**Les Collections**

L'espace de noms ***System.Collections.Generic*** contient des interfaces et des classes qui définissent des collections génériques permettant aux utilisateurs de créer des collections fortement typées. Celles-ci fournissent une sécurité et des performances meilleures que les collections non génériques.

**La classe List**

Représente une liste fortement typée d'objets (objets de même type) accessibles par index. Fournit des méthodes de recherche, de tri et de manipulation de listes. Sa taille augmente dynamiquement au cours de l'exécution du programme.

[**Constructeurs publics**](javascript:void(0))

|  |  |
| --- | --- |
| **Nom** | **Description** |
| [List](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.arraylist(v=VS.80).aspx) | Surchargé. Initialise une nouvelle instance de la classe [List](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist(v=VS.80).aspx). |

[**Propriétés publiques**](javascript:void(0))

|  |  |
| --- | --- |
| **Nom** | **Description** |
| [Capacity](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.capacity(v=VS.80).aspx) | Obtient ou définit le nombre d'éléments que **List** peut contenir. |
| [Count](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.count(v=VS.80).aspx) | Obtient le nombre d'éléments réellement contenus dans **List**. |

La capacité de List correspond au nombre d'éléments que peut contenir List. La capacité initiale par défaut de List est 0. Lorsque des éléments sont ajoutés à List, la capacité augmente automatiquement par réallocation. La capacité peut être diminuée en appelant TrimExcess (Capacity va être égale à Count en appelant TrimExcess) ou en définissant explicitement la propriété [Capacity](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.capacity(v=VS.80).aspx).

[**Méthodes publiques**](javascript:void(0)) **de la classe ArrayList**

|  |  |
| --- | --- |
| **Nom** | **Description** |
| [Add](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.add(v=VS.80).aspx) | Ajoute un objet à la fin de **List**. |
| [AddRange](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.addrange(v=VS.80).aspx) | Ajoute les éléments de la collection spécifiée à la fin de **List**. |
| [Clear](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.clear(v=VS.80).aspx) | Supprime tous les éléments de l'objet **List**. |
| [Contains](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.contains(v=VS.80).aspx) | Détermine si un élément est dans **List**. |
| [CopyTo](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.copyto(v=VS.80).aspx) | Surchargé. Copie **List** ou une partie de celui-ci dans un tableau unidimensionnel. |
| [Equals](http://msdn.microsoft.com/fr-fr/library/system.object.equals(v=VS.80).aspx) | Surchargé. Détermine si deux instances de [Object](http://msdn.microsoft.com/fr-fr/library/system.object(v=VS.80).aspx) sont égales. (Hérité de [Object](http://msdn.microsoft.com/fr-fr/library/system.object(v=VS.80).aspx).) |
| [GetType](http://msdn.microsoft.com/fr-fr/library/system.object.gettype(v=VS.80).aspx) | Obtient le [Type](http://msdn.microsoft.com/fr-fr/library/system.type(v=VS.80).aspx) de l'instance en cours. (Hérité de [Object](http://msdn.microsoft.com/fr-fr/library/system.object(v=VS.80).aspx).) |
| [IndexOf](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.indexof(v=VS.80).aspx) | Surchargé. Retourne l'index de base zéro de la première occurrence d'une valeur dans **List** ou dans une partie de celui-ci. |
| [Insert](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.insert(v=VS.80).aspx) | Insère un élément dans **List** à l'index spécifié. |
| [InsertRange](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.insertrange(v=VS.80).aspx) | Insère les éléments d'une collection **List** à l'index spécifié. |
| [LastIndexOf](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.lastindexof(v=VS.80).aspx) | Surchargé. Retourne l'index de la dernière occurrence d'une valeur dans **List** ou dans une partie de celui-ci. |
| [Remove](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.remove(v=VS.80).aspx) | Supprime la première occurrence d'un objet spécifique de **List**. |
| [RemoveAt](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.removeat(v=VS.80).aspx) | Supprime l'élément au niveau de l'index spécifié de **List**. |
| [RemoveRange](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.removerange(v=VS.80).aspx) | Supprime une plage d'éléments de **List**. |
| [Reverse](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.reverse(v=VS.80).aspx) | Surchargé. Inverse l'ordre des éléments dans **List** ou dans une partie de celui-ci. |
| [Sort](http://msdn.microsoft.com/fr-fr/library/system.collections.arraylist.sort(v=VS.80).aspx) | Surchargé. Trie les éléments dans **List** ou dans une partie de celui-ci. |
| [TrimExcess](http://msdn.microsoft.com/fr-fr/library/ms132207(v=VS.80).aspx) | Affecte à la capacité le nombre réel d'éléments dans **List**, si ce nombre est inférieur à une valeur de seuil. |

**Exemple**

L'exemple de code suivant illustre de nombreuses propriétés et méthodes de la classe générique **List**. Le constructeur par défaut permet de créer une liste de chaînes d'une capacité de 0. La propriété [Capacity](http://msdn.microsoft.com/fr-fr/library/y52x03h2(v=VS.80).aspx) s'affiche, puis la méthode [Add](http://msdn.microsoft.com/fr-fr/library/3wcytfd1(v=VS.80).aspx) est utilisée pour ajouter plusieurs éléments à la fin de la liste. La propriété **Capacity** s'affiche à nouveau, avec la propriété [Count](http://msdn.microsoft.com/fr-fr/library/27b47ht3(v=VS.80).aspx), pour indiquer que la capacité a été augmentée en fonction des besoins.

La méthode **Contains** permet de tester la présence d'un élément dans la liste. La méthode [Insert](http://msdn.microsoft.com/fr-fr/library/sey5k5z4(v=VS.80).aspx) permet d'insérer un nouvel élément dans une position bien précise au sein de la liste, et le contenu de la liste s'affiche une nouvelle fois.

La méthode [Remove](http://msdn.microsoft.com/fr-fr/library/cd666k3e(v=VS.80).aspx) permet de supprimer la première instance d’un élément ajouté précédemment, et le contenu s'affiche une nouvelle fois.

La méthode [TrimExcess](http://msdn.microsoft.com/fr-fr/library/ms132207(v=VS.80).aspx) permet de réduire la Capacity pour qu’elle soit égale à Count, et les propriétés **Capacity** et **Count** sont affichées.

Enfin, la méthode [Clear](http://msdn.microsoft.com/fr-fr/library/dwb5h52a(v=VS.80).aspx) permet de supprimer tous les éléments de la liste, et les propriétés **Capacity** et **Count** sont affichées.

*Programme*

static void Main(string[] args)

{

List<string> noms = new List<string>();

Console.WriteLine("\nCapacity: {0}", noms.Capacity);

noms.Add("salwa");

noms.Add("adil");

noms.Add("samir");

noms.Add("imane");

noms.Add("issam");

Console.WriteLine();

foreach (string nom in noms)

{

Console.WriteLine(nom);

}

Console.WriteLine("\nCapacity: {0}", noms.Capacity);

Console.WriteLine("Count: {0}", noms.Count);

Console.WriteLine("\nContains(\"samir\"): {0}",

noms.Contains("samir"));

Console.WriteLine("\nInsert(2, \"jihane\")");

noms.Insert(2, "jihane");

Console.WriteLine();

foreach (string nom in noms)

{

Console.WriteLine(nom);

}

Console.WriteLine("\nnoms[3]: {0}", noms[3]);

Console.WriteLine("\nRemove(\"adil\")");

noms.Remove("adil");

Console.WriteLine();

foreach (string nom in noms)

{

Console.WriteLine(nom);

}

noms.TrimExcess();

Console.WriteLine("\nTrimExcess()");

Console.WriteLine("Capacity: {0}", noms.Capacity);

Console.WriteLine("Count: {0}", noms.Count);

noms.Clear();

Console.WriteLine("\nClear()");

Console.WriteLine("Capacity: {0}", noms.Capacity);

Console.WriteLine("Count: {0}", noms.Count);

Console.Read();

}