

How to extract text from PDF in C++ using ByteScout PDF Extractor SDK

This code in C++ shows how to extract text from PDF with this how to tutorial

We made thousands of pre-made source code pieces for easy implementation in your own programming projects. ByteScout PDF Extractor SDK is the SDK is designed to help developers with pdf tables and pdf data extraction from unstructured documents like pdf, tiff, scans, images, scanned and electronic forms. The library is powered by OCR, computer vision and AI to provide unique functionality like table detection, automatic table structure extraction, data restoration, data restructuring and reconstruction. Supports PDF, TIFF, PNG, JPG images as input and can output CSV, XML, JSON formatted data. Includes full set of utilities like pdf splitter, pdf merger, searchable pdf maker and other utilities. It can extract text from PDF in C++.

The SDK samples like this one below explain how to quickly make your application do extract text from PDF in C++ with the help of ByteScout PDF Extractor SDK. In order to implement the functionality, you should copy and paste this code for C++ below into your code editor with your app, compile and run your application. Further enhancement of the code will make it more vigorous.

Trial version of ByteScout PDF Extractor SDK is available for free. Source code samples are included to help you with your C++ app.

C++ - CPPExample.cpp

```
#include "stdafx.h"
#include "comip.h"

#import "c:\\Program Files\\Bytescout PDF Extractor
SDK\\net4.00\\Bytescout.PDFExtractor.tlb" raw_interfaces_only

using namespace Bytescout_PDFExtractor;

int _tmain(int argc, _TCHAR* argv[])
{
    // Initialize COM.
    HRESULT hr = CoInitializeEx(NULL, COINIT_APARTMENTTHREADED);

    // Create the interface pointer.
    _TextExtractorPtr pITextExtractor(__uuidof(TextExtractor));

    // Set the registration name and key
    // Note: You should use _bstr_t or BSTR to pass string to the library
    // because of COM requirements
    _bstr_t bstrRegName(L"DEMO");
    pITextExtractor->put_RegistrationName(bstrRegName);
}
```

```

    _bstr_t bstrRegKey(L"DEMO");
    pITextExtractor->put_RegistrationKey(bstrRegKey);

    // Load sample PDF document
    _bstr_t bstrPath(L"..\\..\\sample3.pdf");
    pITextExtractor->LoadDocumentFromFile(bstrPath);

    // Save extracted text to file
    _bstr_t bstrOutputFile(L"output.txt");
    pITextExtractor->SaveTextToFile(bstrOutputFile);

    pITextExtractor->Release();

    CoUninitialize();

    return 0;
}

```

C++ - stdafx.cpp

```

// stdafx.cpp : source file that includes just the standard includes
// CPPEXAMPLE.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

```

C++ - stdafx.h

```

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

#pragma once

```

```
#include "targetver.h"

#include
#include

// TODO: reference additional headers your program requires here
```

C++ - targetver.h

```
#pragma once

// Including SDKDDKVer.h defines the highest available Windows platform.

// If you wish to build your application for a previous Windows platform, include
WinSDKVer.h and
// set the _WIN32_WINNT macro to the platform you wish to support before including
SDKDDKVer.h.

#include
```

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout PDF Extractor SDK](#)

[Explore documentation](#)

[Visit www.ByteScout.com](#)

or

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