



普通高等教育“十二五”规划教材

# Visual Basic 程序设计语言

李良俊◎主编



科学出版社

普通高等教育“十二五”规划教材

# Visual Basic 程序设计语言

李良俊 主编

科学出版社

北京

## 内 容 简 介

本书详细介绍了 Visual Basic 6.0 程序设计的基本内容、基本应用, 在介绍可视化程序设计的同时, 也介绍了结构化程序设计的思想、方法。

本书共分 12 章, 主要内容包括: Visual Basic 概述、简单的 Visual Basic 程序设计、Visual Basic 程序设计基础、Visual Basic 程序控制结构、数组、Visual Basic 常用标准控件、过程和函数、应用程序调试、可视界面程序设计、图形操作、文件、数据库应用技术。

本书内容丰富, 知识讲解循序渐进, 并有配套的实验教材、电子教案和网络版考试系统, 适合作为普通高等院校本科及高职高专非计算机专业学生的使用教材, 可也供从事计算机应用和开发的相关人员参考。

### 图书在版编目(CIP)数据

Visual Basic 程序设计语言/李良俊主编. —北京: 科学出版社, 2015

(普通高等教育“十二五”规划教材)

ISBN 978-7-03-043503-3

I. ①V… II. ①李… III. ①BASIC 语言—程序设计—高等学校—教材  
IV. ①TP312

中国版本图书馆 CIP 数据核字 (2015) 第 034715 号

责任编辑: 宋 丽 张 斌 / 责任校对: 马英菊

责任印制: 吕春珉 / 封面设计: 多边数字

科学出版社 出版

北京东黄城根北街 16 号

邮政编码: 100717

<http://www.sciencep.com>

三河市骏杰印刷有限公司印刷

科学出版社发行 各地新华书店经销

\*

2015 年 2 月第 一 版 开本: 787×1092 1/16

2015 年 2 月第一次印刷 印张: 23 1/2

字数: 557 000

定价: 43.00 元

(如有印装质量问题, 我社负责调换<骏杰>)

销售部电话 010-62134988 编辑部电话 010-62135741

版权所有, 侵权必究

举报电话: 010-64030229; 010-64034315; 13501151303

# 前 言

信息技术的飞速发展，促进了计算机基础教育内容的更新。教育部非常重视普通高等院校非计算机专业的计算机基础教学，提出了“计算机文化基础”、“计算机技术基础”和“计算机应用基础”三个层次的课程体系，而后再根据形势的发展提出了“模块化”的教学体系。根据国家计算机基础教育教学指导委员会的指导意见，对非计算机专业的学生来说，学习高级语言程序设计课程，有利于培养其逻辑思维能力和计算思维能力，以及应用程序设计语言解决实际问题的能力。

Visual Basic 程序设计语言提供了开发 Windows 应用程序最迅速、最简洁的方法。它提供了一整套工具，供用户开发应用程序。鞍山师范学院的高级语言程序设计课程于 2008 年被评为省级精品课程，结合 2013 年学校立体化教材建设项目，综合多年的教学实践经验，我们参照全国高校非计算机专业计算机基础教育改革方案，在原来使用教材的基础上，结合学生及专业特点对教材内容进行了重新整理，补充了部分章节，使知识体系更为完整，教材配套更为齐全。整套教材由教材、实验教材、经典案例教程及学习指导与习题集组成。同时，我们制作了与教材配套的电子教案，开发了本课程的网络版考试系统。本书的编者都是从事高级语言程序设计课程教学多年的教师，具有丰富的教学经验和实践经验，在编写本书时力求做到理论和实践相结合，注重教学实践环节和学生应用能力培养相结合。

为了兼顾文、理、艺术、体育学生本身知识体系存在的差异，本书的内容尽可能详细明了、深入浅出，各章内容相对独立，可以根据实际教学情况进行取舍。

本书的编写分工如下：第 1 章和第 12 章由李良俊编写，第 2 章和第 3 章由徐然编写，第 4 章和第 5 章由姜红艳编写，第 6 章和第 7 章由何丽宏编写，第 8 章和第 9 章由姜秀玉编写，第 10 章和第 11 章由车元媛编写，全书由李良俊担任主编并统稿审定。

在编写本书过程中，得到了许多教师的大力支持和帮助，在此一并表示感谢。

由于编者水平有限，书中难免有不足之处，恳请读者批评指正！

编 者

2014 年 11 月

# 目 录

第 1 章 Visual Basic 概述	1
1.1 程序设计语言与程序设计	1
1.1.1 程序设计语言的发展	1
1.1.2 程序设计语言的分类	2
1.1.3 常见的程序设计语言	3
1.2 关于 Visual Basic	5
1.2.1 Visual Basic 的发展	5
1.2.2 Visual Basic 的功能特点	6
1.3 Visual Basic 6.0 的安装和启动	7
1.3.1 Visual Basic 6.0 的运行环境	7
1.3.2 安装 Visual Basic 6.0	7
1.3.3 启动 Visual Basic 6.0	8
1.4 Visual Basic 集成开发环境	9
1.4.1 标题栏	10
1.4.2 菜单栏	10
1.4.3 标准工具栏	11
1.4.4 工具箱	13
1.4.5 窗口	14
1.5 Visual Basic 6.0 帮助系统的使用	17
1.5.1 使用 MSDN Library 查阅器	18
1.5.2 上下文帮助	18
1.5.3 运行所提供的样例	19
习题	19
第 2 章 简单的 Visual Basic 程序设计	21
2.1 Visual Basic 中的基本概念	21
2.1.1 对象和类	21
2.1.2 对象的属性	22
2.1.3 对象的事件及事件过程	24
2.1.4 对象的方法	26
2.2 窗体的常用属性、方法和事件	27
2.2.1 窗体的添加与启动	27
2.2.2 窗体的属性	29
2.2.3 窗体的事件	36

2.2.4	窗体的方法	37
2.3	控件和常用控件的使用	40
2.3.1	控件概述	40
2.3.2	命令按钮控件	46
2.3.3	标签	47
2.3.4	文本框	49
2.4	Visual Basic 应用程序的结构及工作模式	53
2.4.1	Visual Basic 应用程序的结构	53
2.4.2	Visual Basic 应用程序的工作模式	55
2.4.3	代码窗口的设置	55
2.5	创建简单的 Visual Basic 应用程序	57
2.5.1	创建 Visual Basic 应用程序的一般步骤	57
2.5.2	创建 Visual Basic 应用程序实例	57
	习题	61
<b>第 3 章</b>	<b>Visual Basic 程序设计基础</b>	<b>64</b>
3.1	Visual Basic 字符集及程序的书写规则	64
3.1.1	Visual Basic 字符集	64
3.1.2	程序的书写规则与约定	65
3.2	数据类型	66
3.2.1	字符串类型	66
3.2.2	数值类型	67
3.2.3	其他类型	68
3.2.4	用户自定义类型	69
3.3	常量和变量	70
3.3.1	常量	70
3.3.2	变量	72
3.3.3	赋值语句	75
3.4	运算符与表达式	76
3.4.1	运算符与表达式概述	76
3.4.2	算术运算符与算术表达式	77
3.4.3	字符串运算符与字符串表达式	79
3.4.4	关系运算符与关系表达式	80
3.4.5	逻辑运算符与逻辑表达式	81
3.4.6	日期型运算符与日期型表达式	83
3.4.7	运算符的执行顺序	83
3.5	常用的内部函数	84
3.5.1	数学函数	84
3.5.2	转换函数	86



3.5.3 字符串函数 .....	87
3.5.4 日期函数 .....	88
3.5.5 其他函数 .....	90
3.5.6 立即窗口 .....	94
3.6 InputBox 函数 .....	95
3.7 MsgBox 函数和 MsgBox 过程 .....	97
习题 .....	100
<b>第 4 章 Visual Basic 程序控制结构 .....</b>	<b>103</b>
4.1 算法概述 .....	103
4.1.1 算法的相关知识 .....	103
4.1.2 算法的特性 .....	103
4.1.3 算法的表示 .....	104
4.2 顺序结构 .....	104
4.3 选择结构 .....	104
4.3.1 单行结构条件语句 .....	104
4.3.2 块结构条件语句 .....	106
4.3.3 Select Case 语句 .....	109
4.3.4 Iif 函数 .....	113
4.4 循环结构 .....	114
4.4.1 For 循环控制结构 .....	114
4.4.2 Do 循环控制结构 .....	119
4.4.3 当循环控制结构 .....	123
4.4.4 多重循环控制结构 .....	126
4.5 其他辅助控制语句 .....	128
4.5.1 GoTo 语句 .....	128
4.5.2 End 结束语句 .....	129
4.6 与控制结构相关的算法实例 .....	129
习题 .....	133
<b>第 5 章 数组 .....</b>	<b>140</b>
5.1 数组的概念 .....	140
5.2 静态数组 .....	142
5.2.1 静态数组的定义 .....	142
5.2.2 默认数组 .....	147
5.3 动态数组 .....	148
5.4 数组的基本操作 .....	150
5.4.1 数组的引用 .....	150
5.4.2 数组元素的输入 .....	151





5.4.3	数组元素的输出	152
5.4.4	数组元素的复制	153
5.4.5	求数组中最大元素及其下标和各元素之和	153
5.4.6	交换数组中各元素	154
5.4.7	插入和删除问题	154
5.4.8	数组的初始化	155
5.5	数组的应用举例	157
5.5.1	一维数组的应用	157
5.5.2	二维数组的应用	163
5.5.3	动态数组的应用	168
5.6	控件数组	169
5.6.1	控件数组的概念	169
5.6.2	创建控件数组	170
	习题	173
第 6 章	Visual Basic 常用标准控件	178
6.1	单选按钮和复选框	178
6.1.1	单选按钮	178
6.1.2	复选框	180
6.2	框架	183
6.3	滚动条	184
6.4	列表框、组合框	185
6.4.1	列表框	186
6.4.2	组合框	193
6.5	计时器	195
6.6	图形控件	196
6.7	焦点与 Tab 顺序	199
6.7.1	设置焦点	199
6.7.2	Tab 顺序	200
6.8	多重窗体	200
6.8.1	建立多重窗体应用程序	201
6.8.2	多重窗体程序的执行与保存	204
6.9	鼠标与键盘事件过程	206
6.9.1	鼠标事件	206
6.9.2	键盘事件	207
	习题	209



<b>第 7 章 过程和函数</b> .....	212
7.1 函数过程的定义和调用 .....	213
7.1.1 函数过程的定义 .....	214
7.1.2 函数过程的调用 .....	216
7.2 子过程的定义与调用 .....	217
7.2.1 子过程的定义 .....	218
7.2.2 子过程的调用 .....	219
7.3 参数传递 .....	221
7.4 过程、变量的作用域 .....	227
7.4.1 过程的作用域 .....	229
7.4.2 变量的作用域 .....	235
7.4.3 静态变量 .....	240
7.5 递归 .....	242
7.6 常用算法 .....	244
习题 .....	250
<b>第 8 章 应用程序调试</b> .....	251
8.1 程序错误分类 .....	251
8.1.1 编译错误 .....	251
8.1.2 运行错误 .....	252
8.1.3 逻辑错误 .....	252
8.2 调试应用程序 .....	253
8.2.1 Visual Basic 6.0 的工作模式 .....	253
8.2.2 调试工具 .....	255
8.2.3 调试方法 .....	256
8.2.4 使用调试窗口 .....	258
8.3 处理运行错误 .....	262
8.3.1 On Error 语句 .....	262
8.3.2 Resume 语句 .....	263
8.3.3 Err 对象 .....	264
<b>第 9 章 可视界面程序设计</b> .....	265
9.1 对话框 .....	265
9.1.1 自定义对话框 .....	265
9.1.2 通用对话框控件 .....	269
9.1.3 文件对话框 .....	270
9.1.4 “颜色”对话框 .....	273
9.1.5 “字体”对话框 .....	274

9.1.6	“打印”对话框	276
9.1.7	“帮助”对话框	277
9.2	菜单设计	277
9.2.1	菜单概述	277
9.2.2	菜单编辑器	278
9.2.3	创建下拉菜单	280
9.2.4	动态菜单	282
9.2.5	弹出菜单	283
9.3	多文档窗体 (Form)	284
9.3.1	多文档界面的结构	285
9.3.2	窗体菜单设计	286
9.3.3	多窗口管理	289
9.4	工具栏	290
9.4.1	关于 ActiveX 控件	291
9.4.2	创建工具栏	291
9.5	状态栏	294
	习题	295
<b>第 10 章</b>	<b>图形操作</b>	<b>300</b>
10.1	图形操作基础	300
10.1.1	坐标系统	300
10.1.2	自定义坐标系	301
10.2	绘图属性	303
10.2.1	设置当前坐标	303
10.2.2	AutoRedraw 属性	304
10.2.3	线宽、线型样式、填充	304
10.2.4	定义颜色	306
10.3	图形控件	307
10.3.1	直线控件	307
10.3.2	形状控件	308
10.4	图形方法	309
10.4.1	Cls 方法	309
10.4.2	Pset 方法	309
10.4.3	Line 方法	309
10.4.4	Circle 方法	310
	习题	311
<b>第 11 章</b>	<b>文件</b>	<b>314</b>
11.1	文件系统控件	314
11.1.1	驱动器列表框控件	314

11.1.2	目录列表框控件	315
11.1.3	文件列表框控件	316
11.1.4	3个控件的联动	317
11.2	文件及其结构	319
11.2.1	文件系统的基本概念	319
11.2.2	文件操作语句和函数	320
11.3	顺序文件	322
11.4	随机文件	326
11.5	二进制文件	329
11.6	文件的基本操作	330
	习题	332
<b>第 12 章</b>	<b>数据库应用技术</b>	<b>335</b>
12.1	数据库概述	335
12.1.1	关系数据库的基本结构	335
12.1.2	数据访问对象模型	336
12.1.3	记录集对象	337
12.2	本地数据库设计	337
12.2.1	可视化数据管理器	337
12.2.2	具体实现	339
12.3	数据控件	341
12.3.1	数据控件的属性	341
12.3.2	数据控件的事件	344
12.3.3	数据控件的常用方法	345
12.3.4	记录集的属性与方法	346
12.3.5	数据库记录的增、删、改操作	349
12.4	ADO 数据控件	353
12.4.1	ADO 对象模型	353
12.4.2	使用 ADO 数据控件	354
12.4.3	ADO 数据控件上新增绑定控件的使用	355
12.5	Visual Basic 中 SQL 的实现	356
	习题	359
	参考文献	361

# 第 1 章 Visual Basic 概述

程序设计语言是指能被计算机执行、完成用户指定功能的所有程序指令的集合。计算机只有接受并执行由程序设计语言编写的程序，才能完成用户的任务。本章从程序设计语言入手，以 Visual Basic 为例介绍程序设计语言的相关知识。

## 1.1 程序设计语言与程序设计

程序设计语言（也称算法语言）是人与计算机交流的工具，是由字、词和语法规则构成的指令系统。它是一种被标准化的交流技巧，用来向计算机发出指令。一种计算机语言让程序员能够准确地定义计算机所需要使用的数据，并精确地定义在不同情况下应当采取的行动。

人们应用程序设计语言完成程序设计任务时，一般分为 3 步：①系统分析；②编写程序；③调试运行。其中，系统分析和调试运行是程序设计的重中之重，关系着程序功能能否正确实现。

### 1.1.1 程序设计语言的发展

从 1946 年第一台计算机诞生起，在这短短的 60 多年间，计算机技术发展迅速，程序设计语言经历了机器语言、汇编语言到高级语言的转变，程序设计方法也从结构化程序设计方法向面向对象的程序设计方法过渡。

#### 1. 机器语言

机器语言与硬件直接相关，不同类别的计算机有着自己的机器语言。机器语言是一种二进制代码语言，由数字 0 和 1 组成。

用机器语言完成用户任务，就是编写出一串由“0”和“1”组成的指令序列交由计算机执行。由于机器语言程序完全由 0 和 1 代码组成，因此机器语言程序烦琐且难以读懂。此外，由于不同类别计算机的指令系统往往各不相同，因此，在一台计算机上执行的程序，要想在另一台计算机上执行，必须另编程序，这就造成了重复性工作。但由于机器语言程序能直接被计算机识别并执行，因此效率是所有语言中最高的。

#### 2. 汇编语言

为了减少使用机器语言编程的困难，在机器语言的基础上，人们进行了一种有益的改进：用一些英文缩写表示机器语言中的指令，如用 ADD 代表加法，MOVE 代表数据传递等。这样一来，就增强了程序的可读性、可理解性和可维护性，这种符号化的程序设计语言称为汇编语言。但汇编语言程序不能直接被计算机执行，这就需要有一个专门的程序负责将这些符号翻译成二进制的机器语言，这种翻译程序称为汇编程序。

汇编语言同样与硬件相关，移植性不好，但编程效率十分高。针对计算机特定硬件而编制的汇编语言程序，能准确发挥计算机硬件的功能和优势，程序精练且质量高，所以至今仍是一种常用的高效软件开发工具。

### 3. 高级语言

虽然汇编语言比机器语言容易理解，但即使实现简单的功能，其程序代码也很长，与数学语言或人类的自然语言差别较大。于是，在汇编语言的基础上，逐渐形成了与数学语言或人类的自然语言更为接近的高级语言。这类语言不依赖于计算机硬件，编制的程序能在所有机器上通用。1954年，第一种完全脱离机器硬件的高级语言 FORTRAN 问世，近60年来，共有几百种高级语言出现，其中影响较大、使用较普遍的有几十种，如 FORTRAN、ALGOL、COBOL、BASIC、LISP、SNOBOL、PL/1、Pascal、C、PROLOG、Ada、C++、Visual C++、Visual Basic、Delphi、Java 等。

高级语言程序依然不能被计算机直接识别并执行，而要通过翻译程序将其转换为计算机可直接执行的代码。翻译程序有两种工作方式：一种是用户每输入一条高级语言语句就将其转换为机器语言语句，这种方式称为解释方式，完成解释功能的程序称为解释程序；另一种是用户输入完所有高级语言程序语句后，对整个程序进行转换，这种方式称为编译方式，完成编译功能的程序称为编译程序。解释方式的交互性强，编译方式的速度更快。Visual Basic 中常采用解释方式。

高级语言的发展经历了从早期语言到结构化程序设计语言，从面向过程到面向对象程序设计语言的发展过程。

1969年，为克服软件生产中出现的各种问题，人们提出了结构化程序设计方法，1970年，第一种结构化程序设计语言 Pascal 出现，标志着结构化程序设计时期的开始。

从20世纪80年代初开始，在软件设计思想上又产生了一次革命，其成果就是面向对象的程序设计。在此之前的高级语言几乎都是面向过程的，程序的执行是流水线式的，在一个模块被执行完成前，人们无法动态地改变程序的执行方向。这和人们日常处理事务的方式是不一致的，因为人们希望发生一件事就处理一件事，也就是说，不能面向过程，而应面向具体的应用功能，即对象 (Object)。其方法就是软件的集成化，如同硬件的集成电路，生产一些通用的、封装紧密的功能模块，称为软件集成块，它与具体应用无关，但能相互组合，完成具体的应用功能，同时又能重复使用。对使用者来说，只关心它的接口 (输入量、输出量) 及能实现的功能，至于如何实现则是它内部的事，使用者完全不用关心，C++、Visual Basic、Delphi 就是其中的典型代表。

高级语言的发展趋势是模块化、简明性和形式化。

1) 模块化。不仅语言具有模块成分，程序由模块组成，而且语言本身的结构也是模块化的。

2) 简明性。涉及的基本概念不多，成分简单，结构清晰，易学易用。

3) 形式化。发展合适的形式体系，以描述语言的语法、语义、语用。

另外，程序设计语言的并行化和可视化也是人们追求的目标。

#### 1.1.2 程序设计语言的分类

构成程序设计语言的基本成分包括：①数据成分，用于描述程序所涉及的数据；②运

算成分, 用于描述程序中所包含的运算; ③控制成分, 用于描述程序中所包含的控制; ④传输成分, 用于表达程序中数据的传输。

程序设计语言按照语言级别可以分为机器语言、汇编语言和高级语言。机器语言、汇编语言与特定的机器有关, 效率高, 但使用复杂、烦琐、费时, 易出差错。高级语言的表示方法要比机器语言、汇编语言更接近于待解决问题的表示方法, 其特点是在一定程度上与具体机器无关, 易学、易用、易维护。

程序设计语言按照用户的要求, 有过程式语言和非过程式语言之分。过程式语言的主要特征是, 用户可以指明一系列可顺序执行的运算, 以表示相应的计算过程, 如 FORTRAN、COBOL、Pascal 等。

程序设计语言按照应用范围, 有通用语言与专用语言之分。例如, FORTRAN、COBOL、Pascal、C 等都是通用语言。目标单一的语言称为专用语言, 如 APT 等。

程序设计语言按照使用方式, 有交互式语言和非交互式语言之分。具有反映人机交互作用的语言称为交互式语言, 如 BASIC 等。不反映人机交互作用的语言称为非交互式语言, 如 FORTRAN、COBOL、ALGOL69、Pascal、C 等。

程序设计语言按照成分性质, 有顺序语言、并发语言和分布语言之分。只含顺序成分的语言称为顺序语言, 如 FORTRAN、C 等; 含有并发成分的语言称为并发语言, 如 Pascal、Modula 和 Ada 等; 考虑到分布计算要求的语言称为分布语言, 如 Modula 等。

### 1.1.3 常见的程序设计语言

#### 1. ActionScript

ActionScript 是 Flash 的脚本语言, 是在 Flash 动画设计中使用的编程语言, 是一种功能强大的、面向对象的编程语言, 可以制作出极富感染力的 Web 内容。

#### 2. Ada

Ada 是一种表现能力很强的通用程序设计语言, 它是美国国防部为克服软件开发危机, 耗费巨资, 历时近 20 年研制成功的。它被誉为第四代计算机语言的成功代表。

#### 3. 汇编语言

汇编语言 (Assembly Language) 是面向机器的程序设计语言。在汇编语言中, 用助记符 (Mnemonic) 代替操作码, 用地址符号 (Symbol) 或标号 (Label) 代替地址码。这样用符号代替机器语言的二进制码, 就把机器语言变成了汇编语言。汇编语言也称为符号语言。

#### 4. BASIC

BASIC 是一种高级语言, 它的英文含义是“初学者通用符号指令代码”, 是在 1965 年 5 月, 由美国科学家托马斯·库尔兹研制出来的。十多年后, 比尔·盖茨把它移植到 PC 上。以后的 30 多年来, BASIC 语言一直是初学计算机语言者使用最广泛的一种高级语言。它能进行数值计算、画图、演奏音乐, 功能十分强大, 学习起来也非常容易。



## 5. FORTRAN

FORTRAN 是英文“FORmula TRANslator”的缩写，译为“公式翻译器”，它是世界上最早出现的计算机高级程序设计语言，广泛应用于科学和工程计算领域。

## 6. VBScript

VBScript 是 Visual Basic Script 的简称，即 Visual Basic 描述语言，有时也被缩写为 VBS。VBScript 是 Microsoft 公司开发的一种脚本语言，可以视为 Visual Basic 语言的简化版，与 VBA 的关系也非常密切。它具有源语言容易学习的特性。目前，这种语言广泛应用于网页制作和 ASP 程序中。

## 7. C

C 语言是目前世界上非常流行且被广泛使用的高级程序设计语言之一。在操作系统、系统使用程序及需要对硬件进行操作的场合，用 C 语言明显优于其他高级语言，许多大型应用软件都是用 C 语言编写的。

## 8. COBOL

COBOL 是适合于商业及数据处理的类似英语的程序设计语言，可使商业数据处理过程精确表达。

## 9. HTML

HTML (Hypertext Markup Language, 超文本置标语言) 是 WWW 的描述语言。HTML 也是网络的通用语言，是一种简单、通用的全置标语言。

## 10. Java

Java 包含了一种计算机编程语言和一个平台。由 Sun 微系统公司发布，并作为一种开放的标准进行提供。Java 平台包括 Java 虚拟机和 Java 应用程序接口 (API)。Java 非常适合于企业网络和 Internet 环境，现在已成为 Internet 中最受欢迎、最有影响力的编程语言之一。

## 11. JavaScript

Java Script 是一种由 Netscape 公司的 LiveScript 发展而来的脚本语言，它提高了与 Java 的兼容性，是适应动态网页制作需要而诞生的一种新的编程语言。

## 12. SQL

SQL 的全称是“结构化查询语言”(Structured Query Language)。目前，常用的数据库开发系统都支持 SQL 语言作为查询语言。

## 13. Visual Basic

Visual Basic (VB) 是一种由 Microsoft 公司开发的包含协助开发环境的事件驱动编程



语言，是一种具有很高通用性的高级程序设计语言。

## 1.2 关于 Visual Basic

什么是 Visual Basic? “Visual”的英文原意是“可视的”、“视觉的”，在这里指的是开发图形用户界面（GUI）的方法，即“可视化程序设计”。在图形用户界面下，不需要编写大量代码去描述界面元素的外观和位置，只要把预先创建的控件加到屏幕上的适当位置，再进行简单的设置即可。“Basic”指的是 BASIC（Beginners All-Purpose Symbol Code）语言，是一种应用十分广泛的计算机语言。Visual Basic 在原有 BASIC 语言的基础上进一步发展，至今包含了数百条语句、函数及关键词，其中很多和 Windows GUI 有直接关系。专业人员可以用 Visual Basic 实现其他任何 Windows 编程语言的功能，而初学者只要掌握几个关键词就可以创建简单的应用程序。

### 1.2.1 Visual Basic 的发展

美国 Microsoft 公司于 1991 年推出 Visual Basic 1.0 版，随后于 1992 年秋推出 2.0 版，1993 年 4 月推出 3.0 版，1995 年 10 月推出 4.0 版，1997 年推出 5.0 版，1998 年推出 6.0 版。随着版本的提升和功能的改进，Visual Basic 已逐渐成为简单易学、功能强大的编程工具。从 1.0 版到 4.0 版，Visual Basic 只有英文版；而 5.0 版以后的 Visual Basic 在推出英文版的同时，又推出了中文版，这大大方便了中国用户。

Visual Basic 6.0 是专门为 Microsoft 的 32 位操作系统设计的，可用来建立 32 位的应用程序。在 Windows 9x、Windows NT 或 Windows 2000 环境下，用 Visual Basic 6.0 的编译器可以自动生成 32 位应用程序。这样的应用程序在 32 位操作系统下运行，速度更快更安全，并且更适合在多任务环境下运行。

Visual Basic 6.0 包含 3 种版本，分别为学习版、专业版和企业版。这些版本是在相同的基础上建立起来的，因此大多数应用程序可在这 3 种版本中通用。3 种版本适合于不同的用户层次。

1) 学习版是 Visual Basic 的基础版本，可使编程人员轻松开发 Windows 应用程序。该版本包括所有的内部控件（标准控件）、网格（Grid）控件、Tab 对象及数据绑定控件。

2) 专业版为专业编程人员提供了一整套功能完备的软件开发工具。该版本包括学习版的全部功能，同时包括 ActiveX 控件、Internet 控件、Crystal Report Writer 和报表控件。

3) 企业版使得专业编程人员能够开发功能强大的组内分布式应用程序。该版本包括专业版的全部功能，同时具有自动化管理器、部件管理器、数据库管理工具、Microsoft Visual SourceSafe 面向工程版的控制系统等。

3 种版本中，企业版功能最全，而专业版包括了学习版的功能，用户可以根据自己的需要选用不同的版本。对于大多数用户来说，专业版完全可以满足需要。

本书介绍和应用的是 Visual Basic 6.0 中文企业版，其内容适用于专业版和学习版，所有程序都可以在专业版和学习版中运行。

## 1.2.2 Visual Basic 的功能特点

### 1. 面向对象的可视化设计工具

在 Visual Basic 6.0 中,应用面向对象的程序设计方法(OOP),把程序和数据封装在一起视为一个对象,而且每个对象都是可视的。程序员在设计时只需用现有工具根据界面设计的要求,直接在创建程序时系统自动生成的“窗体”上,画出如按钮、滚动条、文本框等不同类型的对象,并为它们设置属性,编写事件代码即可。从而大大简化了设计的复杂程度,使程序设计效率大大提高。

### 2. 面向对象的程序设计

在一般的面向对象程序设计语言中,对象由程序代码和数据组成,是抽象的概念;而 Visual Basic 则是应用面向对象的程序设计方法,把程序和数据封装起来作为一个对象,并为每个对象赋予应有的属性,使对象成为实在的东西。在设计对象时,不必编写创建和描述每个对象的程序代码,而是用工具画在界面上,Visual Basic 自动生成对象的程序代码并封装起来。每个对象以图形方式显示在界面上,都是可见的。

### 3. 事件驱动的编程机制

事件驱动非常适合图形用户界面的编程方式。在图形界面的应用程序中,用户的动作(即事件)掌握着程序的运行走向,每个事件都可以驱动一段程序的运行。程序员只需要编写响应用户动作的代码,各个动作之间不一定有必然的联系。这样的代码一般比较短,使得程序易于开发和维护。

Visual Basic 通过事件来执行对象的操作。一个对象可能会产生多个事件,每个事件都可以通过一段程序来响应。例如,命令按钮是一个对象,当用户单击该按钮时,将产生一个“单击”(Click)事件,而在产生该事件时将执行一段程序,用来实现指定的操作。

### 4. 结构化的程序设计语言

Visual Basic 具有丰富的数据类型和结构化程序结构,其特点如下:

- 1) 增强了数值和字符串处理能力,比传统的 BASIC 语言有许多的改进。
- 2) 提供了丰富的图形及动画指令,可方便地绘制各种图形。
- 3) 提供了定长和动态(变长)数组,有利于简化内存管理。
- 4) 增加了递归过程调用,使程序更为简练。
- 5) 提供了一个可供应用程序调用的包含多种类型的图标库。
- 6) 具有完善的调试、运行出错处理能力。

### 5. 提供了应用程序集成开发环境

在 Visual Basic 6.0 集成开发环境(IDE)中,用户可以设计程序的界面、编写事件代码、调试程序,直至最后将程序编译成可执行文件。所有操作均可以通过 IDE 提供的各种菜单或工具按钮来完成。