# 什么是 VBScript?

* VBScript 是一种脚本语言
* 脚本语言是一种轻量级的编程语言
* VBScript 是微软的编程语言 Visual Basic 的轻量级的版本

# VBScript 变量

## VBScript 变量

VBScript 变量用于保存值或表达式。

变量可以有一个短的名称，如 x，或一个更具描述性的名称，如 carname。

VBScript 变量名称的规则：

* 必须以字母开头
* 不能包含点号（.）
* 不能超过 255 个字符

在 VBScript 中，所有的变量都与类型 *variant* 相关，可存储不同类型的数据。

## 声明（创建）VBScript 变量

在 VBScript 创建变量通常指"声明"变量。

您可以通过 Dim、Public 或 Private 语句声明 VBScript 变量。如下所示：

Dim x  
Dim carname

现在您已经创建了两个变量。变量的名称是 "x" 和 "carname"。

您也可以在脚本中通过使用它的名称来声明变量。如下所示：

carname="Volvo"

现在您又创建了一个变量。变量的名称是 "carname"。然后，这个做法不是一个好习惯，因为您可能会在脚本中拼错变量名，那样可能会在脚本运行时引起奇怪的结果。

如果您拼错变量名，比如 "carname" 变量错拼为 "carnime"，脚本会自动创建一个名为 "carnime" 的新变量。为了防止脚本这样做，您可以使用 Option Explicit 语句。如果您使用这个语句，就必须使用 dim、public 或 private 语句来声明所有的变量。

把 Option Explicit 语句放置于脚本的顶端，如下所示：

Option Explicit  
Dim carname  
carname=some value

## 为变量赋值

您可以为某个变量赋值，如下所示：

carname="Volvo"  
x=10

变量名是在表达式的左侧，需要赋给变量的值在表达式的右侧。现在变量 "carname" 的值是 "Volvo"，变量 "x" 的值是 "10"。

## 变量的生存期

变量的生存期指的是它可以存在的时长。

当您在一个子程序中声明变量时，变量只能在此程序内进行访问。当退出此程序时，变量也会失效。这样的变量称为本地变量。您可以在不同的子程序中使用名称相同的本地变量，因为每个变量只能在声明它的程序内得到识别。

如果您在子程序以外声明了一个变量，在您的页面上的所有子程序都可以访问它。这类变量的生存期始于它们被声明，止于页面被关闭。

## VBScript 数组变量

数组变量用于在一个单一的变量中存储多个值。

在下面的实例中，声明了一个包含 3 个元素的数组：

Dim names(2)

括号内显示的数字是 2。数组的下标以 0 开始，因此该数组包含 3 个元素。这是容量固定的数组。您可以为数组的每个元素分配数据，如下所示：

names(0)="Tove"  
names(1)="Jani"  
names(2)="Stale"

同样地，通过使用特定数组元素的下标号，您可以取回任何元素的值。如下所示：

mother=names(0)

您可以在一个数组中使用多达 60 个维数。声明多维数组的方法是在括号中用逗号来分隔数字。这里，我们声明了一个包含 5 行 7 列的 2 维数组：

Dim table(4,6)

# VBScript 程序

VBScript 可使用两种程序：

* 子程序
* 函数程序

## VBScript 子程序

子程序：

* 是一系列的语句，被封装在 Sub 和 End Sub 语句内
* 可执行某些操作，但**不会返回**值
* 可带有参数

Sub mysub()  
*some statements*  
End Sub

或者

Sub mysub(argument1,argument2)  
*some statements*  
End Sub

实例

Sub mysub()  
document.write("I was written by a sub procedure")  
End Sub

## VBScript 函数程序

函数程序

* 是一系列的语句，被封装在 Function 和 End Function 语句内
* 可执行某些操作，并**会返回**值
* 可带有通过程序调用来向其传递的参数。
* 如果没有参数，必须带有空的圆括号 ()
* 通过向函数程序名赋值的方式，可使其返回值

Function myfunction()  
*some statements*  
myfunction=*some value*  
End Function

或者

Function myfunction(argument1,argument2)  
*some statements*  
myfunction=*some value*  
End Function

实例

function myfunction()  
myfunction=Date()  
end function

## 调用程序

这个简单的函数程序被调用来计算两个参数的和：

Function myfunction(a,b)  
myfunction=a+b  
End Function  
  
document.write(myfunction(5,9))

函数 "myfunction" 将返回参数 "a" 和参数 "b" 的和。这里返回的是 14。

当您调用程序时，您可以使用 Call 语句，如下所示：

Call MyProc(argument)

或者，您可以省略 Call 语句，如下所示：

MyProc argument

# VBScript 条件语句

## 条件语句

条件语句用于根据不同的情况执行不同的操作。

在 VBScript 中，我们可以使用四种条件语句：

* **If 语句** - 假如您希望在条件为 true 时执行一系列的代码，可以使用这个语句
* **If...Then...Else 语句** - 假如您希望执行两套代码其中之一，可以使用这个语句
* **If...Then...ElseIf 语句** - 假如您希望选择多套代码之一来执行，可以使用这个语句
* **Select Case 语句** - 假如您希望选择多套代码之一来执行，可以使用这个语句

## If...Then...Else

在下面的情况中，您可以使用 If...Then...Else 语句：

* 在条件为 true 时，执行某段代码
* 选择两段代码之一来执行

如果在条件为 true 时只执行**一条**语句，可以把代码写为一行：

If i=10 Then alert("Hello")

在上面的代码中，没有 ..Else.. 语句。我们仅仅让代码在条件为 true 时（当 i=10 时）执行**一项操作**。

如果在条件为 true 时执行**不止一条**语句，那么就必须在一行写一条语句，然后使用关键词 "End If" 来结束这个语句：

If i=10 Then  
alert("Hello")  
i = i+1  
End If

在上面的代码中，同样没有 ..Else.. 语句。我们仅仅让代码在条件为 true 时执行了**多项操作**。

假如您想要在条件为 true 时执行某条语句，并在条件不为 true 时执行另一条语句，就必须添加关键词 "Else"：

实例

i=hour(time)  
If i < 10 Then  
document.write("Good morning!")  
Else  
document.write("Have a nice day!")  
End If

在上面的代码中，当条件为 true 时会执行第一段代码，当条件不成立时执行第二段代码（当 i 大于 10 时）。

## If...Then...ElseIf

如果您想要选择多套代码之一来执行，可以使用 If...Then...ElseIf 语句：

i=hour(time)  
If i = 10 Then  
document.write("Just started...!")  
ElseIf i = 11 Then  
document.write("Hungry!")  
ElseIf i = 12 Then  
document.write("Ah, lunch-time!")  
ElseIf i = 16 Then  
document.write("Time to go home!")  
Else  
document.write("Unknown")  
End If

## Select Case

如果您想要选择多套代码之一来执行，可以使用 "Select Case" 语句：

d=weekday(date)  
Select Case d  
Case 1  
document.write("Sleepy Sunday")  
Case 2  
document.write("Monday again!")  
Case 3  
document.write("Just Tuesday!")  
Case 4  
document.write("Wednesday!")  
Case 5  
document.write("Thursday...")  
Case 6  
document.write("Finally Friday!")  
Case else  
document.write("Super Saturday!!!!")  
End Select

以上代码的工作原理：首先，我们需要一个简单的表达式（常常是一个变量），并且这个表达式会被做一次求值运算。然后，表达式的值会与每个 Case 中的值作比较。如果匹配，被匹配的 Case 所对应的代码会被执行。

# VBScript 循环

## 循环语句

循环语句用于运行相同的代码块指定的次数。Looping statements are used to run the same block of code a specified number of times.

在 VBScript 中，我们可以使用四种循环语句：

* **For...Next 语句**- 运行一段代码指定的次数
* **For Each...Next 语句**- 针对集合中的每个项目或者数组中的每个元素来运行某段代码
* **Do...Loop 语句**- 运行循环，当条件为 true 或者直到条件为 true 时
* **While...Wend 语句**- 不要使用这个语句 - 请使用 Do...Loop 语句代替它

## For...Next 循环

请使用 **For...Next** 语句运行一段代码指定的次数。

**For** 语句规定计数变量（**i**）以及它的初始值和结束值。**Next** 语句会以 1 作为步进值来递增变量（**i**）。

### **Step 关键词**

通过 **Step** 关键词，您可以规定计数变量递增或递减的步进值。

在下面的实例中，计数变量（**i**）每次循环的递增步进值为 2。

For i=2 To 10 Step 2  
some code  
Next

如果要递减计数变量，您就必须使用负的 **Step** 值。并且必须规定小于开始值的结束值。

在下面的实例中，计数变量（**i**）每次循环的递减步进值为 2。

For i=10 To 2 Step -2  
some code  
Next

### **退出 For...Next**

您可以通过 Exit For 关键词退出 For...Next 语句。

For i=1 To 10  
If i=5 Then Exit For  
some code  
Next

## For Each...Next 循环

**For Each...Next** 针对集合中的每个项目或者数组中的每个元素来重复运行某段代码。

## Do...Loop

如果你不知道重复多少次，可以使用 Do...Loop 语句。

Do...Loop 语句重复执行某段代码直到条件是 true 或条件变成 true。

重复执行代码直到条件是 true

您可以使用 While 关键字来检查 Do... Loop 语句的条件。

Do While i>10  
some code  
Loop

如果 **i** 等于 9，上述循环内的代码将终止执行。

Do  
some code  
Loop While i>10

这个循环内的代码将被执行至少一次，即使 **i** 小于 10。

### **重复执行代码直到条件变成 true**

您可以使用 Until 关键字来检查 Do...Loop 语句的条件。

Do Until i=10  
some code  
Loop

如果 **i** 等于 10，上述循环内的代码将终止执行。

Do  
some code  
Loop Until i=10

这个循环内的代码将被执行至少一次，即使 **i** 等于 10。

### **退出 Do...Loop**

您可以通过 Exit Do 关键词退出 Do...Loop 语句。

Do Until i=10  
i=i-1  
If i<10 Then Exit Do  
Loop

这个循环内的代码，只要 **i** 不为 10 且 **i** 大于 10 时都将被执行。

# VBScript 函数

本页列出了所有内建的 VBScript 函数，主要分为以下几类：

|  |  |  |
| --- | --- | --- |
| * [Date/Time 函数](https://www.runoob.com/vbscript/vbscript-ref-functions.html" \l "date) * [Conversion 函数](https://www.runoob.com/vbscript/vbscript-ref-functions.html" \l "conversion) * [Format 函数](https://www.runoob.com/vbscript/vbscript-ref-functions.html" \l "format) | * [Math 函数](https://www.runoob.com/vbscript/vbscript-ref-functions.html" \l "math) * [Array 函数](https://www.runoob.com/vbscript/vbscript-ref-functions.html" \l "array) | * [String 函数](https://www.runoob.com/vbscript/vbscript-ref-functions.html" \l "string) * [其他函数](https://www.runoob.com/vbscript/vbscript-ref-functions.html" \l "other) |

## Date/Time 函数

|  |  |
| --- | --- |
| **函数** | **描述** |
| [CDate](https://www.runoob.com/vbscript/func-cdate.html) | 把有效的日期和时间表达式转换为日期（Date）类型。 |
| [Date](https://www.runoob.com/vbscript/vb-func-date.html) | 返回当前的系统日期。 |
| [DateAdd](https://www.runoob.com/vbscript/vb-func-dateadd.html) | 返回已添加指定时间间隔的日期。 |
| [DateDiff](https://www.runoob.com/vbscript/vb-func-datediff.html) | 返回两个日期之间的时间间隔数。 |
| [DatePart](https://www.runoob.com/vbscript/vb-func-datepart.html) | 返回给定日期的指定部分。 |
| [DateSerial](https://www.runoob.com/vbscript/func-dateserial.html) | 返回指定年、月、日的日期。 |
| [DateValue](https://www.runoob.com/vbscript/func-datevalue.html) | 返回日期。 |
| [Day](https://www.runoob.com/vbscript/func-day.html) | 返回代表一月中的一天的数字（介于并包括 1 到 31 之间）。 |
| [FormatDateTime](https://www.runoob.com/vbscript/func-formatdatetime.html) | 返回格式化为日期或时间的表达式。 |
| [Hour](https://www.runoob.com/vbscript/func-hour.html) | 返回代表一天中的小时的数字（介于并包括 0 到 23 之间）。 |
| [IsDate](https://www.runoob.com/vbscript/func-isdate.html) | 返回指示计算表达式能否转换为日期的布尔值。 |
| [Minute](https://www.runoob.com/vbscript/func-minute.html) | 返回一个数字，代表小时的分钟（介于并包括 0 到 59 之间）。 |
| [Month](https://www.runoob.com/vbscript/func-month.html) | 返回一个数字，代表年的月份（介于并包括 1 到 12 之间）。 |
| [MonthName](https://www.runoob.com/vbscript/func-monthname.html) | 返回指定月份的名称。 |
| [Now](https://www.runoob.com/vbscript/vb-func-now.html) | 返回当前的系统日期和时间。 |
| [Second](https://www.runoob.com/vbscript/func-second.html) | 返回一个数字，代表分钟的秒（介于并包括 0 到 59 之间）。 |
| [Time](https://www.runoob.com/vbscript/func-time.html) | 返回当前的系统时间。 |
| [Timer](https://www.runoob.com/vbscript/func-timer.html) | 返回自 12:00 AM 以来的秒数。 |
| [TimeSerial](https://www.runoob.com/vbscript/func-timeserial.html) | 返回特定小时、分钟和秒的时间。 |
| [TimeValue](https://www.runoob.com/vbscript/func-timevalue.html) | 返回时间。 |
| [Weekday](https://www.runoob.com/vbscript/func-weekday.html) | 返回一个数字，代表一周的天数（介于并包括 1 到 7 之间）。 |
| [WeekdayName](https://www.runoob.com/vbscript/func-weekdayname.html) | 返回一周中指定的一天的星期名。 |
| [Year](https://www.runoob.com/vbscript/func-year.html) | 返回一个数字，代表年份。 |

## Conversion 函数

|  |  |
| --- | --- |
| **函数** | **描述** |
| [Asc](https://www.runoob.com/vbscript/func-asc.html) | 把字符串中的首字母转换为 ANSI 字符代码。 |
| [CBool](https://www.runoob.com/vbscript/func-cbool.html) | 把表达式转换为布尔（Boolean）类型。 |
| [CByte](https://www.runoob.com/vbscript/func-cbyte.html) | 把表达式转换为字节（Byte）类型。 |
| [CCur](https://www.runoob.com/vbscript/func-ccur.html) | 把表达式转换为货币（Currency）类型。 |
| [CDate](https://www.runoob.com/vbscript/func-cdate.html) | 把有效的日期和时间表达式转换为日期（Date）类型。 |
| [CDbl](https://www.runoob.com/vbscript/func-cdbl.html) | 把表达式转换为双精度（Double）类型。 |
| [Chr](https://www.runoob.com/vbscript/func-chr.html) | 把指定的 ANSI 字符代码转换为字符。 |
| [CInt](https://www.runoob.com/vbscript/func-cint.html) | 把表达式转换为整数（Integer）类型。 |
| [CLng](https://www.runoob.com/vbscript/func-clng.html) | 把表达式转换为长整型（Long）类型。 |
| [CSng](https://www.runoob.com/vbscript/func-csng.html) | 把表达式转换为单精度（Single）类型。 |
| [CStr](https://www.runoob.com/vbscript/func-cstr.html) | 把表达式转换为字符串（String）类型。 |
| [Hex](https://www.runoob.com/vbscript/func-hex.html) | 返回指定数字的十六进制值。 |
| [Oct](https://www.runoob.com/vbscript/func-oct.html) | 返回指定数字的八进制值。 |

## Format 函数

|  |  |
| --- | --- |
| **函数** | **描述** |
| [FormatCurrency](https://www.runoob.com/vbscript/func-formatcurrency.html) | 返回作为货币值进行格式化的表达式。 |
| [FormatDateTime](https://www.runoob.com/vbscript/func-formatdatetime.html) | 返回作为日期或时间进行格式化的表达式。 |
| [FormatNumber](https://www.runoob.com/vbscript/func-formatnumber.html) | 返回作为数字进行格式化的表达式。 |
| [FormatPercent](https://www.runoob.com/vbscript/func-formatpercent.html) | 返回作为百分数进行格式化的表达式。 |

## Math 函数

|  |  |
| --- | --- |
| **函数** | **描述** |
| [Abs](https://www.runoob.com/vbscript/func-abs.html) | 返回指定数字的绝对值。 |
| [Atn](https://www.runoob.com/vbscript/func-atn.html) | 返回指定数字的反正切。 |
| [Cos](https://www.runoob.com/vbscript/func-cos.html) | 返回指定数字（角度）的余弦。 |
| [Exp](https://www.runoob.com/vbscript/func-exp.html) | 返回 *e*（自然对数的底）的幂次方。 |
| [Hex](https://www.runoob.com/vbscript/func-hex.html) | 返回指定数字的十六进制值。 |
| [Int](https://www.runoob.com/vbscript/func-int.html) | 返回指定数字的整数部分。 |
| [Fix](https://www.runoob.com/vbscript/func-fix.html) | 返回指定数字的整数部分。 |
| [Log](https://www.runoob.com/vbscript/func-log.html) | 返回指定数字的自然对数。 |
| [Oct](https://www.runoob.com/vbscript/func-oct.html) | 返回指定数字的八进制值。 |
| [Rnd](https://www.runoob.com/vbscript/func-rnd.html) | 返回小于1但大于或等于0的一个随机数。 |
| [Sgn](https://www.runoob.com/vbscript/func-sgn.html) | 返回可指示指定数字的符号的一个整数。 |
| [Sin](https://www.runoob.com/vbscript/func-sin.html) | 返回指定数字（角度）的正弦。 |
| [Sqr](https://www.runoob.com/vbscript/func-sqr.html) | 返回指定数字的平方根。 |
| [Tan](https://www.runoob.com/vbscript/func-tan.html) | 返回指定数字（角度）的正切。 |

## Array 函数

|  |  |
| --- | --- |
| **函数** | **描述** |
| [Array](https://www.runoob.com/vbscript/vb-func-array.html) | 返回一个包含数组的变量。 |
| [Filter](https://www.runoob.com/vbscript/func-filter.html) | 返回下标从零开始的数组，其中包含基于特定过滤条件的字符串数组的子集。 |
| [IsArray](https://www.runoob.com/vbscript/func-isarray.html) | 返回一个指示指定的变量是否为数组的布尔值。 |
| [Join](https://www.runoob.com/vbscript/func-join.html) | 返回一个由数组中若干子字符串组成的字符串。 |
| [LBound](https://www.runoob.com/vbscript/func-lbound.html) | 返回指示数组维数的最小下标。 |
| [Split](https://www.runoob.com/vbscript/func-split.html) | 返回下标从零开始的一维数组，包含指定数量的子字符串。 |
| [UBound](https://www.runoob.com/vbscript/func-ubound.html) | 返回指示数组维数的最大下标。 |

## String 函数

|  |  |
| --- | --- |
| **函数** | **描述** |
| [InStr](https://www.runoob.com/vbscript/func-instr.html) | 返回字符串在另一字符串中首次出现的位置。搜索从字符串的第一个字符开始。 |
| [InStrRev](https://www.runoob.com/vbscript/func-instrrev.html) | 返回字符串在另一字符串中首次出现的位置。搜索从字符串的最末字符开始。 |
| [LCase](https://www.runoob.com/vbscript/func-lcase.html) | 把指定字符串转换为小写。 |
| [Left](https://www.runoob.com/vbscript/func-left.html) | 从字符串的左侧返回指定数量的字符。 |
| [Len](https://www.runoob.com/vbscript/func-len.html) | 返回字符串中的字符数量。 |
| [LTrim](https://www.runoob.com/vbscript/func-ltrim.html) | 删除字符串左侧的空格。 |
| [RTrim](https://www.runoob.com/vbscript/func-rtrim.html) | 删除字符串右侧的空格。 |
| [Trim](https://www.runoob.com/vbscript/func-trim.html) | 删除字符串左侧和右侧的空格。 |
| [Mid](https://www.runoob.com/vbscript/func-mid.html) | 从字符串中返回指定数量的字符。 |
| [Replace](https://www.runoob.com/vbscript/func-replace.html) | 使用另一个字符串替换字符串的指定部分指定的次数。 |
| [Right](https://www.runoob.com/vbscript/func-right.html) | 从字符串的右侧返回指定数量的字符。 |
| [Space](https://www.runoob.com/vbscript/func-space.html) | 返回由指定数量的空格组成的字符串。 |
| [StrComp](https://www.runoob.com/vbscript/func-strcomp.html) | 比较两个字符串，返回代表比较结果的一个值。 |
| [String](https://www.runoob.com/vbscript/func-string.html) | 返回包含指定长度的重复字符的字符串。 |
| [StrReverse](https://www.runoob.com/vbscript/func-strreverse.html) | 反转字符串。 |
| [UCase](https://www.runoob.com/vbscript/func-ucase.html) | 把指定的字符串转换为大写。 |

## 其他函数

|  |  |
| --- | --- |
| **函数** | **描述** |
| [CreateObject](https://www.runoob.com/vbscript/func-createobject.html) | 创建指定类型的对象。 |
| [Eval](https://www.runoob.com/vbscript/func-eval.html) | 计算表达式，并返回结果。 |
| [GetLocale](https://www.runoob.com/vbscript/func-getlocale.html) | 返回当前的 locale ID。 |
| [GetObject](https://www.runoob.com/vbscript/func-getobject.html) | 返回对文件中 automation 对象的引用。 |
| [GetRef](https://www.runoob.com/vbscript/func-getref.html) | 允许您把 VBScript 子程序连接到页面上的一个 DHTML 事件。 |
| [InputBox](https://www.runoob.com/vbscript/func-inputbox.html) | 显示对话框，用户可在其中输入文本，并/或点击按钮，然后返回内容。 |
| [IsEmpty](https://www.runoob.com/vbscript/func-isempty.html) | 返回一个布尔值，指示指定的变量是否已被初始化。 |
| [IsNull](https://www.runoob.com/vbscript/func-isnull.html) | 返回一个布尔值，指示指定的表达式是否包含无效数据（Null）。 |
| [IsNumeric](https://www.runoob.com/vbscript/func-isnumeric.html) | 返回一个布尔值，指示指定的表达式是否可作为数字来计算。 |
| [IsObject](https://www.runoob.com/vbscript/func-isobject.html) | 返回一个布尔值，指示指定的表达式是否是一个 automation 对象。 |
| [LoadPicture](https://www.runoob.com/vbscript/func-loadpicture.html) | 返回一个图片对象。仅用于 32 位平台。 |
| [MsgBox](https://www.runoob.com/vbscript/func-msgbox.html) | 显示消息框，等待用户点击按钮，并返回指示用户点击了哪个按钮的值。 |
| [RGB](https://www.runoob.com/vbscript/func-rgb.html) | 返回一个表示 RGB 颜色值的数字。 |
| [Round](https://www.runoob.com/vbscript/func-round.html) | 对数字进行四舍五入。 |
| [ScriptEngine](https://www.runoob.com/vbscript/func-scriptengine.html) | 返回使用中的脚本语言。 |
| [ScriptEngineBuildVersion](https://www.runoob.com/vbscript/func-scriptengine.html) | 返回使用中的脚本引擎的内部版本号。 |
| [ScriptEngineMajorVersion](https://www.runoob.com/vbscript/func-scriptengine.html) | 返回使用中的脚本引擎的主版本号。 |
| [ScriptEngineMinorVersion](https://www.runoob.com/vbscript/func-scriptengine.html) | 返回使用中的脚本引擎的次版本号。 |
| [SetLocale](https://www.runoob.com/vbscript/func-setlocale.html) | 设置 locale ID，并返回之前的 locale ID。 |
| [TypeName](https://www.runoob.com/vbscript/func-typename.html) | 返回指定变量的子类型。 |
| [VarType](https://www.runoob.com/vbscript/func-vartype.html) | 返回指示变量子类型的值。 |

# VBScript 关键字

## VBScript 关键字

|  |  |
| --- | --- |
| **关键字** | **描述** |
| Empty | 用于指示一个未初始化的变量值。当第一次创建变量时或变量值显式设置为空时，变量值未初始化且变量为被赋值。  实例： Dim x '变量 x 未初始化！ x="ff" '变量 x 不再是未初始化 x=Empty '变量 x 未初始化！  **注意：**这和 Null 不一样！！ |
| IsEmpty | 用于测试一个变量是否未初始化。  实例：If (IsEmpty(x)) '变量 x 未初始化？ |
| Nothing | 用于指示一个未初始化的对象值，或者把对象变量从对象分离用于释放系统资源。  实例: Set myObject=Nothing |
| Is Nothing | 用于测试一个值是否是初始化的对象。  实例：If (myObject Is Nothing) '它是否未设置？  **注意：**如果您把一个值与 Nothing 作比较，您将不会得到正确的结果！实例: If (myObject = Nothing) '总是错误！ |
| Null | 用于指示变量不包含有效数据。  Null 把值设置为"无效"，Empty 则表示值"未设置"。  **注意：**这不同于 Empty 或 Nothing！！  实例：x=Null 'x 不包含有效数据 |
| IsNull | 用于测试一个值是否包含无效数据。  实例: if (IsNull(x)) 'x 是无效的？ |
| True | 用于指示一个布尔条件是正确的（True 为 -1） |
| False | 用于指示一个布尔条件是不正确的（False 为 0） |

# msgbox

### 语法

MsgBox (Prompt [,Buttons] [,Title] [,Helpfile,Context])

### 参数

|  |  |
| --- | --- |
| **MsgBox函数参数** | |
| **部分** | **描述** |
| Prompt | 必选。字符串[表达式](https://baike.baidu.com/item/%E8%A1%A8%E8%BE%BE%E5%BC%8F?fromModule=lemma_inlink" \t "https://baike.baidu.com/item/msgbox/_blank)，显示在对话框中的消息。Prompt的最大长度大约为  1024个字符，由所用字符的字节大小决定。如果Prompt的内容超过一行，则可以在每一行之间用回车符（Chr(13)）、换行符（Chr(10)）或是回车与换行符的组合（Chr(13) & Chr(10)，即[vbCrLf](https://baike.baidu.com/item/vbCrLf?fromModule=lemma_inlink" \t "https://baike.baidu.com/item/msgbox/_blank)）将各行分隔开来。 |
| Buttons | 可选。数值表达式，是一些数值的总和，指定所显示的按钮的数目及形式、使用的图标样式（及声音），缺省按钮以及消息框的强制性等。如果省略，则其缺省值为0。具体数值见“参数设置值”表格。 |
| Title | 可选。字符串表达式，在对话框标题栏中显示的内容。如果省略Title，则将应用程序标题（App.Title）放在标题栏中。 |
| Helpfile | 可选。字符串表达式，用来向对话框提供上下文相关帮助的帮助文件。如果提供了Helpfile，则也必须提供Context。 |
| Context | 可选。数值表达式，由帮助文件的作者指定给适当的帮助主题的帮助上下文编号。如果提供了Context，则也必须提供Helpfile。 |

### 参数设置值

|  |  |  |
| --- | --- | --- |
| **Buttons参数设置值** | | |
| **常数** | **值** | **描述** |
| **对话框中显示的按钮的类型与数目** | | |
| vbOKOnly | 0 | 只显示“确定”按钮。（缺省） |
| vbOKCancel | 1 | 显示“确定”和“取消”按钮。 |
| vbAbortRetryIgnore | 2 | 显示“终止”、“重试”和“忽略”按钮。 |
| vbYesNoCancel | 3 | 显示“是”、“否”和“取消”按钮。 |
| vbYesNo | 4 | 显示“是”和“否”按钮。 |
| vbRetryCancel | 5 | 显示“重试”和“取消”按钮。 |
| **图标的样式（根据系统设置，可能伴有声音）** | | |
| vbCritical | 16 | 显示“错误信息”图标。 |
| vbQuestion | 32 | 显示“询问信息”图标。 |
| vbExclamation | 48 | 显示“警告消息”图标。 |
| vbInformation | 64 | 显示“通知消息”图标。 |
| **默认按钮** | | |
| vbDefaultButton1 | 0 | 第一个按钮是默认按钮。（缺省） |
| vbDefaultButton2 | 256 | 第二个按钮是默认按钮。 |
| vbDefaultButton3 | 512 | 第三个按钮是默认按钮。 |
| vbDefaultButton4 | 768 | 第四个按钮是默认按钮。 |
| **对话框的强制返回性** | | |
| vbApplicationModal | 0 | 应用程序强制返回；应用程序一直被挂起，直到用户对消息框作出响应才继续工作。 |
| vbSystemModal | 4096 | 系统强制返回；全部应用程序都被挂起，直到用户对消息框作出响应才继续工作。 |
| **对话框特殊设置** | | |
| vbMsgBoxHelpButton | 16384 | 将帮助按钮添加到消息框。 |
| vbMsgBoxSetForeground | 65536 | 指定消息框窗口作为前景窗口。 |
| vbMsgBoxRight | 524288 | 文本为右对齐。 |
| vbMsgBoxRtlReading | 1048576 | 指定文本应为在希伯来和阿拉伯语系统中的从右到左显示。 |

这些常数都是Visual Basic for Applications（[VBA](https://baike.baidu.com/item/VBA?fromModule=lemma_inlink" \t "https://baike.baidu.com/item/msgbox/_blank)）指定的。所以，可以在程序代码中使用这些常数名称，而不使用实际数值。

### 返回值

|  |  |  |
| --- | --- | --- |
| **MsgBox函数返回值** | | |
| **常数** | **值** | **描述** |
| vbOK | 1 | 单击了“确定”按钮。 |
| vbCancel | 2 | 单击了“取消”按钮。 |
| vbAbort | 3 | 单击了“终止”按钮。 |
| vbRetry | 4 | 单击了“重试”按钮。 |
| vbIgnore | 5 | 单击了“忽略”按钮。 |
| vbYes | 6 | 单击了“是”按钮。 |
| vbNo | 7 | 单击了“否”按钮。 |