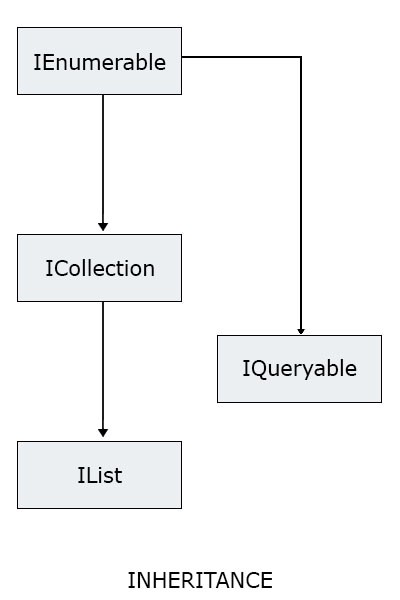
IEnumerable vs ICollection vs IList vs IQueryable in C#

In C#, we have been using collections like IEnumerable, ICollection, IList interchangeably but do you really know what is it that separates each other.

They all have specific characteristics that differentiate them and makes them adaptable to certain scenarios.

All of the above holds a list maintaining the order of items but which one should we use and in which case, let’s see.



IEnumerable vs ICollection vs IList vs IQueryable in C#

**IEnumerable**

Namespace: **System.Collections**

An IEnumerable is a list or a container which can hold some items. You can iterate through each element in the IEnumerable. You can not edit the items like adding, deleting, updating, etc. instead you just use a container to contain a list of items. It is the most basic type of list container.

All you get in an IEnumerable is an enumerator that helps in iterating over the elements. An IEnumerable does not hold even the count of the items in the list, instead, you have to iterate over the elements to get the count of items.

An IEnumerable supports filtering elements using where clause.

**ICollection**

Namespace: **System.Collections**

ICollection is another type of collection, which derives from IEnumerable and extends it’s functionality to add, remove, update element in the list. ICollection also holds the count of elements in it and we does not need to iterate over all elements to get total number of elements.

The count of total items can be retrieved in O(1) time.

ICollection supports enumerating over the elements, filtering elements, adding new elements, deleting existing elements, updating existing elements and getting the count of available items in the list.

**IList**

Namespace: **System.Collections**

IList extends ICollection. An IList can perform all operations combined from IEnumerable and ICollection, and some more operations like inserting or removing an element in the middle of a list.

You can use a foreach loop or a for loop to iterate over the elements.

**IQueryable**

Namespace: **System.Linq**

IQueryable extends ICollection. An IQueryable generates a LINQ to SQL expression that is executed over the database layer. Instead of the generating a Func<T, bool> like the ones above, IQueryable generates an expression tree and gives Expression<Func<T, bool>> that is executed over the database layer to get data set.