

How to create simple spreadsheet in C++ (unmanaged) using ByteScout Spreadsheet SDK

This tutorial will show how to create simple spreadsheet in C++ (unmanaged)

With this source code sample you may quickly learn how to create simple spreadsheet in C++ (unmanaged). ByteScout Spreadsheet SDK can create simple spreadsheet. It can be used from C++ (unmanaged). ByteScout Spreadsheet SDK is the SDK component for writing, reading, modifying and calculating Excel and CSV spreadsheets. Can calculate and recalculate formulas with Excel installed. You may import or export data to and from CSV, XML, JSON. Supports export to databases, arrays, streams.

C++ (unmanaged) code samples for C++ (unmanaged) developers help to speed up coding of your application when using ByteScout Spreadsheet SDK. In your C++ (unmanaged) project or application you may simply copy & paste the code and then run your app! Implementing C++ (unmanaged) 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. C++ (unmanaged) and other programming languages are supported.

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:

HelloWorld.cpp

```
// IMPORTANT: Copy ByteScout.Spreadsheet.tlb into the project folder
// HelloWorld.cpp : Defines the entry point for the console application.
//

#include "stdafx.h"
#include <windows.h>

#pragma warning (disable: 4278)

#import "c:\windows\system32\stdole2.tlb" rename_namespace("BytescoutSpreadsheet") exc
"OLE_XPOS_PIXELS", "OLE_YPOS_PIXELS", "OLE_XSIZE_PIXELS", "OLE_YSIZE_PIXELS",
"OLE_YPOS_HIMETRIC", "OLE_XSIZE_HIMETRIC", "OLE_YSIZE_HIMETRIC", "OLE_XPOS_CON
"OLE_YPOS_CONTAINER", "OLE_XSIZE_CONTAINER", "OLE_YSIZE_CONTAINER", "OLE_HANDL
"OLE_CANCELBOOL", "OLE_ENABLEDEFAULTBOOL", "FONTSIZE", "OLE_COLOR")

// To use managed-code servers like the C# server,
// we have to import the common language runtime:
#import <microsoft.tlb> raw_interfaces_only
#import <System.tlb> raw_interfaces_only
#import <System.Drawing.tlb> raw_interfaces_only

#import "Bytescout.Spreadsheet.tlb" rename_namespace("BytescoutSpreadsheet") //no_name
int main(int argc, char* argv[])
{
    // initialize OLE
    HRESULT hr = CoInitialize(NULL);

    // check for errors
    if (FAILED(hr)) {
        MessageBox(0,"OLE initialization errp","error",MB_OK);
        return -1;
    }
    // declare document object

    ISpreadsheet * Doc = NULL;
    CLSID clsid;
        _Worksheets* worksheets = NULL;
        _Worksheet* worksheet = NULL;
        _Cell* cell = NULL;

    // get inuque ID for ISpreadsheet interface
    hr = CLSIDFromProgID(OLESTR("Bytescout.Spreadsheet.Spreadsheet"), &clsid);
    // check for errors
    if (FAILED(hr)) {
        MessageBox(0,"Can't get CLSID for ISpreadsheet interface","error",MB_OK);
        goto Uninit;
    };
    // create Spreadsheet object
    hr = CoCreateInstance(clsid, NULL, CLSCTX_INPROC_SERVER, __uuidof(ISpreadsheet)
    // check for errors
    if (FAILED(hr)) {
        MessageBox(0,"Can't create Spreadsheet object","error",MB_OK);
        goto Uninit;
    }
}
```

```

}

//IWorksheets* worksheets = NULL;

Doc->get_Worksheets(&worksheets);

worksheet = reinterpret_cast<IWorksheets*>(worksheets)->Add("Sheet 1");

cell = reinterpret_cast<IWorksheet*>(worksheet)->Cell("A1");
reinterpret_cast<ICell*>(cell)->PutValueAsHTML("test value");

Doc->SaveAs("output.xls");

// release PDFDoc object
Doc->Release();

// uninitialized OLE libraries
Uninit:
CoUninitialize();
return 0;
}

```

StdAfx.cpp

```

// stdafx.cpp : source file that includes just the standard includes
//      HelloWorld.pch will be the pre-compiled header
//      stdafx.obj will contain the pre-compiled type information

#include "stdafx.h"

// TODO: reference any additional headers you need in STDAFX.H
// and not in this file

```

StdAfx.h

```
// stdafx.h : include file for standard system include files,
// or project specific include files that are used frequently, but
//         are changed infrequently
//

#if !defined(AFX_STDAFX_H__59C6B829_CE1C_476C_B7F7_AADB9A4AC838__INCLUDED_)
#define AFX_STDAFX_H__59C6B829_CE1C_476C_B7F7_AADB9A4AC838__INCLUDED_

#if _MSC_VER > 1000
#pragma once
#endif // _MSC_VER > 1000

#define WIN32_LEAN_AND_MEAN           // Exclude rarely-used stuff from Windows headers

#include <stdio.h>

// TODO: reference additional headers your program requires here

//{{AFX_INSERT_LOCATION}}
// Microsoft Visual C++ will insert additional declarations immediately before the previous line.

#endif // !defined(AFX_STDAFX_H__59C6B829_CE1C_476C_B7F7_AADB9A4AC838__INCLUDED_)
```

VIDEO

https://www.youtube.com/watch?v=nm_7IOPN1TY

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