

# C# Essentials Veel gebruikte klassen

Koen Bloemen

Wim Bervoets

Kristof Palmaers



**DE HOGESCHOOL  
MET HET NETWERK**

Elfde-Liniestraat 24, 3500 Hasselt, [www.pxl.be](http://www.pxl.be)





- `System.Math`
- `System.String`
- `System.Text.StringBuilder`
- `System.Random`
- `System.Windows.MessageBox`
- `Microsoft.VisualBasic.Interaction`



# Veelgebruikte klassen

- Zelf documentatie opzoeken tijdens coderen
  - CTRL-klik op de method binnen VS
  - Class opzoeken op <https://docs.microsoft.com>

```
... public static void WriteLine(object value);  
//  
// Summary:  
//     Writes the specified string value, followed by the current line  
//     the standard output stream.  
//  
// Parameters:  
//     value:  
//     The value to write.  
//  
// Exceptions:  
//     T:System.IO.IOException:  
//     An I/O error occurred.  
public static void WriteLine(string value);  
... public static void WriteLine(string format, object arg0);  
... public static void WriteLine(string format, object arg0, object  
... public static void WriteLine(string format, object arg0, object  
... public static void WriteLine(string format, params object[] arg);  
... public static void WriteLine(char[] buffer, int index, int count);  
... public static void WriteLine(decimal value);
```

Docs / .NET / .NET API browser / System / Console / Methods / WriteLine

Version

.NET 5

Search

write

WriteLine

> Events  
> ConsoleCancelEventArgs  
ConsoleCancelEventHandler  
ConsoleColor  
ConsoleKey  
> ConsoleKeyInfo  
ConsoleModifiers  
ConsoleSpecialKey  
> ContextBoundObject  
> ContextMarshalException  
> ContextStaticAttribute  
> Convert  
Converter<TInput,TOutput>  
> DataMisalignedException

## Console.WriteLine Method

### Definition

Namespace: System

Assembly: System.Console.dll

Writes the specified data, followed by the current line terminator, to the standard output stream.

### Overloads

<a href="#">WriteLine(String, Object, Object)</a>	Writes the text representation of the specified objects, followed by the current line terminator, to the standard output stream using the specified format information.
<a href="#">WriteLine(String)</a>	Writes the specified string value, followed by the current line terminator, to the standard output stream.
<a href="#">WriteLine(Char[], Int32, Int32)</a>	Writes the specified subarray of Unicode characters, followed by the current line terminator, to the standard output stream.

# System.Math (namespace: using System;)

- Constantes
  - $\text{Math.PI} = 3.14159265358979$  (double)
  - $\text{Math.E} = 2.71828182845905$  (double)
- Functies
  - Absolute waarde
    - $\text{Math.Abs}(-12.34) = 12.34$
  - Minimum en maximum
    - $\text{Math.Min}(17, 35) = 17$
    - $\text{Math.Max}(17, 35) = 35$
  - Afronden naar dichtstbijzijnde even (!) getal i.g.v. randgevallen
    - $\text{Math.Round}(10.5) = 10$
    - $\text{Math.Round}(10.51) = 11$  (11 is niet even, maar wel het dichtstbijzijnde getal)
    - $\text{Math.Round}(11.5) = 12$
    - $\text{Math.Round}(-12.3456, 2) = -12,35$

# System.Math (namespace: using System;)

- Functies
  - Afronden naar boven
    - $\text{Math.Ceiling}(-11.5) = -11$
    - $\text{Math.Ceiling}(11.5) = 12$
  - Afronden naar beneden
    - $\text{Math.Floor}(-11.5) = -12$
    - $\text{Math.Floor}(11.5) = 11$
  - Machtsverheffing
    - $\text{Math.Pow}(2, 3) = 2^3 = 2*2*2 = 8$
  - Logaritme (inverse van machtsverheffing)
    - $\text{Math.Log}(8, 2) = 3$  (tot de hoeveelste macht 2 nemen om 8 te krijgen)
  - Vierkantswortel
    - $\text{Math.Sqrt}(16) = 4$

# System.String (namespace: using System;)

- `String str = "C Sharp";`
- Testen of string leeg is
  - `if (str.Length == 0) {}`
  - NIET DOEN: `if (str == "") {}`
- Strings vergelijken
  - `if (str.Equals("C Sharp")) {}`
  - NIET DOEN: `if (str == "C Sharp") {}`
- Equals geeft foutmelding als string null is => kan opgevangen worden

# System.String (namespace: using System;)

Functie	Resultaat	
<code>string.concat("pannen", "koek");</code>	"pannenkoek"	String samenvoegen
<code>string.Compare("aa", "AA");</code>	-1	String vergelijken 0 = gelijk -1 = linkse kleiner dan rechtse 1 = linkse groter dan rechtse
<code>string.Compare("aa", "AA", true);</code>	0	true negeert hoofdletters!
<code>string.CompareOrdinal("A", "a");</code>	-32	Vergelijkt ASCII-code (A=65 en a=97)
<code>string.Equals("abcd", "Abcd");</code>	false	Zijn strings gelijk
<code>string.Equals("A", "a", StringComparison.OrdinalIgnoreCase)</code>		Zijn strings gelijk Negeer hoofdletters



# System.String (namespace: using System;)

Method	Resultaat	
"Visual CSharp".Substring(2);	"sual CSharp"	Deel uit string halen vanaf 2 <sup>de</sup> teken tot einde (start vanaf 0!!!)
"Visual CSharp".Substring(2,5);	"sual "	Deel uit string halen vanaf 2 <sup>de</sup> teken, 5 tekens lang (start vanaf 0!!!)
"Visual CSharp".PadLeft(20); "Visual CSharp".PadRight(20);	" Visual CSharp" "Visual Csharp "	Vult string aan langs links/rechts met spaties tot 20 chars.
"Visual CSharp".PadLeft(15,'*'); "Visual CSharp".PadRight(15,'*');	"**Visual CSharp" "Visual Csharp**"	Vult string aan langs rechts met * tot 15 chars.
"Visual CSharp".Length;	13	Lengte van string (property)
"Visual CSharp".Contains("CSharp") "Visual CSharp".Contains("C#")	true false	Kijkt of string substring bevat
"Visual CSharp".IndexOf("*"); "Visual CSharp".IndexOf("CSh");	-1 7	Geef positie vanaf 0 waar de substring in de string zit. -1 indien niet gevonden



# System.String (namespace: using System;)

Method	Resultaat	
" Visual CSharp ".Trim();	"Visual CSharp"	Verwijdert whitespaces (spaties, newlines, ...), zowel aan begin als einde
" Visual CSharp ".TrimEnd();	" Visual Csharp"	Enkel aan einde trimmen
" Visual CSharp ".TrimStart();	"Visual CSharp "	Enkel aan begin trimmen
"Visual CSharp".Remove(4);	"Visu"	Verwijder vanaf 4 <sup>de</sup> karakter (0-based!)
"Visual CSharp".Remove(4,3);	"VisuCSharp"	Verwijder 3 karakters vanaf 4 <sup>de</sup> karakter (0-based!)
"Visual C#".StartsWith("Visual");	true	Start de string met een bepaalde substring? (true/false)
"Visual CSharp".Insert(2,"TEST");	"ViTESTsual CSharp"	Voegt string "TEST" toe vanaf het 2 <sup>de</sup> karakter (0-based!)
"Visual CSharp".Replace("a","o");	Visuol Cshorp	Vervangt elke a door o
"Visual CSharp".ToUpper();	VISUAL CSHARP	String in hoofdletters
"Visual CSharp".ToLower();	visual csharp	String in kleine letters

# System.Text.StringBuilder (using System.Text;)

- StringBuilder is performanter dan string concatenatie!
- Append()  
Nieuwe tekst toevoegen
- AppendLine()  
Nieuwe tekst toevoegen en nieuwe regel starten

```
StringBuilder sb = new StringBuilder(); // start van lege string
sb.Append("Nieuwe tekst:\r\n");
sb.AppendLine("Punt:").Append(" 10").AppendLine();
string str = sb.ToString(); // op het einde omvormen naar een string voor gebruik
```

# System.Text.StringBuilder (using System.Text;)

```
StringBuilder sb = new StringBuilder(); // start van lege string
sb.Append("Nieuwe tekst:\r\n");
sb.AppendLine("Punt:").Append(" 10").AppendLine();
Console.WriteLine(sb);
Console.WriteLine();
sb.Insert(0, "StringBuilder:").Insert(14, Environment.NewLine);
sb.Replace(":", "-", 15, sb.Length - 15);
for (int i = 0; i < 3; i++)
{
    for (int j = 0; j < 2; j++)
    {
        sb.AppendFormat("{0}-{1}", i, j);
    }
    sb.Append("|");
}
Console.WriteLine(sb);
Console.WriteLine();
sb.Remove(0, 14);
Console.WriteLine(sb);
Console.WriteLine();
sb.Clear();
Console.WriteLine(sb);
Console.WriteLine();
```

# System.Random

- Willekeurig (random) getal genereren
- Maak eerst instantie aan
- Lege constructor  
telkens verschillende stroom
- Seed meegeven  
in dezelfde stroom

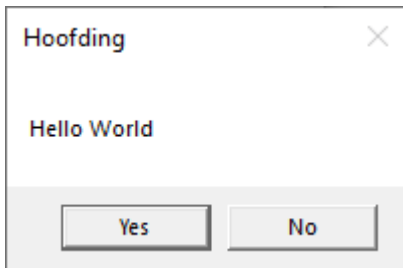
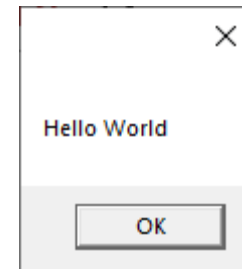
```
Random rnd = new Random();  
int getal = rnd.Next(1, 7); //genereer nieuw getal van 1 t.e.m. 6 (!)  
double dbl = rnd.NextDouble(); //dbl >= 0.0 en < 1.0  
Console.WriteLine(getal);  
Console.WriteLine(dbl);
```



# System.Windows.MessageBox

- Berichtvenster om mededeling te tonen
- **NIET** in console applicatie, **ENKEL** in WPF!!!

```
MessageBox.Show("Hello World");
```



```
MessageBox.Show("Hello World", "Hoofding", MessageBoxButton.YesNo);
```

# System.Windows.MessageBox

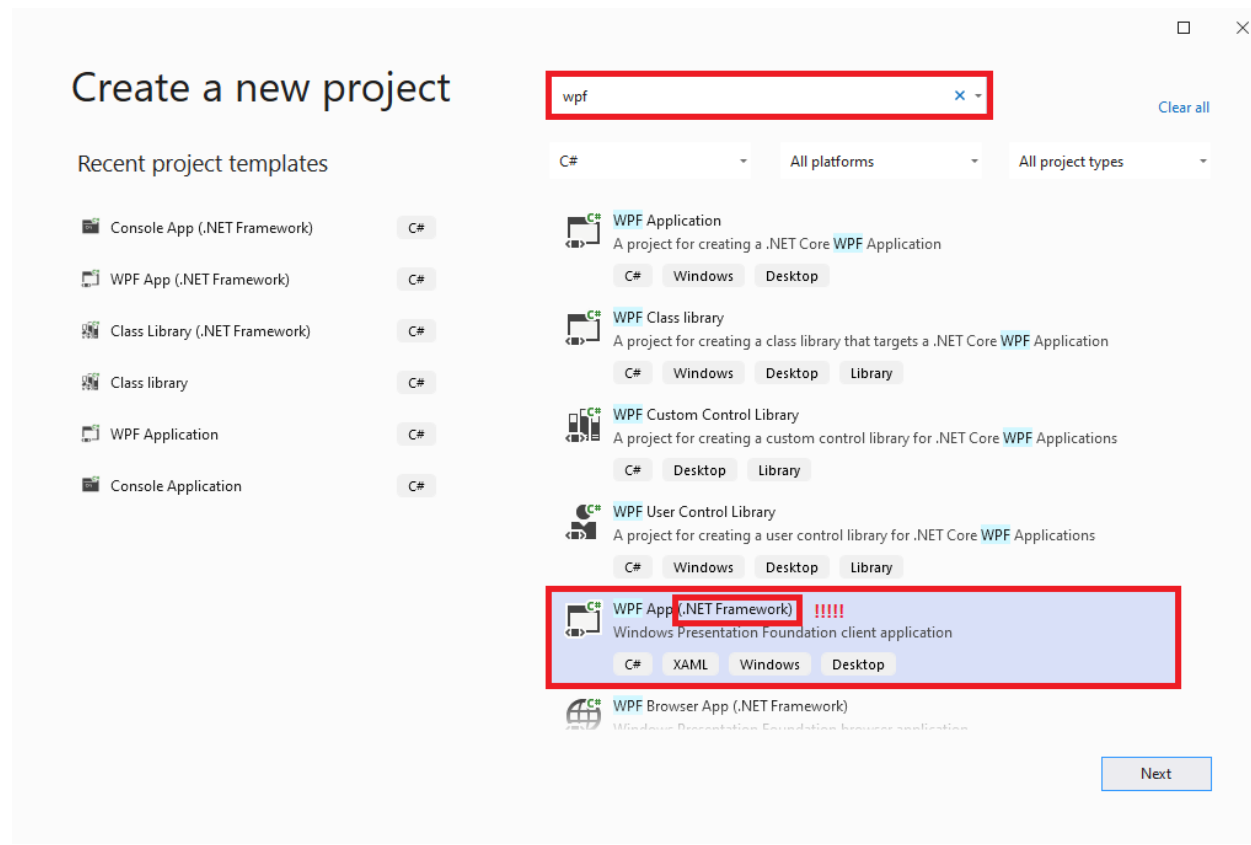
- Aangeklikt resultaat opvragen

```
MessageBoxResult antwoord = MessageBox.Show("Wil je echt afsluiten?", "Project  
afsluiten", MessageBoxButton.YesNo, mess?MessageBoxImage.Question,  
MessageBoxResult.No);
```

```
if (antwoord == MessageBoxResult.Yes)  
{  
    //Project wordt afgesloten  
}  
else  
{  
    //Project wordt niet afgesloten  
}
```

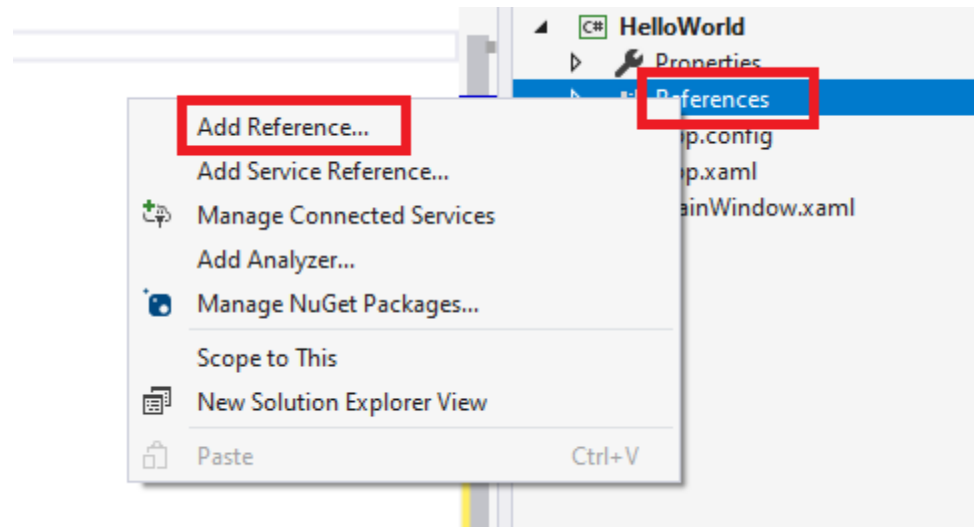
# Microsoft.VisualBasic.Interaction

- Bestaat **niet** in native C#
- Werkt voorlopig nog niet in .NET Core!



# Microsoft.VisualBasic.Interaction

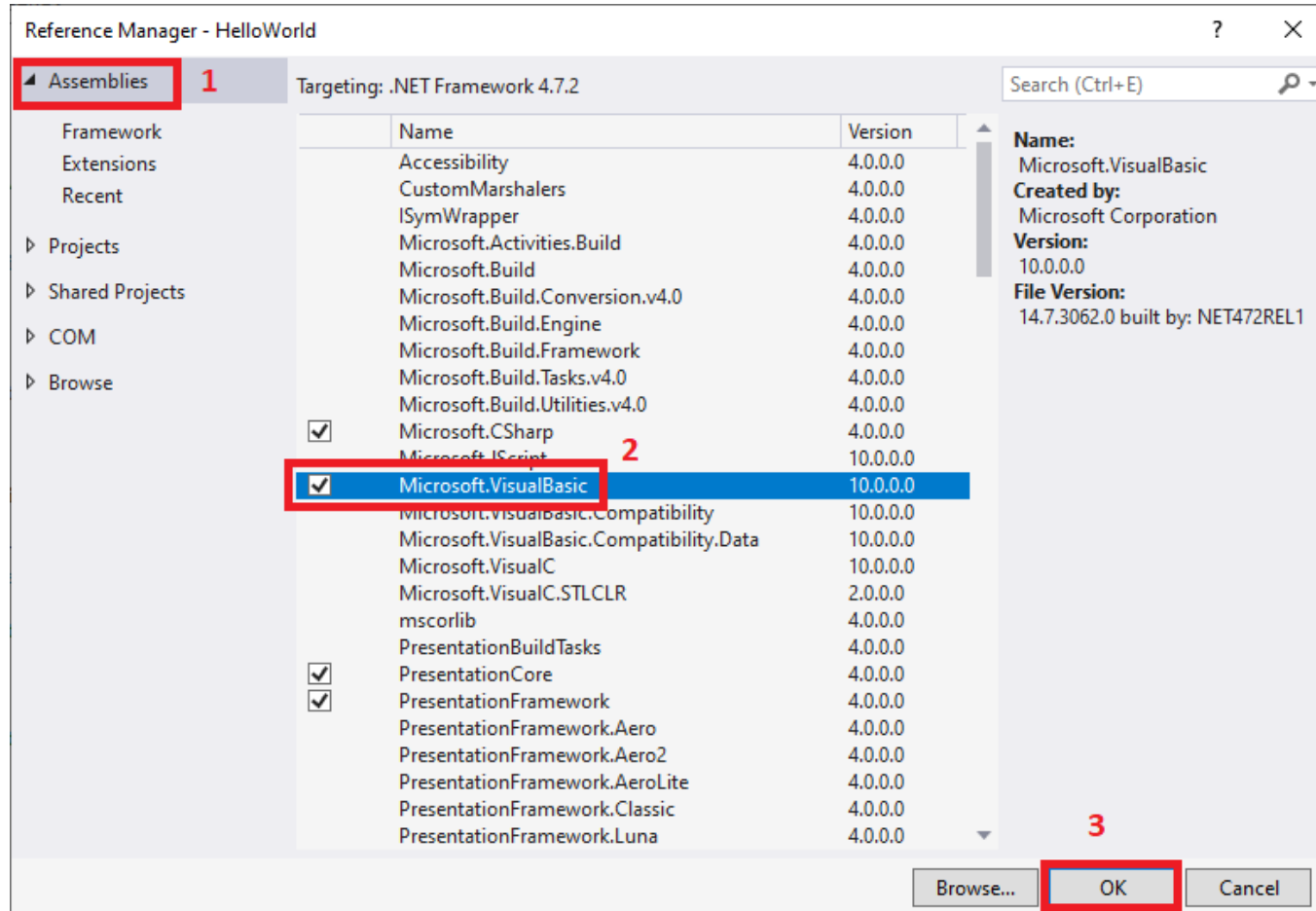
- Rechtsklik op References => Add Reference...





# Microsoft.VisualBasic.Interaction

- Duid bij Assemblies aan: Microsoft.VisualBasic



# Microsoft.VisualBasic.Interaction

```
string antwoord = Interaction.InputBox("Geef een getal", "Invoer", "50", 500);  
while (string.IsNullOrEmpty(antwoord))  
{  
    MessageBox.Show("Geef getal!", "Foutieve invoer");  
    antwoord = Interaction.InputBox("Geef een getal", "Invoer", "50", 500);  
}
```

