**VBA – PDF to Excel with one click of a button for Foxit Phantom PDF**

**Key words:** VBA, PDF, Excel

**Note:** Detailed coding of this subject can be found on my GitHub account -

1. **Introduction**

VBA is a coding language used by millions of people across the world to automate tasks in Microsoft Office products. It’s a language that has been around for decades and is one of the easiest coding languages to learn if you don’t have a computer science background. I am a chemist by degree and it was my first introduction to coding before the “Software developer” course from SAIT.

It is the easiest languages to learn compare to some others, there is no brackets, no long weird phrases and so on. Because VBA is something that makes sense to anyone familiar with Excel (or Word or PowerPoint). There’s no weird syntax like curly brackets or a complex lexicon of phrases to memorize, you simply write what you want to do and it is done.

I always choose work smarter not harder approach. It was the case when I decided to write a macro to convert PDF file to excel in specific way, to avoid typing the same information multiple times a day.

**02. Understanding the PDF file**

Before jumping into the code, itself, it is important to understand the difference between PDF software available for Windows: Adobe, Adobe Light, Adobe Pro, Foxit, Foxit Pro and so on.

There is more than enough information on the web to find the best one:

### [Best PDF editors 2020: Reviewed and rated | PCWorld](https://www.pcworld.com/article/3215231/best-pdf-editors-reviewed-and-rated.html)

### [Best PDF readers for Windows of 2020 | TechRadar](https://www.techradar.com/best/best-pdf-readers-for-windows)

The chances are there is a PDF software already installed for you and there is high chance that PRO option will allow you to convert file to excel.

I started with Foxit light version and had to upgrade to Foxit Pro ( version 9.6) to be able to achieve my goal.

I have a feeling all Pro versions will allow pdf file to be converted to excel with no issue, but in my case, it also has to be assigned in specific way. I had access to Foxit Pro PDF software and that’s what I had to use to achieve my goal.

**03. Setting up the project**

There are three main blocks:

1. Convert PDF to Excel
2. Find the information and assign to the right cells in excel
3. Make it user-friendly

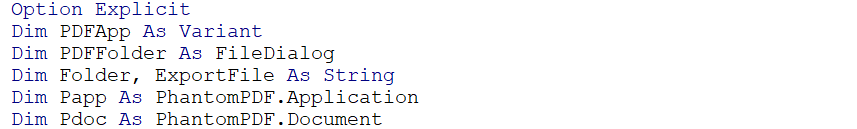
**03.1 Convert PDF to Excel**

This was by far the most challenging part of the VBA coding for me. Currently available information on the Foxit software website, can we found here:

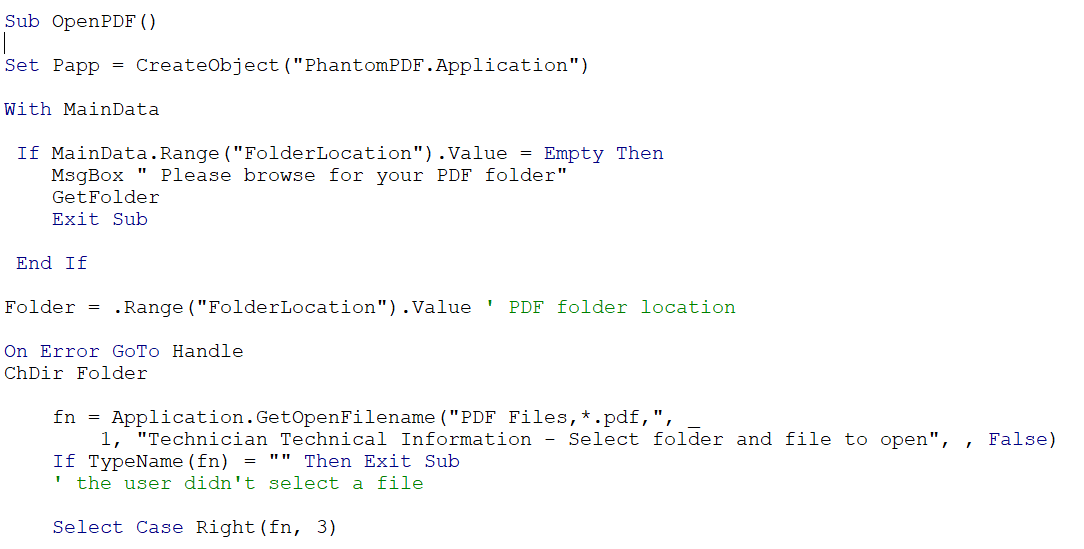
1. <http://cdn01.foxitsoftware.com/pub/foxit/manual/phantom/en_us/FoxitPhantomPDF9.6_Manual.pdf>
2. <http://cdn01.foxitsoftware.com/pub/foxit/manual/phantom/en_us/API%20Reference%20for%20Application%20Communication.pdf>

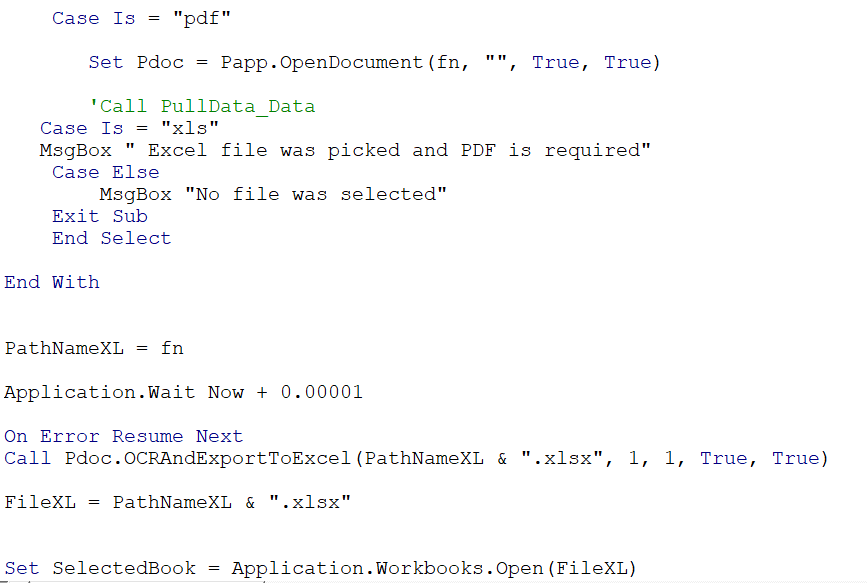
It took a while for me to combine it in one code, but here is my spin on it:

1. **Declare variables**



1. **After selecting the file to be converted, we call PDF code to convert PDF to Excel**

****

****

* 1. **At this point the PDF file is converted to Excel and we are ready to pull the data as needed. My personal choice is to use Find() method and Offset function when I need to find a value and assigned it to another cell in excel.**

****

* 1. Anyone working with coding languages knows, that code has to be robust and bullet-proof to prevent users from breaking your code, so error handling it is one of the most important things to do after the code is complete. As a final result, user able to convert PDF file into Excel with one click of a button.

Work Smarter, not harder.

I hope you found this blog useful. Work smarter, not harder and VBA gives you plenty of opportunities to do so.

**References**

**[1]**[**Web Scraping news articles in Python – TowardsDataScience.com**](https://towardsdatascience.com/web-scraping-news-articles-in-python-9dd605799558)