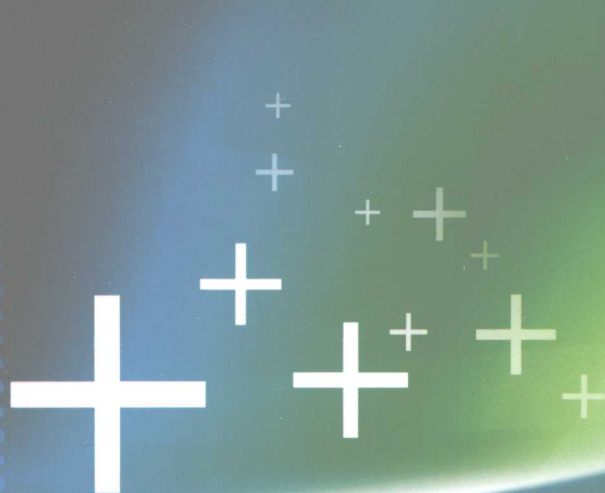
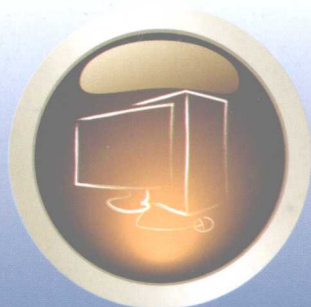




王永新 曲国军 主 编
柏晓辉 杨显辉 副主编

Visual Basic

程序设计教程



黑龙江大学出版社
HEILONGJIANG UNIVERSITY PRESS



Visual Basic

程序设计教程

王永新 曲国军 主 编
柏晓辉 杨显辉 副主编



黑龙江大学出版社
HEILONGJIANG UNIVERSITY PRESS

图书在版编目(CIP)数据

Visual Basic 程序设计教程/王永新,曲国军主编.
哈尔滨:黑龙江大学出版社,2008.10

ISBN 978-7-81129-067-7

I. V… II. ①王…②曲… III. BASIC 语言-程序设计-
教材 IV. TP312

中国版本图书馆 CIP 数据核字(2008)第 157020 号

责任编辑 刘春雷
封面设计 乐然纸尚

Visual Basic 程序设计教程

VISUAL BASIC CHENGXUSHEJI JIAOCHENG

王永新 曲国军 主 编

柏晓辉 杨显辉 副主编

出版发行 黑龙江大学出版社

地 址 哈尔滨市南岗区学府路 74 号 邮编 150080

电 话 0451-86608666

经 销 新华书店

印 刷 哈尔滨市石桥印务有限公司

版 次 2008 年 10 月 第 1 版

印 次 2008 年 10 月 第 1 次印刷

开 本 787×1092 毫米 1/16

印 张 19.75

字 数 487 千

书 号 ISBN 978-7-81129-067-7/T·10

定 价 34.00 元

凡购买黑龙江大学出版社图书,如有质量问题请与本社发行部联系调换

版权所有 侵权必究

前 言

Visual Basic 是一个可视化的面向对象的程序设计环境,是设计 Windows 应用程序的重要工具。由于它的功能强、容易掌握、使用方便,因此,从推出以来,受到广大程序设计人员的热烈欢迎。经过不断的改进和完善,到 1998 年已经推出了 Visual Basic 6.0 版,功能大大增强,编程效率和操作智能化程度明显提高。事实上,它已经成为目前许多专业人员和广大编程爱好者不可缺少的有力工具。虽然随着 Microsoft 的 .Net 战略的推广,Visual Studio. Net 随之逐渐替代早期的 Visual Studio 6.0,但 Visual Basic 6.0 不仅是计算机专业学生,而且也是非计算机专业学生学习程序设计的一门重要基础课程。目前我国大部分高等学校都以 Visual Basic 语言作为学生面向对象程序设计入门的语言,因此编写这本教材有着十分重要的意义。

程序设计语言的课程需要学生进行充分的实践练习,这就要求我们的教材能比较详细地叙述课程内容,并能提供学习指导,使学生在课后自学起来比较方便。正是基于这一目标,我们在教材的编写上针对知识点举了丰富的实例,以便学生进行掌握。

本教材的编写采用以知识点为单元的编写方式,将整个 Visual Basic 语言的内容按若干个知识点来编排、编写,使学生在此学习的基础上能系统、有序地掌握一个个完整的概念,并能灵活应用。

本书主要由王永新、曲国军、柏晓辉和杨显辉编著,王永新编写了第 1、第 3、第 9 章,曲国君编写了第 4、第 5、第 8 章,柏晓辉编写了第 2、第 4、第 7、第 11 章的内容,杨显辉编写了第 10 章。

由于时间仓促,作者水平有限,书中错误难免。我们恳请广大师生在使用教材的过程中,对本教材提出宝贵意见,以便我们作进一步的修改。对此,我们表示衷心的感谢。

编 者
2008 年 8 月

目 录

第1章 Visual Basic 6.0 概述	1	2.1.1 数值(Numeric)型数据	28
1.1 面向对象程序设计(OOP)	1	2.1.2 字符(String)型数据	29
1.1.1 面向过程与面向对象程序 设计	1	2.1.3 布尔(Boolean)型数据	29
1.1.2 对象及其属性、方法和类	2	2.1.4 日期(Date)型数据	30
1.1.3 事件、事件过程及事件 驱动	3	2.1.5 对象(Object)型数据	30
1.2 Visual Basic 简介	4	2.1.6 可变(Variant)类型数据	30
1.2.1 Visual Basic 的发展历史及 特点	5	2.2 常量与变量	30
1.2.2 Visual Basic 6.0 的运行	9	2.2.1 常量	30
1.3 Visual Basic 6.0 集成开发 环境	9	2.2.2 变量	32
1.3.1 Visual Basic 6.0 集成的开发 环境	9	2.3 表达式	35
1.3.2 Visual Basic 6.0 的模式	11	2.3.1 算术表达式	35
1.3.3 Visual Basic 6.0 的菜单 系统	11	2.3.2 字符串表达式	36
1.3.4 工具栏和工具箱	14	2.3.3 日期表达式	37
1.3.5 窗口简介	16	2.4 常用内部函数	37
1.4 Visual Basic 6.0 控件的操作	19	2.4.1 数学函数	37
1.4.1 控件对象的建立	19	2.4.2 字符串函数	38
1.4.2 控件的属性设置	19	2.4.3 日期/时间函数	39
1.5 Visual Basic 创建应用程序的 基本步骤	20	2.4.4 格式输出函数	40
1.5.1 Visual Basic 6.0 代码的 书写规范	20	2.4.5 随机数语句和函数	41
1.5.2 创建应用程序的基本 步骤	21	2.4.6 类型转换函数	42
1.6 Visual Basic 6.0 工程管理	23	2.5 程序代码编写规则	43
1.6.1 Visual Basic 应用程序的 组织	23	2.5.1 语句及语法	43
1.6.2 工程管理	24	2.5.2 代码书写规则	44
习题一	25	2.5.3 语句格式中的符号约定	44
第2章 程序设计基础	27	2.5.4 基本语句	45
2.1 标准数据类型	27	习题二	48
		第3章 赋值与输入/输出	50
		3.1 使用 Print 方法输出数据	50
		3.1.1 使用 Print 方法	50
		3.1.2 特殊打印格式	51
		3.2 窗 体	54
		3.2.1 窗体的基本属性	54
		3.2.2 窗体的事件	56
		3.2.3 窗体的方法	57
		3.2.4 焦点与 Tab 键序	58
		3.3 基本控件	59

3.3.1 控件的公共属性	59	6.2.3 数组的应用	124
3.3.2 命令按钮	61	6.3 动态数组	135
3.3.3 标签	61	6.4 For Each...Next 循环语句	138
3.3.4 文本框	64	6.5 控件数组	139
3.4 使用对话框	66	6.5.1 控件数组的概念	139
3.4.1 输入对话框	66	6.5.2 控件数组的建立	139
3.4.2 消息对话框	67	6.5.3 控件数组的使用	140
习题三	70	习题六	141
第4章 选择结构设计	73	第7章 过程	147
4.1 条件表达式	73	7.1 通用过程	147
4.1.1 关系表达式	73	7.1.1 Sub 过程	147
4.1.2 逻辑表达式	74	7.1.2 Function 过程	152
4.2 条件语句	76	7.2 参数传递	154
4.2.1 If...Then...语句和 If...Then...Else...语句	76	7.2.1 形参与实参	155
4.2.2 IIf 函数	80	7.2.2 按地址传递和按值传递	155
4.2.3 条件语句的嵌套	80	7.2.3 传递数组	158
4.3 多分支语句	83	7.3 过程的嵌套与递归调用	159
4.4 选择性控件	86	7.3.1 过程的嵌套调用	159
4.4.1 单选按钮	86	7.3.2 过程的递归调用	160
4.4.2 复选框	88	7.4 变量的作用范围	162
4.5 计时器控件	89	7.4.1 代码模块的概念	162
4.6 程序举例	92	7.4.2 变量的作用域	164
习题四	96	7.4.3 变量的生存期	166
第5章 循环结构设计	99	7.5 多窗体与 SubMain 过程	167
5.1 循环语句	99	7.5.1 多窗体处理	167
5.1.1 For...Next 循环语句	99	7.5.2 SubMain 过程	171
5.1.2 Do...Loop 循环语句	101	习题七	172
5.1.3 While...Wend 循环语句	105	第8章 文件	179
5.2 多重循环	106	8.1 文件的基本概念	179
5.3 列表框与组合框	107	8.2 文件系统控件	180
5.3.1 列表框	107	8.2.1 驱动器列表框	180
5.3.2 组合框	110	8.2.2 目录列表框	180
5.4 程序举例	112	8.2.3 文件列表框	181
习题五	115	8.3 常用的文件操作语句和函数	183
第6章 数组	121	8.3.1 与文件、文件夹操作有关的函数	183
6.1 数组的概念	121	8.3.2 与文件、文件夹操作有关的语句	188
6.1.1 数组与数组元素	121	8.4 顺序文件	191
6.1.2 下标和数组的维数	121	8.4.1 顺序文件的打开和关闭	191
6.2 数组的声明和应用	122		
6.2.1 数组声明语句	122		
6.2.2 Array 函数	123		

8.4.2 顺序文件的读写	193	9.8.3 自定义数据类型	242
8.5 随机文件	199	9.8.4 定义枚举类型	242
8.5.1 随机文件的打开和 关闭	200	9.8.5 GoTo 型控制	243
8.5.2 随机文件的读写	200	9.8.6 打印机输出	244
8.6 二进制文件	205	9.8.7 闲置循环与 DoEvents 语句	245
8.6.1 二进制文件的打开和 关闭	205	习题九	246
8.6.2 二进制文件的读写	205	第 10 章 程序调试与错误处理	248
习题八	208	10.1 Visual Basic 程序中的错误 类型	248
第 9 章 其他常用语句及控件	212	10.2 程序工作模式	249
9.1 键盘与鼠标事件	212	10.3 程序调试	250
9.1.1 键盘事件	212	10.3.1 静态检查与动态检查	250
9.1.2 鼠标事件	213	10.3.2 使用 Visual Basic 调试 工具	251
9.1.3 拖放操作	214	10.4 错误处理	254
9.2 菜单设计	215	10.4.1 错误处理的步骤	254
9.2.1 下拉式菜单	215	10.4.2 Err 对象	254
9.2.2 菜单编辑器	216	10.4.3 捕获错误语句	255
9.2.3 菜单的 Click 事件	217	10.4.4 退出错误处理语句	255
9.2.4 运行时改变菜单属性	220	习题十	256
9.2.5 弹出式菜单	220	第 11 章 数据库编程	257
9.3 工具栏和状态栏	221	11.1 数据库的基本概念	257
9.3.1 ActiveX 控件简介	222	11.1.1 关系数据库的结构	257
9.3.2 工具栏	222	11.1.2 数据访问对象模型	258
9.3.3 状态栏	225	11.1.3 结构化查询语言	259
9.4 对话框	226	11.2 可视化数据管理器	259
9.4.1 通用对话框	226	11.2.1 启动可视化数据 管理器	259
9.4.2 自定义对话框	232	11.2.2 新建数据库	260
9.5 框架	232	11.2.3 打开数据库	261
9.6 滚动条	234	11.2.4 添加表和修改表	262
9.7 图形方法和图形控件	236	11.2.5 添加、删除和修改 数据	265
9.7.1 坐标系	236	11.2.6 查询数据	267
9.7.2 图形方法	236	11.2.7 数据窗体设计器	271
9.7.3 图片框	238	11.3 ADO 数据对象访问技术	271
9.7.4 图像框	239	11.3.1 ADO 编程模型简介	271
9.7.5 Shape 形状控件	239	11.3.2 使用 ADO 编程模型	273
9.7.6 Line 直线控件	240	习题十一	275
9.8 其他常用语句	241	参考答案	277
9.8.1 强制显式声明变量 语句	241		
9.8.2 定义变量类型	242		

第1章 Visual Basic 6.0 概述

Visual Basic 是一种面向对象的程序设计工具。本章首先介绍面向对象程序设计的基本知识,然后介绍 Visual Basic 的 IDE 以及 Visual Basic 的工程管理。

1.1 面向对象程序设计(OOP)

在 21 世纪的今天,计算机正以前所未有的速度在全世界普及开来,它的应用早已远远超出最初纯数值计算的范畴,广泛应用于金融管理、广播通讯、交通能源、家用电器、科学研究、文化教育、办公自动化等各个领域,计算机应用实际上已经成为现代生活的一个重要组成部分。

我们都知道,一个完整的计算机系统是由硬件系统和软件系统构成的,而在硬件系统功能相差不多的时候,软件系统的丰富和完备将决定该计算机系统的整体性能。不管是在系统底层的操作系统、编译程序,还是与用户关系最紧密的应用程序,都需要使用相关的编程技术来进行开发。

本书所介绍的“Visual Basic 程序设计”,就是一种先进的“面向对象”的程序设计技术。

1.1.1 面向过程与面向对象程序设计

面向对象的程序设计(Object-Oriented Programming, OOP)是一种以对象为基础的、以事件驱动对象执行的程序设计技术。OOP 是将计算过程看做对象的分类过程加上其状态变换的过程,即将一个应用程序逐步划分成相互关联的多个对象,并建立起与这些对象相互关联的事件过程,通过对象对所发生的事件的响应,并执行相应的事件过程以引发其对象状态的变化,而最终达到完成计算的目的。

传统的结构化程序设计(Structured Programming)即面向过程的程序设计,则强调了功能上的抽象和模块性,它把解决问题的过程看做是一个处理过程。这就是说在结构化程序设计中模块是对功能的抽象,而每个模块都是一个处理单位,都包含输入和输出。

面向对象的程序设计(OOP)则综合了功能抽象和数据抽象,它把解决问题的过程看做是一个对象分类演绎的过程。在 OOP 中,对象(Object)是一个包括数据和方法(操作)的、并被封装起来的整体,它是对数据以及功能的抽象和统一,在对象中包含了模块的概念。

面向对象程序设计是一种全新的程序设计方法,它从根本上改变了传统的程序设计模式。其基本思想是:通过建立和具体实际应用相对应的对象,并通过这些对象的组合来创建具体的应用程序。

1.1.2 对象及其属性、方法和类

1. Visual Basic 的对象

Visual Basic 是一种面向对象的新型程序设计语言,这里的“对象”一词,实际就是日常生活中的一个普通术语。现实生活中的任何一个实体都是对象,大到地球、宇宙,小到一张桌子、椅子、一本书、一支笔等,均可视为一个对象,整个世界也就是由这些各种各样不同的对象构成的。为了区别和标识不同的对象,每个对象应该有一个属于自己的名字,对象的名字通常称为对象名,如桌子、椅子、汽车等均是一些不同对象的名字。在 Visual Basic 中,对象是 Visual Basic 系统中的基本运行实体,应用程序的窗体、标签、文本框、命令按钮等都是对象。

Visual Basic 中的对象主要分为两大类:一类是由系统设计好的,称为预定义对象,可以直接使用,如工具箱中的控件。另一类是由用户自定义的类实例化的对象。

对象具有属性、事件和方法三个要素。建立一个对象后,对对象的操作可以通过修改对象的属性、定制对象的事件、调用对象的方法来进行。

2. 对象的属性(Properties)

用以描述对象的各种特征,就是对象的属性,如名称(Name)、颜色(Color)、窗口形态(WindowState)、字体大小(FontSize)、是否可见(Visible)等。不同的对象具有不同的属性,例如,命令按钮具有名称、标题、大小、未知等属性,文本框具有名称、文本内容等属性。

每一个对象属性一般都具有一组默认值,如控件的名称默认为该控件的类别名称加上数字后缀,比如标签控件的名称为 Label1, Label2, Label3..., 其中的标题文本属性也默认为 Label1, Label2, Label3...。

设置对象的属性通常有两种方法。

(1) 在设计状态下,选中该对象,在属性窗口中对各项属性值进行修改和设置。

(2) 在程序运行中更改对象的属性。使用赋值语句,动态地改变对象的属性。其语法格式为。

[对象名称]. 属性名称 = 属性值

例如,在窗体窗口中已画有文本框控件,该控件的名称属性仍用缺省值 text1,而该对象的 Text 属性在程序代码中赋 Visual Basic,则语句为:

```
text1.Text = "Visual Basic"
```

执行此语句后,在窗口界面(text1)框中将显示出“Visual Basic”。

3. 对象方法(Method)

Visual Basic 的对象除拥有属于自己的属性和事件外,还拥有属于自己的方法。这里所谓的“方法”是指对象本身所内含的一些特殊函数或过程,利用这些内部自带的函数或过程,可以实现一些特殊的功能或动作,故称之为“方法”。如同现实世界中的不同类别的事物具有不同的功能一样,不同类型的对象所拥有的“方法”在数量和功能上不完全相同,这是由于不同类型的对象完成的任务不同所决定的。当用方法来控制某个对象的行为时,其实质就是调用该对象内部的某个特殊的函数或过程。比如,窗体对象拥有 Hide 方法和 Show 方法,调用 Hide 方法可以使窗体隐藏起来成为不可见,而调用 Show 方法可以使窗体显示成为可见。至于 Hide 方法是如何将窗体隐藏起来的,不必去管它。

在 Visual Basic 中,方法的调用格式为:

对象名.方法名#[参数项]

如果此方法需要带一些参数,此时,只需将所需参数放在方法名后面即可。比如对象的移动方法 Move,后面需要带有一个要移动到的坐标位置参数项。

例 1.1 现有一命令按钮对象 cmdMove,当用户单击该按钮时,将其移动到坐标为(400,550)的位置。

分析:要移动一个对象,可采用对象的 Move 方法来实现。其实现的事件过程如下。

```
Private Sub cmdMove_Click( )  
    cmdMove.Move 400,550  
End Sub
```

例 1.2 试用窗体对象的打印输出方法 Print,在窗体上打印输出“欢迎学习 Visual Basic 6.0”字符串。

分析:按照方法的一般调用格式,则 Print 方法的具体调用格式如下。

```
Private Sub cmdPrint_Click( )  
    Form1.Print "欢迎学习 Visual Basic 6.0"  
End Sub
```

若要清除在窗体上打印的字符串,则:

```
Private Sub cmdClear_Click( )  
    Form1.CLS  
End Sub
```

4. 类

类(Class)是某一类具有共性对象的集合定义,它包含了同一类对象的属性、方法、事件的定义。简单地说,类是创建对象的模型,对象则是按模型生产出来的每一个产品。比如,人是一个类,区别于其他动物,而我们每一个人,则是人类的一个对象。

在 Visual Basic 中,工具箱中的每一个控件,如文本框、标签、命令按钮等都是一个类。当我们将这些控件添加到窗体上时就创建了相应的对象。由同一个类创建的对象(如 Command1,Command2 等)具有预定义的公共属性、方法和事件,不同的类创建的对象具有不同的属性、方法和事件。

1.1.3 事件、事件过程及事件驱动

1. 事件

在现实世界中,我们会对一些动作作出反应,比如说敲击膝跳反射区域,我们会出现膝跳反射。Visual Basic 的对象事件,是 Visual Basic 对各类对象预先设置的,能够被对象识别的动作。如单击(Click)事件、双击(DbClick)事件、装载(Load)事件、按键(KeyPress)事件、获得焦点(GotFocus)事件等。

对于不同的对象,具有不同的事件。例如,窗体能识别单击和双击事件,而命令按钮能识别单击但是不能识别双击动作。

既然是事件,那么对象的事件需要由相应的动作引发,而这些动作是由用户操作或系统

触发实现。

2. 事件过程 (Event Procedure)

识别事件的目的是为了对事件进行有针对性地处理。既能识别事件又能正确地处理事件,反映了一个对象对事件的响应能力。识别能力是 Visual Basic 预先定义的,而处理能力则不是。这是因为,同一个类型的对象在不同的应用程序中所起的作用和完成的任务是不同的,而具体起什么作用、完成什么任务、如何响应这个事件完全取决于设计者。

因此,处理所有对象事件의编写代码任务也应该必须由设计者来完成。由于事件隶属于对象,为便于正确识别和管理,Visual Basic 提供了一个统一的编写处理事件过程代码的框架格式。以单击“打开”按钮(对象名为 cmdOpen)为例,框架格式如下:

```
Private Sub cmdOpen_Click( )  
    ' 对事件处理的程序代码  
End Sub
```

上述格式连同其中的程序代码称为“事件过程”,实现了对象(cmdOpen)对单击事件(Click)的响应。如果在上述框架内没有编写对事件处理的程序代码,那么,即使发生了单击事件(cmdOpen_Click),程序也不会作出任何反应。

3. 事件驱动

Visual Basic 应用程序运行时,通常先装载和显示一个窗体,之后会等待下一个事件(一般由用户操作来触发)的发生。当某一事件发生时,程序就会执行此事件的事件过程。当完成一个事件过程后,程序又会进行事件的等待,直到下一个事件发生为止。如此周而复始地执行,直到程序结束。也就是说,事件过程要经过事件的触发才能被执行,这种工作模式称为事件驱动方式。

Visual Basic 采用事件驱动的运行机制,程序的执行顺序不再按预先设计好的固定程序流程执行,而是通过响应不同的事件执行不同的事件过程。用户的操作不同,产生事件的次序就不同,响应事件的顺序自然也不同,即事件发生的顺序决定了整个程序的执行流程。

以上介绍了有关 Visual Basic 的几个重要概念,为了能尽快理解和掌握 Visual Basic 程序设计的思想和方法,建议读者一定要将对象、对象的属性、对象的事件以及对象的方法紧密联系起来,学会用对象的眼光去看待整个 Visual Basic 程序设计。

1.2 Visual Basic 简介

Visual Basic 是微软(Microsoft)公司推出的 Windows 环境下基于对象的可视化程序设计语言,可以用来开发 Windows 环境下的各种应用软件,也可用来开发多媒体应用软件。它是功能强大的可视化编程工具,即使没有编程经验的用户,也可以很容易地使用 Visual Basic 开发出功能强大的 Windows 应用程序。它提供了开发 Windows 应用程序最迅速、最简捷的方法,不论是 Windows 应用程序开发的专业人员还是初学者,Visual Basic 都为他们提供了整套工具,以方便开发应用程序。目前 Visual Basic 已经成为 Windows 环境下最主要的应用开发工具,使用这种开发工具,程序员不需要编写大量的代码去描述界面元素的外观和位置,只要把预先建立的界面元素用鼠标拖放到屏幕上的适当位置即可设计出一个易于使用的 Windows 应用程序。

1.2.1 Visual Basic 的发展历史及特点

1. Visual Basic 的发展过程

Visual Basic 是目前使用最广泛的一种程序设计语言,它是在原有的 BASIC 语言基础上发展起来的。BASIC(Beginning All-purpose Symbolic Instruction Code)语言是 1963 年由 John Kemeny 和 Thomas Kurtz 设计的,这种程序设计语言对于初学者来说确实是最容易学会的。它包含了数百条语句、函数及关键词,其特点很明显:每个语句前都有行号;书写不自动缩进;GOTO 和 GOSUB 都以行号为控制转移的目的。但是,它不符合后来发展起来的结构化程序设计的概念。随着计算机技术的发展和软件技术的进步,BASIC 也从一个效率低下、非结构化的解释型语言发展成为一个效率较高、结构化的解释/编译型语言,可用于开发范围广泛的应用程序的、基于事件驱动的可视化程序设计语言。Microsoft 公司的 Windows 系列操作系统(Windows 3X, Windows 95/98 和 Windows NT)以新颖、直观的、方便易用的图形工作环境赢得了广大用户的青睐,成为个人计算机上最为重要的运行操作环境。但同时也使 Windows 环境下的程序设计工作变得越来越复杂。例如,在 MS-DOS 下用任何一种程序语言编写一个在屏幕上显示一条信息的程序,最多只需 4 行语句就够了,而在 Windows 下编写同一功能的程序则其代码却可能长达 2~3 页之多。对于专业人员来说,这也许不是一个大问题,但是对于非计算机专业的人员来说,要在 Windows 下设计满足自己需要的应用程序却相当烦琐。如果能有一个高效、易用的 Windows 应用程序开发环境,则开发基于 Windows 的应用程序就会容易些。Microsoft 公司开发设计出的可视化的程序设计开发工具 Visual Basic,给广大的程序开发人员带来了福音,大大简化了他们的工作。

2. Visual Basic 的特点

Visual Basic 简单易学,其中的 Visual 指的是开发图形用户界面的方法。Visual Basic 语言具有简单易学的特性,在原有 BASIC 基础上增加了很多 Windows 图形用户界面(Graphical User Interface, GUI)的元素。Visual Basic 的可视化的用户界面设计功能,把开发人员从烦琐复杂的界面设计中解脱出来。可视化编程环境中的“所见即所得”功能,使界面设计如搭积木一样方便。强大的多媒体功能可轻而易举地开发出 Windows 下的集图像、动画、声音于一体的多媒体应用程序。新的网络功能提供了编写具有客户机/服务器和浏览器/服务器功能程序的能力。Visual Basic 的主要特点如下:

(1) 可视化的界面设计

以往的 Windows 应用程序开发工具在设计图形用户界面时,都是直接采用程序来实现的。一个大型应用程序大概有 90% 的程序代码用来处理用户界面,只有 10% 的程序才是真正用来实现应用程序功能的,而且在程序设计过程中看不到界面显示的效果,只有在程序执行时才能观察到,当界面效果不佳时还需回到程序中去修改。Visual Basic 提供新颖的可视化设计工具,可以直接可视化地设计用户界面,开发人员不必再为界面设计而编写大量程序代码,仅需采用现有工具按要求在屏幕上进行布局设计,对对象属性进行设置即可。Visual Basic 自动产生界面设计代码,开发人员的编程工作仅需要写出功能实现的那部分代码。这样可以大幅度提高程序设计的效率。

(2) 基于对象的程序设计思想

在一般面向对象的程序设计语言(如 C++)中,对象是由程序代码和数据组成的,是抽象的概念。而在 Visual Basic 中,已经把程序和数据封装起来成为一个个对象,并为每个对

象赋予各种属性,使对象成为实在的东西,开发人员在设计时无需编写每个对象的创建和描述程序代码,只管直接使用即可。开发人员所要做的仅仅是编写针对某个事件需要完成功能的程序。这种基于对象的程序设计具有简便、高效的特点。

(3) 事件驱动的编程机制

在 Visual Basic 程序中对象与程序的交互是通过事件来实现的,即针对每个对象,可能会有多种事件产生,每个事件都能驱动一段程序运行,完成相应的工作。如按钮(Button)是一个对象,用户单击一个按钮(Click)就会产生一个事件,响应此事件的程序代码放在 Button_Click 子程序中,单击按钮就可得到需要的功能。对于大型的 Windows 应用程序,用 Visual Basic 编程,无需构造一个完整的具有明显开始和结束的大型程序,而只需创建若干个较小的子程序,这些子程序均对应某个对象,由用户操作引发的事件来驱动完成某种特定功能,或是由事件驱动程序调用完成某功能的公共子模块程序。这样,开发人员可以方便地设计功能强大的 Windows 应用程序,提高工作效率。

(4) 方便、易用

Visual Basic 是一种高级语言,具有其他高级语言所具有的优秀特性:结构化的程序结构、更接近于人类的语言和逻辑思维方式、语句简单易懂等。此外,Visual Basic 还支持巨型数组、长字符串、动态数组等。Visual Basic 程序编辑器还支持彩色代码,可用于语法错误检查、判别标识符和断点。Visual Basic 还具有功能强大而且灵活的调试器和编译器。Visual Basic 是解释性语言,在编写代码的同时,解释程序将高级语言分解成计算机能识别的机器指令,并判断语句的语法错误。Visual Basic 程序随时可以运行,而且在整个应用程序调试好以后可编译生成 .exe 文件,使得用 Visual Basic 语言开发的 Windows 应用程序可脱离 Visual Basic 开发环境,直接在具有相关动态连接库支持的 Windows 环境下运行。

(5) 基于构件的程序开发

Visual Basic 程序设计中可以使用已经设计好的程序控件,通过控件的重用和共享,迅速建立应用系统,完成应用程序的开发。可以使用 Visual C++, Borland C++, Delphi 等开发用于 Visual Basic 的专用控件,以扩展 Visual Basic 的应用开发能力(在 5.0 以后的版本中,也可以使用 Visual Basic 开发自己的控件)。Visual Basic 控件可以加入到 Visual Basic 工具箱中供开发人员使用。

(6) 对象的链接与嵌入(OLE)

Visual Basic 3.0 以上的专业版中将工业标准 OLE 2.0 Automation 技术加入到 Visual Basic 中。OLE 是一种应用程序访问其他应用程序对象的方法。Visual Basic 的核心技术就是其对 OLE 的支持,OLE 技术允许 Visual Basic 导出任何遵循 OLE 标准的应用程序。利用 OLE 技术,在 Visual Basic 程序中,可以将其他应用软件作为一个对象嵌入到 Visual Basic 应用程序中,而且对嵌入对象的处理就好像处理 Visual Basic 本身的对象一样,可以对它赋予各种属性。可通过 OLE 技术把符合 OLE 标准的各种不同的 Windows 的应用程序嵌入到 Visual Basic 程序中,实现声音、视频、影像、动画、三维立体、变换几何形状、绘制各种图形图像等多媒体功能。即 Visual Basic 允许开发人员通过对现有应用程序的集成建立自己的应用程序,这就为方案设计人员、软件开发人员和独立的软件商带来了更加广泛的市场发展机会。

(7) 数据库访问功能

Visual Basic 系统本身提供了非常好的数据库管理功能,利用数据库管理窗口,用户可直接创建 Access 数据库,还可直接编辑和访问其他外部数据库,如 FoxPro, dBASE, FoxBase 等。

此外, Visual Basic 还提供了开放式数据库访问功能, 可通过 Visual Basic 提供的数据库访问对象方式直接访问外部数据库, 还可以通过 ODBC 建立连接的方式来使用 and 操纵外部数据库, 如 SQL Server, Oracle, Informix, Sybase 等。

(8) 功能强大的辅助开发工具

Visual Basic 中的 Setup Wizard 可为用户制作生成规范的带安装的用户盘, 使开发人员采用 Visual Basic 开发工具设计的应用程序在编译生成可执行文件以后, 脱离 Visual Basic 环境, 并制作成标准安装盘提交给用户使用, 用户可直接在 Windows 环境下进行安装和运行。Visual Basic 中还有报表生成器, 可用来制作各种报表。另外还具有为应用程序设计帮助信息的功能, 可在 Word 文字处理软件的 Help 编辑器中编辑应用程序的帮助, 并生成帮助文件。在采用 Visual Basic 进行应用程序设计时利用菜单和通用对话框的方式将帮助信息集成到应用程序中, 并以 Windows 环境下标准帮助窗口方式进行显示和检索。目前开发人员普遍使用的版本为 Visual Basic 6.0。

3. Visual Basic 版本简介

Microsoft 公司自 1991 年推出 Visual Basic 1.0 以来, Visual Basic 的版本不断更新, 功能不断增强, 目前最新的版本为 Visual Basic 6.0。同时为了满足不同的开发需要, Visual Basic 6.0 有三个不同功能的版本:

(1) 学习版

使开发人员可以轻松开发 Windows 95/98/2000 和 Windows NT 4.0 的应用程序, 该版本包括所有的内部控件以及表格、选项卡和数据绑定控件。与学习版一起提供的文档光盘包括 Learn Visual Basic Now 多媒体光盘, 以及包含全部联机文档的 MSDN (Microsoft Developer Network) 光盘。

(2) 专业版

为专业开发人员提供了一整套功能完备的开发工具, 专业版包括学习版的全部功能, 以及附加的 ActiveX 控件功能、IIS (Internet Information Server) 应用程序设计器、集成的可视化数据库工具 (Visual Database Tools) 和数据环境 (Data Environment) 设计器、ActiveX 数据对象 (ActiveX Data Objects, ADO)、动态 HTML 页面设计器 (Dynamic HTML Page Designer, DHTML)。与专业版一起提供的文档光盘包括 Visual Studio Basic Professional Feature 手册, 以及包含全部联机文档的 MSDN 光盘。

(3) 企业版

使得专业开发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, 加上 Back Office 工具 (如 SQL Server, Microsoft Transaction Server, Internet Information Server, Visual Source Safe 和 SNA Server 等)。与企业版一起提供的文档光盘包括 Visual Studio Enterprise Feature 手册, 以及全部联机文档的 MSDN 光盘。

4. Visual Basic 6.0 的新特征

Visual Basic 是 Microsoft 公司可视化编程语言的一套重要开发工具, 是一种完全的 32 位编程语言, 同时也支持 16 位的编程。Visual Basic 6.0 新增如下的特点:

(1) 增强的数据库访问能力

Visual Basic 6.0 提供了与现有系统的开放式互联性技术, 包括 ActiveX 数据对象 (ActiveX Data Object, ADO), Microsoft 公司还为各种不同的数据库提供了全新的 OLE DB 和 ODBC 驱动程序, 通过第三方提供的 ODBC 和 OLE DB 驱动程序, Visual Basic 可与大多数第三

方数据库系统(如 Oracle, DB/2 等)进行连接。

(2) 新的语言特征

用户自定义类型可以作为参数或公共属性和方法的返回值,函数可以返回数组变量,动态数组可以赋值,文件系统对象,按名称调用、创建增强的对象函数(CreateObject Function Enhancement)、增加的 StrConv 函数等。

(3) 更强大的 Internet 应用程序开发功能

在 Visual Basic 6.0 中,Internet 应用程序的开发变得更加容易,功能更加强大。在应用程序内可以通过 Internet 或 Intranet 访问文档和应用程序;也可以创建 Internet 服务器应用程序,包括 IIS 应用程序;支持动态 HTML 技术(DHTML)的应用程序;具有 Web 应用程序在线发布功能等。

(4) 新增加的控件

- 与数据库相关的控件:ADODataControl, DataGrid, DataListm, DataCombo, DataRepeater, DateTimePicker。

- 其他控件: CoolBar, HierachicalFlexGrid, ImageCombo, MonthView, ImageList, ListView, MsChart, ToolBar。

(5) 新增的部件功能

Visual Basic 6.0 提供了各种基于部件的可视化开发工具以及基于部件的 Web 开发能力。Visual Basic 6.0 包括了新版本的 Microsoft 公司可视化开发工具,可以帮助开发人员创建基于部件(Component)的解决方案。这些工具可用于创建基于 Microsoft DNA 体系的应用系统,包括创建交叉平台的 Web 应用程序、Windows 的客户端/服务器模式应用系统。Visual Basic 6.0 可用于快速创建基于 Windows 环境的客户端/服务器模式应用系统、Web 应用系统和中等规模的应用系统。Visual Basic 6.0 可以创建和重用 COM 部件,创建的部件可以被其他语言重用,而且也可以使用其他语言创建的部件。这使得利用已有的各种第三方发布的部件建立 Web 应用程序成为可能,开发人员可以使用任何高级语言,如 C++ 和 Java 等来建立部件,并将其集成在 Visual Basic 应用程序中。

(6) 增加的新向导

新的向导有数据对象生成向导和工具条向导。

5. 可视化程序设计的基本要素

Visual Basic 因界面美观、使用方便,深受广大开发人员的青睐,用某些开发人员的话说,Visual Basic 是一种“应用程序的构成框架”。Windows 应用程序的所有基本组成部分在 Visual Basic 中都可以得到,开发人员可以拿来即用,而不必专门为显示一个对话框而编写大段程序代码,这样可大大节省开发人员的时间,以设计出更好的应用程序系统。Windows 应用程序最大的特点是消息处理,所有的 Windows 应用程序都是基于消息传递和处理的。处理消息最基本的要素是对象、属性、方法、事件和消息,后面章节会分别介绍这些要素。

6. 什么是可视化程序设计

“Visual”一词的含义是“视觉的、可视的”,引申到程序设计领域中就变成了“可视化程序设计”,指的是开发图形用户界面(GUI)的一种方法。使用这种方法,开发人员不用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界面元素(如文本框、命令按钮等)用鼠标拖放到屏幕上的适当位置,并对这些对象进行属性和方法的设置即可完成程序的开发。将“Visual”和“Basic”这两个词连在一起,就表示用 BASIC 语言来可视化地进行应

用程序的开发。当然,Visual Basic 绝不仅仅是一个可视化的 BASIC 语言,其特征除了“可视化”之外,还包括“事件驱动”。正是“可视化”和“事件驱动”这两个特征构成了使用 Visual Basic 进行 Windows 应用程序开发的精髓。

1.2.2 Visual Basic 6.0 的运行

用户在计算机上安装 Visual Basic 后,就可以启动并在它的集成环境中编程了。

1. 启动 Visual Basic 6.0

要创建 Visual Basic 应用程序,首先要启动 Visual Basic 的集成开发环境。

方法:在 Windows 的任务栏上,单击“开始”→“程序”→“Microsoft Visual Basic 6.0 中文版”→“Microsoft Visual Basic 6.0 中文版”。

2. 退出 Visual Basic 6.0

退出 Visual Basic 的方法很简单,而且有多种。要退出 Visual Basic 集成环境,可以执行以下操作之一。

- ①单击标题栏上的关闭按钮。
- ②从“文件”菜单中选择“退出”。
- ③按“Alt + F4”或“Alt + Q”键。
- ④单击标题栏最左侧的图标,在打开的控制菜单中选择“关闭”按钮。

执行上述操作之一后,如果有修改过的文件没有保存,Visual Basic 会弹出对话框,提示用户是否保存修改。

1.3 Visual Basic 6.0 集成开发环境

本部分主要介绍 Visual Basic 语言的工作环境即集成开发环境(IDE),以及如何利用它创建 Windows 应用程序。

作为一个 32 位的 Windows 应用程序开发环境,Visual Basic 6.0 从开始编码到最终形成可执行的程序,都是通过集成开发环境来实现的。因而学习 Visual Basic 6.0 就首先要了解和掌握集成开发环境。

1.3.1 Visual Basic 6.0 集成的开发环境

启动 Visual Basic 后,出现“新建工程”对话框,如图 1-1 所示。

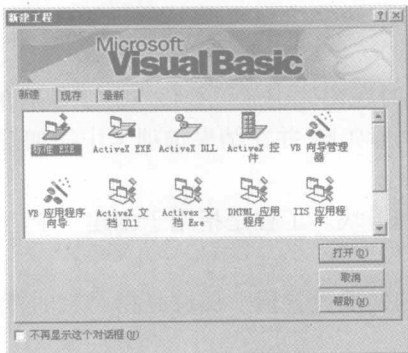


图 1-1 “新建工程”对话框

使用 Visual Basic 开发的应用程序或其他程序都被称为“工程”。选择“新建”选项卡可以建立一个新的工程,选择“现存”选项卡可以打开原来已有的工程,选择“最新”选项卡可以打开最近建立或使用过的工程。

在“新建”选项卡中选中“标准 EXE”,然后单击“打开”按钮,出现集成开发环境的主界面,如图 1-2 所示。

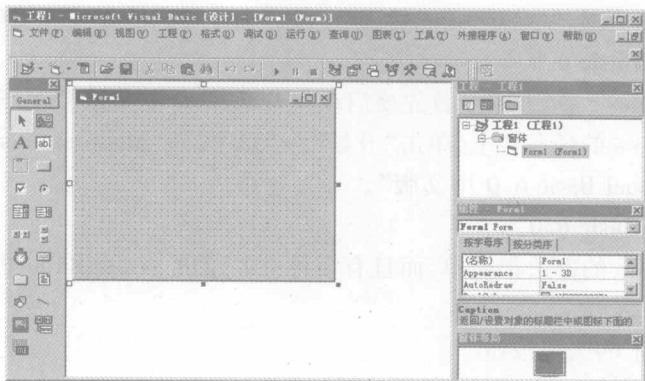


图 1-2 Visual Basic 集成开发环境 (IDE)

Visual Basic 集成开发环境 (IDE) 由以下元素组成。

(1) 菜单条

显示所使用的 Visual Basic 命令。除了提供标准“文件”、“编辑”、“视图”、“窗口”和“帮助”菜单之外,还提供了编程专用的功能菜单,例如“工程”、“格式”或“调试”。

(2) 上下文菜单

包括经常执行的、操作的快捷键。在要使用的对象上单击鼠标键即可打开上下文菜单。在上下文菜单中有效的专用快捷键清单取决于单击鼠标键所在的环境。例如,在“工具箱”上单击鼠标右键时显示的上下文菜单,可以在上面选择:显示“部件”对话框、隐含“工具箱”、连接或挂断“工具箱”或在“工具箱”中添加自定义选项卡。

(3) 工具栏

在编程环境下提供对于常用命令的快速访问。单击工具栏上的按钮,则执行该按钮所代表的操作。按照缺省规定,启动 Visual Basic 之后显示“标准”工具栏。附加的编辑、窗体设计和调试的工具栏可以从“视图”菜单上的“工具栏”命令中移进或移出。

工具栏能紧贴在菜单条之下,或以垂直条状紧贴在左边框上,如果将它从菜单下面拖开,则它能“悬”在窗口中。

(4) 工具箱

提供一组工具,用于设计时在窗体中放置控件。除了缺省的工具箱布局之外,还可以通过从上下文菜单中选定“添加选项卡”并在结果选项卡中添加控件来创建自定义布局。

(5) 工程管理器窗口

列出当前工程中的窗体和模块。工程是指用于创建一个应用程序的文件的集合。

(6) 属性窗口

列出对选定窗体和控件的属性设置值。属性是指对象的特征,如大小、标题或颜色。