

E.S.L.S.C.A.

Programmation VBA

Thème 2 : La gestion des Interfaces par UserForm

ATTN : ce TD contient 4 parties

Partie 1 : Quelques concepts

Les UserForms - Présentation.

La boîte à outils contient un certain nombre d'éléments prédéfinis dont nous allons vous présenter ici les plus courants. Exception faite de l'outil **Sélectionner les objets**, qui correspond à celui de la barre d'outils *Dessin*, tous ces outils s'utilisent de la même façon.

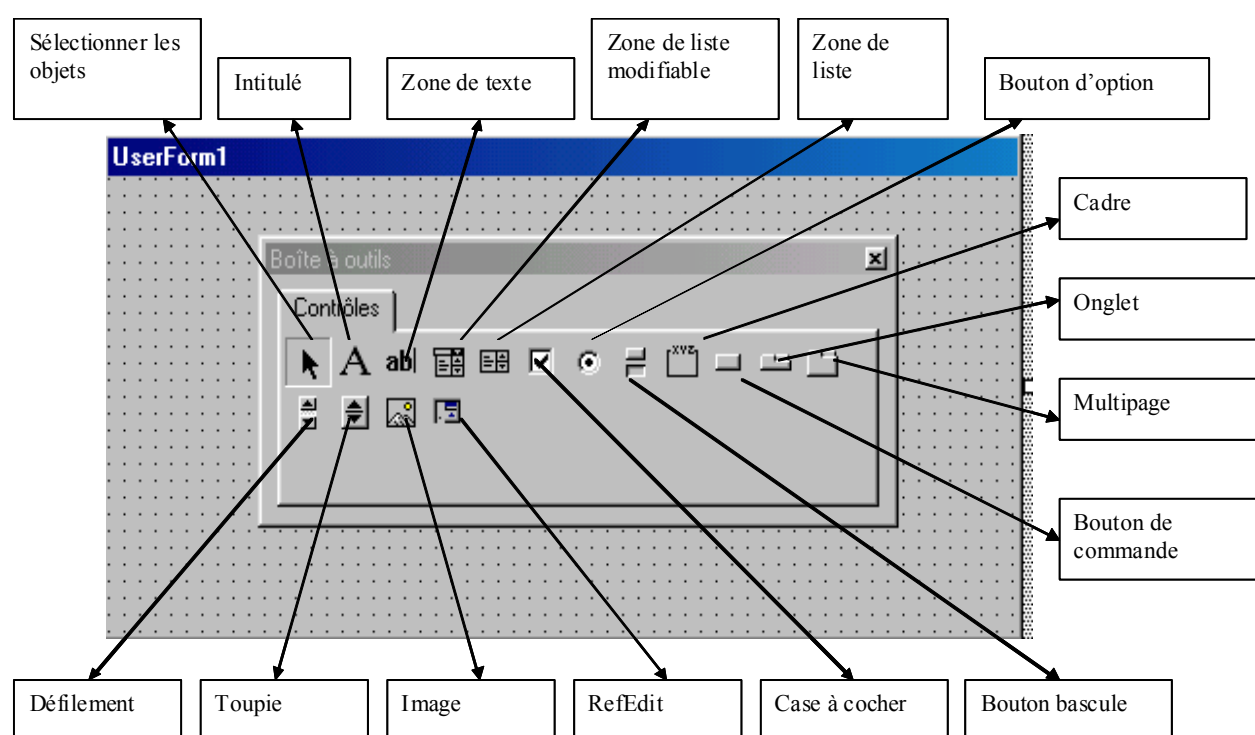
Ouvrir Excel,

Aller sur VBA

Insérer une feuille de type « userform »

Afficher la boîte à outils suivante

Passer en revue l'ensemble des icônes :



- **Intitulé (Label) :** les intitulés peuvent être utilisés à n'importe quel endroit de la boîte de dialogue en guise d'invites, de commentaires, d'incitation, etc. Ils n'ont pas d'autre fonction dans le cadre de l'utilisation ultérieure de la boîte de dialogue, mais ils peuvent être modifiés par le programme. Vous pouvez aussi les utiliser pour afficher du texte qui ne peut pas être modifié.
- **Zone de texte (Textbox) :** pour la saisie des contenus (textes, nombres, références ou formules), utilisez des zones de modification. Elles peuvent avoir n'importe quelle taille et, par conséquent, également contenir plusieurs lignes de texte.
- **Zone de liste modifiable (Combobox) :** zone de liste combinée avec une zone de texte dans laquelle l'entrée de liste sélectionnée s'inscrit et peut être éditée.

- **Zone de liste (Listext):** c'est une liste munie d'une barre de défilement et dans laquelle vous pourrez par la suite sélectionner une entrée.
- **Case à cocher (CheckBox):** les cases à cocher, contrairement aux cases d'option, ne s'exclut pas mutuellement. Vous pouvez aussi les réunir dans des zones de groupe pour montrer quelles sont celles qui appartiennent à une même catégorie ou qui se rapportent à un même domaine.
- **Bouton d'option (OptionButton) :** Ils se rapportent à une alternative donnée, doivent toujours être réunis dans une zone de groupe. Il s'agit d'options qui s'excluent mutuellement ; une seule peut être sélectionnée à la fois.
- **Bouton bascule :** bouton permettant de basculer entre deux états.
- **Cadre (Frame):** si vous avez besoin de cases d'options, autrement dit de plusieurs boutons ronds dont un seul peut être sélectionné à la fois, vous devez les regrouper à l'intérieur d'un cadre. Vous pouvez également rassembler des cases à cocher et d'autres éléments dans une telle zone. Cela n'a cependant aucune répercussion sur leur fonction dans ce cas.
- **Bouton de commande (CommandButton) :** Avec cet outil, vous pouvez créer autant de boutons que vous voulez. Il lance, termine ou interrompt une action ou une série d'actions
- **Défilement (ScrollBar):** lorsqu'il est actionné dans la boîte de dialogue, ce contrôle retourne une valeur correspondant approximativement à la position du curseur dans la barre (par défaut, ces valeurs sont comprises entre 0 et 100). Il peut être combiné avec d'autres éléments qui réagissent à la valeur retournée.
- **Toupie (SpinButton):** cet élément fonctionne comme une barre de défilement à cette différence près que vous ne disposez ici que des deux flèches et pas de curseur.
- **Onglet :** contrôle permettant d'insérer des onglets dans un formulaire.
- **Multipage :** permet d'insérer des onglets dans un formulaire, ces onglets pouvant à leur tour contenir des contrôles.
- **Image :** permet d'insérer une image ou un dessin dans le formulaire
- **RefEdit :** cet outil permet de sélectionner des références de plages de cellules dans un formulaire.

Ex : Vous pouvez créer un contrôle '**CommandButton**' qui ouvre une autre feuille.

Ex : Vous pouvez aussi afficher un texte, une image ou les deux sur un contrôle '**CommandButton**'.

Les contrôles sont d'abord désignés par des noms et des numéros par défaut

La propriété par défaut d'un contrôle CommandButton est la propriété Value.

L'événement par défaut d'un contrôle CommandButton est l'événement Click.

Exemple des propriétés des *contrôles ActiveX* :

Name : Définir le nom du contrôle.

Min : Spécifie la valeur (*Value*) minimale acceptable pour la propriété d'un contrôle.

Max : : Spécifie la valeur (*Value*) maximale acceptable pour la propriété d'un contrôle.

LinkedCell : Spécifie à quelle **cellule** de la feuille le contrôle est-il **lié**

SmallChange : Spécifie le déplacement (valeur du pas) se produisant lorsque l'utilisateur clique sur une flèche de défilement dans un contrôle SpinButton.

LargeChange : Spécifie la distance de déplacement (valeur du pas) intervenant lorsque l'utilisateur clique entre le curseur de défilement et la flèche de défilement.

Style : Pour un contrôle ComboBox, spécifie de quelle façon l'utilisateur va utiliser le contrôle.

ListFillRange : Indiquer le nom de la base où se trouve les informations de la zone de liste modifiable.

ColumnCount : Cette propriété indique à la zone de liste modifiable sur le nombre de colonnes de la base d'infos.

BoundColumn : Cette propriété indique à la zone de liste quelle valeur de colonne mettre dans la cellule lié.

Caption : Texte descriptif apparaissant sur un objet pour l'identifier ou le décrire.

NB : A ne pas Confondre avec Name : Nom par défaut donné aux objets par VB à leur création.

Font : Définit les caractéristiques (Gras, Italique, Police, Taille) du texte utilisé

SpécialEffect : Spécifie l'aspect à l'écran d'un objet.

RowSource : Permet de relié une zone de liste modifiable à une plage de cellules d'une feuille.

Syntaxe : Nom de feuille!cellules à lier

ControlSource : Permet de relier une liste à la cellule correspondante de la feuille cible.

Syntaxe : Nom de feuille!cellule à lier


Partie II : Du développement assisté

Exercice : Développer votre userform.

FEUILLES UTILISATEUR

Les feuilles utilisateur sont des fenêtres ou des boîtes de dialogue qui permettent de saisir des données. Elles sont plus puissantes que les fonctions ou les instructions **InputBox** ou **MsgBox**. Elles intègrent toute une panoplie de contrôles comme les zones de texte, les cases d'option ou les cases à cocher.

Notre exemple gèrera la liste des employés de l'auberge « À la bonne Franquette ».

- Cliquez **Nouveau** .

Base de données

En premier lieu, définissons la base de données des employés.

- Tapez le nom des colonnes comme dans la capture d'écran ci-après.

	A	B	C	D	E	F	G	H	I	J
1	Titre	Nom	Prénom	Téléphone	Extra	Heures	Salle	Emporté	Ecailler	Traiteur
2										

- Double-cliquez l'onglet **Feuil1** puis tapez **Base** .

La colonne **Extra** indique si l'employé accepte de faire un « extra » le samedi ou le dimanche. Nous utiliserons des cases d'option.

La colonne **Heures** indique le nombre d'heures par semaine. Nous utiliserons une barre de défilement.

Les colonnes **Salle**, **Emporté**, **Ecailler** et **Traiteur** indiquent le ou les postes de l'employé. Nous utiliserons des cases à cocher.

La colonne **Titre** contiendra **Madame**, **Mademoiselle** ou **Monsieur**. Nous utiliserons une zone de liste modifiable. Les valeurs de la zone seront contenues dans une feuille de calcul.

- Cliquez le menu **Insertion Feuille**.

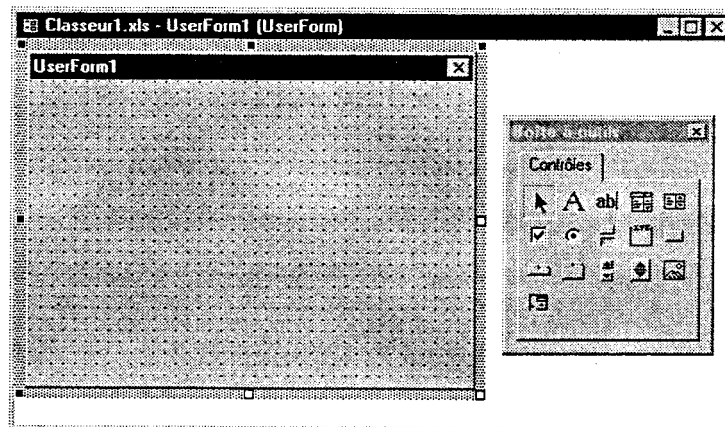
Créer une feuille utilisateur

- Double-cliquez l'onglet **Feuil2** puis tapez **Titres**.
- Tapez **Madame** **Mademoiselle** **Monsieur**.

	A	B
1	Madame	
2	Mademoiselle	
3	Monsieur	
4		

Créer une feuille utilisateur

- Cliquez **Visual Basic Editor** dans la barre d'outils **Visual Basic**.
- Cliquez **Ajouter UserForm** dans la barre d'outils.



Excel ajoute une feuille **UserForm1** dans une nouvelle fenêtre. Une **Boîte à outils** propose tous les objets nécessaires à la réalisation de la feuille.

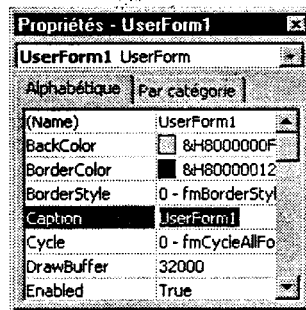
PROPRIÉTÉS DES OBJETS

Chaque objet de la feuille possède des propriétés modifiables à la création.

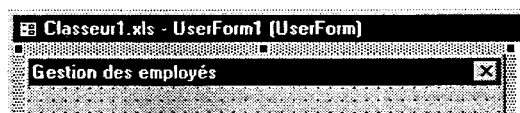
Modifions la barre de titre de la feuille.

- Cliquez **Fenêtre Propriétés** dans la barre d'outils.

- Double-cliquez la propriété **Caption**. Double-cliquez le mot **Caption** et non le contenu **UserForm1**.




- Tapez **Gestion des employés** dans la propriété **Caption**.

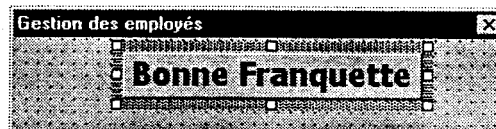


- Utilisez les poignées de redimensionnement pour agrandir la feuille (environ une fois et demie plus large et une fois et demie plus haute).

Intitulés

Les intitulés sont de simples textes que l'utilisateur ne peut pas modifier.

- Cliquez **Intitulé A** dans la **Boîte à outils**.
- Cliquez et faites glisser en haut de la feuille pour ajouter l'intitulé.
- Tapez **Bonne Franquette** dans la propriété **Caption**.
- Cliquez la propriété **Font** puis cliquez .
- Sélectionnez **Gras** pour **Style** et **16** pour **Taille** puis cliquez **OK**.
- Sélectionnez **1 - FmSpecialEffectRaised** dans la propriété **SpecialEffect**.
- Sélectionnez **2 - FmTextAlignCenter** dans la propriété **TextAlign**.
- Au besoin, ajustez la taille de l'intitulé avec les poignées.




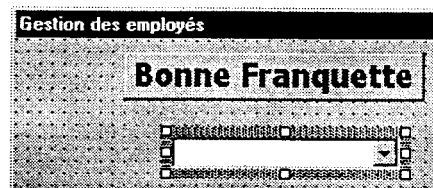
Zones de liste modifiables

En choisissant ce type de zone pour les données **Titre**, il sera possible, soit de sélectionner une des valeurs prédéfinies (feuille **Titres**), soit d'en taper une nouvelle (Maître, Société, etc.).



La **Boîte à outils** n'est plus visible quand on clique la fenêtre des **Propriétés**. Pour qu'elle apparaisse de nouveau, cliquez le fond de la feuille **Gestion des employés**.

- Cliquez **Zone de liste modifiable**  dans la **Boîte à outils**.
- Cliquez et faites glisser pour ajouter la zone de liste modifiable.

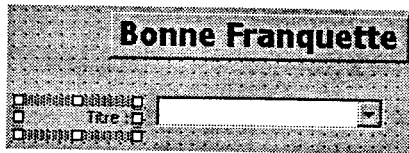


Relions cette liste à la plage **A1:A3** de la feuille **Titres**.

- Tapez **Titres!A1:A3** dans la propriété **RowSource**.


La base de données se trouve dans la feuille **Base**. Les nouveaux employés seront toujours insérés dans la ligne **2**. Relions la liste à la cellule correspondante.

- Tapez **Base!A2** dans la propriété **ControlSource**.
- Ajoutez un **Intitulé A** à gauche de la liste déroulante.
- Tapez **Titre** : dans la propriété **Caption**.
- Sélectionnez **3 - FmTextAlignRight** dans la propriété **TextAlign**.




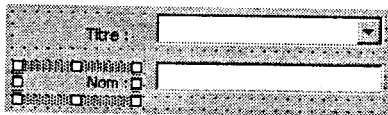
Zones de texte

Ajoutons une zone de texte pour saisir le nom du client.

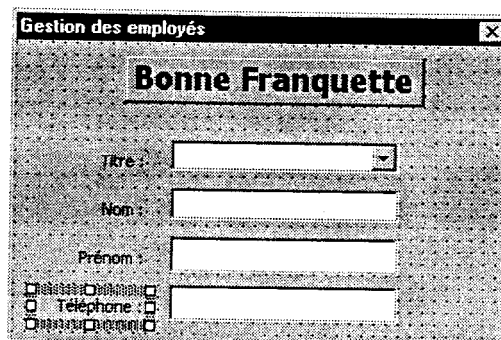
- Cliquez **Zone de texte**  dans la **Boîte à outils**.
- Cliquez et faites glisser en dessous de la liste déroulante.

Relions la zone de texte à la cellule correspondante.

- Tapez `Base!B2` dans la propriété **ControlSource**.
- Ajoutez un **Intitulé**  à gauche de la liste déroulante.
- Tapez `Nom :` dans la propriété **Caption**.
- Sélectionnez **3 - FmTextAlignRight** dans la propriété **TextAlign**.




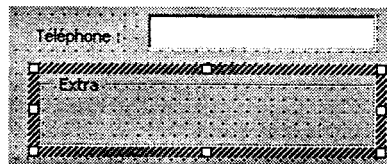
- De la même manière, ajoutez des zones de texte et des intitulés pour le **Prénom** et le **Téléphone** reliés aux cellules **C2** et **D2**.





Cases d'option

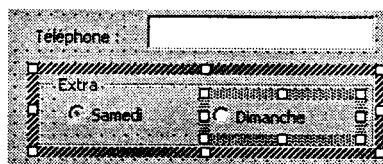
Ajoutons deux cases d'option dans un cadre pour le jour des « extra ».

- Cliquez **Cadre**  dans la **Boîte à outils**.
- Cliquez et faites glisser en dessous de l'étiquette **Téléphone**.
- Tapez **Extra** dans la propriété **Caption**.



- Cliquez **Bouton d'option**  dans la **Boîte à outils** et faites glisser à l'intérieur du cadre **Extra**.
- Tapez **Samedi** dans la propriété **Caption**.
- Tapez **Base!E2** dans la propriété **ControlSource**.
- De la même manière, ajoutez un **Bouton d'option**  nommé **Dimanche**.

Comme il n'y a que deux boutons, ce dernier n'est pas relié à une cellule puisqu'il change la valeur de l'autre bouton quand on clique dessus.

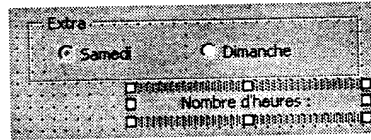



Compteurs et barres de défilement

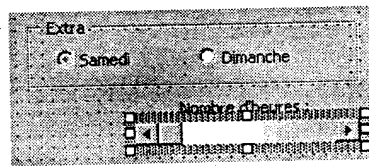
Les compteurs et les barres de défilement permettent de faire varier des valeurs numériques en utilisant la souris. Pour connaître leur valeur courante, il faut les associer à un intitulé ou à une zone de texte. La mise à jour se fait avec VB.

Cases à cocher

- Ajoutez un intitulé Nombre d'heures : en dessous du cadre Type de téléphone.
- Sélectionnez 2 - FmTextAlignCenter dans la propriété TextAlign.



- Cliquez Défilement  et faites glisser en dessous de l'intitulé.




La valeur de la barre variera entre les valeurs des propriétés Min et Max.

- Tapez 50 dans la propriété Max.

La valeur de la barre augmente ou diminue de la valeur définie dans **SmallChange** quand on clique sur les flèches de droite ou de gauche. Elle augmente ou diminue de la valeur définie dans **LargeChange** quand on clique à l'intérieur de la barre.

- Tapez 8 dans la propriété LargeChange.
- Tapez Base!F2 dans ControlSource.

Cases à cocher

- Cliquez Case à cocher  dans la Boîte à outils et faites glisser à droite de la zone de liste Titre.
- Tapez Salle dans la propriété Caption.
- Tapez Base!G2 dans ControlSource.
- De la même manière, ajoutez 3 cases à cocher nommées Emporté, Ecailler et Traiteur liées aux cellules H2, I2 et J2.

Titre : ☒ Salle


Nom : ☒ Emporté

Prénom : ☒ Ecailler

Téléphone : ☒ Traiteur

Bouton de commande

Pour valider ou annuler les données saisies et fermer la feuille, ajoutons deux boutons de commande.

- Cliquez **Bouton de commande**  dans la **Boîte à outils** puis faites glisser à droite du cadre **Extra**.
- Tapez **OK** dans la propriété **Caption**.
- De la même manière, ajoutez en dessous un bouton **Annuler**.

Gestion des employés

Bonne Franquette

Titre : ☒ Salle

Nom : ☒ Emporté

Prénom : ☒ Ecailler

Téléphone : ☒ Traiteur


Extra

☒ Samedi ☐ Dimanche

Nombre d'heures :

OK

Annuler

- Au besoin, déplacez et redimensionnez les contrôles pour obtenir le même agencement que dans l'exemple ci-dessus.
- Cliquez **Enregistrer**  et nommez le classeur **Employés**.

Nom des contrôles

À leur création, VB donne un nom par défaut aux objets (type d'objet suivi d'un numéro chronologique).

- Cliquez la case à cocher **Salle**.

La propriété **Name** donne le nom attribué. VB ne réattribue pas un numéro supprimé sauf si c'est le dernier. Si l'on supprime la **CheckBox5** alors qu'il existe une **CheckBox6**, la prochaine case à cocher portera le nom **CheckBox7**.

Si vous avez supprimé un contrôle pendant la constitution de la feuille, il est possible qu'un objet porte un autre nom. Comme on utilise ces noms pour accéder aux objets avec VB, il faut les nommer explicitement.

- Cliquez la feuille **UserForm1** et tapez **Saisie** dans la propriété **Name**.
- Cliquez l'intitulé **Nombre d'heures** et tapez **Compteur** dans la propriété **Name**.
- Cliquez la barre de défilement et tapez **Heures** dans la propriété **Name**.
- Cliquez l'option **Samedi** et tapez **Samedi** dans la propriété **Name**.
- Donnez à la liste et aux trois zones de texte les noms **Titre**, **Nom**, **Prénom** et **Téléphone**.
- Donnez aux quatre cases à cocher les noms **Salle**, **Emporté**, **Ecailler** et **Traiteur**.

Procédures événementielles

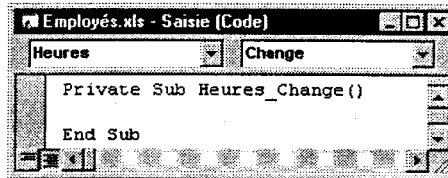
Pour chaque contrôle il existe des procédures événementielles. Elles sont exécutées quand l'utilisateur intervient sur l'objet (en cliquant un bouton, en tapant dans une zone de texte, en cliquant une barre de défilement, etc.).

Pour que notre intitulé **Compteur** varie en fonction de la barre de défilement **Heures**, il faut ajouter une procédure événementielle.

- Double-cliquez la barre de défilement **Heures**.

Une nouvelle feuille de module affiche la structure de la procédure. Le nom de la procédure est constitué du nom de l'objet et du type d'événement.


Exécuter une feuille utilisateur




L'événement **Change** est proposé par défaut car c'est le plus couramment utilisé pour une barre de défilement.

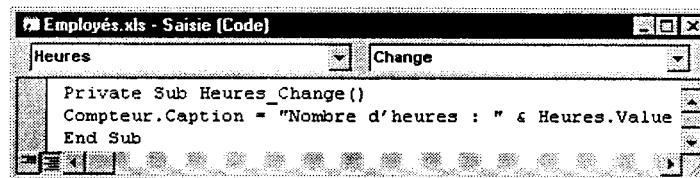
- Cliquez la flèche  de la liste **Procédure** dans la fenêtre **Saisie (Code)**.

On trouve ici tous les événements possibles pour ce type d'objet.

- Tapez  pour fermer la liste **Procédure**.

Pour mettre à jour l'intitulé **Compteur** (propriété **Caption**), il suffit d'y placer la valeur de la barre de défilement **Relance** (propriété **Value**).

- Tapez `Compteur.Caption = "Nombre d'heures : " & Heures.Value`
.

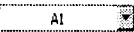



Exécuter une feuille utilisateur

- Cliquez **Affichage Microsoft Excel**  dans la barre d'outils.


- Cliquez l'onglet **Base**.

Comme pour les objets de la feuille utilisateur, il est plus simple de donner un nom aux objets d'Excel (les plages de cellules par exemple). Donnons un nom à la base de données.


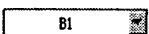

- Sélectionnez la plage **A1:J2**.
- Cliquez **Zone Nom**  (à gauche de la barre de formule) et tapez `ListeEmployés`  pour donner un nom à la liste.

Exécuter une feuille utilisateur





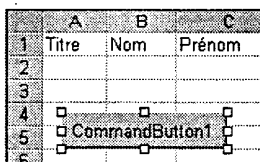
Il est indispensable d'appuyer sur la touche  pour que le nom soit pris en compte.

Les données de la liste seront triées sur les noms. Nommons la colonne des noms pour y accéder plus facilement.

- Sélectionnez la colonne **B** en cliquant .
- Cliquez **Zone Nom**  et tapez **Nom** .



La feuille sera ouverte par un bouton de commande.

- Cliquez **Boîte à outils Contrôles**  dans la barre d'outils **Visual Basic**.
- Cliquez **Bouton de commande**  dans la barre d'outils **Commandes**.
- Cliquez et faites glisser dans la feuille **Base** pour ajouter le bouton.



	A	B	C
1	Titre	Nom	Prénom
2			
3			
4			
5			
6			


Les objets proposés dans la barre d'outils **Boîte à outils Contrôles** sont les mêmes que dans la boîte équivalente de VB.


- Cliquez **Propriétés**  dans la barre d'outils **Boîte à outils Contrôles**.
- Tapez **OuvrirSaisie** dans la propriété **Name**.
- Tapez **Ajouter un employé** dans la propriété **Caption**.
- Cliquez  pour fermer la fenêtre **Propriétés**.

Nous allons créer la procédure qui ouvre la feuille **Saisie**.

- Double-cliquez le bouton **Ajouter un employé**.





Pour accéder aux procédures des objets, le bouton **Mode création**  doit être enfoncé.

- Tapez ' Ouverture de la boîte de saisie .

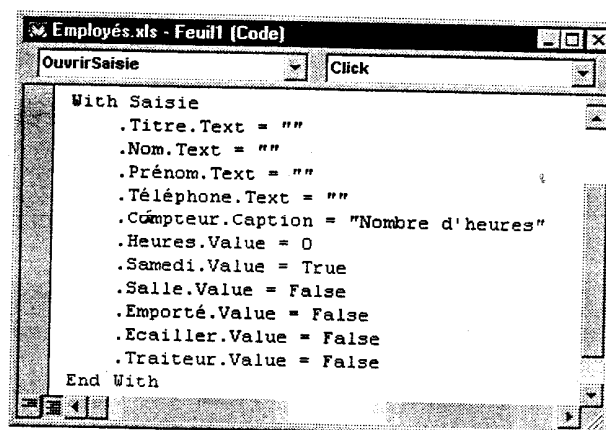
Les valeurs seront toujours insérées dans la première ligne de la liste, après les en-têtes de colonne.

Exécuter une feuille utilisateur

- Tapez `Worksheets("Base").Rows(2).Select` .
- Tapez `Selection.Insert` .

Chaque objet d'une feuille utilisateur conserve la dernière valeur saisie dans sa propriété **Value**, **Text** ou **Caption**. Quand on ouvre une feuille, les anciennes valeurs apparaissent. Il faut donc remettre à zéro tous les contrôles pour obtenir une feuille vierge.



- Tapez les lignes suivantes.




N'oubliez pas le point avant chaque contrôle.

Initialisons aussi les valeurs par défaut des cases d'option et des cases à cocher dans la feuille **Base**.

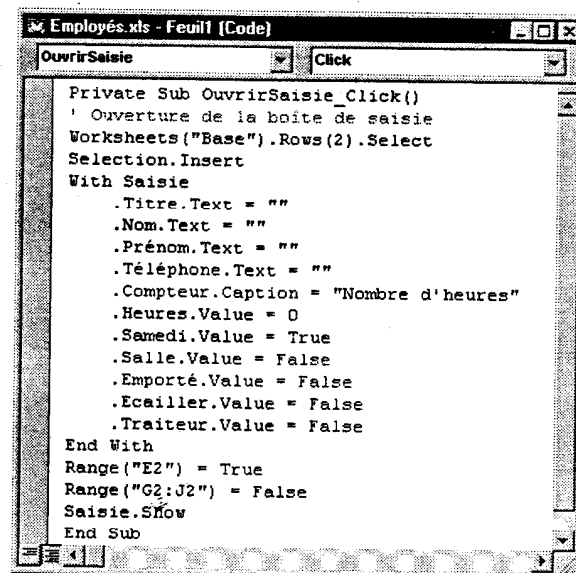
Choisissons l'option **Samedi** pour le jour d'extra par défaut.

- Tapez `Range("E2") = True` .
- Tapez `Range("G2:J2") = False` .

Pour afficher une feuille, il faut utiliser la méthode **Show**.

- Tapez `Saisie.Show` .

Fermeture de la feuille utilisateur






Fermeture de la feuille utilisateur

Avant même de tester la feuille, il faut penser à sa fermeture. Nous disposons de deux boutons qui ferment la boîte.

BOUTON OK



Si l'utilisateur clique le bouton **OK**, les données doivent être triées dans l'ordre croissant des noms.

De plus, les cases d'option et les cases à cocher étant converties en valeur **True** (case cochée) ou en valeur **False** (case non cochée), nous devons les remplacer par des valeurs plus explicites (**Pro**, **Perso**, **Oui** et **Non**).

- Double-cliquez **Saisie**   dans la fenêtre **Projet**.
- Double-cliquez le bouton **OK** pour ouvrir la procédure événementielle **Click**.
Fermons la feuille utilisateur **Saisie**.
- Tapez `Saisie.Hide` .

Fermeture de la feuille utilisateur






Pour éviter une interaction entre les données de la feuille utilisateur et celles de la feuille de calcul, il faut modifier les données sur une autre ligne. Insérons une nouvelle ligne pour modifier les données en ligne 3.

- Tapez `Rows(2).Select` .
- Tapez `Selection.Insert` .








Remplaçons le jour d'extra par **Samedi** ou **Dimanche** à la place de **VRAI** ou **FAUX**.



Si les cases d'option et les cases à cocher contiennent les valeurs **True** ou **False**, ce sont en revanche des valeurs **VRAI** ou **FAUX** qui apparaissent dans la feuille de calcul.


- Tapez `If Range("E3") = True Then` .
- Pour la valeur **True**, c'est la case **Samedi** qui est cochée.
- Tapez `Range("E3") = "Samedi"` .
- Pour la valeur **False**, c'est la case **Professionnel** qui est cochée.
- Tapez `Else` .
- Tapez `Range("E3") = "Dimanche"` .
- Tapez `End If` .

Remplaçons les valeurs **Vrai** ou **Faux** des quatre cases à cocher par des valeurs **Oui** ou des cellules vides. Pour simplifier le code, nous utilisons une boucle **For...Next**. On accède ici aux cellules avec la méthode **Cells** (voir page 63). Les valeurs se trouvent dans les colonnes 7 à 10.


- Tapez `For i = 7 To 10` .
- Tapez `If Cells(3, i) = True Then` .
- Tapez `Cells(3, i) = "Oui"` .
- Tapez `Else` .
- Tapez `Cells(3, i) = ""` .
- Tapez `End If` .
- Tapez `Next i` .

Supprimons maintenant la ligne insérée au début de la procédure.

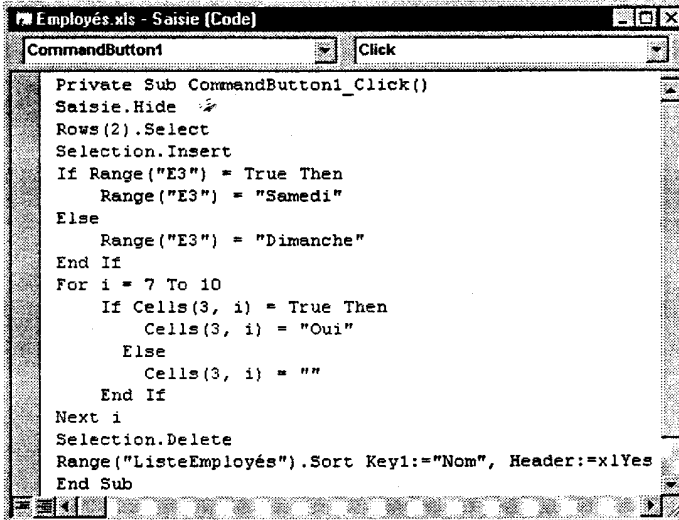
Fermeture de la feuille utilisateur

- Tapez `Selection.Delete` .

Comme des données ont été insérées, il faut maintenant trier la liste **ListeEmployés**.

- Tapez `Range("ListeEmployés").Sort Key1:="Nom", Header:=xlYes` .



- ⇒ **Key1** : correspond à la colonne sur laquelle s'effectue le tri (colonne **B** que nous avons appelée **Nom**).
- ⇒ **Header** : indique que la liste contient des en-têtes de colonne qui ne doivent pas être triées avec les données.



```
Private Sub CommandButton1_Click()  
    Saisie.Hide  
    Rows(2).Select  
    Selection.Insert  
    If Range("E3") = True Then  
        Range("E3") = "Samedi"  
    Else  
        Range("E3") = "Dimanche"  
    End If  
    For i = 7 To 10  
        If Cells(3, i) = True Then  
            Cells(3, i) = "Oui"  
        Else  
            Cells(3, i) = ""  
        End If  
    Next i  
    Selection.Delete  
    Range("ListeEmployés").Sort Key1:="Nom", Header:=xlYes  
End Sub
```




BOUTON ANNULER

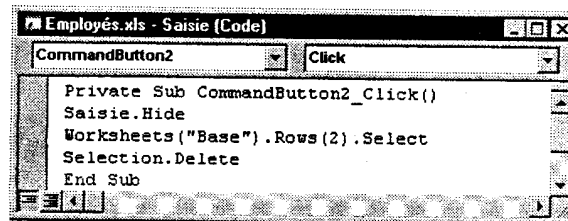
Si l'utilisateur clique le bouton **Annuler**, la ligne ajoutée dans la liste doit être supprimée.


- Double-cliquez **Saisie**   dans la fenêtre de **Projet**.
- Double-cliquez le bouton **Annuler** pour ouvrir la procédure événementielle **Click**.

Fermons la feuille utilisateur **Saisie**.


Ordre de tabulation

- Tapez `Saisie.Hide` .
- Supprimons la liste précédemment insérée.
- Tapez `Worksheets("Base").Rows(2).Select` .
- Tapez `Selection.Delete` .

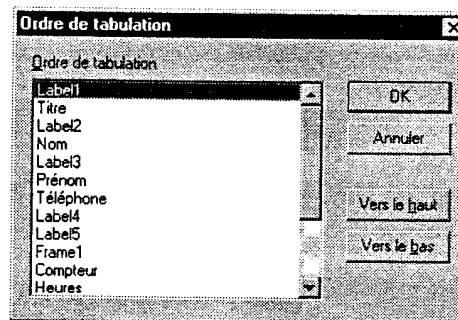


- Cliquez  pour fermer la fenêtre **Saisie (Code)**.


Ordre de tabulation

Il existe un ordre de saisie des données dans une feuille quand l'utilisateur appuie sur la touche  pour passer d'un contrôle à un autre contrôle.

- Cliquez le menu **Affichage** **O**rdre de **t**abulation.



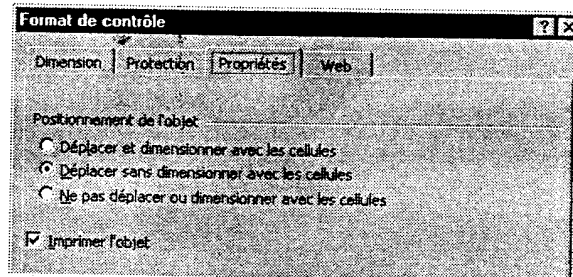
- ⇒ **Vers le haut, Vers le bas** : sélectionnez l'objet à déplacer dans la liste **O**rdre de **t**abulation puis cliquez sur un de ces boutons pour changer l'ordre de tabulation.
- Cliquez **OK** dans la boîte **Ordre de tabulation**.

- Cliquez  pour fermer la fenêtre **Microsoft Visual Basic**.

Propriétés des contrôles

Les contrôles possèdent des propriétés modifiables directement.

- Cliquez avec le bouton droit le bouton de commande **Ajouter un employé**.
- Cliquez **Format de contrôle** dans le menu contextuel.
- Cliquez l'onglet **Propriétés** dans la boîte **Format de contrôle**.





Quand on insère une nouvelle ligne dans la liste des employés, le bouton **Ajouter un employé** est déplacé avec le reste de la feuille.

- Cliquez **Ne pas déplacer ou dimensionner avec les cellules** pour que le bouton reste toujours à sa place.
- Cliquez **OK** pour fermer la boîte **Format de contrôle**.

Saisir des employés

Nous allons maintenant saisir des employés.

- Cliquez **Désactiver le mode création**  pour qu'il ne soit pas enfoncé.
- Cliquez **Ajouter un employé** pour ouvrir la feuille **Gestion des employés**.
- Tapez les données comme dans l'exemple ci-dessous. Utilisez la touche  pour passer d'un contrôle à l'autre. Sélectionnez **Madame** dans la liste **Titre**, ou tapez simplement **M**.

Saisir des employés

Gestion des employés

Bonne Franquette

Titre : ☒ Salle

Nom : ☐ Emporté

Prénom : ☐ Ecailler

Téléphone : ☐ Traiteur

Extra

☒ Samedi ☐ Dimanche

Nombre d'heures : 39

OK Annuler

- Cliquez sur **OK** pour mettre à jour les données dans la liste.

	A	B	C	D	E	F	G	H	I	J
1	Titre	Nom	Prénom	Téléphone	Extra	Heures	Salle	Emporté	Ecailler	Traiteur
2	Madame	Couvair	Armelle	01.34.61.13.13	Samedi	39	Oui			

- Cliquez le bouton **Ajouter un employé**.
- Ajoutez le second employé.

Gestion des employés

Bonne Franquette

Titre : ☐ Salle

Nom : ☐ Emporté

Prénom : ☒ Ecailler

Téléphone : ☐ Traiteur

Extra

☐ Samedi ☒ Dimanche

Nombre d'heures : 35

OK Annuler

- Cliquez sur **OK** pour fermer la boîte de dialogue.

Saisir des employés

- Cliquez sur le bouton de commande **Ajouter un employé**.
- Tapez les données de votre choix.
- Cliquez sur **Annuler** pour fermer la boîte de dialogue.

Excel n'a pas ajouté ces données dans la liste.

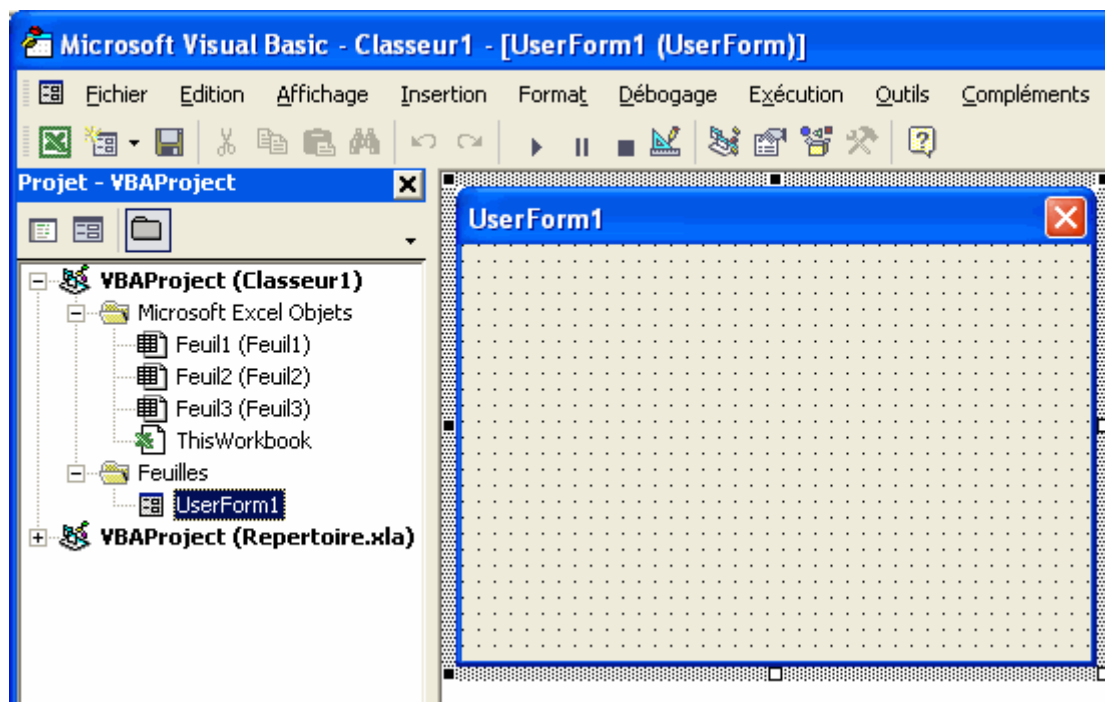
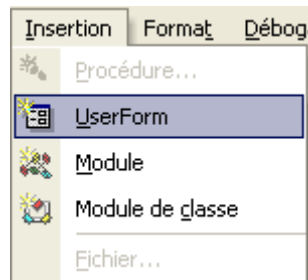
	A	B	C	D	E	F	G	H	I	J
1	Titre	Nom	Prénom	Téléphone	Extra	Heures	Salle	Emporté	Ecailler	Traiteur
2	Monsieur	Arroni	Marc	01.40.36.65.91	Dimanche	35			Oui	
3	Madame	Couvair	Amelle	01.34.61.13.13	Samedi	39	Oui			

- Enregistrez et fermez le classeur.

Partie 3 : Apprendre Plus.


Les UserForm sont des boîtes de dialogues personnalisées, offrant une interface intuitive entre l'application et l'utilisateur.

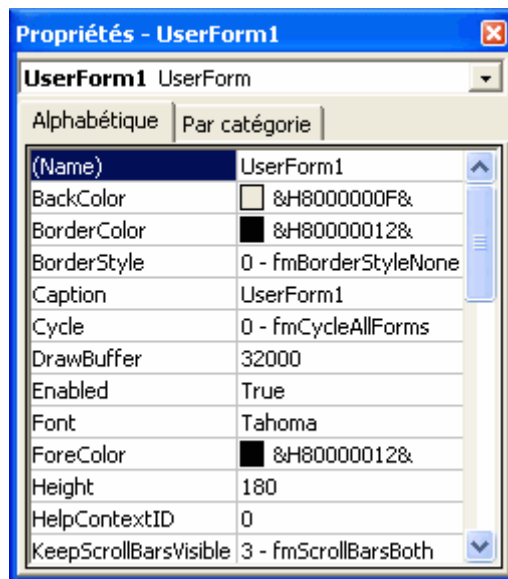
Sous VBE, les UserForm sont créés par le menu "Insertion-UserForm".



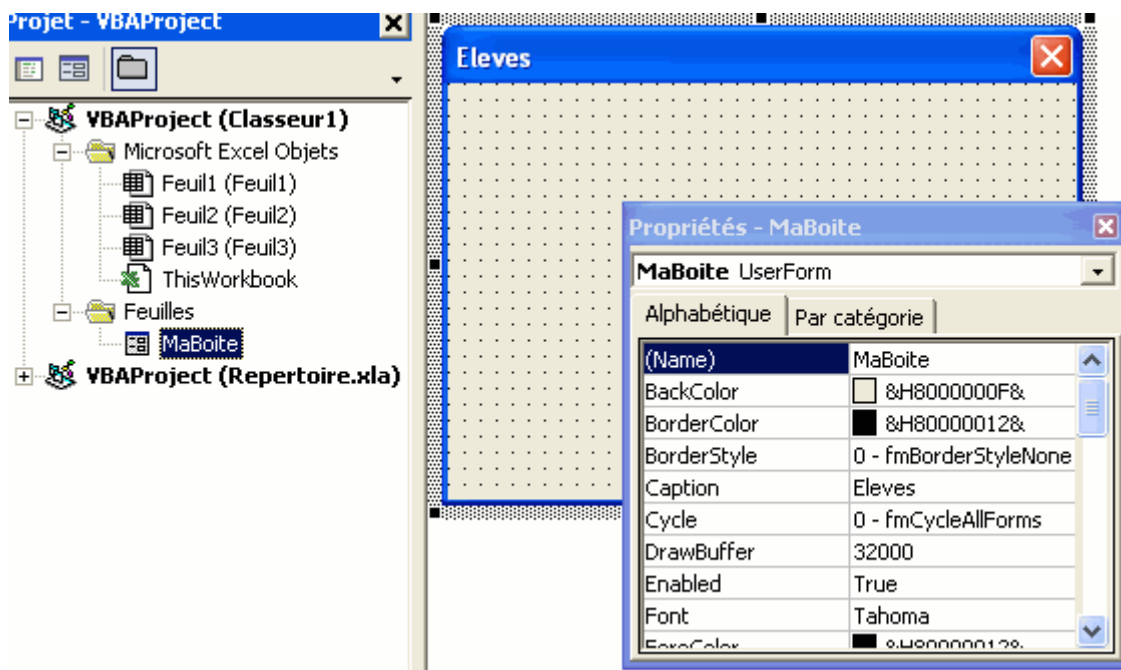
Par défaut, les UserForm sont nommés "UserForm1", "UserForm2" ...

Chaque UserForm possède ses propres propriétés tel que son nom, sa couleur, sa taille, sa position ...

Les propriétés d'un UserForm s'affichent en cliquant sur l'icône , par le menu "Affichage-Fenêtre Propriétés" ou par la touche "F4".




La propriété "Name" change le nom de l'UserForm, la propriété "Caption", son titre.



Les propriétés permettent de personnaliser les UserForm. Vous pouvez changer la couleur de fond par la propriété "BackColor", ajouter une bordure par la propriété "BorderStyle", définir sa couleur par la propriété "BorderColor", mettre une image de fond par la propriété "Picture"...

Le dimensionnement d'un UserForm peut se faire avec la souris ou en définissant sa taille par ses propriétés "Width" (Largeur) et "Height" (Hauteur).

Chaque UserForm va recevoir des contrôles. En cliquant sur le UserForm, une boîte à outils doit apparaître. Si ce n'est pas le cas, affichez la en cliquant sur l'icône  ou par le menu "Affichage-Boîte à outils".

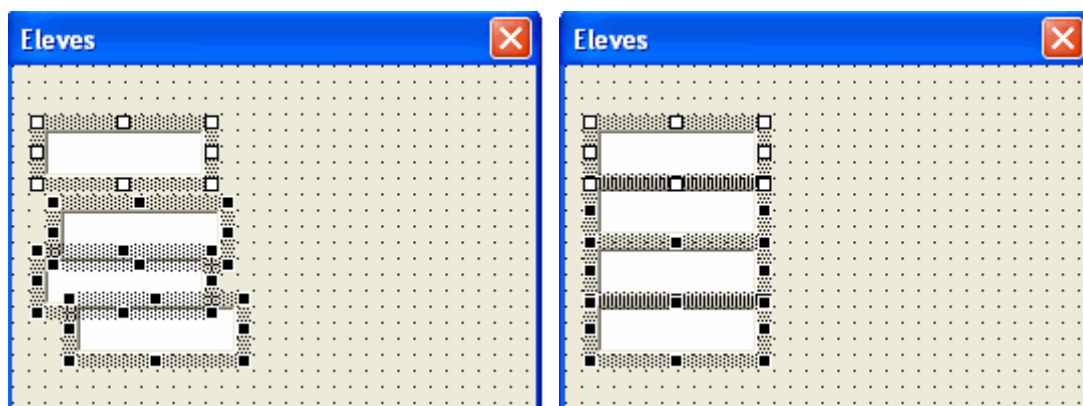


Pour ajouter un contrôle sur le UserForm, vous pouvez soit cliquer sur le contrôle désiré puis, sur le UserForm, tracer un rectangle qui définira sa taille ou simplement faire un cliquer-glisser du contrôle sur l'UserForm.

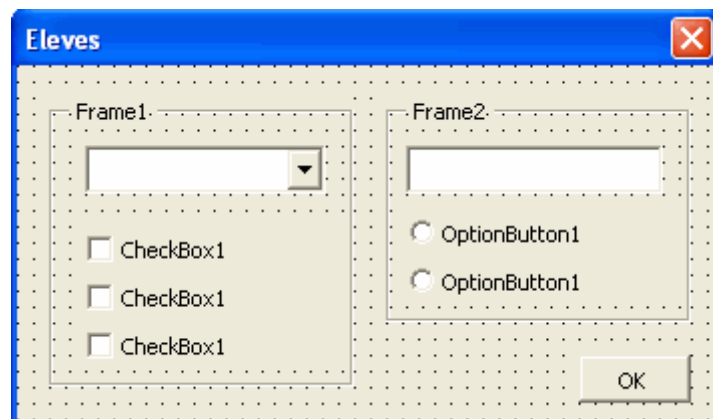
Les UserForm possèdent une grille matérialisée par des points. Elle permet l'alignement des contrôles. Vous pouvez la masquer, la désactiver ou définir sa taille par le menu "Outils-Options" dans l'onglet "Général".

Le menu "Format" de VBE permet d'aligner les contrôles.

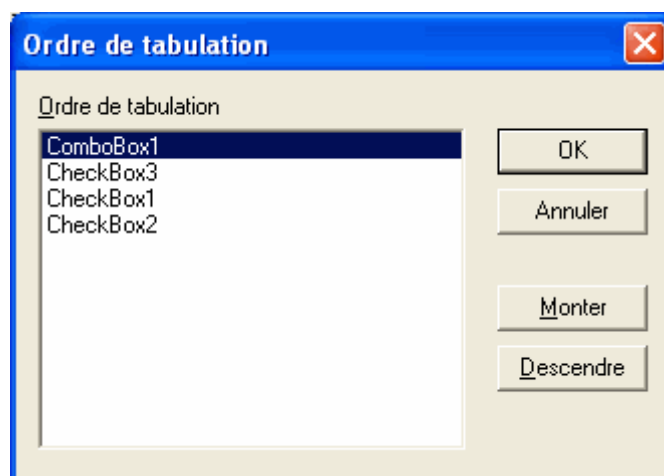
Par exemple le menu "Format-Aligner- Gauche" puis le menu "Espace Vertical-Egaliser" permet un alignement régulier des contrôles:



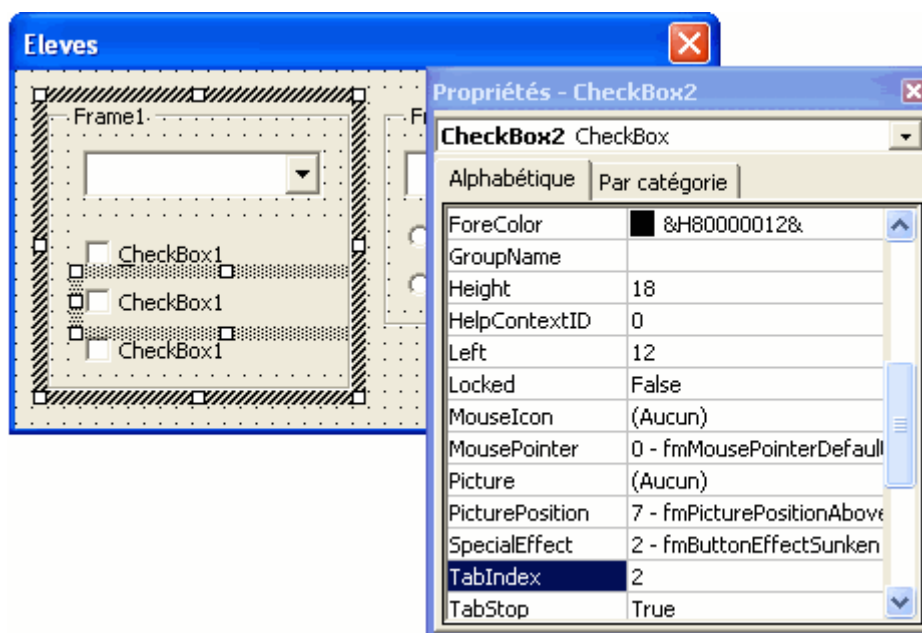
Le contrôle "Frame"  permet de grouper des contrôles.



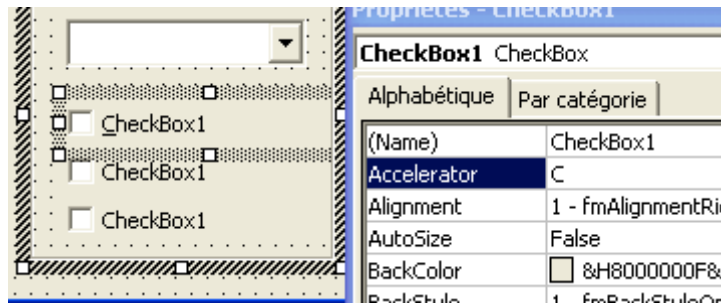
Le UserForm doit permettre à l'utilisateur de passer d'un contrôle à l'autre par la touche "Tabulation" de façon ordonnée. Le menu "Affichage-Ordre de tabulation" permet de paramétrer l'ordre de tabulation. Cliquez sur l'UserForm pour changer l'ordre des deux frames et du bouton "OK" et sélectionnez une frame pour changer l'ordre des contrôles qu'elle contient.



Vous pouvez également changer l'ordre de tabulation par la propriété "TabIndex" de chaque contrôle.

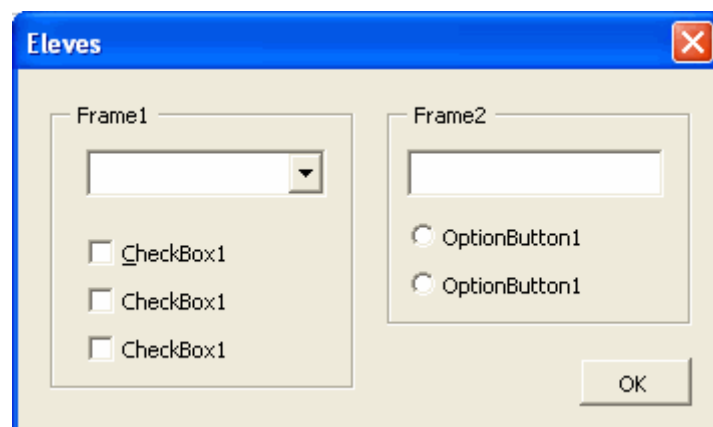


Vous pouvez affecter une touche de raccourci "Alt+caractère" à un contrôle par sa propriété "Accelerator". Utilisez un caractère du nom du contrôle, celui-ci sera souligné, indiquant à l'utilisateur quelle touche de raccourci utiliser :



L'affichage des UserForm s'effectue par la méthode "Show" de l'UserForm. Cette instruction doit être placée à l'intérieur d'une procédure dans un module.

```
Sub AfficheUF()  
    MaBoite.Show  
End Sub
```



Par défaut, un UserForm est modal, c'est à dire que l'utilisateur ne peut effectuer aucune action sur l'application tant qu'il n'est pas fermé. Depuis la version 2000 d'Excel, il est possible d'afficher des boîtes non modal, permettant l'utilisation des feuilles de calcul en gardant le UserForm affichée. La syntaxe est :

```
Sub AfficheUF()  
    MaBoite.Show 0  
End Sub
```

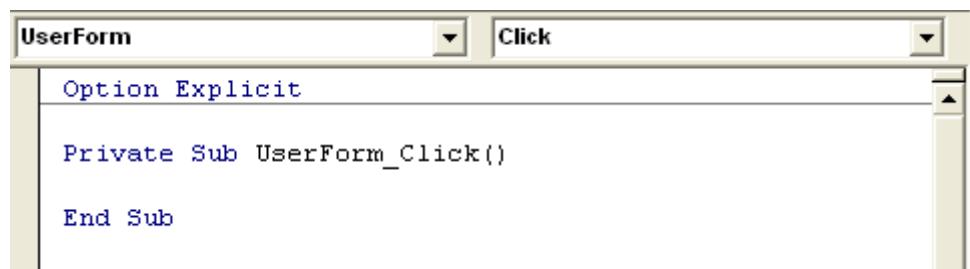
L'instruction Load charge le UserForm en mémoire sans l'afficher.

L'instruction Unload ferme le UserForm en le déchargeant de la mémoire. La syntaxe de cette instruction est : Unload UserForm.

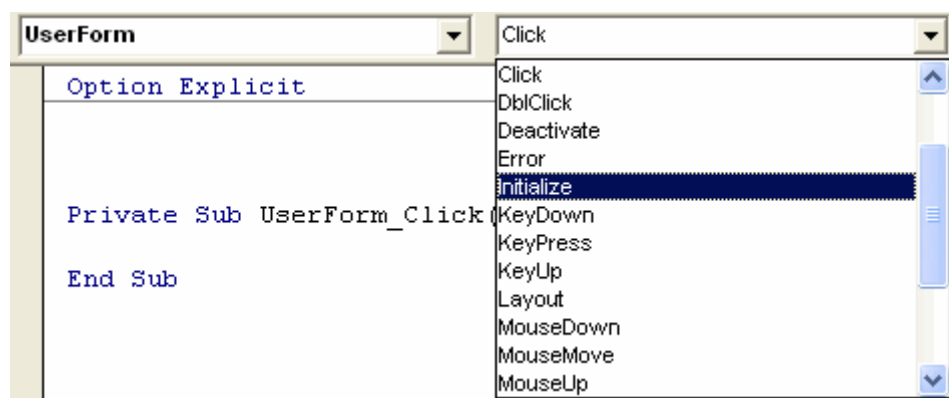
Il est également possible de fermer un UserForm en gardant en mémoire la valeur de ses contrôles par la méthode Hide. La syntaxe devient : UserForm.Hide.

Chaque UserForm possède son propre module.

Pour y accéder, cliquez sur le UserForm ou sur un contrôle puis tapez "F7" ou faites un double-clic sur l'objet. Par défaut, le module s'affichera avec une procédure événementielle de type privée de l'objet sélectionné.

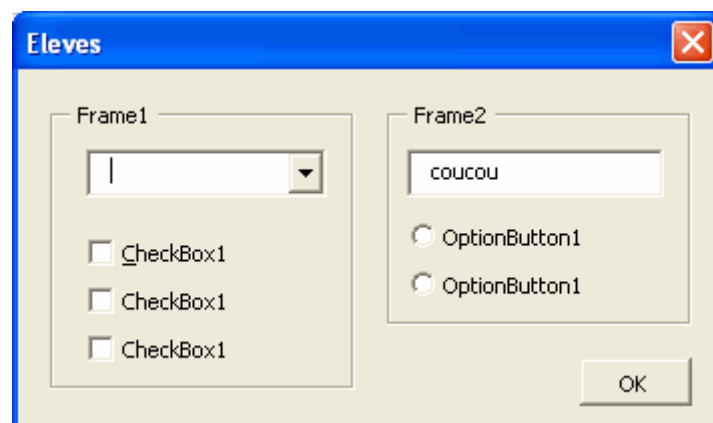


Les deux listes déroulantes en haut du module permettent de sélectionner l'objet et son évènement.



Dans cet exemple, la procédure Initialize de l'objet UserForm va être créée et ses instructions vont être exécutées au chargement de la boîte.

```
Private Sub UserForm_Initialize()
    TextBox1 = "coucou"
End Sub
```



Si l'évènement Initialize se produit au chargement d'un UserForm, l'évènement QueryClose se produit à sa fermeture. Dans l'exemple suivant, un message contenant le texte de l'objet TextBox1 s'affichera à la fermeture de la boîte.

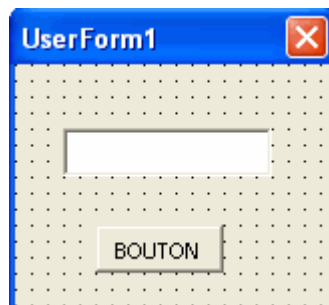
```
Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    MsgBox TextBox1
End Sub
```

L'élément Cancel de l'évènement QueryClose invalide la fermeture de la boîte si sa valeur est 1 et l'élément CloseMode définit la manière dont la boîte cherche à être fermée. Si l'utilisateur cherche à la fermer en cliquant sur la croix, CloseMode prend comme valeur 0, sinon CloseMode prend comme valeur 1. L'exemple suivant montre comment obliger l'utilisateur à fermer la boîte en cliquant sur le bouton "OK".

```
Private Sub CommandButton1_Click()
    Unload MaBoite
End Sub

PrivateSub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    If CloseMode = 0 Then 'Si on clique sur la croix
        MsgBox "Fermez la boîte avec le bouton OK"
        Cancel = 1 'Invalide la fermeture
    End If
End Sub
```

Les contrôles possèdent la propriété Visible qui permet de les rendre visible ou invisible. Dans l'exemple suivant, le click sur BOUTON va masquer ou afficher la zone de texte.



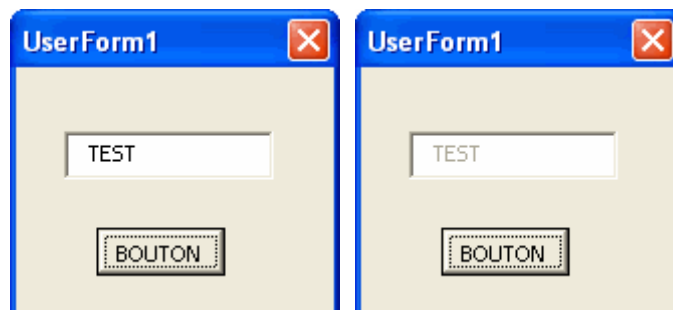
```
Private Sub CommandButton1_Click()
    If TextBox1.Visible = True Then
        TextBox1.Visible = False
    Else
        TextBox1.Visible = True
    End If
End Sub
```

Les contrôles possèdent la propriété Enabled qui peut interdire son accès à l'utilisateur. Lorsqu'un contrôle n'est pas accessible, son aspect change. Dans l'exemple suivant, le click sur BOUTON va interdire ou rendre accessible l'accès à la zone de texte.

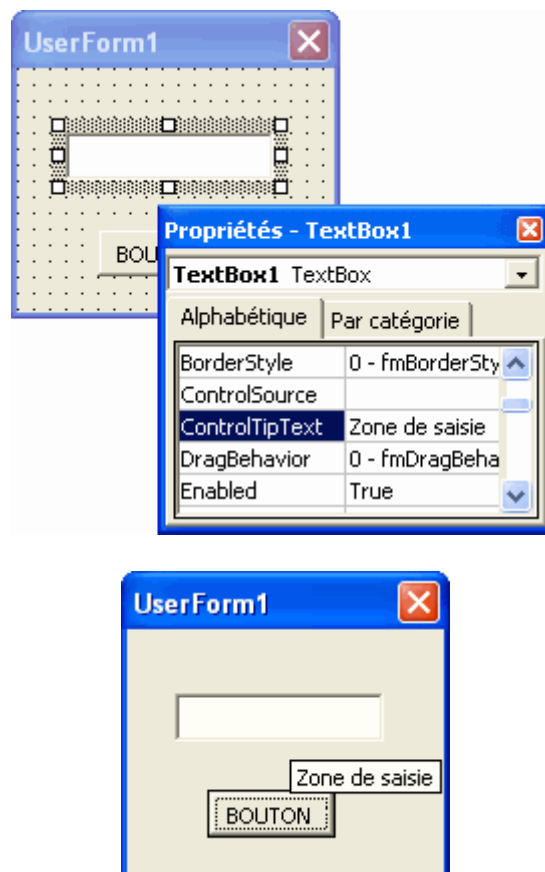
```

Private Sub CommandButton1_Click()
    If TextBox1.Enabled = True Then
        TextBox1.Enabled = False
    Else
        TextBox1.Enabled = True
    End If
End Sub

```

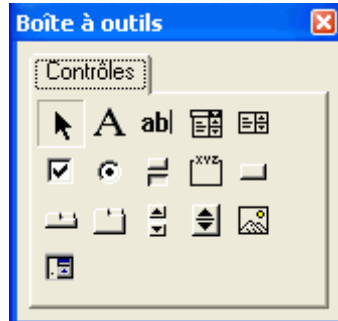


Les contrôles possèdent la propriété ControlTipText qui affiche une étiquette lors du survol de la souris.



UserForms – Les contrôles.

La boîte à outils affiche les contrôles standard de VBA.



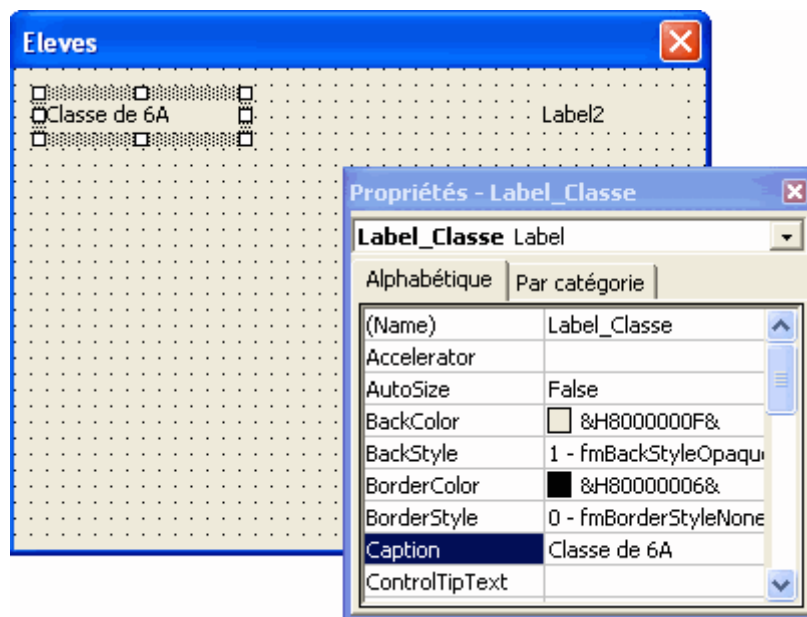
Sélection

Cet outil permet de sélectionner, de déplacer et de redimensionner les contrôles créés sur l'UserForm.



Label ou étiquette

Cet outil permet de créer une zone de texte non modifiable par l'utilisateur.



Dans cet exemple, 2 étiquettes ont été créées. Par défaut leur nom était Label1 et Label2. Pour plus de confort, elles ont été renommées Label_Classe et Label_Date. La valeur de Label_Class étant fixe, elle a été saisie dans sa propriété Caption. La valeur de Label_Date étant variable, elle peut être définie dans l'évènement Initialize de l'UserForm (renommé MaBoite).


```
Private Sub UserForm_Initialize()  
    Label_Date.Caption = Date  
End Sub
```

ab| TextBox ou zone de texte

Cet outil permet de créer une zone de texte pouvant être saisie ou modifiée par l'utilisateur. Une zone de texte peut faire référence à une cellule par la propriété ControlSource.

	A	
1	Excel	
2		
3		
4		

Si l'utilisateur modifie la zone de texte, la valeur de la cellule A1 sera modifiée.

	A	
1	Word	
2		
3		
4		

La valeur de la cellule A1 peut également prendre la valeur de la zone de texte par une procédure événementielle TextBox_Change.

```
Private Sub TextBox1_Change()  
    Range("A1") = TextBox1  
End Sub
```

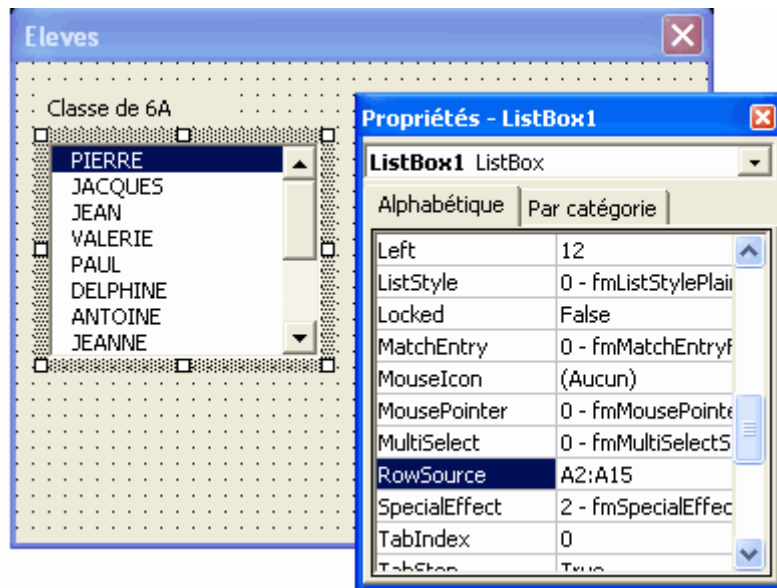


ListBox ou zone de liste

Une zone de liste permet d'afficher une liste d'éléments sélectionnables par l'utilisateur. Reprenons la liste d'élèves.

	A	B
1	ELEVE	NOTE
2	PIERRE	5
3	JACQUES	15
4	JEAN	10
5	VALERIE	12
6	PAUL	18
7	DELPHINE	13
8	ANTOINE	0
9	JEANNE	6
10	ANDRE	19
11	JOCELYNE	8
12	FELIX	12
13	JUSTIN	15
14	ETIENNE	6
15	MARIE	5

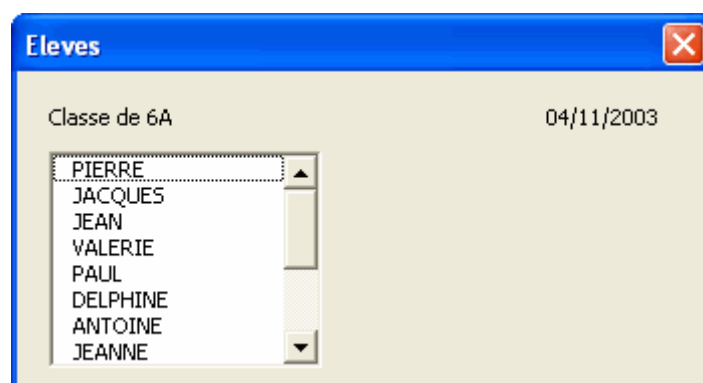
Les listes peuvent de remplir par la propriété RowSource de l'objet ListBox.



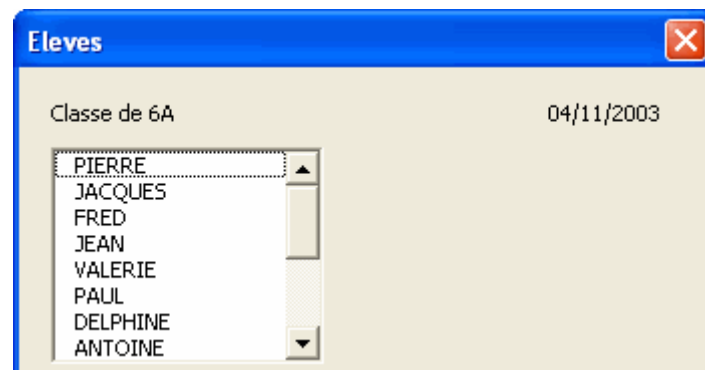
La méthode AddItem d'un objet ListBox permet d'ajouter un élément à la liste. La syntaxe est `ListBox.AddItem "Texte", Index` (Index correspond à l'emplacement du nouvel élément dans la liste). L'index du premier élément d'une liste a pour valeur 0. Si l'index n'est pas indiqué, le nouvel élément sera placé à la fin de la liste.

```
'La liste peut se remplir par la procédure suivante
Private Sub UserForm_Initialize()
    'Appel de la procédure Liste située dans un module
    Liste
End Sub

Sub Liste()
    Dim i As Integer
    For i = 1 To 14
        MaBoite.ListBox1.AddItem Range("A1").Offset(i)
    Next i
End Sub
```



```
'Ajout d'un élément situé en 2ème position
MaBoite.AddItem "FRED", 2
```



La propriété ListCount d'une zone de liste renvoie le nombre d'éléments qu'elle contient, la propriété Value sa valeur et la propriété ListIndex son index.



```
Private Sub ListBox1_Change()
    Dim i As Integer, j As Integer
    Dim Val As String
    i = ListBox1.ListCount 'renvoie 15
    j = ListBox1.ListIndex 'renvoie 4
    Val = ListBox1.Value    'renvoie "VALERIE"
End Sub
```

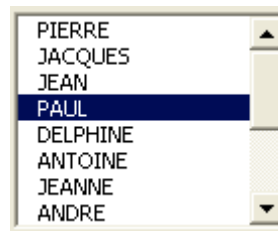
Il est également possible de remplir une zone de liste en utilisant un tableau et la propriété List.

```
Sub Liste()
    Dim i As Integer
    Dim List_Eleve(1 To 14) As String
    For i = 1 To 14
        List_Eleve(i) = Range("A1").Offset(i)
    Next i
    MaBoite.ListBox1.List = List_Eleve
End Sub
```

La suppression d'un élément d'une liste se fait par la méthode RemoveItem. La syntaxe est ListBox.RemoveItem Index. Index correspond à l'élément à supprimer.

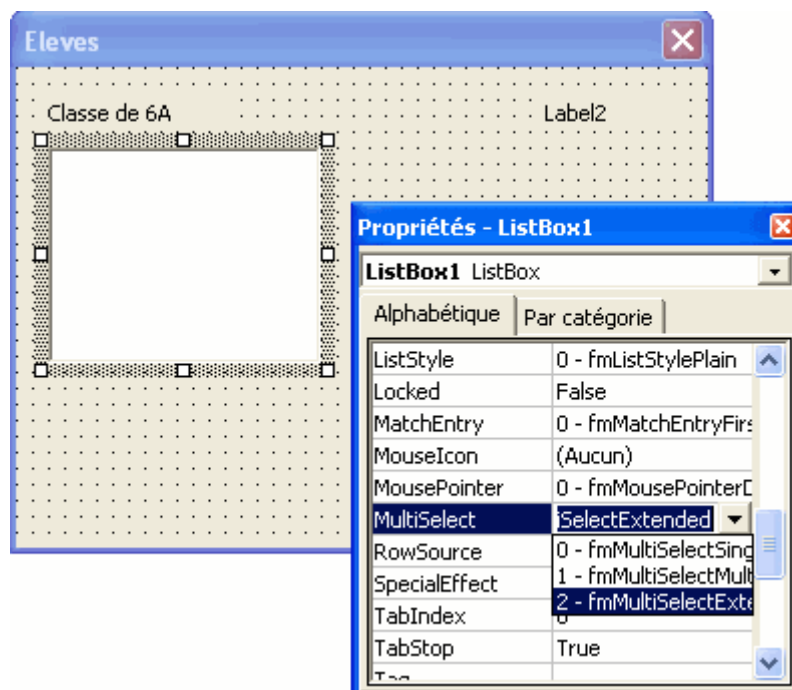


```
Dim i As Integer
i = ListBox1.ListIndex 'renvoie 4
ListBox1.RemoveItem i
```



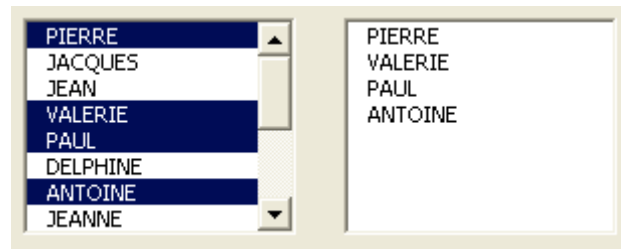
La suppression de tous les éléments d'une liste se fait par la méthode Clear. La syntaxe est `ListBox.Clear`.

Par défaut, l'utilisateur ne peut sélectionner qu'un seul élément de la liste. Pour permettre la sélection de plusieurs éléments, la propriété `MultiSelect` de la zone de texte doit être sur 1 ou sur 2.



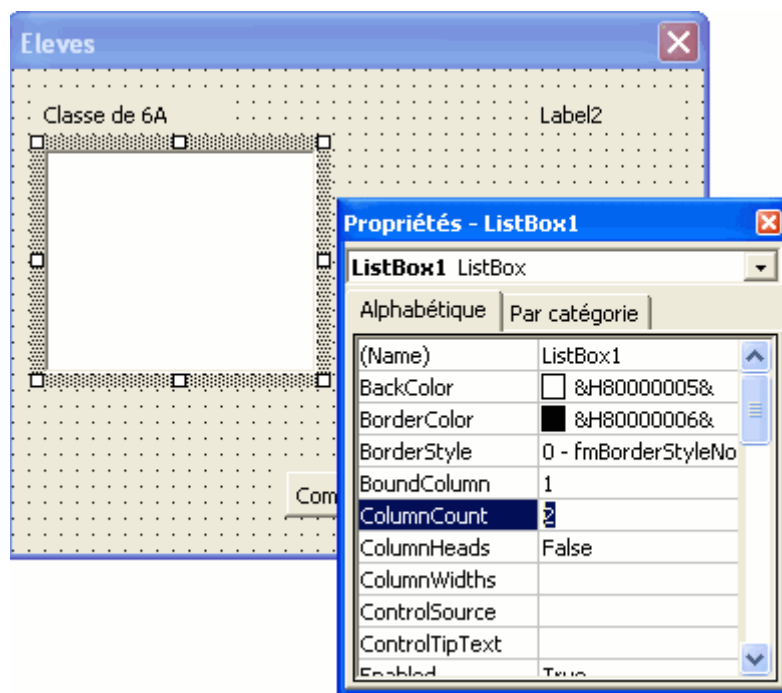
La propriété Selected(Item) détermine si un élément est sélectionné ou non. L'exemple suivant va copier les éléments sélectionnés de la ListBox1 dans la ListBox2.

```
Dim i As Integer
For i = 0 To ListBox1.ListCount - 1
    If ListBox1.Selected(i) = True Then
        ListBox2.AddItem ListBox1.List(i)
    End If
Next i
```



Une zone de liste peut contenir plusieurs colonnes. Le nombre de colonnes est défini par la propriété ColumnCount.

Dans l'exemple suivant, la zone de liste va être composée de deux colonnes.



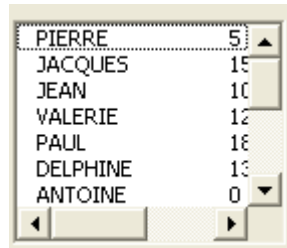
Pour remplir une zone de liste avec plusieurs colonnes, on va utiliser un tableau à plusieurs dimensions.

Dans l'exemple suivant, la zone de liste va recevoir le nom des élèves avec leurs notes.

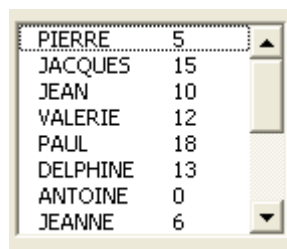
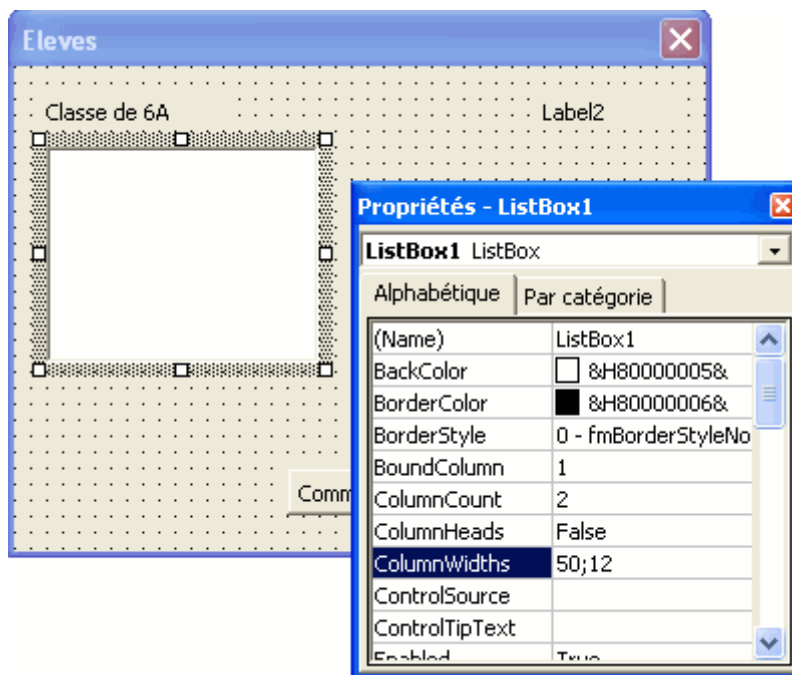
```

Sub Liste()
    Dim i As Integer
    Dim List_Eleve(1 To 14, 1 To 2) As String
    For i = 1 To 14
        List_Eleve(i, 1) = Range("A1").Offset(i)
        List_Eleve(i, 2) = Range("B1").Offset(i)
    Next i
    MaBoite.ListBox1.List = List_Eleve
End Sub

```



La largeur de chaque colonne d'une zone de liste se change par la propriété ColumnWidths. Les différentes largeurs sont séparées par le caractère ";".

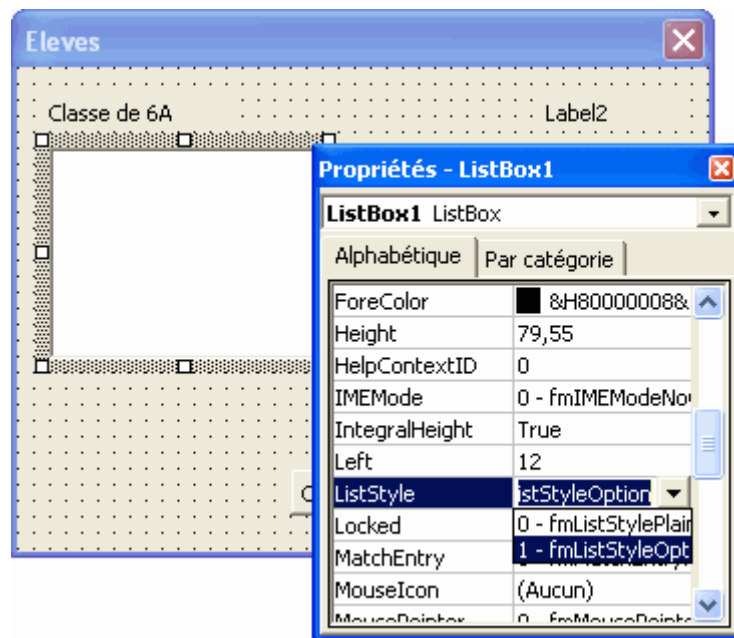


La propriété `BoudColumn` détermine dans quelle colonne la valeur est récupérée.

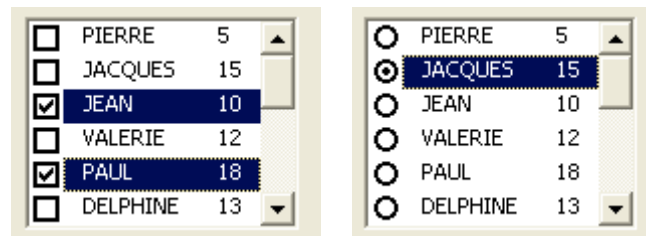


```
Dim Val As String
ListBox1.BoundColumn = 1
Val = ListBox1.Value 'renvoie "VALERIE"
ListBox1.BoundColumn = 2
Val = ListBox1.Value 'renvoie "12"
```

Il est possible de changer l'aspect d'une zone de liste par la propriété `ListStyle`.



L'aspect de la zone de liste change selon la valeur de la propriété `MultiSelect`.





ComboBox ou liste déroulante

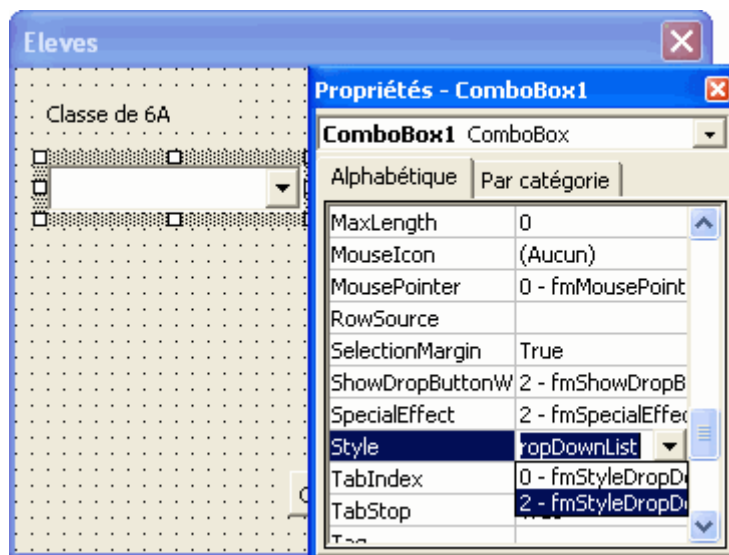
Une liste déroulante se remplit de la même façon qu'une zone de liste.

Contrairement à la zone de liste, la liste déroulante peut permettre à l'utilisateur de saisir une chaîne de caractères qui ne fait pas partie de la liste.



Si la valeur saisie ne fait pas partie de la liste ou est nulle, la propriété `ListIndex` de l'objet `ComBox` prend comme valeur -1.

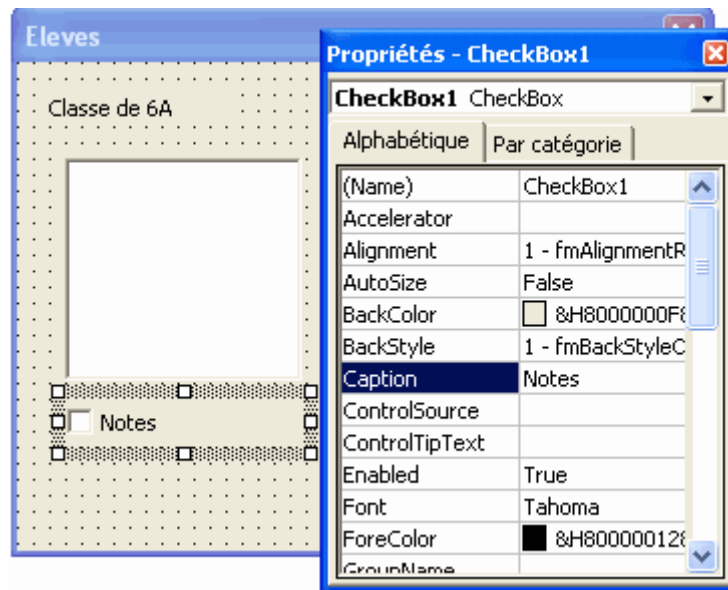
Il est possible d'interdire à l'utilisateur de saisir une chaîne de caractère qui ne fait pas partie de la liste en mettant la propriété `Style` sur 2.





CheckBox ou case à cocher

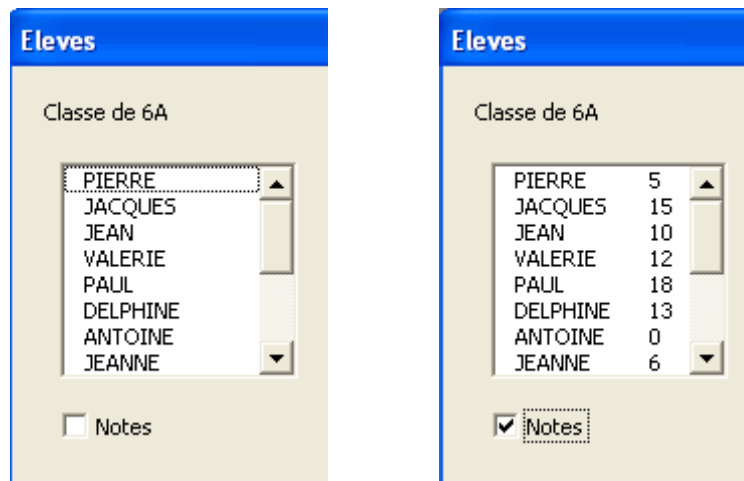
Cet outil crée des cases que l'utilisateur peut activer ou désactiver d'un simple click



Si la case à cocher est activée, sa propriété Value prend comme valeur True, sinon elle prend comme valeur False.

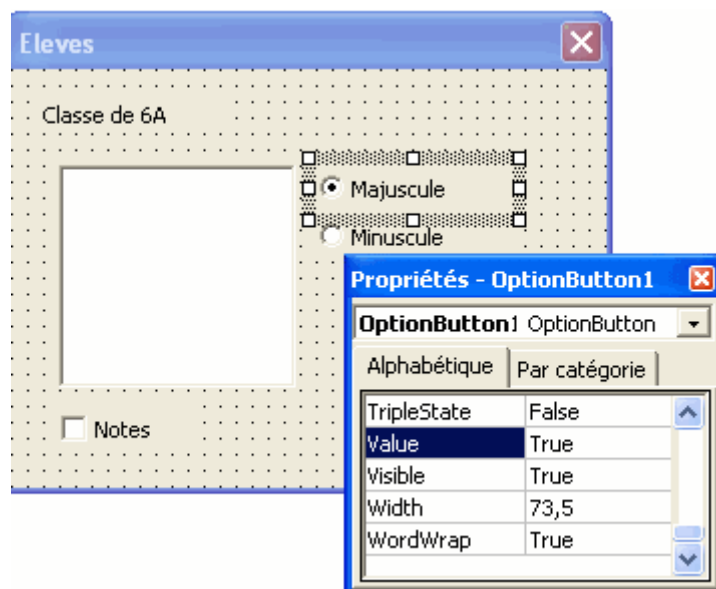
Dans l'exemple suivant, si la case à cocher est activée, les notes apparaissent, sinon elles disparaissent.

```
Private Sub UserForm_Initialize()  
    Dim i As Integer  
    Dim List_Eleve(1 To 14, 1 To 2) As String  
    For i = 1 To 14  
        List_Eleve(i, 1) = Range("A1").Offset(i)  
        List_Eleve(i, 2) = Range("B1").Offset(i)  
    Next i  
    MaBoite.ListBox1.List = List_Eleve  
End Sub  
  
Private Sub CheckBox1_Click()  
    If CheckBox1.Value = True Then  
        ListBox1.ColumnCount = 2  
    Else  
        ListBox1.ColumnCount = 1  
    End If  
End Sub
```



OptionButton ou bouton d'option

Cet outil crée des boutons de d'options. L'utilisateur ne peut sélectionner qu'un seul bouton d'un même groupe.



La propriété Value du bouton sélectionné prend la valeur True alors que la propriété Value des autres boutons du même groupe prend la valeur False.

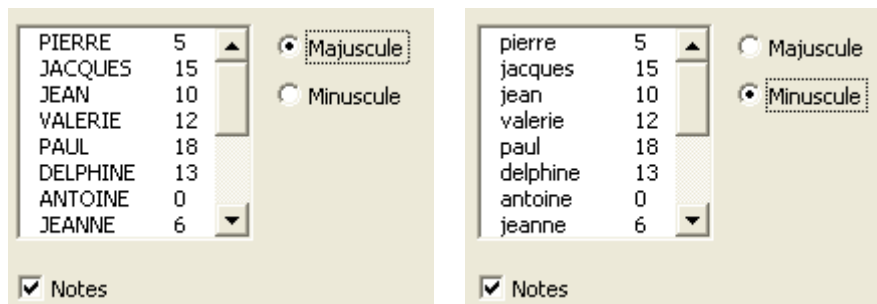
```
Private Sub OptionButton1_Click()  
    Ecrire(True)  
End Sub
```

```
Private Sub OptionButton2_Click()  
    Ecrire(False)  
End Sub
```

```

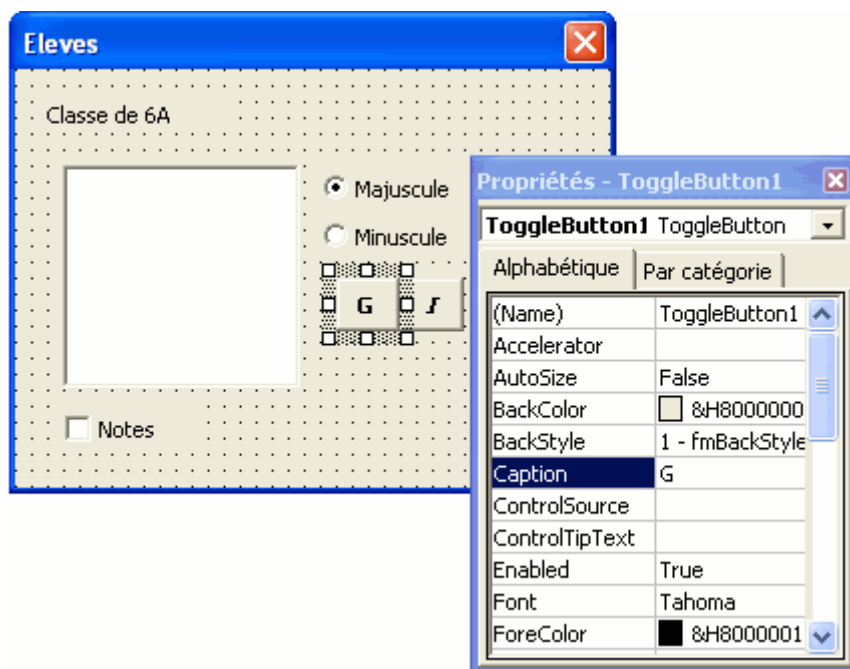
Sub Ecrire(Maj As Boolean)
    Dim i As Integer
    For i = 0 To ListBox1.ListCount - 1
        If Maj = True Then
            'Majuscule
            ListBox1.List(i) = UCase(ListBox1.List(i))
        Else
            'Majuscule
            ListBox1.List(i) = LCase(ListBox1.List(i))
        End If
    Next i
End Sub

```



ToggleButton ou Bouton à bascule

Cet outil crée un bouton qui change d'aspect à chaque click.

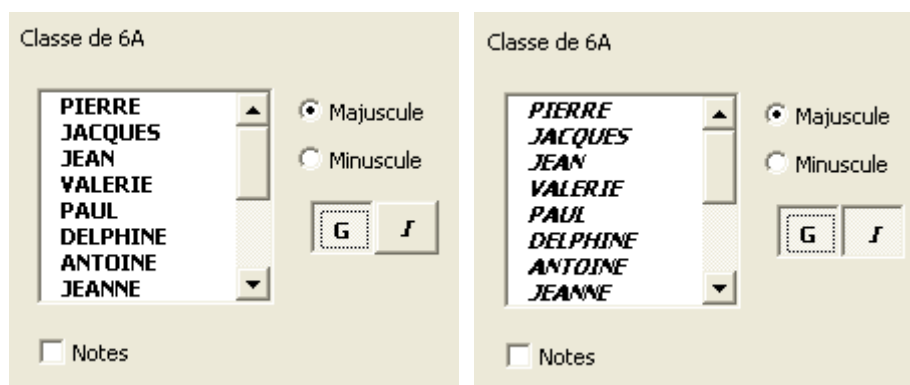


Si le bouton est enfoncé, sa valeur est égale à True sinon elle est égale à False.

```

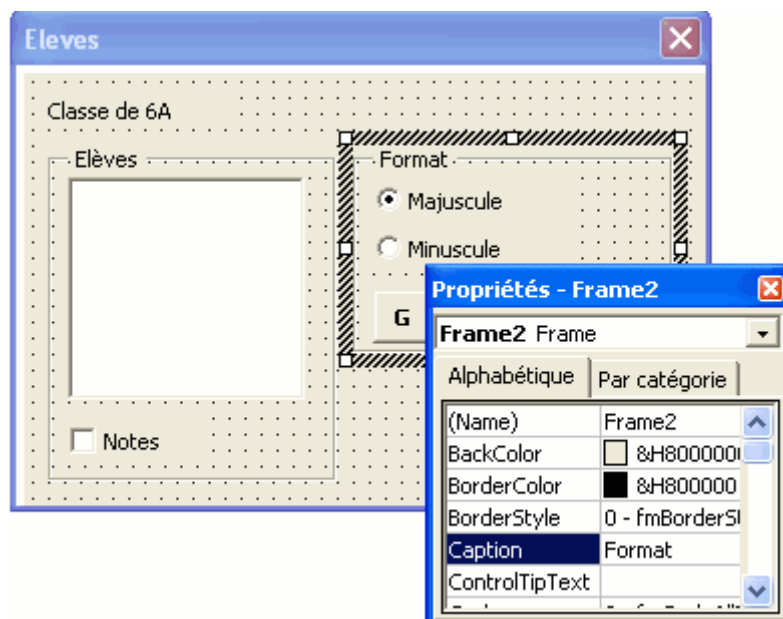
'Bouton Gras
Private Sub ToggleButton1_Click()
    If ToggleButton1 = True Then
        ListBox1.Font.Bold = True
    Else
        ListBox1.Font.Bold = False
    End If
End Sub
'Bouton Italic
Private Sub ToggleButton2_Click()
    If ToggleButton2 = True Then
        ListBox1.Font.Italic = True
    Else
        ListBox1.Font.Italic = False
    End If
End Sub

```



Frame ou cadre

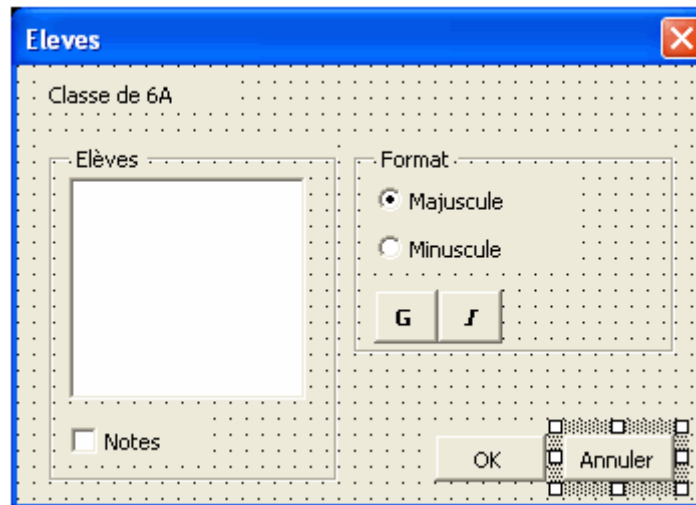
Cet outil crée des cadres permettant de grouper des contrôles.



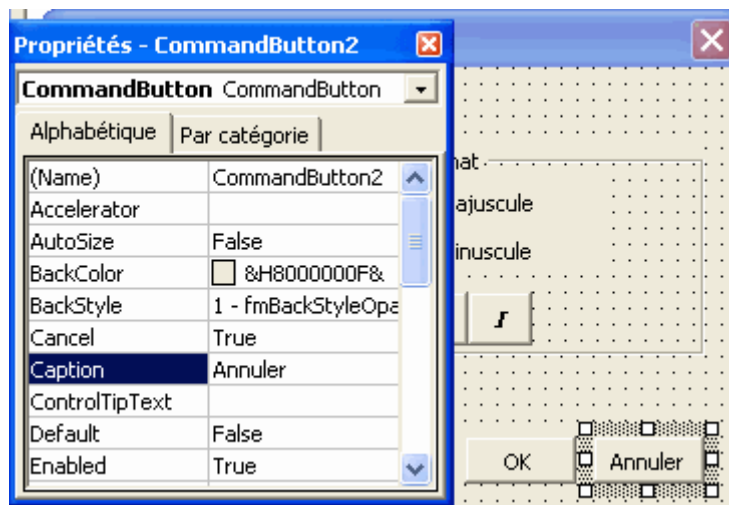


CommandButton ou Bouton de commande

Cet outil crée des boutons de commande tel que des boutons OK ou Annuler.

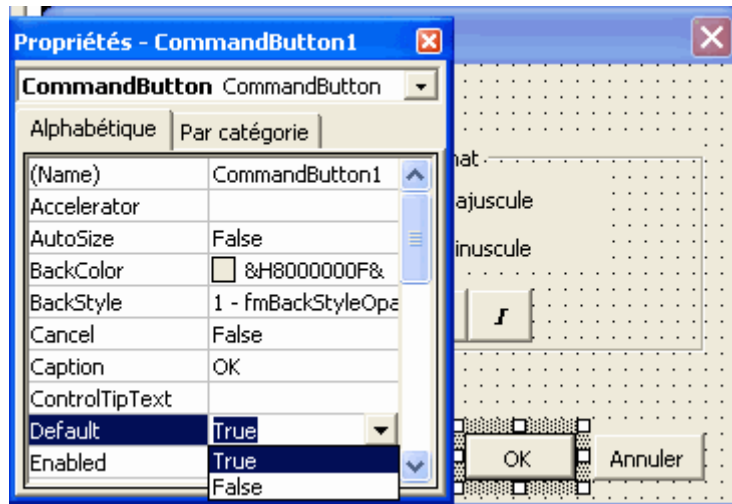


Si vous affectez la valeur True à la propriété Default d'un bouton de commande et si aucun autre contrôle n'est sélectionné, la touche Entrée équivaut à un click sur ce même bouton. De même, si vous affectez la valeur True à la propriété Cancel d'un bouton de commande, la touche Echap équivaut à un click sur le bouton.



Dans cet exemple, le fait de taper sur la touche Echap équivaut à un click sur le bouton Annuler et ferme le UserForm

```
'Bouton Annuler
Private Sub CommandButton2_Click()
    Unload MaBoite
End Sub
```



Dans cet exemple, le fait de taper sur la touche Entrée équivaut à un click sur le bouton OK si aucun autre contrôle n'est sélectionné et met à jour la liste dans la feuille de calcul.

```
'Bouton OK
Private Sub CommandButton1_Click()
    MAJListe
End Sub

Sub MAJListe()
    Dim NreENreg as Integer
    NbreEnreg = ListBox1.ListCount
    Range("A2:B" & NbreEnreg + 1) = ListBox1.List
End Sub
```

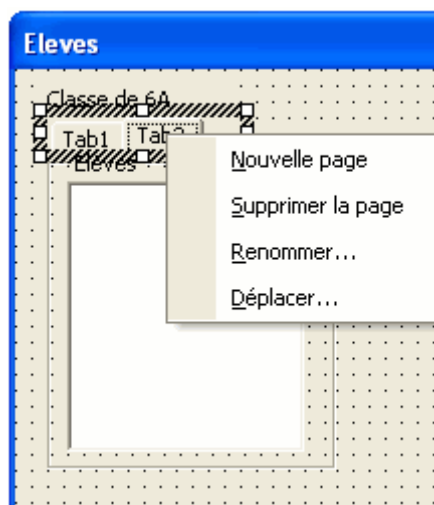


TabStrip ou étiquette d'onglet

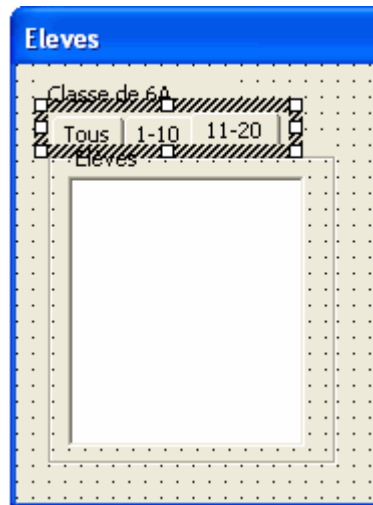
Cet outil crée des étiquettes d'onglet pour des pages identiques.

Par défaut, le nombre d'onglets d'un nouveau TabStrip est de 2 et sont nommés Tab1 et Tab2.

Un simple click avec le bouton droit de la souris permet d'en ajouter, de les renommer, de les déplacer ou de les supprimer.



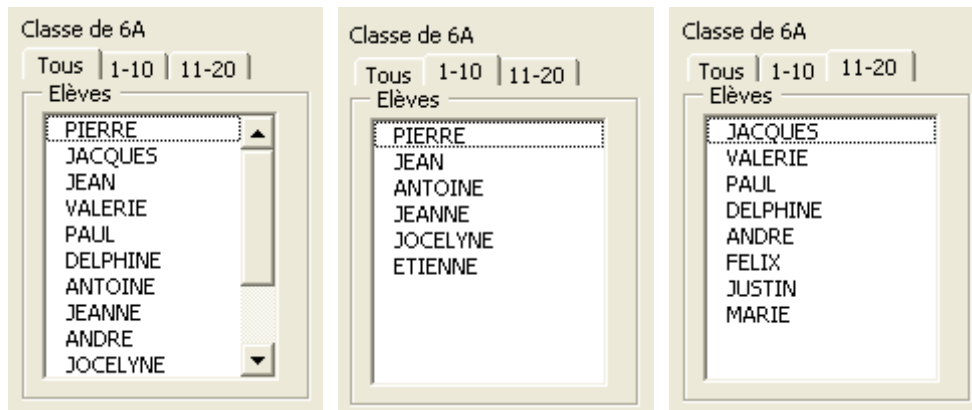
Dans l'exemple suivant, ajoutons un TabStrip avec 3 onglets permettant de classer les élèves suivant leurs notes.



L'onglet sur lequel clique l'utilisateur est déterminé par la propriété Value du Tab Strip qui prend comme valeur 0 si l'utilisateur clique sur le premier onglet, 1 si il clique sur le second, 3 si il clique sur le troisième ...

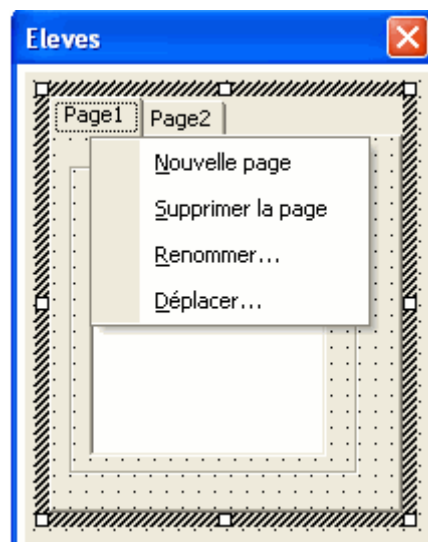
```
Private Sub TabStrip1_Click()  
    Liste(TabStrip1.Value)  
End Sub
```

```
Sub Liste(OptNote As Integer)  
    Dim Note As Integer  
    Dim Eleve As String  
    Dim i As Integer  
    Dim NoteMini As Integer  
    Dim NoteMaxi As Integer  
    MaBoite.ListBox1.Clear 'Efface le contenu de la liste  
    Select Case OptNote  
    Case 0 'Toutes les notes(onglet 1)  
        NoteMini = 0  
        NoteMaxi = 20  
    Case 1 'Notes de 0 à 10(onglet2)  
        NoteMini = 0  
        NoteMaxi = 10  
    Case 2 'Notes de 11 à 20(onglet3)  
        NoteMini = 11  
        NoteMaxi = 20  
    End Select  
    For i = 1 To 14  
        Note = Range("B1").Offset(i)  
        If Note >= NoteMini And Note <= NoteMaxi Then  
            Eleve = Range("A1").Offset(i)  
            MaBoite.ListBox1.AddItem Eleve  
        End If  
    Next i  
End Sub
```

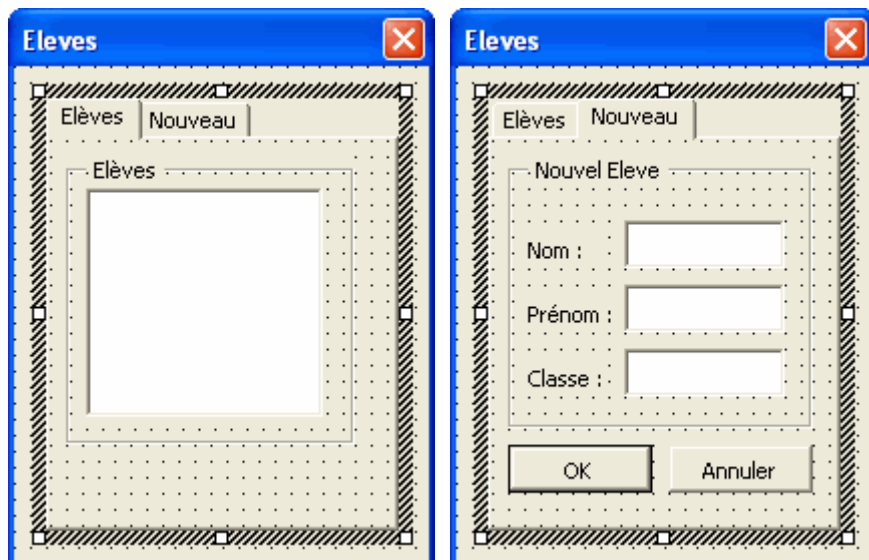



MultiPage

Un multipage peut être comparé à plusieurs UserForm dans le même. Tout comme le TabStrip, le multipage contient des onglets mais à chaque onglet correspond une nouvelle page qui contient des contrôles différents. Sa création est identique à la création d'un TabStrip.



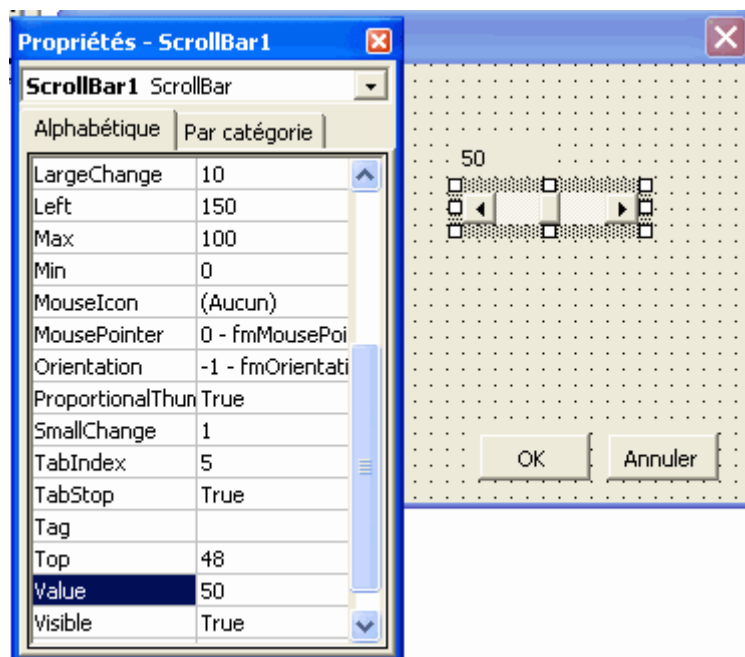
L'onglet sur lequel clique l'utilisateur est déterminé par la propriété Index du multipage qui prend comme valeur 0 si l'utilisateur clique sur le premier onglet, 1 si il clique sur le second, 3 si il clique sur le troisième ...



ScrollBar ou Barre de défilement

Une barre de défilement peut être horizontale ou verticale selon son redimensionnement.

L'exemple suivant se compose d'une barre de défilement et d'une étiquette qui reçoit sa valeur.



La valeur mini d'une barre de défilement se définit par sa propriété Min, sa valeur maxi par sa propriété Max et sa valeur par sa propriété Value.

La propriété LargeChange définit le changement de valeur lorsque l'utilisateur clique entre le curseur et l'une des flèches.

La propriété SmallChange définit le changement de valeur lorsque l'utilisateur clique sur l'une des deux flèches.

La propriété Delay définit le temps (en millisecondes) entre chaque changement lorsque l'utilisateur reste appuyer sur le Scrollbar.

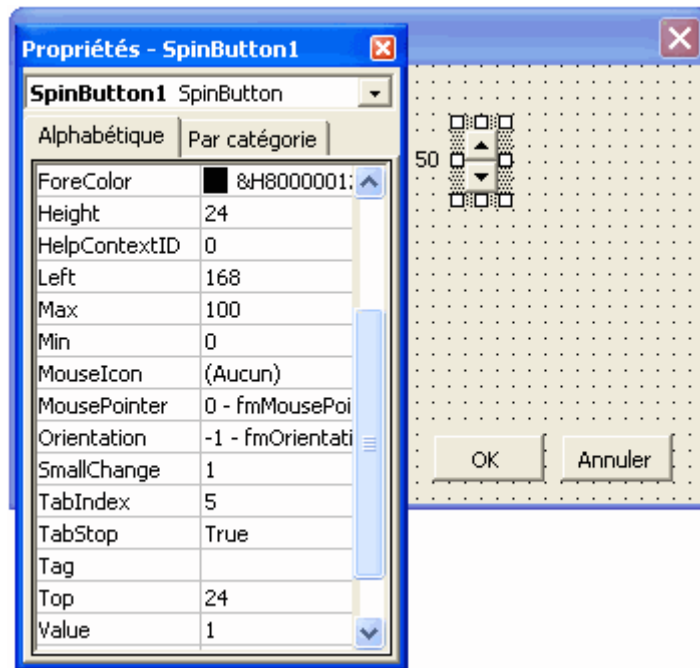
L'étiquette prend la valeur de la barre de défilement par la procédure suivante:

```
Private Sub ScrollBar1_Click()  
    Label1 = ScrollBar1.Value  
End Sub
```



SpinButton ou Bouton rotatif

Le bouton rotatif possède presque les mêmes propriétés qu'une barre de défilement. Il ne peut cependant incrémenter ou décrémenter un nombre que de la même valeur (définie dans sa propriété Value) à chaque fois.



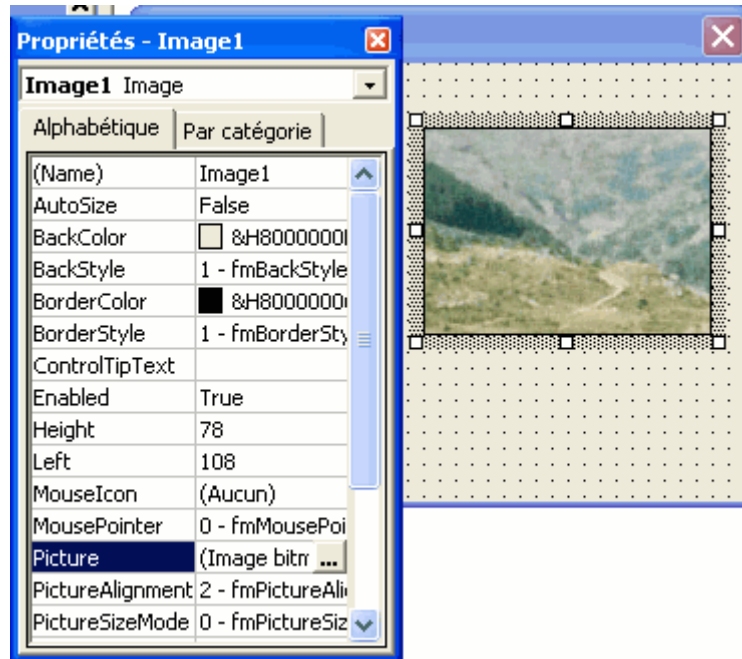
```
Private Sub SpinButton1_Click()  
    Label1 = SpinButton1.Value  
End Sub
```



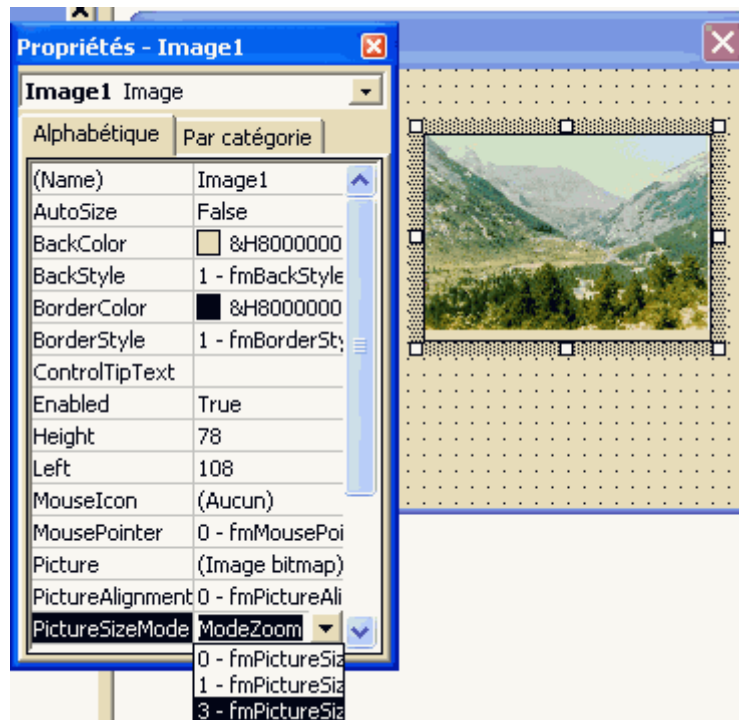
Image

Cet outil permet d'ajouter une image sur un UserForm.

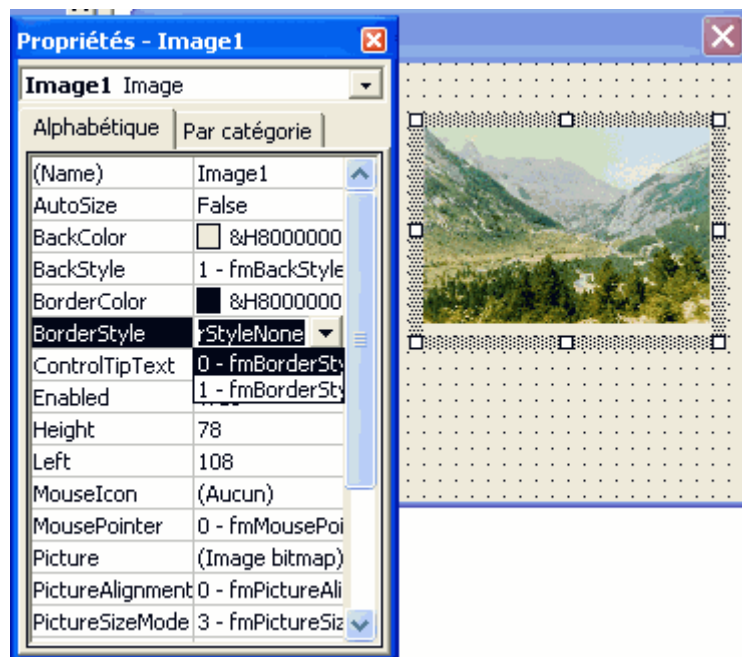
La sélection de l'image à placer se fait en cliquant sur ... de la propriété Picture.



La propriété PictureSizeMode permet de redimensionner l'image.



La propriété BorderStyle permet de supprimer le cadre autour de l'image.



Le code VB permet également de charger ou de décharger une image par la propriété Picture.

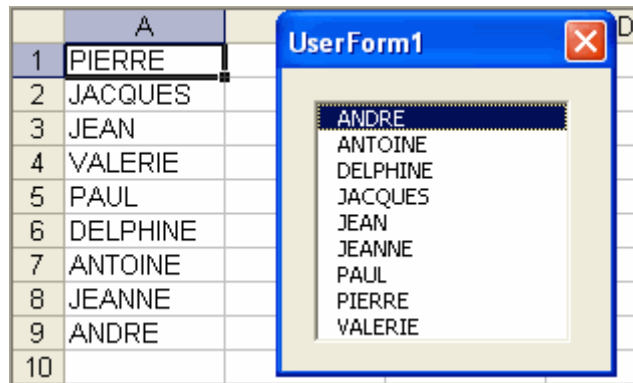
```
Dim Photo As String
Image1.Picture = LoadPicture() 'Décharge l'image
Photo = "c:\cheminphoto\photo.jpg"
Image1.Picture = LoadPicture(Photo) 'charge l'image
```

Le contrôle Image supporte les formats d'image bmp, cur, gif, ico, jpg, et wmf.

Exemples – UserForms.

Mettre en ordre alphabétique une listbox

```
Sub List_Alphab()  
    Dim i As Integer, j As Integer  
    Dim Entree As String  
    Dim Cel As Range  
    Set Cel = Range("A1")  
    'Pour chaque enregistrement  
    For i = 0 To Cel.End(xlDown).Row - 1  
        'Récupère la valeur  
        Entree = Cel.Offset(i)  
        With UserForm1  
            'Pour chaque valeur de la listBox  
            For j = 0 To .ListBox1.ListCount - 1  
                'Si la valeur de la listBox est > à la valeur à entrer  
                'on récupère l'index j et on sort de la boucle  
                If .ListBox1.List(j) > Entree Then  
                    Exit For  
                End If  
            Next j  
            'ajout de la valeur à son emplacement spécifié par l'index j  
            .ListBox1.AddItem Entree, j  
        End With  
    Next i  
    UserForm1.Show  
End Sub
```

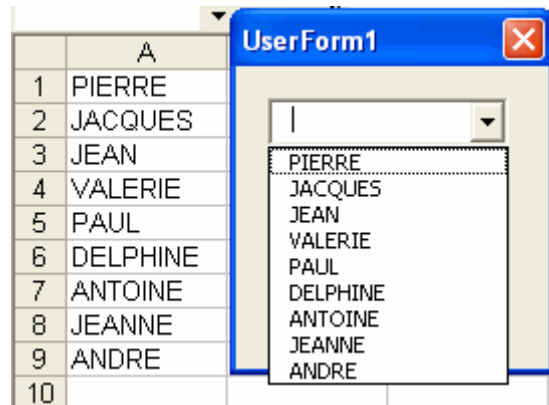


Fermer le classeur lors de la fermeture d'un Userform

```
'Cette procédure événementielle ferme le classeur sans l'enregistrer  
'lors de la fermeture de l'UserForm  
Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)  
    Workbooks("nomduclasseur.xls").Close False  
End Sub
```

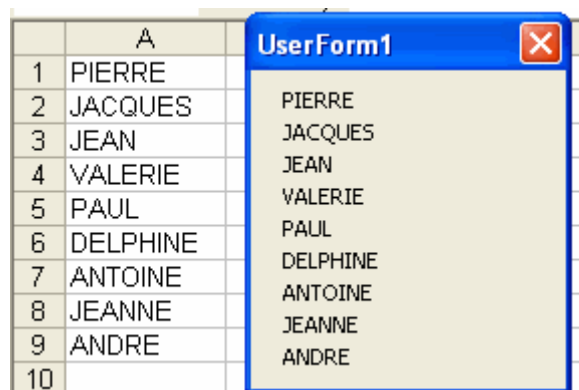
Dimensionner une zone de liste suivant le nombre d'enregistrements

```
Sub Nbre_ZoneListe()  
    Dim NbreEnreg As Integer  
    With UserForm1  
        .ComboBox1.RowSource = "A1:A9"  
        NbreEnreg = .ComboBox1.ListCount  
        .ComboBox1.ListRows = NbreEnreg  
        .Show  
    End With  
End Sub
```



Ecrire dans plusieurs contrôles en utilisant une boucle

```
Sub Ecrire_Control()  
    Dim i As Integer  
    Dim Cel As Range  
    Set Cel = Range("A1")  
    With UserForm1  
        For i = 0 To Cel.End(xlDown).Row - 1  
            .Controls("Label" & i + 1) = Cel.Offset(i)  
        Next i  
        .Show  
    End With  
End Sub
```



Partie 4 : Exercices

Exercice 1:

Vous travaillez dans une agence immobilière. Vous disposez de la base de données suivante concernant des villas de haut standing.

<i>Villa</i>	<i>Ville</i>	<i>Nbre de pièces</i>	<i>Prix</i>
Flairsou	St Tropez	10	10 000 000 €
Picsou	St Barth	14	19 000 000 €
Tirsou	St Jacques	7	13 000 000 €
Rentresou	St Simpson	6	14 000 000 €

Il vous est demandé de réaliser :

- Les formulaires répondant au besoin de cette agence.
- Le code VBA permettant de lancer l’affichage du formulaire
- Le code VBA permettant de transférer les données saisies dans le formulaire dans la feuille de calcul.

Exercice 2:

La Bibliothèque nationale lance actuellement un grand projet concernant l’informatisation de la gestion des ouvrages, périodiques et Supports audiovisuels afin d’assurer un meilleur suivi pour ses adhérents.

Le système d’information se limitera aux seules entités suivantes : livre, périodique, adhérent, auteur, maison d’édition, support audiovisuel.

Il vous est demandé de réaliser :

- Les formulaires répondant au besoin de la Bibliothèque.
- Le code VBA permettant de lancer l’affichage du formulaire
- Le code VBA permettant de transférer les données saisies dans le formulaire dans la feuille de calcul.

Exercice 3:

On se propose de réaliser un formulaire VBA répondant au besoin suivant :

Afin d’offrir un plus grand choix à sa clientèle, l’agence «Tour Opérateur » propose l’organisation d’un voyage touristique. L’agence propose 3 forfaits s’appliquant à la même destination :

- ☐ Forfait 1 : Achat billet : 1000 euros
- ☐ Forfait 2 : Achat billet + Logement inclus : 1500 euros
- ☐ Forfait 3 : Achat billet + Logement + Visite guidée : 2200 euros.

Le formulaire doit renseigner, outre les trois forfaits disponibles, des informations sur le client : Nom, Prénom, N° Tél., Adresse, Destination, Type de forfait (numéroter de 1 à 3), Mode de paiement (Chèque, CB ou cash), quantité de forfait.

Une fois ces informations saisies, le prix total de l’ensemble des forfaits pris par le client doit être affiché automatiquement sur le formulaire.

Il vous est demandé de réaliser :

- un formulaire répondant au besoin de cette agence.
- Le code VBA permettant de lancer l’affichage du formulaire
- Le code VBA permettant de transférer les données saisies dans le formulaire dans la feuille de calcul.