

高职高专计算机系列规划教材



可视化程序设计 — Visual Basic

周志德 刘德强 许敏 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

高职高专计算机系列规划教材

可视化程序设计——Visual Basic

周志德 刘德强 许 敏 编著

电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书共 11 章。前 2 章为 Visual Basic 概述、窗体与基本控件。第 3、4 章讲述了 Visual Basic 语言与菜单、工具栏、状态栏、对话框。第 5、6 章讨论了 Visual Basic 数据库开发环境与数据库应用程序设计。第 7、8 章讲解了数据报表打印程序设计与 Visual Basic 高级控件。第 9、10 章讲述了文件管理、多媒体技术与图形处理，最后一章布置了 Visual Basic 课程设计。全书以学生档案管理系统作为项目案例，采用边讲知识边设计模块程序的项目式教学法，当课程讲完时学生档案管理系统也设计完成。最后再通过学生成绩管理系统的课程设计，使学生能巩固所学知识，同时也能学到程序项目开发的知识。

本书可以作为高职高专学院计算机、管理、电子等专业的教材。本书起点低，可作为可视化程序设计的入门教材来学习。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目 (CIP) 数据

可视化程序设计: Visual Basic/周志德, 刘德强, 许敏编著. —北京: 电子工业出版社, 2006.10

(高职高专计算机系列规划教材)

ISBN 7-121-03233-3

I. 可… II. ①周… ②刘…③许… III. BASIC 语言—程序设计—高等学校: 技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 115093 号

责任编辑: 吕 迈 (lumai@phei.com.cn)

印 刷: 北京季蜂印刷有限公司

装 订: 三河市鹏成印业有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1 092 1/16 印张: 24.25 字数: 621 千字

印 次: 2006 年 10 月第 1 次印刷

印 数: 4 000 册 定价: 31.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系电话: (010) 68279077; 邮购电话: (010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

Visual Basic 是目前许多软件公司开发信息管理等系统所使用的重要开发工具,是计算机专业学生应掌握的一门计算机软件课程。通过这门课程的学习,能使读者了解 Visual Basic 的开发环境、工程、面向对象、事件驱动程序设计的概念,掌握 Visual Basic 语言的语法、程序结构、编程方法,掌握 Visual Basic 基本控件的属性、方法、事件,以及将控件加入到窗体构成用户程序界面的方法,掌握使用 Data 控件、ADO Data 控件、ADO 对象及其绑定控件设计一个信息管理系统的方法,能按系统分析员的要求,用 Visual Basic 设计窗体界面、编写程序代码、进行程序运行与调试,为最终成为一名应用程序程序员打下必要的基础。

本教材有如下特点:

1. 本书以学生档案管理系统作为项目案例,采用边讲知识点边设计模块程序的项目式教学法,当课程讲完时学生档案管理系统也设计完成,最后再通过学生成绩管理系统的课程设计,使学生能巩固所学知识,同时也能学到程序项目开发的知识。

2. 通过 C++中有关类与对象的例题引出类、对象、属性、方法的概念,进一步引出可视化程序设计、事件驱动的编程机制等概念,使读者对上述概念有一个直观的理解。

3. 本教材以“学生档案管理系统”程序作为主要例题与实验题编写。在这个管理信息系统中融入了大量编者开发程序的经验和体会,希望能通过这本实用性很强的教材,使读者能很快掌握用 Visual Basic 开发应用程序的基本方法。

4. 本教材根据信息管理系统编程需要,将常用的控件加以重点介绍,并用实例说明控件的属性、方法与事件及具体的使用方法,避免将教材编成一本 Visual Basic 的说明书(只列出所有各类控件的属性、方法与事件)。

5. 所有应用程序的界面设计均采用商业化应用程序的风格,在课堂教学中就开始培养学生掌握编写商业化应用程序的设计方法和思路。

6. 每章后有小结,并配有一定量的习题与实验题,便于教师教学和学生自学。各章内容充实,安排合理,衔接自然。

7. 本书最后一章通过“学生成绩管理系统”课程设计,比较详细地介绍了开发一个 MIS 管理系统的主要过程,并将本书介绍的许多实用 Visual Basic 控件开发应用技术融合于其中。例如,用 MDI 多文档技术设计主、子窗体;用 Combo 控件进行部门的选择;用 ADO 对象与 TreeView 控件实现树形结构的查询选择;用 MSChart 控件以直方图、曲线图与饼图形式显示各类统计图表;用 ADO 对象与 MSFlexGrid 控件实现数据表的录入与保存工作。若读者能独立完成整个“学生成绩管理系统”的设计工作,则不但能初步掌握 MIS 系统的开发方法,还能掌握 Visual Basic 的许多实用数据库开发技术,并运行这些技术去开发其他的信息管理系统程序,达到学完这门课程后就能用该课程的知识从事程序设计的目的。这也是编写此书的初衷之一。

8. 本教材将“学生档案管理”与“学生成绩管理”两个信息管理系统程序放在电子工

业出版社的华信教育资源网上 (www.hxedu.com.cn), 供教师与学生在教学中参考。

在本书编写过程中, 参考了目前国内比较优秀的有关 Visual Basic 程序设计方面的书籍资料, 在此谨向有关作者表示感谢。对电子工业出版社吕迈老师在编写工作中给予的大力支持表示衷心的感谢。

本书第 1、2、3 章由刘德强编写, 第 4、9、10 章由许敏编写, 第 5、6、7、8、11 章由周志德编写, 全书由无锡职业技术学院周志德副教授统编, 南京邮电大学唐瑞庭副教授审阅。在全书编写过程, 陈天娥、李萍、颜惠琴、王得燕老师对书中内容提出了非常宝贵意见, 在此, 谨对上述老师的工作表示感谢。

本书若有错误及不足之处, 恳请读者给予指正。作者信箱 zzd@wxzyxgx.sina.net。

编者于 2006 年 8 月 10 日

目 录

第 1 章 Visual Basic 概述	1
1.1 Visual Basic 简介	1
1.2 面向对象的程序设计方法	1
1.3 可视化的程序设计方法	4
1.4 事件驱动的编程机制	4
1.5 VB 的集成开发环境	5
1.5.1 标题栏	6
1.5.2 菜单栏	6
1.5.3 工具栏	6
1.5.4 控件工具箱	6
1.5.5 窗体设计器	7
1.5.6 属性设置窗口	7
1.5.7 代码编辑器	7
1.5.8 工程资源管理器	8
1.5.9 窗体布局窗口	9
1.6 VB 的工程管理	9
1.6.1 工程文件 (.vbp)	9
1.6.2 窗体文件 (.frm)	9
1.6.3 其他文件	9
1.6.4 创建 VB 应用程序的步骤	10
1.7 创建一个简单的 VB 应用程序	11
本章小结	13
习题 1	14
实验 1	15
第 2 章 窗体与基本控件	16
2.1 窗体	16
2.1.1 窗体概述	16
2.1.2 窗体的属性	17
2.1.3 窗体的方法	21
2.1.4 窗体的事件	23
2.1.5 窗体的装载和卸载	25
2.2 VB 的常用控件	25
2.2.1 标签控件 (Label)	25
2.2.2 文本框控件 (TextBox)	26
2.2.3 命令按钮控件 (CommandButton)	29

2.2.4	列表框控件 (ListBox)	30
2.2.5	组合框控件 (ComboBox)	32
2.2.6	图像框控件 (Image)	33
2.2.7	图片框控件 (PictureBox)	34
2.2.8	单选按钮、复选框和框架控件	34
2.2.9	滚动条控件 (HScrollBar 和 VScrollBar)	36
2.2.10	计时器控件 (Timer)	39
2.3	控件布局	40
2.3.1	调整控件对象的位置和大小	40
2.3.2	控件对象的对齐	41
2.3.3	控件对象的间距调整	41
2.4	程序界面设计示例	41
	本章小结	45
	习题 2	46
	实验 2	47
第 3 章	Visual Basic 语言	49
3.1	Visual Basic 程序组成	49
3.1.1	模块	50
3.1.2	过程	50
3.1.3	代码行的书写规则	51
3.2	VB 数据类型与表达式	52
3.2.1	数据类型	52
3.2.2	常量	55
3.2.3	变量	56
3.2.4	运算符与表达式	58
3.2.5	VB 公共函数	61
3.3	程序结构与流程控制语句	64
3.3.1	程序的三种基本结构	64
3.3.2	赋值语句及相关函数	65
3.3.3	分支语句	69
3.3.4	循环语句	74
3.3.5	综合应用举例	78
3.4	程序调试	83
3.4.1	程序错误类型	83
3.4.2	VB 调试工具	85
3.4.3	程序调试	86
3.5	数组	89
3.5.1	一维数组定义与引用	90
3.5.2	二维数组的定义与引用	93
3.5.3	动态数组的定义与引用	96

3.5.4	控件数组的定义与引用	97
3.5.5	数组元素的输入与输出	100
3.5.6	数组函数及数组语句	101
3.6	过程与函数	104
3.6.1	事件过程的定义与调用	104
3.6.2	通用过程的定义与调用	106
3.6.3	函数过程的定义与调用	109
3.6.4	数组与对象作为参数	111
3.6.5	调用其他模块中的公有过程和函数	114
3.6.6	静态变量	115
本章小结		116
习题 3		118
实验 3		121
第 4 章	菜单、工具栏、状态栏与对话框	123
4.1	菜单设计	123
4.1.1	下拉式菜单设计	123
4.1.2	弹出式菜单设计	128
4.2	工具栏设计	130
4.2.1	ImageList 控件	130
4.2.2	ToolBar 控件	131
4.2.3	用 ToolBar 与 ImageList 设计工具栏	133
4.3	状态栏设计	137
4.3.1	StatusBar 控件的概述	137
4.3.2	StatusBar 控件的属性	137
4.4	通用对话框与单文档编辑器设计	141
4.4.1	通用对话框设计	141
4.4.2	CommonDialog 控件	142
4.4.3	单文档编辑器设计	144
4.5	MDI 多窗体程序设计	146
4.5.1	创建 MDI 主窗体	147
4.5.2	建立 MDI 子窗体	147
4.5.3	MDI 窗口菜单设计	147
4.6	综合应用（学生档案管理系统设计）	149
4.6.1	学生档案管理系统 MDI 主窗体与子窗体设计	149
4.6.2	学生档案管理系统 MDI 主菜单设计	150
4.6.3	学生档案管理系统主窗体中工具栏设计	151
4.6.4	学生档案管理系统主窗体中状态栏设计	151
本章小结		152
习题 4		153
实验 4		154

第 5 章 Visual Basic 数据库开发环境	155
5.1 数据库基本知识	155
5.1.1 数据库系统的基本概念	155
5.1.2 关系型数据库的基本概念	157
5.2 数据库应用程序的开发环境	158
5.3 可视化数据管理器	159
5.3.1 建立数据库与数据表	159
5.3.2 打开与编辑数据表	162
5.3.3 数据表导入/导出	163
5.3.4 查询生成器	164
5.4 结构化查询语言 (SQL) 简介	165
5.4.1 Select 语句	165
5.4.2 Insert 语句	173
5.4.3 Update 语句	174
5.4.4 Delete 语句	174
5.5 数据窗体设计器	175
本章小结	177
习题 5	178
实验 5	179
第 6 章 数据库应用程序设计	180
6.1 数据库应用程序的结构与设计步骤	180
6.1.1 用 Data 数据控件设计数据库应用程序示例	180
6.1.2 数据库应用程序的结构	181
6.1.3 数据库应用程序的设计步骤	183
6.2 用 Data 数据源控件设计数据库应用程序	183
6.2.1 Data 数据源控件	183
6.2.2 TextBox 控件	188
6.2.3 Label 控件	189
6.2.4 DBGrid 控件	190
6.2.5 DBCombo 控件	193
6.2.6 DBList 控件	193
6.2.7 设计班级编码维护程序	194
6.3 用 ADO Data 数据源控件设计数据库应用程序	197
6.3.1 ADO Data 数据源控件	197
6.3.2 DataGrid 控件	201
6.3.3 DataCombo 控件	203
6.3.4 DataList 控件	203
6.3.5 用 ADO Data 控件设计班级代码表维护程序	204
6.3.6 SQL 语句字符串中变量的表达式	207
6.3.7 用 ADO Data 控件设计学生档案录入程序	208

6.3.8 用 ADO Data 控件设计学生档案查询程序	213
6.4 用 ADO 对象设计数据库应用程序	218
6.4.1 ADO 的基本概念	218
6.4.2 ADO 对象	219
6.4.3 用连接对象打开数据库	220
6.4.4 用记录集对象操作数据表	222
6.4.5 用命令对象操作数据表	228
本章小结	232
习题 6	233
实验 6	234
第 7 章 数据报表打印程序设计	238
7.1 数据环境设计器	238
7.1.1 数据环境设计器的概念与使用	238
7.1.2 用数据环境设计器设计数据库应用程序	243
7.2 数据报表设计器	247
7.2.1 数据报表对象	247
7.2.2 数据报表控件	249
7.2.3 数据报表程序设计	250
本章小结	257
习题 7	258
实验 7	258
第 8 章 Visaul Basic 高级控件	261
8.1 日期类控件与函数	261
8.1.1 MonthView 控件	261
8.1.2 Calendar 控件	262
8.1.3 日期类函数	262
8.2 字符表格控件 MSFlexGrid	263
8.2.1 MSFlexGrid 控件概述	263
8.2.2 MSFlexGrid 控件的主要属性	264
8.2.3 MSFlexGrid 控件的事件与方法	266
8.2.4 校历管理程序设计	271
8.2.5 学生宿舍录入程序设计	276
8.3 树形结构控件 TreeView	280
8.3.1 TreeView 控件概述	280
8.3.2 TreeView 控件的属性	280
8.3.3 TreeView 控件的方法	282
8.3.4 用树形控件 TreeView 设计学生档案查询程序	283
8.4 分页控件 SSTab	286
8.4.1 分页控件 SSTab 概述	286
8.4.2 分页控件 SSTab 的属性	286

8.4.3 用分页控件 SSTab 设计学生档案查询程序	287
8.5 进程条控件 ProgressBar	289
8.5.1 进程条控件 ProgressBar 概述	289
8.5.2 进程条控件 ProgressBar 的属性	289
8.5.3 用 ProgressBar 设计查询进度界面	290
8.6 数据图表控件 MSChart	291
8.6.1 MSChart 控件概述	292
8.6.2 MSChart 控件的属性	292
8.6.3 MSChart 控件的属性对话框	297
8.6.4 用 MSChart 控件设计学生档案统计程序	298
本章小结	302
习题 8	303
实验 8	303
第 9 章 文件系统	304
9.1 文件系统控件	304
9.1.1 驱动器列表框控件 (DriveListBox)	304
9.1.2 目录列表框控件 (DirListBox)	305
9.1.3 文件列表框控件 (FileListBox)	306
9.1.4 用文件系统控件设计文本浏览器	307
9.2 文件操作	309
9.2.1 文件概述	309
9.2.2 顺序文件	310
9.2.3 随机文件	315
9.2.4 二进制文件	317
本章小结	319
习题 9	320
实验 9	320
第 10 章 多媒体技术与图形处理	322
10.1 多媒体技术	322
10.1.1 多媒体技术的基本术语	322
10.1.2 多媒体控件 (MMControl)	323
10.1.3 动画控件 (Animation)	325
10.1.4 在学生档案管理系统中添加媒体播放器	327
10.2 图形处理技术	330
10.2.1 坐标系	330
10.2.2 色彩函数	332
10.2.3 绘图控件	333
10.2.4 绘图方法	337
10.2.5 图片框 (PictureBox)	343
本章小结	344

习题 10	345
实验 10	345
第 11 章 Visual Basic 课程设计	346
11.1 系统需求与功能分析	346
11.2 系统功能模块设计	347
11.3 系统数据流程图与数据结构设计	347
11.3.1 系统数据流程图	347
11.3.2 数据库设计	348
11.3.3 编码表的部分字段内容	350
11.3.4 建立学生成绩管理系统目录与数据表结构	351
11.4 各功能模块程序设计	351
11.4.1 建立学生成绩管理系统工程文件与主窗体文件	351
11.4.2 学生成绩初始化窗体程序设计	352
11.4.3 学生成绩录入窗体程序设计	354
11.4.4 学生班级成绩查询窗体程序设计	354
11.4.5 学生个人成绩查询窗体程序设计	355
11.4.6 学生成绩统计窗体程序设计	356
11.4.7 课程代码窗体程序设计	356
11.4.8 各类数据编码表维护窗体程序设计	356
11.4.9 班级课程表录入窗体程序设计	358
11.4.10 课程表查询窗体程序设计	359
11.4.11 学生补考成绩录入程序设计	359
11.4.12 学生补考成绩查询程序设计	360
附录 A 学生档案管理系统与学生成绩管理系统数据表结构	361
附录 B Visaul Basic 标准函数	367
附录 C Visual Basic 6.0 的常用属性、方法与事件	370
参考文献	376

第 1 章 Visual Basic 概述

1.1 Visual Basic 简介

Visual Basic (以下简称 VB), 意思是“可视化的 Basic”, 它是在 BASIC (Beginners All Purpose Symbolic Instruction Code, 可直译为: 初学者通用符号代码) 语言的基础上, 吸收近年来最新编程技术而成的一门程序设计语言。与 BASIC 语言不同的是, VB 用于在 Windows 操作环境下开发具有图形用户界面 (Graphic User Interface, 简称 GUI) 的应用程序, 而 BASIC 语言则是在 DOS 环境下开发具有字符界面的应用程序。VB 自问世以来, 因为其与 BASIC 语言的渊源和保留了 BASIC 语言易学易用的优点, 因此很快拥有了大量的爱好者和学习者, 时至今日仍然是许多初学者和专业开发人员学习和开发应用程序的首选程序设计语言。

VB 的最早版本 Visual Basic 1.0 由 Microsoft 公司于 1991 年推出, 其后随着计算机技术的发展, 在功能不断完善的同时, 其版本也不断升级换代, 到 1998 年 VB 发展到了 6.0 版。VB 在发展过程中, 功能真正变得强大的是从 5.0 版新增 Internet 开发、数据访问 ActiveX 控件等众多功能后开始的。本书介绍利用 VB 进行程序设计的方法主要以 Visual Basic 6.0 为背景。针对不同的用户, Visual Basic 6.0 提供了三个不同版本, 学习版主要是满足初学者和一般开发人员的需要; 专业版在学习版的基础上进行了功能扩充, 主要是为满足专业开发人员的需要, 企业版主要供开发小组使用, 三个版本中企业版功能最强大。

VB 最显著的特点可以概括为可视化、面向对象和事件驱动。可视化特点使得设计应用程序的界面时不需编写大量代码去描述界面元素的外观和位置, 而只要把预先建立的控件拖放到屏幕上即可。面向对象的程序设计方法 (Object-Oriented Programming, 简称 OOP) 与人的思维活动方式更为接近, 可有效降低编程的复杂性从而提高编程效率。事件驱动使得对用户界面上的任何操作都会自动转到对相应的代码进行处理, 同时也为程序运行过程中各对象之间的关联建立了有效的机制。

与大多数开发工具一样, VB 提供了集编辑、测试和程序调试等各种程序开发工具于一体的集成开发环境 (Integrated Developing Environment, 简称 IDE)。利用该集成环境, 可完成应用程序的界面设计、程序编码、测试和调试、编译并建立可执行程序等各种任务。此外 VB 还提供了对象的链接与嵌入功能 (Object Linking and Embedding, 简称 OLE), 利用该功能, 在 VB 的应用程序中可以使用其他 Windows 应用程序对象的某些功能。在数据操作方面, VB 也具有强大的功能, 利用 VB 能够开发各种数据库应用系统, 建立多种类型的数据库并管理、维护和使用这些数据库。

1.2 面向对象的程序设计方法

面向对象的程序设计方法自 20 世纪 90 年代面世以来, 由于其新颖的设计思想, 更符合

人类的思维习惯，因此迅速在全世界流行起来，并一跃成为程序设计的主流技术。面向对象的程序设计方法，其基本元素是对象，因此要理解和掌握这种程序设计方法，首先应理解对象和类的概念。

1. 对象和类

采用传统的面向过程的程序设计方法进行程序设计时，描述事物的数据和处理数据的函数在程序中是分开的。下面以一个 C++ 程序为例。

【例 1.1】设计程序，用结构体类型和结构体变量求矩形面积。

```
#include <iostream.h>
struct Rectangle
{   float Height,Width;
};
float area(float h,float w)
{   return h*w;
}
void main (void)
{   Rectangle r1;
    float s;
    cout<<"请输入矩形的长度";
    cin>> r1.Height;
    cout<<"请输入矩形的宽度";
    cin>>r1.Width;
    s=area(r1.Height,r1.Width);
    cout<<"矩形面积等于"<<s<<endl;
}
```

程序中定义了一个结构体 `Rectangle`，结构体中有 `Height` 和 `Width` 两个数据成员，它们是描述矩形的数据。程序中还定义了一个 `area` 函数，通过在主函数中调用该函数可以计算矩形的面积。从程序结构可以看出：描述矩形的数据和处理数据的函数是分开的，两者之间没有建立必然的内在联系。

面向过程的程序设计方法在程序的设计、调试、维护、移植以及数据的安全性等方面都存在着缺点，为克服这些缺点，当前均采用面向对象的程序设计方法。

采用面向对象的程序设计方法进行程序设计时，将描述事物的数据和处理数据的函数构成一个整体。同样以一个 C++ 程序为例。

【例 1.2】设计程序，用类和对象求矩形面积。

```
#include <iostream.h>
class Rectangle                                //定义 Rectangle 类
{   private:
        float Height,Width;                    //A
    public:
```

```

Rectangle(float H,float W)
{ Height=H;Width=W; }

float Area(void)                //B
{ return Height*Width;}

};

void main (void)
{   Rectangle  r1(600,1500);      //定义 r1 对象
    cout<<"矩形面积等于"<<r1.Area()<<endl;

}

```

从程序结构可以看出，通过定义一个 **Rectangle** 类，程序将描述矩形的数据成员 **Height** 和 **Width** 与处理数据的函数 **Area** 构成了一个整体。

计算机程序中将描述事物的数据和处理数据的函数（过程）构成的整体称为“类”，用类定义的变量称为“对象”，如例 1.2 中的 **Rectangle** 就是类，而用 **Rectangle** 类在主函数中定义的变量 **r1** 就是对象。计算机程序中类是抽象的，而用类定义的对象，由于为其数据成员赋予了具体的值，是具体的。在进行程序设计时，通常都是先定义类，然后再用类定义对象，同一个类可以定义多个不同的对象。对于常用的类，系统通常都会进行预先定义，在进行程序设计时，可以直接用它定义对象，从而实现对具体事物进行处理。针对特殊要求，用户也可以在程序设计时自己定义类，然后再用类定义对象。在 **VB** 中，系统预先定义了众多的类，如控件工具箱中的命令按钮、文本框、定时器等控件就是 **VB** 系统预先定义的类，设计程序时可以用它们来定义对象。当我们将控件工具箱中某个控件拖放到窗体上时，就相当于用类定义了一个对象。

2. 属性

类由描述事物的数据和处理数据的函数构成，类中描述事物的数据成员称为“属性”，如例 1.2 中 A 语句定义的 **Height** 和 **Width** 就是矩形的属性（如图 1.1 所示），用 **Height** 和 **Width** 属性可以描述矩形的高和宽。使用类定义一个具体对象时，对象将继承类规定的各种属性。**VB** 为每一个控件（类）都规定了若干属性，比如，命令按钮有显示方式、背景颜色、按钮名称、按钮标题、标题字体、标题字号、按钮大小位置和可见性等属性。将按钮控件拖放到窗体上，相当于用按钮类定义了一个按钮对象，按钮对象将继承按钮类的属性。通过为按钮对象设置具体的属性值，就可获得所需要的按钮外观及相关特性。在设计 **VB** 应用程序时，处理的是“实例化”的对象，因此属性通常意义上都是针对对象而言。

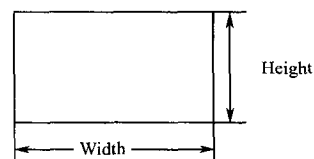


图 1.1 定义矩形类

3. 方法

类中处理数据的函数称为“方法”，如例 1.2 中 B 语句定义的 **Area** 函数就是矩形类的方法，用 **Area** 方法可以计算矩形的面积。使用类定义一个具体对象时，对象将继承类中定义的各种函数（方法）。**VB** 为大部分控件（类）都定义了若干方法，如窗体具有 **Show**（显示）、**Hide**（隐藏）、**Print**（打印）和 **Move**（移动）等方法。窗体调用 **Show** 方法将使自己显示在屏幕上，调用 **Hide** 方法将使自己从屏幕上隐藏起来，调用 **Print** 方法将在窗体上“打

印”文本，调用 Move 方法将可以在屏幕上移动。

面向过程的程序设计将描述事物的数据和处理数据的过程（或函数）分开，设计的程序可以概括为：程序=数据+过程定义+过程调用，这种设计方法的着眼点是过程（或函数），使用这种方法设计程序不但要关心数据，而且要关心数据是如何处理的，即既要告诉程序干什么，又要告诉程序如何进行工作。面向对象的程序设计将存在必然联系的数据描述和数据处理两者构成一个整体，设计的程序可以概括为：程序=对象+事件（或消息），这种设计方法的着眼点是对象，由于对象对数据描述和数据处理进行了封装，因此设计程序时只需要告诉对象干什么，而不必关心它是如何工作的。与面向过程的程序设计相比，面向对象的程序设计的提高程序的重用性和数据的安全性、改善程序的维护性、降低程序的复杂性等方面有诸多优点。面向对象的程序设计是一种全新的程序设计思想，对于初学者来说，开始时理解上可能会遇到一些困难，但一旦掌握了这种方法对于迅速提高编程能力将会产生很大帮助。

1.3 可视化的程序设计方法

视窗操作系统出现后，能够设计出带有图形用户界面的应用程序成了许多编程人员追求的目标，但在 VB 出现之前，想要设计出一个带有 Windows 风格的图形界面的应用程序将是一件非常困难的事情。VB 将描写界面元素外观的复杂技术封装在对象中，使用这种“可视化”工具，即使对于初学者来说，设计出带有美观实用的图形界面应用程序已不再是一件困难的事情。

用 VB 开发应用程序，包括两部分工作：一是设计图形用户界面；二是编写程序代码。VB 提供了一个“画板”（窗体）也就是用户界面，还提供一个“工具箱”，在“工具箱”中放了许多被称为“控件”的工具，比如有制作按钮的工具，有制作文本框的工具，有显示图形数据的工具等。可以从工具箱中取出所需工具，放到“画板”中适当的位置上，这样就形成了“用户界面”，也就是说，屏幕上的用户界面是用 VB 提供的可视化设计工具——“控件”直接“画”出来的，而不是用程序“写”出来的。当然最直观且最麻烦的界面设计也是由程序编写出来的，只不过这些编程工作不用读者来做，而是由 VB 系统自己来完成。

根据上面所述，所谓可视化程序设计是用系统提供的控件在窗体上设计出应用程序界面的过程。

1.4 事件驱动的编程机制

Windows 下的应用程序的用户界面都是由窗体、菜单和控件等对象构成的。程序运行后，各个对象的动作以及各对象之间的关联，完全取决于操作者所做的操作。操作者可能会首先操作界面上的 A 对象，然后再操作 B 对象，也有可能先操作 B 对象，再操作 A 对象，如果没有进行任何操作，程序就处于等待状态，就是说，程序的运行并没有固定的顺序。程序运行中各对象间要产生某种关联，这种关联同样是通过操作者用某种操作激活一个对象时，随着该对象的活动，又引发新的操作作用于其他对象上，而使另一个对象被激活。Windows 程序的上述工作模式，被称为事件（也称消息）驱动方式。用 VB 开发的应用程序是在 Windows 环境下运行的，其工作也要通过事件来驱动。

所谓“事件”，是指作用于某个对象上的一种操作或动作，如运行程序时用鼠标单击窗

体上菜单条的某个命令项时，“单击”（Click）这个动作就是作用在该命令项上的一个事件，也可以描述为该命令项发生了 Click 事件，当双击窗体上的某个命令按钮时，“双击”（Dblclick）就是作用在该命令按钮上的一个事件。VB 中的每个对象通常都可以响应多个不同的事件，一个对象可以响应哪些事件，VB 都预先进行了规定。

用 VB 开发的应用程序的工作方式为事件驱动，因此设计 VB 应用程序时应采用事件驱动的编程机制。程序运行中当作用于某个对象上的事件发生时，就要对相应的信息进行处理，对象就要产生状态和行为的改变，这种处理和改变的实现要通过编写相应的程序代码段来进行规定，在 VB 中这样的代码段称为“事件过程”。“事件过程”是构成一个完整 VB 应用程序不可缺少的组成部分，是 VB 应用程序的基本单元。“事件过程”由事件自动调用，即一个事件的发生驱动一段程序（事件过程）的执行，从而完成对象的某个功能。

根据上面所述，事件驱动的编程机制是指针对激活对象的事件进行相应“事件过程”编写的程序设计方法。

1.5 VB 的集成开发环境

VB 为使用者提供了一个功能强大而又易于操作的集成开发环境，用 VB 开发应用程序的大部分工作都可以通过该集成开发环境来完成。在 Windows 下，启动 VB，这时出现在屏幕上的画面就是 VB 的集成开发环境（IDE）（如图 1.2 所示），VB 的集成开发环境也称为 VB 的主窗口。VB 的集成开发环境是一典型的 Windows 界面，它由“标题栏”、“菜单栏”、“工具栏”、“控件工具箱”、“窗体设计器”、“工程资源管理器”、“属性设置窗口”和“窗体布局窗口”等组成。VB 集成开发环境中还有几个在必要时才会显示出来的子窗口，即“代码编辑器”和用于程序调试的“立即”、“本地”和“监视”窗口等。

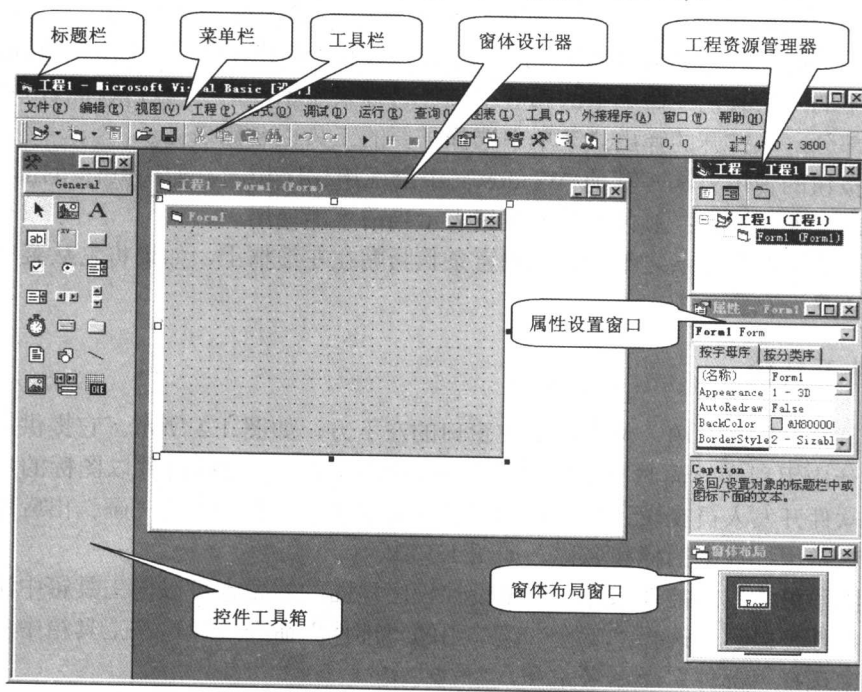


图 1.2 VB 集成开发环境