

Programmation Excel/VBA

VBA ET LES OBJETS EXCEL

MASTER SES

UNIVERSITÉ NANTERRE LA DÉFENSE
SANA BEN HAMIDA

Définitions

2

Un objet est une instance d'une classe

Un objet est caractérisé par :

Un nom : son nom de variable

Un type : appelé également *classe de l'objet*
Le type de base des objets est Object.

Des propriétés : Ce sont des caractéristiques de l'objet
(la couleur, la taille, l'état, etc.).

Chaque propriété possède un type (Boolean, String, Object, ...).

*Si une propriété d'un objet est de type Object, on dit qu'il y a eu **composition**.*

Des méthodes Elles sont communes à tous les objets
de ce type/de cette classe.

- Elles permettent de faire effectuer une action à l'objet

Définitions

3

Accès au propriétés et méthodes

Pour accéder aux propriétés ou aux méthodes d'un objet, il faut utiliser la notation.

unObjet.unePropriete
unObjet.uneMethode(param1, ..., paramk)

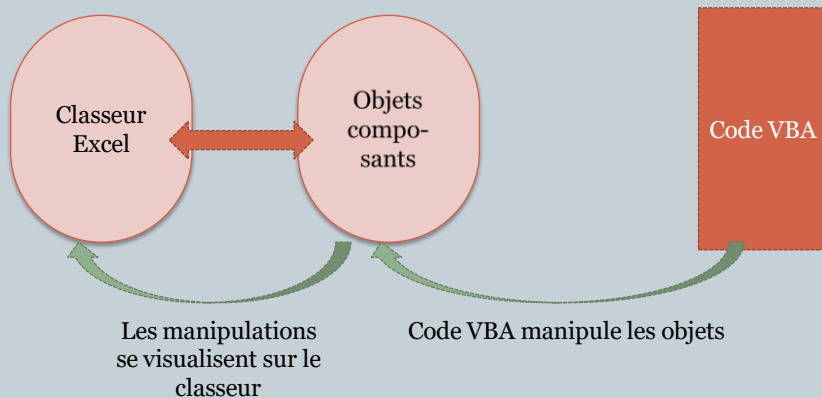
Exemple:

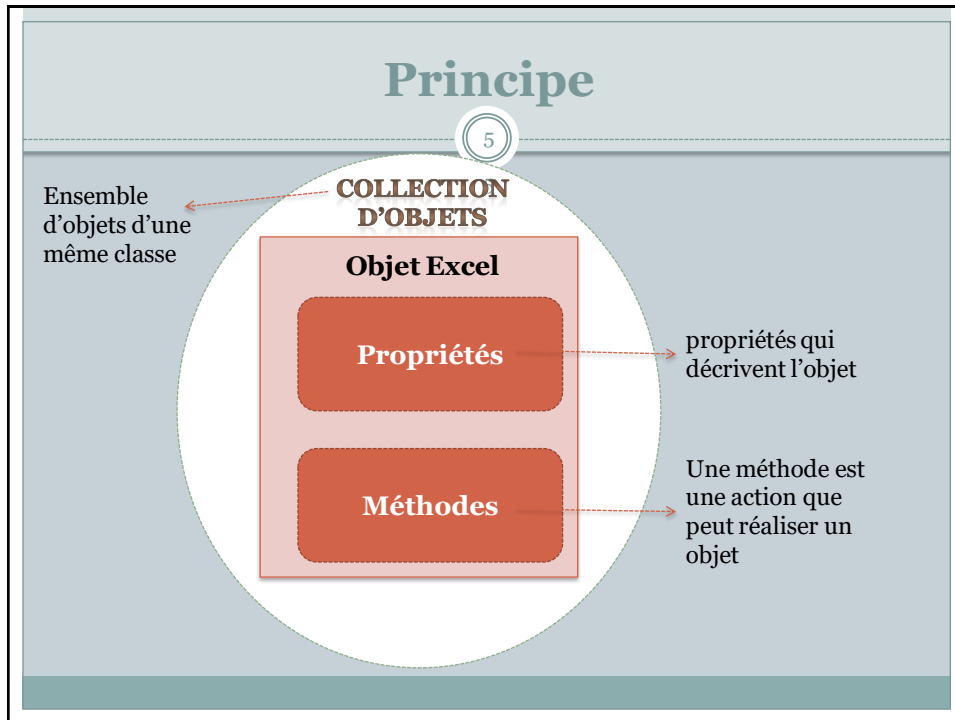
Cells(1,1).value ← propriété

Cells(1,1).select() ← méthode

Principe

4





Principe

6

Le nom d'une collection est généralement le type de ses objets auquel on ajoute un « **s** »

Exemples

- Workbook → Workbooks : les classeurs Excel
- WorkSheet → Worksheets
- Sheet → Sheets : collection des feuilles de calcul d'un classeur
- Cell → Cells : collection des cellules d'une feuille ou d'une plage dans une feuille
 - Column → Columns : collection des colonnes d'une plage
 - Row → Rows : collection des lignes d'une plage

Chemin d'accès

7

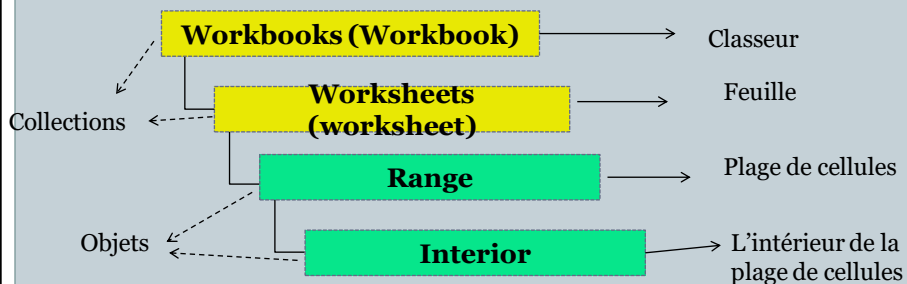
- La hiérarchie des objets détermine le chemin à emprunter pour accéder à un objet donné.
- Le point (.) est utilisé comme séparateur entre les différentes collections et objets.

<Nom collection>.(« Nom Objet 1 »).(« Nom Objet 2 »)....<Nom méthode ou propriété>

Chemin d'accès

8

- Exemple : Chemin d'accès jusqu'à l'intérieur d'une cellule



Chemin d'accès

9

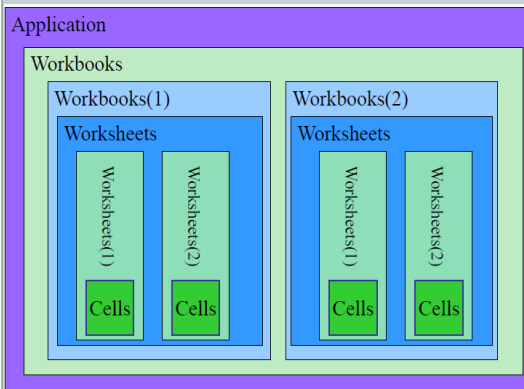
- Exemple :

On souhaite colorer en gris l'intérieur (**Interior**) de la cellule B2 (**Range("B2")**) de l'objet **worksheet** "Feuil3" de la collection **Worksheets** du classeur en cours d'utilisation (**ActiveWorkbook**):

```
ActiveWorkbook.Worksheets(« Feuil3 »).Range(« B2 »).Interior.Color = vbgrey
```

Hiérarchie des Objets

10



→ Dans l'objet **Application**, il existe une collection **Workbooks** qui contient tous les objets **Workbook** ouverts. Chaque objet **Workbook** comporte une collection **Worksheets** qui contient tous les objets **Worksheet** de ce classeur. Chaque objet **Worksheet** contient des objets **Cells**, **Rows** et **Columns**.

Les principaux collections/objets Excel

(A retenir)

11

Workbooks	représente les classeurs Excel
Worksheets/Sheets	pour les feuilles du classeur
Range	pour une plage de cellules
Cells	pour les cellules d'une feuille
Rows	Collection des lignes d'une plage
Colums	Collection des colonnes d'une plage

Méthodes et propriétés des objets

12

- Une propriété ou une méthode d'un objet ne peut être désignée indépendamment de l'objet qui le contient:

« nom objet ».propriété

- **Exemple:**

- **ActiveSheet.Name**

➔ désigne la propriété Name de l'objet ActiveSheet

- **ActiveSheet.Name=" mafeuille"**

➔ Modifie le nom de la feuille active

Méthodes et propriétés des objets

13

- Quelques propriétés utiles

Propriété	Rôle
Name	Nom du classeur ou de la feuille
Value	Valeur d'une cellule
Visible	Objet visible ou masqué (boolean)
Caption	Le texte d'un objet de contrôle
.....	

Exemple:

- ▶ Cells(1,1).Value = 10 → inscrit 10 dans la cellule A1
- ▶ Sheets("Feuil1").Visible = False → Masquer une feuille.

Méthodes et propriétés des objets

14

- Agir sur les objets (multitude de méthodes)
- Quelques exemples sur les objets « Sheets »:

Méthode	Type d'objets	effet
Add	classeur	Ajouter une feuille
Activate	Classeur/feuille Cellule/plage	Activer un objet
Select	Feuille/cellule/ plage	Sélectionner l'objet
...		

Les collections

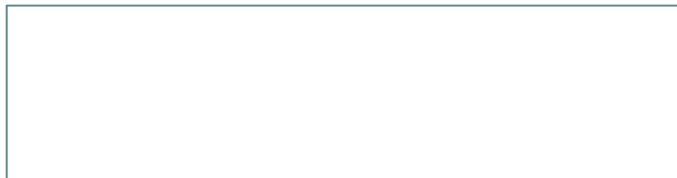
15

Exemples:

- *Sheets.Count* désigne le nombre de feuilles du document actif
- *Sheets(3)* désigne la troisième feuille de la collection des feuilles du document actif
- *Sheets("Feuil2")* désigne la feuille intitulée Feuil2 de la collection des feuilles du document actif
- *Sheets.Add* ajoute une nouvelle feuille au document actif (appel de méthode)

L'objet Range

16



Définition

17

Range → Un objet permettant la définition de cellules et les plages de cellules (objet essentiel dans la programmation VBA).

Déclaration

Dim maPlage As Range

L'objet **maPlage** peut recevoir une cellule ou une plage de cellule. Ex:

Set MaPlage = ActiveSheet.Cells(1,1)

Manipulation des objets Range

18

La manipulation des objets Range se fait principalement en deux étapes:

1- Définition de la plage de l'objet Range

A partir d'une propriété de la collection *Worksheets*

Par sélection de cellules

2-Application des modifications

Définition de l'objet Range

(A partir d'une propriété de la collection *Worksheets*)

19

- **Range** → renvoie une cellule ou une plage de cellules
- **Cells** → renvoie l'ensemble des cellules ou des cellules spécifiques d'une feuille
- **Rows** → renvoie l'ensemble ou une sélection de lignes d'une feuille
- **Columns** → renvoie l'ensemble ou une sélection de colonnes d'une feuille
- **Selection** → renvoie les cellules (contiguës ou non) sélectionnées dans la feuille en cours

Définition de l'objet Range

(Par sélection de cellules)

20

- Deux méthodes:
 - **Select**: sélectionner un objet Range
 - **Activate**: activer un objet Range
- Deux techniques de sélection
 - Par **adressage direct** (en spécifiant l'adresse de la plage à atteindre)
 - Par **adressage indirect** (l'objet visé est atteint à partir d'une cellule ou une plage initiale)

Définition de l'objet Range

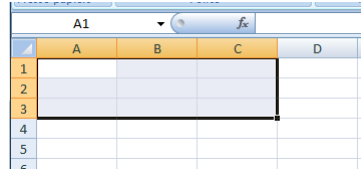
(Par sélection de cellules)

21

- **Sélection par adressage direct** en utilisant la propriété *Range*

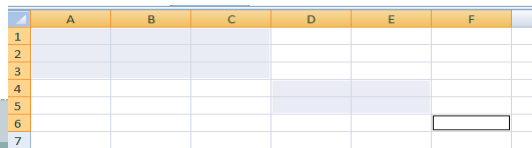
- `Range("A1").select` → sélectionne la cellule A1

- `Range("A1:C3").select` →



	A	B	C	D
1				
2				
3				
4				
5				
6				

- `Range("A1:C3, D4:E5, F6").Select`



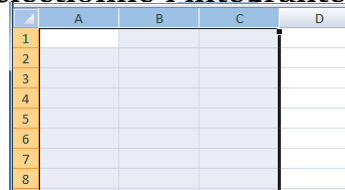
	A	B	C	D	E	F
1						
2						
3						
4						
5						
6						
7						

Définition de l'objet Range

(Par sélection de cellules)

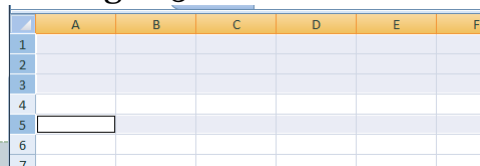
22

- `Range("A:C").select` → sélectionne l'intégralité des colonnes A à C



	A	B	C	D
1				
2				
3				
4				
5				
6				
7				
8				

- `Range("1:3,5:5").select` → sélectionne l'intégralité des lignes 1 à 3 et la ligne 5



	A	B	C	D	E	F
1						
2						
3						
4						
5						
6						
7						

Définition de l'objet Range

(Par sélection de cellules)

23

- **Sélection par adressage direct** en utilisant la propriété **Cells**
- **Cells.select** → sélectionne l'intégralité des cellules de la feuille
- **Cells(16).select** → sélectionne la 16^{ème} cellule (P1)
- **Cells(2,3).select** → sélectionne la cellule C2 (intersection 2^{ème} ligne et 3^{ème} colonne)



Une feuille de calcul est composée de 16 383 colonnes et 1 048 576 lignes. Les cellules sont numérotées de gauche à droite, et de haut en bas:

Cells(16 384) → A2

Définition de l'objet Range

(Par sélection de cellules)

24

- **Sélection par adressage direct** en utilisant les propriétés **Rows** et **columns**
- **Rows.select** → sélectionne l'intégralité des lignes de la feuille (toute la feuille)
- **Columns.select** → sélectionne l'intégralité des colonnes de la feuille (toute la feuille)
- **Rows(1).select** → sélectionne la 1^{ère} ligne
- **Columns("A").select** → sélectionne la 1^{ère} colonne
- **Rows(1:3).select** → sélectionne les ligne 1 à 3
- **Columns("A:C").select** → sélectionne les colonnes A à C.

Manipulation de l'objet Range

25

1. Quelques propriétés et méthodes de l'objet Range
2. Stockage de valeurs et de formules

Manipulation de l'objet Range

1. Quelques propriétés et méthodes de l'objet Range

26

- ❑ **Valeur d'une plage:** La propriété par défaut de l'objet Range est sa valeur (*value*)
`MaVariable = range("A1")` 'est équivalent à
`MaVariable = Range("A1").value`
- ❑ **Nom d'une plage :**
`Range ("A1:C3").Name=" MaPlage"`
- ❑ **Nombre de lignes d'une plage:**
`Range ("A1:C3").Rows.count`
- ❑ **Nombre de colonnes d'une plage:**
`Range ("A1:C3").columns.count`
- ❑ **Nombre de cellules d'une plage**
`Range ("A1:C3").cells.count`

Manipulation de l'objet Range

1. Quelques propriétés et méthodes de l'objet Range

27

❑ Mise en forme d'une plage:

`Range("A1").Interior.ColorIndex = 3` → colorer en rouge la cellule A1

`Range("A1").font.bold = true` → activation du font gras

`Range("A1").font.italic = true` → activation du font italic

❑ Effacer le contenu d'une plage

`Range("A19").ClearContents`

Manipulation de l'objet Range

2. Stockage de valeurs et de formules

28

❑ Stockage de valeur

<Objet Range>.Value = <Expression>

Exemple:

`Worksheets("Feuil1").Range("A1").Value = "Pi : "`

`Worksheets("Feuil1").Range("A2").Value = 3.14159`

Manipulation de l'objet Range

2. Stockage de valeurs et de formules

29

□ Stockage de formule

<Objet Range>.Formula = <formule>

Exemple:

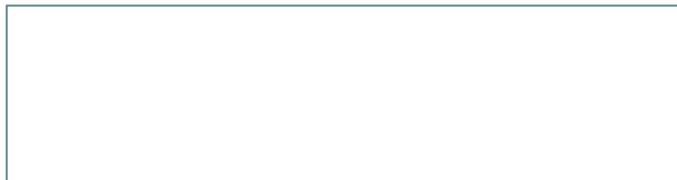
Range("C2").Formula= "=A1"

Range("A2").Formula= "AVERAGE(B1:B10) "

Range("A3").Formula= "STDEV(B1:B10) "

Exemples

30



Exemple 1 : mise en forme

31

```

Sub MiseEnForme()
Dim maplage As Range
Dim mafeuille As Worksheet
Set mafeuille = Sheets(1) 'Sélection de la 1ère feuille du classeur
mafeuille.Activate
Set maplage = mafeuille.Range("A2:A5")
maplage.Font.Size = 18
maplage.Interior.ColorIndex = 5
maplage.Value = "Exemple 1"

'maplage.ClearFormats
'maplage.ClearContents

End Sub

```

	A	B	C
1			
2	Exemple 1		
3	Exemple 1		
4	Exemple 1		
5	Exemple 1		
6			

Exemple 2: parcours d'une feuille

32

```

Sub Rayure()
Dim maplage As Range
Dim mafeuille As Worksheet
Dim i As Integer
Set mafeuille = Sheets(2)
mafeuille.Activate
Set maplage = mafeuille.Cells 'maplage contient toutes les
cellules de la feuille
For i = 1 To maplage.Columns.Count Step 2 'parcours de toutes
les colonnes
maplage.Columns(i).Interior.Color = vbBlue
Next
End Sub

```


Exemple 3: parcours d'une plage à une dimension

33

```

Sub Parcours1()
  Dim maplage As Range
  Dim mafeuille As Worksheet
  Dim i As Integer
  Set mafeuille = Sheets(3)
  mafeuille.Activate
  Set maplage = mafeuille.Range("A2:G2")
  For i = 1 To maplage.Count
    maplage.Cells(i).Value = "Cel" & i
    maplage.Cells(i).Font.Size = 14 + i
  Next
End Sub

```

	A	B	C	D	E	F	G	H
1								
2	Cel1	Cel2	Cel3	Cel4	Cel5	Cel6	Cel7	
3								
4								

```

For i = 1 To plage.Cells.Count
  s = s + plage.Cells(i) 'ajout de la ième température
  If plage.Cells(i) > Max Then
    Max = plage.Cells(i)
  End If
Next
'calcul moyenne
Range("E4") = s / plage.Cells.Count 'moyenne
Range("E5") = Max

```