

OOP

LINQ

Uvod u LINQ

Uvod

- LINQ =Language INtegrated Query
- Osnovna ideja koja stoji iza LINQ-a dolazi iz područja funkcionalnog programiranja i baza podataka (SQL)

Uvod (2)

- LINQ to (SQL, Objects, Entities,...)
- LINQ i funkcionalno programiranje
 - Sagrađeno na ideji iz programskog jezika LISP:
mapiraj ->filtriraj -> reduciraj
 - Lijeno učitavanje (engl. lazy loading)
- LINQ and SQL
 - LINQ je upitni jezik
 - LINQ fragmenti se prevode u SQL

LINQ i C#

- LINQ u C#-u može se koristiti i kao mehanizam obrade raznih kolekcija, ali i kao zamjena za SQL

Elementi LINQ-a

- **Niz (eng. Sequence)** = kolekcija tipa `IEnumerable<T>`
- **Izvor podataka (eng. Data source)** = niz koji je izvor upita (LINQ-a)
- **Upit (eng. Query)** = izraz koji evaluira podatke iz niza
- **Operator upita (eng. Query Operator)** = funkcija definirana na tipu u nizu (T)
- **LINQ provider** = implementacija upita na izvoru podataka (npr. LINQ to Objects)

LINQ arhitektura

- **Mapiranje** – primijeni funkciju na svaki element u nizu i vrati rezultatni niz

LINQ : Select

- **Filtriranje** – primijeni funkciju na svaki element u nizu i vrati samo one koji zadovoljavaju uvjet

LINQ : Where

- **Redukcija** – primjena binarne funkcije (vraća skalar) na svaki par susjednih elemenata u nizu

LINQ : Count, Min, Max, Sum, Avg

DEMO

LINQ operatori koji vraćaju niz

- `s.Select(f)` – proizvoljna element funkc.
- `s.Where(f)` – proizvoljna predikatna funkc.
- `s.Distinct()`
- `s.Take(int)`
- `s.Intersect(s2)`
- `s.Union(s2)`
- `s.Concat(s2)`
- `s.OrderBy(f)` – „key selector”

Pitanja za kraj



Hvala na pažnji!