

VBA EXCEL

Chapitre I

OBJECTIFS

Groupe cible :

Toutes les personnes qui veulent automatiser certaines tâches ou créer leur propre application dans Excel.

Prérequis :

Très bonne connaissance d'Excel

Durée :

2 jours (12h)

1. Macro's
 - a. Enregistreur de macro's
 - b. Sauver une macro
 - c. Retravailler une macro
 - d. Création d'une icône dans la barre d'outils
2. L'éditeur VBA
 - a. Environnement de programmation (fenêtres barre d'outils)
 - b. Création d'une procédure (subroutine, fonction)
 - c. Aide dans la création
3. Les variables
 - a. Déclaration
 - b. Portée

Objectifs :

À l'issue de la formation, les participants seront capables de :

- Créer des macros avec l'enregistreur de macros
- Programmer des procédures et des fonctions
- Faire interagir un programme avec l'utilisateur

Contenu :

4. Références, sélections et manipulations
 - a. Les objets (classeurs, feuilles et cellules),
 - b. Les mots clés
5. Interaction avec l'utilisateur
 - a. MsgBox et InputBox
 - b. formulaires
6. Les conditions
 - a. If
 - b. Select case
7. Les boucles
 - a. For – next
 - b. Do – loop
8. Debugger un programme
 - a. étape par étape
 - b. point d'arrêt
 - c. espion
9. Gestion des erreurs

Chapitre II

ÉNONCÉS

Enregistreur de macros

Exercice 1 :

Enregistrez une macro nommée **Absolu** dans le classeur en cours
Avant de commencer l'enregistrement, cliquez dans la cellule A3
Pendant l'enregistrement

Tapez une valeur dans cette cellule
validez

Cliquez dans la cellule A1 et tapez une autre valeur

Testez

Exercice 2 :

Enregistrer une macro nommée **Relatif** dans le classeur de macros
personnelles

Même exercice que le précédent mais cette fois, activez le bouton relatif avant
de commencer les actions.

Testez

Exercice 3:

Enregistrement d'une macro nommée **GrasItalique** dans le classeur en
cours

Pendant l'enregistrement
utiliser Format/cellule ...

Regardez le code

Nettoyez l'inutile

Exercice 4 :

- Affectez un raccourci clavier à la macro 'Absolu'
- Créez un bouton sur la feuille et affectez-lui la macro 'relatif'
- Affectez la macro 'GrasItalique' à un nouveau bouton dans la
barre d'outils ou dans le menu

Création d'une procédure Sub

Exercice 5 :

Écrivez une procédure qui :

- affiche une boîte de dialogue *Bonjour* avec un bouton unique et le
titre *Message VBA*

Exercice 6 :

Dans la procédure ci-dessous, trouvez l'erreur

```
Sub ChercherErreur  
    MsgBox Voici un nouveau message  
End sub
```

Création d'une fonction personnalisée

Exercice 7 :

Créez une fonction qui :

- permet de calculer la circonférence d'un cercle (Diamètre * 3,14)

Déclaration implicite des variables

Que donnera cette procédure comme résultat ?

```
Sub SurfaceCarré()  
    côté = 10  
    MsgBox côté * côté  
End Sub
```

Déclaration explicite des variables

Exercice 8

Créez une procédure qui

- Demandra à l'utilisateur de taper un nombre compris entre 1 et 10 (bornes comprises)
- Lorsque l'utilisateur validera son entrée, le programme doit afficher une boîte de dialogue affichant le nombre tapé augmenté de l'entrée précédente.

Exemple :

1^{ere} utilisation de la procédure, l'utilisateur tape 10, le programme affiche 10
2^e utilisation de la procédure, l'utilisateur tape 25, le programme affiche 35 (25+10)
et ainsi de suite

Déclaration d'une constante

Exercice 9

Copier la fonction créée à l'exercice 7, la renommer et travaillez avec une constante pour Pi plutôt qu'avec une valeur

Structure du langage

Exercice 10 – affectation d'une valeur à une propriété

Créez une procédure qui :

- Affichera la valeur 12 dans la cellule active.

Exercice 11 – Récupération de la valeur d'une propriété

Créez une procédure qui :

- Affichera dans une boîte de message la valeur contenue dans la cellule active

Exercice 12 – Syntaxe de l'utilisation d'une méthode

Créez une procédure qui fera dans l'ordre les actions suivantes :

- Mettre la valeur 'Hello World' dans la cellule active
- Afficher dans une boîte de message le contenu de la cellule active
- Effacer le contenu de la cellule active (la méthode)
- Pour vérifier, afficher de nouveau dans une boîte de message, le contenu de la cellule active

Exercice 13 – Règles d'écriture

Créez une procédure qui :

- Demandra à l'aide d'une 'inputbox' à l'utilisateur de taper une valeur comprise entre 1 et 100.

Propriétés de l'inputbox

- Affichage dans le coin supérieur gauche de l'écran
- Le message = 'Tape une valeur entre 1 et 100'
- Utilisez des variables

Commentez votre code à l'intérieur de la procédure et rendez-le agréable et aisé à lire

Exercice 14 – Structures de contrôle – With (plusieurs actions sur un objet)

- Dans le module Exercice, copiez la procédure : **exercice 14**
- Insérez un nouveau module que vous nommerez 'Correction'

- Collez-y votre procédure
- Corrigez le code pour éviter la redondance

Exercice 15 – Structures de contrôle – With (plusieurs actions sur un objet)

Même chose que l'exercice précédent mais cette fois vous le faites seuls.

Exercice 16 – Structures de contrôle – If...then

Effacez le contenu de la cellule A1 de la feuille sélectionnée

Créez une procédure qui :

- Émet un bip et affiche le message 'Pas de titre' si la cellule A1 de la feuille active est vide.

Exercice 17 – Structures de contrôle – If...then...end if

Tapez quelque chose dans la cellule A1 de la feuille active

Créez une procédure qui :

- Si la cellule A1 de la feuille active n'est pas vide, affichera son contenu en Gras, Souligné et Rouge

Exercice 18 – Structures de contrôle – If...then...else...end if

Même exercice que le précédent mais en plus si la cellule est vide, vous devez en avertir l'utilisateur

Exercice 19 – Structures de contrôle – If...then...elseif...else...end if

Créez une procédure qui :

- Si l'intérieur de la cellule A1 de la feuille active est en rouge, toutes les cellules de la feuille active seront coloriées en rouge
- Si l'intérieur de la cellule A1 de la feuille active est en vert, toutes les cellules de la feuille active seront coloriées en vert
- Si l'intérieur de la cellule A1 de la feuille active est en bleu, toutes les cellules de la feuille active seront coloriées en bleu
- Si aucune de ces 3 couleurs, toutes les cellules de la feuille active seront coloriées en jaune

Exercice 20 – Structures de contrôle – Select case

Créez une fonction nommée TotalAPayer qui calculera le Total à payer.

Vous devrez fournir 3 arguments à la fonction

- la quantité
- le prix HT
- le port

En fonction de la quantité, vous accorderez une remise :

- 1 pièce pas de remise
- de 2 jusque 10, remise de 1%
- de 11 à 20, remise de 5%

- de 21 à 30, remise de 10%
- de 31 à 40, remise de 15%
- de 41 à 50, remise de 20%
- Si la commande est supérieure à 50 pièces : 'erreur dans la quantité' doit apparaître

Exercice 21 – Structures de contrôle – Do While ...loop

Créez une procédure qui :

- Demandra à l'utilisateur de taper un nombre de 3 chiffres maximum dans une 'inputbox'.
- Tant que la somme des nombres tapés est inférieure à 1000, l'inputbox doit se réafficher pour taper une nouvelle valeur.

Exercice 22 – Structures de contrôle – Do until ...loop

Créez une procédure qui :

- Demandra à l'utilisateur de taper une date dans une 'inputbox'.
- Tant que la frappe n'est pas reconnue comme une date par le système, il faut réafficher l'inputbox.

Exercice 23 – Structures de contrôle – For...next

Renommez vos feuilles :

- 1 = Tournai
- 2 = Mons
- 3 = Charleroi

Créez une procédure qui :

- rendra la feuille 'Charleroi' active
- sélectionnera la cellule E5 de cette feuille
- inscrira dans cette cellule 'Je suis dans la cellule 1 de la colonne E'
- descendra d'1 cellule et y inscrira ' je suis dans la cellule 2 de la colonne E'
- et ainsi de suite jusque la ligne 1000

Exercice 24 – Structures de contrôle – For...next Step

Même exercice que le précédent mais cette fois-ci l'inscription doit se faire toutes les 5 lignes.

Exercice 25– Structures de contrôle – For Each...next

Créez une procédure qui :

- Renommra toutes les feuilles du classeur actif
- La première en 1
- La deuxième en 2
- Et ainsi de suite

Exercice 26– Structures de contrôle – Gestion des erreurs – Variables tableaux

Créez une procédure qui :

- Empêchera l'affichage des messages d'alerte d'Excel
- Fera en sorte que le classeur comporte 12 feuilles (si trop peu ajout, trop retrait) nommées respectivement janvier, février, ..., décembre
- Y inclure une gestion des erreurs qui :
 - o affichera en cas d'erreur
 - Le numéro de l'erreur
 - L'intitulé de l'erreur
 - o Passera au-dessus de l'erreur pour continuer la procédure

Exercice 27– Boîtes de dialogue intégrées - Propriétés

Créez une procédure qui :

- Affichera le nombre de boîtes de dialogue intégrées
- Affichera ensuite la boîte de dialogue du tri

Exercice 28– Boîtes de dialogue intégrées - Méthodes

Créez une procédure qui :

- Affichera une boîte de dialogue qui me permettra de choisir un ou plusieurs fichiers
 - o Excel
 - o Titre de la boîte : Sélectionne un fichier
 - o Je peux choisir plusieurs fichiers
- Après validation une boîte de message s'affichera avec le nom et le chemin de chaque fichier sélectionné sur une ligne différente

Exercice 29– Boîtes de dialogue intégrées – Méthodes

Créez une procédure qui :

- Affichera une boîte de dialogue qui me permettra de choisir un fichier
 - o Excel
 - o Titre de la boîte : Ouvrir fichier
 - o Je peux choisir un fichier
- Ouvrira le fichier sélectionné

Exercice 30– Boîtes de dialogue intégrées – Méthodes

Créez une procédure qui :

- Affichera une boîte de dialogue qui me permettra de donner un nom et un endroit de stockage au fichier en cours
- Enregistrera réellement le fichier

Exercice 31 – Boîtes de dialogue prédéfinies - Inputbox

Créez une procédure qui :

- Affichera dans une boîte de dialogue, sur des lignes distinctes, la moyenne, le maximum et le minimum d'une plage que le système vous invitera à sélectionner.

Exercice 32 – Boîtes de dialogue prédéfinies – MsgBox

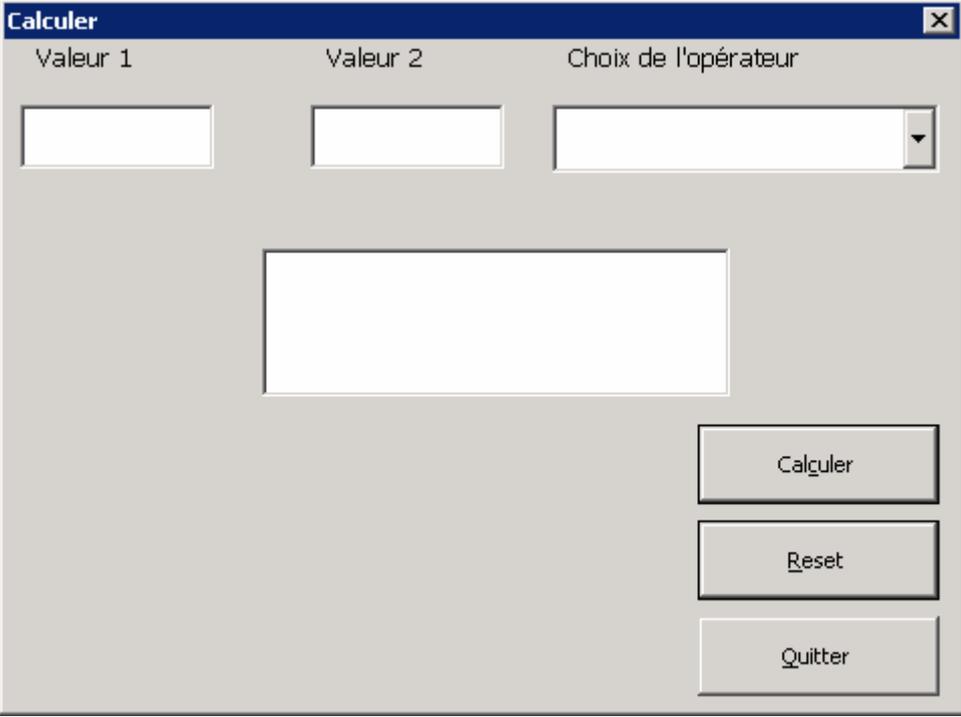
Créez une procédure qui :

- Affichera une boîte de dialogue qui vous demandera si vous voulez oui ou non effacer la(les) colonnes(s) que vous avez sélectionné.
- La boîte de dialogue comprendra les boutons Oui et non et un point d'exclamation. De plus, le bouton OK devra avoir le focus.

Exercice 33 – Formulaires

Créez un formulaire 'calculatrice' qui sera capable de :

- Effectuer les 4 opérations fondamentales
- Attention à la division par 0
- Les différents résultats doivent s'inscrire à partir de la cellule A1 de la feuille 1 et descendre d'une cellule à chaque opération même lorsque Excel a été fermé.
- Le formulaire doit s'afficher à l'ouverture du fichier
- Le fichier doit s'enregistrer quand vous cliquez sur le bouton Quitter



The image shows a Windows-style dialog box titled "Calculer". It has a dark blue title bar with a close button (X) on the right. The dialog contains three input fields at the top: "Valeur 1", "Valeur 2", and "Choix de l'opérateur". Below these is a large empty rectangular area for the result. On the right side, there are three buttons stacked vertically: "Calculer", "Reset", and "Quitter".

Chapitre III

CORRIGÉS

Enregistreur de macros

Exercice 1 :

Enregistrez une macro nommée **Absolu** dans le classeur en cours
Avant de commencer l'enregistrement, cliquez dans la cellule A3

Pendant l'enregistrement

Tapez une valeur dans cette cellule

validez

Cliquez dans la cellule A1 et tapez une autre valeur

Testez

Exercice 2 :

Enregistrer une macro nommée **Relatif** dans le classeur de macros
personnelles

Même exercice que le précédent mais cette fois, activez le bouton relatif avant
de commencer les actions.

Testez

Exercice 3:

Enregistrement d'une macro nommée **GrasItalique** dans le classeur en
cours

Pendant l'enregistrement

utiliser Format/cellule ...

Regardez le code

Nettoyez l'inutile

Exercice 4 :

- Affectez un raccourci clavier à la macro 'Absolu'
- Créez un bouton sur la feuille et affectez-lui la macro 'relatif'
- Affectez la macro 'GrasItalique' à un nouveau bouton dans la
barre d'outils ou dans le menu

Création d'une procédure Sub

Exercice 5 :

Écrivez une procédure qui :

- affiche une boîte de dialogue *Bonjour* avec un bouton unique et le
titre *Message VBA*

```
Sub Exercice5()
```

```
MsgBox "Hello World" 'sortie vers l'utilisateur
```

```
End Sub
```

Exercice 6 :

Dans la procédure ci-dessous, trouvez l'erreur

```
Sub ChercherErreur  
    MsgBox Voici un nouveau message  
End sub
```

Création d'une fonction personnalisée

Exercice 7 :

Créez une fonction qui :

- permet de calculer la circonférence d'un cercle

```
Function Circonférence(diamètre As Long) 'exercice7  
    Circonférence = diamètre * 3.14  
End Function
```

Déclaration implicite des variables

Que donnera cette procédure comme résultat ?

```
Sub SurfaceCarré()  
    côté = 10  
    MsgBox côté * côté  
End Sub
```

0 car côté ne vaut rien

Déclaration explicite des variables

Exercice 8

Créez une procédure qui

- Demandra à l'utilisateur de taper un nombre compris entre 1 et 10 (bornes comprises)
- Lorsque l'utilisateur validera son entrée, le programme doit afficher une boîte de dialogue affichant le nombre tapé augmenté de l'entrée précédente.

Exemple :

1^{ère} utilisation de la procédure, l'utilisateur tape 10, le programme affiche 10

2^e utilisation de la procédure, l'utilisateur tape 25, le programme affiche 35 (25+10)

et ainsi de suite

```
Sub Addition() 'exercice 8 - variable dim ne fonctionne pas  
' le contenu de la variable est réinitialisé à chaque fois (remis à 0)
```

```
Dim Total As Long  
Dim LeNombre As Integer
```

```
LeNombre = InputBox("Tape un nombre entier entre 1 et 10")  
Total = LeNombre + Total
```

```
MsgBox Total
```

```
End Sub
```

```
Sub AdditionStatic() 'exercice 8 bis - variable static fonctionne -  
conserve la valeur
```

```
'déclaration  
Static Total As Long  
Dim LeNombre As Integer
```

```
'affectation  
LeNombre = InputBox("Tape un nombre entier entre 1 et 10")  
Total = LeNombre + Total
```

```
'exécution  
MsgBox Total
```

```
End Sub
```

Déclaration d'une constante

Exercice 9

Copier la fonction créée à l'exercice 7, la renommer et travaillez avec une constante pour Pi plutôt qu'avec une valeur

```
Function CirconférenceBis(diamètre As Long) 'exercice7 adapté  
Const Pi = 3.14  
Circonférence = diamètre * Pi  
End Function
```

Exercice 10 – affectation d'une valeur à une propriété

Créez une procédure qui :

- Affichera la valeur 12 dans la cellule active.

```
Sub PropriétésAffectation() 'exercice 10
'affectation d'une valeur à une propriété
Application.ActiveCell = 12
End Sub
```

Exercice 11 – Récupération de la valeur d'une propriété

Créez une procédure qui :

- Affichera dans une boîte de message la valeur contenue dans la cellule active

```
Sub PropriétésRécupération() 'exercice 11
'récupération de la valeur d'une propriété
MsgBox Application.ActiveCell
End Sub
```

Exercice 12 – Syntaxe de l'utilisation d'une méthode

Créez une procédure qui fera dans l'ordre les actions suivantes :

- Mettre la valeur 'Hello World' dans la cellule active
- Afficher dans une boîte de message le contenu de la cellule active
- Effacer le contenu de la cellule active (la méthode)
- Pour vérifier, afficher de nouveau dans une boîte de message, le contenu de la cellule active

```
Sub méthode() 'exercice 12
Dim Valeur As String

' je mets une valeur dans la cellule active
Application.ActiveCell = "Hello world"
'j'affecte le contenu de la cellule active à valeur
Valeur = Application.ActiveCell.Value
'je l'affiche avec message
MsgBox Valeur
'j'efface le contenu de la cellule active à l'aide de la méthode CLEAR
Application.ActiveCell.Clear
'je vérifie en refaisant la même action
'j'affecte le contenu de la cellule active à valeur
Valeur = Application.ActiveCell.Value
'je l'affiche avec message
MsgBox Valeur
End Sub
```

Exercice 13 – Règles d'écriture

Créez une procédure qui :

- Demandra à l'aide d'une 'inputbox' à l'utilisateur de taper une valeur comprise entre 1 et 100.

Propriétés de l'inputbox

- Affichage dans le coin supérieur gauche de l'écran
- Le message = 'Tape une valeur entre 1 et 100'
- Utilisez des variables

Commentez votre code à l'intérieur de la procédure et rendez-le agréable et aisé à lire

```

Sub argument() 'exercice 13 première possibilité avec ", "
'déclaration
Dim x As Integer, y As Integer
Dim Message As String
Dim Result As Integer

'affectation
x = 10
y = 10
Message = "Tape un nombre plus petit que 100"

'execution
Result = InputBox(Message, , , x, y)
End Sub

```

```

Sub argument2() 'exercice 13 méthode 2 avec les noms d'arguments
'déclaration
Dim x As Integer, y As Integer
Dim Message As String
Dim Result As Integer

'affectation
x = 10
y = 10
Message = "Tape un nombre plus petit que 100"

'execution
Result = InputBox(prompt:=Message, xpos:=x, ypos:=y)

End Sub

```

Exercice 14 – Structures de contrôle – With (plusieurs actions sur un objet)

- Dans le module Exercice, copiez la procédure **:exercice 14**
- Insérez un nouveau module que vous nommerez 'Correction'
- Collez-y votre procédure
- Corrigez le code pour éviter la redondance

```

Sub Exercice14Correction() 'correction exercice 14
'structure avec with
With Application.ActiveCell
'gras à la cellule active
.Font.Bold = True
'fond en rouge
.Interior.ColorIndex = 3
'mot lapin dedans
.Value = "lapin"
End With
End Sub

```

Exercice 15 – Structures de contrôle – With (plusieurs actions sur un objet)

Même chose que l'exercice précédent mais cette fois vous le faites seuls.

```
Sub Exercice15Correction()  
    'sélection de la cellule E5  
    Application.Cells(1, 5).Select  
  
    With Selection  
        ' contenu = frappe de l'utilisateur  
        .Value = InputBox("Tape une valeur")  
        'fond en vert flashy  
        .Interior.ColorIndex = 4  
        With .Font  
            'cellule sélectionnée en gras  
            .Bold = True  
            'en italique  
            .Italic = True  
            ' texte en taille 16  
            .Size = 14  
            ' texte en arial  
            .Name = "arial"  
            'texte en rouge  
            .Color = RGB(255, 0, 0)  
        End With  
    End With  
End Sub
```

Exercice 16 – Structures de contrôle – If...then

Effacez le contenu de la cellule A1 de la feuille sélectionnée

Créez une procédure qui :

- Émet un bip et affiche le message 'Pas de titre' si la cellule A1 de la feuille active est vide.

```
Sub Exercice16()  
    'le if ... then  
    'si la cellule A1 est vide le système émet un beep et un message  
    d'avertissement  
    If IsEmpty(Application.Cells(1, 1)) Then Beep: MsgBox "Pas de titre"  
  
End Sub
```

Exercice 17 – Structures de contrôle – If...then...end if

Tapez quelque chose dans la cellule A1 de la feuille active

Créez une procédure qui :

- Si la cellule A1 de la feuille active n'est pas vide, affichera son contenu en Gras, Souligné et Rouge

```
Sub Exercice17()  
    'le if... then  
    'end if  
    'si la cellule A1 n'est pas vide, elle est mise en gras,rouge et  
    souligné
```

```

    If Not IsEmpty(Range("A1")) Then
        With Range("A1").Font
            .Bold = True
            .Underline = True
            .ColorIndex = 3
        End With
    End If
End Sub

```

Exercice 18 – Structures de contrôle – If...then...else...end if

Même exercice que le précédent mais en plus si la cellule est vide, vous devez en avertir l'utilisateur

```

Sub Exercice18()
    'le if ... then
    'else ... end if
    'si la cellule A1 n'est pas vide, elle est mise en gras,rouge et
    souligné
    'dans le cas contraire, un message apparaît invitant à compléter +
    activer A1
    If Not IsEmpty(Range("A1")) Then
        With Range("A1").Font
            .Bold = True
            .Underline = True
            .ColorIndex = 3
        End With
    Else
        MsgBox "Cellule A1 vide, veuillez compléter"
        Range("A1").Select
    End If
End Sub

```

Exercice 19 – Structures de contrôle – If...then...elseif...else...end if

Créez une procédure qui :

- Si l'intérieur de la cellule A1 de la feuille active est en rouge, toutes les cellules de la feuille active seront coloriées en rouge
- Si l'intérieur de la cellule A1 de la feuille active est en vert, toutes les cellules de la feuille active seront coloriées en vert
- Si l'intérieur de la cellule A1 de la feuille active est en bleu, toutes les cellules de la feuille active seront coloriées en bleu
- Si aucune de ces 3 couleurs, toutes les cellules de la feuille active seront coloriées en jaune

```

Sub Exercice19()
    ' le if... then...
    'elseif...
    'else...
    'End If
    'si le fond de la cellule A1 est rouge, alors toutes les cellules en
    rouge
    'envisager plusieurs couleurs

```

```

If ActiveSheet.Cells(1, 1).Interior.ColorIndex = 3 Then
    ActiveSheet.Cells.Select
    Selection.Interior.ColorIndex = 3 ' rouge
ElseIf ActiveSheet.Cells(1, 1).Interior.ColorIndex = 4 Then
    ActiveSheet.Cells.Select
    Selection.Interior.ColorIndex = 4 'vert
ElseIf ActiveSheet.Cells(1, 1).Interior.ColorIndex = 5 Then
    ActiveSheet.Cells.Select
    Selection.Interior.ColorIndex = 5 'bleu
Else
    ActiveSheet.Cells.Select
    Selection.Interior.ColorIndex = 6 'jaune
End If
End Sub

```

Exercice 20 – Structures de contrôle – Select case

Créez une fonction nommée TotalAPayer qui calculera le Total à payer. Vous devrez fournir 3 arguments à la fonction

- la quantité
- le prix HT
- le port

En fonction de la quantité, vous accorderez une remise :

- 1 pièce pas de remise
- de 2 jusque 10, remise de 1%
- de 11 à 20, remise de 5%
- de 21 à 30, remise de 10%
- de 31 à 40, remise de 15%
- de 41 à 50, remise de 20%
- Si la commande est supérieure à 50 pièces : 'erreur dans la quantité' doit apparaître

```

Function TotalAPayer(quantité, HT, port) 'exercice 20 'permet de calculer
le prix total par rapport à la quantité commandé
' selon la quantité la remise évolue
    Select Case quantité
        Case 1
            TotalAPayer = HT + port
        Case 2 To 10
            TotalAPayer = (HT + port * 0.99)*quantité
        Case 11 To 20
            TotalAPayer = (HT + port * 0.95)*quantité
        Case 21 To 30
            TotalAPayer = (HT + port * 0.9)*quantité
        Case 31 To 40
            TotalAPayer = (HT + port * 0.85)*quantité
        Case 41 To 50
            TotalAPayer = (HT + port * 0.8)*quantité
        Case Else
            TotalAPayer = "Erreur dans la quantité"
    End Select
End Function

```

Exercice 21 – Structures de contrôle – Do While ...loop

Créez une procédure qui :

- Demandra à l'utilisateur de taper un nombre de 3 chiffres maximum dans une 'inputbox'.
- Tant que la somme des nombres tapés est inférieure à 1000, l'inputbox doit se réafficher pour taper une nouvelle valeur.

```
Sub Exercice21()  
    Static Total As Integer  
    Dim nombre As Integer  
    Do While Total < 1000  
        nombre = InputBox("Entrez un nombre de 3 chiffres maximum")  
        Total = Total + nombre  
    Loop  
End Sub
```

Exercice 22 – Structures de contrôle – Do until ...loop

Créez une procédure qui :

- Demandra à l'utilisateur de taper une date dans une 'inputbox'.
- Tant que la frappe n'est pas reconnue comme une date par le système, il faut réafficher l'inputbox.

```
Sub Exercice22() 'procédure de vérification d'une entrée  
    Do Until IsDate(Range("A1"))  
        Range("A1") = InputBox("Entrez la date désirée")  
    Loop  
End Sub
```

Exercice 23 – Structures de contrôle – For...next

Renommez vos feuilles :

- 1 = Tournai
- 2 = Mons
- 3 = Charleroi

Créez une procédure qui :

- rendra la feuille 'Charleroi' active
- sélectionnera la cellule E5 de cette feuille
- inscrira dans cette cellule 'Je suis dans la cellule 1 de la colonne E'
- descendra d'1 cellule et y inscrira ' je suis dans la cellule 2 de la colonne E'
- et ainsi de suite jusque la ligne 1000

```
Sub Exercice23()  
    Dim i As Integer  
    Application.Sheets(3).Activate  
    Cells(1, 5).Select  
    For i = 1 To 1000  
        Selection.Value = "je suis dans la cellule N° " & i & "de la  
        colonne E"  
        Selection.Offset(1, 0).Select  
    Next i  
End Sub
```

```
Next i
End Sub
```

Exercice 24 – Structures de contrôle – For...next Step

Même exercice que le précédent mais cette fois-ci l'inscription doit se faire toutes les 5 lignes.

```
Sub Exercice24()
Dim i As Integer
Application.Sheets(3).Activate
Cells(1, 5).Select
For i = 1 To 20 Step 5
    Selection.Value = "je suis dans la cellule N° " & i & "de la
    colonne E"
    Selection.Offset(5, 0).Select
Next i
End Sub
```

Exercice 25– Structures de contrôle – For Each...next

Créez une procédure qui :

- Renommara toutes les feuilles du classeur actif
- La première en 1
- La deuxième en 2
- Et ainsi de suite

```
Sub Exercice25()
Dim LesFeuilles As Sheets
Dim LaFeuille As Worksheet
Dim i As Integer
i = 1

Set LesFeuilles = Sheets
For Each LaFeuille In LesFeuilles
    LaFeuille.Name = i
    i = i + 1
Next
End Sub
```

Exercice 26– Structures de contrôle – Gestion des erreurs – Variables tableaux

Créez une procédure qui :

- Empêchera l'affichage des messages d'alerte d'Excel
- Fera en sorte que le classeur comporte 12 feuilles (si trop peu ajout, trop retrait) nommées respectivement janvier, février, ..., décembre
- Y inclure une gestion des erreurs qui :
 - o affichera en cas d'erreur
 - Le numéro de l'erreur
 - L'intitulé de l'erreur
 - o Passera au-dessus de l'erreur pour continuer la procédure

```
Sub Exercice26()
```

```

'erreurs que faire ?
On Error GoTo erreur

'déclaration
Dim LesFeuilles As Sheets
Dim LaFeuille As Worksheet
Dim i As Integer
Dim NbFeuille As Integer
Dim mois(12) As String
Dim Différence As Integer
Dim Année As String

Set LesFeuilles = Sheets
mois(1) = "Janvier"
mois(2) = "Février"
mois(3) = "Mars"
mois(4) = "Avril"
mois(5) = "Mai"
mois(6) = "Juin"
mois(7) = "Juillet"
mois(8) = "Août"
mois(9) = "Septembre"
mois(10) = "Octobre"
mois(11) = "Novembre"
mois(12) = "Décembre"

Année = Year(Date)

'je coupe les messages d'alerte
Application.DisplayAlerts = False

'calcul du nombre de feuille
NbFeuille = Application.ActiveWorkbook.Sheets.Count

'calcul du nombre de feuille en trop ou trop peu
Différence = NbFeuille - 12

'ajout ou retrait de feuille
Select Case Différence
    Case 0
        'je continue le traitement sans rien faire

    Case Is < 0
        ' il manque des feuilles, il faut en ajouter la différence
        For i = 1 To -Différence
            Application.ActiveWorkbook.Sheets.Add
        Next i

    Case Is > 0
        ' il y en a trop, il faut en enlever
        For i = 1 To Différence
            Application.ActiveWorkbook.Sheets(1).Delete
        Next i
End Select

'renommer les feuilles
i = 1
For Each LaFeuille In LesFeuilles
    LaFeuille.Name = mois(i) & " " & Année
    i = i + 1
Next

```

```

'je réactive les messages d'alerte
Application.DisplayAlerts = True

' sortie de la procédure sinon je continue vers le traitement des erreurs
Exit Sub

'traitement des erreurs
erreur:
Application.DisplayAlerts = False
MsgBox Err & vbCrLf & Error
Resume Next

End Sub

```

Exercice 27– Boîtes de dialogue intégrées - Propriétés

Créez une procédure qui :

- Affichera le nombre de boîtes de dialogue intégrées
- Affichera ensuite la boîte de dialogue du tri

```

Sub Exercice27()
    MsgBox Application.Dialogs.Count
    Application.Dialogs(xlDialogSort).Show
End Sub

```

Exercice 28– Boîtes de dialogue intégrées - Méthodes

Créez une procédure qui :

- Affichera une boîte de dialogue qui me permettra de choisir un ou plusieurs fichiers
 - o Excel
 - o Titre de la boîte : Sélectionne un fichier
 - o Je peux choisir plusieurs fichiers
- Après validation une boîte de message s'affichera avec le nom et le chemin de chaque fichier sélectionné sur une ligne différente

```

Sub Exercice28()
    Dim fichier, f, Message As String
    fichier = Application.GetOpenFilename("fichier excel(*.xls), *.xls",
    , "Choisi le fichier", , True)

    For Each f In fichier
        Message = Message & vbCrLf & f
    Next
    MsgBox Message

End Sub

```

Exercice 29– Boîtes de dialogue intégrées – Méthodes

Créez une procédure qui :

- Affichera une boîte de dialogue qui me permettra de choisir un fichier
 - o Excel

- Titre de la boîte : Ouvrir fichier
- Je peux choisir un fichier
- Ouvrira le fichier sélectionné

```
Sub Exercice29()
    Dim fichier
    fichier = Application.GetOpenFilename("fichier excel(*.xls), *.xls",
, "Choisi le fichier à ouvrir", , False)
    Application.Workbooks.Open (fichier)
End Sub
```

Exercice 30– Boîtes de dialogue intégrées – Méthodes

Créez une procédure qui :

- Affichera une boîte de dialogue qui me permettra de donner un nom et un endroit de stockage au fichier en cours
- Enregistrera réellement le fichier

```
Sub Exercice30()
    Dim fichier
    fichier = Application.GetSaveAsFilename()
    Application.ActiveWorkbook.SaveAs (fichier)
End Sub
```

Exercice 31 – Boîtes de dialogue prédéfinies - Inputbox

Créez une procédure qui :

- Affichera dans une boîte de dialogue, sur des lignes distinctes, la moyenne, le maximum et le minimum d'une plage que le système vous invitera à sélectionner.

```
Sub Exercice31()
    On Error GoTo erreur ' gestion des erreurs

    Dim Plage As Range ' la plage sur laquelle porte le calcul
    Dim Moyenne As Single, Maximum As Single, Minimum As Single 'les calculs
    Dim Message As String, Titre As String 'les messages
    Dim C As Variant ' les cellules de la plage
    Dim Vide As Boolean ' test de la cellule vide ou non

    Application.DisplayAlerts = False 'suppression des messages d'alerte

    'message et titre de inputbox
    Message = "Sélectionne la(les) plage(s) sur laquelle (lesquelles) doivent
porter les calculs"
    Titre = "Sélection de plage(s)"

    'affectation de la plage
    'boucle do pour empêcher d'aller plus loin si pas de sélection
    Do
    Set Plage = Application.InputBox( _
        prompt:=Message, _
        Title:=Titre, _
        Type:=8)
    Loop While IsEmpty(Plage)

    'traitement
    Moyenne = Application.WorksheetFunction.Average(Plage)
```

```

Maximum = Application.WorksheetFunction.Max(Plage)
Minimum = Application.WorksheetFunction.Min(Plage)

'message et titre de la msgbox
Titre = "Résultat des calculs sur la plage: " & Plage.Address
Message = "Les résultats demandés: " & vbCrLf _
& "La moyenne est égale à: " & Moyenne & vbCrLf _
& "Le maximum est égal à: " & Maximum & vbCrLf _
& "La minimum est égale à: " & Minimum & vbCrLf _
& vbCrLf & " Merci de votre visite" & " " & Application.UserName

MsgBox _
    prompt:=Message, _
    Title:=Titre

'remise en activité des messages d'alerte
Application.DisplayAlerts = True
'sortie de l'application si pas d'erreur
'sinon le traitemnt d'erreur a lieu

Exit Sub

'traitement en cas d'erreur
erreur:
Application.DisplayAlerts = True

    For Each C In Plage
        ' si une des cellules contient une valeur alors ok
        If C.Value <> "" Then
            Vide = False
            Exit For
        Else: Vide = True
        End If
    Next
    If Vide = True Then
        MsgBox "Plage vide"
        Exit Sub
    End If

MsgBox Err & " " & Error
Exit Sub
End Sub

```

Exercice 32 – Boîtes de dialogue prédéfinies – MsgBox

Créez une procédure qui :

- Affichera une boîte de dialogue qui vous demandera si vous voulez oui ou non effacer la(les) colonnes(s) que vous avez sélectionné.
- La boîte de dialogue comprendra les boutons Oui et non et un point d'exclamation. De plus, le bouton OK devra avoir le focus.

```

Sub exercice32()
Dim Colonnes As Range, Colo As Range, C As Variant
Dim Confirmation As Integer
Dim Message As String

Application.DisplayAlerts = False

```

```

Do
Set Colonnes = Application.InputBox( _
    prompt:="Sélectionne la ou les colonnes que tu veux
supprimer", _
    Title:="Sélection", _
    Type:=8)

Loop While IsEmpty(Colonnes)

' je prévois que l'utilisateur ne sélectionne que des cellules
Set Colonnes = Colonnes.EntireColumn
Set Colo = Colonnes.Columns

For Each C In Colo
    Message = Message & " " & C.Address
Next

Confirmation = MsgBox(prompt:="Voulez-vous supprimer les colonnes: " &
Message, _
    Buttons:=vbYesNo + vbExclamation + vbDefaultButton2,
    Title:="Confirmation")

If Confirmation = vbYes Then
    Colonnes.Delete
ElseIf Confirmation = vbNo Then
    MsgBox "Vous avez renoncé à effacer les colonnes sélectionnées", ,
"Renoncement"
End If

Application.DisplayAlerts = True
End Sub

```

Exercice 33 – Formulaires

Créez un formulaire 'calculatrice' qui sera capable de :

- Effectuer les 4 opérations fondamentales
- Attention à la division par 0
- Les différents résultats doivent s'inscrire à partir de la cellule A1 de la feuille 1 et descendre d'une cellule à chaque opération même lorsque Excel a été fermé.
- Le formulaire doit s'afficher à l'ouverture du fichier
- Le fichier doit s'enregistrer quand vous cliquez sur le bouton Quitter

```

Private Sub B_Calcul_Click()
On Error Resume Next
' Vérification si une valeur numérique a été introduite dans les
If Me.T_Valeur1 = "" Then
    MsgBox "Veuillez taper une première valeur"
    Me.T_Valeur1.SetFocus
Exit Sub
ElseIf Me.T_Valeur2 = "" Then
    MsgBox "Veuillez taper une deuxième valeur"
    Me.T_Valeur2.SetFocus
Exit Sub
End If

```

```

' selon l'opération choisie, lancer une procédure
Select Case Me.C_Choix
    Case Is = "Addition"
        Me.T_Resultat = CLng(Me.T_Valeur1) + CLng(Me.T_Valeur2)
    Case Is = "Soustraction"
        Me.T_Resultat = Me.T_Valeur1 - Me.T_Valeur2
    Case Is = "Multiplication"
        Me.T_Resultat = Me.T_Valeur1 * Me.T_Valeur2
    Case Is = "Division"
        If Me.T_Valeur2 = 0 Then
            MsgBox "Division par zéro impossible"
            Me.T_Valeur2.SetFocus
            Exit Sub
        End If
        Me.T_Resultat = Me.T_Valeur1 / Me.T_Valeur2
    Case Else
        MsgBox "Vous n'avez pas choisi d'opérateur"
        Me.C_Choix.SetFocus
End Select
If CLng(Me.T_Resultat.Value) < 0 Then
    Me.T_Resultat.ForeColor = RGB(255, 0, 0)
Else
    Me.T_Resultat.ForeColor = RGB(0, 0, 0)
End If
'appel de la procédure d'inscription
InscriptionFeuille (CLng(Me.T_Resultat.Value))

End Sub

```

```

Private Sub B_Quitter_Click()
'enregistrement du fichier
Application.ActiveWorkbook.Save
End
End Sub

```

```

Private Sub B_Reset_Click()
On Error Resume Next
    Me.T_Valeur1 = ""
    Me.T_Valeur2 = ""
    Me.C_Choix = ""
    Me.T_Resultat = ""
    Me.T_Valeur1.SetFocus
End Sub

```

```

Private Sub C_Choix_Change()
On Error GoTo erreur

Select Case Me.C_Choix
    Case Is = "Addition"
        Me.L_Operation = " + "
    Case Is = "Soustraction"
        Me.L_Operation = " - "
    Case Is = "Multiplication"
        Me.L_Operation = " * "
    Case Is = "Division"
        Me.L_Operation = " / "
End Select

```

```

Exit Sub
erreur:
    MsgBox Err & " " & Error
Exit Sub
End Sub

```

```

Private Sub UserForm_Initialize()
On Error Resume Next
    Me.C_Choix.AddItem "Addition"
    Me.C_Choix.AddItem "Soustraction"
    Me.C_Choix.AddItem "Multiplication"
    Me.C_Choix.AddItem "Division"
End Sub
Sub InscriptionFeuille(LeResultat) ' p&s bon
    Application.Sheets(1).Activate
    Range("A1").Select
    If Selection.Value = "" Then
        Selection.Value = LeResultat
    Else
        Selection.Offset(1, 0).Select
        Do While Selection <> ""
            Selection.Offset(1, 0).Select
        Loop
        Selection.Value = LeResultat
    End If
End Sub

```

The screenshot shows a VBA UserForm window titled "Calculer". The window has a standard Windows title bar with a close button (X). The form is divided into several sections:

- Input Section:** At the top, there are two text boxes labeled "Valeur 1" and "Valeur 2". Above the first text box is a label "I Onerati". To the right of these is a dropdown menu labeled "Choix de l'opérateur". Below each text box and the dropdown menu are yellow callout boxes labeled "T Valeur" and "C Choix" respectively.
- Output Section:** In the center of the form is a larger text box labeled "T Result".
- Control Section:** At the bottom right, there are three buttons stacked vertically: "B_Calcul" (with the text "Calculer" next to it), "B_Reset" (with "Reset" next to it), and "B_Quitter" (with "Quitter" next to it).
- Form Label:** At the bottom left, there is a label "I UserForm".

Chapitre I.....	2
Objectifs.....	2
Groupe cible :.....	3
Prérequis :	3
Durée :.....	3
Objectifs :.....	3
Contenu :	3
Chapitre II.....	4
Énoncés	4
Enregistreur de macros	5
Exercice 1 :.....	5
Exercice 2 :	5
Exercice 3:.....	5
Exercice 4 :.....	5
Création d'une procédure Sub	5
Exercice 5 :.....	5
Exercice 6 :.....	6
Création d'une fonction personnalisée	6
Exercice 7 :.....	6
Déclaration implicite des variables	6
Déclaration explicite des variables	6
Exercice 8.....	6
Déclaration d'une constante	7
Exercice 9.....	7
Structure du langage.....	7
Exercice 10 – affectation d'une valeur à une propriété	7
Exercice 11 – Récupération de la valeur d'une propriété	7
Exercice 12 – Syntaxe de l'utilisation d'une méthode.....	7
Exercice 13 – Règles d'écriture	7
Propriétés de l'inputbox	7
Exercice 14 – Structures de contrôle – With (plusieurs actions sur un objet)	7
Exercice 15 – Structures de contrôle – With (plusieurs actions sur un objet)	8
Exercice 17 – Structures de contrôle – If...then...end if	8
Exercice 18 – Structures de contrôle – If...then...else...end if	8
Exercice 19 – Structures de contrôle – If...then...elseif...else...end if	8
Exercice 20 – Structures de contrôle – Select case	8
Exercice 21 – Structures de contrôle – Do While ...loop	9
Exercice 22 – Structures de contrôle – Do until ...loop.....	9
Exercice 23 – Structures de contrôle – For...next.....	9
Exercice 24 – Structures de contrôle – For...next Step	9
Exercice 25– Structures de contrôle – For Each...next	9
Exercice 26– Structures de contrôle – Gestion des erreurs – Variables tableaux.....	10
Exercice 27– Boîtes de dialogue intégrées - Propriétés.....	10
Exercice 28– Boîtes de dialogue intégrées - Méthodes	10

Exercice 29– Boîtes de dialogue intégrées – Méthodes.....	10
Exercice 30– Boîtes de dialogue intégrées – Méthodes.....	10
Exercice 31 – Boîtes de dialogue prédéfinies - Inputbox.....	10
Exercice 32 – Boîtes de dialogue prédéfinies – MsgBox	11
Exercice 33 – Formulaire	11
Chapitre III.....	12
Corrigés	12
Enregistreur de macros	13
Exercice 1 :.....	13
Exercice 2 :.....	13
Exercice 3:.....	13
Exercice 4 :.....	13
Création d’une procédure Sub	13
Exercice 5 :.....	13
Exercice 6 :.....	14
Création d’une fonction personnalisée	14
Exercice 7 :.....	14
Déclaration implicite des variables	14
Déclaration explicite des variables	14
Exercice 8	14
Déclaration d’une constante	15
Exercice 9	15
Exercice 10 – affectation d’une valeur à une propriété	15
Exercice 11 – Récupération de la valeur d’une propriété	16
Exercice 12 – Syntaxe de l’utilisation d’une méthode.....	16
Exercice 13 – Règles d’écriture	16
Propriétés de l’inputbox	16
Exercice 14 – Structures de contrôle – With (plusieurs actions sur un objet)	17
Exercice 15 – Structures de contrôle – With (plusieurs actions sur un objet)	18
Exercice 16 – Structures de contrôle – If...then.....	18
Exercice 17 – Structures de contrôle – If...then...end if.....	18
Exercice 18 – Structures de contrôle – If...then...else...end if.....	19
Exercice 19 – Structures de contrôle – If...then...elseif...else...end if	19
Exercice 20 – Structures de contrôle – Select case	20
Exercice 21 – Structures de contrôle – Do While ...loop.....	21
Exercice 22 – Structures de contrôle – Do until ...loop.....	21
Exercice 23 – Structures de contrôle – For...next.....	21
Exercice 24 – Structures de contrôle – For...next Step	22
Exercice 25– Structures de contrôle – For Each...next	22
Exercice 26– Structures de contrôle – Gestion des erreurs – Variables tableaux.....	22
Exercice 27– Boîtes de dialogue intégrées - Propriétés.....	24
Exercice 28– Boîtes de dialogue intégrées - Méthodes	24
Exercice 29– Boîtes de dialogue intégrées – Méthodes.....	24
Exercice 30– Boîtes de dialogue intégrées – Méthodes.....	25

Exercice 31 – Boîtes de dialogue prédéfinies - Inputbox.....	25
Exercice 32 – Boîtes de dialogue prédéfinies – MsgBox	26
Exercice 33 – Formulaires	27