



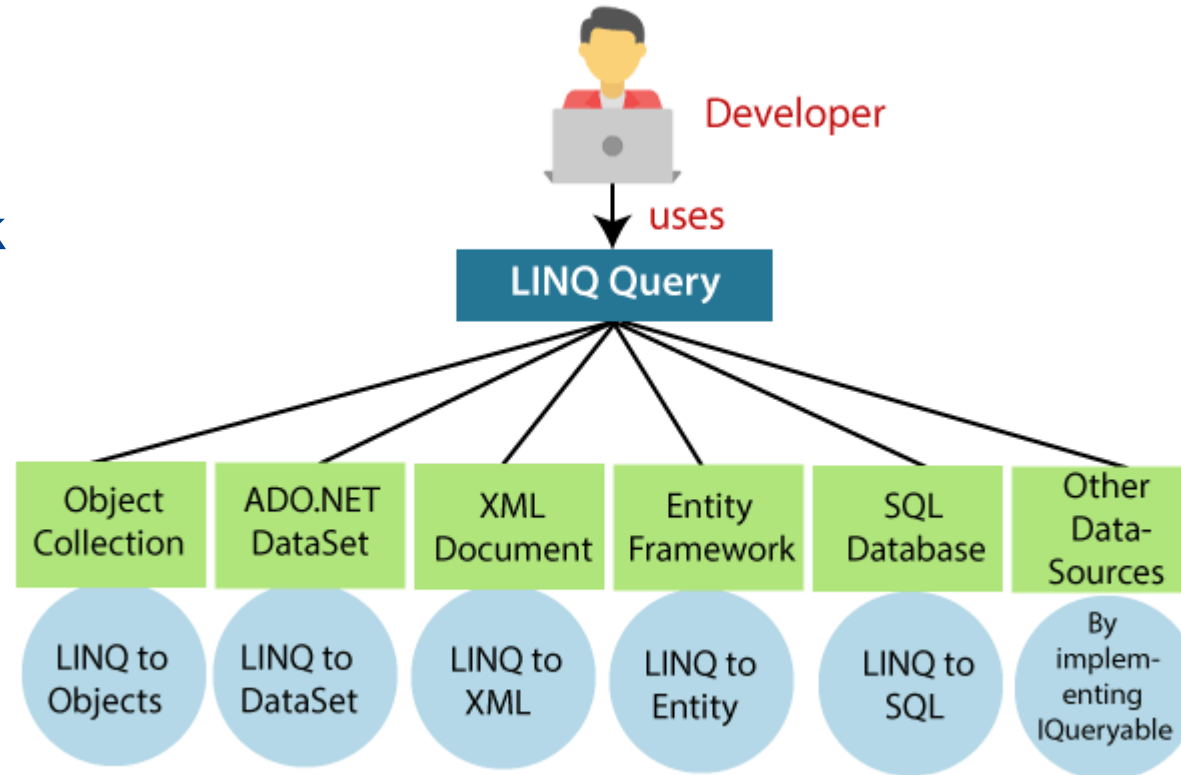
PV178 – Lab06

The background of the image is a stylized world map divided into four quadrants by color: red (top-left), blue (top-right), yellow (bottom-left), and green (bottom-right). The map is rendered in a low-poly, blocky style. The word "Kahoot!" is written across the center in a large, white, rounded, sans-serif font. The exclamation point is notably larger than the other characters.

Kahoot!

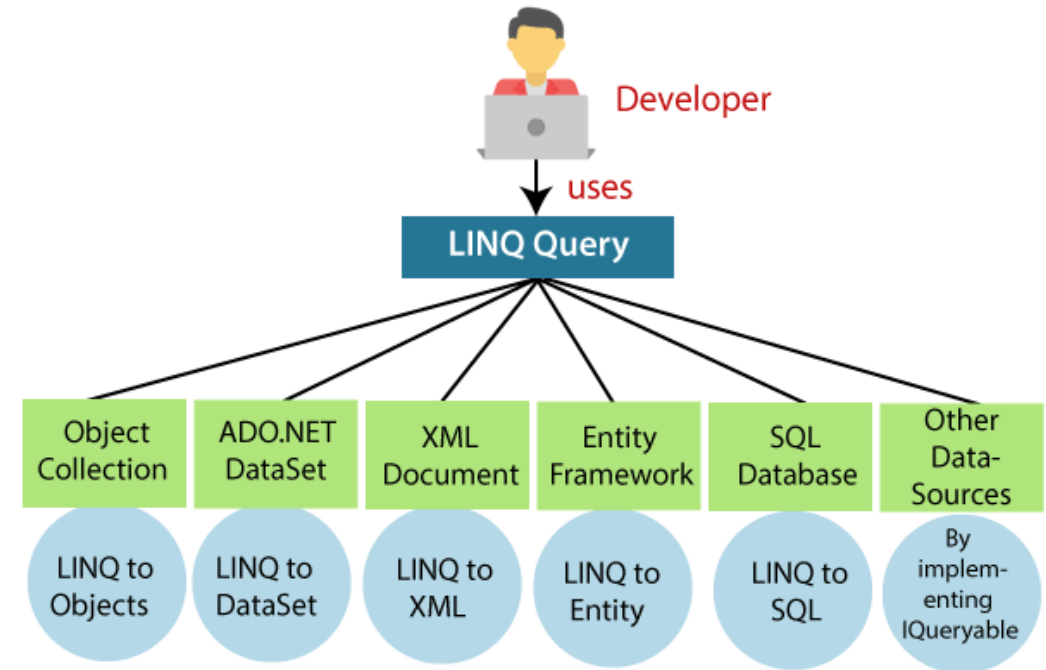
Agenda

- LINQ
- Odložené zpracování
- LINQ a Entity Framework



LINQ

- Language **I**ntegrated **Q**uery
- Jednotný přístup k datům
- Data pro LINQ mohou být v:
 - databázi
 - XML
 - kolekci ...
- Zjednodušení a zpřehlednění kódu
- Soubor rozšiřujících metod
- Dva ekvivalentní typy syntaxe (*query* a **method**)



Query syntaxe

- Připomíná SQL zápis
- Dnes méně používaný
- Méně přehledný (můj osobní názor)

```
var ints = new List<int>{ 4, 5, 85, 67, 87, 98, 150, 170}
```

```
var intsBiggerThan15 = from i in ints  
                        where i > 15  
                        select i;
```

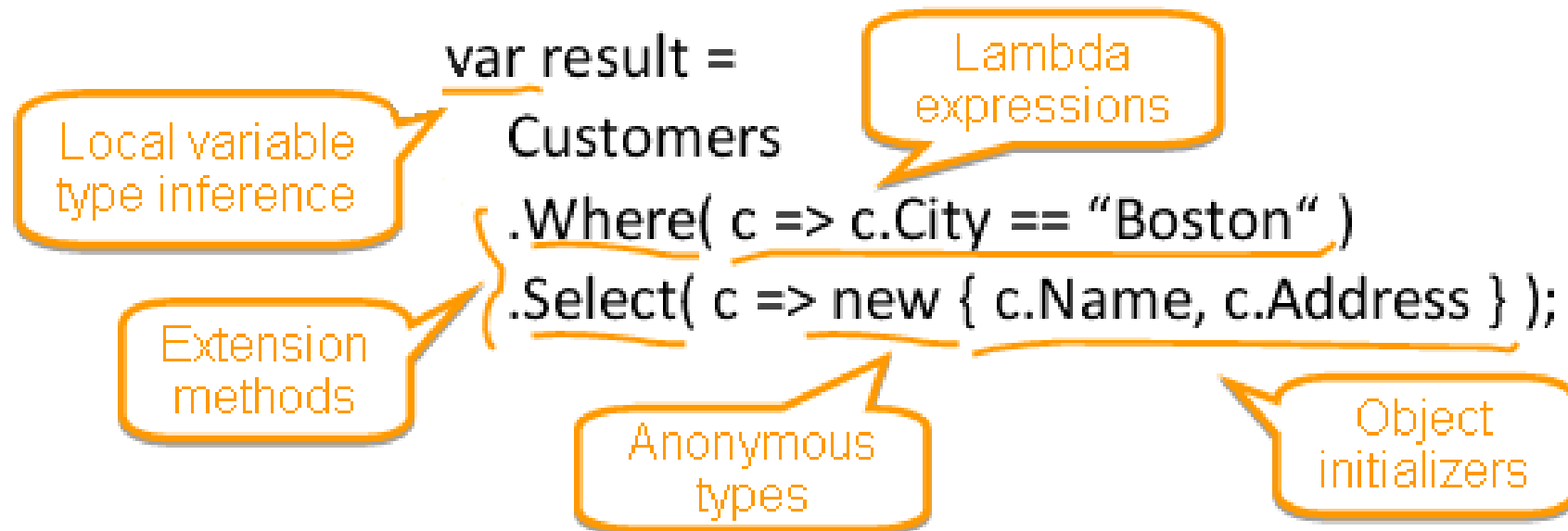
Method syntaxe

- Více používaná
- Kratší zápisy
- Používá tečkovou notaci

```
var ints = new List<int>{ 4, 5, 85, 67, 87, 98, 150, 170}
```

```
var intsBiggerThan15 = ints.Where(i => i > 15);
```

LINQ - filozofie



Odložené zpracování

- Dotazy LINQ jsou prováděny s odloženým spuštěním
- Dotaz je spuštěn až když je k němu přistoupeno – optimalizace výkonu
- Uložení dotazu do proměnné spuštění nevyvolá!
- Pozor na výrazy, které nutí vyhodnotit dotaz okamžitě (ToList, ToArray ...)

Locals	
Name	Value
▸ this	{LINQ.LinqDemo}
▾ filteredInts1	{System.Linq.Enumerable.WhereListIterator<int>}
Current	0
System.Collections.IEnumerator.Current	0
current	0
enumerator	{System.Collections.Generic.List<int>.Enumerator}
predicate	{Method = {Boolean <TestLinq>b_1_0(Int32)}}
source	Count = 8
state	0
threadId	1
Results View	Expanding the Results View will enumerate the IEnumerable
filteredInts2	null

Odložené zpracování – příklad

```
var ints = new List<int>{ 4, 5, 85, 67, 87, 98, 150, 170};  
IEnumerable<int> result = ints.Where(i => i > 15);
```

```
1 result.ToList() // Dotaz se spustí až teď !!!
```

```
2 if (false)  
  {  
    result.ToList() // Dotaz se nespustí nikdy - optimalizace  
  }
```

IEnumerable a vyhodnocení dotazu 1

```
var ints = new List<int>{ 4, 5, 85, 67, 87, 98, 150, 170};
```

```
var result = ints.Where(i => i > 15)
                .Select(i => i + 10)
                .Select(i => i - 15)
                .ToList(); // čekání až se zpracují všechny prvky
```

85

5

4

Where(i => i > 15)

Select(i => i + 10)

Select(i => i - 15)

result

IEnumerable a vyhodnocení dotazu 2 – takto NE!

- Pozor, zpomalení se projeví až u velké kolekce (např.: až v produkci)

```
var ints = new List<int>{ 4, 5, 85, 67, 87, 98, 150, 170};
```

```
var result = ints.Where(i => i > 15)
    .ToList() // čekání až se zpracují všechny prvky
    .Select(i => i + 10)
    .ToList() // čekání až se zpracují všechny prvky
    .Select(i => i - 15)
    .ToList(); // čekání až se zpracují všechny prvky
```

85
5
4

Where(i => i > 15)

Select(i => i + 10)

Select(i => i - 15)

result

ToList()

ToList()

IEnumerable a vyhodnocení dotazu 3 – takto NE!

- Pozor, zpomalení se projeví až u velké kolekce (např.: až v produkci)

```
var ints = new List<int>(100000){ 4, 5, 85, 67, 87, 98, 150, 170 ...};
```

```
// Takto NE!
```

```
var result = ints.Where(i => i > 15)
                .ToList(); // Nutí k vyhodnocení
```

Where se provede pro celou kolekci

```
var first = result.First();
```

(v tomto případě 100000x)

```
// Opraveno
```

```
var result = ints.Where(i => i > 15)
```

```
var first = result.First();
```

Where se provádí pouze než je nalezen prvek větší než 15

(v tomto případě 3x)

```
// Úplně nejlépe
```

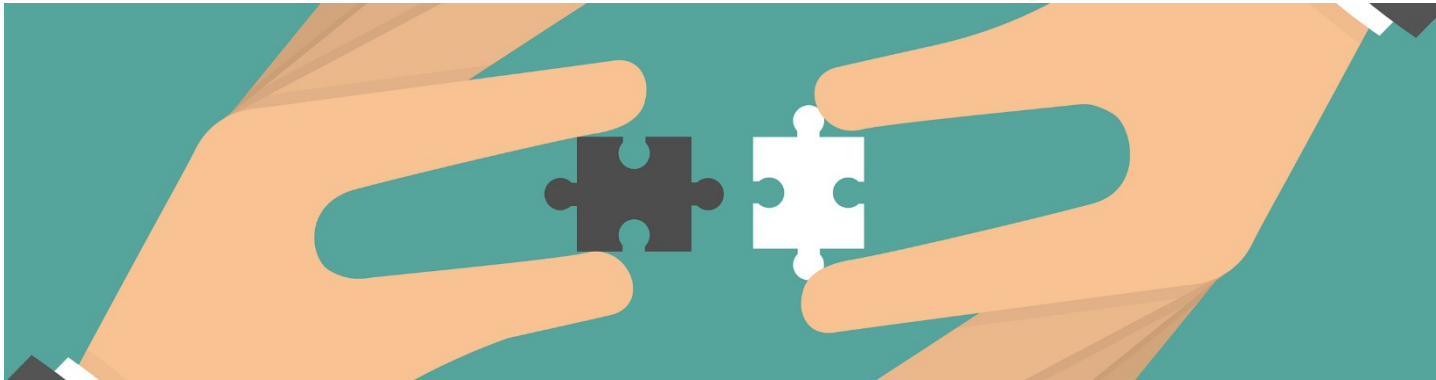
```
var first = ints.First(i => i > 15)
```

LINQ poskytovatelé (providers)

- Provádí překlad LINQ dotazu na pro konkrétní platformu
- **LINQ to Objects:**
 - slouží k práci s libovolnou kolekcí, která implementuje rozhraní `IEnumerable<T>`
- **LINQ to SQL:**
 - mapuje příkazy LINQ na dotazy SQL a umožňuje práci s MS SQL Server
- **LINQ to DataSet:**
 - používá technologii ADO.NET pro komunikaci s databázemi
- **LINQ to XML:**
 - umožňuje práci se soubory XML

LINQ to Objects

- Definován v jmenném prostoru System.Linq
- Pro kolekce implementující IEnumerable



Samostatná práce

1. Rebuild solution
2. Seznamte se s DataLoader projektem (načítá data, definuje typy apod.)
3. Nastavte si LINQ projekt jako Start Up Project
4. Pro získávání dat používejte statickou třídu DataContext
5. Hurá do LINQu (Task01 ...)

