ME 232 Cheat Sheet

by takshimaro via cheatography.com/20184/cs/3030/

VBA Data Types

Variant	any data type
Integer	(2 bytes) integer
Long	(4 bytes) integer
Single	(4 bytes) floating point
Double	(8 bytes) floating point
String	non-numeric data **(declared with quotes)**
Object	any object reference
Date	a date
Boolean	True / False
Byte	0-255

Operators (Syntax)

Comparison	**=**, **<>** (not equal to), **>**, **<** , **>=** , **<=**
Logical (Boolean)	**NOT** (opposite), **AND** (if all *true* then returns true), **OR** (at least 1 *true* returns true)
Mathematical	**+**, **-**, *, **/** , **** (integer division), **Mod** (remainder), **^** (remember to put space)
String Concatenation	**&**

The concatenate operator eg. "A" & "B" becomes "AB".

Operators (Precedence)

1	**^**
2	* OR **/** (division)
3	**** (integer division)
4	**Mod**
5	**+** OR **-**

Declarations

Variables	`Dim [varname] As [type]`
Arrays	`Dim [arrayname(index)] As [type]`
Re-declare Array	`ReDim [arrayname(newindex?)] As [newtype?]`

Declarations (cont)

ReDim and keep values	`ReDim Preserve [arrayname(newindex?)] As [newtype?]`

`Option Explicit` is used to require declarations of all variables. `Option Base 1` makes the index of all arrays starts from 1. If declared array has no numeral `index` provided, then it is dynamic in size.

Data Functions (Conversions)

...value to a boolean	`CBool(value)`
...value to a integer	`CInt(value)`
...value to a double	`CDbl(value)`
...value to a string	`CStr(value)`

`Val` function accepts a string as input and returns the numbers found in that string.

Math Functions

Absolute	`Abs([numeric value])`
Square root	`Sqr([numeric value])`
Exponential, *e*	`Exp([numeric value])`
Natural log, *ln*	`Log([numeric value])`
Is it a number (boolean)?	`IsNumeric([numeric value])`
Truncate to integer	`Int([numeric value])`
`(Num1/Num2)` remainder?	`[Num2] Mod [Num2]`
Round to a decimal place	`Round([numeric value],[# of digits])`

`[Sin/Cos/Tan](x)` for trigonometric functions,
`[ASin/ACos/ATan](x)` for inverse trig functions.

String Functions

All upper case	`UCase([string value] )`
All lower case	`LCase([string value] )`
Length of string (integer)	`Len([string value] )`
Filters a string to a double	`Val([string value])`
Convert number to string	`Str([numeric value])`

`Val` function accepts a string as input and returns the numbers found in that string.

User Interaction and Cell Selection

A popup dialogue box	`MsgBox "dialogue" [& variable etc.]`
Prompt user for input	`InputBox ("dialogue")`
...a single cell **A1**	`Range("A1").Select`
...an active cell	`ActiveCell.Select`
...a contiguous range	`Range("A1:G5").Select`
...offset and select	`[ActiveCell/Range(#)].Offset(1,0).Select`
...set a cell's value	`[ActiveCell/Range(#)].Value = [varname]`

Use `Range` to select specific cells or group of cells. Use `ActiveCell` to select highlighted cell in excel.

Array Functions

Highest element number	`UBound(ArrayName [, Dimension] )`
Lowest element number	`LBound(ArrayName [, Dimension] )`

Highest element number is the size of possible entries a array can hold.

Array Iteration

```
Function MinIntegerofArray(TheArray As Variant) As Integer
Dim i As Integer, placeholder As Integer
placeholder = 0
For i = 1 To UBound(TheArray)
    If TheArray(i) < TheArray(placeholder) Then
        placeholder = i
    End If
Next
'index of min value is at placeholder
MinIntegerofArray = TheArray(placeholder)
End Function
```

Error Handling with GoTo & Labels

```
...
tempstudentName = InputBox("Please enter student name (type exit to end): ")
'error check and force reentry of input
    If IsText(tempstudentName) = False Then
checker1:
        tempstudentName = InputBox("Please enter a valid student name [not blank and letter] (type exit to end): ")
    End If

    If tempstudentName = "exit" Then
        End '(the program)
    ElseIf IsText(tempstudentName) = False Then
        GoTo checker1 'label
    Else
        studentName = tempstudentName
    End If
...
```

'checker1:' is a label: labels only include the next line

IsText, Case Statements, For loop

```
Function IsText(streng As Variant) As Boolean
    Dim i As Integer
    For i = 1 To Len(streng)
    'checks if the text follows the ACII numerials (a-z AND A-Z)

        Select Case Asc(Mid(streng, i, 1))
            Case 65 To 90, 97 To 122
                IsText = True
            Case Else
                IsText = False
                Exit For
        End Select
    Next
End Function
```