using System.Collections.Generic;

using System.Text;

using static System.Console;

namespace FlyWeightPattern

{

public class FormattedText

{

private string plainText;

private bool[] capitalize;

public FormattedText(string plainText)

{

this.plainText = plainText;

capitalize = new bool[plainText.Length];

}

public void Capitalize(int start, int end)

{

for (int i = start; i <= end; ++i)

capitalize[i] = true;

}

public override string ToString()

{

var sb = new StringBuilder();

for (var i = 0; i < plainText.Length; i++)

{

var c = plainText[i];

sb.Append(capitalize[i] ? char.ToUpper(c) : c);

}

return sb.ToString();

}

}

public class BetterFormattedText

{

private string plainText;

private List<TextRange> formatting = new List<TextRange>();

public BetterFormattedText(string plainText)

{

this.plainText = plainText;

}

public TextRange GetRange(int start, int end)

{

var range = new TextRange { Start = start, End = end };

formatting.Add(range);

return range;

}

public override string ToString()

{

var sb = new StringBuilder();

for (var i = 0; i < plainText.Length; i++)

{

var c = plainText[i];

foreach (var range in formatting)

if (range.Covers(i) && range.Capitalize)

c = char.ToUpperInvariant(c);

sb.Append(c);

}

return sb.ToString();

}

public class TextRange

{

public int Start, End;

public bool Capitalize, Bold, Italic;

public bool Covers(int position)

{

return position >= Start && position <= End;

}

}

}

public class Demo

{

static void Main(string[] args)

{

var ft = new FormattedText("This is a brave new world");

ft.Capitalize(10, 15);

WriteLine(ft);

var bft = new BetterFormattedText("This is a brave new world");

bft.GetRange(10, 15).Capitalize = true;

WriteLine(bft);

}

}

}