

# 定位

- 定位允许你从正常的文档流中取出元素，并使他们具有不同的行为

## 四种定位值

### static

- 不指定position的值

### relative: 相对定位

- 占据正常的文档流中
- 主要用于创造定位上下文

### absolute: 绝对定位

- 不在正常的文档流布局中，坐在自己的一层
- 元素设置为absolute之后，默认会把元素内部转换为inline-block

### fixed: 固定定位

- 固定定位相当于浏览器视口本身

### sticky: 粘性定位

- 父级元素有overflow: scroll, auto, overlay, hidden时，粘性定位会失效。
- 粘性定位只在父元素内生效
- 父元素的高度不能低于sticky元素的高度
- 必须指定top, bottom, left, right其中之一。否则会处于相对定位(目前自己只知道top的效果, left的效果不知道是什么)

浮动的定位还是基于正常的文档流，只有改变了文档流的显示。(半脱离，元素会无视它，但是文本能看到)

## 包含块

- 元素的大小和位置通常受包含块的影响，通常包含块是最接近的块级元素的！！！内容！！！位置

#### 1. 元素的属性为position: fixed

- 则元素的包含块为可视窗口。

#### 1. 元素的属性是position: absolute

- 则元素的包含块为最接近的父级元素具position不是static。
- 元素的width用百分比时需要考虑包含块的padding。
- 如果都没有显示的定义position，那么元素的包含块是浏览器窗口(不是body是(html标签的父级))
- 根目录所在的包含块是一个称之为！！！初始包含块！！！的矩形

## 偏移量的设置

- 设置了position: absolute;之后，元素会被在内部转化为inline-block类型
- 这个和其他的位置定位不一样，设置的每一个关键字，代表的是元素的那一边的位置，如果设置了对角的关键字(left right)的话，那么就会固定住了子元素在父元素的相对位置。
- 设置left:0;right:0;top:0;bottom:0;之后，如果自身没有定义宽或者高，就会继承父元素(指的是包含块而不是DOM)的宽高。且内容不能偏移出父级元素的区域,否则会消失。

## z-index的介绍

- 只要是父元素显式的声明了z-index的话，无论子元素的z-index的值多小，子元素也不会到父元素的底下。？？？有疑问
- z-Index 只能应用于(神奇盒子) 默认值为0
- 神奇盒子最常见的定位不为static，opacity不为0,z-Index不为auto的flex，元素的transform不为none。
- 对于普通的盒子，(无特殊定义的，inline-block，float)，z-Index无法影响他们的层叠顺序写在html文档后面的块会挡在前面的盒子上面。  
float的盒子会挡在没有float的盒子上面，文字等inline，inline-block会在block的元素上面
- z-Index是在自己的父级内的层叠顺序，和伯伯元素的层叠顺序建立在父级和伯伯元素的层叠顺序上。
- 有些特定的元素也会在内部创建自己的堆叠上下文，内部的排序不会受到其他堆叠盒子的影响。
- 具有层叠的顺序的盒子和普通盒子的视轴顺序  
HTML > z-Index为负值的盒子 > 块状盒子 > float盒子 > inline,inline-block > z-Index为auto或者0的盒子 > z-Index为正值的盒子