

VisualBasic变量、常数和数据类型及过程概述六 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/137/2021\\_2022\\_VisualBasi\\_c97\\_137584.htm](https://www.100test.com/kao_ti2020/137/2021_2022_VisualBasi_c97_137584.htm) 提供可选参数的缺省值 也可以给可选参数指定缺省值。

在下例中，如果未将可选参数传递到函数过程，则返回一个缺省值。

```
Sub ListText(x As String, Optional y As Integer = 12345) List1.AddItem x List1.AddItem y End Sub Private Sub Command1_Click () strName = "yourname" 未提供第二个参数。 Call ListText (strName) 添加 " yourname " 和 " 12345 "。
```

End Sub 使用不定数量的参数一般说来，过程调用中的参数个数应等于过程说明的参数个数。可用ParamArray 关键字指明，过程将接受任意个数的参数。于是可以这样来编写计算总和的 Sum 函数：

```
Dim x As Integer Dim y As Integer Dim intSum As Integer Sub Sum (ParamArray intNums ()) For Each x In intNums y = y x Next x intSum = y End Sub Private Sub
```

```
Command1_Click () Sum 1, 3, 5, 7, 8 List1.AddItem intSum End
```

Sub 用命名的参数创建简单语句 对许多内建函数、语句和方法，Visual Basic 提供了命名参数方法来快捷传递参数值。对命名参数，通过给命名参数赋值，就可按任意次序提供任意多参数。为此，键入命名参数，其后为冒号、等号和值

(MyArgument := "SomeValue")，可以按任意次序安排这些赋值，它们之间用逗号分开。注意，下例中的参数顺序和所要参数的顺序相反：

```
Function ListText (strName As String, Optional strAddress As String) List1.AddItem strName List2.AddItem
```

```
strAddress End Sub Private Sub Command1_Click () ListText
```

```
strAddress:="12345", strName:="Your Name" End Sub 如果过程有
```

若干不必总要指定的可选参数，则上述内容更为有用。确定对命名参数的支持要确定哪一个函数、语句和方法支持命名参数，用“代码”窗口中的“AutoQuickInfo”功能，检查“对象浏览器”，或者参阅语言参考。使用命名参数时要注意以下几点：在 Visual Basic (VB) 对象库中的对象的方法不支持命名参数。而 Visual Basic for applications (VBA) 对象库中的所有语言关键字都支持命名的参数。在语法中，命名参数是用粗体和斜体字表示的。所有其它参数只用斜体字表示。重点使用命名参数时不能省略所需参数的输入。可以只省略可选参数。对于 Visual Basic (VB) 和 Visual Basic for applications (VBA) 对象库，“对象浏览器”对话框将可选参数用方括号 [ ] 括起来。详细信息请参阅语言参考中的 "ByVal"、"ByRef"、"Optional" 和 "ParamArray"。

控制结构概述 有了控制结构就可控制程序执行的流程。如果未复选控制流语句，程序便从左至右、自顶向下地贯穿这些语句。有些简单程序可以只用单向流程来编写，有些流程可以依靠运算符的优先级来控制，但任何编程语言的效力和用途皆由其通过结构和循环改变语句顺序的能力而得。

判定结构 Visual Basic 过程能够测试条件式，然后根据测试结果执行不同的操作。Visual Basic 支持的判定结构有：1.If...Then 2.If...Then...Else 3.Select Case

If...Then 结构有条件地执行一个或多个语句。单行语法和多行块语法都可以使用：If condition Then statement  
If condition Then statements End If  
condition 通常是比较式，但它可以是任何计算数值的表达式。Visual Basic 将这个值解释为 True 或 False：一个为零的数值为 False，而任何非零数值都被看作 True。若 condition 为 True，则 Visual Basic 执行 Then 关

键字后面的所有 statements。可以使用单行或多行语法有条件地执行一个语句（下面两个例子等价）：  
`If anyDate = Now Then End If` 注意：`If...Then` 的单行格式不用 `End If` 语句。如果 condition 为 True 时要执行多行代码，则必须使用多行块 `If...Then...End If` 语法。  
`If anyDate = Now Then Timer1.Enabled = False End If` 定时器控制失效。  
用 `If...Then...Else` 块定义几个语句块，执行其中一个语句：  
`If condition1 Then [statementblock-1] [Elseif condition2 Then [statementblock-2]] ... [Else [statementblock-n]] End If`  
Visual Basic 首先测试 condition1。如果它为 False，Visual Basic 就测试 condition2，依此类推，直到找到一个为 True 的条件。当它找到一个为 True 的条件时，Visual Basic 就会执行相应的语句块，然后执行 `End If` 后面的代码。作为一个选择，可以包含 `Else` 语句块，如果条件都不是 True，则 Visual Basic 执行 `Else` 语句块。  
100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)