

时尚百例丛书



Visual Basic 6.0 控件 时尚编程百例

100

时尚百例丛书

Visual Basic 6.0控件 时尚编程百例

网冠科技 编著

光盘包含本书素材、效果文件



机械工业出版社

Visual Basic 6.0 是 Microsoft 公司推出的一种可视化编程工具。本书从控件的角度讲解 Visual Basic 6.0 强大的开发工具。

本书通过 100 个实例的制作来叙述 Visual Basic 6.0 在控件方面的强大功能。全书分为九篇。第一至第三篇讲解 Visual Basic 6.0 的基本控件。第四至第八篇讲解 Visual Basic 6.0 中所有常见的 ActiveX 控件。第九篇讲解自制 ActiveX 控件的方法,读者可以按照书中内容的提示,一步步进行 ActiveX 控件的开发工作。

本书既可作为计算机专业不同层次培训班的教材,也可以作为命令参考手册使用查阅。

图书在版编目 (CIP) 数据

Visual Basic 6.0 控件时尚编程百例 / 网冠科技编著.

-北京:机械工业出版社, 2001.1
(时尚百例丛书)

ISBN 7-111-08457-8

I .V... II .网... III .Basic 语言-程序设计
IV .TP312

中国版本图书馆 CIP 数据核字 (2000) 第 55017 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策 划: 胡毓坚

责任编辑: 王琼先

责任印制: 路 琳

155158/06

北京市密云县印刷厂印刷 • 新华书店北京发行所发行

2001 年 6 月第 1 版 • 第 4 次印刷

787mm × 1092mm $\frac{1}{16}$ • 21.5 印张 • 532 千字

15001-20000 册

定价: 35.00 元 (含 1CD)

凡购本图书,如有缺页、倒页、脱页,由本社发行部调换
本社购书热线电话: (010) 68993821、68326677-2527

时尚百例丛书

追求时尚 追求完美

出版说明

随着 21 世纪的到来,人们更深切地感受到了计算机在生活和工作中的作用越来越重要,越来越多的职业需要具有计算机的应用技能。掌握计算机是职业的需要,更是事业发展的需要。

目前计算机技术不但广泛地应用在办公自动化中,它还全面渗透到各行各业。如果要从从事平面设计的相关行业,就应该学会平面设计软件,如 Photoshop、CorelDRAW、FreeHand 等;如果要从从事三维设计的相关行业,就应该学会三维设计软件,如 3DS MAX、Maya、Poser 等;如果要从从事多媒体设计的相关行业,就应该学会多媒体制作软件,如 Authorware、Director、Premiere 等;如果要从从事与网络相关的行业,就应该学会 Flash、Dreamweaver、Fireworks、ASP、PHP、JavaScript 等;如果要从从事建筑产品、工业产品设计的相关行业,就应该学会 AutoCAD、3DS VIZ、Protel 等;如果要从从事软件开发的相关行业,就应该学会 VB、VC、VFP、Delphi、PowerBuilder 等编程。

所有与计算机相关的职业都要求工作者有很强的计算机操作技能,做到运用自如,熟练而且深入地掌握软件的应用。而要做到这一点,必须从软件的各个方面入手,通过实例演练的方式训练自己,而且要反复练习,做到举一反三。

为了让读者能深入而且熟练地掌握相关软件的应用方法,机械工业出版社特别为广大读者推出了这套时尚百例丛书。本丛书对每一个应用软件精心制作了 100 个实例,其宗旨就是让读者全方位掌握软件的应用,为广大读者提供一条快速掌握计算机应用技能的捷径。

本丛书采用新颖的版式,将知识和实例紧密结合,通过对各种实例的详细讲解,使读者不必事先学习各种软件,而从实例的制作过程中体会到每个软件每项功能的使用方法,并自己做出各种实例效果,这样既节省了大量时间,同时也使读者有身临其境的感觉,并可以反复演练,将所学知识运用到职业工作中去。

书山有路勤为径。愿广大读者能通过本丛书的学习掌握计算机技能,并应用到自己的工作和事业中去。

机械工业出版社

前言

《Visual Basic 6.0 控件时尚编程百例》是“时尚百例丛书”中的一本。

Visual Basic 是 Microsoft 公司推出的一种可视化编程工具，自推出以来便以其强大的功能和易用性风靡全球，成为当今最普遍的可视化编程工具之一。Visual Basic 6.0 在各国程序开发人员中获得了广泛的好评。不少程序员已经将 Visual Basic 6.0 作为首选的编程开发工具。

Visual Basic 6.0 在数据库和网络方面的许多开发是通过控件来实现的。Visual Basic 6.0 中携带的控件比以往任何一个版本都多。控件中封装了更多的属性和方法，更好地实现了代码共享，大大节省了程序开发人员的时间。一个初级程序员即使不使用代码或者仅仅编写少量的代码，也能够借助于控件的帮助，开发出功能强大的程序来。不少资深程序员也认为：未来的软件开发，越来越趋向于控件方面的开发，控件将会成为软件开发的下一个浪潮。

本书按照不同的控件类型分成了若干个部分进行叙述，每部分都有一定的主题，并由浅入深地编排了 100 个精炼实用的例子，每个例子都具有较强的针对性，突出讲解了某一控件的具体功能。在书中很少有大段的代码和高深的术语出现，重点放在了实例的创建思路与过程上，便于读者掌握各个实例中包含的知识点。书中的语言通俗易懂，读者可以轻松地完成本书所有内容的学习。

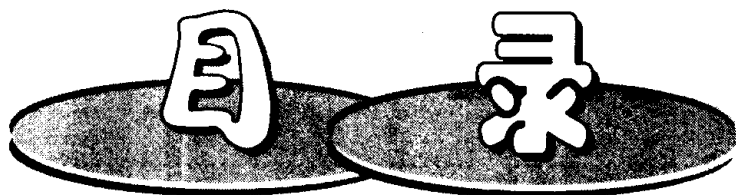
本书内容广泛，涉及到 Windows 编程的各个方面，根据各个控件所实现功能的不同，全书分为九篇。前三篇涉及到 Visual Basic 6.0 中的基本控件，读者可以进行一些初级的程序界面制作，掌握时间控制方面的知识，并就文件管理和数据库操作等内容进行学习。第四至第八篇讲解了 Visual Basic 6.0 中所有常见的 ActiveX 控件，这些控件类型丰富且功能强大，可以进行文字、图像、多媒体、网络与通信等方面的开发。最后一篇为讲解自制 ActiveX 控件的方法，读者可以按照书中内容的提示，一步步进行 ActiveX 控件的开发工作。



网冠科技

本书光盘含配套素材（使用方法请见光盘中“光盘使用说明书”），技术支持请点击网冠科技站点 <http://netking.163.com>。E-mail: netking_@yeah.net。是网易公司的标志。

網易 NETEASE
WWW.163.COM



出版说明 前 言

第一篇 初级界面控件

实例 1	我的工具箱	2
实例 2	加法器	5
实例 3	四则运算器	8
实例 4	造字程序	11
实例 5	调色程序	14
实例 6	图像浏览	17
实例 7	文本框示例	20
实例 8	考勤登记表	23
实例 9	画图板	27
实例 10	万花规	30
实例 11	课程查询	34
实例 12	直线拟合	38
实例 13	三角函数演示	42
实例 14	邮件注册系统	45
实例 15	简易记事本	48
实例 16	读书器	51

第二篇 Timer 控件

实例 17	弹球	55
实例 18	手表	59
实例 19	秒表程序	62
实例 20	数字式电子钟	66
实例 21	日历台	69
实例 22	红绿灯实例	72
实例 23	屏幕保护程序	75
实例 24	倒计时牌	78

实例 25	简单动画的实现	81
-------	---------------	----

第三篇 文件与数据库控件

实例 26	文件定位	85
实例 27	资源管理器	88
实例 28	创建数据库	92
实例 29	数据库管理(一)	96
实例 30	数据库管理(二)	99
实例 31	数据库管理(三)	102

第四篇 高级界面控件

实例 32	三维按钮控件演示	106
实例 33	系统信息	110
实例 34	工具栏制作(一)	114
实例 35	工具栏制作(二)	118
实例 36	模拟拷贝进程	123
实例 37	手动日历台	126
实例 38	打开对话框	130
实例 39	保存对话框	134
实例 40	颜色对话框	137
实例 41	字体对话框	140
实例 42	打印对话框	143
实例 43	帮助对话框	146

第五篇 度量控件

实例 44	温度计程序	151
实例 45	时速表程序	156
实例 46	拷贝进程(一)	159
实例 47	拷贝进程(二)	162
实例 48	雷达控制台	165
实例 49	VB 终端	169
实例 50	图像放大	172

第六篇 图文与多媒体控件

实例 51	图标浏览	176
-------	------------	-----



实例 52	图表控件演示	179
实例 53	高级记事本	182
实例 54	高级画图板	185
实例 55	动画大观	189
实例 56	聊天室注册系统	192
实例 57	VCD 播放器	195
实例 58	WAV 播放器	198
实例 59	CD 播放器	201
实例 60	AVI 播放器	204
实例 61	Midi 播放器	207

第七篇 网络控件

实例 62	网页浏览器	211
实例 63	聊天程序(客户端)	214
实例 64	聊天程序(服务器端)	218
实例 65	电话拨号程序	222

第八篇 其他高级控件

实例 66	显示器设置	226
实例 67	创建状态栏	229
实例 68	插入 BMP 图像	232
实例 69	插入 Word 文档	235
实例 70	插入 Excel 图表	238

第九篇 自制控件

实例 71	第一个用户控件的制作	242
实例 72	第一个 ActiveX 控件	247
实例 73	日期标签	251
实例 74	数字式时间标签	254
实例 75	图像式时间标签	257
实例 76	名片控件(一)	260
实例 77	名片控件(二)	263
实例 78	名片控件(三)	266
实例 79	资源浏览控件	270
实例 80	路径选择控件	272
实例 81	计时控件	275
实例 82	倒计时控件	278

实例 83	追随控件	281
实例 84	逃避控件	284
实例 85	工具栏控件(一)	287
实例 86	工具栏控件(二)	290
实例 87	计算控件	293
实例 88	特制文本框(一)	296
实例 89	特制文本框(二)	299
实例 90	特制文本框(三)	302
实例 91	背景控件	305
实例 92	动画按钮控件	308
实例 93	组合数据控件(一)	311
实例 94	组合数据控件(二)	314
实例 95	闪烁控件	318
实例 96	音乐控件	321
实例 97	图片浏览控件	324
实例 98	动画标签	327
实例 99	常用状态栏	330
实例 100	创建 ActiveX 文档	333



第一篇

初级界面控件

本篇总览

本篇主要介绍 Visual Basic 6.0 中，用于开发应用程序界面的几个基本控件。这些控件包括命令按钮、选择按钮、文本框、图像框等。在这些控件内部封装了大量的编程代码，包含大量的方法和属性。通过它们，读者可以轻易地开发出常见的 Windows 程序界面。

通过本篇中各个实例的学习，读者可以了解 Visual Basic 中控件的基本概念，并运用一些初级的界面控件来开发初级的应用程序。同时，读者也可以对面向对象的编程机制有一个大概的了解。



实例 1 我的工具箱

实例说明

本例的程序运行界面如图 1-1 所示。该程序用于对 Windows 操作系统中的常用工具（如记事本、画图板等）进行方便的管理。程序使用者单击界面中的不同图标，便可以启动相应的 Windows 程序。

本例的知识点有：介绍面向对象的编程机制，使读者对 Visual Basic 的编程原理有一个大概的了解；同时讲解 Visual Basic 6.0 中“CommandButton”控件的使用方法。



图 1-1 效果图

编程思路

本实例的创作主要使用了 Visual Basic 6.0 中的“CommandButton”控件。“CommandButton”是 Visual Basic 中应用最为广泛的一个按钮控件，大部分 Windows 程序都利用它来实现各种各样的功能。

在本例中，首先为“CommandButton”控件设置好外观和属性，为它添加图标，并标明文字。然后编写“Click”事件的驱动代码，当用户单击不同的控件时，能够通过“Shell”函数来调用相应的 Windows 程序，“Shell”函数的完整形式如下：

Shell (pathname [, windowstyle])

其中“pathname”是欲调用程序的路径，“windowstyle”参数则指明被调用函数的窗口以何种方式显示出来。本例中要调用的所有程序，都存放在“Windows”操作系统的根目录下。

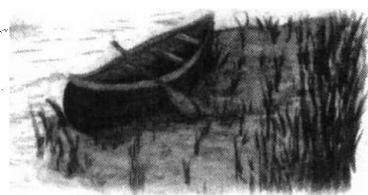
创作步骤

一、建立程序的基本框架

1. 启动 Visual Basic 6.0，程序启动后将首先弹出一个“新建工程”对话框，让使用者选择需要创建的工程类型。绝大部分用 Visual Basic 开发的程序是一般的可执行程序，在“新建工程”对话框中对应的选项是“标准 EXE”。选择该项，建立一个“标准 EXE”工程。如果 Visual Basic 事先已经启动，则单击“文件”菜单，选择“新建工程”项，建立一个同样的工程。

2. Visual Basic 将自动完成工程的各项创建工作，并建立一个用户工程主窗口，命名为“Form1”。这也是将来程序运行时的主窗口。拖动窗口四周的活动句柄，将该窗口调整到合适的大小。

3. 单击 Visual Basic 工具箱中的“CommandButton”控件，将鼠标移动到“Form1”窗口的左上角位



置，单击并拖动鼠标，将该控件添加到窗口中，并命名为“Command1”。也可通过双击 Visual Basic 工具箱中“CommandButton”控件的图标，将该控件添加到窗口中，不过在这种情况下，“Command1”将处于系统默认的位置（一般是窗口中间），并且具有系统默认的大小。

4. 用上述方法继续为“Form1”窗口添加“CommandButton”控件，一共添加 4 个，依次命名为“Command1”、“Command2”、“Command3”、“Command4”。

5. 如图 1-1 所示，将 4 个“CommandButton”控件的大小比例调整好，并放置在“Form1”窗口的相应位置中。

6. 用户控件的基本框架创建完毕。

二、设置各个基本控件的外观属性

1. 单击“Command1”控件，拖动其四周的活动句柄，将该控件大致调整到窗口的 1/4 大小。然后保持该控件的被选中状态，按住“Shift”键或“Ctrl”键不动，用鼠标依次单击余下的“Command”控件，将所有的“Command”控件选中，并确保“Command1”最后处于被激活状态（在选完所有的控件后单击“Command1”，使其四周的句柄为蓝色）。

2. 单击“格式”菜单，选择“统一尺寸”子菜单下的“两者都相同”项，将所有被选中的“Command”控件都调整为统一的尺寸，这项操作对美化程序界面十分有用，读者应该注意掌握。参照图 1-1 所示，排好这 4 个控件在“Form1”窗口中的位置。

3. 依次单击 4 个“Command”控件，在开发界面的属性窗口中将它们的“Caption”属性分别改为“记事本”、“画图板”、“写字板”和“CD 播放器”，如图 1-2 所示。

4. 单击“Command1”控件，在开发界面的属性窗口中找到“Style”选项，该选项用于确定“Command”控件的外观类型，可以有 2 个值供选择：“Standard”和“Graphical”。前者表示“Command”控件在外观上只能显示文字，而后者可在“Command”控件的外观上添加小图标或设置背景颜色等。很明显，在本例中应该选择“Graphical”项。

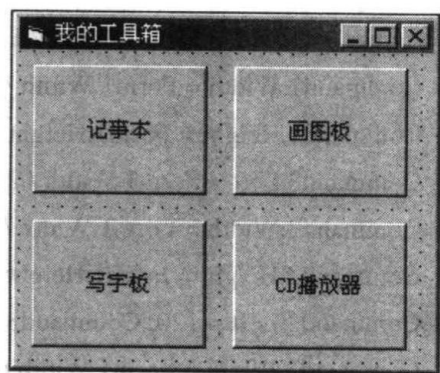


图 1-2 程序的基本框架

5. 设定好控件的外观类型后，继续在属性窗口中找到“Picture”项，单击赋值框中带有省略号的小按钮，则启动“加载图片”对话框让程序开发人员选择欲添加图片的路径。在 Visual Basic 的安装目录下有一个“Common\Graphics”文件夹，其中分类保存了大量用于开发 Windows 程序的图像和图标。我们可在“Common\Graphics\Icons\Writing”目录中任意找出一个笔记本的图标，将其添加到“Command1”控件中。

6. 用同样的方法，依次改动余下的“Command”控件的外观类型，并为它们一一指定相应的图标。

7. 最后单击“Form1”窗口，在属性窗口中将其“Caption”属性改为“我的工具箱”。

8. 程序界面的外观基本设置完毕。

三、为工程编写事件驱动代码

1. 本实例主要使用 Visual Basic 中的一个程序调用函数“Shell()”，实现程序的主要功能。该函数所需的各项参数已经在本文的“编程思程”部分作了详细介绍，只需在各个“Command”控件的“Click”事件中使用该函数即可。下面以“记事本按钮”为例。

```
Private Sub Command1_Click()
```

```
Shell "C:\windows\notepad.exe", 1
```

'1 表示程序窗口将以默认的大小显示

```
End Sub
```

2. 同理可得余下的“Command”控件的代码为:

```
Private Sub Command2_Click()
```

'画图板按钮的代码

```
Shell "C:\windows\pbrush.exe", 1
```

```
End Sub
```

```
Private Sub Command3_Click()
```

'写字板按钮的代码

```
Shell "C:\windows\write.exe", 1
```

```
End Sub
```

```
Private Sub Command4_Click()
```

'CD 播放器按钮的代码

```
Shell "C:\windows\cdplayer.exe", 1
```

```
End Sub
```

3. 最后编写代码来协调窗口和控件之间的大小关系,使得窗口最大化或者被用户调整大小时,各个控件也能随之改变自身大小来适应窗口的变化。实现这项功能需要解决 2 个问题:控制好各个控件的大小、调节好各个控件的坐标,代码如下:

```
Private Sub Form_Resize()
```

```
Command1.Left = 120: Command1.Top = 120
```

```
Command1.Width = Form1.Width / 2 - 480
```

```
Command1.Height = Form1.Height / 2 - 480
```

```
Command2.Left = Form1.Width / 2 + 120: Command2.Top = 120
```

```
Command2.Width = Form1.Width / 2 - 480
```

```
Command2.Height = Form1.Height / 2 - 480
```

```
Command3.Left = 120: Command3.Top = Form1.Height / 2
```

```
Command3.Width = Form1.Width / 2 - 480
```

```
Command3.Height = Form1.Height / 2 - 480
```

```
Command4.Left = Form1.Width / 2 + 120: Command4.Top = Form1.Height / 2
```

```
Command4.Width = Form1.Width / 2 - 480
```

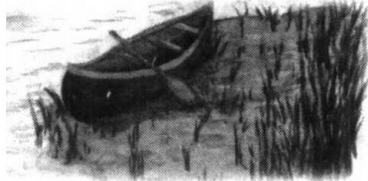
```
Command4.Height = Form1.Height / 2 - 480
```

```
End Sub
```

4. 程序代码编写完毕,存盘。

四、编译并运行程序

按 F5 键或单击 Visual Basic 工具栏上的运行按钮 , 编译并运行上述程序,最终效果如图 1-1 所示。



实例 2 加法器

实例说明

本例的运行效果如图 2-1 所示。这是一个实现简单加法运算的加法器。程序使用者可在加数框内填写好欲进行相加的 2 个加数，然后单击“相加”按钮，完成运算。

本例的知识点有：“Label”控件和“TextBox”控件的使用方法，以及怎样在 Visual Basic 中实现变量的加法运算。

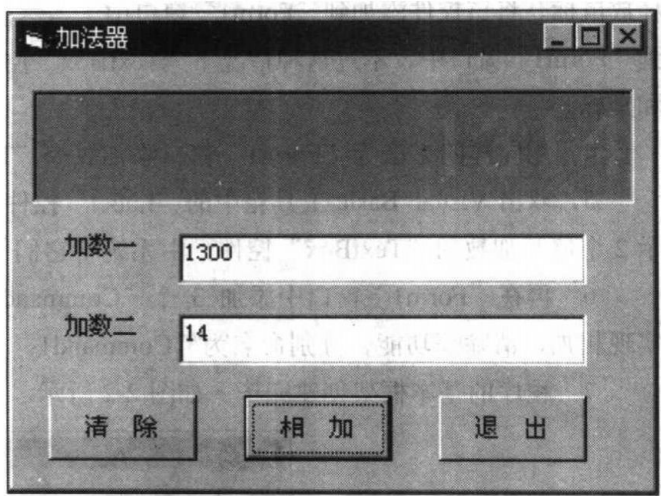


图 2-1 效果图

编程思路

本例使用 Visual Basic 6.0 中的“TextBox”、“Label”和“CommandButton”3 种基本控件。“TextBox”控件用于输入加数和显示运算结果；“Label”控件用于标示加数输入框；“Command”控件用于控制运算及清除结果等操作。

在程序运行之初，定义 3 个运行在程序幕后的变量：num1、num2 和 sum。它们分别代表 2 个加数和运算结果。通过一个变量转换函数 CDbl()，将 2 个“TextBox”控件中的字符型变量转换成双精度型的实型变量。CDbl() 函数的完整形式如下：

CDbl (expression)

其中“expression”代表需要转换的变量。

得到进行相加的 2 个变量后，在“相加”按钮中编写 2 个变量相加的代码，同时将运行的结果在另外一个“TextBox”控件中显示出来，并事先设置好该“TextBox”控件的显示字样，以使结果显眼且易于辨认。“清除”按钮的实现也十分简单，将 3 个用来输入/输出数据的“TextBox”控件清空即可，当然还要记住将 3 个在幕后运行的变量置零。

创作步骤

一、建立程序的基本框架

1. 启动 Visual Basic 6.0，创建一个“标准 EXE”工程。如果 Visual Basic 已经启动，则单击“文件”菜单，选择“新建工程”项来创建工程。

2. Visual Basic 将自动为用户创建一个名为“工程一”的标准工程。同时在工程中创建 2 个窗口：一个用于开发程序的运行界面，命名为“Form1”；另外一个用于填写程序的事件驱动代码。可在 Visual Basic



开发环境的“工程资源管理器”窗口中方便地切换这 2 个窗口。

3. 用鼠标单击 Visual Basic 工具箱中的“TextBox”控件，然后将鼠标移到“Form1”窗口中，单击并拖动鼠标，在拖动过程中可以看到一条虚线，那就是该控件将来的大小形状。调节虚线到合适的大小后释放鼠标，将该控件添加到“Form1”窗口中。当然也可通过双击工具箱中的图标将“TextBox”控件添加到“Form1”窗口中，不过这种情况下“TextBox”控件只有系统默认的大小和位置，还需另外调整其大小和坐标。

4. 用同样的方法为“Form1”窗口添加 3 个“TextBox”控件。

5. 双击 Visual Basic 工具箱中的“Label”控件，为“Form1”窗口添加 2 个“Label”控件，用于标示 2 个填写加数的“TextBox”控件，由系统将它们命名为“Label1”和“Label2”。

6. 再在“Form1”窗口中添加 3 个“CommandButton”控件，本例中将利用它们的“Click”事件来实现相加、清除等功能，分别命名为“Command1”、“Command2”和“Command3”。

7. 程序的基本框架创建完毕，如图 2-2 所示。

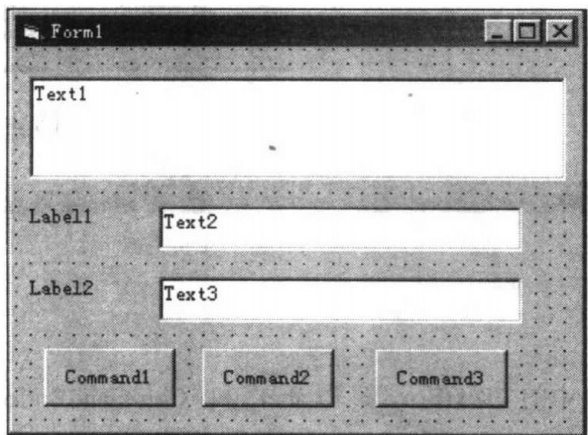


图 2-2 程序的基本框架

二、设置各个基本控件的外观属性

1. 单击“Text1”控件，在开发界面的属性窗口中找到“Font”选项，将该控件的显示字体设置为“黑体、粗体、小初”；再选中“Alignment”属性项，选择“Right”值，使显示的文本向右靠齐，符合人们日常使用计算器的习惯；然后把“BackColor”项属性设置为淡紫色，把“ForeColor”属性项设置为青色，就像图 2-1 所示的效果；最后在“Text”属性项中将该控件的显示内容清空。

2. 分别选中“Text2”控件和“Text3”控件，它们在本程序中起到输入数据的作用，在属性窗口中将它们的“Text”项都设置为“”，其他一切默认属性保持不变。

3. 单击用作输入的 2 个“TextBox”控件旁边的 2 个“Label”控件，分别将它们的“Caption”项属性改为“加数一”和“加数二”。

4. 最后改变三个“CommandButton”控件的外观属性，在 Visual Basic 开发界面的属性窗口中将它们的“Caption”项属性分别改动为“清除”、“相加”和“退出”。

5. 程序中使用的各个控件的外观属性基本设置完毕，如表 2-1 所示。

表 2-1 控件外观属性设置情况

控件名	控件属性项	控件属性值
Form1	Caption	加法器
Text1	Alignment	Right



(续)

控件名	控件属性项	控件属性值
Text1	Font	黑体、粗体、小初
	ForeColor	青色
	BackColor	淡紫色
	Text	“ ”
Text2	Text	“ ”
Text3	Text	“ ”
Label1	Caption	加数一
Label2	Caption	加数二
Command1	Caption	清除
Command2	Caption	相加
Command3	Caption	退出

三、为工程编写事件驱动代码

1. 首先编写代码来实现 2 个加数的输入，其中的最大障碍是变量类型的不匹配：在“TextBox”控件中输入的是字符型的变量，而程序中参与运算的必须是双精度型的实型变量。我们可以用一个变量类型转换函数 CDBl() 来解决上述问题，该函数的具体用法在本文的“编程思程”部分已经详细介绍，现列出其转换代码如下：

```
num1 = CDBl(Text2.Text)      '给加数一赋值
num2 = CDBl(Text3.Text)      '给加数二赋值
```

2. 处理了数据输入的问题，现在可以编写运算的代码了。其基本思路如下：在用户单击“相加”按钮后，将“Text1”控件中的值转换后传递给变量 num1，同样为 num2 赋值，然后将 2 个参与运算的变量进行相加，结果赋值给变量 sum，同时将变量 sum 转变成字符串类型后在“Text1”控件中输出。其实现的具体代码如下：

```
Private Sub Command2_Click()
    num1 = CDBl(Text2.Text)
    num2 = CDBl(Text3.Text)
    sum = num1 + num2
    Text1.Text = CStr(sum)      '显示运算结果
End Sub
```

3. 编写代码来实现“清除”按钮。其基本思路为：在按下该控件的同时，将参与运算的 num1、num2 和 sum 三个变量置零，同时将用于输入和输出的三个“TextBox”控件清空，代码如下：

```
Private Sub Command1_Click()
    Text1.Text = "" : Text2.Text = "" : Text3.Text = ""      '在 VB 中，“”通常代表清空
    num1 = 0 : num2 = 0 : sum = 0      '将参与运算的变量置零
End Sub
```

4. 最后编写代码来初始化程序的主窗口，我们必须在程序运行之初就调节好程序窗口的大小，并且将该窗口的边界锁定，以免用户改变窗口大小后影响各个控件之间的大小比例。

5. 程序的事件驱动代码编写完毕，存盘。

四、编译并运行程序

按 F5 键或单击 Visual Basic 工具栏上的运行按钮，编译并运行上述程序。



实例 3 四则运算器

实例说明

本实例的程序运行界面如图 3-1 所示。其功能类似于 Windows 系统自带的计算器，除了实现加、减、乘、除等简单四则运算外，还可以进行求模（余数）运算。整个程序使用了“CommandButton”、“TextBox”和“Label”三种控件，程序支持鼠标操作，通过鼠标的单击事件便可以实现程序中的所有功能。

本实例的知识点有：Visual Basic 控件的基本工作原理及其面向对象的编程机制，同时介绍 Visual Basic 的运算方法。读者可以在此例的基础上开发出功能强大的计算工具。



图 3-1 运算器

编程思路

本实例的关键之处在于对“CommandButton”控件的处理，围绕“CommandButton”控件的 Click 事件编写过程代码，实现数据运算；程序中全部的数字按钮和运算符按钮是一组“CommandButton”控件，用于显示操作数和计算结果的“屏幕”则是一个“TextBox”控件；程序开始运行时便在幕后定义“num1”、“num2”和“sum”三个单精度类型的变量，其中“num1”和“num2”为操作数，“sum”用于存储运算结果；它们在程序运行的过程中是不可见的，只能通过各个事件过程进行运算操作；另外，本程序将所有的“CommangButton”控件作成一个控件数组，以方便编程；而“TextBox”控件中的字符串则通过 CStr() 和 CDBl() 2 个函数实现与单精度型数据的交换。

创作步骤

一、创建程序的运行界面

1. 运行 Visual Basic 6.0，程序启动后，在弹出的“新建工程”对话框中选择“标准 EXE”工程。
2. 程序将创建一个名为“Form1”的工程窗口，用鼠标左键单击并选中，然后将窗口拖放到合适的大小，这也是将来程序主窗口的大小（为方便起见，本书中一切鼠标的事件，如单击、拖动等，如果不加特殊说明，都是指鼠标左键的事件）。