

Web程序设计

▶▶ 第4章 CSS基础



全国计算机等级考试

National Computer Rank Examination



第4章 CSS基础

- ◆ [4.1](#) **CSS**样式表的概念与作用.....●
- ◆ [4.2](#) **CSS**的基本语法和使用方法.....●
- ◆ [4.3](#) **CSS**的层次及其作用优先级.....●
- ◆ [4.4](#) **CSS**的属性单位.....●
- ◆ [4.5](#) **CSS**的常用属性.....●
- ◆ [4.6](#) **CSS**布局.....●

4.1 CSS样式表的概念与作用

- ◆ **CSS**（**Cascading Style Sheet**，可译为“层叠样式表”或“级联样式表”）是一组格式设置规则，用于控制**Web**页面的外观。
- ◆ 通过使用**CSS**样式设置页面的格式，可将页面的内容与表现形式分离。
- ◆ 页面内容存放在**HTML**文档中，而用于定义表现形式的**CSS**规则则存放在另一个文件（**.css**文件）中或**HTML**文档的某一部分（通常为文件头部分）。
- ◆ 将内容与表现形式分离，不仅可使维护站点的外观更加容易，而且还可以使**HTML**文档代码更加简练，缩短浏览器的加载时间。

4.2 CSS的基本语法和使用方法

- ◆ **CSS**的定义是由三个部分构成：选择符（**selector**）、属性（**properties**）和属性的取值（**value**）。
- ◆ 语法：**selector {property: value}** (选择符{属性:值})
- ◆ 选择符可以是多种形式，一般是要定义样式的**HTML**标记，例如**body**，**p**，**table**，...。可以通过此方法定义它的属性和值，属性和值要用冒号隔开，如：**body {color: black}**。
- ◆ 如果属性的值由多个单词组成，则必须在值上加引号：**p {font-family: "sans serif"}**（定义段落字体为**sans serif**）。
- ◆ 如果需要对一个选择符指定多个属性，则使用分号将所有的属性和值分开。如：**p {text-align: center; color: red}**（段落居中排列，并且段落中的文字为红色）。

4.2 CSS的基本语法和使用方法

1. 选择符组

把相同属性和值的选择符组合起来书写，用逗号将选择符分开，这样可以减少样式重复定义。如：

h1,h2,h3,h4,h5,h6{ color: green } ; p, table{ font-size: 9pt }（段落和表格里文字尺寸为9号字）

2. 类选择符

类选择符能够把相同的元素分类定义不同的样式。在自定义类的名称前面加一个点号。假如想使两个不同的段落一个段落右对齐，一个段落居中，则可以先定义两个类，即**p.right {text-align: right}**和**p.center {text-align: center}**，然后用在不同的段落里。

在选择符中省略HTML标记名，可以把几个不同的元素定义成相同的样式[如，**.center{text-align: center}**]，这样的类可以被应用到任何元素上。



4.2 CSS的基本语法和使用方法

3. id选择符

在HTML页面中，**id**参数指定了某个单一元素，**id**选择符是用来对这个单一元素定义单独的样式。

- ◆ **id**选择符的应用和类选择符类似，只要把**class**换成**id**即可。**id**只能在一个页面中出现一次，而**class**可以多次运用。
- ◆ **id**选择符要在**id**名称前加上一个“#”号。
- ◆ 定义**id**选择符的属性也有两种方法：
 - ① **id**属性将匹配所有**id="intro"**的元素
#intro{.....}
 - ② **id**属性只匹配**id="intro"**的段落元素
p#intro{.....}



4.2 CSS的基本语法和使用方法

4.包含选择符

可以单独对某种元素包含关系定义的样式表，元素1里包含元素2，这种方式只对元素1里的元素2定义，对单独的元素1或元素2无定义，例如：

```
table a
{
font-size: 10px
}
```

在表格内的链接改变了样式，文字大小为10像素，而表格外的链接的文字仍为默认大小。



4.2 CSS的基本语法和使用方法

5. 样式表的层叠性

层叠性就是继承性。样式表的继承规则是外部的元素样式会被这个元素所包含的其他元素继承。所有在元素中嵌套的元素都会继承外层元素指定的属性值，有时会把很多层嵌套的样式叠加在一起，除非另外更改。

不同的选择符定义相同的元素时，要考虑不同的选择符之间的优先级。**id**选择符、类选择符和**HTML**标记选择符三者中，因为**id**选择符是最后加到元素上的，所以优先级最高，其次是类选择符。如果想超越这三者之间的关系，可以用**!important**提升样式表的优先级，如：

```
p{ color: #FF0000!important }
```

```
.blue{ color: #0000FF}
```

```
#id1 { color: #FFFF00}
```



4.2 CSS的基本语法和使用方法

6.注释：/*...*/

可以在**CSS**中插入注释来说明代码的意思，注释有利于以后编辑和更改代码时理解代码的含义。

在浏览器中，注释是不显示的。**CSS**注释以“/*”开头，以“*/”结尾，如下：

```
/*定义段落样式表*/
```

```
p{
```

```
text-align: center; /*文本居中排列*/
```

```
color: black; /*文字为黑色*/
```

```
font-family: arial; /*字体为arial*/
```

```
}
```

4.3 CSS的层次及其作用优先级

- ❖ 按照从底层到高层的顺序，样式表分为行内样式表、文档层样式表和外部样式表三个层次。
- ❖ 行内样式表只能作用于单个标签的内容
- ❖ 文档层样式表则能够对文档的整个主体起作用
- ❖ 外部样式表可以应用到任意数目文档的主体中。
- ❖ 从优先级顺序上讲，行内样式表优先于文档层样式表，而文档层样式表又优先于外部样式表。

4.3 CSS的层次及其作用优先级

1. **行内样式**：表样式表是直接在元素处通过**style**这个属性定义的。如：

```
<div style="color:blue"> hello,world </div>
```

2. **文档层样式表**是为文档的所有内容的呈现提供统一样式的方式。文档层样式表说明位于文档的头部区域，能够作用于文档的整个主体中。如：<style type="text/css">

```
div{ color:Blue } </style>
```

3. **外部样式表**：能够对多个文档起作用，实现跨页面的样式共享。如在**default.css**文件中有：

```
.myclass{ color:yellow; }
```

在页面中引用该样式表文件：

```
<link rel="stylesheet" href="default.css" />
```



4.4 CSS的属性单位

❖ **CSS**是由属性和属性值组成的，有些属性值会用到单位。**css**中属性值的各种单位与**HTML**中有所不同，**css**允许以各种不同的格式指定尺寸和颜色。

❖ 主要有以下3类：

- ❖ （1）长度单位，其中pc、pt、mm、cm、in属于绝对长度单位
- ❖ （2）百分比单位，如：P{line-height:150%}
- ❖ （3）颜色单位，**CSS**中指定颜色可以用5种方法



4.5 CSS的常用属性

4.5.1 字体属性

(1) 字体名称属性font-family

设定字体名称，如Arial, 宋体, 黑体等。如：**H2 {font-family:Arial}**

(2) 字体大小属性font-size

设置字体的大小。字体大小的设置可以有多种方式，最常用的就是pt和px(pixel)。如：

P {font-size:16pt}

(3) 字体风格属性font-style

有三个值可选：**normal, italic, oblique**。**normal**是缺省值，**italic, oblique**都是斜体显示。如：

H2 {font-style:italic}



4.5 CSS的常用属性

4.5.1 字体属性

(4) 字重属性font-weight

常用值是normal和bold，normal是缺省值。如：

```
<p style = "font-weight:bold">
```

(5) 字体属性(font)

这个属性是各种字体属性的一种快捷的综合写法。如：

```
p {font:italic normal bold 11pt arial}
```

(6) 字体颜色(color)

字体颜色用CSS中的color属性来表示

(7) 文字修饰text-decoration

主要有text-decoration: underline;(下划线),
overline;(上划线) line-through;(删除线) blink;(闪烁)



4.5 CSS的常用属性

4.5.2 背景属性

(1) 背景颜色属性**background-color**
为HTML元素设定背景颜色

```
body {background-color:#99FF00;}
```

(2) 背景图片属性**background-image**
为HTML元素设定背景图片

```
<body style="background-  
image:url(background.jpg)">
```

(3) 背景重复属性**background-repeat**
决定背景图片是否重复。

```
body {background-image:url(background.jpg);  
background-repeat:repeat-x}
```



4.5 CSS的常用属性

4.5.2 背景属性

(4) 背景附着属性background-attachment

这个属性和background-image属性决定图片是跟随内容滚动，还是固定不动。如：

```
body { background-image:url(background.jpg);  
background-repeat:no-repeat;  
background-attachment:fixed }
```

(5) 背景位置属性background-position

和background-image属性决定背景图片的最初位置。如

```
body { background-image:url(background.jpg);  
background-repeat:no-repeat;  
background-position:30px 60px }
```

(6) 背景属性background

是设置背景相关属性的一种快捷的综合写法



4.5 CSS的常用属性

4.5.3 CSS边框属性

- ❖ (1) 边框风格属性**border-style**
- ❖ (2) 边框宽度属性**border-width**
- ❖ (3) 边框颜色属性**border-color**
- ❖ (4) 边框属性(**border**)
- ❖ (5) 单边边框属性

4.5 CSS的常用属性

4.5.4 CSS边距属性

边距属性是用来设置页面中一个元素所占空间的边缘到相邻元素之间的距离。

(1) 左边距属性(margin-left)

```
.m1 {margin-left:1cm}
```

(2) 右边距属性(margin-right)

```
.m2 {margin-right:1cm}
```

(3) 上边距属性(margin-top)

```
.m3 {margin-top:1cm}
```

(4) 下边距属性(margin-bottom)

```
.m4 {margin-bottom:1cm}
```

(5) 边距属性(margin)是设定边距宽度的一个快捷的综合写法

```
.m5 {margin:1cm}
```

```
.m6 {margin:1cm 2cm 3cm 4cm}
```



4.5 CSS的常用属性

4.5.5 CSS间隙属性

间隙属性(padding)是用来设置元素内容到元素边界的距离。

(1) 左间隙属性(padding-left)

`.p1 {padding-left:1cm}`

(2) 右间隙属性(padding-right)

`.p2 {padding-right:1cm}`

(3) 上间隙属性(padding-top)

`.p3 {padding-top:1cm}`

(4) 下间隙属性(margin-bottom)

`.p4 {padding-bottom:1cm}`

(5) 间隙属性(padding)

这个属性是设定间隙宽度的一个快捷的综合写法。

`.p5 {padding:1cm}`

`.p6 {padding:1cm 2cm 3cm 4cm}`



4.5 CSS的常用属性

4.5.6 CSS列表样式属性

(1) 列表样式类型属性list-style-type

这个属性用来设定列表项标记前图标的样式。取值：

disc (缺省值, 黑圆点), **circle** (空心圆点), **square** (小黑方块), **decimal** (数字排序)等

(2) 列表样式位置属性list-style-position

两个值: **outside**表示以列表项内容为准对齐; **inside** 表示以列表项标记为准对齐;

(3) 列表样式图片属性(list-style-image)

可以用图片来表示, 用列表样式图片属性list-style-image来设定图片。

```
ul {list-style-image: url(dot02.gif)}
```

(4) 列表样式属性list-style, 是设定列表样式的快捷综合写法。

```
ul {list-style:circle inside url(dot02.gif)}
```



4.5 CSS的常用属性

4.5.7 CSS锚伪类

锚伪类是最常用的伪类，主要有4个，其语法和CSS常用语法有些不同。如：

a:link {color: #FF0000} 未被访问的链接为红色

a:visited {color: #00FF00} 已被访问过的链接为绿色

a:hover {color: #FFCC00} 鼠标悬浮在上的链接为橙色

a:active {color: #0000FF} 鼠标点中激活链接时为蓝色

4.6 CSS布局

❖ **css**中的盒模型（**box model**）用于描述一个为**HTML**元素形成的矩形盒子。盒模型包括为各个元素调整外边距（**margin**）、边框（**border**）、内边距（**padding**）和内容的具体操作。

（1）**border**（边框）属性：●边框样式（**border-style**）●边框粗细（**border-width**）●边框颜色（**border-color**）

（2）**padding**（填充）

padding指内容距离边框的区域，如：**padding:20px;**

（3）**margin**（边界）

可设置四个方向边界，分别通过**margin-top**、**margin-right**、**margin-bottom**和**margin-left**来设置。



4.6 CSS布局

❖ 基于**CSS**的布局能提供更灵活的布局方式和更强的用户视觉体验。

(1) 创建div对象

<div class="center">此处显示新div标签的内容**</div>**

(2) 定位属性

定位属性**position**用于定义一个元素是否是**absolute**（绝对）、**relative**（相对）、**static**（静态）或者**fixed**（固定）。

语法: **position: static|absolute|fixed|relative**

(3) 浮动属性

用于控制对象的浮动布局方式。其语法如下:

Float: none | left | right| top| bottom



4.6 CSS布局

浮动属性示例:

```
#div1{  
height: 250px;  
width: 250px;  
background-color: #ff0000;  
float: left;  
}  
  
#div2{  
background-color: #ffff00;  
width: 300px;  
height: 18px;  
float: left; }
```



4.6 CSS布局

(1) 单列结构

```
#divt1{  
    height: 250px;  
    width:80%;  
    background-color:#FFCC00;  
    margin-top: 0px;  
    margin-right: auto;  
    margin-left: auto;  
}
```

这里就做出了一个高度为**250px**、位于顶部的**div**。

4.6 CSS布局

(2) 二列固定结构

```
#divleft{ float:left;
          height: 100px;
          width: 200px;
          border: 10px solid #CCFF00;
          background-color: #F2FAD1; }
#divright {      float:left;
                 height: 100px;
                 width: 200px;
                 border: 10px solid #00FFCC;
                 background-color: #FFFF00; }
```



4.6 CSS布局

(3) 二列自适应宽度

```
#divleft { margin:10px;
            float:left;
            height: 250px;
            width: 30%;
            border: 10px solid #ccff00;
            background-color: #f2aad1;}

#divright {      margin:10px;
               float:right;
               height: 250px;
               width: 50%;
               border: 10px solid #00ffcc;
               background-color: #ffff00; }
```



4.6 CSS布局

(4) 左列固定、右列自适应

```
#divleft { float:left;
            height: 250px;
            width: 20%;
            border: 10px solid #ccff00;
            margin: 10px;
            background-color: #f2fad1; }
#divright {      height: 250px;
                margin: 10px;
                border: 10px solid #ccff00;
                background-color: #ffff00; }
```



4.6 CSS布局

(5) visibility属性

```
<html>
<head>
<style type="text/css">
h1.visible {visibility:visible}
h1.invisible {visibility:hidden}
</style>
</head>
<body>
<h1 class="visible">这是可见的标题</h1>
<h1 class="invisible">这是不可见的标题</h1>
</body>
</html>
```



本章小结

CSS的引入为指定**HTML**文档的呈现细节提供了一种既统一又一致的方法，样式表分为三个层次：行内样式表、文档层样式表和外部样式表。优先级是行内样式表大于文档样式表，文档样式表大于外部样式表。

样式类是在**<style>**中的内容定义的，它能够使得同样的标签在不同的位置有不同的属性值，通过样式类说明能够使得不同名称的标签使用同样的样式。**CSS**中定义了许多种不同的属性值格式，包括长度值、百分比值、**URL**、颜色等。

在**CSS**布局中，**<div>**是一个常用的标记，利用这个标记，加上**CSS**对其样式的控制，可以很方便地实现各种效果，其中常用的一些属性包括**float**、**position**、**margin**、**padding**、**background**等。