

Linq: Language Integrated Query + method syntax

Csorba Kristóf

IEnumerable

Felsorolható... semmi több.

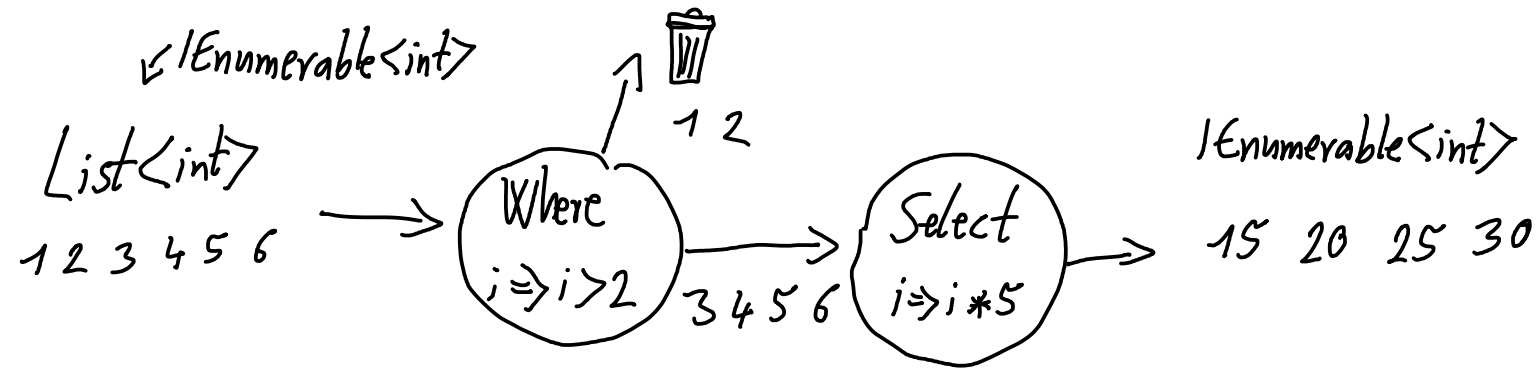
- Egyesével olvasható
- Csak sorban

Nincs

- Darabszám
- Ugrás

Implementálja például a tömb, lista, dictionary...

IEnumerable lán



Query vs method syntax

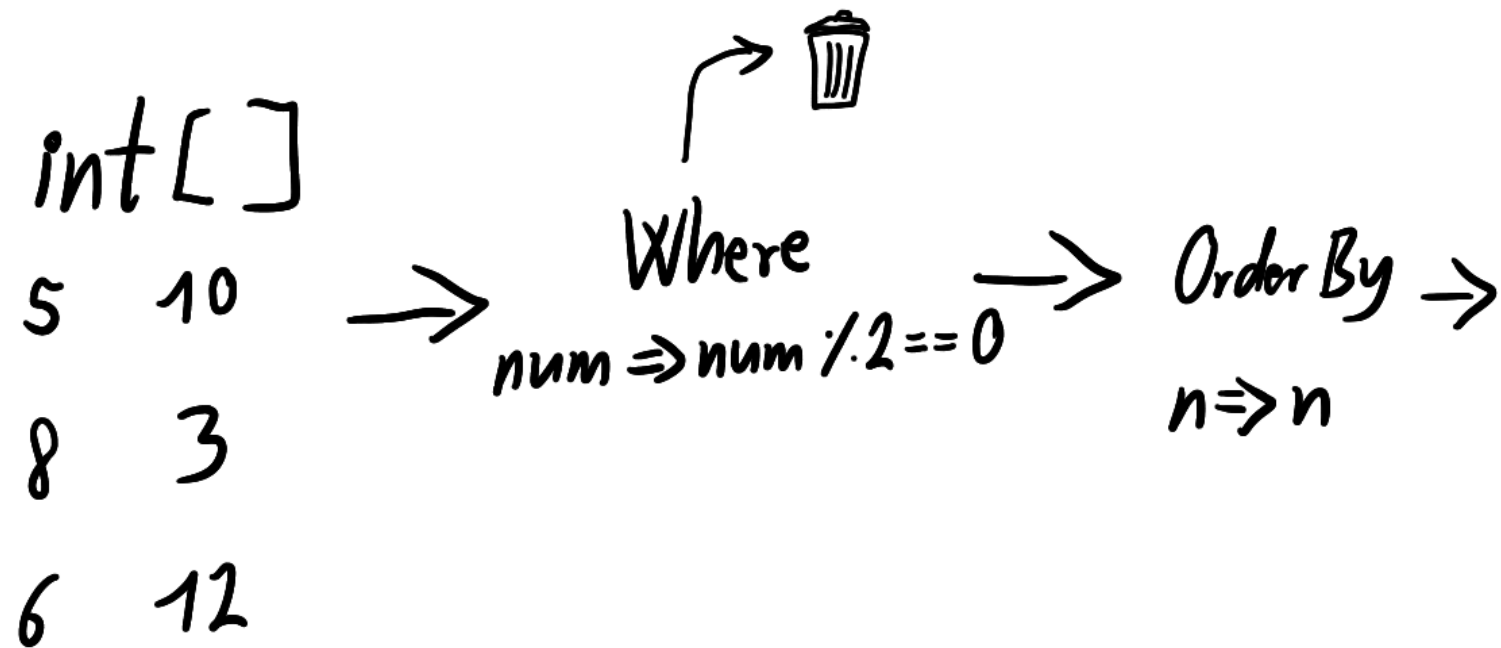
```
int[] numbers = { 5, 10, 8, 3, 6, 12 };
```

```
//Query syntax:
```

```
IEnumerable<int> numQuery1 =  
    from num in numbers  
    where num % 2 == 0  
    orderby num  
    select num;
```

```
//Method syntax:
```

```
IEnumerable<int> numQuery2 =  
    numbers.Where(num => num % 2 == 0)  
    .OrderBy(n => n);
```

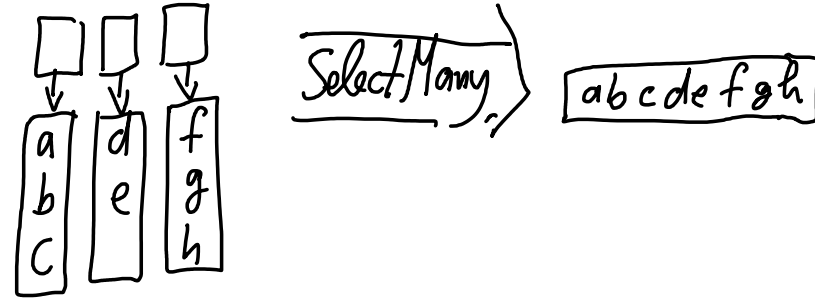


Count, lambda

```
// Check for draw
// EVIP: Linq Count with condition
Winner = (counts.Count(c => c==maxFieldCountPerPlayer) == 1)
    ? playerWithMaxFields : 0;
```

AttaxxPlus\AttaxxPlus\Model\GameBase.cs

SelectMany



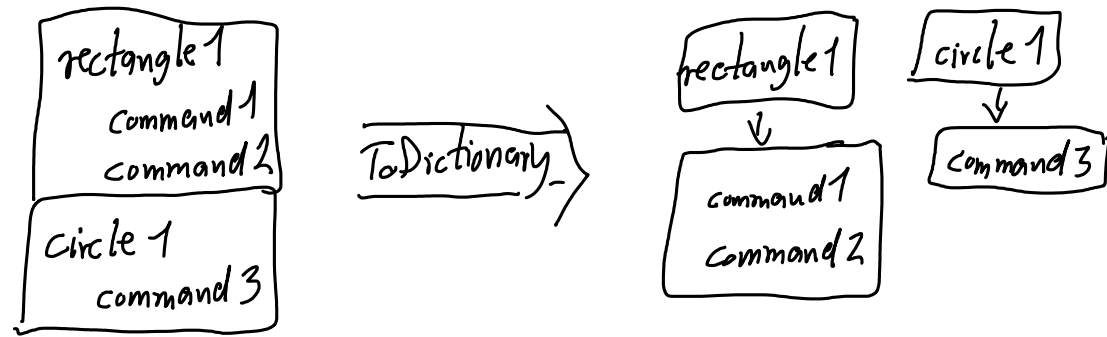
```
public int GetLastFrameIndex()
{
    // EVIP: SelectMany to concatenate arrays in dictionary values.
    var allCommands = Commands.SelectMany(kvp => kvp.Value.Value).ToArray();
    public readonly Dictionary<XElement, AnimationCommand[]> Commands;

    // EVIP: local function having access to local variables of parent method
    void registerFrameIndex(int? frameIndex)
    {
        if (frameIndex.HasValue && lastFrameIndex < frameIndex)
            lastFrameIndex = frameIndex.Value;
    }

    foreach(var cmd in allCommands)
    {
        // Without nested method, registerFrameIndex would need a
        // class level attribute to store the maximum (or a ref parameter).
        registerFrameIndex(cmd.FirstFrame);
    }
}
```

svg2pptx\svg2pptx\SvgWrapper.cs

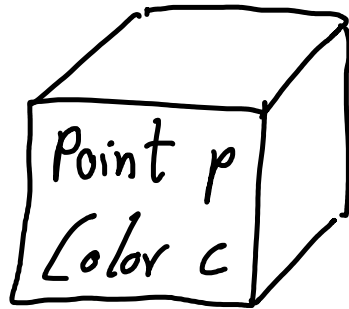
ToDictionary



```
private Dictionary<XElement, AnimationCommand[]> GetCommandDictionary()  
{  
    // EVIP: Linq ToDictionary  
    return GetEntitiesWithInkscapeLabel(root).ToDictionary(  
        e => e,  
        e => AnimationCommand.Parse(e.InkscapeLabel()).ToArray());  
  
    // Note: a longer form of the above code:  
    //var result = new Dictionary<XElement, AnimationCommand[]>();  
    //foreach (var element in GetEntitiesWithInkscapeLabel(root))  
    //    result.Add(element, AnimationCommand  
    //        .Parse(element.InkscapeLabel()).ToArray());  
    //return result;  
}
```


Tuple, expression syntax

```
// EVIP: Tuple as input and return value  
private (Point, Color) Index2HSV((Point p, int index) input)  
    => (input.p, HsvRgbConverter.Hue2RGB(Math.Min(input.index, 255)));
```



Mandelbrot\MandelbrotCommon\Mandelbrot.cs

Linq, típusok követése...

```
protected override void Render(WritableBitmap image)
{
    // EVIP: Linq to render pixel colors in parallel.
    image.GetAllPixelLocations().AsParallel()
        .Select(p => ToScaledComplex(p))
        .Select(c => GetGridColor(c))
        .SetPixels(image);
}
```

```
public static IEnumerable<Point>
    GetAllPixelLocations(this WritableBitmap image)
```

```
protected (Point p, Complex c) ToScaledComplex(Point p)
```

```
private (Point p, Color c) GetGridColor((Point p, Complex c) input)
```

```
public static void SetPixels(this IEnumerable<(Point, Color)> values,
    WritableBitmap image)
```

Mandelbrot\MandelbrotCommon\TestGridImageRenderer.cs

Típusok követése...

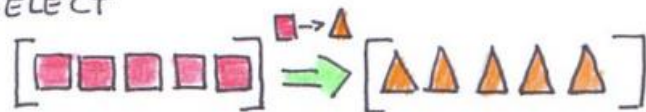
```
protected override void Render(WritableBitmap image)
{
    // EVIP: parallel rendering of Mandelbrot set with
    // Linq (some elements of functional programming)
    image.GetAllPixelLocations().AsParallel()
        .Select(p => ToScaledComplex(p))
        .Select(c => GetMandelbrotDivergenceIndex(c))
        .Select(i => Index2HSV(i))
        .SetPixels(image);
}
```

```
public static IEnumerable<Point> GetAllPixelLocations(this WritableBitmap image)
protected (Point p, Complex c) ToScaledComplex(Point p)
private (Point, int) GetMandelbrotDivergenceIndex((Point p, Complex c) input)
private (Point, Color) Index2HSV((Point p, int index) input)
public static void SetPixels(this IEnumerable<(Point, Color)> values, WritableBitmap image)
```

Mandelbrot\MandelbrotCommon\Mandelbrot.cs

EXPLAIN LINQ TO A FIVE YEAR OLD

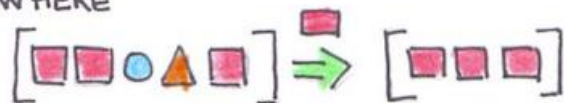
SELECT



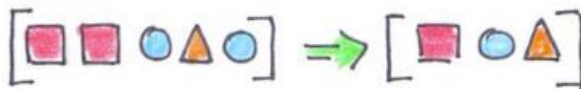
SELECT MANY



WHERE



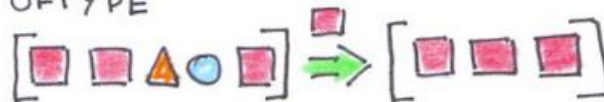
DISTINCT



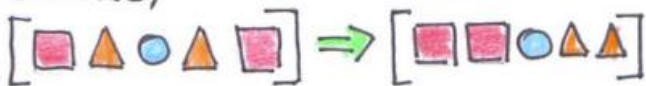
CAST



OF TYPE



ORDER BY



ORDER BY DESCENDING



REVERSE



GROUP BY



SKIP



SKIP WHILE



TAKE



TAKE WHILE



BASED ON THE ORIGINAL SYMBOLS
BY MARTIN FOWLER

© WIDECL

<https://martinfowler.com/articles/collection-pipeline/>

Az XML formátum

```
<svg>
  <g id="topLevelGroup">
    <rect style="fill:#ff00ff;fill-opacity:1.0;stroke:#000000;
      stroke-width:1;stroke-miterlimit:4;
      stroke-dasharray:none;stroke-opacity:1"
      id="rectPurple" width="60.0" height="30.0" x="60.0" y="90.0" />
    <rect style="fill:#ffffff;fill-opacity:1.0;stroke:#000000;
      stroke-width:2;stroke-miterlimit:4;
      stroke-dasharray:none;stroke-opacity:1"
      id="rectWhite" width="90.0" height="30.0" x="30.0" y="130.0" />
  </g>
</svg>
```

- Tag (nyitó, záró, önlezáró)
- Attribútum
- Tartalom

Mozgás az XML-ben (DOM)

```
<svg>
  <g id="topLevelGroup">
    <rect style="fill:#ff00ff;fill-opacity:1.0;stroke:#000000;
      stroke-width:1;stroke-miterlimit:4;
      stroke-dasharray:none;stroke-opacity:1"
      id="rectPurple" width="60.0" height="30.0" x="60.0" y="90.0" />
    <rect style="fill:#ffffff;fill-opacity:1.0;stroke:#000000;
      stroke-width:2;stroke-miterlimit:4;
      stroke-dasharray:none;stroke-opacity:1"
      id="rectWhite" width="90.0" height="30.0" x="30.0" y="130.0" />
  </g>
</svg>
```

```
root.Descendants("rect")
  .Where(r => r.Attribute("style").Value
  .Contains("stroke-width:1"));
```

XML létrehozása LinqToXml alapokon

```
// EVIP: creating Xml using Linq to Xml
var favorites = vm.Favorites
    .Select(areaViewModel => areaViewModel.Model).ToArray();
XElement root = new XElement("favorites");
var xmlElements = favorites.Select(f => new XElement("favorite",
    new XAttribute("top", f.Top), new XAttribute("bottom", f.Bottom),
    new XAttribute("left", f.Left), new XAttribute("right", f.Right)));
foreach (var element in xmlElements)
    root.Add(element);
return root;
```

```
<?xml version="1.0" encoding="utf-8"?>
<favorites>
  <favorite top="-1.2000000476837158" bottom="1.2000000476837158"
    left="-1.6000000238418579" right="0.60000002384185791" />
  <favorite top="-1.2000000476837158" bottom="-0.45943370930383542"
    left="-0.61691267546033124" right="0.061939789126450373" />
</favorites>
```

Mandelbrot\FavoriteMandelbrots\ViewModel\SaveFavoritesCommand.cs: CreateXmlTree()

XML és Linq: XElement

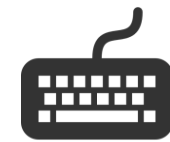
UWP fájlműveletek

```
StorageFile file = await savePicker.PickSaveFileAsync();
if (file != null)
{
    using (Stream stream = await file.OpenStreamForWriteAsync())
    {
        // EVIP: Writing objects into XML file
        XElement treeRoot = CreateXmlTree();
        await treeRoot.SaveAsync(stream, SaveOptions.None,
            new System.Threading.CancellationToken());
    }
}
```

Mandelbrot\FavoriteMandelbrots\ViewModel\AddFavoritesFromFileCommand.cs

Mandelbrot\FavoriteMandelbrots\ViewModel\SaveFavoritesCommand.cs

Linq2XmlSvgLab – rectangles1.svg



```
<g
  id="group1">
  <rect
    y="20"
    x="30"
    height="40"
    width="50"
    id="rectRed"
    style="fill:#ff0000;fill-opacity:1;stroke:#000000;stroke-width:2;stroke-
  <rect
    y="20"
    x="90"
    height="30"
    width="40"
    id="rectYellow"
    style="fill:#ffff00;fill-opacity:1;stroke:#000000;stroke-width:2;stroke-
```