1. Why do we need LINQ ThenBy and ThenByDescending Method in C#?

The LINQ OrderBy and OrderByDescending method works fine when you want to sort the data based on a single value or a single expression. But if you want to sort the data based on multiple values or multiple expressions then you need to use the LINQ ThenBy and ThenByDescending Method along with the OrderBy or OrderByDescending Method.

1. What are the Linq ThenBy and ThenByDescending Methods in C#?

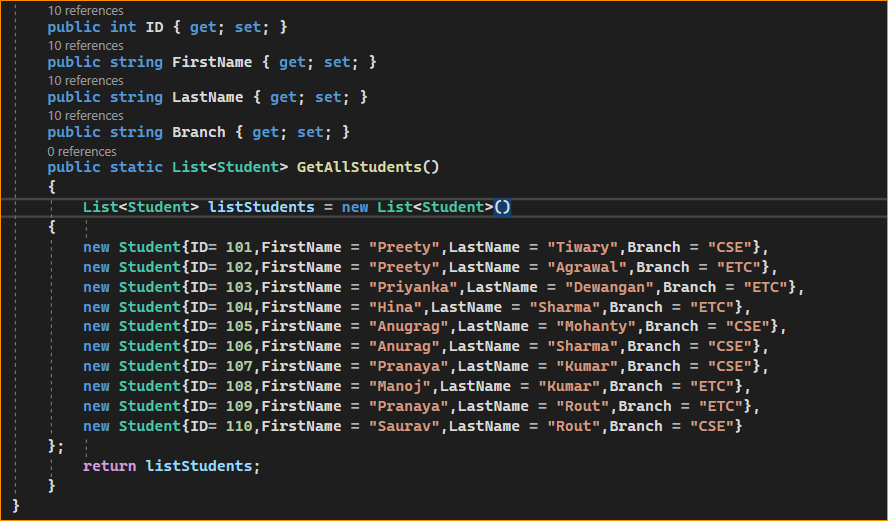
The Linq ThenBy Method in C# is used to sort the data in Ascending order from the second level onwards. On the other hand, the Linq ThenByDescending Method is used to sort the data in Descending order also from the second level onwards. Or you can say if you want to sort the data by multiple keys, then the first key will be sorted by the OrderBy or OrderByDescending method, but from the second key onwards, the data is going to be sorted by ThenBy and ThenByDescending methods.

These two methods are used along with OrderBy or OrderByDescending methods. You can use the ThenBy or ThenByDescending method more than once in the same LINQ query but you can use OrderBy or OrderByDescending methods only once in the same query.

The OrderBy or OrderByDescending method is generally used for primary sorting. The LINQ ThenBy or ThenByDescending Methods are used for secondary sorting and so on. For example, first, sort the student by First Name and then sort the student by Last Name. In this case, First Name needs to be sorted by OrderBy or OrderByDescending method, and the Last Name needs to be sorted by ThenBy or ThenByDescending method based on our requirements.

1. Examples to Understand LINQ ThenBy and ThenByDescending Methods in C#:

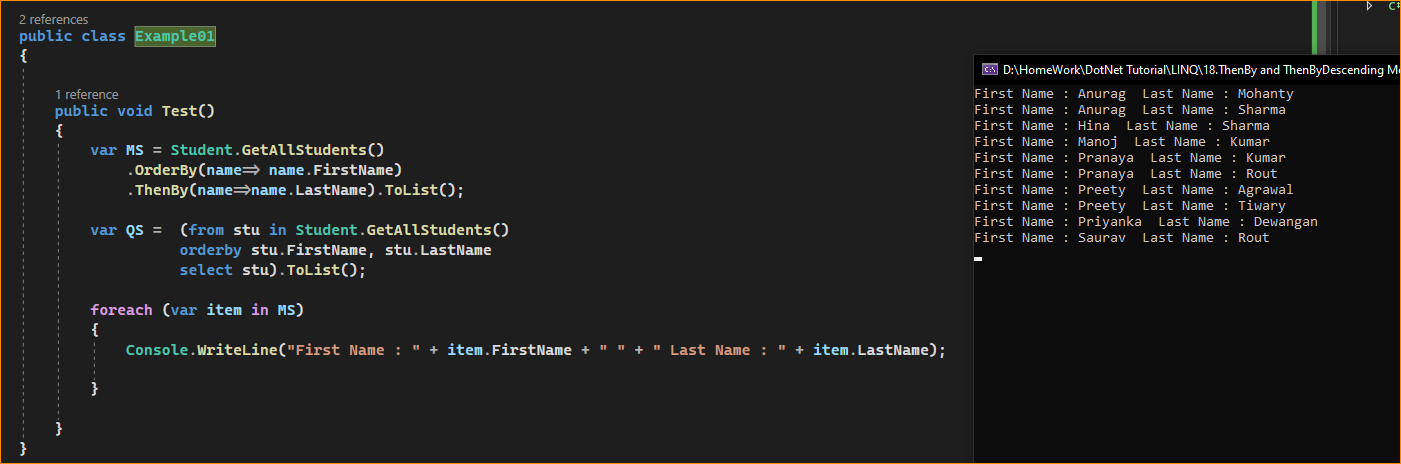
As we are going to use ThenBy or ThenByDescending methods, so need to work with Complex types. It is not possible to use these two methods with primitive data types as we are going to sort the data based on multiple keys. So, let us see how can we work with Complex data types using LINQ ThenBy or ThenByDescending methods. We are going to use the following Student class in order to understand the use of the ThenBy and ThenByDescending methods in C#. So, create a class file with the name Student.cs and then copy and paste the following code into it.



As you can see, we created the above Student class with four properties such as ID, FirstName, LastName, and Brach. We then created one method i.e. GetAllStudents() which is going to return a list of students which is going to be our data source and on this data source, we are going to perform our operations.

1. Example to Understand LINQ ThenBy Method Using Method Syntax in C#:

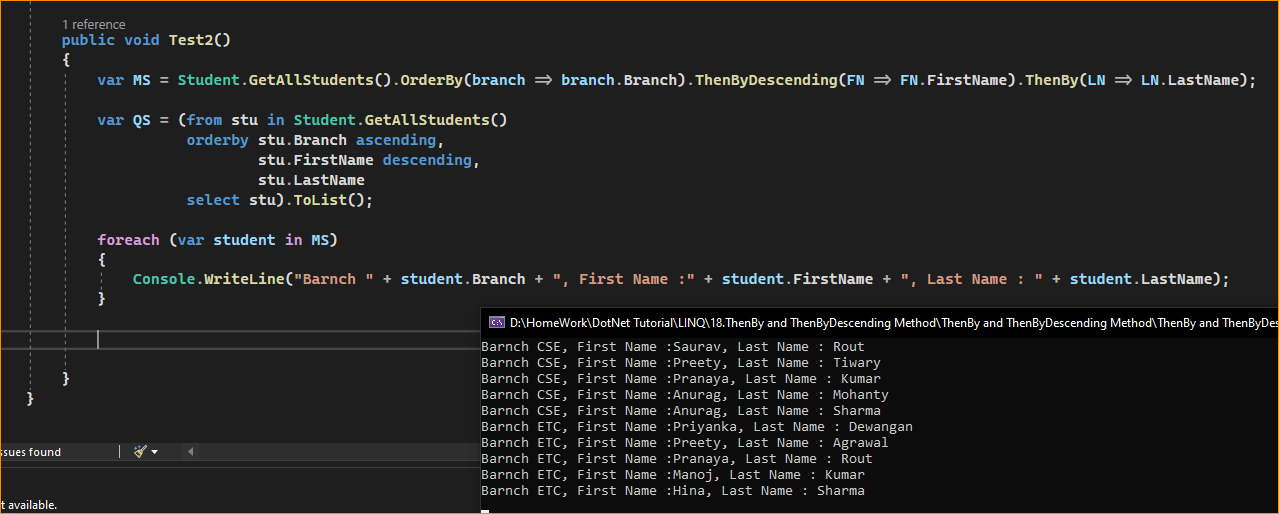
Our requirement is to sort the student by First Name in Ascending Order and then we need to sort the student by Last Name in Ascending order. So, here, we are going to sort the data based on two keys i.e. FirstName and LastName. So, in this case, the first sorting is going to be done by the OrderBy method and the second sorting is going to be done by the ThenBy method which is shown in the below example.



Example to Understand Ascending and Descending with Multiple Key Values:

Let us see an example to Understand How we can Perform the Ascending and Descending Operation with Multiple Key Values using both Method and Query Syntax in C#.

* Our requirement is, we need first sort the students in ascending order based on the Branch. Then we need to sort the students in descending order based on their First Names. And, finally, we need to sort the students in ascending order based on their Last Names. The following example exactly does the same.



1. Example Using ThenBy and ThenByDescending Method with Where Extension Method:

Let us see an example to understand how can we use LINQ ThenBy and ThenByDescending Method with Where Extension Method in C#. Our requirement is,

first we need to fetch only the CSE branch students and then we need to sort the data as follows.

* First sort the students in ascending order based on First Name.
* Then sort the students in descending order based on their Last Names.

