

.NET

Free. Cross-platform. Open source.
A developer platform for building all your apps.

www.dot.net

Generics

- Introduits avec le .NET Framework 2.0
- type-safe: permet de définir des structures de données sans s'engager dans un type de données concret
- performant: évite le boxing et l'unboxing
- Nullable<T> est un générique

<https://learn.microsoft.com/en-us/dotnet/standard/generics>

Collections Génériques

- `IEnumerable<T>` : pour itérer dessus uniquement avec un `foreach`
- `ICollection<T>` : permet de compter, ajouter et supprimer des éléments
- `IList<T>` : ordonne les éléments, permet de récupérer un élément par sa position

<https://learn.microsoft.com/en-us/dotnet/api/system.collections.generic>

Dictionnaires et Set

- `IDictionary<TKey, TValue>` : permet de récupérer un élément par sa clé
- `ISet<T>` : permet de savoir si un élément est présent ou non, et de faire des comparaisons de set (Intersections, overlaps, subset etc...)

<https://learn.microsoft.com/en-us/dotnet/api/system.collections.generic>

Queue et Stack

- Queue<T> : structure first-in-first-out (une file)
- Stack<T> : structure last-in-first-out (une pile)

<https://learn.microsoft.com/en-us/dotnet/api/system.collections.generic>

Autres structures

- SortedList, SortedSet, SortedDictionary : structures ordonnées
- IReadOnlyCollection, IReadOnlyList, IReadOnlySet, IReadOnlyDictionary : Collection en lecture seule

<https://learn.microsoft.com/en-us/dotnet/api/system.collections.generic>

Bonnes pratiques

- Toujours demander le type le moins précis
- Choisir la collection adapté au besoin

Guidelines for collections

Complexité Algorithmique des collections

Linq: Language-Integrated Query

- Ensemble d'API pour interroger des collections
- Execution **lazy**

<https://learn.microsoft.com/fr-fr/dotnet/csharp/linq/>

<https://learn.microsoft.com/fr-fr/dotnet/csharp/tutorials/working-with-linq>

Exercice: katas-linq

<https://programmingheadache.com/2024/08/04/unlocking-linq-basic-katas-for-c-programmers/>

<https://github.com/FractalCodeRicardo/katas-linq>

Entity Framework Core

- ORM maintenu par Microsoft
- Utilise LINQ pour écrire le code des requêtes
- Peut être utilisé avec plusieurs moteurs de bases de données

<https://learn.microsoft.com/fr-fr/ef/>

EFCore cli

```
dotnet add package Microsoft.EntityFrameworkCore.SqlServer
```

```
dotnet add package Microsoft.EntityFrameworkCore.Design
```

```
dotnet tool install --global dotnet-ef
```

```
dotnet ef dbcontext scaffold "Data Source=localhost,1433;Initial  
Catalog=SystemeNotes;Integrated Security=False;User  
ID=sa;Password=MyPasswordIsS3cure!;MultipleActiveResultSets=True;TrustServe  
rCertificate=True" Microsoft.EntityFrameworkCore.SqlServer
```