

Szövegkezelés, I/O

Csorba Kristóf

Előzetes online anyagok (ismétlés)

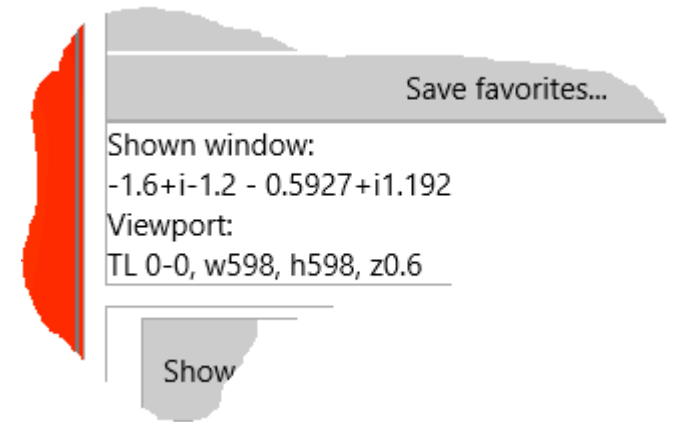
- Stream koncepció, sima szöveg és bináris írás/olvasás (stream, reader). File, írás és olvasás konzol alkalmazásnál (vs UWP!)
- HF1a kiírás: feladat, leadás módja

```
// EVIP: string interpolation  
MessageDialog m = new MessageDialog(  
    $"Game over. Winner is player {Model.Winner}");
```

E:\Projekte\evipdev\AttaxPlus\AttaxPlus\ViewModel\GameViewModel.cs

```
// EVIP: override of ToString, string interpolation
public override string ToString()
{
    return $"SPoint({X},{Y},t={T},color({Color.R},{Color.G},{Color.B}),width({LineWidth}))";
}
```

SpirographLab\SpirographLib\SPoint.cs



```
// EVIP: Getter-only property and string interpolation
```

```
public string AsString => $"{left:G4}+i{top:G4} - {right:G4}+i{bottom:G4}";
```

```
// EVIP: string interpolation
```

```
DebugViewPortDataString = $"TL {x:G4}-{y:G4}, w{w:G4}, h{h:G4}, z{zoomFactor:G4}";
```

Mandelbrot\FavoriteMandelbrots\Model\Area.cs

Mandelbrot\FavoriteMandelbrots\ViewModel\MainViewerViewModel.cs

- keresés és helyettesítés, feldarabolás és összefűzés, string és stringBuilder (ez utóbbiban remove és replace is van!)
<https://docs.microsoft.com/en-us/dotnet/api/system.text.stringbuilder?view=netcore-3.1>
- string: immutable, sealed class
- object.ToString() (Watch ablak is használja!)
- StringBuilder and immutability
<https://docs.microsoft.com/en-us/dotnet/api/system.string?view=netcore-3.0#Immutability>

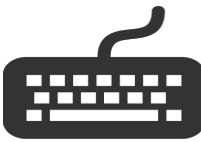
Néhány string metódus

- StartsWith, EndsWith, IndexOf, Contains
- LastIndexOf, LastIndexOfAny(Char[])
- Trim, Trim(Char[]), TrimEnd
- ToUpper
- `public string Substring (int startIndex, int length);`

...és még sok minden más! Ne írd meg nulláról...

<https://docs.microsoft.com/en-us/dotnet/api/system.string?view=netcore-3.0>

Serialization példa



- XML és JSON formátum (elem, attribútum)
- Szerializálás és deszerializálás

HF1a kiadása

OPC

- Regexp
- <https://docs.microsoft.com/en-us/dotnet/standard/base-types/regular-expressions>
(Example 2-vel bezárólag)

```

public static IEnumerable<AnimationCommand> Parse(string s)
{
    // EVIP: Regex to extract values from a string
    Regex r = new Regex(@"([a-z])([0-9]*):([0-9]*)(-([0-9]*))?");
    var matches = r.Matches(s);
    foreach (Match m in matches)
    {
        // Note: group 0 is the entire match. Group 1 is the
        // letter in the first bracket.
        string type = m.Groups[1].Captures[0].Value;
        int? param = GetIntFromRegexMatch(m, 2);
    }
}

private static int? GetIntFromRegexMatch(Match match, int groupIndex)
{
    int? result = null;
    var group = match.Groups[groupIndex];
    if (group.Captures.Count > 0)
    {
        // Note: we assume a single capture for every group
        Capture capture = group.Captures[0];
        if (int.TryParse(capture.Value, out int parsedResult))
            result = parsedResult;
    }
    return result;
}

```

svg2pptx\svg2pptx\AnimationCommands\AnimationCommand.cs