* 使用一些专业技能和工具将UI设计稿实现成最终产品，包括pc端和移动端的网页，并处理视觉和交互的问题。简单来说，就是用户可以看到的和交互的部分，都是前端的工作领域
* 以前的网页：
* 现在的网页：
* 前端基本技能：
* HTML：是一种用于创建网页的标准标记语言。（结构层）
* CSS：层叠样式表（表现层）
* JS：脚本语言（行为层）
* HTML：是一种用于创建网页的标准标记语言。（结构层）



<!DOCTYPE html> 声明为 HTML 文档类型

<html> 元素是 HTML 页面的根元素

<head>文档的头部

<meta charset="utf-8"> 定义网页编码格式为 utf-8

<title>素描述了文档的标题

<body>包含页面可见的内容

<h1> 元素定义一个大标题

HTML标签通常是成对出现的，也有单独的，比如<img>，<input>，可以嵌套

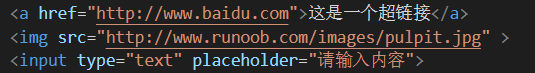


* 常见标签：<div>，<a>，<span>，<input>，<select>......





* 属性：在元素中添加一些附加的信息



* 块级元素，行内元素和行内块级元素：

块级元素会换行，宽度会占满整个浏览器；行内元素不换行，宽度由内容确定，不能设置高宽；行内块级元素也是一行显示，但是可以设置高宽

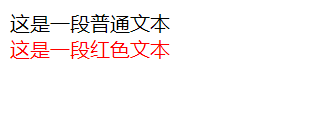
块级元素：<div>，<p>，<ul>，<table>...

行内元素：<span>，,<a>...

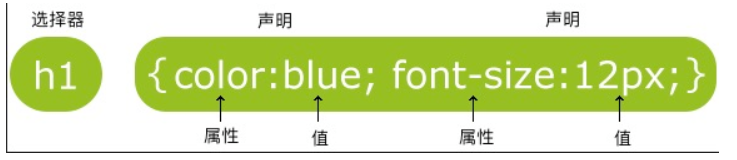
行内块级元素：<img>，<input>...

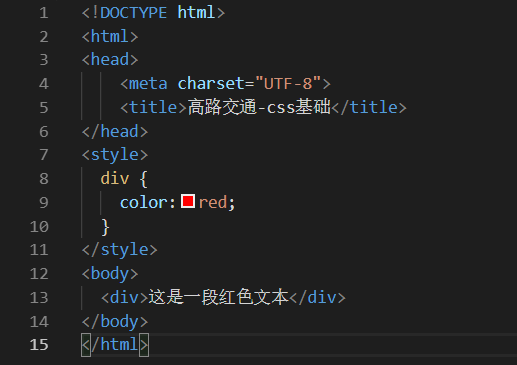
* Css：层叠样式表（表现层），定义了html里面元素的展示方式，大小，位置，颜色，间距等
* 引入css的三种方式：行内引入；内嵌式；外部引入
* 行内引入：



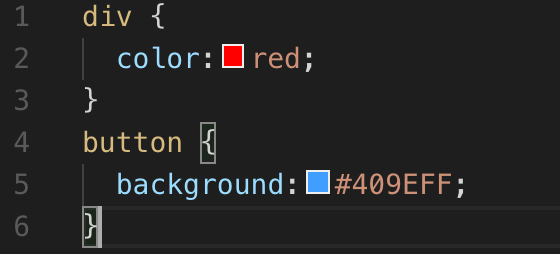


* 内嵌式：css规则有两个主要的部分组成：选择器，以及申明

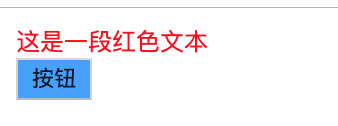




* 外部引入：



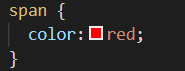




* Css选择器：

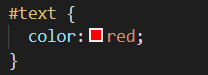
1. 元素选择器





1. id选择器（唯一性）



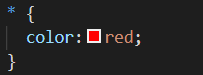


1. 类选择器

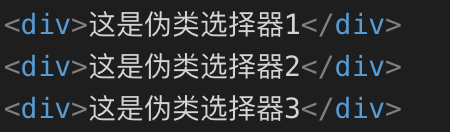


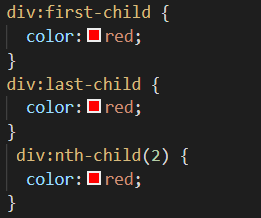


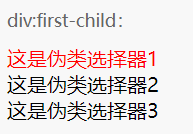
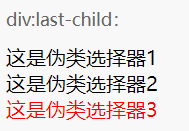
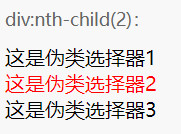
1. 全局选择器



1. 伪类选择器

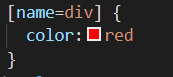




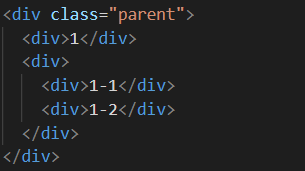
  

1. 属性选择器

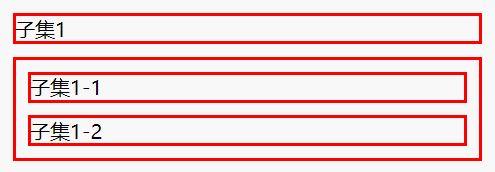




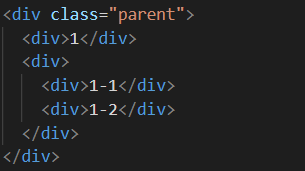
1. 包含选择器

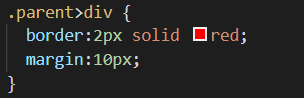


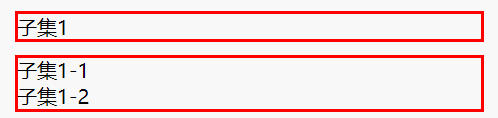




1. 子选择器



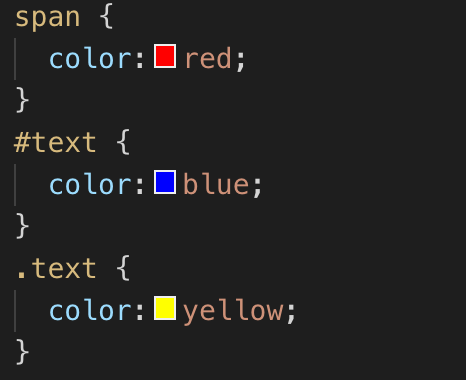




......

* 选择器优先级





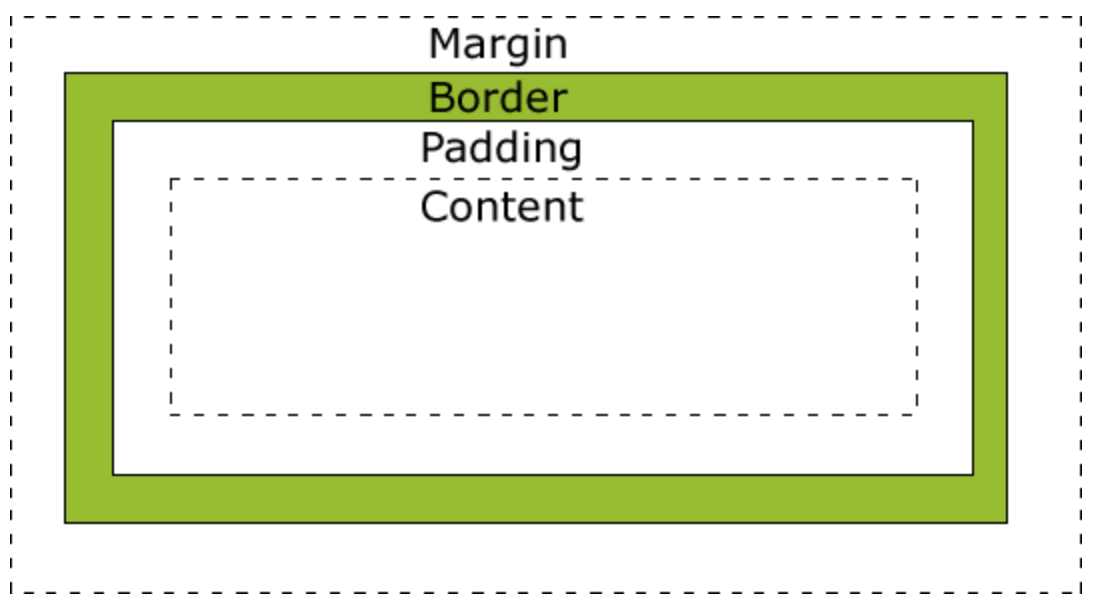






id选择器>类选择器>元素选择器

* Css盒模型：所有的HTML元素都可以看做是一个盒子

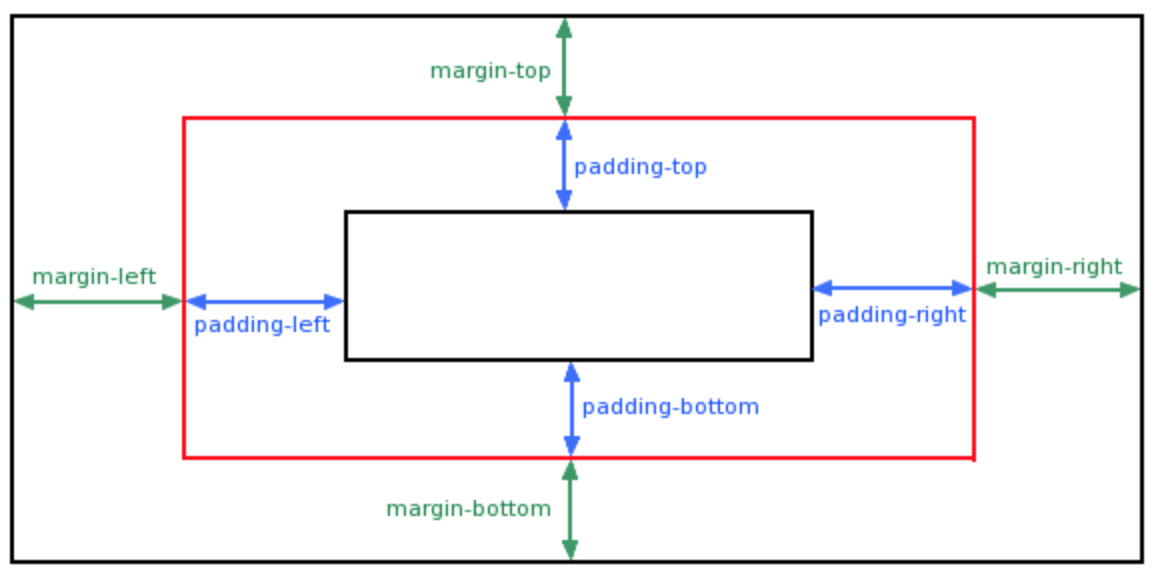


Margin：外边距

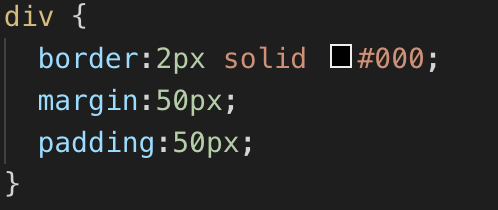
Border：边框

Padding：内边距

Content：内容

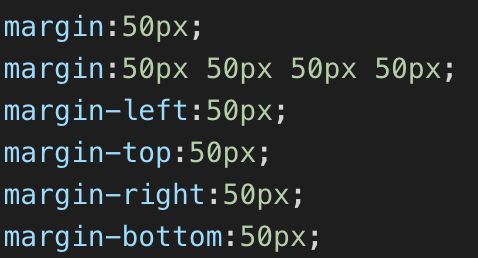






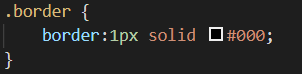


* 边距：

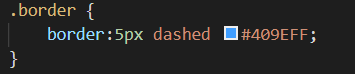


* 边框：



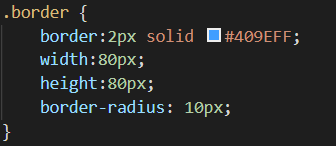


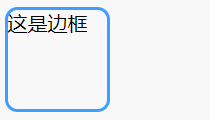




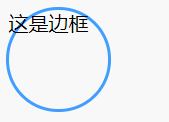


设置圆角：



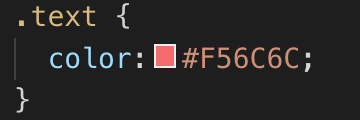






* 文本：



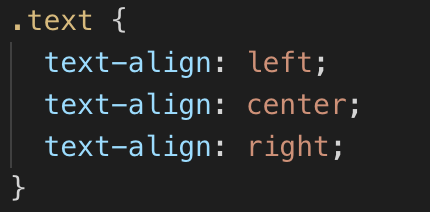


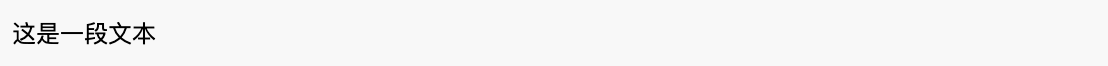


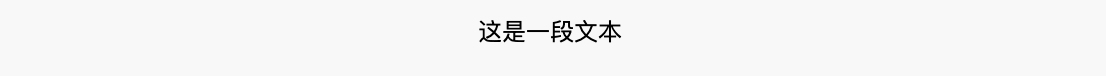
十六进制值 - 如: ＃FF6C6C

一个RGB值 - 如: RGB(255,0,0)

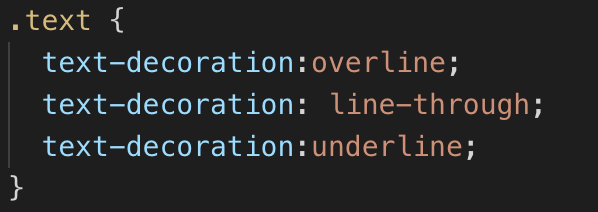
颜色的名称 - 如: red

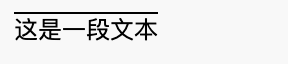






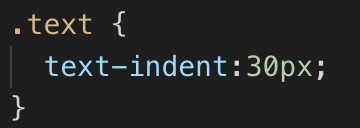














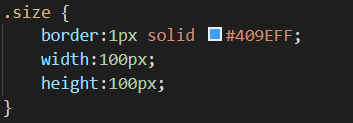


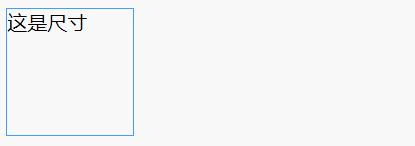




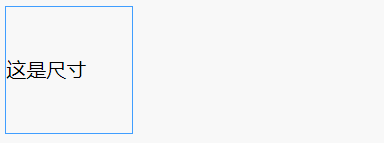
* 尺寸：











注：块级元素和行内块级元素才可以设置高宽，行内元素不能设置

解决方法：

设置成块级元素：

设置成行内块级元素：

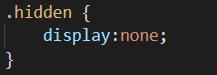
设置成行内元素：

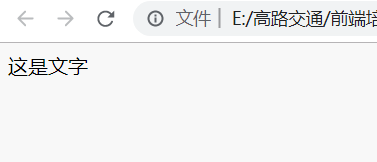
display:none；隐藏

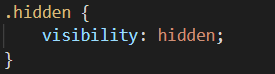
## visibility:hidden；隐藏

区别：visibility隐藏后的元素任然会占用空间，display:none则不会





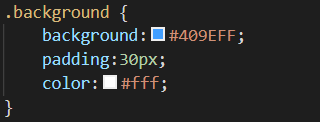


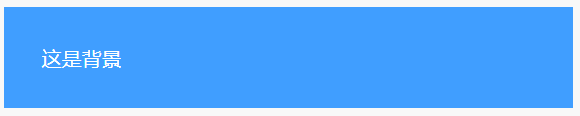


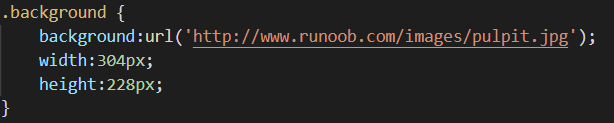


* 背景：

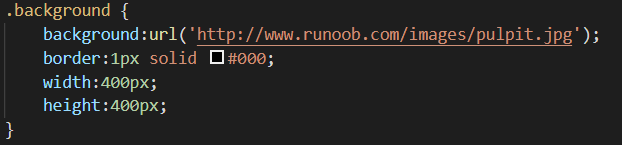


























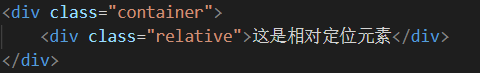
强制铺满，图片不成比例，会变形

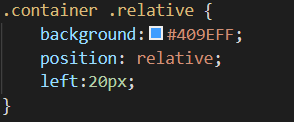
定位：position，元素的位置

常用的postion属性值：static，relative，absolute，fixed；通过left，top，right，bottom来定位

Static：默认值，没有定位，遵循正常的文档流

Relative：相对定位，相对正常的位置













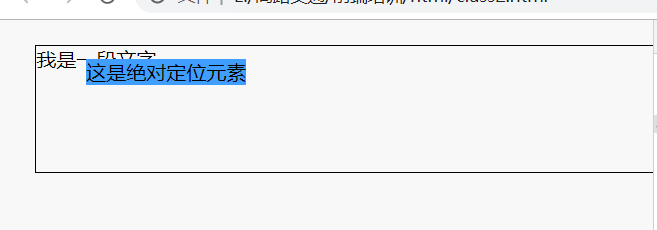


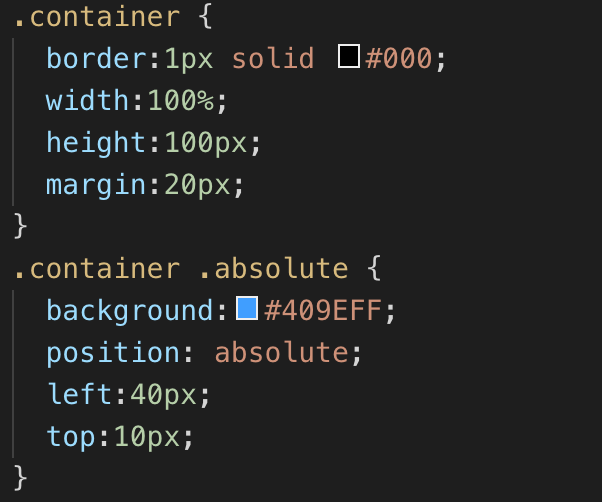




Absolute:绝对定位的元素是相对于最近的已定位父元素，如果元素没有已定位的父元素，那么它的位置相对于<html>，脱离文档流，不占据空间

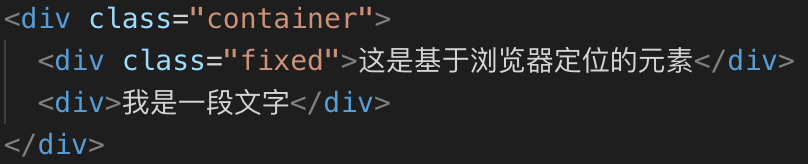


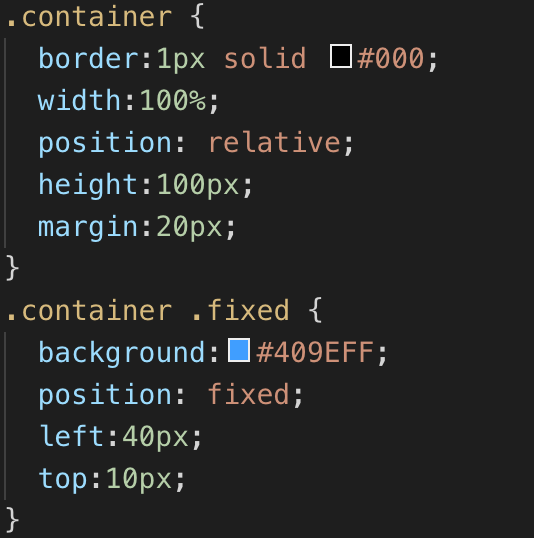


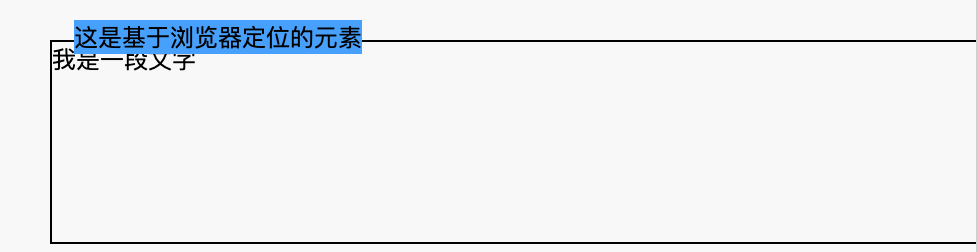


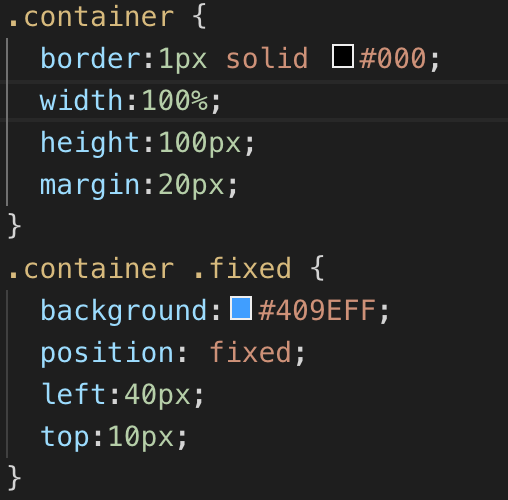


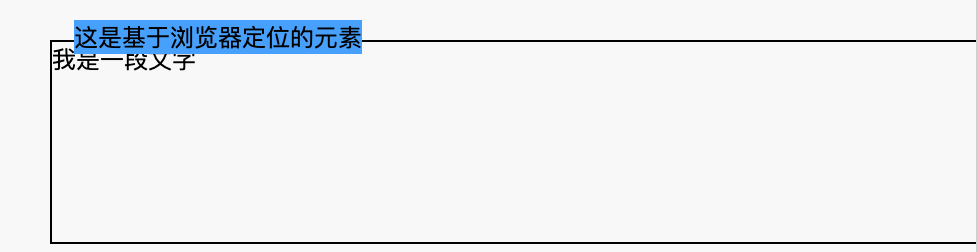
Fixed：基于浏览器窗口定位，也是脱离文档流的





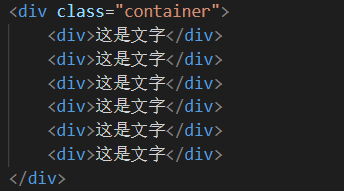


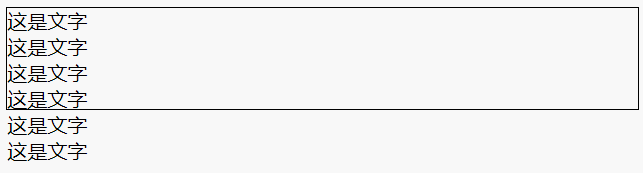
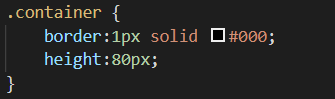




Z-index可以设置层级，两个定位的元素发生重叠，层级高的就会显示在最上面，低的则会被覆盖

Overflow：内容溢出时的显示方式 ,属性值为：hidden，scroll，auto

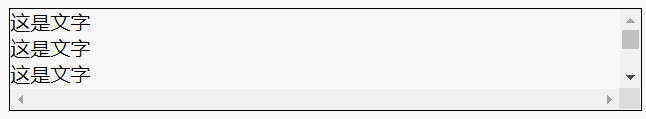




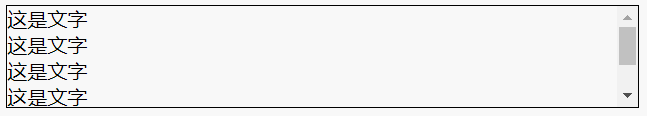












：超过才显示滚动条

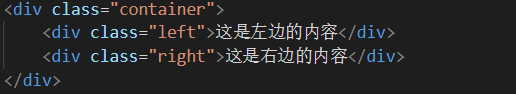
这是设置overflow-y:scroll;

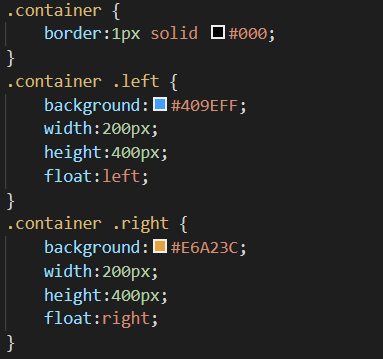


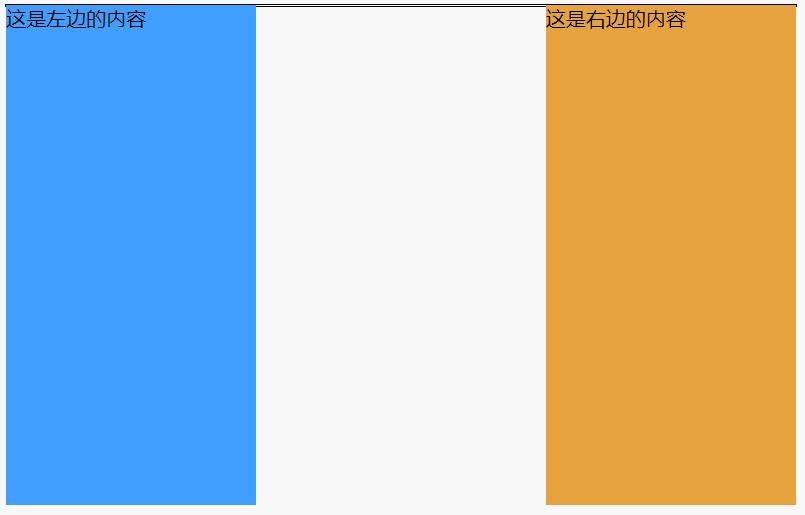
这是设置overfolw:auto;



Float：浮动；值为：none，left，right，浮动的元素也会脱离文档流，可用于布局



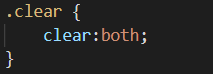


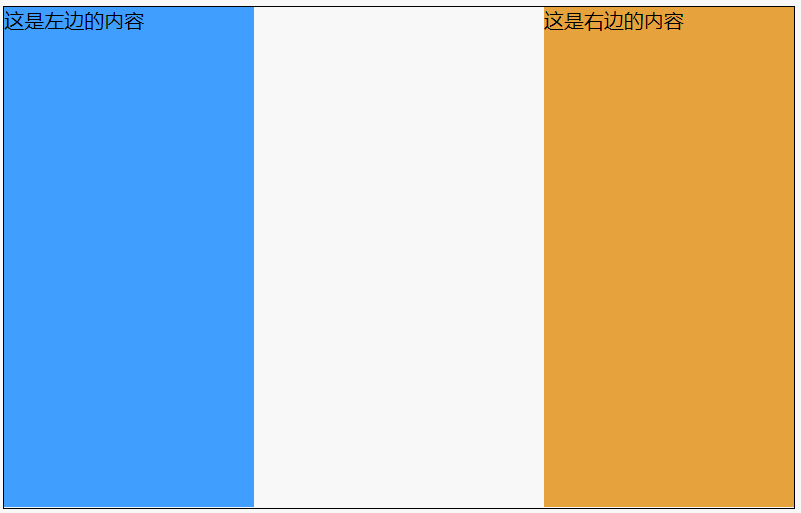


由于脱离文档流，所以父级的高度塌陷了，解决办法：清除浮动

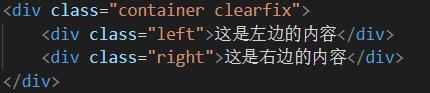
1. 在最后浮动的一个元素后新加一个标签：

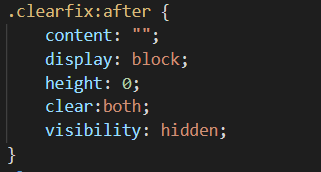


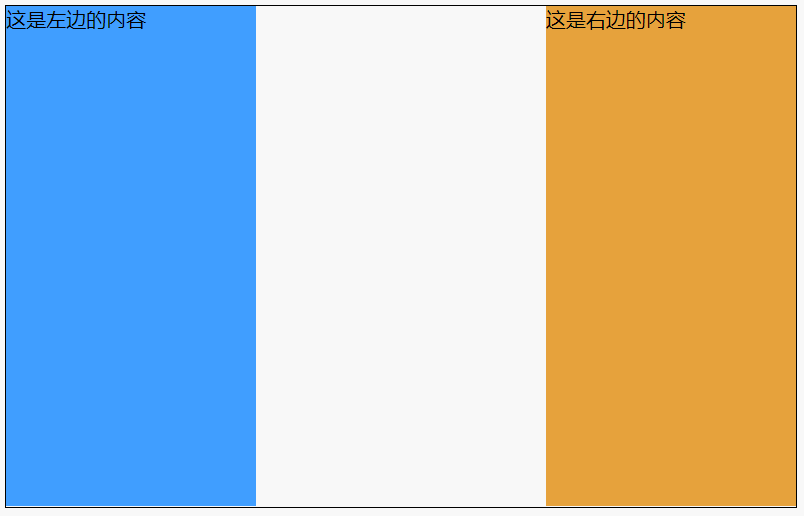




1. 使用after伪元素清除浮动：

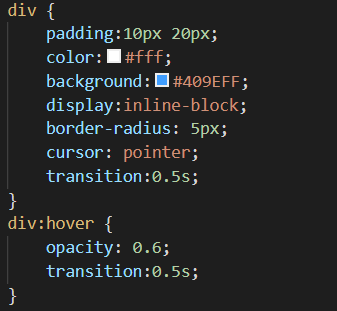






伪类：hover，鼠标移动上去





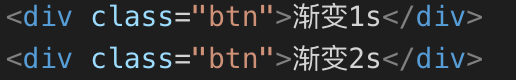


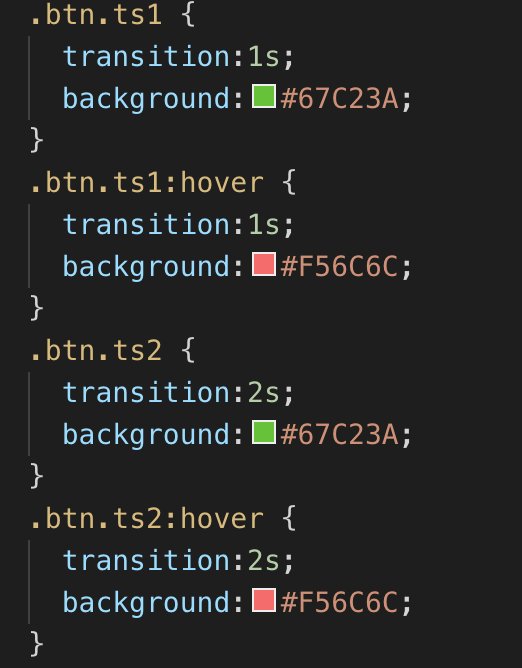
Opacity：透明度，值在0~1之间

Cursor：光标样式：值为pointer，help，wait......

* Transition

1. 颜色渐变





1. 尺寸渐变



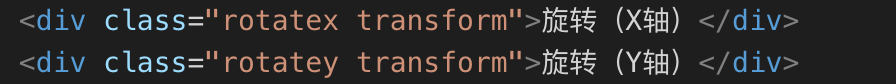


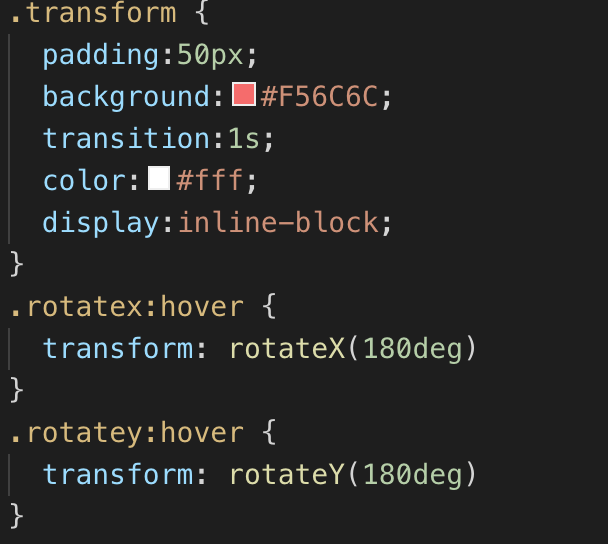
延时



* Transform

1. 旋转





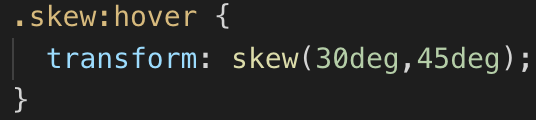
1. 缩放





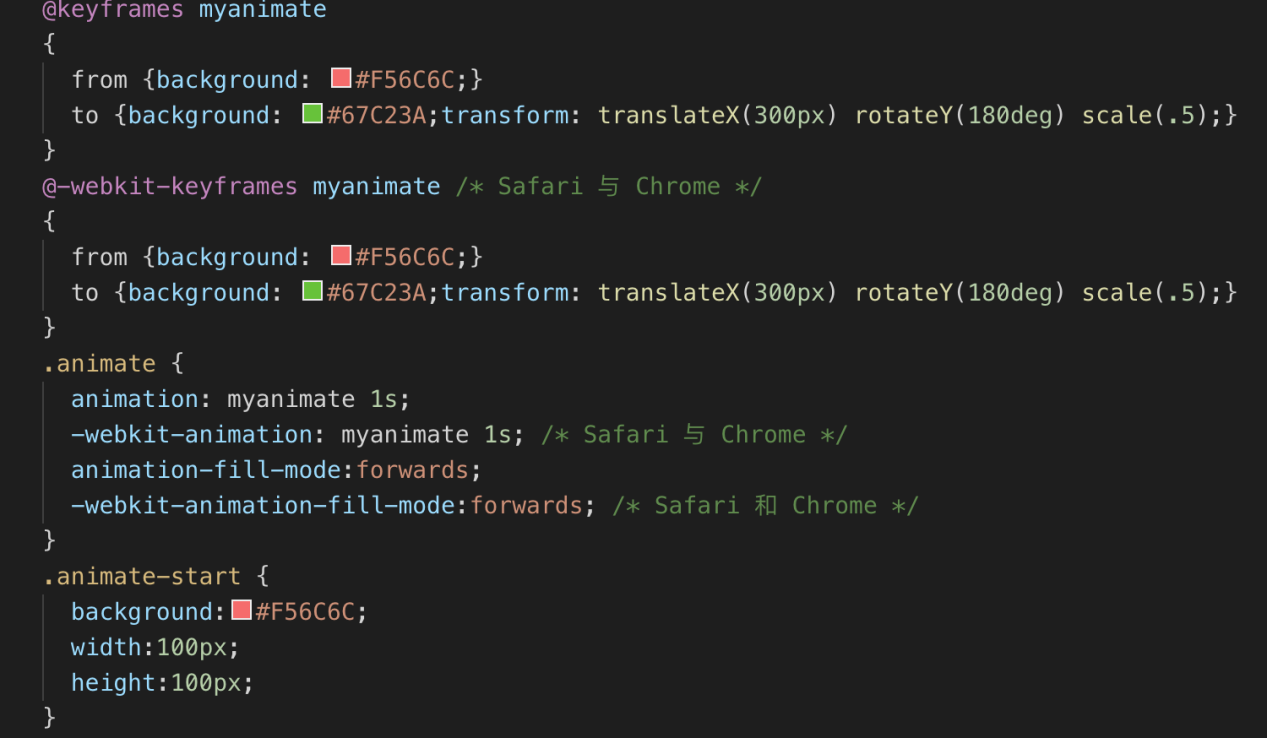
1. 倾斜





动画：





对齐：

文本居中对齐：

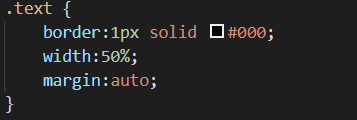






元素水平居中对齐：

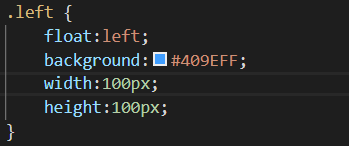


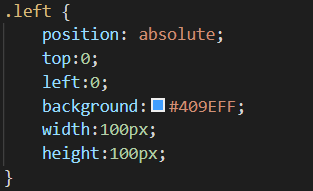




元素左右对齐：

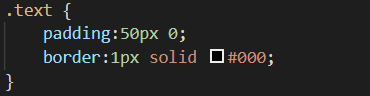




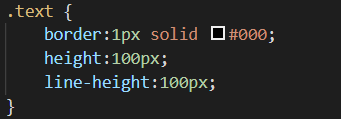


垂直居中：











垂直水平都居中：

