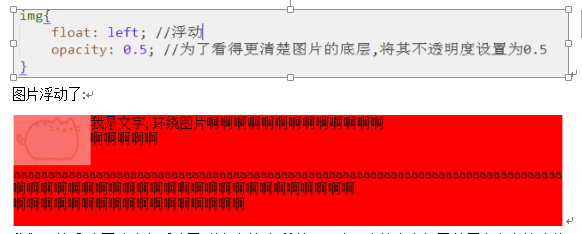


## 浮动:解决文字环绕图片的问题

1. 没浮动时



(2)



可以看到,图片确实浮动了,脱离文档流,所以下一个元素就占据了其原本在文档流的位置.

那为什么文字不会躲到图片下面呢?那是因为浮动的块虽然脱离了正常的文档流，但是依然占据正常文档流的文本空间。

打个比喻,就是: 小明拥有一套房子(标准流中的位置),小明过世了他的灵魂还在(浮动元素),他允许所有的东西进入他的房子(除文本之外的一切),就是不允许其他人(文本)住进来,因为他规定这个房子只能住一个灵魂(浮动元素).

## 浮动的特点:

1. 浮动以后的元素的显示方式与行内块级元素一样;

情景:如果元素A是浮动的，并且A元素的上一个元素也是浮动的，那么A元素会跟随在上一个元素的后边（两个元素会显示在同一行）。如果一行的宽度小于这两个元素的宽度和，那么后面的元素会被挤到下一行。

2. 浮动的元素会脱离标准流，即在标准流中不占位置;

3. 若子元素都浮动了,会存在父元素高度塌陷的问题.

## 浮动的案例：

1.0使用浮动完成页面的布局

2.0制作导航（注意：以后使用a标签直接来制作导航是有问题的）

(1)导航与导航之间应该是列表的关系，所以如果想要将这个些关系通过html语义化表示出来必须要用到ul标签。

(2)如果这些a标签你不用其它的标签包裹起来，那么将来浏览器会将这些a标签中的文字当作全部一起显示。SEO在查看页面的时候会认为这个页面进行作弊，就是进行了关键字的堆砌。必须要用ul中的li标签将a中的内容包裹起来。

(3)文字环绕图片（文本会给浮动以后的元素“让位”。）

## 清除浮动

1. 投机取巧法:

将<div style="clear:both;"></div>当作最后一个子标签放到父标签那.

此方法屡试不爽，兼容性强，使用方便，是初学时使用的上佳之选。

缺点:滥用标签,浪费资源,而且只能使用一次,所以这个方法不推荐。

2. overflow + zoom方法

给需要清除浮动的父元素添加声明: {overflow:hidden; zoom:1;}

缺点:会将超出父元素宽高的内容裁截掉(后面会讲)

1. 伪元素 + zoom方法(后面讲完伪类再讲)



这个样式清除浮动的，不会影响任何其他样式，通用性强，覆盖面广。

## 补充:伪元素

# overflow 溢出

|  |  |
| --- | --- |
| visible | 默认值。内容不会被修剪，会呈现在元素框之外。 |
| hidden | 超出元素的部分会被隐藏 |
| scroll | 给容器加上滚动条 |
| auto | 根据具体情况判断容器是否要加上滚动条 |

元素的两种隐藏方式：

overflow:hidden//将超出的部分裁剪掉

visibility: hidden//可以将元素隐藏，但是在页面上还保留着原来的位置。

display: none;//隐藏元素，并且在页面不占据位置。

# position 定位

## static: 静态的

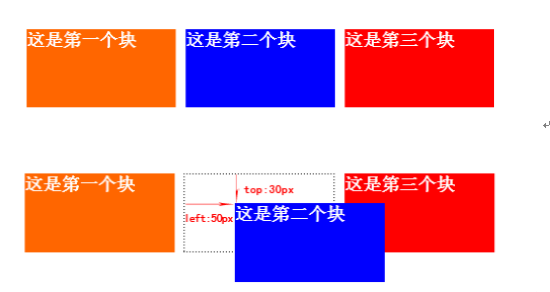
1.position:static; 静态定位。

2.所有的标准流（文档流）中的元素都是静态定位。

## relative:相对的

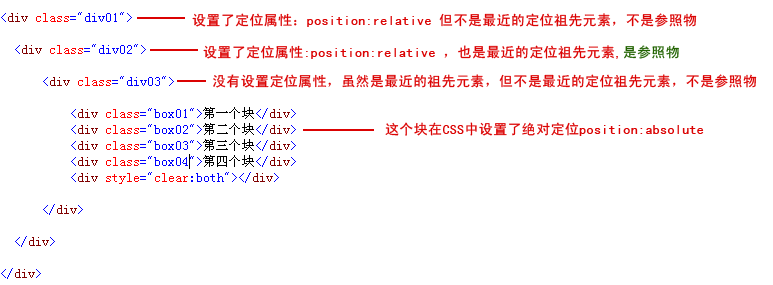
1. position:relative; 相对定位。
2. 使用的时候可配合：top,left,right,bottom来使用。
3. 如果设置了相对定位以及trbl属性，那将来盒子会以其原本的位置发生偏移.
4. 设置了相对定位的元素,即使发生了偏移,在页面上还占据着为变动前的位置.

总结：灵魂出窍。



## absolu**t**e:绝对的

1. position:absolute//绝对定位
2. 使用的时候可配合trbl属性来使用
3. 特点：
4. 如果绝对定位元素没有父元素或者是所有的父元素都没有定位，那么将来trbl是相对于body定位的；
5. 如果绝对定位的元素有父元素，而且父元素有定位（非static），那么这个绝对定位的元素偏移是以自己的父元素为基础。
6. 绝对定位以后的元素会脱离标准流,不占据原来的位置.



4.项目开发中常用:

将来在开发时,用的最多是绝对定位与相对定位一起使用：子元素使用绝对定位，父元素使用相对定位。[子绝父相]

案例:如何让父盒子中的子盒子在父盒子中居中显示?

1. 子绝父相;
2. 让子盒子的left设置为50%;
3. 让子盒子的margin-left为负值，数值为子盒子大小的一半

## fixed：固定

1. position:fixed。
2. 使用的时候也要配合trbl属性来使用
3. 特点
4. 不管页面有多大，trbl永远是相对于浏览器的可视区域来的。
5. 固定定位的元素也脱离了标准流（不在页面上占据位置）

## z-index 层级

1. 作用: 设置盒子的层级,层级越大的元素在最上面
2. 注意：

1. z-index可以取所有的整数（负整数，0，正整数）

2. z-index只能与定位一起使用。

# 比较标准流，浮动，绝对(固定)定位的层级关系：

标准流 < 浮动 < 绝对(固定)定位