# CSS语法

## CSS语法格式

经常用到的CSS元素是选择器、属性和值。其语法格式为： **<style type=”text/css”>**

**选择器1{**

**属性1：值;**

**......**

**}**

**......**

**</style>**

其中，<style>...</style>标签是放在<head></head>标签中的。在<style>......</style>标签中， 可以有多个选择器，每个选择器也可以有多个属性。

## CSS样式的引入

在开发中，通常使用三种方式实现CSS的调用，即外部样式表、内部样式表、内联样式。

1. 外部样式表是指专门编写第三方css文件，然后在页面中通过<head></head>标签中插入<link>标签对外部样式文件进行调用。其语法格式为：

**<link href=”文件名.css” rel=”stylesheet” type=”text/css”>**

css样式文件以css为后缀。当需要使用多个css文件时，可以插入多条该语句。

2）内部样式表是指直接在当前页面内编写CSS样式代码，其语法格式为：  
 **<style type=”text.css”>**

**选择器1**

**......**

**</style>**

3）内联样式是指直接在标签中定义样式，例如：

**<p style=”属性1：值;属性2：值”;>......</p>**

属性和属性之间使用分号隔开。

# 选择器

1. 选择器就是样式，每个选择器就是一个样式。所有的选择器都是定义在<style>...</style>标签中的。
2. 常用的选择器有通配选择器、标签选择器、群组选择器、包含选择器、id选择器、class选择器等。

## 选择器命名规范

选择的命名可以使用如下几类字符：

1. 大小写的英文字母；
2. 数字0-9；
3. 连字号-；
4. 下划线\_；
5. 冒号：；
6. 句号。；

## 通配选择器

通配选择器的语法格式为：  
 **\*{  
 属性：值;**

**......  
}**

使用了通配选择器，则页面中所有的元素都会应用此样式。

## **类型选择器**

类型选择器也叫标签选择器，类型选择器是指以网页中已有的标记作为名称的选择器，如body、div、p等标签作为选择器类型。其语法格式为：

**标签名{**

**属性：值;**

**......**

**}**

定义了类型选择器后，则页面中对应的标签都会应用其定义的样式。

## **群组选择器**

群组选择器又称多标签选择器或分组选择器。当多个选择器定义了相同的样式时，可以 将多个选择器以半角符号“，”分割的方式合并为一组。其语法格式为：

**选择器1，选择器2···{**

**属性：值;**

**......**

**}**

对于多个不同的标签，如果使用的样式是相同的，则可以使用群组选择器。

## **id选择器**

id选择器是使用“#”符号将选择器名称作为一个id，其语法格式为：

**#选择器名称{**

**属性：值;**

**......**

**}**

每一个标签中都可以使用id=”选择器”的形式指定相应的样式。

## class选择器

class选择器类似于id选择器，使用“.”符号将一个选择器作为一个calss，其语法格式 为：

**.选择器名称{**

**属性：值;**

**......**

**}**

在标签中通过class=”选择器”的形式来指定样式。

## 伪类选择器

伪类是一种特殊的选择器，之所以成为伪类，是因为它们所指定的对象在网页文档中并 不存在，它们指定的是元素的某种状态。其语法格式为：

**选择器名称：伪类{属性：值}**

伪类是CSS已经定义好的，常用的伪类有4个：link、表示鼠标未点击过的样式；visited、表示鼠标点击过的样式；hover、表示鼠标移动到该元素时该元素的样式、active、表示鼠标点击该元素时该元素的样式。如a:link{}、#p:hover{}、.s1:active{}等等

## 包含选择器

包含选择器中前后两个对象以空格隔开，例如：

**.aa p{**

**属性：值;**

**}**

表示应用了.aa类选择器的元素中的p元素应用会该样式。

包含选择器可以实现跨级包含，即父元素可以包含该元素以下的各级子元素。

## 子元素选择器

子元素选择器是对某个元素的最近子元素进行样式指定。如：

**#aa > p{ #nav > .top{**

**属性：值; 属性：值;  
 } }**

表示对应用了aa选择器元素的最近子元素p进行样式指定。

表示对应用了nav选择器元素的应用了top选择器的最近子元素进行样式指定。

子元素选择器中所控制的元素只能是”>”前面所指对象的最近子元素。

# CSS属性介绍

CSS常用的属性有字体属性、颜色和背景属性、文本属性、块属性、边框属性、项目符 号和编号属性、层属性等。

## 字体属性

字体属性的功能是用来设置页面字体的显示样式，常用的字体属性有：

|  |  |
| --- | --- |
| 属性 | 含义 |
| font-family | 设置字体样式，如宋体、楷体 |
| font-style | 设置字体倾斜 |
| font-size | 设置字体的大小 |
| font-weight | 设置字体的粗细 |
| color | 设置字体颜色 |

## 颜色和背景属性

颜色和背景属性的功能是设置页面元素的颜色和背景颜色，常用的颜色和背景属性如下：

|  |  |
| --- | --- |
| 属性 | 颜色和背景属性 |
| color | 设置元素前景色 |
| background-color | 设置元素背景色 |
| background-image | 设置背景图像 |
| background-repeat | 设置背景图像重复方式 |
| background-attachmen | 设置字体的大小 |
| background-position | 设置背景图案的初始位置 |

## 文本属性

文本属性的功能是设置页面文本的显示效果，常用的文本属性如下：

|  |  |
| --- | --- |
| 属性 | 文本属性 |
| text-align | 设置文字的对齐方式 |
| text-index | 设置文本的首行缩进 |
| line-height | 设置文本的行高 |
| a:link | 设置链接未访问过的状态 |
| a:visisted | 设置链接访问过的状态 |
| a:hover | 设置链接的鼠标激活的状态 |

## 块属性

块属性的功能是设置页面内块元素的显示效果，常用的块属性如下：

|  |  |
| --- | --- |
| 属性 | 文本属性 |
| margin-top | 设置上边距 |
| margin-right | 设置右边距 |
| padding-top | 设置上填充距 |
| padding-right | 设置右填充距 |

## 边框属性

|  |  |
| --- | --- |
| 属性 | 边框属性列表 |
| border-top-width | 设置上边框宽度 |
| border-right-width | 设置右边框宽度 |
| width | 设置宽度 |
| height | 设置高度 |

边框属性的功能是设置页面内边框的显示效果，常用的边框属性如下：

## 项目符合和编号属性

项目符号和编号属性的功能是，设置页面内项目符合和编号元素的显示效果，常用的项目符合和编号属性如下：

|  |  |
| --- | --- |
| 属性 | 项目符号和编号属性列表 |
| display | 设置是否显示符号 |
| white-spac | 设置空白部分的处理方式 |

## 层属性

层属性的功能是设置页面内层元素的定位方式，常用的层属性如下：

|  |  |
| --- | --- |
| 属性 | 层属性列表 |
| absolute | 设置绝对定位 |
| relative | 设置相对定位 |
| static | 设置无特殊定位 |

# CSS编码规范

CSS编码规范是指在书写CSS代码时遵循的格式，虽然它不是强制性的。但按照标准格式书写的CSS代码不但会便于读者的阅读，而且有利于程序的维护和调试。

## 书写规范

按照Web标准的要求，标准的CSS书写应该包括如下2个方面：

1. 书写顺序

在使用CSS时，最好将CSS文件单独书写并保存为独立文件，而不是把其书写在HTML页面中。这样做的好处是，便于CSS样式的统一管理，便于代码的维护。

1. 书写方式

在CSS中，虽然可以在不违反语法格式的前提下使用任何书写方式都能正确执行，但是还是建议开发者在书写每一个属性时，使用换行和缩进来书写。这样做的好处是，是编写的程序一目了然，便于程序的后续维护。

## 命名规范

1. 命名规范是指CSS选择器在命名时所要遵循的规则。在网页设计过程中，需要定义大量的选择器来实现页面表现，如果没有好的命名规范会导致页面的混乱或名称的重复，从而造成额外的麻烦。所以，CSS在命名时应遵循一定的规范，使页面结构达到最大化。
2. 在CSS开发中，通常使用的命名方式是结构化命名方法。它是相对于传统的表现效果命名方式来说的。例如，当文字颜色为红色时，使用red命名；当某元素位于页面中间时，使用center命令。这种传统的表现效果命名方式看起来比较直观和方便，但是这种方法不能达到标准布局所要求的页面结构和效果相分离的要求。所以，结构化命名方式便结合了表现效果的命名方式，实现样式命名。
3. 常用的页面元素的命名方法如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 页面元素 | 名称 | 页面元素 | 名称 |
| 主导航 | mainnav | 子导航 | subnav |
| 页脚 | foot | 内容 | content |
| 头部 | header | 底部 | footer |
| 商标 | label | 标题 | title |
| 顶部导航 | topnav | 侧栏 | sidebar |
| 左侧栏 | leftsidebar | 右侧栏 | rightsidebar |
| 标志 | logo | 标语 | banner |
| 子菜单 | submenu | 注释 | note |
| 容器 | container | 搜索 | search |
| 登录 | login | 管理 | admin |

# 颜色介绍

颜色在CSS中的地位十分重要，页面元素通过颜色的设置，可以实现页面美观的表现效果。在CSS应用过程中通常通过如下两种方式实现颜色定义：

1. 名称定义；
2. 十六进制定义

## 颜色名称定义

颜色名称定义时指用使用颜色的名称来设置页面元素的颜色值。例如，设置为红色可以使用“red”来实现。但只有一定数量的颜色名称才能被浏览器识别，所以颜色名称定义的方法只能实现比较简单的颜色效果。能够被浏览器识别的颜色名称如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 颜色名称 | 描述 | 颜色名称 | 描述 |
| yellow | 黄色 | white | 白色 |
| blue | 蓝色 | navy | 深蓝 |
| silver | 银色 | olive | 橄榄色 |
| purple | 紫色 | gray | 灰色 |
| green | 绿色 | lime | 浅绿 |
| maroon | 褐色 | aqua | 水绿 |
| black | 黑色 | fuchsia | 紫红 |
| red | 红色 | teal | 深青色 |

## 十六进制定义

十六进制定义是指使用颜色的十六进制数值来定义颜色样式值。任何的颜色都可以使用对应的十六进行数值来表示。使用十六进制定义颜色，能够在页面中定义更加复杂的颜色。如果不知道某个颜色的十六进制代码是多少，可以在某些网站上进行转换：

<https://www.sioe.cn/yingyong/yanse-rgb-16/>

# 长度单位

在CSS中常用的长度单位有绝对长度大单位和相对长度单位两种。

## 绝对长度单位

绝对长度即某元素的实际长度值，现实中常用的的绝对长度单位如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 描述 | 名称 | 描述 |
| mm | 毫米 | pt | 磅 |
| m | 米 | pc | pica |
| in | 英寸 | cm | 厘米 |

## 相对长度单位

相对长度单位即某元素的相对元素值，网页设计中最为常用的相对长度单位如下：

1. 字体大小：em

em通常用来定义文本中font-size（字体大小）的值。

1. 文本高度：ex

ex和em类似，功能是定义文本中元素的高度。

1. 像素：px

像素px是网页设计中作为常用的长度单位。通常显示器的界面将被划分为多个单元格，其中的每个单元格就是一个像素。也就是说，像素px的具体大小是和屏幕的分辨率有关的。