

# How to merge PDF documents from uploaded files for PDF merging API in VB.NET using PDF.co Web API

## How to merge PDF documents from uploaded files for PDF merging API in VB.NET: How To Tutorial

Here you may find thousands pre-made source code pieces for easy implementation in your own programming projects. PDF merging API in VB.NET can be implemented with PDF.co Web API. PDF.co Web API is the Rest API that provides set of data extraction functions, tools for documents manipulation, splitting and merging of pdf files. Includes built-in OCR, images recognition, can generate and read barcodes from images, scans and pdf.

VB.NET code snippet like this for PDF.co Web API works best when you need to quickly implement PDF merging API in your VB.NET application. Follow the instruction and copy - paste code for VB.NET into your project's code editor. Code testing will allow the function to be tested and work properly with your data.

Trial version of ByteScout is available for free download from our website. This and other source code samples for VB.NET and other programming languages are available.

VB.NET - Module1.vb

```
Imports System.IO
Imports System.Net
Imports Newtonsoft.Json.Linq

Module Module1

    ' The authentication key (API Key).
    ' Get your own by registering at https://app.pdf.co/documentation/api
    Const API_KEY As String = "*****"

    ' Source PDF files
    Dim SourceFiles as String() = { ".\sample1.pdf", ".\sample2.pdf" }
    ' Destination PDF file name
    const DestinationFile as string = ".\result.pdf"

    Sub Main()

        ' Create standard .NET web client instance
        Dim webClient As WebClient = New WebClient()

        ' Set API Key
        webClient.Headers.Add("x-api-key", API_KEY)
```

```

' 1. UPLOAD FILES TO CLOUD

Dim uploadedFiles As List(Of String) = New List(Of String)()

Try

    For Each pdfFile As String In SourceFiles

        ' 1a. RETRIEVE THE PRESIGNED URL TO UPLOAD THE FILE.

        ' Prepare URL for `Get Presigned URL` API call
        Dim query As string =
Uri.EscapeUriString(string.Format(
            "https://api.pdf.co/v1/file/upload/get-
presigned-url?contenttype=application/octet-stream&name={0}",
            Path.GetFileName(pdfFile)))

        ' Execute request
        Dim response As string =
webClient.DownloadString(query)

        ' Parse JSON response
        Dim json As JObject = JObject.Parse(response)

        If json("error").ToObject(Of Boolean) = False Then

            ' Get URL to use for the file upload
            Dim uploadUrl As string =
json("presignedUrl").ToString()

            ' Get URL of uploaded file to use with later
            Dim uploadedFileUrl As string =
json("url").ToString()

            ' 1b. UPLOAD THE FILE TO CLOUD.
            webClient.Headers.Add("content-type",
"application/octet-stream")
            webClient.UploadFile(uploadUrl, "PUT",
pdfFile) ' You can use UploadData() instead if your file is byte array or Stream
            uploadedFiles.Add(uploadedFileUrl)

        Else
            Console.WriteLine(json("message").ToString())
        End If

    Next

    if uploadedFiles.Count > 0 Then

        ' 2. MERGE UPLOADED PDF DOCUMENTS

        ' Prepare URL for `Merge PDF` API call
        Dim query As String =
Uri.EscapeUriString(String.Format(
            "https://api.pdf.co/v1/pdf/merge?name=
{0}&url={1}",
            Path.GetFileName(DestinationFile),
            string.Join(",", uploadedFiles)))

```

```

        ' Execute request
        Dim response As String =
webClient.DownloadString(query)

        ' Parse JSON response
        Dim json As JObject = JObject.Parse(response)

        If json("error").ToObject(Of Boolean) = False Then

            ' Get URL of generated PDF file
            Dim resultFileUrl As string =
json("url").ToString()

            ' Download PDF file
            webClient.DownloadFile(resultFileUrl,
DestinationFile)

            Console.WriteLine("Generated PDF file saved
as "{0}" file.", DestinationFile)

        Else
            Console.WriteLine(json("message").ToString())
        End If

    End If

    Catch ex As WebException
        Console.WriteLine(ex.ToString())
    End Try

    webClient.Dispose()

    Console.WriteLine()
    Console.WriteLine("Press any key...")
    Console.ReadKey()

End Sub

End Module

```

---

FOR MORE INFORMATION AND FREE TRIAL:

[Download Free Trial SDK \(on-premise version\)](#)

[Read more about PDF.co Web API](#)

[Explore documentation](#)

[Visit \[www.ByteScout.com\]\(http://www.ByteScout.com\)](#)

or

[Get Your Free API Key for \[www.PDF.co\]\(http://www.PDF.co\) Web API](#)