



王萍萍 赵俊莉 孙 强 编著
周 峰 审校

JavaScript

基础与实践教程



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

JavaScript基础与实践教程

王萍萍 赵俊莉 孙 强 编著

周 峰 审校

電子工業出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书采用案例讲解的方式系统介绍了JSP应用开发技术,包括JSP开发环境的安装与配置、JSP基本编程基础、JSP数据库编程、JSP多媒体编程、Servlet技术、JSP与JavaScript的结合、JSP中XML的应用和表达式语言与标准标签库等几个方面的内容。此外,通过一个实际开发案例,使读者能够结合实际,快速、高效、灵活地设计出专业的企业级应用程序。

本书着重于JSP技术的实用性和可操作性,本书的作者融合了多年JSP实际开发中遇到的各种常见问题和编程技巧,并结合网上各个技术论坛中讨论的热点技术核心问题,针对目前读者的实际需求进行了内容结构的设计和分类,通过实例剖析技术的具体应用,使读者能十分容易入门并逐步精通。

本书可作为初学者的入门教程,更适用于有一定Web编程基础的读者,本书涉及到不同类型的应用,多角度地引导读者学习相关知识。通过本书学习,读者可以迅速提高自己的编程水平,达到实际商业开发水平的要求。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

JavaScript基础与实践教程/王萍萍,赵俊莉,孙强编著.一北京:电子工业出版社,2007.6
ISBN 978-7-121-04023-8

I. J… II. ①王… ②赵… ③孙… III. Java语言—程序设计—教材 IV. TP312

中国版本图书馆CIP数据核字(2007)第037192号

责任编辑:徐云鹏

特约编辑:底波

印刷:北京天竺颖华印刷厂

装订:三河市金马印装有限公司

出版发行:电子工业出版社

北京市海淀区万寿路173信箱 邮编:100036

北京市海淀区翠微东里甲2号 邮编:100036

开本:787×1092 1/16 印张:36.375 字数:930千字

印次:2007年6月第1次印刷

定价:55.00元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至zltz@phei.com.cn,盗版侵权举报请发邮件至dbqq@phei.com.cn。

服务热线:(010)88258888。

前 言

JavaScript是一种解释性的, 基于对象的脚本语言, 它的功能强大、语法简单, 广泛应用于网页设计中。使用JavaScript结合HTML语言, 可以实现丰富的动态效果。同时, JavaScript也是ASP、ASP.NET、PHP、JSP等网络服务器端编程语言需要的客户端搭配语言, 是进行网页设计所必须掌握的基础脚本语言之一。

本书详细介绍了JavaScript的语言基础和大量丰富的实例, 比较全面地介绍了JavaScript的应用。学习内容由浅入深的, 每学完一部分, 读者的JavaScript编程水平就前进了一步。全书共12章, 具体内容如下。

第1章 JavaScript语言概述

本章介绍JavaScript的发展历史和特点, JavaScript的应用领域, JavaScript的编写工具。本章的内容让读者对JavaScript有一个总体的认识, 为学习后续章节做铺垫。

第2章 JavaScript程序设计基础

本章介绍JavaScript的语法规则、数据类型、变量、表达式和运算符、流程控制、函数、正则表达式和高级特性。目的是让读者重点掌握JavaScript的基本语法知识, 为后面的编程奠定基础。

第3章 面向对象的JavaScript

本章介绍JavaScript语言基于对象(Object-Based)的重要特性, 包括对象的基本概念和JavaScript的常用内置对象。这些内置对象能帮助网页编程人员编写更精彩、更实用的页面, 大大简化了JavaScript的程序设计, 使其以更直观、模块化和可重用的方式进行程序开发。

第4章 JavaScript程序调试

本章介绍JavaScript编程过程中的错误及其类型, 针对错误不可避免的情况, 介绍了基本的、常用的调试方法和高级调试方法。本章首先阐述了JavaScript编程过程中出现的错误类型: 语法错误、运行错误、逻辑错误, 然后重点讲解JavaScript的调试技术, 包括基础的、常用的调试方法和高级调试方法。

第5章 Web基础

本章主要介绍两部分内容, 第一节主要介绍了HTML语言的基本知识; 第二节是重点内容, 主要介绍了浏览器对象, 包括浏览器对象模型、window对象、document对象、history对象、navigator对象、location对象、frame对象、screen对象。

第6章 CSS样式

本章主要讲解CSS样式表的概念，从CSS样式的定义出发，由浅入深地讲解了CSS选择器的分类，如何在HTML网页中使用CSS，还重点讲解了CSS的各种属性，包括字体属性、文本属性、颜色和背景属性、容器属性、列表属性、鼠标属性、定位和显示。还通过一个构建图片查看器的例子，讲解如何综合使用CSS。通过本章的学习，读者会对CSS样式表有深刻的理解，本章的实例也会使读者进一步提高水平。

第7章 JavaScript的DOM编程

本章主要包括两方面的内容，一是对XML的概述、语法、元素、XSL、数据岛、解析器的介绍，应用JavaScript对XML文档进行操作；二是介绍了DOM文档对象模型，通过DOM对象的相关属性和方法，应用JavaScript对HTML文档和XML文档进行操作。

第8章 JavaScript基础应用

本章主要介绍了使用JavaScript来实现表单对象的属性方法和事件、Cookie的使用、常用的网页特效、窗口控制、警告和提示功能及密码特效。本章为JavaScript基础应用，是进行高级应用的基础和前提。

第9章 JavaScript服务器端应用

本章介绍基于ASP的运行环境，将JavaScript语言应用到服务器端，制作交互的动态Web应用程序。学习如何在服务器端应用JavaScript脚本语言以及ASP内置对象的使用。

第10章 JavaScript数据库应用

本章在上一章JavaScript服务器端应用的基础上，介绍了JavaScript数据库应用，讲解如何应用ASP的数据库对象（ADO）访问数据库，如何用JavaScript语言来控制数据库的访问。以Access数据库为例，通过实例介绍如何对数据库进行连接、查询等操作。

第11章 JavaScript高级应用

本章通过实现万年历、树型菜单、评估用户输入密码的强度、跟随鼠标移动的时间效果、统计用户来访次数这五个综合实例，介绍了应用JavaScript制作网页特效的过程，并对相应的代码做了详细说明。

第12章 在线测试系统

本章应用JavaScript + ASP + Access数据库技术构建了一个在线测试系统，分析了在线测试系统的系统设计流程，数据库设计和页面之间的关系。在功能上，实现了页面之间数据的传递和共享，数据库的查询、删除、添加、修改等操作和数据库信息分页显示。在设

计思想上，目的是让读者从整体上了解“在线测试系统”这样一个完整的系统，学会如何处理系统流程设计、实现过程、页面之间的关系。

附录

A JavaScript参考

本附录内容给出了JavaScript语言的关键点的参考。

B. CSS参考

本附录给出了CSS所用到的各种属性的参考。

C W3C DOM参考

本附录给出了W3C通过的DOM的主要属性和方法，供读者参考。

本书在编写过程中力求理论和实践紧密结合，采用大量典型实例，并做了详细讲解，使读者快速高效掌握JavaScript的编程方法。书中的综合实例可以帮助读者学习大型系统的设计与开发。

本书由王萍萍、赵俊莉、孙强编著，周峰审校。以下人员对本书的编写提出过宝贵意见并参与了本书的部分资料搜集工作，他们是孙更新、孙宁、王荣芳、李德路、李永杰、韩亚男、陈卓、王伟、姚国发。感谢北京美迪亚电子信息有限公司的各位老师，谢谢你们的帮助和指导。

由于时间仓促，加之水平有限，书中的缺点和不足之处在所难免，敬请读者批评指正。

为方便读者阅读，本书配套资料请登录“华信教育资源网”（<http://www.hxedu.com.cn>），在“资源下载”频道的“图书资源”栏目下载。

目 录

第1章 JavaScript语言概述	1
1.1 JavaScript简介	1
1.2 JavaScript用途	9
1.3 JavaScript编写、运行	13
本章小结	21
本章习题	21
习题答案	22
第2章 JavaScript程序设计基础	23
2.1 语法规则	23
2.2 数据类型	28
2.3 变量	35
2.4 表达式和运算符	40
2.5 流程控制	54
2.6 函数	74
2.7 正则表达式	85
2.8 JavaScript高级特性	95
本章小结	99
本章习题	100
习题答案	103
第3章 面向对象的JavaScript	108
3.1 对象	108
3.2 原型对象与继承	117
3.3 JavaScript内置对象	121
本章小结	146
本章习题	147
习题答案	147
第4章 JavaScript程序调试	149
4.1 常用调试方法	149

4.2 高级调试技术	156
本章小结	162
本章习题	163
习题答案	163
第5章 Web基础	165
5.1 HTML	165
5.2 浏览器对象	191
本章小结	221
本章习题	221
习题答案	222
第6章 CSS样式	225
6.1 CSS简介	225
6.2 CSS属性	237
6.3 CSS滤镜	253
6.4 CSS+JavaScript构建图片查看器	256
本章小结	259
本章习题	260
习题答案	260
第7章 JavaScript的DOM编程	263
7.1 XML概要	263
7.2 XML基础	265
7.3 DOM	280
本章小结	296
本章习题	296
习题答案	297
第8章 JavaScript基础应用	300
8.1 表单验证	300
8.2 Cookie处理	315
8.3 网页特效	321
8.4 窗口控制	354
8.5 警告和提示功能	360
8.6 密码特效	365

本章小结	376
本章习题	376
习题答案	377
第9章 JavaScript服务器端应用	378
9.1 ASP概述	378
9.2 ASP脚本语言	383
9.3 ASP中内置对象	388
本章小结	407
本章习题	407
习题答案	408
第10章 JavaScript数据库应用	410
10.1 JavaScript连接数据库	410
10.2 JavaScript结合ASP动态显示查询结果	417
本章小结	429
本章习题	429
习题答案	430
第11章 JavaScript高级应用	434
11.1 实现万年历	434
11.2 实现树型菜单	439
11.3 评估用户输入密码的强度	446
11.4 跟随鼠标移动的时间效果	449
11.5 统计用户来访次数	455
本章小结	457
本章习题	457
习题答案	458
第12章 在线测试系统	462
12.1 系统主要功能模块	462
12.2 创建数据库	463
12.3 系统首页	464
12.4 学生测试模块	466
12.5 教师管理模块	496
本章小结	539

本章习题	539
习题答案	539
附录	548
A. JavaScript参考	548
B. CSS参考	560
C. W3C DOM参考	563

第1章

JavaScript语言概述

课前导读

随着Internet的飞速发展，越来越多的人每天都要访问各种不同的Internet站点。当你使用浏览器访问这些站点时，可能已经注意到呈现在你面前的页面是如此的五彩缤纷，既有静态的文本、图像，也有游动的动画、信息框以及不断变化的状态栏提示、动态变换的时钟信息等，甚至还有需要填写并提交的消息框，以及很多令我们眼前一亮的网页特效，如：滚动的字幕；鼠标移过时变换的图像；加载网页时自动弹出的窗口等。要实现页面上这些实时的、动态的、可交互的网页效果，可以使用JavaScript来编写。大家想自己来编写这些网页效果吗？那就赶快开始我们的JavaScript学习之旅吧！下面就先让我们来认识一下JavaScript。

重点提示

在本章我们将重点学习：

- * JavaScript的发展历程
- * JavaScript和Java的区别
- * JavaScript的特点
- * JavaScript的用途
- * 如何编写和运行JavaScript程序



1.1 JavaScript简介

JavaScript是目前非常流行的一种开发动态网页的脚本语言，它可以嵌入到HTML文档中，使网页更加生动活泼，并具有交互性。我们都知道，HTML语言是网页设计普遍采用的一种超文本标记语言，但HTML自身不能为网页提供很多动态支持，它只能创建超链接以允许用户浏览不同的HTML文档。HTML也不能接受用户输入，更不能对用户请求做出反应。因而，Netscape公司为了扩展浏览器的功能而开发了一种名为LiveScript的脚本语言，即JavaScript语言的前身。下面，就来详细地讲述JavaScript语言的发展过程。

1.1.1 JavaScript语言简史

谈到JavaScript的发展历史还得从Java谈起。

Java最初由Sun Microsystems公司为了在实时嵌入式系统使用而开发的。最初只有HotJava支持Java，而HotJava也正是Sun公司为支持Java而开发的一种实验性Web浏览器。Netscape在认识到Java的巨大潜力后，把它和自己开发的Web浏览器——Netscape Navigator结合到一起。由于Navigator是目前非常流行的浏览器，故Java得到了很大的普及。1995年，Java成为Internet

上最热门的新词，但很少有人真正了解如何用它编程。Netscape Communications认识到人们需要有一种用于Web页的简单明了的语言，于是引入了由Netscape Communications公司的Brendan Eich设计的一种编程语言——LiveScript，这是第一个Web脚本语言。它可以嵌入到Web页面中，能够处理数字和修改窗体的内容。

LiveScript的句法是以Java为基础的，但是更加简明扼要，也更易于掌握。同时，LiveScript是一种解释型语言，而Java则必须经过编译。Netscape公司将LiveScript嵌入Netscape Navigator的Beta版本中。1995年6月发布的Netscape Navigator 2.0b1版本开始支持LiveScript。

1995年11月，Netscape公司与Sun公司达成合作协议。Sun公司也认识到使用一种简单的脚本语言是一个好主意，于是他们正式认可了LiveScript。这样，LiveScript与Java的联系成为正式的，而LiveScript也改名为我们都熟悉的JavaScript。

注意：LiveScript被改名为JavaScript是因为Java的普及性。

接着Microsoft公司的浏览器Internet Explorer 3.0也开始支持JavaScript了。此后，许多公司相继宣布承认JavaScript为Internet上的开放式脚本编写标准，并且把它添加到了自己的产品中。因此，目前流行的浏览器都支持JavaScript。JavaScript日益流行起来。

1.1.2 对JavaScript的误解

JavaScript中带有“Java”前缀，因而很容易使人联想到Java，造成了一种误解：认为JavaScript是Java的子集或简化版；也很容易让初学者迷惑：它们到底是不是同一种语言？下面，我们就先来澄清这个问题。

事实上，JavaScript和Java虽然有紧密的联系，但它们却是两个公司开发的两种不同的语言。Java是Sun公司推出的新一代面向对象的程序设计语言，其前身是Oak语言，特别适合于Internet应用程序开发；而JavaScript最初是由Netscape公司开发的，前身是LiveScript语言，其目的是为了扩展Netscape Navigator功能而开发的一种可以嵌入Web页面中的基于对象和事件驱动的解释性语言。这两种语言在本质上也有许多不同之处，下面我们将它们的不同之处做如下比较：

1. 执行方式

Java和JavaScript最基本的区别在于它们的执行方式不同。

- JavaScript采用解释方式：JavaScript是一种解释性编程语言，其源代码在发往客户端执行之前不需经过编译，而是将文本格式的字符代码发送给客户，由浏览器解释执行。
- Java采用编译方式：Java的源代码在传递到客户端执行之前，必须经过编译，因而客户端上必须具有相应平台上的仿真器或解释器，它可以通过编译器或解释器实现独立于某个特定的平台编译代码的束缚。

2. 关于对象

Java和JavaScript支持面向对象程序设计的程度不同。

- JavaScript是基于对象的：JavaScript是一种基于对象和事件驱动的编程语言。它支持对象类型和实例，但是不支持类（创建类）、多态性和继承性。JavaScript主要用来访问

预定义对象，操纵它们的属性、方法和关联事件。它本身提供了非常丰富的内部对象供设计人员使用。

- **Java是面向对象的：**Java是一种真正的面向对象的语言，即使是开发简单的程序，也必须设计对象。Java支持面向对象语言中除多继承性外的所有特性，作为面向对象的程序设计语言，Java可以创建强壮可靠的程序。

3. 关于变量

在变量声明与检查方面，Java和JavaScript也不相同。

- **JavaScript中变量声明，采用其弱类型：**即变量在使用前不需做声明，而是解释器在运行时检查其数据类型。JavaScript在脚本执行的过程中自动转换数据类型。这就消除了脚本中显式地声明每一个数据类型的负担，转而把重点放在网页的相关方面。
- **Java采用强类型变量检查：**即所有变量在编译之前必须做声明。对于强类型语言，在程序的编译、装载和执行过程中都可能产生类型转换错误。也就是说，在Java中不恰当的类型转换会引发错误。

4. 编译方式

Java和JavaScript采用不同的编译方式。

- **JavaScript采用动态联编：**即JavaScript的对象引用在运行时进行检查，如不经编译则就无法实现对象引用的检查。
- **Java采用静态联编：**即Java的对象引用必须在编译时进行，以使编译器能够实现强类型检查。

5. 交互方式

在与用户、浏览器及网络其他部分交互方面，Java和JavaScript存在很大不同。

- **Java Applet**只是在浏览器窗口的有限区域内能够和用户交互，不能访问浏览器窗口的其他区域的元素，也不能对这些元素的事件做出响应。
- **JavaScript**允许用户访问网页的任何组件，它允许用户控制整个网页的显示，并对发生在网页任何组件上的事件做出响应。JavaScript的这种特性允许用户将HTML、Java、插件和其他HTML页实体整合到一个完整的Web应用程序中。

6. 代码格式

书写Java和JavaScript的代码格式也是不同的。

- **JavaScript**的代码是一种文本字符格式，可以直接嵌入HTML文档中，并且可动态装载。编写HTML文档就像编辑文本文件一样方便。
- **Java**是一种与HTML无关的格式，必须通过像HTML中引用外部媒体的方式进行装载，其代码以字节代码的形式保存在独立的文档中。

7. 嵌入方式

Java和JavaScript都可以嵌入到HTML文档中，但二者的嵌入方式是不同的。

- 在HTML文档中，JavaScript使用<script>...</script>来标识。在开始标签中，程序通过language属性值来识别脚本类型，即使用的是何种脚本语言。Script标签的典型应用如下：

```
<script language="javascript">
    Javascript语句段
</script>
```

- 在HTML文档中，Java使用<applet>...</applet>来标识，并用code指明字节码所在的文件，width和height指明applet所占的大小。Java Applet小程序的典型应用如下：

```
<applet code="Java类文件名.class" width=宽度 height=高度>
</applet>
```

下面我们以显示“Hello World!”为例，用JavaScript和Java两种不同语言来实现，体验一下二者的不同。

例1-1 用JavaScript在浏览器中显示“Hello World!”。

用嵌入HTML文件中JavaScript实现该功能的程序helloworld.html如下：

```
<HTML>
<HEAD>
  <TITLE>JavaScript程序</TITLE>
  <SCRIPT Language="JavaScript">
    document.write("Hello World!")
  </SCRIPT>
</HEAD>
<BODY>
</BODY>
</HTML>
```

运行时，只需直接在浏览器中执行该网页文件，即被解释执行，运行效果如图1-1所示。

例1-2 用Java Applet小程序实现在浏览器中显示“Hello World!”。

HelloWorldApplet.java文件的内容如下：

```
import java.awt.*;
import java.applet.*;
public class HelloWorldApplet extends Applet
{
    public void paint(Graphics g)
    {
        g.drawString("Hello World!",20,20);
    }
}
```

将Java Applet程序嵌入HTML文件的程序helloworldapplet.html如下：

```
<HTML>
<HEAD>
  <TITLE>Java Applet小程序</TITLE>
</HEAD>
<BODY>
  <applet code="HelloWorldApplet.class" width=200 height=40>
  </applet>
</BODY>
</HTML>
```

运行Java Applet小程序时要先将HelloWorldApplet.java通过Java编译器JDK编译,生成HelloWorldApplet.class文件,然后再在浏览器中运行helloworldapplet.html文件。运行结果如图1-2所示。



图1-1 helloworld.html运行效果



图1-2 helloworldapplet.html运行效果图

可见,JavaScript与Java语言从代码的编写、执行、编译等许多方面都不相同,为方便大家准确地区分和记忆,将二者对比,如表1-1所示。

表1-1 JavaScript同Java的比较

语言	JavaScript	Java
开发公司	Netscape	SUN
前身	LiveScript	Oak
执行方式	解释方式	编译方式
对象	基于对象	面象对象
变量	弱变量	强变量
编译方式	动态联编	静态联编
交互方式	允许访问网页的任何组件	只能在有限区域内交互
代码格式	可直接嵌入HTML中	需在HTML中像引用外部媒体的方式进行装载或独立保存
嵌入方式	<script>...</script>	<applet> ... </applet>

1.1.3 JavaScript的主要特点

JavaScript是一种基于对象(Object)和事件驱动(Event Driven)并具有安全性能的本语言,是专门用来开发Internet客户端和服务端的应用程序,它可以很容易地嵌入到HTML文件中,并通过浏览器中内置的JavaScript解释器来执行。使用JavaScript,浏览器可以回应使用者的需求事件而不用通过网络来回传资料。这样,使用者的资料就可以直接被客户端应用程序所处理。另外它还能方便地同网络客户实现交互,以及使网页变得生动等。JavaScript的出现弥补了HTML语言的缺陷,它是Java与HTML折中的选择,其主要特点如下:

1. JavaScript是一种脚本语言

脚本(Script)是一种能够完成某些特殊功能的指令序列(小程序段)。这些指令序列不

像一般程序那样需要被编译才能执行，而是在程序运行过程中被逐行地解释。这就意味着脚本编写起来比程序容易一些，通常也快一些，这是由于程序是使用更为结构化和经过编译的语言来编写的。在脚本中所使用的命令与语句集称为脚本语言。

JavaScript是一种脚本语言，它的标识符形式上与C、C++、Java等语言十分类似，但它采用小程序段的方式实现编程。像其他脚本语言一样，JavaScript也是一种解释型语言，不需要先编译，而是在程序运行过程中被逐行地解释执行。从而相对于C、Java等语言来说比较容易掌握，而且它与HTML标识结合在一起，方便了用户的使用。

2. JavaScript是一种基于对象的语言

JavaScript是一种基于对象的语言，这意味着它能运用已经创建的对象，但不能派生新的对象，也就是没有面向对象程序设计语言（如Java就是一种面向对象的程序设计语言）所具有的继承、多态等特点，这使得JavaScript更容易学习。JavaScript主要用来访问预定义对象[如窗体（Windows）、框架（Frames）、文档（Documents）、表单（Forms）以及表单元素（Form elements）等]，操纵它们的属性、方法和关联事件。

例如，在JavaScript中，用户不必创建“日期”这个对象，因为该语言已有这一对象，用户可以直接使用。JavaScript本身提供了非常丰富的内部对象供设计人员使用，因而在一般情况下，用户使用这些内部对象就可以满足绝大部分的需要。

3. JavaScript是一种事件驱动的语言

JavaScript可以直接对用户或客户的输入做出响应，无须经过Web服务程序。它对用户的响应，是采用以事件驱动的方式进行的。所谓事件驱动，是通过用户在客户端的某些事件来驱动程序。其中，事件就是指用户执行了某种操作所产生的动作，比如单击鼠标、拖动窗口、选择菜单等都可以视为事件。事件发生后，可能会引起相应的事件响应。JavaScript是一种基于事件驱动的语言，当事件发生时，它可对之做出响应，具体如何响应某个事件取决于相应的事件响应处理程序。

4. JavaScript是一种简单实用的语言

JavaScript是一种脚本编写语言，它提供了一个简易的开发过程。首先，它的基本结构形式与C、C++、Java语言十分类似，因而学习过Java、C或者具有其他编程基础的使用者可以很容易的掌握这门语言。其次，JavaScript的变量类型是采用弱类型，并未使用严格的数据类型，这无疑给初学者带来了便利。第三，JavaScript最杰出之处就在于可以用很小的程序做大量的事，如创建交互式网页、控制浏览器的行为等。用户无须有高性能的电脑，也无需安装复杂的软件，仅需一个字处理软件及一个浏览器，就可以使用JavaScript进行程序设计了。

5. JavaScript是一种安全性的语言

JavaScript是一种安全性的语言，它不允许用户轻易的访问本地硬盘，也不能将数据存入到服务器上，更不允许对网络文档内容进行修改和删除，只能通过浏览器实现信息浏览或动态交互。虽然这在某种意义上削弱了JavaScript的功能，但可以有效地防止恶意程序对系统的破坏，保证了系统的安全。

6. JavaScript是一种跨平台的语言

对于一般的计算机程序，它们的运行与平台有关。除非使用一个仿真器来模拟Windows

环境，否则不可能在Macintosh上运行一个Windows应用程序。JavaScript则不依赖于具体的计算机平台（虽然有一些限制），它只依赖于解释它的浏览器，只要能运行浏览器的计算机，并支持JavaScript的浏览器就可以正确执行。例如，不论你使用Macintosh还是Windows，或是UNIX版本的Netscape Navigator，JavaScript都可以正常运行。

另外，JavaScript还能大大减少服务器的开销，节省CGI（Common Gateway Interface）的交互时间。这是由于JavaScript是一种基于客户端浏览器的语言，用户在浏览中填表、验证的交互过程只是通过浏览器对调入HTML文档中的JavaScript源代码在客户端进行解释执行来完成的，而不用频繁地访问服务器。即使是必须调用CGI的部分，浏览器也只将用户输入验证后的信息提交给远程的服务器，从而大大减少了服务器的开销。

正是以上的这些JavaScript的特点，使得它在Web编程领域中广泛地普及和运用，具有广阔的发展前景。

1.1.4 JavaScript的版本

1. JavaScript的版本

JavaScript语言自从在Netscape 2.0中第一次发布以来，一直在不断进步。下面我们就来简单了解一下JavaScript版本的发展情况。

JavaScript语言的最初版本称做JavaScript 1.0，它是在1995年12月Navigator 2.0正式发布之前，由Netscape和Sun公司联合发布的。这个最初的版本有很多缺陷，现已经被淘汰了。

1996年，Netscape将它的JavaScript语言提交欧洲计算机制造商会（ECMA）进行标准化。ECMA制订了ECMAScript标准，该标准采用了JavaScript的核心语法，但没有定义DOM level 0的所有方面。随着Netscape 3的发行，Netscape发布了JavaScript 1.1，这个版本比JavaScript 1.0要强壮得多，而且还引了一些重要的新特性，如提供了对数组的强化支持，提供了改变图像位置的功能，这种显示技巧被广泛流行。

1997年，随着采用DHTML技术的Netscape 4的发布，产生了JavaScript的新版本JavaScript 1.2。这个新版本含有大量的新特性：如对正则表达式、switch语句和delete运算符的支持等。另外，通过将专用的HTML标记与新的W3C CSS 1建议结合在一起，JavaScript 1.2还为程序员提供了改变文档组件CSS样式的手段。这种分层技术得到了广泛认可，从而诞生了Layers DOM。

1998年，随着Netscape新版本浏览器的发行，发布了新的JavaScript 1.3。这个新版本引入了许多新特性：如对异常处理器（一种高级的错误处理器，在C和Java这样的语言中运用了这种复杂的特性）的支持等。

JavaScript 1.4是只用于Netscape服务器端的JavaScript。

目前，最常用的Netscape 6浏览器支持的JavaScript的版本是JavaScript 1.5。下面的表1-2列举了JavaScript的各个版本及其支持的浏览器。

2. JScript的版本

Microsoft在Netscape公司发布的JavaScript的基础上，也开发了自己的和JavaScript相类似的脚本语言叫做JScript。JScript与JavaScript在基本功能和语法上是相同的，同时还结合Internet浏览器，增加了一些新的功能。