内联样式：style属性

内部样式：head标签下<style type=”text/css”> </style>

外部样式：通过<link rel=”stylesheet” type=”text/css” href=” ”>链接

1. 块和内联
2. <div> </div> ：块元素，每个独占一行(p、h1、h2…均为块元素)

width height

1. 内联元素/行内元素：只占自身大小的元素，不会占用一整行。

<span>文字</span> <a></a> <iframe></iframe> <img></img>

1. rules：块元素用于页面布局，内联元素用于设置文本样式

a元素包含任意元素，除了自身

p元素不包含任何块元素，包括自身也不行

1. 选择器
2. 元素选择器，直接使用标签名
3. id选择器：#id值{ }
4. 类选择器：.class值。class=”类值 类值 类值”
5. 选择器分组（并集选择器）：选择器1，选择器2，选择器3{ }
6. 复合选择器（交集选择器）：条件1条件2条件3{ }（条件之间互相挨着）
7. 通配选择器：\*，选中页面中所有元素
8. 父选择器>子选择器{ } （ie6不支持） 祖先元素 后代元素{ }
9. 子选择器：nth-child（n）{ } n可以是数字，或者**odd奇数**，**even偶数**

子选择器：first-child{ } last-child{ }

p:first-child{ } 必须同时为p标签以及对应父元素的第一个孩子

子选择器：first-of-type / last-of-type / nth-of-type（）{ }

p:first-of-type{ } 指定对应父元素所有孩子中的第一个p元素

1. 兄弟元素选择器： 前一个 + 后一个{ } ***必*须紧挨着，之间不能隔任何东西**

前一个 ~ 后面所有{ }

1. 伪类选择器：用于表示元素的某种特殊状态 格式： 元素：伪类

link - 没访问过的链接；visited - 访问过的链接，仅能设置color；根据历史记录判断

hover - 鼠标悬停；active - 元素被点击并且未松手的状态；ie6不支持超链接外的元素

focus - 获取焦点时；

元素**::**selection - 元素被选中时，（fiefox中使用时需要使用 **元素::-moz-selection**）

1. 伪元素选择器：用于选取元素中特殊位置的内容 格式： 元素：伪元素

first-letter 首字符；first-line 首行；[[1]](#footnote-1)\*

before / after 元素最前/后面的部分，一般没有东西，需要结合content属性使用

e.g: p:before{ content: ”文字内容”；color：··· }，出现的内容不可中

1. 否定伪类选择器： p : not（.hello） 选中除了class值为hello外的p元素
2. 属性选择器：根据元素中的属性或属性值来选择元素

格式：p[属性]{ } p[属性名=”属性值”]{ }

p[属性名**^**=’’属性值的开头’’]{ }

p[属性名**$**=’’属性值的结尾’’]{ }

p[属性名**\***=’’属性值包含的内容’’]{ }

\*选择器的优先级： 1000 100 10 1 0

1. 内联样式 > id > class和伪类 > 元素选择器 > \*
2. 继承的样式优先级最低，低于通配选择器
3. 当元素包含多种选择器时，需将其优先级相加再比较
4. 选择器优先级相加结果不能超过其最大数量级，例如10个class最后结果不超过99
5. 如果选择器优先级一致，选择最靠后的样式
6. 并集选择器的优先级会分别计算。h1,a,p{ }
7. 在样式最后添加“！important”，会使某个属性值获得最高优先级，即使js也改不了。

E.g: a {color : yellow ! important; background : red; }

\*伪类选择器的优先级：

1. link visited hover active优先级一致
2. hover不能写在active之后
3. link visited 均不能写在 hover active 之后（**l**o**v**e **h**a**t**e）
4. 继承：祖先元素的样式被后代元素继承
5. 并不是所有都会被继承：所有background相关属性都不会被继承，但是默认值为transparent（透明的）。
6. 文本标签及样式
7. <em> </em>：表示一段内容中的重点，默认表现为 *斜体*

<strong> </strong>：表示一段内容很重要，默认表现为 **粗体**

1. <i> </i>，*斜体*；<b> </b>，**粗体**
2. <small> </small>：表示细则内容，其内容必父元素的 字体小 一点
3. <cite> 参考的内容 </cite>：参考的内容稍微斜体
4. <q> 行内引用 </q>：默认为行内引用加引号，该引号无法被选中

<blockquote> 长引用 </blockquote>：块元素，独占一行，不能置于p标签内

1. <sup> </sup>：上标

<sub> </sub>：下标

1. <ins> 插入的内容 </ins>：默认加 下划线

<del> 删除的内容 </del>：默认加 ~~删除线~~

1. <pre> </pre>：预格式标签，会将代码中的格式原封不动保存，不会忽略多个空格
2. <code>代码</code>
3. 列表：无序列表ul，有序列表ol，定义列表
4. 无序列表<ul></ul> 列表项<li> </li> ul、li均为块元素

属性：type=” 列表项前的项目符号 ” css除去项目符号：list-style：none;

li的快速写法 li{ $ }\*10

1. 有序列表<ol> </ol>

属性：type=”项目符号类型” 默认为数字，其他：A a i I …

\* 列表可互相嵌套，子列表最好放在父表的<li>标签中

1. 定义列表<dl></dl>

结构：<dl>

<dt> 名词 </dt>

<dd> 描述1 </dd>

<dd> 描述2 </dd>

</dl>

1. 文本格式化
2. 长度单位
3. 像素px： 不同显示器像素大小不同。手机一个像素约为电脑的1/4，手机会 默认将像素\*4
4. 百分比%： width：50%；表示宽度为父元素的50%
5. em： 本元素的字体大小
6. 颜色
7. 颜色单词：red blue green yellowgreen
8. RGB值：rgb（红，绿，蓝），各色浓度值取0-255，或者0-100%
9. 十六进制：格式 - #红色绿色蓝色，每色2位（00 - ff），越大越浓。

#000000黑 #ffffff白

相邻两位均是重复的可简写：#00ff00 简写为 #0f0

1. 字体样式
2. 文字颜色：color
3. 文字大小：font-size，默认16px。其设置的并非文字实际大小，而是每个文字所处格子的高度，该格子不一定比文字大
4. 文字字体：font-family，可以同时指定多个字体，使用“,”隔开，浏览器会优先使用前面的字体。只有客户计算机中有该字体才能使用。名称内有空格，要使用“”括住。
5. 字体的分类：serif衬线字体，sans-serif非衬线字体

monospace等宽，cursive草书，fantasy虚幻字体

以上字体并非具体类型，浏览器会根据该大类选择具体类型的字体，一般放在font-family的最后一个选项

1. font-style字体风格：normal默认，italic斜体，oblique倾斜，inherit从父元素继承
2. font-weight文本加粗效果：bold加粗，normal默认，数字100-900间的9个值（一般用不上），bolder更粗，litter更细
3. font-variant 设置小型大写字母，small-caps所有字母变成大写，但是小写字母对应的大写字母型号会小一些。默认normal无效果。
4. font综合格式：size必须写在倒数第二个，family必须写在最后一个。其他的可写可不写。
5. 行
6. 行间距，css中无对应样式。 **行间距=行高-字体大小 对于单行文本，设置行**

**高与父元素高度一致，可使文本垂直居中**

1. 行高：line-height，文字会默认在行中垂直居中，并不会靠上或者靠下

单位px，%（相对于字体大小），单存数字（相对于字体大小的倍数）

**字体大小/行高**

1. font中指定行高： font：30px/50px,微软雅黑；若未设置则使用默认值。该种规定方式必须在line-height之前，即使未设置行高。
2. 其他
3. 大小写text-transform：none默认；uppercase全部大写；lowercase全部小写

capitalize **单词**首字母大写，通过空格识别；

1. 修饰 text-decoration：none默认；underline下划线（超链接的默认值）；

overline上划线； ~~line-through删除线~~；

1. letter-spacing：字符间距，中英文均可

word-spacing：单词间距。中英文均根据空格识别

* 1. 对齐text-align：left默认左对齐；right；center居中；justify两端对齐（通过调整空格大小对齐）
  2. 首行缩进text-indent：px em，可设负值
  3. white-space：nowrap；使文字强制一行显示

1. 盒子模型（框模型）
2. 内容区：content；边框border；内边距padding；[[2]](#footnote-2) 外边距margin
3. 内容大小width height ；并非整个盒子的大小，为元素所在的区域
4. 边框 border-width（默认3px），border-color（默认black），border-style

border：1px, solid, red; **没有顺序要求，所有边的样式都一样，但是有**

**border-top border-left border-bottom border-right单独指定**

\*border-width：上，右，下，左；（顺时针）/ 上，左右，下；/ 上下，左右；

（亦适用于另外两个属性）

border-top-width，border-right-width，border-bottom-width，border-left-width

\*border-style：**none默认**；solid实线；dotted点状虚线；dashed线状虚线；

double双实线；

\*border-shadow

1. 内边距padding，会扩大盒子大小，background-color会扩展到padding区

padding规则和border类似，但是没有padding-style，padding-color

1. 外边距margin，不属于盒子的一部分，表示与其他盒子的边的最小距离

\*设置置顶居中margin：0，auto；

\*垂直外边距重叠：垂直方向的兄弟元素之间的相邻外边距取最大值，而非求和

使之不相邻就可以取消重叠，变成求和

\*子元素和父元素**外边**垂直相邻，则其**外边距**会被设置给父元素，可通过父元素加padding border来取消，或者干脆不用margin，而使用padding。

#给父标签添加一个空的<div>无法取消垂直外边距重叠，但是使用<table>可以

.clearfix : before{ content : ””; **display : table;**  }

1. 取消默认边距 \*{ margin：0；padding：0；}
2. 内联元素的盒模型：1、不能设置width height

2、可以设置水平方向内边距、边框，而垂直方向内边距、边框不会影响布局，即不会把其他元素挤开

3、水平方向外边距不会重叠，只会求和；不支持垂直外边距

1. display和visibility
2. display：inline将元素作为**行内元素**显示

block将元素做为**块元素**显示

inline-block将元素转换为行内块元素，既有行内元素的特点：**不会独占一行**，又有块元素特点：**可以设置宽高**

none**不显示元素并且不占用空间**

1. visibility：visible默认会在页面显示

hidden **不显示元素**但是**依然占用空间**

1. **\*子元素默认在父元素的内容区里，如果子元素大小超过内容区，会默认超出父元素来显示，称为溢出**。

overflow：设置父元素如何处理溢出visible默认可见

hidden 溢出内容不会被显示

scroll 添加滚动条，通过滚动条来查看完整内容，无论有无溢出均会添加滚动条

auto 看情况添加滚动条

1. 文档流

处在网页最底层，表示一个页面中的位置。元素默认处在文档流中

1. 块元素：在文档流中自上向下显示，weight height为auto。当宽度高度为auto时，指定内边距并不会影响可见框大小。块元素的大小默认会被内容撑开
2. 内联元素：只占自身大小，从左向右排列。内联元素的宽高默认会被内容撑开
3. 浮动

使元素脱离文档流，其下元素会立即向上移动。**浮动的元素比非浮动元素高一层**

**float**：none默认不浮动

left使元素脱离文档流，向左上侧浮动，不会超过父元素或者其他浮动元素

如果浮动元素上面是一个块元素，他也不会超过块元素

浮动元素不会超过它的兄弟元素，最多平齐

right

**\*ctrl+方向键 可以调整代码位置**

浮动的元素不会盖住文字，文字会自动环绕在元素周围

**块元素脱离文档流后，宽高不再占据父元素全部，而是被内容撑开。**

**内联元素脱离文档流后变成块元素**

**高度塌陷问题：**父元素默认被子元素撑开，但是为子元素设置浮动后，脱离文档流，不足以撑开父元素，发生高度塌陷

**解决方法：# 将高度写死，不推荐**

**# 所有元素都有一个隐含属性BFC（Block Formatting Context）默认关闭**

**开启bfc的元素具有如下特性： 1、父元素垂直外边距不会和子元素重叠**

**2、不会被浮动元素所覆盖**

**3、可以包含浮动子元素**

开启方式：设置元素浮动 宽度丢失并且布局混乱

设置元素绝对定位 宽度丢失并且布局混乱

设置元素为display：inline-block，可以解决问题但是宽度会丢失

**将overflow设置为非visible的值，overflow：hidden auto**

**（在ie6中需要开启Has Layout属性： zoom：1；表示放大倍数，仅兼容ie8以下、或者指定宽度）**

**#清除高度塌陷的最终方案：在父元素最末添加一个人空白的div，使用clear清除兄弟元素对他的影响，足以撑开父元素**

**或者使用css样式：after伪类添加空白块元素（ie6不支持）**

**子元素：after{ content：“”；**

**display：block；**

**clear：both}**

**clear清除其他元素浮动对本元素产生的影响：**none默认不清除 **left**清除左浮动的影响

**right** 清除右浮动 both清除两侧浮动的影响，实际上是清除対之影响最大的一个

同时解决高度塌陷及父子元素垂直外边距重叠的方法：

.clearfix : after, clearfix : after{ content : “” ; display : table ;

clear : both ; }

1. 定位 - 将指定元素摆放到任意位置 属性position
2. 默认无定位static，忽略偏移量的设置（非static值时开启定位，可通过left right top bottom属性值来设置偏移量，偏移量指元素相对于某个方向的参照物的位移）
3. 相对定位relative：i. 不设置偏移量时不会出现任何变化

ii. 参照物为自身原来的位置

iii. 元素**不会脱离文档流**，原来的位置依然占据空间

iv．元素的z轴会提升一个层级

v．不会改变元素的性质

1. 绝对定位absolute： i. 使元素**脱离文档流，元素的z轴会提升一个层级**

ii．未设置偏移量，元素位置不会发生变化

iii. 参照物是**离他最近**的 **开启了定位**的祖先元素，如果所有的祖先元素均无定位，则相对于浏览器窗口定位。但会随着浏览器滚动而滚动

1. 固定定位fixed：i. 固定定位也是一种绝对定位，但是参照物永远是**浏览器窗口**

ii. 元素不会随着窗口滚动而滚动

iii. 固定定位必须有宽度

1. 层级关系：如果定位层级一致，则下面的元素会盖住上面的元素

z-index属性设置元素层级：正整数，层级越高，越优先显示。

**对于未开启定位的元素，不起作用。父元素的层级再高，也不会覆盖子元素。**

1. **设置元素透明度opacity：0-1的值。不兼容ie8及以下浏览器**

filter：alpha（opacity=0-100），兼容ie6，但是在IE TEXTER中无法测试

1. 背景
2. background-image：url（ 链接 ）[[3]](#footnote-3)

\* 如果背景图片较大，那么只显示图片的一部分。可设置元素与图片一样大。

\* 如果背景图片较小，默认会把图片平铺直至填满元素。

background-repeat：repeat默认双方向重复 no-repeat不重复

repeat-x水平方向重复 repeat-y垂直方向重复

# 背景颜色和图片同时设置，那么背景颜色将会作为图片的底色

^ 背景图片默认紧贴左上角显示

background-position：

\*可使用top left bottom right center则两个值来规定位置

background-position：top center。如果只给出一个值，第二个值默认**center**

\***可以使用两个数字值来规定x y的位置，可正负。**~：20px 15px；

\*

& background-attachment：scroll默认图片随页面滚动而滚动

fixed 图片不随页面滚动而滚动，**使得图片的定位永远相对于浏览器窗口，都是图片不会超过元素。**

**一般只设置给body**

**@** 简写属性 background：#bfa url（） center center no-repeat fixed; **没有顺序要求**

**ie6中，如果图片使用的是png24格式，会导致透明效果失灵。应该把格式改为png8，图片清晰度下降；或者使用JavaScript。**

1. **表格**

<table height=10px weight=15px align=“center”>

<tr>

<th></th>表头的单元格，内容默认加粗居中

</tr>

<tr>

<td></td> 单元格

</tr> 一行

</table> 一个表格

\*合并单元格 td属性 colspan=“占据的格数”横向合并单元格

rowspan=“占据的格数”纵向合并单元格

\*table是块元素

css：\*table和td的border之间默认有一个距离，border-spacing

亦可用border-collapse：collapse；使td的边框合并，不需给table再设置边框

\*如果collapse，则border-spacing自动失效

#长表格：分为三个部分，表头<thead>、主体<tbody>、底部<tfoot>，都是table子标签，<tr>被置于这些标签内。三者在代码中可以调乱顺序，都是显示出来的顺序不会乱。

如果普通表格中无<tbody>，则浏览器会自动加上一个<tbody>， 因此**<table>的子元素并不是<tr>，而是<tbody>**

1. 表单

<form>表单</form>：form的属性action=”目标地址”，指向**表单提交的目标服务器地址**

**表单元素：<input>**

**type=” ” : text默认为单行文本框**

**submit 提交按钮，需要配合属性name=”查询字符串”，用户提交的信息会附在url地址后提交给服务器，格式：url？查询字符串。**

**例如，mmmp.html?username=123456 & hello=22623**

**password密码框**

**radio 单选按钮，通过name进行分组，同名为一组，只能选一个**

**必须指定value值，被选中的按钮在提交后，value值会被赋予查询字符串。例如，mmp.html?gendername=male**

**checkbox 多选框。用name分组，value用于区分提交**

**reset 重置按钮，是表单元素恢复为默认值**

**bottom 创建单纯按钮 直接使用<bottom></bottom>也可以，可规定其type为submit reset bottom，与input类似**

**属性disable =true，禁用按钮**

**value=” ”填写文本框的默认值、按钮的内容**

**<select></select>创建下拉列表，子元素<option></option>指定选项**

**<select name=” ”>**

**<option value=” ”>选项</option>**

**<option value=” “>选项</option>**

**</select>**

**\*多选的下拉列表属性：multiple=”multiple”**

**\*使用<optgroup></optgroup>进行分组，通过属性label=” ”规定分组名**

**<select name=” ”>**

**<optgroup label=” ”>**

**<option value=” ”>选项</option>**

**<option value=” “>选项</option>**

**</optgroup>**

**<optgroup label=” ”>**

**<option value=” ”>选项</option>**

**<option value=” “>选项</option>**

**</optgroup>**

**</select>**

***%* 单选框多选框的默认选中属性，checked=”checked”**

**下拉列表的默认选中属性，selected=”selected”**

**<textarea></textarea>多行文本框，使用name属性**

**默认可随用户调整大小，使用resize：none取消该默认功能**

**ie默认添加滚动条，使用overflow：auto取消滚动条**

**placeholder属性指定提示词，不兼容ie8及以下**

**<laber></label>圈起表单的提示文字。属性for=“对应input框的id值”，可协助焦点的定位**

**长表单：<fieldset>多个表单项</fieldset>，可用于表单项的分组。**

**<legend>分组的组名</legend>**

1. 框架集（不建议使用）

与内联框架作用类似，用于在一个页面中引入外部页面，但是可以引入多个页面

<frameset>必须得有属性：rows=”50%,20%,30%”指定框架集一行行排列；或者cols=” , , ”一列列排列。若使用\*，可以设定为自动，如cols=”20% , **\*** , 30%”

<frame src=”目标地址”>

<frame src=”目标地址”>

<frame src=”目标地址”>

</frameset>，不能和<body></body>出现在同一个页面，但是可以套多层的<frameset></frameset>

1. hack，其实是利用浏览器的bug，不建议使用

某些代码只需要在一些特殊的浏览器中执行。css Hack实际上指一些特殊代码，通过这种方式，为一些浏览器设置特殊的代码。

\*条件hack：< ! - - [if IE] 代码 <![endif] - - > 该种代码**只在ie9及以下版本可识别**

**快捷键alt /**

**可以放在任何地方**

< ! - - [if IE6] 代码 <![endif] - - > 仅ie6可识别

< ! - - [if ***lt*** IE9] 代码 < ! [endif] - - > 小于ie9

gt 大于 gte 大于或等于 lt 小于 lte 小于或等于 ！除了

\*属性hack：\_属性：值 选择ie6及以下

\*属性：值 选择ie7及以下

属性：值\9 选择ie6及以上

属性：值\0 选择ie8及以上及Opera15以下

\* \* html 属性 只有ie6认识，绿色为空格

1. 命名规范

**—** 尽量英文，不要英文拼音混用

**—** 格式：驼峰命名法；单词均小写，使用 \_ 或 - 分隔

1. Ctrl shift F 格式化 [↑](#footnote-ref-1)
2. 这个盒子大小由这三部分决定 [↑](#footnote-ref-2)
3. 第一次切换图片时会出现一个快速的闪烁，原因是背景图片是以外部资源形式加载进网页的，浏览器每加载一次外部资源需要单独发送一次请求，但浏览器只有在需求资源时才加载资源。可以**将需要切换的几张图片集合到同一张上（雪碧图），再通过background-position来切换图片** [↑](#footnote-ref-3)