

网页常用单词代码：(书 p23)

Left 设置标题文字左对齐 (书 p28)
Right 设置标题文字右对齐
Center 设置标题文字居中对齐
如：<h2 align="right">2 级标签右对齐</h2>

水平线标签<hr /> (书 p29)

属性名	含义
align	设置对齐方式
size	设置水平线粗细
color	设置颜色
width	设置宽度

如：<hr color="red">

换行标签
 (强制换行)

文本样式标签 (书 p31)
文本内容

属性名	含义
face	设置文字字体
size	设置字号大小, 1~7
color	设置文字颜色

如：<p>
文本</p>

文本格式化标签 (书 p32)

标签	效果
 	文字以粗体方式显示
<u></u> <ins></ins>	文字加下画线显示
<i></i> 	文字斜体显示
<s></s> 	文字加删除线显示

文本语义标签 (书 p33)

标签	含义
time	定义时间或日期 可以为 24 小时某一时间
datetime	定义相应的时间日期 不定义时由文本定义
pubdate	定义发布日期 一般取值为 pubdate

如：<p>我们早上<time>9.00</time>上班</p>
我们<time datetime="2015-1-1">1 号</time>去旅游
<time datetime="2015-1-1" pubdate =" pubdate">
本消息发布于 2015 年 1 月 1 日

<mark>标签——文本显示高亮
(书 p35)

特殊字符	描述	字符对应代码
	空格符	
<	小于号	<
>	大于号	>
&	和号	&
¥	人民币	¥
©	版权	©
®	注册商标	®
°	摄氏度	°
±	正负号	±
x	乘号	×
÷	除号	÷
A ²	平方 2 (上标)	²
A ³	立方 3 (上标)	³

图像标签 (书 p36)

属性	属性值	描述
src	url	图像的路径;
alt	文本	图像不能显示时替换文本
title	文本	指针悬浮显示时显示文本
width	px	图像宽度
height	px	图像高度;
border	数字	图像边框宽度
vspace	px	图像顶部和底部空白 (垂直边距)
hspace	px	图像左侧和右侧空白 (水平边距)
align	left	图像左对齐
	right	图像右对齐
	top	图像顶部和文本第一行文字对齐, 其他文字居图像下方
	middle	图像水平中线和文本第一行文字对齐, 其他文字居图像下方
	bottom	图像底部和文本第一行文字对齐, 其他文字居图像下方

实际中不建议直接使用 img 标签

CSS 样式 (书 p44)

1.内嵌式

<style type="text/css"></style>

2.外链式

<head>

<link href="CSS 文件路径" type="text/css" rel="stylesheet" />

</head>注意位于头部标签中

CSS 基础选择器 (书 p49)

1.标签选择器 (优先级第三) (为 1)

P{属性 1: 属性值 1; 属性 2: 属性值 2; }

2.类选择器 (优先级第二) (为 10)

.类名{属性 1: 属性值 1; 属性 2: 属性值 2; }

3.id 选择器 (优先级第一) (唯一性) (为 100)

#id 名{属性 1: 属性值 1; 属性 2: 属性值 2; }

4.通配符选择器 (优先级第四)

*{属性 1: 属性值 1; 属性 2: 属性值 2; }

文本样式

CSS 字体样式属性 (书 p52)

font-size 字号大小

相对长度单位	说明
em	倍率, 相对于当前对象文本的字体尺寸
px	像素
绝对长度单位	说明
in	英寸
cm	厘米
mm	毫米
pt	点

Font-family 字体

注: 1.各种字体之间必须用英语逗号隔开

2.中文字体需要加英文的引号

Font-weight 字体粗细

值	描述
normal	默认值
bold	定义粗体字符
bolder	定义更粗字符
lighter	定义更细字符
100~900	定义由细到粗字符 400 为 normal, 700 为 bold 值越大字体越粗

Font-style 字体风格

normal	默认值
oblique	倾斜
italic	斜体 (更常使用)

设置综合字体样式

顺序: p{

1) 字体风格 font-style:normal;

2) 字体粗细 font-weight:bold;

3) 字体大小/行高 font-size:30px; line-height:40px;

4) 字体 font-family:Arial,"宋体";

P{ font: normal bold 30px/40px Arial,"宋体" }

@font-face 定义服务器字体名称

@font-face{ font-family: 自己定义;

Src: url 字体路径; }

P{ font-family: 定义的字体名称; }

CSS 文本外观属性 (书 p55)

Color 文本颜色

颜色	十六进制	Rgb
红色 red	#FF0000	255, 0, 0
绿色 green	#008000	0, 128, 0
白色 white	#FFFFFF	255, 255, 255
黑色 black	#000000	0, 0, 0
黄色 yellow	#FFFF00	255, 255, 0
灰色 gray	#808080	128, 128, 128
粉红色 pink	#FFC0CB	255, 192, 203
银色 silver	#C0C0C0	192, 192, 192
褐色 brown	#A52A2A	139, 35, 35
蓝色 blue	#0000FF	0, 0, 255
紫色 purple	#800080	128, 0, 128
金色 gold	#FFD700	255, 215, 0
橙色 orange	#FFA500	255, 165, 0
水绿色 aqua	#00FFFF	212, 242, 231

Letter-spacing: 字间距

Word-spacing: 单词间距

Line-height: 行间距

Text-transform: 文本转换

默认值 none

首字母大写·capitalize

全部字符转换为大写 uppercase

全部字符转换为小写 lowercase

Text-decoration: 文本装饰

没有装饰 (默认值) none

下画线 underline

上画线 overline

删除线 line-through

Text-align: 水平对齐方式

左对齐 (默认值) left

右对齐 right

居中对齐 center

Text-indent: 首行缩进

(em/px/% 允许使用负值)

仅适用于块级元素, 行内元素无效

White-space: 空白符处理

默认值 normal

保留空格, 空行, 原样显示 pre

强制文本不能换行 nowrap

Text-shadow: 阴影效果

(水平和垂直距离可以为负值)

选择器{text-shadow: 水平阴影距离 垂直阴影距离
模糊半径 阴影颜色; }

(可以多个阴影, 中间用逗号隔开)

{ text-shadow:10px 10px 10px red,20px 20px 20px blue;}

Text-overflow: 表示对象溢出文本

clip 修剪溢出文本, 不显示省略标签...

ellipsis 用省略标签代替被修剪文本

(overflow: hidden 修剪溢出文本)

Word-wrap

Normal 默认允许断字点换行

Break-word 在长单词或 url 地址内部换行

CSS 复合选择器 (书 p64)

1. 标签指定式选择器 (两个选择器之间 **不能** 有空格)

P.head{color: red}

则只会影响<p class="head">标签

不会影响其他 head 类标签

2. 后代选择器 (中间 **必须** 用空格隔开)

P strong{ color: red}

当 p 标签嵌套 strong 标签时, 就可以对 p 中的 strong 进行控制

P strong em{ color: red } (不限于 2 个元素)

控制 strong 中的 em

3. 并集选择器 (通过 **逗号** 连接)

如 h2 和 h3 想要蓝色, h2 和 h4 加下画线

h2,h3{color: blue;}

h2,h4{ text-decoration: underline;}

应用 style 的元素拥有巨大的优先级

P{ h4 style="color: red" }

Important 最大优先级

#header{ color:red !important }

CSS 属性选择器 (书 p74)

1. E[att^=value]选择器

P[id="one"] p 元素中 id 属性以 one 开头就会被选中

2. E[att\$=value]选择器

P[id="one"] p 元素中 id 属性以 one 结尾就会被选中

3. E[att*=value]选择器

P[id="one"] p 元素中 id 属性包含 one 就会被选中

CSS 关系选择器 (书 p77)

1. 子元素选择器 (用于选择某一个元素的第一级子元素)

h1>strong h1 后面的 strong 元素生效

2. 临近兄弟选择器 (前后两个元素, 必须为 **紧跟** 元素)

h1+strong

h1 后面的 strong 元素生效, 第二个即使相同也不会生效

3. 普通兄弟选择器 (两个元素有同一个父亲, 不用紧跟)

h1~strong h1 后面的 strong 都会生效

结构化伪类选择器 (书 p80)

1. root 选择器 (对页面所有标签都生效)

: root{ background: red; }

不需要该样式时可以单独覆盖

2.: not 选择器

P*: not (h3) 对 p 中的 h3 不生效

3.: only-child 选择器

匹配某父元素的唯一一个子元素

Strong: only-child{color: red; }

当 p 标签中只有一个元素且为 strong 时, strong 生效

4.: first-child / : last-child 选择器

用于选择父元素中第一个/最后一个元素

5.:nth-child (n) /:nth-last-child (n) 选择器 (n 一般为 2)

用于选择某一个父元素中第 n 个/最后 n 个子元素

6.:nth-of-type(n) / :nth-last-of type 选择器

用于选择特定类型的第 n 个子元素和倒数第 n 个元素

奇数行 odd

偶数行 even

7:empty 选择器

用于选择没有子元素或者文本内容为空的所有元素

伪元素选择器 (书 p86)

: before 伪元素选择器 (在某元素之前插入内容)

: after 伪元素选择器 (在某元素之后插入内容)

(必须配合 content 元素使用)

P: before{content: url 图片路径; }

盒子模型 (书 p92)

1.边框样式设置 border-style (必要)

none	没有边框
solid	单实线边框
dashed	虚线边框
dotted	点线边框
double	双实线边框

2.边框宽度 border-width (默认为 1px)

3.边框颜色 border-color

注: 如使用百分比颜色时, 0 也要加%

4.综合设置边框 (顺时针顺序)

(4 个属性时: 上, 右, 下, 左)

(3 个属性时: 上, 左右, 下)

(2 个属性时: 上下, 左右)

(1 个属性时: 全生效)

Border-top-style 上边框样式

Border-right-color 右边框颜色

Border-bottom-width 下边框宽度

Border-left-style 左边框样式

P{ border: 5px solid red}

5.内边距 padding (可以%数值) (同上)

Padding-top 上内边距

Padding-bottom 下内边距

Padding-left 左内边距

Padding-right 右内边距

6.外边距 margin (可以为负值) (同上)

margin-left: 50px; margin-right: 40px; 等价于

margin{ 0 40px 0 50px; } (顺时针顺序)

为方便控制网页

*{margin: 0; padding: 0; border: 0; }

可用 height width 控制盒子的宽和高

.box{ height: 200px; width: 200px; }

背景属性 (书 p101)

1.背景颜色 background-color

2.背景图像 background-image: url

3.背景图像平铺 background-repeat

repeat (默认值)	沿水平和垂直平铺
no-repeat	不平铺(位于左上角, 仅显示一次)
repeat-x	水平方向平铺
repeat-y	垂直方向平铺

4.背景图像位置 background-position (可以 px 或%)

水平方向: left center right

垂直方向: top center bottom

5.背景图像固定 background-attachment

scroll (默认) 图像随页面一起滚动

fixed 图像固定于屏幕上, 不随页面滚动

6.综合设置 (中间空格隔开, 实际无顺序要求)

background:颜色 图像位置 平铺 水平位置 垂直位置 固定

新增盒子属性 (书 p106)

透明度

1.Rgb 模式 (透明度 0~1)

P{background-color: rgb (255, 0, 0, 0.5) }

0.5 指的为透明度 (0 完全透明) (1 完成不透明)

2.opacity 属性

opacity: 0.5;

圆角 border-radius (px 或%)

{水平半径参数 1 水平半径参数 2 水平半径参数 3 水平半径参数 4 /

垂直半径参数 1 垂直半径参数 2 垂直半径参数 3 垂直半径参数 4; }

(一个数值时为四个角半径相同)

(两个数值时为 左上角右下角 右上角左下角)

(三个数值时为 左上角 右上角 左下角 右下角)

(四个数值时为 左上角 右上角 右下角 左下角)

图片边框

属性	描述
border-image-source	图片路径
border-image-slice	图片上右下左偏移量 (裁切位置)
border-image-width	边框宽度
border-image-outset	边框背景向盒子外部延伸距离
border-image-repeat	图片平铺方式

border-image-repeat: stretch 图片填充方式为拉伸填充

阴影 box-shadow (书 p111)

参数	描述
h-shadow	水平阴影距离
v-shadow	垂直阴影距离
blur	阴影模糊半径
spread	阴影拓展半径
color	阴影颜色
outset/inset	外阴影/内阴影

{ 5px 5px 10px 2px #999 inset; }

渐变 (书 p112)

1.线性渐变

(沿着一条直线过渡到结束颜色)

background-image:linear-gradient

(渐变角度, 颜色值 1, 颜色值 2,);

注: 可在颜色值后面标注百分比,

用于标示颜色的位置(空格隔开)

(30deg,red, 50%,blue 80%) ;

重复线性渐变 Background-image:repeating-linear-gradient

渐变角度是水平线和渐变线之间的夹角 (deg 为单位)
或在后面加 to

to bottom	180deg	下方
to left	270deg	左方
to right	90deg	右方
to top	0deg	上方

径向渐变 background-image:radient-gradient

(渐变形状 圆心位置,颜色值 1,颜色值 2,……) ;

重复径向渐变 background-image:repeating-radient-gradient

(1) 渐变形状

circle:指定圆形的径向渐变

ellipse:指定椭圆形的径向渐变

50px 80px:水平 80px, 垂直半径 50px 的椭圆形

(2) 圆心位置 (确定元素渐变中心的位置)

使用 at 加上关键词或参数来确定渐变中心位置

参数为像素或百分比 (可以为负值)

(ellipse at center, red, blue)

(ellipse at 50% 90%, red, blue)

多背景图像 (书 p116)

background-image:url(图片 1), url(图片 2), url(图片 3),

修剪背景图像 background-size

(属性值 1 宽度, 属性值 2 高度)

px, %: 只设置一个则第二个默认为 auto

cover: 把背景图像扩展足够大

contain: 扩展到最大尺寸, 完成适合内容区域

设置背景的显示区域 background-origin

padding-box:背景图像相对于内边距区域来定位

border-box: 背景图像相对于边框来定位

content-box: 背景图像相对于内容框来定位

设置背景图像的裁剪区域 background-clip

padding-box: 从内边距向外裁剪背景

border-box: 默认, 从边框区域向外裁剪背景

content-box: 从内容区域向外裁剪背景

元素类型 (书 p120)

块元素 (独自占 1 行或多行, 可设置宽度, 高度, 对齐等)

<h1>~<h6>, <p>, <div>, , , 等

行内元素 (一般不占独立的区域)

<i><s><ins><u><a>等

为范围, 可控制内容

<div>内容<div>

元素类型的转换 display

(行内元素需要设置宽度和高度

或块元素需要拥有某些行内元素的特性)

inline	此元素显示为行内元素
block	此元素显示为块元素
Inline-block	此元素显示为行内快元素， 可以设置宽高对齐属性，不会独占一行
none	此元素将被隐藏，不显示页不占页面空间

列表标签 (书 p131)

无序列表

(是不分顺序的列表，没有顺序级别之分，内部可以嵌套多个标签)

每队 至少嵌套一对标签

type 属性用于指定项目符号

type 属性值	显示效果
dise 默认值	●
circle	○
square	■

有序列表

每对标签至少嵌套一对标签

属性	属性值 (类型)	描述
type	1 默认	项目符号显示为数字 1,2,3
	A 或 a	项目符号显示为 ABC abc
	I 或 i	项目符号显示为 I ii iii
start	数字	规定项目符号的起始值
value	数字	规定项目符号的数字

定义列表<dl>

包含三个列表<dl><dt><dd>

三个标签之间不允许出现其他标签

<dl>标签必须和<dt>标签相连

<dl>

<dt>内容</dt>

<dd>内容 1</dd>

<dd>内容 2</dd>

<dd>内容 3</dd></dl>

CSS 控制列表样式

list-style-type 属性 ul{List-style-type: disc}

属性值	描述	属性值	描述
disc	实心圆 (无序)	none	不使用
circle	空心圆 (无序)	Cjk-ideographic	简单的表意数字
square	实心方块 (无序)	georgian	传统的乔治尼亚编号
decimal	阿拉伯数字	decimal-leading-zero	以 0 开头的阿拉伯数字
lower-roman	小写罗马数字	upper-roman	大写罗马数字
lower-alpha	小写英文字母	upper-alpha	大写英文字母
lower-latin	小写拉丁字母	upper-latin	大写拉丁字母
hebrew	传统希伯来编号方式	armenian	传统的亚美尼亚编号

list-style-image 属性 (引入图片项目符号)

则为 ul 引入，li 使用

list-style-position 属性 (控制项目符号位置)

inside: 列表项目符号位于文本以内

outside: 列表项目符号位于文本以外 (默认值)

list-style 属性 (综合设置列表样式属性)

list-style:列表项目符号 列表项目符号的位置 列表项目图像;

超链接 (书 p139)

文本或
图像(可设置 border="0",清除图像边框效果)

href: 链接目标的地址

Target: 指定页面的打开方式

_self: 默认值，在原窗口打开

_blank: 在新窗口中打开

锚点链接: 直接跳转到指定位置

美美

<h3 id="one">美美</h3>

链接伪类控制超链接 (按表格从上到下顺序设置)

超链接标签 a 的伪类	描述
a:link{CSS 样式规则}	超链接默认样式
a:visited{CSS 样式规则}	超链接被访问后样式
a:hover{CSS 样式规则}	鼠标指针经过，悬停样式
a:active{CSS 样式规则}	鼠标点击不放时样式

a:link , a:visited{color:red}

设置默认和被访问后的颜色为红色

表格和表单（书 p148）

<table><tr><td></td></tr></table>

<table>用于定义表格的开始和结束

<tr>代表行，有几个 tr 就有几行

<td>定义单元格 tr 中有几个就有几列

， <th>头部标签

<table>标签属性 <table border="20" height="400">

属性	描述	常用属性值(默认)
border	表格边框，默认为 0	px
cellspacing	单元格与单元格之间空间	px (2)
cellpadding	单元格内容与单元格边缘空间	px (1)
width	表格宽度	px
height	表格高度	px
align	表格网页中水平对齐方式	left,center,right
bgcolor	表格背景颜色	rgb
background	表格背景图像	url 地址

<tr>标签属性

属性	描述	常用属性值
height	设置行高度	px
align	一行内容水平对齐方式	left,center,right
valign	一行内容垂直对齐方式	top,middle,bottom
bgcolor	设置行背景颜色	rgb
background	设置行背景图像	url 地址

<td>标签属性

属性名	含义	常用属性值
width	单元格宽度	px
height	单元格高度	px
align	单元格内容水平对齐方式	left,center,right
valign	单元格内容垂直对齐方式	top,middle,bottom
bgcolor	单元格背景颜色	rgb
background	单元格背景图像	url 地址
colspan	单元格横跨的列数 用于合并水平方向单元格	正整数
rowspan	单元格竖跨的行数 用于合并垂直方向单元格	正整数

表格的名称<caption></>（书 p157）

<thead></>定义表格的头部

（包含网页 logo，导航头部信息）

<tfoot></>定义表格的尾部

（位于 thead 之后）

<tbody></>定义表格主体内容（位于 thead,tfoot 之后）

CSS 样式表格控制（书 p158）

border-collapse: collapse 去除单元格与边框之间的空间

<th><td>标签（tr 标签无内外边距属性）

可以设置内边距 padding 属性，外边距 margin 属性无效

可为单元格单独设置宽高（同一行/列取决于其中较大者）

表单（书 p163）

创建表单

<form action="url 地址" method="提交方式" name="表单名称"></form>

url 地址：可以为相对路径或绝对路径，或邮箱地址

method: get（默认）提交少量数据，保密性差，浏览器与表单服务器建立链接

post: 分段发送，可以提交大量数据

表单控件<input type="控件类型" />

属性	属性值	描述
type	text	单行文本输入框
	password	密码输入框
	radio	单选按钮
	checkbox	复选框
	button	普通按钮
	submit	提交按钮
	reset	重置按钮
	image	图像形式的提交按钮
	hidden	隐藏域
	file	文件域
name	自定义	控件的名称
value	自定义	Input 控件中默认文本值
size	正整数	Input 控件在页面显示宽度
readonly	readonly	该控件内容为只读 (不能编辑修改)
disabled	disabled	第一次加载时禁用该控件 (默认显示为灰色)
checked	checked	定义选择控件默认被选中的项
maxlength	正整数	控件允许输入的最多字符数

注：单选按钮必须有相同的 name 值才会生效

提交通常配合 value 使用*

textarea 控件，创建多行文本（书 p167）

<Textarea cols="每行中的字符数" rows="显示的行数">

文本内容</>

当 type 属性为 text 时可用

| 其他属性 | 属性值 | 描述 |
|----------|----------|--------------------------|
| name | 自定义 | 控件的名称 |
| readonly | readonly | 该控件内容为只读
(不能编辑修改) |
| disabled | disabled | 第一次加载时禁用该控件
(默认显示为灰色) |

select 控件，包含多个下拉菜单

<select>

<option>选项 1</>

<option>选项 2</>

<option>选项 3</></>

| 标签名 | 常用属性值 | 描述 |
|----------|----------|---|
| <select> | size | 下拉菜单可见数（正整数） |
| | multiple | 定义 Multiple="Multiple"时
下拉菜单具有多项选择功能
(按住 ctrl 键的同时选择多项) |
| <option> | selected | 定义 selected="selected"时
该项为默认选中项 |

全新的 input 控制类型（书 p173）

1.email 类型<input type="email" />

不符合 Email 格式，将提示错误信息

2.url 类型<input type="url" />

输入值不符合 url 地址格式，不允许提交

3.tel 类型<input type="tel" />

提供输入电话号码的文本框，与 pattern 属性配合使用

4.search 类型<input type="search" />

用户输入内容后，会记录，出现一个可以删除的图标

5.color 类型<input type="color" />

输入颜色，可通过 value 属性设置默认颜色

6.number 类型<input type="number" />

会检查输入数值是否正确

| | |
|-------|----------------|
| Value | 指定输入框默认值 |
| Max | 指定输入框可以接受最大输入值 |
| Min | 指定输入框可以接受最小输入值 |
| Step | 输入域合法间隔，默认为 1 |

7.range 类型<input type="range" />

提供一定范围内的输入值

| | |
|------|-------|
| max | 最大值 |
| min | 最小值 |
| step | 滑动的步幅 |

8.date pickers 类型<input type="Date pickers" />

提供时间日期类型输入

| 时间/日期类型 | 说明 |
|----------------|-------------------|
| date | 选取日，月，年 |
| month | 选取月，年 |
| week | 选取周，年 |
| time | 选择时间（小时和分钟） |
| datetime | 选取时间日，月，年（utc 时间） |
| datetime-local | 选取时间日，月，年（本地时间） |

全新 input 属性（书 p177 页）

autofocus 属性

指定页面加载后是否自动获取焦点

form 属性

可以把表单内的子元素写在页面的任意位置

form 属性适合所有 input 输入类型，

使用时只需引用表单所属 id 即可

<form action="#" method="post" id="one">

</form>

<input type="text" form="one"/>

list 属性

datalist 属性实现数据列表的下拉效果，则 list 属性能指定输入框所绑定的 datalist 元素，其值为 datalist 的某个 id

<input type="url" list="one"/>

<datalist id="one">

<option label="百度" value="http"></>

<option label="搜狗" value="http"></>

<option label="京东" value="http"></>

</></>

multiple 属性

可向文本框输入多个邮箱等，用逗号隔开，表示可以选择多个文件

min max step 属性

适用于 date pickers number range 标签

2.相对定位 (p202)

选择器{position: relative; }

可以通过边偏移属性改变标签的位置

3.绝对定位

选择器{position: absolute; }

会依据浏览器窗口进行定位

不论父标签位置如何偏移,

子标签相对于父标签的位置都不会改变

当 left 和 right 标签冲突时, 以 left 为准

当 top 和 bottom 标签冲突时, 以 top 为准

4.固定定位

选择器{position: fixed; }

不管窗口如何变化,

标签始终会显示在浏览器窗口的固定位置

布局的其他属性 (p203)

1.overflow 属性

选择器{overflow: 属性值; }

用于规定溢出内容的显示状态

| 属性值 | 描述 |
|---------|----------------------|
| visible | 内容不会修剪, 会呈现标签框之外(默认) |
| hidden | 溢出内容修剪, 修剪内容不可见 |
| auto | 需要时出现滚动条, 自适应显示内容 |
| scroll | 溢出内容修剪, 滚动条始终可见 |

2.z-index 属性

标签重叠时可以设置

属性取值为正整数, 负整数, 和 0 (默认)

属性取值越大, 设置属性标签在层叠标签中越居上

布局类型 (p206)

1.单列布局

<div id="top">头部</div>

<div id="nav">导航栏</div>

<div id="banner">焦点图</div>

<div id="content">内容</div>

<div id="footer">页面底部</div>

margin: 5px auto 标签上下外边距均为 5px;

且位于水平居中位置

2.两列布局

<div id="top">头部</div>

<div id="nav">导航栏</div>

<div id="banner">焦点图</div>

<div id="content">

<div class="content_left">内容左部分</div>

<div class="content_right">内容右部分</div>

</div>

<div id="footer">页面底部</div>

3.三列布局 (p209)

<div id="top">头部</div>

<div id="nav">导航栏</div>

<div id="banner">焦点图</div>

<div id="content">

<div class="content_left">内容左部分</div>

<div class="content_middle">内容中间部分</div>

<div class="content_right">内容右部分</div>

</div>

<div id="footer">页面底部</div>

全新 HTML5 元素 (p212)

1.header 标签<header></>

用于页面头部内容, 不限制 header 个数

2.nav 标签<nav></>

用于定义导航链接

3.footer 标签<footer></>

定义一个页面或区域的底部

4.article 标签<article></>

代表文档, 页面或程序与上下文不相关的独立部分

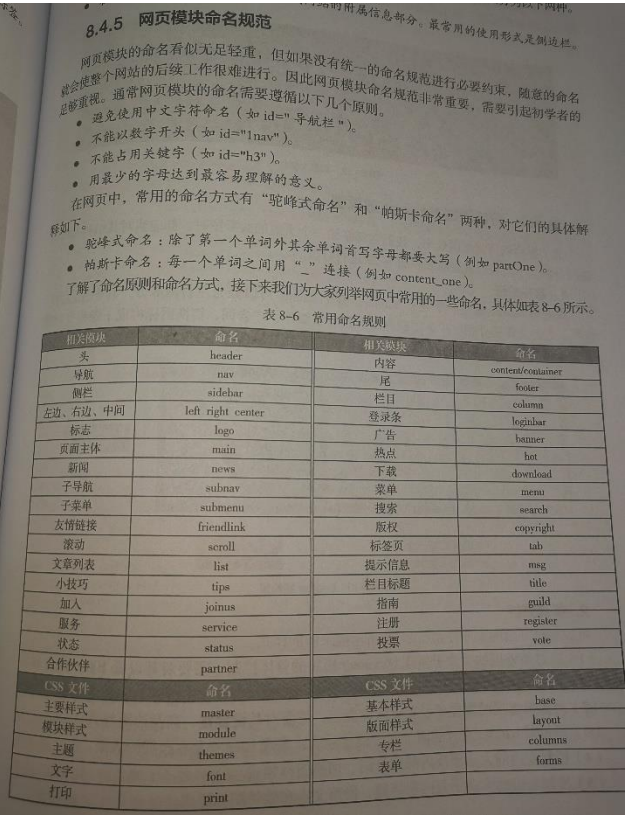
5.section 标签<section></>

表示一段专题性内容, 主要应用文章的章节中

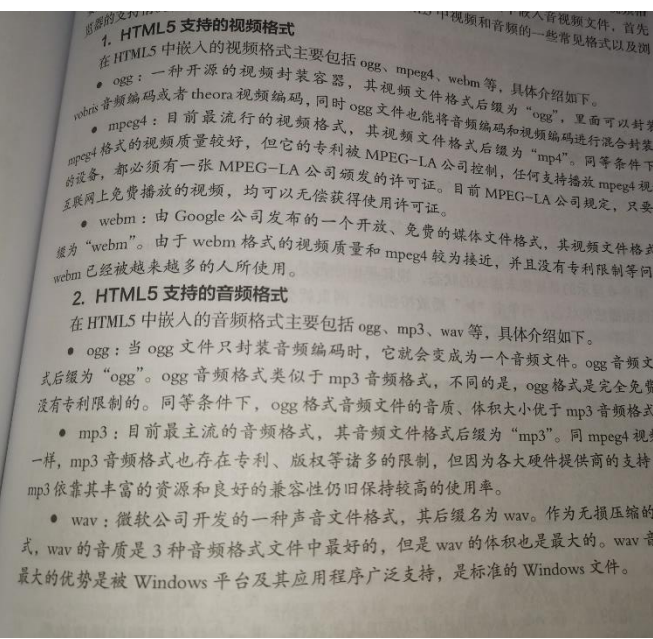
6.aside 标签<aside></>

定义当前页面或主要内容相关的引用等有别于主要内容部分

网页命名规范 (p215)



多媒体嵌入 (p223)



视频 Video 标签 (p224)

```
<video url="视频路径" controls="controls" >
</video>
```

用于定义视频文件支持 ogg, webm, mpeg4

| 属性 | 值 | 描述 |
|----------|----------------|-----------|
| autoplay | autoplay | 自动播放视频 |
| loop | loop | 自动重新播放 |
| preload | auto/meta/none | 预备播放 |
| poster | url | 视频播放失败时图像 |

可以直接使用 height width 为视频设置宽高

音频 audio 标签(p225)

```
<audio url="音频路径" controls="controls" >
</audio>
```

| 属性 | 值 | 描述 |
|----------|----------------|--------|
| autoplay | autoplay | 自动播放视频 |
| loop | loop | 自动重新播放 |
| preload | auto/meta/none | 预备播放 |

url 可以直接引用相应文件网址

过渡, 变形, 动画 (p235)

过渡

1.transition-property 属性

定义属性是否有过渡效果

| 属性值 | 描述 |
|---------------|--------------------------------|
| none | 没有属性获得过渡效果 |
| all | 所有属性都获得过渡效果 |
| property (名称) | 定义应用过渡属性 css 属性名称
多个名称用逗号分隔 |

transition-property: border-radius 变为圆形

2.transition-duration 属性

定义过渡效果持续时间

常用值为秒 (s)

3.transition-timing-function 属性

规定过渡效果的速度曲线

| 属性值 | 描述 |
|-----------------------|---|
| linear | 相同速度开始至结束的过渡效果
等同于 cubic-bezier(0, 0, 1, 1) |
| ease(默认) | 慢速开始, 然后加快, 慢慢结束的过渡效果
等同于 cubic-bezier(0.25, 0.1, 0.25, 1) |
| ease-in | 慢速开始, 逐渐加快 (0.42, 0, 1, 1) |
| ease-out | 慢速结束 (0, 0, 0.58 ,1) |
| ease-in-out | 慢速开始和结束 (0.42, 0.1, 0.58 ,1) |
| cubic-bezier(n,n,n,n) | 自定义, 值在 0~1 之间 |

4.transition-delay 属性

规定过渡效果的开始时间

5. transition 属性 (复合属性)

transition: property duration
timing-function delay;

变形 transform (p240)

1. 平移

transform: translate(x-value,y-value) ;

- (1)x 定义水平轴, y 定义垂直轴坐标
- (2)常用 px 和%为单位
- (3)当参数为负数时则向反方向移动 (向左和向上)
- 如果省略第二个坐标轴, 则默认值为 0, 不移动
- (4)单位不可省略, 否则不起作用

2. 缩放

transform: scale(x-value,y-value) ;

- (1)参数值可以为正值或负值和小数
不需要加单位
- (2)x 定义水平轴, y 定义垂直轴坐标
- (3)正数为放大元素, 负数翻转缩放元素
小于 1 的元素用于缩小元素

3. 倾斜

transform: skew(x-value,y-value) ;

- (1)x 定义水平轴, y 定义垂直轴坐标
- (2)参数值为角度值, 单位为 deg
可以为正值, 可以为负值

4. 翻转

transform: rotate(angle) ;

- (1)angle 表示要旋转的角度, 单位为 deg
- (2)角度为正值, 则按顺时针旋转;
角度为负值, 则按逆时针旋转

注: 如果一个元素需要设置多种变形效果,
中间用空格隔开。

5. 更改变换中心点

默认情况下元素的中心点在 x 轴和 y 轴的 50%位置

transform-origin: x-axis y-axis z-axis;

| 参数 | 描述 |
|--------|--|
| x-axis | 定义视图被置于 x 轴的何处。属性值可以为 em, %, px
也可以为 top, right, bottom, left, center |
| y-axis | 定义视图被置于 y 轴的何处。属性值可以为 em, %, px
也可以为 top, right, bottom, left, center |
| z-axis | 定义视图被置于 z 轴的何处。注意该值不能是一个百分比值, 否则会被示为无效值, 一般单位为 px |

3D 变形

1. rotateX ()

transform: rotateX (a)

CSS3 中, 用于指定元素围绕 x 轴旋转

- (1)a 用于定义旋转的角度值, 单位为 deg
- (2)角度为正值, 则围绕 x 轴顺时针旋转;
角度为负值, 则围绕 x 轴按逆时针旋转

2. rotateY ()

transform: rotateY (a)

CSS3 中, 用于指定元素围绕 y 轴旋转
(同上)

3. rotated3d ()

当想要同时设置 x 轴, y 轴旋转时可用

rotated3d (x, y, z, angle)

xyz 的取值可以取值 0 或 1, 当要沿着某轴移动则该轴设置为 1, 否则为 0, angle 为移动角度

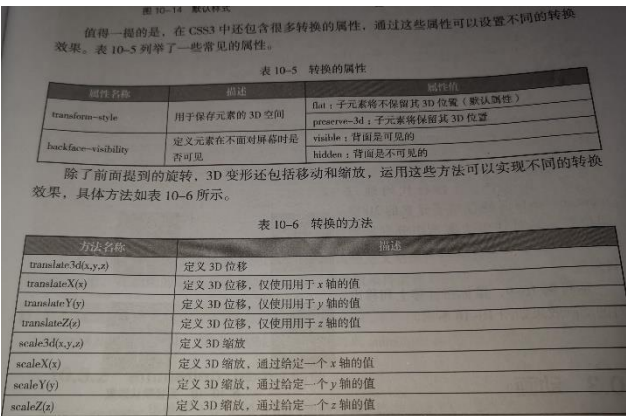
4. perspective 属性

可以理解为视距, 用于呈现良好的 3D 效果

perspective: 参数值;

参数值可以为 none 或数值, (一般为单位 px)

参数值越小, 透视效果越突出



动画(p249)

1.@keyframes 规则，用于创建规则

@keyframes animationname

```
{ keyframes-selector{css-styles;} }
```

(1)animationname:表示当前动画名称，不能为空

(2)keyframes-selector:关键帧选择器，即当前关键帧要应用到整个动画过程的位置，可以为%， from, to

(3)from 和 0%效果相同，表示动画开始

to 和 100%效果相同，表示动画结束

(4) css-styles:定义执行到当前关键帧的动画状态
多个属性之间用分号分隔，不能为空

@keyframes appear (动画名称)

```
{ 0% (开始状态) {opacity: 0;} (完全透明)
```

```
100% (结束状态) {opacity: 1;} (完全不透明) }
```

2.animation-name 属性

定义动画名称，该名称会被@keyframes 规则引用

animation-name: keyframesname / none;

3. animation-duration 属性

定义整个动画效果完成时所需要的时间

animation-duration: time;

参数为 (s) 或 (ms)

当设置为 0 时，则没有任何动画效果

当设置为负数时，则被视为 0

4. animation-timing-function 属性

规定动画的速度曲线，可以定义用那种方式来执行动画速率

animation-timing-function: 属性值;

| 属性值 | 描述 |
|-----------------------|----------------|
| linear | 从头到尾的速度是相同的 |
| ease(默认) | 低速开始，然后加快，结束变慢 |
| ease-in | 低速开始 |
| ease-out | 低速结束 |
| ease-in-out | 低速开始，低速结束 |
| cubic-bezier(n,n,n,n) | 自定义，值在 0~1 之间 |

5.animation-delay 属性

执行动画效果延迟的时间，即规定动画什么时候开始

animation-delay: time;

6. animation-iteration-count 属性

用于定义动画的播放次数

animation-iteration-count: number;

当为 number 时，为播放次数

当为 infinite 时，动画循环播放

7. animation-direction 属性

定义当前动画播放的方向，即动画播放完成后是否逆向交替循环

normal 为默认值，动画正常播放

alternate 值，奇数 (1, 3, 5) 正常播放

偶数 (2, 4, 6) 逆向播放

注: 只有动画播放次数大于等于 2 时此属性才生效，
所有应先定义 animation-iteration-count 属性

8. animation 属性 (综合)

animation-name: my; (动画名称)

animation-duration: 5s; (动画时间)

animation-timing-function: linear; (动画速率)

animation-delay: 2s; (动画延迟时间)

animation-iteration-count: 3; (动画播放次数)

animation-direction: alternate; (动画逆向播放)

可简写为:

animation: my 5s linear 2s 3 alternate;

绘图 (p257)

1.JavaScript 的引入

(1)行内式 (书 p258, 不推荐使用)

(2)嵌入式

```
<script type="text/JavaScript">
```

JavaScript 语句; </ script >

(3)外联式(文件后缀为.js)

```
<script type="text/JavaScript" src="文件路径">
```

```
</ script>
```

也可以省略 type 属性

2.变量

在 JavaScript 中，可以使用字母，数字，符号来命名变量

(1)必须以字母或下滑线开头，中间可以为数字，字母或下划线

(2)变量名不能包括空格，加，减号等

(3)不能使用 JavaScript 中的关键字，如 var int 等

(4)严格区分大小写

3.声明 var 变量与赋值

var unit,room; 声明

var unit=3;赋值

var room=1001;赋值

var name='tom',age=12;声明同时赋值

注：如果声明对象已有初始值，则声明为重新赋值

4.document 对象

想要在 JavaScript 中操作某个标签，首选要获得该标签的属性，可通过 document 获取属性

| |
|---|
| document.getElementById() |
| 返回对拥有指定 id 名的第一个对象的引用
(简单理解为获取指定 id 名的标签) |
| document.getElementsByName() |
| 返回带有指定 name 属性名的对象集合
(简单理解为获取指定 name 名的标签) |
| document.getElementsByTagName() |
| 返回带有指定标签名的对象集合
(简单理解为获取标签名) |
| document.getElementsByClassName() |
| 返回带有指定类名的对象集合
(简单理解为获取指定 class 名的标签) |

HTML5 画布 (p262)

1.创建画布

```
<canvas id="画布名称" width="数值" height="数值"></canvas>
```

2.获取画布

想要通过 JavaScript 控制画布，则必须要获取画布

```
var canvas=document.getElementById('cavs')
```

获取 id 名为 cavs 的画布

3.准备画笔

通过 content 对象，在画布中绘制图形

```
canvas.getContext('2d')
```

2d 代表画布的种类，表示二维图像画图

webgl 代表三维图形

注：在 JavaScript 中，通常会定义一个变量来保存获取的 content 对象

```
var content= canvas.getContext('2d');
```

绘制线

1.初始位置

moveTo(x,y)来定义初始位置

单位默认 px，可省略单位

2.连接端点

```
content.lineTo (x, y)
```

绘制一个端点，并和初始位置相连接

3.描边

```
content.stroke ();
```

(括号不需要加入任何内容)

初始位置和端点的连线并不可见，需要添加描边

线的样式

1.宽度 context.lineWidth='数值';

2.描边颜色 context.strokeStyle='颜色值';

3.端点形状 lineCap='属性值'

| 属性值 | 显示效果 |
|-----------|--------------|
| butt (默认) | 无端点，显示直线方形边缘 |
| round | 显示圆形端点 |
| square | 显示方形端点 |

线的路径

1.重置端点

想要开始新的线，必须要重置端点

```
context.beginPath ();
```

2.闭合路径

```
context.closePath ();
```

3.填充路径

```
context.fill ();
```

4.填充路径图形改变颜色

```
context.fillStyle='颜色值'
```

绘制圆

```
arc (x, y, r, 开始角, 结束角, 方向)
```

(1)x, y 表示圆心的位置

(2)r 表示圆的半径，用于表示圆的大小

(3)方向 false 为顺时针，true 为逆时针 (可省略)

(4)开始角，表示初始弧点位置

用 Math.PI (圆周率，理解为 180 度)

如 270 度为 1.5Math.PI

(5)结束角，结束时弧点的位置

