

Platforma .NET
Zajęcia laboratoryjne

Ćwiczenie nr 4

Zasoby pomocne w trakcie realizacji zadań

<https://code.msdn.microsoft.com/101-LINQ-Samples-3fb9811b>

Zad 1.

Napisz zapytanie Linq, które pobiera z tablicy wartości podzielne przez 2

```
int[] n1 = {  
    1, 3, -2, -4, -7, -3, -8, 12, 19, 6, 9, 10, 14  
};
```

Zad 2.

Napisz zapytanie Linq, które pobiera liczby podzielne przez 2

```
int[] n1 = {  
    1, 3, -2, -4, -7, -3, -8, 12, 19, 6, 9, 10, 14  
};
```

Zad 3.

Napisz zapytanie Linq, które pobiera liczby z przedziału >0 i <12

```
var arr1 = new[] { 3, 9, 2, 8, 6, 5 };
```

Zad 4.

Przy pomocy Linq Znajdź liczby które podniesione do drugiej potęgi dają wartość >20

Zad 5.

Mając daną tablicę

```
int[] arr1 = new int[] { 5, 9, 1, 2, 3, 7, 5, 6, 7, 3, 7, 6, 8, 5, 4,  
9, 6, 2 };
```

wypisz wszystkie liczby oraz częstotliwość ich występowania

Wykorzystaj group by / Key / Count

Zad 6.

Mając dany ciąg znaków

```
Var str = „abeddwkkecjjeksoiekcllkenndkwel”
```

Wypisz wszystkie litery występujące w tekście oraz częstotliwość ich występowania

Wykorzystaj group by / Key / Count

Zad 7.

Mając daną tablicę

```
string[] months = { "January", "February", "March", "April", "May",  
"June", "July", "August", "September", "October", "November",  
"December" };
```

wypisz wszystkie dni tygodnia

Zad 8.

Mając daną tablicę

```
int[] nums = new int[] { 5, 1, 9, 2, 3, 7, 4, 5, 6, 8, 7, 6, 3, 4, 5,  
2 };
```

Wypisz wszystkie unikalne wartości, sumę poszczególnych wartości (iloczyn wartości przez liczbę jej wystąpień), liczbę wystąpień danej wartości

Wykorzystaj group by/ Key / Sum / Count

Zad 9.

Napisz aplikację potrafiącą znaleźć łańcuchy znaków rozpoczynające się i kończące zadanymi przez użytkownika znakami

```
string[] cities =  
{  
    "ROME", "LONDON", "NAIROBI", "CALIFORNIA", "ZURICH", "NEW  
DELHI", "AMSTERDAM", "ABU DHABI", "PARIS"  
};
```

Zad 10.

Stwórz listę liczb całkowitych (pobierz wartości z klawiatury), wypisz elementy większe od wartości zadanej przez użytkownika (pobierz wartość z klawiatury). Wykorzystaj metodę FindAll

Zad 11.

Napisz program przechowujący tablicę liczb (`List<int>`) i wyświetlających `n` ostatnich wartości (`n` – podawane przez użytkownika)

Wykorzystaj metodę `Take`

Zad 12.

Napisz program przechowujący tablicę liczb (`List<int>`) i wyświetlających `n` największych wartości (`n` – podawane przez użytkownika)

Wykorzystaj metody `Sort` / `Revers` / `Take`

Zad 13.

Napisz program wykrywających słowa pisane wielkimi literami w łańcuchu znakowym

Wykorzystaj `Split`, `Where`, `Equals`, `ToUpper`

Zad 14.

Napisz program konwertujący tablicę typu `string` na `string`

Wykorzystaj `Join`

Zad 15.

Mając daną następującą klasę

```
public class Students
{
    public string StudentName { get; set; }
    public int GroupPoint { get; set; }
    public int StudentId { get; set; }

    public List<Students> GtStuRec()
    {
        List<Students> stulist = new List<Students>();
        stulist.Add(new Students { StudentId = 1, StudentName = "
A ", GroupPoint = 800 });
        stulist.Add(new Students { StudentId = 2, StudentName = "
B", GroupPoint = 458 });
    }
}
```

```

        stulist.Add(new Students { StudentId = 3, StudentName =
"C", GroupPoint = 900 });
        stulist.Add(new Students { StudentId = 4, StudentName =
"D", GroupPoint = 900 });
        stulist.Add(new Students { StudentId = 5, StudentName =
"E", GroupPoint = 458 });
        stulist.Add(new Students { StudentId = 6, StudentName =
"F", GroupPoint = 700 });
        stulist.Add(new Students { StudentId = 7, StudentName =
"G", GroupPoint = 750 });
        stulist.Add(new Students { StudentId = 8, StudentName =
"H", GroupPoint = 700 });
        stulist.Add(new Students { StudentId = 9, StudentName =
"I", GroupPoint = 597 });
        stulist.Add(new Students { StudentId = 10, StudentName =
"J", GroupPoint = 750 });
        return stulist;
    }
}

```

Znajdź n studentów, którzy uzyskali najwyższy wynik

Zad 16.

Mając daną tablicę `string[] arr1 = { "a.erc", "b.txt", "c.ldd", "d.pdf", "e.PDF", "a.pdf", "b.xml", "z.txt", "zzz.doc" };`

Napisz program, grupujący poszczególne rozszerzenia i zliczający wystąpienie poszczególnych rozszerzeń

Zad 17.

Napisz program usuwający dane z listy na podstawie przekazanych wartości

Zad 18.

Napisz program generujący iloczyn kartezjański dwóch zbiorów

```

char[] charset1 = { 'A', 'B', 'C', 'D' };
int[] numset1 = { 1, 2, 3, 4 };

```

Zad 19

Napisz program generujący złącze typu INNER JOIN pomiędzy dwoma listami

```
List<Item_mast> itemlist = new List<Item_mast>
{
    new Item_mast { Id = 1, Descr = "A  " },
    new Item_mast { Id = 2, Descr = "B" },
    new Item_mast { Id = 3, Descr = "C" },
    new Item_mast { Id = 4, Descr = "D" },
    new Item_mast { Id = 5, Descr = "E" }
};

List<Purchase> purchlist = new List<Purchase>
{
    new Purchase { No=100, Id = 3, Qty = 55 },
    new Purchase { No =101, Id = 2, Qty = 44 },
    new Purchase { No =102, Id = 3, Qty = 555 },
    new Purchase { No =103, Id = 4, Qty = 33 },
    new Purchase { No =104, Id = 3, Qty = 33 },
    new Purchase { No =105, Id = 4, Qty = 44 },
    new Purchase { No =106, Id = 1, Qty = 343 }
};
```

Zad 19.

Napisz program generujący złącze typu LEFT JOIN pomiędzy dwoma listami

Zad 20.

Napisz program generujący złącze typu RIGHT JOIN pomiędzy dwoma listami