

# Memento C3 infodauphine.com: Objets et cellules

## Cibler une cellule de la feuille active

`Cells( numeroLigne, numeroColonne )` → type `Range`

Exemple : `Cells(3,1).Value = 10` → la valeur de `L3C1` devient 10

## Cibler une plage de la feuille active

`Range( Cellule1 As Range, Cellule2 As Range )` → type `Range`

`Range( Cells( numLigne1, numColonne1 ), Cells( numLigne2, numColonne2 ) )`

Exemple : `Range( Cells(3,1) , Cells(5,3) ).Value = 10` → toute la plage `L3C1:L5C3` prend la valeur 10

`Range( Cells(1,1) , Cells(5,1) ).Value = Range( Cells(1,2) , Cells(5,2) ).Value`

→ on copie les valeurs de `LIC2:L5C2` dans la plage `LIC1:L5C1` (les valeurs sont en double du coup)

Une **boucle** est toujours **superflue** pour un traitement **identique** sur plusieurs cellules d'une plage.

Les **boucles** sont **nécessaires** si le traitement dépend d'une **condition** ou d'un calcul

## Parcourir les cellules

Tous les exemples testent si la cellule est négative dans une plage `L1C1:L4C5` (donc 4 lignes, 5 colonnes)

**Rappel:** utiliser `For nl` (`nl` comme **num**éro de **l**igne) et `For nc` (**num**éro de **c**olonne) évite beaucoup d'erreurs.

<b>Parcourir la ligne 3</b> (donc de <code>L3C1</code> à <code>L3C5</code> )	<b>Parcourir la colonne 2</b> (donc de <code>LIC2</code> à <code>L4C2</code> )
<pre>Dim nc As Integer ... For nc = 1 To 5   If Cells(3, nc).Value &lt; 0 Then     ...   End If Next nc</pre>	<pre>Dim nl As Integer ... For nl = 1 To 4   If Cells(nl, 2).Value &lt; 0 Then     ...   End If Next nl</pre>
<b>Parcourir les cellules de chaque colonne</b> (dans <code>LIC1:L4C5</code> )	<b>Parcourir toute la plage avec une seule boucle</b> (dans <code>LIC1:L4C5</code> )
<pre>Dim nl As Integer, nc As Integer ... For nc = 1 To 5   [initialisations variables par colonne]   For nl = 1 To 4     If Cells(nl, nc).Value &lt; 0 Then       ...     End If   Next nl Next nc</pre>	<pre>Dim c As Range ... For Each c In Range( Cells(1,1) , Cells(4,5) ).Cells   If c.Value &lt; 0 Then     ...   End If Next c</pre>
<b>Parcourir la ligne 3 tant que la cellule n'est pas vide</b>	<b>Parcourir la colonne 2 tant que la cellule n'est pas vide</b> (à partir de <code>LIC2</code> )
<pre>Dim nc As Integer ... nc=1 Do While Cells(3,nc).Value &lt;&gt; ""   If Cells(3, nc).Value &lt; 0 Then     ...   End If   nc = nc + 1 Loop</pre>	<pre>Dim nl As Integer ... nl = 1 Do While Cells(nl,2).Value &lt;&gt; ""   If Cells(nl, 2).Value &lt; 0 Then     ...   End If   nl = nl + 1 Loop</pre>

<b>Formatage d'un objet <code>Range</code></b> ( <i>très importants</i> ) <ul style="list-style-type: none"> <li><code>.Interior.Color = RGB(rouge, vert, bleu)</code> → couleur de la cellule</li> <li><code>.Font.Bold = true</code> → gras</li> <li><code>.Font.Italic = true</code> → italique</li> <li><code>.Font.Color = RGB(rouge, vert, bleu)</code> → couleur du texte</li> </ul>	<b>Vidage d'un objet <code>Range</code></b> <ul style="list-style-type: none"> <li><code>.Value = ""</code> → vide la valeur, équivalent à:</li> <li><code>.ClearContents</code> → vide la valeur (et formule)</li> <li><code>.Clear</code> → efface aussi mise en forme</li> <li><code>.ClearFormats</code> → efface juste mise en forme</li> </ul>
---	--

## Autres propriétés d'un objet `Range`

(en gras celles importantes)

<code>.Value</code> (Variant)	La valeur	Si la cellule <code>L1C2</code> est vide (attention pas d'espace entre les " )	<code>If Cells(1,2).Value = "" Then</code>
<code>.Formula</code> (String)	La formule	<code>.FormulaR1C1Local</code> (String)	La formule références L1C1
<code>.Row</code> (Long)	N° de ligne	<code>.Column</code> (Long)	N° de colonne
<code>.Count</code> (Long)	Nombre de cellules	<code>.Cells</code> (Range)	Collection des cellules
<code>.Rows</code> (Range)	Collection des lignes	<code>.Columns</code> (Range)	Collection des colonnes

## Règles notation objet

Je peux mettre un **point** et une **propriété** après un **objet** (objets `Range` grâce à `Cells(nl, nc)` ou `Range(...)`, propriétés `.Font` et `.Interior`, ...), mais **jamais** après une **valeur** (`.Value`, `.Bold`, `.Color`, ...).

Exemple: `Cells( 3,1).Font.Bold = Cells(2,1).Font.Italic`

Je peux mettre **des parenthèses** après une propriété **collection** (`.Cells`, `.Rows`, `Worksheets`, ...) ou une **méthode fonction** (`.Range`, `.InputBox`, ...)

Retrouve le cours complet, avec exemples exécutables étape par étape, dans **EasyVBA**

[www.infodauphine.com](http://www.infodauphine.com)