



世纪高职高专规划教材

Visual Basic 程序设计

■ 黄振中 侯 杰 编著



北京理工大学出版社

BEIJING INSTITUTE OF TECHNOLOGY PRESS

21 世纪高职高专规划教材

Visual Basic 程序设计

黄振中 侯杰 编著

 **北京理工大学出版社**

BEIJING INSTITUTE OF TECHNOLOGY PRESS

内 容 简 介

本书首先从 VB 的发展入手,循序渐进地介绍了 VB 程序设计的基础知识、程序设计的基本技巧,以及编程人员应该掌握的基本控件;接着结合实例介绍了如何用 Visual Basic 6.0 来设计程序,包括如何使用控件设计应用程序,如何设计 MDI 窗体,如何设计 Windows 标准菜单,以及如何使用面向对象编程技术,如何使用 VB 进行数据库编程等内容;同时还介绍了在使用 VB 中文版集成开发环境编程的过程中调试程序及遇到问题的解决方法。

本书内容实用,实例丰富,特别适合做高职高专的教材使用。通过阅读本书,读者可以举一反三,从而对 Visual Basic 6.0 编程有个比较全面的认识。

版权专有 侵权必究

图书在版编目(CIP)数据

Visual Basic 程序设计/黄振中,侯杰编著. —北京:北京理工大学出版社,2005.7

21 世纪高职高专规划教材

ISBN 7-5640-0536-X

I. V… II. ①黄…②侯… 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(2005)第 049521 号

出版发行 / 北京理工大学出版社

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010)68914775(办公室) 68944990(发行部)

网 址 / <http://www.bitpress.com.cn>

电子邮箱 / chiefedit@bitpress.com.cn

经 销 / 全国各地新华书店

印 刷 / 北京市顺义向阳胶印厂

开 本 / 787 毫米×1092 毫米 1/16

印 张 / 16.25

字 数 / 351 千字

版 次 / 2005 年 7 月第 1 版 2005 年 7 月第 1 次印刷

定 价 / 21.00 元

责任校对 / 郑兴玉

责任印制 / 刘京凤

* 图书出现印装质量问题,本社负责调换 *

丛书序

计算机高职高专教育发展非常迅速,在数量上已经超过了对本科人才的需求。高职高专这个层次有自己的特殊性,学习时间较短,要学习的内容很多,在教学要求上,既要有相当坚实的理论基础,又要能运用理论解决实际问题。目前在培养这个层次人才过程中,突出的矛盾之一是缺乏合适的教材,合适的教材既要高于中专层次,又不同于本科教材。

根据高职计算机专业(非计算机专业)计算机教学的特点,这套教材在注重系统性、科学性的基础上重点突出了实用性和操作性,将重点讲述计算机的基本概念和基本操作方法。按照由浅入深的教学原则,把各册教材的内容分割成若干个模块,采取循序渐进的教学方法,力求通俗而不肤浅,深入而不玄奥。对重点概念、重要的操作技能,力争讲深讲透。

本教材在内容安排上,以培养计算机应用能力为主线,构造该专业的课程设置体系和教学内容体系;从计算机应用需求出发,进行理论教学,强调理论教学与实验实训密切结合,尤其突出实践体系与技术应用能力的实训环节的教学;教材编写力求内容新颖、结构合理、概念清楚、实用性强、通俗易懂、前后相关课程有较好的衔接。与本科教材相比,本套教材在培养学生的应用技能上更有特色。和其他同类教材相比,本系列教材具有如下优势:

1. 自成体系:本系列教材覆盖了计算机基础教学各个层次的教学内容。其中既包括所有学生都必须掌握的计算机文化基础,也包括适用于各专业的软、硬件基础知识;既包括基本概念、方法与规范,也包括计算机应用开发的工具与环境。

2. 内容先进:注重将计算机技术的最新发展适当地引入教学中来,保持了教学内容的先进性。

3. 讲解到位:讲透基本理论、基本原理、方法和技术,在写法上力求叙述详细,算法具体,通俗易懂,便于自学。

4. 理论结合实际:计算机是一门实践性很强的科学,本套教材贯彻从实际中来到实践中去的原则,许多技术理论结合实例讲,以便于学习和理解。

5. 学练结合:每章都配有教学重点和课后练习,供学生总结提高使用。

这套教材都是聘请大专院校有丰富教学实践经验、工作在第一线的教师编写。在编写过程中,充分考虑了大专院校特点,在选材上贯彻少而精的原则,力求理论密切联系实际,深入浅出,便于教学。

因此本系列教材适用于高等职业学校、高等专科学校、成人高校及本科院校举办的二级职业技术学院、继续教育学院和民办高校,同时也可作为非计算机专业的本科生使用。

编委会

2005年3月

前 言

1991年,Microsoft公司推出了Windows应用程序开发工具——Visual Basic。Visual Basic的出现使得编程技术向前迈进了一大步。如今,使用VB编程已经成为了很多编程初学者首选的语言。Visual Basic功能强大,内容十分丰富,涉及很多方面。在5.0版本以后,Visual Basic已发展成为大型程序设计语言。Visual Basic6.0所提供的开发环境与Windows系统具有完全一致的界面,使用起来十分方便,其代码效率非常高,已达到Visual C++水平。在面向对象程序设计方面,6.0版的Visual Basic全面支持面向对象的程序设计,包括数据抽象、封装、对象与属性等。对于初学者来说,Visual Basic是十分理想的选择。

由于本书是面向高职高专的学生和Visual Basic的初学者,所以特别注重技术应用,在讲述方法上尽量做到简单明了、通俗易懂。本书的特点是起点低、入门快,不仅有各种功能和概念的详细说明,还结合大量编程实例进行分析和训练,使读者能够学练结合,更加容易接受并吸收所学习的知识。本书内容翔实,基本上涵盖了VB的方方面面。但我们求全更求精,在各方面都有更加高深的东西,并没有被纳入其中,读者如果需要,参考VB的使用手册即可得到相关信息。

本书主要包括如下内容:

第一章:Visual Basic基础知识,讲解该软件的应用领域和基本概念,以及如何搭设Visual Basic的开发环境。

第二章:Visual Basic的基本操作,通过一些简单解释,让读者快速入门,掌握对象、属性、方法的内涵,同时以一个小程序的开发,讲解VB的使用流程。

第三章:Visual Basic编程基础,讲解了变量、数组、内部函数以及程序结构的基本的语法。

第四章:窗体设计,讲解如何设计不同类型的程序界面。

第五章:基本控件,讲解了标签、文本框、按钮、复选项、图形、框架、滚动条等10多种常用控件的使用方法。

第六章:菜单设计,菜单是Windows编程中最常用到的,用户大部分操作都可以从中得到,本章着重讲解各种菜单的设计方法。

第七章:对话框设计,讲解了输入框、消息框、颜色对话框、字体对话框等的设计和使用方法。

第八章:文件操作,文件操作是一个完整的系统必需的功能,本章讲解了如何对文件中的内容进行读、写、删除等操作。

第九章:调试程序和创建安装文件。

第十章:数据库编程,讲解了使用VB开发数据库系统的各种方法,为以后工作打下基础。

第十一章:应用程序的开发。讲解了透明窗体、文件属性设置程序和学生信息管理系统的制作方法。让读者在实践中掌握Visual Basic。

学习任何一门编程语言,最重要的就是实践。笔者一向反对只学不练,或者学多于练。因此,本书提供了大量的编程实例供读者练习使用。对于从未接触过VB的读者来说,可以按照本书安排的顺序进行学习;对于已经有一定基础的读者,可以根据自身的情况选择合适的章节学习。

由于时间仓促,加上作者水平有限,书中错误之处在所难免,希望读者能够多提宝贵意见,以便更加完善本书的教材体系。

编 者

2005年3月

目 录

第1章 Visual Basic 基础知识 1	2.3 编写一个简单的 Visual Basic 应用程序 19
1.1 Visual Basic 的特点 1	2.3.1 用 Visual Basic 开发应用程序的一般步骤 20
1.1.1 Visual Basic 是可视的 1	2.3.2 编写 Visual Basic 应用程序 20
1.1.2 Visual Basic 是面向对象的 2	2.3.3 代码编辑器 26
1.1.3 Visual Basic 采用事件驱动编程机制 2	2.4 工程的管理 28
1.1.4 访问数据库 2	2.4.1 工程资源管理器 29
1.1.5 动态数据交换 2	2.4.2 创建和打开工程 29
1.1.6 对象链接与嵌入 2	2.4.3 保存程序 30
1.1.7 动态链接库 2	2.4.4 程序的装入 32
1.2 Visual Basic 的安装 3	2.4.5 程序的运行 34
1.2.1 Visual Basic 6.0 的系统要求 3	2.5 外接程序 35
1.2.2 Visual Basic 6.0 的安装 3	2.5.1 加载外接程序 35
1.2.3 Visual Basic 6.0 组件的添加和删除 3	2.5.2 使用向导 36
1.3 Visual Basic 的启动与退出 4	习题 41
1.3.1 启动 4	
1.3.2 退出 6	第3章 Visual Basic 编程基础 42
1.4 主窗口 6	3.1 常量与变量 42
1.4.1 标题栏和菜单栏 7	3.1.1 常量 42
1.4.2 工具栏 7	3.1.2 变量 43
1.5 其他窗口 9	3.2 运算符和表达式 50
1.5.1 窗体设计器和工程资源管理器 9	3.2.1 运算符 50
1.5.2 属性窗口和工具箱窗口 10	3.2.2 表达式的执行顺序 53
习题 12	3.3 数组 54
第2章 Visual Basic 的基本操作 13	3.3.1 改变数组的大小 55
2.1 对象 13	3.3.2 UBound 函数 56
2.1.1 属性 13	3.4 常用内部函数..... 56
2.1.2 方法 14	3.4.1 数学函数 57
2.1.3 事件 15	3.4.2 转换函数 58
2.2 工程、方案和集合 16	3.4.3 字符串函数 59
2.2.1 工程 16	3.4.4 日期函数 60
2.2.2 方案 17	3.5 程序设计结构 61
2.2.3 集合 19	3.5.1 赋值语句 61
	3.5.2 选择结构 62
	3.5.3 循环结构 64

3.6 过程	70	5.2 文本框控件	112
3.6.1 Sub 过程	70	5.2.1 文本框的基本属性	112
3.6.2 函数过程	72	5.2.2 选择文本	113
3.6.3 属性过程	74	5.2.3 文本框事件和方法	114
习题	76	5.2.4 字体与字号	115
		5.2.5 密码框	116
		5.2.6 使用剪贴板交换文本	116
第4章 窗体的设计	77	5.3 复选框和单选按钮	117
4.1 窗体概述	77	5.3.1 Value 属性	117
4.1.1 Windows 窗体	77	5.3.2 Alignment 属性	118
4.1.2 窗体的属性	77	5.3.3 Style 属性	118
4.2 窗体的事件	83	5.4 按钮控件	119
4.2.1 鼠标事件	83	5.4.1 按钮控件的常用属性	119
4.2.2 键盘事件	84	5.4.2 按钮的有效性	120
4.2.3 其他事件	85	5.4.3 多功能按钮	120
4.3 窗体的方法	86	5.4.4 图片按钮	122
4.3.1 Print 方法	86	5.4.5 使用键盘操作按钮	122
4.3.2 Cls 方法	88	5.5 图形控件	123
4.3.3 Move 方法	88	5.5.1 图片框和图像框	123
4.4 Windows 窗体设计的一般步骤	89	5.5.2 图形文件的导入	124
4.4.1 Windows 窗体设计步骤	89	5.5.3 直线和形状	126
4.4.2 Windows 窗体设计实例	90	5.6 列表框控件	127
4.5 使用窗体	99	5.6.1 属性	127
4.5.1 开始窗体	99	5.6.2 列表框事件	129
4.5.2 显示和隐藏窗体	99	5.6.3 列表框方法	129
4.5.3 使用 Sub Main	101	5.7 组合框控件	130
4.5.4 使用模态窗体	101	5.8 框架控件	131
4.6 Web 窗体的设计	102	5.9 滚动条控件	132
4.6.1 Web 窗体设计步骤	102	5.9.1 滚动条	132
4.6.2 Web 窗体设计实例	103	5.9.2 滚动条的主要属性	132
习题	110	5.10 计时器控件	133
		5.11 设置焦点	135
		5.12 Tab 顺序	136
		习题	137
第5章 基本控件	111		
5.1 标签控件	111	第6章 菜单设计	138
5.1.1 Alignment 属性	111	6.1 菜单简介	138
5.1.2 AutoSize 属性	111	6.2 菜单编辑器	139
5.1.3 BackStyle 属性	111	6.2.1 数据区	139
5.1.4 BackColor 属性	111	6.2.2 编辑区	140
5.1.5 BorderStyle 属性	111	6.2.3 菜单项显示区	141
5.1.6 Caption 属性	112		
5.1.7 Font 属性	112		

6.3 菜单的设计	141	8.3.3 顺序文件操作	179
6.3.1 建立菜单	141	8.4 随机文件操作	181
6.3.2 编写代码	147	8.4.1 随机文件的写操作	181
6.4 在运行时控制菜单	149	8.4.2 随机文件的读操作	182
6.4.1 有效性控制	149	8.5 二进制文件操作	183
6.4.2 菜单项标记	149	习题	184
6.4.3 菜单项的隐藏与显示	150		
6.5 快捷菜单	150	第9章 调试程序与创建安装程序	185
习题	153	9.1 调试程序与错误捕获	185
第7章 对话框的设计	154	9.1.1 错误类型	185
7.1 对话框的概述	154	9.1.2 调试程序	186
7.2 预定义对话框	155	9.1.3 错误捕获及处理	191
7.2.1 输入框	155	9.1.4 条件编译	195
7.2.2 消息框	156	9.2 创建安装程序	196
7.3 通用对话框	158	9.2.1 安装和部署项目	196
7.3.1 打开对话框的方法	158	9.2.2 CAB 项目	200
7.3.2 文件对话框	159	9.2.3 合并模块项目	201
7.3.3 颜色对话框	162	习题	202
7.3.4 字体对话框	164		
7.4 自定义对话框	165	第10章 数据库编程	203
习题	170	10.1 数据库概述	203
第8章 文件操作	171	10.1.1 基本概念	203
8.1 用文件系统对象模型操作文件	171	10.1.2 数据库系统	204
8.1.1 FSO 对象	171	10.2 数据库设计工具	204
8.1.2 TextStream 对象	172	10.2.1 数据库设计器	204
8.1.3 Drive 对象	174	10.2.2 表设计器	205
8.1.4 Folder 对象	174	10.2.3 SQL 编辑器	206
8.1.5 File 对象	175	10.3 ADO 简介	206
8.2 用命名空间操作文件	177	10.3.1 操作性	207
8.2.1 通过 FileStream 类读写文件	177	10.3.2 维护性	207
8.2.2 通过 BinaryReader 和 BinaryWriter 类读写文件	177	10.3.3 编程性	207
8.2.3 通过 StreamReader 和 StreamWriter 类读写文件	178	10.3.4 性能	207
8.2.4 驱动器和文件夹操作	178	10.3.5 可缩放性	207
8.3 直接操作文件	179	10.4 ADO 的使用	207
8.3.1 文件操作类型、函数和语句	179	10.4.1 引入 ADO 名称空间	207
8.3.2 数据操作函数	179	10.4.2 使用 Connection 对象连接数据库	208
		10.4.3 使用 Command 对象操作数据库	209
		10.4.4 读取和显示数据	211
		10.5 ADO 程序设计的一般步骤	215

10.6 XML 简介	220	11.2.1 创建界面	233
习题	228	11.2.2 编写代码	234
第 11 章 应用程序系统的开发	229	11.3 文件属性	235
11.1 基本程序设计练习	229	11.3.1 创建窗体	235
11.1.1 创建窗体	229	11.3.2 添加代码	236
11.1.2 添加控件	229	11.4 学生信息管理系统	240
11.1.3 添加类模块	230	11.4.1 功能介绍	240
11.1.4 添加代码	230	11.4.2 数据库设计	240
11.2 透明窗体	233	11.4.3 界面设计	241
		11.4.4 程序代码介绍	244

第 1 章 Visual Basic 基础知识

Visual Basic 是在 BASIC 语言的基础上研制而成的。

Basic 诞生于 20 世纪 60 年代初期,是 Beginner's all-purpose Symbolic Instruction Code(初学者通用符号指令代码)的缩写。随着计算机技术的快速发展,硬件功能的增强,以及 Windows 操作系统的流行,Basic 的缺点逐渐显现出来。1991 年 Microsoft 公司推出了 Visual Basic。

用 Visual Basic 来设计应用程序,将 Basic 语言应用于程序中,提供了编程的简易性;又采用了可视化设计工具,具有“所见即所得”的可视性,适应了 Windows 所特有的优良性能。Visual Basic 采用的是事件驱动编程机制,用户不必像以前使用 Basic 语言编写程序那样,精确写出执行的每一步骤;不必写出很长程序,只要写出简短的程序片断,就可以完成所需的操作。Visual Basic 同时采用图形工作环境,通过图形对象来设计应用程序,用户可以很方便地设计出具有 Windows 风格图形界面的应用软件。

自 1991 年推出 Visual Basic 1.0 至今已经历了 6 个版本。3.0 以前的 Visual Basic 只能在 Windows 3x 环境下运行,4.0 以后的版本要在 Windows 9x 或 WindowsNT 3x 及更高的版本中运行。5.0 以前版本是 16 位应用程序的开发工具,5.0 以后的版本是 32 位应用程序的开发工具。

Visual Basic 6.0 共有 3 个版本:标准版、专业版和企业版。标准版主要是为初学者开发的,是基于 Windows 的应用程序而设计的;专业版主要是为专业人员开发的,是基于客户/服务器的应用程序开发而设计的;企业版则是为专业编程人员开发的,为创建更高级的分布式、高性能的客户/服务器或 Internet/Intranet 上的应用程序而设计的。

这些版本是在相同的基础上建立起来的,大多数应用程序可在 3 种版本中通用。

1.1 Visual Basic 的特点

1.1.1 Visual Basic 是可视的

Visual 这一英文单词的意思是“可视的”、“视觉的”,Visual Basic 在这里包含“可视化程序设计”的含义。

可视化设计工具把 Windows 界面设计的复杂性“封装”起来,用户不必为界面设计编写程序代码,只需要利用系统提供的工具,在屏幕上画出各种对象,并设置对象的属性即可。“可视化程序设计”为用户制作具有 Windows 风格的应用程序提供了简化编程难度的有效方法。

1.1.2 Visual Basic 是面向对象的

面向对象的程序设计方法是把程序和数据封装起来作为一个对象,并为每一个对象设置所需要的属性。这些图形对象的建立不必用语句来描述,而是用工具画在界面上。

为什么采用面向对象的程序设计方法后用户不必用语句来描述所要画出的图形呢?因为设计这些图形对象的程序代码由 Visual Basic 自动生成并封装起来,计算机是执行了这些封装起来的数据和程序代码,才画出各种各样图形界面的。

直观的、图形丰富的工作环境,已是当今所有应用程序必须具备的共同点,利用面向图形对象的程序设计方法设计应用程序,可使图形对象的建立变得十分简单和容易。

1.1.3 Visual Basic 采用事件驱动编程机制

所谓事件就是对象上所发生的事情。Visual Basic 通过事件来执行对象的操作。当我们单击(或双击)一个对象时,该对象将会触发一个事件,该事件又通过一个程序段来响应,从而实现指定的操作。这就是所谓的事件驱动机制。

所以一般用 Visual Basic 设计的应用程序,无须具有明显的开始部分和结束部分,而是编写若干过程,不同的对象分别对应不同的过程,由用户操作触发某个事件来执行相应的过程,从而完成某种特定的功能。

事件驱动是一种适用于图形用户界面(GUI)的编程方式。

1.1.4 访问数据库

在 Visual Basic 应用程序中,可直接建立 Access 格式的数据库或访问 Access 中的数据,并进行数据存储和检索。Visual Basic 还能编辑和访问如 FoxPro、Paradox 等外部数据库。

Visual Basic 提供开放式数据链接功能,可直接访问或建立链接的方式使用并操作后台大型网络数据库。

1.1.5 动态数据交换

把一种应用程序中的数据动态地链接到另一种应用程序中,使两种不同的应用程序之间交换数据。这种使两种完全不同的应用程序进行通信的技术称为动态数据交换。

Visual Basic 提供了动态数据交换的编程技术,实现了 Visual Basic 应用程序与其他 Windows 应用程序之间的动态数据交换。

1.1.6 对象链接与嵌入

Visual Basic 采用了对象链接与嵌入技术,将每个应用程序都看作一个对象,将不同的对象链接起来,再嵌入到 Visual Basic 应用程序中,得到具有声、像和文字等信息的集合式文件。

1.1.7 动态链接库

Visual Basic 使用了动态链接库技术,这样可以在 Visual Basic 应用程序中调用其他语言

编写的函数,或者将 C/C++ 或汇编语言写的程序加入到 Visual Basic 应用程序中。

1.2 Visual Basic 的安装

1.2.1 Visual Basic 6.0 的系统要求

目前常用的计算机系统配置一般都能满足 Visual Basic 6.0 的要求。其中有 3 个主要的系统要求:

- ①安装 Visual Basic 6.0 的计算机要求 486DX66、Pentium 或更高的微处理器。
- ②在 Windows 98 下至少需要 16 MB 以上内存,Windows NT4.0 下需要 32 M 以上内存。
- ③硬盘空间:
标准版:典型安装 48 MB,完全安装 80 MB。
专业版:典型安装 48 MB,完全安装 80 MB。
企业版:典型安装 128 MB,完全安装 147 MB。
MSDN:至少需要 67 M。

MSDN 是 Visual Basic 帮助文件所必需的,它包含了 Visual Basic 的编程技术信息及其他资料。

1.2.2 Visual Basic 6.0 的安装

Visual Basic 6.0 的安装过程与其他 Microsoft 应用软件的安装过程类似,首先将 Visual Basic 6.0 的安装光盘放入光驱,然后在“我的电脑”或“资源管理器”中执行安装光盘上的 Setup 程序(若没有取消“自动播放”功能,则安装程序将会自动运行)。

1.2.3 Visual Basic 6.0 组件的添加和删除

在使用 Visual Basic 6.0 的过程中,由于硬盘空间的原因,有时需要添加未安装的组件或删除不需要的组件。安装或删除 Visual Basic 6.0 组件的方法是:再次运行 Visual Basic 6.0 安装程序,打开 Visual Basic 6.0 中文企业版安装向导对话框。

- ①选择“工作站工具和组件”选项。
- ②单击“下一步”按钮,打开“添加/删除”对话框,对话框中有“添加/删除”、“重新安装”和“全部删除”3 个按钮。
- ③在对话框中根据需要选择相应的选项,再根据提示进行操作即可。

1.3 Visual Basic 的启动与退出

1.3.1 启动

启动 Visual Basic 有多种方法,主要有:

(1) 使用快捷方式

双击 Windows 桌面上的 Visual Basic 快捷图标即可启动 Visual Basic。

(2) 使用菜单命令

步骤如下:

①单击 Windows 操作系统的“开始”按钮,弹出下拉菜单,把鼠标指针移到“程序”命令,弹出级联菜单。

②把鼠标指针移到“Micorsoft Visual Basic 6.0 中文版”上,弹出 Visual Basic 6.0 程序组的级联菜单。

③单击“Microsoft Visual Basic 6.0 中文版”,即可进入启动 Visual Basic。

(3) 使用“我的电脑”启动

步骤如下:

①在 Windows 桌面双击“我的电脑”,弹出“我的电脑”窗口,然后单击 Visual Basic 6.0 所在硬盘的盘符,打开相应的驱动器窗口。

②依次进入文件夹 D:\Program Files\Microsoft Visual Studio\VB98。

③双击“Visual Basic 6”图标,即可启动 Visual Basic。

启动 Visual Basic 后将首先显示版权信息,稍后便显示“新建工程”对话框,如图 1-1 所示。对话框中显示出可以在 Visual Basic 6.0 中使用的工程类型,主要有:

(1) 标准 EXE 程序

建立标准 Windows 下的可执行文件(.EXE 文件)。它是“新建”环境中最基本的类型,也是缺省类型。

(2) ActiveX EXE 程序

这种程序只能在专业版和企业版中建立,用于建立进程外的对象的链接与嵌入服务器应用程序项目类型。这种程序可包装成可执行文件。

(3) ActiveX DLL 程序

这种程序与 ActiveX EXE 程序是一致的,只是包装不一样,ActiveX DLL 只能包装成动态链接库。

(4) ActiveX 控件

只能在专业版或企业版中建立,用于开发用户自定义的 ActiveX 控件。



图 1-1 “新建工程”对话框

(5) Visual Basic 应用程序向导

用于在开发环境中建立新的应用程序框架。

(6) 数据工程

提供开发数据报表应用程序的框架,选中该图标后,将自动打开数据环境设计器和数据报表设计器。

(7) 外接程序

用于建立 Visual Basic 外接程序,并在开发环境中自动打开联接设计器。

(8) ActiveX 文档 EXE 和 ActiveX 文档 DLL 程序

建立可以在超链接环境中运行的 Visual Basic 应用程序,即 Web 浏览器。

(9) Visual Basic 企业版控件

该选项不是用来建立应用程序的,而是在工具箱中加入企业版控件图标。

在对话框中选择要建立的工程类型,对初学者应选择“标准 EXE”,然后单击“打开”按钮,即打开 Visual Basic 集成环境界面,如图 1-2 所示。

Visual Basic 6.0 提供了许多种工程类型以满足不同的需要。每次启动 Visual Basic 时,都要显示“新建工程”对话框。在一般情况下,可能主要使用“标准 EXE”工程,因此,没有必要在每次启动 Visual Basic 时都显示该对话框,这可以通过选择对话框左下角的“不再显示这个对话框”选项来实现(用鼠标单击小方框,使框内有“√”)。选择该选项后,再选择“标准 EXE”,然后单击“打开”按钮,进入 Visual Basic 开发环境。以后再启动 Visual Basic 时,即不再显示“新建工程”对话框,直接进入开发环境。

在默认情况下,Visual Basic 6.0 的集成开发环境为传统的 Windows MDI(多文档界面)方

式;此外,也可以用 SDI(单文档界面)方式启动 Visual Basic 设计。在多数情况下,使用 SDI 方式可能会更方便。为了把编程环境变为 SDI 方式,可执行“工具”菜单中的“选择项”命令,打开“选项”对话框,选择“高级”选项卡,在对话框中选择“SDI 开发环境”选项,然后单击“确定”按钮。这样设置后,退出 Visual Basic,然后重新启动,即可按 SDI 方式进入 Visual Basic 集成开发环境。



图 1-2 Visual Basic 集成环境

从图 1-2 中可以看出,启动 Visual Basic 后,屏幕上分为若干部分,包括标题栏、菜单栏、工具栏、工具箱、窗体设计器窗口、工程资源管理器、窗体布局窗口和属性窗口。

1.3.2 退出

退出 Visual Basic 的方法是:

- ①单击 Visual Basic 窗口右上角的“关闭”按钮。
- ②选择“文件”菜单中的“退出”命令。
- ③按下 Alt + Q 组合键。

退出 Visual Basic 时,若新建立的程序或已修改过的原有程序没有存盘,系统将显示一个对话框,询问用户是否将其存盘,用户作出应答后才能退出 Visual Basic。

1.4 主窗口

主窗口也称设计窗口。启动 Visual Basic 后,主窗口位于集成环境的顶部,该窗口由标题栏、菜单栏和工具栏组成。

1.4.1 标题栏和菜单栏

1. 标题栏

标题栏是屏幕顶部的水平条,它显示的是应用程序的名字。用户与标题栏之间的交互关系由 Windows 来处理,而不是由应用程序处理。启动 Visual Basic 后,标题栏中显示的信息为:

工程 1-Microsoft Visual Basic [设计]

方括号中的“设计”表明当前的工作态是“设计阶段”。随着工作状态的不同,方括号中的信息也随之改变,可能会是“运行”或“Break”,分别代表“运行阶段”或“中断阶段”。这3个阶段也分别称为“设计模式”、“运行模式”和“中断模式”。

2. 菜单栏

在标题栏下面是集成环境的主菜单。菜单栏中的菜单命令提供了开发、调试和保存应用程序所需要的工具。Visual Basic 6.0 中文版的菜单栏共有 13 个菜单项,即文件、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。每个菜单项含有若干个菜单命令,执行不同的操作。用鼠标单击某个菜单项,即可打开该菜单,然后用鼠标单击菜单中的某一条就能执行相应的菜单命令。例如,单击“文件”,就可以打开文件菜单。打开菜单后,若单击“打开工程”,就可以打开已有的工程文件;而若单击“工程另存为”,就可以保存文件,等等。在以后的叙述中,我们把上面的操作记为“执行文件菜单中的‘打开工程’命令”、“执行文件菜单中的‘工程另存为’命令”。

菜单中的命令分为两种类型,一类是可以直接执行的命令,这类命令的后面没有任何信息(例如“保存工程”);另一类在命令名后面带有省略号(例如“打开工程”),需要通过打开“对话框”来执行。在用鼠标单击后一类命令后,屏幕上将显示一个对话框,利用该对话框可以执行各种有关的操作。在“文件”菜单中,“新建工程”、“保存工程”等是可以直接执行的命令,而“打开工程”、“工程另存为”等命令则必须通过对话框来执行。此外,从“文件”菜单可以看出,在有些命令的后面还带有其他信息,例如:

打开工程... Ctrl + O

保存 Form1 Ctrl + S

其中“Ctrl + O”等叫做“热键”(或快捷键)。在菜单中,热键列在相应的菜单命令之后,与菜单命令具有相同的作用。使用热键方式,不必打开菜单就能执行相应的菜单命令。例如,按 Ctrl + O,可以立即执行“打开工程”命令。注意,只有部分菜单命令能通过热键执行。

菜单被打开后,在屏幕上显示相应的菜单命令。若打开了不适当或不需要的菜单,或者执行菜单命令时打开了不需要的对话框,可以用 Esc 键关闭。

Visual Basic 应用程序的编辑、编译、连接、运行、调试及文件的打开、保存等都可以通过相应的菜单命令来实现,其用法与上面介绍的类似。

1.4.2 工具栏

Visual Basic 6.0 提供了 4 种工具栏,包括编辑、标准、窗体编辑器和调试,并可根据需要定

义用户自己的工具栏。在一般情况下,集成环境中只显示标准工具栏,其他工具栏可以通过“视图”菜单中的“工具栏”命令打开(或关闭)。每种工具栏都有固定和浮动两种形式。把光标移到固定形式工具栏中没有图标的地方,按住鼠标左键向下拖动,或者双击工具栏左端的两条浅色竖线,即可把工具栏变为浮动的;而若双击浮动工具栏的标题条,则可变为固定工具栏。

固定形式的标准工具栏位于菜单栏的下面,即主窗口的底部,它以图标的形式提供了部分常用菜单命令的功能。只要用鼠标单击代表某个命令的图标按钮,就能直接执行相应的菜单命令。标准工具栏中有 21 个图标,代表 21 种操作。大多数图标都有与之等价的菜单命令。

表 1-1 列出了标准工具栏中各图标的作用。

表 1-1 标准工具栏图标及作用

编号	名 称	作 用
1	添加工程	添加一个新工程,相当于“文件”菜单中的“添加工程”命令
2	添加窗体	在工程中添加一个新窗体,相当于“工程”菜单中的“添加窗体”命令
3	菜单编辑器	打开菜单编辑对话框,相当于“工具”菜单中的“菜单编辑器”命令
4	打开工程	用来打开一个已经存在的 Visual Basic 工程文件,相当于“文件”菜单中的“打开工程”命令
5	保存工程(组)	保存当前的 Visual Basic 工程(组)文件,相当于“文件”菜单中的“保存工程”命令
6	剪切	把选择的内容剪切到剪贴板,相当于“编辑”菜单中的“剪切”命令
7	复制	把选择的内容复制到剪贴板,相当于“编辑”菜单中的“复制”命令
8	粘贴	把剪贴板的内容复制到当前插入位置,相当于“编辑”菜单中的“粘贴”命令
9	查找	打开“查找”对话框,相当于“编辑”菜单中的“查找”命令
10	撤消	撤消当前的修改
11	重复	对“撤消”的反操作
12	启动	用来运行一个应用程序,相当于“运行”菜单中的“启动”命令
13	中断	暂停正在运行的程序(可以用“启动”按钮或 Shift + F5 继续),相当于热键 Ctrl + Break 或“运行”菜单中的“中断”命令
14	结束	结束一个应用程序的运行并回到设计窗口,相当于“运行”菜单中的“结束”命令
15	工程资源管理器	打开工程资源管理器窗口,相当于“视图”菜单中的“工程资源管理器”命令
16	属性窗口	打开属性窗口,相当于“视图”菜单中的“属性窗口”命令
17	窗体布局窗口	打开窗体布局窗口,相当于“视图”菜单中的“窗体布局窗口”命令
18	对象浏览器	打开“对象浏览器”对话框,相当于“视图”菜单中的“对象浏览器”命令
19	工具箱	打开工具箱,相当于“视图”菜单中的“工具箱”命令
20	数据视图	打开数据视图窗口
21	组件管理器	管理系统中的组件(Component)

在工具栏的右侧还有两个栏,分别用来显示窗体的当前位置和大小,其单位为 twip,1 英寸等于 1440 twip。左边一栏显示的是窗体左上角的坐标,右边一栏显示的是窗体的长×宽。

twip 是一种与屏幕分辨率无关的计量单位,无论在什么屏幕上,若画了一条 1440 twip 的直线,打印出来都是 1 英寸。这种计量单位可以确保在不同的屏幕上都能保持正确的相对位置或比例关系。

在 Visual Basic 中,twip 是默认单位,可以通过 ScaleMode 属性改变。