Foxit PDF SDK

DEVELOPER GUIDE
FOXIT PDF CONVERSION SDK

For Python

TABLE OF CONTENTS

1 INTRODUCTION TO FOXIT PDF CONVERSION SDK

Foxit PDF Conversion SDK is a flexible high-performance library for conversion between PDF files and MS Office files 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 engines for creating MS office files or PDF files are 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 Python API

In this guide, we focus on the introduction of Foxit PDF Conversion SDK for Python API on Windows and Linux platforms. Foxit PDF Conversion SDK for Python API ships with simple-to-use APIs that can help Python developers do conversion between PDFs and MS Office files (only support docx, pptx and xlsx format files).

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 Python API 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 about our SDK package. 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 Vista, 7, 8 and 10 (32-bit and 64-bit) Windows Server 2003, 2008 and 2012 (32-bit and 64-bit)	It only supports for Windows 8/10 classic style, but not for Store App or Universal App.
Linux	x86/x64 (32-bit and 64-bit OS) - The minimum supported version of the GCC compiler is gcc4.9.4. - The minimum supported version of GLIBC is GLIBC_2.17.	The sample for Linux x86/x64 has been tested on Ubuntu16.0 32/64 bit.

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.

- For PDF2Office, it supports Windows and Linux platforms. For Office2PDF, it only supports Windows platform currently.
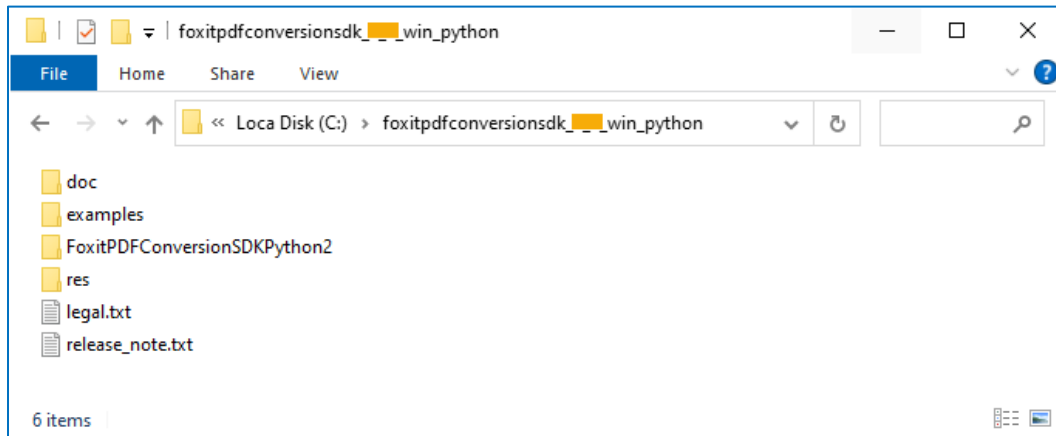
2.2   Windows

2.2.1   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 2.0, so it shows 2_0.

Download Foxit PDF Conversion SDK zip for Windows python API package and extract it to a new directory "foxitpdfconversionsdk_2_0_win_python", which is shown in Figure 2-1. The release package contains the following folders:

doc: developer guide

examples: sample demo

FoxitPDFConversionSDKPython2:   libraries of the Python 2.7

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 PDF to Office conversion.

- In "res\office2pdf" folder, it contains the resource files which are used to simulate the Office format document typesetting process during Office to PDF conversion.

2.2.2   How to run a demo

Requirement

1) Python 2.7 or 3.6-3.10, a single version of Python installed on the system is allowed.

2) Make sure that the default command is python, and it is added in the system path environment variables.

3) Python 3.6-3.10 use pip to install FoxitPDFConversionSDKPython3 module from pypi.

```
> pip install FoxitPDFConversionSDKPython3
```

About package directory structure

The folder **FoxitPDFConversionSDKPython2** is used for Python 2.7. If you use Python 2.7 32-bit, you need to manually copy from FoxitPDFConversionSDKPython2/x86_vc15/_fpdfconversionsdk.pyd

to FoxitPDFConversionSDKPython2/. If you use Python 2.7 64-bit, you need to manually copy from FoxitPDFConversionSDKPython2/x64_vc15/_fpdfconversionsdk.pyd to FoxitPDFConversionSDKPython2/. You can also run examples/simple_demo/rundemo_python.py for automatic copying.

How to run the demo

1) Use rundemo_python.py to run the demo on Python.

   To run pdf2office demo:

   ```
   > cd examples/simple_demo/
   > python rundemo_python.py pdf2office
   ```

   To run office2pdf demo, first you should copy all the files under "res\office2pdf" folder to the directory where the python.exe is located. Then run the command as follows:

   ```
   > cd examples/simple_demo/
   > python rundemo_python.py office2pdf
   ```

2) Run the demo script directly with python.

   For Python 2.7, if _fpdfconversionsdk.pyd in the corresponding directory matches the current system python version, you can use python to run the demo directly. For the correspondence between _fpdfconversionsdk.pyd and python version, please see **About package directory structure**. For Python 3, if you have installed FoxitPDFConversionSDKPython3 module, you can use python to run the demo directly.

   To run pdf2office demo:

   ```
   > cd examples/simple_demo/pdf2office/
   > python -u pdf2office.py
   ```

   To run office2pdf demo, first you should copy all the files under "res\office2pdf" folder to the directory where the python.exe is located. Then run the command as follows:

   ```
   > cd examples/simple_demo/office2pdf/
   > python -u office2pdf.py
   ```
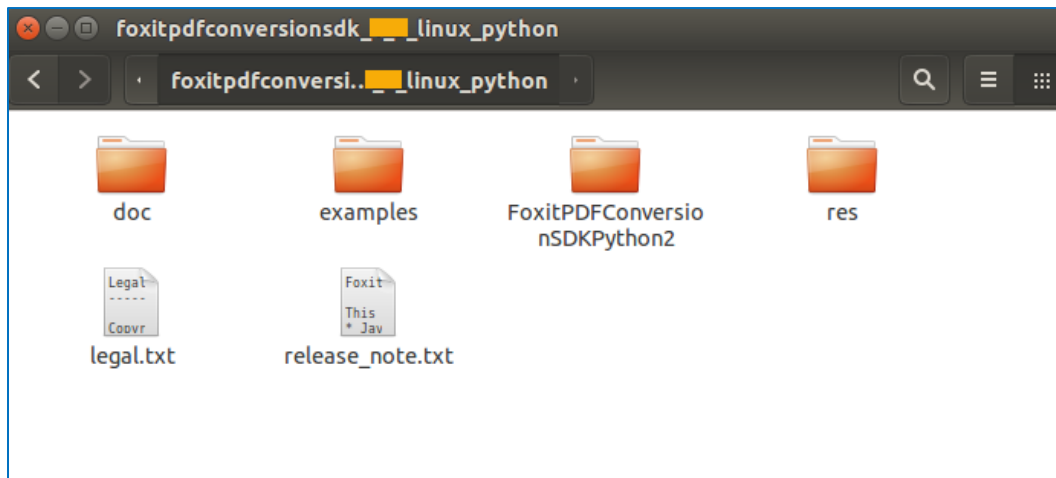
2.3 Linux

2.3.1 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 2.0, so it shows 2_0.

Download Foxit PDF Conversion SDK zip for Linux x86/x64 python API package and extract it to a new directory "foxitpdfconversionsdk_2_0_linux_python", which is shown in Figure 2-2. The release package contains the following folders:

doc: developer guide

examples: sample demo

FoxitPDFConversionSDKPython2:   libraries of the Python 2.7

res: font-related configuration files

Figure 2-2

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.

2.3.2 How to run a demo

Requirement

1) Python 2.7 or 3.6-3.10, a single version of Python installed on the system is allowed.

2) Make sure that the default command is python, and it is added in the system path environment variables.

3) Python 3.6-3.10 use pip to install FoxitPDFConversionSDKPython3 module from pypi.

```
> pip install FoxitPDFConversionSDKPython3
```

About package directory structure

The folder **FoxitPDFConversionSDKPython2** is used for Python 2.7. If you use Python 2.7 32-bit, you need to manually copy from FoxitPDFConversionSDKPython2/x86_vc15/_fpdfconversionsdk.so to FoxitPDFConversionSDKPython2/. If you use Python 2.7 64-bit, you need to manually copy from FoxitPDFConversionSDKPython2/x64_vc15/_fpdfconversionsdk.so to FoxitPDFConversionSDKPython2/. You can also run examples/simple_demo/rundemo_python.py for automatic copying.

How to run the demo

1) Use rundemo_python.py to run the demo on Python.

```
> cd examples/simple_demo/
> python rundemo_python.py pdf2office
```

2) Run the demo script directly with python.

For Python 2.7, if _fpdfconversionsdk.so in the corresponding directory matches the current system python version, you can use python to run the demo directly. For the correspondence between _fpdfconversionsdk.so and python version, please see **About package directory structure**. For Python 3, if you have installed FoxitPDFConversionSDKPython3 module, you can use python to run the demo directly.

```
> cd examples/simple_demo/pdf2office/
> python -u pdf2office.py
```

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.

```python
sn = " "
key = " "
if __name__ == '__main__':
    code = Library.Initialize(sn, key)
    if code == e_ErrSuccess:
        main()
    Library.Release()
```

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