

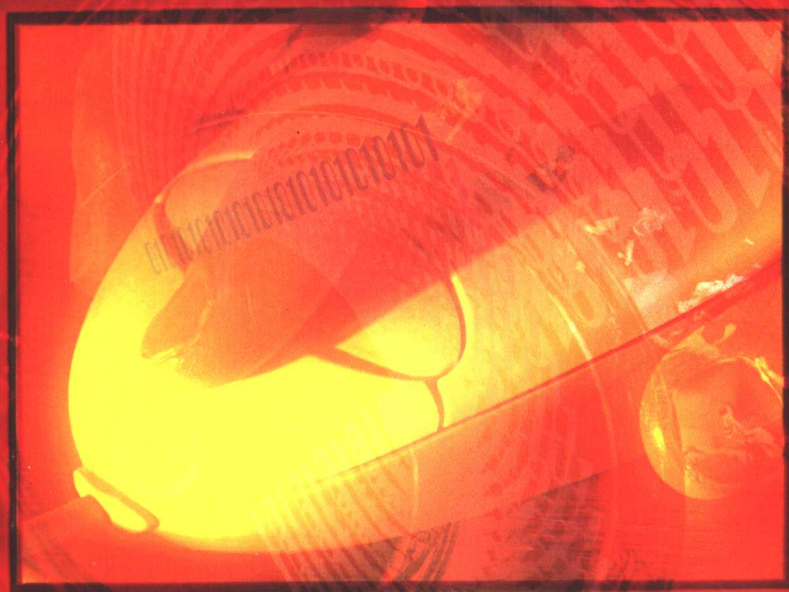
高职高专计算机类规划教材

面向对象程序设计

—— Visual Basic 6.0

王建明 主 编

康英健 副主编



机械工业出版社
China Machine Press

高职高专计算机类规划教材

面向对象程序设计

—— Visual Basic 6.0

主 编 王建明

副主编 康英健

参 编 倪浩如 龚戈淬 李春表

主 审 黄清虎



机械工业出版社

本书通过大量的实例,深入浅出地介绍了如何使用 Visual Basic 6.0 开发应用程序的方法与技巧。全书共分 13 章,主要内容包括:面向对象程序设计基本概念、Visual Basic 6.0 程序设计基础、窗体设计、文件系统操作、多媒体、API 函数的应用、数据库编程等。各章均附有习题,可供老师和学生选用。

本书将 Visual Basic 6.0 的可视化界面设计与程序代码设计部分有机地融合在一起,突出程序设计这条主线,注重实际操作能力的培养。内容精练,文字简洁,结构合理,例题经典,综合性强,符合高职教育的目标要求和教师教学与学生自学的习惯。可作为高职高专计算机专业及其它专业的可视化程序设计的教材,也可供广大计算机程序设计人员参考。

图书在版编目(CIP)数据

NJS304/06

面向对象程序设计——Visual Basic 6.0/ 王建明主编.

—北京:机械工业出版社,2001.12

高职高专计算机类规划教材

ISBN 7-111-09695-9

I.面... II.王... 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(2001)第 097929 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:王小东

封面设计:姚毅 责任印制:路琳

中国建筑工业出版社密云印刷厂印刷·新华书店北京发行所发行

2002 年 1 月第 1 版第 1 次印刷

1000mm×1400mm B5·8.5 印张·328 千字

0 001—5 000 册

定价:22.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话(010) 68993821、68326677-2527

前 言

计算机应用是高职各专业能力培养目标的重要组成模块，而掌握面向对象程序设计是其能力组成之一。Visual Basic 6.0 是一种可视化程序设计工具，采用面向对象、事件驱动的程序设计模式，同时又保留了结构化程序设计语言的优点，其解决问题的方式更符合人们思维活动方式。因此，在国内外各个领域应用非常广泛，许多计算机专业人员常利用它开发应用软件。

本书是根据高职高专计算机专业程序设计课程教学要求编写的。教材的编者都是从事程序设计教学多年的教师，了解教学目标要求，熟知教学内容与学生在学过程中可能出现的问题。因此，本教材适合于学生的学习习惯，符合教学规律要求。

本书以程序设计为主线，重点讲解计算机语言、结构化程序设计的基本知识和常用算法，面向对象程序设计的概念、方法和常用控件，并且对多媒体、数据库、API 函数应用程序开发等也作了较详细的讲解。对 Visual Basic 6.0 的可视化界面设计内容与代码设计部分进行了较好的融合，使得教材结构合理，内容精练，实用性强。不仅让学生了解面向对象程序设计概念，掌握可视化程序设计开发工具，而且通过大量典型示例，强化学生实际应用能力培养，使学生能够利用所学知识解决实际问题。

全书既注重原理又重视实践，配有大量的例题和一定的习题，这些精选的典型实例能帮助读者正确理解面向对象程序设计的相关概念。所有的例题均在计算机上运行调通，并作了较详细的注解，使学生更容易理解程序设计的基本思想。

本书由王建明主编并执笔其中第 1、2、3、4 章，第 5、7 章由倪浩如执笔，第 6 章由李春表执笔，第 8、9、10、11、12、13 章由康英健、龚戈淬执笔。本书由黄清虎老师主审，另外，包头医学院胡广春老师对全书进行了认真的审阅，并提出了许多宝贵意见，包头职业技术学院杨晶老师也对本书提出了有益的建议，在此谨表示衷心感谢。

由于时间仓促，书中缺点或错误在所难免，殷切希望广大读者批评指正。

编 者

2001 年 11 月

目 录

前言

第1章 面向对象程序设计的基本

概念.....1

1.1 Visual Basic 6.0 简介1

1.2 面向对象程序设计的基本概念 ...1

1.2.1 对象的属性、事件和方法1

1.2.2 窗体与控件3

1.2.3 事件驱动编程3

1.2.4 面向对象编程3

1.3 Visual Basic 6.0 集成开发环境.....4

1.3.1 集成开发环境的启动4

1.3.2 集成开发环境的构成5

1.3.3 集成开发环境的配置7

1.4 Visual Basic 6.0 编程步骤8

1.4.1 建立窗体.....8

1.4.2 添加控件.....8

1.4.3 设置属性.....10

1.4.4 编写代码和运行程序10

1.4.5 保存工程.....13

1.5 MSDN 的使用13

1.5.1 使用 MSDN 帮助系统13

1.5.2 使用“帮助”中的代码

示例14

思考与练习15

第2章 数据类型与表达式16

2.1 标准数据类型16

2.2 变量16

2.3 常量17

2.4 运算符与表达式19

2.5 常用内部函数22

2.5.1 数学函数22

2.5.2 字符串函数22

2.5.3 日期和时间函数23

2.5.4 数据类型转换函数23

2.5.5 格式输出函数 (Format) ...24

思考与练习25

第3章 Visual Basic 6.0 程序设计

基础26

3.1 输入输出控件与函数26

3.1.1 命令按钮的使用26

3.1.2 标签的使用28

3.1.3 文本框的使用29

3.1.4 输入和输出函数31

3.1.5 焦点与 Tab 键序34

3.2 赋值语句34

3.3 选择结构35

3.3.1 If 语句35

3.3.2 使用 If 函数37

3.3.3 Select Case 语句.....38

3.4 循环结构40

3.4.1 Do...Loop 语句40

3.4.2 For ...Next 语句.....41

3.4.3 循环的嵌套42

3.4.4 强制退出循环43

3.4.5 For Each ... Next 语句44

3.4.6 With 语句44

3.5 数组与控件数组45

3.5.1 数组声明	45	4.6.1 窗体模块	79
3.5.2 数组元素的使用	46	4.6.2 标准模块	80
3.5.3 控件数组	47	4.6.3 类模块	80
3.6 实现选项操作	51	4.7 变量的作用范围与生存期	81
3.6.1 框架	52	4.7.1 局部变量	81
3.6.2 选项按钮	52	4.7.2 模块变量	81
3.6.3 复选框	54	4.7.3 全局变量	82
3.6.4 列表框	55	思考与练习	82
3.6.5 组合框	59	第5章 窗体的设计	83
3.7 其它控件	60	5.1 窗体	83
3.7.1 定时器	60	5.1.1 窗体的属性	83
3.7.2 滚动条	62	5.1.2 窗体的方法	86
3.8 对象变量的使用	64	5.1.3 窗体的事件	88
3.8.1 对象变量	64	5.2 多重窗体的程序设计	92
3.8.2 New 关键字	66	5.2.1 添加新的窗体	92
3.8.3 确定对象变量所属的类	66	5.2.2 启动窗体	92
3.8.4 常用集合对象	67	5.3 多文档窗体	94
3.8.5 常用系统对象	67	5.4 菜单的设计	95
思考与练习	69	5.4.1 菜单的基本结构	96
第4章 过程与模块	71	5.4.2 Visual Basic 6.0 菜单 编辑器	96
4.1 事件过程	71	5.4.3 下拉式菜单的设计	98
4.2 通用过程	71	5.4.4 弹出式菜单的设计	100
4.2.1 通用过程的创建	72	5.5 工具栏的设计	102
4.2.2 通用过程的调用	72	5.6 状态栏的设计	108
4.3 函数过程	73	5.7 鼠标事件与键盘事件	110
4.3.1 函数过程的建立	73	5.7.1 鼠标事件	110
4.3.2 函数过程的调用	74	5.7.2 键盘事件	111
4.4 参数传递	75	5.7.3 窗体级键盘处理	112
4.4.1 形式参数与实际参数	75	5.7.4 控件的拖放	113
4.4.2 按值传递和按地址传递	75	思考与练习	116
4.5 过程的嵌套调用与递归调用	77	第6章 图形操作	117
4.5.1 过程的嵌套调用	77	6.1 坐标系统	117
4.5.2 过程的递归调用	78	6.1.1 标准坐标系统	117
4.6 代码模块	79		

6.1.2 自定义坐标系统	119	7.4.2 驱动器相关的方法和属性	161
6.2 图形控件	123	7.4.3 子目录相关的方法和属性	163
6.2.1 直线控件	123	7.4.4 与文件相关的方法和属性	166
6.2.2 形状控件	124	思考与练习	168
6.2.3 图形属性	125	第 8 章 错误处理与程序调试	169
6.3 图形方法	130	8.1 错误类型	169
6.3.1 Line 方法	130	8.1.1 语法错误	169
6.3.2 Circle 方法	131	8.1.2 运行错误	170
6.3.3 Pset 方法	132	8.1.3 逻辑错误	170
6.3.4 Point 方法	132	8.2 程序调试	171
6.4 图片框和图像框	133	8.2.1 设置断点	171
6.4.1 图片框	133	8.2.2 调试菜单	171
6.4.2 图像框	134	8.2.3 调试工具栏	173
思考与练习	137	8.2.4 调试窗口	174
第 7 章 文件操作	138	8.3 错误处理	176
7.1 文件系统控件	138	思考与练习	179
7.1.1 驱动器列表框	138	第 9 章 多媒体控件	180
7.1.2 目录列表框	138	9.1 Animation 控件	181
7.1.3 文件列表框	139	9.1.1 Animation 控件的属性	182
7.2 公共对话框的使用	142	9.1.2 控件方法	182
7.2.1 公共对话框控件	142	9.2 MMcontrol 控件	183
7.2.2 文件对话框	143	9.2.1 MCI 设备及 MCI 命令	184
7.2.3 颜色对话框	145	9.2.2 MMControl 控件的主要属性与事件	185
7.2.4 字体对话框	146	9.3 ActiveMovie 控件	191
7.2.5 打印对话框	147	9.3.1 ActiveMovie 控件的常用属性	192
7.2.6 帮助对话框	149	9.3.2 ActiveMovie 控件的主要方法	192
7.3 文件的访问	150	思考与练习	193
7.3.1 数据文件类型	150		
7.3.2 顺序文件的存取	151		
7.3.3 随机文件的存取	155		
7.3.4 二进制文件的存取	159		
7.4 文件系统对象	160		
7.4.1 FileSystemObject 对象的使用	160		

第 10 章 数据库编程基础	194	10.7.3 数据报表中的计算	230
10.1 数据库的基本概念	194	10.7.4 在数据表中添加日期、 时间、页号和标题	232
10.1.1 什么是数据库	194	思考与练习	233
10.1.2 关系数据库的基本概念	194	第 11 章 ActiveX 控件	234
10.1.3 数据库的分类	196	11.1 ActiveX 控件概述	234
10.2 数据管理器的使用	196	11.2 创建 ActiveX 控件	234
10.2.1 VisData 的工作环境	196	11.2.1 设计控件的界面	235
10.2.2 创建一个数据库	198	11.2.2 添加属性	236
10.2.3 打开数据库	199	11.2.3 测试用户控件	239
10.2.4 添加数据表	199	11.2.4 添加事件与方法	240
10.2.5 数据的添加、修改、 删除	201	11.2.5 添加属性页	242
10.2.6 建立一个查询	203	11.2.6 编译 ActiveX 控件	243
10.2.7 数据窗体设计器	205	11.2.7 使用 ActiveX 控件	243
10.3 Data 控件	205	第 12 章 使用动态链接库	245
10.3.1 Data 控件的常用属性	206	12.1 DLL 过程的声明和调用	245
10.3.2 Data 控件的 Recordset 对象	207	12.1.1 DLL 过程的声明	245
10.3.3 Data 控件的常用方法	207	12.1.2 使用 DLL 过程示例	246
10.4 数据绑定控件	208	12.1.3 访问 Windows API	247
10.4.1 数据绑定控件的常用 属性	209	12.2 Windows API 函数应用 示例	250
10.4.2 DBGrid 控件	215	思考与练习	252
10.5 结构化查询语言	217	第 13 章 发布应用程序	253
10.5.1 SQL 简介	217	13.1 程序打包	253
10.5.2 SQL 语句	217	13.2 展开应用程序	259
10.6 ADO 数据控件	220	附录 Visual Basic 6.0 中可以捕获的 错误信息	261
10.7 数据报表	224	参考文献	263
10.7.1 数据报表设计器概述	225		
10.7.2 创建数据报表	226		

第 1 章 面向对象程序设计的基本概念

1.1 Visual Basic 6.0 简介

Visual Basic 6.0 是微软公司 1998 年推出的面向图形用户界面的可视化程序设计工具。相对其它 Windows 编程语言, Visual Basic 6.0 具有更简捷的程序开发环境, 可以快速构建图形用户界面, 使用方便, 简单易学。对于初学者, Visual Basic 6.0 无疑是最理想的 Windows 编程工具。Visual Basic 6.0 比以往版本具有更新、更强的功能: 从开发小型应用程序到大型应用程序, 从数据库管理、客户/服务器软件到 Internet/Intranet 分布式应用程序的开发, Visual Basic 6.0 都能胜任。使用 Visual Basic 6.0 不仅可以感受到 Windows 编程的新概念、新技术、新方法, 而且是众多软件开发工具中编程效率较高的一种。

Visual Basic 6.0 是一种面向对象的程序设计语言。其特点是对对象的属性和方法都包含在对象的定义之中, 即将代码与数据封装到对象中。这项技术的应用, 使得 Windows 编程变得较为容易, 用户不需要关心对象的内部工作机制, 只要编写简单的代码就可以完成较为复杂的工作。

事件驱动编程是 Visual Basic 6.0 的另一特征。程序员只需编写响应用户动作的程序, 而不必花费精力去考虑程序的执行顺序。因此, 对于一个大型的应用程序, 可以将其分解成许多小的程序段, 而每个程序段都是针对某一对象的特定事件编写的, 从而使程序开发的周期大为缩短。

1.2 面向对象程序设计的基本概念

面向对象的程序设计 (OOP) 是一种新的程序设计方法, 它吸取了结构化程序设计的先进思想, 并结合了最新程序设计的新概念、新方法, 为软件开发提供了一种更为快捷、方便的工具, 使程序员更容易理解和管理更大、更复杂的程序。同时, 也引入了许多新的概念, 这些概念是理解和使用面向对象编程的基础和关键。

1.2.1 对象的属性、事件和方法

对象是程序中可以直接控制的实体。例如, 程序中的控件和窗体是对象,

整个应用程序也可以看成是一个对象。凡是在程序中能够直接引用、操作的实体都是对象。对象是由代码和数据组成的集合。对象是一个非常广泛的概念，就像自然界中的任何实体都可以称为对象一样。

1. 对象的属性

属性是指对象的特征。例如，控件的大小、位置和名称等。不同的对象其属性不同，通过修改对象的属性能够控制对象的外观和特性。

设置对象属性的方法有两种：在设计程序界面时利用属性窗口直接设置；在编写程序代码中进行设置，格式为：

对象名. 属性名 = 属性值

例如：Text1.Text = "欢迎使用 Visual Basic 6.0"

2. 对象的事件

事件是对象上发生的并能被对象识别的特定动作。例如，单击鼠标、鼠标移动等。事件是用户与程序交互的媒介，用户通过在不同的对象上按动鼠标或键盘，触发对象的事件，向程序下达命令。对象通过代码对所发生的事件作出反应，执行相应操作。离开事件，程序就难以确定用户要干什么，想干什么。这就是 Windows 程序的事件驱动机制。

对象的事件附属于对象，是由 Visual Basic 6.0 系统预先定义好的，用户不能在已有的对象上创建新的事件。

3. 对象的方法

方法是对象所能完成的某种操作。例如，控件的移动，窗体的显示，对象的打印操作等。方法对应于对象内部的一段代码，对象所能完成的特定功能，就是由这段代码来实现的。它既可能是一个函数，也可能是一个过程。

方法的内容是不可见的，用户只能按照约定的格式，在程序代码中调用它们。调用的格式为：

对象名. 方法名[参数]

例如：Label1.Move Left+100, Top+100 方法调用将 Label1 控件向右下方移动了 (100, 100) 个单位。

4. 事件过程

对象对事件的反应，是通过事件过程来完成的。事件过程是程序员为响应对象事件所编写的一段程序代码。当对象事件发生时，如单击鼠标，则去执行相对应事件过程中的代码。若对象的事件未发生，则程序代码不被执行。例如：命令按钮的单击事件。

```
Private Sub Command1_Click()
```

```
    Command1.Caption = "单击事件发生"
```

```
End Sub
```

当单击按钮 `Command1` 时, 按钮的标题才会改变。否则, 事件过程不被执行, 按钮的标题不会被改变。

一个对象能够响应的事件有很多, 但程序员只需编写必须响应的事件过程即可。

1.2.2 窗体与控件

窗体与控件是 Visual Basic 中两个最基本的对象, 是构成程序界面的主要元素。同时, 也是程序设计的基础。

1. 窗体

窗体是开发者定制应用程序界面的窗口。应用程序的图形界面, 就是由各种控件在窗体上的合理布局而构成的。窗体是可视化图形界面的基础, 我们在 Visual Basic 6.0 中创建的窗体就是 Windows 操作系统下运行的程序窗口。一个应用程序可以有一个或多个窗体。

2. 控件

控件是放置在窗体上的图形元件, 是 Visual Basic 6.0 中预先定义好的对象。用户可以通过控件向应用程序提供各种命令和控制信息。应用程序通过控件向用户显示处理结果及系统提示信息。

1.2.3 事件驱动编程

所谓事件驱动就是当应用程序中某对象的特定事件发生时, 通过执行一段 Basic 程序代码, 来完成该特定事件所对应的操作。这与我们生活和工作中完成任务的方式相同, 例如, 按动电源开关(触发事件), 电动机转动(作出反应, 完成某项操作)。事件可以由用户通过鼠标或键盘触发, 也可以由系统触发。当没有任何事件发生时, 应用程序就处于空闲阶段, 不进行任何操作, 并保持原状态不变。

在事件驱动的应用程序中, 代码的执行顺序是由事件发生的顺序决定, 而不是按照预定的路线执行。事件的触发顺序决定了代码的执行顺序。由于事件发生的顺序是不能预测的, 因此, 开发者必须对用户的动作、各种事件发生的可能性做出必要的假设, 从而确定程序代码和代码的结构。

1.2.4 面向对象编程

面向对象编程是使得 Windows 编程变得较为容易的关键技术之一。对象是由一组数据和处理数据的代码组成的独立单元, 是构成 Windows 程序的重要组成元素。

Visual Basic 6.0 编程是面向对象的编程, 这意味着利用 Visual Basic 6.0 编程, 只需跟对象打交道即可。需要完成的工作就是创建对象, 设置对象的

属性，捕捉来自对象的事件，处理发生的事件，而不必关心对象是如何产生和如何工作。例如，在窗体上显示一个按钮，只需创建一个按钮对象，不必使用一长串的作图语句绘制按钮。面向对象编程方法相对用户而言，它进行了数据封装，从而掩蔽了实现的复杂性，并且最大限度地提高了代码的可重用性，大大缩短了程序开发的周期。

1.3 Visual Basic 6.0 集成开发环境

启动 Visual Basic 6.0，将进入 Visual Basic 6.0 集成开发环境的主界面。它是一个集界面设计、代码编写、程序编译、测试与查错等多种功能于一体的程序开发工作环境。

1.3.1 集成开发环境的启动

Visual Basic 6.0 安装完成后，在开始菜单的程序组中将加入一个“Microsoft Visual Basic 6.0 中文版”的菜单选项。单击该选项菜单，即可启动 Visual Basic 6.0。

Visual Basic 6.0启动后，将显示如图1-1所示的“新建工程”对话框。若在其中的“新建”选项卡中，双击“标准 EXE”选项，就可以进入 Visual Basic 6.0的集成开发环境，同时，系统自动创建一个新的工程，如图1-2所示。如果选择对话框中的“现存”选项卡，那么可以打开一个已有的工程；选择对话框中的“最新”选项卡可以打开最近建立或最近使用过的一个工程。

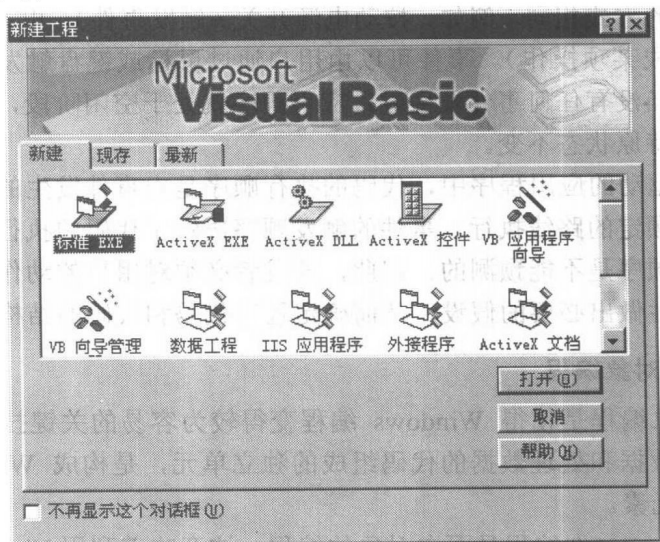


图 1-1 新建工程对话框

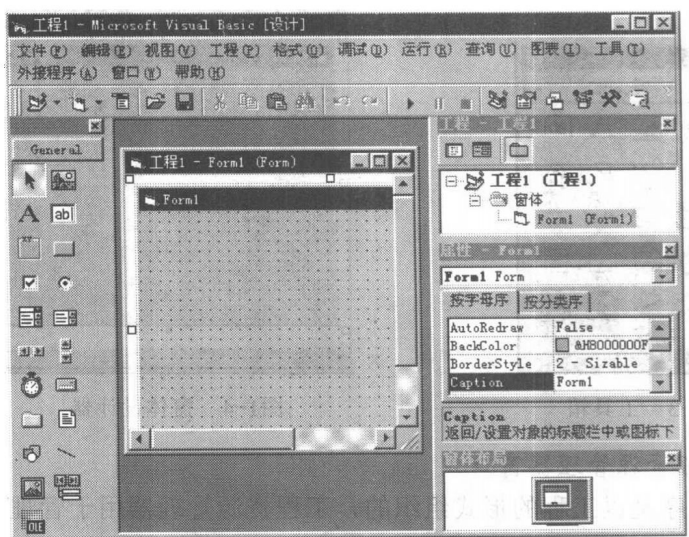


图1-2 集成开发环境

1.3.2 集成开发环境的构成

Visual Basic 6.0 的集成开发环境的桌面由菜单栏、工具栏、工具箱、窗体设计器、代码编辑窗口、工程资源管理器窗口、属性窗口以及窗体布局窗口、对象浏览器等部分组成。

1. 工具箱

工具箱中提供了 Visual Basic 6.0 中常用的内部控件，如图1-3所示。也可以向工具箱中添加新的控件。设计时双击工具箱中的控件，可直接将控件放置到窗体上。或者在工具箱中选择控件，然后在窗体上拖动，就可以画出控件。标准工具栏右侧的两组数据表示添加控件的位置和大小（单位为 $\text{Twip}=1/1440 \text{ in}$ ）。

2. 窗体设计器

窗体设计器用于创建应用程序的图形界面。启动一个新的工程，系统将自动创建一个空白窗体，名为 Form1，如图1-4所示。在空白窗体中添加控件、图形、图像就可绘制出自己的程序界面。

3. 代码编辑窗口

代码编辑窗口用于编写、显示应用程序代码。工程中的每一个窗体都对应于一个代码编辑窗口，所有程序代码均在此窗口中编写和修改。双击窗体的任意位置都可以打开代码编辑窗口，如图1-5所示。代码编辑窗口由对象列表框、过程列表框和代码编辑区组成，其中，对象列表框中列出了当前窗体中所有对象的名称，过程列表框中列出了当前所选对象的所有事件名。

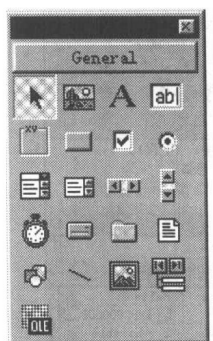


图1-3 工具箱

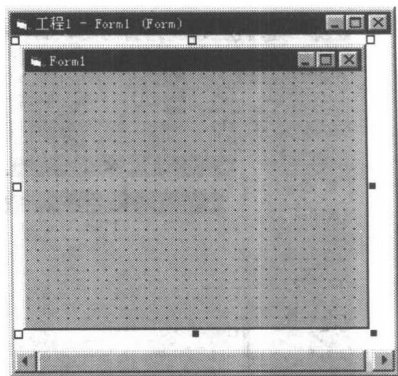


图1-4 窗体设计器

4. 工程资源管理器窗口

应用程序是以工程的形式组织的，工程资源管理器用于管理工程中所需要的各类文件和资源。单击标准工具栏的“工程资源管理器”按钮，可以打开工程资源管理器窗口，如图1-6所示。该窗口中列出了当前工程中使用的窗体及模块等项目文件。窗口上有三个按钮，利用它们可以在窗体设计窗口与代码编辑窗口之间快速切换。

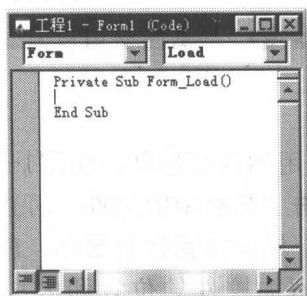


图1-5 代码编辑窗口

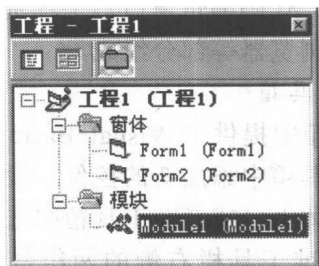


图1-6 工程资源管理器

5. 属性窗口

属性窗口用于设置和显示选定对象的属性值。单击标准工具栏的“属性窗口”按钮，可打开属性窗口，如图1-7所示。在程序设计阶段，可以通过属性窗口修改对象的属性值。该值将作为程序运行时，对象属性的初始值。

6. 窗体布局窗口

窗体布局窗口用于调整程序运行时，窗体在屏幕中的显示位置。单击标准工具栏的“窗体布局窗

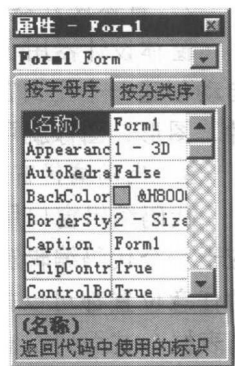


图1-7 属性窗口

口”按钮，可打开窗体布局窗口，如图1-8所示。拖动窗口中的小屏幕图标，可以改变窗体的显示位置，同时在标准工具栏右侧的两组数据，表示出调整后窗体相对屏幕的位置和大小，如图1-9所示。

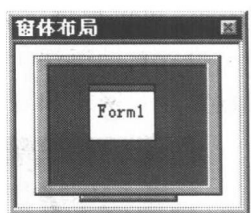


图1-8 窗体布局窗口

7. 对象浏览器

对象浏览器用于查看工程中有效的对象、类以及对象的属性、事件、方法和常数，还可查看工程的类代码、模块代码和过程代码，并可将代码过程粘贴到自己的应用程序中。按<F2>键或选择“视图”菜单的“对象浏览器”可显示对象浏览器窗口，如图1-10所示。对象浏览器的检索部分由工程/库列表、类列表、成员列表组成，窗口下面的文字部分是对所选内容的简要说明。

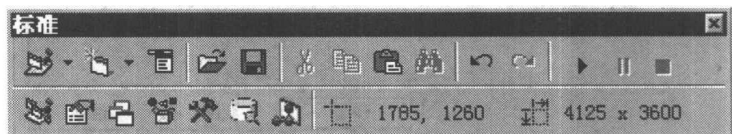


图1-9 标准工具栏

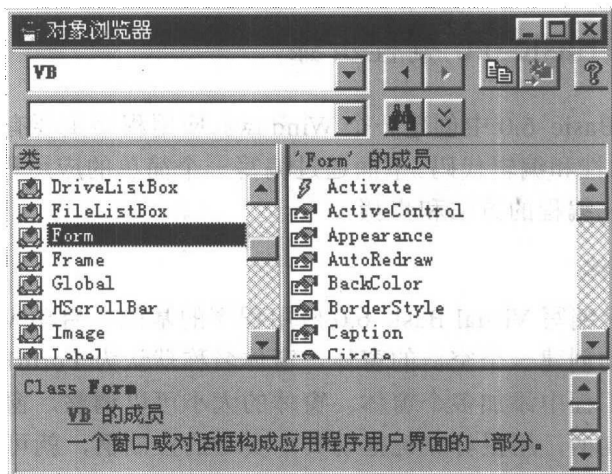


图1-10 对象浏览器

1.3.3 集成开发环境的配置

Visual Basic 6.0 允许程序设计员根据自己的需要重新配置集成开发环境。方法为：从“工具”菜单中选择“选项”命令，打开选项对话框，如图

1-11 所示。可以根据需要对其中的项目进行重新设置。例如，在编辑器选项卡中，设置自动语法检查，要求变量声明，自动列出成员等；在通用选项卡中，设置显示网格和网格大小等。

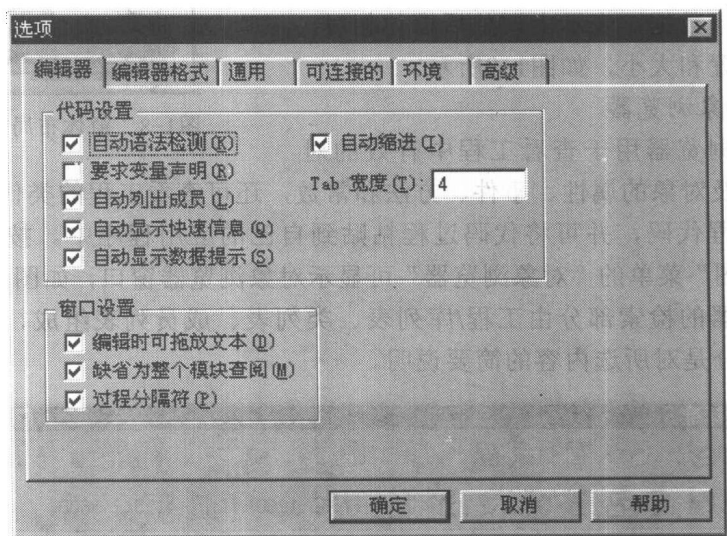


图1-11 选项对话框

1.4 Visual Basic 6.0 编程步骤

在 Visual Basic 6.0 中创建一个 Windows 应用程序主要有三大任务：设计界面、设置属性和编辑代码。下面通过编写一个简单的应用程序说明 Visual Basic 6.0 可视化编程的方法和步骤。

1.4.1 建立窗体

建立窗体是编写 Visual Basic 6.0 应用程序的基础。启动 Visual Basic 6.0 后，系统会自动创建一个空白的窗体，窗体名称默认为 Form1。根据需要的不同，可以向工程中添加多个窗体。窗体的大小可以调整，窗体周围有八个小块，叫控制柄，用鼠标左键拖动其中实心的控制柄，就可以改变窗体的大小，还可以通过改变窗体的 Width 和 Height 属性来改变窗体的大小。

窗体的位置可以通过改变窗体的 Top 和 Left 属性值来实现精确定位；也可通过窗体布局窗口调整程序运行时窗体在屏幕中的位置。

1.4.2 添加控件

通过向窗体中添加控件，实现图形用户界面的设计。向窗体上添加控件

的方法为：单击工具箱中控件图标，然后在窗体中用左键拖动鼠标，即可在窗体上画出相应的控件。

1. 改变控件大小

选中需要调整大小的控件，用鼠标左键拖动周围的控制柄，直到理想大小后，松开鼠标即可；也可以通过改变控件的 **Width** 和 **Height** 属性，改变控件的大小。

2. 改变控件位置

用鼠标左键拖动想改变位置的控件至新的位置后，放开鼠标即可；也可通过改变控件的 **Top** 和 **Left** 属性值来精确定位控件在窗体中的位置。

注意：对窗体而言，**Top** 和 **Left** 属性值是以屏幕的左上角为坐标原点的；对控件而言，**Top** 和 **Left** 属性值是以窗体工作区的左上角为坐标原点的。

3. 锁定控件

当控件的位置已经确定，不希望随意改动时，可选择“编辑”菜单中的“锁定控件”命令，锁定当前窗体上的所有控件。但其它窗体上的控件不受影响。由于该命令具有复选框的性质，所以利用同样的方法可以撤消控件的锁定。

注意：当控件锁定后，仍可通过改变控件的 **Top** 和 **Left** 属性，改变控件的位置。

4. 选择多个控件

为了同时对多个控件进行操作，需要同时选择多个控件，方法与 Windows 下选择多个文件的方式相同，即利用 <Shift> 或 <Ctrl> 键，再单击其它要选择的控件即可。

当多个控件被选择后，若移动其中的一个控件，将会引起其它控件一起移动，但各控件之间的相对位置不变。若需要调整多个控件的位置，则可先选中要调整的多个控件，然后使用“格式”菜单中的“对齐”、“水平间距”、“垂直间距”命令来完成。当两个控件重叠时，使用“格式”菜单中的“顺序”命令，可设置哪个控件在上。

5. 删除控件

要删除窗体上的控件，只要选中它，然后按下 键即可。

6. 复制控件

当窗体上需要同类的多个相同的控件时，可通过复制控件的方法来实现。方法为：选中要复制的对象，使用“复制”、“粘贴”命令，系统会弹出一个对话框，询问是否创建一个控件数组，如图 1-12 所示。选择“否”，复制好的控件将出现在窗体的左上角，然后将其移动到需要的位置即可。如果