

第 1 章 网页标记语言 HTML 入门基础

本章要点

随着网络的不断普及，网页已经被大多数人所熟悉。但是这些页面是怎样搭建起来的呢？又是怎样显示的呢？其实网页是由一种简单的标记语言 HTML 构成的。HTML 语言是组成网页的基本语言，它是一切网页制作的基础。如果能够熟练掌握并应用 HTML 代码，大到做网站，小到做个人网页等都会有很大的好处。这就需要对 HTML 有个基本的了解。因此具备一定的 HTML 语言的基本知识是必要的。本章主要内容包括：

- (1) 什么是 HTML；
- (2) HTML 文件的构成；
- (3) HTML 文件的编写方法；
- (4) 网页设计与开发的过程。

1.1 HTML 概述

HTML 的英文全称是 HyperText Markup Language，中文通常称作超文本标记语言或超文本标签语言。HTML 是 Internet 上用于编写网页的主要语言，它提供了精简而有力的文件定义，可以设计出多姿多彩的超媒体文件。借助 HTTP 通信协议，HTML 文件可以在全球互联网(World Wide Web，也称万维网)上进行跨平台的文件交换。

HTML 文件为纯文本的文件格式，可以用任何的文本编辑器或者使用 FrontPage、Dreamweaver 等网页制作工具来编辑。至于文件中的文字、字体、字体大小、段落、图片、表格及超链接，甚至是文件名称都是用不同意义的标签来描述的，以此来定义文件的结构与文件间的逻辑关联。简而言之，HTML 是以标签来描述文件中的多媒体信息的。

1. HTML 的特点

HTML 文档制作简单，且功能强大，支持不同数据格式的文件导入，这也是 WWW 盛行的原因之一，其主要特点如下。

- (1) HTML 文档容易创建。只需要一个文本编辑器就可以完成。
- (2) HTML 文件所占存储空间小。能够尽可能快地在网络环境下传输与显示。
- (3) 平台无关性。HTML 独立于操作系统平台，它能对多平台兼容，只需要一个浏览器，就能够在操作系统中浏览网页文件。可以使用在广泛的平台上，这也是 WWW 盛行的另一个原因。
- (4) 容易学习。不需要很深的编程功底。
- (5) 可扩展性。HTML 语言的广泛应用带来了加强功能，增加标识符等要求，HTML 采取子类元素的方式，为系统扩展带来保证。



2. HTML 的历史

HTML 1.0 —— 1993 年 6 月, 互联网工程工作小组(IETF)工作草案发布。

HTML 2.0 —— 1995 年 11 月发布。

HTML 3.2 —— 1996 年 1 月 W3C 推荐标准。

HTML 4.0 —— 1997 年 12 月 W3C 推荐标准。

HTML 4.01 —— 1999 年 12 月 W3C 推荐标准。

HTML 5.0 —— 2008 年 8 月 W3C 工作草案。

1.2 HTML 文件的构成

编写 HTML 文件时, 必须遵循一定的语法规则。一个完整的 HTML 文件由标题、段落、表格、文本等各种嵌入的对象组成, 这些对象统称为元素。HTML 使用标记(也称标签)来分隔并描述这些元素。整个 HTML 文件其实就是由元素与标记组成的。

1.2.1 HTML 文件结构

HTML 的任何标记都用“<”和“>”围起来, 如<HTML>。在起始标记的标记名前加上符号“/”便是其终止标记, 如</HTML>, 夹在起始标记和终止标记之间的内容受标记的控制。超文本文档分为头部和主体两部分, 在文档头部, 对文档进行了一些必要的定义, 而文档主体则包含了要显示的各种文档信息。

基本语法:

```
<html>  
<head>网页头部信息</head>  
<body>网页主体正文部分</body>  
</html>
```

语法说明:

其中<html>在最外层, 表示这对标记间的内容是 HTML 文档。一个 HTML 文档总是以<html>开始, 以</html>结束。<head>之间包括文档的头部信息, 如文档标题等, 若不需头部信息则可省略此标记。<body>标记一般不能省略, 表示正文内容的开始。

下面就以一个简单的 HTML 文件来熟悉 HTML 文件的结构。代码如下:

实例代码:

```
<!doctype html>  
<html>  
<head>  
<meta charset="utf-8">  
<title>简单的 HTML 文件结构</title>  
</head>  
<body>  
<p>HTML 文件的基本结构!  
</p>
```

```
</body>
</html>
```

这一段代码是使用 HTML 中最基本的几个标记所组成的，运行代码，在浏览器中预览效果，如图 1-1 所示。



图 1-1 HTML 文件结构

下面解释一下上面的例子。

- (1) HTML 文件就是一个文本文件。文本文件的后缀名是.txt，而 HTML 的后缀名是.html。
- (2) 在 HTML 文档中，第一个标记是<html>，这个标记告诉浏览器这是 HTML 文档的开始。
- (3) HTML 文档的最后一个标记是</html>，这个标记告诉浏览器这是 HTML 文档的终止。
- (4) 在<head>和</head>标记之间的文本是头信息，在浏览器窗口中，头部信息是不被显示在页面上的。
- (5) 在<title>和</title>标记之间的文本是文档标题，它被显示在浏览器窗口的标题栏。
- (6) 在<body>和</body>标记之间的文本是正文，会被显示在浏览器中。
- (7) <p>和</p>标记代表段落。

1.2.2 编写 HTML 文件的注意事项

HTML 是由标记和属性构成的。在编写文件时，要注意以下几点。

- (1) “<”和“>”是任何标记的开始和结束。元素的标记要用这对尖括号括起来，并且在结束标记的前面加一个“/”斜杠，如<table></table>。
- (2) 在源代码中不区分大小写。
- (3) 任何回车和空格在源代码中均不起作用。为了代码的清晰，建议不同的标记之间用回车进行换行。
- (4) 在 HTML 标记中可以放置各种属性，如：

```
<h1 align="right">2017 年北京欢迎您</h1>
```

其中 align 为 h1 的属性，right 为属性值，元素属性出现在元素的<>内，并且和元素名之间有一个空格分隔，属性值可以直接书写，也可以使用" "括起来，如下两种写法都是正



确的。

```
<h1 align="right">2017 年北京欢迎您</h1>  
<h1 align=right>2017 年北京欢迎您</h1>
```

(5) 要正确输入标记。输入标记时，不要输入多余的空格，否则浏览器可能无法识别这个标记，导致无法正确地显示信息。

(6) 在 HTML 源代码中注释。<!--要注释的内容-->注释语句只出现在源代码中，不会在浏览器中显示。

1.3 怎样编写 HTML 文件


由于 HTML 语言编写的文件是标准的 ASCII 文本文件，因此可以使用任意一个文本编辑器来打开并编写 HTML 文件，如 Windows 系统中自带的“记事本”程序。如果使用 Dreamweaver、FrontPage 等软件，则能以可视化的方式进行网页的编辑制作等。

1.3.1 使用记事本文件编写页面


HTML 是一个以文字为基础的语言，并不需要什么特殊的开发环境，可以直接在 Windows 系统自带的“记事本”程序中编写。HTML 文档以.html 为扩展名，将 HTML 源代码输入到记事本文件中并保存，可以在浏览器中打开文档以查看其效果。使用记事本文件手工编写 HTML 页面的具体操作步骤如下。

(1) 在 Windows 系统中，运行“记事本”程序，在记事本文件中输入以下代码(见图 1-2):

```
<!doctype html>  
<html>  
<head>  
<meta charset="utf-8">  
<title>使用记事本编写 HTML 页面</title>  
</head>  
<body>  
  
</body>  
</html>
```

 **说明：** 新建记事本很简单，在电脑桌面上或者磁盘中空白地方单击鼠标右键(即右击)，在弹出的快捷菜单中选择“新建”|“文本文档”命令。

(2) 当编辑完 HTML 文件后，选择“文件”|“另存为”菜单命令，弹出“另存为”对话框，将它存为扩展名为.htm 或.html 的文件即可，如图 1-3 所示。

 **说明：** 注意是“另存为”命令，而不是“保存”命令，因为如果选择“保存”命令的话，Windows 系统会默认地把它存为.txt 记事本文件。.html 是个扩展名，注意前面是个点，而不是句号。

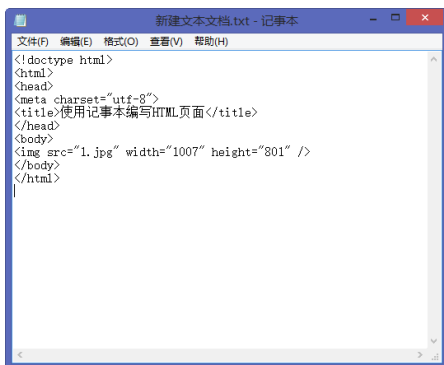


图 1-2 在记事本文件中输入代码

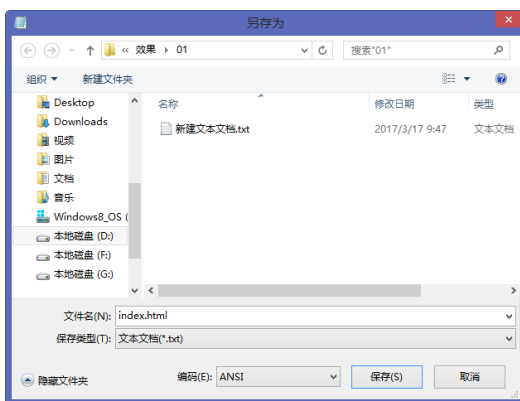


图 1-3 保存文件

(3) 单击“保存”按钮，这时该文本文件就变成了 HTML 文件，在浏览器中浏览效果，如图 1-4 所示。



图 1-4 浏览网页效果

1.3.2 使用 Dreamweaver 编写 HTML 页面

在 Dreamweaver CC “代码视图”中可以查看或编辑源代码。为了方便手工编写代码，Dreamweaver CC 增加了标签选择器和标签编辑器。使用标签选择器，可以在网页代码中插入新的标签；使用标签编辑器，可以对网页代码中的标签进行编辑，如添加标签的属性或修改属性值。在 Dreamweaver 中编写代码的具体操作步骤如下。

(1) 打开 Dreamweaver CC 软件，新建空白文档，在“代码视图”中编写 HTML 代码，如图 1-5 所示。

(2) 在 Dreamweaver 中编辑完代码后，返回到“设计视图”中，效果如图 1-6 所示。

(3) 选择“文件”|“保存”菜单命令，保存文档，即可完成 HTML 文件的编写。

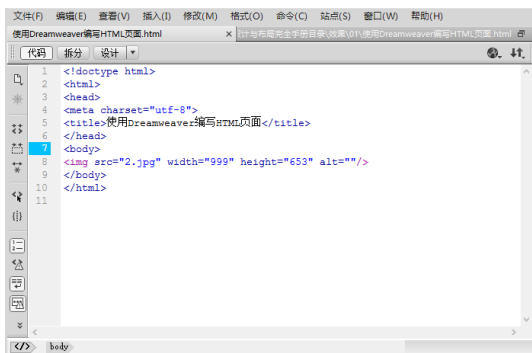


图 1-5 编写 HTML 代码



图 1-6 设计视图

1.4 网页设计与开发的过程

创建完整的网站是一个系统工程，有一定的工作流程，只有遵循这个步骤，按部就班地来，才能设计出满意的网站。因此，在设计网页前，先要了解网页设计与开发的基本流程，这样才能制作出更好、更合理的网站。

1.4.1 明确网站定位

在创建网站时，确定站点的目标是第一步。设计者应清楚建立站点的目标定位，即确定它将提供什么样的服务，网页中应该提供哪些内容等。要确定站点目标定位，通常应该从以下 3 个方面考虑。

1. 网站的整体定位

网站可以是大型商用网站、小型电子商务网站、门户网站、个人主页、科研网站、交流平台、公司和企业介绍性网站、服务性网站等。首先应该对网站的整体进行一个客观的评估，同时要以发展的眼光看待问题，否则将带来许多升级和更新方面的不便。

2. 网站的主要内容

如果是综合性网站，那么对于新闻、邮件、电子商务、论坛等都要有所涉及，这样就要求网页要结构紧凑、美观大方。对于侧重某一方面的网站，如书籍网站、游戏网站、音乐网站等，则往往对网页美工要求较高，使用模板较多，更新网页和数据库较快。如果是个人主页或介绍性的网站，那么一般来讲，网站的更新速度较慢，浏览率较低，并且由于链接较少，内容不如其他网站丰富，但对美工的要求更高一些，可以使用较鲜艳明亮的颜色，同时可以添加 Flash 动画等，使网页更具动感和充满活力，否则网站没有吸引力。

3. 网站浏览者的教育程度

对于不同的浏览者群体，网站的吸引力是截然不同的，例如：针对少年儿童的网站，



卡通和科普性的内容更符合浏览者的兴趣，也能够达到网站寓教于乐的目的；针对学生的网站，往往对网站的动感程度和特效技术要求更高一些；对于商务浏览者，网站的安全性和易用性更为重要。

1.4.2 收集信息和素材

首先要创建一个新的总目录(文件夹)，比如“D:\我的网站”，来放置建立网站的所有文件，然后在这个目录下建立两个子目录：“文字资料”和“图片资料”。放入目录中的文件名最好全部用英文小写，因为有些主机不支持大写和中文，以后增加的内容可再创建子目录。

1. 文本内容素材的收集

具体的文本内容，可以让访问者清楚地明白作者的 Web 页中想要说明的东西。我们可以从网络、书本、报刊上找到需要的文字材料，也可以使用平时的试卷和复习资料，还可以自己编写有关的文字材料，将这些素材制作成 Word 文档保存在“文字资料”子目录下。收集的文本素材既要丰富，又要便于有机地组织，这样才能做出内容丰富、整体感强的网站。

2. 艺术内容素材的收集

只有文本内容的网站对于访问者来讲，是枯燥乏味、缺乏生机的。如果加上艺术内容素材，如静态图片、动态图像、音像等，将使网页充满动感与生机，也将吸引更多访问者。这些素材主要来自以下 4 个方面。

(1) 从 Internet 上获取。可以充分利用网上的共享资源，可使用百度、雅虎等引擎收集图片素材。

(2) 从光盘中获取。在市面上，有许多关于图片素材库的光盘，也有许多教学软件，可以选取其中的图片资料。

(3) 利用现成图片或自己拍摄。既可以从各种图书出版物(如科普读物、教科书、杂志封面、摄影集、摄影杂志等)获取图片，也可以使用自己拍摄和积累的照片资料。将杂志的封面彩图用彩色扫描仪扫描下来，经过加工后，整合制作到网页中。

(4) 自己动手制作一些特殊效果的图片，特别是动态图像，自己动手制作往往效果更好。可采用 3ds Max 或 Flash 进行制作。

鉴于网上只能支持几种图片格式，所以可先将通过以上途径收集的图片用 Photoshop 等图像处理工具转换成 JPG、GIF 形式，再保存到“图片资料”子目录下。另外，图片应尽量精美而小巧，不要盲目追求大而全，要以在网页的美观与网络的速度两者之间取得良好的平衡为宜。

1.4.3 规划栏目结构

合理地组织站点结构，能够加快对站点的设计，提高工作效率，节省工作时间。当需要创建一个大型网站时，如果将所有网页都存储在一个目录下，当站点的规模越来越大时，管理起来就会变得很困难。因此，合理地使用文件夹管理文档就显得很重要。

网站的目录是指在创建网站时建立的目录。要根据网站的主题和内容来分类规划，不同的栏目对应不同的目录。在各个栏目目录下也要根据内容的不同对其划分不同的分目录，如页面图片放在 `images` 目录下，新闻放在 `news` 目录下，数据库放在 `database` 目录下等。同时，要注意目录的层次不宜太深，一般不要超过 3 层。另外，给目录起名的时候要尽量使用能表达目录内容的英文或汉语拼音。这样会更加方便日后的管理和维护。如图 1-7 所示，这是企业网站的站点结构。

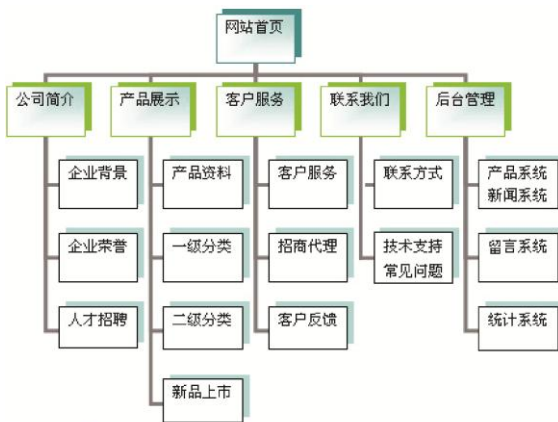


图 1-7 企业网站的站点结构

1.4.4 设计页面方案

在规划好网站的栏目结构和搜集完资料后就需设计网页图像了。网页图像设计包括 Logo、标准色彩、标准字、导航条、首页布局等。可以使用 Photoshop 或 Fireworks 软件来具体设计网站的图像。有经验的网页设计者，通常会在使用网页制作工具制作网页之前，设计好网页的整体布局，这样在具体设计过程将会胸有成竹，大大节省工作时间。如图 1-8 所示，这是设计的网页整体图像。



图 1-8 设计网页图像

1.4.5 制作页面

具体到每一个页面的制作时，首先要做的就是设计版面布局。就像传统的报纸杂志一样，需要将网页看作一张报纸、一本杂志来进行排版布局。

版面指的是在浏览器中看到的完整的一个页面的大小。因为每个人的显示器分辨率不同，所以同一个页面的大小(下面数值均以像素(px)为单位)可能出现 640×480、800×600 或 1024×768 等不同尺寸。目前主要以 1024×768 分辨率的用户为主。在实际制作网页时，应将网页内容宽度限制在 778 以内(可以用表格或层来进行限制)，这样在用 1024×768 分辨率的显示器进行浏览时，除去浏览器左右的边框后，刚好能完全显示出网页的内容。

布局就是以最适合浏览的方式将图片和文字排放在页面的不同位置。这是一个创意的过程，需要一定的经验，当然也可以参考一些优秀的网站来寻求灵感。

版面布局完成后，就可以着手制作每一个页面了。通常都从首页做起，制作过程中可以先使用表格或层对页面进行整体布局，然后将需要添加的内容分别添加到相应的单元格中，并随时预览效果进行调整，直到整个页面完成并达到理想的效果。接下来使用相同的方法完成整个网站中其他页面的制作。

网页制作是一个复杂而细致的过程，一定要按照先大后小、先简单后复杂的顺序制作。所谓先大后小，就是在制作网页时，先把大的结构设计好，然后再逐步完善小的结构设计。所谓先简单后复杂，就是先设计出简单的内容，然后再设计复杂的内容，以便出现问题时好修改。在制作网页时要灵活运用模板和库，这样可以大大提高制作效率。如果很多网页都使用相同的版面设计，就应为这个版面设计一个模板，然后就可以以此模板为基础创建网页。以后如果想要改变所有网页的版面设计，只需要简单地改变模板即可。图 1-9 所示为制作的网页。



图 1-9 制作的网页



1.4.6 实现后台功能

页面设计制作完成后, 如果还需要动态功能的话, 就需要开发动态功能模块, 网站中常用的功能模块有搜索功能、留言板、新闻信息发布、在线购物、技术统计、论坛及聊天室等。

1. 留言板

留言板、论坛及聊天室是为浏览者提供信息交流的地方。浏览者可以围绕个别的产品、服务或其他话题进行讨论。顾客也可以提出问题、提出咨询, 或者得到售后服务。但是聊天室和论坛是比较占用资源的, 一般不是大中型的网站没有必要建设论坛和聊天室。如果访问量不是很大的话, 做好了也没有人来访问。图 1-10 所示为留言板页面。



图 1-10 留言板页面

2. 搜索功能

搜索功能是为了帮助浏览者在短时间内快速地从大量的资料中找到符合要求的资料。这对于资料非常丰富的网站来说非常有用。要建立一个搜索功能, 就要有相应的程序以及完善的数据库支持, 可以快速地从数据库中搜索到所需要的数据。

3. 新闻发布管理系统

新闻发布管理系统提供方便直观的页面文字信息的更新维护界面, 提高工作效率、降低技术要求, 非常适合用于经常更新的栏目或页面。图 1-11 所示为新闻发布管理系统。

4. 购物网站

购物网站是实现电子交易的基础。用户将感兴趣的产品放入自己的购物车, 以便最后统一结账。当然用户也可以修改购物的数量, 甚至将产品从购物车中取出。用户选择结算后系统自动生成本系统的订单。图 1-12 所示为一个购物网站。



图 1-11 新闻发布管理系统



图 1-12 购物网站

1.4.7 网站的测试与发布

在将网站的内容上传到服务器之前，应先在本地区点进行完整的测试，以保证页面外观和效果、链接和页面下载时间等与设计相同。站点测试主要包括检测站点在各种浏览器中的兼容性，检测站点中是否有断掉的链接。用户可以使用不同类型和不同版本的浏览器预览站点中的网页，检查可能存在的问题。

在完成了对站点中页面的制作后，就应该将其发布到 Internet 上供大家浏览和观赏了。但是在此之前，应该对所创建的站点进行测试，对站点中的文件逐一进行检查，在本地计算机中调试网页以防止包含在网页中的错误，以便尽早发现问题并解决问题。

在测试站点过程中应该注意以下几个方面。

(1) 在测试站点过程中应确保在目标浏览器中，网页如预期的显示和工作，没有损坏的链接，以及下载时间不宜过长等。

(2) 了解各种浏览器对 Web 页面的支持程度，不同的浏览器观看同一个 Web 页面，可能会有不同的效果。很多制作的特殊效果，在有些浏览器中可能看不到，为此需要进行浏览器兼容性检测，以找出不被其他浏览器支持的部分。

(3) 检查链接的正确性，可以通过 Dreamweaver 提供的检查链接功能来检查文件或站点中的内部链接及孤立文件。

网站的域名和空间申请完毕后，就可以上传网站了，可以采用 Dreamweaver 自带的站点管理上传文件。

1.5 本章小结

HTML 是目前网络上应用最为广泛的语言，也是构成网页文档的基本语言。本章介绍了 HTML 的基本概念、编写方法和 HTML 页面基本标记以及网页设计与开发的基本流程。



通过本章的学习，读者能够对 HTML 有个初步的了解，从而为后面设计制作更复杂的网页打下良好的基础。

1.6 练 习 题

1. 填空题

(1) 一个 HTML 文档总是以_____开始，以_____结束。_____之间包括文档的头部信息，如文档标题等，若不需要头部信息则可省略此标记。_____标记一般不能省略，表示正文内容的开始。

(2) 由于 HTML 语言编写的文件是标准的 ASCII 文本文件，因此可以使用任意一个文本编辑器来打开并编写 HTML 文件，如 Windows 系统中自带的_____。如果使用_____、_____等软件，则能以可视化的方式进行网页的编辑制作等。

2. 操作题

(1) 用 IE 浏览器打开网上的任意一个网页，选择“查看”|“源文件”菜单命令，在打开的记事本文件中查看各代码，并试着与浏览器中的内容进行对照。

(2) 分别利用“记事本”程序和 Dreamweaver 创建一个简单的 HTML 网页。

第 2 章 HTML 网页文档的结构

本章要点

本章就来讲解这些基本标记的使用，它们是一个完整的网页必不可少的。掌握这些页面的基本元素是定义 HTML 页面的关键，通过它们可以了解网页的基本结构及工作原理。本章主要内容包括：

- (1) HTML 文档主体标记；
- (2) 头部标记。

2.1 HTML 文档主体标记

在<body>和</body>之间放置的是页面中所有的内容，如图片、文字、表格、表单、超链接等设置。<body>标记有自己的属性，包括网页的背景设置、文字属性设置、链接设置等。设置<body>标记内的属性，可以控制整个页面的显示方式。

2.1.1 bgcolor 属性

对大多数浏览器而言，其默认的背景颜色为白色或灰白色。在网页设计中，bgcolor 属性定义整个 HTML 文档的背景颜色。

基本语法：

```
<body bgcolor="背景颜色">
```

语法说明：

背景颜色有以下两种表示方法。

- (1) 使用颜色名指定，如红色、绿色等分别用 red、green 等表示。
- (2) 使用十六进制格式数据值#RRGGBB 来表示，RR、GG、BB 分别表示颜色中的红、绿、蓝三基色的两位十六进制数据。

实例代码：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>网页背景色</title>
</head>
<body bgcolor="#0446B9">
</body>
</html>
```



在上述代码中，加粗部分的代码表示为页面设置背景颜色，在浏览器中预览效果，如图 2-1 所示。

背景颜色在网页上很常见，如图 2-2 所示的网页使用了大面积的蓝色背景。



图 2-1 设置页面的背景颜色



图 2-2 使用背景颜色的网页

2.1.2 background 属性

网页的背景图片可以衬托网页的显示效果，从而取得更好的视觉效果。背景图片的选择不仅要注重好看，而且还要注意不要“喧宾夺主”，影响网页内容的阅读。通常使用深色的背景图片配合浅色的文本，或者是浅色的背景图片配合深色的文本。**background** 属性用来设置 HTML 网页的背景图片。

基本语法：

```
<body background="图片的地址">
```

语法说明：

background 属性值就是背景图片的路径和文件名。图片的地址可以是相对地址，也可以是绝对地址。

实例代码：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>设置背景图片</title>
</head>
<body background="047.jpg">
</body>
</html>
```

在上述代码中，加粗部分的代码表示为网页设置背景图片，在浏览器中预览可以看到背景图像，如图 2-3 所示。

在网络上除了可以看到各种带有背景色的页面之外，还可以看到一些以图片作为背景的网页。如图 2-4 所示的网页使用了背景图像。



图 2-3 页面的背景图像



图 2-4 使用了背景图像

- 提示：**
- ① 网页中可以使用图片做背景，但图片一定要与插图以及文字的颜色相协调，才能达到美观的效果，如果色差太大则会使网页失去美感。
 - ② 为保证浏览器载入网页的速度，建议尽量不要使用字节过大的图片作为背景图片。

2.1.3 text 属性

通过 text 属性可以设置 body(主体)内所有文本的颜色。在没有对文字的颜色进行单独定义时，这一属性可以对页面中所有的文字起作用。

基本语法：

```
<body text="文字的颜色">
```

语法说明：

在该语法中，text 的属性值与设置页面背景色的相同。

实例代码：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>设置文字颜色</title>
</head>
<body text="#FF0000">
<p>人人都喜欢花草的围绕，行走在花草间，心情飞扬，梦色彩而缤纷。可见美让人眷恋也让心温暖。走进自然尽情把心情释放，让生命不再有忧烦的围绕，让生命增添一些春光，每天欣然迎着风雨，每天静静的如荷绽放。</p>
</body>
</html>
```

在上述代码中，加粗部分的代码表示为文字设置颜色，在浏览器中预览可以看到文档中文字的颜色，如图 2-5 所示。

在网页中需要根据网页整体色彩的搭配来设置文字的颜色。如图 2-6 所示的文字和整个网页的颜色相协调。

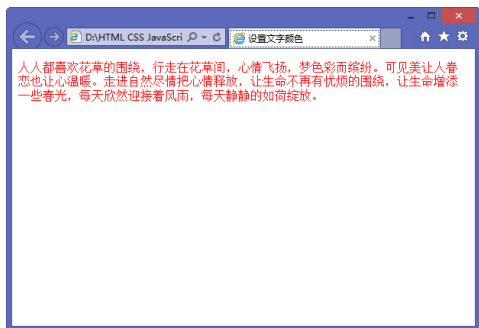


图 2-5 设置文字的颜色



图 2-6 文字的颜色

2.1.4 设置链接文字属性

为了突出超链接，超链接文字通常采用与其他文字不同的颜色，而且超链接文字的下端还会加一条横线。网页的超链接文字有默认的颜色。在默认情况下，浏览器以蓝色作为超链接文字的颜色，而访问过的文字则变为暗红色。在<body>标记中也可自定义这些颜色。

基本语法：

```
<body link="颜色">
```

语法说明：

这一属性的设置与前面几个设置颜色的参数类似，都是与 body 标记放置在一起，表明它对网页中所有未单独设置的元素起作用。

实例代码：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>设置链接文字的颜色</title>
</head>
<body link="#9933ff">
<center>
<p><a href="#">公司简介</a></p>
<p><a href="#">产品展示</a></p>
<p><a href="#">联系我们</a></p>
</center>
</body>
</html>
```

在上述代码中，加粗部分的代码表示为链接文字设置颜色，在浏览器中预览效果，可以看到链接的文字已经不是默认的蓝色，如图 2-7 所示。

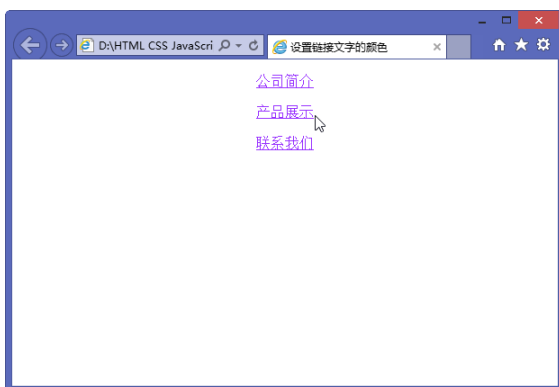


图 2-7 设置链接文字的颜色

使用 `alink` 可以设置当鼠标单击超链接时的颜色，举例如下：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>链接文字的颜色</title>
</head>
<body link="#9933ff" alink="#0066FF">
<center>
<p><a href="#">公司简介</a></p>
<p><a href="#">产品展示</a></p>
<p><a href="#">联系我们</a></p>
</center>
</body>
</html>
```

在上述代码中，加粗部分的代码表示为链接的文字设置单击时的颜色，在浏览器中预览效果，可以看到单击链接的文字，文字已经改变了颜色，如图 2-8 所示。



图 2-8 单击链接文字时的颜色



使用 vlink 可以设置已访问过的超链接颜色，举例如下：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>无标题文档</title>
</head>
<body link="#9933ff" alink="#0066FF" vlink="#FF0000">
<center>
<p><a href="#">公司简介</a></p>
<p><a href="#">产品展示</a></p>
<p><a href="#">联系我们</a></p>
</center>
</body>
</html>
```

在上述代码中，加粗部分的代码表示为链接文字设置访问后的颜色，在浏览器中预览效果，可以看到单击链接后文字的颜色已经发生改变，如图 2-9 所示。



图 2-9 访问后的链接文字的颜色

在网页中，一般文字上的超链接都是蓝色(当然，也可以自己设置成其他颜色)，同时文字下面有一条下划线。当移动鼠标指针到该超链接上时，鼠标指针就会变成一只手的形状，此时用鼠标左键单击，就可以直接跳到与这个超链接相连接的网页。如果已经浏览过某个超链接，这个超链接的文本颜色就会发生改变。图 2-10 所示为网页中的超链接文字颜色。



图 2-10 网页中的超链接文字颜色



2.1.5 设置页面边距

有的朋友在做页面的时候，感觉文字或者表格怎么也不能靠在浏览器的最上边和最左边，这是怎么回事呢？因为一般用的制作软件或 HTML 语言默认的都是 `topmargin`、`leftmargin` 值等于 12，如果你把它们的价值设为 0，就会看到网页的元素与左边距离为 0 了。

基本语法：

```
<body topmargin=value leftmargin=value rightmargin=value bottommargin=value>
```

语法说明：

通过设置 `topmargin/leftmargin/rightmargin/bottommargin` 不同的属性值来设置显示内容与浏览器的距离。在默认情况下，边距的值以像素为单位。

- (1) `topmargin` 设置到顶端的距离。
- (2) `leftmargin` 设置到左边的距离。
- (3) `rightmargin` 设置到右边的距离。
- (4) `bottommargin` 设置到底边的距离。

实例代码：


```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>设置边距</title>
</head>
<body topmargin="80" leftmargin="80">
<p>页面上边距</p>
<p>页面左边距</p>
</body>
</html>
```

在上述代码中，加粗部分的代码表示设置上边距和左边距，在浏览器中预览效果，可以看出定义的边距效果，如图 2-11 所示。



图 2-11 设置的边距效果



 **提示：** 一般网站的页面左边距和上边距都设置为 0，这样看起来页面不会有太多的空白。

2.2 head 部分的标记

HTML 中的 head 标记是网页标记中一个非常重要的符号。head 标记中包含的内容基本上描述了所属页面的基本属性，包括标题、字符集、站点信息、网站作者信息、站点描述、站点关键字、刷新及跳转、样式表链入以及其他一些有用的附加功能。做好 head 标记中的内容对整个页面有着非常重要的意义。下面介绍 head 标记中比较常用的一些东西。

2.2.1 title 标记

HTML 页面的标题一般是用来说明页面的用途，它显示在浏览器的标题栏中。在 HTML 文档中，标题信息设置在<head>与</head>之间。标题标记以<title>开始，以</title>结束。

基本语法：


```
<head>
<title>...</title>
...</head>
```

语法说明：

页面的标题只有一个，它位于 HTML 文档的头部，即<head>和</head>之间。

实例代码：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>标题标记 title</title>
</head>
<body>
</body>
</html>
```

 **提示：** 如何设置网站标题呢？首先应该明确网站的定位，希望对哪类词感兴趣的用户能够通过搜索引擎来到我们的站点，在经过关键字调研之后，选择几个能带来不菲流量的关键字，然后把最具代表性的关键字放在 title 的最前面。

2.2.2 定义页面关键字

在搜索引擎中，检索信息都是通过输入关键字来实现的。关键字是整个网站登录过程中最基本也是最重要的一步，是进行网页优化的基础。关键字在浏览时是看不到的，它可

供搜索引擎使用。当用关键字搜索网站时，如果网页中包含该关键字，就可以在搜索结果中列出来。

基本语法：

```
<meta name="keywords" content="输入具体的关键字">
```

语法说明：

在该语法中，name 为属性名称，这里是 keywords，也就是设置网页的关键字属性，而在 content 中则定义具体的关键字。

实例代码：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<meta name="keywords" content="插入关键字">
<title>插入关键字</title>
</head>
<body>
</body>
</html>
```

在上述代码中，加粗的代码表示设置网页关键字。



- 提示：**
- ① 要选择与网站或页面主题相关的文字。
 - ② 选择具体的词语，别寄望于行业或笼统的词语。
 - ③ 揣摩用户会用什么作为搜索词，把这些词放在页面上或直接作为关键字。
 - ④ 关键字可以不止一个，最好根据不同的页面，制定不同的关键字组合，这样页面被搜索到的概率将大大增加。

2.2.3 定义页面描述

描述的英文是 description，网页的描述属性为搜索引擎提供了关于这个网页的总括性描述。网页的描述元标记是由一两个语句或段落组成的，内容一定要具有相关性，描述不能太短、太长或过分重复。

基本语法：

```
<meta name="description" content="设置页面描述">
```

语法说明：

在该语法中，name 为属性名称，这里设置为 description，也就是将元信息属性设置为页面说明，在 content 中定义具体的描述语言。

实例代码：

```
<!doctype html>
<html>
```



```
<head>
<meta charset="utf-8">
<meta name="description" content="设置页面描述">
<title>设置页面描述</title>
</head>
<body>
</body>
</html>
```



提示： 在创建描述元标记 `description` 时请注意避免以下几点误区。

- ① 把网页的所有内容都复制到描述元标记中。
- ② 使用与网页实际内容不相符的描述元标记。
- ③ 使用过于宽泛的描述，比如“这是一个网页”或“关于我们”等。
- ④ 在描述部分堆砌关键字。堆砌关键字不仅不利于排名，而且会受到惩罚。
- ⑤ 所有的网页或很多网页使用千篇一律的描述元标记。这样不利于网站优化。

2.2.4 定义编辑工具

现在有很多编辑软件都可以制作网页，在源代码的头部可以设置网页编辑工具的名称。与其他 `meta` 元素相同，这些编辑工具也只是在页面的源代码中可以看到，而不会显示在浏览器中。

基本语法：

```
<meta name="generator" content="编辑软件的名称">
```

语法说明：

在该语法中，`name` 为属性名称，设置为 `generator`，即设置编辑工具，在 `content` 中定义具体的编辑工具名称。

实例代码：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<meta name="generator" content="FrontPage">
<title>设置编辑工具</title>
</head>
<body>
</body>
</html>
```

在上述代码中，加粗部分的代码表示为网页定义编辑工具。

2.2.5 定义作者信息

在源代码中还可以设置网页制作者的姓名。

基本语法:

```
<meta name="author" content="作者的姓名">
```

语法说明:

在该语法中, `name` 为属性名称, 设置为 `author`, 即设置作者信息, 在 `content` 中定义具体的信息。

实例代码:

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<meta name="author" content="云庭">
<title>设置作者信息</title>
</head>
<body>
</body>
</html>
```

在上述代码中, 加粗部分的代码表示为网页设置作者信息。

2.2.6 定义网页文字及语言

在网页中还可以设置语言的编码方式, 这样浏览器就可以正确地选择语言, 而不需要人工选取。

基本语法:

```
<meta http-equiv="content-type" content="text/html; charset=字符集类型" />
```

语法说明:

在该语法中, `http-equiv` 用于传送 HTTP 通信协议的标头, 而在 `content` 中才是具体的属性值。 `charset` 用于设置网页的内码语系, 也就是字符集的类型。国内常用的是 GB 码, `charset` 往往设置为 `gb2312`, 即简体中文。英文是 `ISO-8859-1` 字符集, 此外还有其他字符集。

实例代码:

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Untitled Document</title>
</head>
<body>
</body>
</html>
```

在上述代码中, 加粗部分的代码表示为网页设置文字及语言, 此处设置的语言可以是日语。



2.2.7 定义网页的定时跳转

在浏览网页时经常会看到一些欢迎信息的页面，在经过一段时间后，这些页面会自动转到其他页面，这就是网页的跳转。用 `http-equiv` 属性中的 `refresh` 不仅能够完成页面自身的自动刷新，也可以实现页面之间的跳转过程。通过设置 `meta` 对象的 `http-equiv` 属性来实现跳转页面。

基本语法：

```
<meta http-equiv="refresh" content="跳转的时间;URL=跳转到的地址">
```

语法说明：

在该语法中，`refresh` 表示网页的刷新，而在 `content` 中设置刷新的时间和刷新后的链接地址，时间和链接地址之间用分号相隔。在默认情况下，跳转时间以秒为单位。

实例代码：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<meta http-equiv="refresh" content="10;url=index1.html">
<title>定义网页的定时跳转</title>
</head>
<body>
10 秒后自动跳转
</body>
</html>
```

在上述代码中，加粗部分的代码表示为网页设置定时跳转特性，这里设置为 10 秒后跳转到 `index1.html` 页面。在浏览器中预览可以看出，跳转前如图 2-12 所示，跳转后如图 2-13 所示。



图 2-12 跳转前的网页



图 2-13 跳转后的网页

2.3 综合实例——创建基本的 HTML 文件

本章主要学习了 HTML 文件整体标记的使用。下面就用所学的知识来创建最基本的 HTML 文件。

(1) 使用 Dreamweaver CC 打开网页文档，如图 2-14 所示。



图 2-14 打开原始文档

(2) 打开“拆分”视图，在代码<title> </title>之间输入标题，如图 2-15 所示。



图 2-15 设置网页的标题

(3) 打开“拆分”视图，在<body>标记中输入 `bgcolor="#FFB5B6"`，用来定义网页的背景颜色，如图 2-16 所示。



图 2-16 定义网页的背景颜色



(4) 打开“拆分”视图，在图片的上面输入文字“七夕节快乐”，在“属性”面板中将字体大小设置为 36，如图 2-17 所示。



图 2-17 输入文字

(5) 打开“拆分”视图，在<body>语句中输入“text="#F91212"”，设置整个文档的文本颜色，如图 2-18 所示。



图 2-18 设置文字的颜色

(6) 在<body>标记中输入“topmargin="0" leftmargin="0"”，用于设置网页的上边距和左边距，将上边距设置为 0，左边距设置为 0，如图 2-19 所示。

(7) 保存网页，在浏览器中预览，如图 2-20 所示。





图 2-19 设置页面的边距



图 2-20 预览效果

2.4 本章小结

一个完整的 HTML 文档必须包含 3 个部分：一是由<html>元素定义的文档版本信息；二是由<head>定义各项声明的文档头部；三是由<body>定义的文档主体部分。本章介绍了 HTML 的主体标记、头部标记。通过本章的学习，读者能够对 HTML 有个初步的了解，从而为后面的学习打下基础。



2.5 练 习 题

1. 填空题

(1) 使用<body>标记的_____属性可以为整个网页定义背景颜色。使用_____属性可以将图片设置为背景，还可以设置背景图片的平铺方式、显示方式等。

(2) 在 HTML 语言的头部元素中，一般需要包括标题、基础信息和元信息等。HTML 的头部元素是以_____为开始标记，以_____为结束标记的。

(3) meta 元素的属性有_____和_____，其中_____属性主要用于描述网页，以便于搜索引擎查找、分类。

(4) 使用_____标记可以使网页在经过一定时间后自动刷新，这可通过将_____属性值设置为 refresh 来实现。_____属性值可以设置为更新时间。

2. 操作题

创建 HTML 网页文件，在浏览器中的预览效果如图 2-21 所示。



图 2-21 要创建的 HTML 网页文件的预览效果