

David Liska's Excel Formula & Macro Cheat Sheet

Formula Components

VLOOKUP/HLOOKUP

Explanation:

VLOOKUP is the easiest method to provide the critical link between a template table and the data which will fill it. By specifying the value to match as a combination of values, it is possible to populate a table based on its leading column and row.

Syntax:

=VLOOKUP(*value to match, target range, column to reference relative to the left edge of target range, [Range? TRUE or FALSE]*)

IF

Explanation:

IF statements may not be necessary in early endeavors, but the decision structure they provide, especially when nested, becomes critical in more advanced data-handling applications.

Syntax:

=IF(*conditional test, value if true, value if false*)

IFERROR

Explanation:

If VLOOKUP targeting with a data set where some elements may be missing, recommended to prevent N/A errors from cascading through your formula .

Syntax:

=IFERROR(*value, alternate value if value results in error*)

COUNTIF

Explanation:

Countif works well with the template & macro combo method. Conditionally counting cells with values may be especially useful if you wish templates to morph to accommodate various rows of data.

Syntax:

=COUNTIF(*range of cells, criteria*)

TRIM

Explanation:

Trim removes preceding/trailing spaces from your data, which can cause unintended results for comparisons.

Syntax:

=TRIM(*text or cell reference*)

XOR

Explanation:

Exclusive OR statement evaluator will return TRUE if one and only one of the criteria is TRUE.

Syntax:

=XOR(*1st logical condition, 2nd condition, [3rd condition, ...]*)

AND/OR

Explanation:

The AND and OR functions can require an entire set of conditions to be true or a single condition from a set of conditions to be true for an IF statement to be fulfilled.

Syntax:

=AND(*1st condition, 2nd condition, [3rd condition, ...]*)

Macro Components

Variable Declarations

	Type	Properties
Dim <i>VariableName</i> As	Integer	Numbers ± 32,767
	String	Text up to 2 billion characters
	Long	Numbers ± 2.1B
	Boolean	True or False
	Date	Jan. 1, 0100 to Dec 31, 9999

Cell Selection

Select cell A1 on currently active worksheet:

```
Range("A1").Select
```

Select cells A1 through D4 on active worksheet:

```
Range("A1:D4").Select
```

Select cell A1 on worksheet Sheet2:

```
Range("Sheet2!A1").Select
```

Cell Value Assignment

Set cell value:

```
Worksheets("Sheet1").Range("A1").Value = 3458
```

```
Worksheets("Sheet1").Range("A1").Value = strVariableName
```

Remove cell contents:

```
Worksheets("Sheet1").Range("A1:A30").Clear
```

6 Variable Value Assignment Methods

```
strVariableName = "Yada, yada, yada"
```

```
strVariableName = "Total: " & intVariableName
```

```
strVariableName = Range("Sheet1!A1")
```

```
intVariableName = 3720
```

```
intVariableName = intVariableName + 1
```

```
strVariableName = ActiveCell.Value
```

Looping

For...Next

Running through a loop 20 times:

```
For intCounterVariable = 1 To 20
```

 Tasks

```
Next
```

Do While

Running through a loop while Boolean variable is true:

```
Do While boolVariable = True
```

 Tasks

```
Loop
```

Copying & Pasting

Copy from cell A1 to cell D4:

```
Range("A1").Copy Range("D4")
```

Notifications

Generate a popup notification with either literal text, a variable's value, a cell value, or any combination:

```
MsgBox "Literal text" & VariableName & Range("B2")
```

Filtering a List

Filters an 800 row table on the current worksheet according to column A (Because it is field 1) for matches to variable.

```
ActiveSheet.Range  
("$A$1:$C$800").AutoFilter Field:=1,  
Criteria1:= strVariableName
```

Output

Export to PDF in the C:\Folder location with the filename equal to value contained in string variable strVariableName:

```
ActiveSheet.ExportAsFixedFormat Type:=xlTypePDF, Filename:"C:\Folder\" & strVariableName, Quality:=xlQualityStandard,  
IncludeDocProperties:=True, IgnorePrintAreas:=False, OpenAfterPublish:=False
```

Send to printer:

```
ActiveWindow.SelectedSheets.PrintOut Copies=1, Collate:=True, IgnorePrintAreas=False
```