

# Ajouter un contrôle Listview à un formulaire

Continuons notre exemple d'utilisation d'un contrôle Listview dans un Userform.

## Les données sources

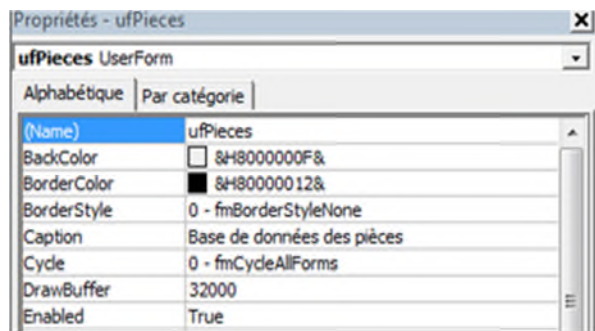
Pour commencer, nous avons besoin d'une base de données, une liste d'items. Pour notre exemple, nous allons construire un Listview à partir d'une liste de pièces contenues dans un classeur.

No Piece	Description	Fabricant	No Fabricant	Prix	Devis	Catégorie
A01101	Vis	Bizzbizz	1893142	2.01	CAD	1
A22314	Boulon	Dollaramama	A283.153.2	1.38	CAD	2
A34125	Plaque de support	MacMouf	95345KA	70.25	CAD	1
A12715	Fenêtre	Happlez	3034-ND	90	US	4
A17352	Boîtier	Super Top Inc.	N0135262	300.5	CAD	9

## Création du formulaire et du Listview

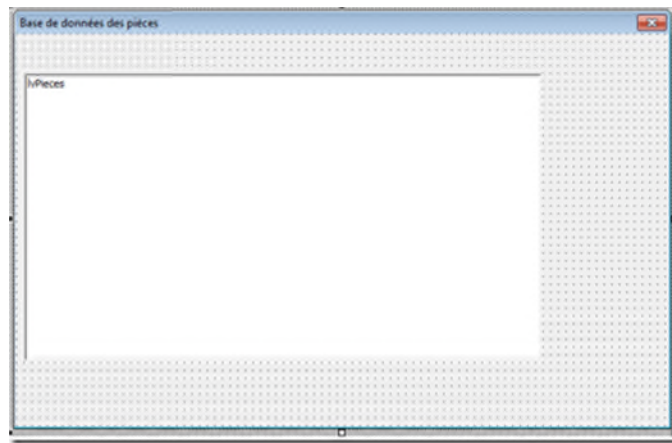
Maintenant, créons notre formulaire.

- Aller dans l'éditeur VBA et insérer un formulaire (Insertion | Userform)
- Changer le nom (name) de cet userform par **ufPieces**
- Changer le caption pour "**Base de données des pièces**"



- Faire apparaître votre Boîtes à Outils (Affichage | Boîte à outils)
- Cliquer sur l'icône du contrôle Listview et ajouter un Listview à votre Userform
- Cliquer sur le Listview pour le sélectionner et changer son nom pour "lvPieces"
- Ajuster les dimensions au besoin, pour avoir à peu près ceci :





Nous venons d'ajouter notre contrôle Listview à notre formulaire. La prochaine étape consistera à l'alimenter à partir des données contenues dans notre liste de pièces, mais avant, un petit mot pour comprendre comment est construit un Listview et les contrôles de base.

## La structure d'un Listview

Il faut voir le Listview comme un tableau. Chaque ligne d'une Listview peut être définie en 2 parties :

Listview1.ListItems(x) : La ligne X et la 1re colonne de cette ligne

Listview1.ListItems(x).ListSubItems(y) : Les colonnes suivantes de la ligne X

Pour notre exemple, on pourrait illustrer ainsi :

	A	B	C	D	E	F	G	H	I
1	EXEMPLE LISTVIEW								
2									
3		No Piece	Description	Fabricant	No Fabricant	Prix	Devis	Catégorie	
4	ListItems(1) >>	A01101	Vis	Bizzbizz	189314Z	2.01	CAD	1	
5	ListItems(2) >>	A22314	Boulon	Dollaramama	A283.153.2	1.38	CAD	2	
6	ListItems(3) >>	A34125	Plaque de support	MacMouf	95345KA	70.25	CAD	1	
7	ListItems(4) >>	A12715	Fenêtre	Happlez	3034-ND	90	US	4	
8	ListItems(5) >>	A17352	Boîtier	Super Top Inc.	N0135262	300.5	CAD	9	
9									
10			.ListSubItems(1)	.ListSubItems(2)	-	-	-	.ListSubItems(7)	
11									
12									
13									
14	Note :	ListItems(x)	Les lignes de notre Listview (et la 1re colonne)						
15		ListSubItems	Les colonnes de notre Listview						
16									
17									

Ainsi, dans notre exemple

`lvPieces.ListItems(3).ListSubItems(3) = 95345KA`

## Les contrôles simples pour bâtir un Listview

Pour bâtir un Listview, on doit ajouter :

- Les colonnes (texte d'entête, dimensions, nombre, ...)
- Les lignes (ListItems)
- Les sous-éléments des lignes (ListSubItems)

### La syntaxe pour définir les colonnes est la suivante :

Listview1.ColumnHeaders.Add [Index], [Key], [Text], [Width], [Alignment], [Icon]

où :

[Index] : Facultatif. Indique le numéro de colonne

[Key] : Facultatif. Attribuer une clé unique qui facilite l'identification des colonnes.

[Text] : Facultatif. Détermine le texte qui s'affichera dans l'entête

[Width] : Facultatif. Détermine la largeur de la colonne en points. (Valeur par défaut = 72 points)

[Alignment] : Facultatif. Détermine l'alignement dans la colonne. Les valeurs possibles sont : lvwColumnLeft (par défaut), lvwColumnCenter, lvwColumnRight

[Icon] : Facultatif. Spécifie l'image qui à afficher dans l'entête.

### **La syntaxe pour ajouter une ligne est la suivante:**

ListView1.ListItems.Add [Index], [Key], [Text], [Icon], [SmallIcon]

où :

[Index] : Facultatif. Indique le numéro de ligne où ajouter le nouvel élément. Si rien n'est spécifié, les données sont ajoutées après la dernière ligne.

[Key] : Facultatif. Donner une clé unique qui facilite l'identification des lignes.

[Text] : Facultatif. Détermine le texte à afficher dans la première colonne dans le ListView.

[Icon] : Facultatif. Spécifie l'image qui s'affichera quand le ListView est dans le mode lvwIcon.

[SmallIcon] : Facultatif. Spécifie l'image qui s'affichera quand le ListView est en mode lvwList, lvwSmallIcon, lvwList.

### **La syntaxe pour ajouter un sous-élément dans une ligne est la suivante:**

ListView1.ListItems(x).ListSubItems.Add [Index], [Key], [Text], [ReportIcon], [TooltipText]

où

x: Indique le numéro de ligne dans le ListView.

[Index] : Facultatif. Indique le numéro de colonne pour ajouter une donnée. Note : 1 correspond à la 2e colonne du ListView.

[Key] : Facultatif. Donne une clé unique qui facilite l'identification des éléments.

[Text] : Facultatif. Indique le texte qui va s'afficher dans la ListView.

[ReportIcon] : Facultatif. Pour afficher un icône ou une image dans le sous-élément spécifié.

[TooltipText] : Facultatif. Pour ajouter une infobulle dans le sous-élément spécifié

Nous venons de voir les contrôles de base qui servent à construire un Listview. Nous sommes maintenant prêts à appliquer ceci à notre exemple.

## **Construire un Listview**

Dans le dernier article, nous avons vu les instructions pour ajouter des éléments. Il est maintenant temps d'appliquer ceci à notre exemple et de construire notre Listview.

- D'abord, un rappel sur les données que nous utilisons :

EXEMPLE LISTVIEW							
ColumnHeaders >>	No Piece	Description	Fabricant	No Fabricant	Prix	Devis	Catégorie
ListItems(1) >>	A01101	Vis	Bizzbiz	189314Z	2.01	CAD	1
ListItems(2) >>	A22314	Boulon	Dollaramama	A283.153.2	1.38	CAD	2
ListItems(3) >>	A34125	Plaque de support	MacMouf	95345KA	70.25	CAD	1
ListItems(4) >>	A12715	Fenêtre	Happlez	3034-ND	90	US	4
ListItems(5) >>	A17352	Boîtier	Super Top Inc.	N0135262	300.5	CAD	9
		.ListSubItems(1)	.ListSubItems(2)	...	...	...	.ListSubItems(6)

Note :

- ListItems(x) Les lignes de notre Listview (et la 1re colonne)
- ListSubItems(x) Les colonnes de notre Listview
- ColumnHeaders(x) Le nom de colonne du Listview

- Maintenant, nommons la cellule “B3” de notre tableau par “refTab”. Ceci sera très utile dans notre code VBA lorsque nous allons complexifier notre Listview. En nommant la cellule de départ, on peut ainsi l’utiliser comme référence dans notre code VBA même si on ajoute des lignes en haut de notre tableau.  
([rappel sur les champs nommés](#))
- Dans notre exemple, nous devons définir :
  - 7 entêtes de colonnes
  - 5 items
  - 6 sous-items

Pour ce faire nous utiliserons une boucle pour être plus efficace (Lean...)

- Ouvrir l’éditeur VBA et cliquer avec le bouton de droit sur votre userform nommé **ufPieces** et ensuite sur **Code**
  - Choisir **Userform** et **Initialize** dans les 2 listes en haut. La macro va en effet s’exécuter quand le userform est “initialisé” à son ouverture.
  - Ajouter le code suivant :
- 

```
Private Sub UserForm_Initialize()
'Exemple 1 : Comment alimenter un Listview
' -- Identification des variables
Dim ws As Worksheet
Dim rg As Range
Dim i As Integer
Dim j As Integer
' -- Initialisation des variables
Set ws = ThisWorkbook.Sheets("BD") 'Nom de l'onglet contenant nos données
Set rg = ws.Range("refTab") 'Notre cellule de référence de notre tableau
' -- Construction du tableau
With ufPieces.lvPieces 'Nous travaillons avec notre Listview "lvPieces" dans notre userform
' -- Ajout des colonnes
For i = 1 To 7 '7 colonnes à ajouter
'Ajouter une colonne en spécifiant le nom de l'entête seulement, largeur par défaut
'les titres sont sur la même ligne que notre référence (rg)
.ColumnHeaders.Add , , rg.Offset(0, i - 1)
Next i
' -- Ajouter les éléments de la 1re colonne
For i = 1 To 5
'Ajouter un élément en spécifiant la valeur seulement
'les éléments sont dans la même colonne que notre référence (rg)
.ListItems.Add , , rg.Offset(i, 0)
Next i
' -- Ajouter les sous-éléments
For i = 1 To 5 'Nous bouclons sur les 5 éléments (lignes)
For j = 1 To 6 'Nous bouclons sur les 6 colonnes
'Ajouter un sous-élément en spécifiant la valeur seulement
.ListItems(i).ListSubItems.Add , , rg.Offset(i, j)
Next j
Next i
' -- Spécifier l'affichage du Listview en mode "Détails"
.View = lvwReport
End With
End Sub
```

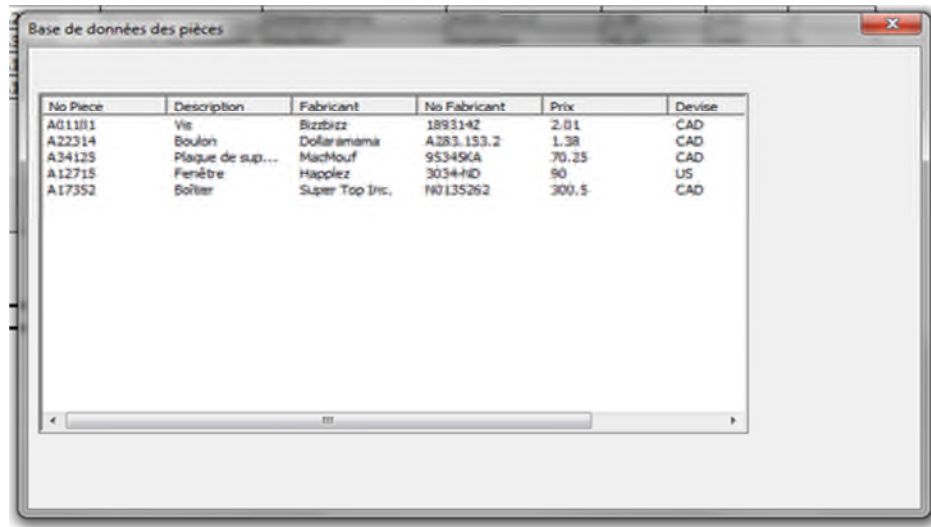
---

- Il ne reste maintenant qu'à afficher notre Formulaire. Pour ce faire, vous pouvez ajouter un bouton à votre onglet et affecter cette simple macro (à mettre dans un module)
- -----

```
Sub MontrerFormulaire()  
ufPieces.Show  
End Sub
```

-----

Vous devriez avoir ceci :



No Piece	Description	Fabricant	No Fabricant	Prix	Devise
A011011	Vis	Bizzbizz	1893142	2.01	CAD
A22314	Boulon	Dollarimamit	A283.153.2	1.38	CAD
A34125	Plaque de sup...	MacHouf	95349KA	70.25	CAD
A12715	Fenêtre	Happlez	3054-ND	90	US
A17352	Boîtier	Super Top Inc.	N0135262	300.5	CAD

Notre formulaire contient maintenant un Listview peuplé par les données de notre base de données. Il reste encore du travail à faire pour définir la largeur des colonnes, donner des clés "(Key) à nos éléments pour faciliter leur identification, etc...

En utilisant une cellule de référence et des boucles, nous avons facilement pu alimenter notre Listview.