

How to convert PDF to XLSX from URL asynchronously for PDF to excel API in VB.NET and ByteScout Cloud API Server

Learn to write code convert PDF to XLSX from URL asynchronously for PDF to excel API in VB.NET: Simple How To Tutorial

Every ByteScout tool includes sample VB.NET source codes that you can find here or in the folder with installed ByteScout product. 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..

VB.NET code snippet like this for ByteScout Cloud API Server works best when you need to quickly implement PDF to excel API in your VB.NET application. This sample code in VB.NET is all you need. Just copy-paste it to the code editor, then add a reference to ByteScout Cloud API Server and you are ready to try it! Enjoy writing a code with ready-to-use sample VB.NET codes to add PDF to excel API functions using ByteScout Cloud API Server in VB.NET.

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.

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](http://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", "ByteScoutWebApiExample.csproj", "{F184B08F-C81C-45F6-A57F-5ABD9991F28F}"
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|Any CPU
        {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Debug|Any CPU.Build.0 = Debug|Any CPU
        {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Release|Any CPU.ActiveCfg = Release|Any CPU
        {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Release|Any CPU.Build.0 = Release|Any CPU
    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 System.Threading
Imports Newtonsoft.Json.Linq

' Please NOTE: In this sample we're assuming Cloud Api Server is hosted at "https://localhost:5000/"
' If it's not then please replace this with your hosting url.

' Cloud API asynchronous "PDF To XLSX" job example.
' Allows to avoid timeout errors when processing huge or scanned PDF documents.

Module Module1

    ' Direct URL of source PDF file.
    Const SourceFileUrl As String = "https://bytescout-com.s3.amazonaws.com/files/example.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 = ""
' Destination XLSX file name
Const DestinationFile As String = ".\result.xlsx"
' (!) Make asynchronous job
Const Async As Boolean = True

```

```

Sub Main()

```

```

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

    ' Prepare URL for `PDF To XLSX` API call
    Dim query As String = Uri.EscapeUriString(String.Format(
        "https://localhost/pdf/convert/to/xlsx?name={0}&password={1}&pages={2}&sourceFileUrl={3}&async={4}",
        Path.GetFileName(DestinationFile),
        Password,
        Pages,
        SourceFileUrl,
        Async))

```

```

    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

```

```

            ' Asynchronous job ID
            Dim jobId As String = json("jobId").ToString()
            ' URL of generated XLSX file that will be available after
            Dim resultFileUrl As String = json("url").ToString()

```

```

            ' Check the job status in a loop.
            ' If you don't want to pause the main thread you can re
            ' to use a separate thread for the status checking and
            Do

```

```

                Dim status As String = CheckJobStatus(jobId) '

```

```

                ' Display timestamp and status (for demo purposes)
                Console.WriteLine(DateTime.Now.ToLongTimeString() & " Job Status: " & status)

```

```

                If status = "success" Then

```

```

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

                    Console.WriteLine("Generated XLSX file successfully")
                    Exit Do

```

```

                ElseIf status = "working" Then

```

```

                    ' Pause for a few seconds
                    Thread.Sleep(3000)

```

```

                Else
                    Console.WriteLine(status)
                    Exit Do
                End If
            Loop

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

    Function CheckJobStatus(jobId As String) As String

        Using webClient As WebClient = New WebClient()

            Dim url As String = "https://localhost/job/check?jobid=" + jobId

            Dim response As String = webClient.DownloadString(url)
            Dim json As JObject = JObject.Parse(response)

            Return Convert.ToString(json("status"))

        End Using

    End Function

End Module

```

packages.config

```

<?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](http://www.ByteScout.com)

[visit www.PDF.co](http://www.PDF.co)

www.bytescout.com