<i>斜体标记</i>

<b>粗体标记</b>

<u>下划线</u>

<s>删除线</s>

<strike>也是删除线</strike>

上标标记：10<sup>3</sup>

下标标记：h<sub>2</sub>o



<em>强调文字</em>

<strong>字体加重</strong>

<code>显示编程代码</code>

<samp>显示示例文字</samp>

<kbd>显示键盘按键文字</kbd>

<small>缩小文字</small>

<big>放大文字</big>

不换行控制标记：<nobr></nobr>

只要文本在标记里面，不管有多长，都在一行里。(除非加入换行符<br>)

对齐标记：align=”#”，参数为left，right，center。

<center>居中</center>也可以。

设置文字滚动标签：<marquee></marquee>

默认从右到左滚动

scrollamount：用于设置内容滚动速度

behavior：设置内容滚动方式，

|  |  |
| --- | --- |
| 默认为scroll | 循环滚动 |
| alternate | 内容将来回滚动 |
| slide | 内容滚动一次即停止，不循环 |
| loop | 默认无限制次数，可设置循环次数 |

scrolldellay：设置内容滚动时间间隔

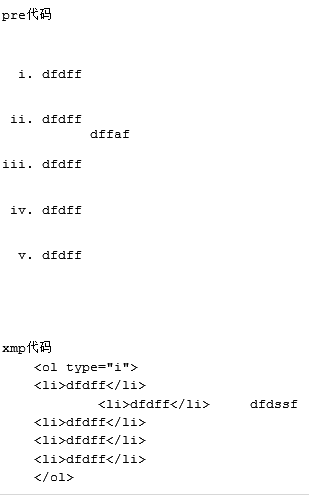
bgcolor：设置内容滚动背景色

direction：设置内容滚动方向，up, right, down, left

欲格式化文本：在网页中输出的是代码中的格式。

<pre></pre>会解释HTML代码，并且输入文本里的格式，输出空格，换行等。

<xmp></xmp>把HTML代码看做文本



图像与文本对齐

<img src=”xxx.jpg” alt=”风景” align=”top”>图像与文本顶端对齐

默认align=”bottom”，设置对齐底部。

还可以设置top，right，left，middle。

超链接网页的打开方式target参数

\_self：在同一框架或窗口中打开所链接的文档，默认参数。

\_blank：在新浏览器窗口中打开网页，比较常用。

\_parent：将要链接的文件载入含有该链接框架的父框架集或父窗口中，如果含有该链接的框架不是嵌套的，则在浏览器全屏窗口中载入链接的文件，类似\_self。

\_top：在当前的整个浏览器窗口中打开所链接的文档，因此会删除所有框架。

**链接的注释**

<a href="http://www.baidu.com" title="zz">accelworld</a>

**瞄点链接：**在其名前加上“#”，一是设定name的名称，二是设置一个href指向这个name。

<a name=”html”>瞄点名称</a>

<a href=”#html”>点击到瞄点位置</a>

Name的值可以随便设。

**跨行用rowspan，跨列用colspan**

<table class="b" border="1px">

<tr>

<th colspan="2">表头1</th>

</tr>

<tr>

<td>表元2</td>

<td>表元2</td>

</tr>

</table>

**表格在网页对齐：<table align=”#”></table>**

**表格内文字对齐：<tr align=”#”></table>**

**多媒体标记embed**

设置自动播放：autostart=true|false，默认自动播放为true。

设置循环播放：loop=true|false，等于数值则播放多少次。

设置控制面板隐藏：hidden=true|false，默认不隐藏。

当使用float设置后的层，在接下来用到的层受空间漂移后的影响，也会出现漂移情况，可以使用属性clear来设置对象不往某方向漂移。

**Span：**span的宽度依照包围着的对象的宽度而定，故不能用宽度属性width设置span标记内对象的宽度，但可以设置span的margin(外边距)值，设置span中的对象位置与父对象的距离。

Div标记块级元素，sapn标记指行内元素，主要用于文本的容器。

一个div内容不超过一行，所占的就是一行文本内容，因为div默认宽度是父对象的100%，而span标记中的内容宽度是指其内容所占的宽度，即不能设置span的宽度，所以在一行文本中，想对其中的文字设置样式，而又不使该行文本换行的条件下，可以选择span标记。

实例：

<style type="text/css">

#span1{

color:blue;

}

#span2{

font-size:15px;

font-weight:600px;

}

</style>

</head>

<body>

<span id="span1">HTML+CSS</span>完全<span id="span2">自学</span>手册

</body>

**伪类：**是一组基于预定义性质的选择符。

link：用在对象处于的链接状态上。

visited：用在已经访问过的链接上。

active：用在获得焦点(比如被单击)的链接上。

hover：用于鼠标光标置于其上的链接。

**字体的复合属性**

font-style：normal(正常)，italic(斜体)，oblique(倾斜)

font-variant：normal(正常)，small-caps(小型大写字母)

font-weight：

font-size:

font-family:

行距：line-height

**字母大小写转换** text-transform：

|  |  |
| --- | --- |
| capitalize | 使每个词的首字母大写 |
| uppercase | 使每个词的所有字母大写 |
| lowercase | 使每个词的所有字母小写 |
| none | 以正常形式显示 |

**文本修饰** text-decoration:

|  |  |
| --- | --- |
| underline | 给文字加下划线 |
| overline | 给文字加上划线 |
| line-through | 给文字加删除线 |
| blink | 文字在闪烁 |
| none | 使原本有文本修饰的文字回复默认值 |

white-space：空格处理方式

|  |  |
| --- | --- |
| normal | 默认处理方式 |
| pre | 用等宽字体显示预先格式化的文本，但并不合并字间的空白距离和进行两端对齐 |
| nowrap | 强制在同一行内显示所有文本，直达文本结束或遭遇br对象 |

实例：

<style type="text/css">

#ws1

{

width: 200px;

border: 1px solid #000000;

white-space: normal;

}

#ws2

{

width: 200px;

border: 1px solid #000000;

white-space: pre;

}

#ws3

{

width: 150px;

border: 1px solid #000000;

white-space: nowrap;

padding: 10px;

}

</style>

</head>

<body>

<div id="ws1">

html+css完全自学手册html+css完全自学手册

</div>

<div id="ws2">

html+css完全自学手册

以下是本对象的css代码：

#ws2

{

width: 200px;

border: 1px solid #000000;

white-space: pre;

}

</div>

<div id="ws3">

html+css完全自学手册html+css完全自学手册

</div>

疑惑：white-space：pre和格式化文本<pre>有什么区别，感觉还不够后者好用。

HTML中，input标签默认是有边框（border）和背景色属性的。可以在CSS中加入border:none;（或者在html中加入style："border:none"）可以去掉边框。input标签在鼠标点击时，会有黄色的边框。如果不需要，可以加input{outline:medium;}属性去掉。

**vertical-align：文本垂直对齐**

**垂直对齐属性可以作用于一个内部元素的纵向位置，相对于它的上级元素或相对于元素行(一个内部元素是没有行在其前后断开的)，它的值可以是一个相对于元素行高属性的百分比，它会在上级基线上增高基线的指定数量。**

默认底部对齐，允许使用负值，也可以是一个方向关键字。

|  |  |
| --- | --- |
| top | 使元素和行中最高的元素向上对齐 |
| middle | 纵向对齐元素基线加上上级元素的x-高度 |
| bottom | 使元素和行中最高的元素向下对齐 |
| super | 使对象以上标的形式显示 |
| sub | 使对象以下标的形式显示 |
| ?text-top | 使元素与上级元素的字体向上对齐 |
| ?text-bottom | 使元素与上级元素的字体向下对齐 |
| ?baseline | 使元素与上级元素的基线对齐 |

text-align：文本水平对齐

参数：left、right、center、justify(左右对齐，貌似不常用…)

text-indent：文本缩进，可设长度与百分比两种。

背景颜色：background-color：transparent | color

前者表示背景色透明，后者直接接颜色值。

background-position：设置背景图像的位置

background-position：length|length

background-position：position|position

第一种表示后面可接百分数或者浮点数字和单位标识符组成的长度值，可写两个值，分别表示X轴和Y轴，只有一个值则表示x轴，y轴默认50%。

第二种表示可接top, center ,bottom, left , right。

<style type="text/css">

.box{

height: 70px;

width: 200px;

margin-top: 10px;

border: 1px solid red;

background-image: url("1.jpg");

background-repeat: no-repeat;

background-size: 100px 50px;

}

#id1{

background-position: top right;

}

#id2{

background-position: left 50%;

}

#id3{

background-position: 30px 20%;

}

</style>

</head>

<body>

<div class="box" id="id1"></div>

<div class="box" id="id2"></div>

<div class="box" id="id3"></div>

</body>

class和id一起用，还是第一次见，这就是发现新的用法的雀跃感？

background-attachment：设置背景图像滚动

scroll为默认值，表示图像因内容滚动而滚动

fixed是固定的，表示背景图像固定在网页中。

<style type="text/css">

#id1{

width: 160px;

height: 100px;

color: #ffffff;

background: #ccc url(1.jpg) no-repeat;

background-attachment: fixed;

}

</style>

</head>

<body>

<textarea id="id1">

<span>加速世界1</span>

21313213<br>

21313sdfdsfd213<br>

21313dfsdf213<br>

<span>加速世界1</span>

</textarea>

难道是因为文本域有上下拖动条？

设置背景图像的铺排：background-repeat

repeat：默认值，背景图像在纵向和横向上平铺。

no-repeat：背景图像不平铺。

repeat-x：背景图像仅在横向上平铺。

repeat-y：背景图像仅在纵向上平铺。

overflow-y 属性规定是否对内容的上/下边缘进行裁剪 - 如果溢出元素内容区域的话。

overflow-y: visible|hidden|scroll|auto|no-display|no-content;

|  |  |
| --- | --- |
| **值** | **描述** |
| visible | 不裁剪内容，可能会显示在内容框之外。 |
| hidden | 裁剪内容 - 不提供滚动机制。 |
| scroll | 裁剪内容 - 提供滚动机制。 |
| auto | 如果溢出框，则应该提供滚动机制。 |
| no-display | 如果内容不适合内容框，则删除整个框。 |
| no-content | 如果内容不适合内容框，则隐藏整个内容。 |

**position：定位**

static 默认值

absolute：将对象从文档流中从流出，使用方位等属性表示最接近的一个父对象的绝对定位的方位。

会将对象脱出正常的文档流的绝对定位，而不考虑它周围内容的布局。

此时对象不具有margin，但仍有padding和border。

relative：对象不可层叠，但依据方位等属性在正常文档流中偏移位置。

它的位置会根据前一个对象进行偏移，在相对定位对象之后的文本或对象占有它们自己的空间而不会覆盖被定位对象的自然空间。

z-index：层叠对象

数值越大，越往前端显示。

overflow：表示内容超过其指定大小时该如何管理

visibility：表示的对象在网页中是存在一个空间的，不管它是否显示。

display：表示的对象是在网页中显示才有空间存在，隐藏则消失。

伪类

link：链接颜色

hover：鼠标置于其上的状态

active：获得焦点的状态

visited：访问过后的颜色