# IMG_25610.17IMG_256

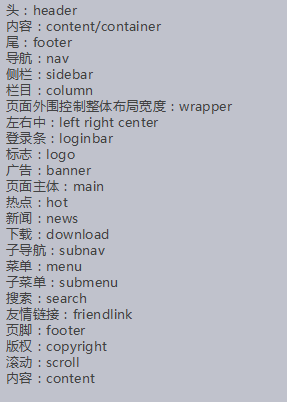
IMG_256

## 特殊字符



IMG_256

## 常见命名：



IMG_256

## H5新增语义标签

多用于移动端 移动端兼容性更好

<header>头部 banner等</header>

<nav>导航</nav>

<article>定义文章</article>

<aside>侧边栏 定义主页面内容之外的内容</aside>

<section>区块 所有浏览器都兼容 任意小模块都能用section</section>

<footer>底部 版权信息等</footer>

IMG_256

## 常见属性

|  |  |
| --- | --- |
| text-align: left | center | right; | 内容的水平对齐方式 |
| background-color: | 背景颜色 |
| text-indent: 2em; | 首行缩进 |
| word-spacing: 20px; | 单词距离（对中文无效） |
| letter-spacing: 20px; | 字与字的距离(对中文有效) |
| text-decoration:none; | 去掉下划线 |
| text-decoration: underline; | 字体加下划线 |

IMG_256

## 快捷创建方式：

.father>p.son>span.grandson >表示包含关系 div可以省略 只写一个“.”

. <div class="father">

<p class="son">

<span class="grandson"></span>

</p>

</div>

a[#]\*4 href为默认值所以可以省略

<a href="#"></a>

<a href="#"></a>

<a href="#"></a>

<a href="#"></a>

IMG_256

## 外链式样式表

放在<head></head>里边

<link rel=”stylesheet” href=”hm.css”>

rel决定了href的类型 stylesheet的意思是样式表IMG_256

## 选择器

### IMG_256标签选择器

div { }

h2 { }

### IMG_256类选择器IMG_256

.box { }

<div class="box">卢锡安</div>

P.s：

可以有很多的class="box"

一个class也可以等于很多值

例如：

<div class="box content">卢锡安</div>

<p class="box">嘉文</p>

命名规范：

☆ 不能以纯数字或者数字开头来定义类名

☆ 定义类名的时候不能出现特殊符号(\_除外)

☆ 不推荐使用中文来定义类名

☆ 不推荐使用属性（属性的值）来定义类名

☆ 使用驼峰命名法 (x\_x)

### IMG_256id选择器IMG_256

#box { }

<div id="box" >啦啦啦</div>

P.s：

一个id只能被调用一次，多次调用不符合w3c规范，更不利于js的调用。

一个标签可同时调用类选择器和id选择器，即id和class可以出现在一个div中

### IMG_256通配符选择器

\* { }

了解即可 不推荐使用 会降低页面的加载速度

### IMG_256交集选择器IMG_256

div.box { }

<div class="box">璐璐</div> // 调用

<h2 class="box">让我们带身上思考帽</h2> // 不调用

即...又的关系，即是这个标签，又调用了对应的类(id)选择器

“.”后面的类位于这个标签括号内部

### IMG_256后代选择器IMG_256

div p span { }

<div class="father">

<p class="son">

<span class="grandson"> 牛小小</span>

</p>

</div>

可以无限制隔代 只要是他的后代 都可以被选择 不管是儿子孙子重孙子

可以用标签、类名、交集等等表示

所以以上选择器也可以写成 .father .son .grandson{ } 中间一定要加空格

空格用来表示头尾标签中间包含的标签

### IMG_256子代选择器

div>span { }

<div>

<div class="father">

<p class="son">

<span class="grandson">呵呵哒</span>

</p>

<span>么么哒</span>

</div>

</div>

选择的是直接下一代 亲儿子 隔代的不选

二者为嵌套关系

快捷创建方式：.father>p.son>span.grandson

### IMG_256相邻选择器

.son+p { }

<div class="son">吃饱了</div>

<p>嘿嘿嘿</p>

+号前面的为所有的，+号后面的为紧邻的

二者必须为并列关系！！

### IMG_256并集选择器IMG_256

.box,h2,p { }

<div class="box">呵呵哒</div>

<h2>么么哒</h2>

<p>萌萌哒</p>

### IMG_256链接伪类选择器IMG_256

a :link{ } 超链接的访问之前状态 link最好省略，对浏览器更友好

a:visited { } 超链接的访问之后状态

a:hover { } 悬停

a:active { } 激活 点击时的状态

把a换成其他同理

如果这个四个状态都需要，需要按照顺序写，不然会出问题。

文本修饰：

text-decoration:none; 清除下划线

underline; 下划线

line-through; 贯穿线

### IMG_256结构(位置)伪类选择器

li:first-child { } 前面写li，ul，ul li都行 写成ul更容易理解

ul :nth-child(2){ } 写成 .box 时要加空格 即作为一个后代选择器

ul li:last-child{ }

对应：

<ul>

<li>饿饿饿</li>

<li>饿饿饿</li>

<li>饿饿饿</li>

<li>饿饿饿</li>

</ul>

### IMG_256目标伪类选择器

:target { }

<a href="#md">锚点</a>

<p>啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊啊<h2 id="md">目标伪类选择器</h2> 啊啊啊啊</p>

设置一个锚点

点击链接后 跳转到锚点 锚点呈现target中的属性

### IMG_256属性选择器

[href] { }

[href="abc.html"] { } 属性为abc.html <a href="abc.html">jinx</a>

[href\*=b] { } 只要属性值里有b <a href="abc.html">jinx</a>

[href^=a] { } 属性值以a开头 <a href="abc.jpg">jinx</a>

[href$=g] { } 属性值以g结尾 <a href="123.jpg">jinx</a>

### IMG_256首行/首字符伪元素选择器

p:first-line { } 首行

p:first-letter { } 首字

:比::兼容性好

### IMG_256文本选择状态伪元素选择器

.box::selection { }

必须写两个“:”

选中后 文字的颜色和背景颜色改变 只能作用于color 和background-color

### IMG_256前后伪元素选择器IMG_256

.box:before {

content:"前伪元素"; content为伪元素的内容 必须写

}

.box:after {

content:"后伪元素";

}

一个：或两个：

IMG_256

## 文本属性

◆文字粗细

font-weight: 400;

默认值为normal， 值从100-900，400正常粗细，700加粗(bold)

不用加px！！

◆文字风格

font-style: normal | italic;

normal默认值 italic是倾斜

◆行高

line-height: 80px;

默认值是normal

◆字体

font-family: arial,”宋体”;

Arial只能解析英文 中文按宋体解析

浏览器解析字体的时候，首先解析第一个，如果第一个能解析全部的字体，就全部解析，如果有不能解析的，交给下一个字体解析。

◆文字属性连写

font: font-weight font-style font-size/line-height font-family;

注意：文本属性连写 文字大小与字体为必写。 别漏了/

行高不写时默认为normal 会覆盖上面定义的行高

◆文字表达方式

font-family : ”宋体”;

font-family : ”microsoft yahei”;

IMG_256写字体的unicode编码

第一步： f12，调出开发者工具，找到控制台(console)

第二步： 在控制台输入escape(“中文字体名称”)，回车

第三步 将%u替换为\ 复制到源代码的“ ”中即可



IMG_256

## 颜色的表达方式

IMG_256十六进制颜色表示法

前边是#，后边是6位，用0-9、a(A)-f(F)来表示颜色

前边2位代表红色，中间代表绿色，后边2位代表蓝色

IMG_256rgb颜色表示法

background-color: rgb(255,0,0);

IMG_256rgba颜色表示法

background-color: rgba(225,0,0,0.5);

只控制该背景颜色的透明度 a为0-1 数值越小透明度越高

IMG_256opacity

opacity:0.5;

控制大括号内整个元素的透明度 范围0-1

IMG_256兼容ie6的透明度

filter: alpha(opacity=30);

此时opacity的范围是0-100

IMG_256

## Css的三大特性

### IMG_256层叠性

发生样式冲突且权重相同时，总是执行后面的样式，其他不冲突的样式还是执行之前的。前后顺序受css决定。

### IMG_256继承性

· font-size、font-weight、color、font-style、font-family、line-height、text-indent、text-align是可以被继承的。

· 只有自身没有样式时才继承父元素的样式

· 即使父元素有important 只要自身有定义样式 则遵循自身样式

· a标签不继承文字颜色

· h1-h6不继承文字大小 但会受父元素文字大小的影响

### IMG_256优先级

默认样式 < 标签选择器 < 类选择器 < id选择器 < 行内样式 < !important

0 1 10 100 1000 1000以上

· 继承的权重为0

· 先看作用到谁身上，如果作用到父元素身上则权重为0，如果作用到自身，则根据权重值推算相加

· 给的权重值只是一个抽象的便于计算的，不是实际的值，所以类选择器的权重是远远大于标签选择器的IMG_256

## 行块元素

行内块元素 行内元素 回车空格都会解析成一个空格(只有在同一行才会有空格)

### IMG_256块元素 block

典型代表:div，p，h1-h6，ul，ol，li，dl，table...

☆独占一行

☆块元素没有定义宽度的时候，会默认父元素的宽度

☆块元素默认高度为0，内容会撑开高度

☆可以设置宽高

### IMG_256行内元素(内联元素) inline

典型代表:span，strong，em，i，b，u，s，del，ins，a.....

☆可以在一行上显示

☆不能手动设置宽高

☆根据自身内容大小自动形成宽高， 如果没有内容则默认宽高为0

### IMG_256行内块元素 inline-block

典型代表: input，img，textarea.....

☆在一行上显示

☆可以设置宽高

### IMG_256元素显示模式的转换

◆转行内 display:inline;

◆转块 display: block;

◆转行内块: display: inline-block;IMG_256

## 背景属性

◆背景颜色

background-color: #123456; transparent 默认值 透明

◆背景图片

background-image: url（2.png）; none为默认值

◆背景平铺

background-repeat: repeat; 平铺 默认值 沿x轴、y轴平铺

repeat-x; 沿水平方向平铺

repeat-y; 沿垂直方向平铺

no-repeat; 不平铺

◆背景定位

方位值： left | right | top | bottom | center

background-position: 12px 50px;

background-position: right top;

★写2个方位值，顺序没有要求

★写1个方位值的时候，另外一个默认center

★写2个数值，第一个是距左边的距离，第二个距顶边的距离

★写1个数值的时候，默认为距左边的距离，垂直方向居中

★既有数值又有方位值时 前面代表水平 后面代表垂直 否则出来效果为默认值

规律： 没有数值 顺序随便 只要有数值 左边代表水平距离 右边代表垂直距离

◆背景附着

background-attachment: scroll; 默认值 背景图片随着滚轴滚动

fixed; 固定 背景图片不随着滚轴滚动 即一直显示在视野中

当值为fixed时 图片的定位依据于浏览器 即可能会跳出当前范围

### ◆背景属性连写

background: #ccc url(2.png) no-repeat 10px 20px scroll;

背景属性连写时候，最好有背景图片IMG_256

## 背景缩放

background-size: 100px (50%) ;

· 可以写数值也可以写百分比

百分比指的是盒子宽高的百分比 不是自身宽度的百分比

· 只写一个值的时候 代表的是宽度 高度默认auto

只要有auto一定为等比例缩放

特殊：

☆ background-size: cover;

背景图片等比例缩放，填满盒子，背景图片有可能显示不完整。

☆ background-size:contain;

背景图片等比例缩放，背景图片完整显示，有可能填不满盒子。

IMG_256

## 多背景图片

background: url(bg/bg1.png) left top,

url(bg/bg2.png) right top,

url(bg/bg3.png) right bottom,

url(bg/bg4.png) left bottom,

url(bg/bg5.png) center;

background-repeat: no-repeat;

注意：图片之间用逗号隔开

IMG_256

## 行高

浏览器默认文字大小16px

行框=文字大小+上间距+下间距

line-height: 80px; 行高

height: 80px; 容器高

行高 = 容器高 文字垂直居中

行高 > 容器高 文字偏下

行高 < 容器高 文字偏上

IMG_256

## 盒模型

### IMG_256 边框

◇ 单边框

单独一个边的属性

border-left-color: red;

border-left-width: 10px;

border-top-style: solid 实线

dashed 虚线

dotted 点线

◇ 单边框 连写

border-right: 20px solid orange;

像素默认值为1px 颜色默认黑色 边框类型必写

◇ 四边框 连写

border:10px solid #666;

### IMG_256 表格边框的合并

border-collapse:collapse;

放在table中 只针对表格

### IMG_256 表单边框

border: 0 none;

边框 0和none 同时写 兼容性更好

outline-style: none;

清除轮廓线

:focus { }

获取光标焦点的状态

### IMG_256 外边距

margin-bottom: 50px;

◇ 外边距连写：

Margin: 50px; 上下左右

Margin: 50px 40px; 上下 左右

Margin: 50px 40px 30px; 上 左右 下

Margin:50px 40px 30px 20px; 上 右 下 左 顺时针

margin：0 auto； 块元素水平居中

★ 做页面时首先清除外边距

body,h4,ul {

margin: 0;

padding: 0;

}

◇ 垂直方向外边距合并 （只发生在块元素身上）

上面的：margin bottom：100px；

下面的：margin top：50px；

则实际间距为100px 取最大值 该现象只存在于垂直方向

◇ 垂直方向外边距塌陷（俗称坑爹）（只发生在块元素身上）

① ②

当给1设置上外边距时 会把2也给带下来（前提是父元素没有上边框）

· 解决方案：

1. 给父元素设置顶边框 不推荐使用
2. 给父元素使用overflow：hidden；

### IMG_256 内边距

padding-bottom: 50px;

内容区域距边框的距离 可以连写

### IMG_256 内边距撑大盒子的问题

盒子的宽度 = 边框的粗细 + 左右内边距 + 设置的宽度

### IMG_256 内边距不撑大盒子的问题

如果子元素没有设置宽度 则默认父元素宽度

大前提 ：当子元素没有设置宽度时

当 子元素内边距<父元素边距 时 不会被撑宽

当 子元素内边距>父元素边距 时 会被撑宽

解决问题时 首先考虑用行高解决 其次考虑用内边距 最后考虑用外边距

IMG_256

## Ps的基本使用

Ctrl+r 隐藏(调出)标尺

Ctrl+: 隐藏(调出)辅助线

清除辅助线：视图——清除辅助线

Ctrl+”+”、 ”\_” 放大、缩小画布显示比例

测量方法：

1. 矩形选区 选择测量区域 F8调出信息面板，可以看到选区的宽高
2. 用切片工具 在切片上双击，调出切片选项



IMG_256

## markman的基本使用

★ 智能测量 按住tab键，L切换测量方向

IMG_256

## 行内元素和行内块元素的内外边距问题

◆行内元素不能设置上下的内外边距，可以设置左右的内外边距

◆行内块元素可以设置上下左右内外边距

IMG_256

## Css3的盒模型

### IMG_256内减模式

box-sizing: content-box; 默认值 正常模式，内边距会撑大盒子

border-box; 盒子不会被撑大 而是内容区域被压缩

内容区域=设置的宽高-内边距的值

### IMG_256盒子的阴影

box-shadow: 10px 10px 10px 10px #666;

五个值分别代表： 水平偏移量（正右负左） 垂直偏移量（正下负上）

阴影的虚化程度 阴影的大小 颜色

分别为0时：水平居中 垂直居中 阴影为实线 阴影和盒子重叠

· 可以设置多个阴影 用逗号隔开

box-shadow: 10px 10px 10px 10px #666，10px 10px 10px 10px #fff;

· 设置内阴影

box-shadow: inset 10px 10px 10px 10px #666;

### IMG_256文字的阴影

text-shadow: 10px 5px 4px #ca64c4;

三个值分别代表 水平偏移 垂直偏移 模糊范围

可以设置多个阴影 用逗号隔开

### IMG_256盒子的圆角

不同的表达方式：

单个角

❀ border-top-left-radius:20px 70px; 左上角 用20x70的圆切 radius要放最后

四个角

❀ border-radius: 20px; 四个角 都用20x20的圆切

❀ border-radius: 10px 20px 30px 40px; 四个角用正圆切

分别代表左上、右上、右下、左下（顺时针）

❀border-radius: 10px 20px 30px 40px / 20px 30px 40px 50px ;四个角用不规则圆切

同一个圆

正圆：

1. 建一个正方形盒子
2. border-radius：50%； 只要大于等于50%都可以

IMG_256

## 文档流（标准流）

在页面上，元素自上而下、自左而右，块元素独占一行，行内元素在一行上显示，直到碰到父元素的边框换行。

IMG_256

## 浮动技术

float: left | right | none;

### IMG_256浮动的特点

❀ 浮动元素脱离标准流，不占据原来的位置

❀ 使元素在一行显示 且代码换行后 元素之间没有空格

❀ 在父元素内部浮动 受父元素内边距约束

❀ 浮动的元素顶对齐

❀ 行内元素浮动之后，转换为了行内块的显示模式（不推荐用这个方法转行内块 因为脱标了）

❀ 浮动元素的默认宽高为0 但会被内容撑开

❀ 代码上一行的块元素不会受浮动影响 代码下一行的块元素会顶上来

❀ 行内元素会被浮动挤到后面 但还是在同一行 且原位置会留出此元素的位置

IMG_256浮动的作用

❀ 文本饶图 （文本不参与浮动）

❀ 制作导航

❀ 页面布局

### IMG_256清除浮动

当父容器没有设置高度，里面的盒子没有设置浮动的情况下会将父容器的高度撑开。一旦父容器中的盒子设置浮动，脱离标准文档流，父容器立马没有高度，下面的盒子会跑到浮动的盒子下面。出现这种情况，我们需要清除浮动

清除浮动不是不浮动，是清除浮动产生的不利影响

❀ 方法一：额外标签法

在最后一个浮动元素的后边，添加额外标签

<div style=“clear:both; ”></div>

不推荐使用，会产生冗余代码。



❀ 方法二：

给浮动元素的父元素使用overflow：hidden

弊端： 如果有浮动元素溢出了父元素，则这个溢出会被隐藏掉

❀ 方法三：

给要清除浮动的元素的父元素使用

◇ 单伪元素清除浮动

.clearfix:after {

content:""; 必写属性 里面可以为空

display: block;

height: 0;

line-height: 0;

visibility:hidden; 显示模式为隐藏

clear:both; 清除浮动

}

.clearfix {

zoom:1; 兼容IE 缩放

}

◇ 双伪元素清除浮动

.clearfix:before,.clearfix:after {

content:””;

dispaly:table;

}

.clearfix:after {

clear:both;

}

Html代码：

Class=” clearfix”

❀ 方法四：

给父元素设置高度 （不推荐）

IMG_256

## 定位

### IMG_256 静态定位

position:static;

默认值 标准流

### IMG_256 绝对定位

position: absolute;

特点：

❀ 脱离标准流 不占据原来的位置

❀ 如果父元素没有使用定位 子元素的绝对定位基于浏览器偏移

如果父元素使用了定位 子元素的绝对定位基于父元素的位置偏移

❀ 行内元素使用绝对定位时 转换为行内块显示模式

（不推荐用这个方式转换 因为脱离了标准流）

### IMG_256 相对定位

position: relative;

❀ 不脱离标准流

❀ 基于自己本身的位置偏移

top：100px； 即 本身在现在的上边

❀ 子绝父相 子元素的位置基于父元素偏移

❀ 行内元素使用想对定位时 不能转换为行内块显示模式

### IMG_256 固定定位

position: fixed;

❀ 脱离标准流 不占据原来的位置

❀ 基于浏览器窗口定位

❀ 行内元素使用固定定位时 转换为行内块显示模式（不推荐使用）

### IMG_256 定位元素层叠顺序

❀ 默认顺序： 在html中越靠后 层数越靠上

❀ z-index: 0;

值越大 层数越靠上，默认值auto或者0 ，范围1-999 可以为负

### IMG_256 定位的盒子居中显示

.father {

width: 500px;

height: 300px;

background-color: red;

position: relative;

}

.son {

width: 50px;

height: 50px;

background-color: #fff;

position: absolute;

top: 50%;

right:50%;

margin-right:-25px;

margin-top: -25px;

}

规律：需要定位的图片放在div中

IMG_256

## vertical-align

用在行内块元素的对齐上

vertical-align: baseline; 基线

top; 顶对齐

bottom; 底对齐

middle; 垂直居中

规律：图片默认和基线对齐 在样式最前面写 vertical-align top

IMG_256

## overflow溢出的可见

overflow: visible; 默认值 溢出可见

overflow: hidden; 溢出隐藏

overflow: auto; 溢出生成滚动条

overflow: scroll; 不管是否溢出 都生成滚动条

IMG_256

## 元素的显示与隐藏

display:none; 元素不可见，不占据原来位置

display:block; 元素可见

visibility:hidden; 元素隐藏，还占据原来的位置

overflow:hidden; 溢出隐藏

IMG_256

## 标签的嵌套规范

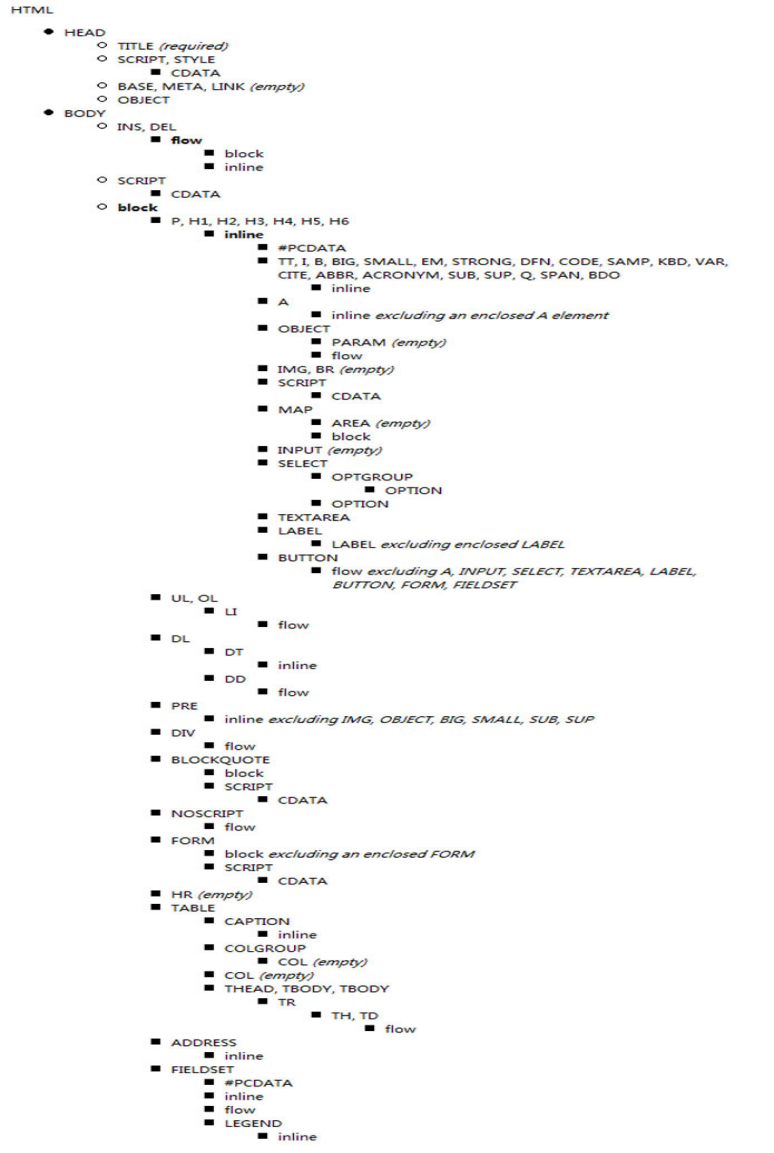
◆div可以嵌套任意标签

◆p标签里不要嵌套其他的块元素

◆块元素里可以嵌套行内元素和行内块元素

◆行内块元素里可以嵌套行内块元素和行内元素

◆行内元素里只能嵌套行内元素

IMG_256

## 文本换行

word-break: break-all; 换行的时候不另起一行 以字母为单位 单词会被拆散

word-wrap:break-word; 换行的时候另起一行 以单词为单位 单词不会被拆散

IMG_256

## 伸缩盒子

一般将父元素设置为伸缩盒子

### IMG_256 主轴元素的对齐方式

justify-content: flex-start; 默认对齐方式 沿主轴开始处对齐

flex-end; 沿主轴结束处对齐

center; 主轴方向居中

space-between; 两端对齐，中间等距

space-around; 中间等距，两边是等距的一半

### IMG_256 侧轴的对齐方式

align-items: flex-start; 默认值 侧轴默认对齐方式

flex-end; 侧轴结束方向对齐

center; 侧轴居中对齐

base-line; 基线对齐

stretch; 拉伸 ，需要将宽度(高度)设置为auto,或者不设置

### IMG_256 调整主轴的方向

flex-direction: row; 默认值 主轴水平方向

column; 主轴垂直方向，侧轴变为水平方向

row-reverse; 水平方向反向显示

column-reverse; 垂直方向反向显示

### IMG_256 多行

flex-wrap: nowrap; 默认值 不换行

wrap; 多行模式

wrap-reverse; 多行反向显示

### IMG_256 多行对齐

align-content: stretch; 拉伸 宽或者高不设置，或者设置为auto

flex-start; 多行开始位置对齐

flex-end; 多行结束位置对齐

space-between; 中间等距，两边对齐

space-around; 中间等距，两边是等距的一半

center; 居中对齐

### IMG_256 伸缩盒子子元素比例显示

给子元素设置,flex:1; flex:2; flex:3; flex:4;将父元素拆分10份,各自占相对应份数。

IMG_256

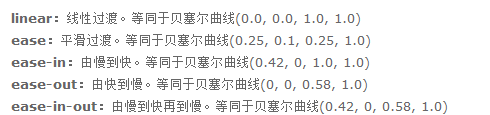
## 过渡

transition-property: all; 属性：全部（默认值） （left、top）

Transition-duration: 1s; 过渡时间 默认值为0

transition-delay: 1s; 过渡延时 默认值0

transition-timing-function: ease; 过渡速度变化 默认值ease



过渡属性连写：

transition: all 1s ease 1s ;

注意：过渡时间为必写属性 只写一个时间时默认为过渡时间

IMG_256

## 2d转换

### IMG_256 位置的偏移

transform:translate(400px,200px);

写1个值代表水平方向平移，2个值第二个值代表垂直平移，负值反方向。

IMG_256 旋转

Transform: rotateX(60deg); 沿x轴旋转

rotateY(120deg); 沿Y轴旋转

rotateZ(360deg); 沿z轴旋转 z轴可省略不写

IMG_256 缩放

Transform: scaleX(2); 宽度2倍

scaleY(3); 高度3倍

Scale(2); 等比例缩放

Scale(2,3); 第一个值宽度缩放比例，第二个高度缩放比例

IMG_256 倾斜

Transform: skewX(30deg); 沿X轴倾斜

skewY(45deg); 沿Y轴倾斜

Skew(30deg,45deg); 第一个值沿X轴倾斜，第二值沿Y轴倾斜

IMG_256 转换中心点

Transform-origin: left top; 写2个方位值

Left； 另外一个值默认center

20px; 距左边20px,另外一个默认center

20px 50px; 距左边20px, 距顶边50px

Rotate 平面旋转 单位deg