

How to add rows into existing spreadsheet in VB.NET with ByteScout Spreadsheet SDK

Write code in VB.NET to add rows into existing spreadsheet with this step-by-step tutorial

The sample source codes on this page shows how to add rows into existing spreadsheet in VB.NET. What is ByteScout Spreadsheet SDK? It is the SDK to create, read, modify and calculate spreadsheets. Formula calculations are supported, import and export to and from JSON, CSV, XML, databases, arrays. It can help you to add rows into existing spreadsheet in your VB.NET application.

You will save a lot of time on writing and testing code as you may just take the VB.NET code from ByteScout Spreadsheet SDK for add rows into existing spreadsheet below and use it in your application. In order to implement the functionality, you should copy and paste this code for VB.NET below into your code editor with your app, compile and run your application. Enjoy writing a code with ready-to-use sample VB.NET codes.

You can download free trial version of ByteScout Spreadsheet SDK from our website to see and try many others source code samples for VB.NET.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout Spreadsheet SDK](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout Spreadsheet SDK](#)

[Get Free API key for Web API](#)

[visit www.Bytescout.com](http://www.Bytescout.com)

Source Code Files:

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

Module Module1

    Sub Main()
        ' Open Spreadsheet
        Dim document As New Spreadsheet()
        document.LoadFromFile("AdvancedReport.xls")

        ' Get Worksheet
        Dim worksheet As Worksheet = document.Workbook.Worksheets(0)

        ' Add new row
        worksheet.Rows.Insert(6, 1)

        ' Set values
        worksheet.Rows(6)(0).Value = "New Name"
        worksheet.Rows(6)(1).Value = "New Full Name"

        ' 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=nm_7I0PN1TY

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout Spreadsheet SDK Home Page](#)
[Explore ByteScout Spreadsheet SDK Documentation](#)
[Explore Samples](#)
[Sign Up for ByteScout Spreadsheet SDK Online Training](#)

ON-DEMAND REST WEB API

[Get Your API Key](#)
[Explore Web API Docs](#)
[Explore Web API Samples](#)

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

[visit www.PDF.co](http://www.PDF.co)

www.bytescout.com