

How to read XML spreadsheet (spreadsheetml format) in VB.NET and ByteScout Spreadsheet SDK

This code in VB.NET shows how to read XML spreadsheet (spreadsheetml format) with this how to tutorial

Read XML spreadsheet (spreadsheetml format) is easy to implement in VB.NET if you use these source codes below. 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 and you can use it to read XML spreadsheet (spreadsheetml format) with VB.NET.

VB.NET code samples for VB.NET developers help to speed up coding of your application when using ByteScout Spreadsheet SDK. This VB.NET sample code is all you need for your app. Just copy and paste the code, add references (if needs to) and you are all set! Implementing VB.NET application typically includes multiple stages of the software development so even if the functionality works please test it with your data and the production environment.

ByteScout Spreadsheet SDK free trial version is available on our website. VB.NET and other programming languages are supported.

VB.NET - Program.vb

```
Imports Bytescout.Spreadsheet

Module Program

    Sub Main()

        Try

            ' Convert SpreadhseetML (xml spreadsheet) to XLSX format
            Using spreadsheet As New Spreadsheet

                spreadsheet.RegistrationName = "demo"
                spreadsheet.RegistrationKey = "demo"

                ' Load input file
                spreadsheet.LoadFromFile("sample.xml")

                ' Generate XLSX format
                spreadsheet.SaveAsXLSX("sample.xlsx")
            End Using

            ' Read spreadsheet and display data to console
```

Using spreadsheet As New Spreadsheet

```
spreadsheet.RegistrationName = "demo"
spreadsheet.RegistrationKey = "demo"

' Load input file
spreadsheet.LoadFromFile("sample.xlsx")

' Reference Worksheet
Dim Worksheet As Worksheet = spreadsheet.Worksheet(0)

Console.WriteLine("=====")
Console.WriteLine("===== Spreadsheet data
=====")

Console.WriteLine("=====")

' Read And output cells to console.
For row = Worksheet.UsedRangeRowMin To Worksheet.UsedRangeRowMax

    For column = Worksheet.UsedRangeColumnMin To
Worksheet.UsedRangeColumnMax

        Console.Write("{0}" + vbTab, Worksheet.Cell(row,
column).ValueAsString)

    Next

    Console.WriteLine()

Next

End Using

Catch ex As Exception
    Console.WriteLine(("ERROR:" + ex.Message))
End Try

Console.WriteLine("Press any key to exit...")
Console.ReadLine()

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](#)