

Visual Basic 程序设计

王编 戴青云 王 静

Visual Basic 程序设计

主 编 戴青云 王 静
副主编 李 毅 孙庆月 陈文三 高亚平
主 审 王金斗

版权专有 侵权必究

图书在版编目(CIP)数据

Visual Basic 程序设计/戴青云,王静主编. —北京:北京理工大学出版社,2012.7

ISBN 978-7-5640-6181-4

I. ①V… II. ①戴…②王… III. ①BASIC 语言-程序设计-高等学校-教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 141408 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(批销中心) 68911084(读者服务部)

网 址 / [http:// www. bitpress. com. cn](http://www.bitpress.com.cn)

经 销 / 全国各地新华书店

印 刷 / 北京泽宇印刷有限公司

开 本 / 787 毫米 × 1092 毫米 1/16

印 张 / 16

字 数 / 364 千字

责任编辑 / 胡 静

版 次 / 2012 年 7 月第 1 版 2012 年 7 月第 1 次印刷

王玲玲

印 数 / 1 ~ 1 500 册

责任校对 / 周瑞红

定 价 / 49.00 元

责任印制 / 王美丽

图书出现印装质量问题,本社负责调换

前言

Preface <<<<

Visual Basic, 简称 VB, 是 Microsoft 公司推出的一种 Windows 应用程序开发工具, 是当今世界上使用最广泛的编程语言之一, 它也被公认为是编程效率最高的一种编程方法。无论是开发功能强大、性能可靠的商务软件, 还是编写能处理实际问题的实用小程序, VB 都是最快速、最简便的方法。因此, 在我国高等院校中很多计算机或与其相关专业, 往往将 Visual Basic 程序设计作为程序设计的首选课程。

本书设置了 7 个教学情境, 其他部分均服务于这 7 个教学情境。在具体内容的选取上, 坚持“必须、够用”的原则, 将传统教材章节内容进行整合或分解, 兼顾了当前科学技术的发展和学生的实际水平, 既考虑了教学内容的完整性和连续性, 同时又降低了学习难度。既考虑了教学内容概念清晰、突出重点; 又考虑了后续课程对本课程的要求。本教材重点强调了理实一体化, 注重培养学生分析和解决实际问题的能力, 满足学生对先进控制技术的应用的需要, 更能适应学生的实际水平。

全书共 10 章分为两部分: 第 1 部分为基础知识由第 1 ~ 第 3 章组成, 从程序设计的基本概念开始, 介绍了用 Visual Basic 设计 Windows 应用程序的基本方法和步骤、窗口应用程序的设计方法和运行模式, 并介绍了基本的面向对象程序设计的重要概念, 以及 Visual Basic 程序的 3 种基本控制结构等。其中, 第 1 章为编写第一个 Visual Basic 应用程序, 第 2 章为 Visual Basic 程序设计基础, 第 3 章为程序控制结构。第 2 部分为第 4 ~ 第 10 章, 在这一部分包括 7 个学习情境, 每个学习情境也是 1 个任务。其中学习情境学生成绩统计程序设计, 介绍了数组的应用以及消息对话框; 在计算器设计、排序插入程序设计、图片自动显示器程序设计等 3 个学习情境中系统地介绍了 Visual Basic 常用控件的使用, 同时还介绍了控件数组的使用, 子过程以及子函数的使用; 学习情境文本编辑器设计, 介绍了菜单制作、通用对话框控件使用、多文档程序开发过程; 学习情境简单财务管理软件设计介绍了 Visual Basic 与数据库编程的初步应用; 学习情境图书综合管理系统设计以一个综合项目的形式系统地介绍了 Visual Basic 应用程序的开发过程。

本书是由具有多年 Visual Basic 程序设计讲授经验和应用程序项目开发经历的“双师型”教师编写。其中, 第 4 ~ 第 9 章由戴青云编写, 第 1、第 2 章由王静、孔庆月、陈文兰编写, 第 3 章由程俊红、李瑗、牛君、陈云尚、李彩玲、雷晨、高亚平编写, 第 10 章由李毅编写, 王金斗对全书进行了审阅, 全书由戴青云拟定编写大纲并负责统稿。

在本书的编写过程中, 得到了许多从事计算机教学工作的同事的帮助和大力支持, 他们对本书提供了很多宝贵的建议, 在此向他们表示衷心的感谢。在编写本书的过程中, 作者参考了多位同行的编著和文献, 在此一并表示衷心的感谢。

由于编者水平有限, 时间仓促, 书中难免存在不足和疏漏之处, 敬请广大读者批评指正。本书编者联系邮箱为 susu_2004@126.com。

编 者

目录

Contents

第1章 编写第一个 Visual Basic 应用程序	1
1.1 Visual Basic 6.0 概述	1
1.1.1 Visual Basic 6.0 的发展历程	1
1.1.2 Visual Basic 6.0 的特色	1
1.2 Visual Basic 6.0 的启动及基本界面	2
1.2.1 Visual Basic 6.0 的启动和退出	2
1.2.2 Visual Basic 6.0 的集成开发环境 (IDE)	2
1.3 面向对象程序设计基本概念	3
1.3.1 Visual Basic 的对象与类	3
1.3.2 对象的属性、方法与事件	4
1.3.3 窗体	5
1.4 应用程序举例	9
1.4.1 创建工程	9
1.4.2 设计应用程序界面	10
1.4.3 编写程序代码	11
1.4.4 运行、保存工程	12
1.5 获取帮助	13
1.5.1 在线帮助	13
1.5.2 上下文帮助	14
1.6 习题	14
第2章 Visual Basic 程序设计基础	17
2.1 程序设计基础	17
2.1.1 关键字	17
2.1.2 标识符	17
2.2 数据类型	17
2.2.1 基本数据类型	18
2.2.2 用户自定义类型	20
2.3 常量与变量	20
2.3.1 常量	20
2.3.2 变量	21
2.3.3 Option Explicit 语句	22
2.4 运算符和表达式	22

2.4.1	算术运算符和算术表达式	22
2.4.2	字符串运算符	23
2.4.3	关系运算符和关系表达式	23
2.4.4	逻辑运算符和逻辑表达式	24
2.4.5	运算符的优先级别	25
2.5	常用内部函数	25
2.5.1	数学函数	25
2.5.2	字符串函数	26
2.5.3	转换函数	27
2.5.4	日期函数	27
2.5.5	测试函数	27
2.5.6	其他函数	28
2.6	习题	30
第3章	控制结构	32
3.1	顺序结构	32
3.1.1	赋值语句	32
3.1.2	With 语句	33
3.1.3	Print 语句	33
3.1.4	End 语句	34
3.1.5	程序代码书写规则	34
3.2	选择结构	35
3.2.1	If/Then 语句	35
3.2.2	Select Case 语句	39
3.3	循环结构	41
3.3.1	Do...Loop 循环	42
3.3.2	While...Wend 循环	45
3.3.3	For...Next 循环	45
3.3.4	循环嵌套	46
3.4	习题	48
第4章	学生成绩统计程序设计	55
4.1	学生成绩统计程序设计任务描述及分析	55
4.2	知识链接	56
4.2.1	输入对话框和消息对话框	56
4.2.2	一维数组	59
4.2.3	二维数组	62
4.3	简单学生成绩统计程序设计与实现	63
4.4	通用学生成绩统计程序	65

4.4.1 动态数组	65
4.4.2 通用学生成绩统计程序设计与实现	66
4.5 习题	68
第5章 计算器设计	74
5.1 计算器程序任务描述及分析	74
5.2 知识链接	75
5.2.1 标签控件 (Label)	75
5.2.2 命令按钮控件 (CommandButton)	76
5.2.3 文本框控件 (TextBox)	76
5.2.4 选择类控件	79
5.2.5 框架 (Frame)	81
5.2.6 控件数组	84
5.2.7 变量作用域	85
5.3 计算器程序设计	87
5.3.1 界面设计	87
5.3.2 计算器程序编制	88
5.4 习题	92
第6章 排序插入程序设计	96
6.1 排序插入程序设计描述及分析	96
6.2 知识链接	96
6.2.1 Sub 过程	97
6.2.2 Function 过程	99
6.2.3 过程间参数传递	100
6.2.4 过程作用域	103
6.2.5 列表框控件 (ListBox)	103
6.3 排序插入程序实现	106
6.3.1 排序程序实现	107
6.3.2 插入程序实现	107
6.4 习题	108
第7章 图片自动显示器程序设计	115
7.1 图片自动显示器任务描述及分析	115
7.2 知识链接	116
7.2.1 图片框控件 (PictureBox)	116
7.2.2 图像框控件 (Image)	116
7.2.3 定时器控件 (Timer)	117
7.2.4 驱动器列表框 (Drive List Box)	118

7.2.5	目录器列表框 (Directory List Box)	119
7.2.6	文件列表框 (File List Box)	120
7.2.7	组合框 (ComboBox)	122
7.3	图片自动播放器编程实现	124
7.3.1	界面设计	124
7.3.2	手动浏览图片的实现	125
7.3.3	自动显示实现	127
7.4	习题	127
第8章 文本编辑器设计		131
8.1	文本编辑器任务描述及分析	131
8.2	知识链接	133
8.2.1	RichTextBox 控件	133
8.2.2	通用对话框控件	136
8.2.3	菜单	139
8.2.4	工具栏	144
8.2.5	剪切板	147
8.2.6	MDI 窗体	148
8.3	文本编辑器编程实现	149
8.3.1	界面设计	149
8.3.2	菜单制作	150
8.3.3	工具栏制作	151
8.3.4	主程序编制	152
8.4	习题	162
第9章 简单财务管理软件设计		166
9.1	简单财务管理软件任务描述及分析	166
9.2	知识链接	167
9.2.1	Visual Basic 访问数据库技术	167
9.2.2	Data 控件	169
9.2.3	MSFlexGrid 控件	174
9.2.4	ADO 数据控件	175
9.2.5	DataGrid 控件	179
9.3	简单财务管理软件编程实现	179
9.3.1	登录窗体设计	179
9.3.2	主窗体设计	181
9.3.3	添加用户窗体设计	182
9.3.4	删除用户窗体设计	183
9.3.5	收支录入窗体设计	185

9.3.6 修改收支信息窗体设计	187
9.3.7 家庭财务检索窗体设计	189
9.4 习题	190
第10章 图书综合管理系统设计	192
10.1 图书综合管理系统任务描述及分析与知识链接	192
10.1.1 图书综合管理系统任务描述及分析	192
10.1.2 知识链接	193
10.2 系统整体设计	195
10.2.1 系统数据库设计	195
10.2.2 系统流程设计	198
10.3 图书综合管理系统编程实现	198
10.3.1 数据库结构实现	198
10.3.2 公共模块创建	199
10.3.3 登录模块	200
10.3.4 主窗体创建	202
10.3.5 系统用户管理模块	203
10.3.6 读者信息管理模块	212
10.3.7 图书管理模块	219
10.3.8 借书管理模块	226
10.3.9 还书管理模块	230
10.3.10 读者查询模块	232
10.4 系统编译与发行	234
10.4.1 打包和展开向导	234
10.4.2 制作安装程序	234
10.5 习题	238
参考文献	240

到目前为止，人们已经开发了数百个高级语言，但只有为数不多的语言得到了认可，Visual Basic 即为其中之一。Visual Basic 提供了诸如图形用户界面、事件处理、访问 Win32 API、面向对象、错误处理、结构化编程等功能。本章将介绍 Visual Basic 的特色，集成开发环境，面向对象程序设计基本概念，并通过一个应用实例介绍设计 Visual Basic 应用程序的开发步骤。

1.1 Visual Basic 6.0 概述

1.1.1 Visual Basic 6.0 的发展历程

Visual Basic，简称 VB，是 Microsoft 公司推出的一种 Windows 应用程序开发工具。它是当今世界上使用最广泛的编程语言之一，也被公认为是编程效率最高的一种编程方法。无论是开发功能强大、性能可靠的商务软件，还是编写能处理实际问题的实用小程序，VB 都是最快速、最简便的方法。“Visual”指的是可视化的开发图形用户界面的方法，一般不需要编写大量代码去描述界面元素的外观和位置，只要把需要的控件拖放到屏幕上的相应位置即可；“Basic”指的是 BASIC 语言，因为 VB 是在原有的 BASIC 语言的基础上发展起来的，至今包含了数百条语句、函数及关键字。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字就可以建立实用的应用程序。

自 1991 年 Microsoft 推出 Visual Basic 1.0 后，版本不断升级，相继推出了 Visual Basic 2.0、Visual Basic 3.0、Visual Basic 4.0、Visual Basic 5.0，1998 年又推出了 Visual Basic 6.0。

VB 6.0 提供了学习版、专业版和企业版三种版本，用以满足不同的开发需要。学习版使编程人员很容易地开发 Windows 和 Windows NT 的应用程序；专业版为专业编程人员提供了功能完备的开发工具；企业版允许专业人员以小组的形式来创建强健的分布式应用程序。

Visual Basic 系统软件的安装过程同其他软件的安装过程一样，按照屏幕提示一步步操作即可，十分简单。

1.1.2 Visual Basic 6.0 的特色

1. 面向对象的可视化编程

在 Visual Basic 6.0 中，采用面向对象程序设计方法（Object - Oriented Programming），

把程序和数据封装起来作为一个对象，每个对象都是可视的。开发人员在界面设计时，可以直接用 Visual Basic 6.0 的工具箱在屏幕上“画”出窗口、菜单、命令按键等不同类型的对象，并为每个对象设置属性。开发人员要做的仅仅是对要完成事件过程的对象编写代码，因而程序设计的效率可大大提高。

2. 事件驱动的编程机制

面向过程的程序是由一个主程序和若干个子程序及函数组成的。程序运行时总是先从主程序开始，由主程序调用子程序和函数，开发人员在编程时必须事先确定整个程序的执行顺序。而 Visual Basic 6.0 事件驱动的编程是针对用户触发某个对象的相关事件进行编码，每个事件都可以驱动一段程序的运行。开发人员只要编写响应用户动作的代码。这样的应用程序代码精简，比较容易编写与维护。

3. 结构化的程序设计语言

Visual Basic 6.0 具有丰富的数据类型和众多的内部函数。其采用模块化和结构化程序设计，结构清晰，语法简单，容易学习。

4. 充分利用 Windows 资源

VB 程序开发与 Windows 系统紧密相连，充分利用 Windows 操作系统的环境和功能。可以方便地调用 Windows 的函数和应用程序，开发出各种功能的应用程序。

5. 支持多种数据库系统的访问

VB 具有很强的数据管理功能，使用户能在很短的时间内编写数据库管理系统。VB 可以直接建立或处理 Microsoft Access 格式的数据库，同时，VB 还能直接编辑和访问其他一些数据库。VB 还提供开放式数据互连（ODBC）功能、ActiveX 数据对象（ADO）和 ADO 数据控件、远程数据对象（RDO）和远程数据控件等，利用结构化查询语言 SQL，访问或操作大型网络数据库。同时 VB 采用了对象的链接与嵌入（Object Linking and Embedding, OLE）技术可以很方便地开发出集声音、图像、动画、字处理、Web 等对象于一体的应用程序。

1.2 Visual Basic 6.0 的启动及基本界面

1.2.1 Visual Basic 6.0 的启动和退出

VB 6.0 的启动步骤如下：

① 单击“开始”按钮。

② 选择“程序”菜单，并将鼠标移动到“Microsoft Visual Basic 6.0 中文版”程序项，单击“Microsoft Visual Basic 6.0 中文版”程序项即可启动中文版 VB 6.0。

若在桌面上有 VB 6.0 快捷方式图标，用户也可直接在桌面上双击该图标启动 VB 系统。若要退出 VB 环境，只需打开“文件”菜单，选择“退出”即可。

1.2.2 Visual Basic 6.0 的集成开发环境（IDE）

VB 拥有一个功能强大而又易于操作的集成开发环境 IDE（Integrated Development Environment），所有的图形界面设计和代码的编写、调试、运行、编译均在该集成环境中完成。

启动 Visual Basic 6.0 后，就进入了 Visual Basic 的集成开发环境，VB 的集成开发环境如

图 1-1 所示，主要由以下窗口组成：主窗口、窗体设计器窗口、代码编辑窗口、属性窗口、工程资源管理器窗口、窗体布局窗口、工具箱窗口、立即窗口。各窗口作用如下：

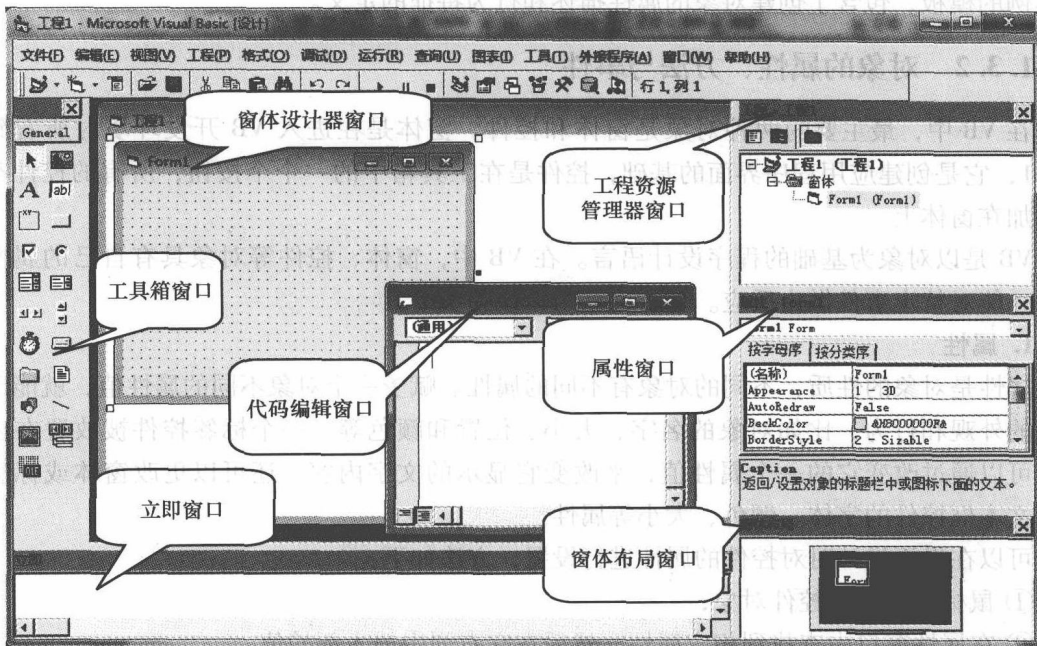


图 1-1 VB 集成开发环境

主窗口：由标题栏、菜单栏、工具栏等组成。

工具箱窗口：工具箱中有 20 个标准控件，可以选择控件。

窗体设计器窗口：简称窗体（Form），是最终用户看到的软件界面，应用程序的运行结果、各种图形、图像、数据等都是通过窗体或窗体中的控件显示出来的。

工程资源管理器窗口：显示当前工程文件所包含的各种文件。

窗体布局窗口：设置窗体在屏幕上的显示位置。用来指定程序运行时的窗体初始位置，可以通过拖动窗体图标的位置来调整程序运行时窗体显示的位置。

属性窗口：显示当前窗体或控件的所有属性，可按字母分类显示。

代码编辑窗口：用来编辑修改过程或事件过程的代码。

立即窗口：主要用于调试程序时，检查变量或表达式的值。

1.3 面向对象程序设计基本概念

1.3.1 Visual Basic 的对象与类

对象（Object）是代码和数据的集合。一个典型的 Windows 应用程序具有用户可视的界面，这个界面以窗体的形式呈现出来，窗体上有各种可视元素，如文本框、按钮等，所有这些可视的元素都是对象。

对象是具有特殊属性（数据）和行为方式（方法）的实体。建立对象后，用其操作与

该对象有关的属性、事件和方法来描述。

类是同一种对象的统称，是一个抽象的概念，它通过实例化创建对象。所以类是创建对象实例的模板，包含了创建对象的属性描述和行为特征的定义。

1.3.2 对象的属性、方法与事件

在 VB 中，最主要的两种对象是窗体和控件。窗体是在进入 VB 开发环境时所看到的 Form1，它是创建应用程序界面的基础，控件是在工具箱中的一个个按钮，所有的控件都可以添加在窗体上。

VB 是以对象为基础的程序设计语言。在 VB 中，窗体、控件等对象具有自己的属性和方法，能对特定事件做出反应。

1. 属性

属性是对象的性质，不同的对象有不同的属性。赋予一个对象不同的属性值，就能改变对象的外观和行为，比如对象的名字、大小、位置和颜色等。一个标签控件被放置在窗体上，可以通过改变它的一个属性值，来改变它显示的文字内容。还可以更改窗体或标签控件、文本框控件的字体、颜色、大小等属性。

可以在界面设计时对控件的属性进行设置，方法如下：

- ① 鼠标单击选中控件对象；
- ② 在属性窗口中查找到相应属性，然后在它右列中填入新的值。

也可以在程序运行时改变控件的属性，这需要在设计时的代码窗口中通过编程实现，其格式为：

对象名. 属性名 = 属性值

2. 方法

方法是对对象能执行的操作或具有的功能，是一段不必知道细节同时也无法改变其代码的程序。方法不是随意的，每个对象都有自己特定的方法。程序员可以直接使用对象提供的方法来完成某些操作或功能，调用对象方法的格式为：

[对象名.]方法 [参数列表]

其中，如果省略了对象名，则表示当前对象，一般是当前窗体。

3. 事件

事件是由用户或系统触发，可以由对象识别的操作。当在对象上发生了某个事件时，如果要处理这个事件，就必须设计事件处理的步骤。事件处理的步骤称为事件过程。事件的发生也不是随意的，某些事件仅发生在特定对象上。VB 程序设计的主要任务就是为对象编写事件过程中的程序代码。

事件过程的语法如下：

```
Private Sub 对象名_事件名 ([参数表])  
    处理事件的代码  
End Sub
```

这里的对象名可以是窗体或控件的名称，如果对象是窗体，则一律用 Form_ 事件名。

1.3.3 窗体

窗体是 Visual Basic 中经常用到的对象，是用户与应用程序之间进行交互的窗口，是构成应用程序的基本模块，是可视化程序设计的基础。窗体又称窗口，在程序运行前，即程序设计阶段，称为窗体；程序运行后也可称为窗口。当新建一个标准工程时，系统将自动生成一个窗体，名为 Form1。窗体具有自己的属性、事件和方法。

1. 窗体的常见属性

窗体的属性决定了窗体的外观和操作，共有 50 个属性，全部列于属性窗口中。对于窗体的大部分的属性，既可以通过属性窗口改变，也可以在程序中通过代码进行改变。仅有少部分的属性只能通过属性窗口改变。

(1) Name 属性

窗体的名称，用于在程序中唯一的标识窗体。在程序中每添加一个新窗体时，系统会自动给其一个默认的名称 Form 加上一个序号如：Form1、Form2……

对象都有名字，计算机把名字看成对象与对象之间的根本差异，因此在同一窗体里不允许出现重名的情况（除非这是一个控件数组），且名字不得超过 40 个字符。在简单的程序里，给控件命名不是很必要，完全可以使用控件 Name 属性的默认值。

但在有几十个控件的复杂窗体里，就很难区分它们。所以，VB 推荐由三个小写字母的前缀和一个第一个字母为大写的描述性单词组成的名字。例如 frmMyForm 是一个窗体的名称（前缀是 frm）。

(2) Caption 属性

窗体的标题，即显示在标题栏内控制图标右面的标题，用来向用户说明窗体的作用。系统默认值和 Name 属性的默认值相同。

Caption 属性是许多对象都具有的属性，而“名称”属性则是每个对象必备的，主要作用是设置对象在窗体上显示的文字，向用户提示本对象的功能。

对象的“名称”属性则是对象的名字，在程序中如果要引用对象的属性或方法，都要用到这个属性。

二者的区别可以用一个人的名字和这个人身上穿的衣服来类比，Name 属性好比人的名字，Caption 属性好比人穿的衣服，不能因为某天换了一件衣服就不认得这个人了。并且，在程序运行时 Name 属性值不可改变，而 Caption 属性值是可以改变的。

(3) Left、Top、Width 和 Height 属性

所有可视化的对象（窗体和控件）都具有这些属性，它们决定了对象的位置和大小。

功能：Left 和 Top 分别用来设置对象左边和顶边的坐标。Width 和 Height 分别用来设置对象的宽度和高度。

对应的属性值表示一个相对距离，当“对象”为窗体时，Left 指的是窗体左边界与屏幕左边界的相对距离，Top 指的是窗体的顶边与屏幕顶边的相对距离；而当“对象”为控件时，Left 和 Top 分别指控件的左边和顶边与窗体的左边和顶边的相对距离。

属性值以 Twip 为单位，1Twip = 1/1440 英寸。

(4) ForeColor 和 BackColor 属性

ForeColor：用来定义文本或图形的前景颜色。

BackColor: 用来定义窗体或控件的背景颜色。

设置这两个属性时, 可以通过打开的调色板选取适当的颜色。

此外, VB 提供了 8 个颜色常数, 可在代码中直接用于颜色设置: **vbBlack** (黑色)、**vbRed** (红色)、**vbGreen** (绿色)、**vbYellow** (黄色)、**vbBlue** (蓝色)、**vbMagenta** (洋红)、**vbCyan** (青色) 和 **vbWhite** (白色)。

例如, 可通过以下语句设置窗体的背景色为黄色, 前景色为红色:

```
Form1.BackColor = vbYellow
```

```
Me.ForeColor = vbRed ' Me 表示当前窗体
```

(5) Appearance 属性

设置窗体在运行时是否以立体的效果显示。0 表示以平面的形式显示, 1 表示以 3D 的立体效果显示。

(6) BorderStyle 属性

设置窗体的边框样式, 该属性值可以为 0~5。0 表示无边框; 1 表示窗体大小固定且具有单线边框; 2 (默认值) 表示窗体具有大小可调整的边框; 3 表示窗体具有大小固定的边框; 4 表示窗体为单线固定工具窗口, 且只包含关闭按钮; 5 表示窗体为可调整的工具窗口, 且只包含关闭按钮。设计者可以根据需要选择窗体相应的样式。

(7) MaxButton、MinButton 属性

设置窗体的标题栏中是否具有最大化/最小化按钮, 该属性为布尔值。

(8) ContrlBox 属性

设置窗体是否具有关闭功能。若该属性为 **True**, 则窗体中存在关闭按钮, 可以被用户关闭, 并且用鼠标单击控制图标也可以打开控制菜单; 若该属性为 **False**, 则窗体中不存在关闭按钮和控制菜单。

(9) WindowState 属性

设置窗体运行时的初始状态。该属性值可以为 0~2 的一个整数。

0: 正常状态。运行后窗体的大小以设计阶段为准。

1: 最小化状态。运行后窗体缩小为一个图标显示在任务栏中。

2: 最大化状态。运行后窗体充满整个屏幕。

(10) StartUpPosition 属性

用于控制窗体首次显示时的位置。该属性值可以为 0~3 的一个整数。

0: 窗体的初始位置由其 **Left** 和 **Top** 属性决定。

1: 窗体显示在所有者的中央, 若是 MDI 窗体, 则显示在其父窗体的中央。

2: 窗体显示在屏幕中央。

3: 窗体显示在屏幕的左上角。

(11) Font 属性

该属性表示返回一个 **Font** 对象。对 **Font** 属性可以通过单击属性窗口中的 **Font** 属性所弹出的字体对话框进行修改, 如图 1-2 所示。修改方法与 Word 中“格式”菜单中的“字体”一样。

(12) AutoRedraw 属性

设置窗体是否具有自动重画的功能。若将该属性设为 **True**, 则当窗体被其他窗体覆盖又

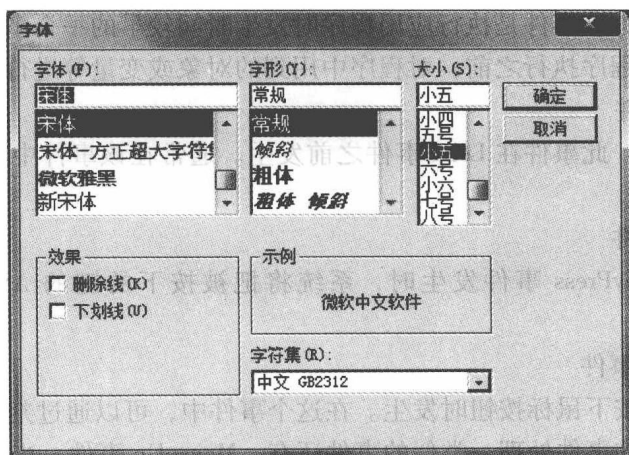


图 1-2 Font 属性设置

回到该窗体，或窗体的大小发生改变时，会自动刷新显示在窗体上的图形和文本。

(13) Enabled 属性

每个对象都具有 Enabled 属性，用来设置对象是否可以被用户激活，即对象是否接受并响应用户事件。若该属性设置为 False，则对象一般呈暗淡显示，说明用户不可使用该对象。这个属性通常用来在程序运行时控制对象是否达到可以使用的条件。

(14) Visible 属性

该属性是用于设置对象在程序运行时是否可见。若该属性为 True，则对象显示在屏幕上；若该属性为 False，则对象隐藏起来，不在屏幕上显示。

(15) Picture 背景图片

该属性用于设置窗体中要显示的背景图片。用来在对象中显示一个图形。该属性可以显示 BMP、JPG、GIF 等多种格式的图形文件。

2. 窗体的事件

(1) Click (单击) 事件

单击事件是几乎每个对象都具有的事件，当用户在一个对象上按下并释放鼠标按钮时发生。对窗体而言，当在窗口内没有其他控件的任何位置单击鼠标时，就会触发窗体的 Click 事件。

(2) DblClick (双击) 事件

对象的双击事件，也是大多数对象都具有的一个事件。当用户在一个对象上双击鼠标时发生。

(3) Activate 事件

窗体的激活事件。在窗口由非活动窗口变为活动窗口时的瞬间发生，窗口一旦成为活动窗口，该事件就会消失。

(4) Deactivate 事件

与 Activate 事件相对，在窗口由活动窗口变为非活动窗口时的瞬间发生。

(5) Load 事件

窗体的加载事件，当窗体被调入内存时发生。每执行一个应用程序，在屏幕上都会至少

打开一个窗口。所以，该事件是执行应用程序时发生时间较早的一个事件。因此，在这个事件中通常加入一些在程序执行之前，对程序中用到的对象或变量等进行初始化的语句。

(6) Initialize 事件

初始化窗口事件，此事件在 Load 事件之前发生。通常在该事件中初始化变量或设置窗体属性。

(7) KeyPress 事件

按键事件。当 KeyPress 事件发生时，系统将把被按下的键的 ASCII 码传给 KeyAscii 参数。

(8) MouseDown 事件

当用户在对象上按下鼠标按钮时发生。在这个事件中，可以通过判断用户按下的是左键还是右键而做出不同的事件处理。类似的事件还有，MouseUp 事件：当用户在对象上释放鼠标键时触发该事件；MouseMove 事件：当用户在对象上移动鼠标时触发该事件。

(9) Resize 事件

当窗体第一次显示或用户改变窗体的大小时会触发这个事件的发生。

(10) Unload 事件

当窗体被关闭而从屏幕上消失时发生。

3. 窗体的常见方法和语句

(1) Show 方法

用来将一个已经装入内存的窗体显示出来。其语法为：窗体名.Show [模式]。

其中“模式”参数有两种取值：0（默认值）表示非模化的窗体，1 表示模化的窗体。模化的窗体是指该窗口出现后，用户必须对其作出响应，否则不能进入其他任何操作的窗口。这种窗口经常用在一些对话框窗口上。

(2) Hide 方法

用于隐藏显示在屏幕上的窗体。隐藏窗体时，用户将无法访问该窗体上的控件，但是运行中的 VB 应用程序并不卸载它，仍然可以使用语句访问隐藏窗体上的控件。其语法为：

[窗体名]. Hide

(3) Cls 方法

用于清除在窗体上显示的文本，并把光标重新定位到窗体的左上角。其语法为：

[窗体名]. Cls

(4) Move 方法

Move 方法用于移动窗体或控件，并可以改变其大小。格式如下：

[对象]. Move 左边距离[, 上边距离[, 宽度[, 高度]]]

对象：可以是窗体以及除菜单以外的所有可视控件，若省略对象则默认为当前窗体。

左边距离、上边距离、宽度、高度均为数值，以 Twip 为单位。如果对象是窗体，则“左边距离”和“上边距离”以屏幕左边界和上边界为准，否则以窗体等容器内部的左边界和上边界为准。“宽度”和“高度”指定对象的新宽度和新高度。调用 Move 方法后将自动改变对象的 Left、Top、Width 和 Height 四个属性。

(5) Load 语句

该语句用来将新创建的窗体加载到内存中，但此时窗体并没有显示出来，其语法格式