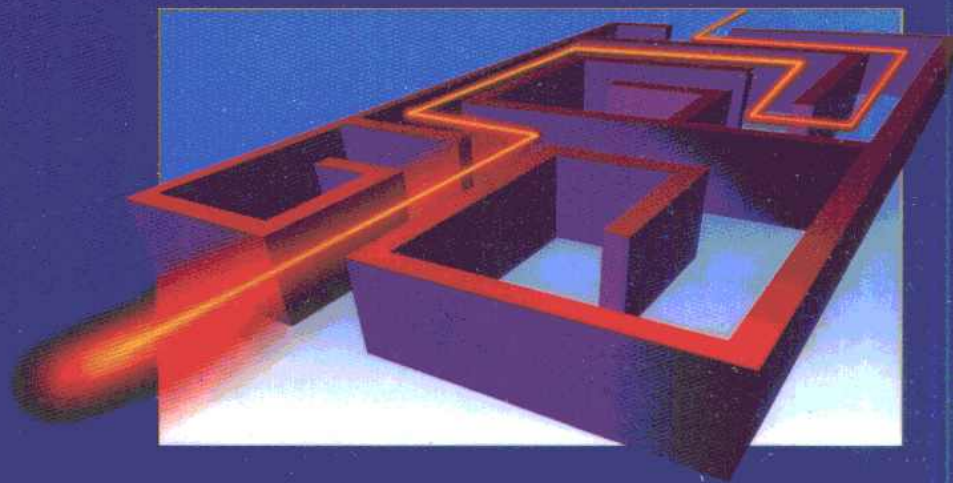


教给你编程起步的基本技能

JavaScript

编程起步

- 学习用JavaScript编程的基础知识
- 编写表单、框架、动画的实现脚本
- 按部就班地学习代码的调试



[美] John Pollock 著
云巅工作室 译

人民邮电出版社
www.pptph.com.cn

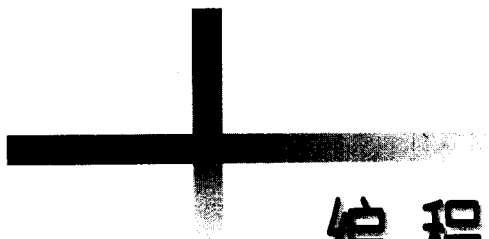
麦格劳-希尔国际公司
www.mhhe.com

Mc
Graw
Hill

284

TP3-059
587

JavaScript



编程起步

[美] John Pollock 著

云巅工作室 译

人民邮电出版社 麦格劳-希尔国际公司



图书在版编目 (CIP) 数据

Java Script 编程起步/(美)波洛克 (Pollock,J.) 编著;云巅工作室译.

—北京:人民邮电出版社,2001.9

ISBN 7-115-09621-X

I.J... II.①波...②云... III.JAVA 语言—程序设计 IV.TP312

中国版本图书馆 CIP 数据核字 (2001) 第 057713 号

JavaScript 编程起步

◆ 著 [美] John Pollock

译 云巅工作室

责任编辑 陈冀康

◆ 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@pptph.com.cn

网址 <http://www.pptph.com.cn>

读者热线 010-67129212 010-67129211(传真)

北京鸿佳印刷厂印刷

新华书店总店北京发行所经销

◆ 开本:787×1092 1/16

印张:30.25

字数:725 千字

2001 年 9 月第 1 版

印数:1—4 000 册

2001 年 9 月北京第 1 次印刷

著作权合同登记 图字:01—2001—3053 号

ISBN 7-115-09621-X/TP·2445

定价:49.00 元

本书如有印装质量问题,请与本社联系 电话:(010)67129223

第 1 章

JavaScript 简介

本章目的:

- 理解文本编辑器在 JavaScript 脚本编程中的作用;
- 理解 Web 浏览器在 JavaScript 脚本编程中的作用;
- 推荐用于编写 JavaScript 脚本程序的文本编辑器和 Web 浏览器;
- 简要介绍 JavaScript 脚本语言;
- 理解如何在 HTML 中嵌入 JavaScript 脚本。

欢迎学习《JavaScript 编程起步》！显然，你是有兴趣学习 JavaScript 脚本语言的，但可能你还不能确定如果要使用它你需要先了解或掌握哪些知识。不用担心，本章将告诉你学习和使用 JavaScript 脚本语言过程中所需的基本概念和相关知识，比如说什么是 JavaScript 脚本语言，它有哪些优点和局限，如何使用它来创建动态 Web 网页和访问 Web 网页。另外，本章还将简要介绍 JavaScript 脚本语言的发展历史。

JavaScript 在 WWW 中无处不在，它可以使你的网页交互性更强，因为它允许页面对浏览者的行为做出反应，你也可以用它在你的页面上做一些特效（在视觉或其它方面）。

JavaScript 作为一种被推荐给初级 Web 开发者的语言，经常用于 HTML（Hypertext Markup Language）中（无论是出于商业目的还是因兴趣而创建网站）。当然，你也可以只使用 HTML 来创建 Web 页，但 JavaScript 允许你添加进附加的特性，而静态的 HTML 网页若没有某种脚本或程序的帮助是不可能提供这种功能的。

1.1 学习本书所需的基本知识

在学习 JavaScript 脚本语言之前，你应该先了解下列基本知识：

- HTML（超文本标记语言）；
- 文本编辑器；
- Web 浏览器；
- JavaScript 脚本语言的不同版本。

如果你已经具备了上述的基本知识（相信应该如此），那么你就可以轻松地学好本书所介绍的内容了。学习本书并不需要你先掌握其它的编程语言或脚本语言，也不需要你已经具备用 JavaScript 编程的经验。本书的对象就是 JavaScript 脚本语言的初学者。

如果你对上述基本知识的某一方面还不甚了解，那也没有关系。下面各小节将简要介绍上述的几个基本知识点。

1.1.1 HTML（超文本标记语言）的基本知识

即使不想成为 HTML 的编程高手，你仍需知道该在 HTML 文本中的哪个位置编写哪部分内容（如 head 部分和 body 部分），并且知道如何添加你自己的属性。本书将经常提到 head 部分（HTML 文本中<HEAD>标记和</HEAD>标记之间的那部分内容）和 body 部分（HTML 文本中<BODY>标记和</BODY>标记之间的那部分内容）的脚本。

有时候，你需要将某一属性添加到某一标记中，以使脚本执行适当的功能。例如，你可能需要用 name 属性命名一个表单（form）元素，可以使用如下代码：

```
<INPUT type="text" name="thename">
```

如果你已经掌握了如何使用标记和属性的基本知识，那么在学习 JavaScript 时，有关 HTML 的那部分内容将不会给你带来任何问题。

如果你不具备 HTML 的基本知识，建议你通过其它方式简要地学习一下以帮助你更好

地学习本书。在此给你介绍两个途径，一是阅读由 Osborne 出版社 2000 年出版的由 Wendy Willard 编写的《HTML A Beginners Guide》(《HTML 编程起步》，人民邮电出版社)；二是通过因特网获得有关 HTML 语言的帮助信息，如可以访问如下网站：<http://www.pageresource.com/html/> 和 <http://htmlgoodies.earthweb.com>。

1.1.2 文本编辑器和 Web 浏览器的基本知识

在学习和使用 JavaScript 脚本语言编程之前，你必须掌握如何使用文本编辑器或 HTML 编辑器，同时还要会使用 Web 浏览器。你将使用这些工具来编写你的脚本。

一、文本编辑器

很多文本编辑器和 HTML 编辑器都支持 JavaScript 脚本¹。如果你了解 HTML 代码，你可能已经使用过 HTML 编辑器来创建 HTML 文件，那么你可以继续使用这个编辑器。

然而，有一些 HTML 编辑器在你加入 JavaScript 代码时可能会带来一些问题。比如当你保存那些添加了 JavaScript 代码的 HTML 文件时，HTML 编辑器可能会自动地改变代码的位置或改变代码本身而造成错误。这时，建议你使用一个简单的编辑器，或寻找一个能够轻松处理用户添加的 JavaScript 代码的 HTML 编辑器，例如 Notepad、TextPad 和 Simple Text。

二、Web 浏览器

同样，如果你已经编写过 HTML 代码，或使用过 Web 浏览器来检验你创建的 HTML 文件，那么你也可以继续使用这个 Web 浏览器。但是，有一些 Web 浏览器可能会不支持新版本的 JavaScript 语言。当然，只要这个 Web 浏览器与 JavaScript 兼容，那么最终采用什么样的 Web 浏览器，由你自己来决定。建议使用如下两者之一来验证你所编写的 JavaScript 代码的正确性：

- 微软公司的 Microsoft Internet Explorer 4.0 或更高版本；
- 网景公司的 Netscape Navigator 4.0 或更高版本。

这两种 Web 浏览器的新版本还将不断地推出。截止到写作本书之时，最新的版本为 Internet Explorer 5.5 和 Netscape Navigator 4.7。

为了使你对这两种 Web 浏览器有一个直观的认识，图 1-1 显示了用 Microsoft Internet Explorer 浏览器看到的一个 Web 页，图 1-2 显示了用 Netscape Navigator 浏览器看到的同样的 Web 页。

如果你正在使用一个老版本的 Web 浏览器而且又不能升级，那么可能会有不少特性（大部分将在后续的章节中介绍）无法在你的 Web 浏览器中正常显示。即使如此，如果只能使用老版本的 Web 浏览器，你也没有必要放弃对 JavaScript 脚本语言的学习，这本书仍将帮助你有效地自学。

上面推荐的那两个主要的 Web 浏览器的不同版本和它们所支持的 JavaScript 语言版本如表 1-1 所示。

¹ JavaScript 脚本就是 JavaScript 程序，或称 JavaScript 代码段。

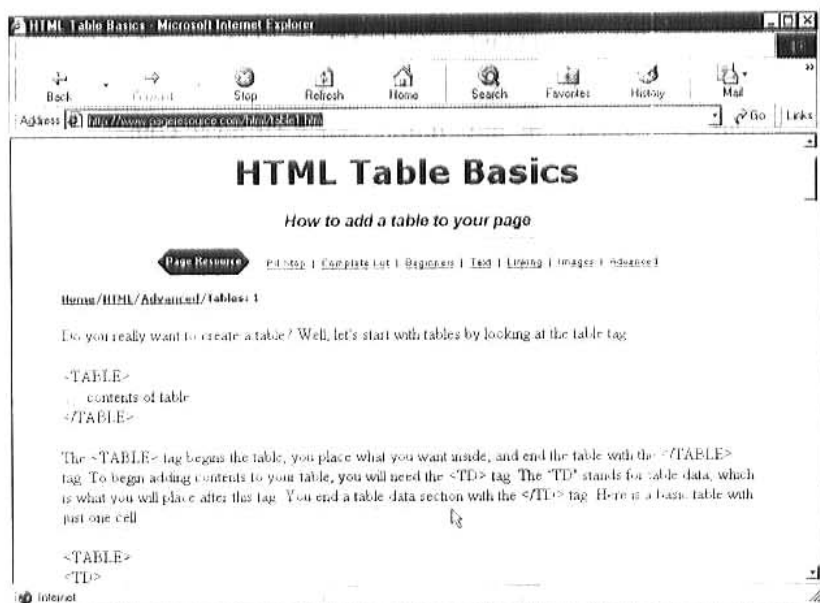


图 1-1 用 Microsoft Internet Explorer 浏览器看到的 Web 页

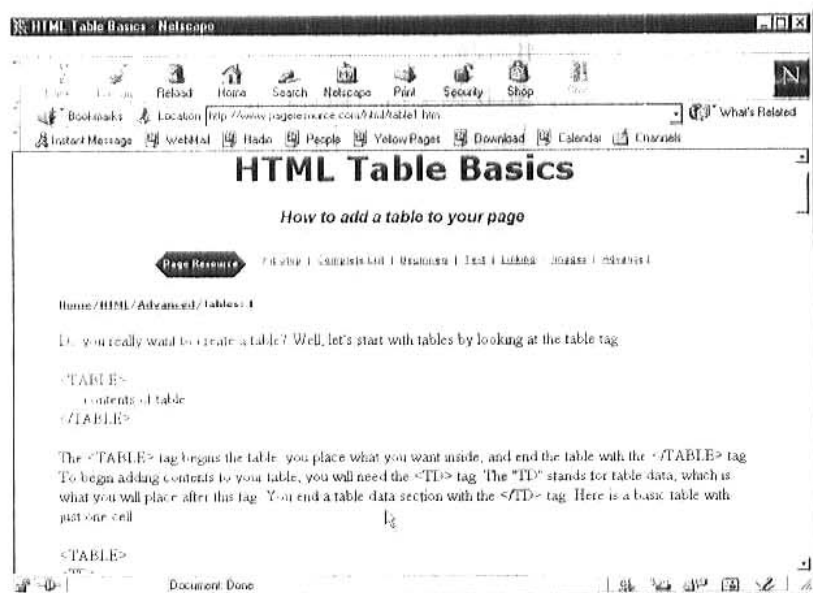


图 1-2 用 Netscape Navigator 浏览器看到的 Web 页

表 1-1 两种主要的 Web 浏览器所支持的 JavaScript 版本

Microsoft Internet Explorer 版本	Netscape Navigator 版本	所支持的 JavaScript 版本
5.0	4.5	1.3
4.0	4.0	1.2
3.0	3.0	1.0/1.1

下一小节“JavaScript 语言的不同版本”将更详细地解释版本号的含义。参见以下在线资源：<http://www.webreview.com/browsers/browsers.html>，可获得完整列表（包括许多常用浏览器，并提出了使用平台问题，即在相同的浏览器上，哪一版本用于 Windows 平台，哪一版本用于 Macintosh 平台）。

如果你已经确定自己符合学习 JavaScript 语言的基本要求，你就可以开始学习这门语言了。

1.1.3 JavaScript 语言的不同版本

本书编写之时，在本章前面所推荐的那两个 Web 浏览器都至少支持 JavaScript 1.2 版本，一些更高版本的浏览器还能支持 JavaScript 1.3 版本。

版本为 3.0 及更早版本的浏览器仅支持 JavaScript 1.0 或 1.1 版本，这也是本书推荐使用上述两种浏览器 4.0 或以上版本的原因。另外，JavaScript 1.4 版本目前也已经推出了，并且可以在网景公司的网站上（<http://developer.netscape.com/docs/manuals/js/core/jsguide/index.htm>）查到相关的资料。

你可能也见到过或听说过 JScript 或 ECMAScript，这两个又是什么呢？JScript 是微软公司在其 Web 浏览器 Microsoft Internet Explorer 中所采用的 JavaScript 版本，由于它是以 Windows 脚本引擎（Windows Script Engine）的方式实现的，因此它具有一些附加的特性，它可以使用服务器端语言（server-side language）来完成类似于更新数据库那样的复杂任务。如果你想进一步了解 JScript，可以查看如下网站：<http://msdn.microsoft.com/scripting/default.htm>。

ECMAScript 是 JavaScript 脚本语言的国际标准名称和规范，所以它并不是一种新的语言，而是一个要求 JavaScript 和 JScript 必须遵循的国际标准。如果你想进一步了解 ECMAScript，请查看网站 <http://www.jsworld.com/ecmascript>。

1.2 切记，JavaScript 不是 Java

JavaScript 和 Java 是两种不同的语言。Java 是一种完整的程序设计语言。一个 Java 程序（通常叫做 Java applet）要变得可执行，必须先通过编译。Java 的功能要比 JavaScript 强大得多，但也要复杂得多。JavaScript 语言则不需要一个专门的编译器，并且它在许多方面要比 Java 语言宽松得多，比如语法。

1.3 与其它语言的相似之处

JavaScript 语言与其它程序设计语言和脚本语言也有相似之处。如果你已经有了 Java、C++ 或者 C 语言的程序设计经验，那么你将发现 JavaScript 语言与它们在语法上的一些相似之处，这有助于你更好、更快地学习。由于 JavaScript 语言是一种脚本语言，它更类似于 Perl 语言那样，直接通过解释器就可以执行，而不需要执行前的编译。

如果你事先已经有了使用其它语言编写程序的经验，那么学习掌握 JavaScript 语言对你来说很轻松，不过，即使你没有任何程序设计经验，你也同样可以学好 JavaScript 语言。



请教专家

问：您说我可以根据我的选择使用一个文本编辑器或 **HTML** 编辑器，但我还是不很清楚这到底是什么意思。什么是文本编辑器？我在哪里可以找到它？

答：文本编辑器是一个用来编辑和保存文本的程序。文本编辑器有简单的，也有复杂的，如 Notepad、WordPad、Simple Text、Microsoft Word、Corel WordPerfect Millennium Edition 等等。另外，你还可以去购买或从因特网上下载一些像 NoteTab 或 TextPad 那样的文本编辑器。

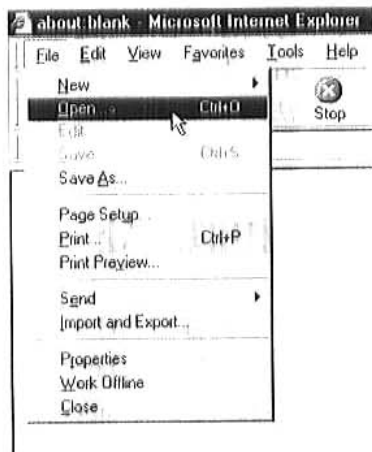
HTML 编辑器不仅是一个更加复杂的文本编辑器，而且只需通过点击按键或其它方式就可以增加代码，这就是所谓的“所见即所得”（WYSIWYG）。我所推荐的文本编辑器或 HTML 编辑器是那种不会改变你所手动加进去的代码的编辑器，如 Microsoft FrontPage、Allaire HomeSite 和 Macromedia Dreamweaver。

问：使用文本编辑器，我该掌握哪些知识？

答：简单地说，使用文本编辑器，你只要知道如何在编辑器中输入普通文本，以扩展名 .html 或 .htm 来保存文件、在需要的时候重新打开文件并编辑它即可。由于 HTML 文件本身由普通文本构成，因此你可以不必了解文本编辑器的其它特性。

问：使用 **Web** 浏览器，我需要了解哪些知识？

答：使用 Web 浏览器，你该知道的仅仅是如何在你的计算机（或网站上）上打开一个本地的 HTML 文件，以及如何重新装载一个页面。如果你不知道如何从自己的计算机中打开一个 HTML 文件，那么打开你的浏览器找到 File 菜单，在该下拉菜单中找到写着类似于“Open...”或“Open File”的选项，然后点击它。你可以像在其它软件中打开一个文件那样来浏览你所打开的 HTML 文件。下图显示了如何在 Microsoft Internet Explorer 5.5 中找到打开文件的选项：



问：我可以从哪里获得你所推荐的浏览器？

答：你可以从下列网站下载：

- 微软公司的 Microsoft Internet Explorer: <http://www.microsoft.com/ie>。
- 网景公司的 Netscape Navigator: <http://home.netscape.com/computing/download>。

1.4 JavaScript 语言简介

JavaScript 语言是由 Netscape Communications Corporation 和 Sun Microsystems, Inc. 联合开发的。1995 年 12 月 4 日，它们联合发布了这个新语言，当 Netscape Navigator 还是 2.0 版测试版时就支持这种语言（可通过网页 <http://home.netscape.com/newsref/pr/newsrelease67.html> 查看有关它发布的新闻）。不久，JavaScript 1.0 就在新版本的浏览器中采用了。另外，该语言在被命名为 JavaScript 之前，称为 LiveScript。

JavaScript 是一个基于对象的客户端脚本语言。利用它来编写网页，可以使网页能够根据情况动态地发生改变。为了更好地理解这个定义，下面我们来逐个解释它的各个重要部分。

1.4.1 基于对象 (Object Based)

基于对象是指 JavaScript 语言可以使用一个叫做对象的项。但是，这种对象并不是基于类的（也就是说类和实例是没有区别的），相反，它们只是普通对象¹。我们将在第 8 章中详细介绍如何使用 JavaScript 对象。在掌握 JavaScript 语言其它的一些特性之前，你没有必要详细了解这个概念。

1.4.2 客户端 (Client Side)

客户端是指 JavaScript 脚本是在浏览器所使用的客户程序中执行的，而不是在网站的 Web 服务器上先解释执行再送到用户的浏览器中显示的。在这种情况下，客户程序就是 Web 浏览器本身。为了更好地理解这个概念，让我们来比较一下服务端语言和客户端语言分别是如何工作的。

一、服务器端语言

服务器端语言需要从网页或 Web 浏览器中获得信息，然后将这些信息传递给在主服务器端运行的某个程序，并由这个程序来产生相应的结果，送回到 Web 浏览器。因此，在 Web 浏览器中，想看到程序的结果必须通过一个中间步骤，即先将数据送给服务器，再由服务器产生结果后送回 Web 浏览器。

服务器端语言常常给程序员一些客户端语言所没有的好处，比如将数据保存在 Web 服务器中以备后用，或者使用新数据来更新网页并保存更新的数据。

但是，服务器端语言可能在实现 Web 浏览器窗口特性的能力方面比较有限，而客户端语言就可以对 Web 浏览器窗口进行灵活操作。比如 Web 浏览器窗口的状态栏中的内容的

¹ 这与“面向对象”中所说的对象是不同的。

动态显示，或者判断在提交给服务器之前表单中的各种数据是否符合要求等等。

二、客户端语言

客户端语言编写的脚本可以直接被浏览者所使用的客户程序运行。对 JavaScript 语言而言，客户程序就是 Web 浏览器本身。因此，JavaScript 脚本可以直接在 Web 浏览器中执行，而不需要通过额外的步骤来向 Web 服务器发送数据并等待其发回的数据以显示结果。

使用客户端语言，浏览器直接读取并解释执行代码，产生的结果直接返回浏览器而不需要先从服务端获得数据。这种处理方式可以使某些任务得以快速地完成。

同时，客户端语言还可以直接对 Web 浏览器窗口的特定属性进行操作，这对于服务端语言来说是不可能的。但是，客户端语言缺乏在 Web 服务器上保存文件或更新文件的能力，而这些对服务端语言来说却是轻而易举的。

对于处理像改变 Web 浏览器窗口的某个属性，或者是在提交给服务端程序或脚本处理之前检验各种数据是否符合要求这样的任务，客户端语言就显得非常有用。例如，JavaScript 可以打开一个特定大小、具有特定控件（如工具栏或状态栏等）的窗口，并且可以将这个窗口放置在计算机屏幕的特定位置。

JavaScript 也可以用来检验在提交给服务端程序处理之前，表单中的各种信息是否符合要求，这种方法可以防止将不准确的或不完整的数据提交给 Web 服务器处理，从而避免服务器出现异常。或者说，与其让服务器不断地来分析、处理提交的各种数据的正确性和完整性，还不如在客户端先进行判断，然后一次性将正确的数据传送给服务器。

1.4.3 脚本语言

脚本语言所编写的程序在运行之前并不需要先进行编译，所有的代码都是在客户程序中直接被解释执行的。

用一个常规的程序设计语言进行编程，你必须经历这样的一个过程：先编写代码，接着用一个特定的编译器对代码进行编译以确保它在语法上没有任何错误并生成目标代码，然后再利用链接程序将其链接成可执行文件，最后运行可执行文件。而使用一个脚本语言进行程序设计时，你所编写的代码在客户程序装载它的时候就直接被解释执行了。这样，你就可以更快更直接地检查你所编写的程序是否能得到期望的正确结果。

然而，在脚本运行之前错误是会被发现的，如果脚本不能把错误处理好，就会给用户带来问题。对于 JavaScript 脚本来说，错误的处理取决于用户所使用的浏览器。

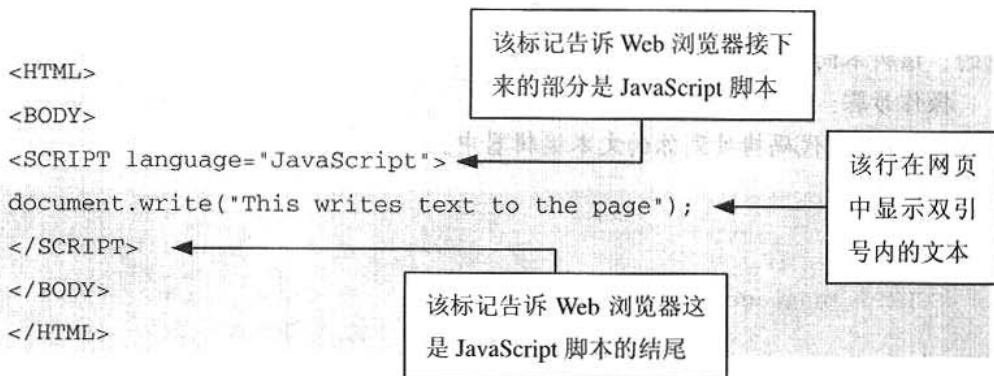
1.5 用 JavaScript 语言编写脚本

有了上面所讲述的概念之后，你可能还想知道 JavaScript 脚本是如何在浏览器中运行的，该在哪个地方添加你编写的 JavaScript 代码，浏览器又是如何在一个网页中将你编写的 JavaScript 代码与其它部分区分开来的等等。下面将简要地回答这些问题，但在下一章中我们将详细地讲述。

说起来也非常简单，只要将 JavaScript 脚本直接加入到一个已有的 HTML 文件中就可以被浏览器执行了。但在加入的过程中，你需要在 HTML 代码中加入一个特定的标记和命

令来告诉浏览器将要运行一个脚本。一旦浏览器执行到这些标记和命令时，它就会将你所编写的 JavaScript 代码逐个解释执行。这样，你只需要简单地对一个 HTML 文件进行编辑，就可以开始在你的网页中使用 JavaScript 语言并在浏览器中查看其结果了。

我们举例说明。在下面的 HTML 文件所显示的代码中，我们加入了一些 JavaScript 代码。这些 JavaScript 代码的功能是在网页中写一些文本内容。注意，我们在代码中添加了 `<SCRIPT>` 和 `</SCRIPT>` 标记，在这两个标记之间就是 JavaScript 代码。



在下一章中我们将讲述在一个 HTML 文件中，如何使用 `<SCRIPT>` 和 `</SCRIPT>` 这两个 HTML 标记来添加 JavaScript 代码。这是你想成为 JavaScript 程序员所必须迈出的第一步！

1.6 在线资源

为了使你能够获得更多的有关 JavaScript 的信息，下面列出一些非常有用的在线资源：JavaScript 语言编程实例的教学网站：

<http://www.pageresource.com/jscript>

包含大量 JavaScript 脚本剪辑的自学网站：

<http://www.wsabstract.com>

可提供专家咨询的 JavaScript 程序员/爱好者网站：

<http://www.javascriptcity.com/forums>



一分钟练习

- 什么是 JavaScript 语言？
- JavaScript 语言中的对象是基于类的吗？

- JavaScript 脚本能够在 Web 服务器上保存文件吗？



项目 1-1：使用 JavaScript 来输出文本

本项目将告诉你当 Web 浏览器打开一个 HTML 文件时 JavaScript 脚本是怎样被执行的。该脚本向网页输出一行特定的文本。

操作步骤：

1. 将下列代码拷贝到你的文本编辑器中。

```
<HTML>
<BODY>
<SCRIPT language="JavaScript">
<!--
document.write("This text was written with JavaScript!");
//-->
</SCRIPT>
</BODY>
</HTML>
```

2. 将该段代码保存为 pr1_1.html 文件，然后用你的 Web 浏览器打开这个文件。此时你就可以看到，在这个网页中有一行用 JavaScript 脚本输出的文本。

1.7 小测验

1. 要使用 JavaScript 语言，你必须了解下列哪些内容？
 - A. Perl
 - B. C++
 - C. HTML
 - D. SGML
2. 要使用 JavaScript 语言，你必须使用下列哪些工具或设备？
 - A. Web 浏览器
 - B. C++ 编译器

- JavaScript 是一个基于对象的客户端脚本语言，利用它可以使你的网页能够根据情况动态地发生改变。
- 不是。
- 不能。

- C. 50GB 的硬盘
 - D. 光盘刻录机
3. 在网页中使用表格时，客户端语言起什么作用？
- A. 它能将数据保存到服务器中
 - B. 它能将数据在送到服务器之前先检验其正确性和完整性
 - C. 它能更新文件并将新的数据保存到文件中
 - D. 根本不起作用
4. JavaScript 脚本是怎样添加到网页中的？
- A. 在浏览器的一个特殊的编辑器中添加
 - B. 利用服务器上的一个编译器来添加
 - C. 将代码自动地添加到一个文件中并打开它
 - D. 添加到一个 HTML 文件中

第 2 章

在 HTML 文件中添加 JavaScript 代码

本章目的:

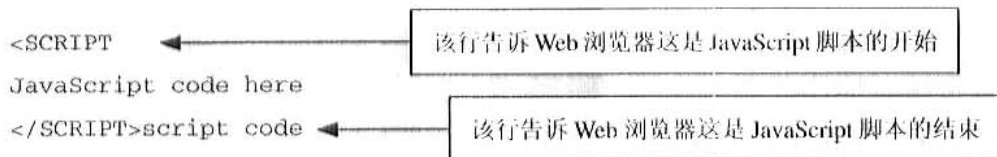
- 了解 SCRIPT 标记的功能;
- 掌握如何将 JavaScript 脚本添加到 HTML 代码中;
- 学会如何从 HTML 代码中调用 JavaScript 外部脚本文件;
- 学会如何在代码中添加 JavaScript 注释。

既然你已经对 JavaScript 脚本语言有了大致的了解，现在就开始讲述如何进行 JavaScript 语言的编码。由于 JavaScript 代码是通过 HTML 文件来执行的，所以你需要知道如何让 Web 浏览器运行你编写的脚本。最普通的方式就是用一组 HTML 语言中的 `<SCRIPT>` 和 `</SCRIPT>` 标记标注出来，表明它们之间的那部分代码是一个脚本。然后将脚本添加到你的 HTML 文件的 HEAD 或 BODY 部分。

本章先讲述如何利用 SCRIPT 标记来设置 JavaScript 代码段的开始和结束；然后教你开始创建和运行你的第一个 JavaScript 脚本；本章的最后将讲述如何添加 JavaScript 注释以说明你的脚本。

2.1 使用 HTML 语言中的 SCRIPT 标记

SCRIPT 标记是用来告诉浏览器在一个 HTML 文件中某一种类型的脚本语言的代码段是从哪开始又是在哪结束的。SCRIPT 标记的基本形式与 HTML 语言中其它标记的形式非常相似。如：



正如你所看见的，JavaScript 脚本以 `<SCRIPT>` 标记开始，接着是 JavaScript 代码，最后以 `</SCRIPT>` 标记结尾。当你的脚本采用像上面所示的开始和结束标记时，许多 Web 浏览器就假设它们中间的代码采用的是 JavaScript 脚本语言。但是，有一些 Web 浏览器则需要你告诉它你所采用的是哪一种脚本语言。

SCRIPT 标记除了可以为 Web 浏览器指示一个脚本的开始和结束之外，还可以告诉浏览器所采用的脚本语言的类型和定义一个 JavaScript 外部脚本文件的位置。这两个附加的功能是通过分别设置 `language` 和 `src (source)` 属性实现的。

2.1.1 定义脚本语言的类型

在 SCRIPT 开始标记和结束标记之间的代码可以采用 JavaScript 脚本语言、VBScript 脚本语言和一些其它的脚本语言。尽管 Web 浏览器通常采用 JavaScript 作为它默认的脚本语言，但仍有可能有一些 Web 浏览器并不是这样。为了安全起见，清楚地定义脚本语言的类型为 JavaScript 是一个很好的主意。为此，你只需要在 SCRIPT 开始标记中增加一个 `language` 属性，并将该属性的值设为“JavaScript”。如下所示：

