



Java 经典教材译丛

# JavaScript

## 深度剖析(第2版)

THOMSON



TM

著者: [美] Don Gosselin

译者: 卢云鹏 李筱青 沈维伦

北京大学出版社

<http://cbs.pku.edu.cn>

Java 经典教材译丛

# JavaScript 深度剖析 (第 2 版)

JavaScript: Comprehensive(Second Edition)

[美] Don Gosselin 著  
卢云鹏 李筱青 沈维伦 译

北京大学出版社

· 北 京 ·

## 内 容 简 介

本书适合于大中专院校计算机相关专业作为教材,也是 JavaScript 初学者以及 JavaScript 爱好者的理想参考用书。书中详细介绍了基本的 JavaScript 程序设计原理以及实现它们的语法,内容包括 JavaScript 简介,变量、函数、对象和事件,数据类型、运算符,结构化逻辑控制结构和语句,窗口和框架,表单,动态 HTML 和动画,cookie 和安全性,服务器端 JavaScript,数据库连接,使用 Java 小程序和嵌入式数据。

本书内容全面,讲解清晰,遵循案例教学和循序渐进的学习方法,非常利于学习。

Original English language title: JavaScript: Comprehensive, 2nd Edition by Don Gosselin.

EISBN: 0-619-06334-3

Copyright © 2002 by Course Technology, a division of Thomson Learning.

Original language published by Thomson Learning (a division of Thomson Learning Asia Pte Ltd). All rights reserved. 本书原版由汤姆森学习出版集团出版。版权所有,盗印必究。

Peking University Press is authorized by Thomson Learning to publish and distribute exclusively this simplified Chinese edition. This edition is authorized for sale in the People's Republic of China only (excluding Hong Kong, Macao SARs and Taiwan). Unauthorized export of this edition is a violation of the Copyright Act. No part of this publication may be reproduced or distributed by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

本书中文简体字翻译版由汤姆森学习出版集团授权北京大学出版社独家出版发行。此版本仅限在中华人民共和国境内(不包括中国香港、澳门特别行政区及中国台湾)销售。未经授权的本书出口将被视为违反版权法的行为。未经出版者预先书面许可,不得以任何方式复制或发行本书的任何部分。

981-254-151-9

北京市版权局著作权合同登记号 图字: 01-2003-8585 号

### 图书在版编目(CIP)数据

JavaScript 深度剖析(第2版)/(美)戈瑟林(Gosselin, D.)著;卢云鹏等译. —北京:北京大学出版社, 2004.10  
(Java 经典教材译丛)

ISBN 7-301-07667-3

I. J… II. ①戈… ②卢… III. JAVA 语言—程序设计—IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 072368 号

书 名: JavaScript 深度剖析(第2版)

著作责任者: [美] Don Gosselin 著

译 者: 卢云鹏 李筱青 沈维伦

责任编辑: 胡伟晔

标准书号: ISBN 7-301-07667-3/TP·0771

出 版 者: 北京大学出版社

地 址: 北京市海淀区中关村北京大学校内 100871

网 址: <http://cbs.pku.edu.cn>

电 话: 邮购部 62752015 发行部 62750672 编辑部 62750581

电子信箱: [xxjs@pup.pku.edu.cn](mailto:xxjs@pup.pku.edu.cn)

排 版 者: 北京东方人华北大彩印中心 电话: 62754190

印 刷 者: 河北涿县鑫华书刊印刷厂

发 行 者: 北京大学出版社

经 销 者: 新华书店

787 毫米×1092 毫米 16 开本 30.25 印张 937 千字

2004 年 10 月第 1 版 2004 年 10 月第 1 次印刷

定 价: 49.00 元



# 编者序

欢迎学习 Java 程序设计语言!

自计算机发明以来,无处不在显示着它对整个人类文明和社会进步产生着如此巨大、如此深刻的影响,推动着人类社会一日千里地向前发展

历经半个多世纪,计算机软件——计算机的重要组成部分,也发生了日新月异的变化。从机器语言到汇编语言再到高级语言,从结构化语言到面向对象语言,人们不断地探索和发明新的、功能强大又简单易学的计算机语言。

用计算机语言编程,有如我们用世界上的任何一种语言写作一样,两者都是创造性的工作,而且都要用到相应的技术。如果您掌握了一种语言,熟知它的词汇和语法规则,也许您能熟练地运用这种语言进行表达,但这并不能使您成为一位作家。写作是一种创造性的工作,它需要有创造性的人来做。用计算机语言编程也是同样的道理。也许您掌握了当今最流行的程序设计语言,而且也能迅速地写出程序语句,但是,程序的内容只能靠您的创造性来产生。

## 1. 为什么要学习 Java 程序设计语言?

工欲善其事,必先利其器。没有得心应手的工具又怎么能随心所欲地发挥创意?计算机高级语言众多,而 Java 语言的得天独厚的优点,使它从众多语言中脱颖而出,越来越受到人们的青睐。Java 是一种跨平台、适合于分布式计算环境的面向对象编程语言。具体来说,它具有如下特性:简单、面向对象、分布式、易解释、可靠、安全、与平台无关、体系结构中立、可移植、高性能、多线程、动态等。

鉴于此,我们从国外遴选了 7 本流行的 Java 书,组成了《Java 经典教材译丛》,以期读者能从中受益。

## 2. 您渴望从《Java 经典教材译丛》学到什么?

博大精深是我们选此套丛书的宗旨,从 Java 基本概念和技术到其高级主题,丛书的内容涵盖了 Java 的全部知识。原书多由国外教学经验丰富的教师编写,书中提供大量的实例和案例。为了给读者提供良好的服务,我们将大部分书中的实例中的源代码放在 <http://cbs.pku.edu.cn> 中的【下载专区】上,读者可从中下载。

## 3. 什么人使用《Java 经典教材译丛》?

无论您是一位计算机语言初学者,还是一位熟知计算机语言的编程人员,《Java 经典教材译丛》中必有一本适合您。当然,本套丛书主要针对在校大专院校的学生而编写,所以大多数读者都可以轻松上手,教师更可以从中找到教学用书和参考读物,部分书中带有星号(\*)的章节则是为更高层次的读者而设计,为选学内容。

## 4. 对计算机的硬件和软件有什么需求?

- 软件 可以通过访问 Sun 的网站(<http://java.sun.com>)下载 JDK 编译程序,下载的软件大约为 20MB。
- 硬件 必须具备可以运行 Windows 操作系统的计算机或者 Windows NT 工作站,100MB 的空闲磁盘空间,最少 32MB 的内存(如果使用 Windows NT,建议准备 64MB 的内存)。

丛书编委会

2004 年 10 月

# 本书说明

本书指导初级编程者使用 JavaScript 程序设计语言开发 Web 应用程序，同时介绍了作为必备知识的 HTML 标记和语法。本书内容全面，结构合理，按照案例教学和循序渐进的方法进行讲解，这种方式不但可以增强理解，而且有利于记忆。

通过本书的学习，您将从底层开发应用程序，而不是利用事先写好的代码，这样能够更深入地理解利用 JavaScript 的 Web 程序设计原理。学完本书，您将学会创建和修改简单的 JavaScript 应用程序，并掌握开发更复杂应用程序的工具，还将学会基本的程序设计原理。这无论对进一步学习 JavaScript 语言，还是其他脚本语言或者面向对象的语言，例如 C++ 或者 Visual Basic，都是有很大好处的。

本书在下面几个方面有别于其他讲述 JavaScript 语言的书籍：

- 特别为没有程序设计经验或者 HTML 知识的读者编写。
- 举例代码简短，一个例子演示一个概念。
- 较早地涉及变量、函数、对象和事件。这就使读者能够更好地理解后文中提到的更深层次的概念和技术，并允许他们从开始就接触重要的项目。
- 文字阐述和步进式练习交叉进行。
- 全书对 HTML 标记和语法在必要的时候都会做介绍。
- JavaScript 应用程序从底层开发，读者能够获得关于复杂程序开发的清晰图像。
- 在简单易学的课程中介绍程序设计技巧。

## 本书特点

- 案例教学的方法。每章都给出了读者可能会遇到的问题，在所有的案例后面都有应用程序的示范，这些示范就能够解决各个问题。尽管读者还不会编写复杂的应用程序，但提前接触能够有的放矢。读者看到在学完该章后将能够编写的应用程序，就会更有兴趣去学习本章涉及的内容，因为他们看到将要学习的编程概念如何应用，同时也会了解这些概念的重要性。
- 步进式的学习方法。这是一种独特的教学方法。读者一直都需要编写程序代码来解决书中提出的问题，并且能够持续地接受书本的指导，清楚自己处于解决问题的哪个阶段。大量的示例将指导读者创建有用的程序。
- 提示。这些提示提供了许多补充信息，例如，执行过程的另一种方法、技术背景信息、须引起注意的常见错误、调试技巧或者读者能够获得更多信息的网址等。
- 总结。在每一章后面都有一个总结来扼要重述每个部分所涉及的概念和命令。
- 复习题。每章都以有意义的概念性的复习题结束，这些复习题可以检查读者在本章中学到的内容。
- 练习题。程序练习题为读者提供了附加的实践机会来运用该章所学的技巧和概念。这些练习逐渐增加难度，并且可使读者独立地摸索程序设计语言和程序设计环境。

## 学生文件

要完成本书练习和教学内容，需要从 <http://cbs.pku.edu.cn> 的【下载专区】中下载学生文件。将这些文件复制到自己的磁盘中，就是书中所说的数据磁盘。

## 使用个人计算机

也可以使用自己的计算机完成本书练习和教学内容。要使用自己的计算机，必须满足下列条件：

- Web 浏览器软件。Netscape 6 或者更高版本，可以到 Netscape 产品主页下载，网址是

[home.netscape.com/download/index.html](http://home.netscape.com/download/index.html)。Internet Explorer 的最新版本，可以从 Internet Explorer 主页 [www.microsoft.com/windows/ie/](http://www.microsoft.com/windows/ie/) 下载。如果要从 Internet Explorer 4 或更高版本升级到最新的 JavaScript 版本，就要到 [www.microsoft.com/msdownload/vbscript/scripting.asp](http://www.microsoft.com/msdownload/vbscript/scripting.asp) 下载最新版本的 Windows JavaScript。

- 文本编辑器或者 HTML 编辑器。要完成本书练习必须要有文本编辑器或者 HTML 编辑器。可以使用 Notepad、WordPad 或者任何能够创建简单文本文件的字处理程序。当然也可以使用 HTML 编辑器，例如 Microsoft FrontPage 或者 Adobe PageMill。这种编辑器采用绘图界面，可以创建 Web 页面并马上看到结果。

# 目 录

第1章 JavaScript 简介.....1	2.2.6 创建图像映射.....56
1.1 程序设计、HTML 和 JavaScript.....1	2.2.7 总结.....60
1.1.1 万维网.....1	2.2.8 复习题.....60
1.1.2 JavaScript 在 Web 中的作用.....2	2.2.9 练习.....63
1.1.3 超文本标记语言.....3	第3章 数据类型和操作符.....64
1.1.4 创建 HTML 文档.....6	3.1 数据类型.....64
1.1.5 JavaScript 程序设计语言.....7	3.1.1 数据类型.....64
1.1.6 逻辑与调试.....8	3.1.2 数值数据类型.....67
1.1.7 总结.....9	3.1.3 布尔值.....68
1.1.8 复习题.....9	3.1.4 字符串.....69
1.1.9 练习.....11	3.1.5 数组.....73
1.2 第一个 JavaScript 程序.....13	3.1.6 总结.....77
1.2.1 <SCRIPT>标记.....13	3.1.7 复习题.....78
1.2.2 创建 JavaScript 源代码文件.....17	3.1.8 练习.....80
1.2.3 在 JavaScript 程序中添加注释.....19	3.2 表达式和操作符.....81
1.2.4 从不兼容的浏览器中隐藏 JavaScript.....20	3.2.1 表达式.....81
1.2.5 在 HEAD 段或者 BODY 段放置 JavaScript.....22	3.2.2 算术运算符.....82
1.2.6 总结.....22	3.2.3 赋值操作符.....85
1.2.7 复习题.....23	3.2.4 比较操作符.....88
1.2.8 练习.....26	3.2.5 逻辑操作符.....90
第2章 变量、函数、对象和事件.....28	3.2.6 使用字符串.....92
2.1 变量、函数和事件的使用.....29	3.2.7 操作符优先级.....96
2.1.1 变量.....29	3.2.8 创建计算器程序.....96
2.1.2 定义自定义函数.....30	3.2.9 总结.....98
2.1.3 调用函数.....31	3.2.10 复习题.....99
2.1.4 JavaScript 内置函数.....34	3.2.11 练习.....101
2.1.5 理解 JavaScript 对象.....35	第4章 使用控制结构和语句进行 流程控制.....103
2.1.6 变量作用域.....41	4.1 条件语句.....104
2.1.7 总结.....42	4.1.1 if 语句.....104
2.1.8 复习题.....42	4.1.2 if...else 语句.....109
2.1.9 练习.....45	4.1.3 嵌套的 if 和 if...else 语句.....110
2.2 使用事件.....48	4.1.4 switch 语句.....112
2.2.1 理解事件.....48	4.1.5 总结.....115
2.2.2 HTML 标记和事件.....49	4.1.6 复习题.....115
2.2.3 事件处理器.....50	4.1.7 练习.....117
2.2.4 链接.....52	4.2 循环语句.....118
2.2.5 链接事件.....53	4.2.1 while 语句.....118

4.2.2 do...while 语句.....	122	6.2 验证表单的输入数据.....	202
4.2.3 for 语句.....	124	6.2.1 隐藏表单域.....	202
4.2.4 for...in 语句.....	128	6.2.2 Form 对象.....	204
4.2.5 with 语句.....	130	6.2.3 将表单数据提交给电子信箱地址.....	213
4.2.6 continue 语句.....	131	6.2.4 总结.....	215
4.2.7 总结.....	132	6.2.5 复习题.....	215
4.2.8 复习题.....	133	6.2.6 练习.....	217
4.2.9 练习.....	135	<b>第7章 调试 JavaScript</b> .....	219
<b>第5章 窗口和框架</b> .....	137	7.1 基本调试技术.....	219
5.1 窗口.....	138	7.1.1 理解调试.....	219
5.1.1 JavaScript 对象模型.....	138	7.1.2 错误消息.....	220
5.1.2 Window 对象.....	140	7.1.3 利用 alert()方法跟踪错误.....	222
5.1.3 打开和关闭窗口.....	141	7.1.4 用 write()和 writeln()方法跟踪错误.....	224
5.1.4 超时和间隔.....	146	7.1.5 使用注释来定位错误点.....	226
5.1.5 总结.....	147	7.1.6 其他调试技术.....	227
5.1.6 复习题.....	148	7.1.7 总结.....	229
5.1.7 练习.....	150	7.1.8 复习题.....	230
5.2 框架和其他对象.....	151	7.1.9 练习.....	232
5.2.1 创建框架.....	151	7.2 高级的调试技术和资源.....	235
5.2.2 使用 TARGET 属性.....	154	7.2.1 使用 for...in 语句来检查对象属性.....	235
5.2.3 嵌套的框架.....	157	7.2.2 Netscape 中的观察点.....	236
5.2.4 框架的格式化.....	159	7.2.3 Netscape JavaScript Debugger.....	237
5.2.5 NOFRAMES 标记.....	161	7.2.4 Microsoft Script Debugger.....	245
5.2.6 Location 对象.....	162	7.2.5 JavaScript 语言本身的 bug 和调试资源.....	249
5.2.7 History 对象.....	163	7.2.6 总结.....	250
5.2.8 Navigator 对象.....	165	7.2.7 复习题.....	251
5.2.9 引用框架和窗口.....	167	7.2.8 练习.....	252
5.2.10 总结.....	169	<b>第8章 动态 HTML 和动画</b> .....	254
5.2.11 复习题.....	170	8.1 动态 HTML.....	255
5.2.12 练习.....	172	8.1.1 简介.....	255
<b>第6章 表单</b> .....	175	8.1.2 文档对象模型.....	258
6.1 JavaScript 中的表单.....	176	8.1.3 Image 对象.....	262
6.1.1 表单概述.....	176	8.1.4 用 Image 对象创建动画.....	265
6.1.2 通用网关接口.....	177	8.1.5 图像缓存.....	270
6.1.3 <FORM>标记.....	178	8.1.6 总结.....	273
6.1.4 表单元素概述.....	182	8.1.7 复习题.....	273
6.1.5 输入域.....	183	8.1.8 练习.....	275
6.1.6 选择列表.....	194	8.2 动画和层叠样式表.....	277
6.1.7 多行文本域.....	196	8.2.1 层叠样式表.....	277
6.1.8 总结.....	197	8.2.2 在 JavaScript 代码中使用 CSS 样式.....	280
6.1.9 复习题.....	197	8.2.3 CSS 定位.....	284
6.1.10 练习.....	200	8.2.4 跨浏览器的兼容性.....	294



8.2.5 总结 .....	296	11.1.3 结构化查询语言 .....	383
8.2.6 复习题 .....	297	11.1.4 LiveWire 数据库服务 .....	385
8.2.7 练习 .....	299	11.1.5 执行数据库命令 .....	387
<b>第 9 章 cookie 和安全性</b> .....	<b>301</b>	11.1.6 LiveWire 的事务处理 .....	396
9.1 状态信息和 cookie .....	302	11.1.7 LiveWire 的错误处理 .....	397
9.1.1 状态信息 .....	302	11.1.8 总结 .....	399
9.1.2 用查询字符串保存状态信息 .....	303	11.1.9 复习题 .....	399
9.1.3 用 cookie 保存状态信息 .....	306	11.1.10 练习 .....	402
9.1.4 总结 .....	314	<b>11.2 使用 ASP 连接到数据库</b> .....	<b>404</b>
9.1.5 复习题 .....	315	11.2.1 简介 .....	404
9.1.6 练习 .....	317	11.2.2 ADO Connection 对象 .....	409
<b>9.2 安全性</b> .....	<b>319</b>	11.2.3 执行 SQL 命令 .....	411
9.2.1 JavaScript 安全性问题 .....	319	11.2.4 ADO 的事务处理 .....	420
9.2.2 同一来源策略 .....	320	11.2.5 使用 ADO Error 对象处理错误 .....	420
9.2.3 签名脚本和数字证书 .....	322	11.2.6 总结 .....	421
9.2.4 总结 .....	330	11.2.7 复习题 .....	422
9.2.5 复习题 .....	331	11.2.8 练习 .....	424
9.2.6 练习 .....	332	<b>第 12 章 使用 Java 小程序和嵌入式数据</b> .....	<b>426</b>
<b>第 10 章 服务器端 JavaScript</b> .....	<b>334</b>	12.1 Java 教程 .....	427
10.1 Netscape SSJS .....	335	12.1.1 Java 小程序和嵌入式数据 .....	427
10.1.1 客户机/服务器体系结构 .....	335	12.1.2 Java 简介 .....	428
10.1.2 服务器端 JavaScript 开发 .....	337	12.1.3 类 .....	429
10.1.3 创建 SSJS 应用程序 .....	338	12.1.4 方法 .....	430
10.1.4 SSJS 核心对象 .....	343	12.1.5 编译 Java 程序 .....	431
10.1.5 建立留言簿 .....	349	12.1.6 创建小程序 .....	432
10.1.6 总结 .....	352	12.1.7 Java 变量和数据类型 .....	433
10.1.7 复习题 .....	353	12.1.8 向 HTML 文档添加小程序 .....	435
10.1.8 练习 .....	355	12.1.9 使用 JavaScript 控制 Java 小程序 .....	437
<b>10.2 微软活动服务器页面技术</b> .....	<b>356</b>	12.1.10 总结 .....	440
10.2.1 活动服务器页面简介 .....	356	12.1.11 复习题 .....	441
10.2.2 创建 ASP 应用程序 .....	356	12.1.12 练习 .....	444
10.2.3 对象集合 .....	361	<b>12.2 LiveConnect、插件和 ActiveX</b> .....	<b>444</b>
10.2.4 ASP 核心对象 .....	362	12.2.1 概览 .....	444
10.2.5 创建留言簿 .....	368	12.2.2 Java 包和 LiveConnect .....	445
10.2.6 总结 .....	371	12.2.3 Java 和 JavaScript 之间的数据转换 .....	446
10.2.7 复习题 .....	372	12.2.4 使用 Java 控制 JavaScript .....	447
10.2.8 练习 .....	373	12.2.5 从 JavaScript 直接访问 Java 类 .....	450
<b>第 11 章 数据库连接</b> .....	<b>375</b>	12.2.6 嵌入式数据 .....	452
11.1 数据库概览、使用 SSJS 连接到数据库 .....	376	12.2.7 总结 .....	457
11.1.1 理解数据库 .....	376	12.2.8 复习题 .....	458
11.1.2 数据库管理系统 .....	379	12.2.9 练习 .....	460
		<b>附录 A JavaScript 参考</b> .....	<b>462</b>

---

A.1 注释类型 .....	462	A.7 原始数据类型 .....	464
A.2 JavaScript 保留字 .....	462	A.8 JavaScript 转义序列 .....	465
A.3 标识符 .....	462	A.9 类型转换函数和方法 .....	465
A.4 JavaScript 内置函数 .....	463	A.10 操作符 .....	465
A.5 JavaScript 内置对象 .....	463	A.11 流程控制结构和语句 .....	466
A.6 事件 .....	463	A.12 对象 .....	467

# 第 1 章 JavaScript 简介

## ★ 案例

你刚刚被 WebAdventure 公司录用,这是一个从事网站设计的公司,在 WebAdventure 公司里从事开发是一件令人兴奋的事情。像现在的许多人一样,你也习惯于花很多时间在 Internet 上冲浪,浏览着数以万计的网页。网页是由不同格式、不同类型的信息组成的,相比之下,有些网站非常具有吸引力,而且网页上提供了很多在几年前根本没有的特性,像游戏、表单、动画以及各种可视化的效果现在几乎已经是每个网页的组成部分了,而学会开发这些类型的网页将是件令人兴奋的事情。那么在进入 WebAdventure 公司后将要面临的第一个挑战,就是理解如何进行网页的制作和程序设计开发,并且了解程序开发语言 JavaScript 可以用来做什么。

## 1.1 程序设计、HTML 和 JavaScript

### 1.1.1 万维网

JavaScript 存在并工作于万维网上的网页里。要想很好地理解 JavaScript,首先应该先了解一下什么是万维网。

万维网(World Wide Web, WWW)在 1989 年由位于瑞士日内瓦的欧洲粒子物理实验室创建,最初使用万维网是为了轻松访问 Internet 上的交叉引用文档。在万维网上有一些术语是需要知道的:超文本链接(Hypertext Links)是用于定位并打开文档的,它包含着对指定文档的描述;超文本标记语言(Hypertext Markup Language, HTML)是一种用于设计网页的简单语言;而 Web 浏览器就是把 HTML 文档显示在你的显示器上的程序。目前,两种主流的浏览器是 Netscape Navigator 和 Microsoft Internet Explorer。

**提示** 在本书中 HTML 文档和网页是交替使用的。

每个网页或者 HTML 文档都有一个惟一的地址,叫做统一资源定位符(Uniform Resource Locator, URL)。URL 是统一资源标识符(Uniform Resource Identifier, URI)中的一种,URI 标识网页在万维网上的名称和地址,可以把 URL 看作网页的电话号码。每个 URL 由 4 部分组成:网络协议(通常是 HTTP)、服务、Web 服务器的域名或者网络协议地址(IP),以及 Web 文件名。超文本传输协议(Hypertext Transfer Protocol, HTTP)管理用于网页导航的超文本链接,可以认为是 HTTP 驱动 Web 的。HTTP 确保 Web 浏览器正确读取并显示包含在网页中的各种类型的信息(文本、图形、声音等)。在浏览器的地址栏中,URL 的协议后面紧跟着一个冒号、两个斜杠和服务(通常是 www,代表万维网)。域名是网络中一个惟一的地址,用来标识 Internet 上的一台计算机(通常是服务器)和一个或多个 IP 地址。域名由一个句点隔开的两部分组成:前半部分通常由容易标识某个人或者组织的文本组成,例如 DonGosselin 或者 Course;后半部分代表着机构或者组织的类型,例如,com(company)代表公司,gov(government)代表政府机构,而 edu(education)代表教育机构。course.com 是 Course Technology 的域名,而一个完整的 URL 则被表示为 http://www.DonGosselin.com 或者 http://www.course.com。

**提示** 网际协议地址或者 IP 地址是另外一种惟一标识连接在 Internet 上的计算机或者其他设备的方法。它是一串分为 4 组的数字,各组间用句点隔开。所有的 Internet 域名都对应一个惟一的 IP 地址。

URL 中的域名或者 IP 地址后面可以跟一个指定的文件名,或者是由路径与文件名组成的字符串。当

然，URL 后面也可以不指定文件名，如果不指定文件名，那么接收打开页面请求的服务器会打开根目录或者指定目录中下列名字中的某个文件：`index.html`、`index.shtml`、`index.htm`、`default.html`、`default.shtml` 或者 `default.htm`。图 1-1 指出了 URL 的各部分，这个 URL 用来打开名叫 `index-f.html` 的 HTML 文档。

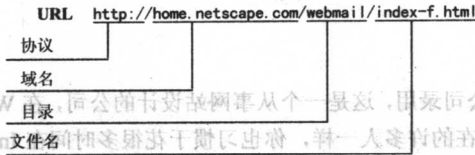


图 1-1 URL 示例

### 1.1.2 JavaScript 在 Web 中的作用

Web 最初的目的是定位和显示信息，在 Web 发展的最初阶段，基本的 HTML 创建的文档都是静态的，HTML 的主要目的只是告诉浏览器文档的显示方式，HTML 文档就相当于用字处理或者桌面印刷程序创建的文档，你只能查看或者打印它；但是当 Web 发展到超出单个学术或者科学团体的范围后，人们开始意识到这种强烈的交互性会使 Web 变得更有用处，而且随着商业应用的发展，信息的交互性和可视化的要求也在不断增加。随着 Web 技术的不断发展，为了满足更好的交互需求，Netscape 开发了在 Navigator 中使用的 JavaScript 程序设计语言。

JavaScript 给了 HTML 生命力，并且使网页由静态变成动态。与静态的 HTML 文档不一样，JavaScript 把它们变成了应用程序，例如游戏或者订单。JavaScript 能够在浏览器显示网页之后再改变页面内容，可以通过表单和控制来与用户交互、创建动画等可视化效果，甚者控制 Web 浏览器窗口本身，这些特性在 JavaScript 出现以前是不可能有的。

今天网页已经被用于各种不同的目的，例如广告或娱乐，这在很大程度上归功于 JavaScript 的出现。现在许多公司都已经有了自己的网站，为了吸引人们到自己的网站上浏览并把他们留住，这些公司的网站必须是吸引人、具有交互性而且刺激的。商业网站通常用闪烁的符号来为特价商品做广告，使用动画、交互操作、直观的导航控制，以及许多其他的特殊效果来推销商品。另外，在网站上要找寻游戏、动画或者其他的娱乐方式也是很容易的。所有的这些应用程序和特殊效果都可以通过 JavaScript 实现。

为了更好地了解 JavaScript 的用处，下面举了图 1-2、图 1-3、图 1-4 三个例子。地图是一个图像映射的例子如图 1-2 所示，当鼠标指针指在某个国家上面时，这个国家的地图就会变成高亮的，并在网页下方显示国家名。在第 2 章中将会学到如何创建图像映射。在线计算器如图 1-3 所示，这将在第 3 章中学到。最后一个程序如图 1-4 所示，是一个在线产品注册表，在第 6 章中将会创建。这些例子仅仅是在本书中要用 JavaScript 创建的程序的一部分。



图 1-2 图像映射

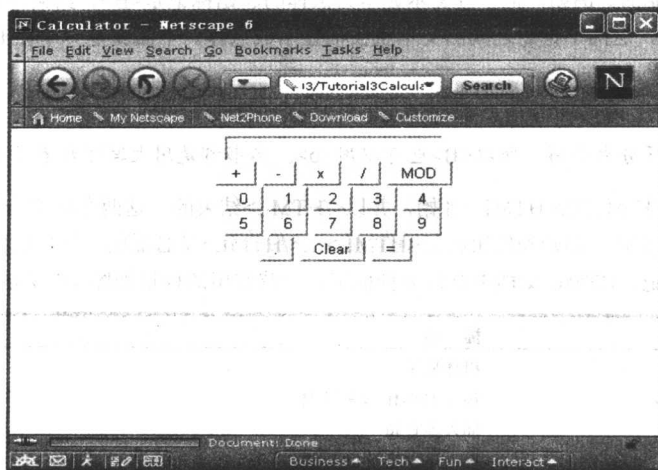


图 1-3 在线计算器

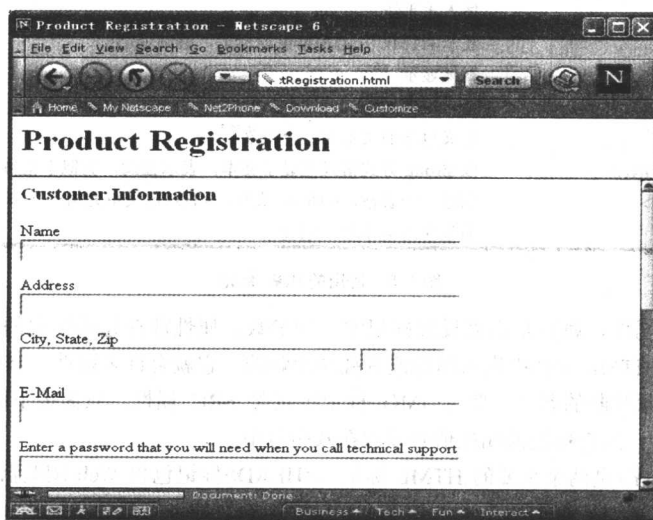


图 1-4 产品注册表

### 1.1.3 超文本标记语言

要使用 JavaScript，必须理解 HTML 是怎样构造网页的，因为 JavaScript 必须存在于网页中。本节讲解基本的 HTML 规则，这是学习 JavaScript 之前必须了解的。在本书中，还会讲述一些其他的 HTML 知识。如果熟悉 HTML，可以跳过本节。

HTML 文档必须是文本文档，里面包含有叫做标记的格式说明以及在网页上需要显示的文本。HTML 标记包括了控制黑体、斜体的格式命令，允许用户输入的控制项，例如单选按钮和复选框，以及一些用来在文档或者网页中显示图形图像及其他对象的 HTML 标记。

**提示** HTML 文档必定是以 .html 或者 .htm 为扩展名的。

当用浏览器打开一个 HTML 文档时，浏览器会按照标记里的说明，对该文档进行解释和格式化。浏览器对 HTML 文档进行解释和格式化的过程称为解析或者翻译。标记被放在尖括号中，大多数标记都有一个起始标记和一个结束标记，在一对标记之间包含着需要显示的文本或者其他项目，这些项目是由该标

记指明显示格式和控制的。例如，把一行文本显示为黑体的起始标记是<B>，结束标记是</B>。在用浏览器打开 HTML 文档时，这两个标记之间的文本就会被显示成黑体。下面就是一个在 HTML 文档中显示黑体文本的例子：

```
<B>This text will appear in boldface in a Web browser.</B>
```

**提示** HTML 不区分大小写，所以<B>也可以用<b>。本书都是用大写字母表示 HTML 标记。

所有的 HTML 文档都是以<HTML>开始，并以</HTML>结束的。这两个标记告诉浏览器标记中间的说明要翻译成 HTML 文档。起始和结束标记<HTML>...</HTML>是必需的，并且包含着组成 HTML 文档的所有文本和其他标记。HTML 文档中有许多种标记，一些常用的标记如图 1-5 所示。

HTML 标记	描述
<B></B>	粗体显示
<BODY></BODY>	包含 HTML 文档主体
 	插入换行符
<DIV></DIV>	把一个文档分成几部分
<HEAD></HEAD>	包含页面头部和整个页面信息
<Hn></Hn>	标题层次标记，n 代表 1~6 的数字
<HR>	插入水平线
<HTML></HTML>	表示 HTML 开始和结束的标记，必须使用
<I></I>	斜体显示
<IMG SRC="...">	插入图像文件
<P></P>	定义包含的文本为一个段落
<STRONG></STRONG>	以 strong 方式格式化显示文本，表示强调，类似于粗体字
<TABLE></TABLE>	创建一个表格，<TR>定义行，<TD>定义单元格
<U></U>	为包含的文本加下划线

图 1-5 常用的 HTML 标记

许多标记都具有属性，属性是用来设置标记的一些参数。属性放在起始标记的右括号前面，例如，<IMG>标记是用来在 HTML 文档中嵌入图像或者视频片断的，它就有许多属性，其中一项是 SRC 属性，用来指定图像或者视频片断的名字。要在<IMG>标记中包含 SRC 属性，只需按下面的格式输入：<IMG SRC="mygraphic.gif">，所有标记的属性值必须放在双引号中。

<HEAD>和<BODY>是两个重要的 HTML 标记。<HEAD>标记包含 Web 浏览器需要的信息，它总是处于 HTML 文档的开始处。在<HEAD>...</HEAD>这对标记之间可以放置其他的标记来管理文档内容。<TITLE>标记包含的文本会显示在浏览器的标题栏中，它是<HEAD>标记中惟一的必需元素。除了<TITLE>标记之外，包含在<HEAD>标记中的其他元素不会影响 HTML 文档的翻译。JavaScript 程序在 HTML 文档中书写时，通常出现在<HEAD>...</HEAD>标记对之间。出现在<HEAD>...</HEAD>中的 HTML 标记如图 1-6 所示。

HTML 标记	描述
<TITLE>...</TITLE>	包含页面名字，该名称显示在浏览器标题栏中
<STYLE></STYLE>	标识级联样式表(CSS)属性
<LINK>	指定一个 HTML 文档和外部资源之间的外部链接
<SCRIPT></SCRIPT>	声明嵌入脚本
<BASE>	标识文档 URL 基地址
<META>	包含文档属性

图 1-6 <HEAD>...</HEAD>中的 HTML 标记



<HEAD>...</HEAD>标记后面是<BODY>标记，它包含了 HTML 页面的主体。<BODY>标记的属性决定了 HTML 文档的外观。部分<BODY>标记的属性如图 1-7 所示。

属性	控制内容
ALINK	激活链接的颜色
BACKGROUND	背景图案
BGCOLOR	背景颜色
LINK	未访问链接的颜色
TEXT	文本颜色
VLINK	已访问链接的颜色

图 1-7 <BODY>标记属性

Web 浏览器解析或者翻译 HTML 文档时，会忽略代码中的非打印字符，例如空格、制表符和回车换行符。只有可识别的 HTML 标记和文本会在 Web 浏览器最终显示的文档中出现。在 HTML 文档主体里不能通过回车换行在网页中的段首或段尾插入空格，要实现这个目的，必须使用浏览器能够识别的<P>和<BR>标记。如果需要使用<P>标记在 HTML 文档中建立空白行，那么请注意，对于某些浏览器，如 Internet Explorer，会忽略没有包含内容的标记，也就是说，用<P>标记在 HTML 文档中创建空行，Web 浏览器可能会完全忽略它。为了防止这种情况发生，需要在<P>标记后面加上不可分空格代码——&nbsp;——并在后面加上结束标记，完整的段落标记应该是<P>&nbsp;</P>。一个 HTML 文档的例子如图 1-8 所示，而这个文档在浏览器中显示的内容则如图 1-9 所示。

```
<HTML>
<HEAD>
<TITLE>Hello World</TITLE>
</HEAD>
<BODY>
<H1>Hello World(This is the H1 tag)</H1>
<H2>This line is formatted with the H2 tag</H2>
<P>This body text line contains several character formatting tags including
<I>italics</I>,<B>bold</B>,<U>underline</U>, and <STRIKE>strickethough</STRIKE>.The
following code line creates a line break followed by a horizontal rule:<BR>
<HR>
<IMG SRC="Checkmrk.jpg">This line contains an image.
</BODY>
</HTML>
```

图 1-8 一个 HTML 文档

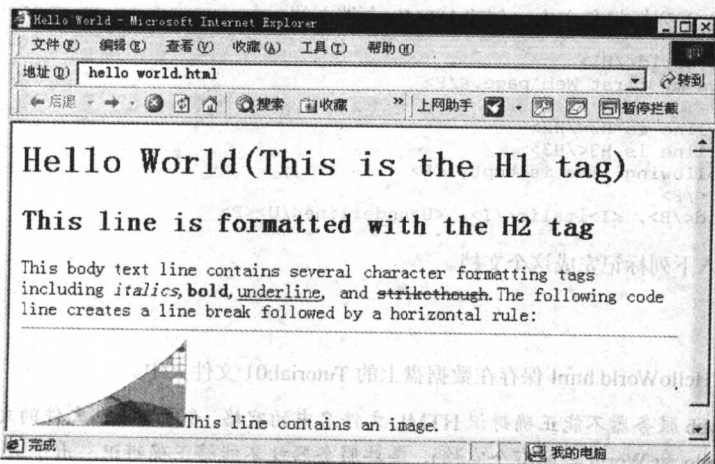


图 1-9 Web 浏览器中的 HTML 文档

**提示** 某些 HTML 标记,如段落标记<P>,并不需要结束标记。此外,不同浏览器对是否需要结束标记的要求是不同的。

### 1.1.4 创建 HTML 文档

由于 HTML 文档是文本文件,因此可以用任何文本编辑器来创建它。例如 Notepad、WordPad,或者其他字处理程序,只要能够创建简单的文本文件就行(令初次使用 HTML 的人惊讶的是,Web 浏览器居然不能创建 HTML 文档)。如果使用文本编辑器创建 HTML 文档,那么只能在用浏览器打开后才能看到最终结果。现在,已经有许多专门用来创建 HTML 文档的应用程序了(叫做 HTML 编辑器)。一些主流的 HTML 编辑器(如 Microsoft FrontPage 和 Adobe PageMill),还具有图形界面,这就能够在创建网页后立即看到网页的显示结果,也就是字处理程序中的所见即所得。另外,目前许多字处理程序(包括 Microsoft Word 和 WordPerfect),允许将文件存为 HTML 文档格式。所有这些 HTML 编辑器仍然是创建简单的文本文件,但是它们都能够自动完成标记应用的过程。例如,在 Word 里面创建了一个包含有粗体字的文档,当它以 HTML 文档格式保存的时候,<B>标记就会自动加入到创建的 HTML 文本文件中去。

**提示** 要注意不同的浏览器翻译 HTML 文档的方式是不同的。例如,在 Internet Explorer 中显示的 HTML 文档放到 Navigator 中显示就会不同。

接下来,就可以开始用本节学过的一些标记创建 HTML 文档了。可以使用任何文本编辑器,如 Notepad、WordPad 或者 HTML 编辑器。

#### 创建 HTML 文档:

- (1) 启动文本编辑器或者 HTML 编辑器新建一个文档。
- (2) 在 HTML 文档开始处输入<HTML>。记住,所有的 HTML 文档都必须以<HTML>和</HTML>标记对作为开始和结束。
- (3) 回车并把<HEAD>标记和<TITLE>标记加入到文档中。标题会出现在 Web 浏览器的标题栏中。记住,<HEAD>...</HEAD>标记对中间必须包括<TITLE>...</TITLE>标记对。<TITLE>...</TITLE>标记对不能出现在<HEAD>...</HEAD>标记对之外。

```
<HEAD>
<TITLE>Web Page Example</TITLE>
</HEAD>
```

- (4) 回车并输入<BODY>,开始 HTML 文档的主体部分。
- (5) 回车并输入下列标记和文本,创建 HTML 文档主体。

```
<H1>Hello World</H1>
<P>This is my first Web page.</P>
<HR>
<H2>This line is H2</H2>
<H3>This line is H3</H3>
<P>The following line is empty.<P>
<P>&nbsp;</P>
<P><B>bold</B>, <I>italic</I>, <U>underline</U><P>
```

- (6) 回车,输入下列标记完成这个文档。

```
</BODY>
</HTML>
```

- (7) 将文件以 HelloWorld.html 保存在数据盘上的 Tutorial.01 文件夹中。

**提示** 有的 Web 服务器不能正确辨识 HTML 文件名中的空格。例如,一个文件的文件名为 Hello World.html(Hello 和 World 之间有个空格),某些服务器就不能够正确辨识。出于这个原因,本书中的文件名都不包括空格。

- (8) 启动 Navigator、Internet Explorer 或者其他的 Web 浏览器。在浏览器里打开 HelloWorld.html 文件。在 Navigator 6 打开的 HelloWorld.html 文件如图 1-10 所示。

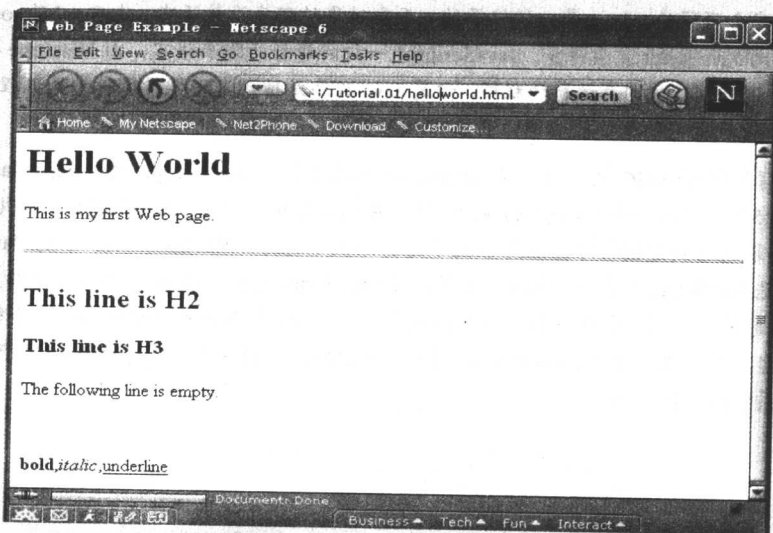


图 1-10 Navigator 6 中的 HelloWorld.html

- (9) 单击 Close 按钮结束浏览器运行。

### 1.1.5 JavaScript 程序设计语言

JavaScript 是一种脚本语言。脚本语言是指通过 Web 浏览器的解释程序解释后执行的程序设计语言。在每次运行网页的时候，解释程序都会把程序代码翻译成可执行格式——每次一行。使用脚本语言编写的程序，例如 JavaScript，都是在脚本引擎载入 HTML 页面的时候解释执行的。脚本引擎就是一种解释程序，它是浏览器的一部分。包含脚本引擎的 Web 浏览器被称作脚本宿主，Navigator 和 Internet Explorer 就是 JavaScript 程序脚本宿主。

JavaScript 首先出现在 Navigator 中，最初叫做 LiveScript。随着 Navigator 2.0 的发行，改名为 JavaScript 1.0。随后，Microsoft 发行了它自己的 JavaScript 版本用于 Explorer 4.0，称之为 JScript。目前最新版本是 Navigator 中的 JavaScript 1.5 和 Microsoft 的 JScript 5.5，后者可用于 Internet Explorer 4.0 或者更高版本。

**提示** 如果使用 Internet Explorer 4.0 或者更高版本，可以安装最新版本的 Windows Script 来升级 JavaScript 版本，下载网址为：[www.microsoft.com/msdownload/vbscript/scripting.asp](http://www.microsoft.com/msdownload/vbscript/scripting.asp)。

Microsoft 发行 JScript 以后，一系列问题也随之而来。例如，Netscape 和 Microsoft 的 JavaScript 程序设计语言差异是如此之大，以至于程序员需要为 Navigator 和 Internet Explorer 编写几乎完全不同的 JavaScript 程序，这个分歧导致了一种国际标准化 JavaScript 版本的诞生，这就是 ECMAScript。ECMAScript 已经发展到第三版，JavaScript 1.5 和 JScript 5.5 都符合 ECMAScript 第三版。尽管如此，这两者仍然有自己独特的程序设计特性，其他语言不支持这些特性。

虽然 JavaScript 和 JScript 之间存在着很多不同，但是本书的目标还是建立在 Internet Explorer 和 Navigator 上都能够运行的 JavaScript 程序。为了达到这个目的，本书主要讲述 ECMAScript 第三版，这是 Netscape 6 及以上版本和 Internet Explorer 4.0 及以上版本都支持的。在写本书的时候，还有很多不完全与 ECMAScript 第三版兼容的旧版本的 Navigator 或者 Internet Explorer 被广泛使用，尤其是 Navigator 浏览器。对于 Internet Explorer 4.0 或者更高版本，可以升级浏览器使用的 JScript 版本。然而，对于 Navigator，JavaScript 版本是直接建立在浏览器里的。为了升级 Navigator 里的 JavaScript 版本，必须升级浏览器本身。