

ASP

网 络 编 程 技 术

肖金秀 主编
冯沃辉 黄元龙 田建辉 何钟颖 参编



清华大学出版社
<http://www.tup.tsinghua.edu.cn>



ASP 网络编程技术

肖金秀 主编

冯沃辉 黄元龙

田建辉 何钟颖 参编

清华大学出版社

(京)新登字 158 号

内 容 简 介

ASP (Active Server Pages) 是 Microsoft 于 1996 年推出的 Web 应用程序开发技术。ASP 的主要特性是能够把 HTML 文件、脚本、基于 COM 标准的组件有机地组合在一起, 形成一个能够在服务器上运行的动态 WEB 应用程序, 并能将按用户要求制作的标准 HTML 页面送给客户端浏览器。本书除了讲述 ASP 的使用知识、编程技巧外, 还专门介绍了几个实用的制作实例, 让读者可以模仿制作某些网上的功能, 从而尽快地将所学到的知识用到实际工作中。

本书内容全面, 包含 ASP 编程技术的大部分关键知识; 举例典型实用, 综合了众多 ASP 程序的实际开发经验。本书适合广大网络设计和开发人员, 以及对 ASP 感兴趣的编程爱好者阅读。

版权所有, 翻印必究。

本书封面贴有清华大学出版社激光防伪标签, 无标签者不得销售。

书 名: ASP 网络编程技术

作 者: 肖金秀 主编 冯沃辉 黄元龙 田建辉 何钟颖 参编

出 版 者: 清华大学出版社(北京清华大学学研大厦, 邮编 100084)

<http://www.tup.tsinghua.edu.cn>

责任编辑: 吴宏伟

印 刷 者: 清华大学印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印 张: 34 字 数: 780 千字

版 次: 2001 年 10 月第 1 版 2001 年 10 月第 1 次印刷

书 号: ISBN 7-302-04794-4/TP·2844

印 数: 0001~5000

定 价: 44.00 元

前言

目前, ASP (Active Server Pages) 技术已风靡全球, 在 Internet 上几乎处处都能看到它的身影。应用 ASP 技术, 不需要进行复杂的编程, 就可以开发出专业的动态 WEB 网站。ASP 技术的出现和发展给曾经以静态内容为主的 WEB 带来了全新的动态效果, 使其具有更加灵活和方便的交互特性, 在 Internet 中实现信息的传递和检索越来越容易。正因为如此, ASP 迅速被广大网络设计和开发人员所接受, 成为他们在 Windows 环境下首选的网站开发和编程技术。

由于 ASP 使用基于开放设计环境的 ActiveX 技术, 用户可以将自己定义和制作的组件加入其中, 使用自己的动态网页具有几乎无限的扩充能力。它还利用 ADO (Active Data Object, 微软的一种新数据访问模型) 方便地访问数据库, 能很好地对数据库进行处理。有了 ASP, 就不必担心客户的浏览器是否能读懂动态网页创作所编写的代码, 因为所有的程序都将在服务器端执行, 包括所有嵌在普通 HTML 中的脚本程序。当程序执行完毕后, 服务器端将执行的结果返回给客户浏览器, 这样也就减轻了客户端浏览器的负担, 大大提高了交互的速度。

本书共分为 6 章, 具体如下:

第 1 章主要讲述 ASP 的基础知识。具体讲述了 ASP 的脚本语言 VBScript 和 JavaScript, 重点讲述了 ASP 中的对象, 以及指令 SSI 与 @ 的用法等。

第 2 章主要讲述了 ASP 的组件。具体介绍了在进行 ASP 编程时所用的一些常用组件。

第 3 章主要讲述的 ADO 的使用方法。具体介绍了 ADO 的使用方法, 并围绕 ADO 的三大对象进行讲述。

第 4 章主要讲述了 ADO 与 SQL 语言。先简介 SQL 语法, 然后通过利用 ADO 特性, 具体结合 SQL 语句, 对数据库进行操作。

第 5 章主要讲述 ASP 如何访问数据库。介绍了多种利用 ASP 访问数据库的方法和方式。

第 6 章为综合实例。其中, 实例包括邮件发送、论坛、网上调查、聊天室、网上调试 SQL 以及一个交友网站。这些实例都是作者从事 ASP 开发的工作成果, 具有很强的实用性。每一个实例都进行详尽的分析注解。

最后的附录列举了 VBScript 常用内部函数、VB 常用函数和作者自编的程序库。作者自编的程序库是作者从业以来自编的“独家秘技”, 这些短程序是使用频率极高的语言, 有较高的使用价值。

本书的讲解全面细致, 举例典型实用, 使用读者能够轻松走入 ASP 的编程天地, 继而

熟练地使用 ASP 进行开发。同时，本书为专业的 ASP 开发人员提供了详细的参考，有利于进一步提高编程水平，掌握更加科学的网站设计思路。

由于时间仓促、水平有限，缺点错误在所难免，恳请读者批评指正。

编 者

2001 年 5 月

目 录

第 1 章 ASP 应用基础.....	1
第 1 节 ASP 基本用法	1
1.1.1 基本脚本语言	1
1.1.2 ASP 变量	4
1.1.3 ASP 的运算符与表达式	7
1.1.4 VBScript 的程序流程控制	11
1.1.5 子过程	20
1.1.6 函数	20
1.1.7 JScript 的程序流程控制	21
1.1.8 JavaScript 函数	25
1.1.9 事件驱动及事件处理	26
第 2 节 Server-Side Include 指令	28
1.2.1 SSI 配置指令	28
1.2.2 SSI 插入环境变量指令	30
1.2.3 SSI 执行指令	32
1.2.4 SSI 指定文件路径指令	33
1.2.5 SSI 指定文件大小指令	34
1.2.6 SSI 包含指令	35
第 3 节 ASP 对象	37
1.3.1 Request 对象	38
1.3.2 Response 对象	51
1.3.3 Application 对象	62
1.3.4 Server 对象	65
1.3.5 Session 对象	69
1.3.6ObjectContext 对象	73
第 4 节 @指令的使用	75
第 5 节 Global 文件使用	77
1.5.1 Application 事件	78
1.5.2 Session 事件	79
1.5.3 OBJECT 声明	82

1.5.4	TypeLibrary 声明	83
第 2 章	ASP 组件	86
第 1 节	Ad Rotator 组件	86
2.1.1	Ad Rotator 组件概述	86
2.1.2	RotatorSchedule 文件	87
2.1.3	重定向文件	88
2.1.4	Ad Rotator 组件的属性	88
2.1.5	Ad Rotator 组件的方法	90
第 2 节	Browser Capabilities 组件	90
第 3 节	Content Linking 组件	92
2.3.1	Content Linking 组件的方法	93
2.3.2	Content Linking 组件使用示例	95
第 4 节	File Access 组件	96
2.4.1	File Access 组件简介	96
2.4.2	FileSystemObject 对象写文件功能	97
2.4.3	FileSystemObject 对象文件读取功能	98
2.4.4	FileSystemObject 对象权限许可设置功能	99
2.4.5	使用 FileSystemObject 对象进行搜索	100
2.4.6	用 FileSystemObject 对象进行内容管理	101
2.4.7	FileS FileSystemObject 对象方法	103
2.4.8	FileSystemObject 对象的不足	122
第 5 节	TextStream 组件	123
2.5.1	TextStream 对象	123
2.5.2	TextStream 对象的方法	124
2.5.3	TextStream 对象的属性	128
第 6 节	性能组件	131
2.6.1	浏览器组件	131
2.6.2	文件超链接组件	137
第 3 章	ADO 简介	145
第 1 节	使用 ADO 对象	145
3.1.1	ADO 对象模型	145
3.1.2	配置 DSN 使用 ADO 对象	147
3.1.3	打开和关闭数据库连接	151
3.1.4	使用企业管理器创建数据表	152
3.1.5	使用 SQL SCRIPT 脚本创建数据表	153
第 2 节	Connection 对象	155

3.2.1	Connection 对象的主要属性.....	156
3.2.2	Connection 的主要方法	157
3.2.3	Connection 对象的集合	158
3.2.4	记录集游标和锁定类型.....	158
3.2.5	Filed 对象与字段	159
3.2.6	使用 RECORDSET 对象注意事项.....	160
3.2.7	范例	160
第 3 节	Error 对象	171
第 4 节	Command 对象	172
3.4.1	建立 Command 对象	172
3.4.2	Command 对象属性	172
3.4.3	Command 对象方法	175
3.4.4	Command 对象的 Parameters 数据集合	178
第 5 节	RecordSet 对象	188
3.5.1	游标	188
3.5.2	Recordset 对象支持的更新	189
3.5.3	RecordSet 对象的方法、属性与集合	190
3.5.4	实现分页.....	190
第 4 章	ADO 与 SQL.....	194
第 1 节	SQL 基本使用方法	194
4.1.1	INSERT 语句	195
4.1.2	SELECT 语句	197
4.1.3	UPDATE 语句	207
4.1.4	DELETE 语句.....	208
第 2 节	ADO 操作 SQL 语句的几种语法	208
4.2.1	ADO 中建立修改删除表	208
4.2.2	ADO 中新增修改删除数据库记录	212
4.2.3	ADO 使用 Insert 语法	214
4.2.4	ADO 使用 SELECT 语法.....	218
4.2.5	ADO 使用 Update 语法.....	236
4.2.6	ADO 使用 Delete 语法.....	240
第 5 章	ASP 访问数据库	242
第 1 节	访问数据库.....	242
5.1.1	创建 ODBC DSN 文件.....	242
5.1.2	ADO 通过 ODBC 存取各种数据库	248
5.1.3	连接数据库.....	253

5.1.4 管理数据库连接.....	259
第 2 节 访问数据库的三种方式.....	261
5.2.1 Internet 数据库接口 IDC.....	262
5.2.2 ActiveX 数据对象 ADO.....	263
5.2.3 远程数据服务 RDS.....	266
第 3 节 ASP 与 ADO 在数据库查询中的应用.....	268
5.3.1 站点数据库访问的几种解决方案.....	269
5.3.2 在网上数据查询中的应用实例.....	270
第 4 节 不用 DSN 访问数据库.....	271
第 5 节 ASP 应用中数据库记录的选取与过滤.....	275
5.5.1 构造过滤条件.....	275
5.5.2 提交表单.....	276
5.5.3 显示结果.....	277
第 6 节 在 ASP 中用集成批操作数据库.....	278
5.6.1 HTML 的集合属性.....	279
5.6.2 HTML 的集合属性的应用.....	280
第 6 章 ASP 综合实例.....	284
第 1 节 用 ASP 来发电子邮件专题.....	284
6.1.1 用 CDO 的 NTS 发送邮件.....	284
6.1.2 用 JMAIL 发送邮件.....	290
第 2 节 论坛.....	300
6.2.1 论坛简介.....	300
6.2.2 论坛的设计.....	302
6.2.3 论坛各程序详解.....	307
第 3 节 网上调查.....	338
6.3.1 网上调查简介.....	338
6.3.2 设计网上调查.....	339
6.3.3 网上调查的数据库结构.....	340
6.3.4 网上调查各程序的详解.....	342
第 4 节 聊天室.....	353
6.4.1 聊天室简介.....	353
6.4.2 设计聊天室.....	354
6.4.3 聊天室各程序的详解.....	358
第 5 节 ASP 网上调试 SQL.....	377
6.5.1 网上调试 SQL 命令简介.....	377
6.5.2 dbase.asp.....	377
6.5.3 Run_sql.asp.....	378

第 6 节 交友网站.....	381
6.6.1 前言	381
6.6.2 网站流程图.....	382
6.6.3 ASP 页面功能例表	383
6.6.4 数据库结构.....	383
6.6.5 GLOBAL.ASA	386
6.6.6 ASP 功能模块	387
附录 A VBSCRIPT 常用内部函数列表.....	487
附录 B VB 常用函数.....	490
附录 C 作者自编程序库	509

第 1 章 ASP 应用基础

第 1 节 ASP 基本用法

1.1.1 基本脚本语言

脚本语言是介于 HTML 和 Java、C++、Visual Basic 等编程语言之间的语言。HTML 通常用于格式化文本和链接网页。编程语言通常用于向计算机发送一系列复杂指令。脚本语言也可用来向计算机发送指令，但它们的语法和规则与可编译的编程语言相比较，没有那么严格和复杂。脚本语言主要用于格式化文本和使用以编程语言编写的已编译好的组件。

1.1.1.1 使用脚本语言

Active Server Pages 使 Web 开发商用多种脚本语言编写完整的过程成为可能。事实上，在单个 .asp 文件内部就可以使用多种脚本语言。另外，由于脚本是在服务器端读取和处理的，所以请求 .asp 文件的客户端浏览器不需要支持脚本。

可以使用任意一种，其脚本引擎已安装在 Web 服务器上的脚本语言。脚本引擎是处理用某种语言书写的命令的程序。Active Server Pages 带有两个脚本引擎：Microsoft Visual Basic Scripting Edition (VBScript) 和 Microsoft JScript。您也可以安装和使用其他的脚本语言引擎，例如 REXX 和 Perl。

如果您已经是 Visual Basic 程序员，就可以马上使用 VBScript，它是 Visual Basic 的子集。如果您是 Java、C 或 C++ 程序员，您会发现 JScript 语法对您来说是熟悉的，虽然 JScript 与 Java 或 C 并没有联系。

如果您熟悉其他脚本语言，例如 REXX 或 Perl，您可获取和安装相应的脚本引擎，这样就可以使用您熟悉的脚本语言了。Active Server Pages 是 ActiveX 脚本的宿主。要使用一种语言，必须安装脚本引擎，该引擎必须遵循 ActiveX 脚本标准并作为一个 COM（组件对象模型）对象驻留在 Web 服务器上。

1.1.1.2 设置主脚本语言

ASP 主脚本语言是用来处理在分界符 `<%` 和 `%>` 内部的命令的语言。默认时，主脚

本语言是 VBScript。您可以将任何一种具有脚本引擎的脚本语言作为主脚本语言。可以逐页设置主脚本语言，也可以在 ASP 应用程序中设置所有页的主脚本语言。

1. 为某页设置语言

要设置单个页的主脚本语言，可将 `<%@ LANGUAGE %>` 指令添加到 .asp 文件的开始位置。

语法：

```
<%@ LANGUAGE=ScriptingLanguage %>
```

参数：

ScriptingLanguage 就是所设置的页的主脚本语言。如果对某页进行了设置，那么该页将忽略在应用程序中对所有页的全局设置。

注意：

若要将不支持 Object.Method 语法的语言作为主脚本语言使用，必须先创建 LanguageEngines 注册表键。

2. 为应用程序设置语言

在应用程序中，如要为所有页设置主脚本语言，可在 Internet 服务管理器的“App 选项”选项卡上设置“默认的 ASP 语言”属性。

1.1.1.3 在服务器上使用 VBScript 和 JScript

在服务端通过 ASP 使用 VBScript 时，两个 VBScript 特征将失效。由于 Active Server Pages 脚本是在服务器端执行的，表示用户接口元素的 VBScript 语句 InputBox 和 MsgBox 将不被支持。另外，在服务器端的脚本中，请不要使用 VBScript 函数 CreateObject 和 GetObject，而要使用 Server.CreateObject，这样 ASP 就可以跟踪对象实例了。用 CreateObject 或 GetObject 创建的对象不能访问 ASP 内建对象，也不能参与事务处理。详细信息，请参阅 IIS Admin 对象和创建 Java 类的实例。

关于所有 VBScript 和 JScript 的操作符、函数、语句、对象、属性及方法的列表及详细说明，请参阅 VBScript 语言参考和 JScript 语言参考。

1. 包含注释

由于 ASP 脚本是在服务器端处理的，所以客户端的浏览器可以不支持脚本语言；也不必通过包含 HTML 注释标记来隐藏脚本，但客户端的脚本则通常需要这样处理：在内容送到浏览器之前，所有 ASP 命令都已被处理好了。您可用 HTML 注释将注释加进 HTML 页。注释将返回给浏览器，若用户浏览 HTML 源文件，就可以看见。

➤ VBScript 注释

VBScript 支持撇号型注释。与 HTML 注释不同，当处理脚本时，它们将被删除，而不

是被送到浏览器。

```
<%  
'这一行和下面的两行都是注释.  
'PrintTable是个函数.  
'元素是一个数组.  
Call PrintTable(myarray())  
%>
```

输出表达式不能包括注释。例如，下面的第一行将正常运行，而第二行则不能，因为它是以<%= 开头的。

```
<% i = i + 1 '增量是1. %>  
<%= name %>
```

➤ JScript注释

JScript 支持 // 注释符。每一个注释行中都必须使用该注释符。

```
<% Call PrintDate %>  
<SCRIPT LANGUAGE=JScript RUNAT=Server>  
function PrintDate()  
{  
var x  
x = new Date()  
// 这一行发送当前的日期到浏览器,  
// 转换成--一个字符串.  
Response.Write(x.toString())  
}  
}  
</SCRIPT>
```

2. 大小写敏感性

VBScript 不区分大小写。例如，您可以用 Request 或 request 来引用 ASP Request 对象。不区分大小写的后果是您不能用大小写来区分变量名。例如，您不能创建两个名为 Color 和 color 的单独变量。

JScript 区分大小写。要在脚本中使用 JScript 关键字，就必须按参考页中所示的大小写来书写。例如，用 date 来代替 Date 将导致错误。在 JScript 中，对象名必须大写；方法名

和属性名可大写也可小写。本文所示的 ASP 内建对象的情况将在 JScript 命令中正常工作。

1.1.2 ASP 变量

1.1.2.1 变量

变量是计算机内存中已命名的存储位置，其中包含了数字或字符串等数据。变量包含的信息被称为变量的值。变量使用户便于理解脚本操作的名称，为用户提供了一种存储、检索和操作数据的途径。

1. 声明和命名变量

命名和声明变量应遵循脚本语言的规则及指导。即使在使用变量前不需要去声明变量，也应养成在编程时声明变量的良好习惯，因为这样有助于防止错误发生。声明一个变量意味着告诉脚本引擎，有一个特定名称的变量，这样就可以在脚本中引用该变量。

➤ VBScript

VBScript 并不需要声明变量，但在使用所有变量前声明它们是一种好的脚本书写习惯。要在 VBScript 中声明变量，请用 Dim、Public 或 Private 语句。例如：

```
<% Dim UserName %>
```

可以在需要显式声明变量的 .asp 文件中用 VBScript Option Explicit 语句。Option Explicit 必须在任何一个 ASP 指令之后及任何一个 HTML 文本或脚本命令之前出现。该语句仅影响用 VBScript 书写的 ASP 命令，而不会影响 JScript 命令。

关于这些命令的详细信息，请参阅 VBScript 语言参考。

➤ JScript

Microsoft JScript 仅当变量为本地过程变量时才需声明，但在使用所有变量前声明它们是一种好的脚本书写习惯。要声明一个变量，请使用 var 语句。例如：

```
<% var UserName; %>
```

有关 var 语句的详细信息，请参阅 JScript 语言参考。

2. 变量作用域

变量的作用域即生命期，决定哪些脚本命令可访问变量。在过程内部声明的变量具有局部作用域。每执行一次过程，变量就被创建然后消亡。而过程外部的任何命令都不能访问它。在过程外部声明的变量具有全局作用域，其值能被 ASP 页上的任何脚本命令访问和修改。

声明变量时，局部变量和全局变量可以有相同的名称。而改变其中一个变量的值并不会改变另一个变量的值。如果没有声明变量，则可能会不小心改变一个全局变量的值。

例如，以下脚本命令返回 1，虽然有两个名为 Y 的变量：

```
<%
```

```
Dim Y
Y = 1
Call SetLocalVariable
Response.Write Y
Sub SetLocalVariable
Dim Y
Y = 2
End Sub
%>
```

由于变量没有显式声明，以下的脚本命令将返回 2。当过程调用将 Y 设置为 2 时，脚本引擎认为该过程是要修改全局变量：

```
<%
Y = 1
Call SetLocalVariable
Response.Write Y
Sub SetLocalVariable
Y = 2
End Sub
%>
```

养成显式声明所有变量的编程习惯可以避免许多问题。尤其在当用 `#include` 语句将文件包含进 ASP 主页时，就显得更为重要。一个独立文件中被包含的脚本是被当作整个包含它的文件的一部分来处理的。用不同的名称来命名主脚本和被包含脚本中用到的变量，这容易被遗忘，除非声明变量。

3. 为变量赋予会话或应用程序作用域

局部变量仅在单个 ASP 页中可用，要使它在一个 ASP 页之外可用，就必须为变量赋予会话或应用程序作用域。会话作用域变量对一个用户所请求的 ASP 应用程序中的所有页都是可用的。应用程序作用域变量也如此。对单个用户来说，会话变量是存储信息的最佳途径，例如，首选项、用户名或用户标识。对于一个特殊应用程序的所有用户，应用程序作用域是存储信息的最佳途径，例如，应用程序特定的问候语或应用程序所需的初始值。

ASP 提供两个内建对象来让您存储变量：Session 对象和 Application 对象。

➤ 会话作用域

要为变量赋予会话作用域，将该变量存储到 Session 对象中即可，方法是为该对象的已命名条目指派一个值。例如，以下命令将两个新的变量存储到 Session 对象中。

```
<%
```

```
Session("FirstName") = "crazy"  
Session("LastName") = "drink"  
%>
```

要从 Session 对象中检索信息, 可以用输出指令 (<%=) 或 Response.Write 访问已命名的条目。以下示例用输出指令显示 Session("FirstName") 的当前值:

```
Welcome <%= Session("FirstName") %>
```

可在 Session 对象中存储用户首选项, 然后就可以访问这些首选项, 从而决定将哪一页返回给用户。例如, 您可以允许用户在应用程序的第一页指定纯文本内容, 然后将这一选择应用到用户在该应用程序中访问的所有后续页。

```
<% If Session("ScreenResolution") = "Low" Then %>  
This is the text version of the page.  
<% Else %>  
This is the multimedia version of the page.  
<% End If %>
```

➤ 应用程序作用域

要为变量赋予应用程序作用域, 将该变量存储到 Application 对象中即可, 方法是为该对象的已命名条目指派一个值。例如, 以下命令将一个应用程序特定的问候语存储到 Application 对象中:

```
<% Application("Greeting") = "Welcome to Exploration Air" %>
```

要从 Application 对象中检索信息, 可以使用 ASP 输出指令(<%=)或 Response.Write 从该应用程序的任意后续页中访问已命名条目。以下示例用输出指令显示 Application("Greeting") 的值:

```
<%= Application("Greeting") %>
```

1.1.2.2 常量

常量是用来代替一个数或字符串的名称。一些随 ASP 提供的基本组件, 例如 ActiveX Data Objects (ADO), 定义了在本脚本中可以使用的常量。组件可在一个组件类型库中声明常量, 组件类型库是一个包含 ActiveX 组件所支持的对象和类型的信息的文件。一旦在 Global.asa 文件中声明了一个类型库, 就可以在该应用程序的任意页中使用已定义的常量。

可在 Global.asa 中用<METADATA>标记为应用程序声明类型库。例如, 要声明 ADO 类型库, 可用以下语句:

```
<!--METADATA TYPE="typelib"  
FILE="c:\program files\common files\system\ado\msado15.dll"
```

-->

这样，就可以在与 Global.asa 文件相同的应用程序中使用 ADO 常量。在下面的示例中，adOpenKeyset 和 adLockOptimistic 是 ADO 常量：

```
'Create and Open Recordset Object
Set RsCustomerList = Server.CreateObject("ADODB.Recordset")
RsCustomerList.ActiveConnection = OBJdbConnection
RsCustomerList.CursorType = adOpenKeyset
RsCustomerList.LockType = adLockOptimistic
```

有关 <METADATA> 标记的参考信息，请参阅 TypeLibrary 声明。

在以前版本的 ASP 中，一些组件在文件中提供常量定义，每个使用那些组件的 ASP 文件必须包含该文件。用 #include 指令来包含常量定义的法仍被支持，但使用类型库通常更方便并且能使脚本更易于升级。在以后版本的 ASP 中，组件可能将不再提供常量定义文件。

可以定义自己的常量。在 VBScript 中，用 Const 语句；在 JScript 中，用 var 语句。如果在多个 ASP 页中使用常量，可在独立的文件中放置常量，然后将其包含进每一个使用该常量的 ASP 文件中。

1.1.3 ASP 的运算符与表达式

运算是数据的加工。最基本的运算形式常常可以用一些简单的符号来描述，这些符号称为运算符或操作符。被运算的对象，即数据，称为运算量或操作数。操作数可以是常量也可以变量，还可以是函数。例如：A + B = C，A * B，B = A + 3，T + Cint(B)等都是表达式，单个的变量或常量也可以看成是表达式。

VBScript 具有一套完整的操作符，包括：算术运算符、比较运算符、联接运算符和逻辑运算符。下面我们对它们作一一介绍。

1.1.3.1 算术运算符

算术运算符是常用的运算符，用来执行简单的算术运算。VBScript 提供了 9 个算术运算符，表 1-1 按优先级列出了这些算术运算符。

在 9 个运算符中，除取负 (-) 是单目运算符外，其他均为双目运算符（需要两个操作符）。加 (+)、减 (-)、乘 (*)、取负 (-) 等几个运算符的含义与数学中的基本相同，下面介绍其他几个运算符的操作。