

Subiectul01 - Rezolvare

Subiectul01 - Rezolvare

FullCode

Output

FullCode

Link: <https://dotnetfiddle.net/ZXgJez>

```
using System;
using System.Collections.Generic;
using System.Text;
using System.IO;
using Newtonsoft.Json;
using System.Linq;

namespace examen {
    public delegate List<string> TextFormatter(List<string> s); /// 3
    public class TextDocument {
        public List<string> TextLines { get; set; }
        public char LineSeparator { get; set; }
        public TextFormatter Formatter { get; set; }

        public TextDocument() {
            TextLines = new List<string>();
            LineSeparator = '\n';
        }

        public void Format() ///3
        {
            this.TextLines = Formatter(TextLines);
        }

        public override string ToString() ///1
        {
            StringBuilder s = new StringBuilder();
            List<string> a = this.TextLines;
            foreach(var c in a)
                s.Append(c + this.LineSeparator);
            return s.ToString();
        }

        public override bool Equals(Object o) ///2
        {
            TextDocument textDocument = new TextDocument();
            if (o != null)
                textDocument = (TextDocument) o;
            else
                return false;
            List<string> a = this.TextLines;
            List<string> b = textDocument.TextLines;
```

```

        if (a.SequenceEqual(b) && textDocument.LineSeparator == this.LineSeparator
&& this.Formatter.Equals(textDocument.Formatter))
            return true;
        return false;
    }

    public void SaveOnDisk() ///5
    {
        this.TextLines = this.Formatter(this.TextLines);
        string filename =
System.Configuration.ConfigurationManager.AppSettings["File"];
        File.WriteAllText(Path.Combine(filename, "SaveDocument.txt"),
this.ToString());
    }

    public void LoadFromDisk() ///6
    {
        string Load =
System.Configuration.ConfigurationManager.AppSettings["Load"];
        TextDocument t = JsonConvert.DeserializeObject<TextDocument>
(Path.Combine(Load, "LoadDocument.txt"));
        this.TextLines = t.TextLines;
        this.LineSeparator = t.LineSeparator;
    }

    public List<string> String() ///7
    {
        return this.TextLines.Where(x => x.Length > 100 && x.Where(c => c == '
').Count() > 10).ToList();
    }

    public override int GetHashCode() {
        return base.GetHashCode();
    }
}

public class Program {
    static void Main(string[] args) {
        TextDocument t = new TextDocument();
        t.Formatter = new TextFormatter((List<String> text) => {
            List<String> formatted = new List<string>();
            foreach(var line in text) {
                formatted.Add(line.ToUpper());
            }
            return formatted;
        });

        t.TextLines.Add("maria");
        t.TextLines.Add("maria1");
        t.TextLines.Add("maria2");
        t.Format();

        TextDocument t1 = new TextDocument();
        t1.Formatter = new TextFormatter((List<String> text) => {
            List<String> formatted = new List<string>();
            for (int i = 0; i < text.Count; i++) {
                formatted.Add(i + 1 + "." + text[i].ToUpper());
            }
        });
    }
}

```

```

        return formatted;
    });

    t1.TextLines.Add("maria mihai maria maria mihai maria maria mihai maria
maria mihai maria maria mihai maria maria mihai maria maria mihai maria maria
mihai maria maria mihai maria maria mihai maria maria mihai maria maria mihai
maria maria mihai maria");
    t1.TextLines.Add("maria1");
    t1.TextLines.Add("maria2");
    t1.Format();

    List<string> a = t1.String();
    foreach(var b in a)
        Console.WriteLine(b);
    Console.WriteLine(t.Equals(t1) + "\n" + t1.ToString() + "\n" +
t.ToString());
    Console.ReadKey();
    }
}
}

```

Output

```

1.MARIA MIHAI MARIA MARIA MIHAI MARIA  MARIA MIHAI MARIA  MARIA MIHAI MARIA MARIA
MIHAI MARIA MARIA MIHAI MARIA MARIA MIHAI MARIA MARIA MIHAI MARIA MARIA MIHAI
MARIA MARIA MIHAI MARIA MARIA MIHAI MARIA MARIA MIHAI MARIA MARIA MIHAI MARIA
False
1.MARIA MIHAI MARIA MARIA MIHAI MARIA  MARIA MIHAI MARIA  MARIA MIHAI MARIA MARIA
MIHAI MARIA MARIA MIHAI MARIA MARIA MIHAI MARIA MARIA MIHAI MARIA MARIA MIHAI
MARIA MARIA MIHAI MARIA MARIA MIHAI MARIA MARIA MIHAI MARIA MARIA MIHAI MARIA
2.MARIA1
3.MARIA2

MARIA
MARIA1
MARIA2

```