

## LEKCIA 6 – LINQ

[LINQ](#) je súčasť .NET frameworku určená pre jednotné prístupovanie k dátam a ich spracovanie deklaratívnym a funkcionálnym spôsobom.

[LINQ](#) je rozšírením syntaxe jazyka C# o natívne písanie dotazov v programovom kóde, nie nepodobným jazyku SQL.

Zároveň je [LINQ](#) sadou nástrojov pre pracovanie s dátami vo forme objektov (kolekcií), s dátami s relačných databáz alebo dátami reprezentovanými ako XML. Ďalšie zdroje dát je možné využívať vďaka možnosti rozšíriteľnosti, ktorá je základným konceptom [LINQ](#).

Výhody používania [LINQ](#) spočívajú najmä v zjednodušení a sprehľadnení kódu. V oslobodení od nástrojov špecifických pre daný zdroj dát (XPath, SQL), ktoré nemajú podporu priamo v syntaxi C#, a teda Visual Studio k nim neposkytuje typovú kontrolu či Intellisense (automatické dokončovania). Veľkou výhodou je tiež jednotná forma prístupu a práce (transparentnosť) s týmito rôzne reprezentovanými dátami.

### LANGUAGE ENHANCEMENTS

Môžnosť sprehľadniť prácu s dátami v jazyku C# za pomoci LINQ je dosiahnutá pomocou využitia nasledovných moderných nielen syntaktických „vylepšení“ :

- [Implicitly Typed Local Variables](#)
- [Object and Collection Initializers](#)
- [Lambda Expressions](#)
- [Extension Methods](#)
- [Anonymous Types](#)

### ZÁKLADNÉ PRVKY LINQ

#### ENUMERABLE

Objektové dáta, ktoré chceme spracovávať pomocou LINQ musia byť reprezentované vo forme *sekvencie*, teda prístupné pomocou [iterátoru](#) /generátoru.

#### LINQ PROVIDERS

Preklad z LINQ dotazu na dotaz do konkrétneho typu dát je realizovaný pomocou LINQ providers. Najpoužívanejšie „príchute“ sú :

- [LINQ to Objects](#)
- [LINQ to XML](#)
- [LINQ to SQL](#)
- [LINQ to Entities](#)

---

## STANDARD QUERY OPERATORS

[Množina základných operácií](#) nad sekvenciami dát, ktorú implementujú LINQ providers. Sú prístupné ako Extension metódy. Ukážka a výčet týchto operátorov je výborne demonštrovaný pomocou stiahnuteľných [101 LINQ Samples](#).

---

## DEFERRED EXECUTION

Je názov pre jav, že samotná transformácia dát (vykonanie LINQ dotazu) sa vykonáva až v momente, keď k výslednej sekvencii pristupujeme (používame ju). Prvky sú postupne jeden po druhom transformované reťazcom operátorov v určenom poradí. Nazývané tiež pipeline pattern.

---

## QUERY EXPRESSIONS

[LINQ Query Expressions](#) je syntax (from.. in.. where..) ktorá sprehľadňuje používanie niektorých *standard query operators* pomocou natívneho zápisu podobnému jazyku SQL.

---

## EXPRESSION TREES

Pre dáta ktoré nie sú reprezentované vo forme Enumerable sekvencie, napríklad dáta v relačnej databáze, je dôležitá možnosť reprezentovať LINQ dotaz (vrátane lamda výrazov) vo forme, ktorú môže LINQ provider transformovať do natívneho spôsobu práce so zdrojom dát. Toto umožňujú [Expression Trees](#), ktoré reprezentujú lambda výrazi ako dáta (a nie ako funkcie). Vďaka tomu sa napríklad pri LINQ to SQL, transformuje LINQ dotaz na dotaz v jazyku T-SQL priamo, a využijú sa takmer všetky výhody použitia SQL Serveru..