

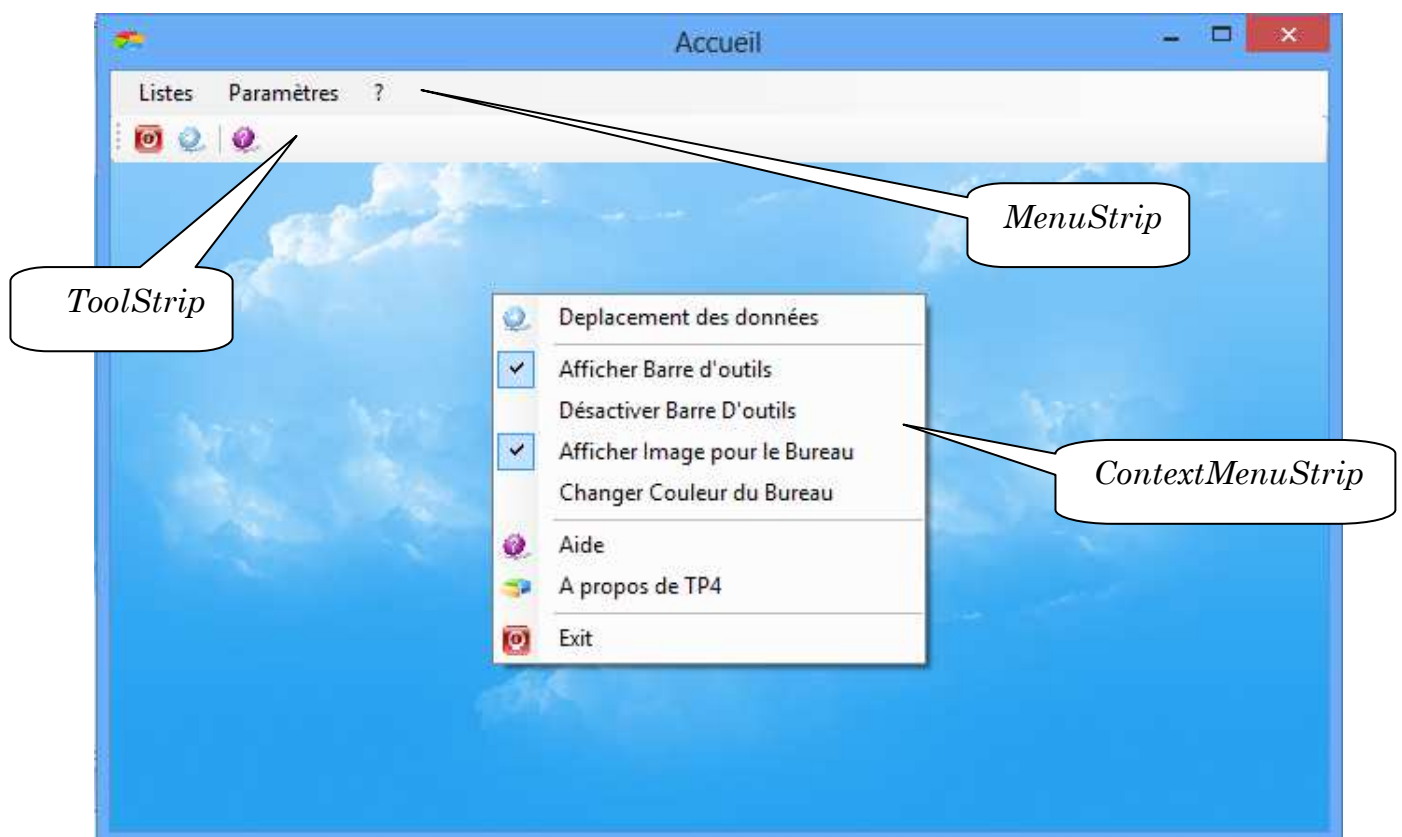
TP 4: LES MENUS ET LES LISTES

OBJECTIFS :

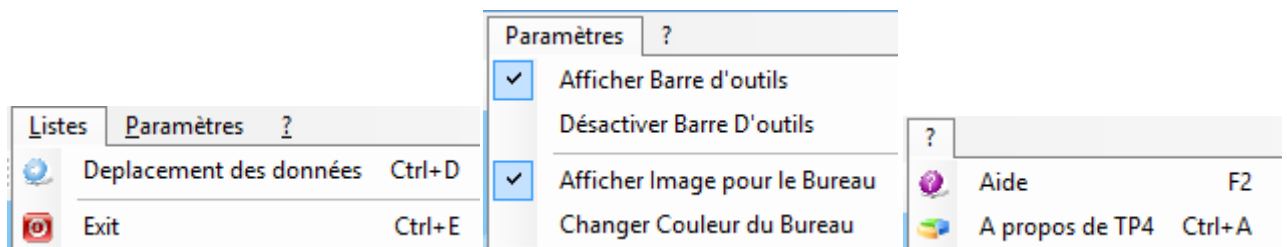
Comprendre l'utilisation des Menus (*MenuStrip*, *ContextMenuStrip* et *ToolStrip*) et quelques manipulations sur les listes.

Interface d'accueil:

- Créer l'interface principale suivante :



- Composition du sous-menu de chaque Menu :

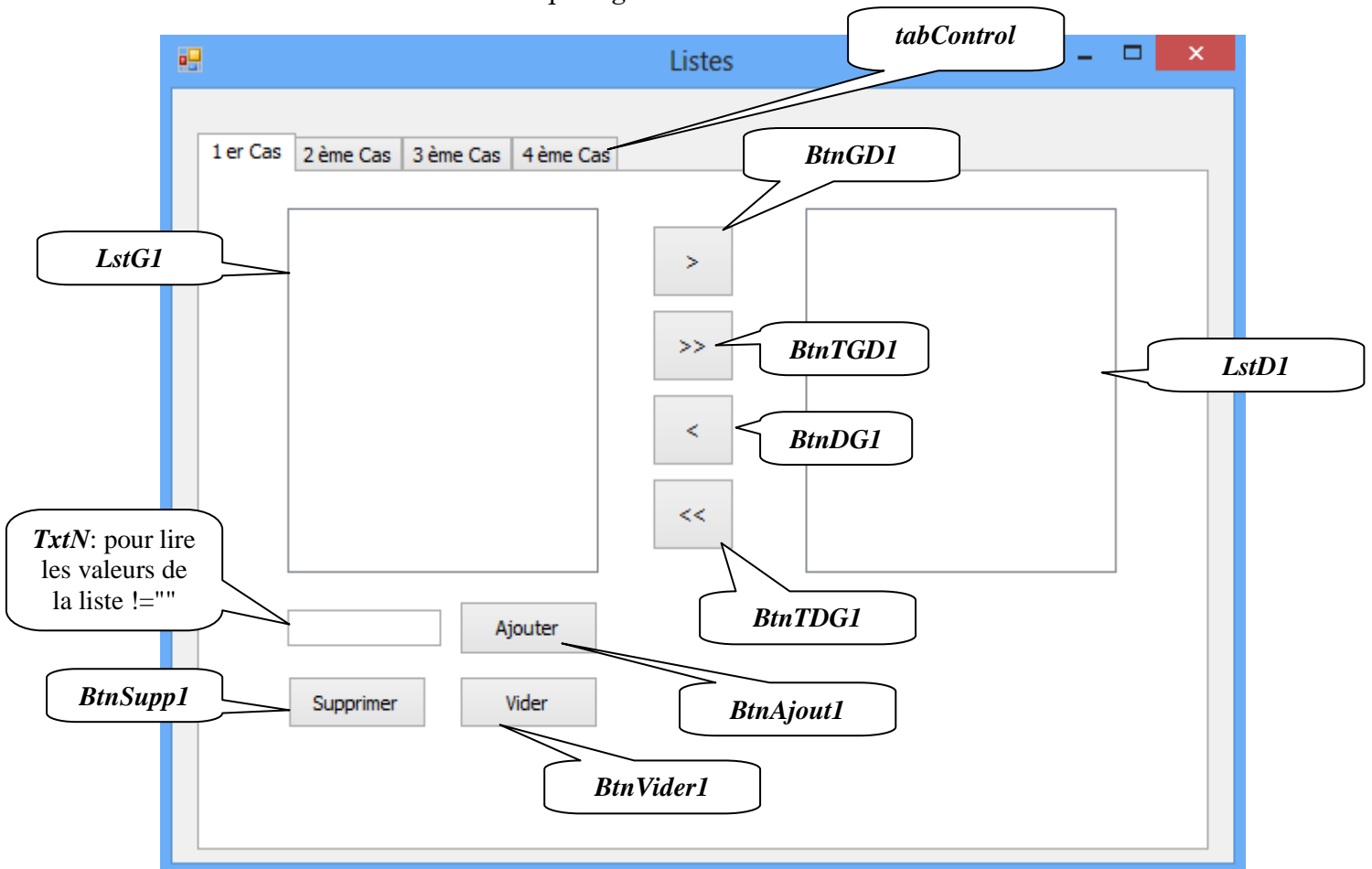


- C'est un menu qui s'ouvre quand, le curseur de l'utilisateur est sur un objet, et qu'il clique sur le bouton droit de la souris. Il faut ensuite affecter ce *ContextMenu* à un contrôle; pour cela donner à la propriété *ContextMenu* de *frmAccueil* *ContextMenu1*.

Interface de manipulation des listes:

On veut réaliser une application qui permet de déplacer les éléments d’une liste vers une autre selon les cas suivants :

- 1^{er} cas : Un seul élément est sélectionné dans une liste
- 2^{ème} cas : Il est possible de sélectionner plusieurs éléments dans une liste
- 3^{ème} cas : Utilisation des checked listbox au lieu de ListBox
- 4^{ème} cas : Utiliser le cliquer-glisser



La liste de texte est un tableau de chaînes de caractères dont le premier élément a l’indice 0. L’accès aux éléments du tableau se fait à partir de la propriété **Items**. Les propriétés principales d’une liste de texte sont :

Propriété et méthode	Description
Items(i)	Le ième élément de la liste
SelectedItem	l’élément de la liste sélectionné
SelectedIndex	Indice de l’élément sélectionné de la liste
Items.Add(element)	Ajoute <i>element</i> a la fin de la liste
Items.Remove(element)	Supprimer le premier élément <i>element</i> de la liste
Items.Insert(indexe, element)	Insérer <i>element</i> a la position indiquée par <i>indexe</i>
Items.RemoveAt(indexe)	Supprimer l’élément a la position <i>indexe</i>
Items.Clear()	Vider toute la liste
Items.Count	Nombre des éléments de la liste

Indication sur le code de la Page 1:

- Le bouton BtnAjout1:

```

if (TxtN.Text.Trim() != "")
{
    .....
    TxtN.Text = "";
    TxtN.Focus();
}
else
    MessageBox.Show("Il faut saisir une valeur");

```

- Le bouton BtnSuppl:

```

if (LstG1.SelectedIndex != -1)
    .....
else
    MessageBox.Show("Aucun élément n'est sélectionné");

```

- Le bouton BtnGD1

```

if (LstG1.SelectedIndex != -1)
{
    LstD1.Items.....
    LstG1.Items.....
}
else
    MessageBox.Show("Aucun élément n'est sélectionné");

```

- Le bouton BtnGD1

```

foreach (Object x in LstG1.Items)
    LstD1.Items.Add(x);

LstG1.Items.Clear();

```

-

Indication sur le code de la Page 2:

Pour donner la possibilité de sélectionner plusieurs éléments d'une liste il faut changer la propriété *SelectionMode* en *MultiExtended* (ou bien *MultiSimple*)

- Le bouton BtnGD2

```

List<Object> coll = new List<Object>();
foreach (Object x in ListG2.SelectedItems)
{
    coll.Add(x);
    LstD2.Items.Add(x);
}
foreach (Object y in coll)
    LstG2.Items.Remove(y);

```

- Le bouton BtnTGD2

```

foreach (Object x in LstG2.Items)
    LstD2.Items.Add(x);

LstG2.Items.Clear();

```

- Le bouton BtnSupp2:

```
List<Object> coll = new List<Object>();
foreach (Object x in LstG2.SelectedItems)
    coll.Add(x);
foreach (Object y in coll)
    LstG2.Items.Remove(y);
```

Indication sur le code de la Page 3:

Objectif: donner la possibilité de cocher directement des éléments de la liste. Remplacer donc les LisBox par des CheckedListbox.

- Le bouton BtnGD3:

```
List<Object> coll = new List<Object>();
foreach (Object x in LstG3.CheckedItems)
{
    coll.Add(x);
    LstD3.Items.Add(x);
}
foreach (Object y in coll)
```

- Le bouton BtnTGD3:

```
ListG3.Items.Remove(y);
foreach (Object x in ListG3.Items)
    ListD3.Items.Add(x);
ListG3.Items.Clear();
```

Indication sur le code de la Page 4:

Objectif: Utiliser le cliquer-glisser pour déplacer les élément entre les listes, supprimer donc tous les boutons de déplacement.

- L'évènement **DragDrop** de la liste LstG4:

```
ListG4.Items.Add(LstD4.SelectedItem);
LstD4.Items.Remove(LstD4.SelectedItem);
```

- L'évènement **DragOver** de la liste LstG4:

```
e.Effect = DragDropEffects.Move;
```

- L'évènement **MouseDown** de la liste LstG4:

```
if (LstG4.SelectedIndex != -1)
    LstG4.DoDragDrop(LstG4.SelectedItem, DragDropEffects.Move);
```

- L'évènement **DragDrop** de la liste LstD4:

.....

-