



PDF Merge Library for .NET

The **PDF Merge** library for .NET can be linked into any .NET application, either ASP.NET web sites or Windows Forms applications, to add pdf merge capabilities to your application.

With Winnovative PDF Merge for .NET you can combine existing PDF documents, images and texts in a single PDF document.

If you need a ready to use tool for combing your documents and images you can use our command line utility. It allows you to specify the merge elements in any order with a simple syntax.

The integration is extremely easy and no additional installation is necessary in order to get started. It's just a .NET assembly that you have to reference in your application.

Features

- ✓ Easy integration, no installation or setup is necessary on the server
- ✓ Delivered as a single strong named .NET assembly (can be installed in GAC)
- ✓ Can be deployed on the server by simple copy (xcopy deployment support)
- ✓ Can be used from ASP.NET, Windows Forms, WPF, Web Service or Console applications
- ✓ The same assembly can be used both on 32-bit and 64-bit Windows servers
- ✓ Command line and Windows Forms samples with full C# source code
- ✓ Merge PDF documents from files and streams
- ✓ Get number of pages of a PDF document
- ✓ Specify the range of PDF pages to be merged from each PDF document
- ✓ Save merged document in memory or in file
- ✓ Merge PDF bookmarks
- ✓ Set compression level of the resulted PDF document
- ✓ Set merged document info (title, author, creation date)
- ✓ Password protect the merged document
- ✓ Set merged document rights for print, copy, assembling



[Getting Started](#)
[Code Sample](#)
[API Reference](#)
[Download](#)
[Buy Now](#)

Code Sample

```
1: PdfDocumentOptions pdfDocumentOptions =  
new PdfDocumentOptions();  
2:  
3: pdfDocumentOptions.PdfCompressionLevel = PDFCompressionLevel.Normal;  
4: pdfDocumentOptions.PdfPageSize = PdfPageSize.A4;  
5: pdfDocumentOptions.PdfPageOrientation = PDFPageOrientation.Portrait;  
6:  
7: PDFMerge pdfMerge = new PDFMerge(pdfDocumentOptions);  
8:  
9: pdfMerge.AppendPDFFile(pdfFilePath);  
10: pdfMerge.AppendImageFile(imageFilePath);  
11: pdfMerge.AppendTextFile(textFilePath);  
12: pdfMerge.AppendEmptyPage();  
13:  
14:  
15: pdfMerge.SaveMergedPDFToFile(outFile);
```