VBA Arrays, Collections and Dictionaries

STATIC ARRAY

Declare	Dim arr(0 To 5) As Long Dim arr(5) As Long
Reset all values	Erase arr

DYNAMIC ARRAY

Declare	Dim arr() As Long
Set size	ReDim arr(1 To 10)
Increase size of existing array	ReDim Preserve arr(1 To 10)
Set size to zero	Erase arr

DYNAMIC AND STATIC ARRAY

Assign a value	arr(1) = 56
Go through all items(For)	For i = LBound(arr) To UBound(arr) Debug.Print arr(i) Next i
Go through all items(For Each)	For Each v In arr Debug.Print v Next v

ARRAYS AND RANGES

Create variant array	Dim arr() As Variant
Read cell values to array	arr = Range("A1:Z2").Value
Write array values to Range	Range("A3:Z4").Value = arr

ARRAYS AND SUBS/FUNCTIONS

Use array parameter	Sub UseArray(ByRef arr()) As Long)
Pass to procedure	Dim arr(0 To 2) As Long UseArray arr

Use as return value	Function GetArray() As Long()
	  Dim arr(0 To 2) As Long
	  GetArray = arr
	End Sub

Return from procedure	Dim arr() As Long arr = GetArray

TWO DIMENSIONAL ARRAY

Create	Dim arr(0 To 2,0 To 4) As Long
Assign value	arr(0,0) = 45 arr(2,4) = 67

Read through array	Dim i As Long, j as Long
	For i = LBound(arr) To UBound(arr)
	  For j = LBound(arr, 2) To UBound(arr, 2)
	  Debug.Print arr(i, j)
	  Next j
	Next i

COLLECTION

Declare and create	Dim coll As New Collection
Declare and create in two steps	Dim coll As Collection Set coll = New Collection
Add item	coll.Add "Apple" coll.Add 55
Access item	Range("A1") = coll(1)
Remove item at index two	Coll.Remove 2
Go through all items (For)	For i = 1 To coll.Count Debug.Print coll(i) Next i
Go through all items(For Each)	For Each v In coll Debug.Print v Next v

DICTIONARY

Early binding reference	"Microsoft Scripting Runtime" (Add using Tools->References)
Declare(early binding)	Dim dict As Scripting.Dictionary
Create(early binding)	Set dict = New Scripting.Dictionary
Declare(late binding)	Dim dict As Object
Create(late binding)	Set dict = CreateObject("Scripting.Dictionary")
Add item. Key must not already exist.	dict.Add <key>,<value> dict.Add "Apples",50
Silent Add. Updates value if key exists.	dict(<key>)= value dict("Orange") = 67
Access item	value = dict("Apple")
Check if item exists	If dict.Exists("Apple") Then
Remove item	dict.Remove "Apple"
Remove all items	dict.RemoveAll
Go through all items	Dim key As Variant For Each key In dict.Keys Debug.Print key, dict(key) Next
Number of items	dict.Count
Make key non case sensitive. Dictionary must be empty.	dict.CompareMode = TextCompare