



LINQ et collections





Qu'est-ce que Linq ?

LinQ est un langage de requête qui permet de manipuler des collections de données.

Il permet d'interroger des collections de données de type `IEnumerable`.

Il permet de faire des requêtes de type SQL.





Quels sont les avantages de Linq ?

Les avantages de Linq sont :

- ◆ Permettre de faire des requêtes de type SQL.
- ◆ Faciliter la manipulation des collections de données.
- ◆ Creation d'objet à partir d'une collection
- ◆ Selection en fonction de critères





Comment utiliser Linq ?

Presence des mots clés suivants :

- ♦ `from ... in ...`
- ♦ `where` conditions
- ♦ `select` item to return ou `new` objet

```
var result = from item in collection
              where condition
              select item;
```





Les clauses de Linq

- ◆ Clause `from` : définit la forme de la source de données.
- ◆ Clause `in` : définit la source de données.
- ◆ `where` : filtre les éléments d'une collection en fonction d'une condition donnée.
- ◆ `select` : sélectionne les propriétés d'un objet ou les transforme en un nouveau type.
- ◆ `orderby` : trie les éléments d'une collection en fonction d'une propriété donnée.





Exemple de Linq simple

Exemple type :

```
var source = ... // Définir la source de données (liste, tableau, etc.)

var result = from item in source
              where condition
              select item;

foreach (var item in result)
{
    ... // faire quelque chose avec les résultats
}
```





Exemple de Linq conversion en tableau

On peut transformer le IEnumerable en tableau

```
var source = ... // Définir la source de données (liste, tableau, etc.)  
  
var result = from item in source  
              where condition  
              select item;  
  
var array = result.ToArray();
```





Exemple de Linq dictionary

```
var dictionary = new Dictionary<int, string>
{
    { 1, "John" },
    { 2, "Jane" },
};

var result = from item in dictionary
              where item.Value.StartsWith("J")
              select item;

foreach (var item in result)
{
    Console.WriteLine("Key: {0}, Value: {1}", item.Key, item.Value);
}
```





Exemple de Linq en Objet

```
var students = ... // Définir la source de données (liste de étudiants)

var result = from student in students
              where student.Age > 18
              select new Student
              {
                  Name = student.Name,
                  Age = student.Age,
                  Address = student.Address
              };

foreach (var student in result)
{
    ... // faire quelque chose avec les étudiants sélectionnés
}
```

