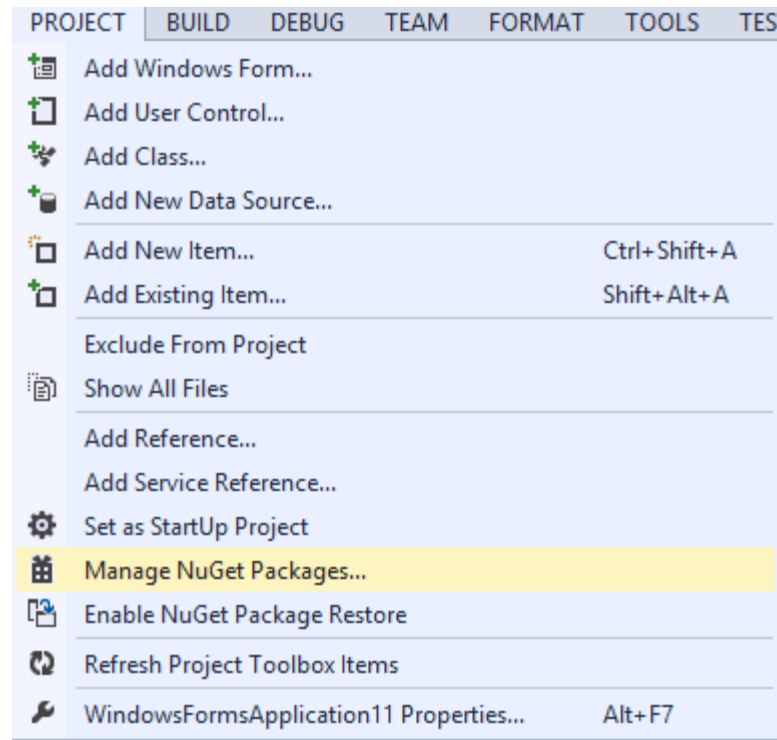


C# Word

Use NuGet Packages



WindowsFormsApplication11 - Manage NuGet Packages



Installed packages

Online

- All
- nuget.org
- Microsoft and .NET
- Search Results

Updates

Each package is licensed to you by its owner. Microsoft is not responsible for, nor does it grant any licenses to, third-party packages.

Stable Only Sort by: Relevance

- DocX** DocX is a .NET library that allows developers to manipulate Word 2007 files, in an easy and intuitive manor.
- Aspose.Words for .NET** Aspose.Words for .NET is a class library for .NET that enables you to perform a wide range of document processing tasks d...
- ABCpdf .NET** A powerful and easy to use PDF component for .NET.
- TemplateEngine.Docx** Simple template engine for generating Word docx documents on the server-side with a human-created Word templates, ba...
- Fluent OpenXml** Fluent OpenXml Generator is .NET library which provides a fluent API for generating, changing and templating docume...
- DocXTemplateEngine** A template engine for the .NET platform which takes a DOCX file and applies a data object to generate reports, do mail me...
- Templater** Templater is a small reporting library with a minimal interface. Use your favorite editor (Word, Excel) to create report templa...

1 2 >

docx

Created by: Cathal Coffey
Id: DocX
Version: 1.0.0.13
Last Published: 3/12/2556
Downloads: 8108
License
[View License](#)
MS-PL
[Project Information](#)
[Report Abuse](#)
Description:
DocX is a .NET library that allows developers to manipulate Word 2007 files, in an easy and intuitive manor.
Tags: Word
Dependencies:
No Dependencies

Settings

Close

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using Novacode;

using System.Diagnostics;

```
string fileName = @"e:\DocXExample.docx";  
  
// Create a document in memory:  
  
var doc = DocX.Create(fileName);  
  
// Insert a paragrpah:  
  
doc.InsertParagraph("This is my first paragraph");  
  
// Save to the output directory:  
  
doc.Save();  
  
// Open in Word:  
  
Process.Start("WINWORD.EXE", fileName);
```

More example

```
string fileName = @"e:\DocXExample2.docx";  
string headlineText = "Constitution of the United States";  
string paraOne = ""  
    + "We the People of the United States, in Order to form a  
    more perfect Union, "  
    + "establish Justice, insure domestic Tranquility, provide for  
    the common defence, "  
    + "promote the general Welfare, and secure the Blessings of  
    Liberty to ourselves "  
    + "and our Posterity, do ordain and establish this Constitution  
    for the United "  
    + "States of America.";
```

```
// A formatting object for our headline:
var headLineFormat = new Formatting();
headLineFormat.FontFamily = new System.Drawing.FontFamily("Arial Black");
headLineFormat.Size = 18D;
headLineFormat.Position = 12;

// A formatting object for our normal paragraph text:
var paraFormat = new Formatting();
paraFormat.FontFamily = new System.Drawing.FontFamily("Calibri");
paraFormat.Size = 10D;

// Create the document in memory:
var doc = DocX.Create(fileName);

// Insert the now text objects;
doc.InsertParagraph(headlineText, false, headLineFormat);
doc.InsertParagraph(paraOne, false, paraFormat);

// Save to the output directory:
doc.Save();

// Open in Word:
Process.Start("WINWORD.EXE", fileName);
```

Workshop

- ให้ออกกรายงานของสินค้าจากร้านค้าที่ได้สร้างไว้เป็น Word และ Excel