

21世纪高等职业教育信息技术类规划教材

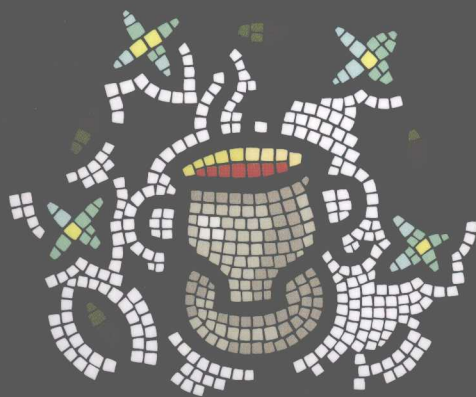
21 Shiji Gaodeng Zhiye Jiaoyu Xinxi Jishulei Guihua Jiaocai

JavaScript 程序设计基础教程

JavaScript CHENGXU SHEJI JICHU JIAOCHENG

曾海 主编 高春艳 于一 张春 副主编

- 内容全面，由浅入深
- 结合案例，贴近实践
- 综合实训提高实践能力



 人民邮电出版社
POSTS & TELECOM PRESS

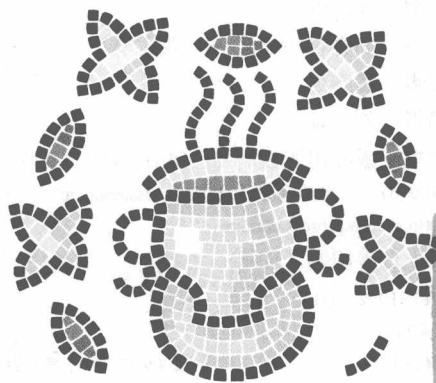
21世纪高等职业教育信息技术类规划教材

21 Shiji Gaodeng Zhiye Jiaoyu Xinxu Jishulei Guihua Jiaocai

JavaScript 程序设计基础教程

JavaScript CHENGXU SHEJI JICHU JIAOCHENG

曾海 主编 高春艳 于一 张春 副主编



人民邮电出版社

样书

专用章

人民邮电出版社

北京

图书在版编目 (CIP) 数据

JavaScript程序设计基础教程 / 曾海主编. —北京: 人民邮电出版社, 2009.10
21世纪高等职业教育信息技术类规划教材
ISBN 978-7-115-20739-5

I. J… II. 曾… III. JAVA语言—程序设计—高等学校: 技术学校—教材 IV. TP312

中国版本图书馆CIP数据核字 (2009) 第058631号

内 容 提 要

本书由浅入深地介绍了 JavaScript 开发所涉及各类知识。全书共分 11 章, 内容包括 JavaScript 概述、HTML 文档、JavaScript 语言基础、JavaScript 基本语句、函数、对象编程、事件处理、浏览器对象、CSS+DIV 技术、Ajax 技术和综合实训。本书配有大量的实例, 介绍详略得当, 有益于读者快速掌握 JavaScript 编程技术。

本书可作为计算机相关专业的教材, 也适合 JavaScript 编程初学者及程序开发人员学习参考。

21 世纪高等职业教育信息技术类规划教材

JavaScript 程序设计基础教程

-
- ◆ 主 编 曾 海
 - 副 主 编 高春艳 于 一 张 春
 - 责任编辑 潘春燕
 - 执行编辑 刘 琦

 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京昌平百善印刷厂印刷

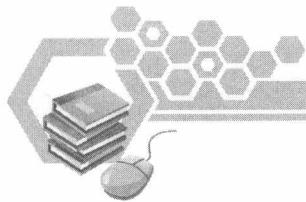
 - ◆ 开本: 787×1092 1/16
印张: 14.25
字数: 362 千字
印数: 1—3 000 册
- 2009 年 10 月第 1 版
2009 年 10 月北京第 1 次印刷

ISBN 978-7-115-20739-5/TP

定价: 25.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223
反盗版热线: (010)67171154

前 言



JavaScript 是 Web 页面中的一种脚本编程语言，也是一种通用的、跨平台的、基于对象和事件驱动并具有安全性的脚本语言。它不需要进行编译，而是直接嵌入在 HTML 页面中，把静态页面转变成支持用户交互并响应相应事件的动态页面。

高职高专院校已将 JavaScript 作为一门重要的程序设计课程。本书在编写过程中力求突出高职教育特点，坚持基础知识管用、理论知识够用、专业知识实用、专业技能会用的编写原则，以培养专业技术应用能力和职业操作技能为教材编写重点，将教材中的教学内容与职业培养目标相结合。

本书内容分为 JavaScript 脚本编程语言讲解与综合实训两部分，内容讲解由浅入深、循序渐进，将理论知识和实际操作相结合，着眼于让读者快速掌握 JavaScript 编程技术。在内容编排方面，注意难点分散、循序渐进；在文字叙述方面，注意言简意赅、重点突出；在实例选取方面，注意实用性强、针对性强。

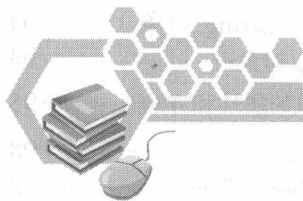
本书配有大量实例并在每一章的后面还提供了习题，方便读者及时验证自己的学习效果。同时，本书还制作了多媒体课件，供教师教学参考使用。

本书由曾海任主编，高春艳、于一、张春任副主编，参加编写工作的还有张乐仲、邹天思、梁冰、房大伟、安剑、孙秀梅、陈丹丹等。

虽然在编写本书的过程中，我们本着科学、严谨的态度，力求精益求精，但错误、疏漏之处在所难免，敬请广大读者批评指正。

编 者

2009 年 2 月



第 1 章 JavaScript 概述 1	第 3 章 JavaScript 语言基础 37
1.1 JavaScript 简述..... 1	3.1 关键字和标识符..... 38
1.1.1 什么是 JavaScript..... 1	3.1.1 关键字..... 38
1.1.2 JavaScript 的主要特点..... 2	3.1.2 标识符..... 38
1.1.3 JavaScript 的应用..... 3	3.2 数据类型..... 39
1.2 JavaScript 的环境要求..... 4	3.2.1 数值型..... 39
1.2.1 硬件要求..... 4	3.2.2 字符串型..... 40
1.2.2 软件要求..... 4	3.2.3 布尔型..... 41
1.3 编写 JavaScript 的工具..... 5	3.2.4 特殊数据类型..... 41
1.3.1 使用记事本..... 6	3.3 常量..... 42
1.3.2 使用 FrontPage..... 6	3.4 变量..... 43
1.3.3 使用 Dreamweaver..... 8	3.4.1 变量的命名..... 43
1.4 编写第一个 JavaScript 程序..... 10	3.4.2 变量的声明与赋值..... 43
1.4.1 编写 JavaScript..... 10	3.4.3 变量的作用域..... 44
1.4.2 运行 JavaScript 程序..... 12	3.4.4 变量的生存期..... 44
1.4.3 调试 JavaScript 程序..... 13	3.5 运算符..... 45
习题..... 14	3.5.1 算术运算符..... 45
第 2 章 HTML 文档 15	3.5.2 比较运算符..... 46
2.1 HTML 文档基础..... 15	3.5.3 逻辑运算符..... 47
2.1.1 HTML 标记..... 16	3.5.4 字符串运算符..... 47
2.1.2 HTML 文档的基本结构..... 18	3.5.5 赋值运算符..... 48
2.2 HTML 文档的常用标记..... 19	3.5.6 条件运算符..... 49
2.2.1 文字标记..... 20	3.5.7 其他运算符..... 50
2.2.2 段落标记..... 23	3.5.8 运算符的优先级..... 51
2.2.3 列表标记..... 24	3.6 表达式..... 52
2.2.4 超链接标记..... 26	习题..... 52
2.2.5 图像标记..... 28	第 4 章 JavaScript 基本语句 54
2.2.6 表格标记..... 29	4.1 条件语句..... 55
2.2.7 框架标记..... 31	4.1.1 if 语句..... 55
2.2.8 多媒体标记..... 34	4.1.2 if...else 语句..... 58
习题..... 35	4.1.3 if...else if...else if 语句..... 58



4.1.4 if 语句的嵌套	59	6.5 字符串对象 (String 对象)	94
4.1.5 switch 语句	61	6.5.1 创建 String 对象	94
4.2 循环语句	64	6.5.2 String 对象的属性和方法	95
4.2.1 while 语句	65	6.6 Number 对象	98
4.2.2 do...while 语句	66	6.6.1 创建 Number 对象	98
4.2.3 for 语句	67	6.6.2 Number 对象的属性	98
4.2.4 for...in 语句	68	6.6.3 Number 对象的方法	99
4.3 跳转语句	69	6.6.4 Number 对象的最大值或最小可能值	99
4.3.1 break 语句	69	6.6.5 Number 对象的正负无穷大的值	99
4.3.2 continue 语句	69	6.6.6 将数字以不同的格式转换成字符串	100
4.4 异常处理语句	71	6.7 Array 对象	102
4.4.1 throw 语句	71	6.7.1 什么是数组	102
4.4.2 try...catch...finally 语句	72	6.7.2 创建和访问数组	102
习题	73	6.7.3 Array 对象的输入输出	103
第 5 章 函数	74	6.7.4 Array 对象的常用属性和方法	104
5.1 什么是函数	74	6.7.5 二维数组	104
5.2 函数的定义	75	习题	105
5.3 函数的调用	76	第 7 章 事件处理	107
5.3.1 函数的简单调用	76	7.1 事件的基本概念	107
5.3.2 在事件响应中调用函数	77	7.1.1 什么是事件	107
5.3.3 通过链接调用函数	77	7.1.2 事件处理程序的调用	108
5.4 函数参数的使用	78	7.1.3 JavaScript 的常用事件	111
5.5 使用函数返回值	79	7.2 鼠标键盘事件	113
5.6 函数的嵌套	80	7.2.1 鼠标的单击事件	113
5.7 递归函数	81	7.2.2 鼠标的按下和松开事件	114
5.8 JavaScript 中的内置函数	82	7.2.3 鼠标的移入移出事件	115
习题	84	7.2.4 鼠标移动事件	116
第 6 章 对象编程	86	7.2.5 键盘事件	117
6.1 对象概述	86	7.3 页面事件	118
6.1.1 什么是对象	86	7.3.1 加载与卸载事件	119
6.1.2 创建和删除对象	88	7.3.2 页面大小事件	120
6.1.3 对象的属性和方法	88	7.4 表单事件	121
6.2 在 JavaScript 中使用对象	89	7.4.1 获得焦点与失去焦点事件	121
6.3 日期对象 (Date 对象)	89	7.4.2 失去焦点修改事件	123
6.3.1 创建 Date 对象	89	7.4.3 表单提交与重置事件	124
6.3.2 Date 对象的主要属性和方法	90	7.5 滚动字幕事件	126
6.4 数学对象 (Math 对象)	93		



7.5.1 onbounce 事件	126	9.5.4 容器属性	164
7.5.2 onstart 事件	127	9.5.5 列表属性	165
7.6 编辑事件	128	9.5.6 鼠标属性	166
7.6.1 文本编辑事件	129	9.5.7 定位和显示	167
7.6.2 对象拖动事件	132	9.5.8 CSS 滤镜	169
习题	134	9.6 CSS+DIV 美化与布局页面	171
第 8 章 浏览器对象	135	9.6.1 <div>标记与标记	171
8.1 窗口对象	136	9.6.2 <div>与的区别	173
8.1.1 Window 对象概述	136	9.6.3 用 CSS+DIV 实现指定尺寸 无边框、无滚动条窗口	175
8.1.2 Window 对象的属性和方法	136	习题	177
8.1.3 打开和关闭窗口	137	第 10 章 Ajax 技术	178
8.1.4 定时打开窗口	139	10.1 Ajax 介绍	179
8.2 文档对象 (document)	139	10.2 Ajax 技术的组成	180
8.2.1 document 对象概述	139	10.2.1 JavaScript	180
8.2.2 document 对象的常用属性、 方法和事件	140	10.2.2 XMLHttpRequest	180
8.2.3 文档对象的应用	141	10.2.3 CSS	182
8.3 表单对象 (form)	142	10.2.4 DOM	183
8.3.1 认识表单对象	142	10.2.5 XML	183
8.3.2 form 对象的属性、方法和事件	142	10.3 实现 Ajax 的基本步骤	183
习题	150	10.3.1 创建 XMLHttpRequest 对象	183
第 9 章 CSS+DIV 技术	151	10.3.2 创建 HTTP 请求	184
9.1 CSS 简介	151	10.3.3 设置响应 HTTP 请求状态 变化的函数	185
9.2 样式定义	153	10.3.4 设置获取服务器返回数据的 语句	185
9.2.1 样式定义的格式	153	10.3.5 发送 HTTP 请求	186
9.2.2 选择符的分类	153	10.3.6 局部更新	187
9.3 使用样式	154	10.4 用 Ajax 实现无刷新广告轮显	189
9.3.1 嵌入样式表	154	习题	191
9.3.2 链接外部样式表	154	第 11 章 综合实训	192
9.3.3 引入外部的样式表	154	11.1 实训 1 JavaScript 基本操作	192
9.3.4 内嵌样式	155	11.1.1 实训目的	192
9.3.5 CSS 样式的优先级	155	11.1.2 实训内容	192
9.4 Style 对象	156	11.1.3 实训步骤	193
9.5 CSS 属性	157	11.2 实训 2 HTML 文档基本操作	193
9.5.1 字体属性	157	11.2.1 实训目的	193
9.5.2 文本属性	159		
9.5.3 颜色和背景属性	161		



11.2.2	实训内容	193	11.6.3	实训步骤	202
11.2.3	实训步骤	194	11.7	实训 7 事件处理的操作	209
11.3	实训 3 JavaScript 语言基础 操作	196	11.7.1	实训目的	209
11.3.1	实训目的	196	11.7.2	实训内容	209
11.3.2	实训内容	196	11.7.3	实训步骤	210
11.3.3	实训步骤	197	11.8	实训 8 浏览器对象的应用	212
11.4	实训 4 JavaScript 基本语句的 应用	198	11.8.1	实训目的	212
11.4.1	实训目的	198	11.8.2	实训内容	212
11.4.2	实训内容	198	11.8.3	实训步骤	213
11.4.3	实训步骤	198	11.9	实训 9 CSS+DIV 技术的 应用	214
11.5	实训 5 函数的应用	200	11.9.1	实训目的	214
11.5.1	实训目的	200	11.9.2	实训内容	214
11.5.2	实训内容	200	11.9.3	实训步骤	214
11.5.3	实训步骤	200	11.10	实训 10 Ajax 技术的应用	217
11.6	实训 6 对象编程的操作	201	11.10.1	实训目的	217
11.6.1	实训目的	201	11.10.2	实训内容	217
11.6.2	实训内容	201	11.10.3	实训步骤	217

第 1 章

JavaScript 概述

学习目标:

JavaScript 是 Web 页面中的一种脚本编程语言,也是一种通用的、跨平台的、基于对象和事件驱动并具有安全性的脚本语言。通过本章的学习,读者可以达到以下学习目标。

- 了解什么是 JavaScript 和 JavaScript 的作用、基本特点。
- 了解 JavaScript 的软硬件环境要求。
- 熟悉编写 JavaScript 所需的工具。
- 独立编写一个 JavaScript 程序。

学习导航:

本章首先介绍什么是 JavaScript, JavaScript 的作用、基本特点,然后介绍 JavaScript 的软硬件环境要求以及编写 JavaScript 所需的工具,最后编写一个 JavaScript 程序。本章内容在书中的位置如图 1.1 所示。

知识框架:

本章内容知识框架如图 1.2 所示。

1.1 JavaScript 简述

1.1.1 什么是 JavaScript

JavaScript 是 Web 页面中的一种脚本编程语言,也是一种通用的、跨平台的、基于对象和事件驱动并具有安全性的脚本语言。它不需要进行编译,而是直接嵌入在 HTML 页面中,把静态页面转变成支持用户交互并响应相应事件的动态页面。

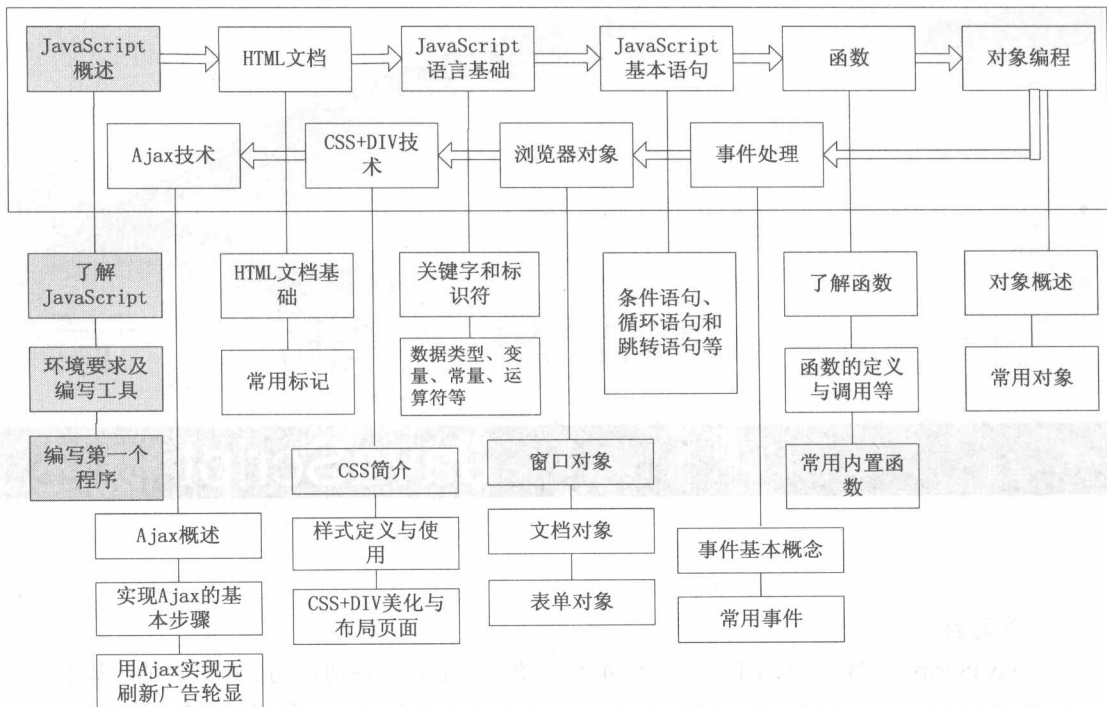


图 1.1 本章内容在书中的位置

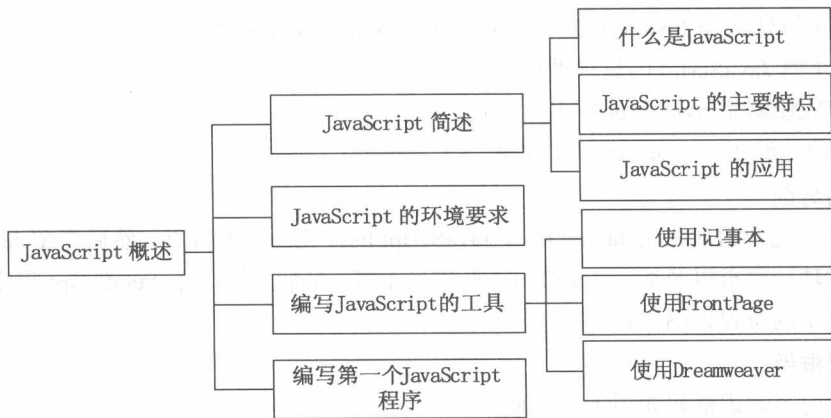


图 1.2 本章内容知识框架

1.1.2 JavaScript 的主要特点

JavaScript 脚本语言的基本特点如下。

(1) 解释性。JavaScript 不同于一些编译性的程序语言，如 C、C++ 等，它是一种解释性的程序语言，它的源代码不需要经过编译，可以直接在浏览器中运行时被解释。

(2) 基于对象。JavaScript 是一种基于对象的语言，这意味着它能运用自己已经创建的对象。因此，许多功能可以来自于脚本环境中对象的方法与脚本的相互作用。

(3) 事件驱动。JavaScript 可以直接对用户或客户输入做出响应，无需经过 Web 服务程序。



它对用户的响应是以事件驱动的方式进行的。所谓事件驱动，就是指在主页中执行了某种操作所产生的动作，此动作被称为“事件”。比如，按下鼠标、移动窗口、选择菜单等都可以被视为事件。当事件发生后，可能会引起相应的事件响应。

(4) 跨平台。JavaScript 依赖于浏览器本身，与操作环境无关，只要计算机能运行浏览器并支持 JavaScript 的浏览器，就可正确执行。

(5) 安全性。JavaScript 是一种安全性语言，它不允许访问本地的硬盘，并不能将数据存入到服务器上，不允许对网络文档进行修改和删除，只能通过浏览器实现信息浏览或动态交互，但这样可有效地防止数据的丢失。

1.1.3 JavaScript 的应用

使用 JavaScript 脚本实现的动态页面在 Web 上随处可见，下面将介绍几种常见的 JavaScript 应用。

(1) 验证用户输入的内容。使用 JavaScript 脚本语言可以在客户端对用户输入的数据进行验证。例如，在制作用户注册信息页面时，要求用户输入确认密码，以确定用户输入的密码是否正确。如果用户在“确认密码”域输入的信息与“密码”域输入的信息不同，将弹出相应的提示信息，如图 1.3 所示。

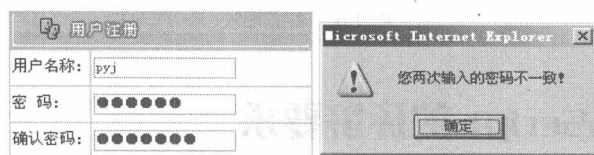


图 1.3 验证 2 次输入的密码是否一致

(2) 动画效果。在浏览网页时，经常会看到一些动画效果，这些动画效果使页面显得更加生动。使用 JavaScript 脚本语言也可以实现动画效果，例如，在页面中实现下雪的效果，如图 1.4 所示。

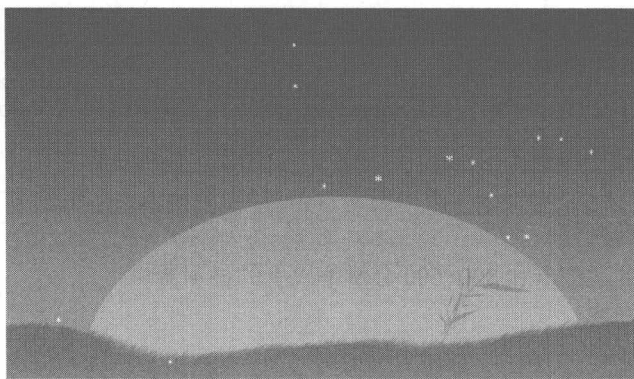


图 1.4 动画效果

(3) 窗口的应用。在打开网页时经常会看到一些浮动的广告窗口。我们也可以通过 JavaScript 脚本语言来实现，如图 1.5 所示的广告窗口。

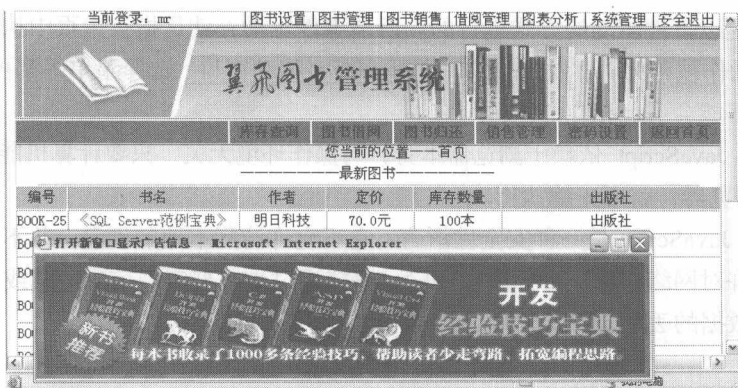


图 1.5 窗口的应用

(4) 文字特效。使用 JavaScript 脚本语言可以使文字实现多种特效，如使文字旋转，如图 1.6 所示。

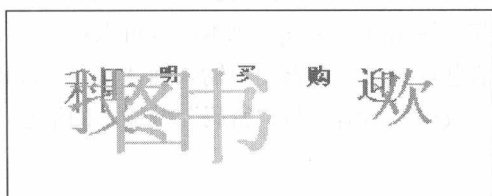


图 1.6 文字特效

1.2 JavaScript 的环境要求

1.2.1 硬件要求

JavaScript 适用于各种支持 Internet 浏览器的平台，包括 Windows、Macintosh、Linux 或 UNIX 变形系统等。

下面介绍在 Windows 系统中安装 Netscape 的 Navigator 7.1 或 Microsoft 的 Internet Explorer 6.0 所需要的最低硬件要求。

- (1) Inter Pentium III 或以上机型。
- (2) Windows 98、Windows NT 及其 Service Pack 6a 或更高版本，Windows 2000 及其 Service Pack 2 或更高版本，Windows XP。
- (3) 64MB 内存。
- (4) 20MB 可用硬盘空间。
- (5) 16 位或以上的适配卡和 800×600 分辨率以上的显示器。
- (6) CD-ROM 驱动器。

1.2.2 软件要求

- Windows 95/98 或 Windows NT 及以上版本。
- Netscape Navigator 3.0 或 Internet Explorer 3.0 以上版本。



● 编辑 JavaScript 程序可以使用任何一种文本编辑器,如 Windows 中的记事本、写字板等应用软件。由于 JavaScript 程序可以嵌入到 HTML 文件中,因此,读者可以使用任何一种编辑 HTML 文件的工具软件,如记事本、FrontPage 和 Dreamweaver 等,具体介绍参见 1.3 节。

下面重点介绍 JavaScript 对浏览器的支持。

(1) Netscape Navigator (网景浏览器)。Netscape Navigator (网景浏览器)是最早也是最具影响力的网页浏览器之一,Netscape Navigator 浏览器 1.0 版发布于 1994 年 12 月,比 Microsoft IE 1.0 浏览器发布时间还早一个多月,但由于 IE 浏览器和 Microsoft 的 Windows 操作系统捆绑在一起,因此对 Netscape 网景浏览器的市场发展造成了巨大影响,最终使得 Netscape 网景浏览器逐渐淡出主流浏览器行列。下面介绍 Netscape Navigator 浏览器版本的变化及其支持的 JavaScript 版本,如表 1.1 所示。

表 1.1 Netscape Navigator 浏览器版本及所支持的 JavaScript 版本

浏览器版本	JavaScript 版本
Navigator 2.0	JavaScript 1.0
Navigator 3.0	JavaScript 1.1
Navigator 4.0	JavaScript 1.2
Navigator 4.5	JavaScript 1.3
Navigator 6.0	JavaScript 1.5
Navigator 7.0	JavaScript 1.5

(2) Microsoft Internet Explorer (Microsoft 浏览器)。Internet Explorer, 全称 Microsoft Internet Explorer, 简称 MSIE (一般称为 Internet Explorer, 简称 IE), 是 Microsoft 公司推出的一款网页浏览器。IE 浏览器不是最早的浏览器,但由于 IE 浏览器自推出之日起就是免费的,因此几乎将其他收费浏览器置于死地。从一定程度上说,是 Microsoft 提供免费的 IE 浏览器后带动了整个互联网的发展。下面介绍 Internet Explorer 浏览器版本的变化及其所支持的 JavaScript 的版本,如表 1.2 所示。

表 1.2 IE 浏览器版本及所支持的 JavaScript 版本

浏览器版本	JavaScript 版本
Internet Explorer 3.0	JavaScript 1.0、2.0
Internet Explorer 4.0	JavaScript 3.0、4.0
Internet Explorer 5.0	JavaScript 5.0
Internet Explorer 5.5	JavaScript 5.5
Internet Explorer 6.0	JavaScript 5.6

1.3 编写 JavaScript 的工具

编写 JavaScript 脚本程序的工具具有多种,主要包括记事本、FrontPage、Dreamweaver、1st JavaScript Editor 和 Ultra Edit 32 等,下面主要介绍前 3 种。



1.3.1 使用记事本

记事本是编写 JavaScript 代码的最简单的工具，它可以进行一些简单的文字处理和 JavaScript 代码的局部修改。虽然记事本使用简单，但如果要使用它编写一些复杂的 JavaScript 代码，则需要熟练掌握 JavaScript 的语法、对象等。

例 1.1 下面介绍如何使用记事本编写 JavaScript 程序，具体步骤如下。

- (1) 单击“开始”菜单，选择“程序”/“附件”/“记事本”选项，打开记事本。
- (2) 在记事本的工作区域输入 HTML 标识符和 JavaScript 代码，具体代码如下。

```
<html>
<head>
<title>用记事本编写 JavaScript 程序</title>
<script language="javascript">
window.alert("Hello Javascript");
</script>
</head>
<body>
<h3>Hello Javascript</h3>
</body>
</html>
```


(3) 编辑完毕后，选择“文件”/“保存”选项，在打开的“另存为”对话框中，输入文件名，将其保存为.html 格式或.htm 格式，之后该文件将会变成一个 IE 浏览器的图标，双击该图标，即可显示结果，如图1.7所示。



图 1.7 用记事本编写的 JavaScript 程序



说明

利用记事本开发 JavaScript 程序也存在着缺点，就是整个编程过程要求开发者完全手工输入程序代码，这就影响了程序的开发速度。所以，在条件允许的情况下，最好不要只选择记事本一种工具来开发 JavaScript 程序。

1.3.2 使用 FrontPage

FrontPage 是 Microsoft Office 办公软件套装中的一款脚本编辑器，它是一种专业化的 Web 编程工具，用于在 HTML 文档和 ASP 文件内查看和编辑 HTML 标签，还可以在文件中添加诸如 JavaScript 和 VBScript 之类的脚本，并调试该脚本。



1. 启动 Microsoft 脚本编辑器

例 1.2 下面介绍如何使用 FrontPage 编写 JavaScript 程序。具体步骤如下。

- (1) 启动 FrontPage，并打开或新建一个 HTML 文件。
- (2) 将工作视图切换到“网页”视图，并选择“HTML”选项卡。此时，在工作区看到的是当前所编辑页面的 HTML 代码。
- (3) 选择“工具”/“宏”命令，然后在弹出的菜单中选择“Microsoft 脚本编辑器”命令，此时将打开 Microsoft 脚本编辑器，如图 1.8 所示。

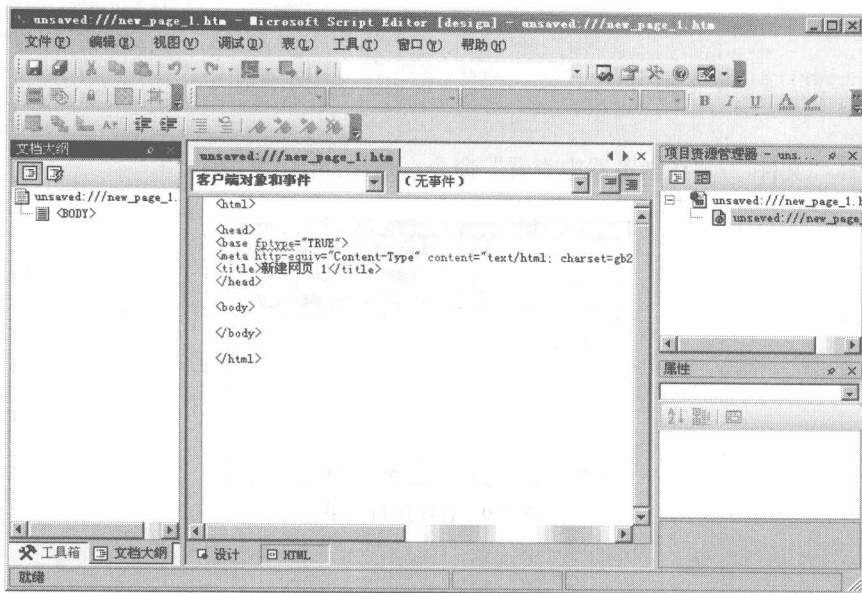


图 1.8 Microsoft 脚本编辑器



提示 如果安装 Office 时，未安装 Microsoft 脚本编辑器，则 FrontPage 会提示安装 Microsoft 脚本编辑器。

在“HTML”视图中，脚本编辑器基于文件中文本的功能对文本进行着色。例如，用紫色表示 HTML 标记，用红色表示属性名称，用蓝色表示属性值。

2. 在 HTML 文档中编写 JavaScript 程序

例 1.3 使用 Microsoft 脚本编辑器可以很容易地加入<script>标记，来编写 JavaScript 程序，具体步骤如下。

- (1) 将光标移至<HTML>标记之前，然后在右边的“属性”面板中找到 DOCUMENT 对象的 defaultClientScript 属性，并将其设置为 JavaScript，此时在<head>部分出现如下代码。

```
<meta name=vs_defaultClientScript content="JavaScript">
```

这段代码用于设置页面所使用的脚本语言是 JavaScript。

- (2) 将光标移至要插入<script>块的位置（如在<head>标记之前的位置），然后单击鼠标右键，在弹出的快捷菜单中，选择“插入脚本块”/“客户端”命令，Microsoft 脚本编辑器将自动在当前



光标所在处生成<script>块标记，代码如下。

```
<script language=javascript>  
<!--  
  
//-->  
</script>
```

在上述代码中，<!--...!-->是 HTML 注释，而//是 JavaScript 的注释。使用这种注释标记的作用在于使较旧的、不支持脚本的浏览器也能正确显示该文档的其他部分。

(3) 编写 JavaScript 程序，在<!--...!-->之间的空行编写代码。例如，要显示“欢迎光临”字样，则代码如下。

```
<script language=javascript>  
<!--  
document.write("欢迎光临! ")  
//-->  
</script>
```

(4) 选择“文件”/“在浏览器中查看”命令，效果如图 1.9 所示。

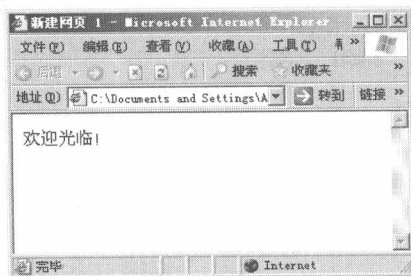


图 1.9 程序运行效果



提示

当输入对象名和句点（如 document.）时，Microsoft 脚本编辑器将自动弹出一个快捷菜单，提示对象所拥有的方法和属性。这一功能可以避免编码时写错方法名和属性名。

1.3.3 使用 Dreamweaver

Adobe 公司的 Dreamweaver 是建立 Web 站点和应用程序的专业工具。该工具可以将可视化工具应用程序开发功能与代码编辑组合在一起，并且内置了一些 JavaScript 小程序。

例 1.4 下面介绍如何使用 Dreamweaver 编写 JavaScript 程序。具体步骤如下。

(1) 安装 Dreamweaver 后，首次运行 Dreamweaver 时，展现给用户的是一个“工作区设置”对话框，在此对话框中，用户可以选择自己喜欢的工作区布局，如“设计者”或“代码编写者”，如图 1.10 所示。这两者的区别是在 Dreamweaver 的右边或是左边显示窗口面板区。

(2) 选择工作区布局，并单击“确定”按钮后。选择“文件”/“新建”命令，将打开“新建文档”对话框。在该对话框中的“类别”列表区选择“基本页”选项，再根据实际情况来选择所应用的脚本语言，这里选择的是“HTML”，如图 1.11 所示，然后单击“创建”按钮，创建以 JavaScript 为主脚本语言的文件。

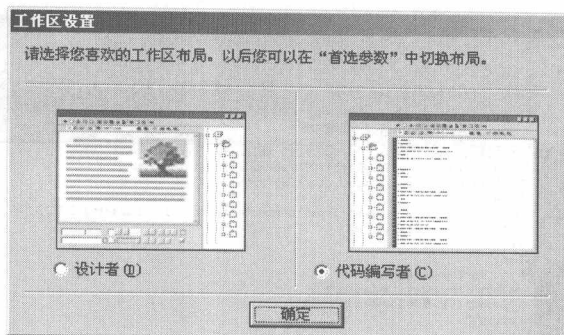


图 1.10 “工作区设置”对话框

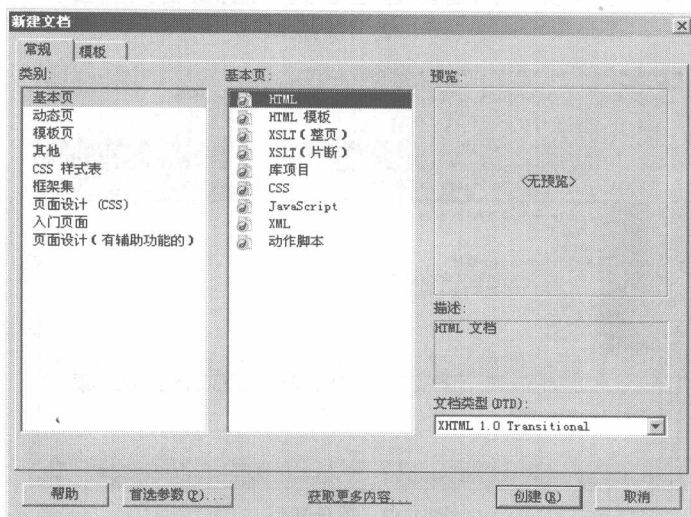


图 1.11 “新建文档”对话框



说明

如果用户选择了“JavaScript”选项，则创建一个 JavaScript 文档。在创建 JavaScript 脚本的外部文件时不需要使用<script>标记，但文件的扩展名必须是.js。调用外部文件可以使用<script>标记的 src 属性。如果 JavaScript 脚本外部文件保存在本机中，src 属性可以是完整路径或是部分路径；如果 JavaScript 脚本外部文件保存在其他服务器中，src 属性需要指定完整的路径。

(3) 在打开的页面中，有 3 种视图形式，分别为代码、拆分和设计。在代码视图中，可以编辑程序代码，如图 1.12 所示；在拆分视图中，可以同时编辑代码视图和设计视图中的内容，如图 1.13 所示；在设计视图中，可在页面中插入 HTML 元素，进行页面布局和设计，如图 1.14 所示。



注意

在代码视图中编写的 JavaScript 脚本，在设计视图中不会输出显示，也没有任何标记。

在 Dreamweaver 中插入 HTML 元素后，通过“属性”面板可以方便地定义元素的属性，使其满足页面布局的要求。在页面中，允许多个表格的嵌套；可以插入图像、flash 等；可以插入表单元素，如文本框、列表/菜单、复选框、按钮等。