

高等院校非计算机专业教学用书

Visual Basic

大学基础教程

麻新旗 梁普选 主编



电子工业出版社

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

高等院校非计算机专业教学用书

Visual Basic

大学基础教程

麻新旗 梁普选 主编

电子工业出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书根据国家教育部对非计算机专业计算机基础教学提出的要求,选择 Visual Basic 6.0 作为“高级语言程序设计”课程的教学内容,通过大量实例,深入浅出地介绍 Visual Basic 6.0 基础知识、程序设计基础知识、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、多重窗体与环境应用、菜单和对话框、数据文件、图形和图像、访问数据库等设计技术。本书在内容安排上紧扣教学要求,充分考虑到目前大学生的实际水平和实际需要,讲解力求基本概念清晰、通俗易懂、例题丰富。本书适合作为大学本科各专业学生“高级语言程序设计”课程的教材,也可作为学习 Windows 编程、Visual Basic 编程的入门教材。

为了便于教师授课和学生复习,本书配套有“电子教案”、“习题答案”、“实例源代码”和“附加练习题及答案”等内容,这些内容可以从华信卓越网站(www.hxex.cn)的“资源下载”栏目查找并免费下载。

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

Visual Basic 大学基础教程 / 麻新旗, 梁普选主编. —北京: 电子工业出版社, 2005.12
ISBN 7-121-02166-8

I.V... II.①麻...②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 (2005) 第 155591 号

责任编辑: 郝志恒

排版制作: 华信卓越公司制作部

印 刷: 北京市天竺颖华印刷厂

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: 22.25 字数: 570 千字

印 次: 2005 年 12 月第 1 次印刷

定 价: 29.00 元

凡购买电子工业出版社的图书,如有缺损问题,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。质量投诉请发邮件到 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书编委会

主 编	麻新旗	梁普选		
副主编	张宗镛	杨秦建	张晓静	
	王春红	王建亮		
审 校	张冬雯	杨秦建	张宗镛	
	张晓静	梁普选		
编 委	(排名不分先后)			
	王建亮	刘紫玉	杨秦建	王巧玲
	刘 伟	张宗镛	武春英	朱 研
	程 欣	杨国霞	张新增	王春红
	张晓静	麻新旗	张冬雯	梁普选

前 言

“高级语言程序设计”是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的第二层次内容的教学要求(第一层次为“计算机文化基础”,使学生了解计算机基础知识,掌握 Windows 平台下的基本操作能力;第三层次为“计算机应用基础”,主要任务是进一步培养学生利用计算机获取信息、处理信息和解决问题的意识与能力,增强学生建构专业及相关领域中计算机应用系统的能力,内容包括计算机信息管理与多媒体应用基础等),目标是使学生掌握程序设计的一般性规律,具有阅读程序、理解程序的能力,并能把程序设计方法应用到本专业与相关领域中,培养学生的程序设计能力。

现实情况是,多数学生在入学前没有计算机编程基础;学校师资条件有限,设备条件有限,大学本科计算机基础课时有限。针对这种情况,选择何种语言作为“高级语言程序设计”课程教学内容,显得尤为重要。BASIC语言作为大众化编程工具,从20世纪80年代初的BASIC语言到 Windows 平台下的开发工具——Visual Basic 集成开发环境(IDE),已经成为最普及的 Windows 应用程序开发环境。Visual Basic 6.0 是基于 Windows 的 32 位程序设计开发平台,它继承了以前 Visual Basic 集成开发环境的优点,简单易学,可以高效地开发出标准的 Windows 应用程序。经历了二十余年的计算机基础课程的教学实践就印证了这一点,我们首选 Visual Basic 6.0 作为程序设计的教学内容也正是基于上述原因。

本书包含 12 个章节:第 1 章由王建亮编写;第 2 章由刘紫玉编写;第 3 章由杨秦建编写;第 4 章由王巧玲编写;第 5 章由刘伟编写;第 6 章由张宗镛编写;第 7 章由武春英编写;第 8 章由朱研编写;第 9 章由程欣编写;第 10 章由杨国霞编写;第 11 章由张晓静编写;第 12 章由王春红编写。全书由麻新旗、梁普选统稿。

为了便于教师授课和学生复习,本书配套有“电子教案”、“习题答案”、“实例源代码”和“附加练习题及答案”等内容,这些内容可以从华信卓越网站(www.hxex.cn)的“资源下载”栏目查找并免费下载。

本书作为大学本科计算机基础课教材,在部分院校使用过程中,任课教师提出了许多宝贵的修改意见和建议,作者在此一并表示感谢。限于作者的学识水平,书中不妥之处难免,敬请读者指正。

编 者

2005 年 11 月

目 录

第1章 Visual Basic入门	1
1.1 集成开发环境	1
1.1.1 启动和退出 Visual Basic	1
1.1.2 菜单栏和工具栏	3
1.1.3 工程资源管理器窗口	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	9
1.1.8 立即窗口	10
1.2 对象及其操作	11
1.2.1 对象	11
1.2.2 窗体	14
1.2.3 控件	17
1.2.4 命令按钮	19
1.2.5 标签	21
1.3 Visual Basic 程序设计工作过程	22
1.3.1 第一阶段——界面设计	22
1.3.2 第二阶段——代码设计	22
1.3.3 第三阶段——运行调试	23
1.3.4 第四阶段——发布应用程序	24
1.4 实例介绍	25
习题	28
第2章 程序设计入门	30
2.1 数据类型	30
2.1.1 数据的分类	30
2.1.2 标准数据类型	30
2.1.3 构造类型	34
2.2 常量和变量	34
2.2.1 常量	34
2.2.2 变量	36
2.3 表达式	39
2.3.1 运算符	40
2.3.2 表达式和运算优先顺序	41
2.4 内部函数	45
2.4.1 数学函数	45
2.4.2 字符串函数	46
2.4.3 常用转换函数	46
2.4.4 日期/时间函数	47
2.4.5 随机数函数	47
2.4.6 格式输出函数	49
2.5 语句和程序	51
2.5.1 常用术语	51
2.5.2 书写程序的规则	51
2.5.3 命令格式的符号约定	52
习题	53
第3章 顺序结构程序设计	56
3.1 基本语句	56
3.1.1 赋值语句	56
3.1.2 赋值号与逻辑表达式	57
3.1.3 注释、暂停与程序结束语句	57
3.2 数据的输入和输出	58

3.2.1	通用对话框的使用	59
3.2.2	直接在窗体上输出	63
3.2.3	输出到立即窗口	69
3.2.4	输出到图片框	70
3.2.5	利用标签输出	71
3.3	文本框的使用	72
3.3.1	常用属性	72
3.3.2	常用事件	73
3.3.3	焦点和 Tab 顺序	74
3.3.4	文本框的应用实例	74
3.4	计时器控件	77
3.4.1	常用属性	77
3.4.2	惟一的 Timer 事件	77
3.4.3	应用实例	77
	习题	79
第 4 章	选择结构程序设计	83
4.1	条件表达式	83
4.1.1	关系表达式	83
4.1.2	逻辑表达式	84
4.2	单行条件语句和 If 函数	85
4.2.1	单行条件语句	85
4.2.2	If 函数	86
4.3	块结构条件语句	87
4.4	条件语句的嵌套	88
4.4.1	条件语句嵌套实例	88
4.4.2	简化的 If 语句嵌套格式——多分支结构	90
4.5	多分支语句结构	92
4.6	单选按钮、复选框和框架	94
4.6.1	框架	94
4.6.2	单选按钮	95
4.6.3	复选框	99
	习题	104
第 5 章	循环结构程序设计	109
5.1	循环结构	109
5.2	GoTo 型控制	110
5.3	Do...Loop 型循环	111
5.3.1	先判断后执行循环体的 Do...Loop 循环	111
5.3.2	先执行循环体后判断的 Do...Loop 结构	113
5.4	For...Next 循环	115
5.5	循环中途退出	118
5.6	循环嵌套与常用算法	120
5.6.1	循环嵌套	120
5.6.2	穷举	122
5.6.3	递推	123
5.6.4	迭代	125
5.7	对象集合 For Each...Next 语句	126
5.8	列表框和组合框	127
5.8.1	列表框	127
5.8.2	组合框	130
	习题	132
第 6 章	数组	139
6.1	数组的概念	139
6.2	静态数组和动态数组	140
6.2.1	静态数组	140
6.2.2	动态数组	142
6.3	数组的基本操作和实例	143
6.3.1	数组的基本操作	143

6.3.2 应用实例	145
6.4 控件数组	157
6.4.1 控件数组的概念	157
6.4.2 创建控件数组	158
6.4.3 控件数组的应用实例	160
6.5 记录类型和枚举类型	161
6.5.1 记录类型的定义和使用	161
6.5.2 枚举类型的定义和使用	163
习题	165
第7章 过程	171
7.1 基本概念	171
7.2 Sub 过程	171
7.2.1 创建通用的 Sub 过程	171
7.2.2 调用通用过程	173
7.3 参数传递	175
7.3.1 值传递	175
7.3.2 引用传递	176
7.3.3 数组参数的传递	177
7.4 可变参数与可选参数	179
7.4.1 可选参数	179
7.4.2 可变参数	180
7.4.3 对象参数	181
7.5 函数过程	182
7.6 变量与过程的作用域	186
7.6.1 变量作用域	186
7.6.2 过程作用域	190
7.7 递归调用	192
7.8 鼠标与键盘事件过程	196
7.8.1 鼠标事件过程	196
7.8.2 键盘事件过程	198
7.8.3 对象拖放	200
习题	201
第8章 菜单和对话框	205
8.1 菜单	205
8.1.1 用菜单编辑器建立菜单	205
8.1.2 运行时创建和修改菜单	210
8.1.3 弹出式菜单的建立	212
8.1.4 菜单设计举例	214
8.2 对话框	217
8.2.1 通用对话框	217
8.2.2 自定义对话框	224
习题	226
第9章 多重窗体与环境应用	228
9.1 多重窗体	228
9.1.1 多重窗体管理	228
9.1.2 设置启动窗体	229
9.1.3 窗体处理	230
9.1.4 多重窗体应用实例	231
9.2 Visual Basic 工程结构	235
9.2.1 标准模块	235
9.2.2 窗体模块	236
9.2.3 Sub Main 过程	237
9.2.4 闲置循环与 DoEvents 语句	241
习题	243
第10章 数据文件	246
10.1 文件的结构与分类	246

10.1.1	文件的结构	246
10.1.2	文件的种类	247
10.2	文件操作语句和函数	248
10.2.1	文件的打开	248
10.2.2	文件的关闭	250
10.2.3	与文件操作有关的语句和函数	250
10.3	顺序文件	253
10.3.1	顺序文件的写操作	253
10.3.2	顺序文件的读操作	256
10.4	随机文件	258
10.4.1	随机文件的打开与读写操作	258
10.4.2	随机文件中记录的添加与删除	261
10.5	文件系统控件	263
10.6	文件基本操作	268
10.7	文件系统对象	271
10.7.1	文件系统的概念	271
10.7.2	管理驱动器	272
10.7.3	管理文件夹	273
10.7.4	管理文件	275
习题		277
第 11 章	图形和图像	280
11.1	图形	280
11.1.1	图形控件和属性	280
11.1.2	Visual Basic 坐标系统	283
11.2	创建图形方法	286
11.2.1	点与直线	286
11.2.2	矩形	291
11.2.3	圆、椭圆和弧	298
11.3	显示图片	303
11.3.1	直接将图片加载到窗体	303
11.3.2	使用图像控件	303
11.3.3	使用图片框控件	304
11.4	滚动条控件	305
11.4.1	滚动条常用属性	305
11.4.2	滚动条的事件	306
11.4.3	滚动条的应用	306
习题		307
第 12 章	访问数据库	309
12.1	数据库	309
12.1.1	关系数据库的基本概念	309
12.1.2	关系数据库的分类	310
12.1.3	关系数据库的设计规则	311
12.2	数据库的创建与维护	311
12.3	访问数据库	315
12.3.1	Visual Basic 的数据访问接口	316
12.3.2	使用 ADO 数据控件访问数据库	317
12.4	使用 ADO 对象访问数据库	324
12.5	Recordset 对象的游标类型	335
12.5.1	选择游标的位置	336
12.5.2	Recordset 对象的锁定类型	336
12.5.3	Recordset 对象的更新模式	337
12.6	结构化查询语言 SQL	341
12.6.1	结构化查询语言概述	341
12.6.2	SQL 语句	342
12.6.3	SQL 中的 Select 语句	344
12.6.4	使用 SQL	344
12.7	小结	347
习题		348

第1章 Visual Basic 入门

Visual Basic 是微软 (Microsoft) 公司开发的高级程序设计语言, 主要用于 Windows 操作系统下的软件开发, 目前普遍使用的版本之一是 6.0。它具有以下特点:

- ◆ **简单易学**——这是 Visual Basic 继承自 BASIC 语言的特点, 不需要太多的计算机知识就可以开发出功能强大的软件, 非常适合于非计算机专业的人员使用。此外, Visual Basic 还支持面向对象技术 (Object Oriented, 简称为 OO), 这是目前软件开发采用的主要技术。因此, Visual Basic 也适合专业软件开发人员使用。
- ◆ **提供功能强大的集成开发环境**——在 Visual Basic 的集成开发环境 (Integrated Development Environment, 简称为 IDE) 中, 可以方便、快捷地进行程序的创建、编辑、编译连接、调试、制作安装软件包等。为了便于编程人员查找编程信息, 该环境还提供了方便的联机帮助 (联机帮助需要单独安装)。
- ◆ **完全支持 Windows 应用程序的开发**——Windows 是一个复杂的操作系统, 编写基于这个平台的程序需要了解 Windows 复杂的运行机制。在 Visual Basic 出现之前, 编制 Windows 应用程序需要非常多的专业知识; 而 Visual Basic 将 Windows 复杂的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, 编程人员不需要了解 Windows 复杂的运行机制和编程知识, 可以很容易、快速地开发出标准的 Windows 应用程序。

要学好 Visual Basic 程序设计, 应了解三个方面的知识: 一是掌握语言本身的语法、语义及常用算法, 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, 这可以保障程序开发的高效率; 三是熟悉常用对象的使用, 这可以保障程序具有强大的功能。

1.1 集成开发环境

使用 Visual Basic 编制程序的工作是在集成开发环境下完成的, 利用集成开发环境提供的各种工具, 可以快速开发出规范的、易于管理的应用程序, 因此, 必须熟练掌握集成开发环境的使用。集成开发环境提供的工具很多, 这些工具是开发程序时必不可少的。本节主要介绍集成开发环境下各种工具的用途和使用方法。

1.1.1 启动和退出 Visual Basic

1. Visual Basic 的启动

启动 Visual Basic 的操作步骤如下 (在不同的系统中路径可能稍有不同):

- (1) 在 Windows 操作系统的桌面上, 单击“开始”按钮。

- (2) 指向“程序”选项。
- (3) 指向“Microsoft Visual Basic 中文版”选项。
- (4) 单击“Microsoft Visual Basic 中文版”选项，便会启动 Visual Basic 的集成开发环境，启动完成后会显示一个工程对话框，让用户进行操作方式选择。

工程是 Visual Basic 中一个非常重要的概念（详细内容请参考 1.1.3 节），每个应用程序对应一个工程，每个工程有一个工程文件，用于组织和管理与工程有关的文件。图 1-1 是新建工程的界面，此外，该对话框还提供另外两个选项卡，可以按照不同方式打开已经存在的工程。

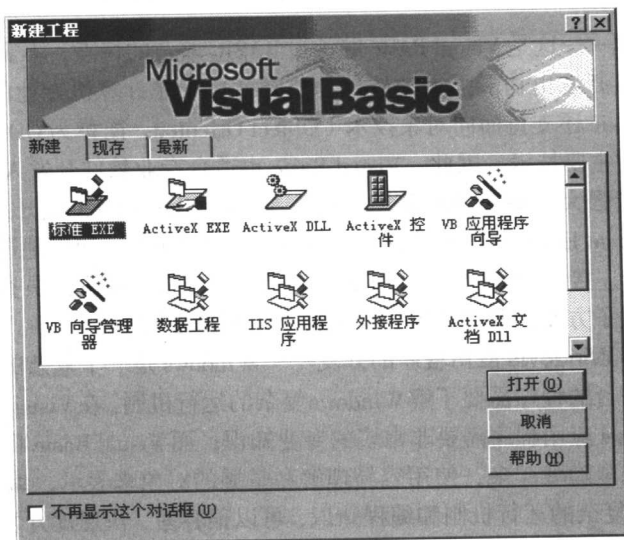


图 1-1 新建工程

Visual Basic 集成开发环境启动完成后，屏幕显示如图 1-2 所示。

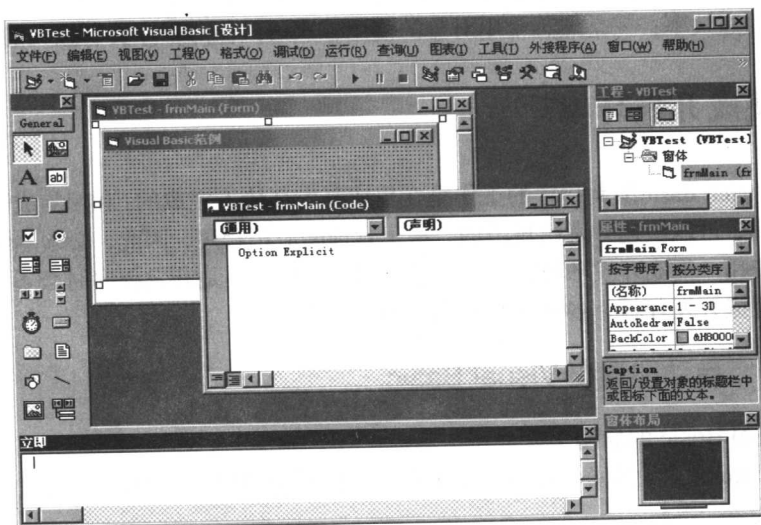


图 1-2 Visual Basic 集成开发环境

2. Visual Basic 的退出

Visual Basic 退出时就会关闭集成开发环境,退出前系统会检查目前打开的工程是否已经保存过。如果本次操作对工程做了修改但没有保存,系统会提示用户保存工程。如果该工程是新建立的,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,这样便于对工程进行管理。

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。

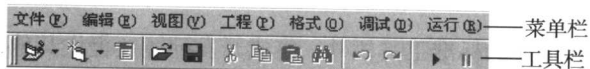


图 1-3 Visual Basic 集成开发环境的菜单栏和工具栏

1. 菜单栏

集成开发环境的菜单栏包含了 Visual Basic 中的大部分功能,图 1-3 的上半部分显示了 Visual Basic 集成开发环境的部分菜单。需要注意的是,Visual Basic 有 3 种版本:学习版、专业版和企业版,每种版本的菜单都有所不同。常用的菜单有:

- ◆ **文件 (File)** ——其选项内容较多,主要由 3 部分组成:一是工程的创建、打开、关闭、另存为等功能;二是打印设置,如窗体界面打印、源代码打印、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)** ——包括常用的编辑选项,如查找、复制、粘贴、删除、撤销、重做等。由于集成开发环境是用于程序编制的,因此还包括一些与此有关的特殊选项,如代码行的缩进设置、显示过程、显示属性、书签操作、定位到语句、插入文件等。
- ◆ **视图 (View)** ——主要用于控制和管理集成开发环境中的子窗口及工具栏,如显示 / 隐藏控件工具箱、显示 / 隐藏属性窗口、设置工具栏、查看对象等。
- ◆ **工程 (Project)** ——用于对工程中的各种部件进行管理,如添加窗体、添加类、添加模块、删除部件、设置工程属性等。该菜单很重要,在规模较大的工程中使用频率很高。
- ◆ **格式 (Format)** ——在界面设计过程中,经常需要调整控件的大小、位置、间距、对齐方式等,使用“格式”菜单,可以很快捷地完成这些操作。要快速设计出美观的程序界面,熟练掌握“格式”菜单的使用是非常必要的。
- ◆ **调试 (Debug)** ——程序的编制不是一蹴而就的,对于规模较大的程序更是如此。编制程序时首先要保证程序的正确性,然后还要保证开发的高效性。编写程序时,出现错误是很正常的现象,关键是如何快速找到这些错误并进行改正。要快速找到程序中的错误,除了编程人员的经验外,熟练掌握软件开发环境提供的调试工具也是非常重要的。“调试”菜单提供多种调试工具,借助这些调试工具,编程人员可以实现高效的程序调试。
- ◆ **运行 (Run)** ——主要用于执行应用程序,检验其功能及正确性。通过“运行”菜单,可以将目前的使用模式改变为运行模式或中断模式,一般要与“调试”菜单配合使用。

- ◆ **工具 (Tools)** ——主要包括 3 部分功能：对集成开发环境进行定制，向程序代码中添加过程和属性，激活菜单编辑器。
- ◆ **外接程序 (Add-Ins)** ——包含两部分功能：一是打开 Visual Basic 环境下的数据库管理器；二是管理外部程序，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。

2. 工具栏

在 Visual Basic 集成开发环境中，有些功能使用很频繁，例如保存文件、复制、粘贴等，为了方便用户使用这些功能，将这些功能按照类别放置在标准工具栏内，其使用方法和 Office 系列软件的工具栏相似。标准工具栏外观可参考图 1-3 的下半部分，在此只显示了其中一部分。

除了图 1-3 所描述的标准工具栏外，集成开发环境还提供其他类别的工具栏，如程序调试工具栏。用户可以根据需要对这些工具栏进行定制，包括是否显示某种工具栏以及显示位置，操作方法是：将鼠标指针指向某个工具栏，单击右键，然后从弹出的快捷菜单中选择需要的工具栏，使其显示或隐藏。

1.1.3 工程资源管理器窗口

工程是 Visual Basic 中的重要概念。使用 Visual Basic 开发的应用程序可能规模比较大，包含多个窗体、模块、自定义的类、其他文档等多个逻辑部件，每个部件对应一个文件。为了便于对这些部件进行管理，提出了工程的概念。工程资源管理器是集成开发环境中对工程及其所属文件进行管理的工具，它以树形结构图的方式将工程中所有文件按照类别组织在不同的文件夹下，查看时非常方便。在该窗口下，可以设置工程属性、快速切换要操作的文件，也可以快捷地添加和删除任何类型的文件。

由于工程的规模和功能不同，因此工程资源管理器的组成情况可能也不一样，图 1-4 显示了一个工程的组成。在进行界面设计或代码设计时，如果希望有更大的空间来进行上述操作，可以暂时将工程资源管理器窗口关闭，需要使用时可以利用“视图”菜单中的“工程资源管理器”选项打开。



图 1-4 工程资源管理器窗口

工程资源管理器可以同时管理多个工程，这些工程组成工程组。一般情况下，工程之间存在某种逻辑上的联系。显示在工程资源管理器窗口中的工程可能有多个，但是只能有一个工程是活动的，也就是当运行程序时，运行的是处于活动状态的工程。在多个工程同时存在的情况下，编程人员可以选择运行某个工程。但在运行指定工程之前，需要先把该工程设置

为活动状态，操作步骤是：将鼠标指针移到工程名称上，单击右键，选择快捷菜单中的“设置为启动”选项。

在工程中添加、删除和保存部件的方法如下：

- ◆ **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，此时会显示“添加”对话框。“添加”对话框中包括两个选项卡：“新建”选项卡用于创建新的部件；“现存”选项卡用于将已经存在的部件添加到指定工程中，该功能可以实现多个工程间部件的共享。
- ◆ **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”选项即可。注意，部件被删除后，只是它和工程之间不再有任何关系，但部件对应的文件并没有被删除，在需要时可以重新添加到工程中。
- ◆ **保存部件**——当某些部件做过修改后，应该对修改的内容进行保存操作。保存内容的方式有两种：一是保存，用于将从未保存过的部件以文件的形式保存起来，这时会出现对话框，或者将已存在相应文件的部件再次保存，这时不会出现对话框；二是另存为，指将该部件保存到另外一个新文件中，利用这种方式可以将部件保存成多个文件，供其他工程使用。

1.1.4 窗体窗口

在 Visual Basic 程序设计中，首先要做的工作是界面设计。界面设计在一个窗口中进行，该窗口被称为窗体窗口，如图 1-5 所示。界面设计的主要步骤是：将所需的控件在窗体窗口中的窗体（Form）表面上画出来，然后调整其位置、大小、颜色、标题等，在调整过程中可以用手工方式调整某些项，也可以通过设置相关属性来调整，还可以利用“格式”菜单进行调整。

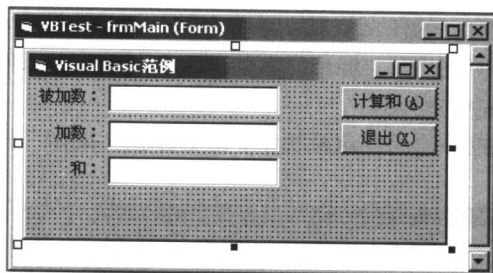


图 1-5 窗体窗口

每个窗体窗口只能容纳一个窗体；如果工程中有多个窗体，则会存在多个窗体窗口。需要对某个窗体进行界面设计时，可以在工程资源管理器中双击该窗体名称，这样包含该窗体的窗体窗口就会显示出来。窗体的界面设计完成后，应该关闭对应的窗体窗口，以保持集成开发环境的简洁。关于界面设计的内容，后面将进行详细讲解。

1.1.5 控件工具箱

标准的 Windows 应用程序以窗口为基本组织单位，在窗口表面有多种通用部件，如文本

框、下拉列表、文本标签等。在 Visual Basic 中, 这些部件实际上称为控件。在进行窗体的界面设计时, 主要工作就是把需要的控件按照功能要求放到合适的位置, 并为之设置合适的属性。为了方便用户使用这些控件, 集成开发环境提供一个窗口, 专门容纳控件, 称为控件工具箱, 如图 1-6 所示。

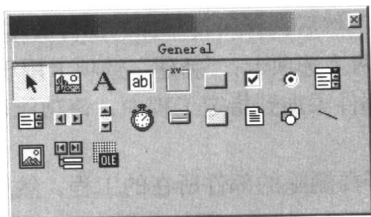


图 1-6 控件工具箱

把控件放置到窗体表面是一个很简单的过程, 操作方法有两种: 第一种是首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; 第二种是双击控件工具箱上的某个控件, 它就会自动出现在窗体中央, 其大小是默认的。这两种方式的结果都是在窗体上创建一个特定类别的具体对象。

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。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操作步骤如下:

- (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。
- (2) 单击“部件”选项。此时, 屏幕显示如图 1-7 所示的“部件”对话框, 在列表框中显示可以使用的控件列表。
- (3) 将需添加的各个控件前的复选框选中, 然后单击“确定”按钮, 选定的控件就会出现在工具箱中。

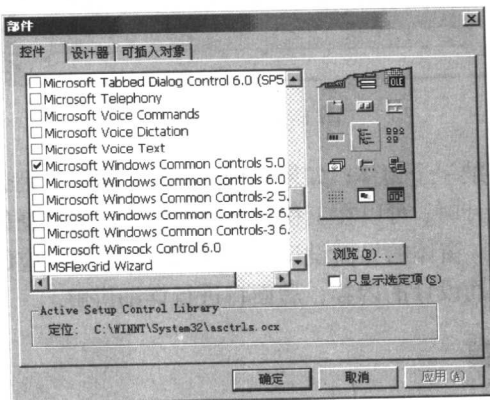


图 1-7 “部件”对话框

1.1.6 属性窗口

在 Visual Basic 中，对象的数据称为属性。属性包含两个内容：属性名称和属性值。例如，每个窗体都有一个标题，标题内容取决于窗体对象（Form）的 Caption 属性，如果其值为“VB 范例”，则窗体的标题会显示为“VB 范例”；如果将窗体的 Caption 属性的值设置为“我的第一件作品”，窗体的标题就会变成“我的第一件作品”。在该例中，Caption 是属性名称，“VB 范例”或“我的第一件作品”是属性值。

在 1.1.5 节中，我们了解到 Visual Basic 中有两种类型的对象：一种是界面设计阶段可以使用的控件，另一种是只能在代码中使用的引用对象。这两种对象都有属性，但设置方式不同：大部分控件对象的属性都可以在界面设计阶段设置，小部分属性只能在代码中设置或读取；引用对象的属性只能在代码中读取或改变。

在进行界面设计时，设置控件对象的属性是很重要的步骤。一般情况下每类控件都有多种属性，属性的设置也就要重复多次。属性设置的工作在属性窗口中进行，如图 1-8 所示。如果属性窗口没有显示出来，可通过以下步骤将打开属性窗口：打开“视图”菜单，选择“属性窗口”命令。



图 1-8 按属性名称的字母顺序组织的属性窗口

属性窗口由 3 部分组成：最上方是窗体及控件名称的下拉列表，列表中每项由两部分组成，左侧是控件或窗体的名称，右侧是控件或窗体的类型；中间部分是一个两列的属性表，左列是属性名称，右列是属性值；最下方是属性表中被选中属性的介绍。属性窗口所包含的内容是选中控件在设计阶段可用的属性，而另一部分只能在运行阶段使用的属性不会出现在属性窗口中。

控件的属性可能很多。因此，为便于使用，属性列表将所有属性按两种方式组织：一种按字母顺序排列，如图 1-8 所示；另一种按属性的类别组织，如图 1-9 所示。

窗体上可能有多个控件，如何设置指定控件的属性呢？有两种方法：第一种是将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”命令，属性窗口上方的下拉列表中会显示该控件的名称及类别，同时其属性显示在属性表格中；第二种是在属性窗口上方的下拉列表中选择相应的项，该项对应的控件或窗体的属性也会显示在属性表格中。

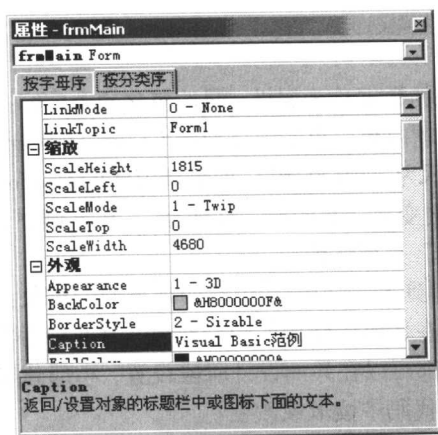


图 1-9 按属性类别组织的属性窗口

在界面设计中,只要向窗体中添加一个控件,则该控件的属性(在设计阶段可用的部分)就会显示在属性窗口中,并且每个属性都有一个系统事先规定的属性值,这种属性值称为默认值。一般情况下,大部分属性值采用默认值就可以了,但一些关键的属性值需要适当改变,例如窗体的名称、命令按钮上的文字等。要改变属性值,有三种方法:

- ◆ **第一种**——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。
- ◆ **第二种**——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,只需单击该三角形,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。
- ◆ **第三种**——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,表示该属性的值在对话框内进行设置,如字体 Font。

在设置窗体或者控件的属性时,首先应设置的是窗体或控件的名称,即 Name 属性。原因很简单,因为一个工程中可以包含多个窗体,这些窗体是通过 Name 属性来区分的。同样,一个窗体中可以包含多个相同类别的控件,例如文本框,这些文本框也是通过 Name 属性来区分的。

在添加一个窗体或控件时,它们都有一个系统默认的名称。例如,窗体名称一般是 Form1, Form2, Form3 等,命令按钮名称一般是 Command1, Command2, Command3 等。根据这些名称,我们只能推测它们的类别,但不能了解它们的功能。因此,最好不要使用这种系统默认的名称,应该按照功能的要求立即进行修改。在设置窗体或控件的 Name 属性时,要遵循见名知义的原则,即根据 Name 属性值,可以很清晰地判断它的类别及功能。

Name 属性值一般由两部分组成,即类别前缀和功能描述。例如 cmdAdd, cmd 是 CommandButton 的缩写,表示“命令按钮”,Add 的英文含义是“加”,表示“加”,所以 cmdAdd 的含义就是“加按钮”。表 1-1 给出了常用控件的前缀及范例。