

How to use rich formatting to create report with spreadsheet sdk in VB.NET with ByteScout Data Extraction Suite

This code in VB.NET shows how to use rich formatting to create report with spreadsheet sdk with this how to tutorial

The sample source code below will teach you how to use rich formatting to create report with spreadsheet sdk in VB.NET. ByteScout Data Extraction Suite: the set that includes 3 SDK products for data extraction from PDF, scans, images and from spreadsheets: PDF Extractor SDK, Data Extraction SDK, Barcode Reader SDK. It can use rich formatting to create report with spreadsheet sdk in VB.NET.

The following code snippet for ByteScout Data Extraction Suite works best when you need to quickly use rich formatting to create report with spreadsheet sdk in your VB.NET application. IF you want to implement the functionality, just copy and paste this code for VB.NET below into your code editor with your app, compile and run your application. Check VB.NET sample code samples to see if they respond to your needs and requirements for the project.

ByteScout Data Extraction Suite free trial version is available on our website. VB.NET and other programming languages are supported.

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 Bytescout.Spreadsheet.Constants
Imports System.IO
Imports System.Diagnostics
Imports System.Drawing

Module Module1

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

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

        ' Set Values
        worksheet.Cell(0, 0).Value = "Name"
        worksheet.Cell(0, 1).Value = "Full Name"

        Dim headerColor As Color = [Color].FromArgb(75, 172, 198)
        Dim contentColor As Color = [Color].FromArgb(141, 180, 227)

        ' Set font bold for headers
        worksheet.Cell(0, 0).Font = New Font("Arial", 12, FontStyle.Bold)
        worksheet.Cell(0, 1).Font = New Font("Arial", 12, FontStyle.Bold)

        ' Set back color for headers rows
        worksheet.Cell(0, 0).FillPattern = PatternStyle.Solid
        worksheet.Cell(0, 0).FillPatternForeColor = headerColor
        worksheet.Cell(0, 1).FillPattern = PatternStyle.Solid
        worksheet.Cell(0, 1).FillPatternForeColor = headerColor
        worksheet.Cell(6, 0).FillPattern = PatternStyle.Solid
        worksheet.Cell(6, 0).FillPatternForeColor = headerColor
        worksheet.Cell(6, 0).FontColor = Color.White
        worksheet.Cell(6, 1).FillPattern = PatternStyle.Solid
        worksheet.Cell(6, 1).FillPatternForeColor = headerColor

        ' Set back color for content rows
        For i As Integer = 1 To 5
            worksheet.Cell(i, 0).FillPattern = PatternStyle.Solid
            worksheet.Cell(i, 0).FillPatternForeColor = contentColor
            worksheet.Cell(i, 1).FillPattern = PatternStyle.Solid
            worksheet.Cell(i, 1).FillPatternForeColor = contentColor
        Next

        worksheet.Cell(1, 0).Value = "Homer"
        worksheet.Cell(1, 1).Value = "Homer Jay Simpson"

        worksheet.Cell(2, 0).Value = "Marjorie"
        worksheet.Cell(2, 1).Value = "Marjorie Marge Simpson (Bouvier)"

        worksheet.Cell(3, 0).Value = "Bartholomew"
        worksheet.Cell(3, 1).Value = "Bartholomew Jojo Bart Simpson"

        worksheet.Cell(4, 0).Value = "Lisa"
```

```
worksheet.Cell(4, 1).Value = "Lisa Marie Simpson"

worksheet.Cell(5, 0).Value = "Margaret"
worksheet.Cell(5, 1).Value = "Margaret Maggie "

worksheet.Cell(6, 0).Value = "Count"
worksheet.Cell(6, 1).Value = "5"

worksheet.Columns(1).Width = 250
worksheet.Columns(2).Width = 250

' 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](#)

[visit www.PDF.co](#)

[www.bytescout.com](#)