

Visual Basic 程序设计基础

3.1 学习要点

1. 数据类型与变量

1) 数据类型

Visual Basic 提供的数据类型主要分为标准数据类型和自定义数据类型两大类。

(1) 标准数据类型(也称基本数据类型)

经常使用的标准数据类型有 Boolean、Integer、Single、String、Long、Double、Date、Variant 等,不同的数据类型具有不同的取值范围。

(2) 自定义数据类型(也称构造数据类型)

自定义数据类型主要包括枚举类型数据和用户自己定义的记录类型数据。

2) 常量与变量

(1) 变量命名规则

变量的命名必须以字母或汉字开头,由字母、汉字、数字或下划线组成,长度小于等于 255 个字符;变量的命名不能使用 Visual Basic 中的关键字;不区分变量名的大小写。

(2) 变量的声明

① 直接使用:如 Dim、Static、Public、Private 等声明语句显式声明为 As type;

② 也可以在变量名后增加说明后缀,如%(Integer)、&(Long)、!(Single)、#(Double)等。

(3) 变量的初值

系统默认数值型变量为零,字符型变量为空(""),对象变量为 Nothing。

(4) 变量的作用域

局部变量作用域:该变量所在的代码模块。

窗体级变量作用域:该变量所在的窗体文件的所有代码模块。

全局变量作用域:该变量所在的整个程序。

(5) 变量的生存周期

静态变量生存周期:整个程序的运行过程。

动态(局部)态变量生存周期:该变量所在的代码模块被调用(运行)时。

(6) 常量

在 Visual Basic 中的常量可分为 3 种类型：直接常量、符号常量以及系统标准常量。

2. 运算符与表达式

1) 运算符

(1) 算术运算符

算术运算符运算规则如表 3-1 所示。

表 3-1 算术运算符运算规则

运算符	名称	功 能	示 例	结 果
+	加	加法运算	5.6+1.8	7.4
-	减	减法运算或表示负数	3.9-2.4	1.5
*	乘	乘法运算	1.5*2.5	3.75
/	除	除法运算,返回浮点数	4.5/9	0.5
^	指数	乘方运算	7^2	49
\	整除	除法运算,返回浮点数的整数部分	11\5	2
Mod	取余	除法运算,返回余数部分	11 Mod 5	1

在算术运算符中值得注意的是 Mod(取余运算符)和\ (整除运算符)。

(2) 字符串运算符

字符串运算符运算规则如表 3-2 所示。

表 3-2 字符串运算符运算规则

运算符	功 能	示 例	结 果
&	连接两个字符串表达式	"Nes" & "cafe"	"Nescafe"
	若表达式中含有非字符串数据,则强制连接	123 & 123	"123123"
+	当其两端都为字符串表达式时进行连接运算	"123"+"123"	"123123"
	当其中含有数值型数据时则进行加法运算	"123"+123	246

(3) 关系运算符

关系运算符用于对两个值进行比较,其运算规则如表 3-3 所示。

表 3-3 关系运算符运算规则

运算符	名 称	示例	结果	运算符	名 称	示例	结果
<	小于	4<6	True	>=	大于或等于	9>=2	True
<=	小于或等于	8<=8	True	=	等于	7=8	False
>	大于	5>7	False	<>	不等于	7<>6	True

在这里要注意的是关系运算符中的=和赋值语句中的=是两个完全不同的概念,前者表示相等,而后者则表示赋值。例如关系表达式 $3+4=7$ 也可以写成 $7=3+4$,但赋值语句 $a=3+4$ 就绝对不能写成 $3+4=a$ 。

(4) 逻辑运算符

逻辑运算符运算规则如表 3-4 所示。

表 3-4 逻辑运算符运算规则

运算符	功 能	示 例	结 果
Not	逻辑非运算	Not (4>7)	True
And	逻辑与运算	5>8 And 8>7	False
Or	逻辑或运算	5>8 Or 8>7	True
Eqv	逻辑等价运算	9>8 Eqv 7>5	True
Imp	逻辑蕴涵运算	8>9 Imp 5>7	True
Xor	逻辑异或运算	5>7 Xor 3>5	False

Not(非)、And(与)和 Or(或)3 个逻辑运算符通常与关系运算符一起使用,例如,可以用如下的语句来描述要查找的学生记录:

```
If 期末考试成绩>90 And 平时成绩>90 Then Print "总评优秀"  
If 性别="女" And (班级="1 班" Or 班级="3 班") Then 体育成绩="免修"
```

读者可以体会一下上述两条语句的含义。

2) 运算符的运算优先顺序

所有运算符的运算优先顺序(级别数字越小,优先级别越高)如表 3-5 所示。

表 3-5 运算符的运算优先顺序

级别	运 算 符	说 明	级别	运 算 符	说 明
1	^		8	<,<=,>,>=、 =,<>	
2	—	取负值运算	9	Not	
3	*,/		10	And	
4	\		11	Or,Xor	
5	Mod		12	Eqv	
6	+,—		13	Imp	
7	+,&	字符串连接运算			

3) 表达式

(1) 组成

表达式由变量、常量、函数、运算符和圆括号组成。

(2) 书写规则

运算符不能相邻。如 a+-b 是错误的。

乘号不能省略。如 x 乘以 y 应写成 x*y。

括号必须成对出现,均使用圆括号。

表达式从左到右在同一基准上书写,无高低、大小。

(3) 不同数据类型的转换

运算结果的数据类型向精度高的数据类型靠。即 Integer<Long<Single<Double<Currency。

(4) 优先级

算术运算符>=字符运算符>关系运算符>逻辑运算

3. 常用标准函数的使用

Visual Basic 语言处理系统向用户提供了丰富的常用标准函数,在 Visual Basic 教材中所罗列的常用标准函数是希望大家熟练掌握使用的最基本部分,在此不再赘述。

4. 顺序结构语句

顺序结构语句是程序设计中使用的最基本语句,在此主要掌握 3 个方面的内容:

(1) 赋值语句

(2) InputBox()函数的使用

(3) MsgBox()函数或过程的使用

5. 选择结构 If 语句

1) 单分支结构的 If 语句的两种格式和使用

(1) 单行形式:

If <表达式>Then <语句>

(2) 多行形式:

If <表达式>Then

<语句块>

End If

其中表达式可以是算术表达式、关系表达式和逻辑表达式,按表达式的值非 0 为 True、0 为 False 来判断。

多行的 If 必须与 End If 配对,语句块可以是多个语句;单行格式不需要 End If 关键字配对,此时的语句是一个语句或用“:”分隔的多个语句。

2) 双分支结构的 If 语句的两种格式和使用

(1) 单行形式:

If <表达式>Then <语句 1>Else <语句 2>

(2) 多行形式:

If <表达式>Then

<语句块 1>

Else

<语句块 2>

End If

3) 多分支结构的 If 语句的格式和使用
语句格式:

```
If <表达式 1> Then
    <语句块 1>
ElseIf <表达式 2> Then
    <语句块 2>
    :
[Else
    <语句块 n+1>]
EndIf
```

当 If 结构内有多个条件为 True 时, Visual Basic 仅执行第一个为 True 的条件后的语句块, 然后跳出 If 结构。

6. 选择结构 Select Case 语句

语句格式:

```
Select Case <变量或表达式>
    Case <表达式列表 1>
        <语句块 1>
    Case <表达式列表 2>
        <语句块 2>
        :
    [Case Else
        <语句块 n+1>]
End Select
```

其中, 需注意<变量或表达式>和<表达式列表 n>的书写以及设计等编程规则。

7. 条件测试函数

条件测试函数有 3 种类型, 其调用格式分别如下:

```
Iif (<条件表达式>, 当条件为 True 时的值, 当条件为 False 时的值)
Choose (<数字类型变量>, 值为 1 的返回值, 值为 2 的返回值... )
Switch (<条件表达式 1>, 条件表达式 1 为 True 时的值
[, <条件表达式 2>, 条件表达式 2 为 True 时的值... ])
```

8. If 语句的嵌套

If 语句的嵌套格式如下:

```
If <条件表达式 1> Then
    :
    If <条件表达式 n> Then
```

```

        :
    End If
    :
End If

```

区分嵌套的层次方法是：每个 End If 与它上面最接近的 If 配对。

9. 循环结构——For 循环结构

For 循环也称计数循环，其格式如下：

```

For <循环变量>=<初值> To <终值> [ Step <步长>]
    <循环体>
Next <循环变量>

```

其中：

(1) 步长可正可负，可为整数，也可为实数。

(2) 循环次数的计算方式为：

$$\text{Int}((\text{终值} - \text{初值}) / \text{步长}) + 1$$

(3) 当循环变量(步长为正)大于终值，则循环终止。

要注意出了循环时，循环控制变量值的问题。例如以下程序段：

```

For i=2 To 13 Step 3          '循环执行次数
    Print i                  '输出 i 的值,分别为: 2    5    8    11
Next i
Print:Print"I=", i          '出了循环输出为: I=14

```

在循环体内对循环控制变量可多次引用。但最好不要对其赋值，否则影响原来的循环控制规律。

10. 循环结构——Do…Loop 循环结构

根据不同组合，Do…Loop 循环结构有如下几种形式：

(1) Do

 <循环体>

Loop

(2) Do While<条件>

 <循环体>

Loop

(3) Do Until<条件>

 <循环体>

Loop

(4) Do

 <循环体>

Loop While<条件>

(5) Do

<循环体>

Loop Until<条件>

Do...Loop 循环语句的功能是：当 While 指定的<条件>为 True 或 Until 指定的<条件>变为 True 之前重复执行<循环体>内的语句。

11. 循环的嵌套

循环体内再含有循环语句称为循环的嵌套或多重循环。Visual Basic 语言处理系统支持选择结构与循环结构的嵌套。对于循环的嵌套,要注意以下事项:

- (1) 内循环变量与外循环变量不能同名。
- (2) 外循环必须完全包含内循环,不能交叉。
- (3) 不能从循环体外转向循环体内,反之则可以。

12. 转移语句——GoTo 语句

形式:

GoTo {标号|行号}

标号是字母开头的字符序列,任何转移到的标号后应有冒号。行号是一个数字序列。

13. Exit 语句

有多种形式的 Exit 语句,如 Exit For、Exit Do、Exit Sub、Exit Function 等,用于退出某种控制结构的执行。

3.2 难点分析

(1) 掌握面向对象的编程语言的一些基本概念,如类、对象等。掌握对象的三要素的作用及使用方法,如给一个对象的属性赋值:对象名.属性名=属性值。使用一个对象的方法:对象名.方法名[参数],这两个格式在 Visual Basic 程序中处处可见,一定要能灵活掌握。

对于对象、属性、事件、消息、事件过程等名词,如果抽象地去理解,也许会有困难。可以用下面的程序实例来加以说明,如图 3-1 所示。

在这个程序实例中,有 3 个对象,分别是一个窗体对象和两个按钮对象。除了在本例中使用到的这些控件对象外,根据程序的需要,在 Visual Basic 中还经常会用到标签控件 Label 对象、文本框控件 TextBox 对象、框架控件 Frame 对象、单选按



图 3-1 Visual Basic 程序实例

钮控件 OptionButton 对象、复选框控件 CheckBox 对象等,这些控件对象在实验教材的不同实验中都使用到,可以在实验练习时熟悉它们的特点和使用方法。

本例中的 3 个对象分别由以下的程序代码控制,这些程序代码表示了 3 个事件过程。

```
Private Sub Command1_Click()           '当按钮 1 被单击时触发的事件过程
    Print "Shanghai"
End Sub
Private Sub Command2_Click()           '当按钮 2 被单击时触发的事件过程
    End
End Sub
Private Sub Form_DblClick()            '当窗体被双击时触发的事件过程
    Print "Beijing"
End Sub
```

每个对象都有其各自的属性。例如在本例中,按钮 1 的 Caption 属性值为“演示”,按钮 2 的 Caption 属性值为“结束”,而窗体 Form1 的 Font(字体)属性值为二号。

当程序运行后,单击“演示”按钮,就得到了一个事件驱动,即按钮 1 对象发生了一个 Click(单击)事件,程序立即启动相应的事件过程,即执行:

```
Private Sub Command1_Click()           '当按钮 1 被单击时触发的事件过程
    Print "Shanghai"
End Sub
```

这段代码的执行结果为向窗体发出了一个消息,即在窗体 Form1 上显示“Shanghai”。

同理,当双击窗体时,发生了 Form_DblClick(双击)事件,立即就会启动另一个事件过程,执行结果是在窗体上显示“Beijing”。而当单击“结束”按钮时,又发生了按钮 2 的 Click 事件,因此,程序执行 End(结束)的事件过程。

(2) 掌握一些常用对象的常用属性、方法和事件的使用,对于特殊的用法一定要注意。例如,要在按钮的 Click 事件中可以直接使用 Print 方法在窗体上输出内容,而若要在窗体的 Load 事件中使用 Print 方法,则首先必须把窗体的 AutoRedraw 属性设置为 True。

(3) 要比较记忆 InputBox() 函数、MsgBox() 函数和 MsgBox() 过程的参数个数及参数排列顺序,使用时不可混淆。

要注意三者的返回值类型。InputBox() 函数返回结果为字符类型,MsgBox() 函数返回结果为数值类型,MsgBox() 过程无返回值。

(4) 对于选择结构,要注意以下问题。

① End If 不能写成 EndIf, ElseIf 不能写成 Else If。

② 在嵌套的 If 语句结构中,End If 和在该语句上面最靠近的 If 是一对。例如:

```
If i>1 Then
    If j>1 Then
        Print i,j
    End If
```

```
End If
```

在上述程序段中,第 2 行的 If 和第 4 行的 End If 是一对内嵌的 If 结构,第 1 行的 If 和第 5 行的 End If 是一对外层的 If 结构。该程序段的含义是:在外层 If 结构中的条件 $i > 1$ 成立时,才会运行内嵌的 If 结构,只有当内嵌条件 $i > 1$ 和外层条件 $j > 1$ 都成立时才会执行语句“Print i,j”。

(5) 对于循环结构,一定要搞清楚 For 循环的执行过程。下面以二重循环为例说明:

```
For i=1 To 5 step 2
    For j=1 To i
        Print j;
    Next j
    Print
Next i
```

在上述程序段中,共执行 3 次以 i 为循环变量的外层循环,i 变量分别取值为 1、3、5,在每次外循环中,都会执行两件事:第一是执行内循环,其循环次数为 i 的值;第二是 Print 语句,即输出空行。在第一次外循环 i 的值为 1 时,内循环共循环 1 次,只输出内容 1;在第二次外循环 i 的值为 3 时,内循环共循环 3 次,输出内容为 1 2 3;在第三次外循环 i 的值为 5 时,内循环共循环 5 次,输出内容为 1 2 3 4 5。因为每次外循环中,执行完内循环后都有空行输出,所以 3 次外循环的输出结果处于不同行,即:

```
1
1 2 3
1 2 3 4 5
```

(6) 对于 Do...Loop 循环的循环次数,举例说明如下。

形式一:

```
Do While m>1
    m=m-1
    Print m
Loop
```

形式三:

```
Do
    m=m-1
    Print m
Loop While m>1
```

形式二:

```
Do Until m<=1
    m=m-1
    Print m
Loop
```

形式四:

```
Do
    m=m-1
    print m
Loop Until m<=1
```

若假设 m 的初始值为 3,则对上面 4 种形式来说,均执行两次循环,m 最后的值均为 1。

若假设 m 的初始值为 1,则对形式一和形式二来说,由于是先判断条件后执行循环体,所以因 m 的初始值为 1 不满足条件,从而一次也不执行循环体,m 最后的值仍为 1;而对形式三和形式四来说,由于是先执行循环体再判断条件,所以无论条件成立与否,都已

经执行了一次循环体,使 m 的值成为 0,此时再判断,因为不满足条件而退出循环体,但 m 的最终值已为 0。

另外要注意的是,若省略了 While 或 Until 条件,即循环结构仅由 Do...Loop 关键字构成,表示无条件循环,这时在循环体内一定要有 Exit Do 语句来退出循环,否则为死循环。

3.3 经典题解

一、单选题

1. 数据类型中的变体型可以包括数值型、日期型、对象型和字符型等数据,此外,它还可以包含 4 个特殊的数据:_____和 Nothing。

- A. Error、Empty、Object
- B. Error、Object、Null
- C. Empty、Error、Null
- D. Empty、Object、Type

答: C

2. 声明一个长度为 10 个字节的字符串变量 mstr,应使用_____。

- A. Dim mstr As 10
- B. Dim mstr As String * 10
- C. Dim mstr As String(10)
- D. Dim mstr(10) As String

答: B

【解析】 对于字符串类型变量,有两种定义方法: **Dim 字符串变量名 As String** 和 **Dim 字符串变量名 As String * 字符数**。前者定义的是不定长的字符串,最多可存放 2MB 个字符;后者定义的字符串的长度由 * 后面的字符数决定。很显然 A、C 选项定义格式不正确,D 项则定义了字符串类型数组 mstr。

3. 代数式 $ab^2 \div (5c+d)$ 的 Visual Basic 表达式是_____。

- A. $a \times b \times b / (5 \times c + d)$
- B. $a * b^b / (5c + d)$
- C. $a * b * b / (5 * c + d)$
- D. $a * b * b / 5 * c + d$

答: C

【解析】 此题要点是要知道各种算术运算符在 Visual Basic 中的表示法。

4. 表达式 "123" & "100" & 200 的值是_____。

- A. 423
- B. 123100200
- C. "123100200"
- D. 123300

答: C

【解析】 & 连接符两旁的操作数不管是字符型还是数字型,进行连接操作前,系统先将操作数转换成字符型,然后再连接。所以该题实际运算的是 "123" & "100" & "200",即 3 个字符串进行连接,结果为 "123100200"。

5. 表达式 "321" & 100 + "200" 的值是_____。

- A. 321300
- B. 621
- C. "321300"
- D. 321100200

答: C