

# Visual Basic 编程指南

田文胜 刘阳 学勤 编著



清华大学出版社

# Visual Basic 编程指南

田文胜 刘 阳 学 勤 编著

清华大学出版社

## 内 容 简 介

Visual Basic 语言是目前最为流行和通用的程序设计语言。本书结合作者多年的教学实践和编程经验,从最基本的程序设计基础知识讲起,主要包括 VB 概述、窗体、常用控件、Windows 通用控件、菜单、文件处理、SQL 查询语言、数据库编程、图形的处理、网络编程、错误处理与调试、安装程序与帮助文件、多媒体、数据输出与打印、面向对象程序设计、语言基础、Win32 API 等内容。

本书语言通俗易懂,实例丰富实用,对 VB 语言的编程技术进行了详细、生动的讲解,着重培养编程初学者解决实际问题的能力。

本书主要适合没有任何编程经验的初学者使用,也可作为高等学校、高等职业学校、成人高校 Visual Basic 程序设计课程的教材和教学参考书。

**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 编程指南/田文胜,刘阳,学勤编著.北京:清华大学出版社,2003

ISBN 7-302-06249-8

I.V... II.①田... ②刘... ③学... III.BASIC 语言-程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(2003)第 002695 号

**出 版 者:** 清华大学出版社(北京清华大学学研大厦,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

**责任编辑:** 苗建强

**印 刷 者:** 清华大学印刷厂

**发 行 者:** 新华书店总店北京发行所

**开 本:** 787×1092 1/16 **印 张:** 45.75 **字 数:** 1051 千字

**版 次:** 2003 年 2 月第 1 版 2003 年 2 月第 1 次印刷

**书 号:** ISBN 7-302-06249-8/TP·3742

**印 数:** 0001~4000

**定 价:** 68.00 元

# 前 言

Visual Basic 不仅是 Visual Basic 编程语言，Visual Basic 编程系统、Microsoft Excel 的 Applications Edition、Microsoft Access 和 Windows 的许多其他应用程序都使用这一语言。Visual Basic Scripting Edition (VBScript) 是广泛使用的脚本语言，它是 Visual Basic 语言的子集。这样，在学习 Visual Basic 中得到的经验可应用到所有这些领域中。

从开发个人使用或小组使用的小工具，到大型企业应用系统，甚至通过 Internet 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。

在这里，无论你是刚接触 VB，或者一点也不懂 VB，都可以通过 VB 入门系列教程学习 VB，掌握 VB。

本书是一本全面介绍 VB 语言特点和应用技术的实用教材，主要面向初学者和自学者，由浅入深、循序渐进地讲解了 VB 的概念和编程技术，使广大用户既能够全面系统地了解 VB 技术的特点，又能掌握最新的实用技术。

全书主要内容包括：VB 概述、窗体、常用控件、Windows 通用控件、菜单、文件处理、SQL 查询语言、数据库编程、图形处理、网络编程、错误处理与调试、安装程序与帮助文件、多媒体、数据输出与打印、面向对象程序设计、VB 语言基础、Win32 API 等。

虽然在写这本书时，我们抱着比以往更严谨的态度，对书中的每个概念、每个说明都一再推敲，一再求证，但仍然无法满足每位读者的需求。因此，书中若有任何错误或用词不当，希望读者不吝赐教；而当书中的说明让您感到疑惑时，欢迎您来信询问，我们一定给您满意的答复。请发电子邮件到：[myran@163.net](mailto:myran@163.net)。

最后感谢所有帮助作者完成此书的朋友和同仁。本书由田文胜、刘阳、学勤、李诚、张艳伟、张志新、李盛、陈辉、郭志敏、孙亮、赵春民、张军、丁武昌、刘德春、吴振刚、王秀稳、孔宪英、杨自军、王晓堂、陈启顺、刘先忠、王春光、张永英等共同编写，程序员刘爱青、肖占国、张志民提供技术支持，在此表示由衷的感谢。正是他们的积极努力和全身心的投入，才使该书能早日与读者见面！同时向清华大学出版社金地公司编辑苗建强表示由衷的感谢，本书的成功出版，和苗建强先生勤奋而高效的工作是分不开的。

编 者

2002 年 10 月

# 目 录

第 1 章 概述 .....	1
1.1 Visual Basic 的特点 .....	1
1.2 初识 Visual Basic 6.0 .....	3
1.3 Visual Basic 6.0 开发环境组成部分 .....	5
1.3.1 Form 窗口 .....	5
1.3.2 标题栏 .....	6
1.3.3 菜单栏 .....	6
1.3.4 工具栏 .....	18
1.3.5 工具箱 .....	19
1.3.6 工程资源管理器 .....	22
1.3.7 属性窗口 .....	24
1.3.8 属性设置 .....	25
1.3.9 “窗体布局”窗口 .....	26
1.3.10 “立即”、“本地”和“监视”窗口 .....	27
1.3.11 “调色板”窗口 .....	27
1.3.12 “代码”窗口 .....	28
1.3.13 “对象浏览器”窗口 .....	29
1.3.14 “菜单编辑器”窗口 .....	30
1.3.15 程序的保存和装入 .....	31
1.4 Visual Basic 应用程序的结构与工作方式 .....	32
1.5 编写简单的 Visual Basic 应用程序 .....	33
1.5.1 保存文件 .....	37
1.5.2 编译应用程序 .....	37
第 2 章 窗体 .....	41
2.1 窗体的属性 .....	42
2.2 窗体程序设计举例 .....	45
2.3 多窗体 .....	56
2.3.1 创建 MDI 应用程序 .....	58
2.3.2 MDI 特性 .....	60

<b>第 3 章 Visual Basic 常用控件</b> .....	70
3.1 标签和文本框 .....	70
3.1.1 标签 .....	70
3.1.2 文本框 .....	73
3.2 命令按钮、复选框和单选按钮 .....	83
3.2.1 命令按钮应用程序 .....	84
3.2.2 复选框应用程序 .....	95
3.2.3 单选按钮应用程序 .....	98
3.3 图片框与图像框 .....	100
3.3.1 属性、事件和方法 .....	101
3.3.2 图形文件的装入 .....	102
3.4 列表框和组合框 .....	111
3.4.1 列表框 .....	112
3.4.2 组合框 .....	120
3.5 滚动条 .....	130
3.6 计时器 .....	136
3.7 直线和形状控件 .....	145
3.8 框架 .....	148
3.9 文件系统控件 .....	154
3.9.1 驱动器列表框 .....	154
3.9.2 目录列表框 .....	157
3.9.3 文件列表框 .....	157
<b>第 4 章 Windows 通用控件</b> .....	168
4.1 ImageList 控件 .....	168
4.1.1 设置 ImageList 的图像 .....	169
4.1.2 读取 ImageList 的图像 .....	172
4.1.3 动态加载图像 .....	174
4.1.4 ImageList 控件的例程 .....	175
4.2 ToolBar 控件 .....	179
4.2.1 ToolBar 与 ImageList 的链接 .....	180
4.2.2 为工具栏加入程序 .....	182
4.2.3 Button 对象的相关属性设置 .....	185
4.2.4 动态设置按钮 .....	186
4.2.5 ToolBar 控件的应用实例 .....	187
4.3 StatusBar 控件 .....	189
4.3.1 几个重要的属性 .....	189

4.3.2	处理窗格单击事件 .....	192
4.3.3	StatusBar 控件的几个例子 .....	192
4.4	ProgressBar 控件 .....	194
4.5	UpDown 控件 .....	195
4.5.1	自动设置伙伴控件 .....	196
4.5.2	设置 UpDown 控件的最小值和最大值 .....	196
4.5.3	UpDown 控件的事件 .....	196
4.5.4	UpDown 控件使用举例 .....	197
4.6	MSFlexGrid 控件 .....	198
4.6.1	MSFlexGrid 控件的重要属性 .....	199
4.6.2	MSFlexGrid 控件的重要方法 .....	200
4.6.3	MSFlexGrid 控件的重要事件 .....	202
4.6.4	MSFlexGrid 控件的几个例子 .....	203
4.7	ListView 控件 .....	207
4.7.1	ListView 控件的四种外观 .....	208
4.7.2	ListView 控件的组成对象及代表性属性成员 .....	210
4.7.3	如何在 ListView 中显示“列表”数据 .....	211
4.7.4	如何在 ListView 中加入图标 .....	212
4.7.5	如何存取 ListView 数据项内容 .....	212
4.7.6	如何读取用户所选取的数据项 .....	213
4.7.7	多列的 ListView .....	216
<b>第 5 章</b>	<b>Visual Basic 的菜单 .....</b>	<b>220</b>
5.1	菜单编辑器 .....	220
5.2	建立菜单 .....	224
5.3	把代码连接到菜单上 .....	227
5.4	快捷菜单 .....	231
<b>第 6 章</b>	<b>文件处理 .....</b>	<b>243</b>
6.1	文件处理概述 .....	243
6.1.1	FileSystemObject 对象及其子对象 .....	243
6.1.2	Folder 对象与文件夹的浏览 .....	244
6.1.3	Files 集合对象和 File 对象 .....	245
6.1.4	SubFolders 集合对象 .....	246
6.2	Folder 对象和 File 对象的属性应用 .....	247
6.2.1	如何知道某个文件夹所占用的磁盘空间 .....	247
6.2.2	如何显示文件夹的所有子文件夹 .....	248
6.3	文件与文件夹的存取 .....	253

6.3.1	TextStream 对象与文件的读写.....	253
6.3.2	FileSystemObject 对象与文件的操作.....	255
6.3.3	FileSystemObject 对象与文件夹的操作.....	258
6.4	Drive 对象及磁盘驱动器的操作.....	260
<b>第 7 章</b>	<b>SQL 查询语言</b> .....	<b>263</b>
7.1	select 语句.....	263
7.1.1	指定欲检索的列.....	264
7.1.2	搜索条件.....	265
7.1.3	排序顺序.....	266
7.1.4	几个重要的关键字的应用.....	267
7.1.5	子查询.....	267
7.1.6	几个重要的条件.....	268
7.1.7	有关常量、函数和表达式的应用.....	270
7.1.8	Group By 子句.....	275
7.1.9	多表查询.....	275
7.2	使用 DML 修改表数据.....	281
7.2.1	Insert 语句.....	281
7.2.2	Update 语句.....	282
7.2.3	Delete 语句.....	283
<b>第 8 章</b>	<b>数据库编程</b> .....	<b>284</b>
8.1	关于数据库的基本知识.....	284
8.1.1	什么是数据库.....	284
8.1.2	关系数据库模型.....	285
8.1.3	几个基本术语.....	285
8.1.4	优秀数据库所具有的特点及设计原则.....	286
8.2	数据访问控件.....	287
8.2.1	数据访问控件.....	287
8.2.2	Data 控件.....	290
8.2.3	ADO 数据控件.....	308
8.3	ActiveX 数据对象.....	322
8.3.1	Visual Basic 的数据访问接口.....	322
8.3.2	ADO 操作数据的步骤.....	325
8.3.3	Connection 对象.....	326
8.3.4	Recordset 对象.....	331
8.3.5	操作字段.....	336
8.3.6	Command 对象.....	337



---

8.3.7	Parameter 对象.....	338
8.3.8	错误信息.....	339
8.3.9	应用 ADO 对象操作数据库.....	340
8.4	学生信息管理系统.....	368
<b>第 9 章</b>	<b>图形的处理.....</b>	<b>410</b>
9.1	图形处理的基础知识.....	410
9.1.1	图形功能.....	410
9.1.2	图形方法和图形控件.....	410
9.1.3	Visual Basic 的坐标系.....	411
9.2	图形方法实现.....	412
9.2.1	重绘 (Redraw).....	412
9.2.2	AutoRedraw 属性.....	412
9.2.3	Paint 事件.....	412
9.2.4	清除绘图区域.....	413
9.2.5	颜色设置.....	414
9.2.6	文本绘制.....	415
9.2.7	TextWidth 和 TextHeight 方法.....	416
9.2.8	字体的选择.....	417
9.2.9	直线的绘制.....	417
9.2.10	DrawStyle 属性.....	419
9.2.11	点的绘制.....	420
9.2.12	Circle 方法.....	421
<b>第 10 章</b>	<b>Visual Basic 网络编程.....</b>	<b>455</b>
10.1	网络编程的基础知识.....	455
10.2	使用 Microsoft 提供的 ActiveX 控件进行网络通信.....	457
10.2.1	用 Internet Transfer 控件下载文件.....	457
10.2.2	使用 WinSock 控件进行网络通信.....	469
<b>第 11 章</b>	<b>错误处理与调试.....</b>	<b>483</b>
11.1	错误类型.....	483
11.1.1	编译错误.....	483
11.1.2	运行错误.....	485
11.1.3	逻辑错误.....	489
11.1.4	分析错误类型的意义.....	489
11.2	错误处理.....	490
11.3	调试功能和调试工具.....	494

第 12 章 安装程序与帮助文件 .....	499
12.1 安装程序 .....	499
12.1.1 安装程序的综述 .....	499
12.1.2 安装程序的实现 .....	499
12.2 帮助文件 .....	506
12.2.1 帮助文件的综述 .....	506
12.2.2 帮助文件的实现 .....	507
第 13 章 多媒体 .....	512
13.1 多媒体元素 .....	512
13.2 多媒体 MCI 控件 .....	514
13.2.1 在代码中使用多媒体控件 .....	514
13.2.2 使用动画控件 .....	517
13.2.3 设置多媒体控件的时间格式 .....	517
13.2.4 从代码中控制多媒体控件 .....	518
13.2.5 停止和暂停多媒体控件 .....	519
13.2.6 关闭多媒体控件 .....	519
13.3 MCI 控件属性 .....	520
13.4 MCI 控件程序举例 .....	530
13.5 文字处理 .....	548
第 14 章 数据输出与打印 .....	556
14.1 Print 方法 .....	556
14.2 与 Print 方法有关的函数 .....	559
第 15 章 面向对象程序设计 .....	586
15.1 面向对象程序设计原理 .....	586
15.1.1 面向对象综述 .....	586
15.1.2 面向对象概念 .....	587
15.1.3 面向对象程序设计方法 .....	588
15.2 Visual Basic 的面向对象特性 .....	589
15.2.1 操作 Visual Basic 中的对象 .....	589
15.2.2 对象编程的经典例子——MDI 窗体 .....	611
15.3 创建自己的类及对象 .....	620
15.3.1 创建自己的 Visual Basic 对象 .....	620
15.3.2 建立自己的类 .....	621

第 16 章 VB 语言基础.....	631
16.1 基本概念.....	631
16.1.1 标识符.....	631
16.1.2 运算符.....	631
16.1.3 书写规则.....	634
16.2 常量、变量和数据类型.....	634
16.2.1 常量.....	634
16.2.2 变量.....	636
16.2.3 数据类型.....	638
16.3 数组.....	640
16.3.1 固定数组.....	641
16.3.2 动态数组.....	642
16.3.3 自定义类型.....	642
16.4 顺序结构.....	644
16.4.1 赋值语句.....	645
16.4.2 注释语句.....	645
16.4.3 End 语句.....	645
16.5 选择结构.....	647
16.5.1 If 语句.....	647
16.5.2 Select Case 语句.....	651
16.6 循环结构.....	654
16.6.1 For...Next 循环.....	654
16.6.2 While 循环.....	665
16.6.3 Do 循环.....	666
16.6.4 中途跳出语句.....	670
16.7 过程.....	670
16.7.1 Sub 过程.....	671
16.7.2 Function 函数.....	676
16.7.3 Property 属性过程和 Event 事件过程.....	685
16.8 常用函数.....	688
16.8.1 数学函数.....	688
16.8.2 字符串函数.....	689
16.8.3 转换函数.....	690
16.8.4 判断函数.....	690
16.8.5 其他函数.....	691

第 17 章 Win32 API .....	694
17.1 Win32 API 概述 .....	694
17.1.1 什么是 API .....	694
17.1.2 静态与动态链接 .....	695
17.1.3 如何找到恰当的 API .....	695
17.1.4 安装 API Viewer 外接程序 .....	696
17.1.5 使用 API Viewer 插入声明 .....	697
17.2 API 声明 .....	701
17.2.1 作用域 .....	701
17.2.2 声明类型 .....	702
17.2.3 API 名称 .....	702
17.2.4 库 .....	703
17.2.5 别名 .....	703
17.2.6 参数 .....	704
17.3 Windows 消息 .....	705
17.4 例子程序 .....	706
17.4.1 如何制作浮动窗体 .....	706
17.4.2 如何读取系统的存储器容量及可用存储器 .....	707
17.5 API 快速参考 .....	709
17.5.1 参数转换表 .....	709
17.5.2 虚拟键代码 .....	710
17.5.3 Windows 消息 .....	711

# 第 1 章 概 述

Microsoft Windows 软件的出现,为 PC 用户提供了一个直观的、图形丰富的工作环境,图形用户界面 (GUI) 则使应用程序更易于学习和使用。用户只要简单地用鼠标按钮单击“菜单”中的命令就可以执行指令的操作,而不必键入复杂的命令。屏幕上的多窗口可以让用户同时运行多个程序,并可以通过对话框输入所需要的信息供用户作出选择。

毋庸置疑,对于用户来说,Windows 的环境相当出色。但对程序员来说,其难度反而增大了。为了编写在 Windows 下运行的程序,必须建立相应的窗口、菜单、字形、对话框等各种“构件”。因此,程序员一方面为 Windows 提供了编写图形界面友好的应用程序平台而感到欣喜,另一方面却为编程工作越来越复杂而苦恼。可视化程序设计语言正是在这种背景下应运而生。

Visual Basic 是 Microsoft 公司在 1991 年推出的可视化 Basic 语言,在语言功能方面,与以前的 Basic 语言基本兼容,但功能方面更为强大,使用起来更加方便。随着版本的改进,Visual Basic 已逐渐成为简单易学、功能强大的编程工具。不但在一般程序设计方面比传统的程序设计语言具有明显的优势,而且可作为多媒体创作工具,支持程序跳转,即从静态目标跳转到音乐、声音、动画、图形等目标;同时具有脱机播放动画的能力,并能较好地控制动画的清晰度、运动速度和方向、动画过渡技术等。

Visual Basic 编程系统使用一种十分巧妙的方法将 Windows 编程的复杂性“封装”起来。它综合运用了 Basic 语言和新的可视化设计工具,既没有牺牲 Windows 所特有的优秀性能和图形工作环境,同时又提供了编程的简易性。Visual Basic 通过图形对象来设计应用程序。图形对象的建立非常简单,只需要为数不多的几行程序就可以控制这些图形对象。

Visual Basic 是第一批采用事件驱动编程机制的计算机语言之一。事件驱动是一种适合于图形用户界面 (GUI) 的编程方式。传统的编程是面向过程、按规定顺序进行的,程序设计人员总是要关心什么时候发生什么事情。对于现代的计算机应用来说,必须能让用户操作程序的执行,而这实际上就是事件驱动程序所要解决的问题。

## 1.1 Visual Basic 的特点

Visual Basic 是一种可视化的、面向对象和采用事件驱动方式的结构化高级程序设计语言,可用于开发 Windows 环境下的各类应用程序,它简单易学、效率高、功能强大。在 Visual Basic 环境下,利用事件驱动的编程机制、新颖易用的可视化设计工具,使用 Windows

内部的应用程序接口 (API) 函数以及动态链接库 (DLL)、动态数据交换 (DDE)、对象的链接与嵌入 (OLE)、开放式数据连接 (ODBC) 等技术, 可以高效、快速地开发出 Windows 下功能强大、图形界面丰富的应用软件系统。

### 1. 可视化编程

在用传统程序设计语言来设计程序时, 都是通过编写计算来设计用户界面。在设计过程中看不到界面的实际显示效果, 必须编译后运行程序才能观察。如果对界面的效果不满意, 还要回到程序中去修改。Visual Basic 提供了可视化设计工具, 把 Windows 界面设计的复杂性“封装”起来, 开发人员不必为界面设计而编写大量程序代码。只需要按设计要求的屏幕布局, 用系统提供的工具, 在屏幕上画出各种“部件”, 即图形对象, 并设置这些图形对象的属性, Visual Basic 即自动产生界面设计代码, 程序设计人员只需要编写相应能实现程序功能的那部分代码, 从而可以大大提高程序设计的效率。

### 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 (OOP), 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, 不必编写建立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, 都是可视的。

### 3. 结构化程序设计语言

Visual Basic 是在 Basic 语言的基础上发展起来的, 具有高级程序设计语言的语句结构, 接近于自然语言和人类的逻辑思维方式, 其语句简单易懂。其编辑器支持彩色代码, 可自行进行语法错误检查, 同时具有功能强大且使用灵活的调试器和编译器。

Visual Basic 是解释型语言, 在输入代码的同时, 解释系统将高级语言分解翻译成计算机可以识别的机器指令, 并判断每个语句的语法错误。在设计 Visual Basic 程序的过程中, 随时可以运行程序, 而在整个应用程序设计好之后, 可以编译生成可执行文件 (.EXE), 脱离 Visual Basic 环境, 直接在 Windows 环境下运行。

### 4. 事件驱动编程机制

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来实现指定的操作。

Visual Basic 使用了若干个微小的子程序, 即过程, 这些过程分别面向不同的对象, 由用户操作引发某个事件来驱动某种特定的功能, 或者由事件驱动程序调用通用过程来执行指定的操作。这样可以方便编程人员, 提高效率。

### 5. 访问数据库

Visual Basic 系统具有很强的数据库管理功能。利用数据控件和数据库管理窗口, 可以直接建立或处理 Microsoft Access 格式的数据库, 并提供了强大的数据存储和检索功能。同时, Visual Basic 还能直接编辑和访问其他可支持的外部数据库。

Visual Basic 提供开放式数据连接 (Open DataBase Connectivity), 即 ODBC 功能, 可通过直接访问或建立连接的方式使用并操作后台大型网络数据库, 如 SQL Server、Oracle 等。在应用程序中, 可以使用结构化查询语言 SQL 数据标准, 直接访问 Server 上的数据库, 并提供了简单的面向对象的库操作指令和多用户数据库访问的加锁机制和网络数据库的 SQL 编程技术, 为单机上运行的数据库提供了 SQL 网络接口, 以便在分布式环境中快速而有效地实现客户/服务器 (Client/Server) 方案。

#### 6. 动态数据交换 (DDE)

利用动态数据交换 (Dynamic Data Exchange) 技术, 可以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, 使两种完全不同的应用程序可以交换数据、进行通信, 在 Windows 环境中为多个应用程序之间以 Client/Server 方式建立起一条动态数据链路。当原始数据变化时, 可以自动更新链接的数据。Visual Basic 提供了动态数据交换的编程技术, 可以在应用程序中实现与其他 Windows 应用程序的动态数据交换, 从而实现了不同应用程序之间的通信。

#### 7. 对象的链接与嵌入 (OLE)

对象的链接与嵌入 (Object Linking and Embedding) 将每个应用程序都看作一个对象 (Object), 将不同的对象链接 (Link) 起来, 再嵌入 (Embed) 某个应用程序中, 从而可以得到具有声音、影像、图像、动画、文字等各种信息的集合式文件。OLE 文件把多个应用程序合为一体, 将每个应用程序看作一个对象进行链接和嵌入, 是一种应用程序一体化的技术。利用 OLE 技术, 可以方便地建立复合式文档 (Compound Document), 这种文档由来自多个不同应用程序的对象组成, 文档中的每个对象都与原来的应用程序相联系, 并可执行与原应用程序完全相同的操作。

#### 8. 动态链接库 (DLL)

Visual Basic 是一种高级程序设计语言, 不具备低级语言的功能, 访问机器硬件的操作不太容易实现。但它可以通过动态链接库 (Dynamic Linking Library) 技术将 C/C++ 或汇编语言编写的程序加入到 Visual Basic 应用中, 可以像调用内部函数一样调用其他语言编写的函数。此外, 通过动态链接库, 还可以调用 Windows 应用程序接口 (API) 函数, 实现 SDK (Software Development Kit, 软件开发工具包) 所具有的功能。

## 1.2 初识 Visual Basic 6.0

单击 Windows 98 的“开始”按钮, 在弹出的菜单中单击“程序”菜单, 将鼠标移动到“Visual Basic 6.0 中文版”命令, 然后单击它, 就启动了 Visual Basic 6.0。如图 1.1 所示。



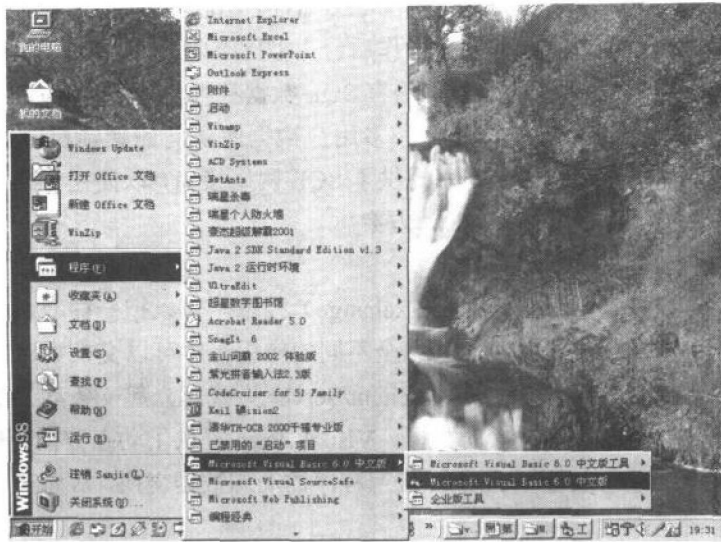


图 1.1 启动 Visual Basic 6.0 中文版

Visual Basic 6.0 中文版启动后就出现如图 1.2 所示的对话框，选取“标准 EXE”，然后用鼠标单击“打开”按钮，出现如图 1.3 所示的界面，这就是 Visual Basic 6.0 的编程环境。

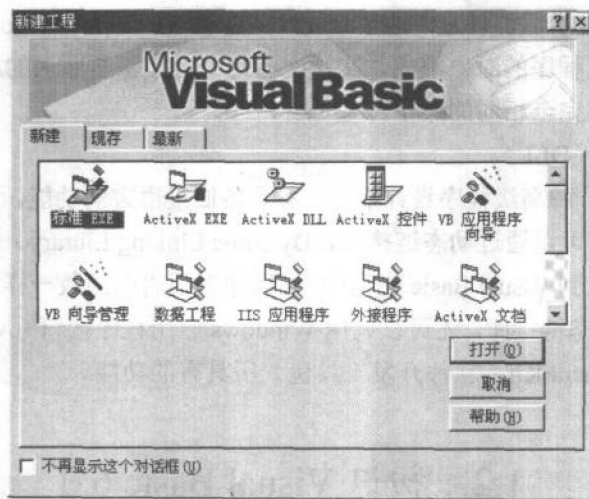


图 1.2 新建工程对话框

Visual Basic 集成开发环境（Integrated Development Environment, IDE）是提供设计、运行和测试应用程序所需的各种工具的一种工作环境。这些工具相互协调、相互补充，大大减少了应用程序的开发难度。





图 1.3 Visual Basic 的集成开发环境

Visual Basic 启动后，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、工程资源管理器窗口和属性窗口。根据需要，和 Windows 下的窗口一样，集成开发环境中的每个窗口都可以在屏幕上移动、缩小、放大或关闭。此外，Visual Basic 保存上一次使用时屏幕上各部分最后的排列方式，并作为下一次启动 Visual Basic 后的屏幕布局。

## 1.3 Visual Basic 6.0 开发环境组成部分

Visual Basic 6.0 的开发环境，在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、工程资源管理器窗口和属性窗口等，接下来逐一介绍。

### 1.3.1 Form 窗口

窗体 (Form) 是应用程序最终面向用户的窗口，对应于应用程序的运行结果。程序正是在窗体上设计和展开的，各种图形、数据、图像等都是通过窗体或窗体中的控件显示出来的。窗体上布满了小点，是供设计时控件对齐用的，运行时消失。

用户可以在如图 1.4 所示的界面中间位置上看到 Form1 窗口。它是用来生成应用程序的编辑窗口，是放置其他控件的一个容器，一般称为“窗体”。

一个程序可以拥有许多个窗体，但它们的名称必须不同，以免运行时发生错误。默认状态下窗体分别以 Form1、Form2、Form3...命名，程序员也可以创建自己的命名，以便于记忆理解每个窗体的功能和作用。