Lekce 5

LINQ

LINQ

- LINQ (Language INtegrated Query) je **univerzalni dotazovaci jazyk** na platforme .NET
- Lze jím provádět základní datovou analýzu objektů v paměti počítače, v databázi, v souborech na disku, a dalších.
- Pro práci s LINQ je nutné do záhlaví programu přidat

```
using System.Linq;
```

Jak fungují dotazy

- podobně jako u foreach **pracujeme s každým prvkem z kolekce**, ten si označíme klíčovým slovem from
- in označuje příslušnou kolekci
- selekce: za where následuje podmínka
- projekce: klíčovým slovem select můžeme označit, jestli nás zajímá celý výsledný prvek z kolekce, nebo jen jeho část

Jak fungují dotazy

```
var zoo = new Zvire[] { lev1, lev2, slon, opice, had1, had2, had3 };
//pole objektů třídy Zvire, každý objekt má vlastnosti Jmeno a Vaha

var dotaz = from z in zoo
            where (z.Jmeno.StartsWith("L") && z.Vaha>150)
            select z.Jmeno;
```

Jak vypsat výsledek

- Pokud nám stačí zpracování pomocí foreach, není nutné převádět výsledek na kolekci.

```
foreach (var z in dotaz)
        {
            Console.WriteLine(z);
        }
//vypíše to, co bylo definováno v select
```

Funkce konverze výsledku

- Pokud potřebujeme výsledek uložit do pole nebo do Listu, použijeme příslušné funkce ToArray resp. ToList přímo v dotazu

```
var dotaz = (from z in zoo
             where (z.Jmeno.StartsWith("L") && z.Vaha>150)
             select z.Jmeno).ToArray();
```

Dotaz pomocí lambda výrazů

- Klíčová slova Where a Select jsou zde funkce, které lze spustit nad zkoumanou kolekcí zoo.
- Proměnná z opět značí každý jednotlivý prvek v kolekci
- Operátor => ukazuje, jakým způsobem se v dané funkci prvek použije

```
var dotaz2 = zoo.Where(z => z.Vaha > 200)
                .Where(z => z.Jmeno.StartsWith("L"))
                .Select(z => z.Jmeno);
```

Funkce - Count

- Spočítá všechny prvky v kolekci

  ```
  Console.WriteLine(zoo.Count());
  ```

- Spočítá prvky, které splňují podmínku (není potřeba použít Where)

  ```
  Console.WriteLine(zoo.Count(z => z.Vaha > 300));
  ```

Funkce výběru

- Funkce First vybere první prvek z kolekce, Last poslední
- Pokud v kolekci není žádný prvek, překladač vrátí chybu
- Varianty FirstOrDefault a LastOrDefault vrací v takovém případě null

```
Console.WriteLine(zoo.FirstOrDefault().Jmeno);
```

Funkce vyhodnocení

- Splňují všechny prvky dané kritérium?

  ```
  Console.WriteLine(zoo.All(z => z.Vaha > 200));
  ```

- Splňuje některý prvek dané kritérium?

  ```
  Console.WriteLine(zoo.Any(z => z.Vaha > 200));
  ```

- Vyber minimum/maximum - funkce Min/Max

  ```
  Console.WriteLine(zoo.Min(z => z.Vaha));
  ```

- Vypočítej průměr/součet - funkce Average/Sum

  ```
  Console.WriteLine(zoo.Average(z => z.Vaha));
  ```

Funkce řazení

- Funkce OrderBy seřadí prvky vzestupně

```
foreach (var z in zoo.OrderBy(z => z.Jmeno))
        {
            Console.WriteLine(z.Jmeno);
        }
```

- Funkce OrderByDescending seřadí prvky sestupně (od největšího k nejmenšímu)

Funkce seskupování

- Funkce GroupBy vytvoří kolekci jejíž prvky obsahují klíč a kolekci k němu příslušných (seskupených) prvků
- Vlastnost Key obsahuje klíč dané skupiny - v tomto příkladu počáteční písmeno jména zvířete (z.Jmeno[0]):

```
foreach (var skupina in zoo.GroupBy(z => z.Jmeno[0]))
        {
            var jmenaZeSkupiny = skupina.Select(j => j.Jmeno);
            Console.WriteLine(skupina.Key + " - " + string.Join(", ", jmenaZeSkupiny));
        }
```

Funkce speciální selekce a projekce

```
var cisla = new int[] {2, 3, 4, 5, 6, 7, 8, 9, 10, 7, 3}
```

- Funkce Distinct vrátí každou hodnotu pouze jednou. Lze použít jen na jednoduché datové typy.
  - `Console.WriteLine(string.Join(", ", cisla.Distinct()));`
- Funkce Reverse otočí pořadí prvků v kolekci.
  - `cisla.Reverse();`
- Funkce Skip a Take se nejlépe použijí při stránkování a vzorkování
- Skip přeskočí na daný index a Take vyjme daný počet prvků počínaje zadaným indexem
  - `Console.WriteLine(string.Join(", ", cisla.Skip(3).Take(3)));`

Funkce speciální selekce a projekce

- Funkce SelectMany se používá na zploštění hierarchie - tj. na získání hodnot z kolekcí v kolekcích

```
var listZoo = new List<Zoo> { zoo1, zoo2 };
//zoo1 a zoo2 jsou objekty třídy Zoo, který každý obsahují kolekci Zvirata
var jmenaZvirat = listZoo.SelectMany(v => v.Zvirata).Select(v => v.Jmeno);
//výsledkem jsou jména všech zvířat ve všech zoo v listZoo
```