

21世纪高等学校应用型规划教材

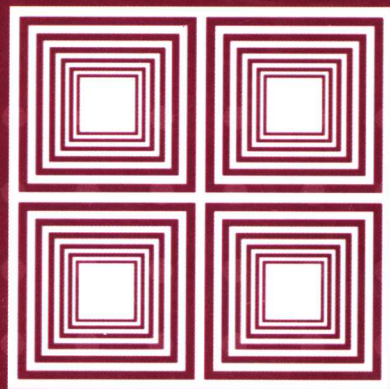


计 算 机 系 列

Visual Basic

程序设计学习指导

费贤举 王文琴 编著



中国电力出版社

www.infopower.com.cn

21世纪高等学校应用型规划教材



计算机系列

TP312
1933

Visual Basic

程序设计学习指导

费贤举 王文琴 编著



中国电力出版社

www.infopower.com.cn

内容提要

本书为《Visual Basic 程序设计教程》的配套学习辅导教程,面向初、中级用户,尤其适合于使用 Visual Basic 语言编写 C/S、B/S 数据库应用程序的用户,在内容安排上,本着从入门到精通的原则,内容合理、语言通俗易懂、事例详尽。全书分三部分,第一部分为实验指导,共 14 个实验(含两个综合实验);第二部分为模拟试题,考虑到 Visual Basic 作为全国计算机等级考试二级中比较多的一门语言课,根据国家最新等级考试大纲,精选了几套笔试与上机模拟试卷,并附了参考答案。第三部分附录,提供了一些实用资料。

本书可作为高等院校本专科 Visual Basic 课程配套实验指导教材使用,也可作为计算机等级考试参考书。

图书在版编目(CIP)数据

Visual Basic 程序设计学习指导 / 费贤举, 王文琴主编. —北京: 中国电力出版社, 2005.9

21 世纪高等学校应用型规划教材·计算机系列

ISBN 7-5083-1548-0

I.V... II.①费...②王... III. BASIC 语言—程序设计—高等学校—自学参考资料

IV.TP312

中国版本图书馆 CIP 数据核字(2005)第 066894 号

责任编辑: 逢积仁

丛 书 名: 21 世纪高等学校应用型规划教材·计算机系列

书 名: Visual Basic 程序设计学习指导

出版发行: 中国电力出版社

地 址: 北京市三里河路 6 号 邮政编码: 100044

电 话: (010) 68358031 (总机) 传 真: (010) 68316497, 88383619

本书如有印装质量问题, 我社负责退换

服务电话: (010) 88515918 (总机) 传 真: (010) 88518169

E-mail: infopower@cepp.com.cn

印 刷: 北京同江印厂

开本尺寸: 185×260 **印 张:** 13.25 **字 数:** 297 千字

书 号: ISBN 7-5083-1548-0

版 次: 2005 年 9 月北京第 1 版

印 次: 2005 年 9 月第 1 次印刷

印 数: 0001—5000 册

定 价: 20.00 元

版权所有, 翻印必究

21 世纪高等学校应用型规划教材·计算机系列

编 委 会

主任委员:

宗 健 常明华

副主任委员:

顾元刚 陈 雁 杨翠南 林全新 华容茂 曹泰斌

魏国英 邵晓根 庄燕滨 邓 凯 吴国经 常晋义

许秀林 谢志荣 张家超 陶 洪 龚兰芳 刘广峰

丁 雁 方 岩 王一曙

委 员:(以姓氏笔画为序)

丁志云 及秀琴 石振国 李 翊 吕 勇 朱宇光

任中林 刘红玲 刘 江 刘胤杰 许卫林 杨劲松

杨家树 杨伟国 郑成增 张春龙 闵 敏 易顺明

周维武 周 巍 胡顺增 袁太生 高佳琴 唐学忠

徐煜明 曹中心 曾 海 颜友钧

序 言

进入 21 世纪,世界高等教育已从精英教育走向了大众教育。我国也适应这一潮流,将高等教育逐步推向大众化。培养应用型人才已成为国家培养国际人才的重要组成部分,且得到了社会各界的广泛支持。于是一大批有规模、有实力、规范化、以培养应用型人才为己任的高等学校得到了长足发展。这类高校办学的一个显著的特点是按照新时代需求和当地的需要来培养学生,他们重视产学研相结合,并紧密地结合当地经济状况,把为当地培养应用型人才作为学校办学的主攻方向。

这类学校的教学特点是:在教授“理论与技术”时,更注重技术方法的教学。在教授“理论与实践”时,更注重理论指导下的可操作性,更注意实际问题的解决。因此,这些学生善于解决生产中的实际问题,受到地方企事业单位的普遍欢迎。

为满足这类高校的教学要求,达到培养应用型人才的目的,根据教育部有关重点建设项目的要求和相关教学大纲,我们组织了多年在这类高校中从教,并具有丰富工程经验的资深教授、高级工程师、教师来编写这套教材。

在这套教材的编写中,我们提倡“实用、适用、先进”的编写原则和“通俗、精练、可操作”的编写风格,以解决多年来在教材中存在的过深、过高且偏离实际的问题。

实用——本套教材重点讲述本行业中最广泛应用的知识、方法和技能。使学生学习后能胜任岗位工作,切实符合当地经济建设的需要和社会需要。

适用——本套教材是以工程技术为主的教材,所以它适用于培养应用型人才的所有高校(包括本科、专科、技术学院、高职等),既符合此类学生的培养目标,又便于教师因材施教。

先进——本套教材所选的内容是当今的新技术、新方法。使学生在掌握经典的技术和方法之后,可用教材中的新技术、新方法去解决工程中的技术难题,为学生毕业后直接进入生产第一线打下坚实的基础。

通俗——本套教材语言流畅、深入浅出、容易读懂。尽量避开艰深的理论和长篇的数学推导,尽量以实例来说明问题,在应用实例中掌握理论,使学生轻松掌握所学知识技能,达到事半功倍的效果。

精练——本套教材选材精练、详细而不冗长、简略得当,对泛泛而谈的内容将一带而过,对学生必须掌握的新技术、新方法详细讲,讲透、讲到位,为教师创造良好的教学空间和结合当地情况调整教学内容的余地。

可操作——本套教材所有的实例均是容易操作的,且是有实际意义的案例。把这些案例连接起来,就是一个应用工程的实例。通过举一反三的应用,使学生能够在更高层次上创造性地应用教材中的新思想、新技术、新方法去解决问题。

本套教材面向培养应用型人才的高等学校,同时也可作为社会培训高级技术人才的教材和需要加深某些方面知识技能的人员的自学教材。

编委会

前 言

Visual Basic 是在 Basic 的基础上经过不断改进和完善发展起来的一种可视化编程语言, 它覆盖了程序设计领域中文件访问技术、数据库访问技术、图形处理技术、多媒体处理技术、Internet 技术、通信技术等。Visual Basic 语言从 1991 年问世以来, 深受广大编程爱好者和专业编程人员的青睐。一方面, 它继承了 Basic 语言简单易学的特点, 使得初学者很容易地从入门到精通; 另一方面, 它巧妙地将 Windows 复杂的编程过程“封装”起来, 用户只要书写少量的代码就可以达到专业软件同样的水准。因此, 使用它设计程序, 可以极大地提高程序开发效率, 使程序员不再为语言而烦恼。

作为一门高质量的开发语言, 很多高校纷纷开设了相关课程, 但遗憾的是, 虽然目前国内 Visual Basic 程序设计语言方面的教材品种繁多, 但真正适合教学的却很少。

2001 年, 我们组织了几名多年从事 Visual Basic 语言教学的教师和科研人员, 根据基础教学的特点, 精心组织和编写了《Visual Basic 程序设计教程》。从 Visual Basic 语言的发展开始, 由浅入深地介绍了 Visual Basic 的基本语法、可视化窗体设计、图形处理技术、跟踪调试技术、数据库编程技术等方面内容, 结合具体的实例, 系统地讲述了 Visual Basic 程序设计方法。

本书为《Visual Basic 程序设计教程》的配套学习辅导教程, 面向初、中级用户, 尤其适合于使用 Visual Basic 语言编写 C/S、B/S 数据库应用程序的用户。在内容安排上, 本着从入门到精通的原则, 内容合理, 语言通俗易懂, 示例详尽, 如果能上机认真实践, 一定能获得很好的效果。同时考虑到 Visual Basic 作为全国计算机等级考试二级中的一门语言课, 根据国家最新等级考试大纲, 精选了几套笔试与上机模拟试卷, 并附了参考答案, 相信对许多非计算机专业选择 Visual Basic 为二级考试语言的学生会起到很好的帮助作用。

本书由费贤举提出编写思路与大纲。共 14 个实验, 包括两个综合实验, 其中实验 1、2、3、4、5、综合实验 (1) 由王文琴编写; 实验 6、7、8、9、10、11、12、综合实验 (2)、模拟试卷、附录由费贤举编写, 最后由费贤举统稿。另外, 唐学忠、王文和李亦飞为本书的出版做了大量的文字工作, 华容茂老师提出了一些很好的建议, 在此一并感谢。

限于水平, 书中不足之处在所难免, 敬请广大读者批评指正。读者在学习过程中如有其他问题或意见和建议, 可以直接与编者联系, 具体联系方式:

E-mail: feixj@czu.cn、wwqin@czu.cn。

作 者

2005 年 6 月

目 录

序 言

前 言

第 1 部分 实验指导	1
实 验 须 知	1
实验 1 Visual Basic 编程基础	2
实验 2 常用控件的使用	23
实验 3 多文档文本编辑器的程序设计	34
实验 4 Visual Basic 的程序调试方法	44
实验 5 文件访问方法	49
实验 6 图形图像处理技术	59
实验 7 创建和使用自己的类	70
实验 8 ActiveX 技术	77
实验 9 数据库访问方法	80
实验 10 多媒体播放器	91
实验 11 注册表访问与 API 调用	101
实验 12 网络编程	113
实验 13 综合实验 (1)	120
实验 14 综合实验 (2)	123
第 2 部分 模拟试题	125
全国计算机等级考试二级 Visual Basic 语言程序设计笔试模拟试卷 (1)	125
全国计算机等级考试二级 Visual Basic 语言程序设计笔试模拟试卷 (2)	132
全国计算机等级考试二级 Visual Basic 语言程序设计笔试模拟试卷 (3)	143
全国计算机等级考试二级 Visual Basic 语言程序设计笔试模拟试卷 (4)	154
全国计算机等级考试二级 Visual Basic 语言程序设计上机模拟试卷 (1)	164
全国计算机等级考试二级 Visual Basic 语言程序设计上机模拟试卷 (2)	167
全国计算机等级考试二级 Visual Basic 语言程序设计上机模拟试卷 (3)	170
全国计算机等级考试二级 Visual Basic 语言程序设计上机模拟试卷 (4)	173
全国计算机等级考试二级 Visual Basic 语言程序设计笔试模拟试卷参考答案	176

第1部分 实验指导

实验须知

上机实验是学习 Visual Basic 程序设计的重要环节。只有通过大量上机实践,才能熟练掌握 Visual Basic 的语法规则及界面设计,也只有通过认真实践,才能积累经验,逐步掌握程序设计的基本方法和技术。因此,根据具体的学习要求,有计划、有目的地安排好上机实验是非常重要的。

与传统的程序设计语言不同, Visual Basic 提供了良好的集成开发环境、所见即所得的用户界面、事件驱动的运行方式,有助于初学者调试程序和查找错误。

为了使同学们能学好 Visual Basic 程序设计这门课程,在本指导书中我们设计了 12 个实验和两个综合练习作业,每一个实验大体与一个章节相对应。综合练习旨在提高学生综合运用所学知识独立进行程序设计的能力,逐步培养学生进行较大项目开发的能力。本书从内容上主要分为 7 个部分:一是实验目的,即所做实验要达到的目标;二是实验预习与准备,是所涉及的知识内容及实验前应掌握的内容;三是实验内容与要求,其中一部分是例题示范,通过典型例子及详细的编程、界面设计的讲解,学会知识的应用,另一部分是练习题,通过课内实验和课后练习相结合的形式完成,加深巩固知识;四是实验示例,通过一些示例的练习,引导学生理解本章实验的基本方法和思路;五是实验过程与结果记录;六是实验心得与体会;七是课外练习题,目的是在课内实验的基础上适当提高要求,能够让学生融会贯通,举一反三。

为了提高上机的效率,请同学按照实验要求,在上机前设计好应用程序界面,编写好程序代码,仔细阅读并检查程序,选择好调试程序的数据,进行静态人工运行,给出程序预期的输出结果,最后再上机实现。这样可以使我们有效地利用上机老师辅导的时间,达到事半功倍的效果。

实验1 Visual Basic 编程基础

一、实验目的

1. 熟悉 Visual Basic 6.0 开发环境的基本组成及选项设置。
2. 掌握使用 Visual Basic 6.0 建立一个应用程序的基本操作步骤。
3. 掌握运行程序和编译程序的基本方法。
4. 掌握 Visual Basic 联机帮助的使用。
5. 掌握变量的定义和使用方法。
6. 掌握 If、Select 语句的使用方法。
7. 了解 Inputbox 函数和 MsgBox 函数的基本用法。
8. 掌握 For、Do 循环语句。
9. 掌握过程的定义及使用方法。
10. 掌握过程和函数中参数“按值传递方式”和“按地址传递方式”的不同。
11. 掌握函数的定义与使用方法。
12. 掌握过程和函数的区别与联系。
13. 理解和掌握数组、自定义类型、枚举的概念和定义。
14. 能熟练地运用数组、自定义类型、枚举编制简单的程序。
15. 掌握静态数组和动态数组的使用差别。
16. 应用数组解决与数组有关的常用算法。

二、实验环境

1. Visual Basic 6.0。

2. 在启动 Visual Basic 之前先建立一个 Vbwork 文件夹（例如：E:\Vbwork\实验1）用来存放本次 Visual Basic 实验所建立的文件。

三、实验预习与准备

1. 熟悉 Windows 98 或 Windows 2000 的操作环境。
2. 一个 Visual Basic 6.0 的应用程序也称为一个工程，由以下几部分组成。
 - ①跟踪所有部件的工程文件（.vbp）。
 - ②每个窗体的文件（.frm）。
 - ③每个窗体的二进制数据文件（.frx），它含有窗体上控件的属性数据。对含有二进制

属性（例如图片或图标）的任何.frm 文件都是不可编辑的，这些文件都是自动产生的。

- ④每个类模块的文件（.cls），该文件是可选项。
- ⑤每个标准模块的文件（.bas），该文件是可选项。
- ⑥一个或多个包含 ActiveX 控件的文件（.ocx），该文件是可选项。
- ⑦单个资源文件（.res），该文件是可选项。

3. 设计 Visual Basic 应用程序的步骤。

- ①设计用户界面。
- ②设置属性：对每一个控件进行属性的设置。
- ③编写程序（事件过程）代码。
- ④编译：使程序脱离 Visual Basic 环境，直接在 Windows 环境下运行。
- ⑤打包发布。

4. 复习变量、常量的定义和使用方法。

(1) 系统提供的常量。

Visual Basic 系统中还提供了为应用程序和控件定义的常量，这些常量位于对象库中，可通过对象浏览器查看。

如：颜色常量 VbRed、VbBlue 等，符号常量（Const 语句定义）

Const 语句的形式：

```
[ Public | Private ] Const <常量名> [ As Type ] = <表达式>
```

(2) Visual Basic 中变量的声明分为两种方法。

①隐式声明：允许对使用的变量未声明就直接使用

```
Private Sub Form_Click()  
    k&=1  
    Country$="China"  
    Print k,Country  
End Sub  
Option Explicit 语句
```

在模块中使用 Option Explicit 语句，系统将检查模块中所有未加显式说明的变量，一旦发现这样的变量存在，就会产生一个出错信息，提示使用者改正错误。

②要求变量声明

可以通过“工具”菜单→“选项”→“编辑器”选项卡设置。

(3) 变量的作用域与变量说明语句。

1) 作用域。

①局部变量：在过程中说明，仅在说明它的过程中有效。

②窗体/模块变量：在窗体或标准模块中说明，在定义该变量的模块或窗体的所有过程内均有效。

③全局变量：在模块或窗体中说明，在工程内的所有过程中都有效。

2) 作用时间。

①普通局部变量：只在声明它的过程内部有效

变量在过程真正执行时才分配空间。过程执行完毕后即释放空间，变量中存放的值也就不复存在。

②静态局部变量：只在声明它的过程内部有效

变量在第一次定义时分配空间。过程执行完毕后，只要程序还未结束，该变量的值就依然存在，该变量占用的空间不被释放。

```
Private Sub Command1_Click()
    Dim i As Integer
    Print i
    i = i + 1
    Print i
End Sub
```

3) 变量说明语句。

```
Dim <变量名> As <类型> [, <变量名> As <类型>]
```

—用于说明窗体变量或局部变量（最常用）

窗体级和全局变量应在“代码编辑器”窗口中的（通用）（声明）部分进行说明

```
例：Dim count As Integer
      Public name As String , age As Integer
```

局部变量仅在使用它的过程中说明。

```
例：Private Sub Command1_Click()
      Dim count As Integer
      .....
End Sub
```

用 Dim 语句说明的变量初值为 0 或空串。

变量名自动转换。

4) 不同作用范围的 3 种变量声明及使用规则

变长字符串（默认）：

```
Dim <字符串变量名> As String
```

```
例：Dim strName As String
```

定长字符串：

```
Dim <字符串变量名> As String * size
```

```
例：Dim empname As String*50（不足补空，太长截取）
```

相关函数：

```
Trim, Rtrim, Ltrim, Left, Right, MID$
```

5. 熟悉各种 If 语句的格式及 Select 语句的格式。

(1) 分支结构。

1) 单分支结构语句，包括两种形式：

a. If e Then

[A 组语句]

End If

b. If e Then [A 组语句]

其流程图如图 1.1 所示。

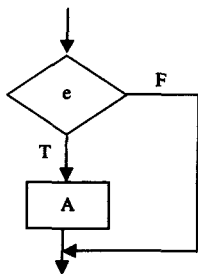


图 1.1 单分支结构语句流程图

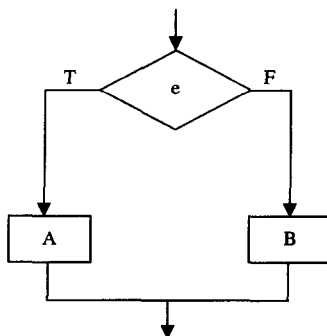


图 1.2 双分支结构流程图

2) 双分支结构语句，包括两种形式：

a. If e Then

[A 组语句]

Else

[B 组语句]

End If

b. If e Then 语句 1 Else 语句 2

其流程图如图 1.2 所示。

3) 多分支结构。

If e1 Then

[A1 组语句]

ElseIf e2 Then

[A2 组语句]

.....

Else

[An+1 组语句]

End If

4) 注意。

①条件：是必要的参数，其结果为 True 或 False。

②ElseIf 是一个整体，中间没有空格。

③在块形式中，If 块必须以一个 End If 语句结束。Else 和 ElseIf 子句都是可选的。

在 If 块中，可以放置任意多个 ElseIf 子句，但是都必须在 Else 子句之前。

(2) Select Case 结构语句。

```

Select Case e
  Case c1
    A 组语句
  Case c2
    B 组语句
    .....
  Case Else
    n 组语句
End Select

```

c1,c2 情况表达式, 可以为以下几种形式:

- ①常量 and 表达式, 如: Case 6 和 Case "6"
- ②表达式 To 表达式, 如: Case 1 To 9
- ③Is <关系运算符><表达式>, 如: Case is <10
- ④多个情况表达式值, 中间用逗号(,)进行分隔。如: Case 1,3,5,7,9

6. 熟悉各种循环语句的格式和执行过程。

(1) For 循环语句。

```

For v=e1 To e2 [Step e3 ]
.....
[Exit For]
.....
Next [v]

```

当知道确定循环次数的时候, 使用上述结构:

①当循环控制变量在给定范围内, 则执行循环体内的语句块, 否则跳出循环, 执行 Next 后的语句。

②当执行到 Next 时, 循环控制变量自加步长, 然后跳回第一步, 重新判断循环控制变量是否在给定范围内。

(2) Do...Loop 循环语句。

用 Do 循环执行某一语句块, 且重复次数不定。

Do...Loop 语句有 4 种形式, 但每种形式都会对循环条件进行判断以决定是否继续执行循环。循环条件必须是一个值为 True 或 False 的表达式。

```

Do While e
...
[Exit Do]
...
Loop
Do
...
[Exit Do]
...
Loop While e
Do Until e
...
[Exit Do]

```

```

...
Loop
Do
    ...
    [Exit Do]
    ...
Loop Until e

```

首先测试循环条件：如果循环条件为 False 或零，则跳过后面的循环语句序列；如果循环条件为 True 或非零，则 Visual Basic 将进入循环体执行语句，执行语句后退回到 Do While 语句再测试循环条件。

(3) 多重循环语句。

多重循环语句也称为循环语句的嵌套，就是在一个循环体内又出现另外的循环语句。在进行嵌套的时候必须注意：

①内、外循环的循环变量不能同名。

②内循环必须完全处于外循环之中，不能出现任何交叉。

另外，循环体如果出现在 If 语句、Select 等语句中时，则也必须作为一个整体出现在条件语句的语句块中。

7. 了解 InputBox 函数和 MsgBox 函数的基本格式。

(1) MsgBox 函数。

对话框中显示一条消息，等待用户单击按钮，并返回用户所单击的按钮类型。

语法

变量% = MsgBox(消息内容[,对话框类型] [,标题])

当对话框只显示单个按钮时，通常采用如下语法格式。

MsgBox 消息内容[,对话框类型] [,标题]

整型变量：返回用户所单击的按钮类型。

消息内容：在对话框中出现的消息文本。

对话框类型：指定在对话框中出现的按钮类型、图标类型及默认按钮设置，可同时设置 3 组参数，它们之间以“+”相连。

如：Amyexit=MsgBox("您确实想退出吗?", VbExclamation+VbYesNo, "退出")

(2) InputBox 函数。

在对话框中显示提示信息，等待用户输入文本或选择按钮，并返回文本框的内容。

语法

变量\$ = InputBox(消息内容[,标题栏内容] [,默认值])

消息内容：对话框的提示文本。当提示文本超过一行时，可用 chr(13)+chr(10) 来分隔。

当用户单击“确定”按钮时，字符串变量返回文本框中的内容；如果单击“取消”按钮，则返回一个零长度的字符串。

8. Print 语句的用法。

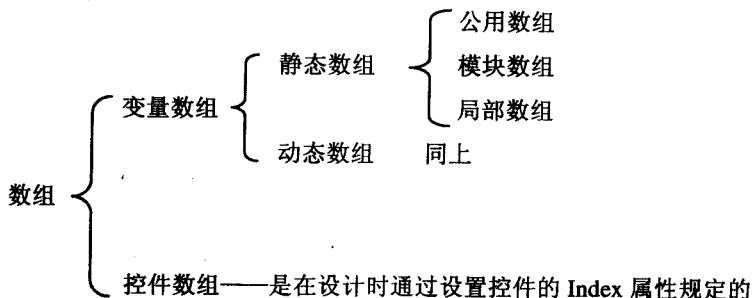
9. 固定数组及动态数组的定义及使用方法。

(1) 数组的概念。

数组是若干具有相同类型的数据的集合。数组用统一的名称来标识；数组中的每一数据称为数组元素，通过数组名和该数据在数组中的序号来标识。如 10 个人的工资，可用 $Gz(0), Gz(1), \dots, Gz(9)$ 来表示。

(2) 数据类型。

Visual Basic 中有如下的数据类型。



(3) 数组的声明与访问。

1) 静态数组。

静态数组有一维和多维之分。

一维数组

格式：{Dim|Static|Public} 数组名[下界 to]上界 As 数据类型，其中：

①上界不得超过 Long 数据类型的范围。

②可同时声明多个数组，各数组之间用逗号分隔。

③当数据类型为 Variant 时，各个元素能够包含不同种类的数据（对象、字符串、数值等）。

例：Dim Sums(20) As Double '21 个元素
Public Counters(2 to 14) As Integer '13 个元素
Static A(3) As Integer, b1(4) As String '同时声明两个数组

一维数组的访问：一维数组的访问有 4 种方式。

①赋值方法：通过“数组名（索引值）”这样的方式对每一个数组元素进行访问。

例：A1(0)=12.1:A1(1)=3:A1(3)=123.2

②通过循环语句：有 for...next 和 for each...next 两种格式。

例：Dim Counters(2 to 14) as Integer
for i=2 to 14
Counters(i)=i+1
Next i

③输入函数：通过单个或循环调用 InputBox 函数，用键盘从对话框输入数组的值。

④使用文件的方法。

多维数组

格式: {Dim|Static|Public} 数组名 ([下界 to]上界, [下界 to]上界, ...) As 数据类型
其中:

- ①可同时声明多个数组, 各数组之间用逗号分隔。
- ②在 Visual Basic 中, 最大数组维数可达 60 维。
- ③其他与一维数组相同。

多维数组的访问: 同一维数组一样, 可“单独”访问, 也可利用多层循环进行访问。

例: Dim I As Integer, J As Integer
Static MstrisS(1 To 10, 1 To 10) As Double
For i=1 to 10
 For j=1 to 10
 MatrixA(i, j)=i*10+j
 Next j
Next i

2) 动态数组。

数组在使用前必须先声明, 那么到底声明多大才合适呢? 太大会占用较多的内存, 太小不够用, 所以希望能够在运行时动态地改变数组大小。动态数组就可以在任何时候改变大小。在 Visual Basic 中, 动态数组可以灵活、方便、有效地管理内存。使用动态数组需要经过两个步骤。

①动态数组的声明。

格式: {Dim|Static|Public|Private} 数组名 ()

注意: 这要的声明对一维、多维数组都适用。

②数组实际维数、元素个数。

格式: ReDim [Preserve]变量名 ([下界 to]上界) As 数据类型

说明:

- a. 可同时给多个数组进行重分配, 各数组之间用逗号分隔。
- b. 用 Preserve 参数, 只能定义数组最末维的大小, 保留除最末维之外的数组的其他数据。
- c. 允许重复定义数组的元素个数, 但不允许重复定义数组的维数。

例: Dim MyArray() As Integer '声明动态数组。
 Redim MyArray(5) '分配 5 个元素。
 For i=1 To 5
 MyArray(i) =i
 Next i
 Redim MyArray(10) '大小重定为 10 个元素。
 For I=0 To 10
 MyArray(i)=i
 Next i

10. 自定义类型。

自定义类型通常有两类, 一类叫自定义数据类型, 另一类叫自定义默认类型。用 Type

语句可将固有类型的数据组合成一个有机整体,形成新的数据类型,这种数据类型称为自定义数据类型,它的值称为“记录”。用 DefType 语句定义的变量类型称为默认数据。

(1) 自定义数据类型。

格式: Type 数据类型名

```

    元素名 1 As 类型
    元素名 2 As 类型
    ...
    元素名 n As 类型
End Type
```

例: 在全局模块定义 typedemo 数据类型如下。

```

Type typedemo
    Num1 As Double
    Num2 As Integer
    Var1 As String*9
    Var2 As String*8
End Type
```

一旦说明了 typedemo 后,便可在各窗体、模块、过程中引用了,如:

```

Sub Form_Click()
    Static Genvar As typedemo
    Genvar.var1="Microsoft"
    Genvar.var2="12345678"
    For i=1 to 9
        Genvar.num2=Genvar.num2+i
    Next i
End sub
```

(2) 自定义默认类型。

格式: Deftype 字母范围

其中: Type 可以是所有固有数据类型。

例: DefInt A~C 表示所有以 A、B、C 字母开头的变量皆为整形变量。

DefInt L~O 表示所有以 L、M、N、O 字母开头的变量皆为字符串变量。

Deftype 语句只影响它所定义的那层。类型定义符(%, &, #, !) Deftype 语句的优先级高。

```

例: DefInt A~Z
    Dim Testvar As Double
    ...
    Sub Demo()
    ...
    String Var#="Microsoft"
    ...
    End Sub
```

11. 枚举。

枚举类型也可以说是一种自定义类型,所谓枚举类型是指通过声明将一系列有关的