

Temat : LINQ¹ integracja języka programowania z językiem zapytań

Historia zmian

<i>Data</i>	<i>Wersja</i>	<i>Autor</i>	<i>Opis zmian</i>
7.10.2013	- 1.0 -	Tomasz Kowalski	Utworzenie dokumentu.
14.10.2013	- 1.1 -	Tomasz Kowalski	Przeniesienie Linq to SQL do innego laboratorium.
3.10.2016	- 1.2. -	Tomasz Kowalski	Aktualizacja linku do projektu 101 samples.

¹ Langauage Integrated Queries

1. Cel laboratorium

Głównym celem laboratoriów jest zapoznanie się z tworzeniem zapytań w języku LINQ, który jest elementem platformy .NET, przy użyciu składni Query Expressions.

2. Materiały pomocnicze

Dokumentacja MSDN:

<http://msdn.microsoft.com/pl-pl/library/system.linq.aspx>

101 przykładów:

<http://code.msdn.microsoft.com/LINQ-Sample-Queries-13a42a54>

Ćwiczenia z rozwiązaniami:

http://download.microsoft.com/download/0/e/2/0e255cf3-b11f-44cb-b42c-7d55ed7b556c/linq_hands_on_lab.docx

3. Laboratorium:

1. Pobierz projekt zawierający przykładowy zastosowania języka LINQ w kontekście kolekcji obiektów, zbiorów XML i danych przechowywanych w relacyjnej bazie:
<http://coach.kis.p.lodz.pl/LINQ-SampleQueries.zip>
2. Uruchom Microsoft Visual Studio i otwórz w/w projekt.
3. Uruchom oraz przetestuj i zapoznaj się z projektem. W szczególności zwróć uwagę na zadania należące do następujących działów:
 1. LINQ to Objects – (plik LinqSamples)
 2. LINQ to XML – (plik LinqToXMLSamples)

Zwróć uwagę na sposób wprowadzania za pomocą anotacji przykładów (grupowanie i nazywanie).

3.1. Zapytania w formie Query Expression

Napisz zapytania rozwiązujące następujące problemy. Umieść je w odpowiednich działach tworząc w każdym nową podgrupę o nazwie *lab1*.

1. Dla kolekcji produktów w LINQ to Objects podaj:
 1. nazwy i ceny jednostkowe wszystkich produktów,
 2. nazwy produktów, które są na stanie, kosztują mniej niż 10 i należą do kategorii Seafood.
 3. produkty, których cena jednostkowa, jest równa cenie produktu o nazwie Ikura.
 4. średnią cenę produktów w każdej kategorii (użyj **group by**).
2. Dla kolekcji klientów w LINQ to XML podaj:
 1. nazwy i miasta klientów nie posiadających nr faksu,
 2. miasta wraz z liczbą klientów w kolejności malejącej.
3. * W dziale LINQ to Objects napisz zapytanie zwracające liczby pierwsze z zakresu od 1 do 888. (WSKAZÓWKA: użyj metody Enumerable.Range(...))

3.2. Proste testowanie wydajności zapytań

1. W nowym pliku cs zaimplementuj mechanizm ułatwiający testowanie wydajności zapytań. Może być, to np. statyczna metoda, przyjmująca jako argument (m.in.) zapytanie, które ma być przetestowane.

Do pomiaru czasu możesz wykorzystać klasę Stopwatch.

2. Zapewnij możliwość określenia liczby powtórzeń testu, a jako ostateczny wynik zwróć medianę z wszystkich czasów (użyj do tego LINQ).
3. Zweryfikuj działanie opracowanego rozwiązania przynajmniej na jednym złożonym zapytaniu w kontekście LINQ to Objects, LINQ to XML.