

第2章

程序设计基础

本章导读

- 数据类型
- 常量与变量
- 表达式
- 常用内部函数
- 程序代码编写规则

作为一门程序设计语言,其中两个重要的方面便是数据及程序控制。数据是程序的必要组成部分,也是程序处理的对象;而程序控制则是对程序运行流程的控制。本章主要介绍程序中的数据及运算,包括数据类型、常量、变量、表达式和函数等。

2.1 数据类型

在使用 VB 程序处理问题时,会遇到各种不同类型的数据。例如,一个人的姓名是由一串字符组成的,成绩、年龄和工资数都是数值,而是否大学毕业则是一个逻辑值,等等。为此,VB 定义了多种数据类型(见表 2.1)。

表 2.1 VB 的基本数据类型

数 据	关 键 字	占用字节数	类 型 符	范 围
整型	Integer	2	%	−32768~32767
长整型	Long	4	&	−2147483648~2147483647
单精度型	Single	4	!	±1.4E−45~±3.40E38
双精度型	Double	8	#	±4.94D−324~±1.79D308
货币型	Currency	8	@	
字节型	Byte	1		0~255
字符型	String	字符串长度	\$	
布尔型	Boolean	2		True 或 False
日期型	Date	8		1/1/100~12/31/9999
对象型	Object	4		任何对象引用
变体型	Variant	按需分配		

VB 提供的数据类型主要有数值型、字符型、布尔型、日期型、变体型和对象型。数值型数据分为整数、浮点数、字节型数和货币型数。其中整数又分为整型数和长整型数,浮点数(也称实型数或实数)分为单精度型数和双精度型数。

不同类型的数据,所占的存储空间不一样,因此选择使用合适的数据类型,可以节省存储空间和提高运行速度。

1. 数值型数据

(1) 整型(Integer)和长整型(Long)。整型和长整型用于保存整数,整数运算速度快、精确,但表示数的范围小。

整型数在计算机中用 2 字节存储,可表示的数值范围为−32768~32767,当使用的数大于 32767 或小于−32768 时,如 33000,−32800 等,程序运行时就会产生“溢出”错误而中断。这时,应采用长整型 Long 甚至采用单精度或双精度型。

长整型数在计算机中用 4 字节存储,可表示的数值范围为−2147483648~2147483647。长整型数可用来表示比较大的整数。

(2) 字节型(Byte)。字节型数可以表示无符号的整数,范围为 0~255,主要用于存储二进制数。

(3) 单精度型(Single)和双精度型(Double)。单精度型数和双精度型数是带小数部分的数,表示的数的范围大,但运算时可能有误差,且运算速度慢。

单精度数最多可以表示 7 位有效数字,小数点可位于这些数字的任何位置。单精度数可用小数形式和指数形式来表示。小数形式是数学中常用的形式如 7.8, -0.514 等;指数形式采用 10 的整数次幂表示数,以 E(或 e)表示底数 10,如 6.53E8(6.53×10^8), 9.273E-14(9.273×10^{-14})等。

双精度数最多可以表示 15 位有效数字,小数点可位于这些数字的任何位置。双精度数也有小数和指数两种表示形式,其指数形式中用 D(或 d)表示底数 10,如 7.14D23(7.14×10^{23}), -3.736014D-13($-3.736014 \times 10^{-13}$)等。

说明: E(或 e)和 D(或 d)可以作为数的指数符号,它只能出现在数的中间,否则是无效的,如 E, E-5, 9DX 等都是错误的。E 和 D 后面的指数必须是整数。

(4) 货币型(Currency)。货币型数是一种专门为处理货币而设计的数据类型。它用于表示定点数,精确到小数点后 4 位,小数点前最多可有 15 位。

2. 字符型数据

字符型(String)数据又称字符串,用于存放文本型的数据。字符可以包括所有的西文符号和汉字,使用时用双撇号""括起来。例如,"Canton","1+2=?","Good Morning"(\表示空格),"程序设计方法"等都是字符串数据。其中,双撇号称为起、止界限符,它不是字符串的一部分。

字符串中包含的字符个数称为字符串长度。空字符串不含任何字符(即长度为 0),用一对双撇号""表示。在 VB 中,通常把一个汉字作为一个字符来处理,长度为 1。

说明: VB 采用 UniCode(统一编码)字符编码方式。UniCode 是全部用 2 字节表示一个字符的字符集。在这种编码机制下,一个英文字母或一个汉字都看作是一个字符(长度为 1),所占用的存储空间均为 2 字节。

字符串分为变长字符串和定长字符串。变长字符串的长度不固定,随着对字符串变量赋予新的字符串,它的长度可增可减。定长字符串的长度保持不变。

字符串数据可以由任意字符组成,但长度不能超过 65536 个字符(定长字符串)或 20 亿个字符(变长字符串)。

3. 布尔型数据

布尔型(Boolean)又称逻辑型,其数据只有 True(真)和 False(假)两个逻辑值。常用于表示逻辑判断的结果。

当把数值型数据转换为逻辑值时,0 会转换为 False,其他非 0 值转换为 True。把逻辑值转换为数值时,False 转换为 0,True 转换为 -1。

4. 日期型数据

日期型(Date)数据用来表示日期和时间。它采用两个“#”符号把日期和时间的值括起来,就像字符型数据用双撇号括起来一样,如 #08/20/2001#, #2001-08-20#, #08/20/2001 2:55:10 AM#。

5. 变体型数据

变体型(Variant)数据是一种可变的数据类型,可以存放任何类型的数据。当指定变量为 Variant 变量时,不必在数据类型之间转换,VB 会自动完成必要的转换。在程序中不特别说明时,VB 会自动将该变量默认为 Variant 型变量。例如:

```
Temp="99"           '赋值一个字符串
Temp=Temp-2         '转换为数值型运算
Temp=#01/01/2002#   '赋值一个日期
```

上述 Temp 的类型随赋值类型不同而不同,其转换处理是由 VB 自动完成的。

6. 对象型数据

对象型(Object)数据可用来表示程序中的对象。使用时先用 Set 语句给对象赋值,其后才能引用对象。

2.2 常量与变量

在程序中,不同类型的数据既能以常量的形式出现,也能以变量的形式出现。

2.2.1 常量

所谓常量就是在程序运行期间,其值始终保持不变的量。VB 中有两种形式的常量:直接常量和符号常量。

1. 直接常量

直接常量是在程序代码中直接给出的数据。例如:

```
数值常量: 321, -463, -85.32, 12E-7
字符串常量: "Visual Basic", "13.57", "02/01/1998"
逻辑值常量: True, False
日期常量: #09/11/2000#, #Jan 1, 2001#
```

说明:(1) 数值常量可以是直接给出的数(如 123, 45.5 等),也可以是采用指数形式的数(如 1.3E+4, .6D-2 等)。在 VB 中还允许使用八进制数和十六进制数,以 &O 开头的数为八进制数,以 &H 开头的数为十六进制数,例如 &O12, &H4E, &H3F2D。

(2) VB 在判断数值常量类型时有时存在多义性,例如,数值 5.4 可能是单精度类型,也可能是双精度类型或货币类型。在默认情况下,VB 将选择需要内存容量最小的数据类型,故数值 5.4 将被作为单精度数处理。为了指明常量的类型,可以在常量后面加上类型符,如 3.6#(表示双精度数),84.13@(表示货币型数据)等。

(3) 字符串常量用来表示文字信息,如姓名、地址、编号等,其内容必须用英文双撇号(作为起止界限符)括起来,如"北京市"、"Visual Basic 程序设计"等。

要注意的是,VB 系统是通过双撇号来识别字符串常量的,因此双撇号是一个不可缺少的

关键符号。例如,将字符串常量“计算机世界”赋值给变量 s,应写成 s=“计算机世界”,而不能写成 s=计算机世界或 s=“计算机世界”(不能用中文引号替代双撇号)。

2. 符号常量

在程序中往往多次用到某些常量(如圆周率 $\pi=3.14159\cdots$),为避免重复书写该常量,VB 提供了符号常量。也就是说,用一个符号来代替一个常量(如 PI 来代替 3.14159)。这样可以增加程序的可读性和可维护性。

符号常量分为两大类,一类是系统内部定义的符号常量,这类常量用户随时可以使用,例如系统定义的颜色常量 vbBlack(代表黑色,见附录 B),vbRed(代表红色)等。

另一类符号常量是用户用 Const 语句定义的,这类常量必须先声明后才能使用。Const 语句的语法格式如下:

[Public|Private]Const 常量名[As 数据类型]=表达式

功能:将表达式表示的数据值赋给指定的符号常量。

其中:

(1) Public 用于在标准模块中声明符号常量,这些常量可在整个应用程序中使用。

(2) Private 声明所定义的符号常量只能在该过程范围内使用。若省略 Public 和 Private,则默认为 Private。

(3) 常量名的命名规则与变量名相同。为了便于辨认,习惯上,符号常量名采用大写字母表示。

示例:

```
Const PI=3.14159           '定义单精度常量 PI,不指定数据类型时,默认为单精度
Const MAX As Integer=100   '定义常量 MAX,整型数
s=2 * PI * 5               '引用符号常量 PI
```

注意:符号常量有点像变量,但不能像修改变量的值那样来修改符号常量,也不能对符号常量赋以新值。

2.2.2 变量

变量是指在程序运行过程中,其值可以发生变化的量。变量代表内存中指定的存储单元,是程序中数据的临时存放场所。每个变量都有一个名字和数据类型,通过变量名可以引用这个变量,数据类型决定了变量的存储形式。

1. 变量的命名规则

给变量命名时应遵守以下规则:

(1) 变量名必须以字母或汉字开头。

(2) 只能由字母、汉字、数字和下划线组成,不能含有小数点、空格等字符。

(3) 字符个数不得超过 255 个。

(4) 不能使用 VB 的关键字(也称保留字,如语句名、函数名等)作为变量名,因为关键字是 VB 使用的词,具有特定的意义。例如,Print,Sub,End 等都是 VB 的关键字,不能作为变量

名,但 Print_1,Sub1 等可以作为变量名。

(5) VB 不区分变量名中字母的大小写,如 Hello,HELLO,hello 指的是同一个名字。

为变量命名时,最好使用有实际意义、容易记忆的变量名,例如用 average 代表平均数,用 sum 代表总和。在本书的一些实例中,为简单起见,仍用单字母的变量名(如 a,b,c 等)。

2. 变量的声明

变量的声明就是向程序说明要使用的变量,以便系统为其分配内存单元。

用 Dim 语句可以声明变量,其语法格式如下:

```
Dim 变量名[As 数据类型][,变量名[As 数据类型]...]
```

例如:

```
Dim total As Integer           '把 total 定义为整型变量(2 字节)
Dim nam As String              '把 nam 定义为变长字符串变量
Dim addr                       '默认为变体型变量
```

说明:(1)在用 Dim 语句声明一个变量后,VB 系统会自动为该变量赋初值。若变量是数值类型,则初值为 0;若变量为变长字符类型,则初值为空字符串。未定义数据类型的变量,默认为 Variant。

(2)可以使用数据类型符来声明变量类型。例如,前面两个声明语句也可写成:

```
Dim total%
Dim nam$
```

(3)默认情况下,字符串变量是不定长的。声明定长字符串的方法是:

```
Dim 变量名 As String * 长度
```

例如,声明一个长度为 20 的定长字符串 stname,可用以下语句:

```
Dim stname As String * 20
```

如果赋予字符串变量的字符少于 20 个,则用空格将 stname 的不足部分填满;如果赋予字符串的长度超过 20 个,VB 会自动截去超出部分的字符。

(4)除了用 Dim 语句声明变量外,还可以用 Public,Private 或 Static 语句来声明变量,但作用有些差异,详见 7.4 节。

在使用变量之前,采用 Dim,Public,Private 或 Static 语句来预先声明变量,称为显式声明变量。

3. 隐式声明

在 VB 中,允许对使用的变量未进行上述的声明而直接使用,称为隐式声明。此时默认的变量类型为变体类型(Variant)。例如,下列语句:

```
Dim a As Double           '显式声明双精度变量 a
a=23.1234                  'a 占 8 字节,有效数字最多可达 15 个
b=1.2                     'b 没有被显式声明,默认类型为变体型(由系统自动确定数据类型)
c="AB12"                  'c 没有被显式声明,默认类型为变体型(由系统自动确定数据类型)
```

为了使程序具有较好的可读性,并利于程序的调试,一般应对使用的变量进行显式声明。

也可使用 Option Explicit 语句(见 11.4 节)要求所有变量都要显式声明。

4. 变量的两个重要特点

(1) “值可变”:一个变量某个时刻只能保存一个值,当将某个数据赋值给一个变量时,就把变量中原有的值“冲”掉,换成新的值。例如,以下两条语句:

```
a=3
a=8
```

执行第 1 条语句“a=3”时,将 3 赋值给变量 a,再执行第 2 条语句“a=8”时,就把 a 中原有的值 3“冲”掉,换成新值 8。

因此,同一个变量在程序中不同时刻,可以根据需要赋予不同的值。

(2) “取之不尽”:程序中可使用变量进行各种运算。在运算过程中,如果没有改变该变量的值,那么,不管使用变量的值进行多少次运算,其值始终保持不变。例如:

```
x=5
a=3 + x
b=x * x - 4 * x
```

变量 x 在后两条语句中被多次使用,但它的值始终保持为 5,因为变量值被读出后,其值没有被改变。

2.3 表达式

运算符是指定某种运算的操作符号。将常量、变量、函数等用运算符连接起来的运算式称为表达式。单个的常量、变量或函数也可看成简单的表达式。VB 中有 5 类表达式:算术表达式、字符串表达式、日期表达式、关系表达式和逻辑表达式。本节介绍前 3 类,后两类表达式将在 4.1 节介绍。

2.3.1 算术表达式

算术表达式也称数值表达式,由算术运算符、数值型常量、变量、函数及括号组成,其运算结果是一个数值。

VB 有 8 种算术运算符,如表 2.2 所示。

表 2.2 算术运算符

运 算 符	名 称	优 先 级	例 子
^	幂运算	1	a^b
-	取负	2	-a
*,/	乘、除	3	a * b, a/b
\	整除	4	a\b
Mod	求余的模运算	5	a Mod b
+, -	加、减	6	a + b, a - b

同一表达式中若有两个同优先级的运算符,则运算顺序从左到右。有括号时括号内优先。

(1) 幂运算用来计算乘方和方根。例如, 2^5 表示 2 的 5 次方,而 $2^{(1/2)}$ 或 $2^{0.5}$ 是计算 2 的平方根。

(2) / 和 \ 的区别: $1/2=0.5$, $1\backslash 2=0$, 整除号 \ 用于整数除法。在进行整除时,如果参加运算的数含有小数,则将它们四舍五入,使其成为整型数或长整型数,然后再进行运算,其结果截尾成整型数或长整型数。

(3) 模运算符 Mod 用来求余数,其结果为第一个操作数整除第二个操作数所得的余数。求余运算一般用于整数。如果操作数带小数,则首先被四舍五入为整数,然后求余数。运算结果的符号取决于第一个操作数。例如:

9 Mod 7	'结果为 2
16 Mod 25	'结果为 16
25.56 Mod 6.91	'先四舍五入再求余数,结果为 5

(4) 在表达式中乘号不能省略,如 $a * b$ 不能写成 ab (或 $a \cdot b$), $(a+b) * (c+d)$ 不能写成 $(a+b)(c+d)$ 。

(5) 括号不分大、中、小,一律采用圆括号。圆括号可以嵌套使用,即在圆括号的里面再套圆括号,但层次一定要分明,左圆括号和右圆括号要配对。例如,可以把数学式子 $x[x(x+1)+1]$ 写成 VB 表达式 $x * (x * (x+1) + 1)$ 。

以下是一些算术表达式的例子。

平常写法	VB 算术表达式
$5x^{10} + \frac{x}{6} + \sqrt[3]{x}$	$5 * x^{10} + x/6 + x^{(1/3)}$
$(-3)^5 + \frac{4}{ab}$	$(-3)^5 + 4/(a * b)$
$8\sin x^3 - \sin^2 x$	$8 * \sin(x^3) - \sin(x)^2$

【例 2.1】 求算术表达式 $2 + 3.2 * 4 \text{ Mod } 17.52 \backslash 4.32 / 2$ 的值。

根据运算符的优先级,该表达式的计算步骤如下:

- ① 计算乘除,得到 $2 + 12.8 \text{ Mod } 17.52 \backslash 2.16$ 。
- ② 计算整除(\),得到 $2 + 12.8 \text{ Mod } 9$ (17.52 和 2.16 两数先四舍五入为 18 和 2)。
- ③ 求余运算(Mod),得到 $2 + 4$ (12.8 先四舍五入为 13)。
- ④ 求和运算,得到表达式的最后结果为 6。

2.3.2 字符串表达式

字符串表达式由字符串常量、字符串变量、字符串函数和字符串运算符组成。运算符有两个: & 和 +,它们的作用都是将两个字符串连接起来,合并成一个新的字符串。例如:

"计算机" & "软件"	'结果是: "计算机软件"
"Windows" & 98	'结果是: "Windows 98"
"Windows" + 98	'错误。+号只能用于连接两个字符串
"123" + "45"	'结果是: "12345"

使用“+”实现字符串的连接时,要求其前后的两个操作数都必须是字符串,而使用“&”实现字符串连接时,没有这样的要求,因为“&”运算符能自动将其前后的操作数都转换成对应的字符串。

建议使用“&”实现字符串的连接运算,但在输入程序时,变量与运算符“&”之间应加一个空格,如 `a & b` 而不能是 `a&b`。这是因为,符号“&”还是长整型的类型符,如果变量与符号“&”接在一起(如 `a&`),VB 先把它作为一个长整型变量处理,因而就会出现语法错误。

2.3.3 日期表达式

日期表达式由运算符(+或-)、算术表达式、日期型常量、日期型变量和函数组成。有以下3种运算方式:

(1) 两个日期型数据相减,其结果是一个数值型数据(相差的天数)。例如:

```
# 8/8/2001 # - # 6/3/2001 #           '结果为: 66
```

(2) 日期型数据加上天数,其结果为一个日期型数据。例如:

```
# 12/1/2000 # + 31                     '结果为: # 01/01/2001 #
```

(3) 日期型数据减去天数,其结果为一个日期型数据。例如:

```
# 12/1/2000 # - 32                     '结果为: # 10/30/2000 #
```

2.4 常用内部函数

函数是一段程序,完成一种特定的运算。VB 中有两类函数:内部函数和用户自定义函数。用户自定义函数就是用户根据需要定义的函数,也就是后面所要介绍的 Function 过程。

内部函数也称标准函数,是由 VB 系统提供的。这些内部函数使用非常方便,用户不必了解函数内部的处理过程,只需给出函数名和适当的参数,就能得到它的函数值。如要计算 x 的平方根,只要写出:

$$y = \text{Sqr}(x)$$

其中, Sqr 是内部函数名, x 为参数。运行时,该语句调用内部函数 Sqr 来求 x 的平方根,其计算结果由系统返回作为 Sqr 的函数值。

VB 的内部函数大体上分为四大类:数学函数、字符串函数、日期/时间函数和转换函数。

说明:本节介绍的函数较多,读者不妨先粗略看过,以后用到某函数时,再回过头来仔细体会它的功能和用法。

2.4.1 数学函数

在数值计算中,经常会遇到一些常用算术函数的计算,如 $\sin x$, $\cos x$, $\sqrt{x}$, 取整数等,如果用到这些算术函数时,都要由使用者自己编写计算程序,那将是十分繁琐的工作。为此,VB 提供了一批算术函数,在程序中要使用某个函数时,只要调用该函数就可以了。

VB 提供的 13 种常用数学函数如表 2.3 所示。

使用数学函数的几点说明:

(1) 三角函数的自变量单位是弧度,如 $\text{Sin}47^\circ$ 应写成 $\text{Sin}(47 * 3.14159/180)$ 。

(2) 要区分取整函数 Int()、Fix() 和 Round() 的异同。

表 2.3 常用数学函数

函 数	返回值类型	功 能	例 子	结 果
Abs(x)	与 x 相同	x 的绝对值	Abs(−4.6)	4.6
Sqr(x)	Double	x 的平方根	Sqr(9)	3
Sin(x)	Double	x 的正弦值	Sin(30 * 3.14/180)	0.499...
Cos(x)	Double	x 的余弦值	Cos(60 * 3.14/180)	0.500...
Tan(x)	Double	x 的正切值	Tan(60 * 3.14/180)	1.729...
Atn(x)	Double	x 的反正切	4 * Atn(1)	3.14159...
Exp(x)	Double	e(自然对数的底)的幂值	Exp(x)	e ^x
Log(x)	Double	x 的自然对数值	Log(x)/Log(10)	Log ₁₀ x
Int(x)	Double	取不大于 x 的最大整数	Int(99.8) Int(−99.8)	99 −100
Fix(x)	Double	取 x 的整数部分	Fix(99.8) Fix(−99.8)	99 −99
Round(x[, n])	Double	对 x 四舍五入,保留的小数位数由 n 指定	Round(1.35,1) Round(1.236,2)	1.4 1.24
Sgn(x)	Integer	取 x 的符号	Sgn(5) Sgn(0) Sgn(−5)	1 0 −1

(3) 函数 Int(x)是求小于或等于 x 的最大整数。例如,Int(2)=2,Int(−2.5)=−3,Int(−3)=−3,也就是说,当 x≥0 时就直接舍去小数,若 x<0 则取小于或等于 x 的第一个负整数。

Int 函数有两种常用用法:

① 对数值四舍五入,其作用与函数 Round(x,n)相同。例如,对一个正数 x 舍去小数位时进行四舍五入,可采用如下式子:

$$\text{Int}(x+0.5)$$

当 x=9.4 时,Int(9.4+0.5)=9。

当 x=9.5 时,Int(9.5+0.5)=10。

② 求余数。例如求一个正整数 x 整除以 k 所得的余数,可以采用 $x-k * \text{Int}(x/k)$,其作用与 $x \text{ Mod } k$ (或 $x-k * \text{Fix}(x/k)$) 相同。

【例 2.2】 已知 a=6,b=9,计算 $c=\sqrt{a^2+b^2}$ 。

编写的窗体单击事件过程代码如下:

```
Private Sub Form_Click()  
    Dim a As Single, b As Single, c As Single
```

```
a=6
b=9
c=Sqr(a * a + b * b)
Print "c="; c
```

End Sub

运行程序后单击窗体,输出结果如下:

```
c=10.81665
```

2.4.2 字符串函数

在计算机的各种应用中,有大量的文字处理操作,如字符串的查找、比较、截取等。为此,VB 提供了一批用于字符串处理的函数。表 2.4 列出了常用的字符串函数。

表 2.4 字符串函数

函 数	返回值类型	功 能	例 子	结 果
Len(字符串)	Integer	字符串长度	Len("ABCD")	4
Left(字符串,n)	String	取左边 n 个字符	Left("ABCD",3)	"ABC"
Right(字符串,n)	String	取右边 n 个字符	Right("ABCD",3)	"BCD"
Mid(字符串,p[,n])	String	从第 p 个开始取 n 个字符	Mid("ABCDE",2,3)	"BCD"
Instr([f,]字符串 1,字符串 2[,k])	Integer	求字符串 2 在字符串 1 中出现的起始位置	Instr("ABabc","ab")	3
String(n,字符)	String	生成 n 个字符	String(4,"*")	"****"
Space(n)	String	生成 n 个空格	Space(5)	5 个空格
Ltrim(字符串)	String	去掉左边空格	Ltrim(" AB")	"AB"
Rtrim(字符串)	String	去掉右边空格	Rtrim("AB ")	"AB"
Trim(字符串)	String	去掉左、右边空格	Trim(" AB ")	"AB"
Lcase(字符串)	String	转成小写	Lcase("Abab")	"abab"
Ucase(字符串)	String	转成大写	Ucase("Abab")	"ABAB"

使用字符串函数的几点说明:

(1) 在函数 Mid 中,若省略 n,则得到的是从 p 开始的往后所有字符,如Mid("ABCDE",2)的结果为"BCDE"。

(2) 在字符串处理中,还常用一个插入字符串语句 Mid,该语句格式如下:

```
Mid(字符串,p[,n])=子字符串
```

该语句用“子字符串”替换“字符串”中从 p 开始的与“子字符串”等长的一串字符。例如,假设 s="ABCDE",执行语句 Mid(s,3)="99"后,s 的值为"AB99E"。

若带参数 n,则用“子字符串”左起 n 个字符来替换“字符串”中从 p 开始的 n 个字符。

(3) 在函数 Instr 中,f 和 k 均为可选参数,f 表示开始搜索的位置(默认值为 1),k 表示比较方式。若 k 为 0(默认),表示区分大小写;若 k 为 1,则不区分大小写。例如,Instr(3,

"A12a34A56","A")的结果为 7,而 Instr(3,"A12a34A56","A",1)的结果为 4。

(4) 在函数 String 中,字符也可以用 ASCII 代码(见附录 A)来表示,如 String(6,42)与 String(6,"*")作用相同。

(5) 表 2.4 及表 2.6 所列出的函数中,凡返回值为字符串(String)的函数,其函数名的尾部也可加入"\$",如 Left("ABC",1)也可写为 Left\$("ABC",1)。

【例 2.3】 使用字符串函数示例。从字符串中取出头部、尾部各一个字符,连接后组成一个新字符串并显示出来。例如,如果字符串为"computer",处理后的新字符串为"cr"。

编写的窗体单击事件过程代码如下:

```

Private Sub From_Click()
    Dim x As String,a As String,b As String,y As String
    x = "computer"           '假设该字符串 x 内容为"computer"
    a = Left(x, 1)           '取左边一个字符
    b = Right(x, 1)          '取右边一个字符
    y = a & b                 '连接起来
    Print "新组成的字符串: " & y '显示处理的结果
End Sub

```

程序运行后单击窗体,输出结果是:

新组成的字符串: cr

2.4.3 日期/时间函数

日期/时间函数用于进行日期和时间处理。表 2.5 列出了常用的日期/时间函数。

表 2.5 日期/时间函数

函 数	返回值类型	功 能	例 子	结 果
Date	Date	返回系统日期	Date	示例:11/03/2002
Time	Date	返回系统时间	Time	示例:7:03:28
Now	Date	返回系统日期和时间	Now	示例:11/03/2002 7:03:28
Day(日期)	Integer	返回日数	Day(# 2002/9/24 #)	24
Month(日期)	Integer	返回月份数	Month(# 2002/9/24 #)	9
Year(日期)	Integer	返回年度数	Year(# 2002/9/24 #)	2002
Weekday(日期)	Integer	返回星期几	Weekday(# 2002/9/24 #)	3
Hour(时间)	Integer	返回小时数	Hour(# 8:3:28 PM #)	20
Minute(时间)	Integer	返回分钟数	Minute(# 8:3:28 PM #)	3
Second(时间)	Integer	返回秒数	Second(# 8:3:28 PM #)	28

说明:函数 Weekday 返回值 1~7,依次表示星期日到星期六。

2.4.4 类型转换函数

类型转换函数用于数据类型的转换。表 2.6 列出常用的类型转换函数。

表 2.6 类型转换函数

函 数	返回值类型	功 能	例 子	结 果
Val(x)	Double	将数字字符串 x 转换为数值	2+Val("12")	14
Str(x)	String	将数值转换为字符串,字符串首位表示符号	Str(5)	"␣ 5"
Asc(x)	Integer	求字符串中首字符的 ASCII 码	Asc("AB")	65
Chr(x)	String	将 x(ASCII 码值)转换为字符	Chr(65)	"A"
Cint(x)	Integer	将 x 转换为整型数,小数部分四舍五入	Cint(1234.57)	1235
Clng(x)	Long	将 x 转换为长整型数,小数部分四舍五入	Clng(325.3)	325
Csng(x)	Single	将 x 舍入为单精度数	Csng(56.5421117)	56.54211
Cdbl(x)	Double	将 x 转换为双精度数	Cdbl(1234.5678)	1234.5678
Ccur(x)	Currency	把 x 转换为货币型数,小数部分最多保留 4 位且自动四舍五入	Ccur(876.43216)	876.4322
Cvar(x)	Variant	把 x 转换为变体型数	Cvar(99 & ".00")	"9900"
Hex(x)	String	把十进制数 x 转换为十六进制数	Hex(31)	"1F"
Oct(x)	String	把十进制数 x 转换为八进制数	Oct(20)	"24"

说明:(1) Val 函数将数字字符串转换为数值型数字时,自动将字符串中的空格去掉,并依据字符串中排列在前面的数值常量来定值,例如:

- Val("12A12")的值为 12 '以前面 12 来定值
- Val("1.2e2")的值为 120 '1.2e 2 是单精度数的指数形式
- Val("bc12")的值为 0 '前面部分不是数值形式,其函数值为 0

可以看出,当字符串中出现数值字符串之外的字符(如字母 A)时,则停止转换,函数返回停止转换前的结果。

(2) Str()函数将数值转换为字符串,字符串首位表示符号(正数用空格表示)。例如,Str(-32)的值为"-32",而 Str(32)的值为"␣ 32"。

(3) Chr()和 Asc()互为反函数,即 Chr(Asc())、Asc(Chr())的结果为原来各自变量的值。例如,Chr(Asc(65))的结果还是 65。

2.4.5 格式输出函数

函数格式:Format(表达式[,格式串])

功能:根据“格式串”规定的格式来输出表达式的值。

其中,“表达式”为要输出的内容,可以是数值、日期/时间型或字符串型表达式;“格式串”表示输出表达式时采用的输出格式,不同数据类型所采用的格式串是不同的。

数值类格式串的常用符号及其含义如表 2.7 所示。

表 2.7 数值类格式串的常用符号及其含义

符 号	功 能
#	数字占位符,显示一位数字。如 121.5 采用格式"###",显示为 122(后一位四舍五入)
0	数字占位符,前、后会补足 0。如 121.5 采用格式"000.00",显示为 121.50
.	小数点
%	百分比符号
,	千位分隔符
E-, E+	科学计数法格式
-, +, \$	负、正号及美元符号,可以原样显示

对于符号 0 和 #,若数值的小数位数多于格式串的位数,按四舍五入处理。

以下是几个简单的示例:

```
Print Format(12345.6, "##,###.##")
Print Format(12345.6, "0000000")
Print Format(12345.6, "$####,#.00")
Print Format(12345.67, "+#####,#.##") '逗号可放在小数点左边的任何位置(占位符的中间)
Print Format(123.45, "0.000E+00")
```

输出结果分别是:

```
12,345.6
0012346
$12,345.60
+12,345.7
1.235E+02
```

对于日期/时间型的格式串符号和字符串型的格式串符号,请参考 VB 的帮助文件,这里不再做介绍。

2.4.6 随机函数

在编写程序时,有时需要产生一定范围内的随机数,这就要用到随机函数和随机函数初始化语句。

1. 随机函数 Rnd

语法格式:Rnd [(x)]

该函数产生一个介于 0 和 1 之间的单精度随机数,一般省略随机函数的参数 x 和圆括号。所谓随机数是人们不能预先估计到的数。

通常把 Rnd 函数与 Int 函数配合使用,例如 Int(4 * Rnd+1)可以产生 1~4 范围内(含 1 和 4)的随机整数,也就是说,该表达式的值可以是 1,2,3 或 4,这由 VB 运行时随机给定。

要生成[a,b]区间范围内的随机整数,可以使用公式:

$$\text{Int}((b-a+1) * \text{Rnd}+a)$$

2. 随机函数初始化语句 Randomize

语法格式: Randomize [n]

该语句用于初始化随机数生成器,一般可省略参数 n。

默认情况下,每次运行一个应用程序,Rnd 都会产生相同序列的随机数。Randomize 语句可以使随机函数 Rnd 产生不同序列的随机数,因此一般在使用 Rnd 之前,都要先执行 Randomize 语句。

【例 2.4】 通过随机函数产生两个两位正整数,求这两个数之和并显示出来。

编写的窗体单击事件过程代码如下:

Private Sub Form_Click()

Dim a As Integer, b As Integer, c As Integer

Randomize

'初始化随机数发生器

a=Int(90 * Rnd + 10)

'产生[10,99]区间内的随机整数

b=Int(90 * Rnd + 10)

c=a + b

'求两数之和

Print "产生的两个随机数: "; a, b

Print "和数: "; c

End Sub

运行程序后单击窗体,输出结果是:

产生的两个随机数: 56 71

和数: 127

单击工具栏上的“结束”按钮,可结束程序的运行。再次运行程序后单击窗体,可得到另一组输出结果:

产生的两个随机数: 18 37

和数: 55

2.5 编码规则及结构化程序设计

VB 程序由一系列语句组成。语句是执行具体操作的指令,它主要包括 VB 关键字(如 Print、Dim 等)、属性、表达式、有效符号等。这些语句元素在构成程序时必须遵守一定的规则。

2.5.1 编码规则

(1) 通常一行写一条语句。在同一行中也可以书写多条语句,语句之间用冒号“:”分隔。例如:

```
sum=sum+x : count=count+1
```

VB 规定,一行允许多达 1 023 个字符。

(2) 有时一条语句很长,一行写不下,可使用续行符(一个空格后面跟随一个下划线“_”),

将长语句分成多行。例如：

```
Print Text1. Text &. Text2. Text &. Text3. Text &. Text4. Text &_
&. Left(Text5. Text,3)
```

但要注意,续行符后面不能加注释,也不能将 VB 关键字或字符串分隔在两行。

(3) 不区分大小写字母。VB 不区分应用程序代码字母的大小写,用户可以随意使用大小写字母编写代码。为了便于阅读,VB 会自动将代码中关键字的首字母转换为大写,其余字母转换为小写。

(4) 各关键字之间,关键字和变量名、常量名、过程名之间一定要用空格分隔。

(5) 使用左缩进格式。为便于程序阅读和维护,建议采用左缩进格式来体现代码的逻辑结构和嵌套关系。

【例 2.5】 程序中使用左缩进和空格分隔示例。

```
Private Sub Form_Click()      '用空格分隔各关键字
    x=4                        '左缩进 2 个字符
    If x<0 Then                '通过按表格 Tab 键或加入空格符可以实现左缩进效果
        Print "x<0"           '通过左缩进来反映各语句的嵌套关系
    Else
        Print "x>=0"
    End If
End Sub
```

(6) 严格按照 VB 规定的格式和符号编写程序。程序中除字符串常量及注释内容外,语句中使用的逗号、分号、引号、括号(如,、;、“”、<、>等)等都是英文状态下的半角符号,不能使用中文标点符号(如,、;、“”、<、>等)和全角符号(如/、+、=等)。

不少同学习惯在中文状态下录入程序,而英文标点符号与中文标点符号在外观上很相似,所以许多错误都是由此而引起的。

2.5.2 格式约定

任何一个程序设计语言都要满足一定的语法要求。

为了便于说明,本书对语句、方法及函数的语法格式中的符号将采用统一约定,专用符号如下：

- (1) []表示方括号内的内容是可选项,用户可根据需要选用或不选用。书写代码时方括号本身不要写入。
- (2) {}表示多项中之一项。
- (3) |(竖线)用来分隔多个选择项,表示选择其中之一。
- (4) ,… 表示同类项目的重复出现。
- (5) …省略了在叙述中不涉及的部分。

2.5.3 结构化程序设计

从下一章开始,我们将要学习顺序、选择和循环 3 种程序结构设计方法。为了使读者对这

3 种程序结构有一个初步认识,下面简单介绍结构化程序设计的基本概念。

面向对象编程尽管采用事件驱动的机制,调用功能相对简单的过程,但在设计过程的程序代码时,仍然要用到结构化程序设计方法,对过程的流程进行控制。

按照结构化程序设计的观点,任何复杂的程序都是由若干种简单的基本结构组成的,这些基本结构就是顺序结构、选择结构和循环结构。

图 2.1 是这三种基本结构的流程图。这三种基本结构具有单入口,单出口的特点。

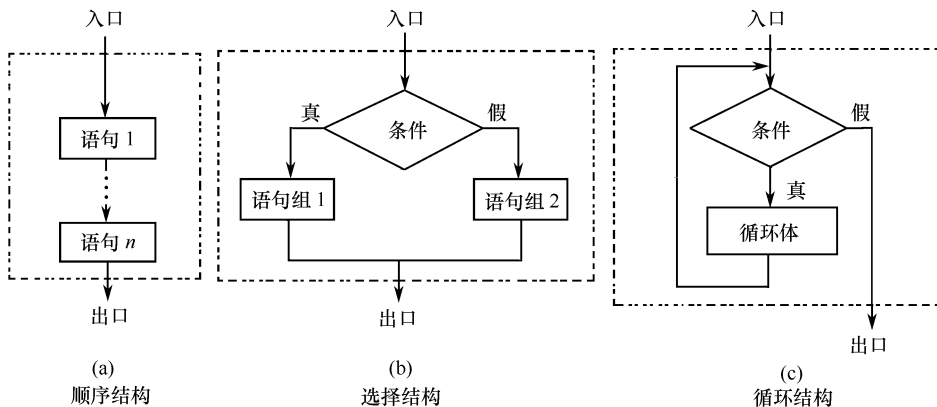


图 2.1 三种基本控制结构

顺序结构由一串按顺序排列的语句组成,如图 2.1(a)所示。运行时,按语句出现的先后次序执行,如从语句 1 顺序执行到语句 n ,程序的流程是按照一个方向进行的。顺序结构是任何程序的主体基本结构,即使在选择结构和循环结构中也常以顺序结构作为其子结构。

选择结构(又称分支结构),如图 2.1(b)所示,程序的流程出现一个或多个分支。通过“条件”判断后分支,条件成立(值为真)时执行语句组 1(一条或多条语句),否则(条件不成立,值为假)执行语句组 2。

循环结构如图 2.1(c)所示,程序流程是按一定的条件重复执行一段程序。先判断循环“条件”,若条件成立(值为真)时执行循环体(语句组),条件不成立(值为假)时跳出循环(出口)。

VB 支持结构化的程序设计方法,人们可以用这 3 种基本结构及其组合来描述程序,从而使程序结构清晰,可读性好,也易于查错和修改。

习 题 2

一、单选题

1. 下列(1)各项中,可以作为变量名的是_____;(2)各项中,____不能作为变量名。

(1) A. a1-0 B. Dim C. K6/600 D. CD[1]

(2) A. ABCabc B. A12345 C. 18AB D. Name1

2. 空字符串是指_____。

A. 长度为 0 的字符串 B. 只包含空格字符的字符串
C. 长度为 1 的字符串 D. 不定长的字符串

3. 使用变量 x 存放数据 12345678.987654,应该将 x 声明为_____类型。
A. 单精度(Single) B. 双精度(Double)
C. 长整型(Long) D. 货币型(Currency)
4. 要定义 1 个整型变量和 2 个字符串变量,正确的语句是_____。
A. Dim a As Integer,b,c As String
B. Dim a%,b\$,c As Integer
C. Dim a,b As Integer,c As String
D. Dim a\$,b As Integer,c As String
5. 表达式 $33 \text{ Mod } 17 \setminus 3 * 2$ 的值为_____。
A. 10 B. 1 C. 2 D. 3
6. 如果 a,b,c 的值分别是 3,2,-3,则下列表达式的值是_____。
 $\text{Abs}(b + c) + a * \text{Int}(\text{Rnd} + 3) + \text{Asc}(\text{Chr}(65 + a))$
A. 10 B. 68 C. 69 D. 78
7. 设 m="morning",下列_____表达式的值是"mor"。
A. Mid(m,5,3) B. Left(m,3) C. Right(m,4) D. Mid(m,3,1)
8. 设 A="12345678",则表达式 $\text{Val}(\text{Left}(A, 4) + \text{Mid}(A, 4, 2))$ 的值是_____。
A. 123456 B. 123445 C. 8 D. 6
9. 在下列函数中,_____函数的执行结果与其他 3 个不一样。
A. String(3, "5") B. Str(555)
C. Right("5555",3) D. Left("55555",3)
10. 设变量 A 的值为 -2,则_____函数的执行结果与其他 3 个不一样。
A. Val("A") B. Int(A) C. Fix(A) D. -Abs(A)
11. 要在窗体 Form1 的标题栏上显示“统计程序”,可用_____语句。
A. Form1. Name="统计程序" B. Form1. Caption="统计程序"
C. Form1. Caption=统计程序 D. Form1. Caption="统计程序"
12. 设 x=10,y=20,要求生成一个字符串变量 s,其值为"10 * 20 = 200",其中数字由变量 x 和 y 的值或表达式来表示。
A. s = "x * y=" & x * y
B. s = "x" & "*" & y & "=" & x * y
C. s = x & "*" & y & "=" & x * y
D. s = x + "*" + y + "=" + x * y
13. 从字符串变量 S 中取出最后(右边)两个字符,可以采用_____函数。
A. Right(2, S) B. Mid(S, Len(S)-1)
C. Mid(S, 2, 2) D. Right(S, Len(S)-2)
14. 已知 A=Space(1),要产生 3 个空格,可以采用_____。
A. Right(A, 3) B. Space(3 * A)
C. String(3, A) D. 3 * A
15. 要求一个正整数 n 除以 8 所得的余数,不可以采用_____。