**CSS知识点总结**

学号：20171310640 姓名：满婉婷

内部样式表

内嵌式是将CSS代码集中写在HTML文档的head头部标签中，并且用style标签定义，其基本语法格式如下：

<head>

<style type="text/css">

选择器 {属性1:属性值1; 属性2:属性值2; 属性3:属性值3;}

</style>

</head>

语法中，style标签一般位于head标签中title标签之后，也可以把他放在HTML文档的任何地方。type="text/CSS" 在html5中可以省略， 写上也比较符合规范， 所以这个地方可以写也可以省略。

行内式（内联样式）

内联样式，又有人称行内样式、行间样式、内嵌样式。是通过标签的style属性来设置元素的样式，其基本语法格式如下：

<标签名 style="属性1:属性值1; 属性2:属性值2; 属性3:属性值3;"> 内容 </标签名>

语法中style是标签的属性，实际上任何HTML标签都拥有style属性，用来设置行内式。其中属性和值的书写规范与CSS样式规则相同，行内式只对其所在的标签及嵌套在其中的子标签起作用。

外部样式表（外链式）

链入式是将所有的样式放在一个或多个以.CSS为扩展名的外部样式表文件中，通过link标签将外部样式表文件链接到HTML文档中，其基本语法格式如下：

<head>

<link href="CSS文件的路径" rel="stylesheet" />

</head>

该语法中，link标签需要放在head头部标签中，并且必须指定link标签的三个属性，具体如下：

href：定义所链接外部样式表文件的URL，可以是相对路径，也可以是绝对路径。

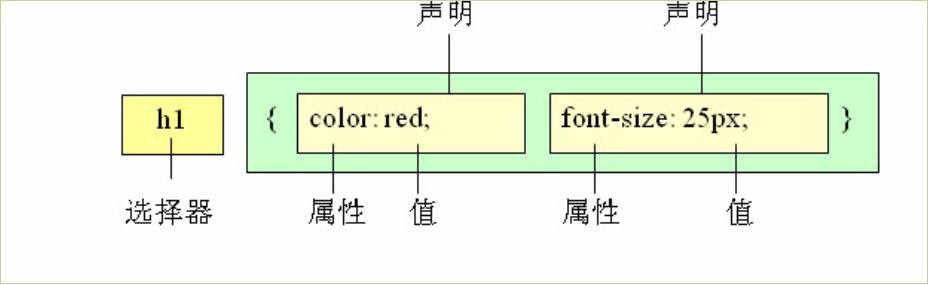
type：定义所链接文档的类型，在这里需要指定为“text/css”，表示链接的外部文件为CSS样式表。

| 样式表 | 优点 | 缺点 | 使用情况 | 控制范围 |
| --- | --- | --- | --- | --- |
| 行内样式表 | 书写方便，权重高 | 没有实现样式和结构分离 | 较少 | 控制一个表情 |
| 内部样式表 | 部分结构和样式分离 | 没有彻底分离 | 较多 | 控制一个页面 |
| 外部样式表 | 完全实现结构和样式相分离 | 需要引入 | 最多，强烈推荐 | 控制整个站点 |

rel：定义当前文档与被链接文档之间的关系，在这里需要指定为“stylesheet”，表示被链接的文档是一个样式表文件。

三种样式表总结

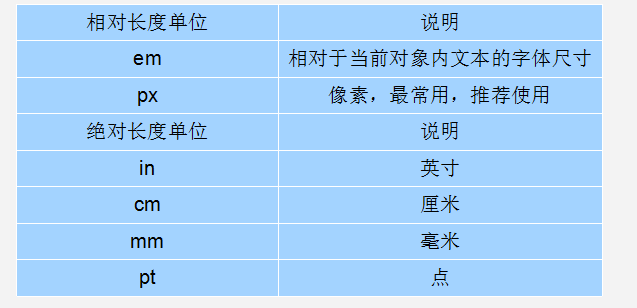
CSS样式规则

使用HTML时，需要遵从一定的规范。CSS亦如此，要想熟练地使用CSS对网页进行修饰，首先需要了解CSS样式规则，具体格式如下：  


在上面的样式规则中:  
1.选择器用于指定CSS样式作用的HTML对象，花括号内是对该对象设置的具体样式。  
2.属性和属性值以“键值对”的形式出现。  
3.属性是对指定的对象设置的样式属性，例如字体大小、文本颜色等。  
4.属性和属性值之间用英文“:”连接。  
5.多个“键值对”之间用英文“;”进行区分。

CSS字体样式属性

font-size:字号大小

font-size属性用于设置字号，该属性的值可以使用相对长度单位，也可以使用绝对长度单位。其中，相对长度单位比较常用，推荐使用像素单位px，绝对长度单位使用较少。具体如下：

font-family:字体

font-family属性用于设置字体。网页中常用的字体有宋体、微软雅黑、黑体等，例如将网页中所有段落文本的字体设置为微软雅黑，可以使用如下CSS样式代码：

p{ font-family:"微软雅黑";}

可以同时指定多个字体，中间以逗号隔开，表示如果浏览器不支持第一个字体，则会尝试下一个，直到找到合适的字体。

常用技巧：  
1.现在网页中普遍使用14px+。  
2.尽量使用偶数的数字字号。ie6等老式浏览器支持奇数会有bug。  
3.各种字体之间必须使用英文状态下的逗号隔开。  
4.中文字体需要加英文状态下的引号，英文字体一般不需要加引号。当需要设置英文字体时，英文字体名必须位于中文字体名之前。  
5.如果字体名中包含空格、#、$等符号，则该字体必须加英文状态下的单引号或双引号，例如font-family: "Times New Roman";。  
6.尽量使用系统默认字体，保证在任何用户的浏览器中都能正确显示。

CSS Unicode字体

在 CSS 中设置字体名称，直接写中文是可以的。但是在文件编码（GB2312、UTF-8 等）不匹配时会产生乱码的错误。xp 系统不支持 类似微软雅黑的中文。

方案一： 你可以使用英文来替代。 比如 font-family:"Microsoft Yahei"。

方案二： 在 CSS 直接使用 Unicode 编码来写字体名称可以避免这些错误。使用 Unicode 写中文字体名称，浏览器是可以正确的解析的。  
font-family: "\5FAE\8F6F\96C5\9ED1"，表示设置字体为“微软雅黑”。

可以通过 js 的 escape() 方法来测试属于什么字体。为了照顾不同电脑的字体安装问题，尽量只使用宋体和微软雅黑中文字体

| 字体名称 | 英文名称 | Unicode 编码 |
| --- | --- | --- |
| 宋体 | SimSun | \5B8B\4F53 |
| 新宋体 | NSimSun | \65B0\5B8B\4F53 |
| 黑体 | SimHei | \9ED1\4F53 |
| 微软雅黑 | Microsoft YaHei | \5FAE\8F6F\96C5\9ED1 |
| 楷体\_GB2312 | KaiTi\_GB2312 | \6977\4F53\_GB2312 |
| 隶书 | LiSu | \96B6\4E66 |
| 幼园 | YouYuan | \5E7C\5706 |
| 华文细黑 | STXihei | \534E\6587\7EC6\9ED1 |
| 细明体 | MingLiU | \7EC6\660E\4F53 |
| 新细明体 | PMingLiU | \65B0\7EC6\660E\4F53 |

font-weight:字体粗细

字体加粗除了用 b 和 strong 标签之外，可以使用CSS 来实现，但是CSS 是没有语义的。

font-weight属性用于定义字体的粗细，其可用属性值：normal、bold、bolder、lighter、100~900（100的整数倍）。

数字 400 等价于 normal，而 700 等价于 bold。 但是常用数字来表示。

font-style:字体风格

字体倾斜除了用 i 和 em 标签之外，可以使用CSS 来实现，但是CSS 是没有语义的。平时很少给文字加斜体，反而喜欢给斜体标签（em，i）改为普通模式。

font-style属性用于定义字体风格，如设置斜体、倾斜或正常字体，其可用属性值如下：

normal：默认值，浏览器会显示标准的字体样式。

italic：浏览器会显示斜体的字体样式。

oblique：浏览器会显示倾斜的字体样式。

font:综合设置字体样式

font属性用于对字体样式进行综合设置，其基本语法格式如下：

选择器{font: font-style font-weight font-size/line-height font-family;}

使用font属性时，必须按上面语法格式中的顺序书写，不能更换顺序，各个属性以空格隔开。

注意：其中不需要设置的属性可以省略（取默认值），但必须保留font-size和font-family属性，否则font属性将不起作用。

CSS外观属性

color:文本颜色

color属性用于定义文本的颜色，其取值方式有如下3种：  
1.预定义的颜色值，如pink，red，green，blue等。  
2.十六进制，如#FF0000，#F00，#FF6600，#29D794等。实际工作中，十六进制是最常用的定义颜色的方式。  
3.RGB代码，如红色可以表示为rgb(255,0,0)或rgb(100%,0%,0%)。

line-height:行间距

line-height属性用于设置行间距，就是行与行之间的距离，即字符的垂直间距，一般称为行高。line-height常用的属性值单位有三种，分别为像素px，相对值em和百分比%，实际工作中使用最多的是像素px。

一般情况下，行距比字号大7.8像素左右就可以了。

text-align:水平对齐方式

text-align属性用于设置文本内容的水平对齐，相当于html中的align对齐属性。其可用属性值如下：

left：左对齐（默认值）

right：右对齐

center：居中对齐

text-indent:首行缩进

text-indent属性用于设置首行文本的缩进，其属性值可为不同单位的数值、em字符宽度的倍数、或相对于浏览器窗口宽度的百分比%，允许使用负值, 建议使用em作为设置单位。

1em 就是一个字的宽度 如果是汉字的段落， 1em 就是一个汉字的宽度，首行缩进 2 个汉字，就是 2em。

text-decoration 文本的装饰

text-decoration 通常我们用于给链接修改装饰效果：

| 值 | 描述 |
| --- | --- |
| none | 默认。定义标准的文本。 |
| underline | 定义文本下的一条线。下划线 也是我们链接自带的 |
| overline | 定义文本上的一条线。 |
| line-through | 定义穿过文本下的一条线。 |

开发者工具（chrome）

“按F12”或者是 “shift+ctrl+i” 打开 开发者工具。或者 菜单： 右击网页空白出---查看/检查

小技巧：  
1.ctrl+滚轮 可以 放大开发者工具代码大小。  
2.左边是HTML元素结构 右边是CSS样式。右边CSS样式可以改动数值和颜色查看更改后效果。

选择器

要想将CSS样式应用于特定的HTML元素，首先需要找到该目标元素。在CSS中，执行这一任务的样式规则部分被称为选择器（选择符）。

CSS基础选择器

标签选择器（元素选择器）

标签选择器是指用HTML标签名称作为选择器，按标签名称分类，为页面中某一类标签指定统一的CSS样式。其基本语法格式如下：

标签名{属性1:属性值1; 属性2:属性值2; 属性3:属性值3; } 或者

元素名{属性1:属性值1; 属性2:属性值2; 属性3:属性值3; }

标签选择器最大的优点是能快速为页面中同类型的标签统一样式，同时这也是他的缺点，不能设计差异化样式。

类选择器

类选择器使用“.”（英文点号）进行标识，后面紧跟类名，其基本语法格式如下：

.类名{属性1:属性值1; 属性2:属性值2; 属性3:属性值3; }

标签调用的时候用 class=“类名” 即可。

类选择器最大的优势是可以为元素对象定义单独或相同的样式。 可以选择一个或者多个标签。

长名称或词组可以使用中横线来为选择器命名。  
不建议使用“\_”下划线来命名CSS选择器。  
不要纯数字、中文等命名， 尽量使用英文字母来表示。

多类名选择器

我们可以给标签指定多个类名，从而达到更多的选择目的。  
注意：

样式显示效果跟HTML元素中的类名先后顺序没有关系,受CSS样式书写的上下顺序有关。

各个类名中间用空格隔开。

<div class="pink fontWeight font20">亚瑟</div>

<div class="font20">刘备</div>

<div class="font14 pink">安其拉</div>

<div class="font14">貂蝉</div>

id选择器

id选择器使用“#”进行标识，后面紧跟id名，其基本语法格式如下：

#id名{属性1:属性值1; 属性2:属性值2; 属性3:属性值3; }

该语法中，id名即为HTML元素的id属性值，大多数HTML元素都可以定义id属性，元素的id值是唯一的，只能对应于文档中某一个具体的元素。用法基本和类选择器相同。

id选择器和类选择器区别

W3C标准规定，在同一个页面内，不允许有相同名字的id对象出现，但是允许相同名字的class。类选择器（class） 好比人的名字， 是可以多次重复使用的。id选择器 好比人的身份证号码， 全中国是唯一的， 不得重复。

通配符选择器

通配符选择器用 " \* "号表示，他是所有选择器中作用范围最广的，能匹配页面中所有的元素。其基本语法格式如下：

\* { 属性1:属性值1; 属性2:属性值2; 属性3:属性值3; }

例如下面的代码，使用通配符选择器定义CSS样式，清除所有HTML标记的默认边距。

\* {

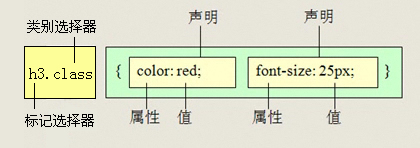
margin: 0; /\* 定义外边距\*/

padding: 0; /\* 定义内边距\*/

}

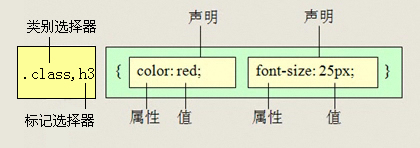
CSS复合选择器

复合选择器是由两个或多个基础选择器，通过不同的方式组合而成的,目的是为了可以选择更准确更精细的目标元素标签。

交集选择器

交集选择器由两个选择器构成，其中第一个为标签选择器，第二个为class选择器，两个选择器之间不能有空格，如h3.special。

交集选择器 是 并且的意思。 即...又...的意思

比如： p.one 选择的是： 类名为 .one 的 段落 p 标签。

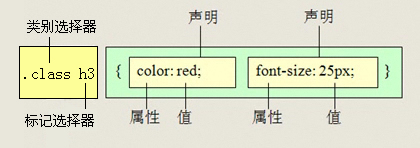
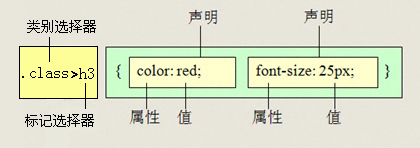
并集选择器

并集选择器（CSS选择器分组）是各个选择器通过 逗号 连接而成的，任何形式的选择器（包括标签选择器、class类选择器id选择器等），都可以作为并集选择器的一部分。如果某些选择器定义的样式完全相同，或部分相同，就可以利用并集选择器为它们定义相同的CSS样式。

并集选择器 和 的意思， 就是说，只要逗号隔开的，所有选择器都会执行后面样式。

比如 .one, p , #test {color: #F00;} 表示 .one 和 p 和 #test 这三个选择器都会执行颜色为红色。 通常用于集体声明。

后代选择器

后代选择器又称为包含选择器，用来选择元素或元素组的后代，其写法就是把外层标签写在前面，内层标签写在后面，中间用空格分隔。当标签发生嵌套时，内层标签就成为外层标签的后代。

子孙后代都可以这么选择。 或者说，它能选择任何包含在内 的标签。

子元素选择器

子元素选择器只能选择作为某元素子元素的元素。其写法就是把父级标签写在前面，子级标签写在后面，中间跟一个 > 进行连接，注意，符号左右两侧各保留一个空格。

这里的子 指的是 亲儿子 不包含孙子 重孙子之类。

比如： .demo > h3 {color: red;} 说明 h3 一定是demo 亲儿子。 demo 元素包含着h3。

伪类选择器

伪类选择器用于向某些选择器添加特殊的效果。比如给链接添加特殊效果， 比如可以选择 第1个，第n个元素。

为了和我们刚才学的类选择器相区别， 类选择器是一个点 比如 .demo {} 而我们的伪类 用 2个点 就是 冒号 比如 :link{}

链接伪类选择器

:link /\* 未访问的链接 \*/

:visited /\* 已访问的链接 \*/

:hover /\* 鼠标移动到链接上 \*/

:active /\* 选定的链接 \*/

注意写的时候，他们的顺序尽量不要颠倒 按照 lvha 的顺序。 love hate 爱上了讨厌 记忆法 或者 lv 包包 非常 hao

a { /\* a是标签选择器 所有的链接 \*/

font-weight: 700;

font-size: 16px;

color: gray;

}

a:hover { /\* :hover 是链接伪类选择器 鼠标经过 \*/

color: red; /\* 鼠标经过的时候，由原来的 灰色 变成了红色 \*/

}

CSS注释

CSS规则是使用 /\* 需要注释的内容 \*/ 进行注释的，即在需要注释的内容前使用 /\* 标记开始注释，在内容的结尾使用 \*/ 结束。

标签显示模式（display）

为了更好的适应环境而完成的自然选择。HTML标签一般分为块标签和行内标签两种类型，它们也称块元素和行内元素。

块级元素(block-level)

每个块元素通常都会独自占据一整行或多整行，可以对其设置宽度、高度、对齐等属性，常用于网页布局和网页结构的搭建。

常见的块元素有<h1>~<h6>、<p>、<div>、<ul>、<ol>、<li>等，其中<div>标签是最典型的块元素。

块级元素的特点：  
1.总是从新行开始  
2.高度，行高、外边距以及内边距都可以控制。  
3.宽度默认是容器的100%  
4.可以容纳内联元素和其他块元素。

行内元素(inline-level)

行内元素（内联元素）不占有独立的区域，仅仅靠自身的字体大小和图像尺寸来支撑结构，一般不可以设置宽度、高度、对齐等属性，常用于控制页面中文本的样式。

常见的行内元素有<a>、<strong>、<b>、<em>、<i>、<del>、<s>、<ins>、<u>、<span>等，其中<span>标签最典型的行内元素。

行内元素的特点：

和相邻行内元素在一行上。

高、宽无效，但水平方向的padding和margin可以设置，垂直方向的无效。

默认宽度就是它本身内容的宽度。

行内元素只能容纳文本或则其他行内元素。（a特殊）  
注意：

只有 文字才 能组成段落 因此 p 里面不能放块级元素，同理还有这些标签h1,h2,h3,h4,h5,h6,dt，他们都是文字类块级标签，里面不能放其他块级元素。

链接里面不能再放链接。

行内块元素（inline-block）

在行内元素中有几个特殊的标签——<img />、<input />、<td>，可以对它们设置宽高和对齐属性，有些资料可能会称它们为行内块元素。

行内块元素的特点：

和相邻行内元素（行内块）在一行上,但是之间会有空白缝隙。

默认宽度就是它本身内容的宽度。

高度，行高、外边距以及内边距都可以控制。

标签显示模式转换 display

块转行内：display:inline;

行内转块：display:block;

块、行内元素转换为行内块： display: inline-block;

CSS书写规范

空格规范

【强制】 选择器 与 { 之间必须包含空格。示例： .selector { }

【强制】 属性名 与之后的 : 之间不允许包含空格， : 与 属性值 之间必须包含空格。示例：font-size: 12px;

选择器规范

【强制】 当一个 rule 包含多个 selector 时，每个选择器声明必须独占一行。

/\* good \*/

.post,

.page,

.comment {

line-height: 1.5;

}

【建议】 选择器的嵌套层级应不大于 3 级，位置靠后的限定条件应尽可能精确。

/\* good \*/

#username input {}

.comment .avatar {}

属性规范

【强制】 属性定义必须另起一行。

/\* good \*/

.selector {

margin: 0;

padding: 0;

}

【强制】 属性定义后必须以分号结尾。

CSS 三大特性

层叠 继承 优先级 是学习CSS 必须掌握的三个特性。

CSS层叠性

所谓层叠性是指多种CSS样式的叠加。

是浏览器处理冲突的一个能力,如果一个属性通过两个相同选择器设置到同一个元素上，那么这个时候一个属性就会将另一个属性层叠掉

比如先给某个标签指定了内部文字颜色为红色，接着又指定了颜色为蓝色，此时出现一个标签指定了相同样式不同值的情况，这就是样式冲突。

一般情况下，如果出现样式冲突，则会按照CSS书写的顺序，以最后的样式为准。

样式冲突，遵循的原则是就近原则。 那个样式离着结构近，就执行那个样式。

样式不冲突，不会层叠

CSS继承性

所谓继承性是指书写CSS样式表时，子标签会继承父标签的某些样式，如文本颜色和字号。想要设置一个可继承的属性，只需将它应用于父元素即可。简单的理解就是： 子承父业。

恰当地使用继承可以简化代码，降低CSS样式的复杂性。子元素可以继承父元素的样式（text-，font-，line-这些元素开头的都可以继承，以及color属性）

CSS优先级

定义CSS样式时，经常出现两个或更多规则应用在同一元素上，这时就会出现优先级的问题。

在考虑权重时，还需要注意一些特殊的情况，具体如下：

继承样式的权重为0。即在嵌套结构中，不管父元素样式的权重多大，被子元素继承时，他的权重都为0，也就是说子元素定义的样式会覆盖继承来的样式。

行内样式优先。应用style属性的元素，其行内样式的权重非常高，可以理解为远大于100。总之，他拥有比上面提高的选择器都大的优先级。

权重相同时，CSS遵循就近原则。也就是说靠近元素的样式具有最大的优先级，或者说排在最后的样式优先级最大。

CSS定义了一个!important命令，该命令被赋予最大的优先级。也就是说不管权重如何以及样式位置的远近，!important都具有最大优先级。

CSS特殊性（Specificity）

关于CSS权重，我们需要一套计算公式来去计算，这个就是 CSS Specificity，称为CSS 特性或称非凡性，它是一个衡量CSS值优先级的一个标准 具体规范入如下：

specificity用一个四位的数 字串(CSS2是三位)来表示，更像四个级别，值从左到右，左面的最大，一级大于一级，数位之间没有进制，级别之间不可超越。

| 继承或者\* 的贡献值 | 0,0,0,0 |
| --- | --- |
| 每个元素（标签）贡献值为 | 0,0,0,1 |
| 每个类，伪类贡献值为 | 0,0,1,0 |
| 每个ID贡献值为 | 0,1,0,0 |
| 每个行内样式贡献值 | 1,0,0,0 |
| 每个!important贡献值 | ∞ 无穷大 |

权重是可以叠加的，比如下面的例子：

div ul li ------> 0,0,0,3

.nav ul li ------> 0,0,1,2

a:hover ----—> 0,0,1,1

.nav a ------> 0,0,1,1

#nav p ------> 0,1,0,1

注意：  
数位之间没有进制 比如说： 0,0,0,5 + 0,0,0,5 =0,0,0,10 而不是 0,0, 1, 0， 所以不会存在10个div能赶上一个类选择器的情况。继承的 权重是 0

总结优先级：  
1.使用了 !important声明的规则。  
2.内嵌在 HTML 元素的 style属性里面的声明。  
3.使用了 ID 选择器的规则。  
4.使用了类选择器、属性选择器、伪元素和伪类选择器的规则。  
5.使用了元素选择器的规则。  
6.只包含一个通用选择器的规则。  
7.同一类选择器则遵循就近原则。

总结：权重是优先级的算法，层叠是优先级的表现

CSS 背景(background)

CSS 可以添加背景颜色和背景图片，以及来进行图片设置。

| background-color | 背景颜色 |
| --- | --- |
| background-image | 背景图片地址 |
| background-repeat | 是否平铺 |
| background-position | 背景位置 |
| background-attachment | 背景固定还是滚动 |
| 背景的合写（复合属性） |  |
| background:背景颜色 背景图片地址 背景平铺 背景滚动 背景位置 |  |

背景图片(image)

background-image : none | url (url)

参数：  
none : 　无背景图（默认的）  
url : 　使用绝对或相对地址指定背景图像，url一般不要加引号。

background-image 属性允许指定一个图片展示在背景中（只有CSS3才可以多背景）可以和 background-color 连用。 如果图片不重复地话，图片覆盖不到地地方都会被背景色填充。 如果有背景图片平铺，则会覆盖背景颜色。

背景平铺（repeat）

background-repeat : repeat | no-repeat | repeat-x | repeat-y

参数：  
repeat : 　背景图像在纵向和横向上平铺（默认的）  
no-repeat : 　背景图像不平铺  
repeat-x : 　背景图像在横向上平铺  
repeat-y : 　背景图像在纵向平铺

背景位置(position)

background-position : length || length

background-position : position || position

参数：  
length : 　百分数 | 由浮点数字和单位标识符组成的长度值。请参阅长度单位  
position : 　top | center | bottom | left | center | right

说明：  
设置或检索对象的背景图像位置。必须先指定background-image属性。默认值为：(0% 0%)。  
如果只指定了一个值，该值将用于横坐标。纵坐标将默认为50%。第二个值将用于纵坐标。

注意：  
1.position 后面是x坐标和y坐标。 可以使用方位名词或者 精确单位。  
2.如果和精确单位和方位名字混合使用，则必须是x坐标在前，y坐标后面。比如 background-position: 15px top; 则 15px 一定是 x坐标 top是 y坐标。

背景附着

background-attachment : scroll | fixed

参数：  
scroll : 　背景图像是随对象内容滚动  
fixed : 　背景图像固定  
设置或检索背景图像是随对象内容滚动还是固定的。

背景简写

background属性的值的书写顺序官方并没有强制标准的。为了可读性，建议如下写：

background:背景颜色 背景图片地址 背景平铺 背景滚动 背景位置

background: color url(image.jpg) repeat-y scroll 50% 0 ;