

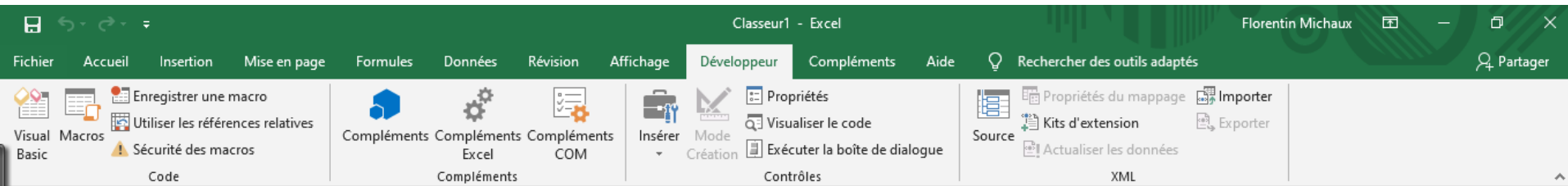


florentin.michaux@univ-lorraine.fr

06. Initiation au Visual Basic (VBA) sous Excel

1. Premières fonctions
2. Boucles : calcul factoriel
3. Première macro
4. Exemple bonus

Installation de l'outil "développeur"

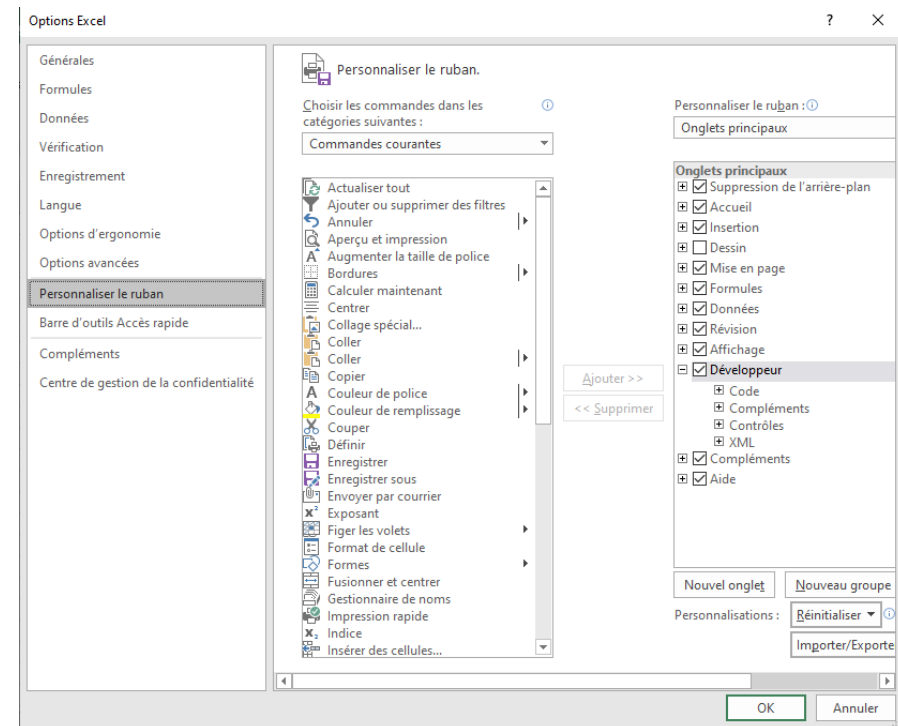


L'onglet **développeur** ne s'affiche pas par défaut, mais vous pouvez l'ajouter au ruban.

1. Dans l'onglet **fichier**, accédez à **options** > **personnaliser le ruban**.
2. Sous **Personnaliser le ruban** et **Onglets principaux**, activez la case à cocher **Développeur**.

L'onglet **développeur** est l'endroit où vous voulez faire ou utiliser ce qui suit :

- Écrire des macros.
- Exécuter des macros enregistrées précédemment.



Installation de l'outil "développeur"

Accès à l'environnement de programmation visual basic

The image shows the Microsoft Excel interface with the 'Développeur' (Developer) ribbon selected. A yellow arrow points to the 'Visual Basic' button on the ribbon. Below the ribbon, the 'Microsoft Visual Basic pour Applications' window is open, displaying the 'TD_07_initiation_VBA.xlsx - Module1 (Code)' window. The 'Projet - VBAProject' pane on the left shows a tree view of the project, including 'atpvbaen.xls (ATPVBAEN.XLAM)', 'Solver (SOLVER.XLAM)', 'VBAPProject (FUNCRES.XLAM)', 'VBAPProject (sigmaplot.xla)', 'VBAPProject (spremove.xla)', and 'VBAPProject (TD_07_initiation_VBA.xlsx)'. The 'Module1' is highlighted under the 'Modules' folder. Red text annotations explain the project structure and the code editor.

Projets associés à d'autres applications (ne pas s'en soucier !!)

Projet associé à votre fichier excel

Module créé dans le projet (clic droit, insertion → module)

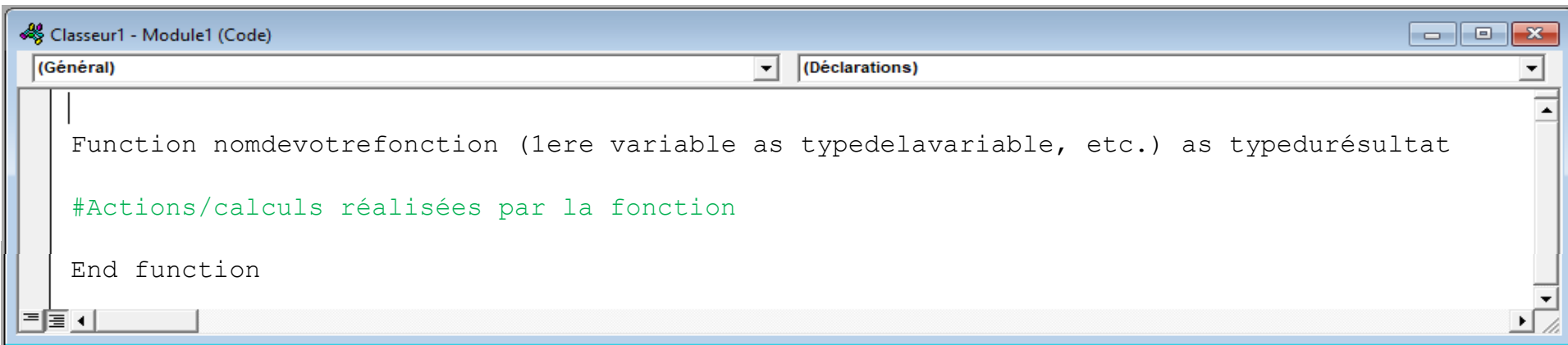
Taper votre code dans cette fenêtre

1. Premières fonctions

a. Créer deux fonctions de conversion permettant de convertir des °C en °F (CtoF) et inversement (FtoC).

Pour rappel, $T(^{\circ}\text{C}) = \frac{5}{9} (T(^{\circ}\text{F}) - 32)$ et $T(^{\circ}\text{F}) = \frac{9}{5} T(^{\circ}\text{C}) + 32$

1. Dans votre module, démarrer votre programmation par :



```
Function nomdevotrefonction (1ere variable as typedelavARIABLE, etc.) as typedurésultat

#Actions/calculs réalisées par la fonction

End function
```

Les différents types de variables :

VBA propose the following type of variables:

► NUMERICAL

- **Integer** (from -32768 to 32767) 
- **Long** (from -2147483648 to 2147483647) 

- **Single** (from -3.402823 10³⁸ to 3.402823 10⁻³⁸) 
- **Double** (from -1.79769313486232 10³⁰⁸ to 1.79769313486232 10⁻³⁰⁸) 

► CHARACTER

- **String** (using " " as delimiters of the character string) 

► BOOLEAN

- **Boolean** (with only two possible values : True or False) 

► OTHER

- **Variant** any kind of content
- **Object** Windows Object (presented later) 

1. Premières fonctions

2. Dans votre fichier excel associé, appeler la fonction dans une cellule par "`=lenomdevotrefonction`" et appliquer la à une valeur numérique placée dans une autre cellule tel que vous le feriez pour n'importe quelle fonction de base d'Excel



B	C
°C	°F
90	=CtoF(B2)

B	C
°C	°F
90	194
=FtoC(C3)	150

1. Premières fonctions

b. Créer une fonction conversion prenant 2 arguments, la valeur à convertir (valeur numérique) et son unité (chaîne de caractère) en utilisant les fonctions précédentes et qui renvoie un message d'erreur en cas de mauvais arguments.



=conversion(B2;"C")

Fonction conversion convertissant la température donnée dans la cellule B2 qui est en °C en °F

°C	°F
90	194
=conversion(B2;"C")	

1. Il sera nécessaire de tester une condition sur le 2^{ème} argument → utilisation d'une commande "If Then Else End If"

2. Le message d'erreur peut être affiché via la fonction *MsgBox* ""

3. Ne pas oublier de bien soigner la forme et les commentaires

4. Enregistrement du fichier excel en .xlsm

```
Classeur1 - Module1 (Code)
(Général)

If delta > 0 Then
    x1 = (-b - Sqr(delta))/(2*a)
    x2 = (-b + Sqr(delta))/(2*a)
Elseif delta = 0 Then
    x1 = (-b)/(2*a)
    x2 = x1
Else
    MsgBox ("Pas de solution")
End If
```

3. Boucles

Les boucles *Faire pour i valant de a jusqu'à b [avec un pas c]* se codent de la façon suivante :

```
For i=a to b Step c
...
Next
```

Il existe aussi des boucles *Faire ... jusqu'à ce que a=b* qui se codent ainsi :

```
Do
...
Loop Until a=b
```

Et enfin, les boucles *Faire ... tant que a=b* s'écrivent :

```
Do while a=b
...
Loop
```

3. Boucles : calcul de factorielle

La factorielle $n!$ est le produit de tous les nombres entiers positifs non nuls inférieurs ou égaux à n

$$5! = 1*2*3*4*5 = 120$$

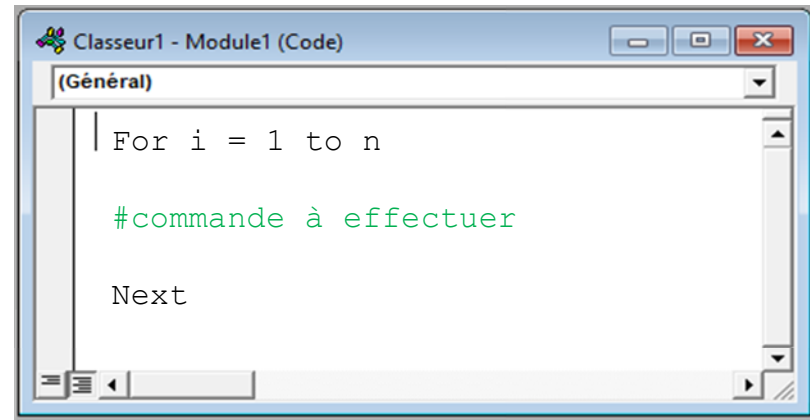


Calcul répétitif parfait pour l'utilisation d'une boucle !

1. Le calcul sera réalisé "pas à pas" (calcul itératif) de 1 jusqu'à n et s'arrêtera lorsque n sera égal à 5.
2. Le pas d'itération sera de 1, la première valeur de l'itération sera 1, la dernière n



Utilisation d'une boucle For



```
Classeur1 - Module1 (Code)
(Général)
For i = 1 to n
    #commande à effectuer
Next
```

La boucle For a l'avantage de faire automatiquement l'incrémentation de la variable d'un entier avec la commande next. Dans les autres boucles, il faut coder l'incrémentation avec la commande $i = i + x$ (avec x la valeur souhaitée).

3. Boucles : calcul de factorielle

3. *Le calcul à réaliser et le suivant* : Itération sur i

i = 1 : résultat = $1 * 1 = 1$

i = 2 : résultat = $1 * 2 = 2$

i = 3 : résultat = $2 * 3 = 6$

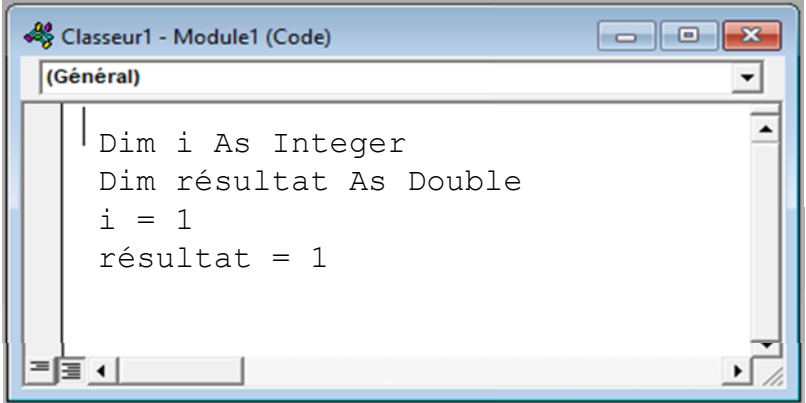
i = 4 : résultat = $6 * 4 = 24$

i = 5 (= n) résultat = $24 * 5 = 120$

Le calcul consiste à partir de 1 et à chaque itération, de multiplier la valeur "actuelle" par l'entier suivant.

➡ Pour que la fonction fonctionne, elle nécessite l'utilisation de 2 variables (lieu de stockage de valeurs (valeur numérique dans ce cas)).

➡ Déclarations et initialisations des variables "i" et "résultat"
(la variable "i" n'est en réalité par nécessaire car elle est déclarée et initialisée automatiquement dans la boucle For)

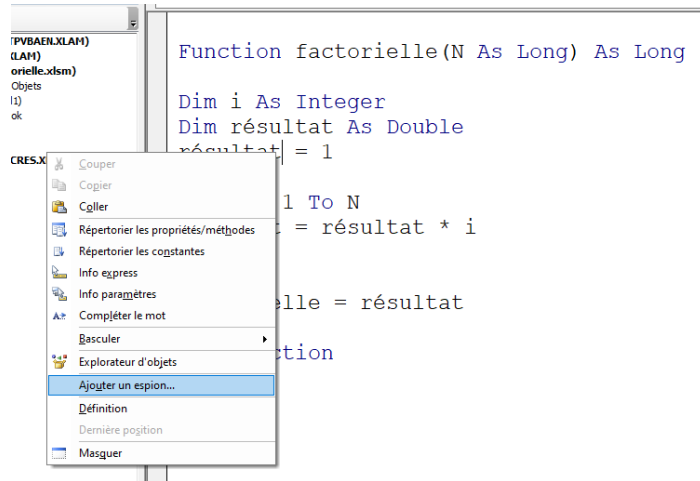


```
Classeur1 - Module1 (Code)
(Général)
Dim i As Integer
Dim résultat As Double
i = 1
résultat = 1
```

3. Boucles : Astuce pour suivre l'évolution des variables

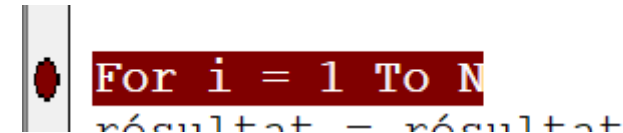
Astuce pour suivre le calcul effectué lors de la boucle pas à pas en suivant les valeurs des variables

1. Faire un click droit sur la variable à suivre → sélectionner ajouter un espion (apparaît dans la fenêtre espion)



Espions	
Expression	Valeur
résultat	2

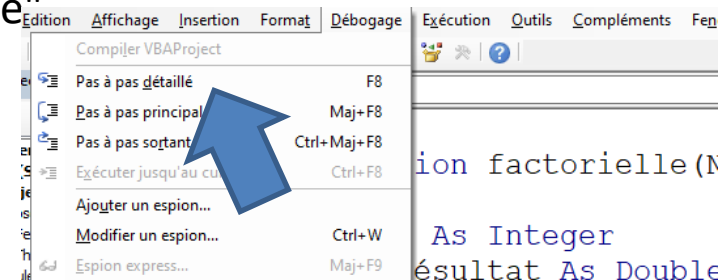
2. Ajouter un arrêt dans la colonne à gauche de votre code



3. Sélectionner "pas à pas détaillé" dans l'onglet "débogage"

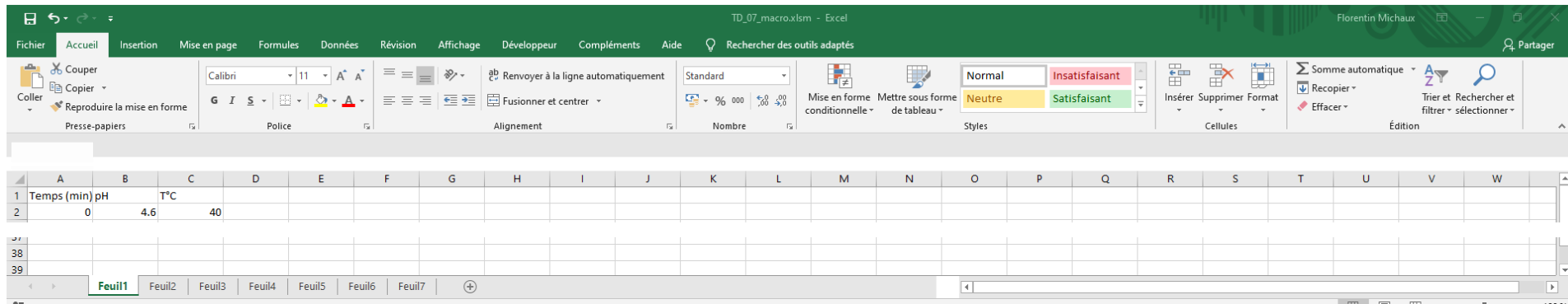
4. Exécuter votre fonction dans le tableur

5. Appuyer sur F8 pour faire avancer votre programme et suivre l'évolution de vos variables

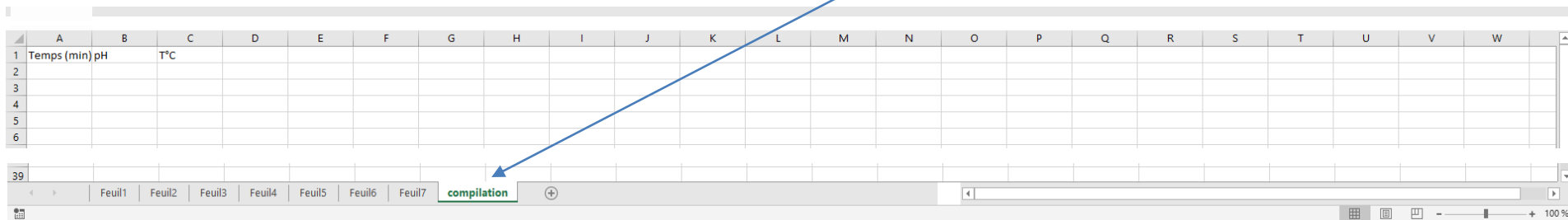


4. Première macro

Vous disposez d'un enregistreur automatique relié à un réacteur relevant toutes les 10 minutes le pH et la température d'un milieu réactionnel. A chaque nouvel enregistrement, une nouvelle feuille est créée dans un fichier Excel tel que :



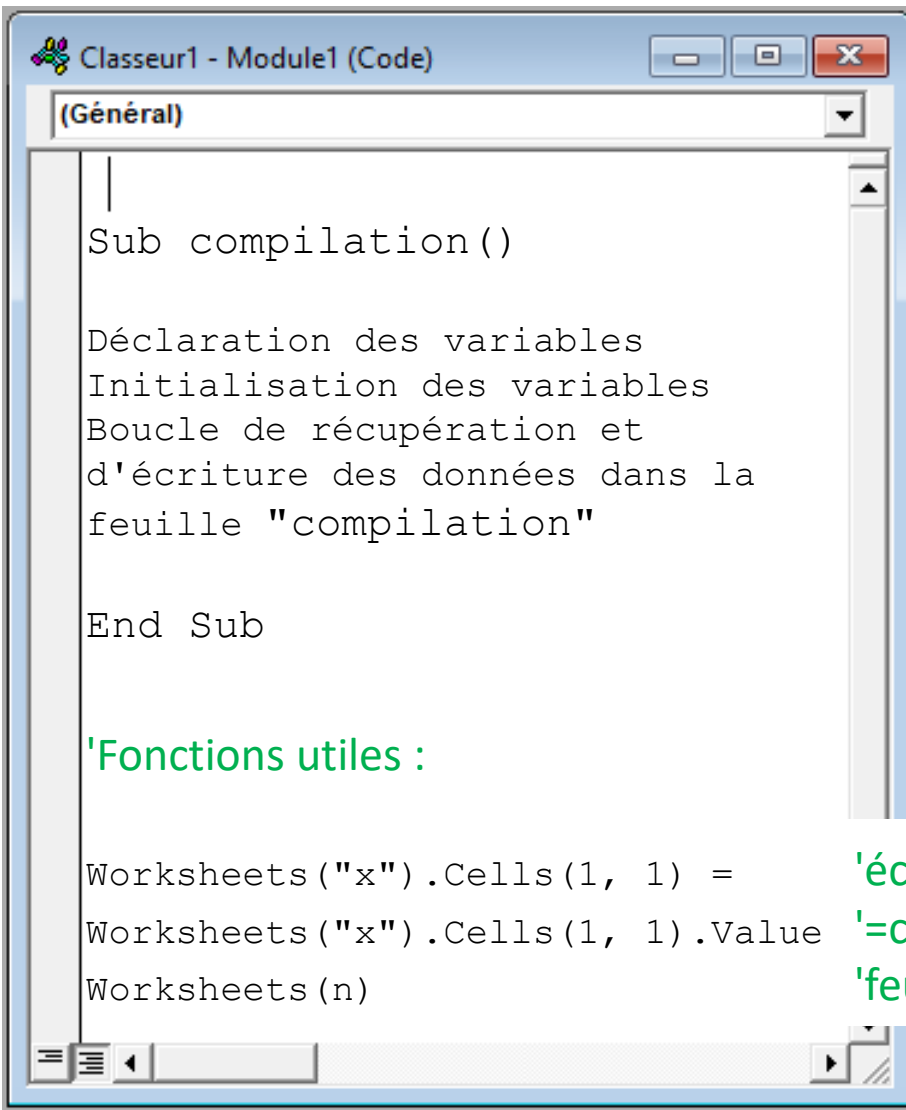
Vous avez ainsi 7 enregistrements disponibles et vous voulez les récupérer sous forme d'un tableau dans une feuille déjà créée et baptisée "compilation" tel que :



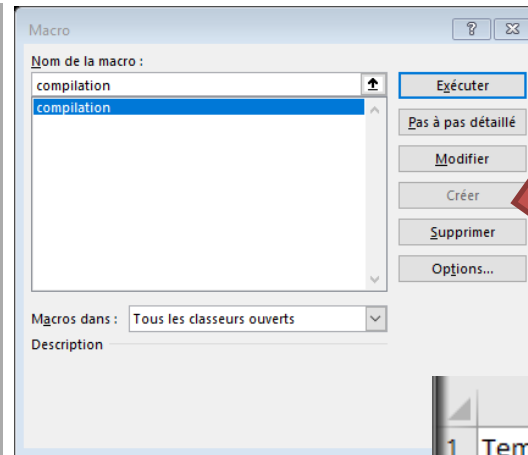
Créer la macro "compilation" permettant de faire cette opération automatiquement

4. Première macro

Lancement de la macro (développeur → macros)



```
Sub compilation()  
  
Déclaration des variables  
Initialisation des variables  
Boucle de récupération et  
d'écriture des données dans la  
feuille "compilation"  
  
End Sub  
  
'Fonctions utiles :  
  
Worksheets("x").Cells(1, 1) =  
Worksheets("x").Cells(1, 1).Value  
Worksheets(n)
```



	A	B	C
1	Temps (min)	pH	T°C
2	0	4.6	40
3	10	5.4	45
4	20	6.7	53
5	30	5.9	52
6	40	7.5	64
7	50	5.9	48
8	60	6.5	50

'écrire le contenu de la cellule (A1) de la feuille x
'=contenu de la cellule (A1) de la feuille x
'feuille numéro n

4. Pour aller plus loin

Démarrer comme si la feuille "compilation" n'était pas créée.

1. Ajouter une feuille en dernière position
2. Nommer cette feuille "compilation"
3. Remplir les entêtes de colonnes avec les temps, pH et température

Fonctions utiles :

Gestion des feuilles :

<code>Worksheets.Add()</code>	' ajout d'une feuille (par défaut, en dernière position)
<code>Worksheets.Count()</code>	' renvoie le nombre de feuilles
<code>Worksheets.Name()</code>	' permet de nommer une feuille

Ex : `Worksheets.Add(after:=Worksheets(Worksheets.Count)).Name = "compilation"`

'Ajoute une feuille en dernière position que l'on nomme "compilation"

5. Exemple bonus (chatGPT est votre ami)

Un suivi cinétique est réalisé sur un appareil. Un logiciel d'analyse produit un fichier excel d'export reprenant pour chaque temps analysé 2 colonnes. Sur ces 2 colonnes, la 1^{ère} colonne reste inchangée mais elle est néanmoins répétée à chaque temps.

→ Concevoir une macro VBA permettant de supprimer une colonne sur deux dans le fichier excel d'export.

1^{ère}
colonne
hauteur à
conserver

Temps à chaque analyse

A	B	C	D	E	F	G	H
Hauteur (µm)	00m:00s	Hauteur (µm)	00m:30s	Hauteur (µm)	01m:00s	Hauteur (µm)	01m:30s
0,000000	0,000000	0,000000	0,01150	0,000000	0,005013	0,000000	0,004828
40,000000	0,000000	40,000000	-0,046611	40,000000	-0,038581	40,000000	-0,061896
80,000000	0,000000	80,000000	-0,085566	80,000000	-0,022831	80,000000	-0,078848
120,000000	0,000000	120,000000	-0,094404	120,000000	-0,008755	120,000000	-0,135705
160,000000	0,000000	160,000000	-0,107675	160,000000	-0,009209	160,000000	-0,241581
200,000000	0,000000	200,000000	-0,127271	200,000000	0,012583	200,000000	-0,286347
240,000000	0,000000	240,000000	-0,188424	240,000000	-0,103350	240,000000	-0,335100
280,000000	0,000000	280,000000	-0,241241	280,000000	-0,054433	280,000000	-0,435250
320,000000	0,000000	320,000000	-0,263055	320,000000	0,036154	320,000000	-0,500224
360,000000	0,000000	360,000000	-0,469715	360,000000	0,086158	360,000000	-0,826875
400,000000	0,000000	400,000000	-0,513961	400,000000	0,063517	400,000000	-1,119266
440,000000	0,000000	440,000000	-0,402712	440,000000	0,172549	440,000000	-1,344128
480,000000	0,000000	480,000000	-0,538998	480,000000	0,170917	480,000000	-1,801039
520,000000	0,000000	520,000000	-0,384831	520,000000	0,494324	520,000000	-1,880538
560,000000	0,000000	560,000000	-0,143533	560,000000	0,867493	560,000000	-1,895352
600,000000	0,000000	600,000000	0,111509	600,000000	0,996988	600,000000	-1,988175
640,000000	0,000000	640,000000	0,346600	640,000000	1,098184	640,000000	-2,000978
680,000000	0,000000	680,000000	0,640953	680,000000	1,205180	680,000000	-1,801868
720,000000	0,000000	720,000000	0,627598	720,000000	1,276354	720,000000	-1,731048
760,000000	0,000000	760,000000	0,548089	760,000000	1,018469	760,000000	-1,754535
800,000000	0,000000	800,000000	0,517352	800,000000	0,924856	800,000000	-1,795986
840,000000	0,000000	840,000000	0,394426	840,000000	0,586344	840,000000	-1,718790
880,000000	0,000000	880,000000	0,290976	880,000000	0,266660	880,000000	-1,700722
920,000000	0,000000	920,000000	0,274427	920,000000	0,076865	920,000000	-1,572209
960,000000	0,000000	960,000000	0,219957	960,000000	-0,028575	960,000000	-1,477958
1000,000000	0,000000	1000,000000	0,029197	1000,000000	-0,308332	1000,000000	-1,555511
1040,000000	0,000000	1040,000000	0,030968	1040,000000	-0,248724	1040,000000	-1,368823
1080,000000	0,000000	1080,000000	-0,053174	1080,000000	-0,279121	1080,000000	-1,243511
1120,000000	0,000000	1120,000000	-0,134953	1120,000000	-0,318524	1120,000000	-1,179864
1160,000000	0,000000	1160,000000	-0,181126	1160,000000	-0,282258	1160,000000	-1,040091
1200,000000	0,000000	1200,000000	-0,176834	1200,000000	-0,257106	1200,000000	-0,958115
1240,000000	0,000000	1240,000000	-0,137678	1240,000000	-0,253311	1240,000000	-0,881970
1280,000000	0,000000	1280,000000	-0,181713	1280,000000	-0,255420	1280,000000	-0,840368
1320,000000	0,000000	1320,000000	-0,171512	1320,000000	-0,223849	1320,000000	-0,747640
1360,000000	0,000000	1360,000000	-0,114213	1360,000000	-0,180192	1360,000000	-0,656753
1400,000000	0,000000	1400,000000	-0,128275	1400,000000	-0,246217	1400,000000	-0,610790
1440,000000	0,000000	1440,000000	-0,061012	1440,000000	-0,204625	1440,000000	-0,577069

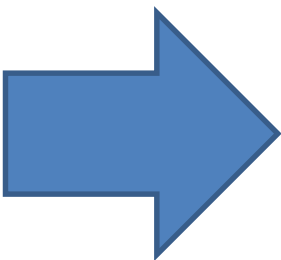
En vert en
orange =
colonnes à
conserver

En violet =
colonne à
supprimer
(une colonne
sur deux à
partir de la
3^{ème})

84 colonnes au total

5. Exemple bonus (chatGPT est votre ami)

+ récupération des valeurs de temps en entête de colonne dans une seule colonne intitulée "temps"



	A	B	C	D	E	F	G	H	I
1	Temps	Hauteur (µm)	00m:00s	00m:30s	01m:00s	01m:30s	02m:00s	02m:30s	03m:00s
2	00m:00s	0,000000	0,000000	0,011350	0,005013	0,004628	0,008780	-0,007526	-0,002331
3	00m:30s	40,000000	0,000000	-0,046611	-0,038581	-0,061896	-0,056625	-0,074083	-0,067662
4	01m:00s	80,000000	0,000000	-0,085566	-0,022831	-0,078848	-0,027624	-0,049684	-0,029413
5	01m:30s	120,000000	0,000000	-0,094404	-0,008755	-0,135705	0,006775	-0,092573	-0,035427
6	02m:00s	160,000000	0,000000	-0,107675	-0,009209	-0,241581	-0,099310	-0,200619	-0,078467
7	02m:30s	200,000000	0,000000	-0,127271	0,012583	-0,286347	-0,115856	-0,259621	-0,004283
8	03m:00s	240,000000	0,000000	-0,188424	-0,103350	-0,335100	-0,246465	-0,382914	-0,130829
9	03m:30s	280,000000	0,000000	-0,241241	-0,054433	-0,435250	-0,324468	-0,529566	-0,304568
10	04m:00s	320,000000	0,000000	-0,263055	0,036154	-0,500224	-0,318177	-0,607057	-0,344374
11	04m:30s	360,000000	0,000000	-0,469715	0,086158	-0,826875	-0,532551	-0,863043	-0,598368
12	05m:00s	400,000000	0,000000	-0,513961	0,063517	-1,119266	-0,587269	-1,003592	-0,867395
13	05m:30s	440,000000	0,000000	-0,402712	0,172549	-1,344128	-0,645014	-1,062058	-1,184351
14	06m:00s	480,000000	0,000000	-0,538998	0,170917	-1,801039	-0,787730	-1,359112	-1,765996
15	06m:30s	520,000000	0,000000	-0,384831	0,494324	-1,880538	-0,776487	-1,400066	-2,077631
16	07m:00s	560,000000	0,000000	-0,143533	0,867493	-1,895352	-0,752873	-1,419865	-2,266517
17	07m:30s	600,000000	0,000000	0,111509	0,996988	-1,988175	-0,839337	-1,411385	-2,382914
18	08m:00s	640,000000	0,000000	0,346600	1,098184	-2,000978	-1,026490	-1,421731	-2,595879
19	08m:30s	680,000000	0,000000	0,640953	1,205180	-1,801868	-0,940496	-1,181210	-2,478051
20	09m:00s	720,000000	0,000000	0,627598	1,276354	-1,731048	-0,852970	-0,956800	-2,444405
21	09m:30s	760,000000	0,000000	0,548089	1,018469	-1,754535	-0,906509	-0,753723	-2,391015
22	10m:00s	800,000000	0,000000	0,517352	0,924856	-1,795986	-0,705051	-0,575563	-2,285649