

How to add images to spreadsheet in VBScript using ByteScout Spreadsheet SDK

The tutorial shows how to add images to spreadsheet in VBScript

ByteScout tutorials are designed to explain the code for both VBScript beginners and advanced programmers. ByteScout Spreadsheet SDK: the SDK that can write and read, modify and calculate Excel and CSV spreadsheets. Most popular formulas are supported. You may import or export data to and from CSV, XML, JSON as well as to and from databases, arrays. It can add images to spreadsheet in VBScript.

This rich sample source code in VBScript for ByteScout Spreadsheet SDK includes the number of functions and options you should do calling the API to add images to spreadsheet. In order to implement the functionality, you should copy and paste this code for VBScript below into your code editor with your app, compile and run your application. This basic programming language sample code for VBScript will do the whole work for you to add images to spreadsheet.

Our website provides trial version of ByteScout Spreadsheet SDK 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 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:

```
Set document = CreateObject("Bytestout.Spreadsheet.Spreadsheet")

document.RegistrationName = "demo"
document.RegistrationKey = "demo"

' Add new worksheet
Set worksheet = document.Workbook.Worksheets.Add("HelloWorld")

Set pictures = worksheet.Pictures

' add a picture to specified cell
pictures.Add 1, 1, "image1.jpg"

' add a picture by absolute pixel coordinates
pictures.Add_3 "image2.jpg", 550, 100

' fit picture into cell range
Set picture = pictures.AddAndFitInto(27, 1, 32, 5, "image1.jpg")

' rotate the last added picture
picture.Rotation = 45

' delete output file if exists already
Set fso = CreateObject("Scripting.FileSystemObject")
If (fso.FileExists("Output.xls")) Then fso.DeleteFile("Output.xls")
Set fso = nothing

' save document
document.SaveAs "Output.xls"

' close Spreadsheet
Set document = Nothing
```

VIDEO

https://www.youtube.com/watch?v=nm_7I0PN1TY

[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