

21

世纪高等教育
计算机技术规划教材



Visual Basic 程序设计

闵忠保 肖守柏 金玲 主编
黄丽 孙敏燕 副主编



人民邮电出版社
POSTS & TELECOM PRESS

21 世纪高等教育计算机技术规划教材

Visual Basic 程序设计

闵忠保 肖守柏 金 玲 主 编

黄 丽 孙敏燕 副主编

人 民 邮 电 出 版 社

北 京

图书在版编目 (CIP) 数据

Visual Basic 程序设计 / 闵忠保, 肖守柏, 金玲主编.
北京: 人民邮电出版社, 2008.8
21 世纪高等教育计算机技术规划教材
ISBN 978-7-115-17907-4

I. V… II. ①闵…②肖…③金… III. BASIC 语言—程序设计—高等学校: 技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2008) 第 103707 号

内 容 提 要

本书以全国计算机等级考试二级 Visual Basic 考试大纲为指导, 以 Visual Basic 6.0 中文版为背景, 通过大量实例, 深入浅出地介绍了 Visual Basic 的基本概念, Visual Basic 语言基础, 顺序结构程序设计, 选择结构程序设计, 循环结构程序设计, 数组, 过程与作用域, 界面设计, 数据文件, 常用键盘、鼠标事件及图形控件, 公共基础知识等。本书概念清楚、循序渐进、层次分明、例题丰富, 适合教师课堂教学和学生自学。书中各章均附有精心设计的练习题。

本书可作为高等职业院校、应用型本科学校 Visual Basic 课程的教材, 也可作为参加全国计算机等级考试的读者的自学用书。

21 世纪高等教育计算机技术规划教材

Visual Basic 程序设计

-
- ◆ 主 编 闵忠保 肖守柏 金 玲
 - 副 主 编 黄 丽 孙敏燕
 - 责任编辑 李 凯
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京世纪雨田印刷有限公司印刷
 - ◆ 开本· 787×1092 1/16
印张 17.25
字数· 423 千字
印数· 1—3 000 册
 - 2008 年 8 月第 1 版
2008 年 8 月北京第 1 次印刷

ISBN 978-7-115-17907-4/TP

定价: 36.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223

反盗版热线: (010)67171154

前 言

Visual Basic 是开发 Windows 环境下图形界面应用程序的计算机语言。它继承了 BASIC 语言易学易用的优点，同时又引入了可视化图形用户界面的程序设计方法和面向对象的编程机制，开发应用效率高，深受广大专业和非专业计算机程序设计人员喜爱，成为当今广泛采用的计算机应用开发语言。

本书主要讲述 Visual Basic 语言的基础知识，并以实例形式详细介绍该软件的基本功能。全书共分 11 章，分别为 Visual Basic 概述，Visual Basic 语言基础，顺序结构程序设计，选择结构程序设计，循环结构程序设计，数组，过程与作用域，界面设计，数据文件，常用键盘、鼠标事件及图形控件，公共基础知识等。为配合非计算机专业人员参加等级考试，本书对数据结构、软件工程、数据库等公共基础知识做了简单介绍，可供读者学习参考。

本书由多年从事计算机基础教学工作的教师编写，在内容编排上注重由浅入深、循序渐进、繁简适当，使之更符合教学规律；在编写过程中，力求内容准确、通俗易懂、简洁实用，使读者能轻松而快速地学会 Visual Basic。为了巩固每章的学习，各章配有精心设计的练习题。

本书可作为高等职业院校、应用型本科学校 Visual Basic 课程的教材，也可作为参加全国计算机等级考试的读者的自学用书。

本书由江西蓝天学院闵忠保、肖守柏、金玲担任主编，黄丽、孙敏燕担任副主编。第 1 章、第 9 章、第 10 章由闵忠保编写，第 11 章由肖守柏编写，第 2 章、第 3 章、第 4 章由金玲编写，第 5 章、第 6 章由黄丽编写，第 7 章、第 8 章由孙敏燕编写。

由于编者水平有限，书中错漏之处在所难免，恳请读者批评指正。

编 者
2008 年 7 月

目 录

第 1 章 Visual Basic 概述	1	2.3 运算符和表达式	26
1.1 Visual Basic 的特点和版本	1	2.3.1 算术运算符和算术表达式	26
1.1.1 Visual Basic 的特点	1	2.3.2 字符串运算符和字符串表达式	27
1.1.2 Visual Basic 的版本	2	2.3.3 日期运算符和日期表达式	27
1.2 Visual Basic 的安装、启动和退出	3	2.3.4 关系运算符和关系表达式	28
1.2.1 Visual Basic 的安装	3	2.3.5 布尔运算符和布尔表达式	29
1.2.2 Visual Basic 的启动和退出	3	2.3.6 运算符的优先顺序	29
1.3 Visual Basic 的集成开发环境	4	2.4 常用内部函数	30
1.3.1 Visual Basic 的主窗口	4	2.4.1 数学函数	31
1.3.2 Visual Basic 的其他窗口	5	2.4.2 字符串函数	32
1.4 可视化编程的一般步骤	8	2.4.3 日期和时间函数	33
1.5 可视化编程的基本概念	11	2.4.4 格式输出函数	33
1.5.1 对象和类	11	2.4.5 随机函数	34
1.5.2 对象的属性、事件和方法	12	2.4.6 颜色函数	35
1.5.3 窗体对象	13	2.5 程序语句	36
1.5.4 控件	15	2.5.1 程序语句书写规则	36
1.6 使用 Visual Basic 帮助系统	16	2.5.2 命令语法格式中的符号约定	37
习题 1	18	习题 2	37
第 2 章 Visual Basic 语言基础	19	第 3 章 顺序结构程序设计	40
2.1 基本数据类型	19	3.1 基本语句	40
2.1.1 数值型数据	19	3.1.1 赋值语句	40
2.1.2 字符型数据	21	3.1.2 注释与暂停语句	41
2.1.3 布尔型数据	21	3.1.3 卸载对象与结束程序	42
2.1.4 日期型数据	21	3.2 数据输出	43
2.1.5 可变类型数据	22	3.2.1 使用 Print 方法输出到窗体	43
2.1.6 对象类型数据	22	3.2.2 输出文本到图片框	48
2.2 常量和变量	22	3.2.3 使用标签输出文本	49
2.2.1 直接常量	22	3.3 数据输入	51
2.2.2 符号常量	23	3.4 使用对话框	55
2.2.3 系统常量	24	3.4.1 输出信息对话框	55
2.2.4 变量及其命名规则	24	3.4.2 输入对话框	58
2.2.5 变量的声明	25	习题 3	60

第 4 章 选择结构程序设计	63	5.7.5 递推法	124
4.1 If 语句	63	5.7.6 最大值、最小值问题	124
4.1.1 使用 IIF 函数	63	习题 5	125
4.1.2 单行选择结构语句		第 6 章 数组	133
If...Then...Else	63	6.1 数组的基本概念	133
4.1.3 块结构条件语句		6.1.1 引例	133
If...Then...Else...End If	65	6.1.2 基本概念	134
4.1.4 嵌套结构条件语句		6.2 静态数组	134
If... ElseIf...Else...End If	66	6.2.1 静态数组的声明	135
4.2 多分支条件选择语句 Select Case	72	6.2.2 数组的基本操作	136
4.3 计时器控件	76	6.2.3 与数组相关的函数	144
4.4 单选按钮和复选框	78	6.3 动态数组	145
4.4.1 单选按钮	78	6.3.1 建立动态数组	146
4.4.2 复选框	81	6.3.2 保留动态数组中原有的数据	149
4.5 框架控件	84	6.4 For Each...Next 语句	150
4.6 命令按钮	85	6.5 控件数组	151
4.6.1 Visible 属性	85	6.5.1 创建控件数组	151
4.6.2 Enabled 属性	86	6.5.2 控件数组的使用	152
4.6.3 Default 属性和 Cancel 属性	86	习题 6	158
4.6.4 Name 属性和 Caption 属性	87	第 7 章 过程与作用域	165
习题 4	88	7.1 Visual Basic 应用程序模块结构	165
第 5 章 循环结构程序设计	94	7.1.1 窗体模块	166
5.1 循环结构的概念	94	7.1.2 标准模块	166
5.1.1 循环结构引例	94	7.1.3 类模块	166
5.1.2 循环结构基本概念	95	7.2 事件过程	167
5.2 Do...Loop 语句	95	7.3 Sub 子过程	167
5.2.1 前测型 Do...Loop 循环	96	7.3.1 引例	168
5.2.2 后测型 Do...Loop 循环	98	7.3.2 Sub 子过程的定义	168
5.3 For...Next 语句	101	7.3.3 Sub 子过程的调用	169
5.4 While...Wend 循环结构	107	7.4 Function 函数过程	170
5.5 循环嵌套	108	7.4.1 Function 函数过程的定义	170
5.6 列表框和组合框	111	7.4.2 Function 函数过程调用	172
5.6.1 列表框	112	7.5 参数传递	173
5.6.2 组合框	116	7.5.1 形参和实参	174
5.7 常用算法介绍	119	7.5.2 传址与传值方式	174
5.7.1 累加、累乘问题	119	7.5.3 传递数组	176
5.7.2 利用循环嵌套打印图形	120	7.6 过程的嵌套与递归调用	177
5.7.3 查找指定条件的数	121	7.6.1 过程的嵌套调用	177
5.7.4 穷举法	123	7.6.2 过程的递归调用	178

7.7 变量的作用域与生存周期	180	9.5.1 随机文件的打开和关闭	217
7.7.1 变量的作用域	181	9.5.2 随机文件的读写操作	218
7.7.2 变量的生存周期	182	9.5.3 随机文件中记录的增、删、 改操作	220
7.8 过程的作用域	183	9.6 二进制文件操作	222
7.8.1 模块级过程	183	9.6.1 二进制文件的打开和关闭	222
7.8.2 全局级过程	183	9.6.2 二进制文件的读写操作	222
7.8.3 调用其他模块中的过程	184	9.7 文件管理	223
习题 7	185	9.8 文件系统控件	224
第 8 章 菜单、工具栏和对话框	187	9.8.1 驱动器列表框	224
8.1 菜单	187	9.8.2 目录列表框	225
8.1.1 菜单编辑器	187	9.8.3 文件列表框	225
8.1.2 下拉式菜单设计	189	习题 9	227
8.1.3 弹出式菜单设计	192	第 10 章 常用键盘、鼠标事件及图形控件	229
8.2 工具栏	194	10.1 键盘事件	229
8.2.1 用手工方式制作工具栏	194	10.1.1 Keypress 事件	229
8.2.2 使用 ToolBar 控件和 ImageList 控件	195	10.1.2 Keydown 事件和 Keyup 事件	230
8.3 公用对话框	197	10.1.3 Keypreview 属性	231
8.3.1 添加公用对话框	197	10.2 鼠标事件	231
8.3.2 公用对话框的使用	197	10.3 图形控件	233
习题 8	200	10.3.1 直线控件	233
第 9 章 数据文件	202	10.3.2 形状控件	234
9.1 文件的概念	202	10.3.3 显示图片	234
9.1.1 文件的分类	202	10.3.4 图片框控件	235
9.1.2 文件的读写和文件缓冲区	203	10.3.5 图像控件	236
9.2 文件指针	203	10.4 滚动条控件	237
9.2.1 文件指针的概念	203	10.4.1 滚动条控件的属性	238
9.2.2 文件指针的操作	204	10.4.2 滚动条控件的常用事件	238
9.3 用户自定义数据类型	205	10.4.3 滚动条控件的应用示例	238
9.3.1 用户自定义数据类型的概念	205	习题 10	241
9.3.2 创建用户自定义类型	205	第 11 章 公共基础知识	242
9.3.3 建立和使用用户自定义类型 变量和数组	206	11.1 基本数据结构与算法	242
9.4 顺序文件操作	210	11.2 程序设计基础	254
9.4.1 顺序文件的打开和关闭	210	11.3 软件工程基础	256
9.4.2 顺序文件的写操作	211	11.4 数据库设计基础	263
9.4.3 顺序文件的读操作	213	习题 11	267
9.5 随机文件操作	217	参考文献	270

第 1 章

Visual Basic 概述

Visual Basic 是美国 Microsoft 公司推出的 Windows 环境下应用程序开发工具。从数学计算、数据库管理、客户机/服务器软件、通信软件、多媒体软件到 Internet/Intranet 软件，都可以用 Visual Basic 开发完成。由于 Visual Basic 易学好用、编程效率高，目前被广泛采用。

1.1 Visual Basic 的特点和版本

Visual Basic 是从 BASIC 语言发展而来的，是开发 Windows 环境下图形用户界面软件的可视化工具。Visual 意指“可视的”，Basic 意指“初学者通用符号指令代码”(Beginners All Purpose Symbolic Instruction Code)。Visual Basic 采用 BASIC 语言作为程序代码，并在原有 BASIC 语言基础上进行了进一步发展。在 Visual Basic 中引入了面向对象的概念，把各种图形用户界面元素抽象为不同的控件，如各种各样的按钮、文本框和图片框等。Visual Basic 把这些控件模式化，为每个控件赋予若干属性和方法来控制其外观及行为。这样，在开发 Visual Basic 应用程序过程中，无须编写大量代码去描述界面元素的外观和位置，只要从 Visual Basic 工具箱中把预先建立好的控件直观地加到屏幕上，这就像使用“画图”之类的绘图程序，通过选择画图工具来画图一样，从而极大地提高了编程效率。

1.1.1 Visual Basic 的特点

Visual Basic 是目前所有图形用户界面程序开发语言中最简单、最容易使用的语言之一。Visual Basic 主要有以下特点。

1. 面向对象的可视化设计平台

利用传统的程序设计语言进行程序设计时，需要花费大量的精力去设计用户界面，且在设计过程中看不到程序的实际显示效果，必须在程序运行时才能观察。如发现界面不满意，还要回到程序中去修改，这一过程常常需要反复多次。Visual Basic 提供的可视化设计平台，为程序员创造了所见即所得的开发环境，程序员不必再为界面设计而编写大量程序代码，只需按设计要求，用系统提供的工具在屏幕上“画出”各种对象，无须知道对象的生成过程，Visual Basic 将自动生成界面设计代码。程序员所要编写的只是实现程序功能的那部分代码。

2. 事件驱动式编程机制

传统的编程方式是面向过程的，程序员必须考虑程序每一步执行的顺序，即程序的执行完全按事先设计的流程来运行，无疑增加了程序员的思维负担。Visual Basic 引入了面向对象

的概念,采用事件驱动式编程机制,在 Visual Basic 图形用户界面应用程序中,用户的动作(即事件)掌握着程序的运行流向,每个事件都驱动一段程序的运行。程序员在设计应用程序时,只要编写若干个具有特定功能的子程序(即事件过程和通用过程),这些过程分别面向不同的对象,但无须考虑它们之间的先后次序,各过程的运行由用户操作对象时引发的某个事件来驱动。

3. 结构化的设计语言

结构化的程序设计语言,是指它能够方便地实现“自顶向下、分而治之、模块化”的程序设计方法。Visual Basic 是在结构化的 BASIC 基础上发展起来的,具有高级程序设计语言的结构化语句、丰富的数据类型、众多的内部函数,便于程序的模块化、结构化设计。其结构清晰,简单易学。在输入代码的同时,编辑器自动进行语法检查。在设计过程中,可随时运行程序,随时调试改正错误,而且在整个应用程序设计好后,编译生成的可执行文件(.exe),可脱离 Visual Basic 环境,直接在 Windows 环境下运行。

4. 充分利用 Windows 资源

Visual Basic 提供的动态数据交换(DDE)编程技术,可以在应用程序中实现与其他 Windows 应用程序建立动态数据交换,以及在不同的应用程序之间进行通信。Visual Basic 提供的对象链接与嵌入(OLE)技术,可以将不同的应用程序链接起来,从而开发出集声音、图像、动画、字处理、电子表格等对象于一体应用程序。Visual Basic 还将 Windows 常用功能程序封装在对象中,如各种通用对话框以及文件菜单的设计等,均共享 Windows 资源。

5. 开放的数据库功能与网络支持

Visual Basic 系统具有很强的数据库管理功能,不仅可以管理 MS Access 格式的数据库,还能访问其他外部数据库,如 FoxPro、dBase、Paradox 等格式的数据库。另外,Visual Basic 还提供了开放式数据连接(ODBC)功能,可以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)直接访问服务器上的数据库,并提供简单的面向对象的库操作命令、多用户数据库的加锁机制和网络数据库编程技术,为单机上运行的数据库提供 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。

1.1.2 Visual Basic 的版本

Microsoft 公司为了简化 Windows 应用程序的开发过程,于 1991 年推出了 Visual Basic 1.0 版,并获得巨大成功。随着 Windows 操作系统版本的不断更新,Visual Basic 的版本也不断更新升级,到 1998 年推出 Visual Basic 6.0 版,已经经历了 6 个版本。Visual Basic 的最新版本是 2002 年发布的 Visual Studio.net 套件中的 Visual Basic.net,该版本网络功能更强,但由于 Visual Basic.net 对运行环境要求较高,目前使用还不够广泛。

目前拥有用户最多的 Visual Basic 6.0 有 3 种版本:学习版、专业版和企业版。3 种版本适合于不同的用户层次,大多数应用程序可在 3 种版本中通用。

- 学习版(Learning Edition)。学习版是 Visual Basic 的基础版本,可用来开发 Windows 应用程序。该版本包括了所有内部控件(标准控件)、网格(Grid)控件以及数据绑定控件。
- 专业版(Professional Edition)。专业版包括了学习版的全部功能,同时还包括 ActiveX 控件、Internet 控件和报表控件等。该版本为专业编程人员提供了一套功能完备的开发工具。
- 企业版(Enterprise Edition)。企业版可供专业编程人员使用,是功能强大的客户/服

务器或 Internet/Intranet 应用程序开发工具。它包括了专业版的全部功能,还增加了自动化管理器、部件管理器、数据库管理工具等。

本书以 Visual Basic 6.0 企业版作为学习环境,但书中程序仍然可在专业版中运行,大多数程序可在学习版中运行。为叙述方便,除特别声明外,在本书中 Visual Basic 6.0 简称 VB。

1.2 Visual Basic 的安装、启动和退出

1.2.1 Visual Basic 的安装

1. 系统要求

Visual Basic (以下简称 VB) 可以运行在 Windows 9x/Me/NT/2000/XP 环境下,安装时对软、硬件没有特殊要求。对环境的要求与 Windows 9x/Me/NT/2000/XP 环境基本相同。如果安装企业版,对硬盘的要求为 150MB 左右,除此之外,安装帮助系统 MSDN 需硬盘空间 70MB 左右。

2. 安装

VB 系统存放在一张安装光盘(CD)上。安装过程与其他 Microsoft 应用程序的安装过程类似,首先将 VB 安装盘放入光驱,然后在“我的电脑”或“资源管理器”中执行安装光盘上的 Setup 程序,启动安装过程,在安装程序的提示下进行安装。对于初学者可采用“典型安装”方式,但该方式不会将系统提供的图库(即界面设计时可能用到的一些图形文件)装入计算机。另外,VB 联机帮助文件使用 MSDN (Microsoft Developer Network Library) 文档的帮助方式,MSDN 与 VB 系统不在一张 CD 盘上,而与 Visual Studio 产品的帮助集合在另外两张 CD 盘上,在安装过程中,系统会提示插入 MSDN 盘。

1.2.2 Visual Basic 的启动和退出

1. VB 的启动

启动 VB 的步骤如下。

(1) 单击 Windows 任务栏中的“开始”按钮  ,在弹出的“开始”菜单中,从“程序”组中选择“Microsoft Visual Basic 6.0 中文版”下的“Microsoft Visual Basic 6.0 中文版”项,即可启动 VB。

(2) 启动 VB 后,首先显示“新建工程”对话框,如图 1-1 所示。

(3) 在对话框中直接单击“打开”按钮,进入 VB 的集成开发环境。

2. VB 的退出

如果要退出 VB,可单击 VB 窗口的“关闭”按钮,或者选择“文件”菜单中的“退出”命令,VB 会自动判断用户是否修改了当前工程的内容,并询问用户是否保存文件或直接退出。

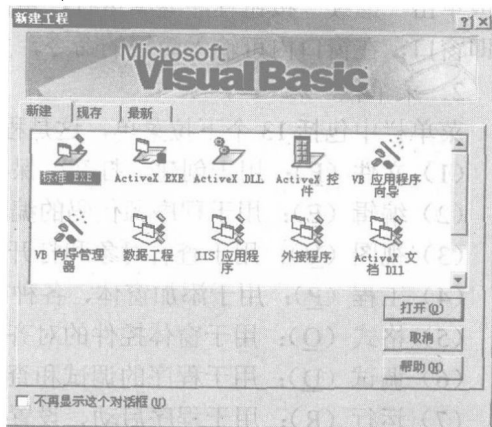


图 1-1 “新建工程”对话框

1.3 Visual Basic 的集成开发环境

VB 的集成开发环境除了 Microsoft 应用软件常规的标题栏、菜单栏、工具栏外，还包括 VB 的几个独立的窗口，如图 1-2 所示。VB 应用程序的开发过程几乎都可以在集成环境中完成。

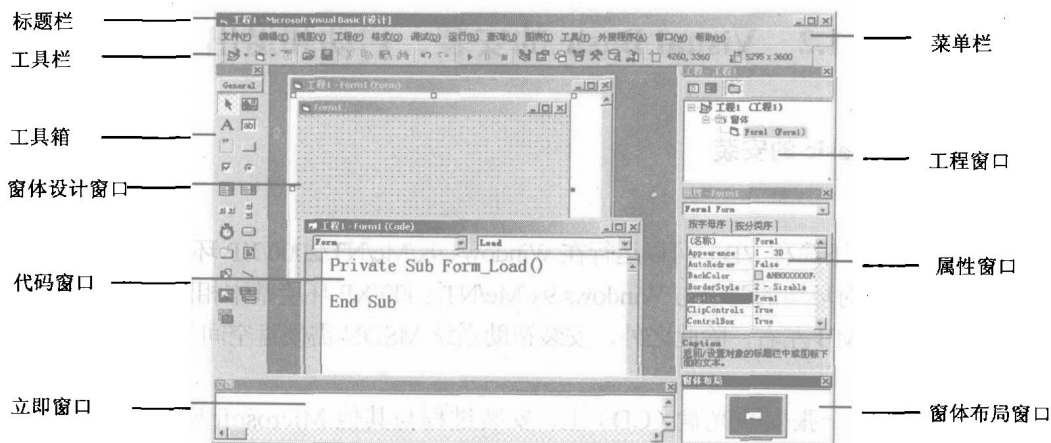


图 1-2 VB 应用程序集成开发环境

1.3.1 Visual Basic 的主窗口

1. 标题栏

标题栏中显示的内容包括窗体控制菜单图标、当前激活的工程名称、当前工作模式以及最小化、最大化/还原、关闭按钮。标题栏中的标题为“工程 1-Microsoft Visual Basic[设计]”，说明此时集成开发环境处于设计模式，在进入其他状态时，方括号中的文字会有相应的变化。VB 有如下 3 种工作模式：

- (1) 设计模式：可进行用户界面的设计和代码的编写，完成应用程序的开发。
- (2) 运行模式：运行应用程序，此时不可编辑代码，也不可编辑界面。
- (3) 中断模式：应用程序运行暂时中断，这时可以编辑代码，但不可编辑界面。按【F5】键或单击“继续”按钮程序继续运行；单击“结束”按钮停止程序运行。在此模式中会弹出立即窗口，在窗口内可输入简短的命令，并立即执行，以便检查程序运行状态。

2. 菜单栏

菜单栏中包括 13 个下拉菜单，这是程序开发过程中的常用命令。

- (1) 文件 (F)：用于创建、打开、保存工程以及生成可执行文件等。
- (2) 编辑 (E)：用于程序源代码的编辑。
- (3) 视图 (V)：用于查看对象和打开各种窗口。
- (4) 工程 (P)：用于添加窗体、各种模块和控件。
- (5) 格式 (O)：用于窗体控件的对齐格式化。
- (6) 调试 (D)：用于程序的调试和查错。
- (7) 运行 (R)：用于程序启动、设置中断和停止运行等。
- (8) 查询 (U)：VB6.0 新增，在设计数据库应用程序时用于设置 SQL 属性。

(9) 图表 (I): VB6.0 新增, 在设计数据库应用程序时用于编辑数据库。

(10) 工具 (T): 用于集成开发环境下工具的扩展。

(11) 外接程序 (A): 用于增加或删除外接程序。

(12) 窗口 (W): 用于窗体的层叠、平铺等布局, 以及窗体的切换。

(13) 帮助 (H): 用于在线帮助。

3. 工具栏及对象指示区

利用工具栏可快速访问常用的菜单命令。除了如图 1-3 所示的“标准”工具栏外, 还有“编辑”、“窗体编辑器”、“调试”等专用工具栏。要显示或隐藏工具栏, 可以选择“视图”菜单的“工具栏”命令或在“标准”工具栏处单击鼠标右键进行所需工具栏的选取。

工具栏的右端是窗体或控件指示区, 左边数字表示对象的坐标位置 (窗体工作区左上角为坐标原点), 右边数字表示对象的宽度和高度, 其默认单位是 twip (1 英寸=1440twip), 可以通过窗体的 ScaleMode 属性改变。

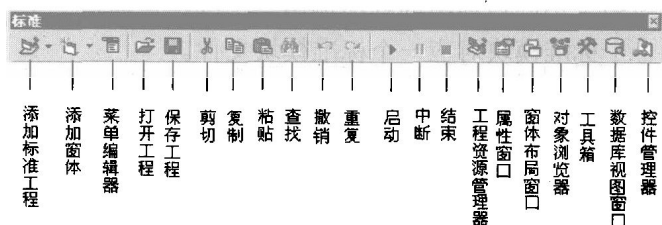


图 1-3 “标准”工具栏

1.3.2 Visual Basic 的其他窗口

除主窗口外, VB 的编程环境中还有一些其他窗口, 包括: 窗体设计窗口、属性窗口、工程资源管理器窗口、工具箱窗口、调色板窗口、代码窗口和立即窗口。

1. 窗体设计窗口

窗体设计窗口简称窗体 (Form), 就是应用程序最终面向用户的窗口。在应用程序运行时, 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。在设计状态, 窗体中布满了排列整齐的网格点, 如图 1-2 所示, 这些网格方便设计者对控件的定位。如果要清除网格点或者改变点与点之间的距离, 可通过执行“工具”菜单中的“选项”命令, 在其中的“通用”选项卡中调整。程序运行时窗体的网格不显示。

窗体的左上角显示的是窗体的标题, 右上角有 3 个按钮, 其作用与 Windows 下普通窗口中的作用相同。

窗体设计是应用程序设计的第一步。在设计应用程序时, 窗体就像一块画布, 程序员根据程序界面的要求, 从工具箱中选取需要的控件, 在窗体中画出来。一般地, 窗体中的控件可在窗体上随意移动、改变大小, 锁定后则不可随意修改。

2. 工程资源管理器窗口

工程是指一个应用程序的所有文件的集合。工程资源管理器窗口 (简称工程窗口) 采用 Windows 资源管理器式的界面, 层次分明地列出当前工程中的所有文件的清单, 一般包括窗体文件 (frm) 和标准模块文件 (bas) 等类型文件, 如图 1-4 所示。

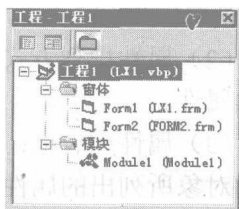


图 1-4 工程资源管理器窗口

另外，每个工程对应一个工程文件（.vbp）。

（1）工程文件。工程文件的扩展名为.vbp，工程文件用来保存与该工程有关的所有文件和对象的清单，这些文件和对象自动链接到工程文件上，每次保存工程时，其相关文件信息随之更新。在工程的所有对象和文件被汇集在一起并完成编码以后，就可以编译工程，生成可执行文件。

（2）窗体文件。窗体文件的扩展名为.frm，该文件存储窗体上使用的所有控件对象和有关属性、对象的事件过程和通用代码等信息。每个窗体对应一个窗体文件，一个应用程序至少有一个窗体，可以拥有多个窗体。执行“工程”菜单中的“添加窗体”命令或单击工具栏中的“添加窗体”按钮，可以增加一个窗体；而执行“工程”菜单中的“移除窗体”命令，可以删除当前的窗体。每建立一个窗体，工程窗口中就增加一个窗体文件，每个窗体都有一个不同的名字，可以通过属性窗口设计 Name 属性，窗体默认名称为 Form1、Form2、Form3 等。

（3）标准模块文件。标准模块文件的扩展名是.bas，它是为合理组织程序而设计的。标准模块是一个纯代码性质的文件，主要用来声明全局变量和定义一些通用的过程，它不属于任何一个窗体，可以被多个不同窗体中的程序调用。标准模块通过“工程”菜单中的“添加模块”命令来建立。一个标准模块对应一个标准模块文件。

在工程窗口的顶部还有 3 个按钮，分别是：“查看代码”、“查看对象”和“切换文件夹”。单击“查看代码”按钮可打开代码窗口，显示和编辑代码；单击“查看对象”按钮可打开窗体设计窗口，查看和设计当前窗体；单击“切换文件夹”按钮则可以隐藏或显示文件夹中的个别项目列表。



图 1-5 属性窗口

3. 属性窗口

在 VB 集成环境的默认视图中，属性窗口位于工程窗口的下面。按【F4】键，或者单击工具栏中的“属性窗口”按钮，或者选取“视图”菜单中的“属性窗口”子菜单，均可打开属性窗口，如图 1-5 所示。

在 VB 中，窗体和控件被称为对象。每个对象都可以用一组属性来描述其特征，如颜色、字体、大小等，属性窗口就是用来设置窗体和

窗体中控件的属性的。属性窗口中包含选定对象（窗体或控件）的属性列表，系统为每个属性预置了一个默认值，用户在程序设计时可通过修改对象的属性来改变其外观和相关特性，这些属性值将是程序运行时各对象的初始属性。

属性窗口由以下部分组成。

（1）对象列表框：修改对象的属性首先要选定对象，对象列表框中显示了当前窗体和其中所有对象的名称及所属的类。单击右端的下拉箭头，可打开列表框，从中可选择要更改其属性的对象。

（2）属性排列方式选项卡：可采用“按字母序”或“按分类序”两种方式来显示所选对象的属性。

（3）属性列表框：其中列出了所选对象在设计模式下可更改的属性及其默认值，对于不同的对象所列出的属性也不同。列表框左半边显示所选对象的所有属性名，右半边显示相应的属性值。用户可以选定某一属性，然后对该属性值进行设置和修改。在实际的应用程序设

计中，没有必要设置对象的所有属性，大多数属性可以使用默认值。

(4) 属性说明：显示当前属性的简要说明。可通过鼠标右键快捷菜单中的“描述”命令来切换显示或隐藏属性说明。

4. 代码窗口

代码窗口是专门用来进行程序代码设计的窗口，它可以显示和编辑程序代码，如图 1-6 所示。每个窗体都有各自的代码窗口。打开代码窗口有 4 种方法：双击窗体或控件、单击工程窗口中的“查看代码”按钮、选择“视图”菜单中的“代码窗口”命令、选择鼠标右键快捷菜单中的“查看代码”命令。

代码窗口主要包括如下内容。

(1) 对象下拉列表框。用来显示窗体及其所有对象的名称，供用户编写代码时选择操作对象，其中“通用”用来编写通用段代码，一般在此声明模块级变量或编写自定义过程。

(2) 过程下拉列表框。用来显示选定对象的所有事件名，供用户编写事件过程时选择触发事件。不同的对象会有不同的事件名。先在对象下拉列表框选择对象名，再在过程下拉列表框选择事件名，即可构成选中对象的事件过程模板，用户可在该模板内输入代码。

(3) 代码区。是编辑程序代码的地方，能够方便地进行代码编辑修改工作。

(4) 代码查看按钮。窗口的左下角有“过程查看”按钮和“全模块查看”按钮，“过程查看”只显示所选的一个过程，“全模块查看”显示模块中所有过程。

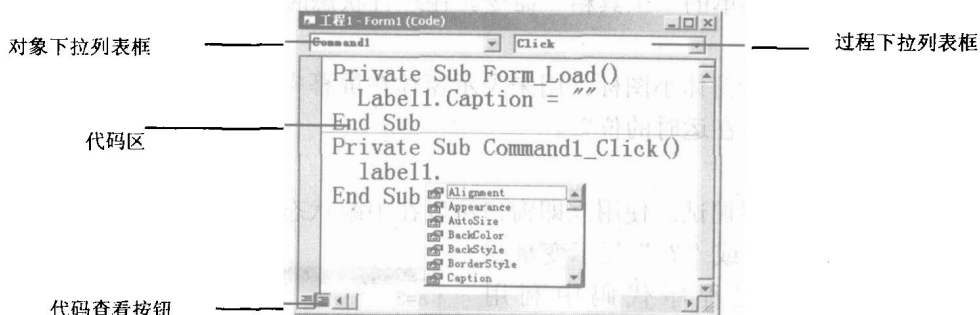


图 1-6 代码窗口

(5) 在输入和编辑代码时，VB 提供了自动列出成员特性和在线提示函数语法特性。当要输入控件的属性和方法时，在控件名后输入小数点，VB 会自动显示一个下拉列表框，其中包含了该控件的所有成员（属性和方法），如图 1-6 下方所示。依次输入成员的前几个字母，系统会自动检索并显示出需要的成员，从列表中选中成员并按【Tab】键即可完成输入。当不熟悉控件有哪些属性时，该项功能非常有用。

如果系统设置禁止“自动列出成员”特性，可使用快捷键【Ctrl+J】获得该特性。

5. 工具箱窗口

工具箱窗口如图 1-7 所示，它由工具图标组成，这些图标是 VB 应用程序的构件，称为图形对象或控件，每个控件由工具箱中的一个图标来表示。工具箱主要用于应用程序的界面设计。在设计阶段，首先用工具箱中的工具（控件）在窗体上建立用户界面，然后编写程序代码。界面的设计完全通过控件来实现。

VB 默认的工具箱中有 21 个图标，其中 20 个控件被称为标准控件。

注意：指针不是控件，它仅用于移动窗体和控件，以及调整它们的大小。用户也可通过“工

程”菜单的“部件”命令将 Windows 中注册过的其他控件（ActiveX 控件）装入到工具箱中。

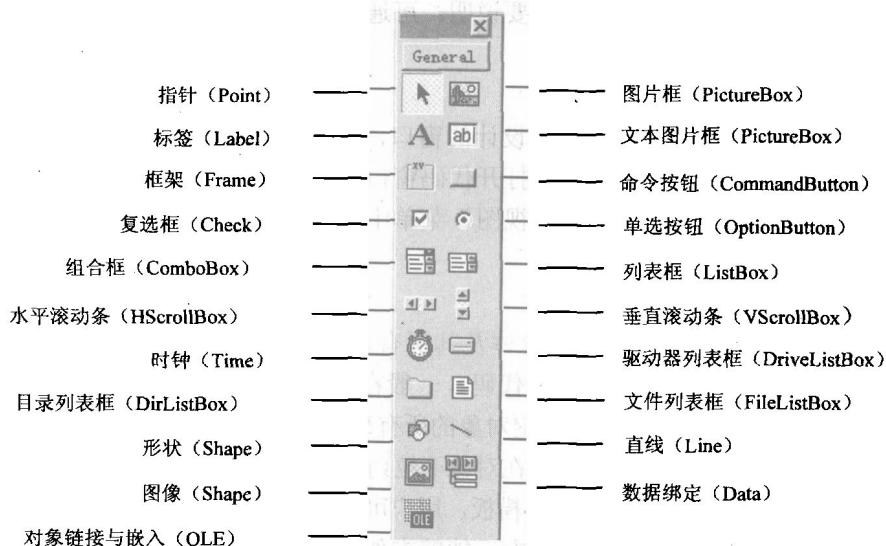


图 1-7 工具箱窗口

在设计状态时，工具箱通常是出现的，若不想显示工具箱，可以关闭工具箱窗口；若要再显示，可选择“视图”菜单的“工具箱”命令。在运行状态时，工具箱自动隐藏。

6. 窗体布局窗口

窗体布局窗口中有一个窗体小图标，用来表示窗体在屏幕中的位置。可以用鼠标拖动其中的窗体小图标来调整窗体在运时的位置。

7. 立即窗口

立即窗口主要用于程序调试。使用立即窗口可以在中断状态下查询对象的值，也可以直接在该窗口使用 Print 语句或“?”显示变量或表达式的值，还可以在程序代码中利用 Debug.Print 方法，把输出送到立即窗口。立即窗口可通过选择“视图”菜单的“立即窗口”命令打开，如图 1-8 所示，前 3 行是输入的命令，第 4 行是输出的结果。

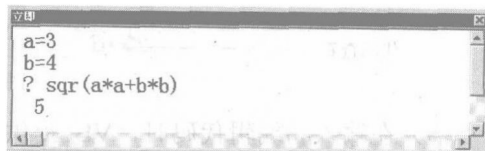


图 1-8 在立即窗口中输出表达式的值

1.4 可视化编程的一般步骤

VB 可视化编程不需要编写大量的代码去描述界面元素的外观和位置，而是采用面向对象、事件驱动的方法。VB 的对象已被抽象为窗体和控件，因而大大简化了程序设计。用 VB 开发应用程序，一般包括 3 个主要步骤：建立用户界面、设置窗体和控件的属性、编写代码。

(1) 建立用户界面。用户界面由窗体和控件组成，所有控件都放在窗体上，程序中的所有信息都要通过窗体显示出来，它是应用程序的最终用户界面。在应用程序中要用到哪些控件，就在窗体上建立相应的控件。程序运行后，将在屏幕上显示由窗体和控件组成的用户界

面。所以,要先建立窗体,然后在窗体上创建各种控件。

(2) 设置窗体和控件的属性。建立界面后,就可以设置窗体和每个控件的属性。在实际的应用程序设计中,建立界面和设置属性可以同时进行,即每画完一个控件,接着就可以设置该控件的属性。当然,也可以在所有对象建立完成后再回来设置每个对象的属性。

(3) 编写代码。由于 VB 采用事件驱动编程机制,因此大部分程序都是针对窗体中各个控件所能支持的方法或事件编写的,这样的程序称为事件过程。例如,命令按钮可以接收鼠标事件,如果单击该按钮,鼠标事件就调用相应的事件过程来做出相应的反应。

下面以图 1-9 所示的“改变字体”程序为例,叙述可视化编程的一般步骤。

(1) 新建一个工程。在 VB 中,开发的每个应用程序都被称为工程。新建一个工程有两种方法:

- 启动 VB 后,系统显示“新建工程”对话框,在“新建”选项卡中选择“标准 EXE”项,然后单击“打开”按钮。
- 选择“文件”菜单中的“新建工程”命令,在“新建工程”对话框中选择“标准 EXE”项,然后单击“确定”按钮。

采用上述任一种方法进入 VB 的集成开发环境,开始设计工程,即应用程序。系统默认的窗体只有一个 Form1。

(2) 向窗体添加控件。向窗体中添加控件的方法是:单击工具箱中的控件图标,移动鼠标到窗体,鼠标指针变成十字形状,此时按下鼠标左键并拖动,即可在窗体上画出对应控件。

在窗体 Form1 上绘出程序所需的控件,本例包括一个标签控件 Label1,两个命令按钮控件 Command1、Command2 (同类型控件的序号依次自动增加),如图 1-10 所示。



图 1-9 程序运行界面

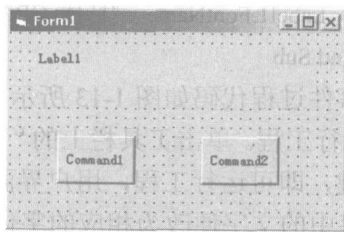


图 1-10 添加控件进行界面设计

(3) 设置窗体和控件的属性。对象属性的设置一般可在属性窗口中进行,其操作方法如下。

单击窗体的空白区域选中窗体,在属性窗口中找到标题属性 Caption,将其值改为“改变字体”,如图 1-11 所示。

单击窗体上的某控件,确认选中该控件,根据需要逐一设置控件的各属性。单击选中标签控件 Label1,将其 Caption 属性设为“欢迎使用 Visual Basic”;将其 AutoSize 属性改为“True”,使标签自动改变大小以适应文本的长短;在属性窗口找到并选中 Font “字体”属性,单击其右边的对话框按钮,在打开的“字体”对话框中设置字体大小。依次单击选中命令按钮 Command1 和 Command2,分别将它们标题属性 Caption 设为“黑体”和“楷体”。属性设置后的窗体如图 1-12 所示。

(4) 编写代码。

编写代码只能在代码窗口进行。用前面介绍的方法首先打开代码窗口,接着在窗口的对象下拉列表框中选中对象 Command1,再在过程下拉列表框中选择 Click (单击) 事件,此时

系统在代码区自动生成该事件过程的首行和尾行代码：

```
Private Sub Command1_Click()
```

```
End Sub
```



图 1-11 设置窗体 Form1 的属性



图 1-12 设置属性后的窗体

首尾两行代码程序员不必重复输入，只要在首、尾两行代码之间输入该事件过程必须实现的功能的代码：

```
Private Sub Command1_Click()
```

```
Label1.FontName = "黑体"
```

'将标签中字体改为黑体

```
EndSub
```

用同样的方法输入命令按钮 Command2 的单击事件过程代码：

```
Private Sub Command2_Click()
```

```
Label1.FontName = "楷体_GB2312"
```

```
End Sub
```

输入事件过程代码如图 1-13 所示。

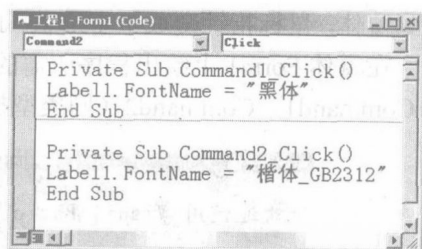


图 1-13 在代码窗口输入事件过程代码

(5) 运行工程。单击工具栏上的“启动”按钮  或按【F5】键，即可运行工程，用户界面如图 1-9 所示。单击界面中的“黑体”或“楷体”按钮时，标签中的文字便改为相应的字体。

单击窗体标右上角“关闭”按钮 ，便可关闭该窗口，结束运行，返回窗体设计窗口。

(6) 修改工程。修改工程包括修改对象的属性和代码，或者添加新的对象和代码，或者调整控件的大小等。直到满足工程设计的需要为止。

(7) 保存工程。在程序调试正确后需要保存工程，即以文件的方式保存到磁盘上。常用下面两种方法保存工程：

- 单击“文件”菜单中的“保存工程”或“工程另存为”命令，如图 1-14 所示。
- 单击工具栏上的保存工程按钮 .

如果新建工程从未保存过，系统将打开“文件另存为”对话框，如图 1-15 所示。由于一个工程可能含有多种文件，如工程文件和窗体文件等，这些文件集合在一起才能构成应用程序。因此，在“文件另存为”对话框中，需注意保存类型，并且将窗体文件（.frm）保存到指定文件夹中。窗体文件存盘后系统会弹出“工程另存为”对话框，保存类型为“工程文件（.vbp）”，默认工程文件名为“工程 1.vbp”，保存工程文件到指定文件夹中。建议将同一工程所有类型的文件存放在同一文件夹中。