How to convert PDF to XLSX from uploaded file for PDF to excel API in VB.NET with ByteScout Cloud API Server

How to convert PDF to XLSX from uploaded file for PDF to excel API in VB.NET: Step By Step Instructions

These source code samples are listed and grouped by their programming language and functions they use. ByteScout Cloud API Server helps with PDF to excel API in VB.NET. ByteScout Cloud API Server is API server that is ready to use and can be installed and deployed in less than 30 minutes on your own Windows server or server in a cloud. It can save data and files on your local server-based file storage or in Amazon AWS S3 storage. Data is processed solely on the API server and is powered by ByteScout engine, no cloud services or Internet connection is required for data processing..

Want to learn quickly? These fast application programming interfaces of ByteScout Cloud API Server for VB.NET plus the instruction and the code below will help to learn how to convert PDF to XLSX from uploaded file. Open your VB.NET project and simply copy & paste the code and then run your app! Use of ByteScout Cloud API Server in VB.NET is also described in the documentation given along with the product.

Free! Free! Free! ByteScout free trial version is available for FREE download from our website. Programming tutorials along with source code samples are assembled.

FOR MORE INFORMATION AND FREE TRIAL:

Download Free Trial SDK (on-premise version)

Read more about ByteScout Cloud API Server

Explore API Documentation

Get Free Training for ByteScout Cloud API Server

Get Free API key for Web API

visit www.ByteScout.com

Source Code Files:

ByteScoutWebApiExample.sln

```
Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 15
VisualStudioVersion = 15.0.26730.10
MinimumVisualStudioVersion = 10.0.40219.1
Project("{F184B08F-C81C-45F6-A57F-5ABD9991F28F}") = "ByteScoutWebApiExample", "ByteScou
EndProject
Global
        GlobalSection(SolutionConfigurationPlatforms) = preSolution
                Debug|Any CPU = Debug|Any CPU
                Release|Any CPU = Release|Any CPU
        EndGlobalSection
        GlobalSection(ProjectConfigurationPlatforms) = postSolution
                {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Debug|Any CPU.ActiveCfg = Debug
                {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Debug|Any CPU.Build.0 = Debug|A
                {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Release|Any CPU.ActiveCfg = Rele
                {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Release|Any CPU.Build.0 = Releas
        EndGlobalSection
        GlobalSection(SolutionProperties) = preSolution
                HideSolutionNode = FALSE
        EndGlobalSection
        GlobalSection(ExtensibilityGlobals) = postSolution
                SolutionGuid = {4576C9BB-A42D-46A8-9198-7E2982E122FA}
        EndGlobalSection
EndGlobal
```

Module1.vb

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

' Please NOTE: In this sample we're assuming Cloud Api Server is hosted at "https://loc
' If it's not then please replace this with with your hosting url.

Module Module1

        ' Source PDF file
        const SourceFile as String = ".\sample.pdf"
        ' Comma-separated list of page indices (or ranges) to process. Leave empty for
        const Pages as String = ""
        ' PDF document password. Leave empty for unprotected documents.
        const Password As String = ""
```

```vbnet
        ' Destination XLSX file name
        const DestinationFile as String = ".\result.xlsx"

Sub Main()

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

        ' 1. RETRIEVE THE PRESIGNED URL TO UPLOAD THE FILE.
        ' * If you already have a direct file URL, skip to the step 3.

        ' Prepare URL for `Get Presigned URL` API call
        Dim query As string = Uri.EscapeUriString(string.Format(
                "https://localhost/file/upload/get-presigned-url?contenttype=ap
                Path.GetFileName(SourceFile)))

        Try
                ' 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 API calls
                        Dim uploadedFileUrl As string = json("url").ToString()

                        ' 2. UPLOAD THE FILE TO CLOUD.

                        webClient.Headers.Add("content-type", "application/octe
                        webClient.UploadFile(uploadUrl, "PUT", SourceFile) ' Yo

                        ' 3. CONVERT UPLOADED PDF FILE TO XLSX

                        ' Prepare URL for `PDF To XLSX` API call
                        query = Uri.EscapeUriString(String.Format(
                                "https://localhost/pdf/convert/to/xlsx?name={0}
                                Path.GetFileName(DestinationFile),
                                Password,
                                Pages,
                                uploadedFileUrl))

                        ' Execute request
                        response = webClient.DownloadString(query)

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

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

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

                                ' Download XLSX file
                                webClient.DownloadFile(resultFileUrl, Destinati

                                Console.WriteLine("Generated XLSX file saved as
```

```vbnet
                        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
```

packages.config

```xml
<?xml version="1.0" encoding="utf-8"?>
<packages>
  <package id="Newtonsoft.Json" version="10.0.3" targetFramework="net40" />
</packages>
```

VIDEO

https://www.youtube.com/watch?v=NEwNs2b9YN8

ON-PREMISE OFFLINE SDK

60 Day Free Trial or Visit ByteScout Cloud API Server Home Page

[Explore ByteScout Cloud API Server Documentation](#)
[Explore Samples](#)
[Sign Up for ByteScout Cloud API Server Online Training](#)

ON-DEMAND REST WEB API

[Get Your API Key](#)
[Explore Web API Docs](#)
[Explore Web API Samples](#)

[visit www.ByteScout.com](#)

[visit www.PDF.co](#)