How to add surface chart with spreadsheet sdk in VB.NET and ByteScout Barcode Suite

If you want to learn more then this tutorial will show how to add surface chart with spreadsheet sdk in VB.NET

An easy to understand guide on how to add surface chart with spreadsheet sdk in VB.NET with this source code sample. Want to add surface chart with spreadsheet sdk in your VB.NET app? ByteScout Barcode Suite is designed for it. ByteScout Barcode Suite is the set that includes three different SDK products to generate barcodes, read barcodes and read and write spreadsheets: Barcode SDK, Barcode Reader SDK and Spreadsheet SDK.

This prolific sample source code in VB.NET for ByteScout Barcode Suite contains various functions and other necessary options you should do calling the API to add surface chart with spreadsheet sdk. Just copy and paste the code into your VB.NET application's code and follow the instructions. Applying VB.NET application mostly includes various stages of the software development so even if the functionality works please test it with your data and the production environment.

Our website gives trial version of ByteScout Barcode Suite for free. It also includes documentation and source code samples.

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.Spreadsheet
Imports Bytescout.Spreadsheet.Charts


Module Module1

    Sub Main()

        ' Create new Spreadsheet object
        Dim spreadsheet As New Spreadsheet()
        spreadsheet.RegistrationName = "demo"
        spreadsheet.RegistrationKey = "demo"

        ' Add new worksheet
        Dim sheet As Worksheet = spreadsheet.Workbook.Worksheets.Add("Sample")

        ' Add a data for Surface chart
        Dim count As Integer = 10
        Dim [step] As Double = 2 * Math.PI / (count - 1)
        For i As Integer = 0 To count - 1
            For j As Integer = 0 To count - 1
                sheet(i, j).Value = Math.Sin([step] * Math.Sqrt(i * i + j * j))
            Next
        Next

        ' Add charts to worksheet
        Dim surfaceChart As Chart = sheet.Charts.AddChartAndFitInto(11, 1, 28, 8, Char
        For i As Integer = 0 To count - 1
            surfaceChart.SeriesCollection.Add(New Series(sheet.Range(0, i, 9, i)))
        Next

        surfaceChart = sheet.Charts.AddChartAndFitInto(11, 10, 28, 17, ChartType.Surfac
        For i As Integer = 0 To count - 1
            surfaceChart.SeriesCollection.Add(New Series(sheet.Range(0, i, 9, i)))
        Next

        surfaceChart = sheet.Charts.AddChartAndFitInto(29, 1, 46, 8, ChartType.SurfaceT
        For i As Integer = 0 To count - 1
            surfaceChart.SeriesCollection.Add(New Series(sheet.Range(0, i, 9, i)))
        Next

        surfaceChart = sheet.Charts.AddChartAndFitInto(29, 10, 46, 17, ChartType.Surfac
        For i As Integer = 0 To count - 1
            surfaceChart.SeriesCollection.Add(New Series(sheet.Range(0, i, 9, i)))
        Next

        ' Save it as XLS
        spreadsheet.SaveAs("Output.xls")

        ' Close the document
        spreadsheet.Close()

        ' Cleanup
```

```
        spreadsheet.Dispose()

        ' Open generated XLS file in default associated application
        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 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