

How to find text in pdf with smart match with pdf extractor sdk in VB.NET using ByteScout Premium Suite

If you want to learn more then this tutorial will show how to find text in pdf with smart match with pdf extractor sdk in VB.NET

ByteScout simple and easy to understand tutorials are planned to describe the code for both VB.NET beginners and advanced programmers. ByteScout Premium Suite is the bundle that includes twelve SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording. It can be applied to find text in pdf with smart match with pdf extractor sdk using VB.NET.

These VB.NET code samples for VB.NET guide developers to speed up coding of the application when using ByteScout Premium Suite. Just copy and paste the code into your VB.NET application's code and follow the instructions. Check VB.NET sample code samples to see if they respond to your needs and requirements for the project.

Trial version of ByteScout Premium Suite is available for free. Source code samples are included to help you with your VB.NET app.

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 Bytescout.PDFExtractor

Class Program

    Friend Shared Sub Main(args As String())

        Dim extractor As TextExtractor = New TextExtractor("demo", "demo")

        ' Load the document
        extractor.LoadDocumentFromFile("sample2.pdf")

        ' Smart match the search string like Adobe Reader
        extractor.WordMatchingMode = WordMatchingMode.SmartMatch

        Dim searchString As String = "land"

        ' Get page count
        Dim pageCount As Integer = extractor.GetPageCount()

        ' Iterate through pages
        For i As Integer = 0 To pageCount - 1

            ' Search through page
            If extractor.Find(i, searchString, False) Then

                Do

                    ' Output search results
                    Console.WriteLine("Found on page " + i.ToString() + " at location ")

                    ' Now we are getting the found text
                    Dim extractedString As String = extractor.FoundText.Text
                    Console.WriteLine("Found text: " + extractedString)

                    Loop While extractor.FindNext() ' Search next occurrence of the search

                End If

            Next

            ' Cleanup
            extractor.Dispose()

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

        End Sub

    End Class
```

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)

www.bytescout.com