



普通高等教育“十一五”国家级规划教材

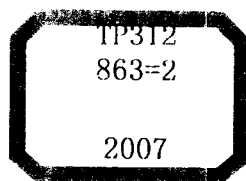
Visual Basic程序设计 实用教程 (第3版)

王 栋 编著



清华大学出版社

普通高等教育“十一五”国家级规划教材



Visual Basic程序设计 实用教程 (第3版)

王 栋 编著

清华大学出版社
北京

内 容 简 介

本书是 Visual Basic 的基础教程,详细介绍了 Visual Basic 语言与算法,通过大量的实例阐述了 Visual Basic 的基本概念、语言特性、程序设计的基本方法和技巧,以及面向对象的程序设计思想与事件驱动的编程机制。本书在叙述上力求做到严谨、详尽而又深入浅出,知识点的安排和格式的编排符合认知规律,是为方便教学而专门设计的。本书中每章都配有习题,所有的例子都调试通过。

本书的主要内容包括: Visual Basic 集成开发环境,数据类型、变量、常量、数组、运算符、表达式、控制结构与过程,窗体与各种内部控件的常用属性、方法和事件,常用内部函数,控件数组、菜单、绘图和文件操作等。

本书适合作为高等学校学生第一门计算机程序设计语言的教材,或供高年级学生学习使用;也可作为培训教程以及各类人员的学习资料和参考手册。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话: 010-62782989 13501256678 13801310933

图书在版编目(CIP)数据

Visual Basic 程序设计实用教程/王栋编著. —3 版. —北京:清华大学出版社,2007.2
ISBN 978-7-302-14447-2

I. V… II. 王… III. BASIC 语言—程序设计—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2006)第 163962 号

责任编辑: 闫红梅

责任校对: 时翠兰

责任印制: 王秀菊

出版发行: 清华大学出版社 地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编: 100084

c-service@tup.tsinghua.edu.cn

社 总 机: 010-62770175 邮购热线: 010-62786544

投稿咨询: 010-62772015 客户服务: 010-62776969

印 刷 者: 北京嘉实印刷有限公司

装 订 者: 三河市新茂装订有限公司

经 销: 全国新华书店

开 本: 185×260 印 张: 23 字 数: 554 千字

版 次: 2007 年 2 月第 3 版 印 次: 2007 年 2 月第 1 次印刷

印 数: 1~5000

定 价: 29.50 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系
调换。联系电话: (010)62770177 转 3103 产品编号: 020881-01

前言

这是《Visual Basic 程序设计实用教程》的第3版。

1999年,作者在写第1版时,那时的 Visual Basic 教材还很少,但作者已经预测到 Visual Basic 将作为高校非计算机专业第一门编程语言被广泛采用。七年的时间和市面上琳琅满目的 Visual Basic 书籍证明了这一点。

本书的第1版荣获南京理工大学优秀教材一等奖,第2版被评为江苏省精品教材,多所高校选为教学用书,受到了教师、学生和各类自学人员的好评。第3版入选教育部普通高等教育“十一五”国家级规划教材。借本次改版的机会,作者对内容和格式进行了全方位的删改和更新,使之达到作者心目中的“好学、好教、好用”的教材标准。


本书具有以下特点,也是作者的编写准则:(1)完全面向第一门语言,不需要任何编程经验;(2)基础知识、基本概念讲深、讲透;(3)知识点安排循序渐进,符合认知规律,便于自学与施教;(4)精心设计的近百个实例贯穿了基本算法,针对性极强的大量习题使读者可以巩固所学知识;(5)图表、附录、索引以及源程序的行号使得本书处处体现作者的匠心独具。

与本书配套的课件和例题、习题的源程序可在清华大学出版社的网站上(www.tup.tsinghua.edu.cn)下载,也可直接与作者联系获取更新的版本。

作者的另外两本后续教材——《Visual Basic 课程设计》和《Visual Basic 程序开发实例教程》(已由清华大学出版社出版)适用于学过“Visual Basic 程序设计”课程之后的实践环节和提高,欢迎选用。

感谢读者,感谢教师,感谢所有为本书出版和使用而辛勤劳作的人们。王涛、陆静、李向东、王芳、吴志林、袁红兵、张小兵、李忠新、张越、宋斌等教师也参与了本书的编写和改版工作,在此表示真诚的谢意。

因作者水平有限,书中定会有不少缺点或错误,恳请读者将意见和建议告知作者。



2006年12月于金陵月牙湖
wangdong@mail.njust.edu.cn

符号说明

(1) 为了便于阐述,本书使用了下列符号。这些符号在实际编程时不能使用。

【“|”表示“多项选其一”】 【“[]”表示“可选、可省略的内容”】



(2) 章节标题上或习题中标有“*”者,表示选学内容,对于 48 学时之内的学生不作要求。

(3) 附录 H“知识点索引”中,标有“★”者,表示重要的知识点。

常用关键字

And	As	Boolean	ByRef	Byte
ByVal	Call	Case	Change	CheckBox
Click	Close	ComboBox	CommandButton	Const
Currency	Data	Date	Debug	Dim
DblClick	Do	Double	Else	End
Eqv	Error	Exit	False	For
Form	Function	GoSub	GoTo	HScrollBar
If	Image	Imp	Input	Integer
Is	Label	Like	Line	ListBox
Long	Loop	Mod	Move	Next
Not	Object	On	Open	Optional
OptionButton	Or	ParamArray	PictureBox	Preserve
Print	Private	Public	ReDim	Rem
Return	Select	Shape	Single	Static
Stop	String	Sub	TextBox	Then
Timer	To	True	Type	TypeOf
Unload	Until	Variant	VScrollBar	Wend
While	Xor			

目 录

符号说明与常用关键字表	III
第 1 章 引言	1
1.1 程序设计语言	1
1.1.1 机器语言	1
1.1.2 汇编语言	1
1.1.3 面向过程的语言	2
1.1.4 面向对象的程序设计语言	2
1.1.5 程序的执行方式和运行环境	2
1.2 面向对象的基本概念	3
1.2.1 对象与类	3
1.2.2 属性	4
1.2.3 方法	4
1.2.4 事件	5
1.2.5 PME 模型	5
1.3 Visual Basic 简介	6
1.3.1 Visual Basic 发展史	6
1.3.2 Visual Basic 的特点	7
1.3.3 Visual Basic 的版本	7
1.3.4 Visual Basic 6.0 中文企业版的安装与启动	8
1.3.5 获得帮助信息	11
习题 1	14
第 2 章 程序设计入门	15
2.1 Visual Basic 集成开发环境	15
2.1.1 “工具箱”窗口	17
2.1.2 “工程”窗口	17
2.1.3 “属性”窗口	18
2.1.4 “窗体布局”窗口	19
2.1.5 “对象”窗口	19

2.1.6	“代码”窗口	21
2.1.7	窗口的排布	21
2.1.8	关闭工程与退出 Visual Basic 集成环境	22
2.2	程序的设计、运行和中断状态	22
2.3	窗体	23
2.3.1	窗体对象的创建	23
2.3.2	窗体对象的常用属性	24
2.3.3	通过程序代码为对象的属性赋值	27
2.3.4	窗体对象的常用方法	27
2.3.5	窗体对象的常用事件	29
2.4	编写事件过程	30
2.4.1	使用“代码”窗口	30
2.4.2	一个“最简单”的程序	32
2.4.3	两个简单程序	32
2.4.4	事件驱动机制	33
2.5	命令按钮控件	34
2.5.1	命令按钮的常用属性	34
2.5.2	命令按钮的 Move 方法	36
2.5.3	命令按钮的 Click 事件	36
2.6	文本框控件	37
2.6.1	文本框的常用属性	38
2.6.2	文本框的 Move 方法	40
2.6.3	文本框的常用事件	40
2.7	标签控件	41
2.7.1	标签的常用属性	41
2.7.2	标签的 Move 方法	41
2.7.3	标签的常用事件	42
2.8	Visual Basic 语法规则	42
2.9	开发应用程序的主要步骤	44
2.10	工程的管理和可执行文件的生成	45
2.10.1	工程中的模块与文件	45
2.10.2	向工程中添加模块	46
2.10.3	保存工程	48
2.10.4	打开工程	48
2.10.5	移除模块	49
2.10.6	工程组*	49
2.10.7	生成可执行文件	49
2.10.8	发布应用程序	50
	习题 2	50

第 3 章 数据类型、常量与变量	53
3.1 基本数据类型	53
3.1.1 数值型	53
3.1.2 String 型	53
3.1.3 Boolean 型	54
3.1.4 Date 型	54
3.2 直接常量	54
3.2.1 整型常量	54
3.2.2 浮点型常量	55
3.2.3 字符串型常量	55
3.2.4 逻辑型常量	56
3.2.5 日期时间型常量	56
3.3 变量	56
3.3.1 变量命名规则	56
3.3.2 定义变量	57
3.3.3 变量的赋值与取值	60
3.3.4 变量的同名问题	63
3.3.5 定长字符串与变长字符串变量	64
3.3.6 对象型变量	64
3.3.7 变体数据类型	65
3.3.8 类型转换	66
3.3.9 类型声明符 *	68
3.3.10 DefType 语句 *	68
3.4 符号常量	69
习题 3	70
第 4 章 运算符与表达式	74
4.1 运算符	74
4.1.1 算术运算符	74
4.1.2 比较运算符	75
4.1.3 字符串运算符	76
4.1.4 日期时间运算符	78
4.1.5 对象型比较运算符	78
4.1.6 逻辑运算符	79
4.2 表达式	82
4.2.1 表达式的求解顺序	82
4.2.2 运算符的优先级	83
4.2.3 使用括号改变计算顺序	84

4.2.4	正确编写表达式	84
4.2.5	表达式求值	86
习题 4	87
第 5 章	控制结构	90
5.1	If 语句	91
5.1.1	单行形式的 If...Then...语句	91
5.1.2	块形式的 If...Then...End If 语句	91
5.1.3	单行形式的 If...Then...Else...语句	92
5.1.4	块形式的 If...Then...Else...End If 语句	92
5.1.5	If 语句的嵌套	93
5.1.6	If...Then...ElseIf...End If 语句	96
5.2	Select Case 语句	98
5.2.1	Select Case 语句的语法结构	98
5.2.2	关于“匹配”的定义	99
5.3	Do...Loop 语句	100
5.3.1	Do While...Loop 形式	101
5.3.2	Do...Loop While 形式	102
5.3.3	Do Until...Loop 形式	103
5.3.4	Do...Loop Until 形式	103
5.3.5	Do...Loop 形式	104
5.3.6	Exit Do 语句	104
5.4	For...Next 语句	104
5.4.1	For...Next 语句	104
5.4.2	Exit For 语句	106
5.4.3	For...Next 循环的“终止值”和“步长”问题	106
5.5	While...Wend 语句*	106
5.6	循环的嵌套	106
5.6.1	嵌套的规则	106
5.6.2	Exit Do 和 Exit For 语句在循环嵌套时的作用	107
5.6.3	循环嵌套的执行流程	108
5.7	GoTo 语句、GoSub...Return 语句*	109
5.7.1	GoTo 语句	109
5.7.2	GoSub...Return 语句	109
5.8	With 语句	110
5.9	控制结构的应用	111
习题 5	122
第 6 章	过程	128
6.1	Sub 过程	128

6.1.1	定义 Sub 过程	128
6.1.2	调用 Sub 过程	130
6.1.3	通用过程的重名问题	131
6.1.4	过程调用时的执行流程	131
6.2	Function 过程	132
6.2.1	定义 Function 过程	132
6.2.2	调用 Function 过程	133
6.3	过程的参数传递方式	136
6.3.1	按值传递参数(ByVal)	136
6.3.2	按地址传递参数(ByRef)	137
6.4	可选参数 *	140
6.5	命名参数 *	140
6.6	递归	141
习题 6	143
第 7 章	数组与自定义数据类型	149
7.1	数组概述	149
7.2	常规数组	150
7.2.1	一维数组	150
7.2.2	二维数组	152
7.2.3	多维数组	153
7.2.4	常规数组占用的内存大小	153
7.3	动态数组	156
7.4	数组函数与语句	160
7.5	变体类型数组 *	161
7.6	数组作参数与返回值	163
7.6.1	数组作参数	163
7.6.2	不定数量的参数(ParamArray) *	165
7.6.3	函数返回数组 *	166
7.7	自定义数据类型	168
7.7.1	定义自定义数据类型	168
7.7.2	自定义类型的变量和数组	169
7.7.3	自定义数据类型参数	170
7.7.4	函数返回自定义类型值	170
习题 7	171
第 8 章	内部控件	177
8.1	图形与图像类控件	177
8.1.1	直线控件	177
8.1.2	形状控件	178

8.1.3	图像控件	180
8.1.4	图片框控件	181
8.1.5	使用图片框作控件容器	182
8.2	滚动条、框架与定时器控件	182
8.2.1	滚动条控件	182
8.2.2	框架控件	184
8.2.3	定时器控件	185
8.3	提供选项的控件	186
8.3.1	复选框控件	186
8.3.2	单选框控件	188
8.3.3	列表框控件	190
8.3.4	组合框控件	196
8.4	文件系统控件 *	197
8.4.1	驱动器列表框控件	198
8.4.2	目录列表框控件	199
8.4.3	文件列表框控件	199
8.4.4	联合使用三个文件系统控件	200
8.5	控件的键盘输入焦点与 Tab 键次序 *	200
8.6	鼠标与键盘事件	201
8.6.1	MouseDown 事件、MouseUp 事件、MouseMove 事件	202
8.6.2	MousePointer 属性、MouseIcon 属性 *	203
8.6.3	KeyDown 事件、KeyUp 事件	204
8.6.4	KeyPress 事件	204
8.6.5	KeyPreview 属性	205
8.6.6	SendKeys 语句 *	205
8.7	控件数组	206
8.7.1	创建控件数组	207
8.7.2	编写事件过程	208
8.7.3	动态添加、删除控件数组元素	210
8.8	菜单	210
8.8.1	菜单控件的属性	211
8.8.2	创建菜单	212
8.8.3	设置菜单控件的属性	213
8.8.4	菜单控件的 Click 事件	213
8.8.5	弹出式菜单 *	214
习题 8	215
第 9 章	内部函数	218
9.1	数学函数	218

9.2	字符串函数	220
9.3	日期与时间函数	229
9.4	类型测试函数 *	232
9.5	分支函数 *	233
9.6	预定义对话框函数	233
9.6.1	消息框函数 MsgBox	234
9.6.2	输入框函数 InputBox	236
习题 9	237
第 10 章	绘图 *	242
10.1	颜色	242
10.2	绘制文字与图形	244
10.2.1	输出文字	244
10.2.2	绘制图形	246
10.3	与绘图有关的属性、事件和方法	250
10.4	与文字输出有关的属性和方法	255
10.5	绘图坐标系统	257
习题 10	259
第 11 章	多模块程序设计与调试	263
11.1	多模块程序设计	263
11.1.1	启动对象	263
11.1.2	窗体的加载与卸载	264
11.1.3	窗体加载时的事件	266
11.1.4	窗体卸载时的事件	267
11.1.5	多模块之间的数据共享	268
11.1.6	程序的终止	269
11.2	程序的调试 *	270
11.2.1	错误的种类	270
11.2.2	调试窗口	270
11.2.3	切换到中断状态的方法	275
11.3	捕获并处理运行时错误 *	277
11.3.1	Err 对象	277
11.3.2	On Error 语句	277
11.3.3	Resume 语句	277
11.3.4	错误的捕获与处理	278
11.3.5	Err 对象的 Raise 方法和 Clear 方法	279
习题 11	279

第 12 章 文件操作	281
12.1 文件操作概述	281
12.1.1 文件操作的必要性	281
12.1.2 文件的标识方法	281
12.2 顺序访问文件	282
12.2.1 打开顺序文件	282
12.2.2 关闭文件	283
12.2.3 写顺序文件	283
12.2.4 读顺序文件	284
12.2.5 关于顺序文件的几点说明	285
12.3 随机访问文件	287
12.4 二进制文件	294
12.5 文件的共享与访问权限*	297
12.6 文件操作函数与语句	297
习题 12	306
附录 A 习题参考答案	313
附录 B 对象的命名前缀与默认属性	339
附录 C 变量的命名前缀	340
附录 D 键码	341
附录 E ASCII 码字符集	343
附录 F SendKeys 语句特殊击键	344
附录 G 可捕获的错误	345
附录 H 知识点索引	347
参考文献	351

引 言

从现在开始,要学习使用 Visual Basic 编制 Windows 操作系统下的应用程序了。如果将使用程序比作开汽车,那么编程就是制造汽车,在技术上属于更高的一个层次。编程是一项既有挑战性又有乐趣的智力活动,让我们在学习编程的过程中有所收获吧!

程序是计算机的灵魂,正是在各种类型程序的功能实现下,计算机才能像今天这样被广泛地应用。本书所提到的“程序”(Program)是指人们使用编程语言开发的、为了解决一定问题的、计算机能够执行的指令代码。

对于 Visual Basic 来说,我们是“用户”,因为我们使用它来编程;对于我们所编制的程序来说,我们是“作者”,我们要对程序和它将来的用户负责。有时,我们编程只是为了自己使用,所以我们又是自己所编程序的“用户”。作为“用户”,我们要掌握 Visual Basic 的编程知识;作为“作者”,我们应使用 Visual Basic 编写出好的程序。

1.1 程序设计语言

计算机程序设计人员(又称为编程人员、程序员,Programmer)编制程序是在一个被称为“计算机程序设计语言”的环境中进行的。计算机程序设计语言(又称为计算机语言、编程语言)是人为规定的、编程人员应遵守的、计算机可以识别的程序代码规则,是人指挥计算机进行工作、与计算机进行交流的工具。

程序设计语言是随着计算机技术的进步而不断地发展的。纵观其历史,可以将其分为机器语言、汇编语言、面向过程的高级语言和面向对象的高级语言四大类。

1.1.1 机器语言

机器语言是一种 CPU 指令系统,也称为 CPU 的机器语言,它是 CPU 可以识别的一组由 0 和 1 序列构成的指令码。用机器语言编程序,就是从所使用的 CPU 的指令系统中挑选合适的指令,组成一个指令序列。这种程序可以被机器直接理解并执行,速度很快,但由于它们不直观、难记、难以理解、不易查错、开发周期长,所以,现在只有专业人员在编制对于执行速度有很高要求的程序时才采用。

1.1.2 汇编语言

为了减轻编程者的劳动强度,人们使用一些帮助记忆的符号来代替机器语言中的 0、1 指令,使得编程效率和质量都有了很大的提高。由这些助记符组成的指令系统,称为汇编语

言。汇编语言指令与机器语言指令基本上是一一对应的。因为这些助记符号不能被机器直接识别,所以汇编语言程序必须被编译成机器语言程序才能被机器理解和执行。编译之前的程序被称为“源程序(Source Program)”,编译之后的程序被称为“目标程序(Object Program)”。

汇编语言与机器语言都是因 CPU 的不同而不同,所以统称为“面向机器的语言”。使用这类语言,可以编出效率极高的程序,但对程序设计人员的要求也很高。他们不仅要考虑解题思路,还要熟悉机器的内部结构,所以,非专业人员很难掌握这类程序设计语言。

1.1.3 面向过程的语言

与机器语言和汇编语言相比,面向过程的语言被称为“高级语言”。这类语言提供了大量的与人类语言相类似的控制结构和具有通用功能的函数库,使程序设计者可以不关心机器的内部结构甚至工作原理,把主要精力集中在解决问题的思路和方法上。这类摆脱了硬件束缚的程序设计语言的出现是计算机技术发展的里程碑,使得编程不再是少数专业人员的“专利”。

面向过程的语言引入了丰富的数据类型、关键字和语句,并允许将程序分解为多个子程序(或函数)。这使得同一个程序可由多人分工开发,大大提高了编程效率,使人们能够开发出规模越来越大、功能越来越强的应用软件和系统软件。

常用的面向过程语言有 C、Fortran、Basic、Pascal 等。

1.1.4 面向对象的程序设计语言

随着计算机技术的进一步发展,特别是像 Windows 这类具有图形用户界面(GUI, Graphics User Interface)的操作系统的广泛使用,人们又形成了一种面向对象的程序设计思想。这种思想把整个现实世界或是其一部分看作是由不同种类对象(Object)组成的有机整体。同一类型的对象既有共同点,又有各自不同的特性。各种类型的对象之间通过发送消息进行联系,消息能够激发对象作出相应的反应,从而构成了一个运动的整体。采用了面向对象思想的程序设计语言就是面向对象的程序设计语言,是面向过程高级语言的发展。当前使用较多的面向对象的语言有 Visual Basic、C++、C#、Java、Object Pascal 等。面向对象的程序设计方法又称为 OOP(Object Oriented Programming)。

随着人们对客观世界认识的不断变化以及计算机技术的不断发展,可以预料,将来一定会有新的计算机程序设计语言类型出现。

1.1.5 程序的执行方式和运行环境

计算机硬件在安装操作系统之后才能使用各种应用软件。作为最重要的一类系统软件,操作系统的功能包括处理器管理、存储管理、文件管理、设备管理和任务管理等。如图 1.1 所示,一般程序并不能直接操作计算机硬件,必须在操作系统的支持与控制下进行。

根据所开发程序的运行方式不同,高级语言可分为“解释型”和“编译型”两种。如果使用解释型语言,编好的源程序必须在程序设计语言环境的支持下,将语句转换为机器代码后执行,如图 1.1(a)所示。每次运行程序时,都要进行转换和执行两个步骤,所以解释型语言

的执行速度不快,源程序保密性不强,并且每次执行都不能离开语言环境。

编译型语言在程序设计完成之后,使用语言本身提供的编译(Compile)程序与连接(Link)程序把源程序编译为目标程序,再与库文件连接成为可执行程序(扩展名一般为“.exe”)。可执行程序能脱离源程序和语言环境独立运行,如图 1.1(b)所示。因为可执行程序是二进制格式,所以其执行效率和保密性都很高。

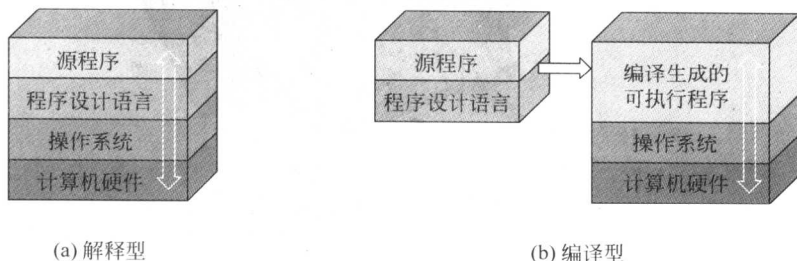


图 1.1 程序的运行环境

目前常用的编程语言多数是编译型的,如 C、C++、C#、Java 等。也有些语言(如我们学习的 Visual Basic)既可以解释型地运行程序,也可以对程序进行编译连接后运行。解释型运行一般用在程序的调试过程中,而设计完成之后便可将其编译成为独立的可执行文件发布给用户。

因为有了图 1.1 所示的程序执行层次关系,所以在大多数情况下,使用高级语言的编程人员没必要关心程序每一个细节在计算机中的执行情况(因为这些操作已交给操作系统或硬件完成),而是把主要精力集中于分析实际问题并使用语言提供的功能来解决问题。当然,对语言、操作系统及计算机硬件的结构与工作原理了解得越多,对编程的帮助就越大。

1.2 面向对象的基本概念

Visual Basic 是面向对象的编程工具,在面向对象的程序设计思想中,自然界中所有的事物(包括计算机中的事物)都可以被看作一个个的对象(Object)。针对对象的描述和操作方式是面向对象编程的技术核心。

1.2.1 对象与类

在图 1.2 所示的画面中,我们可以认为共有 7 个对象:3 个人、3 只杯子和 1 张桌子,都是独立的对象。在这 7 个对象中,3 个人属于同一类,3 只杯子又属于同一类,1 张桌子单独属于一个类。三个人虽然属于同一个类(Class),但是作为独立的对象又各有不同。所以,我们认为:类是同种对象的统称(或抽象),而对象是类的具体表现。

在编程过程中,我们可以将“对象”理解为编程的基本单元。如图 1.3 所示,程序的窗口界面就是由多个不同类型的对象组成的。文本框、单选框、复选框和按钮等类型的对象将在本书的第 2 章和第 8 章中详细讲解。

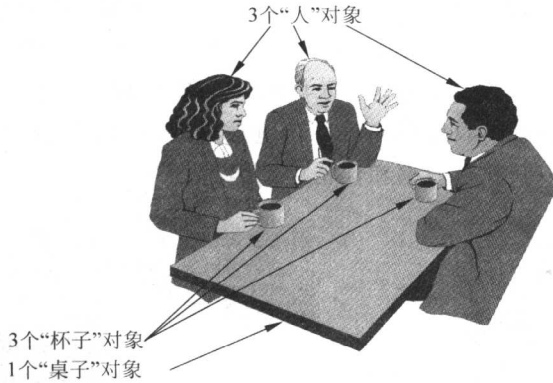


图 1.2 生活中的“对象”

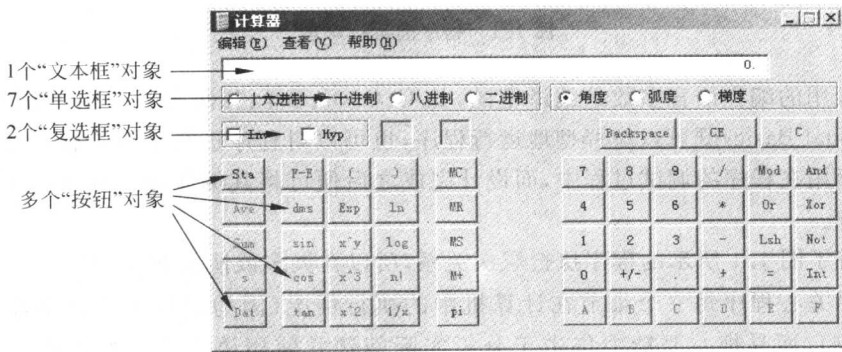


图 1.3 编程中的“对象”

在面向对象的思想中,对于任何一个对象,都可以从属性(Property)、方法(Method)与事件(Event)三个方面去描述它,这就是“PME 模型”。PME 模型将贯穿本书的始终,也是 Visual Basic 编程的基础。

1.2.2 属性

属性是指一个对象所具有的性质、特征。这些特征可能是看得见、摸得着的,也可能是内在的。例如,某个人(看作一个对象)的姓名叫张三,性别为男,身高是 1.75 米,学历为博士。这里的姓名、性别、身高和学历是这个人的属性。其中,“姓名”、“性别”、“身高”和“学历”称为属性名,而相应的“张三”、“男”、“1.75 米”和“博士”是属性值。

对于自然界中任何一个对象,可以从不同方面概括出它的许多属性来,并且每一个属性均有相应的属性值。比如,桌子可以有以下的属性:长、宽、高、材质、颜色、桌腿的数目、价格、制造厂、生产日期等。

在编程时,一个按钮是一个对象,它具有高度、宽度、颜色、所显示的文字等属性。

1.2.3 方法

方法指的是对象所具有的动作和行为。例如,一个人能够执行的动作和行为有:呼吸、