

www.bytescout.com

parallel processing of pdf files with pdf extractor sdk in VB.NET using ByteScout Premium Suite

parallel processing of pdf files with pdf extractor sdk in VB.NET

Easy to understand coding instructions are written to assist you to try-out the features without the requirement to write your own code. ByteScout Premium Suite was made to help with parallel processing of pdf files with pdf extractor sdk in VB.NET. ByteScout Premium Suite is the set that includes 12 SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording.

VB.NET code snippet like this for ByteScout Premium Suite works best when you need to quickly implement parallel processing of pdf files with pdf extractor sdk in your VB.NET application. If you want to know how it works, then this VB.NET sample code should be copied and pasted into your application's code editor. Then just compile and run it. These VB.NET sample examples can be used in one or many applications.

If you want to try other samples for VB.NET then free trial version of ByteScout Premium Suite is available on our website.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout Premium Suite](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout Premium Suite](#)

[Get Free API key for Web API](#)

[visit www.Bytescout.com](http://www.Bytescout.com)

Source Code Files:

```
Imports System.IO
Imports System.Threading
Imports Bytescout.PDFExtractor

Module Module1

    ' Limit to 4 threads in queue.
    ' Set this value to number of your processor cores for max performance.
    ReadOnly ThreadLimiter As Semaphore = New Semaphore(4, 4)

    <MTAThread(>
    Sub Main()

        ' Get all PDF files in a folder
        Dim files = Directory.GetFiles("..\\..\\..\\..\\", "*.pdf")
        ' Array of events to wait
        Dim doneEvents(files.Length - 1) As ManualResetEvent

        For i As Integer = 0 To files.Length - 1

            ' Wait for the queue
            ThreadLimiter.WaitOne()

            ' Start thread with filename and event in params
            doneEvents(i) = New ManualResetEvent(False)
            Dim threadData As Object = New Object() {files(i), doneEvents(i)}
            ThreadPool.QueueUserWorkItem(AddressOf ConvertPdfToTxt, threadData)

        Next

        ' Wait until all threads finish
        WaitHandle.WaitAll(doneEvents)

        Console.WriteLine()
        Console.WriteLine("All is done.")
        Console.WriteLine()
        Console.WriteLine("Press any key to exit...")
        Console.ReadKey()

    End Sub

    Private Sub ConvertPdfToTxt(ByVal state As Object)

        ' Get filename and event from params
        Dim file As String = state(0)
        Dim doneEvent As ManualResetEvent = state(1)

        Dim resultFileName = Path.GetFileName(file) + ".txt"

        Try

            Console.WriteLine("Converting " + file)

            Using extractor As New TextExtractor("demo", "demo")
```

```
extractor.LoadDocumentFromFile(file)
extractor.SaveTextToFile(resultFileName)
```

```
End Using
```

```
Console.WriteLine("Finished " + resultFileName)
```

```
Finally
```

```
' Signal the thread is finished
doneEvent.Set()
' Release semaphore
ThreadLimiter.Release()
```

```
End Try
```

```
End Sub
```

```
End Module
```

VIDEO

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

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout Premium Suite Home Page](#)

[Explore ByteScout Premium Suite Documentation](#)

[Explore Samples](#)

[Sign Up for ByteScout Premium Suite 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)

