

How to convert PDF to JPEG from URL asynchronously for PDF to image API in VB.NET with ByteScout Cloud API Server

How to convert PDF to JPEG from URL asynchronously for PDF to image API in VB.NET: Step By Step Instructions

The documentation is written to assist you to apply all the necessary features on your side. PDF to image API in VB.NET can be applied with ByteScout Cloud API Server. ByteScout Cloud API Server is the ready to deploy Web API Server that can be deployed in less than thirty minutes into your own in-house Windows server (no Internet connection is required to process data!) or into private cloud server. Can store data on in-house local server based storage or in Amazon AWS S3 bucket. Processing data solely on the server using built-in ByteScout powered engine, no cloud services are used to process your data!.

This simple and easy to understand sample source code in VB.NET for ByteScout Cloud API Server contains different functions and options you should do calling the API to implement PDF to image API. 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! Want to see how it works with your data then code testing will allow the function to be tested and work properly.

Our website provides free trial version of ByteScout Cloud API Server that gives source code samples to assist with your VB.NET project.

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://lo
' If it's not then please replace this with your hosting url.

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

Module Module1

```
    ' Source PDF file
    Const SourceFileUrl As String = "https://bytescout-com.s3.amazonaws.com/files/o
    ' 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 = ""
' (!) 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 JPEG` API call
    Dim query As String = Uri.EscapeUriString(String.Format(
        "https://localhost/pdf/convert/to/jpg?password={0}&pages={1}&url={2}&password={3}",
        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 JSON file available after the job completion
            Dim resultJsonFileUrl As String = json("url").ToString()

            ' Check the job status in a loop.
            ' If you don't want to pause the main thread you can replace this loop
            ' to use a separate thread for the status checking and download
            Do
                Dim status As String = CheckJobStatus(jobId)

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

                If status = "success" Then

                    ' Download JSON file as string
                    Dim jsonFileString As String = webClient.DownloadString(resultJsonFileUrl)

                    Dim resultFileUrls As JArray = JArray.Parse(jsonFileString)

                    ' Download generated JPEG files
                    Dim page As Integer = 1
                    For Each token As JToken In resultFileUrls

                        Dim resultFileUrl As String = token.ToString()
                        Dim localFileName As String = "page" & page & ".jpg"

                        webClient.DownloadFile(resultFileUrl, localFileName)

                        Console.WriteLine("Downloaded " & localFileName)
                        page = page + 1
                    Next
                End If
            Loop While status < "success"
        End If
    Catch ex As Exception
        Console.WriteLine(ex.Message)
    End Try
End Sub

```

```

        Next

        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

```

```
<?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