Foxit PDF SDK

DEVELOPER GUIDE
FOXIT PDF CONVERSION SDK

For .NET Core

©Foxit Software Incorporated. All Rights Reserved.

TABLE OF CONTENTS

1 Introduction to Foxit PDF Conversion SDK 1

1.1 Why Choose Foxit PDF Conversion SDK 1

1.2 Foxit PDF Conversion SDK for .NET Core 1

1.3 Evaluation 2

1.4 License 2

1.5 About this guide 2

2 Getting Started 3

2.1 System Requirements 3

2.2 What is in the package 3

2.3 How to run a demo 4

2.4 How to initialize Foxit PDF Conversion SDK 4

Support 6

1 INTRODUCTION TO FOXIT PDF CONVERSION SDK

Foxit PDF Conversion SDK is a flexible high-performance library to convert PDF files to MS office suite formats while maintaining the layout and format of your original documents. It offers advanced conversion capabilities as well as the same quality and user experience you can expect from Foxit. The engine for creating MS office files is developed independently by Foxit and no 3rd party office engine is required to be pre-installed on your machine. Foxit PDF Conversion SDK can be used as a standalone library or as an add-on with Foxit PDF SDK.

1.1 Why Choose Foxit PDF Conversion SDK

Foxit is a leading software provider of solutions for reading, editing, creating, organizing, and securing PDF documents. Foxit PDF SDK libraries have been used in many of today's leading apps, and are proven, robust, and battle-tested to provide the quality, performance, and features that the industry's largest apps demand. Foxit PDF Conversion SDK is a new SDK product independently developed by Foxit. Customers choose Foxit PDF Conversion SDK for the following reasons:

- **Easy to integrate**

Developers can seamlessly integrate Foxit PDF Conversion SDK into their own applications.

- **Lightweight footprint**

Does not exhaust system resource and deploys quickly.

- **Independently-developed engine**

The core technology of the Conversion SDK is developed independently by Foxit.

- **Premium World-side Support**

Foxit offers premium support for its developer products because when you are developing mission critical products you need the best support. Foxit has one of the PDF industry's largest team of support engineers. Updates are released on a regular basis to improve user experience by adding new features and enhancements.

1.2 Foxit PDF Conversion SDK for .NET Core

In this guide, we focus on the introduction of Foxit PDF Conversion SDK for .NET Core on Windows platform.

.NET Core is an open-source, general-purpose development platform maintained by Microsoft and the .NET community on GitHub. Foxit PDF Conversion SDK for .NET Core ships with simple-to-use

APIs that can help .NET Core developers convert PDFs to office files (including Word/Excel/PowerPoint).

1.3 Evaluation

Foxit PDF Conversion SDK allows users to download a trial version to evaluate the SDK. The trial version has no difference from a standard version except for the 14-day limitation trial period. After the evaluation period expires, customers should contact Foxit sales team and purchase licenses to continue using Foxit PDF Conversion SDK.

1.4 License

Developers should purchase licenses to use Foxit PDF Conversion SDK in their solutions. Licenses grant users permissions to release their applications based on PDF SDK libraries. However, users are prohibited to distribute any documents, sample codes, or source codes in the SDK released package to any third party without the permission from Foxit Software Incorporated.

1.5 About this guide

This guide is intended for developers who need to integrate Foxit PDF Conversion SDK for .NET Core into their own applications. It aims at introducing the installation package, and the usage of SDK.

2 GETTING STARTED

It's very easy to setup Foxit PDF Conversion SDK and see it in action! This guide will provide you with a brief introduction on how to integrate Foxit PDF Conversion SDK for .NET Core into the projects on Windows platform. The following sections introduce the contents of system requirements, the installation package as well as how to run a demo, and how to initialize Foxit PDF Conversion SDK.

2.1 System Requirements

Platform	System Requirement	Note
Windows	Windows 10 (32-bit, 64-bit)	Visual Studio 2017 version 15.9 or higher Prerequisites for .NET Core 2.1 or higher on Windows

Note: Starting from version 1.1, Foxit PDF Conversion SDK supports machine learning-based recognition functionality which can identify borderless tables in PDF documents. By default, it is disabled, if you want to enable it, please make sure the network is connected first.

2.2 What is in the package

In this guide, the highlighted rectangle in the figure is the version of the SDK. Here the SDK version is 1.1, so it shows 1_1.

Download the Foxit PDF Conversion SDK zip for .NET Core package and extract it to a new directory "foxitpdfconversionsdk_1_1_win_dotnetcore", which is shown in Figure 2-1.

Note: You can also get this package from NuGet.

The release package contains the following folders:

doc: API reference, developer guide

examples: sample demo

lib: libraries and license files

res: font-related configuration files

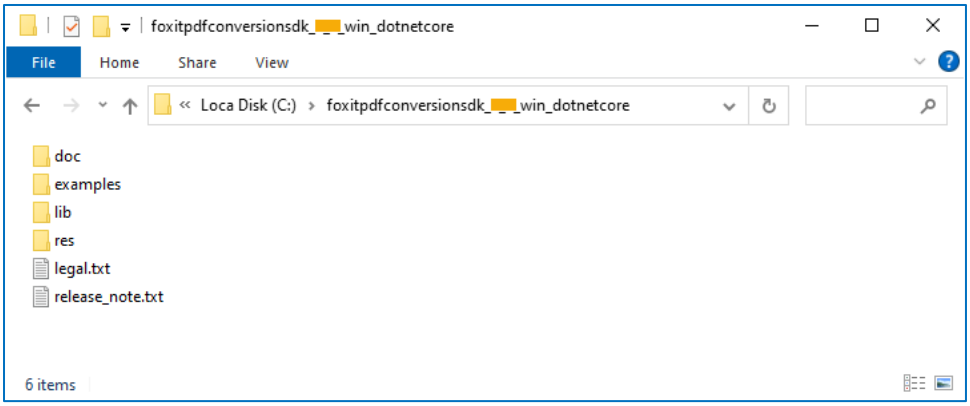


Figure 2-1

Note: In "res\metrics_data" folder, it contains the metrics data files which are used to simulate the Office format document typesetting process during conversion.

2.3 How to run a demo

Foxit PDF Conversion SDK for .NET Core provides a pdf2office demo in the "examples" folder. It can be run directly in a command prompt using the "RunDemo.bat" file in directory "\examples\simple_demo".

Open a command prompt, navigate to "\examples\simple_demo", and then run the following command:

- For **32-bit** operating system, type *"**RunDemo.bat pdf2office x86**"* to run the demo.
- For **64-bit** operating system, type *"**RunDemo.bat pdf2office**"* to run the demo.

The output files (Word, Excel, and Powerpoint) will be generated in the "examples\simple_demo\output_files\pdf2office" folder.

2.4 How to initialize Foxit PDF Conversion SDK

It is necessary for apps to initialize Foxit PDF Conversion SDK using a license before calling any APIs. The trial license files can be found in the "lib" folder.

```
string sn = " ";
string key = " ";
ErrorCode error_code = Library.Initialize(sn, key);
if (error_code != ErrorCode.e_ErrSuccess)
{
  return;
}
```

Note *The value of "sn" can be got from "**conversionsdk_sn.txt**" (the string after "SN=") and the value of "key" can be got from "**conversionsdk_key.txt**" (the string after "Sign=").*

SUPPORT

Foxit Support

In order to provide you with a more personalized support for a resolution, please log in to your Foxit account and submit a ticket so that we can collect details about your issue. We will work to get your problem solved as quickly as we can once your ticket is routed to our support team.

You can also check out our Support Center, choose Foxit PDF SDK which also has a lot of helpful articles that may help with solving your issue.

Phone Support:

Phone: 1-866-MYFOXIT or 1-866-693-6948