Opgave 2 WPF - String

Objekter og nedarvning

# Formål

Formålet med opgaven er at konsolidere din viden om hvordan man anvender WPF til at lave en GUI som gør brugeren i stand til at løse en opgave. Du skal ligeledes lære noget om hvordan man anvender de metoder der findes i *String* klassen. Du skal også lære det grundlæggende omkring *collection type Dictionary<T,T>.* I forbindelse med *Split* metoden i *String* klassen, skal vi kigge lidt på *Linq*.

# Mål

Opgavens mål er

1. at kunne oprette et WPF-projekt ud fra et biled
2. at kunne lave en passende struktur i din solution
3. at kunne benytte standard metoder i klassen *String*
4. at kunne benytte og udnytte fordelene ved *Dictionary*
5. at kunne anvende metoder til udførelse af funktionalitet

# 2 Struktur – Grid med Row & Colum

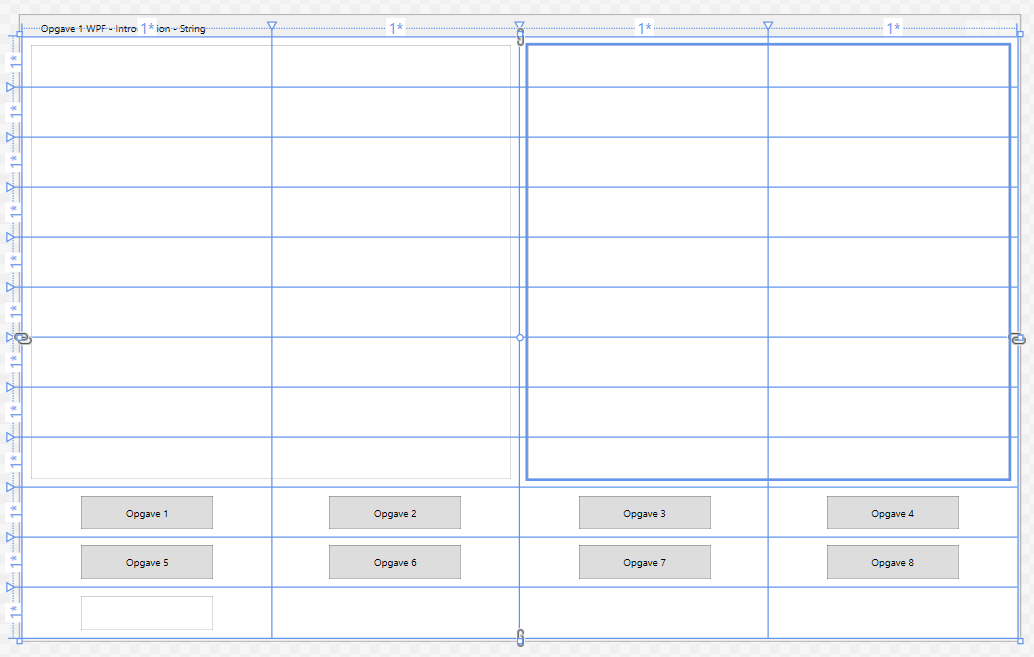
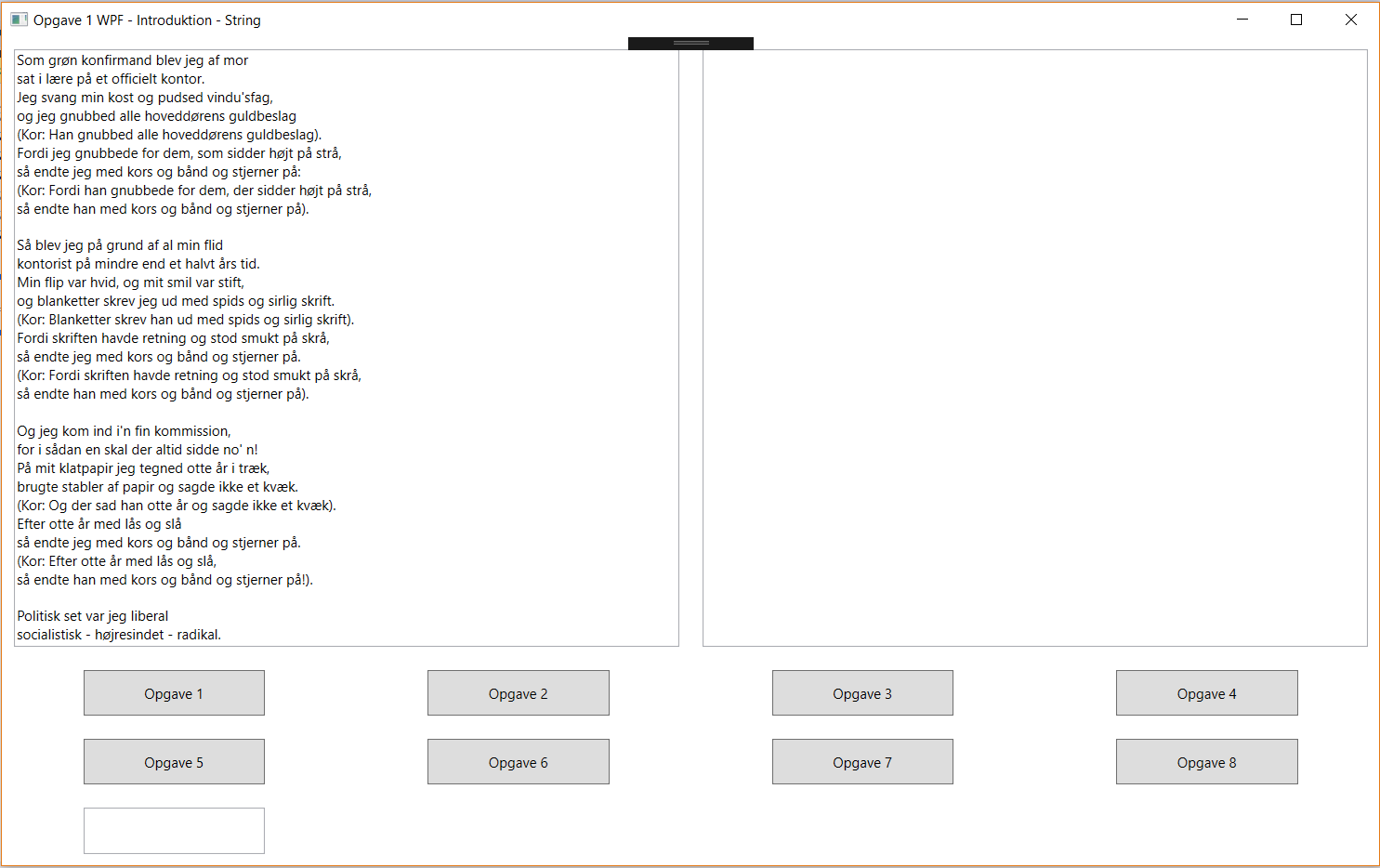
Til denne opgave skal du oprette et nyt projekt i din kodemappe. Det skal være et *WPF App(.NETFramework)* og navngives *WpfAppIntroduktionString.* Projektet skal gemmes i din kodemappe, i din *WPF* mappe.

## 2.1 Opgave 1

Du skal tilrette *MainWindow*, så det får størelsen: Højde-750 og Brede-1200 og får titlen *Opgave 1 WPF – IntroduktionString*.

Du skal oprette fire kolonner. Alle kollonner skal være flydende med lige fordeling af plads.

Du skal nu oprette 12 rækker. Alle rækker skal være flydende og med lige fordeling af plads.

Du skal benytte 8 Button og 3 tekstbokse, som placeres som vist her:  
 

Du skal nu tilføje to nye *Class* til projektet, det gøres på samme måde som du gjorde da du arbejdede med *ConsoleApp*. Den første *Class* skal have navnet *ClassBIZ* og skal placeres i en mappe med navnet *BIZ*. I *ClassBIZ* skal du oprette en *constructor*. Den anden *Class* skal have navnet *ClassIO* og skal placeres i en mappe med navnet *IO*. I *ClassIO* skal du oprette en *constructor*.

I den *Class* der knytter sig til selve *MainWindow* (MainWindow.xaml.cs), skal du lave en instans af din *ClassBIZ* så du kan tilgå de metoder, du snart skal kode for at kunne løse de følgende opgaver.

*Constructoren* i Class *MainWindow* skal lave et kald til en metode i ClassBIZ der skal lave et kald til ClassIO, hvor metoden *GetTestString(TextBox textBox)* som skal indlæse indholdet af filen Admiralen.txt, som du skal [hente her](https://intranet.aspit.dk/Fagets_Univers/Softwarekonstruktion/Documents/Materialer/S1/Brugerflader/Grundl%C3%A6ggende%20WPF/Admiralen.txt) og placere i din kodemappe.

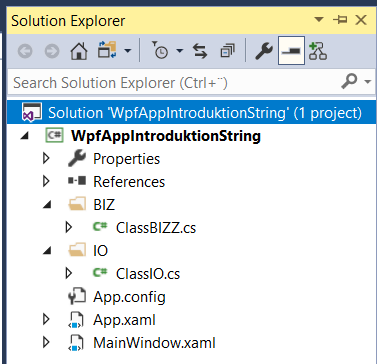
Opret en *Click Event* til knappen Opgave 1. Lav denne metode i ClassBIZ:  
 public int CountAllLines(TextBox textBox)

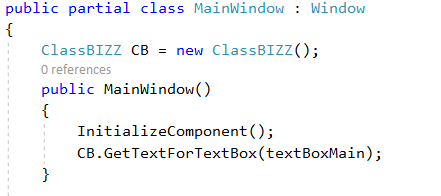
Metoden skal tælle hvor mange linjer, som indeholder tekst, der er i venstre *TextBox* og returnere antallet, så der kan indsættes en passende tekst med angivelse af antal linjer teksten og placere det i højre TextBox.

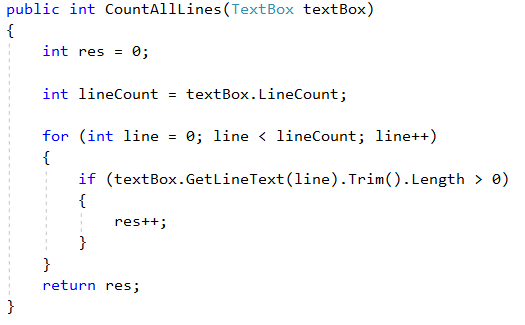
### Hjælp

MainWindow XAML:



ClassBIZ og ClassIO  


Instans af ClassBIZZ:  


ClassBIZZ:  


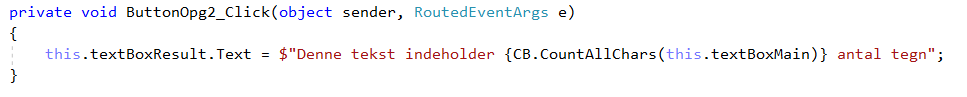
## 2.2 Opgave 2

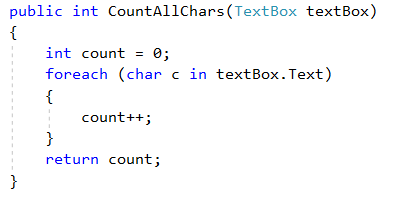
Du skal oprette en Click Eventhandler til knappen *Opgave 2*.  
Fra Eventhandleren skal der foretages et kald til denne metode i ClassBIZZ:  
public int CountAllChars(TextBox textBox)

Som det fremgår af metodenavnet, skal du lave noget kode der kan tælle hvor mange tegn dokumentet indeholder. Tallet med værdien returneres til Eventhandleren hvor resultatet skrive i højre TextBox.

Test om programmet virker efter hensigten.

### Hjælp

MainWindow.xaml.cs :  


ClassBIZ :  


## Opgave 3

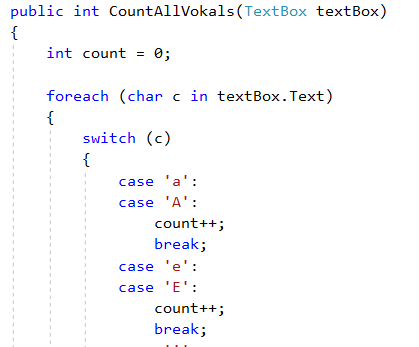
I denne opgave skal du benytte meget af det du lavede i opgave 2.1 og 2.2 .

Du skal i *ClassBIZ* lave følgende metode:  
public int CountAllVokals(TextBox textBox)

Som du kan se af metodenavnet, skal denne metode tælle hvor mange vokaler der findes i teksten og returnere antallet til EventHandleren.

Det kan anbefales at der benyttes en [*Switch*](http://www.tutorialsteacher.com/csharp/csharp-switch) til at afgøre om karakteren er en vokal ([Hvad er en vokal](http://opslagsvaerker.gyldendal.dk/en/OpslagsvaerkerVirtuelle/DanskLexMellemtrin/konsonanter_og_vokaler.aspx)).

### Hjælp

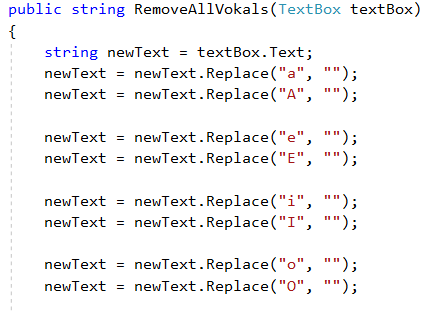


## Opgave 4

I denne opgave skal du som i opgave 2.3 manipulere med vokalerne i teksten.

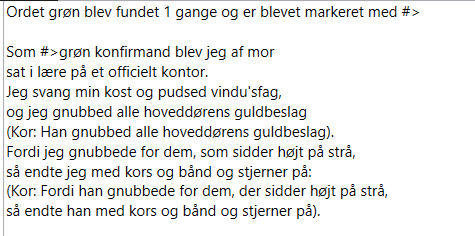
Du skal i ClassBIZZ oprette følgende metode:  
public string RemoveAllVokals(TextBox textBox)   
   
Som det fremgår af ovenstående tekst, skal din kode fjerne alle vokaler i teksten og returnere teksten uden vokaler til EventHandleren som så skal indsætte teksten uden vokaler i højre TextBox.

### Tip!

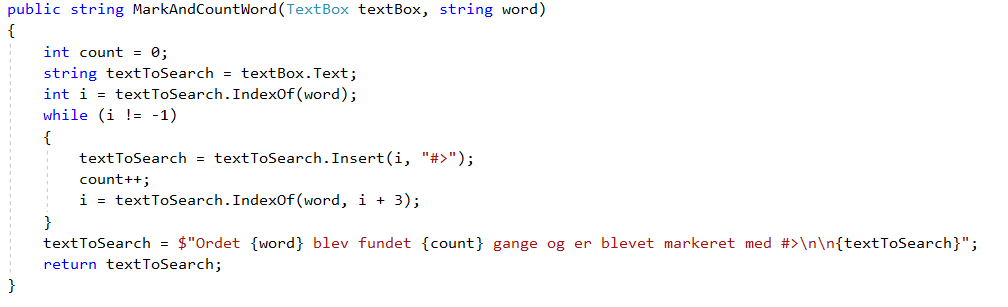


## Opgave 5

I denne opgave skal du finde ord i teksten.   
De ord der skal findes, skal angives af brugeren i TextBox’en under knappen Opgave 5.  
Når ordet findes i teksten, skal det markeres ved at indsætte dette tegn: #> til venstre for ordet der skal søges efter.

Resutatet skal indsættes i højre TextBox som vist her:  


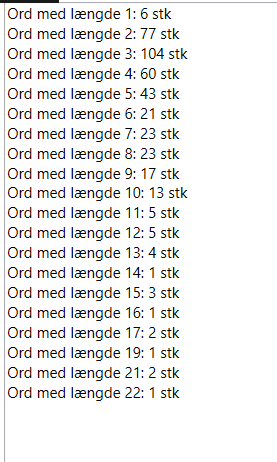
### Tip!



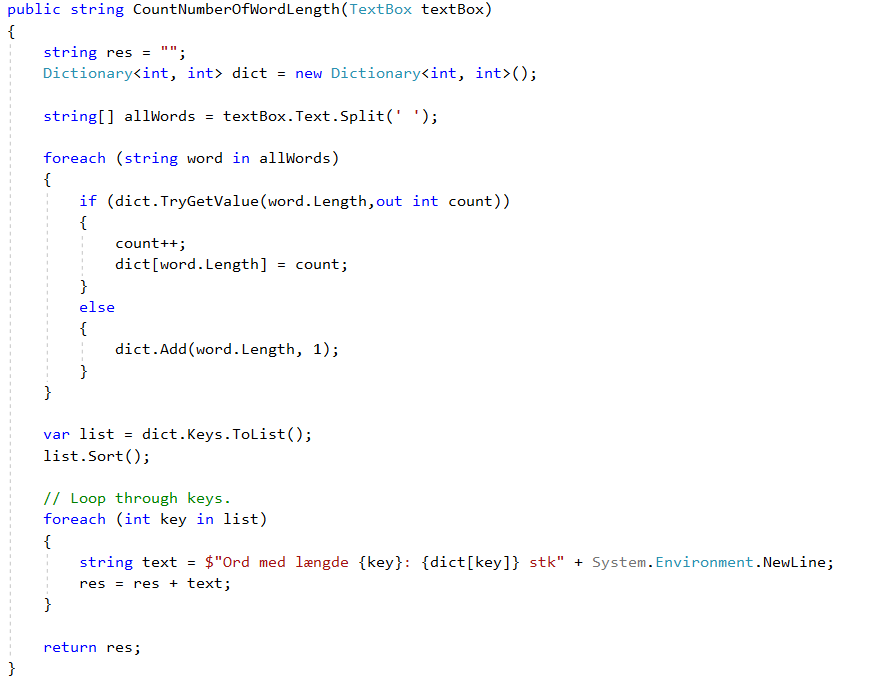
## Opgave 6

I denne opgave skal du lave en metode i ClassBIZZ som kan lave en liste hvor det angive hvor mange ord der findes i teksten af forskellig længde.

Metoden skal oprettes som følger:  
public string CountNumberOfWordLength(TextBox textBox)

Du skal til denne løsning anvende *Dictionary* for at strukturere resultatet. Du skal især have fokus på metoden i [*Dictionary*](http://www.tutorialsteacher.com/csharp/csharp-dictionary) som heder *TryGetValue*.  
Resultatet skal se ud som vist her:  


### Tips!



## Opgave 7

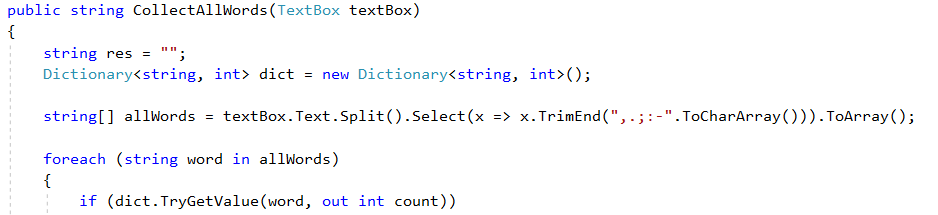
Denne opgave minder meget om opgave 2.6, men til forskel fra opgave 2.6 skal du i denne opgave tælle hvor mange gange de enkelte ord optræder i teksten.

Du skal benytte samme fremgangsmåde, men i stedet for at registrere om ord af denne længde forefindes i dit Dictionary, skal du registrere om ordet er oprettet og hvis det er opskrive tælleren med 1.

I denne kode vil du også stifte bekendtskab med [*System.Linq*](http://www.tutorialsteacher.com/csharp/csharp-linq)som er en klasse der giver dig mulighed for at lave langt bedre udvælgelse af data i forskellige datatyper.  
string[] allWords = textBox.Text.Split().Select(x => x.TrimEnd(",.;:-".ToCharArray())).ToArray();

Resultatet skal præsenteres på følgende måde:  


### Tips!



## Opgave 8

I denne opgave skal du i ClassBIZZ oprette en metode der skal fjerne sidste bogstav i alle ord der er længere end 3 karaktere.

Resultatet skal præsenteres i højre TextBox.

Metoden skal ikke returnere nogen værdi, men foretage opdateringen af TextBox’en i metoden i ClassBIZZ.

Metoden skal oprettes således:  
public void CutAwayLastLetterWhereWordIsLongerThan3Chars(TextBox oldText, TextBox newText)

Du skal benytte funktionen [*Substring(int, int)*](https://www.dotnetperls.com/substring) til at manipulere teksten med.

Du skal ligeledes benytte [*StringCollection*](https://www.c-sharpcorner.com/UploadFile/mahesh/stringcollection-in-C-Sharp/), en særlig form for *collection* tilhåndtering af datatypen *String*.

### Tips!

