

Option Explicit
Option Base 1

Sub Understanding_Ranges()

Dim RR1 As Range

Set RR1 = ThisWorkbook.Worksheets("Sheet1").Range("A4:B6")

RR1.Value = 1

RR1.Interior.Color = vbRed

MsgBox RR1.Address

'MsgBox RR1.Address(0, 0)

'MsgBox RR1.Address(0, 0, xlR1C1) 'Relative to .Cells(1,1)

Dim RR2 As Range

Set RR2 = ThisWorkbook.Worksheets("Sheet1").Range(RR1.Address)

RR2.Name = "MyRange"

RR1.ClearContents

RR1.ClearFormats

End Sub

Sub Useful_Range_Properties_methods()

'--- Range.Cells

'--- Range.Offset

'--- Range.Resize

'--- Range.Rows.Count

'--- Range.Columns.Count

'--- Range.Worksheet.Name

'--- Range.Address

'Range.Cells :

'The return value is a Range consisting of single cells,

'which allows to use the version of the Item with two parameters

'and lets For Each loops iterate over single cells.

'Using Cells without an object qualifier is equivalent to ActiveSheet.Cells.

Dim RR1 As Range, RR2 As Range

Set RR1 = ThisWorkbook.Worksheets("Sheet3").Range("D5:D18")

Set RR2 = RR1.Cells(1, 1)

RR2.Select

Range("B2").Select

```
Set RR2 = RR1.Cells(0, 0)
RR2.Select
Range("B2").Select
```

'--- Range.Offset

```
Set RR2 = RR1.Cells(1, 1).Offset(1, 1)
RR2.Select
Range("B2").Select
```

'--- Range.Resize

```
Set RR2 = RR1.Cells(1, 1).Resize(1, 1)
RR2.Select
Range("B2").Select
```

```
Set RR2 = RR1.Cells(1, 1).Resize(1, 5)
RR2.Select
Range("B2").Select
```

```
Set RR2 = RR1.Cells(1, 1).Resize(5, 1)
RR2.Select
Range("B2").Select
```

'--- Range.Resize with Range.Rows.Count and Range.Columns.Count

```
Set RR2 = RR1.Cells(1, 1).Resize(RR1.Rows.Count, RR1.Columns.Count)
RR2.Select
Range("B2").Select
```

```
Set RR2 = RR1.Cells(1, 1).Offset(0, 1).Resize(RR1.Rows.Count, RR1.Columns.Count)
RR2.Select
Range("B2").Select
```

```
Set RR2 = RR1.Cells(1, 1).Offset(0, 1).Resize(RR1.Rows.Count, RR1.Columns.Count)
RR2.Select
Range("B2").Select
```

```
MsgBox "Row number is " & RR1.Row
MsgBox "Column number is " & RR1.Column
```

```
End Sub
```

```
Sub Selecting_Dynamic_Range_EndXlup()
```

'Range navigation

'One landmark known, The top left cell known

'We will find the last filled row and last filled column for the table.

'Assumptions :

'-- All cells are empty below the table of our concern

'-- The column adjacent right to the table is empty.

'Approach :

'-- Start from the bottommost (last row) and the column same as our landmark cell
'-- Using End(xlUp), we will mimic the cursor behaviour of going upto the next non empty cell

```
Dim wsh As Worksheet
Set wsh = ThisWorkbook.Worksheets("Sheet1")

Dim RR1 As Range
Set RR1 = wsh.Range("B4")

Dim LastRow As Long
Dim LastCol As Long

LastRow = wsh.Cells(Rows.Count, RR1.Column).End(xlUp).Row
MsgBox LastRow

LastCol = RR1.End(xlToRight).Column
MsgBox LastCol

With RR1.Cells(1, 1)
    Set RR1 = .Resize(LastRow - .Row + 1, LastCol - .Column + 1)
End With

MsgBox RR1.Address

RR1.Select

End Sub
```

```
Sub Selecting_Dynamic_Range_Find()
```

'Range Navigation

'One landmark known, The top left cell known
'We will find the last filled row and last filled column for the table.

'Assumptions :
'-- All cells are empty below the table of our concern
'-- The column adjacent right to the table is empty.

'Approach :
'-- Instead of start from the bottommost (last row) and the column same as our landmark cell
'-- We will use .Find() method to search for the last row with nonwhite character in a given column
'We used .Find() Method instead of EndXlUp which can give inaccurate results
'when we are navigating through a column that overlaps with a ListObject
'instead of a normal range

```
Dim wsh As Worksheet
Set wsh = ThisWorkbook.Worksheets("Sheet1")
```

```
Dim RR1 As Range
Set RR1 = wsh.Range("B4")
```

```
Dim LastRow As Long, LastRowB As Long
Dim LastCol As Long
```

```
'Find the last row using .Find() method
LastRow = wsh.Columns(RR1.Column).Find(What:="*", _
LookIn:=xlValues, _
SearchOrder:=xlByRows, _
SearchDirection:=xlPrevious).Row
```

```
MsgBox LastRow
```

```
'LastRowB = wsh.Columns(Range("B1").Column).Find(What:="*", LookIn:=xlValues,
SearchOrder:=xlByRows, SearchDirection:=xlPrevious).Row
'MsgBox LastRowB
```

```
'Find the Last column using End(xlToRight)
LastCol = RR1.End(xlToRight).Column
MsgBox LastCol
```

```
With RR1.Cells(1, 1)
    Set RR1 = .Resize(LastRow - .Row + 1, LastCol - .Column + 1)
End With
```

```
MsgBox RR1.Address
```

```
RR1.Select
```

```
End Sub
```

```
Sub Selecting_Dynamic_Range_Find2(ByRef RR1 As Range, ByVal LandmarkCell As String,
ByVal ShtName As String)
```

```
'Modular procedure to find the dynamic Range based on the topmost cell
```

```
'One landmark known, The top left cell known
```

```
'We will find the last filled row and last filled column for the table.
```

```
'Assumptions :
```

```
'-- All cells are empty below the table of our concern
```

```
'-- The column adjacent right to the table is empty.
```

```
'Approach :
```

```
'-- Instead of start from the bottommost (last row) and the column same as our landmark cell
```

```
'-- We will use .Find() method to search for the last row with nonwhite character in a given column
```

```
'We used .Find() Method instead of EndXIUp which can give inaccurate results
```

```
'when we are navigating through a column that overlaps with a ListObject
```

```
'instead of a normal range
```

```
Dim wsh As Worksheet
Set wsh = ThisWorkbook.Worksheets(ShtName)
```

```
Dim RR1 As Range
Set RR1 = wsh.Range(LandmarkCell)
```

```
Dim LastRow As Long, LastRowB As Long
Dim LastCol As Long
```

```
'Find the last row using .Find() method
LastRow = wsh.Columns(RR1.Column).Find(What:="", _
LookIn:=xlValues, _
SearchOrder:=xlByRows, _
SearchDirection:=xlPrevious).Row
```

```
MsgBox LastRow
```

```
'LastRowB = wsh.Columns(Range("B1").Column).Find(What:="", LookIn:=xlValues,
SearchOrder:=xlByRows, SearchDirection:=xlPrevious).Row
'MsgBox LastRowB
```

```
'Find the Last column using End(xlToRight)
LastCol = RR1.End(xlToRight).Column
MsgBox LastCol
```

```
With RR1.Cells(1, 1)
    Set RR1 = .Resize(LastRow - .Row + 1, LastCol - .Column + 1)
End With
```

```
MsgBox RR1.Address
```

```
RR1.Select
```

```
End Sub
```

```
Sub NumberFormatting()
```

```
Dim LandmarkCell As String
LandmarkCell = "B4"
ShtName = "Sheet1"
```

```
Dim wsh As Worksheet
Set wsh = ThisWorkbook.Worksheets("Sheet1")
```

```
Dim RR1 As Range
Call Selecting_Dynamic_Range_Find2(RR1, LandmarkCell, ShtName)
```

```
RR1.Cells(1, 3).Resize(RR1.Rows.Count, 1).NumberFormat = "0.00"
RR1.Cells(1, 3).Resize(RR1.Rows.Count, 1).NumberFormat = "#,##0"
```

```
End Sub
```

```
Sub demo()
```

```
Dim DT As Date
```

DT = Date

With Range("a1")

.NumberFormat = "@"

.Value = Format(DT, "mmmm yyyy")

End With

ActiveSheet.Cells(1, 2) = Date

ActiveSheet.Cells(1, 2).NumberFormat = "mm/dd/yyyy"

NumberFormat = "dd/mm/yy;@"

End Sub