

21世纪高等教育计算机规划教材

COMPUTER

HTML+CSS+JavaScript 网站开发实用技术

Web Development Technique of
HTML, CSS & JavaScript

■ 刘德山 杨彬彬 主编

- 知识系统，覆盖网站前端设计核心内容
- 案例丰富，体现 HTML5+CSS3 新技术应用
- 易学易用，提供全部的代码和素材资源



 人民邮电出版社
POSTS & TELECOM PRESS

21世纪高等教育计算机规划教材

COMPUTER

HTML+CSS+JavaScript 网站开发实用技术

Web Development Technique of
HTML, CSS & JavaScript

■ 刘德山 杨彬彬 主编

福建师范大学
图书馆
藏书印



1195480



人民邮电出版社

北京

图书在版编目(CIP)数据

HTML+CSS+JavaScript网站开发实用技术 / 刘德山, 杨彬彬主编. — 北京: 人民邮电出版社, 2014.7
21世纪高等教育计算机规划教材
ISBN 978-7-115-35350-4

I. ①H… II. ①刘… ②杨… III. ①超文本标记语言—程序设计—高等学校—教材②网页制作工具—高等学校—教材③JAVA语言—程序设计—高等学校—教材 IV. ①TP312②TP393.092

中国版本图书馆CIP数据核字(2014)第106802号

内 容 提 要

本书系统地讲述了利用 HTML、CSS 和 JavaScript 进行网页设计的相关知识, 案例部分结合了 HTML5 与 CSS3 的典型应用, 可以帮助读者快速掌握网站开发技术, 实现网站开发从传统的 HTML+CSS 向 HTML5+CSS3 过渡。

本书内容主要分为 4 部分。第一部分包括第 1 章~第 3 章, 介绍了 HTML 中广泛使用的标记和属性, 指明了那些已经过时的或很少使用的标记和属性, HTML5 中有变化的标记或 HTML5 新增的常用标记以示例的形式给出说明。第二部分包括第 4 章~第 5 章, 介绍了 CSS 的概念、应用, 通过示例说明了部分 CSS3 对 CSS2 改进的内容。第三部分包括第 6 章~第 8 章, 介绍 JavaScript 中的概念、对象和事件处理。第 4 部分, 介绍了网站的发布与管理。

本书涵盖知识全面, 案例丰富, 易学易用, 将知识点融于案例之中, 并提供了全部代码和素材资源, 能够更好地帮助读者了解和使用网站的前端开发技术。

本书可作为高等院校、高职高专院校网站设计课程的教学用书, 也可作为信息技术类及相关专业的读者或从事网站前端开发人员的参考用书。

-
- ◆ 主 编 刘德山 杨彬彬
责任编辑 邹文波
责任印制 彭志环 焦志炜
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
大厂聚鑫印刷有限责任公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 20.25 2014 年 7 月第 1 版
字数: 534 千字 2014 年 7 月河北第 1 次印刷
-

定价: 43.00 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316
反盗版热线: (010)81055315

前言

Web 技术的发展总是给人耳目一新的感觉。Web1.0 时代是一个用户完全被动接受信息的时代，用户可以用 HTML 来表示信息。当前正在享受的博客、微博、维基、在线流媒体都是 Web2.0 时代的产物。到目前为止，Web3.0 的概念已经被提出，并在信息聚合与智能搜索等方面得到应用。

在 Web 技术的应用与发展过程中，追求的是用户良好的体验和丰富的交互。而实现良好体验和丰富交互的主流技术之一则是 HTML+CSS+JavaScript，也称为 Web 前端开发技术。2010 年，CSS3 标准有了大的改进；2012 年，HTML5 标准定稿，HTML5 和 CSS3 成为 Web 技术中最受关注的话题，W3C 称：“HTML5 是开放的 Web 网络平台的奠基石。”

本书以传统的 HTML 为基础，系统阐述 CSS 和 JavaScript 的内容，并提供了一些案例，结合了 HTML5 与 CSS3 的典型应用，能够帮助读者快速掌握网站开发技术，实现从传统的 HTML+CSS 向 HTML5 和 CSS3 的过渡。本书可以作为网站前端开发人员提高技能的参考用书，也适用于信息技术类及相关专业的读者。

目前，市场上有很多关于 Web 前端开发设计方面的书籍。这些书籍在介绍 HTML、CSS 和 JavaScript 方面各有特色，介绍 HTML5 和 CSS3 时也系统详尽。但 Web 技术应用的现状是，浏览器还不能完全支持 HTML5 和 CSS3，完全支持 HTML5 的编辑器还有待完善，习惯还会使一些读者继续使用 HTML4 以前版本或 CSS2。在一段时间内，HTML5 和 CSS3 将和以前的技术共存。本书的写作定位和 HTML5 的定位是一致的，是“非革命性的发展”，尝试以传统的 HTML 和 CSS 为基础，对比介绍了 HTML5 或 CSS3 的应用，以引导读者用较短时间掌握 Web 前端开发知识，并识得 HTML5 和 CSS3 全貌。

本书包含以下写作特色。

(1) 知识全面。知识讲解由浅入深，内容覆盖了 HTML+CSS+JavaScript 的主要内容，并适当新增了 HTML5 和 CSS3 中经常使用的内容。对于 HTML5 或 CSS3 不再支持的内容，书中给出了说明或解决办法，方便读者了解和使用网站的前端开发技术。

(2) 案例丰富。全书将知识点融于 200 个案例之中，并结合典型案例进行讲解和拓展，达到“知其然，用其长”的效果。其中一些案例给出了 HTML 和 CSS 不同实现方法的对比；另外一些案例给出了早期 HTML 和 HTML5 不同代码的对比；还有一些案例给出了 CSS2 和 CSS3 代码的比较。

(3) 易学易用。教材提供了案例的全部代码和素材资源，读者可以用尽可能少的时间掌握教材内容。

本书内容主要包括以下 4 部分。

第 1 部分：包括第 1 章、第 2 章、第 3 章，介绍 HTML 主要的、广泛使用的标记和属性，这是全书的基础。过时的或现在很少使用的标记和属性书中都给予了指明。HTML5 中有变化的标记或 HTML5 新增的常用标记以示例的形式给出介绍。

第2部分:包括第4章、第5章,介绍CSS的概念、应用和示例,包括基本选择器、复合选择器、用CSS设置文字与字体、背景、图像、盒模型、多种页面布局方式等内容。在示例中介绍了CSS3对CSS2的改进部分。

第3部分:包括第6章、第7章、第8章,介绍JavaScript,包括JavaScript的语法基础,JavaScript的内置对象、浏览器对象和HTML DOM对象,还介绍了事件处理的相关内容。

第4部分:包括第9章,介绍网站的发布与管理。

全书的示例都经过了作者的上机实践,结果运行无误。示例代码及各种资源文件可以到人民邮电出版社的教学服务资源网(www.ptpedu.com.cn)上下载。

本书由刘德山、杨彬彬主编并统稿,闫俊伢担任副主编。由于时间及编者水平上的限制,书中可能存在疏漏或错误,敬请读者批评指正。

编者

2014年3月

目 录

第 1 章 网站开发基础知识1

1.1 互联网的访问过程	1
1.2 互联网的工作机制	3
1.3 网站设计中的基本概念	4
1.4 制作网站的标准和常用技术	6
1.4.1 Web 标准	7
1.4.2 网站开发常用技术	8
1.5 常用开发工具	9
1.6 网站建设的流程	10
1.6.1 明确网站的定位	10
1.6.2 确定网站的主题	11
1.6.3 网站的整体规划	12
1.6.4 收集资料与素材	13
1.6.5 设计制作网页页面	14
1.6.6 测试和发布	15
1.6.7 网站的维护与推广	16
1.7 建立站点	16
1.7.1 Dreamweaver CS5 介绍	16
1.7.2 Dreamweaver 站点组成	19
1.7.3 创建站点	19
1.7.4 管理站点	20
1.8 应用案例	21
本章小结	22
思考与练习	23

第 2 章 静态网页制作—— 使用 HTML 技术24

2.1 HTML 概述	24
2.1.1 HTML 简介	24
2.1.2 用记事本创建 HTML 文件	26

2.1.3 HTML 文档结构与书写规范	26
2.1.4 建立 HTML 文件	27
2.2 HTML 页面修饰标记	30
2.2.1 传统 HTML 页面标记	30
2.2.2 HTML5 改进的页面标记	32
2.3 列表	33
2.3.1 有序列表 	33
2.3.2 无序列表 	35
2.3.3 自定义列表<dl>	36
2.4 多媒体	37
2.4.1 图像标记	37
2.4.2 背景音乐标记<bgsound>和滚动字 幕标记<marquee>*	41
2.4.3 多媒体文件标记<embed>	43
2.4.4 HTML5 新增视频标记<video>	44
2.4.5 HTML5 新增音频标记<audio>	46
2.5 超链接	47
2.5.1 超链接标记<a>	48
2.5.2 超链接类型	48
2.5.3 超链接路径	49
2.6 表单	51
2.6.1 表单定义标记<form>	52
2.6.2 输入标记<input>	52
2.6.3 HTML5 新增 input 类型	55
2.6.4 列表框标记<select>	58
2.6.5 文本域输入标记<textarea>	59
2.7 应用案例	60
2.7.1 多层嵌套列表案例	60
2.7.2 会员注册表单案例	62
2.7.3 旅游网站首页案例	64
本章小结	66
思考与练习	67

第3章 规划页面——HTML 的 页面布局设计69

3.1 表格.....69	
3.1.1 HTML 的表格标记.....69	
3.1.2 HTML 表格的属性.....71	
3.1.3 表格嵌套.....74	
3.2 框架.....75	
3.2.1 框架基本结构.....75	
3.2.2 框架分割方式.....76	
3.2.3 框架集常用属性.....77	
3.2.4 内嵌框架.....80	
3.3 使用层进行网页布局.....82	
3.4 应用案例.....83	
3.4.1 用表格实现页面布局案例.....83	
3.4.2 用框架实现布局案例.....87	
本章小结.....93	
思考与练习.....93	

第4章 美化网页—— 使用 CSS 技术.....95

4.1 CSS 概述.....95	
4.1.1 CSS 的引入.....95	
4.1.2 CSS 简介.....98	
4.2 CSS 基本选择器.....99	
4.2.1 标记选择器.....99	
4.2.2 类选择器.....100	
4.2.3 ID 选择器.....101	
4.3 在 HTML 中使用 CSS 的方法.....103	
4.3.1 行内样式.....103	
4.3.2 嵌入样式.....103	
4.3.3 链接样式.....104	
4.3.4 导入样式.....105	
4.3.5 样式的优先级.....106	
4.4 CSS 复合选择器.....109	
4.4.1 交集选择器.....109	
4.4.2 并集选择器.....110	
4.4.3 后代选择器.....111	

4.4.4 子选择器.....113	
4.4.5 相邻选择器.....113	
4.4.6 伪类选择器.....114	
4.4.7 伪对象选择器.....116	
4.5 用 CSS 设置文本样式.....117	
4.5.1 字体属性.....117	
4.5.2 文本属性.....122	
4.6 用 CSS 设置颜色与背景.....125	
4.7 用 CSS 设置图像效果.....129	
4.7.1 为图片添加边框.....129	
4.7.2 图片缩放.....131	
4.7.3 图文混排.....132	
4.8 应用案例.....135	
4.8.1 用 CSS 美化表单案例.....135	
4.8.2 用 CSS 设计网站页面案例.....137	
本章小结.....140	
思考与练习.....141	

第5章 使用 DIV+CSS 实现 精美布局143

5.1 CSS 盒模型.....143	
5.1.1 内容.....144	
5.1.2 边界.....145	
5.1.3 填充.....147	
5.1.4 边框.....148	
5.2 CSS 布局常用属性.....151	
5.2.1 定位属性 (position).....151	
5.2.2 浮动属性 (float).....156	
5.3 DIV+CSS 网页布局.....158	
5.3.1 基本布局.....158	
5.3.2 复杂布局.....162	
5.4 应用案例.....166	
5.4.1 用 DIV+CSS 实现图文混排案例.....166	
5.4.2 用 CSS 设计圆角案例.....171	
5.4.3 二级导航菜单制作案例.....173	
本章小结.....178	
思考与练习.....179	

第6章 让网页动起来—— 使用 JavaScript 技术.....180

6.1 JavaScript 简介.....180	
6.1.1 JavaScript 的概念和特点.....180	
6.1.2 JavaScript 与 Java.....181	
6.1.3 第一个 JavaScript 程序.....182	
6.2 JavaScript 语言基础.....184	
6.2.1 JavaScript 语法格式.....185	
6.2.2 JavaScript 代码书写位置.....185	
6.2.3 JavaScript 语句.....186	
6.2.4 JavaScript 注释.....187	
6.2.5 数据类型.....187	
6.2.6 变量.....189	
6.2.7 常量.....190	
6.3 表达式与运算符.....191	
6.3.1 表达式.....191	
6.3.2 运算符.....192	
6.4 JavaScript 控制结构与函数.....197	
6.4.1 JavaScript 控制结构.....197	
6.4.2 函数.....205	
6.5 应用案例.....209	
本章小结.....210	
思考与练习.....211	

第7章 使用 JavaScript 对象编程.....212

7.1 JavaScript 对象概述.....212	
7.1.1 对象.....212	
7.1.2 对象的引用.....213	
7.1.3 对象的操作.....214	
7.1.4 JavaScript 的对象类型.....216	
7.2 JavaScript 内置对象.....216	
7.2.1 String 对象.....216	
7.2.2 Array 对象.....220	
7.2.3 Date 对象.....226	
7.2.4 Math 对象.....228	
7.3 浏览器对象.....230	

7.3.1 BOM 概述.....230	
7.3.2 窗口对象 (Window).....231	
7.3.3 浏览器对象 (Navigator).....236	
7.3.4 屏幕对象 (Screen).....237	
7.3.5 地址对象 (Location).....240	
7.3.6 历史对象 (History).....240	
7.3.7 文档对象 (Document).....241	
7.4 HTML DOM 对象.....245	
7.4.1 DOM 概述.....245	
7.4.2 HTML DOM 对象.....247	
7.4.3 访问 HTML DOM 对象.....247	
7.5 自定义对象.....252	
7.5.1 定义对象.....252	
7.5.2 创建对象实例.....252	
7.5.3 自定义对象的应用.....253	
7.6 应用案例.....254	
7.6.1 表单验证案例.....254	
7.6.2 实现复选框的全选效果案例.....258	
本章小结.....260	
思考与练习.....260	

第8章 实现用户与页面的交互 ——事件处理.....262

8.1 事件的基本概念.....262	
8.1.1 事件和事件处理.....262	
8.1.2 事件绑定.....263	
8.2 JavaScript 常见事件.....264	
8.3 事件对象.....265	
8.4 鼠标和键盘事件.....267	
8.4.1 鼠标事件.....267	
8.4.2 键盘事件.....271	
8.5 页面事件.....273	
8.6 表单事件.....273	
8.6.1 提交与重置事件.....273	
8.6.2 元素内容修改事件.....277	
8.6.3 获得焦点事件与失去焦点事件.....278	
8.7 编辑事件.....279	
8.7.1 文本编辑事件.....279	

8.7.2 拖放事件·····	280	9.3 构建自己的 Web 服务器·····	298
8.8 应用案例·····	281	9.3.1 Web 服务器简介·····	299
8.8.1 商品评级网页案例·····	281	9.3.2 安装 Web 服务器·····	299
8.8.2 跟随鼠标文字特效案例·····	283	9.4 在互联网上建立 Web 站点·····	302
8.8.3 网络相册案例·····	285	9.4.1 域名注册·····	302
本章小结·····	288	9.4.2 虚拟空间申请·····	305
思考与练习·····	289	9.4.3 发布网站·····	307
第 9 章 网站的发布与管理 ·····	290	9.5 网站维护·····	308
9.1 网页中的<meta>标记·····	290	9.5.1 网站的推广·····	309
9.1.1 name 属性和 content 属性·····	291	9.5.2 网站的安全·····	310
9.1.2 http-equiv 属性和 content 属性·····	292	本章小结·····	310
9.2 测试网站·····	294	思考与练习·····	310
9.2.1 代码测试·····	294	附录 HTML 标记列表 ·····	312
9.2.2 验证 HTML·····	295	参考文献 ·····	316
9.2.3 检查链接·····	296		
9.2.4 检查屏幕分辨率和颜色·····	298		
9.2.5 可访问性检验工具·····	298		
9.2.6 检查页面在不同浏览器中的显示效果·····	298		

第 1 章

网站开发基础知识

网络是现代获取信息和发布信息的重要工具。几乎每个人都从网络上获取信息，并以电子邮件、博客、微博等不同形式来传播信息。网站是信息传播的重要载体，商业网站可以提供销售平台，个人网站用于发布个人信息。越来越多的人希望了解网站制作和维护的过程，了解网站开发和网页设计的各种技术。

在开始设计网页或网站之前，需要了解一些基础知识。这些知识包括对上网过程，也就是浏览器工作过程的理解，还包括网页的内容描述和格式设置方法，深入的问题包括网站开发标准和标准下的各种技术。上述知识内容并不复杂，但对以后的学习过程有非常重要的影响。

本章主要包括以下内容：

- 互联网的访问过程和工作机制
- 网页设计中的基本概念和常用技术
- 常用的开发工具
- 网站建设的流程
- 站点的建立过程

1.1 互联网的访问过程

网站是由网页组成的，网站设计首先从网页设计开始。在学习网页设计之前，先看一下浏览网页的过程。

打开浏览器并在地址栏中输入某个网站的地址，就会展示出相应的网页内容，如图 1-1 所示。

从图 1-1 可以看到，网页中包含了多种类型的内容，这些内容通常被称为网页元素。最基本的网页元素是文字，此外还包括静态的图形和有动态效果的动画，以及声音和视频等其他形式的媒体文件。网页的目的是向访问者显示有价值的信息或进行交互。浏览网站时会涉及一些非常基本的概念，包括浏览器与服务器、WWW 与万维网，以及 IP 地址与域名等，下面逐一介绍。

1. 浏览器与服务器

浏览网页，首先应当知道什么是浏览器和服务器。互联网是由世界各地的计算机互相连接而成的一个计算机网络。当我们查看各类网站上的内容时，实际上就是从远程计算机中读取内容，然后在本地计算机上显示出来。这和我们打开本地计算机中 D 盘或 E 盘的文件类似，不同在于，浏览网站是从远程计算机中获取内容的。

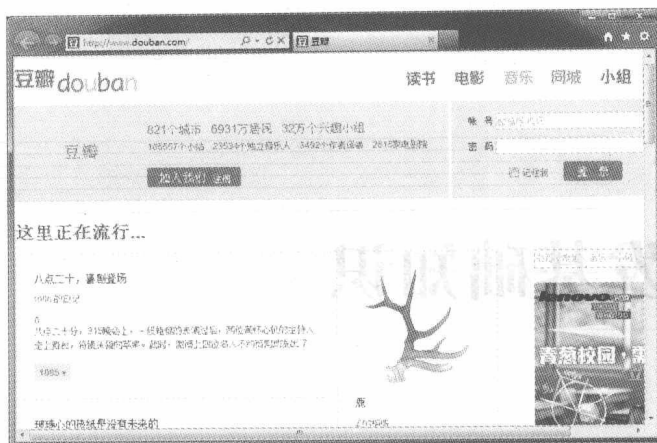


图 1-1 在浏览器中查看网页

提供内容信息的计算机就称为服务器，访问者用于浏览网页的软件称为浏览器。例如，常用的微软公司 Internet Explorer 或者奇虎公司的 360 浏览器都属于浏览器。通过浏览器可以从网络上获取服务器上的文件以及其他信息。服务器可以供许多不同的用户（浏览器）同时访问。

2. WWW 与万维网

我们浏览的网络称为互联网，也叫万维网，英文名称是“World Wide Web”，简称 WWW，也称作 Web。所以，WWW、万维网和 Web 是同义词，是一个大型的相互链接的文件所组成的集合体，范围包括了整个网络世界。

一个完整的 WWW 系统包括服务器、浏览器、HTML 文件和网络。当用户的计算机接入互联网后，通过浏览器发出访问某个站点的请求，然后这个站点的服务器就把信息传送到用户浏览器上，将文件下载到本地计算机，由浏览器显示出文件内容。这就是互联网的访问过程，也称为 WWW 服务，采用的是浏览器/服务器方式（B/S 方式），如图 1-2 所示。

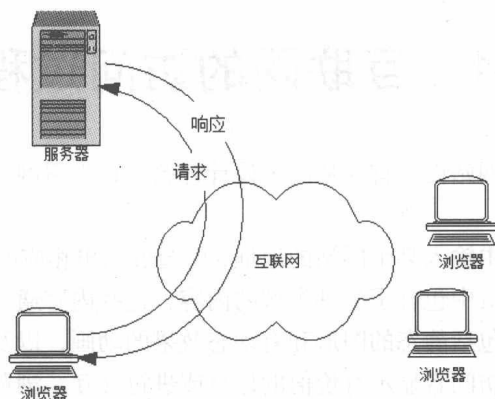


图 1-2 互联网的访问过程

实际上，WWW 服务可以认为是互联网提供的众多功能中的一个。互联网还提供了很多其他功能，例如网站制作好后，需要把网站传送到远程服务器上，这时要用到 FTP 服务，就不属于 WWW 的范畴了。

3. IP 地址和域名

要浏览服务器上的资源，必须知道服务器在网络中的地址，这是通过 IP 地址来实现的。为了

使 IP 地址容易理解和识别,又引入了域名的概念。

(1) IP 地址

IP 地址是识别互联网上计算机和网络设备的标识。网络中的每台计算机都有一个 IP 地址(可能不是固定的),目前使用的 IP 是四段地址,由小数点“.”分隔的 4 段十进制数组成,共 4 个字节,即 32 位。例如:221.11.172.173 是天涯社区(focus.tianya.cn)的 IP 地址。目前,IP 地址总数接近 43 亿个,但 IP 地址数量是有限的,也是非常宝贵的资源。考虑到 IP 地址会用尽的情况,目前互联网有关机构正在对 IP 进行版本升级,即从现在的 IPv4 升级到新的版本 IPv6。

(2) 域名

IP 地址可以用来标识网络上的计算机,但是,要让大多数人记住一个 IP 地址并不是一件容易的事。因此,人们为网络上的服务器指定了一个易于记忆的域名,来标识网络上的计算机。域名是 IP 地址的一种符号化表示。域名通过域名解析系统(Domain Name System, DNS)保证每台主机的域名与 IP 地址一一对应。在网络通信时由 DNS 进行域名与 IP 地址的转换。

域名的一般格式为:主机名.三级域名.二级域名.顶级域名。前面的 IP 地址 221.11.172.173 对应的域名为 focus.tianya.cn。

4. URL 和 HTTP

WWW 上的地址通过 URL 指明,HTTP 是用于浏览网站的基本约束或规则。

(1) URL

URL 是 Uniform Resource Location 的缩写,含义是统一资源定位器,用来指明文件在互联网中的位置。

URL 由协议名、服务器地址、文件路径及文件名组成。WWW 服务使用的是 HTTP,服务器地址可以是 IP 地址,也可以是域名。文件通常以.htm 或.html 为后缀名,这两种文件格式在显示时没有区别,但是在链接时不能互相转换。

例如,http://focus.tianya.cn/是一个 URL,其中,http 是协议,focus.tianya.cn 是服务器地址(域名),这里省略了文件的位置描述。

(2) HTTP

浏览器和服务器之间传输文件时,要遵循一定的规则,这个规则就是协议。HTTP 是 HyperText Transport Protocol 的缩写,即超文本传输协议,它制订了 HTML 文档运行的统一规则和标准,增强了文件的适应性。正是通过 HTTP,客户端的浏览器才能把服务器上的 HTML 文档提取出来,翻译成网页。

HTTP 采用的是客户端请求、服务器响应的工作模式,这个工作由 4 个步骤组成——客户端与服务器建立连接;客户端向服务器发出请求;服务器接受请求,发送响应;客户端接收响应,客户端与服务器断开连接。

1.2 互联网的工作机制

前面简单介绍了互联网的访问过程和其中涉及的一些概念。访问互联网过程中,提供浏览服务的服务器一般被称为 Web 服务器,如果涉及数据检索和查询操作,还会涉及数据库服务器。互联网的具体工作过程如图 1-3 所示。

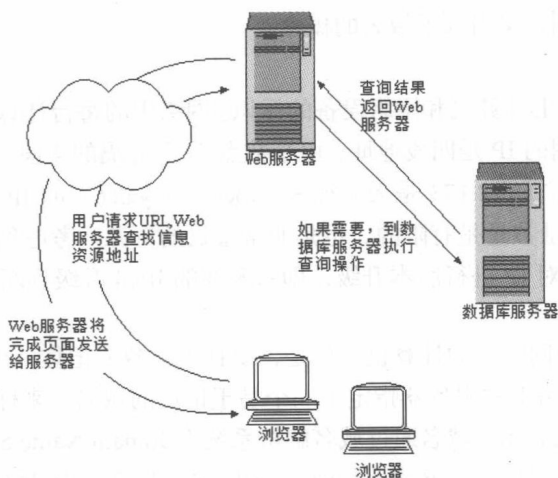


图 1-3 互联网的工作机制

① 启动客户端浏览器后, 在浏览器中输入要访问页面的 URL 地址。由 DNS 进行域名地址解析, 找到服务器 IP 地址, 向该地址所指向的 Web 服务器发出请求。

② Web 服务器根据浏览器送来的请求, 把 URL 地址转换成页面所在服务器上的文件全名, 查找相应的文件。

③ 如果 URL 指向静态 HTML 文档, Web 服务器使用 HTTP 把该文档直接送给浏览器。如果 HTML 文档中嵌入了 ASP、PHP 或 JSP 程序, 则由 Web 服务器运行这些程序, 并把结果送到浏览器。如果 Web 服务器运行的程序包含对数据库的访问, 则服务器将查询指令发送给数据库服务器, 对数据库执行查询操作。

④ 操作成功后, 数据库将查询结果返回 Web 服务器, 再由 Web 服务器将结果数据嵌入页面, 并以 HTML 格式发送给浏览器。

⑤ 浏览器解释 HTML 文档, 在客户端屏幕上展示结果。

1.3 网站设计中的基本概念

前面介绍了浏览器与服务器的概念, 还介绍了 IP 地址和域名的概念。下面进一步学习网站(页)制作过程中涉及的网站、网页、静态网页和动态网页的概念。

1. 网站

网站即 Website, 也称作站点, 是指在互联网上根据一定的规则使用 HTML 语言编写的用于展示内容的网页的集合。在本地计算机上, 网站体现为一组文件夹。网站是一种信息交流工具, 人们可以通过网站来发布信息, 或者通过浏览器来访问网站, 获取自己需要的资讯或者享受其他网络服务。

网站由域名、网站空间、网页 3 部分组成。域名就是访问网站时在浏览器地址栏中输入的网址(URL), 多个网页、网页所需资源由超链接联系起来组成网站。网站空间可以是专门的独立服务器或租用的虚拟主机, 网站需要上传到服务器的网站空间中, 从而可以被浏览者访问。

2. 网页

网站是一个整体, 网站为用户(浏览者)提供的内容是通过网页展示出来的, 用户浏览网站

其实就是浏览网页。网页实际上是用 HTML 语言编写的文本文件。在浏览网页时,浏览器将 HTML 语言翻译成用户看到的网页。

浏览网页时,如果执行浏览器的菜单命令[查看][源文件],即可在浏览器中查看该网页的 HTML 语言描述,如图 1-4 所示。



图 1-4 查看网页的 HTML 代码

不同的网页虽然内容有差别,但都是由网页基本元素组成的,一般包括图片、文字、动画、视频、音频等元素中的一种或多种。网页文件扩展名一般为.htm 或.html,但与 Word、PDF 等文件不同,一个网页实际上并不是由一个单独的文件构成的,网页显示的图片、声音以及其他多媒体文件都是单独存放的。

在 IE8 浏览器下,执行[文件][另存为]命令,并选择保存类型为“网页,全部”,会将网页下载到本地计算机,生成一个网页文件和一个资源文件夹,如图 1-5 所示。

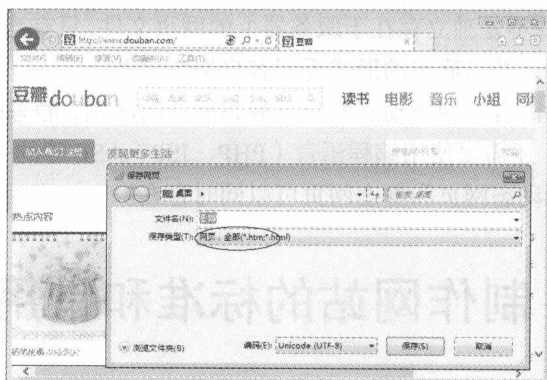


图 1-5 保存网页

网站中最重要的主页 (Home Page),它是网站的门户网页,通常命名为 index.html 或者 index.htm。主页可以是一个单独的网页,同一般网页一样,可以存放各种信息;还可以是一个特殊的网页,它是访问者浏览网站的入口。主页中一般会给出网站的概述,包括网站的主要内容、各种信息的向导。访问者在看到主页后,便会对这个网站有一个大致的了解,以确定要浏览的内容。

根据网页的功能,还可以将网页分为静态网页和动态网页。

3. 静态网页

静态网页是指在浏览器中运行、不需要到后台数据库检索数据、不含有程序的纯 HTML 格式的网页文件，其文件后缀一般为 .html、.htm、.shtml 等。静态网页并不是指网页中的所有元素都静止不动，而是指浏览器与服务器不发生交互，但是在网页中可能会包含各种动态效果，如 GIF 格式的动画、Flash 动画、JavaScript 脚本等。

静态网页的特点如下。

- 静态网页不需要数据库的支持，网站信息量很大时，查找网页内容比较困难，维护方面工作量较大。

- 静态网页的内容相对稳定，因此容易被搜索引擎检索。

- 静态网页的交互性差，在功能方面有较大的限制。

- 网页内容一经发布到网站服务器上，无论是否有用户访问，每个静态网页的内容都是保存在网站服务器上的。也就是说，静态网页是实实在在保存在服务器上的文件，每个网页都是一个独立的 HTML 文件。

4. 动态网页

动态网页是指网页文件中不仅包含 HTML 标记，还包含需要在服务器上执行的程序代码。动态网页需要后台数据库与 Web 服务器交互，利用数据库实现数据更新和查询服务。动态网页的扩展名一般是 .asp、.jsp、.php 等，在动态网页网址中通过符号“?”表明查询条件，如 <http://news.lnnu.edu.cn/showoa.php?id=22846>。

动态网页与网页上的各种动画、滚动字幕等视觉上的“动态效果”没有直接关系，无论网页最终是否具有动态效果，采用动态网站技术生成的网页都可以被称为动态网页。动态网页的特点如下。

- 动态网页以数据库技术为基础，可以大大降低网站维护的工作量。

- 采用动态网页技术的网站可以实现更多功能，如用户注册、用户登录、在线调查、用户管理、订单管理等。

- 动态网页实际上并不是独立存在于服务器上的网页文件，只有当收到用户请求时服务器才动态生成一个完整的页面，并以静态的形式返回客户端浏览器。

可以根据使用的编程语言来判断动态网页与静态网页的区别，静态网页使用 HTML 语言；动态网页使用 HTML 语言，还需要使用编程语言（PHP、JSP、ASP 等的一种）。静态网页是网站建设的基础，在同一网站，动态网页和静态网页可以同时存在。

1.4 制作网站的标准和常用技术

随着 Web 的发展，各种 Web 开发技术不断涌现。而且，各种类型和版本的浏览器越来越多，网页在不同的浏览器中的表现也有区别。因此，依据一定的标准来指导 Web 开发和应用，实现 Web 开发应用的有序、高效，实现 Web 站点的可维护和可扩展，实现网页在不同浏览器中可以获得一致的表现效果，这些内容已经成为 Web 发展过程中越来越重要的问题。

Web 开发应用遵循的标准就是 Web 标准，这个标准也是不断发展和完善的。本节介绍 Web 标准及网页开发的常用技术。

1.4.1 Web 标准

Web 标准是由 W3C 和其他标准化组织共同制定的, 该标准用来创建和解释基于 Web 的内容, Web 标准可以使得在网上发布的文档向后兼容, 使其能够被大多数人所访问。

Web 标准包括一系列标准。网页部分的标准通过 3 部分来描述: 结构 (Structure)、表现 (Presentation) 和行为 (Behavior)。对应的标准也分 3 方面: 结构标准语言主要包括 XHTML 和 XML, 表现标准语言主要包括 CSS, 行为标准主要包括对象模型 (如 W3C DOM)、ECMAScript 等。这些标准大部分由 W3C 起草和发布, 也有一些是其他标准组织制订的标准, 比如 ECMA (European Computer Manufacturers Association) 的 ECMAScript 标准。

就网站开发而言, Web 标准的结构、表现和行为对应于 3 种常用的技术, 即 (X) HTML、CSS 和 JavaScript。(X) HTML 用来决定网页的结构和内容, CSS 用来设计网页的表现形式, JavaScript 用来控制网页的行为, 这 3 部分即是本书的框架。

1. 结构标准

(1) XML

XML 是 The Extensible Markup Language (可扩展标识语言) 的简写。目前推荐遵循的标准是 W3C 于 2000 年 10 月发布的 XML1。和 HTML 一样, XML 同样来源于 SGML, 但 XML 是一种能定义其他语言的语言。XML 最初设计的目的是弥补 HTML 的不足, 以强大的扩展性满足网络信息发布的需要, 后来逐渐用于网络数据的转换和描述。

(2) XHTML

XHTML 是 Extensible HyperText Markup Language (可扩展超文本标记语言) 的缩写。XML 虽然数据转换能力强大, 完全可以替代 HTML, 但 XML 功能过于丰富, 完全替代 HTML 并不是最佳方案。因此, 在 HTML4 的基础上, 用 XML 的规则对其进行扩展, 得到了 XHTML。简单地说, 建立 XHTML 的目的就是实现 HTML 向 XML 的过渡。2010 年以后, HTML5 成为新一代的技术标准, 将逐渐取代 XHTML。

2. 表现标准

CSS 是 Cascading Style Sheets (层叠样式表) 的缩写。W3C 创建 CSS 标准的目的是以 CSS 取代 HTML 表格式布局、帧和其他表现的语言。纯 CSS 布局与结构式 XHTML 相结合能帮助网页设计者分离外观与结构, 使站点的访问及维护更加容易。现在使用的 CSS 都是在 1998 年推出的 CSS2 基础上发展而来的, 2010 年推出的 CSS3 极大简化了 CSS 的编程模型, 将逐渐占据主导地位。

3. 行为标准

(1) DOM

DOM 是 Document Object Model (文档对象模型) 的缩写。根据 W3C DOM 规范, DOM 是一种浏览器、平台、语言的接口, 使得用户可以访问页面的其他标准组件。简单理解, DOM 解决了 Netscape 的 JavaScript 和 Microsoft 的 JScript 之间的冲突, 给 Web 设计师和开发者一个标准的方法, 让他们来访问他们站点中的数据、脚本和表现层对象。

(2) ECMAScript

ECMAScript 是 ECMA (European Computer Manufacturers Association) 制定的标准脚本语言 (JavaScript)。目前推荐遵循的是 ECMAScript 262。

1.4.2 网站开发常用技术

制作静态网页和动态网页分别需要不同的技术。现在，静态网页制作技术主要有 HTML、XHTML、CSS、JavaScript 等，这些技术都是静态网页制作初学者常用的技术。制作动态网页，除了要学习静态网页制作的相关技术外，还需要动态网页的制作技术，如动态网页的制作语言 PHP、ASP、JSP 等，数据库 SQLServer、MySQL、Oracle 等方面的知识。

1. HTML

HTML 是 HyperText Markup Language 的缩写，即超文本标记语言，是一种用来制作超文本文档的简单标记语言，是网页制作的基本语言。用 HTML 编写的超文本文档称为 HTML 文档，它能独立于各种操作系统平台（如 UNIX、Windows 等）。HTML 文档是一个包含 HTML 标记、文本内容，并按照 HTML 文档结构描述的文本文件，文件的后缀名为 .html 或 .htm。浏览器读取网站上的 HTML 文档，再根据此类文档中的描述组织并显示相应的 Web 页面。

2. XHTML

XHTML 即可扩展超文本标记语言。实际上，XHTML 是一个过渡语言，结合了 XML 的强大功能及 HTML 的简单特性。现在所有的浏览器都支持 XHTML。

HTML 在使用初期，为了能被更广泛地接受，语言标准相对不够严格，例如标记可以不封闭，属性可以加引号，也可以不加引号等。这种情况导致出现了很多混乱和不规范的代码，这不符合标准化的发展趋势，影响了互联网的进一步发展。

为此，W3C 组织作为相关规范的制订者，逐步推出新的版本规范。尽管目前浏览器都兼容 HTML，但是为了使网页能够符合标准，网页设计者应该尽量使用 XHTML 规范来编写代码，部分需要注意的事项如下。

- 在 HTML 中，标记名和属性名称可以大写或者小写；但是在 XHTML 中，标记名和属性名必须小写。
- HTML 对标记的嵌套没有严格的规定，但是在 XHTML 中这是不允许的，必须严格地使标记封闭。
- 在 XHTML 中即使是空元素的标记也必须封闭。这里说的空元素的标记是指如 、
 等不成对的标记，它们也必须封闭。

例如，下面的写法在 XHTML 中是错误的。

```
水平线<hr>
图像<img src ="tul.jpg">
```

而正确的写法应该是：

```
水平线</hr>
图像<img src ="tul.jpg"/>
```

- 在 XHTML 中，属性值必须用双引号括起来；在 HTML 中，属性可以不必使用双引号。

3. CSS

CSS（Cascading Style Sheets，层叠样式表）是标准的布局语言，用来控制元素的尺寸、颜色和排版，用来定义如何显示 HTML 元素。纯 CSS 的布局与 XHTML 相结合可使内容表现与结构相分离，并使网页更容易维护，易用性更好。