

Visual Basic 6.0

编程

新思维创作室 编著



人民邮电出版社

内 容 提 要

本书主要介绍 Visual Basic 6.0 编程的基础知识和实用技巧。书中内容包括: Visual Basic 语言、ActiveX 控件、窗体设计、鼠标和键盘事件、数据库编程、客户和服务端编程、ActiveX 技术, 以及制作帮助系统等。

本书内容浅显易懂、概念性强, 适合 Visual Basic 的初学者和具有一定基础的编程人员阅读, 也可供大专院校的师生学习和参考。

计算机精品软件丛书

Visual Basic 6.0 编程

-
- ◆ 编 著 新思维创作室
责任编辑 潘 涛
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销
 - ◆ 开本: 787 × 1092 1/16
印张: 21.25
字数: 515 千字
印数: 1 - 8 000 册

1998 年 12 月第 1 版

1998 年 12 月北京第 1 次印刷

ISBN 7-115-07577-8/TP·961

定价: 29.00 元



随着 Microsoft Windows 操作系统的出现，计算机给人们带来了一个直观的、崭新的世界。但是在 Visual Basic 出现之前，由于进行 Windows 下的应用程序开发需要训练有素的 C++ 程序员和丰富的 MFC 知识，这些给常规的应用程序的设计人员带来了巨大的障碍，使得人们对进行 Windows 下的应用程序开发敬而远之。但是，随着 1991 年 MS Visual Basic 的出现，一切都变了。人们发现，利用 Visual Basic，开发 Windows 下的应用程序原来是这样的简单。所有的一切只需要从本书开始。

Visual Basic 是 Microsoft 公司于 1991 年推出的 Windows 应用程序开发工具，短短的几年时间，就成为 Windows 流行的编程语言，从根本上改变了传统的程序设计模式，大大简化了 Windows 应用程序设计。

Visual Basic 与传统的语言不同，它使用可视化、交互式的方法来开发 Windows 下的应用程序，由于 Visual Basic 的交互特性，因此你可以一边运行着你的应用程序，一边进行界面的调整和程序代码的调试。通过这种方式，代码运行的效果可以在开发时进行测试，而不必等到编译完成以后。

Visual Basic 是在原有 BASIC 语言的基础上进一步发展，至今包含了数百条语句、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。

Visual Basic 3.0 版之前的控件 (Controls) 上 16 位的 VBX 文件，自从 4.0 版之后，Visual Basic 就提供了 32 位 OCX 的支持了。自从 Visual Basic 5.0 版之后，Microsoft 公司将 OCX 技术和其他的一些关键技术组合到了一起，制订出了新的技术规范，并且称之为 ActiveX(TM) 技术。ActiveX(TM) 技术仍然遵从 OLE2.0 的规范，故而继续可以像原来的 OLE 应用那样使用其他应用程序提供的功能，例如 Microsoft Word 字处理器，Microsoft Excel 电子数据表及其他 Windows 应用程序。甚至可直接使用 VBP 或 VBE 创建的应用程序和对象。

Visual Basic 具有强大的数据库访问能力，在 Visual Basic 内部具有内嵌的 DAO 对象，在进行远程访问的时候，还可以使用 RDO 对象。Visual Basic 在进行数据库对象编程时，是将数据库系统作为一个特殊的对象来进行处理的。

利用 ODBC，Visual Basic 的数据库对象可以非常方便地访问无论是本地的还是远程的数据库。对于具有安全性设置的数据库系统，Visual Basic 可以充分利用其安全机制，对于有事务和日志的大型数据库系统，Visual Basic 可以方便地实现分布式的数据库处理，例如数据访问的并发问题、批量数据的更新的恢复等问题。

在 Windows 操作系统下应用程序的一个显著的特点就是多媒体化。Visual Basic 可以非常方便地实现 Windows 下的多媒体应用程序的设计或者是功能的添加。

第一章 Visual Basic 入门.....	1
1.1 VB 开发环境.....	1
1.1.1 IDE (集成式开发环境)	1
1.2 用 VB 开发项目	12
1.2.1 第一个 VB 项目	12
1.3 几个常用属性、方法、事件	21
1.3.1 常用属性.....	21
1.3.2 常用方法.....	22
1.3.3 输入焦点.....	23
1.3.4 TabIndex	24
1.4 定制工作环境	24
1.4.1 Editor 标签.....	25
1.4.2 Editor Format 标签.....	26
1.4.3 General 标签.....	26
1.4.4 Docking 标签.....	27
1.4.5 Environment 标签.....	28
1.4.6 Advanced 标签	29
第二章 Visual Basic 语言.....	31
2.1 常量.....	31
2.2 变量.....	32
2.2.1 变量声明.....	33
2.2.2 变量类型.....	35
2.2.3 变换变量类型	39
2.3 运算符和表达式	40
2.3.1 算术运算符	41
2.3.2 关系运算符和逻辑运算符	42
2.4 流程控制语句	43
2.4.1 条件语句.....	43
2.4.2 循环语句.....	45
2.5 子程序	47
2.6 函数.....	48

第三章 基本 ActiveX 控件	49
3.1 文本框 (TextBox) 和标签 (Label) 控件	49
3.1.1 文本框 (TextBox) 控件	49
3.1.2 标签 (Label) 控件	52
3.2 命令按钮控件 (CommandButton)	52
3.3 列表框控件和组合框控件	53
3.3.1 列表框(ListBox)控件	54
3.3.2 组合框 (ComboBox) 控件	56
3.4 公共对话框控件	59
3.4.1 文件对话框	60
3.4.2 颜色对话框	65
3.4.3 字体对话框	67
3.4.4 打印对话框	69
3.5 文件列表框、目录列表框与盘符列表框	71
第四章 窗体设计	75
4.1 用户接口界面——窗体的设计准则	75
4.2 设置窗体属性	75
4.2.1 窗体标题栏	76
4.2.2 窗体的装入、显示和隐藏	77
4.2.3 窗体事件	78
4.3 窗体菜单栏	79
4.3.1 菜单程序设计	81
4.3.2 菜单编辑器	82
4.3.3 菜单项的控制	84
4.3.4 弹出式菜单	92
第五章 鼠标和键盘事件	97
5.1 鼠标事件	97
5.1.1 Click 事件	97
5.1.2 DblClick 事件	98
5.1.3 MouseDown 事件	98
5.1.4 MouseUp 事件	99
5.1.5 MouseMove 事件	99
5.2 键盘事件	101
5.2.1 KeyPress 事件	101
5.2.2 KeyDown 和 KeyUp 事件	102
5.3 鼠标光标的形状	103
5.4 拖放	104

5.4.1 与拖放有关的属性、事件和方法	105
5.4.2 改变控件的位置	108
第六章 程序调试与错误处理	111
6.1 Visual Basic 运行模式及错误类型	111
6.1.1 Visual Basic 的模式	111
6.1.2 Visual Basic 的错误类型	113
6.2 中断模式	114
6.2.1 用监视表达式监视数据	116
6.2.2 Immediate 窗口	118
6.3 错误处理	119
6.3.1 错误处理子程序	119
6.3.2 错误的模拟与识别	122
6.3.3 特别的调试考虑	125
第七章 数据库基础	129
7.1 基本概念和术语	130
7.2 结构化查询语言 SQL 简介	132
7.3 创建一个数据库	132
7.3.1 规划一个关系型数据库	133
7.3.2 数据库管理程序	134
7.3.3 创建一个关系型数据库	136
7.3.4 数据表结构的更改和数据输入	141
7.4 Data 数据控件	143
7.4.1 不写代码的数据库访问程序	144
7.4.2 不编写代码的数据库访问程序	144
7.4.3 Data 控件的属性、方法和事件	147
7.4.4 记录集 (Recordset) 小结	149
7.4.5 DBComobox 数据绑定控件	158
7.4.6 DBListBox 数据绑定控件常用属性、方法和事件	158
7.4.7 DBComobox 数据绑定控件的属性、方法和事件	160
7.4.8 DBListBox 和 DBComobox 的重要附加属性	160
7.5 DBGrid 控件的使用方法	161
7.5.1 DBGrid 控件概述	161
7.5.2 DBGrid 控件的属性、方法和事件	162
7.5.3 数据绑定控件	169
7.6 数据窗体向导的使用	170
7.6.1 关于可添加工具 (Add-ins)	170
7.6.2 把数据窗体向导加入到 Visual Basic 中	171
7.6.3 数据窗体向导的使用方法	172

第八章 数据访问对象的使用	179
8.1 几个概念的解释	180
8.1.1 数据库引擎	180
8.1.2 数据仓库	180
8.1.3 数据库编程分类	180
8.2 数据访问对象的编程初步	181
8.3 数据访问对象的层次模型	185
8.3.1 数据对象层次模型概述	186
8.4 数据对象层次模型的属性和方法	189
8.4.1 DBEngine 对象	190
8.4.2 工作区集和对象	192
8.4.3 数据库集和对象	193
8.4.4 TableDef 集对象	194
8.4.5 字段和索引集中的对象	197
8.5 DAO 所能够访问的三类数据库	200
8.6 数据访问对象的深入编程	200
8.6.1 创建和修改一个数据库	200
8.6.2 关于关系和引用完整性	204
8.6.3 记录集的操作	205
8.6.4 记录集中的记录浏览	214
8.6.5 记录集中的记录位置	216
第九章 客户机/服务器编程	219
9.1 客户机/服务器基础	219
9.1.1 客户机/服务器的系统结构	221
9.1.2 客户端/服务器的工作分工	222
9.1.3 三层客户端/服务器模型	223
9.1.4 客户/服务器的工作方式	223
9.1.5 客户/服务器工作方式的优点	224
9.2 使用 ODBC	225
9.2.1 ODBC 概述	225
9.2.2 安装和配置 ODBC 驱动程序	226
9.3 Visual Basic ODBC 的访问	232
9.4 远程数据对象的层次模型	234
9.4.1 rdoEngine 对象	235
9.4.2 rdoEnvironment 对象和 rdoenvironment 对象集	237
9.4.3 rdoConnection 对象和 rdoConnection 对象集	240
9.4.4 rdoTable 对象	244
9.4.5 rdoColumn 对象	246

9.4.6 rdoPreparedstatement 对象和 rdoPreparedstatement 对象集	249
9.4.7 rdoParameter 对象	252
9.4.8 rdoResultset 对象	252
9.5 远程数据服务	255
9.5.1 远程数据服务	255
9.6 远程数据控件	255
第十章 OLE 和 ActiveX 技术	257
10.1 OLE 技术	258
10.1.1 OLE Automation 技术	258
10.1.2 对象链接与嵌入的介绍	259
10.1.3 OLE 容器控件	260
10.1.4 在设计阶段使用 OLE 控件	260
10.1.5 OLE 容器控件的弹出式菜单	264
10.1.6 在运行阶段使用 OLE 控件	268
10.1.7 OLE 控件的属性	270
10.1.8 动词和对象激活的使用	274
10.1.9 Updated 事件	276
10.2 ActiveX 技术	276
10.2.1 OLE 与 ActiveX	277
10.2.2 ActiveX 技术综述	277
10.2.3 加载 ActiveX 控件	279
10.2.4 ActiveX 部件	279
10.2.5 创建 ActiveX DLL	280
10.2.6 创建 ActiveX EXE 部件	296
10.2.7 创建 ActiveX 文档	307
第十一章 制作帮助系统	321
11.1 确定所需的信息	321
11.2 将信息分解为模块主题	321
11.3 确定主题之间的连接关系	322
11.4 设计文档的结构	322
11.5 创建文档	322
11.6 创建帮助主题文档	323
11.7 创建帮助项目文件	325
11.8 编译帮助文件	327

第一章 Visual Basic 入门

本章将介绍 Visual Basic 的集成开发环境（IDE）及用可视工具和事件编程开发应用程序的基本原理。我们将建立几个简单的应用程序，介绍可视开发环境的构件。

1.1 VB 开发环境

从本节开始，我们将开始讨论 Visual Basic 编程开发环境（集成式开发环境），本节将简单介绍 Visual Basic 的开发环境（集成式开发环境），主要是 Visual Basic 的菜单、工具条、工具框和程序设计窗口等部分的简单介绍。

1.1.1 IDE（集成式开发环境）

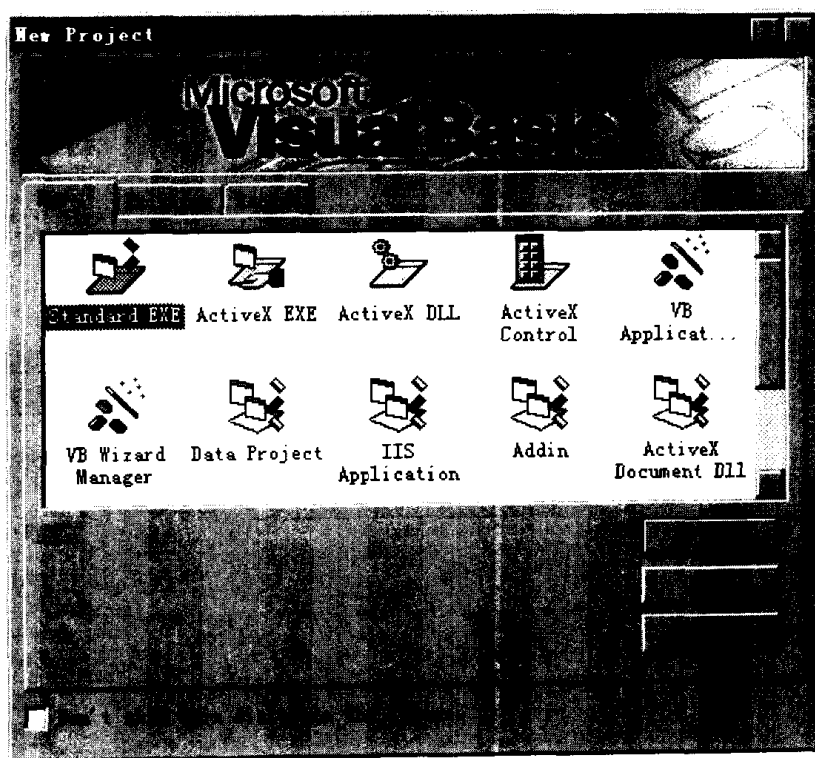


图 1.1 Visual Basic 开启界面

Visual Basic 不仅仅是一种语言，如概述所说，更是一个集应用程序开发、测试和查错等功能于一身的集成式开发环境。

启动 Visual Basic，即可看到如图 1.1 所示的窗口，其中会提示选择要建立的项目类型。

在图 1.1 上, 可以看到该对话框中有三个标签, 它们分别是:

- ✓ New (新建项目)
- ✓ Existing (现存项目)
- ✓ Recent (最新项目)

在 New (新建项目) 标签中, 列有 Visual Basic 所能生成的应用程序类型, 它们是:

Standard EXE: 标准 EXE 程序 (Standard EXE) 是典型的应用程序。本书的大多数应用程序都属于标准 EXE 程序。

ActiveX EXE 和 Active DLL: 这类程序在专业版里才有。ActiveX 构件是支持 OLE 的自动化服务器程序, 该两种程序在功能上是一致的, 只是编译结果不同, 一个编译成 EXE 类型的可执行文件, 一个编译成动态链接库。

ActiveX Control: 这类项目也是专业版中才用的特性, 用于开发自己的 ActiveX 控件。

ActiveX Document EXE 和 ActiveX Document DLL: ActiveX 文档实际上是在支持超链接的容器环境中运行的 Visual Basic 应用程序。简单地说, 这个超链接环境就是个 Web 浏览器, 如 Netscape、Internet Explorer 等。

VB Application Wizard: 利用向导(Wizard)可以建立新应用程序的框架, 这个向导通常只有在开发自己的项目时才会用到。修改向导生成的框架代码和从头生成自己的应用程序相比, 即使不会更费劲, 也不会更轻松。

Addin: 可以建立自己的 Visual Basic IDE 加入件。为此要选择 Addin 项目类型。这个课题较深, 相对来说比较复杂。

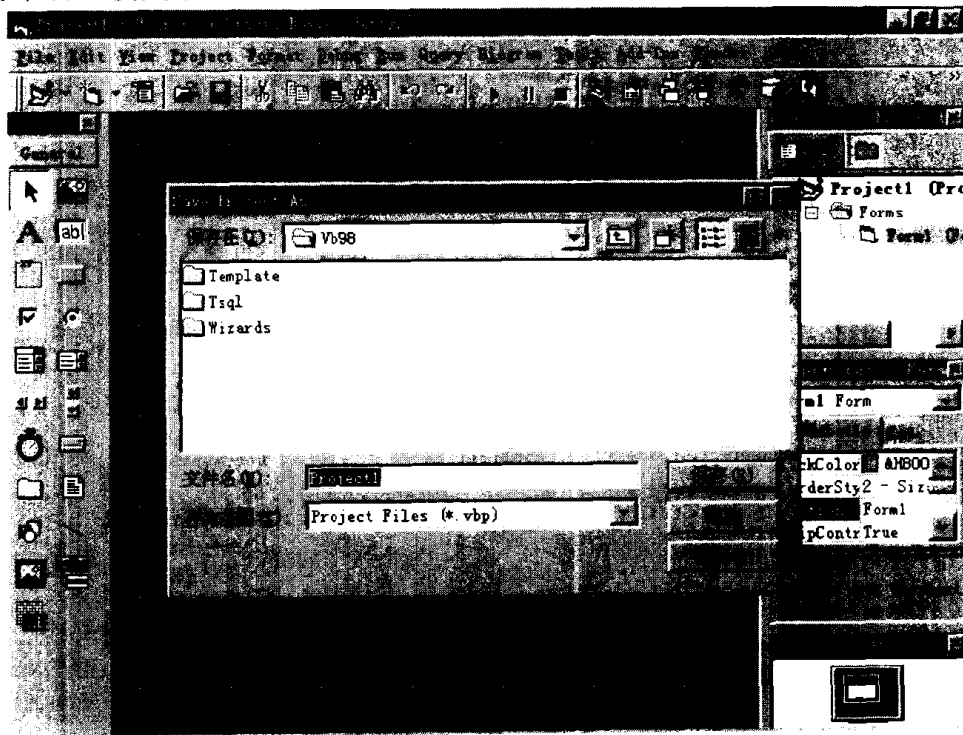


图 1.2 现存项目对话框

切换到 Existing 项目标签下, 可以选择和打开现有的项目。而在 Recent 项目标签下, 可以看到最近用过的项目列表。在日常工作中, 经常要处理同一个项目, 则利用该选项, 可以直接找到新近用过的项目。它们的形式分别如图 1.2 和图 1.3 所示。

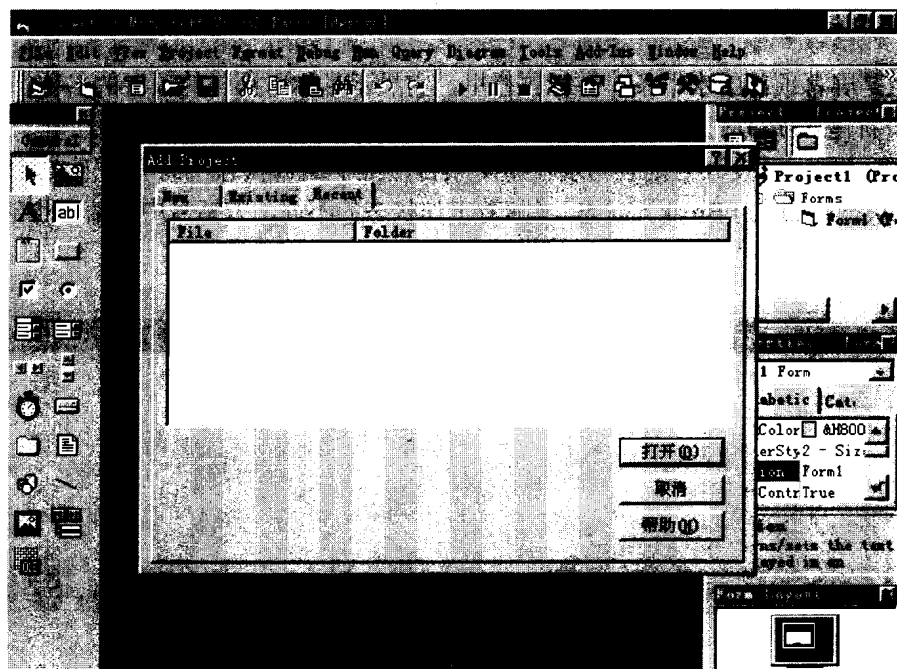


图 1.3 最新对话框

由于第一次使用此对话框，在 Recent 对话框中并不存在最近的工程项目。

选择 New 标签下的 Standard EXE 项，在下一个对话框中选择 OK，则可打开如图 1.4 所示的窗口。

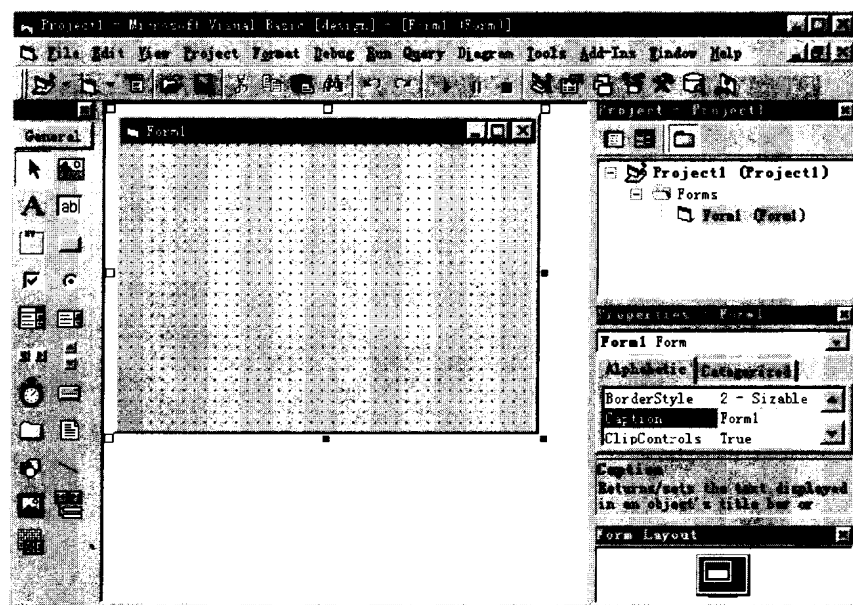


图 1.4 新 Visual Basic 项目

图 1.4 所示的正是 Visual Basic 集成开发环境 IDE。下面分别加以介绍。

1.1.1.1 标题栏

标题栏是屏幕顶部的水平条。在标题栏里，它显示了你所操作的项目的名称，如图 1.5 所示，由于新建项目时，Visual Basic 默认项目名为 Project1，所以在标题栏里，显示的是 Project1。而 design（设计）则表明现在整个项目正处于编辑状态，而不是下文将提到的运行（Run）状态或断点（Break）状态。（随着工作状态的不同，方括号中的信息也随之改变，可能是“run”模式或“break”模式）。



图 1.5 Visual Basic IDE 环境标题栏

1.1.1.2 菜单条



图 1.6 Visual Basic IDE 菜单条

标题栏的下面是集成环境的主菜单，如图 1.6 所示，在菜单条里，包含了运行 Visual Basic 进行编程开发项目所需要的命令。这些菜单命令提供了开发、调试、保存应用程序所需要的工具。基本的菜单命令有：

- File（文件）：如图 1.7 所示，包含打开（Open Project）、保存（Save Project）和删除（Remove）项目、保存（Save Form）和加入窗体（Add Form）以及生成执行文件（Make Project1.exe），打印窗体代码等与文件操作有关的一系列命令和最多到四个的最近用到的项目。

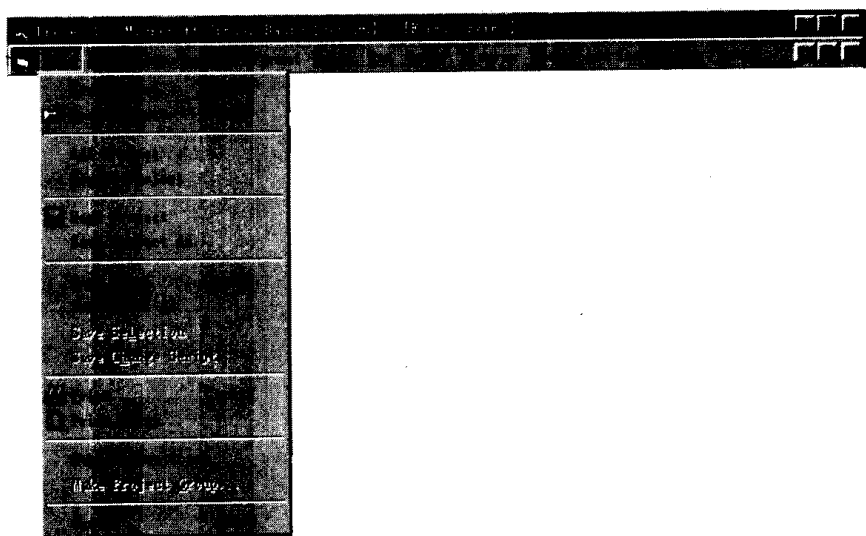


图 1.7 文件菜单项

- Edit（编辑）：如图 1.8 所示，包含了一系列与编辑有关的命令，如撤消上次操作（Undo），剪切（Cut）、粘贴（Paste）以及查找（Find）、替换（Replace）等常用编辑命令和缩进（Indent）、凸出（Outdent）等排版命令。

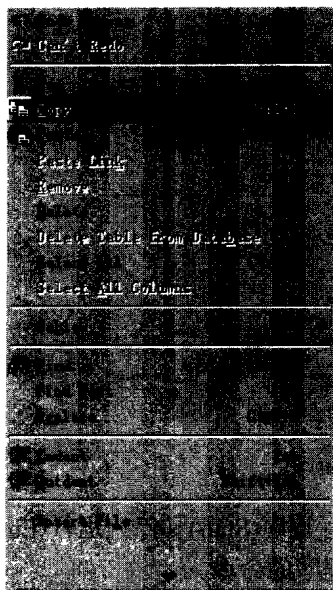


图 1.8 编辑菜单项

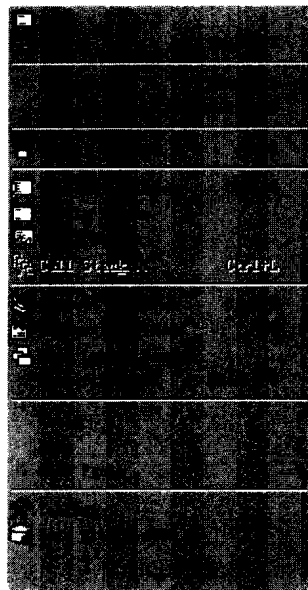


图 1.9 视图菜单项

- **View (视图):** 如图 1.9 所示, 包含了显示和隐藏集成开发环境 (IDE) 元素的命令。如显示或隐藏代码窗口(Code)、显示或隐藏属性窗口(Properties Window)、显示和隐藏工具栏等命令。

- **Project (工程)**: 如图 1.10 所示, 包含将窗体、模块加入当前项目、引用 Windows 对象和工具框新工具的命令。

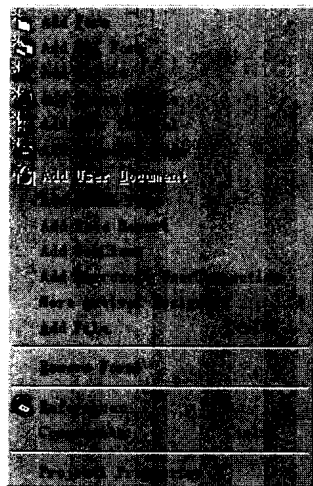


图 1.10 工程菜单项

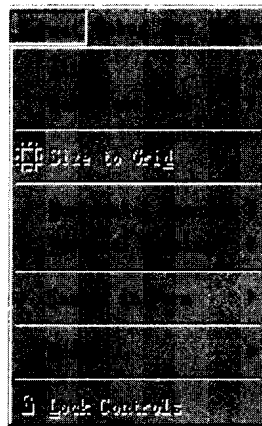


图 1.11 格式菜单项

- **Format (格式):** 如图 1.11 所示, 包含对齐窗体控件 (Align) 和锁定窗体控件使之不能移动(Lock Controls)等命令。

- **Debug (调试)**: 如图 1.12 所示, 包含对程序进行调试的各个命令, 如逐语句(Step Into)、逐过程(Step Over)运行程序, 设置监视(Add Watch)等命令。

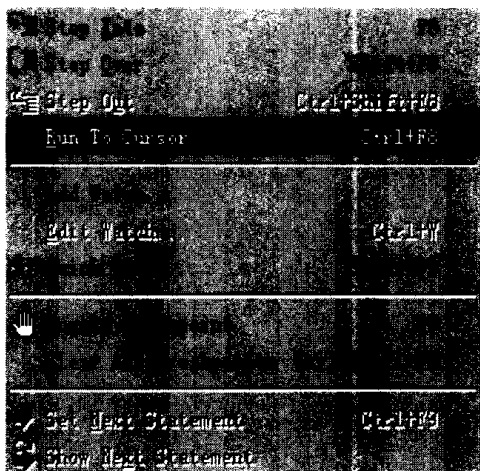


图 1.12 调试菜单项

• Run (运行): 如图 1.13 所示, 包含了运行程序(Start)和停止运行(End), 重新运行程序(Restart)等在 Visual Basic IDE 环境下运行应用程序的命令。

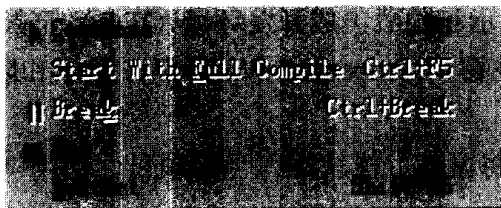


图 1.13 运行菜单项图

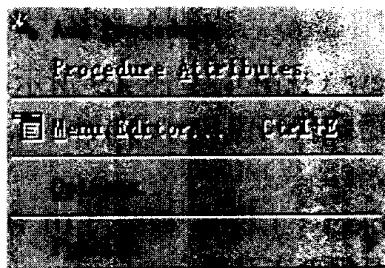


图 1.14 工具菜单项

• Tools (工具): 如图 1.14 所示, 包含了启动菜单编辑器 (Menu Editor) 的命令和对 IDE 环境进行配置 (Options) 的选项, 还包含了建立 ActiveX 构件和 ActiveX 控件时所要的工具。

• Add-Ins (外接程序): 如图 1.15 所示, 缺省包含了可视化数据管理器 (VisData) 外接程序 (Visual Data Manager), 编程人员可以通过外接程序管理器加入或删除其它的外接程序。

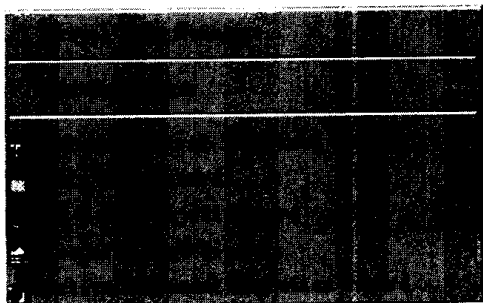


图 1.15 外接程序菜单

• Window (窗口): 如图 1.16 所示包含了设计窗体设计窗口中各个窗体和代码窗口的布局的命令, 如水平平铺(Tile Horizontally)、垂直平铺(Tile Vertically)、层叠(Cascade)、排列图标(Arrange Icons)等命令。

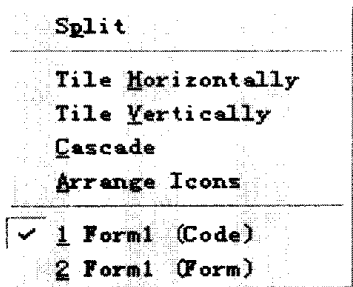


图 1.16 窗口菜单项图

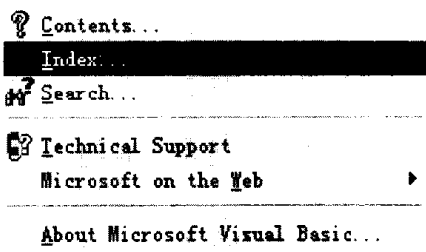


图 1.17 帮助菜单项

- Help (帮助): 如图 1.17 所示, 包含了 Visual Basic 的帮助信息。

1.1.1.3 工具条

工具条位于菜单条的下面, 即主窗口的底部, 它以图标形式提供了部分常用菜单命令的功能, 利用工具条, 可以快速访问这些常用的菜单命令, 而不必下拉菜单条来找菜单命令。除了缺省显示在菜单条下面的主工具条外, 通过选择菜单 View 下的 toolbars 子菜单中的工具条子菜单的名字, 来打开或关闭工具条。可供选择的其它专用工具条有编辑(Edit)工具条、窗体设计(Form Editor)工具条, Debug 调试工具条等。

- Standard 标准工具条: 在菜单的正下方, 如图 1.18 所示。

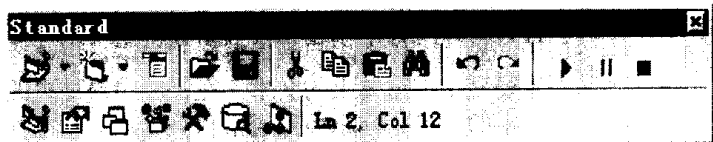


图 1.18 Standard 工具条

- Edit 编辑工具条: 图 1.19 是 Edit 菜单命令的快捷方式, 包含了 Edit 菜单的命令。



图 1.19 编辑工具条

- Debug 调试工具条: 图 1.20 所示是 Debug 菜单命令的快捷方式。



图 1.20 调试工具条

- Form Editor 窗体设计工具条: 图 1.21 所示是 Format 菜单命令的快捷方式。

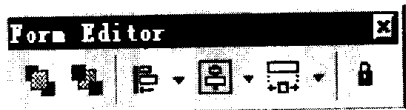


图 1.21 窗体设计工具条

1.1.1.4 工具框(Toolbox)

Toolbox 工具框中包含了控件的图标，可以将这些控件拖到窗体中，生成应用程序的人机对话接口。

缺省情况下，工具框中包含有如图 1.22 所示的指针图标和 20 个 ActiveX 控件图标，它们的简单介绍如表 1.1 所示。

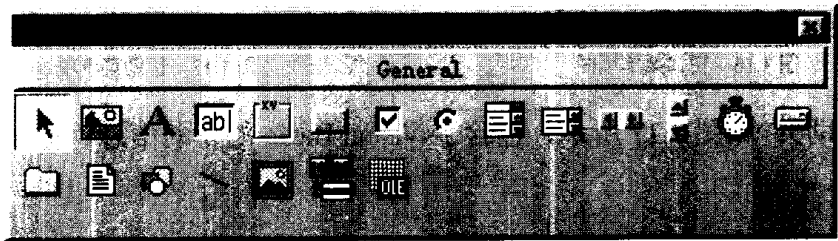


图 1.22 Visual Basic 工具框

表 1.1 VB 常用控件介绍

图标	控件名	功能
	PictureBox（图形框控件）	该控件用于显示图形，用控件的 Picture 属性设置所要显示的图形。
	Label（标记控件）	该控件显示窗体上用户不能编辑的文本。用该控件的 Caption 属性来设置要显示在窗体上的文本。在程序中通过更改 Caption 属性达到更改文本的要求。标记控件通常用来标识其它控件。
	TextBox（文本框控件）	该控件用于显示用户可以编辑的文本。与 Label 控件不同的是，该控件是个微型文字编辑器，利用该控件的 Text 属性，可以设置控件上的文本或读取用户输入的文本。
	Frame（桢控件）	该构件用于组合其它元素，如复选框。
	Command Button（命令按钮控件）	该控件用于在人机交互中提供确定的命令要求。改变该控件的 Caption 属性将改变显示在命令按钮上的文本。
	CheckBox（复选框控件）	复选框控件提供给用户若干个可以选择的选项，用户一次可以选择一个或若干个选项。程序通过判断控件的 Value 属性来判断用户是否选中该选项。选中该控件时，控件的 Value 值为 1，未选中时，控件的 Value 值为 0。
	Option Button（选项钮控件）	选项钮控件一般是成组出现的，用户只能选择一个选项，这一点与 CheckBox 控件不同。选定控件时，控件的 Checked 属性值为 True.否则，控件的 Checked 属性值为 False。
	ListBox（列表框控件）	该控件包含了一系列选项，用户可以选择其中的一个或几个选项。列表框控件中所选项目由控件的 Text 属性给定。而列表框控件的 Sorted 属性确定了表中项目是否排序。

续表

图标	控件名	功能
	ComboBox (组合框控件)	组合框控件与 ListBox 控件类似, 但不同的是组合框控件包含了一个文本编辑字段, 用户既可以选择表中的项目, 亦可以在编辑字段中输入新字符串。利用该控件的 Text 属性就可以得到用户所选择的项目或用户输入的字符串。
	Horizontal ScrollbBox (水平滚动条控件)	用于定性输入一个数据, 由在控件的最小值和最大值之间的位置确定用户的输入值的大小。
	Vertical ScrollBar (垂直滚动条控件)	其作用与水平滚动条控件的作用是一致的, 唯一的区别是滚动的形状不一样。
	Timer (定时器控件)	用于完成确定时间间隔必须进行的任务。用 Interval 属性确定事件发生的时间间隔。
	Drive ListBox (盘列表框控件)	显示系统上的驱动器, 用户在其中可以选择某个驱动器, 如 A 盘、C 盘等。
	Directory ListBox (目录列表框控件)	显示当前盘上的一系列文件目录。
	File ListBox (文件列表框控件)	显示当前目录下的所有文件清单。与上面两个控件一起可以组成一个文件系统, 为用户出现提供文件处理能力。提供给用户访问和处理磁盘、目录、文件的接口。
	Shape (形状控件)	用于在窗体上绘制方框和圆形等图形的功能。
	Line (直线控件)	其属性与 Shape 控件基本一致, 用于在窗体上绘制直线。
	Image (图像控件)	类似于图形框控件, 用于显示图形, 但它只支持图形框控件的几个特性, 需要更少的资源。
	Data (数据控件)	该控件提供了存放在数据库中的数据的一系列访问命令, 它有很多属性, 是数据库编程中很重要的一个数据访问控件, 在数据库操作一章中将重点介绍。
	OLE Container (OLE 容器控件)	OLE 容器控件是一个窗口, 可以将其放在窗口中, 用于放置其它应用出现中的文档, 通过该控件, 用户可以访问支持 OLE 的其它应用程序, 如 Microsoft Excel 文档。