



高等学校电子信息类专业规划教材

Visual Basic 程序设计教程

宁正元 主 编
潘晓文 蒋萌辉 林大辉 副主编



本书配光盘



清华大学出版社
<http://www.tup.tsinghua.edu.cn>



北京交通大学出版社
<http://press.bjtu.edu.cn>

21 世纪高等学校电子信息类专业规划教材

Visual Basic 程序设计教程

宁正元 主编

潘晓文 蒋萌辉 林大辉 副主编

清华大学出版社
北京交通大学出版社

• 北京 •

内 容 简 介

Visual Basic 是目前最流行的可视化开发工具之一。内容分四个部分：第一部分是 Visual Basic 编程基础，初步介绍了面向对象的概念、使用 Visual Basic 集成开发环境编辑 Visual Basic 应用程序的方法，以及 Visual Basic 语言要点；第二部分以 Visual Basic 的可视化设计为主要内容，主要介绍了窗体和一些标准控件的属性、方法和事件，Visual Basic 菜单技术和多文档界面；第三部分介绍文件管理与操作，以及用户自定义类；第四部分是 Visual Basic 高级应用，内容包括 Visual Basic 的多媒体设计、Visual Basic 的数据库应用等，当前较为流行的实用技术，这部分内容是为了读者增加知识面、提高 Visual Basic 6.0 应用能力而编写的。

本书图文并茂、详略适当、循序渐进、由浅入深地介绍了 Visual Basic 的知识，是初学者入门的教材，也可作为参加全国计算机等级考试考生的考前培训和模拟练习使用，大量的实例对 Visual Basic 程序员也极具参考价值。书后所附光盘是模拟练习及考试系统，对读者考前上机训练有良好的指导作用。

版权所有，翻印必究。

本书封面贴有清华大学出版社激光防伪标签，无标签者不得销售。

图书在版编目 (CIP) 数据

Visual Basic 程序设计教程 / 宁正元主编. —北京：清华大学出版社；北京交通大学出版社，2004. 4

(21 世纪高等学校电子信息类专业规划教材)

ISBN 7-81082-288-8

I. V… II. 宁… III. BASIC 语言-程序设计-教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2004) 第 017566 号

责任编辑：陈川

出版者：清华大学出版社 邮编：100084 电话：010-62776969
北京交通大学出版社 邮编：100044 电话：010-51686045, 62237564

印刷者：北京瑞达方舟印务有限公司

发行者：新华书店总店北京发行所

开 本：185×260 印张：21.25 字数：488 千字 附光盘 1 张

版 次：2004 年 4 月第 1 版 2004 年 4 月第 1 次印刷

书 号：ISBN 7-81082-288-8 / TP·108

印 数：1~8 000 册 定价：32.00 元 (含光盘)

目 录

第1章 Visual Basic 入门.....	1
1.1 程序和程序设计语言	1
1.1.1 低级语言	1
1.1.2 高级语言	2
1.1.3 高级语言程序的执行	3
1.2 类、对象、属性、方法与事件	3
1.2.1 类与对象	3
1.2.2 属性	4
1.2.3 方法	4
1.2.4 事件	4
1.2.5 Windows 操作系统中的对象	5
1.3 Visual Basic 简介	5
1.3.1 Visual Basic 特点	5
1.3.2 Visual Basic 版本简介	6
1.3.3 Visual Basic 的安装	6
1.4 Visual Basic 基础	7
1.4.1 启动 Visual Basic	7
1.4.2 什么是集成开发环境 (IDE)	7
1.5 Visual Basic 应用程序开发过程	8
1.5.1 创建应用程序界面	9
1.5.2 设置窗体和控件属性值	11
1.5.3 编写事件过程	13
1.6 Visual Basic 应用程序的结构	16
1.6.1 工程 (Project) 与工程文件	16
1.6.2 工程资源管理器	16
1.6.3 模块	17
习题	18
第2章 Visual Basic 的代码编程基础	19
2.1 编码基础	19
2.1.1 将单行语句分成多行	19
2.1.2 将多个语句合并到同一行上	19
2.1.3 在代码中添加注释	19
2.1.4 理解数字系统	20
2.2 数据类型及运算	20

2.2.1 数据类型	20
2.2.2 变量	23
2.2.3 常量	32
2.2.4 常用内部函数	34
2.3 顺序结构程序设计	44
2.3.1 赋值语句	44
2.3.2 运算符和表达式	45
2.3.3 数据输入输出	48
2.4 选择结构程序设计	61
2.4.1 用 if 语句实现选择结构	61
2.4.2 多分支结构	66
2.4.3 选择结构应用	69
2.5 循环结构程序设计	71
2.5.1 用 If 语句和 GoTo 语句实现循环	72
2.5.2 While 语句和 Wend 实现循环	72
2.5.3 Do...Loop 语句实现循环	73
2.5.4 用 For...Next 语句实现循环	76
2.5.5 循环的嵌套	78
2.5.6 循环控制程序应用	79
2.6 数组	85
2.6.1 数组的概念	86
2.6.2 数组的基本操作	90
2.6.3 动态数组	94
2.6.4 数组的初始化	96
2.6.5 数组程序举例	97
习题	101
第 3 章 Visual Basic 的过程、函数和方法	104
3.1 Sub 过程	104
3.2 Function 过程	108
3.3 参数传递	111
3.3.1 形参与实参	111
3.3.2 命名参数	113
3.3.3 数组做参数	113
3.3.4 可选参数和可变参数	117
3.3.5 对象型参数	119
习题	122
第 4 章 创建用户界面	123

4.1 Visual Basic 的窗体	123
4.1.1 窗体常用的属性	123
4.1.2 窗体的事件	127
4.1.3 窗体的生命周期	128
4.1.4 窗体的方法	132
4.2 Visual Basic 的控件	134
4.2.1 内部控件（也称标准控件）	134
4.2.2 ActiveX 控件	134
4.2.3 可插入的对象	134
4.2.4 控件的属性、事件和方法	136
4.3 常用标准控件	137
4.3.1 文本控件	137
4.3.2 命令按钮	142
4.3.3 选择控件	143
4.3.4 列表框和组合框	145
4.3.5 滚动条	152
4.3.6 计时器	156
4.3.7 图形与图像控件	157
4.3.8 框架	164
4.3.9 焦点和 Tab 顺序	168
4.3.10 控件数组	170
4.4 常用的绘图方法	175
4.5 键盘与鼠标事件过程	181
4.5.1 响应键盘事件	181
4.5.2 响应鼠标事件	184
4.6 多重窗体程序设计	193
4.7 对话框	198
4.7.1 Common Dialog 控件介绍	198
4.7.2 Common Dialog 控件的使用	199
习题	201
第 5 章 菜单设计与多文档界面	207
5.1 菜单的基本概念	207
5.1.1 菜单栏	207
5.1.2 弹出式菜单	209
5.2 菜单编辑器（Menu Editor）简介	210
5.2.1 显示“菜单编辑器”	210
5.2.2 菜单编辑器窗口	210

5.3 设计菜单栏	211
5.3.1 建立菜单栏	211
5.3.2 编写菜单控件的代码	212
5.4 设计弹出式菜单	212
5.4.1 菜单的准备	213
5.4.2 Popup Menu 方法	213
5.5 多文档界面 (MDI) 应用程序	215
5.5.1 Windows 的应用程序用户界面样式	215
5.5.2 创建多文档界面 (MDI) 应用程序	216
5.5.3 MDI 应用程序中的菜单	217
习题	218
第 6 章 文件管理与操作	220
6.1 文件系统的基本概念	220
6.1.1 文件系统的概念	220
6.1.2 文件系统的基本操作	220
6.1.3 文件的类型	221
6.1.4 对文件访问的基本步骤	221
6.2 FSO 对象模型	222
6.2.1 使用 FSO 对象对目录的操作	223
6.2.2 使用 FSO 对象对文件的操作	225
6.3 文件系统控件	228
6.3.1 文件系统控件及其属性	228
6.3.2 使用文件系统控件的组合	229
6.4 用传统的文件 I/O 语句和函数处理文件	230
6.4.1 访问顺序型文件	230
6.4.2 随机文件访问	234
6.4.3 二进制文件的访问	236
习题	238
第 7 章 面向对象技术及其在 Visual Basic 中的实现	244
7.1 面向对象的程序设计	244
7.1.1 面向对象与代码重用	244
7.1.2 面向对象程序设计的基本特性	245
7.1.3 Visual Basic 的对象浏览器	246
7.2 利用类模块自定义类	247
7.2.1 Visual Basic 的类模板	247
7.2.2 为类添加属性	248
7.2.3 为类添加方法	251

7.2.4 设置类的默认属性与方法	253
7.3 声明和使用对象	254
7.3.1 用 New 关键字创建新对象	254
7.3.2 前期绑定与后期绑定	260
7.4 添加用户事件	264
习题	270
第 8 章 Visual Basic 多媒体程序设计	271
8.1 Visual Basic 直接编写动画效果应用程序	271
8.2 Visual Basic 的 OLE	273
8.3 使用 MCI 控件编写多媒体应用程序	275
8.3.1 MCI 设备和 MCI 指令	275
8.3.2 Visual Basic 的 MCI 控件	279
8.4 Visual Basic 调用 API 函数的多媒体实现	281
8.4.1 Windows API 函数简介	281
8.4.2 Windows API 函数实现多媒体应用实例	282
习题	287
第 9 章 数据库编程	288
9.1 数据库基础	288
9.1.1 数据库基本概念	288
9.1.2 结构化查询语言	289
9.2 可视化数据管理器	291
9.2.1 打开可视化数据管理器	291
9.2.2 创建新的数据库	292
9.2.3 添加表与索引	292
9.2.4 修改表结构	294
9.2.5 添加和修改表中记录	294
9.2.6 执行 SQL 命令	295
9.3 Activex 数据对象与数据环境设计器	295
9.3.1 Activex 数据对象	295
9.3.2 数据环境设计器	296
9.3.3 查询设计器与数据视图窗口	299
9.4 记录集对象	301
9.4.1 记录集对象的类型与引用	301
9.4.2 Fields 集合	302
9.4.3 定位记录集中的对象	303
9.4.4 更新记录集	306
习题	310

第 10 章 实验篇	311
10.1 Visual Basic 环境和程序设计初步	311
10.1.1 实验目的	311
10.1.2 实验内容	311
10.2 顺序程序设计	313
10.2.1 实验目的	313
10.2.2 实验内容	313
10.3 选择结构程序设计	314
10.3.1 实验目的	314
10.3.2 实验内容	314
10.4 循环结构程序设计及数组的运用	316
10.4.1 实验目的	316
10.4.2 实验内容	316
10.5 过程	318
10.5.1 实验目的	318
10.5.2 实验内容	318
10.6 窗体和常用控件（一）	329
10.6.1 实验目的	319
10.6.2 实验内容	319
10.7 常用控件（二）	321
10.7.1 实验目的	321
10.7.2 实验内容	321
10.8 菜单设计和多文档界面	323
10.8.1 实验目的	323
10.8.2 实验内容	323
10.9 文件管理与操作	324
10.9.1 实验目的	324
10.9.2 实验内容	324
10.10 Visual Basic 中的面向对象技术	325
10.10.1 实验目的	325
10.10.2 实验内容	325
10.11 Visual Basic 多媒体程序设计	326
10.11.1 实验目的	326
10.11.2 实验内容	326
10.12 数据库编程	326
10.12.1 实验目的	326
10.12.2 实验内容	326

第 1 章 Visual Basic 入门

对许多计算机初学者来说,能够在 Windows 环境下编制一个界面友好的应用程序是一件梦寐以求的事情。那么,什么是计算机程序?我们用什么工具来编制程序?

1.1 程序和程序设计语言

有人认为计算机是万能的,只要把任务告诉计算机,计算机就会自动完成一切,并给出正确的结果,其实这是一个误解。现在人们使用的大多数计算机系统都是按照人们预先设定的方案,执行人们指定的步骤,才能得到相应的结果。每个步骤都是一条计算机指令,当我们要计算机完成一个特定任务时,应事先把一系列指令输入计算机,然后计算机执行这些指令所规定的每一个步骤,直至任务完成。这些指令的集合我们就称之为程序。计算机程序就是计算机为完成一个特定任务而执行的一系列指令集合。描述这些指令用的是程序设计员和计算机都能识别的计算机程序设计语言(简称计算机语言或编程语言)。因此,程序设计语言是人与计算机之间进行通信的工具,是一组记号和一些用以组合这些记号的规则。程序设计语言是随着计算机不断进步而发展的。纵观其历史,可以把计算机程序设计语言按其特点、应用范围和使用方式分为以下几类。

1.1.1 低级语言

低级语言有机器语言与汇编语言,它们都是针对特定的计算机而设计的语言。即在某一指令集 CPU 的机器上可运行的程序在另一指令集 CPU 的机器上可能不能运行,程序可移植性差。因此,这两种语言又称为面向机器的语言。

1. 机器语言

机器语言是 CPU 的指令系统,它是 CPU 可以识别的一组由 0 和 1 构成的指令码。用机器语言编写的程序可以被计算机直接理解和执行。但是,它们不直观、难记、难理解、不易查错、开发周期长,要求用户必须具有较多的计算机方面的知识,熟悉机器内部结构,并且不同指令集的 CPU 机器语言也不尽相同,使得程序可移植性差。

2. 汇编语言

为了减轻程序员的负担,人们使用一些用于帮助记忆的符号来代替机器语言中 0 与 1 的指令。如用 ADD, SUB 等英语单词或缩写来代替用二进制数字来书写机器指令代码。由这些助记符组成的指令系统称为汇编语言。汇编语言较机器语言更易于人们的理解与交流,也方便了程序编制与修改。但汇编指令与机器指令基本一一对应,因此,不同指令集

CPU 的汇编语言也不尽相同, 所以汇编语言仍是低级语言。用汇编语言编写的程序实际上还是要被汇编成机器语言后才能被计算机直接执行。

1.1.2 高级语言

1. 面向过程的语言

为了克服低级语言的弱点, 从 20 世纪 50 年代起, 相继研制出了 FORTRAN, C, Pascal, BASIC 等高级语言。高级语言是一种类似自然语言和数学公式的程序设计语言, 它们独立于计算机, 所以, 程序的可移植性好。另外高级语言的语句功能强, 通常一个语句相当于许多条机器指令, 使编程效率大为提高。使用高级语言, 程序员可以不关心机器的内部结构, 把主要精力放在解决问题的思路和方法上, 大大减轻了程序员的负担。使用这一时期的高级语言, 程序员注意力集中在算法(用计算机在有限步骤内解决问题的方法)的描述上。程序员不仅要告诉计算机“做什么”, 而且还要告诉计算机“如何做”。用这类编程语言编程, 通常把解题的过程看成是数据加工的过程。正是由于用这些语言来解决问题时, 需要详细地描述解题过程, 因此, 这类语言也可称为“面向实现的语言”或“过程化”语言。

2. 面向对象的语言

为了更好地描述与解决现实中的问题, 人们又形成了一种面向对象的程序设计思想。面向对象的程序设计(OOP, Object-Oriented Programming)方法和编程技术不同于面向过程的程序设计的一个不同之处是: 它把要处理的数据以及处理这些数据的算法“封装”在一起, 用于抽象地描述现实世界的对象, 所以, 这类语言又可称为“面向现实”的语言。以面向对象的观点来观察世界, 世界上所有的事物都可以看成是对象, 每个对象都有其特征。具有相同特征的对象同属一个类。对象的特征分为静态特征(属性)和动态特征(方法)。对象的属性用具有一定数据结构的数据来表示, 对象的方法用施加于这些数据之上的操作来表示。这样对问题的解决就变成对问题的描述, 即设置对象的属性和调用对象的某个方法。这种思维方法与人们对现实世界的认识相一致, 比较容易被人们理解和掌握。

面向对象的程序设计方法与编程技术另一个不同于面向过程的程序设计之处是: 程序设计人员在进行面向对象的程序设计时, 不再是为了描述解题过程, 代码从第一行一直编到最后一行, 而是考虑如何利用类来创建对象, 然后设置对象的属性, 调用对象的方法, 利用对象来简化程序设计。对象的创建既可以使用外部提供的类, 也可以是用户自己定义的类, 甚至可以用一个现成的类为基础, 通过“继承”来获得一个满足用户需求的类。这样提高代码的可重用性, 有利于编程效率的提高。

在面向对象程序设计中, 对象之间的联系是通过事件传递消息来实现的。所谓“事件”是对象能够识别的一个外部刺激。当事件发生时, 接收该事件的对象将依据接收到的消息做出相应的反应, 对象反应的内容由用户根据需要编写代码。按这种模式工作的应用程序称为“事件驱动”应用程序。这决定了面向对象应用程序的执行过程也与面向过程的应用程序不同。在传统的“过程化”的应用程序中, 应用程序自身控制了执行代码的顺序执行。通常从第一行代码开始按应用程序中预定的路线执行。在执行期间, 用户无法对执行顺序(流程)进行干预。而面向对象的程序是事件驱动型。在事件驱动的应用程序中, 代码不

是按照预定的路线执行，而是根据响应事件来决定所执行的代码片段。事件可以由用户操作触发，也可以由来自操作系统或其他应用程序的消息触发，甚至由应用程序本身的消息触发。这些事件发生的顺序决定了代码执行的顺序。也就是说在程序运行过程中，用户可以根据需要决定代码的执行顺序。因此，应用程序每次运行时所经过的代码的路线都是不同的。

常用的面向对象的程序设计语言有 C++，JAVA，Visual Basic，VC++等。

1.1.3 高级语言程序的执行

高级语言方便了程序员对程序设计语言的理解，但是不能被计算机直接理解和执行。计算机只能运行机器语言程序，所以用高级语言编写的程序必须通过相应的语言处理程序翻译成机器语言程序，才能被计算机执行。一般要运行高级语言编制的程序，通常有两种办法。

(1) 解释执行：解释型语言在程序编写完毕后，按照程序编排的顺序，每遇到一条语句就翻译并执行，直到整个程序结束。由于每次运行程序都要进行翻译和执行两个步骤，所以解释型语言的执行速度都不快，并且每次执行都不能脱离编译环境。但是解释型语言方便程序的调试。

(2) 编译执行：编译型语言在程序设计完毕之后，使用语言本身提供的编译（Compile）程序和链接（Link）程序把源程序一次性编译成可执行文件（扩展名一般为“exe”）。可执行文件能脱离编译环境独立运行，执行效率高。

当前比较流行的程序设计语言多数是编译型的。也有的语言在程序调试过程通常采用解释运行，设计完成之后就可以把它编译成可执行文件，比如 Visual Basic。

1.2 类、对象、属性、方法与事件

面向对象的思想就是把整个或部分世界看成是由对象组成的。对象可以是任何概念上相对独立事物，比如一个人、一辆汽车、一个杯子、一张桌子都可以被看做是对象，甚至一件事情，一段程序也可以被看成是一个对象。这种思路与我们人类对世界的认识是相一致的。

1.2.1 类与对象

类是对象的模板，是对象的归纳和总结。即把同属一个类的对象所具有共同的特征（属性和方法）的描述提取出来，作为一个概念来研究。用专业的语言说，类是对象的抽象，而对象是类的实例。比如图 1-1 中的这些人，它们都属于同一个类——人类，他们都具有人类的共同特征，比如，“姓名”、“性别”、“身高”、“学历”、“心跳次数”、“跑”、“学习”、“吃饭”、“说话”等。但他们中的每一个人都是人类的具体成员，是一个个对象，是人类的实例。人类是在对所有人的共同特征归纳和总结的基础上抽象出来的概念，而具体的每

个人我们就称为对象。对象是实际存在的东西，不是概念上的，是类的具体表现，即类的实例。而类是对象在概念上的抽象。在面向对象的概念中，对于任何一个对象都可以用属性、方法两个方面来描述，属于同一个类的对象具有相同的属性和方法。

应注意的是，在面向对象的程序设计中，类与对象的获得与现实世界有些区别。在现实世界中，是先有具体的对象后有概念上的类。



图 1-1 这些人都属于同一个类——人类

以人类为例，人类这个概念是通过对所有具体的人（即一个个对象）进行考察，然后利用归纳和总结的方法获得的。而在面向对象的程序设计中，程序员是先根据程序设计的需要定义出一个类，然后再以这个类为模板“制造”出一个个对象。即先有概念上的类后有具体的对象。

类可以根据需要来划分的。图 1-1 中的人可以划分成一个类——人类，也可以根据需要划分成“男人”与“女人”两个类。

1.2.2 属性

属性是指一个对象所具有的静态特征。同属于一个类的对象具有相同的属性，但每个对象的属性值可能不同。图 1-1 中的每一个人都属于一个类——人类，人类的属性有“姓名”、“性别”、“身高”、“学历”、“心跳次数”等。图中的每个人（对象）都有这些属性，但每个人的属性值可能不同。例如，图中的小孩对应的属性值为“张嘎”、“男”、“1.30”、“小学四年级”、“80 次/秒”，提着包的那位老人对应的属性值为“王英”、“男”、“1.70”、“博士”、“60 次/秒”。在研究对象的属性时，应注意属性与属性值的区别。正是因为同一个类的对象具有不同的属性值才有丰富多彩的世界。

1.2.3 方法

方法是一个对象所具有的动态特征，指的是对象所具有的动作和行为。比如一个人所能执行的动作和行为有：跑、学习、吃饭、说话等。那么这些行为和动作就是这个人（对象）的方法。

1.2.4 事件

事件是指对象能够识别并做出反应的外部刺激。例如，上课铃响了、天下雨了、口渴了等，都是人所能识别并做出反应的事件。事件的主要作用是用来传递信息的。比如：上课铃响了这个事件，就是向人（对象）传递上课时间到的信息，人受到该事件的刺激所做出的反应，我们称之为事件反应，在应用程序中称为事件过程。

同一个对象的属性、方法、事件是相互联系、相互作用的。例如，当“张嘎”这个孩子（对象）听到上课铃响时（事件），知道上课时间到了（消息），于是向教室跑（方法），同时心跳次数（属性）加快 100 次/秒。其中向教室跑、心跳次数增加是对象在接收到上课铃响时事件时的反应，是事件过程。

1.2.5 Windows 操作系统中的对象

Windows 下的应用程序通常由一个或多个窗口组成，每个窗口内又有命令按钮、文本框等控件。这些窗体和控件都可以看做是对象，都有属性、方法和事件。以我们所熟悉的命令按钮为例，在它的上面有标题，比如“确定”，这标题就是命令按钮的属性，标题的内容（“确定”）就是命令按钮标题属性的属性值。命令按钮的方法有：移动和显示焦点（按钮中有虚线的方框）等。命令按钮还有预定义对某些外部事件的响应，比如：用户的单击按钮就是一个事件。当用户把鼠标移到按钮并按下鼠标左键时（该动作称为单击按钮），不仅会执行相应操作（比如进行计算、又打开另一个窗体等），还会使该按钮看上去像是被按下并释放一样。用户单击按钮所执行的操作称为 Click 事件过程。将代码写入 Click 事件过程，就可实现用户在单击按钮时所希望的操作。

1.3 Visual Basic 简介

1.3.1 Visual Basic 特点

Microsoft Visual Basic 提供了开发 Microsoft Windows 应用程序的最迅速、最简捷的方法。不论是 Microsoft Windows 应用程序的资深专业开发人员还是初学者，Visual Basic 都为他们提供了整套工具，以方便开发应用程序。

何谓 Visual Basic？“Visual”中文含义是“可视化”，指的是开发图形用户界面（GUI）的方法。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对象“画”到屏幕上即可。如果已使用过诸如 Windows 附件中的“画图”之类的绘图程序，则实际上已掌握了创建用户界面的必要技巧。

“Basic”指的是 BASIC（Beginners All-Purpose Symbolist Instruction Code）语言，是一种在计算机技术发展历史上应用得最为广泛的语言。Visual Basic 在原有 BASIC 语言的基础上进一步发展，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建立实用的应用程序。

使用 Visual Basic 语言，不仅仅用于编制常规 Windows 下的应用程序，还可以用于其他领域。例如 Visual Basic Scripting Edition（VBScript）是广泛使用的脚本语言，它是 Visual Basic 语言的子集，用于网页设计。还可以使用嵌入式 Visual Basic 语言 Visual Basic For Application（VBA）对一些流行软件（如 Word，Excel，AutoCAD 等）进行二次开发。这

样, 在学习 Visual Basic 中得到的经验可应用到所有这些领域中。

从开发个人或小组使用的小工具, 到大型企业应用系统, 甚至通过因特网的遍及全球分布式应用程序, 都可在 Visual Basic 提供的工具中各取所需。

1.3.2 Visual Basic 版本简介

Visual Basic 有三种版本, 可以满足不同的开发需要。

1. Visual Basic 学习版

Visual Basic 学习版使编程人员轻松开发 Windows 和 Windows NT 的应用程序。该版本包括所有的内部控件以及网格、选项卡和数据绑定控件。学习版提供的文档有 Learn Visual Basic Now CD 和包含全部联机文档的 Microsoft Developer Network CD。

2. Visual Basic 专业版

专业版为专业编程人员提供了一整套功能完备的开发工具。该版本包括学习版的全部功能及 ActiveX 控件, 因特网 Information Server Application Designer, 集成的 Visual Database Tools, Data Environment, Active Data Objects, Dynamic HTML Page Designer。专业版提供的文档有 Visual Studio Professional Features 手册和包含全部联机文档的 Microsoft Developer Network CD。

3. Visual Basic 企业版

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及 Back Office 工具, 例如 SQL Server、Microsoft Transaction Server, 因特网 Information Server, Visual SourceSafe, SNA Server, 以及其他。企业版包括的印刷文档包括 Visual Studio Enterprise Features 手册以及包含全部联机文档的 Microsoft Developer Network CD。

1.3.3 Visual Basic 的安装

在安装 Visual Basic 之前, 必须确认计算机是否满足最低的安装要求。为运行 Visual Basic, 还必须在计算机上安装相应的硬件和软件系统。这些系统要求包括:

- (1) Microsoft Windows NT 95 或更高版本, 或 Microsoft Windows NT Workstation 4.0 (推荐 Service Pack 3) 或更高版本。
- (2) 486DX/66 MHz 或更高的处理器 (推荐 Pentium 或更高的处理器), 或任何运行于 Microsoft Windows NT Workstation 的 Alpha 处理器。
- (3) 一个 CD-ROM 驱动器。
- (4) Microsoft Windows 支持的 VGA 或分辨率更高的监视器。
- (5) 16 MB RAM for Windows 95, 32 MB RAM for Windows NT Workstation。
- (6) 鼠标或其他定点设备。

用户执行光盘中安装程序 (Setup.exe) 来完成 Visual Basic 的安装。安装向导将引导用户, 逐步将 Visual Basic 以及其他产品部件从 CD-ROM 安装到硬盘上。为使用 Visual Basic 的帮助, 用户还必须安装 Microsoft Developer Network CD (MSDN) 中的文档。MSDN Library

是开发人员的重要参考资料, 包含了容量为 1 GB 的编程技术信息, 包括示例代码、文档、技术文章、Microsoft 开发人员知识库、以及用户在使用 Microsoft 公司的技术来开发解决方案时所需要的其他资料。用户可以选择只安装 Visual Basic 文档和示例程序到您的计算机上。注意: 不能直接将 CD-ROM 上的文件复制到硬盘, 然后从硬盘运行 Visual Basic。必须使用安装程序将文件解压缩并安装到合适的目录中。

1.4 Visual Basic 基础

1.4.1 启动 Visual Basic

从 Windows 启动 Visual Basic 有以下两种方法。

方法 1:

- (1) 单击任务条上的“开始”按钮;
- (2) 选择“程序”, 接着选取“Microsoft Visual Basic 6.0”。

方法 2:

- (1) 使用“Windows 资源管理器”寻找 Visual Basic 可执行文件;
- (2) 双击 Visual Basic 图标。

也可以创建一个 Visual Basic 快捷键, 并双击该快捷键。

当启动 Visual Basic 时, 可以见到 Visual Basic 新建工程对话框, 如图 1-2 所示。选择“标准 EXE”图标, 单击打开按钮, 就可以见到图 1-3 界面。

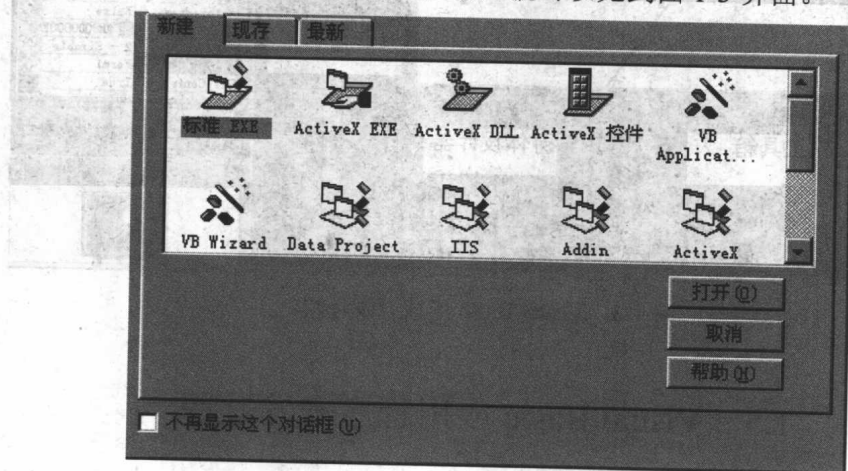


图 1-2 新建工程对话框

1.4.2 什么是集成开发环境(IDE)

Visual Basic 的工作环境常常是指集成开发环境 (IDE, Integrated Development

Environment), 这是因为它在一个公共环境里集成了许多不同的功能, 例如, 设计、编辑、编译和调试。在大多数传统开发工具中, 每个功能都是一个独立的程序, 并都有各自不同的界面。Visual Basic 的集成开发环境是由菜单、工具栏和各种窗口(如属性窗口、工程管理器窗口、窗体)等组成。如图 1-3 所示。菜单栏可以全面地控制和管理程序员的编程活动; 通过工具栏的不同按钮, 程序员能够迅速地实现菜单上的大部分功能; 窗体是用户进行 Visual Basic 应用程序开发的基本模块, 可以通过使用工具箱向窗体添加控件; “工程管理器”窗口能够显示程序员正在工作的工程和该工程的不同部分; 在“属性窗口”中, 程序员可以观察并设置控件、窗体和模块的属性; 在“窗体布局”窗口中, 程序员可以观察和操作窗体在屏幕上的位置。集成开发环境的作用就像工人的工作间一样, 里面的开发工具应有尽有, 方便了程序员开发应用程序的需要。

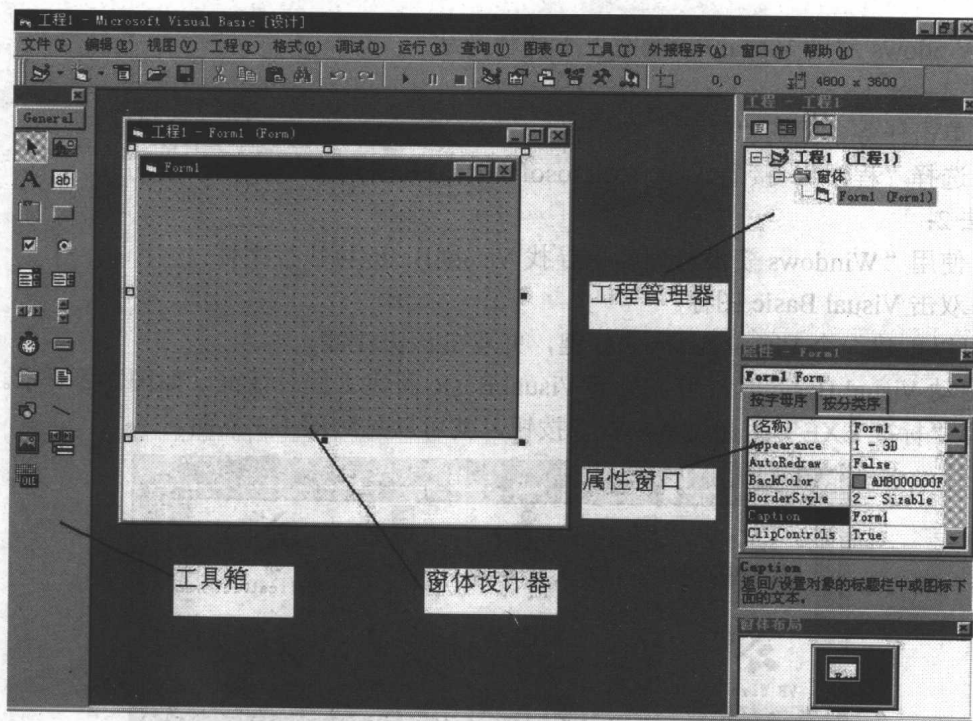


图 1-3 Visual Basic 集成开发环境

1.5 Visual Basic 应用程序开发过程

本节描述使用 Visual Basic 所需要的术语和技巧, 并且一步一步地指导读者创建一个简单的应用程序。

创建 Visual Basic 应用程序有五个主要步骤。

(1) 新建一个工程。