parallel barcode decoding with barcode reader sdk in VB.NET with ByteScout Barcode Suite

Learn parallel barcode decoding with barcode reader sdk in VB.NET

The documentation is crafted to assist you to apply the features on your side easily. ByteScout Barcode Suite was created to assist parallel barcode decoding with barcode reader sdk in VB.NET. ByteScout Barcode Suite is the bundle that privides 3 SDK products to generate barcodes (Barcode SDK), read barcodes (Barcode Reaer SDK) and read and write spreadsheets (Spreadsheet SDK).

Save time on writing and testing code by using the code below and use it in your application. To use parallel barcode decoding with barcode reader sdk in your VB.NET project or application just copy & paste the code and then run your app! Want to see how it works with your data then code testing will allow the function to be tested and work properly.

Our website gives free trial version of ByteScout Barcode Suite. It includes all these source code samples with the purpose to assist you with your VB.NET application implementation.

FOR MORE INFORMATION AND FREE TRIAL:

Download Free Trial SDK (on-premise version)

Read more about ByteScout Barcode Suite

Explore API Documentation

Get Free Training for ByteScout Barcode Suite

Get Free API key for Web API

visit www.ByteScout.com

Source Code Files:

Module1.vb

```vbnet
Imports Bytescout.BarCodeReader
Imports System.Threading

Module Module1

    Const InputFile As String = ".\example.pdf"

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


    <MTAThread()> Sub Main()

        Const numberOfRuns As Integer = 10
        Dim doneEvents(numberOfRuns - 1) As ManualResetEvent

        For i As Integer = 0 To numberOfRuns - 1

            ' Wait for the queue
            ThreadLimiter.WaitOne()

            doneEvents(i) = New ManualResetEvent(False)

            ThreadPool.QueueUserWorkItem(New WaitCallback(AddressOf ThreadProc), New Ob

        Next

        WaitHandle.WaitAll(doneEvents)

        Console.WriteLine("All threads done.")
        Console.WriteLine()

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

    End Sub

    Sub ThreadProc(ByVal stateInfo As Object)

        Dim threadIndex As Integer = stateInfo(0)
        Dim waitEvent As ManualResetEvent = stateInfo(1)

        Console.WriteLine("Thread #" & threadIndex & " started...")

        Try

            Dim reader As New Reader()
            reader.RegistrationName = "demo"
            reader.RegistrationKey = "demo"

            ' Set barcode type to find
            reader.BarcodeTypesToFind.Code128 = True

            ' ----------------------------------------------------------------------------
            ' NOTE: We can read barcodes from specific page to increase performance .
            ' For sample please refer to "Decoding barcodes from PDF by pages" program
            ' ----------------------------------------------------------------------------
```

```vbnet
            ' Read barcodes
            Dim barcodes As FoundBarcode() = reader.ReadFrom(InputFile)

            ' Cleanup
            reader.Dispose()

            Console.WriteLine("Thread #" & threadIndex & " finished with " & barcodes.

        Catch ex As Exception

            Console.WriteLine("Thread #" & threadIndex & " failed with exception:\r\n"

        Finally

            ' Signal the thread is finished
            waitEvent.Set()

            ' Release semaphore
            ThreadLimiter.Release()

        End Try

    End Sub

End Module
```

VIDEO

ON-PREMISE OFFLINE SDK

60 Day Free Trial or Visit ByteScout Barcode Suite Home Page
Explore ByteScout Barcode Suite Documentation
Explore Samples
Sign Up for ByteScout Barcode Suite 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