

Web 前端

HTML 5 + CSS

页面设计

以后的你，一定会感谢现在拼命的自己

温昌锦

# 小知识

## 头文件

【格式】 `<!doctype html>`

【功能】声明: `<!DOCTYPE>`, HTML 也有多个不同的版本, 只有完全明白页面中使用的确切能完全正确地显示出 HTML 页面。这就是 `<!DOCTYPE>` 的用处

## Head 部分

`lang="en"`

【功能】`<head>` 标签中的 `lang="en"`, 功能是表示文字的显示形式, “en”表示英文, “zh”表示中文 (也可以省略, 通常是不写的)

//例: `<head lang="en">…内容…</head>`

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

【功能】对于中文网页需要使用 `<meta charset="utf-8">` 声明编码, 否则会出现乱码

## body 部分

`bgcolor="#8b0000"`

【功能】给 body (即整个页面) 设置背景颜色

【区别】与 background 的区别, background 是设置背景颜色, bgcolor 是设置背景图片

//`<body bgcolor="#8b0000">…内容…</body>`

## 转义字符

字符	代码
“ < ”	&lt;
“ > ”	&gt;
“ ” ”	&quot;
“ ‘ ”	&apos;

# 文字篇

## HTML 文本样式

文字 居中 代码: `<center>…内容…</center>`

文字 加粗 代码: `<b>…内容…</b>`

文字 斜体 代码: `<i>…内容…</i>`

文字 上标 代码: `<sup>…内容…</sup>`

文字 下标 代码: `<sub>…内容…</sub>`

文字 大小 代码: `<h1>…内容…</h1>`

【注】 [h 后面数字的取值范围为 1-6,数字越小文字越大]

【拓展】 文字的对齐方式: <h1 align="center">...内容...</h1/>

//其中 left 为左对齐, center 为居中, right 为右对齐

文字 删除线 代码: <del>...内容...</del>

文字 着重号 代码: <em>...内容...</em>

文字 下划线 代码: <u>...内容...</u>

文字 插入 代码: <ins>...内容...</ins>

文字加重语气 代码: <strong>...内容...</strong>

分隔符(水平线) 代码: <hr>

文字独占一行(段落标签) 代码: <p>...内容...</p>

拓展: 文字 居中: <p style="text-align:center;">...内容...</p>

拓展: 文字 背景颜色: <p style="background-color: gray;">...内容...</p>

换行 代码: <br>

空格 代码: &nbsp;

定义大号字 代码(已淘汰): <big>...内容...</big>

## CSS 文本样式

【格式】 style="文本样式属性"

【属性样式】

字体颜色 代码: color: 颜色;

字体大小 代码: font-size: \* px;

字体 代码: font-family: "楷体";

字符行高 代码: line-height: \* px;

首行缩进 代码: text-indent: \* en;

字符间距 代码: letter-spacing: \* px;

单词间距 代码: word-spacing: \* px;

自动换行 代码: text-wrap: normal;

阴影效果 代码: text-shadow: 20px 30px 5px red;

【注释】 20px : 阴影在文字的基础上向右移动 20px

30px : 阴影在文字的基础上向下移动 30px

5px : 阴影的清晰度

Red : 阴影的颜色

字母大小写规范 代码: text-transform: 规范; { 统一大写: uppercase  
统一小写: lowercase  
首字母大写: capitalize

对齐方式 代码: text-align: 对齐方式; { 左对齐: left  
居中对齐: center  
右对齐: right

# 重难点

## 超链接

打开新窗口: `<a href="链接地址" target="_blank">…网页内显示内容…</a>`

覆盖原窗口: `<a href="链接地址" target="_self">…网页内显示内容…</a>`

//可以省略不写, 默认覆盖原窗口打开指定链接

标签超链接:

**【格式】** `<a name = "标签名称">…内容…</a>`

`<a href = "#标签名称">…内容…</a>`

//例: `<a name = "tips">…内容…</a>`

`<a href = "#tips">…内容…</a>`

去除超链接下划线: `style="text-decoration: none;`

//例: `<a href="链接地址" style="text-decoration: none;">…网页内显示内容…</a>`

超链接 状态属性

**【格式】**

`a:link{` //未被访问时的超链接属性

`color: greenyellow;`

`}`

`a:visited{` //已经访问过的超链接属性

`color: red;`

`}`

`a:hover{` //鼠标指针位于超链接上方时的属性

`color: darkorchid;`

`}`

`a:active{` //被点击时的超链接属性

`color: black;`

`}`

**【位置】** 在外部引用文件 (\*.css) 中间定义

## 选择器

CSS 优先级

行内样式(`<…style=""…>`) > ID 选择器(`#box{…}`) > 类选择器(`.con{…}`)

> 标签选择器(`div{…}`) > 通用选择器(`*{…}`)

ID 选择器

**【格式】** `#元素 id {元素格式}`

**【功能】** 一般用来定义一个元素的样式

类 (class) 选择器

**【格式】** `#元素类别名称 {元素格式}`

**【区别】** 区别于 id 选择器, 一般用来定义多个元素的样式

派生选择器

**【格式】** 外层标签 内层标签 { 标签样式 }

【功能】仅改变嵌套元素内部单个元素样式

【例】 h1 a { color: red; }

属性选择器 + 属性和值选择器

【格式】 定义： title="元素属性名称";

引用： [ title ] { 属性标签样式; }

[ title="属性元素值" ] { 属性标签样式 }

【功能】定义了属性的元素都会根据属性标签内的样式进行相应改变，属性元素内部的值不一样也可以进行进一步的定义，即对不同的属性值进行不同的样式定义

ID 选择器 + 派生选择器 【组合用法】

【功能】仅改变嵌套元素内部单个元素

【例】

```
<style type="text/css">
    #new a {color: red;}
</style>
</head>
<body>
    <div id="new">
        我是一个<a href="#">超链接</a>
    </div>
</body>
```

## HTML 样式 【3 种】

外联样式表：

【格式】 <link rel="stylesheet" type="文件类型" href="文件地址">

【功能】建立一个样式模版，方便后期对大量样式做修改

【注意】css 外联样式一般写在 css 文件中

//例： <link rel="stylesheet" type="text/css" href=" mystyle.css">

内部样式表：

【格式】 <style type= "text/css">

#元素 id (body、html) { 元素格式 }

</style>

【功能】建立一个样式模版，方便后期做修改

【注意】css 内部样式一般写在<head>中

```
//例： <style type= "text/css">
    body {background-color: red}
    p {margin-left: 20px}
</style>
```

内联样式表

【格式】 <p style="样式">…内容…</p>

【功能】对单个元素进行样式修改，方便理解

【注意】css 内联直接写在元素标签内部

//例： <p style="color: red">

## CSS 浮动

元素 **左→右** 浮动: float: left;

元素 **右→左** 浮动: float: right;

元素 **去除** 浮动: clear: 浮动;  $\left\{ \begin{array}{ll} \text{left} & /* 去除左浮动 */ \\ \text{right} & /* 去除右浮动 */ \\ \text{both} & /* 去除所有浮动 */ \end{array} \right.$

## 表格 代码

```
<table border="1px" cellspacing="0">
  <tr><td width="99xp" align="center">...单元格内容...</td><td width="99xp" align="center">...单元格内容...</td></tr>
  <tr><td width="99xp" align="center">...单元格内容...</td><td width="99xp" align="center">...单元格内容...</td></tr>
  <tr><td width="99xp" align="center">...单元格内容...</td><td width="99xp" align="center">...单元格内容...</td></tr>
</table>
```

### 【注释】

注释: 行 代码: <tr>...内容...</tr>

注释: 列 代码: <td>...内容...</td>

注释: 表头 代码: <th>...内容...</th> // th 等价于加粗并水平居中的 td

注释: 单元格宽度 代码: width="\*xp" //可写在<tr><td><table>标签中

注释: 单元格高度 代码: height="\*xp" //可写在<tr><td><table>标签中

注释: 外边框 代码: <table border="\*px">...单元格...</table>

注释: 外边框 (外联): border: \*px solid 边框颜色;

注释: 合并边框 代码: <table style="border-collapse: collapse;">...单元格...</table>

注释: 字体颜色 代码: <td style="color:red;">...内容...</td> (单元格内容为红色)

注释: 表格背景色 代码: <table bgcolor="颜色">...单元格...</table>

注释: 单元格间隙 代码: <table border="1" cellspacing="\*px">...单元格...</table>

注释: 表格 对齐方式: <table align="center">...单元格...</table>

//[center<居中> right<靠右> left<靠左>]

注释: 单元格内容 对齐方式: text-align: center;

//[center<居中> right<靠右> left<靠左>]

注释: 表格背景色 代码: background-color: 颜色;

注释: 单元格 内边距: padding: \*px;

表格 代码<补 简写格式>:

```
<table>
  <col width="99xp"> //第1列宽度
  <col width="99xp"> //第2列宽度
  <col width="99xp"> //第3列宽度
  <col width="199xp"> //第4列宽度
  <tr><td>...内容...</td><td>...内容...</td></tr>
  <tr><td>...内容...</td><td>...内容...</td></tr>
  <tr><td>...内容...</td><td>...内容...</td></tr>
  <tr><td>...内容...</td><td>...内容...</td></tr>
```

或改成如下写法 (可以说是简写)

```
<colgroup span=3 width=99xp>
  <col width="99xp"> //第1列宽度
  <col width="99xp"> //第2列宽度
  <col width="99xp"> //第3列宽度
</colgroup>
```

group 表示分组, 上述代码意思为, 前三列为一组, 宽度 99px

第四个属性不一样, 单独拿出来

</table>

表格 代码<补>

列合并: colspan="合并列数"

行合并: rowspan="合并行数"

## 列表 代码

无序列表(默认) 代码: <ul>

<li>...标题 1 内容...</li>

<li>...标题 2 内容...</li>

</ul>

【注】可以通过以下代码改变无序列表前面的符号

默认【●】→ { 实心圆【●】 代码: <ul type="disc"> ...列表... </ul>  
空心圆【○】 代码: <ul type="circle"> ...列表... </ul>  
实心方块圆【◆】 代码: <ul type="square"> ...列表... </ul>

有序列表(默认) 代码: <ol>

<li>...标题 1 内容...</li>

<li>...标题 2 内容...</li>

</ol>

【注】可以通过 start 代码定义列表从哪个数开始计数

【格式】<ol start="10">...内容...</ol>

【释义】定义数列开始从 10 开始标序号

【注】可以通过以下代码改变有序列表前面的排序

默认【数字】→ { 数字 代码: <ol type="1"> ...列表... </ol>  
大写字母 代码: <ol type="A"> ...列表... </ol>  
小写字母 代码: <ol type="a"> ...列表... </ol>  
大写罗马 代码: <ol type="I"> ...列表... </ol>  
小写罗马 代码: <ol type="i"> ...列表... </ol>

自定义列表 代码:

【格式】<dl>

<dt>...列表...</dt>

<dd>...注释...</dd>

<dt>...列表...</dt>

<dd>...注释...</dd>

<dt>...列表...</dt>

<dd>...注释...</dd>

</dl>

【例如】

<dl>

<dt>苹果</dt>

<dd>这是一个水果</dd>

<dt>猫</dt>

<dd>这是一只动物</dd>

</dl>

【预览结果】

苹果

这是一个水果

这是一只动物

猫

## 动画效果

<!-- 提示信息：并不是所有的浏览器都支持动画效果-->

<!--我们首先要添加要支持的浏览器-->

```
-webkit-transform: translate( * px , * px );      /* 支持 safari   chrome */  
-ms-transform: translate( * px , * px );          /* 支持 IE */  
-o-transform: translate( * px , * px );           /* 支持 opera */  
-moz-transform: translate( * px , * px );          /* 支持 Firefox */
```

### 2D 转换方法

移动 translate()

**【外联格式】** transform: translate(\* px , \* px);

**【功能】** 对元素进行移动

**【参数】** 左边参数值为向 X 轴正方向移动的距离，右边的参数值为向 Y 轴正方向移动的距离

旋转 rotate

**【外联格式】** transform: rotate( \* deg );

**【参数】** 旋转角度

**【功能】** 对元素进行旋转

缩放 scale

**【外联格式】** transform: scale( 参数 1, 参数 2);

**【参数】** 参数 1: 缩放后的宽度倍数 (数值)

参数 2: 缩放后的高度倍数 (数值)

**【功能】** 对元素进行缩放

倾斜 skew

**【外联格式】** transform: skew( 参数 1, 参数 2);

**【参数】** 参数 1: 围绕 X 轴旋转的角度 ( \* deg )

参数 2: 围绕 Y 轴旋转的角度 ( \* deg )

**【功能】** 对元素进行自定义角度倾斜

矩阵 matrix

**【格式】** 不明

**【功能】** 不明

### 3D 转换方法

rotateX()

**【外联格式】** transform: rotateX( \* deg );

**【参数】** 旋转角度

**【功能】** 在立体的角度对元素进行 X 轴的旋转

rotateY()

**【外联格式】** transform: rotateY( \* deg );

**【参数】** 旋转角度



【功能】在立体的角度对元素进行Y轴的旋转

## 过渡效果

过渡属性简写 (transition)

【外联格式】 transition: 过渡效果 过渡时间;

过渡效果属性 { 宽度: *width*  
高度: *height*  
动画效果: *transform*  
.....

【功能】提前定义将要执行动画的过渡时间

//例如: transition: width 2s,height 2s,transform 2s;

过渡名称 (transition-property)

【外联格式】 不明

【功能】 不明

过渡时间 (transition-duration)

【外联格式】 不明

【功能】 不明

过渡效果时间曲线 (transition-timing-function)

【外联格式】 不明

【功能】 不明

效果延迟 (transition)

【外联格式】 transition-delay: \* s;

【参数】 延迟时间

【功能】 效果延迟展示

【 注 】

并不是适配所有的浏览器

```
-webkit-transition-delay: * s; /* 支持safari chrome */
-moz-transition-delay: * s; /* 支持Firefox */
-ms-transition-delay: * s; /* 支持IE */
-o-transition-delay: * s; /* 支持opera */
.....
```

## 动画效果

【外联格式】 div { /\*定义动画的位置\*/

animation: 动画名称 动画过渡时间 infinite alternate;  
/\*上行 infinite alternate 设置动画循环执行 (可省略, 即只执行一次)\*/

}

@keyframes 动画名称 { /\*以下为动画过渡举例\*/

0% {background-color: red;left: 0px;top: 0px;}  
25% {background-color: orange;left: 200px;top: 0px;}  
50% {background-color: yellow;left: 200px;top: 200px;}  
75% {background-color: green;left: 0px;top: 200px;}  
100% {background-color: red;left: 0px;top: 0px;}  
}

【功能】自定义动画过渡效果

## 多列效果

分列的数量 (column-count)

【外联格式】 column-count: 列数;

【功能】 将内容分为多列

列间距 (column-gap)

【外联格式】 column-gap: \* px

【功能】 调整列与列中间的距离

分割线属性设置 (column-rule)

【外联格式】 column-rule: \*px outset 颜色;

【参数】 分割线粗细 & 分割线颜色

【功能】 设置分割线属性

列宽 (column-width)

【外联格式】 column-width: auto | length;

【参数】 auto: 浏览器将决定列的宽度

length: 指定列宽的长度

【功能】 定义列的宽度

//外联样式举例

```
div{ /* 定义容器的初始状态 */
width: 400px; /* 定义在未被触发情况下的容器宽度 */
height: 250px; /* 定义在未被触发情况下的容器高度 */
background-color: red; /* 定义在未被触发情况下的容器背景颜色 */
transition: width 2s,height 2s,transform 2s; /* 定义容器各种效果在变化中的过渡时间 */
-webkit-transition: width 2s,height 2s,-webkit-transform 2s; /* 定义容器各种效果在【safari chrome 浏览器】变化中的过渡时间 */
-ms-transform: width 2s,height 2s,-ms-transform 2s; /* 定义容器各种效果在【IE 浏览器】变化中的过渡时间 */
-o-transform: width 2s,height 2s,-o-transform 2s; /* 定义容器各种效果在【opera 浏览器】变化中的过渡时间 */
-moz-transform: width 2s,height 2s,-moz-transform 2s; /* 定义容器各种效果在【Firefox 浏览器】变化中的过渡时间 */
transition-delay: 2s; /* 效果延迟两秒之后执行 */
-webkit-transition-delay: 2s; /* 定义容器各种效果在【safari chrome 浏览器】中延迟两秒后执行 */
-ms-transition-delay: 2s; /* 定义容器各种效果在【IE 浏览器】中延迟两秒后执行 */
-o-transition-delay: 2s; /* 定义容器各种效果在【opera 浏览器】中延迟两秒后执行 */
-moz-transition-delay: 2s; /* 定义容器各种效果在【Firefox 浏览器】中延迟两秒后执行 */
margin: 100px auto; /* 利用外边距定义容器的位置 */
border-radius: 20px; /* 定义容器的边角的弧度 */
}
div:hover{ /* 定义容器被触发时的状态 触发条件:鼠标指针停留在容器上 */
width: 200px; /* 定义被触发后容器的宽度 */
height: 200px; /* 定义被触发后的容器高度 */
transform: rotate(-360deg); /* 定义容器旋转角度 */
-webkit-transform: rotate(1080deg); /* 定义在【safari、chrome 浏览器】中容器旋转的角度 */
-ms-transform: rotate(1080deg); /* 定义在【IE 浏览器】中容器旋转的角度 */
-o-transform: rotate(1080deg); /* 定义在【opera 浏览器】中容器旋转的角度 */
-moz-transform: rotate(1080deg); /* 定义在【Firefox 浏览器】中容器旋转的角度 */
}
```

# 其他 代码

## 元素 属性

元素属性代码	代码效果
min-width	设置元素最小宽度
max-width	设置元素最大宽度
min-height	设置元素最小高度
max-height	设置元素最大高度

## 框架 结构

【框架集标签】（已淘汰）

【格式】  
`<frameset rows="20%, 50%, 30%">`  
    `<frame src="网页链接或网页文件"></frame>`  
    `<frame src="网页链接或网页文件"></frame>`  
    `<frame src="网页链接或网页文件"></frame>`  
`</frameset>`

【注】rows="20%, 50%, 30%"为网页横向排放比例

【注】cols="20%, 50%, 30%"为网页纵向排放比例

【位置】代替了<body>标签的位置，不可写在<body>标签内

【内联框架】

【格式】  
`<iframe src="网页链接或网页文件"></iframe>`  
【注】去除边框效果：`<iframe src="http://..." frameborder="0"></iframe>`  
【注】框架高度：`<iframe src="http://..." height ="*xp"></iframe>`  
【注】框架宽度：`<iframe src="http://..." width ="*xp"></iframe>`

【位置】放置在<head>标签中

## 图片 格式

图片透明度

【格式】opacity: 透明度;                      //透明度取值范围【透明 [0]~不透明 [1]】

//外联举例：`img {`  
    `opacity: 0.5;`  
    `}`

## 背景 格式

背景颜色

【格式】style="background-color: 颜色;"

【功能】给标签添加背景颜色

//例：`<h1 style="background-color: red;">...内容...</h1>`

## 背景图片

【格式】 style="background-image: url(图片路径);"

【功能】 给标签添加背景图片

//例: <body style="background-image: url(img/longmao.gif);"></body>

### 【背景图片拓展 实例演示】

```
body{
background-image: url(../img/longmao.gif);    //设置背景图片
background-size: 100% 1000px;                //设置背景的高度以及宽度
background-repeat: no-repeat;                 //不允许背景图片重复
background-repeat: repeat;                   //允许背景图片重复
background-repeat: repeat-x;                 //允许x轴重复背景图片
background-repeat: repeat-y;                 //允许y轴重复背景图片
background-attachment: fixed;                //背景不随着内容的滚动而滚动
background-position: 位置属性代码;
}
```

【位置属性代码】

right:	右对齐	居中开始显示
left:	左对齐	居中开始显示
center:	居中	居中开始显示
left top:	居中	顶部开始显示
.....		

【特例】 自定义位置显示: \*px \*px

## 轮廓 代码

### 【格式】

轮廓粗细 代码: outline-width: 1px;

轮廓颜色 代码: outline-color: red;

轮廓样式 代码: outline-style: 样式;

groove	//实线
double	//双实线
dotted	//虚线

【功能】 给容器或者标签添加一些外边框

【位置】 写在外联样式文件 (\*.css) 中

## 边框样式

### 边框样式

边框样式 代码: border-style: 边框样式;

上边框样式 代码: border-top-style: 顶部边框样式;

下边框样式 代码: border-bottom-style: 底部边框样式;

左边框样式 代码: border-left-style: 左边框样式;

右边框样式 代码: border-right-style: 右边框样式;

边框样式	代码
实线	groove
虚线	dotted
双实线	double

## 边框宽度

边框宽度 代码: border-width: \* px;  
上边框宽度 代码: border-top-width: \* px;  
下边框宽度 代码: border-bottom-width: \* px;  
左边框宽度 代码: border-left-width: \* px;  
右边框宽度 代码: border-right-width: \* px;

## 边框颜色

边框颜色 代码: border-color: 颜色;  
上边框颜色 代码: border-top-color: 颜色;  
下边框颜色 代码: border-bottom-color: 颜色;  
左边框颜色 代码: border-left-color: 颜色;  
右边框颜色 代码: border-right-color: 颜色;

## 边框 其他代码

圆角边框 代码: border-radius: \* px;  
边框图片 代码: border-image: url( 图片路径 );  
边框阴影 代码: box-shadow: 20px 30px 5px green;

【注释】 20px : 阴影在文字的基础上向右移动 20px  
30px : 阴影在文字的基础上向下移动 30px  
5px : 阴影的清晰度  
Red : 阴影的颜色

## 代码 格式

### 标签内边距

【格式】 <h1 style="padding: \*px;">... 内容...</h1>

【功能】 让标签内元素跟标签产生一定的内边距

//例: <h1 style="padding: 20px;">内边距为 20px</h1>

## 无分类 代码

### No.1 无名称

【格式】 cursor: {  
    cursor: col-resize;      移动图标  
    cursor: cell;          加号图标  
    cursor: copy;          指针右上角加号  
    .....

【功能】 规定当指向某元素之上显示的指针类型

### No.2 无名称

【格式】 display { inline    内容在同一行显示  
    .....

【功能】 设置是否及如何显示元素

【位置】 适合运用在列表当中

### No.3 无名称

【格式】 visibility { hidden    内容不可见  
    visible    内容可见

【功能】 设置元素是否可见或不可见

# 拓展内容 (CSS 属性代码简写)

## 背景属性:

Background-color           /\*背景颜色\*/  
Background-image           /\*背景图片\*/  
Background-repeat          /\*背景图平铺方式\*/

### 可简写为:

background: gray(背景色) url(xxx/xx.png)(背景图片) no-repeat(平铺方式);  
//可以省略部分属性, 但是顺序不可以打乱

## 边框属性:

border-width               /\*边框宽度\*/  
border-style               /\*边框样式\*/  
border-color               /\*边框颜色\*/

### 可简写为:

Border: 1px(边框宽度) solid(边框样式) #D3f402(边框颜色);  
//顺序不能打乱, 只能省略边框颜色(默认为黑色), 其他属性不可以省略

## 字体属性

font-style:italic;           /\*斜体\*/  
font-weight:bold;           /\*加粗\*/  
font-family:arial,sans-serif; /\*字体种类\*/  
font-size:20px;              /\*字号大小\*/  
line-height:35px;           /\*行高\*/

### 可简写为:

Font: italic(斜体字) bold(加粗) 20px(字号大小)/35px(行高) arial(默认字体),sans-serif(备用字体),"微软雅黑(备用字体)"  
//可以省略部分属性, 但顺序不能打乱

## 边距属性 (外边距)

margin-top:10px;            /\*顶部边距\*/  
margin-right:20px;           /\*右边边距\*/  
margin-bottom:30px;          /\*底边边距\*/  
margin-left:40px;            /\*左边边距\*/

### 可简写为:

margin: 10px(上) 15px(右) 10px(下) 15px(左);  
margin: 10px(上) 15px(左右) 10px(下);  
margin: 10px(上下) 15px(左右);  
margin: 10px(上下左右);  
//每种简写的意思都不一样

## 填充属性（内边距）

```
padding-top      /*上边距*/  
padding-right    /*右边距*/  
padding-bottom   /*下边距*/  
padding-left     /*左边距*/
```

**可简写为：**

```
margin: 10px(上) 15px(右) 10px(下) 15px(左);  
margin: 10px(上) 15px(左右) 10px(下);  
margin: 10px(上下) 15px(左右);  
margin: 10px(上下左右);  
//每种简写的意思都不一样
```

## 关于颜色

```
color: red        /*单词表示*/  
color: rgb(184,134,11) /*RGB表示*/  
color: #B8860B    /*十六进制表示*/
```