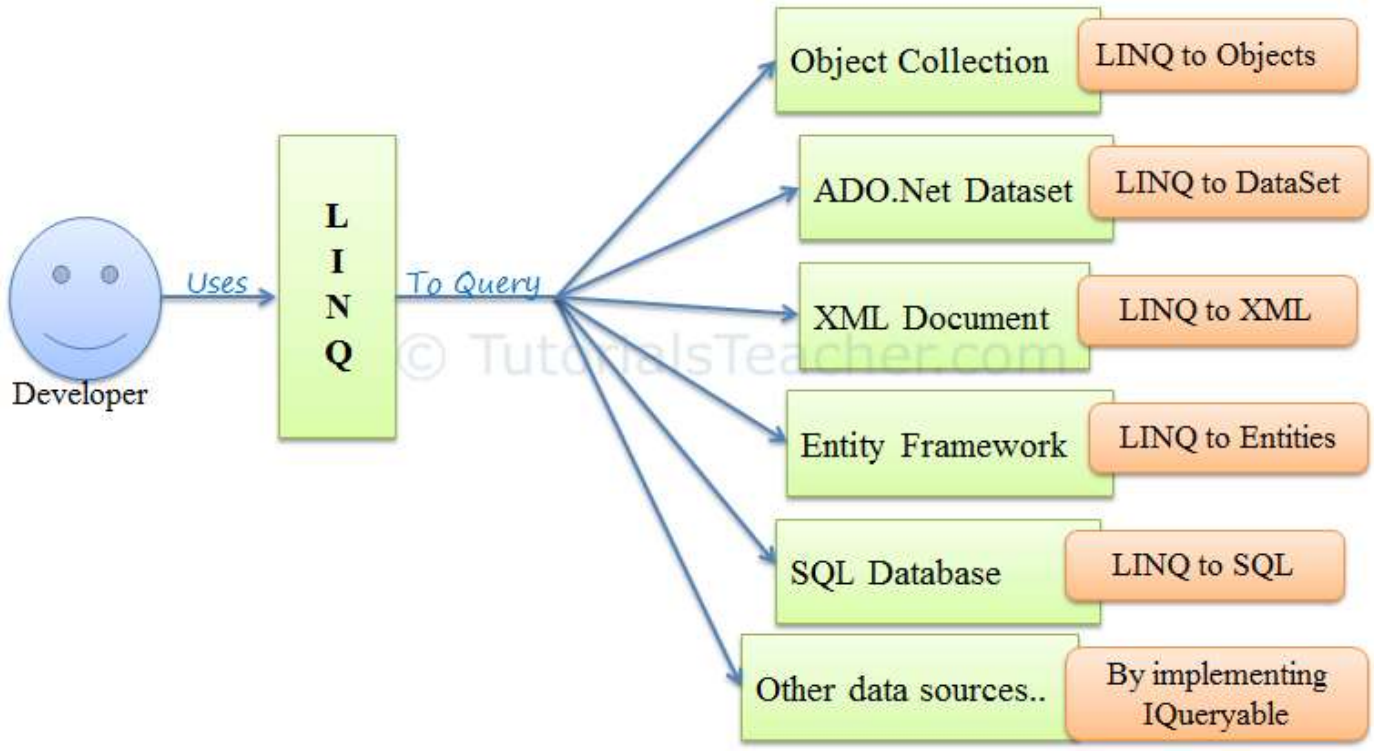
Previous Next

What is LINQ?

LINQ (Language Integrated Query) is uniform query syntax in C# and VB.NET to retrieve data from different sources and formats. It is integrated in C# or VB, thereby eliminating the mismatch between programming languages and databases, as well as providing a single querying interface for different types of data sources.

For example, SQL is a Structured Query Language used to save and retrieve data from a database. In the same way, LINQ is a structured query syntax built in C# and VB.NET to retrieve data from different types of data sources such as collections, ADO.Net DataSet, XML Docs, web service and MS SQL Server and other databases.



LINQ Usage

LINQ queries return results as objects. It enables you to uses object-oriented approach on the result set and not to worry about transforming diffent formats of results into objects.

The following example demonstrates a simple LINQ query that gets all strings from an array which contains 'a'.

Example: LINQ Query to Array

```
// Data source
string[] names = {"Bill", "Steve", "James", "Mohan" };

// LINQ Query
var myLinqQuery = from name in names
                  where name.Contains('a')
                  select name;

// Query execution
foreach(var name in myLinqQuery)
    Console.Write(name + " ");
```

Try it

In the above example, string array names is a data source. The following is a LINQ query which is assigned to a variable `myLinqQuery`.

```
from name in names
where name.Contains('a')
select name;
```

The above query uses query syntax of LINQ. You will learn more about it in the Query Syntax chapter.

You will not get the result of a LINQ query until you execute it. LINQ query can be execute in multiple ways, here we used `foreach` loop to execute our query stored in `myLinqQuery`. The `foreach` loop executes the query on the data source and get the result and then iterates over the result set.

Thus, every LINQ query must query to some kind of data sources whether it can be array, collections, XML or other databases. After writing LINQ query, it must be executed to get the result.

Learn why should we use LINQ in the next chapter.

Previous

Next