Productive Computing

PDF Manipulator

Functions Guide

Revised June 27, 2013

Table of Contents

I. Introduction

Description

The PDF Manipulator plug-in offers the ability to exchange data between FileMaker® and Adobe® Acrobat. With the plug-in FileMaker users are able to "push" and "pull" information to and from PDF form fields. The FileMaker user can also extract page text, extract or modify PDF metadata, and combine several PDF documents into a single PDF. These operations are accomplished by using FileMaker function calls from within FileMaker calculations. These calculations are generally determined from within FileMaker "SetField" or "If" script steps. This document describes the plug-in functions.

Intended Audience

FileMaker developers or persons who have knowledge of FileMaker scripting, calculations and relationships as proper use of the plug-in requires that FileMaker integration scripts be created in your FileMaker solution.

Successful Integration Practices:

1) Read the Developer's Guide
2) Read the Functions Guide
3) Review our FileMaker Demo and video tutorials
Demo and video tutorials: http://www.pdfmanipulator.com
4) Familiarize yourself with Adobe Acrobat and how to create PDFs with form fields

Technical Note

File and full paths are formatted as follows:

Mac - Paths take the form of MountedVolume/path. Paths can also be user- relative (e.g., ~/Desktop)
For Example: /Volumes/MacHD/PDF Template.pdf

Windows - Paths take the form of DriveLetter:\path or \\ServerName\path
For Example: C:\Documents and Settings\User\Desktop\PDF Template.pdf or \\Server1\Docs\PDF Template.pdf

II. Function Descriptions

This section describes the functions that are available with the plug-in.

1) Registration and Other Functions

PCPF_GetLastErrorText

Description:

Returns a verbose description of the last error encountered. This function is called immediately after an error string "!!ERROR!!" is returned to obtain the description of the error.

Parameters:

Return Values:

The description of the error

PCPF_GetOperatingMode

Description:

Returns the current operating mode of the plug-in.

Parameters:

Return Values:

"DEMO" for demo mode, "LIVE" for live mode, or "UNREGISTERED" for unregistered mode, or "EXPIRED" for expired mode after 2 hour or 30 day timeout.

PCPF_Register(ServerName ; ServerPort ; ServerPage ; licenseID)

Description:

Registers the plug-in with Productive Computing's registration server.

Parameters:

ServerName =   "licensing.productivecomputing.com"

ServerPort =   "80"

ServerPage =   "/PCIReg/pcireg.php"

LicenseID =   "your license ID" or "DEMO-PF3"

Demo ID for Windows or Mac is "DEMO-PF3"

Return Values:

0 for success or -10 with error text

PCPF_Version(type)

Description:

Used to identify the plug-in and plug-in version installed on a machine. The current version of the plug-in in string format. A 'Short' version contains only the version number. A 'Long' version contains the product name and the version number.

Parameters:

type = 'Short' or 'Long'

The short version string includes only the version number of the plug-in. Example: "2.0.4.0".

The long includes the name of the plug-in and version number. Example: "PDF Manipulator 2.0.4.0"

The default version string returned is short.

Return Values:

The version string (product name and/or version) or ? if plug-in is not installed.

2) PDF Functions

PCPF_AppendPDF(FilePaths)

Description:

The function appends the specified document(s) to the currently opened PDF document

Parameters:

FilePaths =   A list of full paths to the PDF documents that are to be appended to the currently opened document. The list is of the same format as a FileMaker list - paths are delimited by the new line character (one full path per line).

Return Values:

0 for success, otherwise an !!ERROR!!

PCPF_ClosePDF

Description:

This closes the currently opened PDF document. Any changes to the document are not saved.

Parameters:

Return Values:

0 for success, otherwise an !!ERROR!!

PCPF_DeletePDF(FullPath) or (FilePath)

Description:

This function PERMANENTLY deletes the file at the specified location. Use with caution as the deletion cannot be undone.

Parameters:

FullPath = The full system path to the PDF file to be deleted. (Windows parameter)

FilePath = The full system path to the PDF file to be deleted. (Mac parameter)

Return Values:

0 for success, otherwise an !!ERROR!!

PCPF_GetFieldNames (includeValues) or (optionalIncludeValues)

Description:

This function returns all of the PDF field names and optionally the corresponding field value as a FileMaker list. The name is separated from the value by ':|:' sequence of characters.

Parameters:

includeValues =   (optional Windows parameter) Pass a boolean value (1 or True) to include the field values in the list. The default is false.

optionalIncludeValues = (optional Mac parameter) Pass a boolean value (1 or True) to include the field values in the list. The default is false.

Return Values:

0 for success otherwise an !!ERROR!!

PCPF_GetNumPages

Description:

This function returns the number of PDF pages in the currently opened PDF document.

Parameters:

Return Values:

The number of pages, otherwise an !!ERROR!!

PCPF_GetPDFFieldValue(FieldName)

Description:

This function RETURNS the value contained in the PDF form field referenced by the "FieldName" parameter.

Parameters:

FieldName =   the full name of the PDf form field from which to retrieve a value. Available PDF field names can be retrieved using the PCPF_GetFieldNames function.

Return Values:

The value in the field, otherwise an !!ERROR!!

PCPF_GetPDFInfo(Field)

Description:

This function is used to access a few of the metadata items of the currently opened PDF file.

Parameters:

Field =   The name of the metadata value to retrieve. Valid Values are Title, Author, Subject, Keywords, Creator, Producer, CreationDate, ModDate, or Trapped

All Metadata is typically located in the PDF by selecting "File", then "Properties" and navigating to the "Description" tab. The "Trapped" field is found in the "Advanced" tab from within the PDF Properties area.

Return Values:

The value for the item, otherwise an !!ERROR!!

PCPF_GetTextFromPage(PageNumber)

Description:

This function returns all of the text on a page of the currently opened PDF document.

Parameters:

PageNumber =   the PDF page number to retrieve the text from.

Return Values:

A string containing all of the text on the page, otherwise an !!ERROR!!

PCPF_OpenPDF(FullPath)

Description:

Opens a PDF document with Adobe Acrobat. It is recommended that the user has the desired version of Acrobat opened before calling this function.

Parameters:

FullPath =   the system path to the file

Return Values:

0 for success, otherwise an !!ERROR!!

PCPF_PrintPDF

Description:

This function sends the currently opened PDF document to the default system printer.

Parameters:

Return Values:

0 for success, otherwise an !!ERROR!!

PCPF_SaveAsPDF(FullPath)

Description:

This function saves the currently opened document to the path specified in the parameter.

Parameters:

FullPath =   the full system path including the document name that the file will be saved as.

Return Values:

0 for success, otherwise an !!ERROR!!

PCPF_SavePDF

Description:

This function saves changes to the currently opened document.

Parameters:

Return Values:

0 for success, otherwise an !!ERROR!!

PCPF_SetPDFInfo(Field ; Value)

Description:

The function sets the value of the PDF information items.

Parameters:

Field =   The name of the info item (see GetPDFInfo for a list of available items)

Value =   The value to place in the information item

Return Values:

0 for success, otherwise an !!ERROR!!

PCPF_SetPDFFieldValue(FieldName ; FieldValue)

Description:

The function SETS the value of a PDF form field in the currently opened PDF document.

Parameters:

FieldName =   The full name of the PDF form field which will be set with a value. Available field
                        names can be retrieved using the PCPF_GetFieldNames function.

FieldValue =   The FileMaker value to place in the PDF form field.

Return Values:

0 for success, otherwise an !!ERROR!!

3) Windows Only PDF Functions

PCPF_CloseAcrobat

Description:

Closes the acrobat application.

Parameters:

Return Values:

0 for success, otherwise an !!ERROR!!

PCPF_DeletePages(StartPage ; EndPage)

Description:

Deletes pages from the currently opened PDF document. Pages between StartPage and EndPage inclusive will be deleted

Parameters:

StartPage = The number of the first page to be deleted.

EndPage = The number of the last page to be deleted.

Return Values:

0 for success, otherwise an !!ERROR!!

PCPF_FileBrowse

Description:

Displays a file sector dialog window.

Parameters:

Return Values:

The full path of the selected file, otherwise an !!ERROR!!

PCPF_FolderBrowse

Description:

Displays a folder selector dialog window.

Parameters:

Return Values:

The full path of the selected folder, otherwise an !!ERROR!!

PCPF_GetTextLengthFromPage(PageNumber)

Description:

Returns the number of characters on the desired page.

Parameters:

PageNumber = The page number of the desired page in the currently opened document.

Return Values:

The number of characters on the desired page, otherwise an !!ERROR!!

PCPF_HideAcrobat

Description:

Hides the Acrobat application window.

Parameters:

Return Values:

0 for success, otherwise an !!ERROR!!

PCPF_ShowAcrobat

Description:

Shows the Acrobat application window.

Parameters:

Return Values:

0 for success, otherwise an !!ERROR!!

III. Error Handling

All PDF functions return an error string of !!ERROR!!. When the error string it returned during processing, immediately call PCPF_GetLastErrorText in order to obtain a full description of the error. The text returned by the PCPF_GetLastErrorText function should assist the developer with troubleshooting script or logic failures.

When a function is successfully called, then either a 0 for success or the desired string will be returned. Please check the return values for each function so you can trap for errors accordingly.

IV. Contact Us

Successful integration of a FileMaker plug-in requires the creation of integration scripts within your FileMaker solution. A working knowledge of FileMaker Pro, especially in the areas of scripting and calculations is necessary. If you need additional support for scripting, customization or setup (excluding registration) after reviewing the videos, documentation, FileMaker demo and sample scripts, then please contact us via the avenues listed below.

> Phone: 760-510-1200
> Email: support@productivecomputing.com
> Forum: www.productivecomputing.com/forum

Please note assisting you with implementing this plug-in (excluding registration) is billable at our standard hourly rate. We bill on a time and materials basis billing only for the time in minutes it takes to assist you. We will be happy to create your integration scripts for you and can provide you with a free estimate if you fill out a Request For Quote (RFQ) at www.productivecomputing.com/rfq. We are ready to assist and look forward to hearing from you!