**CSS定位**

1.浮动可以让多个块级盒子一行没有缝隙排列显示，经常用于横向排列盒子。

2.定位则是可以让盒子自由的在某个盒子内移动位置或者固定屏幕中某个位置，并且可以压住其他盒子。

**定位的组成：**

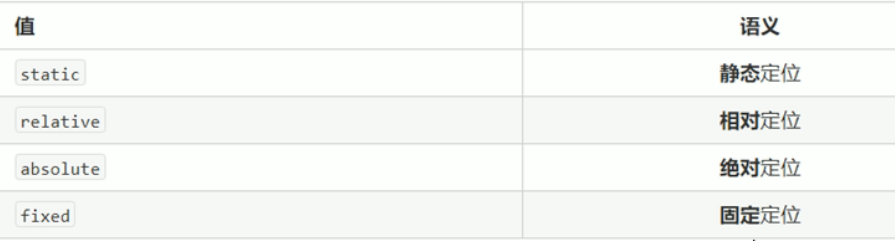
将盒子定位在某一个位置，所以定位也是在摆放盒子，按照定位的方式移动盒子。

定位=定位模式+边偏移

定位模式用于指定一个元素在文档中的定位方式，边偏移则决定了该元素的最终位置

**定位模式：**

定位模式决定元素的定位方式，它通过CSS的position属性来设置，其他可以分为四个



**静态定位：**

静态定位是元素的默认定位方式，无定位的意思

语法： 选择器{position : static ; }

* 静态定位按照标准流特性摆放位置，它没有边偏移
* 静态定位在布局时很少用到

**相对定位：**

相对定位是元素在移动位置的时候，是相对于它原来的位置来说的

语法： 选择器{position :relative ;

Top：100px；

Left：100px；}

* 它是相对于自己原来的位置来移动的（移动位置的时候参照点是自己原来的位置）
* 原来在标准流的位置继续占有，后面的盒子仍然以标准流的方式对待它（不脱标，继续保持原来位置）

**绝对定位：**

绝对定位是元素在移动位置的时候，是相对它祖先元素来说的

语法： 选择器{position : absolute ;

Top：100px；

Right：100px； }

* 如果没有祖先元素或者祖先元素没有定位，则以浏览器为准定位（document文档）
* 如果祖先元素有定位（相对、绝对、固定定位），则以最近一级的有定位祖先元素为参考点移动位置
* 绝对定位不再占有原先的位置（脱标）

**绝对定位的盒子居中**

加了绝对定位的盒子通过margin:0 auto水平居中，但是可以通过以下计算方法是实现水平和垂直居中

1. Left：50%; :让盒子的左侧移动到父级元素的水平中心位置
2. Margin-left:-100px; ：让盒子向左移动自身宽度的一半

**定位的特殊特性**

绝对定位和固定定位也和浮动类似

1. 行内元素添加绝对或者固定定位，可以直接设置高度和宽度
2. 块级元素添加绝对或者固定定位，如果不给宽度或者高度，默认大小是内容的大小

**脱标的盒子不会触发外边距塌陷**

浮动元素，绝对定位（固定定位）元素的都不会触发外边距合并的问题

**绝对定位（固定定位）会完全压住盒子**

浮动元素不同，只会压住它下面标准流的盒子，但是不会压住下面标准流盒子里面的文字（图片）

但是绝对定位（固定定位）会压住下面标准流所有的内容

浮动之所以不会压住文字，因为浮动产生的目的最初是为了做文字环绕效果的，文字会围绕浮动元素

**子决父相:**

子级是绝对定位的话，父级要用相对定位

1. 子级绝对定位，不会占有位置，可以放到父盒子里面的任何一个地方，不会影响其他的兄弟盒子
2. 父盒子需要加定位限制子盒子在父盒子内显示
3. 父盒子布局时，需要占有位置，因此父盒子只能是相对定位

**固定定位：**

固定定位是元素固定与于浏览器可视区的位置，主要使用场景，可以在浏览器页面滚动时元素的位置不会改变。

语法：选择器{position : fixed;

Top：100px；

Right：100px；}

* 以浏览器的可视窗口为参考点移动元素
* 跟父元素没有任何关系
* 不随滚轮的滚动而变化
* 固定定位不再占有原先位置
* 固定定位也是脱标的，其实固定定位也可以看做是一种特殊的绝对定位

**固定定位小技巧：固定在版心右侧位置**

小算法：

* 让固定定位的盒子left：50%，走到浏览器可视区(也可以看做版心)的一半位置
* 让固定定位的盒子margin-left版心宽度的一半距离，多走版心宽度的一般位置

**粘性定位sticky**

粘性定位可以被认为是相对定位和固定定位的混合

语法： 选择器{position : sticky; top： 10px； }

* 以浏览器的可视窗口为参考点移动元素（固定定点特点）
* 粘性定位占有原先的位置（相对定位的特点）
* 必须添加top、left、right、bottom其中一个才有效

**定位叠放次序z-index：**

在使用定位布局时，可能会出现盒子重叠的情况，此时，可以使用z-index来控制盒子的前后次序（z轴）

语法： {z-index：1；}

* 数值可以是正整数、负整数或0，默认auto，数值越大，盒子越靠上
* 如果属性值相同，则按照书写顺序，后来居上
* 数字后面不能加单位
* 只有定位的盒子才有z-index属性

**边偏移：**

边偏移就是定位的盒子移动到最终的2位置，有四个属性：



**元素的显示与隐藏**

类似网站广告，当我们点击关闭就不见了，但是我们重新刷新页面，就会重新出现 本质：让一个元素在页面中隐藏或者显示出来

1. Display显示隐藏

* Display:none ; 隐藏对象
* Display : block ; 除了转换为块级元素之外，同时还有显示元素的信息

Display隐藏元素后，不再占有原来位置

1. Visibility显示隐藏

* Visibility：visible；元素可视
* Visibility：hidden；元素隐藏

Visibility隐藏元素后，继续占有原有的位置

如果隐藏元素想要原来位置，就用visibility：hidden

如果隐藏元素不想要原来位置，就用display：none（用处更多）

1. Overflow溢出显示隐藏



**CSS高级技巧**

**精灵图**

CSS精灵技术：为了有效减少服务器接收和发送请求的次数，提高页面的加载速度

**使用精灵图核心：**

1. 精灵技术主要针对于背景图片使用，就是把多个小背景图片整合到一张大图片中。
2. 这个大图片也称为sprites精灵图或者雪碧图
3. 移动背景图片位置，此时可以使用background-position
4. 移动的距离就是这个目标图片的x和y坐标，注意网页中的坐标有所不同
5. 一般情况下都是往上往左移动，所以数值是负值

**字体图标（iconfont）**

字体图标使用场景：主要用于显示网页中通用、常用的一些小图标

字体图标可以为前端工程师提供一种方便高效的图标使用方式，展示的是图标，本质属于字体

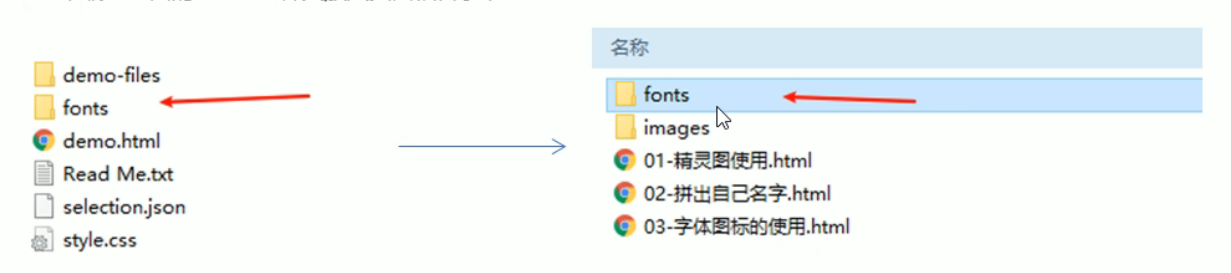
* 轻量型：一个图标字体要比一系列的图像要小，一旦字体加载了，图标就会马上渲染出来，减少了服务器请求
* 灵活性：本质其实是文字，可以很随意的改变颜色、产生阴影、透明效果、旋转等
* 兼容性：几乎支持所有的浏览器，请放心使用
* 字体图标 不能替代精灵技术，只能对工作中图标部分技术的提升和优化

**字体图标的下载：**



**字体图标的引入**

1. 把下载包里面的fonts文件夹放入页面根目录下



1. 在CSS样式中全局声明字体：简单理解把这些字体文件通过CSS引入到我们页面中，一定注意字体文件路径的问题



复制到CSS语句中去

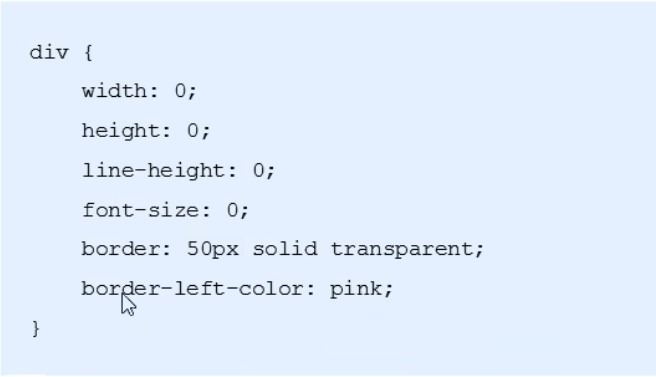
1. Html标签内添加小图标



也可以复制e91b，但是在粘贴在语句中后需要在e前添加转义符号\

**CSS三角制作**

代码如下：

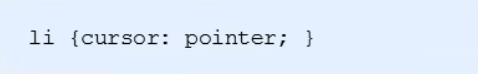


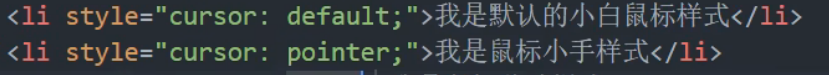
原理是将其余的三个三角形将其变成透明的

**CSS用户界面样式**

**鼠标样式cursor**

语法格式如下：



例如：

设置或检索在对象上移动的鼠标指针采用何种系统预定义的光标形状

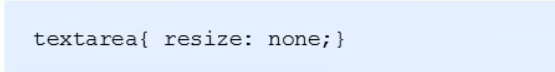


**轮廓线outline**

给表单添加outline:0;或者outline：none；样式之后，就可以去掉默认的蓝色边框

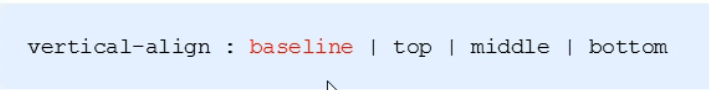
**防止拖拽文本域resize**

实际开发中，我们文本域右下角是不可以拖拽的 语法格式如下：

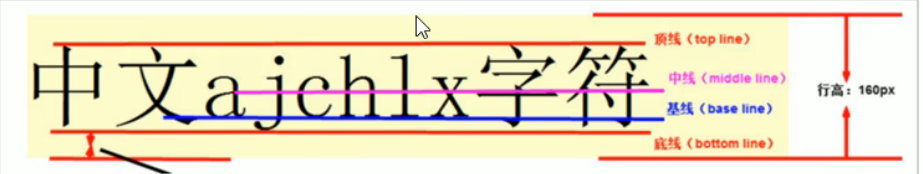


**Vertical-align属性应用**

CSS的vertical-align属性使用场景:经常用于设置图片或者表单（行内元素）和文字垂直对齐 语法格式如下：







* 图片、表单都属于行内块元素，默认的vertical-align是基线对齐

**解决图片底部默认空白缝隙问题**

Bug：图片底侧会有一个空白缝隙，原因是行内块元素会和文字的基线对齐

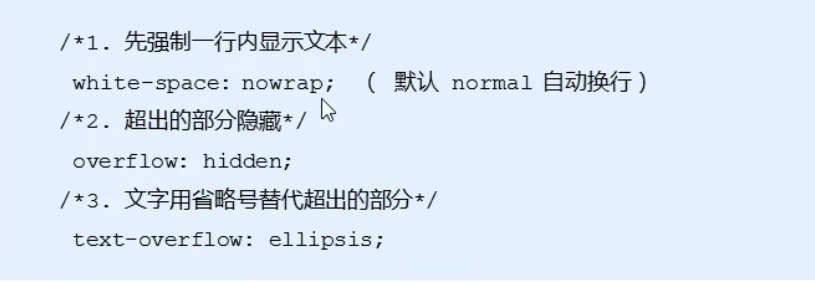
主要解决方法有两种：

1. 给图片添加vertical-align：middle|top|bottom
2. 把图片转换为块级元素display-block

**溢出的文字省略号显示**

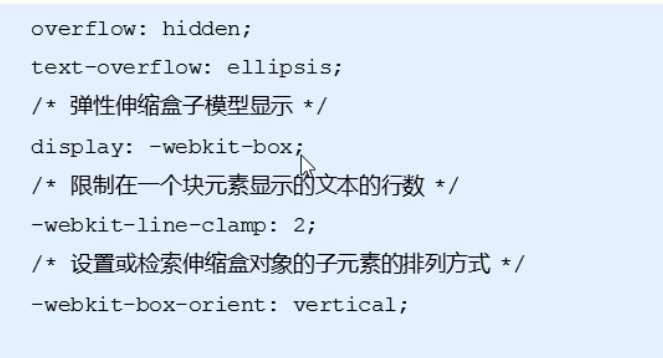
**单行文本溢出显示省略号**

需要满足三条件：



**多行文本溢出显示省略号**

多行文本溢出显示省略号，有较大兼容性问题，适合于webkit浏览器或移动端（移动端大部分是webkit内核）



**常见布局技巧**

**巧妙利用一个技术更快更好的布局：**

1. **Margin负值的运用**

、

让两个盒子可以无视两边相加不会出现加粗现象

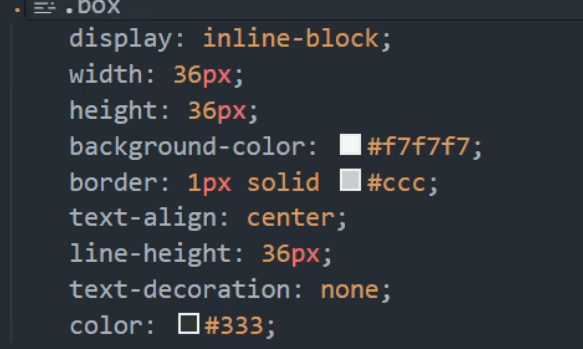
当需要让边框由鼠标经过时变色则需要提高当前盒子的层级即可（如果没有定位，则加相对定位（保留位置），如有定位，则加z-index）

1. **文字围绕浮动元素**

即通过在大盒子里面给照片，视频添加浮动然后字体就会自动形成一个围绕照片成型的样式。

1. **行内块的巧妙运用**

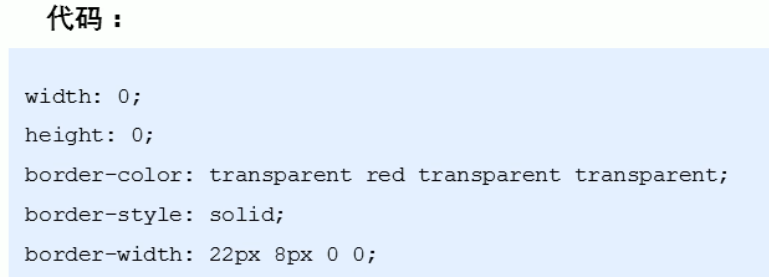




1. **CSS三角强化**



* 把左边和下边的边框宽度设置为0
* 把上边框矿都调大
* 把top属性改成透明色（transparent）



**CSS初始化**

由于不同游览器对有些标签的默认值是不同的，为了消除不同浏览器对HTML文本呈现的差异，照顾浏览器的兼容，我们需要对CSS初始化