

fonts available for cells with spreadsheet sdk in VB.NET and ByteScout Data Extraction Suite

Simple tutorial on how to do fonts available for cells with spreadsheet 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 Data Extraction Suite was created to assist fonts available for cells with spreadsheet sdk in VB.NET. ByteScout Data Extraction Suite is the bundle that includes three SDK tools for data extraction from PDF, scans, images and from spreadsheets: PDF Extractor SDK, Data Extraction SDK, Barcode Reader SDK.

The below SDK samples describe how to quickly make your application do fonts available for cells with spreadsheet sdk in VB.NET with the help of ByteScout Data Extraction Suite. Follow the steps-by-step instructions from the scratch to work and copy and paste code for VB.NET into your editor. Updated and detailed documentation and tutorials are available along with installed ByteScout Data Extraction Suite if you'd like to learn more about the topic and the details of the API.

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

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout Data Extraction Suite](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout Data Extraction Suite](#)

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

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

Source Code Files:

```
Imports Bytescout.Spreadsheet
Imports System.IO

Module Module1

    Sub Main()
        ' Create new Spreadsheet
        Dim document As New Spreadsheet()

        ' Add new worksheet
        Dim worksheet As Worksheet = document.Workbook.Worksheets.Add("HelloWorld")

        ' Create array with font names
        Dim fontNames() As String = {"Helvetica", "Times New Roman", "Verdana", "Times"}

        ' Use all fonts in fontNames array
        For i As Integer = 0 To fontNames.Length - 1
            ' Set font size based on loop counter
            Dim fontSize As Double = 10 + i * 3

            'Set cell font type and font size
            worksheet.Cell(i, 0).Font = New System.Drawing.Font(fontNames(i), fontSize)

            ' Set cell value
            worksheet.Cell(i, 0).Value = fontNames(i)
        Next

        ' remove output file if already exists
        If File.Exists("Output.xls") Then
            File.Delete("Output.xls")
        End If

        ' Save document
        document.SaveAs("Output.xls")

        ' Close Spreadsheet
        document.Close()

        ' open in default spreadsheets viewer/editor
        Process.Start("Output.xls")

    End Sub

End Module
```

VIDEO

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

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout Data Extraction Suite Home Page](#)
[Explore ByteScout Data Extraction Suite Documentation](#)
[Explore Samples](#)
[Sign Up for ByteScout Data Extraction 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