

编程实战与技巧系列

4



VB.NET

编程实例与技巧集粹

北京希望电子出版社 总策划
葛小东 邓秋媛 等 编 著



中国科学技术出版社
CHINA SCIENCE AND TECHNOLOGY PRESS



北京希望电子出版社
Beijing Hope Electronic Press
www.bhpe.com.cn

编程实战与技巧系列

4

VB.NET

编程实例与技巧集粹

北京希望电子出版社 总策划
葛小东 邓秋媛 等 编 著

 中国科学技术出版社
CHINA SCIENCE AND TECHNOLOGY PRESS



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

图书在版编目 (CIP) 数据

VB.NET 编程技巧与实例集粹/葛小东等编. —北京:
中国科学技术出版社, 2003.8

(编程实战与技巧系列)

ISBN 7-5046-3574-X

I.V... II.葛... III.BASIC 语言—程序设计

IV.TP312

中国版本图书馆 CIP 数据核字 (2003) 第 056846 号

系 列 名: 编程实战与技巧系列 (4)

书 名: VB.NET 编程实例与技巧集粹

总 策 划: 北京希望电子出版社

文 本 著 者: 葛小东 邓秋媛等

责 任 编 辑: 周凤明 沈葆华

出版、发行者: 中国科学技术出版社 北京希望电子出版社

地 址: 北京市海淀区中关村南大街 16 号 100081

北京市海淀区知春路甲 63 号卫星大厦三层 100080

网址: www.bhp.com.cn E-mail: lwm@bhp.com.cn zwb@bhp.com.cn

电话: 010-62520290, 62521724, 62528991, 62630301, 62524940, 62521921,

62521724 (发行) 010-82675588-318, 62532258, 62562329 (门市)

010-82675588-501, 82675588-201 (编辑部)

经 销: 各地新华书店、软件连锁店

排 版: 希望图书输出中心 杜海燕

印 刷 者: 北京媛明印刷厂

开 本 / 规格: 787 毫米×1092 毫米 1/16 30 印张 725 千字

版 次 / 印 次: 2003 年 8 月第 1 版 2003 年 8 月第 1 次印刷

印 数: 1~5000 册

书 号: ISBN 7-5046-3574-x/TP·185

定 价: 45.00 元

内 容 简 介

本书是 Visual Basic.NET 编程实例与技巧的专题书籍, 书中设计了 200 个经典实例, 旨在让读者全面、快速、有效地掌握 Visual Basic.NET 编程的方法和技巧。

本书分为 11 章, 第 1 章介绍 VB.Net 的基本语法及与 VB 6.0 的比较; 第 2 章介绍 VB.Net 中面向对象的基本知识, 这部分是 VB.Net 的核心所在; 第 3 章简要介绍 VB.Net 对于文件系统的定义及文件操作技术; 第 4 章与第 5 章主要介绍 VB.Net 中功能强大的控件; 第 6 章介绍 VB.Net 提供的绘图方法; 第 7 章主要介绍多媒体制作, 并且通过使用多媒体控件了解如何使用其他的控件; 第 8 章与第 9 章介绍 VB.Net 中的数据访问技术; 第 10 章简要介绍 VB.Net 中涉及的 Web 编程基础, 主要是简介 HTML 文档结构和常标记、脚本代码编程等; 第 11 章深入介绍如何创建 ASP.Net Web 应用程序, 内容包括 HTML 服务器控件、Web 控件、数据绑定等内容。

本书内容丰富、讲解详细、语言流畅, 书中内容密切围绕 Visual Basic.NET 与 Visual Basic 6.0 相比的优越性展开, 每一个实例都有非常丰富的基本知识介绍以及实例思路分析, 侧重基础性和实用性的结合, 本书可作为初中级程序开发人员的自学丛书, 也可以作为社会培训班的教学参考书。

书中实例代码请从网上 (<http://www.b-xr.com>) 下载。

前 言

Visual Basic.NET 是继 Visual Basic 6.0 之后的新一代语言，它是运行于 Windows 平台上的交互式的可视化集成开发环境，也是 Microsoft Visual Studio.Net 的四种编程语言之一。对于长期使用 VB 6.0 编程工具的人来说，值得欣喜的是，VB.NET 是一种真正面向对象的语言，这是一次历史性的飞跃。使用 VB.NET，可以充分利用 .Net 框架中提供的所有内容，完成从最初的控制台应用程序、Windows 应用程序到 Web 应用程序的各种不同的要求。仅凭这一点，就有足够的理由相信 VB.NET 的前途不会再像 VB 6.0 那样受到怀疑。更何况，VB.NET 的好处还远不止这些，VB.NET 从语法、窗体控件、数据访问、安全性等几乎所有方面对 VB 6.0 进行了改造。

但是，也正因为如此，现实情况是，很多学习 VB 6.0 并长期使用这种工具的人都知道，要纠正由于 VB 6.0 编程工具本身缺陷所带来的错误，是非常不容易的。要把长期习惯于 VB 6.0 的那种似乎面向对象又不完全面向对象的感觉引领到全新的面向对象的世界，这必须让大家清楚地看到，VB.NET 确实有着诸多的好处。

基于这样一种想法，笔者开始构思这本书的组成和形式。本书的写作思路是希望读者通过阅读这本书，逐渐忘却 VB 6.0，在学习与实践过程中，慢慢地领会 VB.NET 的核心。对于使用过 VB 6.0 或者 Visual Basic 更早期版本的读者，我相信阅读本书不会有障碍。因为本书从一开始，就试图突出 VB.NET 与 VB 6.0 的不同，同时也介绍了非常丰富的基本知识。对于已经具备一定 VB.NET 基础知识的读者，本书提供丰富的实例可以用于实践，这有助于更深一步的理解。

本书全部以实例的形式展开，每一个实例都会涉及到大量的知识和问题。这些知识和问题被分为 11 章：

第 1 章，主要介绍 VB.NET 的基本语法，与 VB 6.0 相比，在 VB.NET 中对基本语法作了哪些改进。这部分内容对于我们迅速了解 VB.NET 的语法基础是非常重要的。

第 2 章，较为详细地介绍了 VB.NET 中面向对象的基本知识，这部分所介绍的内容将贯穿全书，也是 VB.NET 的核心问题所在。

第 3 章，以较短的篇幅介绍了 VB.NET 对于文件系统的定义。相比较于 VB 6.0 中所使用的函数以及不得不使用的 API 函数，VB.NET 中能进行的文件操作将让人觉得开心。

第 4 章与第 5 章，主要介绍 VB.NET 中所能用到的丰富的功能强大的控件，这些内容使我们的应用程序在“可视化”方面能做的事情更多。

第 6 章，主要介绍 VB.NET 中所提供的绘图方法。曾经有不少人向我表达他们希望借助于 VB 6.0 绘制一些特殊效果的图形图像的愿望，比如希望把对话框变成圆形的，使用 VB.NET，这一切会变得非常的轻松。

第 7 章，主要介绍多媒体制作，并且通过使用多媒体控件了解如何使用其他的控件。因为能够被我们利用的控件如此之多，在这一本书中不可能全部都介绍。因此，读者需要一种方法。通过本章的学习，对如何使用其他的控件就不会再感到陌生了。

第8章与第9章,主要介绍VB.NET中的数据访问技术,与VB 6.0不同的是,在VB.NET中,对数据访问方法与对象进行了极大的改进。第8章主要集中于这些基本知识的理解,第9章则是侧重于对这些数据访问技术的应用。

第10章,简要介绍VB.NET中涉及到的Web编程基础,主要是简介HTML文档结构和常标记、脚本代码编程等。

第11章,深入介绍如何创建Asp.Net Web应用程序,内容包括HTML服务器控件、Web控件、数据绑定等内容。

以上是本书的基本体系结构。

本书由葛小东和邓秋媛共同执笔编写,此外,张维、吴轶秦、李龙、钱少伟、刘荣强、李伟光、朱峰、许大中、魏勇、萧玉等同志在整理材料方面给予了很大的帮助,在此,编者向他们表示深深的谢意。

由于编者能力和水平有限,本书可能会有谬误之处,恳请读者批评指正。

编 者

2003 年 5 月

目 录

第1章 基本语法.....1	实例 032 类的事件.....73
实例 001 变量的声明及简单运算.....2	实例 033 变量与过程的作用范围.....75
实例 002 变量的数学运算.....5	第3章 文件操作.....77
实例 003 数据类型的转换.....7	实例 034 获取驱动器列表.....78
实例 004 查找字符串中的特定字符的数目.....9	实例 035 获取目录属性.....79
实例 005 输入简单表达式并求值.....12	实例 036 对目录的操作.....80
实例 006 查找随机数组中的质数个数.....14	实例 037 获取当前目录的所有文件及 子目录.....82
实例 007 数组排序.....17	实例 038 磁盘文件操作.....83
实例 008 字符串的有序合并.....20	实例 039 路径操作.....85
实例 009 队列游戏.....23	实例 040 获取文件属性.....87
实例 010 格式化字符串.....25	实例 041 文件流读写二进制文件.....88
实例 011 数组的数组.....27	实例 042 读写文本文件.....90
实例 012 对象数组.....30	实例 043 文件定位.....93
实例 013 自定义类型数据.....31	实例 044 文件观察对象.....94
实例 014 动态数组.....34	第4章 窗体设计.....97
实例 015 过程实现变量替换.....35	实例 045 简单的 Windows 应用程序.....98
第2章 类与对象.....37	实例 046 标签移动与色彩变幻.....99
实例 016 类的声明及类的实现.....38	实例 047 文本框编辑.....102
实例 017 类的构造与析构.....40	实例 048 文本查找与替换.....105
实例 018 类的属性.....42	实例 049 文本框字符长度控制.....107
实例 019 类的方法.....45	实例 050 选择字体.....109
实例 020 类的默认属性.....47	实例 051 倒计时器.....112
实例 021 类的继承.....49	实例 052 三元色配色.....116
实例 022 多个类从一个类继承.....51	实例 053 列表框选择字体.....119
实例 023 公有与私有.....54	实例 054 字体样式.....122
实例 024 保护类型的数据.....56	实例 055 利用字体对话框设置字体.....124
实例 025 类的多态性.....58	实例 056 打开保存文件对话框.....126
实例 026 必须覆盖的类成员和不能覆盖 的类成员.....61	实例 057 读取保存文件.....128
实例 027 隐藏基类的属性与方法.....63	实例 058 图片浏览器.....131
实例 028 方法的重载.....66	实例 059 滚动标签.....133
实例 029 类的共享属性.....68	实例 060 定时猜数游戏.....135
实例 030 类的共享方法.....70	实例 061 读书器字体颜色.....137
实例 031 静态变量.....71	实例 062 登录对话框.....139

实例 063	透明窗体	142	实例 099	花纹字符	228
实例 064	页面设置	143	实例 100	屏幕保护	229
实例 065	打印设置	145	实例 101	椭圆形图画	232
实例 066	打印文本文件	146	实例 102	贝塞尔曲线	234
实例 067	打印图片	148	实例 103	过渡色	236
实例 068	菜单界面	150	实例 104	万花规	239
实例 069	使用工具条	152	实例 105	绘制函数曲线	242
实例 070	使用状态栏	154	实例 106	绘制饼图	246
第 5 章 窗体应用		156	实例 107	鼠标画线	249
实例 071	加法器	156	实例 108	鼠标画圆	251
实例 071	加法器	157	实例 109	绘图路径	253
实例 072	简单计算器	158	实例 110	绘图路径的填充模式	256
实例 073	剪贴板操作	163	实例 111	绘图区域	259
实例 074	记事本文件操作	164	实例 112	特殊形状的对话框	262
实例 075	打开保存图片文件	168	实例 113	特殊形状的对话框	263
实例 076	图片浏览	171	实例 114	颜色拾取器	265
实例 077	图片的缩放	173	实例 115	百叶窗效果	267
实例 078	带格式的简单写字板	177	第 7 章 多媒体设计		271
实例 079	RTF 格式文件的保存	180	实例 116	VCD 播放器	272
实例 080	富文本框中的上下标	183	实例 117	播放器设置	274
实例 081	文件夹列表视图	185	实例 118	WAV 播放器	277
实例 082	修改文件访问日期	187	实例 119	CD 播放器	280
实例 083	树状分支目录结构	190	实例 120	播放视频	282
实例 084	F1 键帮助	192	实例 121	播放动画	284
实例 085	图片框自动调整大小	193	实例 122	有背景音乐的登录界面	286
实例 086	控件停靠与分栏	194	实例 123	简便录音机	289
实例 087	错误提示	195	实例 124	动画屏保	291
实例 088	对话框上的超链接	197	第 8 章 数据库访问		294
实例 089	选项卡	199	实例 125	连接 Access 数据库	295
实例 090	消息框	202	实例 126	连接 SQL Server 数据库	297
第 6 章 图形图像		204	实例 127	连接异常	299
实例 091	绘制直线	205	实例 128	读取数据	302
实例 092	选取绘图笔	207	实例 129	列值查询	305
实例 093	绘制圆形时钟	210	实例 130	特定查询	308
实例 094	绘制多边形	214	实例 131	登录对话框	311
实例 095	红色填充的五角星	217	实例 132	程序创建数据对象	314
实例 096	直线样式	220	实例 133	数据表对象	317
实例 097	绘制字符串	222	实例 134	表的创建、删除、重命名	320
实例 098	封闭图形的花纹填充	224	实例 135	修改数据表	323

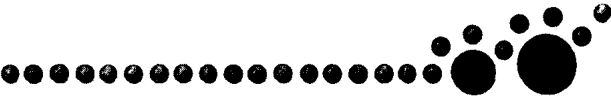
实例 136	修改表中的数据	329	实例 171	页面加载事件	425
实例 137	数据集	332	实例 172	设置页面背景色	426
实例 138	数据适配器	336	实例 173	窗口操作	427
实例 139	数据视图	339	实例 174	历史操作	429
实例 140	带参数命令	343	实例 175	友情链接	430
实例 141	专门访问 SQL Server	345	实例 176	窗口状态栏显示时间	431
实例 142	创建访问 SQL Server 的数据对象	347	第 11 章 Web 应用程序		434
实例 143	带参数的 SqlCommand	350	实例 177	HTMLAncho 服务器控件	435
实例 144	更新数据源	352	实例 178	HTMLButton 服务器控件	436
第 9 章 数据库应用		360	实例 179	利用 HTMLGeneric 服务器控件修改字体	438
实例 145	使用 ADO	361	实例 180	利用 HTMLGeneric 服务器控件修改文档背景色	439
实例 146	使用连接对象操作数据库	364	实例 181	利用 HTMLInputRadioButton 服务器控件修改文档字体	440
实例 147	使用命令对象操作数据库	367	实例 182	利用 HTMLInputCheckBox 服务器控件选购商品	442
实例 148	连接字符串的设置	370	实例 183	利用 HTMLImage 服务器控件显示图片	444
实例 149	记录集对象的游标管理	374	实例 184	利用 HTMLInputText 服务器控件获取输入框的值	445
实例 150	记录集字段属性	377	实例 185	身份验证	447
实例 151	设置记录集的查询条件	379	实例 186	向列表框加入选项	448
实例 152	在记录集中筛选记录	383	实例 187	修改列表框选项	449
实例 153	浏览记录集	385	实例 188	删除列表框选项	451
实例 154	记录集添加、删除或修改记录	388	实例 189	单人留言板	452
实例 155	创建图书信息表	391	实例 190	文件上传	454
实例 156	图书信息录入	394	实例 191	使用 Web 标签控件	456
实例 157	查询书目信息	398	实例 192	使用 Web 日历控件	457
实例 158	登录口令	402	实例 193	使用列表控件修改字体	458
实例 159	图片数据库	404	实例 194	设置字体	459
第 10 章 Web 程序基础		407	实例 195	超链接控件	461
实例 160	静态网页	408	实例 196	设置字体样式	463
实例 161	插入表格	410	实例 197	设置文本框的初始值	465
实例 162	嵌入图片	412	实例 198	图像控件	466
实例 163	列表项	413	实例 199	验证控件	468
实例 164	表单元素	414	实例 200	代码分离	469
实例 165	嵌入脚本	416			
实例 166	处理按钮单击事件	417			
实例 167	计算两个数的和	418			
实例 168	选择计算表达式	420			
实例 169	列表框的选项值	422			
实例 170	焦点设置	423			

第1章 基本语法



Visual Basic.NET 是 Microsoft 公司新推出的编程语言，是真正面向对象的编程语言，因此，在基本语法上，Visual Basic.NET（简称 VB.NET）与 Visual Basic 6.0（简称 VB 6.0）有着较多明显的差异。本章通过各种实例来介绍 VB.NET 的基本语法知识，介绍它与 VB 6.0 的基本语法有哪些不一致的地方。





(续表)

数据类型	CLS 类型	占用字节	值的范围
Decimal	System.Decimal	12	-79228162514264337593950335~ +79228162514264337593950335
String	System.String	10+2*Len	
用户自定义 类型	System.ValueType		

VB.NET 中定义变量的方法与 VB 6.0 基本类似，一般采用如下形式：

Dim 变量 As 数据类型

与 VB 6.0 不同的是，在 VB.NET 中定义变量时，可以同时为变量赋初值，比如：

Public x As Integer = 10

VB.NET 中的变量运算主要包括算术运算、赋值运算、比较运算、逻辑运算、字符连接运算等，各类运算如下：

- 算术运算，包括 +，-，*，/，\，%，^ 等
- 赋值运算，包括 =，+=，-=，*=，/=，\=，^=等
- 比较运算，包括 =，<，>，>=，<=，Is，Like 等
- 逻辑运算，包括 And，Or，Not，Xor，AndAlso，OrElse 等
- 连接运算，&，主要用于连接字符串

如果要将结果显示到屏幕上，可以使用 Console 类的 Write 方法或 WriteLine 方法。

Console 类是 System 名称空间下定义的一个类，用于表示控制台应用程序的输入流与输出流。Write 方法和 WriteLine 方法的使用格式如下所示：

Console.Write 要输出的数据

Console.WriteLine 要输出的数据

Write 和 WriteLine 的区别在于，WriteLine 输出数据后换行，而 Write 方法不换行。

如果要读取用户输入的数据，可以使用 Console 类中定义的 Read 方法和 ReadLine 方法，一般使用格式如下：

存放数据的变量 = Console.Read

存放数据的变量 = Console.ReadLine

与 Write 方法和 WriteLine 方法的区别相类似，Read 方法与 ReadLine 方法分别用于表示读取一个数据和读取一行数据。

在 VB.NET 中，REM 关键字表示该行位于关键字后面的内容是注释，不被执行。REM 关键字可以用一个单引号代替。



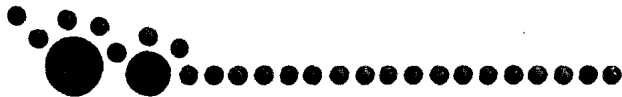
程序设计

新建一个 Visual Basic 项目，选择控制台应用程序类别，项目名称为 VB001，系统自动生成名为 Module1.vb 的文件，并自动在其中添加如下的代码框架：

Module Module1

Sub Main ()



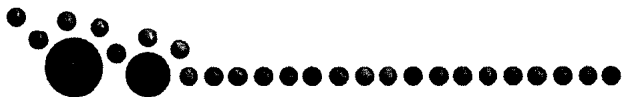


End Sub

End Module

在 Sub Main () 与 End Sub 之间添加如下代码:

```
REM 定义字符串变量, 并赋初值
Dim MyName As String, FirstName As String = "张", SecondName As String = "王"
REM MyName 的值为 FirstName 与 SecondName 拼接值
MyName = FirstName & SecondName
Rem 输出 MyName 变量与 "Is My Name" 常量字符串
Console.WriteLine (MyName & " Is My Name.")
REM 定义日期变量并赋初值, 注意日期常量前后有#号
Dim MyBirthday As Date = #5/30/1980#
Dim NewYearDay As Date = #1/1/2002#, days As Date
REM 定义布尔变量
Dim BeforeBorn As Boolean
BeforeBorn = NewYearDay < MyBirthday
REM 输出 MyBirthday 等变量值
Console.Write (MyBirthday)
Console.Write ("<")
Console.Write (NewYearDay)
Console Write ("ie ")
```



```
Optional ByVal Buttons As MsgBoxStyle = MsgBoxStyle.OKOnly, _  
Optional ByVal Title As Object = Nothing _  
) As MsgBoxResult
```

MsgBox 函数有三个参数,分别用于表示消息框的输出信息、消息框的样式以及消息框的标题,在使用 MsgBox 函数时,后两个参数可以省略。MsgBox 函数的返回值与消息框的样式以及用户的实际操作有关。

在 VB.NET 中定义了一个名为 InputBox 的函数,用于提示并接收用户输入的字符串,InputBox 函数的定义形式如下所示。

```
Public Function InputBox ( _  
    ByVal Prompt As String, _  
    Optional ByVal Title As String = "", _  
    Optional ByVal DefaultResponse As String = "", _  
    Optional ByVal XPos As Integer = -1, _  
    Optional ByVal YPos As Integer = -1 _  
) As String
```

InputBox 函数需要 5 个参数,第 1 个参数为字符串类型的变量,表示输入框的提示信息;第 2 个参数为字符串变量,表示输入框的标题;第 3 个参数是字符串变量,表示输入框的默认输入值;第 4 个参数和第 5 个参数为整型变量,表示输入框显示的坐标位置。在实际使用 InputBox 函数时,后面 4 个参数都可以省略。输入框函数的返回值为用户输入的数据。



程序设计

新建一个 Visual Basic 项目,选择控制台应用程序类别,项目名称为 VB002,系统自动生成名为 Module1.vb 的文件,向 Module1.vb 中添加代码如下。

```
Module Module1
```

```
Sub Main ()
```

```
    Dim x1 As Double, x2 As Double
```

```
    Dim y1 As Double, y2 As Double
```

```
    Console.WriteLine ("输入两个数,计算 x1 的 x2 次方")
```

```
    x1 = InputBox ("输入 x1", "乘方运算底数")
```

```
    ' 使用 InputBox 函数,弹出一个输入框,等待用户输入字符串
```

```
    ' 并将输入的字符串保存到变量中
```

```
    x2 = InputBox ("输入 x2", "乘方运算指数")
```

```
    ' 调用 Math 类的乘方函数,计算 x1 的 x2 次方
```

```
    y1 = Math.Pow (x1, x2)
```

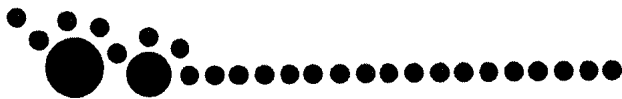
```
    MsgBox (y1, MsgBoxStyle.Information, "乘方运算结果")
```

```
    Console.WriteLine ("输入两个数,计算以 x1 为底, x2 的对数")
```

```
    x1 = InputBox ("输入 x1", "对数运算底数")
```

```
    x2 = InputBox ("输入 x2", "对数运算基数")
```





较小的数据类型向占用内存较大的数据类型转换；收缩是指将占用内存较大的数据类型向占用内存较小的数据类型转换。

VB 6.0 使用函数实现数据类型的转换，这些函数包括 CBool、CByte、CShort、CChar、CDate、CDec、CInt、CLng、CDBl、CSng、CStr 等，它们将函数参数的数据类型转换为函数名所代表的数据类型。VB.NET 兼容了 VB 6.0 利用函数对数据类型进行转换的功能，因此，在 VB.NET 中仍然可以使用这些函数。

在 VB.NET 中，除了使用函数进行数据类型的转换外，还可以使用 CConvert 类实现



表 1-2 System.String 类中定义的主要方法及其功能

String 的常用方法/属性	说明
Clone	对象的复制
Compare	字符串比较
Concat	字符串连接
Copy	把字符串对象复制到另一个对象上
StartWith	测试开始字符串
EndsWith	测试结尾字符串
Equals	比较字符串是否相等
Format	格式化字符串
Insert / Remove	插入/ 删除字符
Replace	替换指定字符
Substring	获取子串
IndexOf	特定字符在字符串中的位置
Length	字符串长度

在 VB.NET 中定义了字符串变量以后, 该字符串变量即具有了上述方法, 比如定义字符串变量 AString 后, 要查看其长度, 可以使用 AString.Length 来获取, 其他方法类似。

VB.NET 兼容 VB 6.0 的基本语法, 因此, VB.NET 允许使用 If 条件语句、For 循环语句、While 循环语句等。



程序设计

新建一个 Visual Basic 项目, 选择控制台应用程序类别, 项目名称为 VB004, 项目中的 VB 文件为 Module1.vb, 在 Module1.vb 中添加代码如下。

Module Module1

Sub Main ()

REM 定义常量字符串

Const AString As String = "AaaAaaaabcccsfeadggeABCDDSSAEGDSAsdfesafEFGA"

Dim BString, CString As String

Dim CCH As Char

Dim iCount, ACounts As Integer

' 定义元素个数不确定的字符串数组

Dim Aarray () As String

Dim StrHead As String = "字符串中出现 A 的次数为"

' 将字符串 Astring 复制到 BString

BString = AString.Copy (AString)

' 输出字符串 AString

Console.WriteLine ("AString is " & AString)

REM 通过 4 种方法查找字符 A

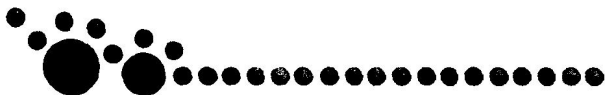
REM 方法一: 将 BString 以 A 字符为分隔符划分为字符串数组



```

Aarray = BString.Split ("A")
' 求数组上界,可知数组元素个数,即字符 A 的个数
ACounts = UBound (Aarray)
Console.WriteLine (StrHead & Convert.ToString (ACounts) )
REM 方法二：使用 String 类的查找方法 IndexOf
ACounts = 0
While BString.IndexOf ("A") >= 0
' 在字符串 BString 中循环查找字符 A, 如果 IndexOf 返回值大于等于 0, 则说明找到 A
' 如果 IndexOf 返回值小于 0, 则说明找不到 A 字符, 结束循环
    ' A 字符的个数加一
    ACounts += 1
    ' 删掉已经找到的字符 A
    BString = BString.Remove (BString.IndexOf ("A"), 1)
End While
Console.WriteLine (StrHead & Convert.ToString (ACounts) )
REM 方法三：依次验证字符串中的每一个字符是否是 A 字符
' 注意 String 类的第一个字符位置是 0
iCount = 0
ACounts = 0
While (iCount <= AString.Length - 1)
    CString = AString.Substring (iCount, 1)
    ' 如果字符串相等, Compare 函数返回值为 0
    If CString.Compare (CString, "A") = 0 Then
        ACounts += 1      ' A 字符个数加 1
    End If
    iCount += 1
End While
Console.WriteLine (StrHead & Convert.ToString (ACounts) )
REM 方法四：使用 Mid 函数依次获取字符串中的每个字符, 并验证其是否是字符 A
' 注意这里字符串的第一个子串位置为 1
iCount = 1
ACounts = 0
While (iCount <= AString.Length)
    CString = Mid (AString, iCount, 1)
    If CString = "A" Then
        ACounts += 1
    End If
    iCount += 1
End While
Console.WriteLine (StrHead & Convert.ToString (ACounts) )

```



```
' 等待用户输入回车
```

```
Console.Read ()
```

```
End Sub
```

```
End Module
```



解释

本例借助 String 类中定义的各种函数, 采用四种方法在字符串 AString (或 BString) 中查找字符 A, 并统计字符 A 在字符串 AString (或 BString) 中的出现次数。方法一是利用 String 类的 Split 函数, 以字符 A 为分割符, 将字符串 BString 划分为一个字符串数组, 统计数组的元素的个数即可知道字符 A 的出现次数。方法二是使用 String 类的 IndexOf 函数查找字符串中字符 A 出现的位置, 如果找到字符 A, 则使用 Remove 方法将字符 A 从原字符串中删除, 然后循环查找并删除下一个字符 A, 最终循环的次数即是原字符串中字符 A 出现的次数。方法三是循环调用 String 类的 Compare 函数, 依次验证原字符串中所有的字符, 并统计字符 A 出现的次数, 循环次数由字符串 AString 的长度 (Length) 确定。方法四是使用 VB 6.0 中定义的 MID 函数, 依次验证字符串 AString 中的每个字符, 并统计字符 A 出现的次数。

实例 005 输入简单表达式并求值



实例说明

本例接收用户输入的一个简单运算表达式, 并根据表达式计算相应的结果。本例综合运用 String 类的方法以及 Convert 类的方法, 对字符串进行处理和转换, 本例效果如图 1-5 所示。

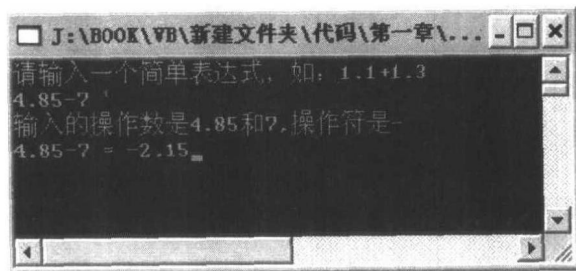


图 1-5 本例效果图



基本知识

String 类的 IndexOf 方法用于在字符串中查找指定的字符, 它需要一个参数, 用于表示要查找的字符, 返回值为该字符在原字符串中出现的位置。String 类的 SubString 方法用于获取原字符串的子串, 它需要两个参数, 分别表示子串的起始位置和长度。

一个运算表达式包含的信息主要有三部分, 即运算符 (操作符) 和参与运算的两个数 (操作数), 要对表达式进行计算, 就必须掌握这三部分的信息。因此, 本例的基本算法是, 利用 IndexOf 方法从输入的表达式字符串中查找操作符 (加号、减号、乘号或除号) 的位置, 然后将表达式字符串中位于操作符前后的两部分分别转化为两个操作数, 并进行

