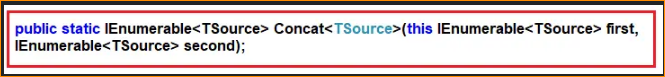
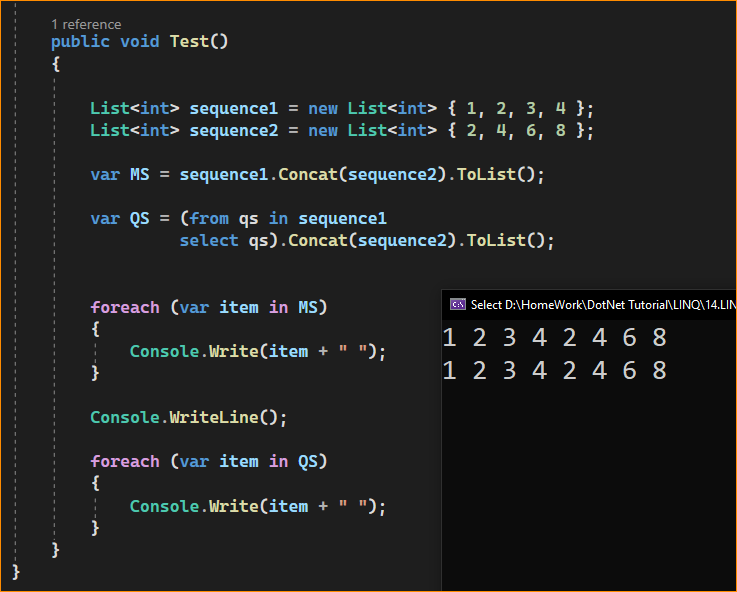
1. LINQ Concat Method in C#:

The LINQ Concat Method in C# is used to concatenate two sequences into one sequence. The point that you need to remember is it is used to concatenate two same types of sequences or collections and return a new sequence or collection without removing the duplicate elements. There is only one version available for this method whose signature is given below.



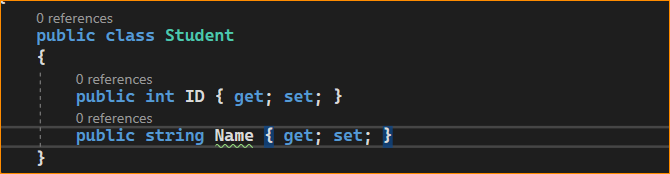
* Example to Understand Concat Method with Primitive Data Types in C#:

Let us see an example to Understand the LINQ Concat Method with Primitive Data Types in C# using both Method and Query Syntax. In the following example, we have created two integer collections which are going to be our data sources, and then we concatenate the two collections into one collection using the Concat Method. Here, you can see, both data collection data type is the same i.e. List<int>, otherwise, we will get a compilation error. In Query Syntax, there is no such operator called concat, so here we need to use the mixed syntax i.e. both the query and method syntax to achieve the same.

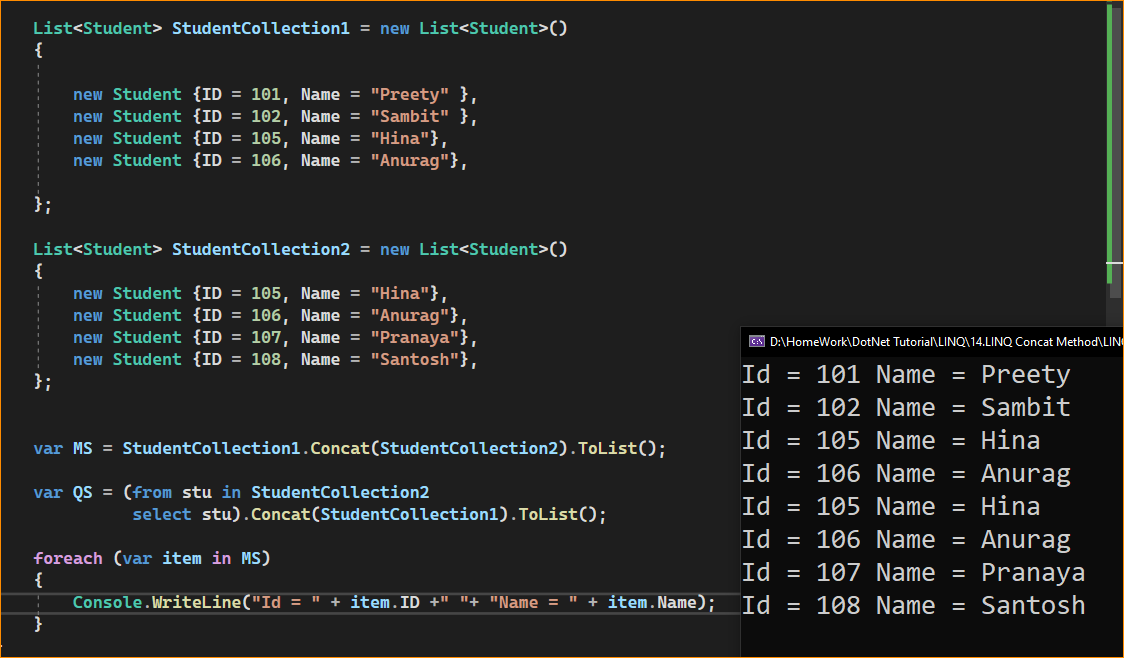


LINQ Concat Method with Complex Data Type in C#:

Let us understand how the LINQ Concat Method works with Complex types. Create a class file with the name Student.cs and then copy and paste the following code into it.



Now, we need to Concatenate the above two data sources into a single data source without removing the duplicate elements. Here, you can see two students appeared in both data sources. Now, to do so, we need to use the LINQ Cancat method. So, modify the Main method of the Program class as follows.



* What is the Difference Between LINQ Concat and Union Method in C#?

**The Concat operator is used to concatenate two sequences into one sequence without removing the duplicate elements**. That means it simply returns the elements from the first sequence followed by the elements from the second sequence. On the other hand, the LINQ Union operator is also used to concatenate two sequences into one sequence by removing duplicate elements.