

- HTML 超文本标记语言 Hyper Text Markup Language
- `<!DOCTYPE html>` *<!--H5 声明 -->*

<!-- HTML 其它版本声明-->

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN"
```

```
"http://www.w3.org/TR/html4/loose.dtd">
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Frameset//EN"
```

```
"http://www.w3.org/TR/html4/frameset.dtd">
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "
```

```
http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
```

- HTML5 支持的两种指定页面使用的字符集的方式:

- 1) 使用 Content-Type 指定字符集: `<meta http-equiv="Content-Type" content="text/html ;charset=UTF-8"/>`

- 2) 直接使用 charset 指定字符集: `<meta charset="UTF-8">`

- `<base target="_blank"/>` 设置这个网页所有链接的目标窗口; `<base href=" " />` 为页面上的所有链接规定默认地址

- `<link/>`

引用外部文档, 常见于引用外部样式。重要属性有三个: rel、href、type。

rel 规定文档与被链接文档之间的关系。

- rel="dns-prefetch" 预先解析缓存文档中使用的域名，目的是为了提高网页访问速度。使用场景：在一个网页频繁使用其他域名资源时。

- rel="shortcut icon"或 rel="icon" 在收藏和标题栏上用于显示的图标。示例：<link rel="icon" href="http://images.cnblogs.com/cnblogs_com/suoning/845162/o_s.png">。注意：IE 浏览器只支持 ico 格式，为了兼容 IE，图片文件采用 ico 格式。

- rel="stylesheet" 引用外部样式表。

- rel="nofollow" 用于指示搜索引擎不要追踪（爬虫抓取），减少垃圾链接。用于<a>标签，使用场景：网页不被信任或是不希望被搜索引擎录入的网站。

- href 资源的路径（相对路径/绝对路径）。

type 规定被连接文档的 MIME 类型，用于明确文件的打开方式。例如：.ico 文件 image/x-icon。

<meta/>

定义关于 HTML 文档的元数据。 重要的属性有三个：http-equiv、name、content

http-equiv 把 content 属性值关联到 http 头部。

- Content-Type（浏览器接受的文档类型，一般是 text/html）

- refresh（网页刷新，以秒为单位）

- expires（设定网页到期时间，一旦过期，必须到服务器上重传）

- <meta http-equiv="Content-Type" content="text/html ;charset=UTF-8"/>

<meta http-equiv="Refresh" content="2">

<meta http-equiv="Refresh"

content="2;URL=https://www.baidu.com">

<meta http-equiv="expires" content="6 Jun 2016"/>

name 把 content 属性关联到一个名称。

- keywords (搜索关键字, 用于搜索引擎抓取信息的显示)

- description (搜索到网站后显示的网页内容简描述)

- author (站点制作者信息)

- generator (用以说明生成工具)

- name 也可以根据特定的功能自定义, 在新浪网中有使用 360 认证和搜狐认证 (<meta name="360-site-verification"content="63349*****"/>、<meta name="sogou_site_verification"content="BVI*****"/>)。

- <meta name="keywords" content="搜索关键字">

<meta name="description" content="简要描述">

<meta name="author"

content="http://cnblogs.com/suoning">

<meta name="generator" content="用以说明生成工具">

content 定义与 http-equiv 或 name 属性相关的元信息, 是必要的属性。

- HTML 元素是指从开始标签到结束标签的所有代码 <开始标签>内容</结束标签>

- 自闭合标签:
折行, <hr/>水平线, <input/>,

- `显示文字`
- `name` 定义文档中的链接方法: `设目标点` ; `连接跳转到设定的目标点`
- ``
- `<h1 align="center">页面标题</h1>`
- `<p>`: 段落标签. 包裹的内容被换行.并且也上下内容之间有一行空白.

`style="text-indent: 2em"`可以设置样式为首行缩进两个字符。

`<blockquote> </blockquote>`可以用来设置整个段落的缩进。

- `background` 定义背景; `bgcolor` 定义背景颜色。
`{background-image: url(图片地址及名称); background-repeat 背景图片是否重复; background-attachment: scroll 默认会滚动, fixed 不滚动, inherit; background-position: 渲染开始位置; }`
- `{background: url("pic/xx.jpg") no-repeat fixed; }` 这行代码可以代替上面的所有内容。
- 格式化: ``粗体, `<small>`小号字, ``加重语气, `<sub>`下标, `<sup>`上标, `<ins>`下划线, ``删除字, `<strike>`: 为文字加上一条中线, `<u>`: 文字下方加下划线, `` `<i>`: 文字变成斜体.

- 表格: <table>表格, {border 边框, align 水平对齐方式, cellpadding 内边距, cellspacing 单元格外边距, } <tr>行, {rowspan}合并行, <td>列, {colspan}合并列, <caption>表格标题。<th>表头单元格, <thead>表头内容, <tbody>主体内容, <tfoot>表注内容 (脚注), <col>列的属性值, <colgroup>供格式化的列组。 frame 不兼容 IE,

- ---

HTML 列表:

1. 无序列表: ,. 属性: <ul type="disc 实心圆点, circle 空心圆, square 方块, none 去除标志", >
2. 有序列表: , 属性: <ol type="A 大写字母序列, a 小写字母序列, I 大写罗马数字, i 小罗马字"; start="从几开始序列" >
3. 嵌套列表: ,, 三个标签的嵌套。
4. 自定义列表: <dl>,<dt>,<dd> 举例: <dl><dt>自定义</dt><dd>自定义内容</dd><dt>自定义</dt><dd>自定义内容</dd></dl>

列表属性

list-style-type: 列表项标志的类型

- none 去除标志
- decimal-leading-zero; 02.
- square; 方框
- circle; 空心圆
- upper-alpha; & disc; 实心圆

list-style-image: 将图象设置为列表项标志

list-style-position: 列表项标志的位置

- inside
- outside

list-style: 缩写

表单

表单用来获取不同类型的用户输入。属性: `action`、`method`、`enctype`

`<form>`

`<input type="text">` `autocomplete` (表单元素要有 `name` 属性自动完成的效果, off, on) , `disabled` (设置或者获取控件的状态, 默认是 false 即可用, 等于 true 时不可用, 不能输入内容)

`<input type="password">` `maxlength` (type 为 text 或 password 时, 表示输入的最大字符数) , 有利于防止 sql 的注入攻击, `readonly` 只读, `placeholder` 框内预置内容(灰色), 写上内容时才消失

`<input type="checkbox">` `<!-- 复选框 -->`

`<input type="radio" name="sex">` `<!-- 单选, 几个选项的 name 值要一致 , value (必须要写, 提交到服务器的 key 值, 实际开发过程中 value 一般是编号) checked (是否被选中的状态) -->`

下拉列表:

`<select>`

`<option>`文字内容 A`</option>`

`<option>`文字内容 B`</option>`

<option>文字内容 C</option>

</select>

<input type="button" value="按钮文字">

<input type="submit" value="提交">

</form>

<textarea cols 列="10" rows 行="20">在文本域中请输入你的信息</textarea>

checkbox 复选框。

name (名字一定要一样一样的，才表示是一组数据，添加到同一 value 值列表提交到服务器)

value (必须要写，提交到服务器的 key 值，实际开发过程中 value 一般是编号)

checked (是否被选中的状态)

file 文件域，上传文件 (不同的浏览器表现形式不同)

submit 提交按钮。用于提交表单。

reset 重置按钮。清空表单的输入，恢复到表单默认的状态。

button 普通按钮。一般结合 javascript 使用。

image 图片按钮，用来提交表单，与 submit 是一样的效果。

src (图片路径)

hidden 隐藏字段。

value (隐藏的内容)

color 颜色标签。value 指定颜色值 (采用#十六进制数表示) 。

date 日期。value 值指定默认日期，格式为****_**_**

(年月日)。

datetime-local 显示本地时间，value 值指定默认的时间，格式为 2016-05-20T11:10:10 (年月日 T 时分秒)。

number 数字向上或者向下滑动。可以填数字然后向上或者向下选择不同的值。

range 滑动标签。min (指定最小值)、max (指定最大值)、value (指定当前默认值)。

week 每年的周数。value 指定哪一年第几周，格式为 2016-W25 (2016 年第 25 周)。

<textarea> 文本域标签。默认表现形式是可以输入很多行文本的文本框。

name (表单提交项的 key)

cols (设置文本域宽度)

rows (设置文本域高度，即行数)

<select> 下拉框标签。使用时要结合<option>子标签一起使用。

name:表单提交项的 key

size: 选项个数

multiple: 多选

<option> 下拉选中的每一项

value (表单提交项的值)

selected (selected 下拉选默认被选中)

<optgroup>为每一项加上分组

<label> 把元素与文本结合起来

友好设计：不只是选中复选框才能选中并打钩，要求点击对应的文字也能选中该复选框。

这种情况下要用到<label>标签的 for 属性（设置或获取给定标签对象指定到的对象，值=另一个元素的 id 号即可）

```
<label for="name">姓名</label>
```

```
<input id="name" type="text">
```

<fieldset> 对表单中的相关元素进行分组

```
<fieldset>
```

```
<legend>温馨提示</legend>
```

```
<div align="middle">不要忘记点赞哦 ==</div>
```

```
</fieldset>
```

value: 表单提交项的值

对于不同的输入类型，value 属性的用法也不同：

- type="button", "reset", "submit" - 定义按钮上的显示的文本
- type="text", "password", "hidden" - 定义输入字段的初始值
- type="checkbox", "radio", "image" - 定义与输入相关联的值

H5 框架

```
<iframe src="**.html" frameborder="0">
```

```
</iframe>
```

```
<frameset rows="20%,50%,30%">
```

```
  <frame src="**.html"></frame>
```

```
  <frame src="*#.html"></frame>
```

```
</frameset> 过时的标签
```

H5 新语义元素明确一个 Web 页面的不同部分：

<header>：头部区域。

<nav>：导航链接。

<section>：定义文档中的区段,比如章节、页眉、页脚或文档中的其他部分，section 通常 包含了一组内容及其标题。

<article>：定义独立的内容。

<aside>：定义页面主区域内容之外的内容（比如侧边栏）。

<figure>：标签规定独立的流内容（图像、图表、照片、代码等等）。

<figcaption>：定义 <figure> 元素的标题。

<footer>：文档的底部区域，一个页脚通常包含文档的作者，著作权信息，链接的使用条款，联系信息等。

- 以下样式代码可以让旧版本浏览器支持本章介绍的块级元素：

```
header, section, footer, aside, nav, article, figure
{
    display: block;
}
```

- IE8 及更早 IE 版本无法在这些元素中渲染 CSS 效果，或者其他的 HTML5 elements.可以使用 HTML5 Shiv 来解决这个问题：

```
<!--[if lt IE 9]>
```

```
<script src="html5shiv.js"></script>
```

```
<![endif]-->
```



<frameset> 框架

用来划分窗体，不能放在<body>中，否则没有效果。

cols （纵向分割页面。其数值表示方法有三种：“30%、30（或者 30px）、*”；数值的个数代表分成的视窗数目且数值之间用“,”隔开。“30%”表示该框架区域占全部浏览器页面区域的 30%；“30”表示该区域横向宽度为 30 像素；“*”表示该区域占用余下页面空间。例如：cols="25%,200,*" 表示将页面分为三部分，左面部分占页面 30%，中间横向宽度为 200 像素，页面余下的作为右面部分。)

rows （横向分割页面。属性和 cols 一样）

frameborder （设置是否显示框架边框。设定值只有 0、1；0 表示不要边框，1 表示要显示边框）

border （框架之间的距离，一般设置为 0）

bordercolor （边框的颜色）

framespacing （设置框架与框架间的保留的空白距离）

<frameset cols="40%,*,*"> 第一个框架占整个浏览器窗口的 40%,剩下的空间平均分配给另外两个框架。

<frameset cols="*,*,*,*"> 浏览器窗口等分为四部分。

<iframe> 框架

元素会创建包含另外一个文档的内联框架（即行内框架）

name （设置框架名称。此为必须设置的属性）

src （设置此框架要显示的网页名称或路径。此为必须设置的属性）

scrolling （设置是否要显示滚动条。设定值为 auto, yes, no)

bordercolor （设置框架的边框颜色）
frameborder （设置是否显示框架边框。设定值只有 0、1；0 表示不要边框，1 表示要显示边框）
noresize （设置框架大小是否能手动调节）
marginwidth （设置框架边界和其中内容之间的宽度）
marginheight （设置框架边界和其中内容之间的高度）
width （设置框架宽度）
height （设置框架高度）

CSS==Cascading Style Sheets

选择器 (Selector)

1. * 通用元素选择器
2. 标签选择器
3. .class 类选择器
4. # ID 选择器
5. , 多元素选择器
6. 后代元素选择器
7. > 子元素选择器
8. + 毗邻元素选择器
9. [] 属性选择器
10. 伪类选择器

常用属性

颜色属性 (color 、transparent、opacity)

字体属性 (font-style、 font-weight、 font-size、 font-family、font)

文本属性 (white-space、direction、text-align、line-height、vertical-align、text-indent、letter-spacing、word-spacing、text-transform、text-overflow、text-decoration、text-shadow、word-wrap)

背景属性 (background-color、background-image、background-position、background-repeat、background-attachment、background)

列表属性 (list-style-type、list-style-image、list-style-position、list-style)

页面布局

边框 (border-style、border-color、border-width、border-radius、border、box-shadow)

盒子模型 (padding、margin)

display

visibility

float 浮动

clip 剪裁图像

overflow 设置当对象的内容超过其指定高度及宽度时如何显示内容

position 规定元素的定位类型 (static、relative、absolute、fixed)

z-index 元素层叠顺序

outline 边框轮廓

zoom 缩放比例

cursor 鼠标的类型形状 (Crosshair、Pointer、Move、

text、wait、help...)

transform、transition 动画效果

CSS 定位

改变元素在页面上的位置 (top,right,bottom,left)

CSS 定位机制：普通流； 浮动； 绝对布局；

div{Position: absolute 绝对定位；

relative 相对定位；

fixed 固定位置；

z-index: -1 /*置于底层*/

}

float

float 属性可用值：

{ left;

right;

none 不浮动

inherit 从父级继承浮动属性

}

清除浮动属性用 clear

clear { left、right、

both 左右两侧均去掉浮动

inherit 从父级继承来的 clear 的值

}

CSS 对齐操作：

div{ margin-left : auto ; margin-right: auto ; }

div{ position:absolute;绝对对齐 left: 0px; }

div{ float:right; 或者居中对齐 center;}

div { margin: 0px auto; }两个值时表示 : 上下; 左右。

文字定位:

垂直对齐:中间 vertical-align:middle

文字水平居中 text-align: center

CSS 引用方法格式:

<style>:样式定义; <link>:资源引用; rel="stylesheet":外部样式表; type="text/css":引入文档的类型; href="规定被链接文档的位置"。CSS 属性和属性值用冒号隔开, 以分号结尾。

1. 行内样式 <div style="..."></div>

2. 内部样式 <style type="text/css">...此处写 CSS 样式</style>

3. 外部样式 <link rel="stylesheet" type="text/css" href="xxx.css">

4. 导入样式 <style> @import "xx.css"; </style>

- 多个样式之间用 ; 分号隔开、一个样式多个属性用空格分开;
- 相同样式并且相同属性, 呈现方式在<head>中的顺序决定, 后面会覆盖前面属性

行内样式>内部样式>外部样式>导入样式 (内部样式 CSS 样式的位置一定在外部样式的后面!)

CSS 样式选择器

又称选择符, 定义了将影响文档中的哪些部分。

CSS 语法由三部分组成 (选择符 { 属性: 值 })

一、元素选择符

A→: ID 选择器#: #名称{属性: 值}; 引用方法<标签 id="名称">

B→: 类(class)选择器: .类名称{属性: 值}; 引用方法<标签 class="类名称">;

C→: *{ } 通配符, 将影响 HTML 中所有此标签的样式, 用来作初始化。

D→: 类型选择符 (标签选择符): 标签名称{ }

二、关系选择符

1.

1. > 子元素选择符: 父亲>儿子

2. ~ 兄弟选择符: 你自己~你的兄弟

3. + 相邻选择器: E + F

4. 空格 包含选择器: E F

三、属性选择符

a[herf] { }

a[herf="www.baidu.com"] { }

四、伪类选择符

- a:link (未访问的链接状态), 用于定义了常规的链接状态。
- a:hover (鼠标放在链接上的状态), 用于产生视觉效果。
- a:active (在链接上按下鼠标时的状态)。
- a:visited (已访问过的链接状态), 可以看出已经访问过的链接。

可靠顺序: l(link)ov(visited)e h(hover)a(active)te,

五、伪对象（伪元素）选择符

Before{ } after{ }

!important>内联>ID>类>标签 | 伪类 | 属性选择器>伪对象>继承>通配符

CSS 元素分类：

- 块级元素** (1)常见块级元素有：<div>, <p>, <h1>...<h6>, , , , <dl>, <dd>, <dt>, <table>, <tr>, <td>, <address>, <blockquote>, <form>; (2)块级元素的特点：a: 新起一行，其后的元素也另起一行； b: 元素的高度、宽度、行高、顶、底边距都可以设置； (3) a{display:block;}设置 display:block 可将元素转化为块级元素；
 - 内联(行内)元素** (1)常用的内联元素有：<a>, , ,
, <i>, , , <label>, <q>, <cite>, <code> ; (2)内联元素的特点： a: 和其他元素都在一行上； b: 元素的高、宽及顶部和底部边距不可设置；（设置高度 height 无效，可以通过 line-height 来设置。） c: 元素的宽度就是它包含的文字或图片的宽度，不可改变。 (3) div{display:inline;} 代码 display:inline 可以将块级元素转换为内联元素；
 - 内联块级元素** (1) 常见的内联块级元素有：, <input>, <iframe>; (2)内联块级元素的特点：a: 和其他元素都在一行上； b: 元素大小边距可以设置； (3) 代码 display:inline-block 就是将元素设置为内联块状元素。
-

透明度 transparent

全透明，使用方式：color: transparent;

opacity

- 元素的透明度，语法：opacity: 0.5;
- 属性值在 0.0 到 1.0 范围内，0 表示透明，1 表示不透明。
- filter 滤镜属性（只适用于早期的 IE 浏览器，语法：filter:alpha(opacity:20);）。

文本属性:

white-space: 设置元素中空白的处理方式

- normal: 默认处理方式。
- pre: 保留空格，当文字超出边界时不换行
- nowrap: 不保留空格，强制在同一行内显示所有文本，直到文本结束或者碰到 br 标签
- pre-wrap: 保留空格，当文字碰到边界时换行
- pre-line: 不保留空格，保留文字的换行，当文字碰到边界时换行

direction: 规定文本的方向

- ltr 默认，文本方向从左到右。
- rtl 文本方向从右到左。

text-align: 文本的水平对齐方式

line-height: 文本行高 (normal 默认)

vertical-align: 文本所在行高的垂直对齐方式:

- baseline 默认
- sub 垂直对齐文本的下标，和<sub>标签一样的效果

- super 垂直对齐文本的上标，和<sup>标签一样的效果
- top 对象的顶端与所在容器的顶端对齐
- text-top 对象的顶端与所在行文字顶端对齐
- middle 元素对象基于基线垂直对齐
- bottom 对象的底端与所在行的文字底部对齐
- text-bottom 对象的底端与所在行文字的底端对齐

text-indent: 文本缩进

letter-spacing: 添加字母之间的空白

word-spacing: 添加每个单词之间的空白

text-transform: 属性控制文本的大小写：

- capitalize 文本中的每个单词以大写字母开头。
- uppercase 定义仅有大写字母。
- lowercase 定义仅有小写字母。

text-overflow: 文本溢出样式：

- clip 修剪文本。
- ellipsis 显示省略符号...来代表被修剪的文本。
- string 使用给定的字符串来代表被修剪的文本。

text-decoration: 文本的装饰

- none 默认。
- underline 下划线。
- overline 上划线。
- line-through 中线。

text-shadow: 文本阴影

- 第一个参数是左右位置
- 第二个参数是上下位置
- 第三个参数是虚化效果

- 第四个参数是颜色
- text-shadow: 5px 5px 5px #888;

word-wrap: 自动换行

- word-wrap: break-word;

text-decoration:none;

/*none(默认,定义标准的文本)

underline(定义文本下的一条线)

overline(定义文本上的一条线)。

line-through(定义穿过文本下的一条线)

blink(定义闪烁的文本)

inherit(规定应该从父元素继承)

text-decoration(属性的值)*/

背景属性

background-color: 背景颜色

background-image 设置图像为背景

- url("http://images.cnblogs.com/cnblogs_com/suoning/845162/o_ns.png"); 图片地址

- background-image:linear-

gradient(green,blue,yellow,red,black); 颜色渐变效果

background-position 设置背景图像的位置坐标

- background-position: center center; 图片置中, x 轴 center, y 轴 center

- 1px -195px 截取图片某部分, 分别代表坐标 x, y 轴

background-repeat 设置背景图像不重复平铺

- no-repeat 设置图像不重复, 常用

- round 自动缩放直到适应并填满整个容器
- space 以相同的间距平铺且填满整个容器

background-attachment 背景图像是否固定或者随着页面的其余部分滚动

background 简写

- background: url("o_ns.png") no-repeat 0 -196px;
- background: url("o_ns.png") no-repeat center bottom 15px;
- background: url("o_ns.png") no-repeat left 30px bottom 15px;

盒子模型

- div{width:220px; height:410px;}宽度、高度
 - div{padding:1px; margin:0px;}内边距、外边距 margin-top:顶
 - div{border: border-width:1px; 边框线条粗细、边框线条样式 border-style: dotted 点状线条, solid 默认实线, double 双线, dashed 虚线;
 - 1 个值, 所有 4 个方向的属性
 - 2 个值 上下/左右
 - 3 个值 上/左右/下
 - 4 个值 上/右/下/左
- border 简写 {border:2px yellow solid;}
- border-color:red;边框颜色
- border-width:边框宽度
- border-radius:10px;圆角效果

border-image:url(bg.jpg)盒子背景图片

box_shadow:{box-shadow: X 轴偏移量 Y 轴偏移量 [阴影模糊半径] [阴影扩展半径] [阴影颜色] [投影方式];} 内阴影用

inset

box-shadow: 边框阴影

- 第一个参数是左右位置
 - 第二个参数是上下位置
 - 第三个参数是虚化效果
 - 第四个参数是颜色
 - box-shadow: 10px 10px 5px #888;
-

display

- none 不显示。
 - block 显示为块级元素。
 - inline 显示为内联元素。
 - inline-block 行内块元素（会保持块元素的高宽）。
 - list-item 显示为列表元素。
-

visibility

- visible 元素可见
 - hidden 元素不可见
 - collapse 当在表格元素中使用时，此值可删除一行或一列，不会影响表格的布局。
-

float 浮动

让一行显示两个块级标签，会脱离文档流

- none
- left 左浮动
- right 右浮动

clear 清除浮动：

- none：默认值。允许两边都可以有浮动对象
 - left：不允许左边有浮动对象
 - right：不允许右边有浮动对象
 - both：不允许两边有浮动对象
-

clip 剪裁图像

rect 剪裁定位元素：

- auto 默认值，无剪切
 - 上-右-下-左（顺时针）的顺序提供四个偏移值
 - 区域外的部分是透明的
 - 必须指定 position:absolute;
 - 例：clip:rect(0px,60px,200px,0px);
-

overflow 设置当对象的内容超过其指定高度及宽度时如何显示内容

- visible 默认值，内容不会被修剪，会呈现在元素框之外。
- hidden 内容会被修剪，并且其余内容是不可见的。
- scroll 内容会被修剪，但是浏览器会显示滚动条以便查看其余的内容。
- auto 如果内容被修剪，则浏览器会显示滚动条以便查看其余的内容。

position 规定元素的定位类型

static

默认值，没有定位，遵从正常的文档流

relative 相对定位元素，相对于其正常位置进行定位，遵从正常的文档流

absolute 绝对定位元素，脱离正常文档流

fixed 绝对定位元素，固定在浏览器某处

- 通过以下四种属性进行定位：

- left
- top
- right
- bottom
- z-index

outline 边框轮廓

- outline-width 轮廓宽度
- outline-color 轮廓颜色
- outline-style 轮廓样式

zoom 缩放比例

cursor 鼠标的类型形状

鼠标放在以下单词上，There will be a miracle:

url: 自定义光标

Auto: 默认

Default: 默认

transform、transition 动画效果

transform 转换, 变形

- origin 定义旋转基点 (left top center right bottom 坐标值) transform-origin: 50px 50px; *transform-origin: left;*
- rotate 旋转 transform: rotate(50deg) 旋转角度可以为负数, 需要先定义 origin。
- skew 扭曲 transform: skew(50deg, 50deg) 分别为相对 x 轴倾斜, 相对 y 轴倾斜。
- scale 缩放 transform: scale(2, 3) 横向放大 2 倍, 纵向放大 3 倍; transform: scale(2) 横竖都放大 2 倍。
- translate 移动 transform: translate(50px, 50px) 分别为相对 x 轴移动, 相对 y 轴移动。

Transition 平滑过渡

- transition-property: 变换的属性 (none(没有属性改变)、all(所有属性改变)、具体属性)
- transition-duration: 变换持续时间
- transition-timing-function: 变换的速率 (ease:(逐渐变慢)、linear:(匀速)、ease-in:(加速)、ease-out:(减速)、ease-in-out:(加速然后减速)、cubic-bezier:(自定义时间曲线))
- transition-delay: 变换延迟时间
- transition: 缩写

#property 指定属性对应类型

1、color: 通过红、绿、蓝和透明度组件变换 (每个数值单独处理), 如: background-color, border-color, color,

outline-color 等 CSS 属性;

2、length: 真实的数字, 如: word-spacing, width, vertical-align, top, right, bottom, left, padding, outline-width, margin, min-width, min-height, max-width, max-height, line-height, height, border-width, border-spacing, background-position 等属性;

3、percentage: 真实的数字, 如: word-spacing, width, vertical-align, top, right, bottom, left, min-width, min-height, max-width, max-height, line-height, height, background-position 等属性;

4、integer 离散步骤 (整个数字), 在真实的数字空间, 以及使用 floor() 转换为整数时发生, 如: outline-offset, z-index 等属性;

5、number 真实的 (浮点型) 数值, 如: zoom, opacity, font-weight 等属性;

6、transform list。

7、rectangle: 通过 x、y、width 和 height (转为数值) 变换, 如: crop;

8、visibility: 离散步骤, 在 0 到 1 数字范围之内, 0 表示“隐藏”, 1 表示完全“显示”, 如: visibility;

9、shadow: 作用于 color、x、y、和 blur (模糊) 属性, 如: text-shadow;

10、gradient: 通过每次停止时的位置和颜色进行变化。它们必须有相同的类型 (放射状的或是线性的) 和相同的停止数值以便执行动画, 如: background-image;

11、paint server (SVG): 只支持下面的情况: 从 gradient 到

gradient 以及 color 到 color，然后工作与上面类似；

12、space-separated list of above：如果列表有相同的项目数值，则列表每一项按照上面的规则进行变化，否则无变化；

13、a shorthand property：如果缩写的所有部分都可以实现动画，则会像所有单个属性变化一样变化。

#支持执行 transition 效果的属性

| Property | Name | Type |
|----------|------|------|
|----------|------|------|

| | |
|------------------|----------|
| background-color | as color |
|------------------|----------|

| | |
|---------------------|--|
| background-position | as repeatable list of simple list of length, percentage, or calc |
|---------------------|--|

| | |
|---------------------|----------|
| border-bottom-color | as color |
|---------------------|----------|

| | |
|---------------------|-----------|
| border-bottom-width | as length |
|---------------------|-----------|

| | |
|-------------------|----------|
| border-left-color | as color |
|-------------------|----------|

| | |
|-------------------|-----------|
| border-left-width | as length |
|-------------------|-----------|

| | |
|--------------------|----------|
| border-right-color | as color |
|--------------------|----------|

| | |
|--------------------|-----------|
| border-right-width | as length |
|--------------------|-----------|

| | |
|----------------|--------------------------|
| border-spacing | as simple list of length |
|----------------|--------------------------|

| | |
|------------------|----------|
| border-top-color | as color |
|------------------|----------|

| | |
|------------------|-----------|
| border-top-width | as length |
|------------------|-----------|

| | |
|--------|--------------------------------|
| bottom | as length, percentage, or calc |
|--------|--------------------------------|

| | |
|------|--------------|
| clip | as rectangle |
|------|--------------|

| | |
|-------|----------|
| color | as color |
|-------|----------|

| | |
|-----------|-----------|
| font-size | as length |
|-----------|-----------|

| | |
|-------------|----------------|
| font-weight | as font weight |
|-------------|----------------|

| | |
|--------|--------------------------------|
| height | as length, percentage, or calc |
|--------|--------------------------------|

left as length, percentage, or calc
letter-spacing as length
line-height as either number or length
margin-bottom as length
margin-left as length
margin-right as length
margin-top as length
max-height as length, percentage, or calc
max-width as length, percentage, or calc
min-height as length, percentage, or calc
min-width as length, percentage, or calc
opacity as number
outline-color as color
outline-width as length
padding-bottom as length
padding-left as length
padding-right as length
padding-top as length
right as length, percentage, or calc
text-indent as length, percentage, or calc
text-shadow as shadow list
top as length, percentage, or calc
vertical-align as length
visibility as visibility
width as length, percentage, or calc

word-spacing as length

z-index as integer

鼠标放在以下图片上，There will be a miracle:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html;  
charset=utf-8" />
```

```
<title>nick</title>
```

```
<meta charset="utf-8" />
```

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

```
.img-see-2016-7-2 {
```

```
background-image:
```

```
url("http://images.cnblogs.com/cnblogs_com/suoning/845  
162/o_sea.jpg");
```

```
background-size: 660px;
```

```
background-repeat: no-repeat;
```

```
height: 300px;
```

```
width: 600px;
```

```
transition-duration: 30s;
```

```
transition-timing-function: ease;
```

```
transition-property: background-size;
```

```
}
```

```
.img-see-2016-7-2:hover {
```

```
background-size: 2000px;
```

```
}
```

```
</style>
</head>
<body>
  <div class="img-see-2016-7-2"></div>
</body>
</html>
```

```
<video src="E:\Web 前端\Html5\html02 基础标签.mp4"
width="320" height="140" controls="controls">
```

```
  <source src="" type="video/ogg">
  <source src="" type="video/mp4">
```

****视频加载不成功显示提示文字****

```
</video>
```

```
<!-- HTML5 中显示视频, ogg、mp4、webM; controls
属性是播放控件按钮 ; source 元素可以链接不同的视频文
件--><hr>
```

常用颜色

值: Cyan #00FFFF DarkBlue #00008B Darkorange #FF8C00 DarkViolet #9400D3 DeepPink #FF1493
DodgerBlue #1E90FF Gold #FFD700 GreenYellow #ADFF2F HoneyDew #F0FFF0 HotPink #FF69B4
Indigo #4B0082 LightGreen #90EE90 LightSlateBl
ue #8470FF

