

number format in cells in VB.NET using ByteScout Spreadsheet SDK

number format in cells in VB.NET

Sample source codes below will show you how to cope with a difficult task, for example, number format in cells in VB.NET. ByteScout Spreadsheet SDK was made to help with number format in cells in VB.NET. ByteScout Spreadsheet SDK is the library (SDK) that is capable of writing, reading, modifying and calculating Excel and CSV spreadsheets. Most popular formulas can be calculated and recalculated with Excel installed. You may import or export data to and from CSV, XML, JSON as well as to and from databases, arrays.

This rich sample source code in VB.NET for ByteScout Spreadsheet SDK includes the number of functions and options you should do calling the API to implement number format in cells. In order 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 in your app.

Free trial version of ByteScout Spreadsheet SDK is available on our website. Get it to try other samples for VB.NET.

VB.NET - 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
```

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout Spreadsheet SDK](#)

[Explore documentation](#)

[Visit www.ByteScout.com](http://www.ByteScout.com)

or

[Get Your Free API Key for www.PDF.co Web API](#)