

● 阎红灿 主编

计算机新技术

DIY

JISUANJI XINJISHU **DIY**

冶金工业出版社

计算机新技术 DIY

阎红灿 主编

北 京
冶金工业出版社

内 容 提 要

本书是非计算机专业研究生的教学用书,与前期计算机技术应用基础课程紧密衔接,主要内容包括:面向对象的概念和程序设计方法、VB 语言程序设计基础及面向对象编程的高级应用、VFP 数据库的维护操作及应用系统开发、网络基础理论及组网技术、网页制作等。

本书内容丰富,编程篇深入浅出,所有实例均上机通过;网络篇言简意赅,适合自学。通过学习本书,读者能够全面掌握面向对象的编程技术及数据库应用软件的开发技术,了解和掌握一般局域网的构建理论,并具有网页制作能力。

图书在版编目(CIP)数据

计算机新技术 DIY/阎红灿主编. —北京:冶金工业出版社,2003.1

ISBN 7-5024-3206-X

I. 计… II. 阎… III. 电子计算机—研究生—教材 IV. TP3

中国版本图书馆 CIP 数据核字 (2002) 第 101417 号

出版人 曹胜利 (北京沙滩嵩祝院北巷 39 号, 邮编 100009)

责任编辑 程志宏 美术编辑 李心 责任校对 符燕蓉 责任印制 李玉山

北京鑫正大印刷有限公司; 冶金工业出版社发行; 各地新华书店经销

2003 年 1 月第 1 版; 2003 年 1 月第 1 次印刷

787mm×1092mm 1/16; 17.5 印张; 423 千字; 272 页; 1—1500 册

36.00 元

冶金工业出版社发行部 电话: (010)64044283 传真: (010)64027893

冶金书店 地址: 北京东四西大街 46 号 (100711) 电话: (010)65289081

(本社图书如有印装质量问题,本社发行部负责退换)

《计算机新技术 DIY》编委会

主 编 阎红灿

主 审 关启明

副主编 张俊一 张淑芬

编 委 付景红 赵艳君 陈学斌 魏明军

张春英 高爱民 富 立 刘保相

前 言

当前,以计算机和信息技术为核心的知识经济时代正在崛起,一场世界范围的信息革命正方兴未艾。为适应社会信息化建设的需要,编者特编写了本书作为理工科非计算机专业类硕士研究生的学位课程教材。期望通过本书的学习能进一步扩大在校研究生知识领域,优化知识结构、强化高层次人才的计算机应用能力和工程设计能力。在计算机和计算机网络日益成为人们工作平台、计算机应用日益普及的时代,为培养计算机知识的学习接受能力和提高计算机应用水平打下理论和应用基础。

Visual Basic 和 Visual FoxPro 作为 Windows 平台的开发工具,既继承了其前期版本所具有的程序设计语言简单易学、易用的特点,又在编程系统中引入了面向对象的机制,用一种巧妙的方法把 Windows 编程的复杂性封装起来,提供了一种可视界面的设计方法,使用窗体和控件来设计应用程序界面,从而极大地提高了用户开发应用程序的效率。为了使初学程序设计的读者能快速地掌握开发基于图形界面应用程序的能力和解决实际问题的能力,编者本着加强基础、注重实践、勇于创新、突出应用的原则,力求使本教材达到可读性、适用性和先进性的和谐统一。为了便于读者自学,在本书的体系结构和内容上注重了由浅入深、深入浅出、循序渐进的方针。为了提高读者编程技巧和能力,本书在大部分章节中都提供了典型例题。

网络构建及网页制作也是高层次人才应具备的技术,在本课程教学中也渗透了这些内容,这部分内容主要靠学生自学完成。

本书共分两部分,第一部分为编程篇,共有 3 章,介绍了面向对象编程特点和 VB、VFP 程序设计方法;第二部分为网络篇,介绍了组建以太网的各种实用技术,包括双绞线的制作、网卡和交换机的选择、交换机和路由器的配置以及网页制作方法等。因为研究生具有一定的自学能力,通过本书内容的学习,学生完全具备了继续自学深造的基础。

本书由具备多年计算机教学经验的教师编写,其中第一章由陈学斌编写,第二章由张淑芬编写,第三章由阎红灿编写,第四章由赵艳君编写,第五章由付景红编写。全书由阎红灿统稿,关启明担任本书主审。本书在编写过程中,得到了河北理工学院及数理系领导、研究生处等有关部门的指导和大力支持。刘保相教授对本书的正式出版提出了许多宝贵意见并给予了有力的支持和帮助,编者在此一并表示衷心的感谢。

在本书的写作过程中曾参考了部分文献资料,编者在此向这些文献资料的作者深表谢意。

计算机技术的发展日新月异,由于编审者水平所限,加之时间十分紧迫,书中如有不妥之处,恳请读者及有关专家批评指正。

编者

2002. 10

目 录

第一部分 编程篇

1 面向对象程序设计概述	1
1.1 什么是面向对象程序设计	1
1.2 面向对象程序设计技术简介	2
1.3 对象和类的概念	2
2 Visual Basic 6.0 高级编程	3
2.1 VB的集成开发环境	3
2.2 VB中的对象	6
2.3 VB程序设计基础	8
2.4 窗体	34
2.5 基本控件	37
2.6 通用对话框和菜单	61
2.7 文件	71
2.8 调试程序和出错处理	89
2.9 生成可执行文件和制作安装盘	93
3 Visual FoxPro 6.0 面向对象程序设计	97
3.1 Visual FoxPro 6.0 概述	97
3.2 数据库语言基础	98
3.3 表的基本操作	104
3.4 用“数据库”管理数据	109
3.5 设计应用程序界面	110
3.6 表单高级设计	127
3.7 信息的显示与输出	138
3.8 VFP6.0 应用程序的集成管理和发布	141

第二部分 网络篇

4 网络技术	160
4.1 网络的基础理论	160
4.2 共享网络	175
4.3 小型交换网络	194
4.4 中型交换式网络	201
4.5 大型交换式网络	207
4.6 接入 Internet	208
4.7 网络布线	214

5 网页设计	217
5.1 FrontPage 2000 的基础知识	217
5.2 网页制作	219
5.3 超链接	226
5.4 表格	232
5.5 框架网页	239
5.6 表单	244
5.7 网页动态元素与动态效果	256
5.8 组织网站	262
5.9 站点的管理与发布	264
附 录	269
参考文献	272

第一部分 编程篇

1 面向对象程序设计概述

面向对象程序设计方法与编程技术不同于标准的过程化程序设计。程序设计人员在进行面向对象的程序设计时,不再是单纯地从代码的第一行一直到最后一行,而是考虑如何创建对象、利用对象来简化程序设计和提供代码的可重用性。对象可以是应用程序的一个自包含组件,一方面具有私有的功能,供自己使用;另一方面又提供公用的功能,供其它用户使用。

1.1 什么是面向对象程序设计

面向对象使程序员的观点从注重程序设计语言如何工作,转向注重于执行程序设计功能的对象模型,即不是着重于每个程序代码如何与程序的其它和系统的交互作用上。面向对象程序着重于建立能够模拟需要解决的现实世界问题的对象。

在面向对象的程序设计中,对象是组成软件的基本元件。每个对象可看成是一个封装起来的独立元件,在程序里承担某个特定的任务。因此,在程序设计时,不必知道对象的内部细节,只要在需要时,对对象的属性进行设定和控制即可。图 1.1 所示为对象和应用程序的关系。

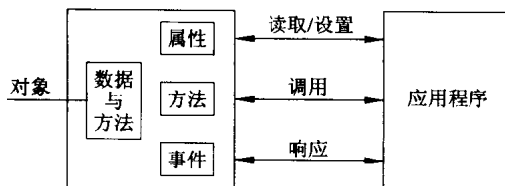


图 1.1 对象和应用程序的关系

程序设计者在使用对象时,虽然无需知道对象的内容,但是必须了解对象对外所提供的属性、方法和事件,就好比用户无需知道电视机内部的运转方式和电路板的焊接,但必须了解遥控器面板上按键的作用和操作程序才能享用它。

1.1.1 属性

所谓属性就是对象表现出来的特征、状态或行为。就像电视机有型号、尺寸、颜色、出厂日期等特征一样。不同的对象可以拥有各种相同或不同的属性,其中有些属性是只读并且无法改变的,而有些属性则可以通过设定来改变。

1.1.2 方法

方法是用来处理或操纵对象的途径。对象通常会提供一些方法,以便应用程序可以使用对象所提供的功能和服务。

1.1.3 事件和事件响应

事件就是对象所碰到的情况,当事件发生时,该事件的过程代码将被执行。也就是说,可以事先指定当事件发生时,对象要作出什么样的反应。例如,当用户单击命令按钮时,打开一个窗口,可把打开窗口的命令代码写在按钮的 Click 事件中。

1.2 面向对象程序设计技术简介

基于 Windows 消息机制的面向对象、事件驱动的应用程序与操作系统在方方面面有着密切的联系,事件的驱动依赖于操作系统的消息传递,直观的 Windows GUI 用户界面设计基于 Windows 友好的用户环境、丰富的系统调用等。面向对象程序设计有着明显的优点。

1.2.1 代码的可重用性

面向对象的程序设计方式引入了类的概念,并由此产生类库。对类的重用,可通过继承、实例化、引用类库等方式,大大加强了代码的可重用性。

1.2.2 程序一致性的维护

面向对象的数据和代码具有封装性,它将数据和代码封装为一体,这就是类的概念。对类进行实例化后,就产生了对象。它将对象作为程序运行的最基本实体,其中具有该类的所有属性和操作。各对象既是独立的实体,又可以通过消息相互作用,给程序的设计带来了很大的灵活性。

1.2.3 模块的独立性

面向对象模式以对象或数据为中心,以数据和方法的封装体为程序设计单位,程序模块之间的交互存在于对象一级,增强了模块的独立性。

1.2.4 可扩充性

类具有继承性、封装性、多态性等特点。继承可以有多种继承方式,从而可以有很多种在原有对象的基础上构造更复杂的类对象,而这种方式对原有对象的完整性没有破坏。

1.3 对象和类的概念

在面向对象的程序设计中,最基本的概念是对象和类。

对象是将数据和对数据的所有必要操作的代码封装起来的程序模块。它是具有各种特殊属性和方法的逻辑实体。

类是对一组具有公共方法和一般特性的对象的描述,它是对象的原型。

在面向对象的程序设计中,对象是构成程序的基本单位,是运行实体,而对象是类的实例化,每个类可以实例化出很多具有基本方法和数据的对象。并且,每个对象都应该属于某一类,只有被实例化以后,它才具备运行条件。

类可以由已存在的类派生而来,类之间是一种层次结构,处于上层的类称为父类,处于下层的类称为派生类或子类。任何一个类都继承了层次结构中上层类的全部特征。

类的继承方式有以下两种:

(1)单一继承:一个派生类至多有一个父类。

(2)多重继承:一个派生类有多个父类,具有各个父类的方法和数据。

2 Visual Basic 6.0 高级编程

2.1 VB 的集成开发环境

2.1.1 VB 的启动

(1)单击“开始”→“程序”→“Microsoft Visual Basic 6.0”→“Microsoft Visual Basic 6.0”，就会打开如图 2.1 所示的窗口。

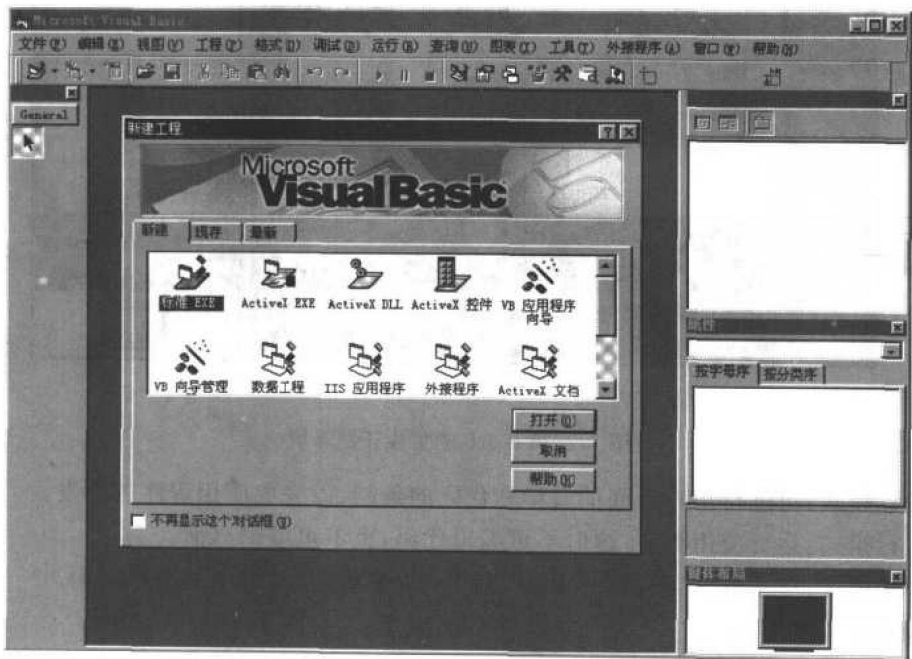


图 2.1 VB 6.0 的启动界面

在“新建工程”对话框中有三个标签：

“新建”：建立新工程。

“现存”：选择和打开现有的工程。

“最新”：列出最近使用过的工程。

(2)若要新建一个工程，在“新建”标签下，选择“标准 EXE”并单击“打开”按钮。

2.1.2 VB 6.0 的集成开发环境

VB 6.0 的集成开发环境如图 2.2 所示。

2.1.2.1 主窗口

(1)标题栏

同 Windows 98 界面一样，标题栏中包括控制菜单图标、标题、最小化、最大化和关闭按钮。

标题栏标题中的“[设计]”说明此时集成开发环境处于设计模式，在进入其它状态时，方括号中的文字将作相应的变化。

VB 有 3 种工作模式：设计 (Design) 模式、运行 (Run) 模式和中断 (Break) 模式。

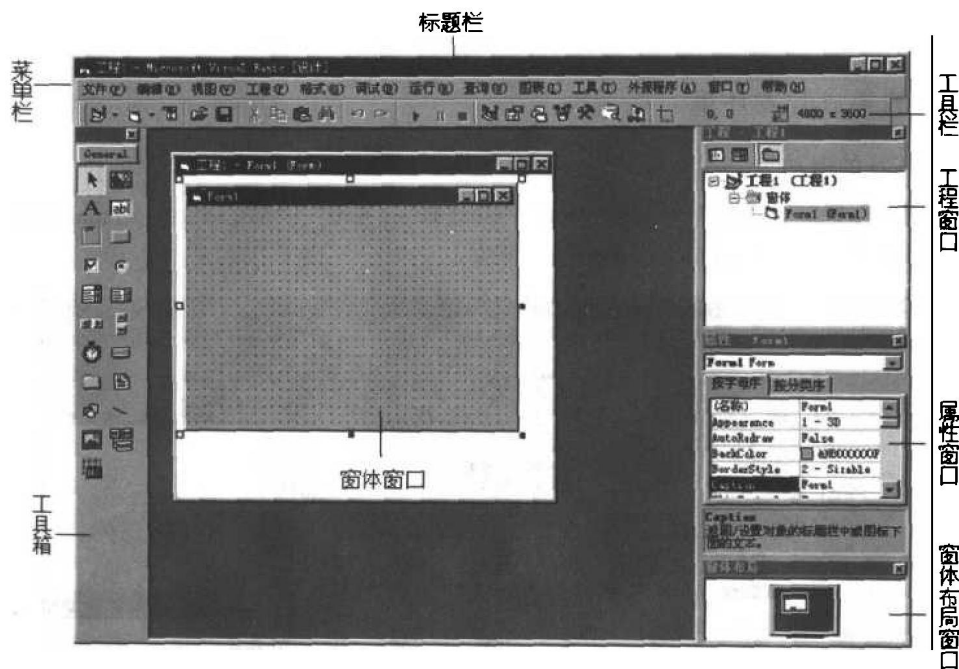


图 2.2 VB 6.0 的集成开发环境

- ① 设计模式:可进行用户界面的设计和代码的编制,以完成应用程序的开发。
- ② 运行模式:运行应用程序,这时不可编辑代码,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。

(2) 菜单栏

菜单栏中包括 13 个下拉菜单,汇集了程序开发过程中所需的命令。

(3) 工具栏

工具栏可以迅速地访问常用菜单命令。

2.1.2.2 窗体窗口

窗体窗口具有标准窗口的一切功能,可被移动、改变大小、最小化成图标。窗体是 VB 应用程序的主要部分,用户通过与窗体上的控件交互来得到结果。每个窗体窗口必须有一个唯一的名字,建立窗体时缺省名为 Form1, Form2...

除了一般窗体外,还有一种 MDI(Multiple Document Interface)多文档窗体,它可以包含子窗体,每个子窗体都是独立的。

2.1.2.3 工程资源管理器窗口

工程资源管理器窗口(如图 2.3)保存一个应用程序的所有属性以及组成这个应用程序的所有文件。VB 6.0 采用层次化管理方式显示各类文件,而且允许同时打开多个工程。

工程资源管理器窗口中有“查看代码”、“查看对象”、“切换文件夹”3 个标签:

- (1) 查看代码:切换到代码窗口,显示和编辑代码。
- (2) 查看对象:切换到模块的对象窗口。
- (3) 切换文件夹:工程中的文件按类型分或不分层次显示的切换。

工程资源管理器中主要包含以下 3 类文件：

(1)窗体文件(.FRM 文件):该文件存储窗体上使用的所有控件对象、对象的属性、对象相应的事件过程及程序代码。一个应用程序至少包含一个窗体文件。

(2)标准模块文件(.BAS 文件):包含所有模块级变量和用户自定义的通用过程。

(3)类模块文件(.CLS 文件):可以用类模块来建立用户自己的对象。



图 2.3 工程资源管理器窗口



图 2.4 属性窗口

2.1.2.4 属性窗口

所有窗体或控件的属性如颜色、字体、大小等都可以通过属性窗口(如图 2.4 所示)来设置或修改。属性窗口由以下部分组成：

(1)对象列表框:用户可以单击对象列表框右边的箭头,弹出所选窗体包含的对象的列表。

(2)属性显示排列方式:有“按字母序”和“按分类序”两个标签。

(3)属性列表:列出所选对象,在设计模式可更改的属性及缺省值,对于不同对象,它所列出的属性也是不同的。属性列表由两部分组成:左边列出的是对象的各种属性;右边列出的是相应的属性值。用户可以选择某一属性,对该属性值进行设置或修改。

(4)属性含义说明:当在属性列表框中选取某一属性时,在该区显示所选属性的含义。

2.1.2.5 代码窗口

代码窗口(如图 2.5)是专门用来进行程序设计的窗口,可在其中显示和编辑程序代码。用户可以打开多个代码窗口,查看不同窗体、标准模块中的代码,并可在各个窗口间复制代码。打开代码窗口有以下 3 种方法:

(1)从工程资源管理器窗口中选择一个窗体或标准模块,并选择“查看代码”标签。

(2)用鼠标双击窗体的某一控件或窗体本身。

(3)从“视图”菜单中选择“代码窗口”命令。

代码窗口主要包括以下 3 部分:

(1)对象列表框:下拉式列表框,存储并显示当前被选中窗体及窗体中的所有控件名。其中“(通用)”用来声明非局部变量或由用户编写自定义函数和过程。

(2)过程列表框:下拉式列表框,存储并显示窗体或所选控件的事件过程。在对象列表框中选择对象,在过程列表框中选择事件过程,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)代码编辑区:输入程序代码。

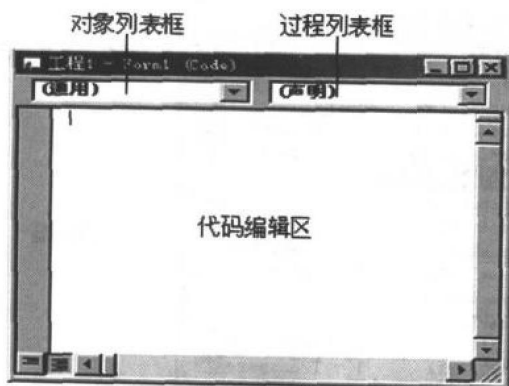


图 2.5 代码窗口

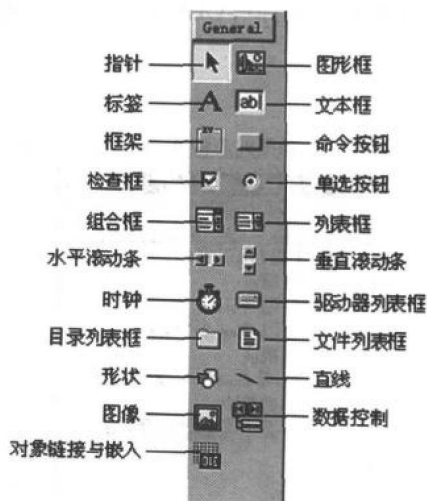


图 2.6 工具箱

2.1.2.6 工具箱窗口

工具箱窗口(如图 2.6),它由 21 个图标按钮构成,其中  为指针控件,用于移动窗体和控件,其余 20 个控件称为标准控件,用户可以通过“工程”菜单的“部件”命令将系统提供的其它控件装入工具箱。

2.2 VB 中的对象

2.2.1 VB 中对象的概念

学习 VB 就要以“对象”的眼光去看待整个程序设计。“对象”是面向对象程序设计的核心,明确这个概念对理解面向对象程序设计来说至关重要。那么,对象是什么?它在程序中是如何体现的?

对象的概念并不神秘,它就来源于生活之中。在现实生活中,我们时时刻刻都在和对象打交道,我们坐的椅子、穿的衣服、住的房子无一例外都是对象。

如果把问题抽象一下,会发现这些现实生活中的对象有两个共同的特点:第一,它们都有自己的状态,例如一个球有自己的质地、颜色和大小;第二,它们都具有自己的行为,比如球可以滚动、停止或旋转。在面向对象的程序设计中,对象的概念就是对现实世界中的对象模型化,它们同样有自己的状态和行为。只不过在这里,对象的状态用数据来表示,就称为对象的属性;而对象的行为用对象中的代码来实现,就称为对象的方法,因此,对象是代码和数据的组合。不同的对象会具有不同的属性和方法,当然也不排除会有部分重叠。

在 VB 中,最主要的两种对象是窗体和控件。窗体就是在进入 VB 开发环境时所看到的 Form1,它是创建应用程序界面的基础。而控件就是在“工具箱”中所看到的一个个按钮,所有

的控件都必须添加在窗体上。

类的概念要比对象更加抽象一些,它是 VB 为了描述具有相同特征的对象而引入的概念。类是用来创建对象的模板,它包含所创建对象的状态描述和方法定义,而对象只是类的一个实例。

为了说明类和对象的关系,我们仍然使用球的例子。其实“球”这个词本身就是一个类,它所描述的只是一个由中心到表面各点距离都相等的立体,它可以有许多属性,例如“颜色”、“质地”、“半径”等;也可以有自己的方法,例如“滚动”。世界上本来没有“球”这个东西存在,我们所见到的足球、铅球、玻璃球都是由“球”这个类创建出的对象,它们具有不同的颜色、大小和质地,但是都遵循“球”这个类的特性。

在 VB 的集成开发环境中,“工具箱”中的每一个控件按钮都可以看作是一个类,单击按钮后在窗体上所创建的诸如按钮、标签等元素,都是由相应的类创建的对象。

2.2.2 对象的属性

通过前面的介绍我们已经了解到,属性是对象中的数据,用来表示对象的状态。在 VB 中,可以通过属性窗口来查看对象的属性。

进行 VB 程序设计时,设置对象的属性是经常要做的事情。对象属性的设置可以在程序设计时进行,也可以在程序运行时使用代码设置属性的值。

(1)如果在程序设计时设置对象的属性,应首先在属性窗口选中要修改的属性,然后键入新的属性值。

(2)如果要在程序运行中更改对象的属性,可以使用赋值语句,它的格式是:

对象.属性 = 新值

注意:符号“.”要使用英文符号。

2.2.3 对象的方法

当我们设置对象的某项属性时,其实质就是把某些数据放入到对象中。根据上面介绍我们知道,对象中不仅可以包含数据,还可以包含代码。对象中的代码(包括过程和函数)就是对象的方法,方法决定了对象可以进行的动作。方法的内容是不可见的,当我们需要使用某个对象的方法时,只须使用这样的格式:

对象.方法

2.2.4 对象的事件及事件过程

世界上的每一种生物都会对外界的刺激做出反应,但是不同的生物对同一种刺激的反应却常常是不同的。在面向对象的程序设计中,同样存在着这样的问题,只不过在面向对象的程序设计中,为来自外部的刺激使用了一个术语——“事件”,而将对象对事件的反应称作“事件过程”。

2.2.4.1 事件

“可视化”和“事件驱动”是使用 VB 进行程序设计的精髓所在。所谓“事件驱动”就是说只有在事件发生时,程序才会运行;在没有事件的时候,整个程序是处于停滞状态的,这一点和以前的 DOS 程序有极大的差别。在 VB 设计的程序中,在程序中流动的不是数据而是事件。如果说属性决定了对象的外观,方法决定了对象的行为,那么事件就决定了对象之间联系的手段。

在 VB 中,事件就是能被对象所识别的动作,使用鼠标单击或者双击就是最常见的事件。此外,用户的键盘输入、鼠标的移动、窗体的载入,还有定时器产生的定时信号等等,都是事件。

一个对象能够响应哪些类型的事件以及它是如何识别这些事件的,实际上并不要用户去关心,VB 的每一个窗体和控件都有一个预定义的事件集,它们能够自动识别属于事件集中的事件。对象所识别的事件类型多种多样,但多数类型为大多数控件所共有。例如一个命令按钮和窗体都可以对 Click(单击)、DbClick(双击)和 KeyPress(按键)这样的事件做出响应。而某些事件只可能发生在某些对象上,

2.2.4.2 事件过程

相同的事件发生在不同的对象上所得到的反应可能是不一样的,造成这种差异是“事件过程”的缘故,每个对象对每个可以识别的事件都有一个“事件过程”,当事件过程不同时,对事件所表现出来的反应自然也会不同。

事件过程的语法是这样的:

```
Sub 对象名_事件()  
    处理事件的代码  
End Sub
```

2.3 VB 程序设计基础

在这一节中,介绍有关 Visual Basic 程序设计语言的基础知识,主要包括:

- (1) Visual Basic 中的数据类型。
- (2) Visual Basic 的常量、变量、运算符和表达式。
- (3) Visual Basic 的常用内部函数。
- (4) Visual Basic 的基本语句及结构。
- (5) Visual Basic 中数组及字符串的使用。

2.3.1 编码规则

VB 主要编码规则如下:

2.3.1.1 VB 代码中字母不区分大小写

(1) 对用户程序代码自动进行转换,对 VB 中的关键字,首字母总被转换成大写字母,其余字母一律转换为小写字母,这样提高了程序的可读性。

(2) 若关键字由多个英文单词组成,系统自动将每个单词的首字母转换成大写字母。

(3) 对于用户自定义的变量名、过程名、函数名,以第一次定义的为准,以后输入的自动转换成首次的形式。

2.3.1.2 语句书写自由

(1) VB 程序由若干行语句组成,在同一行上可以书写一条语句或多条语句,多条语句间用冒号“:”分隔。例如:

```
Form1. width = 2000 : Form1. Height = 1500
```

为了方便阅读,最好一行写一条语句。

(2) 一条语句如果在一行内写不下,VB 允许将单行语句分多行书写,行后加续行符(一个空格后面跟一个下划线“_”)。例:

```
Addresses = "河北理工学院" _
```

“数理系计算机教研室”

原则上,续行符应加在运算符的前后,续行符不应将变量名和属性名分隔在两行上。

2.3.1.3 在 VB 中可以使用注释

在程序中使用注释来说明编写代码或声明变量的目的。这样既方便了开发者自己,也为阅读源代码的人员提供了方便。

注释有三种情况:

(1)用 Rem 开头引导注释行。

(2)用撇号“'”开始引导语句后的注释。

(3)使用“设置/取消注释块”命令将若干语句行或文字设置/取消为注释块。

注释可以和语句在同一行,并写在语句的后面,也可单独占据一整行。例如:

Form1. width = 2000 '设置窗体的宽度

注意:在同一行内,续行符之后不能加注释。

2.3.1.4 VB 中的标识符

标识符是一个字符序列,用来标记变量名、函数名和类型名等。Visual Basic 要求标识符必须符合以下语法规则:

(1)组成标识符的字符有:A~Z、a~z、0~9 或下划线_。

(2)标识符必须是以字母或下划线开始,后跟字母、数字或下划线组成的字符串。

(3)标识符不能分行书写。

2.3.2 数据类型

计算机中的数据是表示信息的方式,它不仅用来表示数值信息,还用来表示诸如姓名、出生日期、是否团员等其它类型的信息,这就是数据类型。不同的程序设计语言,数据类型的规定和处理方法也各不相同。VB 不但提供了丰富的标准数据类型,还可以由用户自定义数据类型。

2.3.2.1 标准数据类型

(1)数值(Numeric)数据类型

数值数据类型有:整型(Integer)、长整型(Long)、单精度型(Single)、双精度型(Double)、货币型(Currency)和字节型(Byte)。

① 整型(Integer)和长整型(Long)用于保存整数。它可以是正整数、负整数或者 0。整型数在计算机中用两个字节存储,可表示数值范围为 $-32768 \sim +32767$;长整型数在计算机中用 4 个字节存储,可表示数值范围为 $-2147483648 \sim +2147483647$ 。

② 单精度浮点型(Single)用来表示带有小数部分的实数。在计算机中用 4 个字节存储,可表示数的范围为 $\pm 1.40 \times 10^{-45} \sim \pm 3.40 \times 10^{38}$ 。单精度浮点数最多有 7 位有效数字,用科学记数法表示如下:

$\pm aE \pm c$ 或 $\pm ae \pm c$

其中:a 为数字,可以是整数,也可以是浮点数;E(或 e)表示该数为单精度数;+、- 则表示数是正数还是负数;c 为 10 的指数值。

例如: $1.234E+4$ 表示 1.234×10^4 $-3.456E-3$ 表示 -3.456×10^{-3}

如果有效数字位数超过 7 位,当把它赋给一个单精度变量时,超出的部分会自动四舍五入。例如:把 213456.2567 赋给单精度变量 aa,在内存中 aa 的值为 213456.3。

③ 双精度浮点型(Double)与单精度数类似,用来表示更大范围的实数。在计算机中,每个双精度浮点数用 8 个字节存储。双精度浮点数最多可有 15 位有效数字。

双精度浮点数的科学记数法格式和单精度浮点数类同,只是把 E(或 e)换为 D(或 d):

$\pm aD \pm c$ 或 $\pm ad \pm c$

例如:314.159265358979D-2 表示 3.14159265358979。

④ 货币型(Currency)是一种专门为处理货币而设计的数据类型,是一种特殊的小数,保留小数点右边 4 位和小数点左边 15 位。货币型数据的范围为 -922337203685477.5808 ~ +922337203685477.5807,在计算机中用 8 个字节存储。如果变量已定义为货币型,且赋值的小数点后超过 4 位,那么超过的部分自动四舍五入。

例如,将 3.12125 赋给货币型变量 bb,在内存中 bb 的实际值是 3.1213。

⑤ 字节型(Byte)用来存储二进制数。如果变量包含二进制数,则将它声明为 Byte 数据类型。除一元减法外,所有可对整数进行操作的运算符均可操作 Byte 数据类型。因为 Byte 为 0~255 的无符号类型,所以不能表示负数。

(2) 日期(Date)数据类型

日期型数据在计算机中按 8 字节的浮点数存储,用来表示从公元 100 年 1 月 1 日到公元 9999 年 12 月 31 日的日期,时间范围则从 0 点 0 分 0 秒到 23 点 59 分 59 秒即 00:00:00~23:59:59。当任何可以作为日期和时间的字符被“#”符号括起来时,都是日期型数值。

例如:#03/10/2000#、#2000-03-10#、#March 1,2000#、#2000-3-15 13:30:15#等都是合法的日期型数据。

(3) 逻辑(Boolean)数据类型

逻辑数据类型用于逻辑判断,只有两个值“真”(True)和“假”(False)。若变量的值是“true/false”、“yes/no”、“on/off”信息,则可将它声明为 Boolean 类型。Boolean 的缺省值为 False。当把逻辑数据转换成整型数据时,“真”转换为 -1,“假”转换为 0;当把其它类型数据转换为逻辑数据时,非 0 数转换为“真”,0 转换为“假”。

(4) 字符(String)数据类型

字符串型数据是一组由计算机字符组成的序列,每个字符都以 ASCII 编码表示,在计算机中,一个字符占用一个字节。在 VB 中,字符串要用双引号括起来。例如“河北理工学院”、“123.456”等都是合法的字符串。

VB 中有两种字符串:变长字符串和定长字符串,其中变长字符串的最大长度为 $2^{31}-1$ 个字符,定长字符串的最大长度为 65535 个字符。

(5) 对象(Object)数据类型

对象数据用 4 个字节来存储,该 32 位地址可以引用应用程序中的对象,可以指定一个被声明为对象的变量去引用应用程序所识别的任何实际对象。

(6) 变体(Variant)数据类型

变体型数据是一种可以随时改变数据类型的数据,它为 VB 的数据处理增加了智能性。一个变量可以在设计时先定义为变体类型的变量,然后,在程序运行过程中,根据上下文的需要,再给变量赋任何类型的值,包括数值型、日期型、对象型、字符型数据,Variant 变量能够存储所有系统定义类型的数据,如果把它们赋予 Variant 变量,则不必在这些数据的类型间进行转换,VB 会自动完成任何必要的转换。例如: