

- I. Mise en place des données de travaux :
 - a. L'environnement

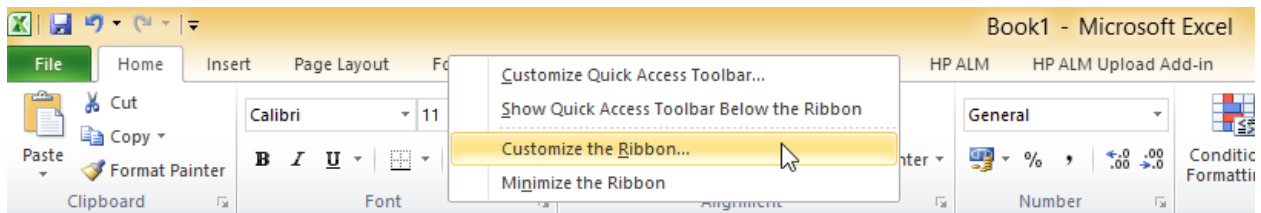
Sur Excel tu vas rentrer un certain nombre de champs, l'objectif est d'insérer un tableau avec les index de colonne suivante : Nom, Prenom, Age, Depart, Arrivé, Temps :

Nom	Prenom	Age	Depart	Arrivé	Temps

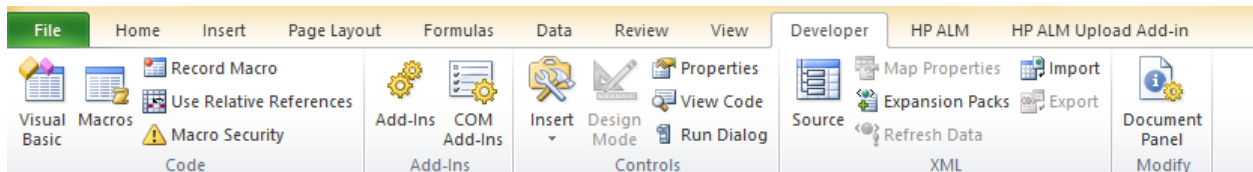
Par la suite, on va vouloir remplir automatique le tableau pour se faire une base de travail aléatoire.

Du coups, tout d'abord tu vas :

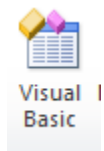
Clic droit sur le ruban -> customiser le ruban :



Et tu ajoutes l'onglet Developer :



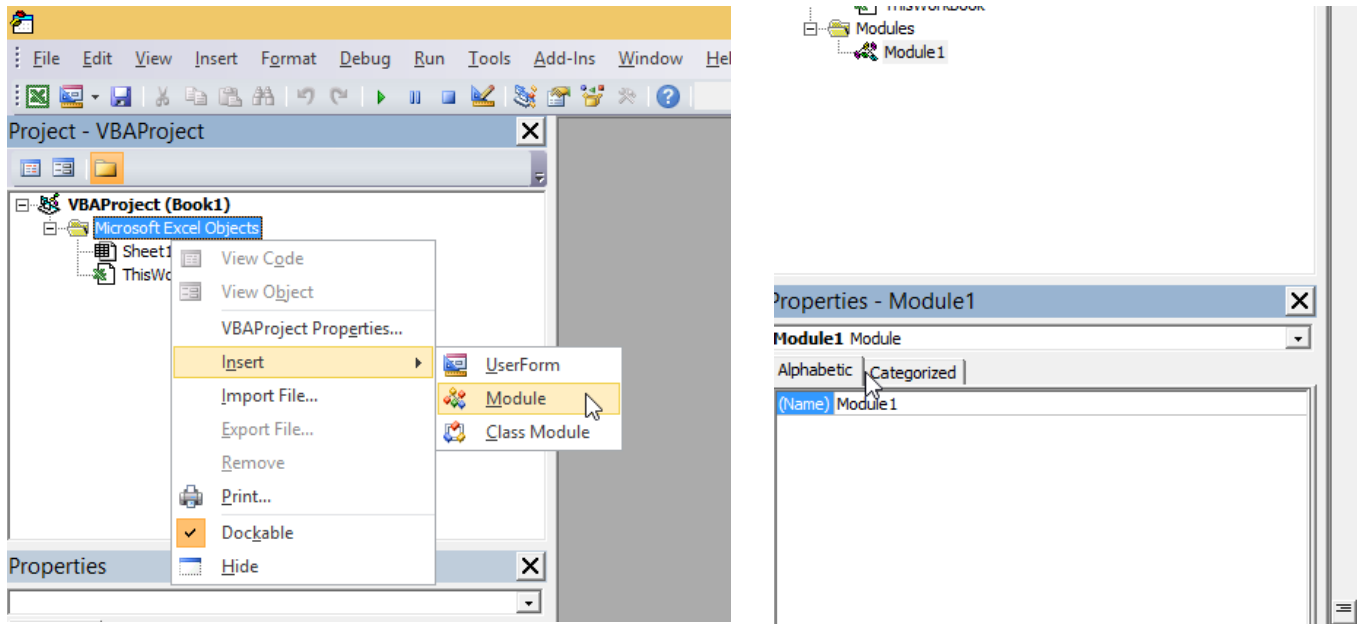
Ensuite tu pourras rentrer dans l'environnement en cliquant sur :



b. Créer son premier module :

Dans VBA tu peux faire un grand nombre de chose, du peux coder directement dans les «Feuille de ton classeur » mais le soucis c'est que sa peut tres vite entrainer des erreurs. Sauf quand tu veux lancer des methode autoatiquement sur une action sur ta feuille : comme un double clics sur une cellule dans la colonne C

Tu vas donc pour etre bien et safe : Inserer un nouveau module qui te servira de base pour ton code :



Une fois ton module créer tu peux voir ses propriétés plus bas :

Tu pourrais le renommer par m_function par exemple pour t'y retrouver si tu as pleins de fonction plus tard. Tu pourrais avoir plusieurs modules pour être mieux organisé : m_function / m_front etc.....

c. Début de code :

Maintenant tu vas pouvoir coder je vais te montrer deux petits exemple et ensuite je te laisserai t'amuser :

```
(General)  Fonction
Option Explicit 'Ultra important pour pas faire n'imp' permet de forcer la declaration des variables.

Sub Methode(Nom As String) 'Pour declarer une méthode qui ne renvoi rien
    MsgBox Nom 'Permet d'afficher un message dans une boite
End Sub

Function Fonction(n As Integer) As Integer 'Pour declarer une fonction qui renvoi un entier
    n = n * 2
    Fonction = n 'Le nom de ta fonction devient une variable qui sera retourné a la fin
End Function
```

d. Gérer les objets Excel :

Tu as pleins d'objets dans Excel, tout est Objet, les feuilles, ton fichier, tes graphiques, tes images inséré (ou «Shapes ») bref tu peux faire beaucoup de chose :

Je vais te montrer le début pour remplir notre tableau :

Option Explicit 'Ultra important pour pas faire n'imp' permet de forcer la declaration des variables.

```
Sub RemplirTableau()  
  
Dim MonTableau As ListObject 'Les tableaux Excel sont des ListObject  
Set MonTableau = Sheet1.ListObjects(1) 'Tu peux recuperer ton tableau par son numero d'index sur ta feuille  
Set MonTableau = ThisWorkbook.Worksheets(1).ListObjects("Table1") ' Ou bien par son nom en partant d'un chemin plus general  
  
Dim cell As Range 'Les cellules ou bien groupe de cellule sont des ranges, on peut faire bcp de chose avec  
Dim i As Integer, n As Integer  
  
If MonTableau.ListRows.Count = 0 Then 'Structure du If, il faut toujours l'ouvrir et le fermer avec un end  
    MonTableau.ListRows.Add 'Si mon tableau est vide je lui rajoute une ligne  
End If  
  
i = 1  
While i <= 10 ' structure du While pareil on le ferme toujours avec un W(hile)END  
    MonTableau.ListRows.Add  
    i = i + 1 'La structure i ++ ou i+= ne marche pas :/  
Wend  
  
End Sub
```

On obtient :

Nom	Prenom	Age	Depart	Arrivé	Temps

On va essayer maintenant de le remplir :

e. Remplir un Tableau :

```

Dim MonTableau As ListObject 'Les tableaux Excel sont des ListObject
Set MonTableau = Sheet1.ListObjects(1) 'Tu peux recuperer ton tableau par son numero d'index sur ta feuille
Set MonTableau = ThisWorkbook.Worksheets(1).ListObjects("Table1") ' Ou bien par son nom en partant d'un chemin plus general

Dim cell As Range 'Les cellules ou bien groupe de cellule sont des ranges, on peut faire bcp de chose avec
Dim i As Integer, n As Integer

If MonTableau.ListRows.Count = 0 Then 'Structure du If, il faut toujours l'ouvrir et le fermer avec un end
    MonTableau.ListRows.Add 'Si mon tableau est vide je lui rajoute une ligne
End If

Dim DernierElement As Integer: DernierElement = MonTableau.ListRows.Count

For i = 1 To DernierElement 'Structure du For on le termine par next
    Randomize
    MonTableau.ListRows(i).Range(1) = "ABC" 'Du coups on va parcourir la premiere colonne,
    MonTableau.ListRows(i).Range(2) = "LOLILOL" 'Deuxieme Colonne
    MonTableau.ListRows(i).Range(3) = Int(30 * Rnd) + 11 'Nombre aleatoire entre 11 et 30

    MonTableau.ListRows(i).Range(4) = DateAdd("d", -1 * (Int(6 * Rnd) + 3), Date) 'Je lui retire entre 3 et 6 Jours
    MonTableau.ListRows(i).Range(5) = Date
    'Ici j' utilise deux fonctions :
    ' Date : Qui me renvoi la date d'aujourd'hui
    ' DateAdd : Qui permet de faire des additions de date facilement, je lui dis que je veux ajouter des
    ' jour avec le "d" = "days"
    MonTableau.ListRows(i).Range(6) = MonTableau.ListRows(i).Range(5) - MonTableau.ListRows(i).Range(4)
Next
End Sub

```

Resultat :

Nom	Prenom	Age	Depart	Arrivé	Temps
ABC	LOLILOL	33	03/06/2018	06/06/2018	3
ABC	LOLILOL	40	02/06/2018	06/06/2018	4
ABC	LOLILOL	16	01/05/2018	06/06/2018	6
ABC	LOLILOL	19	30/05/2018	06/06/2018	7
ABC	LOLILOL	15	03/06/2018	06/06/2018	3
ABC	LOLILOL	37	02/06/2018	06/06/2018	4
ABC	LOLILOL	25	01/06/2018	06/06/2018	5
ABC	LOLILOL	15	29/05/2018	06/06/2018	8
ABC	LOLILOL	13	03/06/2018	06/06/2018	3
ABC	LOLILOL	15	02/06/2018	06/06/2018	4

Voila maintenant tu as les bases pour tester des trucs, je t'ai montré comment, définir tes variables et tes objets avec : « DIM », tu initialise tes objets avec « SET ».

Tout les éléments d'Excel sont des objets donc tu veux souvent les utiliser directement :

Sheet2.ListObjects(1).Select

Les fonctions et méthode commence par Sub ou function et termine par END SUB ou END FUNCTION

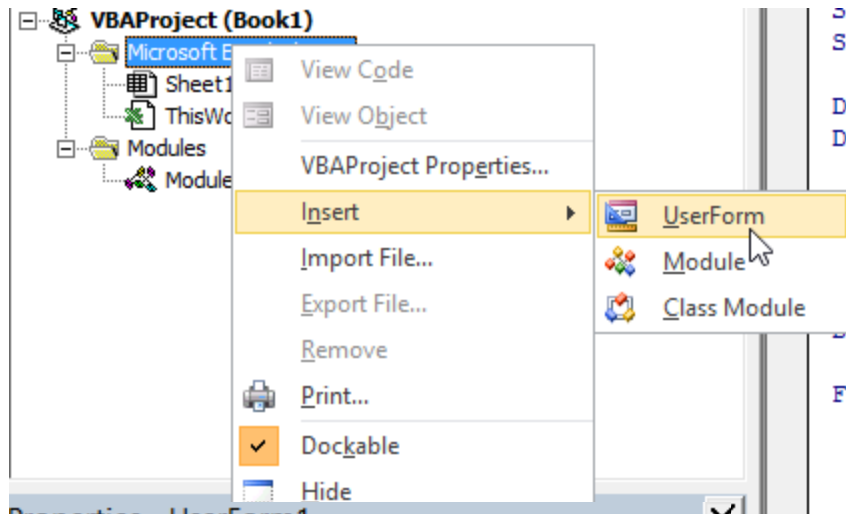
De meme pour les boucles : For / Next ; While / Wend ; If then else End If ;

Toujours penser a coder avec Option Explicit en haut de code sa evite un paaquet d'erreurs !!!

II. Les UserForms

Ce qui est cool en VBA c'est que tu peux faire des fenetres et du front tres rapidement et tres facilement :

On va inserer un UF :



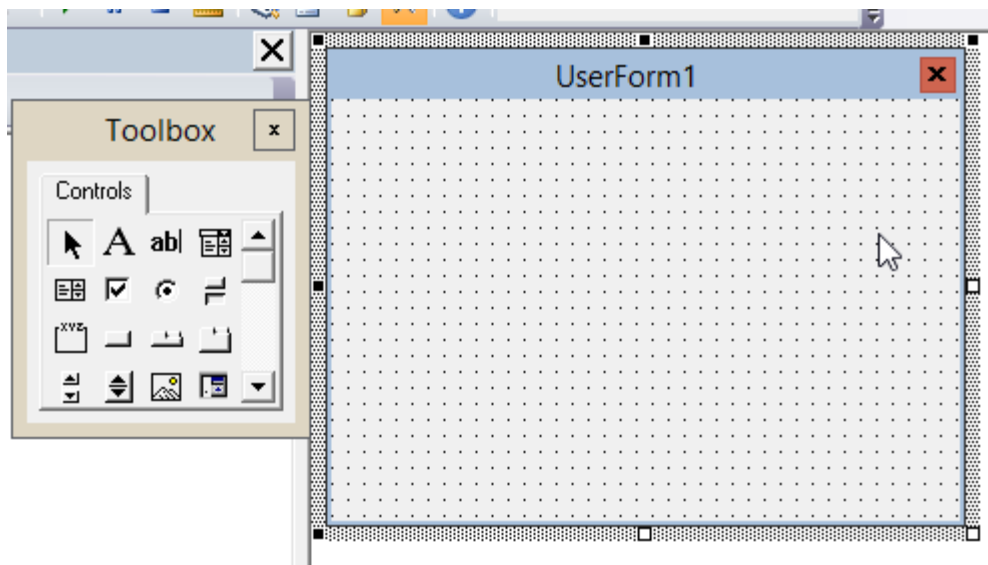
Comme pour le module tu trouveras toutes les options sous la bête que tu auras créé.

Properties - UserForm1

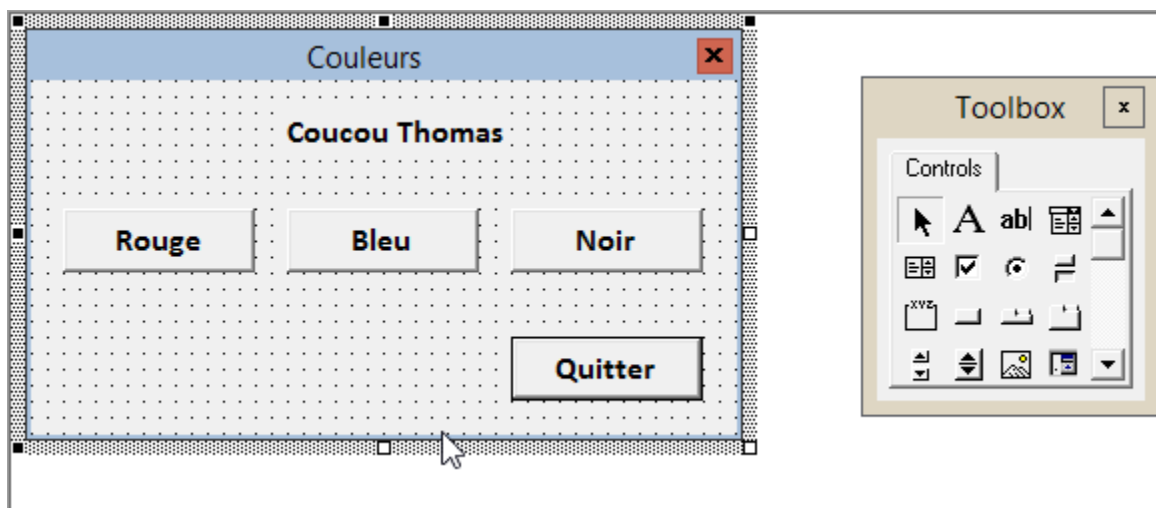
UserForm1 UserForm

Alphabetic | Categorized

(Name)	UserForm1
BackColor	<input type="checkbox"/> &H8000000F&
BorderColor	<input type="checkbox"/> &H80000012&
BorderStyle	0 - fmBorderStyleNone
Caption	UserForm1
Cycle	0 - fmCycleAllForms
DrawBuffer	32000
Enabled	True
Font	Tahoma
ForeColor	<input type="checkbox"/> &H80000012&
Height	180
HelpContextID	0
KeepScrollBarsVisible	3 - fmScrollBarsBoth
Left	0
MouseIcon	(None)
MousePointer	0 - fmMousePointerDefault
Picture	(None)
PictureAlignment	2 - fmPictureAlignmentCenter
PictureSizeMode	0 - fmPictureSizeModeClip
PictureTiling	False
RightToLeft	False
ScrollBars	0 - fmScrollBarsNone
ScrollHeight	0
ScrollLeft	0
ScrollTop	0
ScrollWidth	0



La ToolBox te permettra de faire glisser des objets et ensuite de définir des methode ou des actions en fonction de se que fait l'utilisateur. C'est tres facile tu les places en Drag and Drop.



Pour te monter un exemple, j'ai fais glisser trois boutons sur ma UserForm et un champs de Text

Button : 3eLigne 2eCol / Text : 1eLigne 2eCol

Tu as pleins de possibilité d'objets !

Pour rentrer dans le code on va DOUBLE CLIQUER SUR UN OBJET :

On peut donc faire se genre de chose :

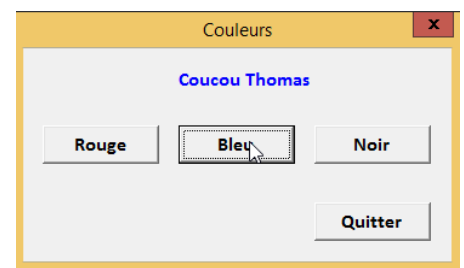
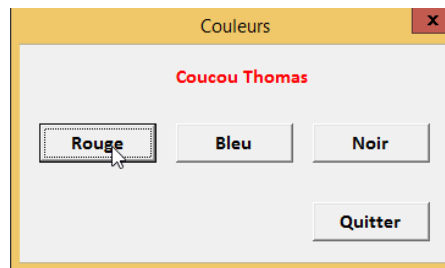
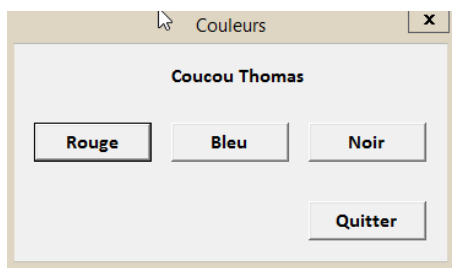
Option Explicit

```
Private Sub RougeBut_Click()  
    Titre.ForeColor = RGB(255, 0, 0)  
End Sub
```

```
Private Sub BleuBut_Click()  
    Titre.ForeColor = RGB(0, 255, 0)  
End Sub
```

```
Private Sub NoirBut_Click()  
    Titre.ForeColor = RGB(0, 0, 0)  
End Sub
```

```
Private Sub ExitBut_Click()  
    Unload Me  
End Sub
```



Comme tu peux le voir on peut faire pleins de chose, a toi après d'explorer et de faire des trucs plus utile !