

# How to convert PDF to text from URL asynchronously for PDF to text API in VB.NET using PDF.co Web API

How to convert PDF to text from URL asynchronously in VB.NET with easy ByteScout code samples to make PDF to text API. Step-by-step tutorial

Every ByteScout tool contains example VB.NET source codes that you can find here or in the folder with installed ByteScout product. PDF.co Web API was made to help with PDF to text API in VB.NET. PDF.co Web API is the flexible Web API that includes full set of functions from e-signature requests to data extraction, OCR, images recognition, pdf splitting and pdf splitting. Can also generate barcodes 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 to text API in your VB.NET application. Sample code in VB.NET is all you need. Copy-paste it to your the code editor, then add a reference to PDF.co Web API and you are ready to try it! Use of PDF.co Web API in VB.NET is also explained in the documentation included along with the product.

Our website provides free trial version of PDF.co Web API that includes source code samples to help with your VB.NET project.

VB.NET - Module1.vb

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

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

Module Module1

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

    ' Direct URL of source PDF file.
    Const SourceFileUrl As String = "https://bytescout-
com.s3.amazonaws.com/files/demo-files/cloud-api/pdf-to-text/sample.pdf"
    ' Comma-separated list of page indices (or ranges) to process. Leave empty
for all pages. Example: '0,2-5,7-'.
    Const Pages As String = ""
    ' PDF document password. Leave empty for unprotected documents.
    Const Password As String = ""
```

```

' Destination TXT file name
Const DestinationFile As String = ".\result.txt"
' (!) Make asynchronous job
Const Async As Boolean = True

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)

    ' Prepare URL for `PDF To TXT` API call
    Dim query As String = Uri.EscapeUriString(String.Format(
        "https://api.pdf.co/v1/pdf/convert/to/text?name={0}&password=
{1}&pages={2}&url={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 TXT file that will 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
            ' to use a separate thread for the status checking
            ' and completion.
            Do
                Dim status As String = CheckJobStatus(jobId)
                ' Possible statuses: "working", "failed", "aborted", "success".
                ' Display timestamp and status (for demo
                ' purposes)
                Console.WriteLine(DateTime.Now.ToLongTimeString() + ": " + status)

                If status = "success" Then

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

                    Console.WriteLine("Generated TXT file

```

```

saved as "{0}" file.", DestinationFile)
        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()
        ' Set API Key
        webClient.Headers.Add("x-api-key", API_KEY)

        Dim url As String = "https://api.pdf.co/v1/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

```



---

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](#)