



# TRAINING RESOURCES

Excel VBA Cheat Sheet

[Enquiries@skilltec.co.uk](mailto:Enquiries@skilltec.co.uk)



## Contents

Sheets .....	2
Cells & Ranges .....	3
Rows .....	4
Columns .....	4
Workbooks .....	5
Settings .....	6
Errors .....	7
Files .....	7
Arrays .....	7
Collections .....	8
Dictionaries .....	9



# Sheets

Description	VBA code
<b>Activate by Tab Name</b>	Sheets("Input").Activate
<b>Activate by VBA Code Name</b>	Sheet1.Activate
<b>Activate by Index Position</b>	Sheets(1).Activate
<b>Next Sheet</b>	ActiveSheet.Next.Activate
<b>Get ActiveSheet</b>	MsgBox ActiveSheet.Name
<b>Select Sheet</b>	Sheets("Input").Select
<b>Set to Variable</b>	Dim ws as Worksheet Set ws = ActiveSheet
<b>Name / Rename</b>	ActiveSheet.Name = "NewName"
<b>Add Sheet</b>	Sheets.Add
<b>Add Sheet and Name</b>	Sheets.Add.Name = "NewSheet"
<b>Add Sheet to Variable</b>	Dim ws As Worksheet Set ws = Sheets.Add
<b>Copy Sheet</b>	Sheets("Sheet1").Copy Before:=Sheets("Sheet2")
<b>Hide Sheet</b>	Sheets("Sheet1").visible = False or Sheets("Sheet1").visible = xlSheetHidden
<b>Unhide Sheet</b>	Sheets("Sheet1").Visible = True or Sheets("Sheet1").Visible = xlSheetVisible
<b>Very Hide Sheet</b>	Sheets("Sheet1").Visible = xlSheetVeryHidden
<b>Delete Sheet</b>	Sheets("Sheet1").Delete
<b>Clear Sheet</b>	Sheets("Sheet1").Cells.Clear
<b>Unprotect (No Password)</b>	Sheets("Sheet1").Unprotect
<b>Unprotect (Password)</b>	Sheets("Sheet1").Unprotect "Password"
<b>Protect (No Password)</b>	Sheets("Sheet1").Protect
<b>Protect (Password)</b>	Sheets("Sheet1").Protect "Password"
<b>Protect but Allow VBA Access</b>	Sheets("Sheet1").Protect UserInterfaceOnly:=True



# Cells & Ranges

Description	VBA Code
<b>Activate Cell</b>	Range("B3").Activate Cells(3,2).Activate
<b>Select Range</b>	Range("a1:a3").Select Range(Range("a1"), Range("a3")).Select Range(Cells(1, 1), Cells(3, 1)).Select
<b>Resize</b>	Range("B3").Resize(2, 2).Select
<b>Offset</b>	Range("B3").Offset(2, 2).Select
<b>Copy</b>	Range("A1:B3").Copy Range("D1")
<b>Cut</b>	Range("A1:B3").Cut Range("D1")
<b>Delete</b>	Range("A1:B3").Delete Range("A1:B3").Delete shift:=xlShiftToLeft
<b>Clear</b>	Range("A1:A3").Clear Range("A1:A3").ClearContents Range("A1:A3").ClearFormat
<b>Count</b>	Range("A1:A3").Count
<b>Set to Variable</b>	Dim rng as Range Set rng = Range("A1")
<b>Merge/UnMerge</b>	Range("A1:A3").Merge Range("A1:A3").UnMerge
<b>Loop Through Cells</b>	Dim cell As Range For Each cell In Range("A1:C3") MsgBox cell.Value Next cell



## Rows

Description	VBA Code
<b>Activate</b>	Rows(1).Activate Rows("1:1").Activate Range("a1").EntireRow.Activate
<b>Height / Width</b>	Range("A1").EntireRow.RowHeight = 30
<b>Delete</b>	Range("A1").EntireRow.Delete
<b>Count</b>	Range("A1").Rows.Count
<b>Insert</b>	Range("A1").EntireRow.Insert
<b>Last</b>	dim IRow as long IRow = Cells(Rows.Count, 1).End(xlUp).Row
<b>Copy</b>	Range("1:1").Copy Range("5:5")
<b>Insert</b>	Range("1:1").Copy Range("5:5").Insert

## Columns

Description	VBA Code
<b>Activate</b>	Columns(1).Activate Columns("a:a").Activate Range("a1").EntireColumn.Activate
<b>Height / Width</b>	Range("A1").EntireColumn.ColumnWidth = 30
<b>Delete</b>	Range("A1").EntireColumn.Delete
<b>Count</b>	Range("A1").Columns.Count
<b>Insert</b>	Range("A1").EntireColumn.Insert
<b>Last</b>	dim ICol as long ICol = Cells(1, Columns.Count).End(xlToLeft).Column
<b>Copy</b>	Range("A:A").Copy Range("E:E")
<b>Insert</b>	Range("A:A").Copy Range("E:E").Insert



## Workbooks

Description	VBA Code
<b>Activate</b>	Workbooks("Book1").Activate
<b>Activate First Opened</b>	Workbooks(1).Activate
<b>Activate Last Opened</b>	Workbooks(Workbooks.Count).Activate
<b>Get ActivateWorkbook</b>	MsgBox ActiveWorkbook.Name
<b>Get ThisWorkbook (containing VBA Code)</b>	MsgBox ThisWorkbook.Name
<b>Add</b>	Workbooks.Add
<b>Add to Variable</b>	Dim wb As Workbook Set wb = Workbooks.Add
<b>Open</b>	Workbooks.Open("C:\example.xlsm")
<b>Open to Variable</b>	Dim wb As Workbook Set wb = Workbooks.Open("C:\example.xlsm")
<b>Close</b>	Workbooks("Book1").Close SaveChanges:=False Workbooks("Book1").Close SaveChanges:=True
<b>Save</b>	Workbooks("Book1").Save
<b>Save As</b>	Workbooks("Book1").SaveAs strFileName
<b>Protect/Unprotect</b>	Workbooks(1).Protect "password" Workbooks(1).Unprotect "password"
<b>Set to Variable</b>	Dim wb as Workbook Set wb = Workbooks("Book1")
<b>Loop Through All Workbook in Workbooks</b>	Dim wb As Workbook For Each wb In Workbooks MsgBox wb.Name Next wb
	Check Exists
<b>Check Exists</b>	If Dir("C:\Book1.xlsx") = "" Then MsgBox "File does not exist." EndIf
<b>Copy Closed</b>	FileCopy "C:\file1.xlsx", "C:\file2.xlsx"



# Settings

Description	VBA Code
<b>Screen Updating</b>	Application.ScreenUpdating = False Application.ScreenUpdating = True
<b>Display Alerts</b>	Application.DisplayAlerts = False Application.DisplayAlerts = True
<b>Events</b>	Application.EnableEvents = False Application.EnableEvents = True
<b>Enable Cancel Key</b>	Application.EnableCancelKey = xlDisabled Application.EnableCancelKey = xlInterrupt
<b>Text Compare – Ignore Case</b>	Option Compare Text
<b>Require Variable Declaration</b>	Option Explicit
<b>Automatic Calculations</b>	Application.Calculation = xlManual Application.Calculation = xlAutomatic
<b>Background Error Checking</b>	Application.ErrorCheckingOptions.BackgroundChecking = False Application.ErrorCheckingOptions.BackgroundChecking = True
<b>Display Formula Bar</b>	Application.DisplayFormulaBar = False Application.DisplayFormulaBar = True
<b>Freeze Panes</b>	ActiveWindow.FreezePanes = False ActiveWindow.FreezePanes = True
<b>Full Screen View</b>	Application.DisplayFullScreen = False Application.DisplayFullScreen = True
<b>PageBreak Preview</b>	ActiveWindow.View = xlPageBreakPreview ActiveWindow.View = xlNormalView
<b>Display Scroll Bars</b>	With ActiveWindow .DisplayHorizontalScrollBar = False .DisplayVerticalScrollBar = False End With With ActiveWindow .DisplayHorizontalScrollBar = True .DisplayVerticalScrollBar = True End With
<b>Display Status Bar</b>	Application.DisplayStatusBar = False Application.DisplayStatusBar = True
<b>Status Bar Contents</b>	Application.StatusBar = "I'm working Now!!!" Application.StatusBar = False
<b>Display Workbook Tabs</b>	ActiveWindow.DisplayWorkbookTabs = False ActiveWindow.DisplayWorkbookTabs = True
<b>UserName</b>	Application.UserName = "AutomateExcel.com"
<b>App Caption</b>	Application.Caption = "AutomateExcel Model"
<b>Zoom</b>	ActiveWindow.Zoom = 80



## Errors

Description	VBA Code
On Error – Stop code and display error	On Error Goto 0
On Error – Skip error and continue running	On Error Resume Next
On Error – Go to a line of code [Label]	On Error Goto [Label]
Clears (Resets) Error	On Error GoTo -1
Show Error number	MsgBox Err.Number
Show Description of error	MsgBox Err.Description
Function to generate own error	Err.Raise

## Files

Description	VBA Code
Copy File	FileCopy "C:\test\test_old.xlsx", "C:\test\test_new.xlsx"
Delete File	Kill "C:\test\example.xlsx"
Make Folder	MkDir "C:\test\"
Delete All Files From Folder	Kill "C:\test\" & "*.*)"
Delete Folder	Kill "C:\test\" & "*.*)" Rmdir "C:\test\"
Current Directory	strPath = CurDir()
ThisWorkbook Path	strPath = ThisWorkbook.Path
Loop Through All Files in Folder	strFile = Dir("C:\test" & "\*") Do While Len(strFile) > 0 Debug.Print strFile strFile = Dir Loop

## Arrays

Description	VBA Code
Create	Dim arr(1 To 3) As Variant arr(1) = "one" arr(2) = "two" arr(3) = "three"
Create From Excel	Dim arr(1 To 3) As Variant Dim cell As Range, i As Integer i = LBound(arr) For Each cell In Range("A1:A3") i = i + 1 arr(i) = cell.value Next cell
Read All Items	Dim i as Long For i = LBound(arr) To UBound(arr) MsgBox arr(i) Next i
Erase	Erase arr
Array to String	Dim sName As String sName = Join(arr, ".")
Increase Size	ReDim Preserve arr(0 To 100)
Set Value	arr(1) = 22





# Collections

Description	VBA Code
<b>Create</b>	Dim coll As New Collection coll.Add "one" coll.Add "two"
<b>Create From Excel</b>	Dim coll As New Collection Dim cell As Range For Each cell In Range("A1:A2") coll.Add cell.value Next cell
<b>Add Item</b>	coll.Add "Value"
<b>Add Item Before</b>	coll.Add "Value", Before:=1
<b>Add Item After</b>	coll.Add "Value", After:=1
<b>Read Item</b>	MsgBox coll (1)
<b>Count Items</b>	coll.Count
<b>Read All Items</b>	Dim item As Variant For Each item In coll MsgBox item Next item
<b>Remove Item</b>	coll.Remove (1)
<b>Remove All Items</b>	Set coll = New Collection



# Dictionaries

Description	VBA Code
<b>Required Reference</b>	Tools > References > Microsoft Scripting Runtime
<b>Create</b>	Dim dict As New Scripting.Dictionary dict.Add "" dict.Add ""
<b>Create From Excel</b>	Dim dict As New Scripting.Dictionary Dim cell As Range Dim key As Integer For Each cell In Range("A1:A10") key = key + 1 dict.Add key, cell.value Next cell
<b>Add Item</b>	dict.Add "Key", "Value"
<b>Change Value</b>	dict("Key") = "Value"
<b>Get Value</b>	MsgBox dict("Key")
<b>Check For Value</b>	If dict.Exists("Key") Then MsgBox "Exists" End If
<b>Remove Item</b>	dict.Remove ("Key")
<b>Remove All Items</b>	dict.RemoveAll
<b>Loop Through Items</b>	Dim key As Variant For Each key In dict.Keys MsgBox key, dict(key) Next key
<b>Count Items</b>	dict.Count
<b>Make Key Case Sensitive</b>	dict.CompareMode = vbBinaryCompare
<b>Make Key Case Insensitive</b>	dict.CompareMode = vbTextCompare