

第 3 章

CSS(层叠样式表)

美轮美奂的网页,往往容易给人留下深刻印象,CSS 就是网页的装饰师。本章主要介绍 CSS 基本格式、应用方式、选择器、常见样式设置方法等内容。熟练掌握 CSS,是 Web 前端开发必备的技能。

3.1 CSS 简介

CSS 全称为 Cascading Style Sheets,层叠样式表,是万维网联盟 W3C 制定的一种标准。CSS 描述了如何在页面上显示 HTML 的各种元素,可以方便地控制网页的样式和布局。

CSS3 是 CSS 的升级版,从多个方面进行了改进和扩充。本章所介绍的内容里面,已经包含了 CSS3 的部分内容。

3.2 CSS 基础

3.2.1 CSS 基本格式

CSS 由选择器和声明两大部分组成,以下面的 CSS 为例:

```
<style type = "text/css">
  div {
    color: #00f;
    font-size:12px;
  }
</style>
```

- ◇ <style>: 为页面定义 CSS 样式信息。
- ◇ div: 选择器。选择器的意思就是选取需要设置样式的 HTML 元素,这里表示该 CSS 是对 div 层进行修饰。

◇ {}：大括号。定义了 CSS 声明块的内容。声明块包含了一条或多条用分号分隔的声明。这里包含了关于颜色、字体大小的两条声明。

◇ 声明。声明是具体用来对网页的各种元素进行修饰的部分。每条声明由“属性名：属性值”数据对组成，多个声明之间用冒号分隔，例如“color: #00f;”中 color 为属性名，#00f 为属性值。

上面示例的 CSS，控制页面中所有 div 层的文本颜色为蓝色，字体大小为 12px。

CSS 的颜色有下面三种表示方法。

(1) 用单词表示，例如 blue 表示蓝色，但单词只能表示有限数目的颜色。

(2) 采用十六进制表示，格式：#RGB。RGB 分别表示红、绿、蓝三原色，取值范围都是 00~FF，大小写均可，例如 #0000FF(蓝色，可简写为 #00F)、#F00(红色)、#7D7DFF(浅蓝色)。

(3) 用 RGB 函数表示，例如红色 rgb(255,0,0)。

3.2.2 应用方式

如果要用 CSS 进行修饰，一般有三种方式。

1. “标签名”作为选择器

```
<style type="text/css">
  div {
    width:125px;
    border: 1px dotted #00f;
  }
</style>
```

页面代码：

```
<span>观书有感</span>
<div>半亩方塘一鉴开</div>
<div>天光云影共徘徊</div>
```

这种方式下，样式会影响页面上所有 <div> 标签，即两句诗所在层都具有同样的样式：长度 125px，边框为蓝色点线，线宽 1px，但 不具有这种样式。

2. “# 自定义名”作为选择器

```
<style type="text/css">
  #poem1 {
    width:125px;
    border: 1px solid #f00;
  }
  #poem2 {
    width:125px;
    border: 1px dotted #00f;
  }
</style>
```

页面代码:

```
<span>观书有感</span>
<div id = "poem1">半亩方塘一鉴开</div>
<div id = "poem2">天光云影共徘徊</div>
```

这种方式下,样式会影响页面上所有 id 值与选择器名称相同的页面元素。所以,“半亩方塘一鉴开”的边框为红色实线、线宽 1px,而“天光云影共徘徊”的边框为蓝色点线、线宽 1px。

3. “.自定义名”作为选择器

```
<style type = "text/css">
  div {
    width: 130px;
    text-align: center;
    padding: 1px;
    background-color: #fff;
    display: table-cell;
  }
  img{
    vertical-align:middle;           ①
  }
  #title {
    color: #f00;
  }
  .poem {
    border: 1px dotted #00f;       ②
  }
  div.poem:nth-child(odd) {
    background-color: #e0e0f3;    ③
  }
  div.poem:nth-child(even) {
    background-color: #e8f1fb;    ④
  }
</style>
```

页面代码:

```
<div>
  <img src = "image/note.png" width = "16" height = "16">
  <span id = "title">观书有感</span>
  <img src = "image/note.png" width = "16" height = "16">
  <div class = "poem">半亩方塘一鉴开</div>
  <div class = "poem">
    <span>天光云影共徘徊</span>
  </div>
  <div class = "poem">问渠那得清如许</div>
  <div class = "poem">为有源头活水来</div>
</div>
```

说明：

- ① 图片与文字在垂直方向居中对齐。
 - ② 这种方式下,样式会影响页面上所有 class 值与选择器名称相同的页面元素。所以,四句诗的边框均为蓝色点线、线宽 1px,而诗名“观书有感”为红色字、无边框。
 - ③ class="poem"的所有<div>中,奇数位置的<div>背景色为 #e0e0f3。
 - ④ class="poem"的所有<div>中,偶数位置的<div>背景色为 #e8f1fb。
- 即③④样式设置了四句诗的奇偶行背景颜色的交替变化,如图 3-1 所示。

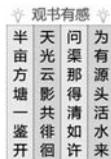


图 3-1 奇偶行变色

结合上面三种应用方式,再拓展介绍几种选择器用法:

- ◇ div.poe^m: 选择 class="poem"的所有<div>元素,例如 div.poe^m{font-size:12px;},下面几种与此类似。
- ◇ div span: 选择<div>内部的元素。
- ◇ img,span: 选择所有、元素。
- ◇ div[class^="poem"]: 选择 class 属性值以"poem"开头的<div>。
- ◇ img[src\$=".png"]: 选择 src 属性值以".png"结尾的。
- ◇ img[src ~="on"]: 选择 src 属性值中包含"on"的。
- ◇ img+span: 选择紧跟的首个。

3.3 CSS 样式设置

熟悉了 CSS 基本语法和应用方式后,就可以进一步学习 CSS 的各种样式设置。CSS 样式设置非常丰富,下面选择一些常用样式设置进行介绍。

3.3.1 文本

通过表 3-1 来学习一些常见的文本设置。

表 3-1 文本设置

名称	说明	举例(作用)
color	颜色	color:#00f;(蓝色文字)
font-family	字体	font-family:Wingdings;(文鼎字体)
font-style	字体样式	font-style:italic;(斜体字)
font-weight	字体粗细	font-weight:bold;(文字加粗)
font-size	字体大小	font-size:1em;(字体大小为 16 像素)
text-align	水平对齐方式	text-align:center;(水平方向居中)
vertical-align	垂直对齐方式	vertical-align:middle;(垂直方向居中)
text-decoration	装饰线	text-decoration:underline;(文字有下画线)
text-transform	文本转换	text-transform:uppercase;(转换为大写)
letter-spacing	字符间距	letter-spacing:-5px;(字符间距 5 像素)
word-spacing	单词间距	word-spacing:5px;(单词间距 5 像素)
line-height	行间距	line-height:1.5;(增大行间距)
text-shadow	文字阴影	text-shadow:1px 1px 1px #ccc;(灰色阴影)

示例：用红色加粗字体突出显示部分文字。

```
< style >
  p.msg span {
    font-weight: bold;
    color: #f00;
    font-size: 1.2em;
    text-shadow: 2px 2px 3px #ccc;
  }
</ style >
< div >
  < p class = "msg" >
    < span >友情提示: </ span >
    优惠活动截止到今天!
  </ p >
</ div >
```

这里仅设置“友情提示：”为红色、粗体、阴影效果，如图 3-2 所示。

友情提示：优惠活动截止到今天！

图 3-2 文本样式

3.3.2 背景

背景主要包括背景颜色、透明度、背景图片等。表 3-2 列出一些常见设置。

表 3-2 背景设置

名称	说明	举例(作用)
background-color	背景色	background-color: #0f0; (绿色背景)
opacity	透明度	opacity: 0.5; (半透明)
background-image	背景图像	background-image: url("image/bg.png"); (背景图像设置为 image 文件夹下的 bg.png)
background-repeat	重复图像	background-repeat: repeat-x; (水平方向拉伸图像) background-repeat: repeat-y; (垂直方向拉伸图像) background-repeat: no-repeat; (不拉伸图像)
background-position	图像位置	background-position: right top; (右上角)
background-attachment	背景图像固定方式	background-attachment: fixed; (固定图像) background-attachment: scroll; (可滚动图像)
background-size	图像大小	background-size: 60px 35px; (图像大小 60×35 像素)

示例：将上面的例子稍微修改一下，加入背景图片。

其他部分不变，只需要在< style >中加入：

```
div{
  width:272px;
  height:75px;
  background-image:url("image/bg.png");
  background-size:272px 75px;
}
```

运行效果如图 3-3 所示。

友情提示：优惠活动截止到今天！



图 3-3 背景图片

3.3.3 边框和边距

边框属性定义了边框的颜色、宽度、形状等,边距则定义了元素与边框的距离。表 3-3 列出了常见边框和边距属性。

表 3-3 边框和边距

名称	说明	举例(作用)
border-style	边框类型	border-style: 1px dotted #00f;(蓝色点线边框,线宽 1px。边框类型还可以是 solid 实线、dashed 虚线、double 双线、none 无边框等)
border-width	边框宽度	border-width: 1em 0.5em 1em 0.5em;(上右下左边框的宽度)
border-color	边框颜色	border-color: #f00 #00f #f00 #00f;(上右下左边框的颜色)
border-radius	圆角边框	border-radius:4px;(也可用百分数表示)
margin	边距	margin: 5px 3px 5px 3px;(上右下左边距,可以分别用 margin-top、margin-right、margin-bottom、margin-left 代替)

示例：将上例中的<div>加上边框“上、下边框为红色点线,左右无边框”。

只需要在<style>的 div 选择器中加入以下代码：

```
border:1px dotted;
border-width: 1px 0px;
border-color: #f00;
```

3.3.4 定位溢出和浮动

position 定位设置元素的位置或前后顺序,overflow 用于设置在内容较多无法容纳时的处理方式,float 则指明了页面元素如何浮动。表 3-4 列出了定位、溢出和浮动的常见用法。

表 3-4 定位、溢出和浮动

名称	说明	举例(作用)
position	定位	position:relative;(相对定位,常见的还有 fixed、absolute 等)
z-index	前后顺序	z-index: -1;(数值越大越靠前)
overflow	溢出	overflow: hidden;(隐藏超出内容,还可以设置为 scroll、auto 等)
float	浮动	float:left(处于左侧)

示例：将各自包含一张图片的两个不同的层<div>,分别放置在背景层的左上角、右上角,如图 3-4 所示。



图 3-4 定位

代码如下：

```
<style>
  div {
    margin: auto;                                ①
    width: 260px;
    height: 75px;
    background-image: url("image/bg.png");      ②
    background-size: 260px 75px;
  }
  .left, .right {
    position: relative;                          ③
    width: 28px;
    height: 28px;
  }
  .left {
    float: left;                                ④
  }
  .right {
    float: right;                               ⑤
  }
</style>
<div>
  <div class = "left">
    <img src = "image/db.png" width = "28" height = "28">
  </div>
  <div class = "right">
    <img src = "image/statistics.png" width = "28" height = "28">
  </div>
</div>
```

说明：

- ① 最外层<div>自动处于屏幕中央。
- ② 设置背景图片为 image/bg.png, 请使用 url() 函数指明图片位置。
- ③ 相同的样式放在一起定义, 位置定义为相对于最外层<div>。
- ④ 浮动在最外层<div>的左边。
- ⑤ 浮动在最外层<div>的右边。

3.3.5 伪类和伪元素

1. 伪类

伪类常用于定义页面元素的特殊状态, 例如鼠标悬停时改变颜色。表 3-5 列出一些常用的伪类。

表 3-5 伪类

名称	说明	举例(作用)
:active	活动的	a:active(活动链接)
:checked	被选中的	input:checked(被选中的复选框)
:disabled	被禁用的	button:disabled(被禁用的按钮)
:enabled	被启用的	button:enabled(被启用的按钮)
:focus	获得焦点的	input:focus(获得焦点的输入域)
:hover	鼠标悬停时	div:hover(鼠标悬停在层上面)
:link	未访问链接	a:link(未访问的链接)
:read-only	具有只读属性	input:readonly(只读的输入域)

示例：用伪类设置六个复选框的状态。当鼠标在复选框上面悬停时，变成红色实线圆角矩形；当勾选复选框后，变成蓝色点线圆角矩形。具体效果如图 3-5 所示。

代码如下：

```
<style>
  input:hover + span {
    border: 1px solid #f00;           ①
    border-radius: 4px;             ②
    color: #f00;
  }
  input:checked + span {
    border: 1px dotted #00f;        ③
    border-radius: 4px;
  }
</style>
<div>
  <input type="checkbox" name="fav" value="01"><span>足球</span>
  <input type="checkbox" name="fav" value="02"><span>游泳</span>
  <input type="checkbox" name="fav" value="03"><span>绘画</span>
  <input type="checkbox" name="fav" value="04"><span>舞蹈</span>
  <input type="checkbox" name="fav" value="05"><span>阅读</span>
  <input type="checkbox" name="fav" value="06"><span>拳击</span>
</div>
```

说明：

- ① 鼠标悬停在复选框上时，改变文字状态。
- ② 设置为圆角矩形边框。
- ③ 复选框被选中后，改变文字状态。

2. 伪元素

伪元素用于设置页面元素指定部分的样式，例如设置一段文字的首行为红色字体，或者设置<div>层中临时动态添加的文字等。表 3-6 列出了伪元素。

足球 游泳 绘画 舞蹈 阅读 拳击

图 3-5 复选框样式

表 3-6 伪元素

名称	说明	举例(作用)
::after	指定元素之后	<pre>span::after { content: url("image/m0.png"); }</pre> (在内部内容的末尾插入一张图片)
::before	指定元素之前	<pre>div::before { content: "欢迎光临!";}</pre> (在<div>内容前插入文字)
::first-letter	首字母	<pre>p::first-letter { color: #f00;}</pre> (段的首字母红色)
::first-line	首行	<pre>p::first-line { color: #f00;}</pre> (段的首行红色)
::selection	用户选择的内容	<pre>p::selection { color: #f00;}</pre> (用户选中的内容为红色)

示例：模拟微信的聊天框，如图 3-6 所示。

今天大家讨论得很热烈啊!

代码如下：

图 3-6 聊天框

```
< style >
  .chat_msg {
    position: relative;
    width: 150px;
    height: 22px;
    font-size: 0.6em;
    background: #9EEA6A;
    border-radius: 4px;
    padding: 2px;
  }
  .chat_msg::after {
    position: absolute;
    content: "";
    top: 100%;
    left: 99%;
    margin-top: -18px;
    border: 6px solid;
    border-color: transparent transparent transparent #9EEA6A;
  }
</ style >
< div class = "chat_msg">今天大家讨论得很热烈啊!</ div >
```

说明：

- ① 设置绿色背景聊天框的样式。
- ② 定义一个伪元素——一个小方块，修饰<div>层。
- ③ 伪元素的内容为空字符。
- ④ 伪元素边框——左边框为绿色实线，其他边透明。

3.3.6 多列

多列允许将文字拆分成几栏，类似于 Word 中的分栏效果。表 3-7 列出了常用的多列属性。

表 3-7 多列属性

名 称	说 明	举例(作用)
column-count	列数	column-count:3; (定义为 3 栏)
column-fill	填充方式	column-fill: balance auto; (平衡或顺序自动填充)
column-gap	列间隙	column-gap:10px; (列间隙为 10 像素)
column-rule	列规则	column-rule: 1px solid #f00; (列之间 1 像素、红色实线)
column-rule-color	列规则(颜色)	column-rule-color: #f00; (列颜色红色)
column-rule-style	列规则(样式)	column-rule-style: solid; (列样式实线)
column-rule-width	列规则(宽度)	column-rule-width: 1px; (列宽度 1 像素)
column-span	跨越的列数	column-span:all; (跨越全部列)
column-width	列宽	column-width:20px; (列宽 20 像素)

示例：将一段出师表古文，用三栏形式展示，如图 3-7 所示。

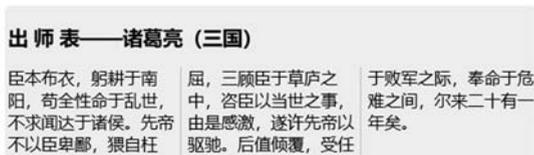


图 3-7 文字分栏

代码如下：

```
<style>
  .csb {
    margin: auto;
    width: 380px;
    font-size: 0.6em;
    column-count: 3;
    column-gap: 10px;
    column-rule: 1px solid #ccc;
    border-radius: 4px;
    background-color: #e6f3fc;
    padding: 2px;
    box-shadow: 2px 2px 2px #ccc;
  }
  h2 {
    column-span: all;
  }
</style>
<div class="csb">
  <h2>出师表——诸葛亮(三国)</h2>
  臣本布衣，躬耕于南阳，苟全性命于乱世，不求闻达于诸侯。先帝不以臣卑鄙，猥自枉屈，三顾臣于草庐之中，咨臣以当世之事，由是感激，遂许先帝以驱驰。后值倾覆，受任于败军之际，奉命于危难之间，尔来二十有一年矣。
</div>
```

3.3.7 动画

通过从一种样式变为另一种样式,CSS 也可实现一些基本动画。我们知道,动画是通过一些关键帧的连续播放来呈现的。CSS 提供了一个重要的属性@keyframes,通过它以及其他属性的组合,就可以实现动画效果。表 3-8 列出了一些常用动画属性。

表 3-8 动画属性

名称	说明	举例(作用)
@keyframes	动画帧	@keyframes mymove { from {background-color: #f00;} to {background-color: #00f;} } (颜色从红色变化为蓝色,也可以使用百分比)
animation-delay	延迟时间	animation-delay:2s(延迟 2s 开始)
animation-direction	播放方向	animation-direction: normal alternate;(正常播放或反向播放)
animation-duration	持续时长	animation-duration:2s;(持续 2s)
animation-iteration-count	播放次数	animation-iteration-count:infinite;(重复播放)
animation-name	动画名称	animation-name:mymove;(动画名称 mymove)
animation-timing-function	速度曲线	animation-timing-function:linear;(恒定速度。还可取值为:ease-in 低速开始、ease-out 低速结束、ease-in-out 低速开始和结束)

示例: 图片的路径动画。一张图片,放置在层<div>里面。动画轨迹:①→②→③→④→①。在运动过程中,层<div>旋转前进,且背景颜色发生变换,如图 3-8 所示。图 3-9 为动画过程中的某一帧画面。

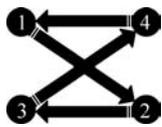


图 3-8 动画轨迹



图 3-9 旋转动画帧

代码如下:

```
< style >
    .move {
        width: 100px;
        height: 110px;
        background: #f00;
        position: relative;
        animation-duration:5s; /* 5s 完成一轮动画 */
        animation-name:moveimg; /* 动画名称 */
```

```
        animation-iteration-count: infinite; /* 循环播放 */
    }
    @keyframes moveimg { /* 定义 5 个关键帧 */
        0% {
            transform: rotate(0deg); /* 转换：旋转 0° */
            background: # f00;
            left: 0px;
            top: 0px;
        }
        25% {
            transform: rotate(90deg); /* 转换：旋转 90° */
            background: # d5d502;
            left: 300px;
            top: 300px;
        }
        50% {
            transform: rotate(180deg); /* 转换：旋转 180° */
            background: # 00f;
            left: 0px;
            top: 300px;
        }
        75% {
            transform: rotate(270deg); /* 转换：旋转 270° */
            background: # 0f0;
            left: 300px;
            top: 0px;
        }
        100% {
            transform: rotate(360deg); /* 还原到起点角度 */
            background: # f00;
            left: 0px;
            top: 0px;
        }
    }
}
</style>
<div class = "move"><img src = "image/m1.gif"></div>
```



视频讲解

3.4 场景应用示例——功能导航条

3.4.1 应用需求

京东购物网在购物页面的右边放置了一个功能导航条。综合前面所学知识，模拟制作一个水平放置的功能导航条，如图 3-10 所示。



图 3-10 功能导航条

具体要求:

(1) 当鼠标悬停在图标按钮上面时,颜色变为红色;

(2) “消息”图标按钮,右上角用红色小圆圈显示有 6 条新消息。这里只是模拟显示,我们将在第 13 章,真正实现消息推送;

(3) 单击图标按钮时,弹出显示文字的对话框,例如“查询……”。当然,这个弹出对话框功能只是示例性质,暂时没有任何实际意义,但体现了基本的处理思路,以后可以慢慢完善按钮单击功能的实际处理。

3.4.2 实现思路

首先定义一个层<div>,容纳整个工具栏图标按钮。其次,图标按钮使用<button>是最合适不过的了。最后,红色消息小圆圈,用层<div>表示,辅以 CSS 属性 border-radius 修饰,并定义好相对于父级<button>的位置即可。

3.4.3 CSS 代码

代码如下:

```
< style >
    .navi { /* 定义最外围层的样式 */
        margin: auto;
        width: 335px;
        height: 50px;
        font-size: 0.6em;
        text-align: center;
        column-count: 6;
        column-gap: 1px;
        column-rule-style: solid;
        column-rule-color: # fff;
    }
    button { /* 图标按钮的样式 */
        border: 0px;
        width: 55px;
        height: 50px;
        border-radius: 4px;
        background-color: # bfbbbb;
    }
    button:hover { /* 鼠标悬停在按钮上背景色变为红色 */
        background-color: # de2525;
    }
    .msg { /* 定义右上角的消息提示 */
        position: relative;
        left: - 4.8em;
        top: 6px;
        border-radius: 50%;
        text-align: center;
```

```
        display: inline-block;
        color: #fff;
        background-color: #f15555;
        width: 1.7em;
        height: 1.6em;
        line-height: 1.6em;
        font-size: 0.6em;
    }
</style>
<div class="navi">
    <button onclick="alert('新增.....')">
        <br/>新增
    </button>
    <button onclick="alert('删除.....')">
        <br/>删除
    </button>
    <button onclick="alert('查询.....')">
        <br/>查询
    </button>
    <button onclick="alert('畅聊.....')">
        <br/>畅聊
    </button>
    <button onclick="alert('报表.....')">
        <br/>报表
    </button>
    <button onclick="alert('消息.....')">
        <br/>消息
        <div class="msg">6 </div>
    </button>
</div>
```

上述代码只是在每个部分的第 1 句加了注释,希望读者认真对照以前所学知识,逐句研读,完全理解每句代码的含义!

3.5 场景任务挑战——导航菜单

制作学院导航菜单,如图 3-11 所示。“学院导航”背景为浅绿色,鼠标移动到其上时,弹出菜单项。当在菜单项中滑动鼠标时,当前菜单(例如“人工智能学院”)的背景色变为灰色。



图 3-11 导航菜单