Version 1.2.0.0  
egonl  
February 2019

SharpDocx

# Summary

SharpDocx is inspired by Web technologies like ASP.NET and JSP. Developers familiar with those technologies should feel right at home.

First you create a view in Word. A view is a Word document which also contains C# code. Code can be inserted anywhere: 24 would insert 24.

The next step is to create documents based on this view. This requires two lines of code:

var document = DocumentFactory.Create("view.cs.docx");

document.Generate("output.docx");

Out of the box SharpDocx supports inserting text, tables, images and more. This tutorial shows you how.

If you want, you can specify a view model to be used in your view. Then you could write things like < % foreach (var item in Model.MyList) { % >. See the Model sample.

If you want to do something that's not supported by SharpDocx, you can do so by creating your own document subclass. See the Inheritance example.

Generating documents with SharpDocx can be very fast: a slightly modified Model sample produced 25 documents per second on my modest laptop. That’s 1500 documents per minute. Single threaded.

# Contents

[Summary 1](#_Toc502494466)

[Contents 2](#_Toc502494467)

[The basics 3](#_Toc502494468)

[Write method 3](#_Toc502494469)

[Conditional content 3](#_Toc502494470)

[Loops 4](#_Toc502494471)

[Tables 5](#_Toc502494472)

[Images 6](#_Toc502494473)

[Replacing text 7](#_Toc502494474)

[Referencing assemblies and importing namespaces 8](#_Toc502494475)

[The Map 9](#_Toc502494476)

# The basics

At any point in the text you can insert C# statements. Like right here.

The result looks like this:



## Write method

If you want to display the value of i, you can use the Write method. Right now, i is 1.

There’s also a shorthand notation for the Write method: i is still 1.

You can insert line breaks by using ‘\n’:

This paragraph  
contains two  
line breaks.

## Conditional content

You can use an if statement to display conditional content.

This will be displayed.

In this case, any formatting will be lost because the code parser ignores any formatting.

If you want to conditionally display text with formatting, use two code blocks and place the text between curly brackets, like this.

This **will** *also* be displayed.

If you want, you can span multiple elements. E.g.

The diverging pronunciation of tomato (though not so much potato) is primarily one of regional dialect.

The pronunciation 'tuh-MAH-toh' is the standard pronunciation in the UK and is accepted in the US regions of New England along with parts of the lower East Coast, while 'tuh-MAY-toh' is found almost everywhere else.

## Loops

If you want to add something in a loop, you should do so programmatically. This example does loop, but doesn’t add anything to the document:

The value of i is **9**.

In the next loop we’ll call the AppendParagraph method, which *will* insert new paragraphs:

The value of i is **0**.

The value of i is **1**.

The value of i is **2**.

The value of i is **3**.

The value of i is **4**.

The value of i is **5**.

The value of i is **6**.

The value of i is **7**.

The value of i is **8**.

The value of i is **9**.

## Tables

Design your tables in Word, and create rows using the AppendRow method.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| i \* 1 | i \* 2 | i \* 3 | i \* 4 | i \* 5 | i \* 6 | i \* 7 | i \* 8 | i \* 9 | i \* 10 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 |
| 3 | 6 | 9 | 12 | 15 | 18 | 21 | 24 | 27 | 30 |
| 4 | 8 | 12 | 16 | 20 | 24 | 28 | 32 | 36 | 40 |
| 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 |
| 6 | 12 | 18 | 24 | 30 | 36 | 42 | 48 | 54 | 60 |
| 7 | 14 | 21 | 28 | 35 | 42 | 49 | 56 | 63 | 70 |
| 8 | 16 | 24 | 32 | 40 | 48 | 56 | 64 | 72 | 80 |
| 9 | 18 | 27 | 36 | 45 | 54 | 63 | 72 | 81 | 90 |
| 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 |

## Images

Insert images using the Image method.

​

If only a file name is specified, SharpDocx searches this file in a directory specified by the ImageDirectory property. Right now this property has been set to ‘D:\projects\asna\contribute\SharpDocx\Samples\SampleProjects\Tutorial\bin\Debug\net35/../../../../../Images’.

The Image method accepts a second optional parameter that specifies the relative size of the image. Here’s  at 15%.

Images that are too wide to be displayed at 100% are automatically scaled back. Here’s an example:



## Replacing text

If you want to replace text, you can use the Replace method.

This will replace *all* occurrences of the specified string.[[1]](#footnote-2)

Here’s the **replaced text**. And here’s some more replaced text.

## Referencing assemblies and importing namespaces

If you want to use specific types in a view, you can use the Assembly and Import directives to get access to them. Directives look like regular code blocks, but they always start with < %@.

Reference an assembly with the Assembly directive.[[2]](#footnote-3)

Import namespaces with the Import directive.

In C# you would write:

using System.Xml.Linq;

Now we can use types in System.Xml.Linq. Let’s read some news.

**Samsung Chips Will Get Faster and Easier on Your Battery in 2020**Processor progress is harder to come by these days, but Samsung says it'll build chips next year that will give you a bit more battery life or a little more speed. From a report: Through improvements …

**New York City Has a Y2K-Like Problem, and It Doesn't Want You To Know About It**On April 6, something known as the GPS rollover, a cousin to the dreaded Y2K bug, mostly came and went, as businesses and government agencies around the world heeded warnings and made software or hard…

**What To Expect From Sony's Next-Gen PlayStation**Daetrin writes: Sony is unwilling to confirm "Playstation 5" as the name, but their next console is "no mere upgrade" according to a report from Wired, which cites Sony executives -- who spoke on the …

**Mark Zuckerberg Leveraged Facebook User Data To Fight Rivals and Help Friends, Leaked Documents Show**Facebook CEO Mark Zuckerberg once considered making deals with third-party developers just to help him find out how much users' data is worth, NBC News reported on Tuesday. The report, which cites 4,0…

**'Avengers: Endgame' Footage Leaks on Reddit, YouTube, and Twitter**Despite Disney's efforts to keep as much of Avengers: Endgame under wraps as possible before the latest Marvel blockbuster hits theaters next week, several minutes of blurry Avengers: Endgame footage …

**Scranos Rootkit Expands Operations From China To the Rest of the World**A malware operation previously limited to China's borders has expanded over the past few months to infect users from all over the world, antivirus firm Bitdefender said in a report published today. Fr…

**Mozilla Wants Apple To Change Users' iPhone Advertiser ID Every Month**Mozilla has launched a petition today to get Apple to rotate the IDFA unique identifier of iOS users every month. From a report: The purpose of this request is to prevent online advertisers from creat…

**US Government Admits It Doesn't Know If Assange Cracked Password For Manning**An anonymous reader quotes a report from Motherboard: The U.S. government does not have any evidence that WikiLeaks founder Julian Assange succeeded in cracking a password for whistleblower Chelsea Ma…

**Researchers 3D-Print Heart From Human Patient's Cells**Researchers have 3D-printed a heart using a patient's cells, providing hope that the technique could be used to heal hearts or engineer new ones for transplants. "This is the first time anyone anywher…

**Traversable Wormholes Can Exist, But They're Not Very Useful For Space Travel, Physicists Say**A new study from physicists at Harvard and Stanford says that wormholes can exist but they're not very useful for humans to travel through. "It takes longer to get through these wormholes than to go d…

**Fukushima: the Removal of Nuclear Fuel Rods From Damaged Reactor Building Begins**An anonymous reader quotes a report from The Guardian: Workers at the wrecked Fukushima Daiichi nuclear power plant have begun removing fuel rods from a storage pool near one of the three reactors tha…

**iOS 13 To Feature Dark Mode and Interface Updates, Report Says**9to5Mac has learned of several new features expected to be included in iOS 13. From the report: Dark Mode: There will be a system-wide Dark Mode that can be enabled in Settings, including a high contr…

**Immune Cells May Play a Role In Causing Cavities**Researchers at the University of Toronto have found that cavities may be collateral damage from an overzealous immune system. New Atlas reports: Traditionally, bacteria have taken most of the blame fo…

**TicTocTrack Smartwatch Flaws Can Be Abused To Track Kids**secwatcher shares a report from Threatpost: A popular smartwatch that allows parents to track their children's whereabouts, TicTocTrack, has been discovered to be riddled with security issues that cou…

**Science and Bicycling Meet In a New Helmet Design**John Timmer from Ars Technica got a chance to take a look at Trek's new bicycle helmet that they claim offers "the first major change in helmet technology in years," and is backed up with peer-reviewe…

In a real world scenario you wouldn’t fetch data or have this much code in a view. But hey, this is just an example.

## The Map

The Map maps OpenXmlElements to plain text and vice versa. It’s being used internally by the Replace method and for finding the C# code in views, among other things. At the moment Map.Text looks something like this:

Version 1.2.0.0  
egonl  
February 2019  
Version 1.2.0.0  
egonl  
February 2019  
SharpDocx  
SharpDocx  
  
  
Summary  
SharpDocx is inspired by Web technologies like ASP.NET and JSP. Developers familiar with those technologies should feel right at home.  
First you create a view in Word. A view is a Word document which also contains C# code. Code can be inserted anywhere: 24 would insert 24.  
The next step is to create documents based on this view. This requires two lines of code:  
 var document = DocumentFactory …

The Map might be handy when you want to search the document for text.

## The SharpDocx solution

### Building the example programs

The Tutorial, Inheritance and Model samples will by default be build for .NET Framework 3.5, 4.5 and .NET Core 2.0. The .NET Core 2.0 builds will use the .NET Standard 2.0 version of SharpDocx.

The Tutorial and Inheritance samples will by default run in .NET Framework 4.5. The Model sample will by default run in .NET Core 2.0. If you want to change this, right click on the solution file in Visual Studio and select **Edit SampleName.csproj**. This will open the csproj-file. The first target named on this line will be used for startup/debugging in Visual Studio:

<TargetFrameworks>net45;net35;netcoreapp2.0</TargetFrameworks

### Unix-like systems

On Unix-like systems, remove the net45 and net35 targets from all projects, because they are only available on Windows. Other than that, the library and samples should compile and run fine.

1. Actually, this will only replace text in the body of the document, and not in headers, footers, end- or footnotes. So this won’t work as expected. But you can use code here. [↑](#footnote-ref-2)
2. When you’re using the .NET Standard version of SharpDocx, all SDK DLL’s (System.\* and Microsoft.\* DLL’s) are automatically referenced. So in this scenario, referencing System.Xml is not needed. [↑](#footnote-ref-3)