www.bytescout.com

number format in cells with spreadsheet sdk in VB.NET using ByteScout Data Extraction Suite

Learn to code number format in cells with spreadsheet sdk in VB.NET: How-To tutorial

An easy to understand guide to learn how to number format in cells with spreadsheet sdk in VB.NET. ByteScout Data Extraction Suite was made to help with number format in cells with spreadsheet sdk in VB.NET. ByteScout Data Extraction Suite is 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.

VB.NET code snippet like this for ByteScout Data Extraction Suite works best when you need to quickly implement number format in cells with spreadsheet sdk in your VB.NET application. If you want to implement this functionality, you should copy and paste code below into your app using code editor. Then compile and run your application. This basic programming language sample code for VB.NET will do the whole work for you in implementing number format in cells with spreadsheet sdk in your app.

Our website gives free trial version of ByteScout Data Extraction 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 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

Source Code Files:

Module1.vb

```
Imports Bytescout.Spreadsheet
Imports System.IO
Imports System.Diagnostics

Module Module1

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

        ' Get worksheet by name
        Dim worksheet As Worksheet = document.Workbook.Worksheets.Add("Sample")

        ' Set header
        worksheet.Cell(0, 0).Value = "Number"
        worksheet.Cell(0, 1).Value = "Format"

        ' Set variables
        Dim plan As Double = 3000
        Dim result As Double = 3100

        ' Fill cells
        For i As Integer = 1 To 5
            worksheet.Cell(i, 0).Value = result / plan
        Next

        ' Set cells format
        worksheet.Cell(1, 0).NumberFormatString = "0.00"
        worksheet.Cell(2, 0).NumberFormatString = "0.00%"
        worksheet.Cell(3, 0).NumberFormatString = "mm:ss"
        worksheet.Cell(4, 0).NumberFormatString = "hh:mm:ss"
        worksheet.Cell(5, 0).NumberFormatString = "##0,0E+0"

        ' Set captions
        worksheet.Cell(1, 1).Value = "0.00"
        worksheet.Cell(2, 1).Value = "0.00%"
        worksheet.Cell(3, 1).Value = "mm:ss"
        worksheet.Cell(4, 1).Value = "hh:mm:ss"
        worksheet.Cell(5, 1).Value = "##0,0E+0"


        ' 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