

TP 3 - L'enregistreur de macros Excel VBA

1 Introduction

Le moyen le plus simple de créer une macro consiste à enregistrer les actions de votre feuille de calcul à l'aide d'un outil précieux appelé **Enregistrer une macro**.

Tout ce que vous avez à faire est de choisir **Enregistrer une macro**, d'effectuer les actions qui composent la tâche que vous souhaitez automatiser, puis d'arrêter **Enregistrer une macro** lorsque vous avez terminé votre tâche.

Lorsque l'enregistreur de macros est activé, chaque action que vous effectuez – sélection d'une cellule, saisie d'un nombre, formatage d'une plage, à peu près tout – est enregistrée et représentée sous forme de code VBA dans une nouvelle macro.

Comme vous le verrez plus tard, lorsque vous exécutez la macro créée par l'enregistreur de macros, votre tâche est traitée automatiquement, comme si vous l'aviez fait manuellement.

L'enregistreur de macros est pratique pour les tâches courantes répétitives que vous préférez ne pas avoir à faire manuellement.

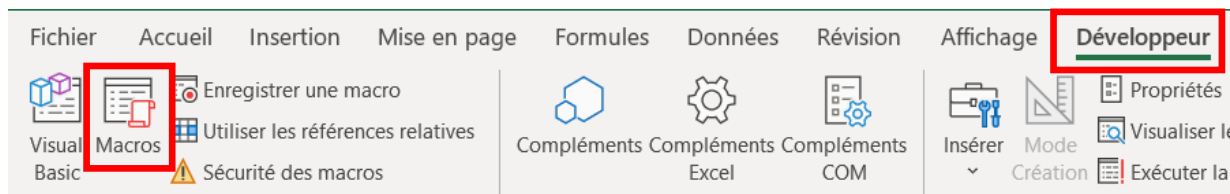
2 Enregistreur votre première macro

Supposons que nous ayons besoin de créer une macro qui active la feuille de calcul « Feuil2 ».

2.1 Enregistrer une macro

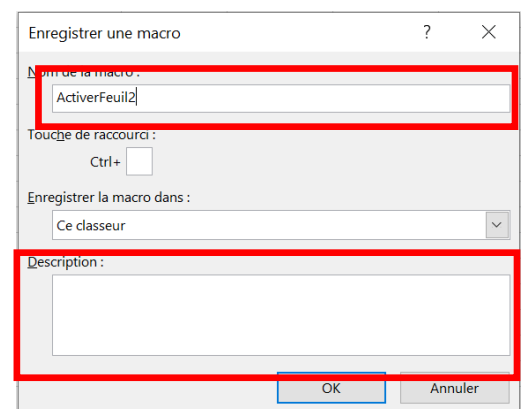
Démarrez Microsoft Excel et assurez-vous que la feuille « Feuil1 » est bien activée.

Pour activer l'enregistreur de macro, allez dans l'onglet **Développeur** de ruban et dans le groupe **Code** cliquez sur le bouton **Enregistrer une macro**.



Question 1 : Dans la fenêtre « Enregistrer une macro » qui s'ouvre, définissez le nom de la macro : « ActiverFeuil2 ». Vous pouvez saisir une description.

Le nom d'une macro ne doit pas contenir d'espaces et la description est importante pour les macros réutilisables, car après un certain temps, vous aurez du mal à vous rappeler pourquoi telle ou telle macro a été créée. Par défaut, les macros sont nommées Macro1, Macro2, etc. Afin de faciliter la reconnaissance d'une macro, il est préférable de ne pas utiliser de nom standard, mais de lui donner un nom unique qui explique à quoi elle est destinée.



2.2 Touche de raccourci

Remarquez la zone de saisie « Ctrl + » dans la section **Touche de raccourci**. Vous pouvez placer n'importe quelle lettre de l'alphabet dans ce champ. Lorsque vous appuyez sur **Ctrl** + **lettre**, vous exécuterez la macro. Une touche de raccourci n'est pas obligatoire, en fait la plupart de vos macros n'en auront pas besoin.

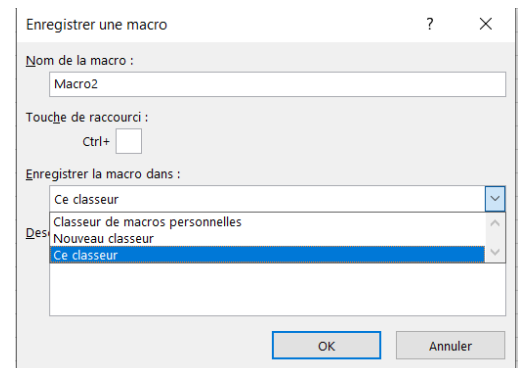
Recommandation : Excel a déjà attribué les 26 lettres de l'alphabet aux principaux raccourcis clavier, prenez la bonne habitude d'attribuer la combinaison **Ctrl** + **Maj** + **lettre**

Question 2 : Cliquez dans le champ **Touche de raccourci**, appuyez sur la touche Maj, et appuyez également sur une lettre de l'alphabet. Vous aurez créé le raccourci clavier Ctrl + Maj + S, qui n'interférera pas avec l'un des principaux raccourcis clavier intégrés d'Excel.

2.3 Liste déroulante : « Enregistrer la macro dans »

La liste déroulante **Enregistrer la macro dans** permet de sélectionner le classeur dans lequel la macro sera enregistrée.

- Si vous sélectionnez **Classeur de macros personnelles**, la macro sera enregistrée dans un classeur caché spécial dans lequel les macros sont stockées. Ce classeur est toujours ouvert, bien qu'il soit masqué, et les macros qui y sont enregistrées sont disponibles pour d'autres classeurs. Pour afficher votre classeur de macros personnelles, allez dans l'onglet **Affichage** et dans le groupe **Fenêtre**, cliquez sur le bouton **Afficher**.
- Si vous sélectionnez **Ce classeur** dans la liste déroulante (c'est-à-dire le choix proposé par défaut par l'ordinateur), la macro sera enregistrée dans le classeur actif
- Si vous sélectionnez **Nouveau classeur**, elle sera enregistrée dans un nouveau classeur.



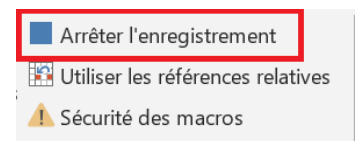
Question 3 : Sélectionnez « Ce classeur »

2.4 Enregistrer les commandes

Question 4 : Sélectionnez la feuille : « Feuil2 ». Ce sera la seule commande pour cette macro !

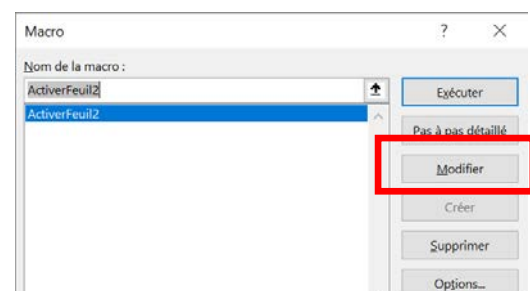
Question 5 : Cliquez sur le bouton Arrêter l'enregistrement.

Question 6 : Enregistrez votre classeur au format « xlsx » (qui prend en charge les macros).



2.5 Voir le code

Question 7 : Allez dans l'onglet **Développeur** et dans le groupe **Code** cliquez sur le bouton **Macros**. Dans la fenêtre Macro

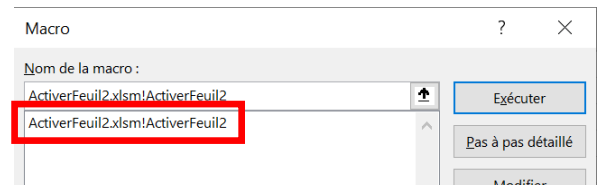


sélectionnez le nom de la macro créée dans la liste et cliquez sur le bouton **Modifier**.

L'écran affichera la fenêtre de l'éditeur VBA, qui contiendra le code de la macro qui vient d'être enregistrée.

```
Sub ActiverFeuil2()  
'  
' ActiverFeuil2 Macro  
' Active la feuille "Feuil2"  
'  
    Sheets("Feuil2").Select  
End Sub
```

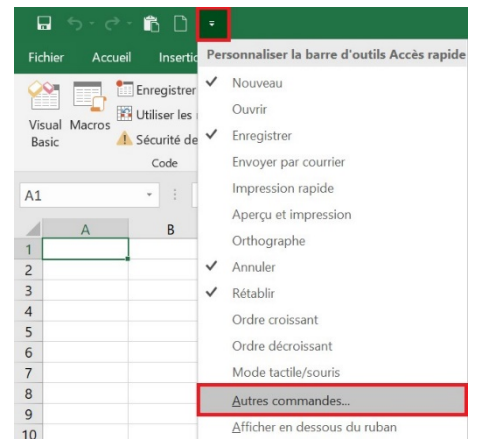
Question 8 : Sans fermer ce classeur, créez un autre classeur. Accédez à l'onglet Développeur, cliquez sur Macros. Dans la fenêtre Macro qui s'ouvre choisissez la macro créée précédemment et cliquez sur le bouton Exécuter. Assurez-vous que dans votre classeur cette macro a activé la feuille : « Feuil2 ».



3 Affecter une macro à un bouton

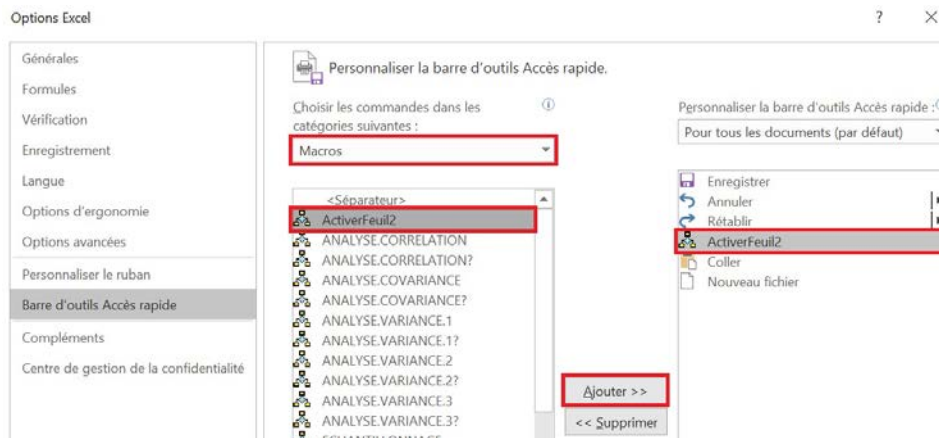
Il est possible de déclencher l'exécution d'une macro à l'aide d'un bouton. Le bouton pourra être placé soit dans la barre d'outils « Accès rapide » soit dans le ruban.

Question 9 : Faites un clic droit dans la barre d'outils d'accès rapide et sélectionnez « Autres commandes... ».



La fenêtre Options s'ouvre dans la catégorie « Barre d'outils d'accès rapide ». Dans la zone de liste déroulante « Choisir les commandes dans les catégories suivantes », sélectionnez « Macros »

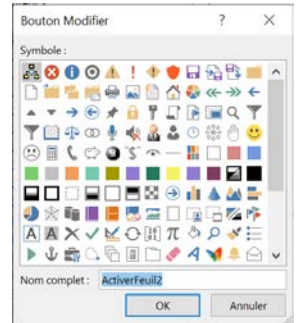
Sélectionnez « ActiverFeuil2 » que vous avez créé dans la colonne de gauche et utilisez le bouton « Ajouter ».



Pour affecter une icône au bouton, cliquez sur le bouton « Modifier » en bas de la fenêtre et choisissez une icône. Vous pouvez aussi modifier le nom qui sera affiché dans une info-bulle.

Cliquez sur OK.

Cliquez sur OK.



Question 10 : Assurez-vous que le bouton avec la macro enregistrée apparaît dans la barre d'outils d'accès rapide. Activer le bouton.

4 Un autre exemple avec l'enregistreur de macros

Par exemple, supposons que vous gérez chaque jour un tableau de données, tel que celui présenté à la Figure 1, qui indique le chiffre d'affaires obtenu par les représentants dans les différentes régions.

	A	B	C
1	Représentants	Région	Chiffre d'affaires
2	CHOPIN	Est	15000
3	DARGAUD	Ouest	18000
4	MARCELIN	Ouest	18000
5	ARNAL	Nord	14000
6	FILLON	Sud	18000
7	LAVAL	Est	16000
8	TRASSARD	Nord	15000
9	VIDAL	Nord	20000

Figure 1

	A	B	C
1	Région	Représentants	Chiffre d'affaires
2	Est	CHOPIN	15 000.00 €
3	Est	LAVAL	16 000.00 €
4	Nord	ARNAL	14 000.00 €
5	Nord	TRASSARD	15 000.00 €
6	Nord	VIDAL	20 000.00 €
7	Ouest	DARGAUD	18 000.00 €
8	Ouest	MARCELIN	18 000.00 €
9	Sud	FILLON	18 000.00 €

Figure 2

La tâche quotidienne consiste à obtenir le résultat de la figure 2 en exécutant les actions suivantes :

- Permuter les colonnes « Représentants » et « Région »
- Trier le tableau par « Région », et par « Représentants »
- Mettre les « Chiffres d'affaires » au format « euro »
- Mettre les titres de colonne en gras.

Question 11 : Saisir les données dans la feuille « Feuil1 »

Question 12 : Copier la feuille « Feuil1 »

Question 13 : Dans la feuille copiée, exécuter toutes les actions nécessaires pour obtenir le résultat

Non seulement ces actions sont monotones, mais il y a un risque de faire des erreurs. Par contre si vous effectuez les étapes nécessaires en enregistrant une macro, la tâche peut être réduite à un simple clic de souris ou un raccourci clavier.

Question 14 : Enregistrez les actions nécessaires dans la macro « FormatageDonnees »

Le code devrait être le suivant :

```
Sub FormatageDonnees ()
'
' FormatageDonnees Macro
'
'
Columns("B:B").Select
Selection.Cut
Columns("A:A").Select
Selection.Insert Shift:=xlToRight
Range("A1:C9").Select
ActiveWorkbook.Worksheets("Feuill (2)").Sort.SortFields.Clear
ActiveWorkbook.Worksheets("Feuill (2)").Sort.SortFields.Add2
Key:=Range( _
    "A2:A9"), SortOn:=xlSortOnValues, Order:=xlAscending,
DataOption:= _
    xlSortNormal
ActiveWorkbook.Worksheets("Feuill (2)").Sort.SortFields.Add2
Key:=Range( _
    "B2:B9"), SortOn:=xlSortOnValues, Order:=xlAscending,
DataOption:= _
    xlSortNormal
With ActiveWorkbook.Worksheets("Feuill (2)").Sort
    .SetRange Range("A1:C9")
    .Header = xlYes
    .MatchCase = False
    .Orientation = xlTopToBottom
    .SortMethod = xlPinYin
    .Apply
End With
Range("A1:C1").Select
Selection.Font.Bold = True
Range("C2:C9").Select
Selection.Style = "Currency"
End Sub
```

Les lignes de code représentent les actions que vous avez effectué pendant l'enregistrement de la macro :

1. Sélectionner la colonne B
2. Couper cette colonne
3. Coller avant la colonne A
4. Sélectionner la plage A1:C9 où se trouve les données
5. Trier la plage sélectionnée
6. Sélectionner A1:C1

7. Mettre en gras
8. Sélectionner la plage C2:C9
9. Mettre en Euro

5 Modification et amélioration du code

Il est facile de générer une macro avec le bouton **Enregistrer une Macro**, mais le code généré manque souvent d'efficacité.

En VBA, vous pouvez agir directement sur la plupart des objets, sans avoir besoin de les sélectionner. Une règle de base dans le développement de VBA est donc de ne sélectionner ou d'activer des objets que si vous en avez besoin. Les méthodes de « Select » et « Activate » font partir des causes d'exécution lente des macros.

Par exemple, les deux lignes de code suivantes :

```
Columns ("B:B").Select  
Selection.Cut
```

Peuvent et doivent être regroupées en une seule ligne, en ignorant la méthodes « Select » :

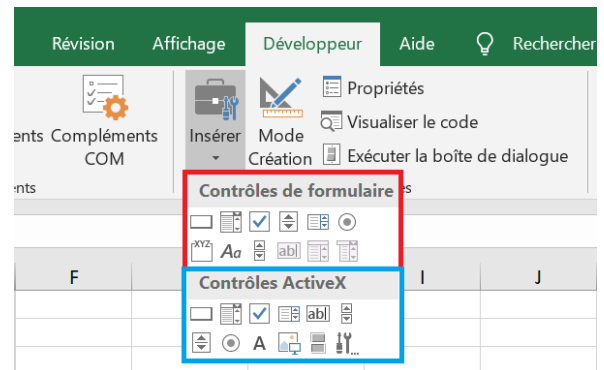
```
Columns ("B:B").Cut
```

Question 15 : Modifier le code pour supprimer toutes les méthodes « Select » pour réduire le nombre de lignes de code et rendre votre macro « FormatageDonnees » plus lisibles et plus efficace

6 Automatiser le travail des feuilles de calcul avec des contrôles

Excel dispose d'un ensemble complet de divers contrôles : bouton de commande, zone de texte, case à cocher, etc., qui, si nécessaire, sont placés sur la feuille de calcul.

Question 16 : Pour voir la liste des contrôles disponibles, accédez à l'onglet **Développeur** et dans le groupe **Contrôles**, cliquez sur le bouton **Insérer**.



- Les contrôles situés dans le groupe **Contrôles de formulaire** sont principalement destinés à garantir la compatibilité avec les fichiers d'anciennes versions d'Excel (jusqu'à Excel 97) qui utilisent ces contrôles. Ils ont beaucoup moins de capacités que les contrôles situés dans le groupe de **contrôles ActiveX**. Certains de ces éléments ne peuvent pas du tout être utilisés dans les documents Excel récents (zone de texte, zone de liste, zone de liste déroulante). Cependant, ces contrôles présentent également un certain nombre d'avantages que les contrôles situés dans le panneau **Contrôles ActiveX** n'ont pas, en particulier, ces contrôles peuvent être placés sur des feuilles de graphique.
- Les **contrôles ActiveX** sont des composants indépendants des applications comme Excel dans lesquels ils peuvent être utilisés. Ce groupe comprend également des contrôles similaires à de nombreux contrôles du groupe **Contrôles de formulaire**.

En plus des commandes du menu **contrôles ActiveX**, vous pouvez utiliser des commandes supplémentaires. Excel est livré avec un certain nombre de ces éléments, par exemple, des contrôles multimédias, avec lesquels vous pouvez lire du son ou une vidéo directement à partir de la feuille de calcul.

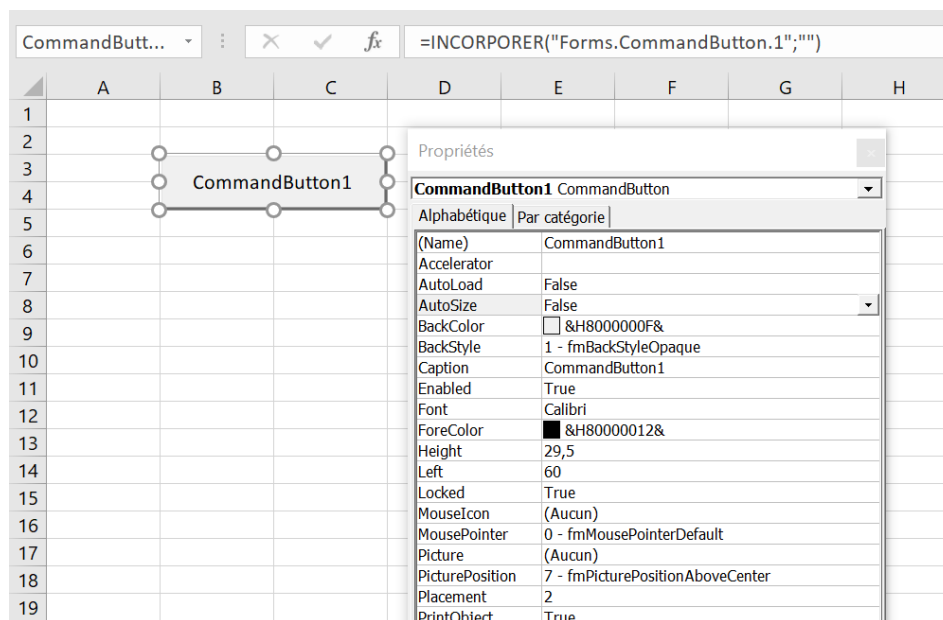
Utiliser les contrôles déclenche des événements que l'on peut associer à du code VBA. Par exemple, appuyer sur un bouton, sélectionner un élément de la liste, sélectionner une case d'option, définir une case à cocher, etc. conduira à exécuter du code VBA pour effectuer un calcul, construire un diagramme etc.

7 Utilisation du contrôle Bouton de commande sur une feuille de calcul

Nous allons utiliser un « Bouton de commande » du groupe **Contrôles ActiveX** placé sur la feuille de calcul « Feuil1 ». Lorsque nous cliquons sur le bouton de commande, la feuille de calcul « Feuil2 » sera activée.

Question 17 : Assurez-vous d'avoir sectionné la feuille de calcul « Feuil1 ».

Question 18 : Accédez à l'onglet **Développeur** et dans le groupe **Contrôles**, cliquez sur **Insérer**. Cliquez sur le contrôle bouton de commande du groupe Contrôles ActiveX. Dessiner un bouton dans la feuille de calcul. Par défaut le premier « bouton de commande » est nommé « CommandButton1 »



Question 19 : Cliquez sur le bouton de commande créé avec le bouton droit de la souris et dans le menu contextuel qui apparaît, sélectionnez la commande **Propriétés** pour ouvrir la fenêtre **Propriétés**.

Le contrôle bouton de commande est un objet, c'est-à-dire qu'il possède, comme tout objet, des **propriétés**, des **méthodes** et des **événements**.

- La propriété « Caption » définit le nom affiché sur la surface du contrôle

Question 20 : Modifiez la valeur de la propriété « Caption » pour « Feuil2 »

- La propriété « Name » identifie l'objet dans le code VBA.

Question 21 : Testez les propriétés BackColor, Font, ForeColor, Shadow.

Nous allons maintenant écrire le code déclenché par l'événement « Click » : activer la feuille « Feuil2 ».

Question 22 : Double-cliquez sur le bouton de commande. L'éditeur VBA s'ouvre dans le module feuille de calcul (Feuil1)

Le squelette du code de la procédure « Click » a été automatiquement ajouté :

```
Private Sub CommandButton1_Click()  
  
End Sub
```

Question 23 : Ajouter la ligne de code :

```
Sheets("Feuil2").Select
```

Question 24 : Revenez à la feuille de calcul Feuil1.

Question 25 : Quitter le **Mode de création** en cliquant sur le bouton **Mode de Création**

Question 26 : Testez **Bouton de commande**, il activera la feuille de calcul « Feuil2 »