

Date: Lundi 02 mai 2011 à 01:58:03
Sujet: Programmation C# .NET

C# Trier un dictionnaire (Dictionary) par valeur

Trier un dictionnaire par valeur peut être utile en C# si on n'a un dictionnaire de clés et de valeurs. Par exemple : la clé peut être un mot (string) et la valeur un entier (int) représentant la fréquence d'apparition de ce mot dans un texte.

La collection Dictionary peut facilement être triée par clés mais le tri par valeur est un peu plus compliqué.

Il y a plusieurs solutions qui pourrait venir à l'esprit pour trier un dictionnaire en C#.

Les SortedDictionary sont une collection adaptée sauf que le tri se fait sur les clés (key) et non sur les valeurs. C'est toutesfois un type de collection qui peut s'avérer fort utile mais dans notre cas, nous ne pourrons pas les utiliser.

Il existe également les SortedList mais celle-ci ne sont utile que pour contenir une liste triée d'un type d'objet (exemple : une liste de mots sous forme de string). Pour trier un dictionnaire en fonction de ses valeurs, nous allons créer une liste de KeyValuePair qui contiendra les 2 éléments du dictionnaire (clé et valeur) triés par valeur.

```
List<KeyValuePair<string, string>> maListe = new List<KeyValuePair<string, string>>(s)
```

```
    maListe.Sort((elt1,elt2) =>
    &nbsp; &nbsp; &nbsp; {
    &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; return elt1.Value.CompareTo(elt2.Value);
    &nbsp; &nbsp; &nbsp; }
    );
```

ou plus simplement :

```
    myList.Sort((x,y)=>x.Value.CompareTo(y.Value));
```

On peut également conserver le dictionnaire initial et utiliser LINQ.

Exemple d'utilisation de LINQ pour trier un dictionnaire par valeur en C# :

```
System.Collections.Generic.Dictionary<string, int> monDictionnaire =
new Dictionary<string, int>();
```

```
&nbsp; &nbsp; myDict.Add("&quot;un&quot;", 1);
&nbsp; &nbsp; myDict.Add("&quot;quatre&quot;", 4);
&nbsp; &nbsp; myDict.Add("&quot;deux&quot;", 2);
&nbsp; &nbsp; myDict.Add("&quot;trois&quot;", 3);
```

```
&nbsp; &nbsp; var DicoTrie = (from entry in monDictionnaire orderby entry.
Value ascending select entry);
```

Bon courage

Publication de Tout sur l'Informatique - Programmation C#,
Sécurité, Divx, P2P:
<http://www.zmaster.fr>

URL de cette publication

<http://www.zmaster.fr/modules.php?name=News&file=article&sid=232>