Seminar

SeC6.3

Chapter 6. Persistence

Language-Integrated Query (LINQ)

Software Engineering

Computer Science School
DSIC – UPV

Goals

- Introduce the LINQ language for data access Visual Studio (C#)

Language-Integrated Query (LINQ)

- .NET Language-Integrated Query defines a set of general purpose standard query operators that allow traversal, filter, and projection operations to be expressed in a direct yet declarative way in any .NET-based programming language. The standard query operators allow queries to be applied to any IEnumerable<T>-based information source.

- The accepted expressions or patterns are similar to lambda expressions in functional programming languages such as Haskell, OCaML, o F#.

- The use of expressions that incorporate functions or lambda expressions is common in most current programming languages (e.g. Java 8 admits lambda expressions to access ojects or in the code associated to graphical UI controls).

From LINQ to SQL

- We will use LINQ to access in a transparent way data stored in relational databases.
- The LINQ expressions are internally converted into SQL expressions.
- In this way we do not have to deal with specific DB architectural or connection details.

LINQ Syntax

- A LINQ expression has three parts:
  1. Obtain data source

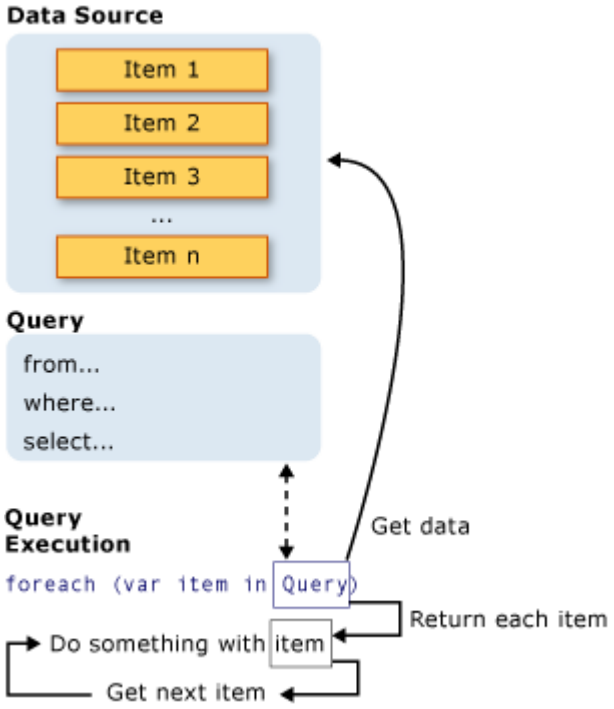
```
Ienumerable<BranOffice> bos = dbcontext.offices
```

  2. Create query

```
var offices_query =
    from office in bos
    where office.Id == 1
    select office;
```

  3. Run the query

```
foreach (BranchOffice office in offices_query)
    Console.WriteLine(office.Id,office.address);
```



Lambda expressions in LINQ

- LINQ allows an abbreviated form for expressions (*method-based)*. This form uses lambda expressions as in functional languages.
- Code 1:

```
var offices_query =
    from office in dbcontext.offices
    where office.Id == 1
    select office;
foreach (BranchOffice office in offices_query)
    Console.WriteLine(office.Id,office.address);
```

- Code 2

```
var offices_query = dbcontext.offices.Where(office => office.Id == 1)
foreach (BranchOffice office in offices_query)
          Console.WriteLine(office.Id,office.address);
```

Where in LINQ

- The Where method in a LINQ expression returns an IEnumerable object.

- This interface has many useful methods, e.g. OrderBy, Distinct, FirstOrDefault, etc.

LINQ in C#

- LINQ is integrated in different aspects of C#:
  - Queries, as shown before
  - Variables without any declared type.

```
var number = 5;
var name = "Virginia";
var query = from str in stringArray
            where str[0] == 'm'
            select str;
```

  - Initializers of objects and collections

```
Customer cust = new Customer { Name = "Mike", Phone = "555-1212" };
```

  - Anonymous types

```
select new {name = cust.Name, phone = cust.Phone};
```

  - Methods extensions
  - Lambda Expressions
  - Properties

```
public string Name {get; set;}
```

Conclusions

- LINQ transforms a query into a first-class language element in C#
- LINQ expressions used to query objects from the DBContext