

大专院校非计算机专业推荐用书

新 编

Visual Basic

程序设计教程

第2版

讲述Visual Basic基础知识

掌握Visual Basic数据存取特性

提供详实的Visual Basic编程实例

主 编 梁普选

副主编 张宗镛 张晓静

王春红 王建亮



电子工业出版社

Publishing House of Electronics Industry
<http://www.phei.com.cn>

新编 Visual Basic 程序设计教程

(第 2 版)

主 编 梁普选
副主编 张宗镛 张晓静
王春红 王建亮

電子工業出版社
Publishing House of Electronics Industry
北京 · Beijing

内 容 简 介

Visual Basic 是微软 (Microsoft) 公司开发的基于 Windows 95/98/NT/2000/XP 平台的 32 位程序设计开发平台, 其最大的优点是简单易学, 并且可以高效地开发出标准的 Windows 应用程序。本教材通过大量实例, 深入浅出地介绍了 Visual Basic 的基础知识, 循序渐进地介绍了程序设计基础及面向对象的程序设计方法。本书基本概念清晰, 通俗易懂, 例题丰富, 可作为非计算机专业学生的《高级语言程序设计》课程教材, 也可作为学习 Windows 编程的入门教材。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有, 侵权必究。

图书在版编目 (CIP) 数据

新编 Visual Basic 程序设计教程 / 梁普选编著. — 2 版. — 北京: 电子工业出版社, 2004.1
大专院校非计算机专业推荐用书
ISBN 7-5053-9435-5

I. 新... II. 梁... III. BASIC 语言 - 程序设计 - 高等学校 - 教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2003) 第 115854 号

责任编辑: 徐津平

印 刷: 北京大中印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

经 销: 各地新华书店

开 本: 787 × 1092 1/16 印张: 21.25 字数: 544 千字

印 次: 2004 年 1 月第 1 次印刷

定 价: 26.00 元

凡购买电子工业出版社的图书, 如有缺页问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系。
联系电话: (010) 68279077。质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权请发邮件至 dbqq@phei.com.cn。

序 言

国家教育部工科计算机基础课程教学指导委员会对非计算机专业的计算机基础教学提出了三个层次构架。第一层次为“计算机文化基础”,以了解计算机基础知识,掌握Windows平台下基本操作能力为教学目标。第二层次内容为“高级语言程序设计”,其教学目标为:掌握程序设计一般性规律,具有阅读、理解程序能力并能把程序设计方法应用到本专业与相关领域中。

选择何种语言作为“高级语言程序设计”课程教学内容,受到师资水平、设备和课时等条件制约。非计算机专业的计算机基础课课时有限,大部分学生入学前没有任何编程基础,而BASIC语言作为大众化编程工具,从20世纪80年代初的DOS下的BASIC到目前Windows平台下的大众化的开发工具——Visual Basic,经历了二十余年的教学实践,已深深扎根在计算机教学沃土之中,我们首选Visual Basic作为程序设计教学内容也正是基于以上原因。

为了讲解上的方便,我们在本教材中以目前使用最多的Visual Basic 6为例。对于最新版本的Visual Basic .NET来说,这里所讲的内容除很少的地方有些小差异之外,基本上也是相符的。所以,本教材完全可以作为Visual Basic 6和Visual Basic .NET的基础教程。

本教材编写人员分工如下:第1章“Visual Basic 基础知识”由王建亮编写;第2章“程序设计基础”、第3章“顺序结构程序设计”、第4章“选择结构程序设计”由张宗镛、王巧玲编写;第5章“循环结构程序设计”由梁普选编写;第6章“数组”由梁普选、齐林编写;第7章“过程”由梁普选、朱研编写;第8章“多窗体程序设计”、第9章“多文档程序设计”由杨国霞、程欣编写;第10章“文件”、第11章“绘图”由张晓静编写;第12章“访问数据库”由王春红编写。全书统稿由梁普选完成。

本教材作为河北科技大学现代教育技术中心“高级语言程序设计”讲义,曾在98、99级两届学生共79个教学班中使用,并在此基础上不断改进完善。在本书的编写过程中,得到了电子工业出版社、河北省计算机教育研究会、河北科技大学现代教育技术中心领导尚爱华的大力支持和帮助。计算中心教师在教学过程中提出许多有益的建议,并为此付出艰辛的劳动,编者在此一并表示感谢。限于时间和编者的学识水平,书中不妥之处在所难免,敬请读者指正。

编 者

2003年11月

目 录

第1章 Visual Basic 基础知识	1
1.1 Visual Basic 集成开发环境的使用	1
1.1.1 Visual Basic 的启动和退出	2
1.1.2 菜单栏和工具栏	3
1.1.3 工程窗口	4
1.1.4 窗体窗口	5
1.1.5 控件工具箱	5
1.1.6 属性窗口	7
1.1.7 代码窗口	8
1.2 对象及事件驱动的概念	8
1.2.1 对象	9
1.2.2 事件驱动	10
1.3 窗体	12
1.3.1 窗体属性	12
1.3.2 方法	14
1.3.3 事件	16
1.4 控件	17
1.4.1 控件通用属性	17
1.4.2 命令按钮	20
1.4.3 文本标签	21
1.5 实例介绍	22
1.5.1 文字显示	22
1.5.2 窗体	26
1.5.3 时钟	29
1.6 Visual Basic 程序设计的过程	30
1.6.1 总体规划	31
1.6.2 模块设计	31
1.6.3 算法设计	32
1.6.4 界面设计	33
1.6.5 编写代码	34
1.6.6 调试运行	34
1.6.7 生成单独的可执行程序	35
习题	35
第2章 程序设计基础	38
2.1 数据类型	38
2.1.1 数据的分类	38
2.1.2 标准数据类型	38

2.2 常量和变量	42
2.2.1 常量	42
2.2.2 变量	43
2.2.3 自定义数据类型及其变量	45
2.3 表达式	46
2.3.1 Visual Basic 中的运算符	46
2.3.2 表达式和表达式中运算的优先顺序	47
2.4 Visual Basic 的内部函数	48
2.4.1 数学函数	48
2.4.2 字符串函数	49
2.4.3 常用转换函数	49
2.4.4 日期/时间函数	49
2.4.5 随机数函数	50
2.4.6 格式输出函数	51
2.5 语句和程序	53
2.5.1 常用术语	53
2.5.2 书写程序的规则	54
2.5.3 命令格式的符号约定	54
习题	55
第3章 顺序结构程序设计	58
3.1 赋值语句	58
3.1.1 格式和说明	58
3.1.2 赋值号和逻辑等号	59
3.2 数据的输入	59
3.2.1 通用对话框介绍	59
3.2.2 输入数据——InputBox 输入框函数	60
3.2.3 显示信息——MsgBox 消息框函数	61
3.3 数据的输出	63
3.3.1 直接输出到窗体	63
3.3.2 输出到图片框	66
3.3.3 输出到标签	67
3.4 文本框的使用	69
3.4.1 文本框的常用属性	69
3.4.2 焦点	70
3.4.3 文本框的常用事件和方法	70
3.4.4 文本框的应用	71
3.5 计时器 (Timer) 控件	72
习题	74
第4章 选择结构程序设计	77
4.1 选择结构程序设计的概念	77

4.2 条件表达式	78
4.2.1 关系表达式	78
4.2.2 逻辑表达式	79
4.3 单行条件语句	80
4.4 块结构条件语句	81
4.5 条件语句的嵌套	82
4.5.1 条件语句嵌套实例	83
4.5.2 简化的 If 语句的嵌套格式	85
4.6 多分支条件选择语句	90
4.7 单选按钮、复选框和框架	94
4.7.1 框架 (Frame)	94
4.7.2 单选按钮 (OptionButton)	95
4.7.3 复选框 (CheckBox)	96
习题	101
第 5 章 循环结构程序设计	104
5.1 循环概念	104
5.2 While ...Wend 循环	105
5.2.1 语句格式	105
5.2.2 当型循环程序流程框图	106
5.3 Do...Loop 循环	107
5.3.1 先判断后执行循环体的 Do ...Loop 结构	107
5.3.2 先执行循环体后判断的 Do...Loop 结构	109
5.4 循环中途退出	111
5.5 For...Next 循环	112
5.6 循环嵌套与常用算法	115
5.6.1 循环嵌套	115
5.6.2 穷举	116
5.6.3 递推	117
5.6.4 迭代	119
5.7 对象集合	120
习题	121
第 6 章 数组	125
6.1 基本概念	125
6.2 声明数组	126
6.3 数组输入与输出	128
6.3.1 数组输入	128
6.3.2 数组输出	129
6.4 动态数组	130
6.4.1 声明动态数组	130
6.4.2 程序中指定动态数组尺寸	130

6.4.3 保留动态数组内容	131
6.4.4 删除动态数组	132
6.5 使用数组	132
6.5.1 一维数组应用	134
6.5.2 二维数组的应用	140
6.6 控件数组	143
6.6.1 创建控件数组	144
6.6.2 控件数组的使用	144
6.7 记录类型	146
6.7.1 声明记录类型	146
6.7.2 定义记录变量	147
6.7.3 使用记录变量	148
习题	148
第7章 过程	151
7.1 基本概念	151
7.2 创建过程	152
7.3 调用过程	153
7.4 参数传递	154
7.4.1 值传递	154
7.4.2 引用传递	155
7.5 变量与过程的作用域	157
7.5.1 变量的作用域	157
7.5.2 过程作用域	161
7.6 函数过程	163
7.7 递归	166
7.8 鼠标与键盘事件过程	169
7.8.1 鼠标事件过程	169
7.8.2 键盘事件过程	171
7.8.3 对象拖放	173
习题	175
第8章 多窗体程序设计	178
8.1 多重窗体	178
8.1.1 多重窗体的管理	178
8.1.2 设置启动窗体	179
8.1.3 窗体处理	180
8.2 通用对话框	187
8.2.1 通用对话框的常用属性	188
8.2.2 通用对话框的方法	188
8.2.3 “打开”对话框	188
8.2.4 “另存为”对话框	190
8.2.5 “颜色”对话框	192

8.2.6 “字体”对话框	193
8.2.7 “打印”对话框	194
8.2.8 “帮助”对话框	195
习题	196
第9章 多文档设计	197
9.1 菜单设计	197
9.1.1 下拉菜单	197
9.1.2 弹出式菜单	198
9.1.3 菜单编辑器	198
9.1.4 创建基本菜单控件	200
9.1.5 创建子菜单	201
9.1.6 分隔菜单项	201
9.2 工具栏与状态条	202
9.2.1 工具栏	202
9.2.2 状态条	206
9.3 多文档设计	211
9.3.1 界面样式	211
9.3.2 创建 MDI 应用程序的一般步骤	211
9.3.3 动态创建多文档	213
习题	217
第10章 文件	218
10.1 顺序文件	218
10.1.1 创建顺序文件	218
10.1.2 读顺序文件	221
10.1.3 与读文件有关的几个函数	223
10.1.4 顺序文件读写实例	224
10.2 随机存取文件	226
10.2.1 打开随机文件	226
10.2.2 写/读随机文件	226
10.2.3 关闭随机文件	227
10.3 二进制文件	231
10.3.1 打开二进制文件	231
10.3.2 读/写二进制文件	231
10.4 文件管理函数与语句	233
10.5 文件系统对象	236
10.5.1 文件系统的概念	236
10.5.2 管理驱动器	237
10.5.3 管理文件夹	238
10.5.4 管理文件	240
习题	243

第 11 章 绘图	244
11.1 Visual Basic 的坐标系统	244
11.1.1 默认规格坐标系	244
11.1.2 标准规格坐标系	245
11.1.3 自定义规格坐标系	245
11.2 图形控件	248
11.3 创建图形的方法	250
11.3.1 点与直线	250
11.3.2 矩形	256
11.3.3 圆、椭圆和弧	263
11.3.4 绘制图形文件	268
11.4 创建动画	269
11.4.1 使用图像控件数组实现动画	269
11.4.2 图形滚动	271
11.4.3 使用清除图形对象实现动画	272
习题	273
第 12 章 访问数据库	274
12.1 关系数据库	274
12.1.1 关系数据库的基本概念	275
12.1.2 关系数据库的设计规则	276
12.2 用可视化数据管理器建立和维护数据库	276
12.2.1 启动可视化数据管理器	276
12.2.2 建立数据库的表	277
12.2.3 建立主索引和关键字	279
12.2.4 创建数据窗体	279
12.2.5 数据库维护	280
12.3 访问数据库	280
12.3.1 Visual Basic 的数据访问接口	281
12.3.2 使用 ADO 数据控件访问数据库	282
12.4 使用 ADO 对象访问数据库	288
12.4.1 连接数据源	289
12.4.2 打开记录集	290
12.4.3 使用结果集	292
12.4.4 中断连接	300
12.5 结构化查询语言 (SQL)	303
12.5.1 结构化查询语言概述	303
12.5.2 SQL 语句	303
12.5.3 SQL 中的 SELECT 语句	305
12.5.4 使用 SQL	306
习题	316
附录 A Visual Basic .NET 介绍	318

第 1 章

Visual Basic 基础知识

本章包括:

- ◆ Visual Basic 集成开发环境
- ◆ Visual Basic 程序设计的一般过程
- ◆ 对象及事件驱动
- ◆ 窗体
- ◆ 控件
- ◆ 实例

Visual Basic 是微软 (Microsoft) 公司开发的基于 Windows 95/98/NT/2000/XP 平台的 32 位程序设计开发平台。它具有三个特点:

- ◆ 一是作为 BASIC 语言中的一种, Visual Basic 具有简单易学的特性, 同时新增了面向对象的功能, 符合当今程序设计的趋势。
- ◆ 二是具有功能强大的集成开发环境(Integrated Development Environment, 简称为 IDE), 在编制简单程序时, 只需通过简单的鼠标拖动, 再辅之以编写少量代码就可以快速开发出标准的 Windows 应用程序。
- ◆ 三是它将复杂的 Windows 运行机制完全隐藏起来, 使用容易理解和掌握的对象来表示, 不需要花费大量的时间来学习 Windows 下的编程知识。

由于 Visual Basic 具有以上几个显著的特点, 利用它编制程序时不用了解很多的计算机知识和 Windows 的运作机制, 非常适合作为学习程序设计的入门语言和快速开发应用程序的有力工具。

学好 Visual Basic 程序设计, 应从以下三个方面抓起:

- ◆ 一是精通语言本身的语法、语义及相关算法, 这是保障程序正确性的基础。
- ◆ 二是掌握集成开发环境的使用, 这可以保障应用程序的开发具有高效率。
- ◆ 三是熟悉常用对象的使用, 这样才能开发出功能强大的程序。

1.1 Visual Basic 集成开发环境的使用

Visual Basic 集成开发环境 (IDE) 是利用 Visual Basic 开发 Windows 应用程序的平台, 利用 Visual Basic 编制程序的过程全部在集成开发环境下完成。因此, 掌握集成开发环境的使用

方法是非常重要的,这可以使我们快速开发出规范和易于管理的应用程序。由于 Visual Basic 采用了对象的概念,并且大部分对象都以直观的形式放在集成开发环境中,这样在使用这些对象时,只需将这些对象从相应的位置找到,然后将其放在窗体设计器中或在工程中加以引用,即可使用这些功能强大的对象。

1.1.1 Visual Basic 的启动和退出

1.1.1.1 Visual Basic 的启动

按照以下步骤逐级打开菜单(在不同的系统中路径可能不同):

“开始”⇒“程序”⇒“Microsoft Visual Basic 中文版”

在此菜单中单击菜单项“Microsoft Visual Basic 中文版”,便会启动 Visual Basic 的集成开发环境。系统启动完成后首先显示一个工程对话框,让用户进行操作方式选择,如图 1-1 所示。

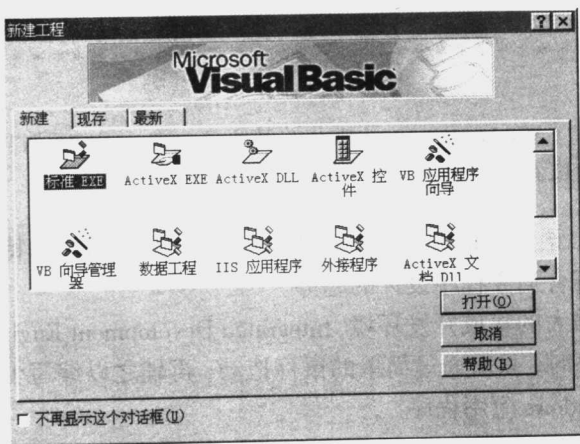


图 1-1 “新建工程”对话框

工程对话框包含三个选项卡,其含义分别如下:

- ◆ **新建:** 创建新的 Visual Basic 应用程序工程。选择该选项后,对话框中将列出若干种工程类型,从中选择一种工程类型,就可以使用最少的时间来完成该应用程序框架的构建。常用的有两个:一是“标准 EXE”,这种方式创建的工程只有一个空白窗体,没有添加任何代码;二是“VB 应用程序向导”,用户可以使用这种方式快速生成一个具有基本功能的应用程序,如多窗体应用程序等——使用这种方式创建的工程可能由多个窗体组成,并且窗体中添加了一些控件和相应代码。
- ◆ **现存:** 打开已经存在的工程,继续进行编辑、修改和调试。一个大的工程是需要经过多次断断续续的修改才能完成的。
- ◆ **最新:** 集成开发环境能够记住最近打开过的工程,这些工程按打开时间的顺序降序排列。在对话框的列表中显示最近一段时间内打开过的工程名称,时间越靠近现在,其排列位置越靠上,单击想要打开的工程名称,集成开发环境会自动打开该工程中的所有文件。

如果不希望 Visual Basic 每次启动时都出现该对话框, 可以选择该对话框下方的“不再显示这个对话框”复选框。在这种情况下, 集成开发环境每次启动时, 会自动创建一个类型为“标准 EXE”的工程。

Visual Basic 启动完成后, 屏幕显示如图 1-2 所示, 这就是我们要介绍的集成开发环境。

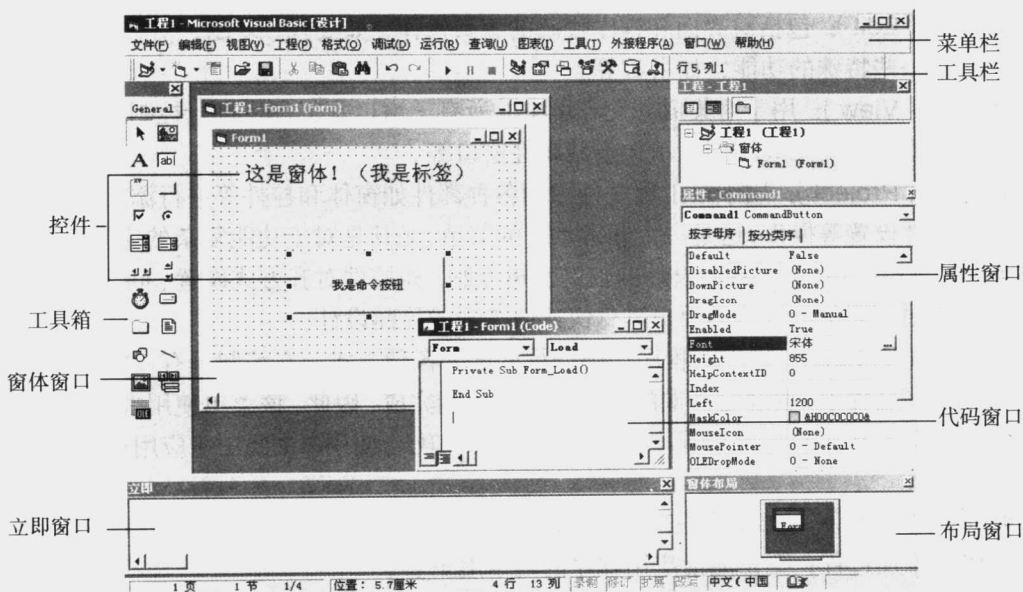


图 1-2 Visual Basic 集成开发环境

1.1.1.2 Visual Basic 的退出

在退出集成开发环境时, 系统会检查目前打开的工程是否做了修改。如果做了修改, 系统会提示保存最新的版本; 如果该工程从未保存过, 系统会显示“保存”对话框, 要求将组成工程的所有文件(工程文件、窗体文件和模块文件等)保存到磁盘上。建议将每个工程的所有文件保存到一个单独的文件夹下, 并且文件夹的名字用工程的名字来命名, 这样可以方便地对工程进行管理。

1.1.2 菜单栏和工具栏

Visual Basic 集成开发环境的菜单栏和工具栏如图 1-3 所示。

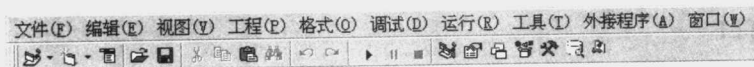


图 1-3 Visual Basic 集成开发环境的菜单栏和工具栏

1.1.2.1 菜单栏

菜单系统是标准 Windows 应用程序的控制中心, Visual Basic 集成开发环境也不例外, 大部分操作都可以通过菜单系统进行。Visual Basic 集成开发环境的顶层菜单如图 1-3 的上半部

分所示,下面简要介绍一下常用的菜单:

- ◆ **文件 (File)**: 主要由三部分内容组成,一是工程的创建、打开、关闭、另存为等功能;二是打印设置,如窗体外观打印、源代码的打印、打印机设置等;三是最近打开的工程列表,列表中的工程按使用时间由近及远排列。
- ◆ **编辑 (Edit)**: 包括对界面和程序代码的编辑操作,如查找、复制、粘贴等。此外,还包括一些特殊的功能,如书签等。
- ◆ **视图 (View)**: 用于对集成开发环境中的所有子窗口及工具栏进行控制和管理,如显示/隐藏控件工具箱、显示/隐藏属性窗口等。
- ◆ **工程 (Project)**: 用于对工程文件中的各种部件如窗体和控件等进行添加、删除和工程属性设置等操作,这是一个非常重要的菜单,尤其是对于功能复杂的应用程序来说。
- ◆ **格式 (Format)**: 对界面设计的辅助编辑控制,如控件对齐方式设置、间距设置等,熟练使用该菜单可以帮助用户进行快速而美观的界面设计。
- ◆ **调试 (Debug)**: 提供对程序代码进行调试的各种方式,在编制具有一定复杂度的应用程序过程中,要经过多次的编辑和调试才能完成,因此,该菜单使用也是很频繁的。
- ◆ **运行 (Run)**: 执行应用程序,与“调试”菜单配合使用,主要用于应用程序的功能检验。
- ◆ **工具 (Tool)**: 主要包括三部分功能,一是对集成开发环境进行定制;二是向程序代码中添加过程;三是激活应用程序的菜单编辑器。
- ◆ **外接程序 (Add-Ins)**: 主要包含两部分功能,一是 Visual Basic 环境下的数据库管理器;二是外部程序管理器,这些外部程序对 Visual Basic 编程有一定的辅助作用,如 Windows API (应用程序编程接口) 的原型说明。

1.1.2.2 工具栏

为了便于用户使用一些最常用的功能,在集成开发环境中,这些最常用的功能全部汇集在标准工具栏中,其使用方法和 Office 系列软件的工具栏相似。标准工具栏外观如图 1-3 的下半部分所示,从左至右依次是:创建工程、添加窗体、菜单编辑器、打开工程、保存工程、剪切、复制、粘贴、查找、撤消、重复、启动、中断、结束、工程资源管理器、属性窗口、窗体布局窗口、对象浏览器、工具箱、数据视图窗口、可视化部件管理器,最右侧显示的动态信息是控件在窗体中的位置和大小。

除上述的标准工具栏外,用户还可随时定制工具栏的组成,操作方法是:将鼠标指针指向工具栏,然后单击右键,此时会弹出菜单,从中可以选择显示或隐藏某个工具栏。

1.1.3 工程窗口

工程窗口是集成开发环境中非常重要的一个组成部分,其主要功能是管理工程及组成工程的相应元素,主要功能包括:添加、删除工程中的组成元素,在代码编辑和界面设计状态之间进行切换,设置工程属性等。该窗口如图 1-4 所示(具体显示内容和实际程序有关)。

工程是 Visual Basic 中一个非常重要的概念,可以认为工程是创建一个应用程序时所包含文件的集合,包括窗体文件、模块文件以及可以由 Visual Basic 应用程序使用的任何类型的文件。工程管理器使用类似于资源管理器的树形文件结构图来显示工程中的文件,通过该窗口

可以快速地切换所编辑的文件,可以快捷地增加和删除任何类型的文件。如果希望给窗体设计窗口或代码设计窗口留出更大的空间,可以将工程资源管理器窗口关闭。

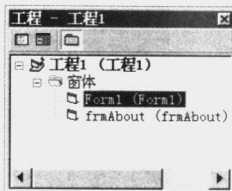


图 1-4 工程资源管理器窗口

工程资源管理器中可以同时容纳多个工程,但在任一时刻只能有一个工程是可运行的,称该工程为可启动的。设置某个工程为可启动工程的步骤是:鼠标指针移到工程名称上,单击右键,选择设置为启动。在工程中添加、删除和保存部件的方法如下:

1. **添加部件:** 鼠标指针指向工程名称,然后单击右键,在快捷菜单中定位到“添加”选项,显示出二级菜单,然后选择所要添加的部件类型即可。
2. **删除部件:** 首先展开要删除的部件所在的工程,然后在要删除的部件项上单击右键,选择“删除”即可。
3. **保存部件:** 当某些文件做过修改后,可以将其及时保存或另外保存为其他文件。选中要保存的部件项,单击右键,选择“保存”或“另存为”即可。

1.1.4 窗体窗口

设计 Visual Basic 应用程序的过程中,最基本的工作是界面设计。界面设计主要在一个窗口中进行,该窗口被称为窗体窗口,在该窗口中可以像搭积木一样将控件放置到用户窗体表面。在设计过程中,需要配合格式菜单进行大小、位置及对齐方式的调整。每个窗体窗口只能容纳一个窗体,应用程序中的窗体可能有多个,相应地窗体窗口也就有多个。对窗体窗口的操作和普通的窗口操作完全相同。对窗体窗口的操作细节在后面还要详细介绍。

1.1.5 控件工具箱

在 Windows 应用程序中,应用程序的基本组成元素是对象。对象分为两种:一种是可见的,通过外观可以了解其位置、颜色等外部特征,当然它还可能完成某些功能,如命令按钮;另一种是不可见的,但它能执行某种功能,如进行数据库操作时使用的 ADO 对象。在 Visual Basic 的集成开发环境中,这两种对象的使用方法是不同的,不可见对象的使用方法是:打开“工程”菜单,选择“引用”选项,然后会显示一个对话框,从中选择要使用的对象,然后单击“确定”按钮。在这里要详细介绍的是可见对象的使用方法。

在窗口表面上有很多种规范的部件,如文本框、下拉列表、文本标签等,这些部件实际上都是可见的对象,在 Visual Basic 中将其称为控件。在进行窗体的界面设计时,最主要的工作就是把合适的控件按照需要放到合适的位置并设置合适的属性,为了便于用户使用这些控件,在集成开发环境中有一个窗口,专门容纳这些控件,称主控件工具箱,如图 1-5 所示。

将控件放置到窗体表面是一个很简单过程,有两种操作方法:一是先单击控件工具箱

上的某个控件，然后使用鼠标拖动的方式将该控件在窗体窗口中画出来；二是双击控件工具箱上的某个控件，该控件对象就会自动出现在窗体中央，其大小是默认的。



图 1-5 标准控件工具箱

应用程序运行过程中，大部分控件都是可见的，并且这些控件的大小、位置及相关属性都可以在设计过程中改变。但有些控件在界面设计时是可见的，而在程序运行时是不可见的，如 Timer 控件，这些控件在程序运行时起作用，但不会显示在窗口表面。因此，在进行界面设计时，这些控件的大小和位置是不重要的。

在编制较复杂的应用程序时，用户可能会发现它们需要的某些控件并没有包含在控件工具箱中，这时就需要手工地将控件添加到控件工具箱中，然后再使用。向工具箱添加控件的过程如下：

1. 将鼠标指向控件窗口的任意位置，单击右键，显示快捷菜单，然后单击“部件”选项。此时，屏幕显示出如图 1-6 所示的“部件”对话框，在列表框中显示出可以使用的控件列表。

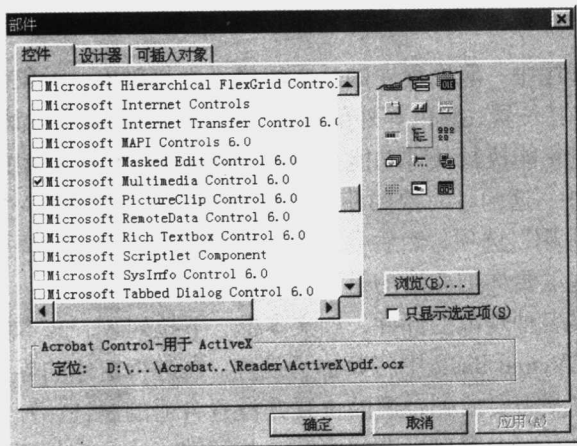


图 1-6 工程部件对话框

2. 选中需要添加到工具箱的控件，全部完成后，单击“确定”按钮，选定的控件就会出现在工具箱中。

1.1.6 属性窗口

在 Visual Basic 中，用于描述对象（如文本框、窗体、命令按钮等）的外观、标题、名称、位置以及其他特性的值称为属性。在进行界面设计时，一个很重要的工作是设置对象的属性（当然，某些属性也可能在程序代码中修改）。在窗体设计阶段通过控件工具箱加入到窗体表面的控件的属性设置可通过属性窗口来进行。如果集成开发环境中没有显示属性窗口，可通过如下步骤将属性窗口显示出来：打开“视图”菜单，选择“属性”。属性窗口的外观如图 1-7 所示。在上一小节中还提到，有些对象必须通过在工程中引用的方式来使用，这种对象的属性设置不能通过属性窗口来完成，必须在程序代码中进行属性的设置。



图 1-7 属性窗口

属性窗口由三部分组成：最上面是操作对象，即正在给哪一个对象设置属性；中间由一个两列的表格组成，左列是属性名称，右列是相应属性的设置值，称为属性值；最下方是目前正在设置属性的简要介绍。

需要注意的是：在 Visual Basic 中，对象的属性分为两种，一种是在设计阶段可用的，还有一种是只能在运行阶段才可用的，属性窗口列出的属性是在设计阶段可用的属性。

窗体上有很多控件，如何设置指定控件的属性呢？操作很简单：鼠标指针指向某个控件，单击右键，然后在快捷菜单中选择“属性”，则该控件的设计阶段可用的属性会显示在属性窗口中。

属性窗口中的每个属性均有一个默认值，要改变该值，有三种方法：

- ◆ **直接输入设置的值：**用鼠标左键单击左列的属性名称，在右列中输入其值，如窗口标题 Caption。
- ◆ **通过下拉列表框选择值：**单击属性名称，如果右列出现向下的三角形，表示不用手工输入属性值，只须单击该三角形，显示下拉列表框，在列表给出的几个值中选择其一即可。如窗口状态 WindowState。
- ◆ **通过对话框设置：**单击属性名称后，如果右边出现带省略号的按钮，表示该属性值在对话框内进行设置，如字体 Font。