

21世纪高等学校计算机规划教材

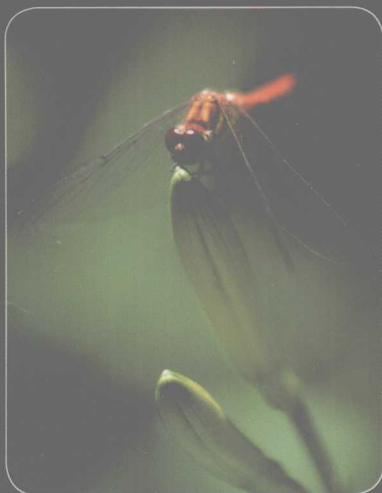
21st Century University Planned Textbooks of Computer Science

# Visual Basic 程序设计实用教程

A Practical Coursebook On Visual Basic  
Programming

匡松 缪春池 主编 王超 何振林 王锦 蒋义军 副主编

- 突出可视化程序设计的思想、方法和技术
- 强调实用性，注重实际编程能力的培养
- 满足计算机等级考试二级VB考试的需要



精品系列

 人民邮电出版社  
POSTS & TELECOM PRESS

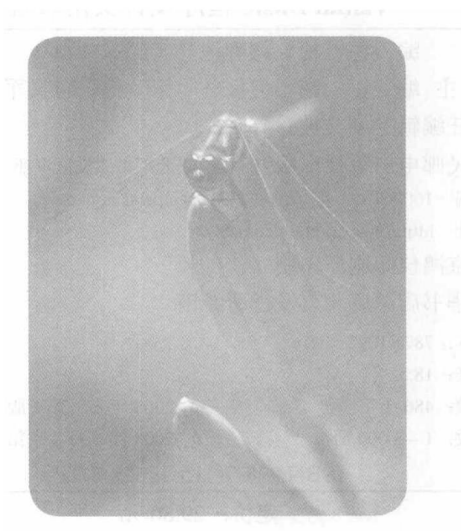
21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

# Visual Basic 程序设计实用教程

A Practical Coursebook On Visual Basic  
Programming

匡松 缪春池 主编 王超 何振林 王锦 蒋义军 副主编



精品系列

人民邮电出版社

北京

## 图书在版编目 (CIP) 数据

Visual Basic 程序设计实用教程/匡松, 缪春池主编.  
北京: 人民邮电出版社, 2008.6  
21 世纪高等学校计算机规划教材  
ISBN 978-7-115-17628-8

I. V… II. ①匡…②缪… III. BASIC 语言—程序设计—  
高等学校—教材 IV.TP312

中国版本图书馆 CIP 数据核字 (2008) 第 018423 号

## 内 容 提 要

本书根据 Visual Basic 教学大纲和全国计算机等级考试新大纲规定的二级 Visual Basic 考试内容的要求编写而成。全书共 14 章, 详细地介绍了 Visual Basic 可视化编程的基本方法, 包括常用控件的使用、Visual Basic 语言基础、数据的输入输出、键盘和鼠标事件、程序控制结构、构造数据类型、过程的建立与调用、对话框设计、菜单设计、多重窗体和多文档界面设计、文件操作、图形设计与操作、Visual Basic 对数据库的访问等内容。每章都提供了多种形式的练习题 (均有参考答案) 和上机操作题, 尤其便于自学。

本书内容系统全面, 实例丰富, 文字叙述简明易懂, 强调实用性和可操作性, 注重程序设计能力的培养。

本书可作为高等学校学生学习 Visual Basic 程序设计的教材, 也可作为参加全国计算机等级考试二级 Visual Basic 的人员或编程初学者的自学用书。

21 世纪高等学校计算机规划教材

### Visual Basic 程序设计实用教程

- 
- ◆ 主 编 匡 松 缪春池  
副 主 编 王 超 何振林 王 锦 蒋义军  
责任编辑 邹文波
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子函件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京鸿佳印刷厂印刷  
新华书店总店北京发行所经销
  - ◆ 开本: 787×1092 1/16  
印张: 18.5  
字数: 486 千字 2008 年 6 月第 1 版  
印数: 1-3 000 册 2008 年 6 月北京第 1 次印刷

---

ISBN 978-7-115-17628-8/TP

定价: 29.80 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223

反盗版热线: (010)67671154

## 出版者的话

计算机应用能力已经成为社会各行业最重要的工作要求之一，而计算机教材质量的好坏会直接影响人才素质的培养。目前，计算机教材出版市场百花争艳，品种急剧增多，要从林林总总的教材中挑选一本适合课程设置要求、满足教学实际需要的教材，难度越来越大。

人民邮电出版社作为一家以计算机、通信、电子信息类图书与教材出版为主的科技教育类出版社，在计算机教材领域已经出版了多套计算机系列教材。在各套系列教材中涌现出了一批被广大一线授课教师选用、深受广大师生好评的优秀教材。老师们希望我社能有更多的优秀教材集中地呈现在老师和读者面前，为此我社组织了这套“21世纪高等学校计算机规划教材-精品系列”。

“21世纪高等学校计算机规划教材-精品系列”具有下列特点。

(1) 前期调研充分，适合实际教学需要。本套教材主要面向普通本科院校的学生编写，在内容深度、系统结构、案例选择、编写方法等方面进行了深入细致的调研，目的是在教材编写之前充分了解实际教学的需要。

(2) 编写目标明确，读者对象针对性强。每一本教材在编写之前都明确了该教材的读者对象和适用范围，即明确面向的读者是计算机专业、非计算机理工类专业还是文科类专业的学生，尽量符合目前普通高等教学计算机课程的教学计划、教学大纲以及发展趋势。

(3) 精选作者，保证质量。本套教材的作者，既有来自院校的一线授课老师，也有来自IT企业、科研机构等单位的资深技术人员。通过他们的合作使老师丰富的实际教学经验与技术人员丰富的实践工程经验相融合，为广大师生编写出适合目前教学实际需求、满足学校新时期人才培养模式的高质量教材。

(4) 一纲多本，适应面宽。在本套教材中，我们根据目前教学的实际情况，做到“一纲多本”，即根据院校已学课程和后续课程的不同开设情况，为同一科目提供不同类型的教材。

(5) 突出能力培养，适应人才市场要求。本套教材贴近市场对于计算机人才的能力要求，注重理论技术与实际应用的结合，注重实际操作和实践动手能力的培养，为学生快速适应企业实际需求做好准备。

(6) 配套服务完善，共促提高。对于每一本教材，我们在教材出版的同时，都将提供完备的PPT课件，并根据需要提供书中的源程序代码、习题答案、教学大纲等内容，部分教材还将在作者的配合下，提供疑难解答、教学交流等服务。

在本套教材的策划组织过程中，我们获得了来自清华大学、北京大学、人民大学、浙江大学、吉林大学、武汉大学、哈尔滨工业大学、东南大学、四川大学、上海交通大学、西安交通大学、电子科技大学、西安电子科技大学、北京邮电大学、北京林业大学等院校老师的大力支持和帮助，同时获得了来自信息产业部电信研究院、联想、华为、中兴、同方、爱立信、摩托罗拉等企业 and 科研单位的领导或技术人员的积极配合。在此，人民邮电出版社向他们表示衷心的感谢。

我们相信，“21世纪高等学校计算机规划教材-精品系列”一定能够为我国高等院校计算机课程教学做出应有的贡献。同时，对于工作欠缺和不妥之处，欢迎老师和读者提出宝贵的意见和建议。

# 前 言

经常有同学问：学习编程，应该从哪种语言入手？绝大多数开发高手会毫不犹豫地给出答案：当然是 Visual Basic。

Visual Basic 以其可视化的应用界面开发方法（如 RAD 等），良好的数据库应用支持（如 ADO 等），加上出自名门（Microsoft），因此不但简单好用，而且功能强大，深得广大开发人员和编程爱好者的喜爱。虽然 VB.net 已推出多年，从目前的情况看，Visual Basic 仍然是程序设计入门和简单系统开发的第一选择。

本书根据 Visual Basic 教学大纲和全国计算机等级考试新大纲规定的二级 Visual Basic 考试内容的要求编写而成。全书共 14 章，详细地介绍了 Visual Basic 可视化编程的基本方法、常用控件的使用、Visual Basic 语言基础、数据的输入输出、键盘和鼠标事件、程序控制结构、构造数据类型、过程的建立与调用、对话框设计、菜单设计、多重窗体和多文档界面设计、文件操作、图形设计与操作、Visual Basic 对数据库的访问等内容。每章都提供了多种形式的练习题（提供有参考答案）和上机操作题，供读者练习和实践。

为了更有利于激发学生学习 Visual Basic 的兴趣，尽快掌握可视化程序设计的方法、技术精髓，与大多数 Visual Basic 教材相比，本书在内容及结构的安排上有较大改变。比如，第 2 章常用控件的使用，传统教材几乎都是安排在后面介绍，而根据编者多年的教学经验，常用控件这部分内容最能激发学生的学习兴趣。对这部分内容的学习，无论是课堂讲授，还是上机实验，最容易让学生体会到可视化编程的美妙。学生通过并不复杂的设计和操作，很快就会看到屏幕上出现的画面，成就感由此而生，同时也理解了什么是面向对象程序设计。当然，使用本教材的教师也可以根据实际教学情况来调整某些章节的教学顺序。

本书内容系统全面，实例丰富，文字叙述通俗易懂，尽量将复杂的问题简单化，设计方法尽可能简捷和明了，程序功能力求完善，强调实用性和可操作性，注重程序设计能力的培养。通过对本书的学习，读者能快速地掌握 Visual Basic 的编程方法和技巧，得心应手地解决实际问题，不仅可以全面掌握 Visual Basic 面向对象可视化程序的设计方法与开发技术，同时能充分满足参加全国计算机等级考试二级 Visual Basic 考试的需要。

本书由匡松、缪春池主编和统稿，王超、何振林、王锦、蒋义军担任副主编。本书的主要编写人员还有吕峻闽、李鹏、邢跃、高志坚、刘亚平、喻敏、张承虎、刘亚蓉、周敏、邱绪桃、程瑶、胡长清等。

本书可作为高等学校学生学习 Visual Basic 程序设计的教材，也可作为参加全国计算机等级考试二级 Visual Basic 的人员或编程初学者的自学用书。

编 者

2008 年 2 月

# 目 录

## 第 1 章 Visual Basic 可视化编程

### 初步 .....1

1.1 Visual Basic 的基本操作 .....1
1.1.1 启动 Visual Basic .....2
1.1.2 退出 Visual Basic .....3
1.1.3 Visual Basic 的集成开发环境 .....3
1.2 对象的基本知识 .....8
1.2.1 对象和类 .....8
1.2.2 对象的属性、事件和方法 .....9
1.2.3 对象属性的设置 .....10
1.3 窗体的基本知识 .....11
1.3.1 窗体的结构 .....11
1.3.2 窗体的属性 .....11
1.3.3 窗体的方法和事件 .....12
1.4 控件的基本知识 .....13
1.4.1 控件的类型 .....13
1.4.2 控件的命名和控件值 .....14
1.4.3 绘制控件 .....14
1.4.4 控件的基本操作 .....14
1.5 Visual Basic 编程基础 .....15
1.5.1 Visual Basic 应用程序的结构与 工作方式 .....15
1.5.2 Visual Basic 程序的书写规则 .....16
1.5.3 面向对象编程的基本步骤 .....16
1.5.4 建立简单的 Visual Basic 应用程序 实例 .....19
练习题 1 .....22

## 第 2 章 常用控件的使用 .....24

2.1 Visual Basic 的控件类型 .....24
2.2 常用标准控件的使用 .....24
2.2.1 文本框控件 .....25
2.2.2 标签控件 .....27
2.2.3 图形控件 .....29
2.2.4 命令按钮控件 .....30

2.2.5 单选按钮控件 .....32
2.2.6 复选框控件 .....33
2.2.7 列表框控件 .....35
2.2.8 组合框控件 .....37
2.2.9 滚动条控件 .....39
2.2.10 定时器控件 .....41
2.2.11 框架控件 .....42
2.2.12 焦点与 Tab 键顺序 .....43
2.3 综合应用——简易计算器的设计 .....44
练习题 2 .....48

## 第 3 章 Visual Basic 语言基础 .....54

3.1 常量 .....54
3.1.1 常数 .....54
3.1.2 符号常量 .....55
3.2 变量 .....56
3.2.1 变量的隐式声明 .....56
3.2.2 变量的显式声明 .....57
3.3 数据的基本类型 .....57
3.3.1 数值型 (Numeric) 数据 .....58
3.3.2 字节型 (Byte) 数据 .....59
3.3.3 字符串型 (String) 数据 .....60
3.3.4 日期型 (Date) 数据 .....60
3.3.5 布尔型 (Boolean) 数据 .....61
3.3.6 对象型 (Object) 数据 .....61
3.3.7 变体型 (Variant) 数据 .....62
3.4 运算符与表达式 .....62
3.4.1 算术运算符 .....63
3.4.2 关系运算符 .....63
3.4.3 逻辑运算符 .....64
3.4.4 字符串连接运算符 .....65
3.5 常用内部函数 .....66
3.5.1 数学运算函数 .....66
3.5.2 字符串处理函数 .....67
3.5.3 日期和时间函数 .....68
3.5.4 类型转换函数 .....69

3.5.5 其他函数	70	6.1.2 选择结构	99
3.6 程序基本语句	72	6.1.3 循环结构	99
3.6.1 Let 语句	72	6.2 选择结构程序设计	99
3.6.2 Rem 语句	73	6.2.1 If 语句	99
3.6.3 Load 语句	73	6.2.2 Select Case 语句	104
3.6.4 Unload 语句	73	6.2.3 条件函数	107
3.6.5 Date 语句	73	6.3 循环结构程序设计	108
3.6.6 Time 语句	74	6.3.1 For 型循环语句	108
3.6.7 Stop 语句	74	6.3.2 While 型循环语句	109
3.6.8 End 语句	74	6.3.3 Do 型循环语句	110
3.6.9 Exit 语句	75	6.3.4 goto 型循环语句	112
练习题 3	75	6.3.5 Exit 语句	112
<b>第 4 章 数据的输入输出</b>	<b>79</b>	6.4 循环控制结构的嵌套	114
4.1 数据输入	79	练习题 6	116
4.1.1 InputBox 函数	79	<b>第 7 章 构造数据类型</b>	<b>126</b>
4.1.2 MsgBox 函数	80	7.1 数组	126
4.1.3 MsgBox 语句	81	7.1.1 数组的概念	126
4.2 数据输出	82	7.1.2 静态数组及其声明	127
4.2.1 Print 方法	82	7.1.3 动态数组及其声明	128
4.2.2 与 Print 方法有关的函数	83	7.1.4 数组的基本操作	130
4.2.3 打印机输出	84	7.1.5 控件数组	133
练习题 4	84	7.2 自定义数据类型	134
<b>第 5 章 键盘与鼠标事件</b>	<b>87</b>	7.2.1 自定义数据类型的定义	134
5.1 Visual Basic 的事件驱动程序开发方式	87	7.2.2 变量的定义和使用	135
5.1.1 事件驱动模型	87	7.2.3 自定义类型数组	135
5.1.2 交互式开发	88	7.3 枚举类型	136
5.1.3 事件驱动应用程序的工作方式	88	7.3.1 枚举类型的定义	136
5.2 键盘事件	89	7.3.2 枚举类型的使用	137
5.2.1 KeyPress 事件	89	7.4 集合类型	138
5.2.2 KeyDown 和 KeyUp 事件	89	7.4.1 集合的创建	139
5.3 鼠标事件	90	7.4.2 集合的使用	139
5.4 鼠标指针的形状	92	练习题 7	140
5.5 拖放	93	<b>第 8 章 过程</b>	<b>150</b>
练习题 5	94	8.1 过程概述	150
<b>第 6 章 程序控制结构</b>	<b>98</b>	8.1.1 代码模块的类型	150
6.1 程序控制结构类型	98	8.1.2 过程的类型	152
6.1.1 顺序结构	98	8.2 子程序	153
		8.2.1 子程序的创建	153

8.2.2 子程序的调用	156	11.1.1 多重窗体的建立	202
8.3 事件过程	159	11.1.2 有关窗体的语句和方法	203
8.3.1 事件过程的创建	159	11.1.3 多重窗体的应用	204
8.3.2 事件过程的调用	159	11.2 多文档界面	206
8.4 函数过程	160	11.2.1 多文档界面的建立	206
8.4.1 函数过程与子程序的异同	160	11.2.2 多文档界面的应用	207
8.4.2 函数使用实例	162	11.3 工具栏和状态栏设计	209
8.5 过程中的参数	163	练习题 11	212
8.5.1 实际参数和形式参数	163	<b>第 12 章 文件操作</b>	215
8.5.2 参数传递方式	164	12.1 文件概述	215
8.6 变量的作用域和生存期	165	12.1.1 文件的结构	215
8.6.1 过程的作用域	165	12.1.2 文件的分类	215
8.6.2 变量的作用域	166	12.2 顺序文件的访问	216
8.6.3 变量的生存期	169	12.2.1 顺序文件的打开	216
8.7 过程的嵌套调用和递归调用	171	12.2.2 顺序文件的关闭	216
8.7.1 过程的嵌套调用	171	12.2.3 顺序文件的读操作	216
8.7.2 过程的递归调用	173	12.2.4 顺序文件的写操作	217
练习题 8	173	12.3 随机文件的访问	218
<b>第 9 章 对话框设计</b>	183	12.3.1 随机文件的打开	218
9.1 对话框的类型	183	12.3.2 随机文件的关闭	218
9.2 通用对话框	183	12.3.3 随机文件的读操作	218
9.3 标准对话框界面设计	184	12.3.4 随机文件的写操作	219
9.3.1 “文件”对话框	184	12.3.5 随机文件中记录的修改、增加与删除	219
9.3.2 “颜色”对话框	187	12.4 二进制文件的访问	219
9.3.3 “字体”对话框	188	12.4.1 二进制文件的打开	219
9.3.4 “打印”对话框	189	12.4.2 二进制文件的关闭	219
9.3.5 “帮助”对话框	190	12.4.3 二进制文件的读操作	220
练习题 9	191	12.4.4 二进制文件的写操作	220
<b>第 10 章 菜单设计</b>	193	12.5 文件操作语句和函数的使用	220
10.1 Visual Basic 中的菜单	193	12.6 文件系统控件	223
10.2 菜单编辑器的使用	194	12.7 文件的其他操作	225
10.3 动态菜单的设计	196	练习题 12	225
10.4 弹出式菜单的设计	197	<b>第 13 章 图形的设计与操作</b>	230
练习题 10	198	13.1 图形操作基础	230
<b>第 11 章 多重窗体与多文档界面设计</b>	202	13.1.1 坐标系统	230
11.1 多重窗体	202	13.1.2 使用颜色	232
		13.2 图形控件的使用	234



13.2.1 直线控件	234	14.1.1 数据库	251
13.2.2 形状控件	235	14.1.2 数据库管理系统 (DBMS)	251
13.3 常用绘图方法	236	14.1.3 数据库系统	252
13.3.1 Line 方法	236	14.1.4 关系数据库系统	252
13.3.2 Circle 方法	237	14.2 可视化数据管理器	252
13.3.3 Pset 方法	238	14.2.1 可视化数据管理器的使用	252
13.3.4 Point 方法	239	14.2.2 建立数据库及表	253
13.4 与绘图有关的操作	239	14.2.3 建立数据查询	255
13.4.1 清除画图区	239	14.3 数据控件的使用	255
13.4.2 设置绘图坐标	240	14.3.1 数据控件概述	256
13.4.3 自动重画	240	14.3.2 数据控件的属性	257
13.4.4 指定线宽	240	14.3.3 数据控件的事件	258
13.5 显示图片	241	14.3.4 数据控件的方法	258
13.5.1 直接加载图片到窗体	241	14.3.5 数据控件应用实例	258
13.5.2 使用图片框 (PictureBox) 控件	242	14.4 ADO 数据控件	263
13.5.3 使用图像框 (Image) 控件	242	14.4.1 使用 ADO 数据控件	263
13.5.4 使用滚动条控件	244	14.4.2 使用数据窗体向导	266
13.6 图形控件与动画	246	14.5 SQL	267
13.6.1 移动图形控件	246	14.5.1 SQL 的组成	268
13.6.2 在多幅图片之间切换	248	14.5.2 SQL 的查询功能——SELECT 语句	270
练习题 13	249	14.5.3 在应用程序中使用 SQL 语句	275
<b>第 14 章 Visual Basic 对数据库的 访问</b>	251	14.6 Visual Basic+Access 开发案例	276
14.1 数据库概述	251	练习题 14	281

# 第 1 章

## Visual Basic 可视化编程初步

【学习内容】本章主要介绍 Visual Basic 的启动与退出、Visual Basic 集成开发环境的组成元素以及各个窗口的功能和特点，重点介绍面向对象的基本概念、特点以及可视化编程的基本方法。

通过本章的学习，要求熟悉 Visual Basic 集成开发环境及其基本操作，熟悉应用程序主窗口的组成和各菜单的主要功能，并对窗体设计器窗口、工具箱窗口、属性窗口、代码窗口、工程资源管理器窗口、立即窗口、本地窗口、监视窗口、窗体布局窗口的布局、作用和打开方式有所了解。

窗体和控件是 Visual Basic 重要的对象，窗体对象是构成一个应用程序的最基本部分，是用户与应用程序之间进行人机对话的界面。控件是具有一定外形特征和操作特点的界面组成元素，分为标准控件、ActiveX 控件和可插入控件 3 类。读者要结合以往在图形操作环境下对话框中各组成元素的操作，对标准控件的分类和组成以及各控件的作用和操作特点进行初步了解。

最后通过介绍一个完整的工程设计实现过程，让读者对 Visual Basic 程序编制的基本步骤有所认识，明确基本设计过程由设计界面、设置对象的属性以及编写代码 3 个环节组成，然后再对程序进行运行、保存、生成可执行文件的后期处理。此外，结合本章简单程序的设计过程，理解对象以及相关概念，便于以后深入学习。

本教材主要介绍 Visual Basic 6.0，在后续文中均称为 Visual Basic。

### 1.1 Visual Basic 的基本操作

Visual Basic 简称 VB，Visual 的含义是“可视的”，Visual Basic 是在 BASIC 语言的基础上发展而来的，是可视化编程技术与 BASIC 语言相结合的产物。Visual Basic 沿用了 BASIC 语言中的一些语法，而且具有可视化、面向对象、事件驱动编程机制等特点，是目前应用最广泛的可视化编程语言之一，也是被公认为编程效率最高的一种编程工具。无论是开发功能强大且性能可靠的商务软件，还是用于编写处理实际问题的小程序，Visual Basic 都显示出其快速和简便的特点，对于初学者开发一般应用程序尤为适合。

Visual Basic 具有以下特点。

(1) 可视化的开发环境。Visual Basic 提供了可视化的设计工具，编程人员只需按设计要求进行界面布局、运行环节设置并编写功能代码，由系统自动生成界面设计代码，从而提高了程序设计的效率。相比由用户自己编程设计界面的传统编程方式，具有简便快捷、所见即所得和编程效

率高等特点。

(2) 面向对象程序设计。Visual Basic 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,并赋予每个对象相关方面的属性,每个对象以图形的方式出现在界面上,形象直观且减少了程序员编写界面代码的繁琐。

(3) 结构化的程序设计语言。Visual Basic 提供的编程语言具有高级程序设计语言的优点,接近于人们自然的逻辑思维方式。语句简单,容易掌握,具有多种结构控制语句,同时具有很强的纠错能力,拥有功能强大的调试器和编译器。

(4) 事件驱动编程机制。Visual Basic 通过事件驱动来执行对象的操作,一个对象可能产生多个事件,每个事件都通过执行一段程序来响应。因此,在设计大型系统时,在界面设计和功能设计的基础上,能够不断化整为零,通过编写若干面向不同对象和不同事件的小程序,相互配合完成任务,从而简化程序员的工作难度和强度。

(5) 强大的第三方支持。目前,有很多应用程序提供了与 Visual Basic 的接口,Visual Basic 可以直接控制第三方应用程序,例如 Microsoft 的 Word、Excel 和 AutoCAD 等。

(6) 开发数据库程序。Visual Basic 可通过 ODBC (Open DataBase Connectivity, 开放数据库连接性) 访问包括 Microsoft Access、Microsoft SQL Server 及 Oracle 等大型数据库在内的各种数据库。Visual Basic 提供了大量支持数据库功能的控件,利用这些控件可以开发出功能强大的数据库应用程序。

(7) 良好的中文支持。Visual Basic 不但全面支持汉字的输入和显示等,还支持中文变量和双字节字符,具有良好的兼容性。

### 1.1.1 启动 Visual Basic

启动 Visual Basic 的方法有以下 4 种。

(1) 单击 Windows 任务栏的“开始”按钮,选择“程序”,单击“Microsoft Visual Basic 6.0 中文版”命令,如图 1-1 所示。

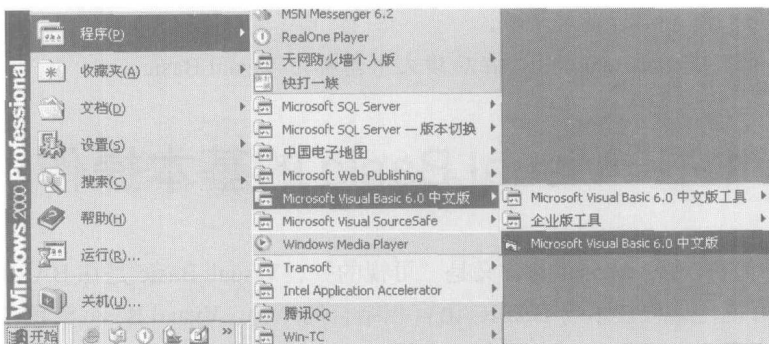


图 1-1 通过“开始”按钮启动 Visual Basic 6.0

(2) 通过“我的电脑”或“资源管理器”,找到 Visual Basic 所在的路径和可执行文件 VB6.EXE,然后双击该文件。

(3) 单击 Windows 任务栏上的“开始”按钮,单击“运行”命令,打开“运行”对话框,如图 1-2 所示。在“打开”文本框内输入可执行文件 VB6.EXE 所在的路径

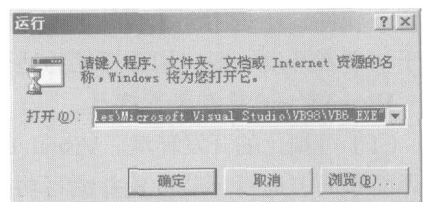


图 1-2 通过“运行”命令启动 Visual Basic 6.0

和文件名，然后单击“确定”按钮，即可运行该程序。也可以通过单击“浏览”按钮，打开“浏览”对话框，然后查找和选择可执行文件 VB6.EXE，确定文件名和所在路径。

(4) 双击桌面上的 Visual Basic 快捷方式图标来快速启动 Visual Basic。

进入 Visual Basic 集成环境后，屏幕上出现“新建工程”对话框，提示用户建立或打开何种类型的文件，如图 1-3 所示。初学者可选择默认的“标准.EXE”，单击“打开”按钮，即可创建一个工程，并打开一个窗体设计器窗口。

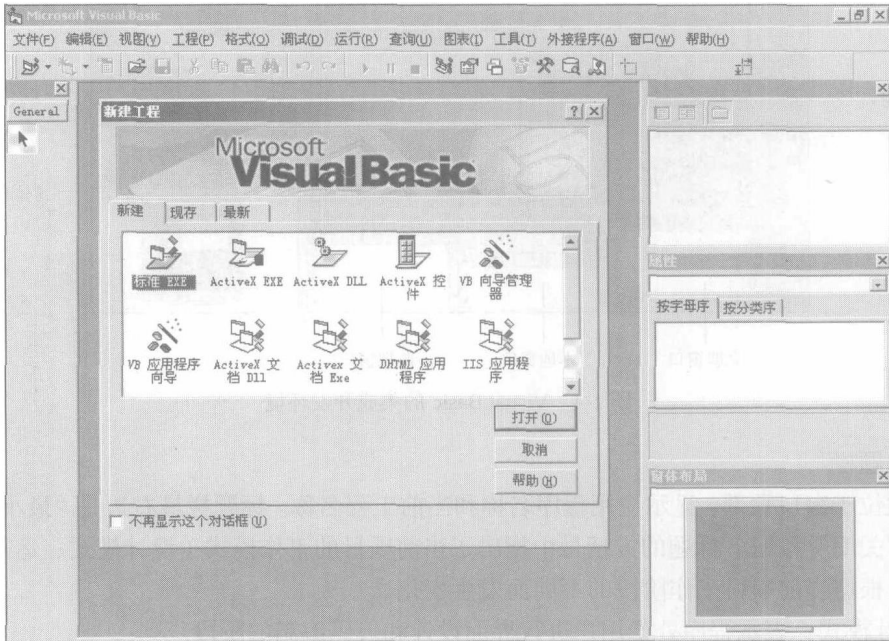


图 1-3 Visual Basic 主界面

## 1.1.2 退出 Visual Basic

退出 Visual Basic 有以下 4 种方法。

- (1) 单击 Visual Basic 程序窗口右上角的“关闭”按钮。
- (2) 打开“文件”菜单，单击“退出”命令。
- (3) 单击 Visual Basic 程序窗口左上角的控制菜单图标，打开快捷菜单，单击“关闭”命令。
- (4) 在桌面下方的任务栏中选 Visual Basic 的任务按钮，单击鼠标右键，打开快捷菜单，单击“关闭”命令。

## 1.1.3 Visual Basic 的集成开发环境

启动 Visual Basic 后，屏幕上出现的窗口有：应用程序主窗口、窗体设计器窗口、工程资源管理器窗口、工具箱窗口、属性窗口、代码窗口、窗体布局窗口、立即窗口、本地窗口和监视窗口等。应用程序主窗口是 Visual Basic 的背景窗口，其余的窗口都包含在主窗口内，各窗口的分布如图 1-4 所示。

### 1. 应用程序主窗口

Visual Basic 的应用程序主窗口位于集成环境的顶部，是用户设计应用程序的界面，主要包括

常规的标题栏、菜单栏和工具栏。

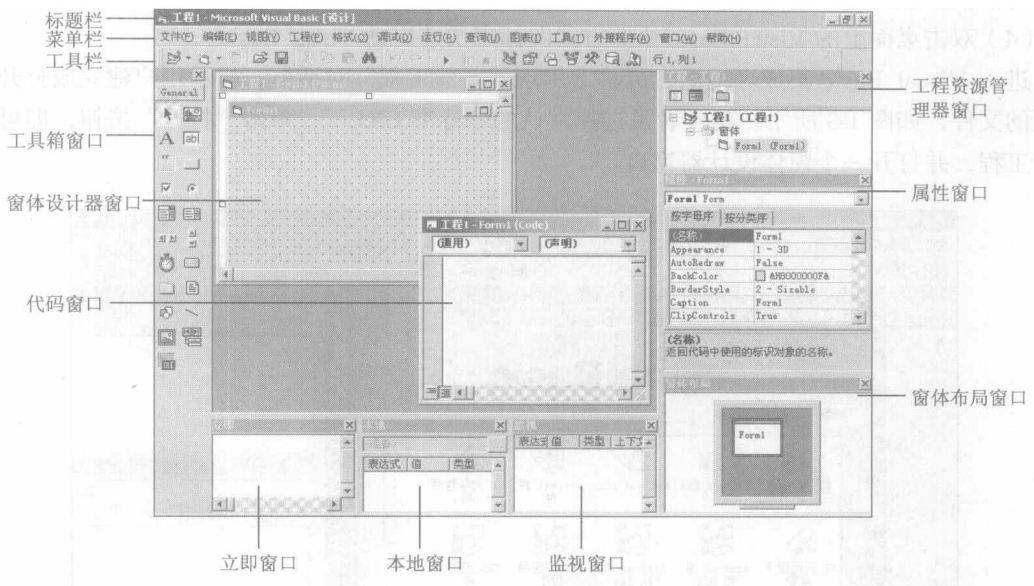


图 1-4 Visual Basic 的集成开发环境

### (1) 标题栏

标题栏位于窗口顶部，显示应用程序名称和当前工程名称。标题栏最右边是“最小化”、“最大化”和“关闭”按钮。标题的方括号中指出了当前项目的工作模式（设计模式、运行模式或中断模式），根据当前项目工作阶段的不同而发生变化。

① 设计模式：表示程序处于用户进行界面设计和代码编制的阶段。

② 运行模式：表示程序处于运行的阶段。在此阶段，不能编辑和修改代码和窗体界面。

③ 中断模式：表示程序处于运行过程中暂时中断的阶段。在此阶段，可以进行代码的修改，但不能修改窗体界面。编辑修改完成后，单击“继续”按钮，继续执行程序。如果要退出程序的执行，则单击“结束”按钮。

### (2) 菜单栏

菜单栏位于标题栏下面，是启动菜单命令的入口，Visual Basic 的菜单栏共有 13 类主菜单，包括文件、编辑、视图、工程、格式、调试、运行、查询、图表、工具、外接程序、窗口和帮助菜单，其对应功能如表 1-1 所示。

表 1-1 Visual Basic 主菜单功能介绍

菜单项	功能介绍
文件	用于工程的创建、打开、保存和生成可执行文件等操作
编辑	用于程序代码的编辑操作
视图	用于各种窗口的打开、查看和编辑等操作
工程	用于对控件、窗体和模块等对象的处理操作
格式	用于对窗体和控件在格式化方面的操作
调试	用于程序执行过程中的调试并检查错误等操作
运行	用于程序执行的启动、中断和停止等操作

续表

菜单项	功能介绍
查询	用于设计数据库应用程序时设置 SQL 属性
图表	用于设计数据库应用程序时编辑数据库的操作
工具	用于集成开发环境下工具的扩充和菜单的编辑等操作
外接程序	用于增加或删除外接程序操作
窗口	用于对窗口排列方式的设置操作
帮助	用于提示 Visual Basic 的使用方法, 帮助用户学习和使用

每一项主菜单包含有若干个菜单命令, 以多级下拉菜单的方式展开。如果某个菜单命令后面带有省略号, 则在执行该命令时, 将会打开一个对话框; 如果某个菜单命令后面带有热键信息, 表示除了可以用鼠标启动执行该命令之外, 还可以通过键盘上的组合键直接启动该命令, 而不必打开菜单。

打开菜单并执行菜单命令的方式有如下多种。

① 按 F10 或 Alt 键激活菜单栏, 键入菜单项后的字母, 打开相应下拉菜单, 再键入要执行菜单命令后的字母执行。

② 按 F10 或 Alt 键激活菜单栏后, 用上、下、左、右方向键选中菜单项中的菜单命令, 按回车键执行。

③ 用鼠标单击菜单项, 打开菜单后用鼠标控制光标移到相应命令, 单击后执行。

### (3) 工具栏

工具栏中以快捷图标的形式提供了菜单中所包含的部分常用命令, 单击快捷图标, 即可快速执行相应的命令。Visual Basic 提供了编辑、标准、窗体编辑器和调试 4 类工具栏, 通常只显示标准工具栏。通过执行“视图”菜单中的“工具栏”命令可打开其他工具栏。每种工具栏都有固定和浮动两种形式。固定式工具栏位于菜单栏的下方, 浮动式工具栏可以用鼠标选中并在屏幕上移动。两种形式可以相互转换: 用鼠标选中固定式工具栏并拖动, 可变成浮动形式; 在浮动式工具栏的标题处双击鼠标, 可变成固定形式。

### 2. 窗体设计器窗口

窗体设计器窗口 (见图 1-5) 又称对象窗口或窗体 (Form), 是用户设计的应用程序界面, 也是对应于应用程序运行结果的界面。Visual Basic 中的可编程对象有 3 种: 窗体、控件和外部对象。其中, 窗体对象是构成一个应用程序最基本的部分, 是用户与应用程序之间进行人机对话的界面。用户在窗体中可以创建各种控件, 并通过修改控件的属性值来改变该控件在窗体上的显示风格。在窗体的左上角是窗体的标题。激活对象窗口的方法有两种: 按下快捷键 Shift+F7, 或从“视图”菜单中单击“对象窗口”命令。

### 3. 工程资源管理器窗口

工程资源管理器窗口主要用于显示用户工程的层次以及工程中的所有文件, 包括窗体、模块、类别模块、用户控件、用户文档、属性页、ActiveX 设计器、相关文档和资源等。工程资源管理器窗口中的文件分为 6 类: 窗体文件 (.frm)、程序模块文件 (.bas)、类模块文件 (.cls)、工程文件 (.vbp)、工程组文件 (.vbg) 和资源文件 (.res)。

在工程资源管理器窗口的上方有 3 个图形按钮, 分别是“查看代码”、“查看对象”和“切换文件夹”, 如图 1-6 所示。单击“查看代码”按钮, 打开所选对象的代码窗口; 单击“查看对象”按钮, 显示对象窗口; 单击“切换文件夹”按钮, 则可以隐藏或显示包含在对象文件夹中的个别项目。

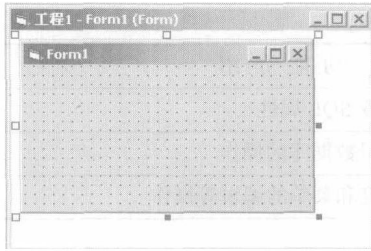


图 1-5 窗体设计器窗口

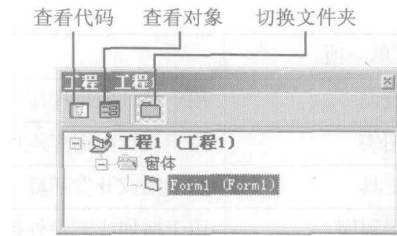


图 1-6 工程资源管理器窗口

在工程资源管理器窗口中单击鼠标右键，在弹出的快捷菜单中选择工程属性，则弹出一个对话框，包括当前工程的各种信息，例如工程名称、启动对象名、工程的版本信息、编译条件等。按快捷键 Ctrl+R，或在“视图”菜单中选择“工程资源管理器”命令，都可以打开该窗口。初学者一般使用以下两个文件。

#### (1) 窗体文件 (.frm)

用于存储窗体上所有对象及其属性、事件过程和程序代码。一个应用程序至少包含一个窗体文件。

#### (2) 程序模块文件 (.bas)

用于所有模块级变量和用户自定义类型的声明以及模块级过程的定义。在此声明的变量和类型，可以被同一模块中的所有过程使用。

### 4. 工具箱窗口

工具箱窗口由控件图标组成，一般位于应用程序窗口的左侧，主要用于应用程序的界面设计。通过把工具箱窗口中的控件拖动到窗体中建立程序界面，然后编写功能代码。其中的工具分为两类：内部控件（标准控件）和 ActiveX 控件。标准控件工具箱如图 1-7 所示。

### 5. 属性窗口

用户可以利用属性窗口修改可编程对象的属性值，操作方法是：先选定要修改的对象（可以在对象窗口中选定对象，也可以在属性窗口上方的对象下拉列表框中进行选择），然后打开属性窗口，找到要修改的属性，输入新的属性值。属性窗口中有两个选项卡：“按字母序”和“按分类序”。根据需要选择对象后，对象的属性就会按照新的顺序显示在属性窗口中。

在 Visual Basic 中，可以方便地同时修改多个对象的属性。操作方法是：选择要修改的多个对象（一般是同一类对象），这时在属性列表框中会列出这些对象所共有的属性。如果属性值为空，表明这些对象的该属性不完全相同，用户可以重新对其赋予相同的属性值。按快捷键 F4 或在“视图”菜单中选择“属性窗口”命令，可以打开属性窗口，如图 1-8 所示。

### 6. 代码窗口

代码窗口（见图 1-9）是用户编写程序的窗口，用户可以在此编写自定义的函数、过程、子程序等。代码窗口分为 3 部分：左上方的对象框是当前应用程序中的对象；右上方的过程框是所选对象的事件过程；位于两个下拉列表框下面的代码编辑区是用户编写代码的区域。如果在对象框中显示的是“通用”，过程框中则会列出所有声明，以及为此窗体所创建的常规过程。如果正在编辑模块中的代码，则过程框列出所有模块中的常规过程。

打开代码窗口有以下方法。

- (1) 在对象窗口中双击控件或窗体。
- (2) 选择“视图”菜单中的“代码窗口”命令。
- (3) 在工程资源管理器窗口中，选择一个窗体或模块，单击“查看代码”按钮。

#### (4) 按快捷键 F7。

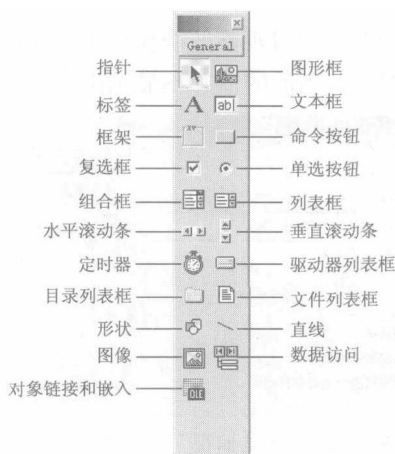


图 1-7 工具箱窗口中的标准控件

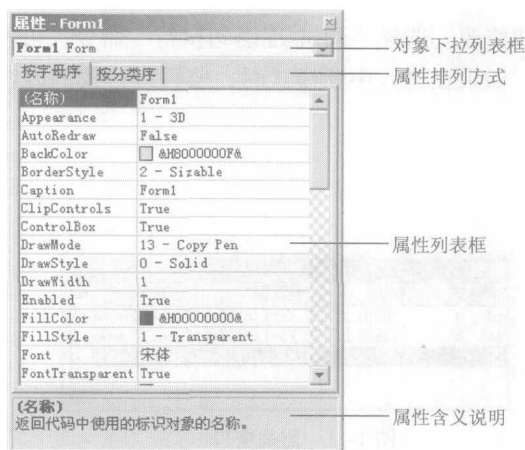


图 1-8 属性窗口

### 7. 窗体布局窗口

在设计时，可以利用窗体布局窗口来可视化地指定一个窗体运行时在屏幕中的位置，也可以通过改变属性窗口中窗体的上边距“TOP”和左边界距“LEFT”属性达到同样的目的，但显然不如利用窗体布局窗口直观。

例如，在窗体布局窗口中缩小了的窗体上单击鼠标右键，在弹出的快捷菜单中选择“启动位置”命令，再选择“屏幕中心”选项，于是窗体运行时就会显示在屏幕的正中央。可以在“视图”菜单中选择“窗体布局窗口”命令来打开窗体布局窗口，如图 1-10 所示。

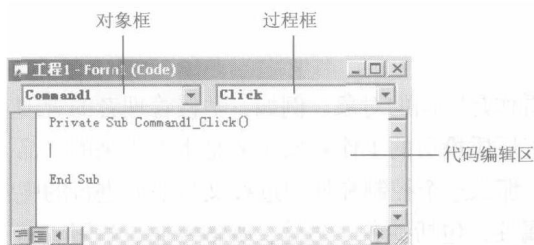


图 1-9 代码窗口

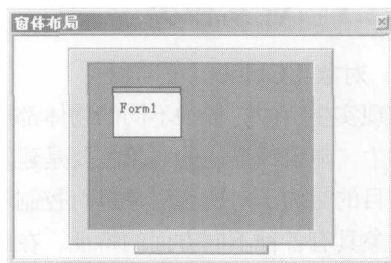


图 1-10 窗体布局窗口

### 8. 立即窗口

在中断模式下立即窗口可以输出运行的结果或查询对象的值。中断模式指的是在程序运行过程中，用户按下快捷键 I+Break 键，或执行菜单“运行”中的“中断”命令，使运行程序处于中断状态。另外，还可以在立即窗口中输入一行代码，然后按回车键来执行该行代码。立即窗口中的代码是不能存储的。按快捷键 Ctrl+G，或在“视图”菜单中选择“立即窗口”命令，可以打开立即窗口。

### 9. 本地窗口

本地窗口可以自动显示出所有在当前过程中的变量声明及变量值。在“视图”菜单中选择“本地窗口”命令，即可打开本地窗口。

### 10. 监视窗口

在“视图”菜单中选择“监视窗口”命令可以打开监视窗口，如图 1-11 所示。监视窗口



中可以自动显示预先定义的监视表达式。在监视窗口中，单击鼠标右键，然后选择“编辑”、“添加”或“删除监视”命令，打开如图 1-12 所示的“添加监视”对话框，可以对监视的表达式进行定义和修改。选择“调试/逐语句执行”命令，可逐条执行命令，同时观察监视窗口中变量的变化情况。如果在进入中断模式时，监视表达式的内容不在范围内，当前的值并不显示出来。

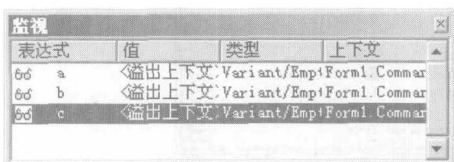


图 1-11 监视窗口

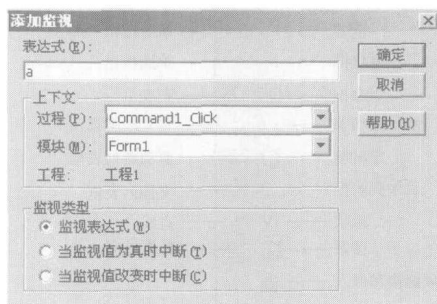


图 1-12 “添加监视”对话框

## 1.2 对象的基本知识

对象就是程序可以控制的运行实体，程序依靠对象的特定动作来完成特定的任务。对象具有属性、事件和方法 3 方面的特征。属性是指通过数据描述的对象的相关特征；事件是指能被对象识别的动作以及由动作引发的结果，需要通过编制代码来设定并执行；方法是指对象要执行的某些特定动作，也是通过编程执行来实现的。

### 1.2.1 对象和类

#### 1. 对象 (Object)

在现实生活中，各种不同的物体都可以被看作是不同的对象。例如，学校管理者的管理对象主要是学生，建筑师描述的对象主要是建筑物，工厂质检员的工作对象主要是生产出来的产品。选择对象的目的是为了对所选对象进行控制和处理，那么这个控制和处理过程又是如何进行的呢？比如学生对象具有各种不同方面的特征，在此称为属性，包括姓名、学号、所学专业、学习成绩、兴趣爱好和基本生活需求等。学校管理者要对学生有序管理，就要从学生的一些普遍共性和不同个性出发，从学习和生活中的不同方面入手实施管理：针对学生的学习，主要由教务处来安排学习课程的内容、进度和时间；针对学生的基本生活需求，主要由后勤部门安排住宿、就餐、洗衣等事项；针对学生的阅读需要，由图书馆进行图书的管理并制定借阅规则供学生阅读图书。由此可见，针对对象的处理，是根据对象的特征和需求制定相应的控制过程。这样的原理同样适用于编程过程中的对象设计。在面向对象的程序设计中，对象的特征被称为属性，针对对象的控制过程和相关操作被称为事件和方法。属性、事件和方法是描述对象的 3 个基本特征。建立一个对象后，可以通过对该对象的属性、事件和方法进行描述，再把对象和对象的属性值以及描述对象行为方式的代码结合成为一个整体，完成对其性质和功能的描述，最终使多个对象组合起来相互配合，实现程序预期的功能。

在面向对象的编程过程中，程序员需要按界面设计要求进行屏幕布局，把若干控件对象拖入到设计窗体中，由系统自动生成界面设计代码。然后，程序员根据功能要求，针对相应的控件对象编写功能代码，这样通过编写若干面向不同对象和不同事件的小程序，使之相互配合完成设计