

高职、高专计算机系列教材



Visual Basic 实用教程

● 刘美珍 主编

华中科技大学出版社



高职、高专计算机系列教材

Visual Basic 实用教程

主编 刘美珍

编者 李 露 孙 俊 冯运仿

华中科技大学出版社

面向 21 世纪计算机教材出版指导委员会

主任 陈火旺（中国科学院院士）
沈绪榜（中国科学院院士）
邹寿彬（华中科技大学博士生导师）

委员（以姓氏笔画为序）

王长胤 韦 敏 卢开澄 卢正鼎
张 峰 何炎祥 苏锦祥

秘书 沈旭日

高职、高专计算机系列教材编委会

顾问 胡金柱 瞿 坦

编委（以姓氏笔画为序）

王绍卜 毛芳烈 王前新 叶远谋 刘小芹 向显智
张学礼 张桂宁 张栲勤 李家瑞 邹依琴 唐铸文
黄元山 黄东峰 程时兴 覃晓康 黎秋萍

秘书 曾 光 谢燕群

内 容 简 介

本书以 Visual Basic 6.0 中文版为语言背景, 通过实例介绍了可视化编程的概念与步骤, 常用内部控件的功能和用法, Visual Basic 语言的基础知识, 选择结构设计、循环结构设计、数组、过程、变量与过程的作用域、菜单与工具栏、图形图像设计、多重窗体与多重文档等程序设计技术。针对初学者的特点, 全书力求通俗易懂、简洁实用。书中选有大量的例题, 便于读者更好地理解。各章均附有习题。

本书可作为高职、高专院校计算机专业的计算机教材, 也可供有关培训班及自学者使用。

前 言

Windows 是一个面向图形对象的操作平台。在这个平台上开发的应用软件具有前后一致的用户界面和指令结构。用户可以在多个窗口之间切换执行和交换数据,它提供了一个直观的、图形丰富的工作环境。但是,在较长一段时间里,支持 Windows 图形环境下应用程序开发的软件却很少,从而使开发应用程序的难度加大,不易掌握。Visual Basic 的出现,给 Windows 应用程序的开发者带来福音,使得在 Windows 下开发应用软件不再让人望而却步。又由于 Visual Basic 简单易学、实用,因此正在受到越来越多的计算机专业和非计算机专业人士的重视。根据高职、高专教学大纲对计算机教学的要求,我们编写了《Visual Basic 实用教程》这本教材。

与传统的程序设计语言相比, Visual Basic 是一种可视化的、面向对象的、采用事件驱动方式的结构化高级程序设计语言。本书从介绍 Visual Basic 的特点出发,介绍了 Visual Basic 面向对象程序设计的概念;用举例的方法说明了面向对象的程序设计方法与步骤;通过大量的实例介绍了常用内部控件、数组、过程、数据文件及菜单、对话框、图形等程序设计技术。

由于本书是针对高职、高专初学者的需要而编写的,因此学习本书时不要求读者具有计算机专业知识基础,也不要求读者有其他计算机高级语言的编程知识,只要求读者有 Windows 的初步知识。

本书的第 1 章、第 8 章由武汉职业技术学院刘美珍编写,第 2 章、第 7 章、第 10 章由武汉职业技术学院李露编写,第 3 章、第 4 章、第 5 章由黄冈职业技术学院孙俊编写,第 6 章、第 9 章由黄石高等专科学校冯运仿编写。全书由刘美珍统稿。

由于作者水平有限,缺点与错误在所难免,恳请专家和读者提出宝贵的意见和建议。

编 者

2001.6

目 录

第 1 章 Visual Basic 简介	(1)
1.1 Visual Basic 的特点与安装	(1)
1.1.1 Visual Basic 的特点	(1)
1.1.2 安装 Visual Basic 的环境要求	(1)
1.2 Visual Basic 程序的构成要素	(2)
1.2.1 对象与属性	(2)
1.2.2 事件与过程	(4)
1.3 Visual Basic 程序开发环境	(5)
1.3.1 主窗口	(5)
1.3.2 窗体窗口	(8)
1.3.3 工具箱窗口	(9)
1.3.4 属性窗口	(10)
1.3.5 项目窗口	(12)
1.4 使用 Visual Basic 建立一个简单程序	(12)
习 题	(17)
第 2 章 使用控件创建用户界面	(18)
2.1 窗体	(18)
2.2 标签与文本框	(23)
2.3 命令按钮、单选框和复选框	(27)
2.4 图片框与图像框	(30)
2.5 计时器	(32)
2.6 滚动条	(33)
2.7 OLE 控件	(35)
2.8 数据控件	(38)
2.9 ActiveX 控件	(41)
习 题	(50)
第 3 章 用 Visual Basic 编程	(52)
3.1 Visual Basic 程序的构成	(52)
3.2 Visual Basic 的基本语法	(53)
3.2.1 数据类型	(53)
3.2.2 常量与变量	(56)
3.2.3 变量作用域	(59)

3.2.4 表达式	(59)
3.3 基本语句	(61)
3.3.1 赋值语句	(61)
3.3.2 输入语句	(61)
3.3.3 输出语句	(62)
3.4 函数	(65)
3.4.1 内部函数	(65)
3.4.2 自定义函数	(70)
3.5 使用基本语句及函数的程序举例	(70)
习 题	(72)
第 4 章 程序流程控制	(73)
4.1 条件测试语句	(73)
4.1.1 If 测试语句	(73)
4.1.2 Select Case 测试语句	(78)
4.2 循环	(80)
4.2.1 For...Next 循环结构	(81)
4.2.2 Do...Loop 循环结构	(82)
4.2.3 While...Wend 循环控制结构	(84)
4.2.4 多重循环	(84)
4.3 分支	(86)
4.3.1 On...Goto 条件分支语句	(86)
4.3.2 GoTo 无条件分支语句	(86)
4.4 流程控制语句程序举例	(87)
习 题	(90)
第 5 章 数 组	(91)
5.1 一维数组	(91)
5.2 多维数组	(92)
5.3 动态数组	(93)
5.4 使用数组的程序举例	(95)
习 题	(98)
第 6 章 过 程	(99)
6.1 过程的定义	(99)
6.1.1 Sub 过程	(99)
6.1.2 Function 过程	(102)
6.2 过程的调用	(104)

6.2.1 Sub 过程的调用	(104)
6.2.2 Function 过程的调用	(106)
6.3 参数的传递	(109)
6.3.1 形参与实参	(109)
6.3.2 参数的传送方式	(110)
6.3.3 数组参数的传送	(113)
6.3.4 对象参数的传送	(116)
习 题	(120)
第 7 章 设计菜单	(121)
7.1 设计下拉式菜单	(121)
7.1.1 认识菜单编辑器	(122)
7.1.2 设计菜单	(123)
7.1.3 编写菜单事件过程	(125)
7.1.4 设计菜单项状态	(126)
7.2 设计弹出式菜单	(128)
习 题	(129)
第 8 章 绘 图	(130)
8.1 坐标系统	(130)
8.1.1 标准系统	(130)
8.1.2 自定义坐标系统	(131)
8.2 绘图方法	(133)
8.2.1 清屏与画点	(133)
8.2.2 画直线	(134)
8.2.3 矩形	(137)
8.2.4 圆	(138)
8.2.5 画椭圆	(138)
8.2.6 画弧	(139)
8.2.7 颜色	(140)
习 题	(142)
第 9 章 文 件	(143)
9.1 文件控件	(143)
9.1.1 驱动器列表框	(144)
9.1.2 目录列表框	(144)
9.1.3 文件列表框	(145)
9.2 文件处理语句和函数	(146)

9.2.1 与文件系统相关的语句	(147)
9.2.2 与文件系统相关的函数	(149)
9.3 顺序文件	(151)
9.3.1 顺序文件的打开	(151)
9.3.2 顺序文件的读操作	(153)
9.3.3 顺序文件的写操作	(154)
9.3.4 顺序文件的关闭	(155)
9.4 随机文件	(157)
9.4.1 声明记录类型变量	(157)
9.4.2 打开随机文件	(158)
9.4.3 随机文件的读操作	(159)
9.4.4 随机文件的写操作	(159)
9.4.5 随机文件的浏览操作	(162)
9.5 二进制文件	(164)
习 题	(165)
第 10 章 多重窗体与多重文档	(166)
10.1 多重窗体	(166)
10.2 多重文档	(169)
习 题	(172)
附录 A 对象的属性	(173)
附录 B 对象的事件	(178)
附录 C 对象的方法	(180)

第 1 章

Visual Basic 简介

1.1 Visual Basic 的特点与安装

1.1.1 Visual Basic 的特点

Visual Basic(简称 VB)是 Microsoft 公司于 1991 年推出的 Windows 应用程序开发工具,也可把它称为 Windows 上开发的程序设计语言。它从根本上改变了传统的程序设计模式,简化了 Windows 程序设计。

由于 Visual Basic 的功能十分完善,又易于学习,使用起来也很方便,因此,是一种受欢迎的高级语言。它的编程系统用一种十分巧妙的方法将 Windows 编程的复杂性“封装”起来了。它综合运用了 Basic 语言和新的可视设计工具,既保留了 Windows 所特有的优良性能和图形工作环境,同时又提供了编程的简易性。

Visual Basic 是首批采用事件驱动编程机制的计算机语言之一。事件驱动是一种适用于图形用户界面的编程方式。传统的编程是面向过程的,程序设计人员总要关心什么时候发生什么事情。而用事件驱动方式设计程序时,程序员不必给出按精确次序执行的每个步骤,而只是编写响应用户动作的程序,例如,选择命令、单击鼠标按键、装入窗体、移动鼠标或选取图片。如果将选择命令、单击鼠标按键、装入窗体、移动鼠标、选取图片动作称为事件,则 Visual Basic 又可称为面向事件的软件。因此学习 Visual Basic 编程则可看成是如何建立对话框或窗口。可以用 Visual Basic 建立菜单、正文框、滚动条、命令按钮、标记、图片等。Visual Basic 6.0 中文版是 Visual Basic 的当前最新版本。该版本在 Visual Basic 版本的基础上增加了企业应用的开发能力、数据访问能力及 Internet 应用能力,改进了集成开发环境和自动提示能力。

1.1.2 安装 Visual Basic 的环境要求

1. 硬件要求

① 80486 或 Pentium(奔腾级)CPU 的个人计算机。

- ② 16MB 以上的内存。
- ③ 128MB 以上的硬盘可用空间。
- ④ 鼠标或类似的设备。
- ⑤ 光驱。

2. 软件要求

- ① Microsoft Windows 95 以上版本或 Windows NT 4.0 操作系统。
- ② Visual Basic 6.0 软件。

1.2 Visual Basic 程序的构成要素

1.2.1 对象与属性

由于 Visual Basic 是一种面向对象的程序设计语言，因此准确地理解对象的概念，是设计 Visual Basic 的重要一环。

1. 对象的概念

现实生活中的“对象”就是人们常说的“东西”。太阳系可以看是一个对象，而太阳系中的太阳、地球又都可看成为一个“对象”。“对象”既然是东西，那么就应该有个名称来代替他，以免相互混淆。

Visual Basic 本身由“对象”组合而成，而这些“对象”又是由系统设计好了的，可以直接使用或对其进行操作。例如，以后要介绍的窗体窗口、工具箱窗口、工具箱中的控件都是对象。由于这些对象是由系统设计好并提供给用户使用的，其移动、缩放等操作也是由系统预先规定好的，因此，使用这些对象比一般的面向对象程序设计中的操作要简单得多。Visual Basic 系统会给每个对象指定一个缺省的名称，例如，窗体可用 From1、命令控件可用 Command1 命名。但一般根据需要，用户都应为其重新命名。命名的原则是能够望文生义。在 Visual Basic 中，为了达到“一看见对象名，就立即知道它是属于哪一种对象类型”的目的，可以在对象之前加上代表该类型的字符串。例如，cmd 代表“命令按钮”，frm 代表“窗体”。

表 1.1 所示是一些常用对象的前导符。

表 1.1 常用对象的前导符

对 象	前导符
窗体 (Form)	frm
复选框(CheckBox)	chk
组合框(ComboBox)	cbo
命令按钮(CommandButton)	cmd
数据控件(data)	dat
目录列表框(DirListBox)	dir

续表

对 象	前导符
磁盘驱动器列表框(DriveListBox)	drv
文件列表框(FileListBox)	fil
框架(Frame)	fra
水平滚动条(HScrollBar)	hsb
图像(Image)	img
标签(Label)	lbl
直线(Line)	lin
列表框(ListBox)	lst
菜单(Menu)	mnu
选项按钮(OptionButton)	opt
OLE	ole
图片框(pictureBox)	pic
形状(Shape)	shp
文本框(TextBox)	txt
记时器(Timer)	tmr
垂直滚动条(VScrollBar)	vsb

2. 对象的属性

属性是一个对象的特征，也就是说，一个对象的特征需用属性来描述。在 Visual Basic 中，把对象的属性常写成“对象·属性”，这进一步说明属性是属于对象的。一个对象会有很多属性，而该对象所有属性的集合就叫属性表。

例如，桌子、车子的属性表：

桌子的属性表

车子的属性表

形状

发动机功率

颜色

颜色

材质

座位

.....

形状、颜色、材质……是桌子的属性表，用来说明桌子的特性。发动机、颜色、座位……是车子的属性表，用来说明车子的特性。

Visual Basic 中常见的属性有标题 (Caption)、控件名称 (Name)、颜色 (Color)、字体大小 (FontSize)、是否可见 (Visible) 等，它们都是用来说明对象的特性。Visual Basic 中有一属性窗口，此窗口中含有各种属性，用户可在属性列表中为具体对象选择所需要的属性。如何为对象设置属性将在本章第三节中介绍。设置属性时一定要注意对象，因不同的对象所具有的属性也可能不一样。例如，窗体 (Form)、命令 (Command) 按钮有 Caption 属性，但 Text 文本框则没有。Text 文本框有 Text 属性而窗体、命令按钮则无。

1.2.2 事件与过程

1. 事件 (Event)

在 Visual Basic 中, 事件定义为: 预先设置好的能够被对象识别的动作。例如, 用鼠标在对象上单击时, 这个单击操作就是一种作用在对象上的事件。事件也可称为是一种加在对象上的作用。Visual Basic 中常用的事件有单击 (Click)、双击 (Double Click)、装入 (Load)、移动鼠标 (Mouse move) 等, 这些不同的事件可以作用在不同的对象上。反之, 不同的对象能够识别的事件是不一样的。为什么不同的对象能识别不同的事件呢? 这就是下面要讨论的问题。

2. 事件过程 (Event Procedure)

何为事件过程? 简单地说, 事件过程就是用户编写的程序代码。若对象响应某个事件后所执行的操作是通过一段程序代码来完成的, 则这样的一段程序代码叫做事件过程。一个对象可以识别一个或多个事件过程, 因此可以用一个或多个事件对用户或系统的事件作出响应。

编写事件的一般格式如下:

```
Sub 对象名称_事件名称()
```

```
.....
```

```
    事件过程
```

```
.....
```

```
End Sub
```

说明: “对象名称”指的是该对象的 Name 属性的值。

“事件名称”是由 Visual Basic 预先定义好的赋予该对象的事件, 而这个事件必须是对象所能识别的。

例如,

```
Sub Command1_Click()
```

```
    Text1.Text= “早安”
```

```
End Sub
```

“Command1”是命令控件的名字, Visual Basic 中的控件都为对象。“Click 的中文含义为单击, 实际上是一个动作, 为事件名称。

执行上面语句后, 会在 Text1 框中显示“早安”两个字。

```
Sub 结束_Dblclick()
```

```
End
```

```
End Sub
```

“结束”是命令控件的名字, “Dblclick”是事件名称。“End”是结束语句, 执行此事件过程, 则程序就结束。

3. 方法 (Method)

Visual Basic 中将对象原来就内含的函数或过程叫做方法, 方法是固定的, 不能修改的,

而且也是看不见的。方法的操作与过程、函数的操作相同。其调用的格式如下：

对象名称.方法名称[参数]

例如，Myform.Print “早安”

其中，“Print”为方法。执行上面语句，则在 Myform 的窗体上显示“早安”两字。

例如，Printer.Print “早安”

执行此句，则在打印机上输出“早安”两字。

从上例中也可以看出，同一方法可以用于不同的对象。对象不同，执行作用的设备也不一样。上例中，一个在屏幕上输出，一个在打印机上输出。

调用方法时，也可以省掉对象，此时将调用的方法应看成为当前对象的方法。一般把当前窗体作为当前对象，例如，Print “早安”运行时，也可在当前窗体上显示字符串“早安”，但为了避免不确定性，最好使用“对象.方法”的形式。

Visual Basic 提供了大量的方法，有些方法可以适用于多种甚至所有类型的对象，而有些方法只适用少数几种对象。

1.3 Visual Basic 程序开发环境

Visual Basic 应用程序都是在一个封闭的集成环境开发的，这样使得应用程序的开发、测试、查错等功能集于一体，大大降低其操作的复杂性，提高了开发效率。在这一节里，介绍 Visual Basic 集成环境的 5 个窗口。5 个窗口分别为主窗口、窗体窗口、工具箱窗口、属性窗口、项目窗口，如图 1.1 所示。

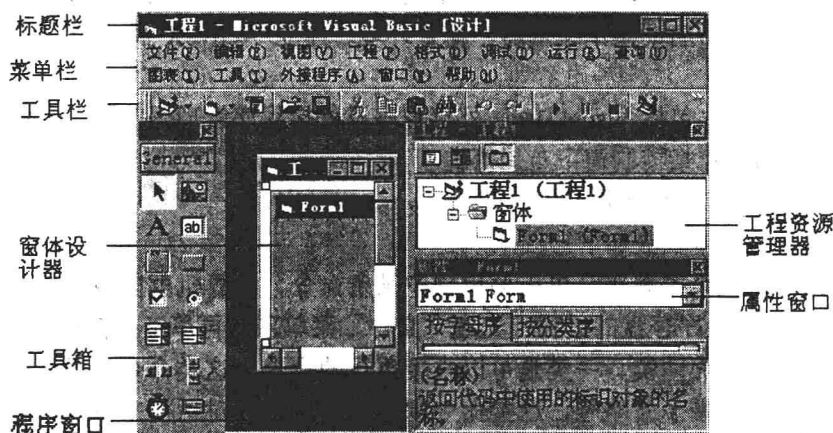


图 1.1 Visual Basic 集成开发环境界面

1.3.1 主窗口

主窗口也称为设计窗口，它位于集成系统的顶部，由标题栏、菜单栏、工具栏等几部分组成，如图 1.2 所示。

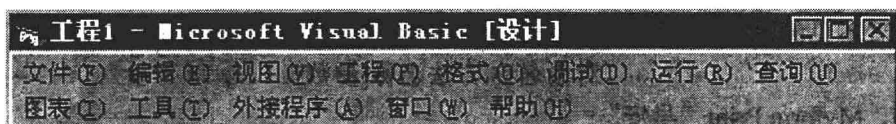


图 1.2 主窗口

1. 标题栏

启动 Visual Basic 后, 标题栏中显示的信息为“工程 1-MicroSoft Visual Basic [设计]”, 其中方括号中的内容“设计”表明是在设计阶段。方括号中的内容也可表明是在运行阶段、中断阶段。

2. 主菜单

主菜单由 13 个菜单项构成。分别是:

- ① 文件 (F): 包括用于文件操作的全部选项, 如打开、保存、添加、删除和打印等选项。
- ② 编辑 (E): 包括正文编辑和控件编辑的操作选项。
- ③ 视图 (V): 包括显示和隐藏集成开发环境构件的操作选项。
- ④ 工程 (P): 包括添加构件到当前工程、在工具箱中加新工具选项等。
- ⑤ 格式 (O): 包括对齐窗体中控件等选项。
- ⑥ 调试 (D): 包括常用查错等选项。
- ⑦ 运行 (R): 包括启动、设置断点和中止当前运行应用程序等选项。
- ⑧ 查询 (U): 包括运行、消除、验证、更改数据、排序等选项。
- ⑨ 图表 (I): 设置文本字体、修改自定义视图、显示分页标记、计算分页等选项。
- ⑩ 工具(T): 包括启动菜单编辑器、配置环境、建立 ActiveX 构件及 ActiveX 控件等选项。
- ⑪ 外接程序(A): 包括 Visual Basic 外接程序及外接程序管理等。
- ⑫ 窗口 (W): 包括控制窗口布局等选项。
- ⑬ 帮助 (H): 包括获取帮助信息的选项。

选菜单命令的具体方法如下:

先用鼠标单击某个菜单项, 再击菜单项中某条命令, 计算机就能执行相应的命令。图 1.3 所示为将写好的程序存盘的过程: 先点“文件”菜单项, 再单击“保存工程”命令。

菜单中的命令分为两类, 一类是可以直接执行的命令, 这类命令后面不带任何信息; 另一类则在命令后面带了省略号, 这类命令则需要通过打开对话框来执行。例如, 当鼠标单击工程“另存为 (E) ...”命令后, 在屏幕上显示图 1.4 所示对话框, 用户只要按提示输入信息就可将信息存入别的磁盘。

在有些命令中还带有其他一些信息。例如, “新建工程(N) CTRL+N”右侧的“Ctrl+N”称为热键, 可以用热键代替所对应的命令, 而且可以不必打开菜单。例如, 按下“Ctrl+N”则可以执行新建工程命令。

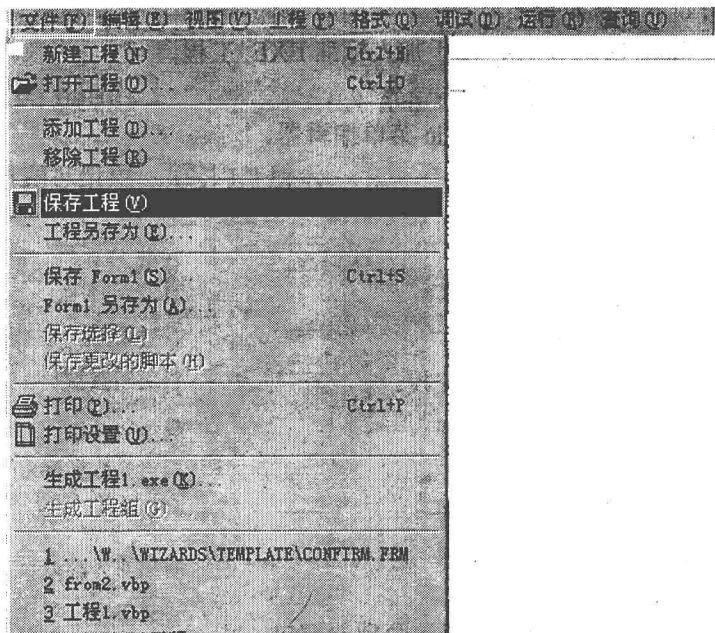


图 1.3 文件菜单

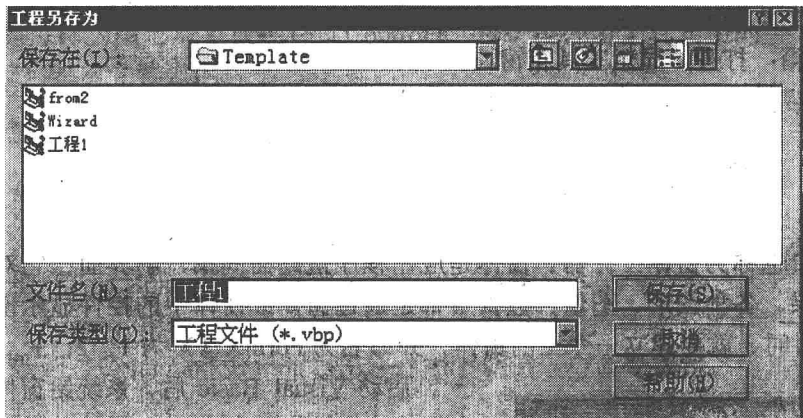


图 1.4 “工程另存为”对话框

3. 工具条

工具条位于主窗口的底部，它以图形的形式提供了部分常用的菜单命令。用户只需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。工具条中有 20 个图标，如图 1.5 所示。

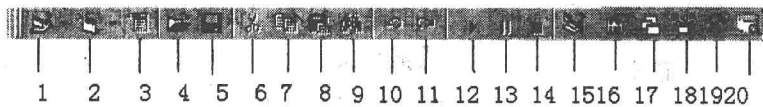


图 1.5 工具条按钮

它们的作用如下（从左至右）

- ① 添加 Standard EXE 工程：用于加入标准 EXE 工程。
- ② 添加窗体：加入新窗体。
- ③ 菜单编辑器：打开 Visual Basic 菜单编辑器。
- ④ 打开工程：打开指定工程。
- ⑤ 保存工程：保存当前工程。
- ⑥ 剪切：剪切文本、控件。
- ⑦ 复制：复制文本、控件。
- ⑧ 粘贴：粘贴文本、控件。
- ⑨ 查找：显示查找窗口，执行查找。
- ⑩ 恢复：恢复以前的操作。
- ⑪ 重复：重复操作。
- ⑫ 启动：开始执行。
- ⑬ 中断：暂停执行。
- ⑭ 结束：终止执行。
- ⑮ 工程资源管理器：显示工程资源管理器窗口。
- ⑯ 属性窗口：显示属性窗口。
- ⑰ 窗体布局窗口：显示窗体布局窗口。
- ⑱ 对象浏览器：显示对象浏览器。
- ⑲ 工具箱：打开、显示工具箱窗口。
- ⑳ 数据视图窗口：显示数据视图窗口。

1.3.2 窗体窗口

窗体窗口也称为窗体设计器。窗体是应用程序最终面向用户的界面，它对应于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。建立一个新的项目时，就要建立一个空的窗体，并命名为 FormX（X 为 1, 2, 3...）。

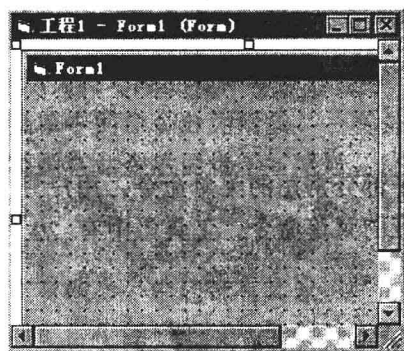


图 1.6 窗体

启动 Visual Basic 后，系统给窗体一个缺省名 Form1。其操作区布满了小点，这些小点是供对齐使用的。如果要清除这些小点，改变点与点之间的距离则可通过“工具”→“选项”→“通用”标签中的“窗体网络设置”栏来设置，如图 1.6 所示。

窗体由窗体设计器生成。在 MDI 开发环境下，窗体设计器里面的窗体会被 8 个“调整句柄”框住，用户可用“调整句柄”来调整对象的大小。但要注意的是，只有“调整句柄”呈现中空为黑点时才能调整对象的大小，也就是说只有窗体是“活动的”时才能对其进行操作。

（窗体也像一块布，在这块“布”上可以画出组成应用程序的各个构件）因此也可以说