# 定义变量:

## 定义一个变量:

Dim 变量名

## 定义多个变量:

Dim 变量名1, 变量名2, 变量名3

## 定义一个点:

Dim 点位名称 As New point

## 定义一个常量:

Const 常量名 = 值

Const 常量名 = "字符串"

常量和变量的不同点在于, 他们在定义的时候被赋值,以后就不能改变了,如果企图改变将会引发一个错误, 一般说来, 常量名全部使用大写

# 注释:

注释(以REM或'开头)行在程序中不起作用, 但能让别人更容易读懂你的程序.

# 给变量赋值:

变量名 = 值

变量名 = "字符串" 以""包裹起来的字符称之为"字符串"

# 获取,设置点值

PointGet("点位名称")

PointSet "点位名称",

# 输出弹框,输入弹框

Dim 变量名

变量名 = InputBox("提示文本","弹框标头","默认输入值")

注意:InputBox的返回值是一个字符串

案例:

Dim a , b , c

a = InputBox("请输入一个数字:","第一个数","1")

b = InputBox("请输入一个数字:","第二个数","2")

c = a + b

MsgBox c

结果:12

使用内建函数Int将输入值转化成整数值

c = (Int(a) + Int(b)) \* 2

结果:6

MsgBox 变量名

MsgBox “字符串”

# 运算符:

## 加减乘除小括号:

vbs中的数学计算和真正的算术没有什么不同, +,-,\*,/, ,()

## 判断:

=

<>

案例:

Dim a , b

a = 2

b = 2

If a = b Then

MsgBox "a等于b"

ElseIf a <> b Then

MsgBox "a不等于b"

End If

结果:a等于b

## 取余运算:

Mod

案例:

Dim a

a = 17 Mod 5

MsgBox a

结果:2

## 乘幂运算:

^

案例:

Dim a

a = 2^3

MsgBox a

结果:8

# 条件语句:

## If...Then…Else:

运行一段代码:

案例:

Dim a

a = 100

If a < 200 Then

a = 300

MsgBox a

End If

结果:300

运行多段代码:

案例:

Dim a

a = InputBox("请输入一个0 - 100的数:","数学成绩")

If a > 79 And a < 101 Then

MsgBox"及格"

ElseIf a > -1 And a < 80 Then

MsgBox"不及格"

Else

MsgBox"输入错误"

End If

## Select Case:

案例:

Dim a

a = InputBox("请输入一个0 - 10的数:","投进球的个数","0")

Select Case a

Case 0,1,2,3

MsgBox "还行"

Case 4,5,6

MsgBox "很好"

Case 7,8,9

MsgBox "很棒"

Case 10

MsgBox "完美"

Case Else

MsgBox "输入错误"

End Select

注意： Select Case结构只计算开始处的一个表达式（只计算一次），而 If...Then...ElseIf 结构计算每个 ElseIf 语句的表达式，这些表达式可以各不相同。仅当每个 ElseIf 语句计算的表达式都相同时，才可以使用 Select Case 结构代替If...Then...ElseIf 结构。

# 循环语句:

## For…Next:

For counter = start To end [Step step]

  [statements]

  [Exit For]

  [statements]

Next

主要参数：

counter：用做循环计数器的数值变量。这个变量不能是数组元素或用户自定义类型的元素。

start：counter的初值。

end：counter的终值。

step：counter的步长。如果没有指定，则step的默认值为1。

案例:

Dim a : a = 0 ' : 相当于换行符

For count = 1 To 10 Step 2 'Step 2 步长为2

a = a + 1

MsgBox "count的值:" & count

If a = 3 Then

Exit For '当a的值等于3时退出For循环

End If

Next

MsgBox "a的值:" & a

结果:count的值1,3,5 a 的值3

## Do…Loop:

重复执行语句直到条件变为True

案例1:

Dim a : a = 6

Do Until a = 0

a = a - 1

MsgBox a

Loop

MsgBox "循环结束"

案例2:

Dim a : a = 6

Do

MsgBox a

a = a - 1

Loop Until a = 0

MsgBox "循环结束"

当条件变为True之前重复执行某语句块

案例1:

Dim a : a = 6

Do While a <> 0 'a不等于0时重复执行该语句块

MsgBox a

a = a - 1

Loop

MsgBox "循环结束"

案例2:

Dim a : a = 6

Do

MsgBox a

a = a - 1

Loop While a <> 0 'a 不等于0时重复执行上面语句块

MsgBox "循环结束"

案例3:

Dim a : a = 6

Do

a = a - 1

MsgBox a

If a = 3 Then '当a等于3时退出Do循环

Exit Do

End If

Loop While a <> 0 'a 不等于0时重复执行上面语句块

MsgBox "循环结束"

## While…Wend:

案例:

Dim a : a = 6

While a <> 0

MsgBox a

a = a - 1

Wend

MsgBox "循环结束"

注意:

While…Wend 没有Exit语句，从头一直循环到尾，若要在中途退出，则需用Do…Loop语句

## For Each...Next:

在Cimplicity中不适用

# 数组:

## 静态数组:

案例:

Dim a (2)

a(0) = "a1"

a(1) = "a2"

a(2) = "a3"

For i = 0 To ubound(a) 'ubound函数,返回指定数组维数的最大可用下标

MsgBox a(i)

Next

结果:a1;a2;a3

## 动态数组:

案例1:

Dim a ()

For i = 0 To 2

ReDim Preserve a(i) 'redim语句,用于过程中声明动态数组变量并分配或重新分配存储空间

a(i) = i + 1

MsgBox a(i)

Next

结果:1;2;3

案例2:

Dim a () '定义一个一维动态数组

ReDim Preserve a(2) '重新定义该数组大小

a(0) = "我"

a(1) = "是"

a(2) = "中"

ReDim Preserve a(4) '重新定义该数组大小

a(3) = "国"

a(4) = "人"

For i = 0 To ubound(a)

MsgBox a(i)

Next

## 一维数组:

格式:

dim 数组名(最大下标)

案例1:

Dim a(6) , Str

For i = 0 To 6

a(i) = i

Str = Str & i & ","

MsgBox a(i)

Next

MsgBox Str

结果:0,1,2,3,4,5,6,

案例2:

Dim name1(2),high1(2),class1(2) '定义三个数组,分别存储姓名,身高,班级

For i = 0 To 2

name1(i) = InputBox("请输入第" & i + 1 & "个学生的姓名:")

high1(i) = InputBox("请输入第" & i + 1 & "个学生的身高:")

class1(i) = InputBox("请输入第" & i + 1 & "个学生的班级:")

Next

Dim cname , temp '定义两个变量,cname用来存储要查询学生的姓名,temp存储下标

cname = InputBox("请输入要查询学生的姓名:")

For i = 0 To 2

If name1(i) = cname Then

temp = i

End If

Next

MsgBox cname & "的身高是:" & high1(temp) & ",班级是:" & class1(temp)

## 二维数组:

格式:

dim 数组名(第一个数组的最大下标, 第二个数组的最大下标)

案例:

Dim a (2,3) '0-2是3,0-3是4,共有3\*4=12个元素

For i = 0 To 2

For j = 0 To 3

Select Case i

Case 0

a(i,j) = "A" & j

Case 1

a(i,j) = "B" & j

Case 2

a(i,j) = "C" & j

End Select

MsgBox a(i,j)

Next

Next

# 常用函数及功能:

## 常用函数:

|  |  |  |
| --- | --- | --- |
| 函数名 | 语法 | 功能 |
| Abs | Abs(number) | 返回一个数的绝对值 |
| Sqr | Sqr(number) | 返回一个数的平方根 |
| Sin | Sin(number) | 返回角度的正玄值 |
| Cos | Cos(number) | 返回角度的余玄值 |
| Tan | Tan(number) | 返回角度的正切值 |
| Atn | Atn(number) | 返回角度的反正切值 |
| Log | Log(number) | 返回一个数的自然对数 |
| Int | Int(number) | 取整函数，返回一个小于number的第一整数 |
| Rnd | Rnd() | 返回一个从0到1的随机数 |
| Ubound | Ubound(数组名，维数) | 返回该数组的最大下标 |
| Lbound | Lbound(数组名，维数) | 返回最小下标数 |

案例:

Dim b(3,4)

Dim a : a = 1.234

msgbox abs(a)

msgbox sqr(a)

msgbox sin(a)

msgbox cos(a)

msgbox tan(a)

msgbox atn(a)

msgbox log(a)

msgbox int(a)

msgbox rnd(a)

msgbox ubound(b,2)

msgbox lbound(b,2)

## 各种转换函数及功能:

|  |  |
| --- | --- |
| 函数 | 功能 |
| CStr(variant) | 将变量variant转化为字符串类型 |
| CDate(variant) | 将变量variant转化为日期类型 |
| CInt(variant) | 将变量variant转化为整数类型 |
| CLng(variant) | 将变量variant转化为长整数类型 |
| CSng(variant) | 将变量variant转化为single类型 |
| CDbl(variant) | 将变量variant转化为double类型 |
| CBool(variant) | 将变量variant转化为布尔类型 |

案例:

Dim a : a = 1.234

msgbox cstr(a)

msgbox cdate(a)

msgbox cint(a)

msgbox clng(a)

msgbox csng(a)

msgbox cdbl(a)

msgbox cbool(a)

注释：

1整型  
以Integer 表示整型，其范围为 -32,768 到 32,767 之间。

2、长整型  
Long(长整型) ，其范围从 -2,147,483,648 到 2,147,483,647。

3、单精度型(Single)   
Single(单精度浮点型)，它的范围在负数的时候是从 -3.402823E38 到 -1.401298E-45，而在正数的时候是从 1.401298E-45 到 3.402823E38。

4、双精度型(Double)  
Double(双精度浮点型)它的范围在负数的时候是从 -1.79769313486232E308 到 -4.94065645841247E-324，而正数的时候是从 4.94065645841247E-324 到 1.79769313486232E308。

### 常用的字符串函数及功能:

|  |  |  |
| --- | --- | --- |
| 函数 | 语法 | 功能 |
| Len | Len(string) | 返回string字符串里的字符数目 |
| Trim | Trim(string) | 将字符串前后的空格去掉 |
| Ltrim | Ltrim(string) | 将字符串前面的空格去掉 |
| Rtrim | Rtrim(string) | 将字符串后面的空格去掉 |
| Left | Left(string,length) | 从string字符串的左边取length长度的字符串 |
| Right | Right(string,length) | 从srting字符串的右边取得length长度的字符串 |
| LCase | LCase(string) | 将字符串里的所有大写字母转化成小写字母 |
| UCase | UCase(string) | 将字符串里的小写字母转化成大写字母 |
| StrComp | Strcomp(string1,string1) | 返回string1字符串与string2字符串的比较结果，如果两个字符串相同，返回0 |
| InStr | InStr(string1,string2) | 返回string2字符串在string1字符串中第一次出现的位置 |
| Asc | Asc(str) | 把字符串中的第一个字母转换为对应的ANSI代码，并返回结果 |

案例:

Dim str : str = " Abc "

Dim Str1 : str1 = " Abc "

Dim str2 : str2 = "A"

'MsgBox Len(str) '5

'MsgBox Len(Trim(str)) '3

'MsgBox Len(Ltrim(str)) '4

'MsgBox Len(Rtrim(str)) '4

'MsgBox Left(str,3) ' Ab

'MsgBox Right(str,3) 'bc

'MsgBox LCase(str) ' abc

'MsgBox UCase(str) ' ABC

'MsgBox StrComp(str,str1) '0

MsgBox InStr(str,str2) '2

msgBox Asc("A") '65

## 常用日期和时间函数及说明:

|  |  |  |
| --- | --- | --- |
| 函数 | 语法 | 功能 |
| Now | Now() | 取得系统当前的日期和时间 |
| Date | Date() | 取得系统当前的日期 |
| Time | Time() | 取得系统当前的时间 |
| Year | Year() | 取得给定日期的年份 |
| Month | Month(Date) | 取得给定日期的月份 |
| Day | Day(Date) | 取得给定日期是几号 |
| Hour | Hour(time) | 取得给定时间是第几小时 |
| Minute | Minute(time) | 取得给定时间是第几分钟 |
| Second | Second(time) | 取得给定时间是第几秒 |
| WeekDay | WeekDay(Date) | 取得给定日期是星期几的整数 1 表示星期一 2 表示星期二 依次类推 |
| DateDiff | DateDiff(“Var”, Var1,Var2) Var:日期或时间间隔因子 Var1：第一个日期或时间 Var2：第二个日期或时间 | 计算两个日期或时间的间隔 |
| DateAdd | DateAdd(“Var”, Var1,Var2) Var:日期或时间间隔因子 Var1：日期或时间 Var2：日期或时间 | 对两个日期或时间作加法 DateAdd(“d”，10，Date()) 10天后是几号 |
| FormatDateTime | FortDateTime(Date,vbShortDate) | 转化为短日期格式 |
| FortDateTime(Date,vblongDate) | 转化为长日期格式 |
| FortDateTime(Date,vbShortTime) | 转化为短时间格式 |
| FortDateTime(Date,vbLongTime) | 转化为长时间格式 |

日期或时间间隔因子

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 间隔因子 | yyyy | m | d | ww | h | s |
| 说明 | 年 | 月 | 日 | 星期 | 小时 | 秒 |

案例:

'MsgBox Now()

'MsgBox Date()

'MsgBox Time()

'MsgBox Month(Now)

'MsgBox Minute(Now)

'MsgBox Second(Now)

'MsgBox DateAdd("yyyy",1,"2017/12/31") '加一年

'MsgBox DateDiff("d",Date,"2017/12/31") '时间间隔天数

Dim d

d="2/10/2012 16:25:56"

MsgBox DatePart("yyyy",d) '返回给定日期的指定部分,年

## 常用的检验函数及功能:

|  |  |
| --- | --- |
| 函数 | 功能 |
| VarType(variant) | 检查变量vriant的值，函数值为该变量的数据子类型，0表示空，2表示整数，7表示日子，8表示字符串，11表示布尔变量，8204表示数组 |
| IsNumeric(variant) | 检查变量variant的值，如果variant是数值类型，则函数值为ture |
| IsNull(variant) | 检查变量variant的值，如果variant为null，则函数值为ture |
| IsEmpty(variant) | 检查变量的值，如果variant是empty，则函数值为ture |
| IsObject(variant) | 检查变量variant的值，如果variant是对象类型，则函数值为ture |
| IsDate(variant) | 检查变量variant的值，如果variant是日期类型，则函数值为ture |
| IsArray(variant) | 检查变量variant的值，如果variant是数组类型，则函数值为ture |

案例:

Dim k : k = ""

Dim Num : Num = 1

Dim Str : Str = "aaa"

Dim Bool : Bool = TRUE

Dim arr(2)

'MsgBox VarType(k) '0

'MsgBox VarType(Num) '2

'MsgBox VarType(Str) '8

'MsgBox VarType(Bool) '11

MsgBox VarType(arr) '8204

# 自定义函数和过程:

案例:

Sub Main()

Dim a1,a2

a1 = InputBox("请输入第一个数:")

a2 = InputBox("请输入第二个数:")

Call CompareNum(a1,a2)

End Sub

Sub CompareNum(a1,a2)

If a1 > a2 Then

MsgBox a1 & "比" & a2 & "大"

ElseIf a1 < a2 Then

MsgBox a1 & "比" & a2 & "小"

Else

MsgBox a1 & "和" & a2 & "相等"

End If

End Sub