# Exercises for Chapter 23 – Introduction to LINQ

## Exercise 1

Write a **ForEach** extension method for any **IEnumerable<T>** that will provide the following semantics:

myEnumerable.ForEach(e => Console.WriteLine(e));

## Exercise 2

Write a **MaxElement** extension method for any **IEnumerable<T>** that will provide the following semantics (assume that the comparable property – e.g. **e.Age** - is always an integer):

Employee oldestEmployee = employees.MaxElement(e => e.Age);

## Exercise 3

Write an extension method that will facilitate turning anonymous types into tuples. A tuple is a generic type of the following form:

public class Tuple<T> { public T R1 { get; set; } }

public class Tuple<T1,T2> : Tuple<T1> { public T2 R2 { get; set; } }

public class Tuple<T1,T2,T3> : Tuple<T1,T2> { public T3 R3 { get; set; } }

…and so on.

Develop a tuple hierarchy of up to 5 generic arguments. Write an extension method or a set of extension methods that will work for any anonymous type (and regular types with properties as well) and provide the following semantics:

var anonType = new { Name=…, Age=… };

Tuple<string, int> tuple = anonType.ToTuple<string, int>();

## Exercise 3

Implement some of the LINQ query operators: **LastOrDefault**, **Cast**, **Concat**, **Aggregate**.

## Exercise 4

Implement the query pattern as extensions methods for the **DataTable** class. Ensure that basic language query operators such as **where**, **select** and **orderby** work correctly.

In order to test your implementation without a database, use an XML file and the **DataSet.ReadXml** method for reading the file contents into the **DataSet**.

## Exercise 5

Write a LINQ query that will display the 10 files that were last modified in the C:\ drive, recursively (including subdirectories).

## Exercise 6

Write a LINQ query that will produce a grouping of type names to the number of methods defined on these types. Use that query to list some of the public types with the largest number of methods in the .NET framework.