



PDF2XML – Command Line

User Guide

Copyright 2000 - 2011 Investintech.com Inc. All rights reserved

Adobe is registered by Adobe Systems Incorporated
Acrobat is registered by Adobe Systems Incorporated
Windows is registered trademark by Microsoft Corporation

Table of contents:

1	INTRODUCTION	4
1.1	System Requirements.....	4
2	INSTALLING PDF2XML COMMAND LINE	5
3	CONVERTING A FILE	10
3.1	Getting Help from within the Prompt	12
3.2	Specifying the Input and Output Files	13
3.2.1	Specifying the Input Location.....	14
3.2.2	Specifying the Output Location	15
3.4	Scheduling a Conversion	16
3.5	Performing Partial Conversion of the PDF File	18
3.6	Specifying the Password for the Protected PDF files	19
4	CONVERTING MULTIPLE FILES	20
4.1	Converting Multiple Files Using Wildcards.....	20
4.2	Converting Files Recursively.....	22
4.3	Force Deletion of Files with the Same Name.....	24
5	SUPPORT	26
5.1	Customer Service and Technical Support	26
5.3	How to Submit an Issue	27
6	TROUBLESHOOTING.....	28

List of figures:

Figure 1:	Installation procedure (step 1/7)	5
Figure 2:	Installation procedure (step 2/7)	6
Figure 3:	Installation procedure (step 3/7)	6
Figure 4:	Installation procedure (step 4/7)	7
Figure 5:	Installation procedure (step 5/7)	8
Figure 6:	Installation procedure (step 6/7)	8
Figure 7:	Installation procedure (step 7/7)	9
Figure 8:	Starting the PDF2XML.....	10
Figure 9:	Issuing the command within the Command Prompt.....	10
Figure 10:	Conversion successful.....	11
Figure 11:	Getting help from within the prompt	12
Figure 12:	Specifying the input and output files.....	13
Figure 13:	Specifying the Input Location	14
Figure 14:	Saving the converted file to a specific location	15
Figure 15:	Scheduling the conversion (step 1/2)	16
Figure 16:	Scheduling the conversion (step 2/2)	16
Figure 17:	Specifying the page range (step 1/2)	18
Figure 18:	Specifying the page range (step 2/2)	18
Figure 19:	Specifying the Password for the Protected PDF files (step 1/2)	19
Figure 20:	Specifying the Password for the Protected PDF files (step 2/2)	19
Figure 21:	Converting Multiple Files Using Wildcards (step 1/2).....	20
Figure 22:	Converting Multiple Files Using Wildcards (step 2/2).....	20
Figure 23:	Converting the Files Recursively (step 1/2).....	22
Figure 24:	Converting the Files Recursively (step 2/2).....	22
Figure 25:	Example of the converted files	23
Figure 26:	Force Deletion of Files with the Same Name (step 1/2).....	24
Figure 27:	Force Deletion of Files with the Same Name (step 2/2).....	25

1 INTRODUCTION

PDF2XML CL is a software tool for converting PDF and XPS files into XML format. It is distributed as an executable file which is used from the [command line](#).

PDF2XML CL can be used in combination with any programming language or application which can pass command line arguments to it (it may be used to automate the conversion of multiple documents). The supported output file format is XML.

The tool consists of several files:

- **PDF2XML.exe** – a main executable which is used to convert PDF files to XML
- Other files

This document is a detailed guide for using the “PDF2XML CL”. Instructions for each of the available tasks are located in the appropriated sections of this guide.

1.1 System Requirements

The minimum requirements to install and run PDF2XML are:

- Microsoft Windows 2000 and newer
- 128 MB of RAM
- 10 MB of available hard disk space

The following are the requirements for using the PDF2XML comfortably:

- Microsoft 2000 and newer
- 512+MB of RAM
- 100 MB of available hard disk space (**note:** the amount of required hard disk space also depends on the size and structure of the source files which are being converted)

2 INSTALLING PDF2XML COMMAND LINE

In order to begin installing the “PDF2XML CL” double-click the executable file. The following window will be displayed.

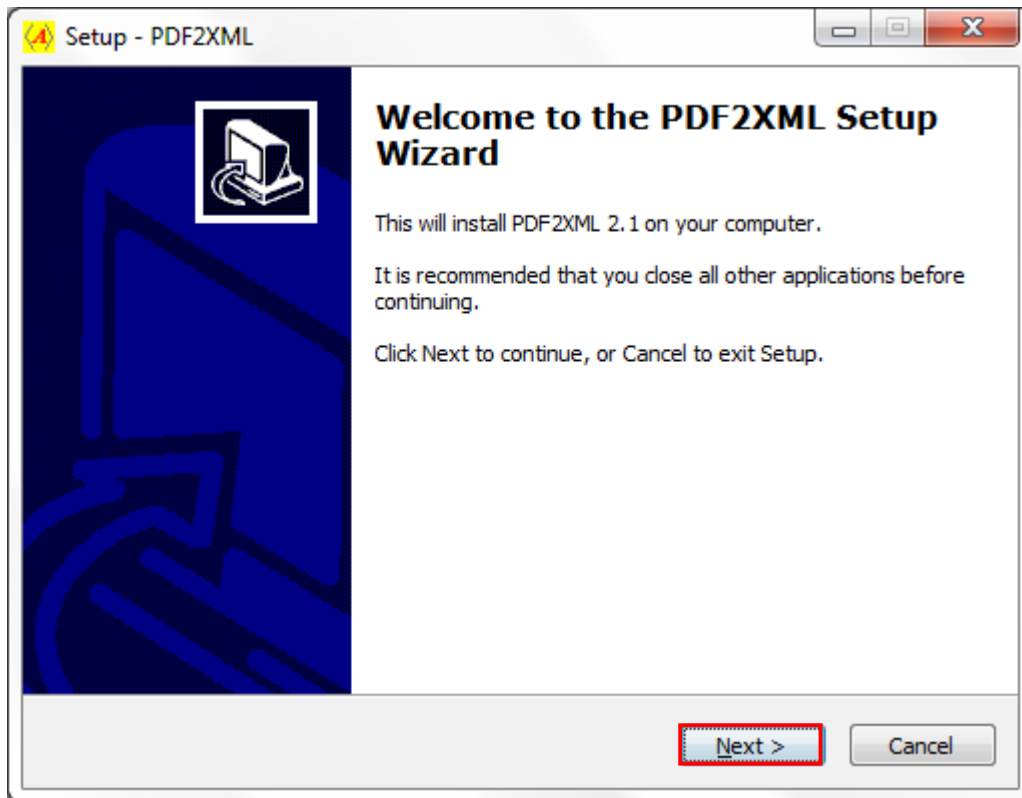


Figure 1: Installation procedure (step 1/7)

Click “Next” and the “License Agreement” window will be displayed.

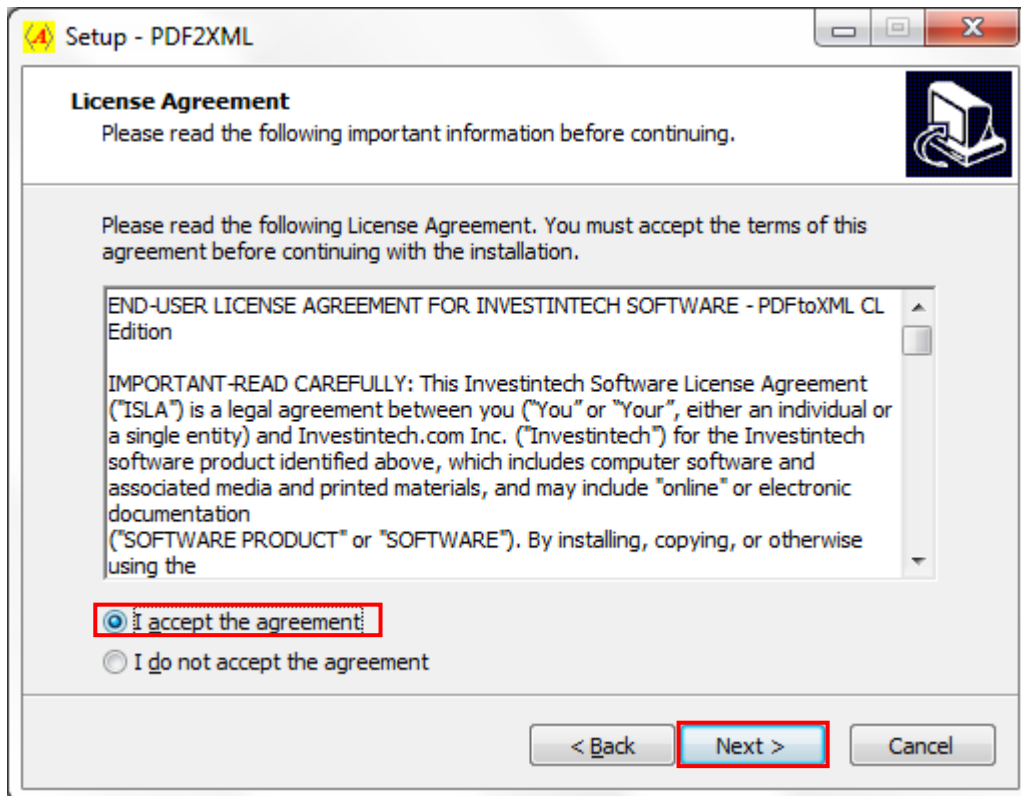


Figure 2: Installation procedure (step 2/7)

By choosing "I accept the agreement" and pressing **Next** you agree to the terms of this license agreement.

You have to accept this agreement in order to install and use the software. The window for choosing the desired destination location will be displayed.

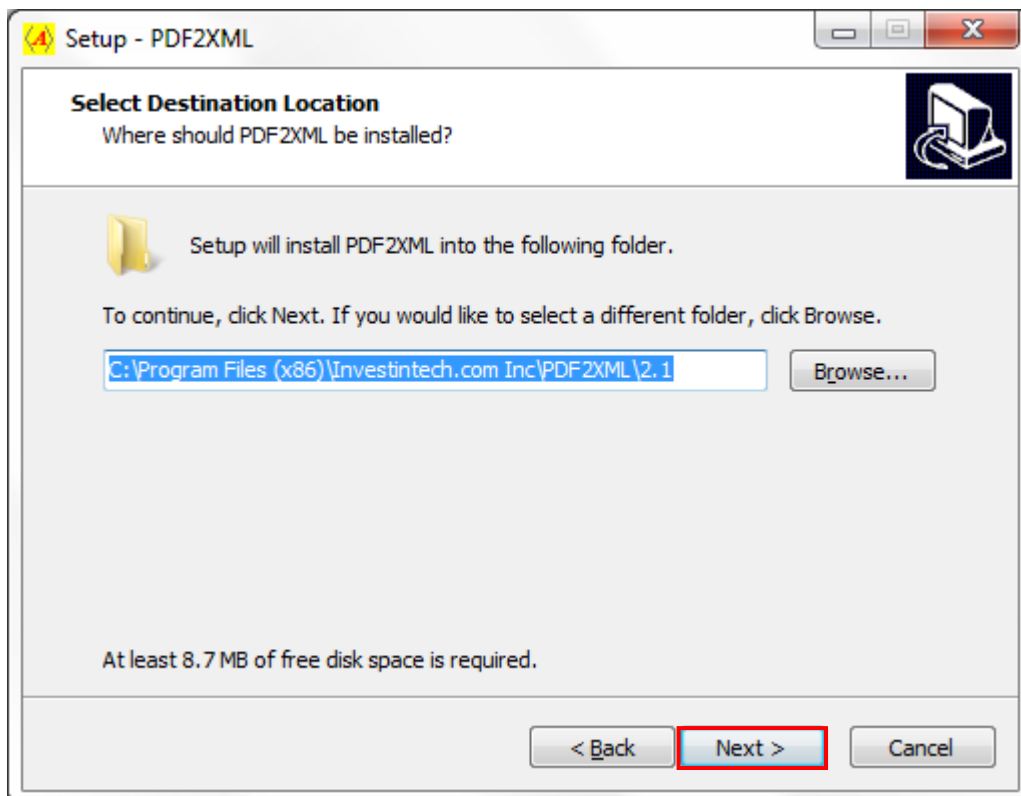


Figure 3: Installation procedure (step 3/7)

You can choose to install the application to the default directory or set the desired folder by clicking **Browse**. After choosing the desired destination location, click **Next**. The following window will be displayed.

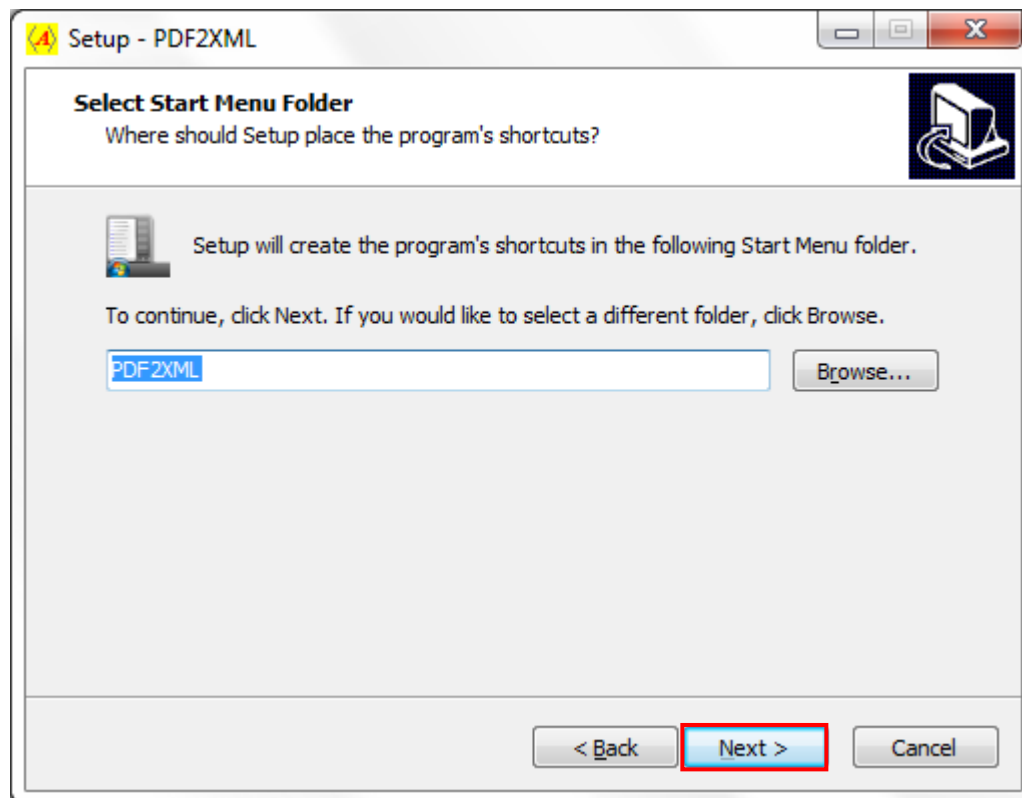


Figure 4: Installation procedure (step 4/7)

Use this window to select the desired Start menu folder i.e. where the setup should place the program's shortcuts.

Either leave the default value or click **Browse** to set a different folder. Click **Next** to continue. The window for specifying additional tasks will be displayed.

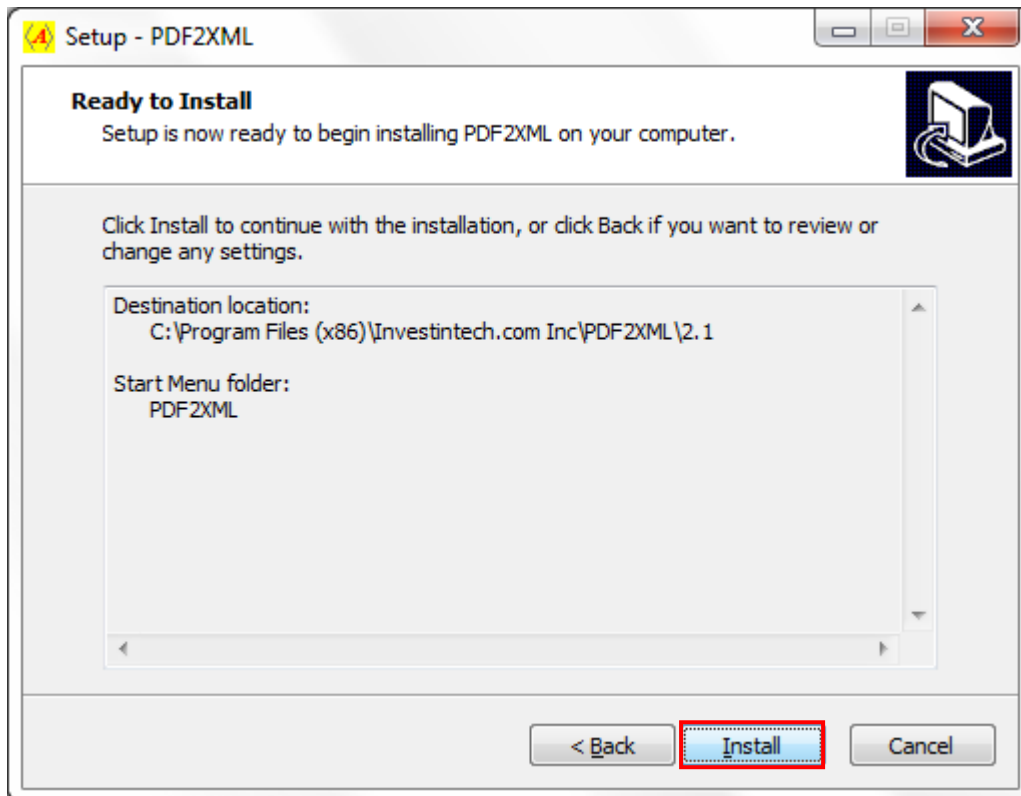


Figure 5: Installation procedure (step 5/7)

The setup will now be ready to install the PDF2XML on your computer. You can still review the settings and choose to go back and modify them.

In case you are satisfied with the chosen parameters, click **Install** to begin installing the program. The following window will be displayed.

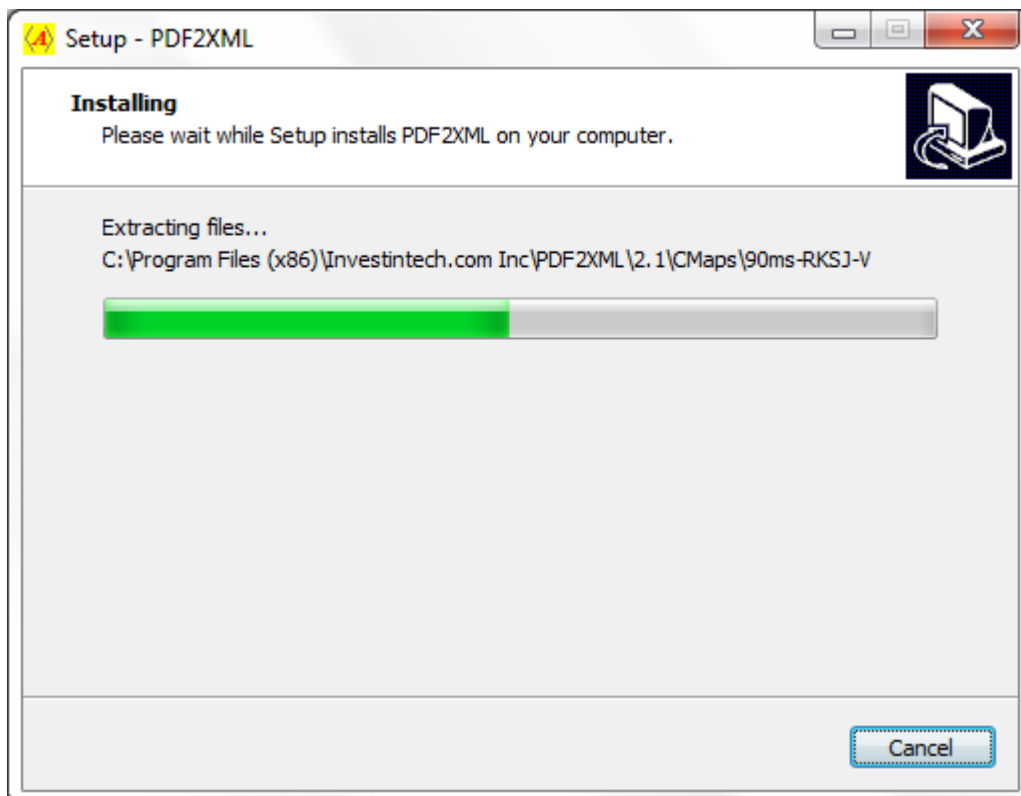


Figure 6: Installation procedure (step 6/7)

After a couple of moments, the installation will be completed and the following window will be displayed.

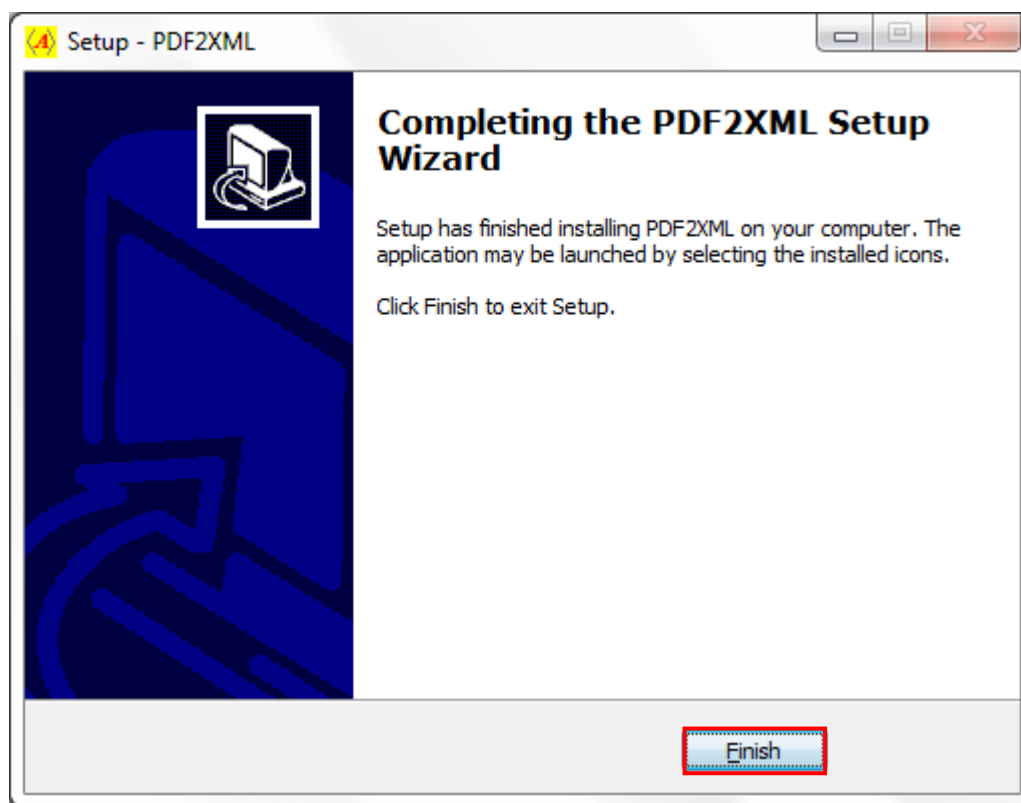


Figure 7: Installation procedure (step 7/7)

Click **Finish** to exit the setup.

Note: the tool is automatically added to the system path and will be available from any location after restart.

3 CONVERTING A FILE

In order to convert a PDF file to an XML, you will need to issue a command from the command prompt.

Start the command prompt by either choosing “PDF2XML” option from the “Start” menu (Start ->All Programs -> PDF2XML -> PDF2XML) or by using the “Run” dialog (Start -> Run).

Note:

- Since PDF2XML is added to the system path after restart, you can start the command prompt by typing `cmd` into within the “Search programs and files” field and typing `PDF2XML`.
- Alternatively, you can just type `PDF2XML` into “Search programs and files” and hit press ENTER on the keyboard.

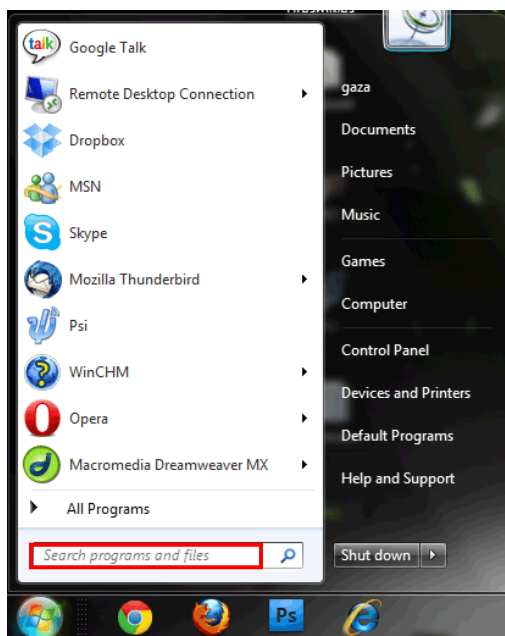


Figure 8: Starting the PDF2XML

The command prompt will be displayed.

```

Administrator: C:\Windows\system32\cmd.exe

C:\repository>PDF2XML
Investintech.com Inc. PDF2XML - Command Line Interface
Copyright 2009 Investintech.com Inc. All rights reserved

Time Limited edition - unlimited conversions are available for 30 days

Usage:
PDF2XML [-r] [-s<schedule>] [-p<range>] -i<input> -o<output> [-w<password>]
[-d]

where:
-r          forces recursive a directory search using the <input> expression
<schedule> is time at which conversion will occur
<range>    indicates page range
<input>    specifies input pdf document (may include wildcards)
<output>   specifies target output file path and extension
<password> specifies password if input file is password protected
-d         forces deleting existing files

Note: input and output files must contain full type extension
Wild cards:
?         Matches any single character in document name
*         Matches 0 or more characters in document name
  
```

Figure 9: Issuing the command within the Command Prompt

Type **PDF2XML** in the command prompt and press enter. The generic command will be displayed containing brief description of all switches within the generic command (see **Figure 9**).

Note: you can also type **PDF2XML -h** to see the list of all available switches and their purpose (see section 3.1).

The Table 1 lists all available switches and their purpose.

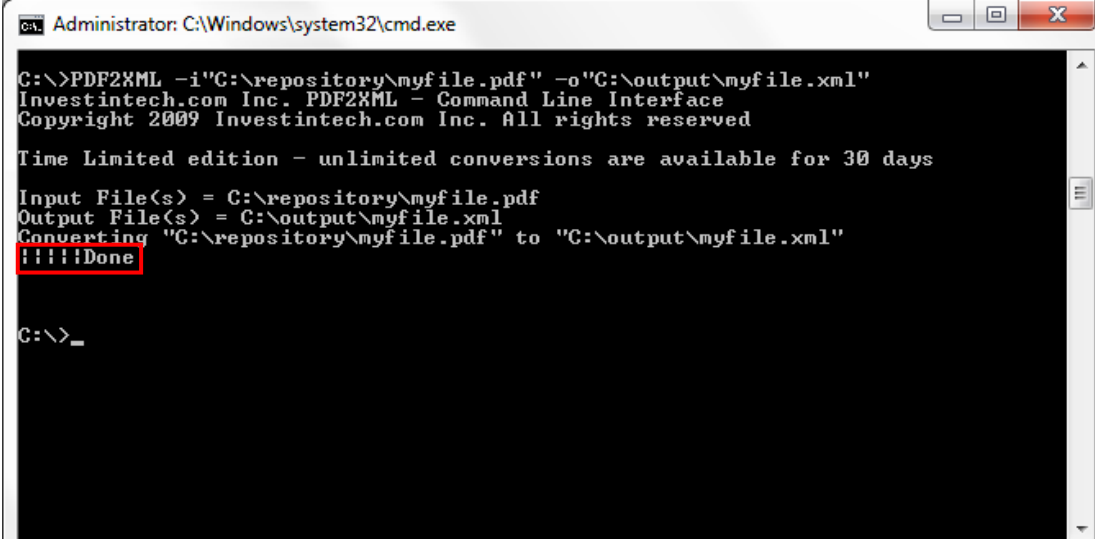
switch	purpose
-i	specifying the name and the location of the input file (see section 3.2)
-o	specifying the name and the location of the output file (see section 3.2)
-p	specifying the range of pages which should be converted (see section 0)
-s	scheduling the conversion to run at specific time (see section 3.5)
-r	converting files recursively (see section 4.2)
-w	specifying the password for the password protected input files (see 3.7)
-d	force deletion of files with the same name (see 4.3)

Table 1: Available switches and their purpose

Note:

- It is recommended that you always specify full paths to both input and output directories (see section 3.2).
- See section 6 for the list of possible reasons and solutions in case the conversion was unsuccessful.

Once the conversion has been completed, the “Done” confirmation line will be displayed.



```

Administrator: C:\Windows\system32\cmd.exe
C:\>PDF2XML -i"C:\repository\myfile.pdf" -o"C:\output\myfile.xml"
Investintech.com Inc. PDF2XML - Command Line Interface
Copyright 2009 Investintech.com Inc. All rights reserved

Time Limited edition - unlimited conversions are available for 30 days

Input File(s) = C:\repository\myfile.pdf
Output File(s) = C:\output\myfile.xml
Converting "C:\repository\myfile.pdf" to "C:\output\myfile.xml"
!!!!Done

C:\>_

```

Figure 10: Conversion successful

The “Done” line denotes successful completion of the file conversion.

Note:

- For each converted page there will be a pipe symbol on the screen to let user know that the conversion is active and the program is still working.
- In the example above 5 pages have been parsed and converted.

3.1 Getting Help from within the Prompt

Help is available directly from within the prompt. Type **PDF2XML** and the list of options and switches containing information on how to use them will be displayed.

```

Administrator: C:\Windows\system32\cmd.exe
C:\>PDF2XML
Investintech.com Inc. PDF2XML - Command Line Interface
Copyright 2009 Investintech.com Inc. All rights reserved

Time Limited edition - unlimited conversions are available for 30 days

Usage:
PDF2XML [-r] [-s<schedule>] [-p<range>] -i<input> -o<output> [-w<password>]
[-d]

where:
-r          forces recursive a directory search using the <input> expression
<schedule> is time at which conversion will occur
<range>    indicates page range
<input>    specifies input pdf document <may include wildcards>
<output>   specifies target output file path and extension
<password> specifies password if input file is password protected
-d         forces deleting existing files

Note: input and output files must contain full type extension
Wild cards:
?         Matches any single character in document name
*         Matches 0 or more characters in document name

Example:
PDF2XML -ic:\documents\Source.pdf -od:\xml\Target.xml
PDF2XML -ic:\documents\Source.pdf -od:\xml\Target.xml -w"password"
PDF2XML -ic:\documents\Source.xps -od:\xml\Target.xml
PDF2XML -i*.pdf -o*.xml
PDF2XML -s15:00:00 -iSource.pdf -oTarget.xml
PDF2XML -p5-9,15 -iSource.pdf -oTarget.xml
PDF2XML -p5-9,15o -iSource.pdf -oTarget.xml
PDF2XML -p4-12,16e -iSource.pdf -oTarget.xml
PDF2XML -tTemplate.ta2e -iSource.pdf -oTarget.xml

-h or -help This help

C:\>_

```

Figure 11: Getting help from within the prompt

3.2 Specifying the Input and Output Files

In order to specify the input file (PDF which will be converted) and output file (the resulting XML file), the following two switches are used:

- **-i** – for specifying the input file
- **-o** for specifying the output file

For the purpose of this example, a PDF file called **myfile.pdf** will be used.

- to specify the **myfile.pdf** as input file, type: **-imyfile.pdf**
- to specify the output file, type: **-omyfile.xml**

The full command for converting myfile.pdf to myfile.xml would be:

```
PDF2XML -i"C:\repository\myfile.pdf" -o"C:\output\myfile.xml"
```

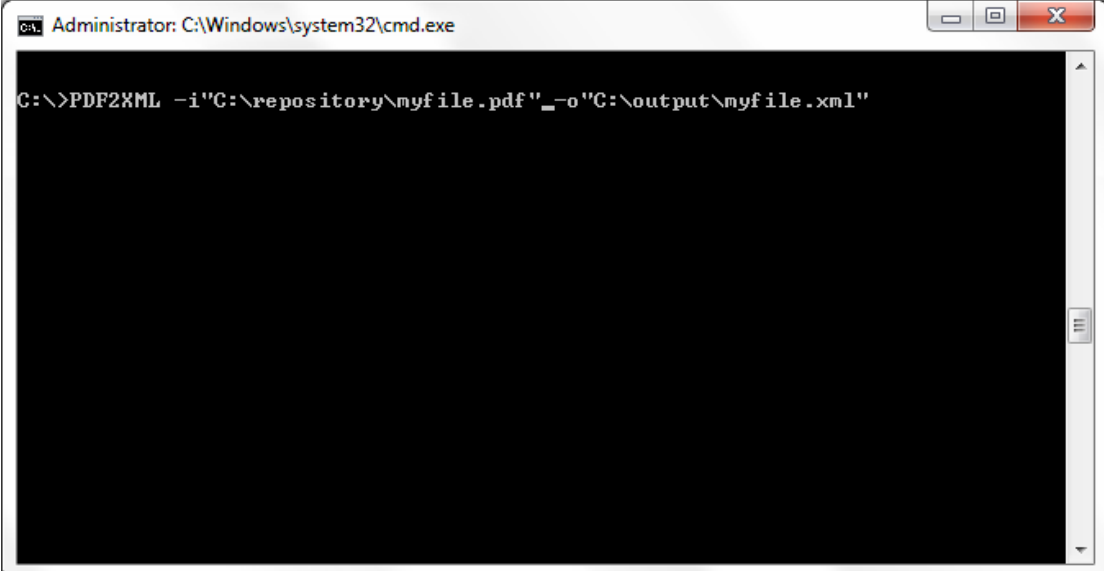
A screenshot of a Windows command prompt window titled "Administrator: C:\Windows\system32\cmd.exe". The window has a black background with white text. The command entered is: `C:\>PDF2XML -i"C:\repository\myfile.pdf" -o"C:\output\myfile.xml"`. The window includes standard Windows window controls (minimize, maximize, close) in the top right corner.

Figure 12: Specifying the input and output files

Note:

- It is recommended that you always specify full paths to both input and output directories.
- The file path should be specified using quotes (e.g. "**C:\Program Files\Folder\myfile.pdf**") – this rule applies to both input and output file paths.
- The output file can be saved under any name i.e. whatever you type after the **-o** switch will become the name of the converted file.

3.3 Specifying the Input Location

In order to specify the input location enter the absolute address i.e. path to the desired file using the input switch.

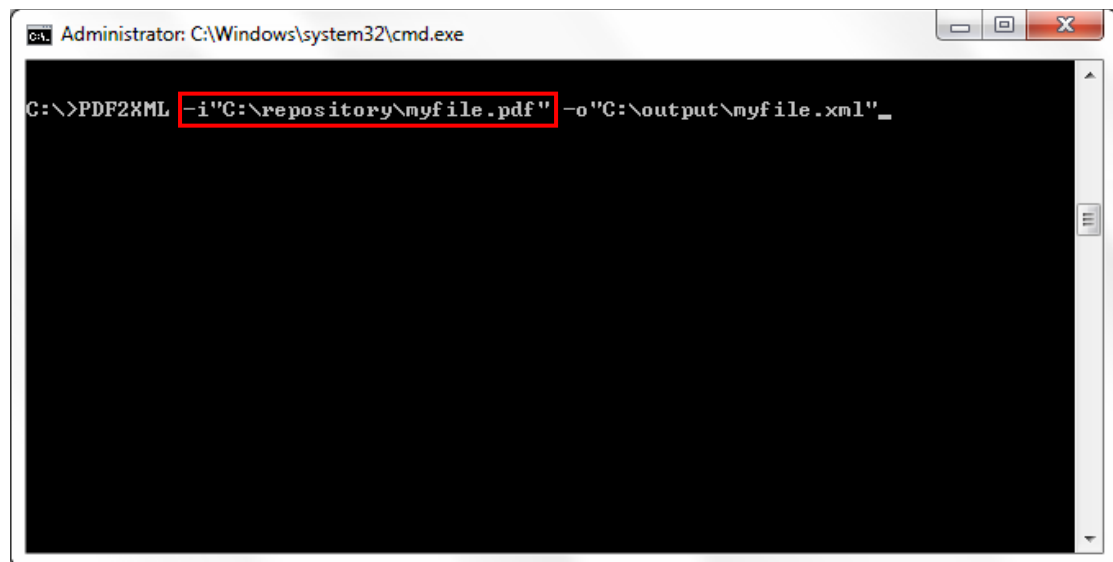


Figure 13: Specifying the Input Location

For example, if the file you wish to convert is located in `C:\repository`, the command for converting that file would be the following:

```
PDF2XML -i"C:\repository\myfile.pdf" -o"C:\output\myfile.xml"
```

This will convert `myfile.pdf` from the "`C:\repository`" folder and save the resulting file to the "`C:\output`" directory.

Note:

- It is recommended that you always specify full paths to both input and output directories.
- The file path should be specified using quotes (e.g. "`C:\Program Files\Folder\myfile.pdf`") – this rule applies to both input and output file paths.

3.4 Specifying the Output Location

In order to specify the output location enter the absolute address i.e. path to the desired file using the output switch.

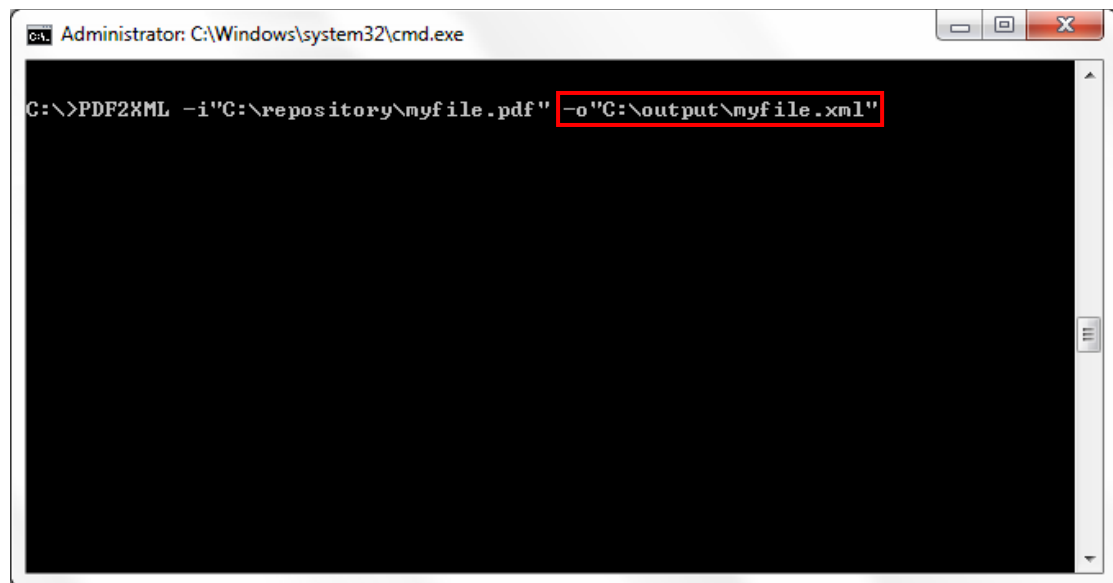


Figure 14: Saving the converted file to a specific location

For example, if the location you wish to save the file to is `C:\output`, the command for saving the converted file to that location would be the following:

```
PDF2XML -i"C:\repository\myfile.pdf" -o"C:\output\myfile.xml"
```

Note:

- It is recommended that you always specify full paths to both input and output directories.
- The file path should be specified using quotes (e.g. "`C:\Program Files\Folder\myfile.pdf`") – this rule applies to both input and output file paths.

3.5 Scheduling a Conversion

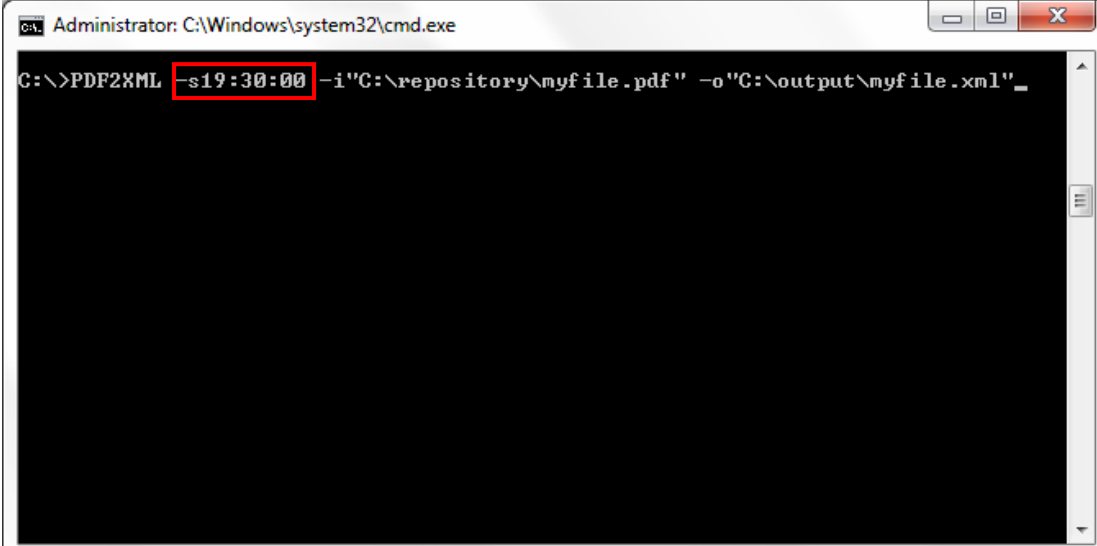
The PDF2XML CL allows you to schedule the file conversion. By using the **-S** schedule switch you can specify the exact time the conversion should occur.

Note: This feature relies on the Windows scheduler functionality, which means you can also manage the scheduled tasks directly from within Windows tool.

For example you can use this option to automate converting files which are placed to a predefined folder, so that you would convert those received files at a specific time.

The switch is followed by the exact time for the conversion in the following format HH:MM:SS (0-24). Below is the example of the command for scheduling the conversion to occur at 18:30.

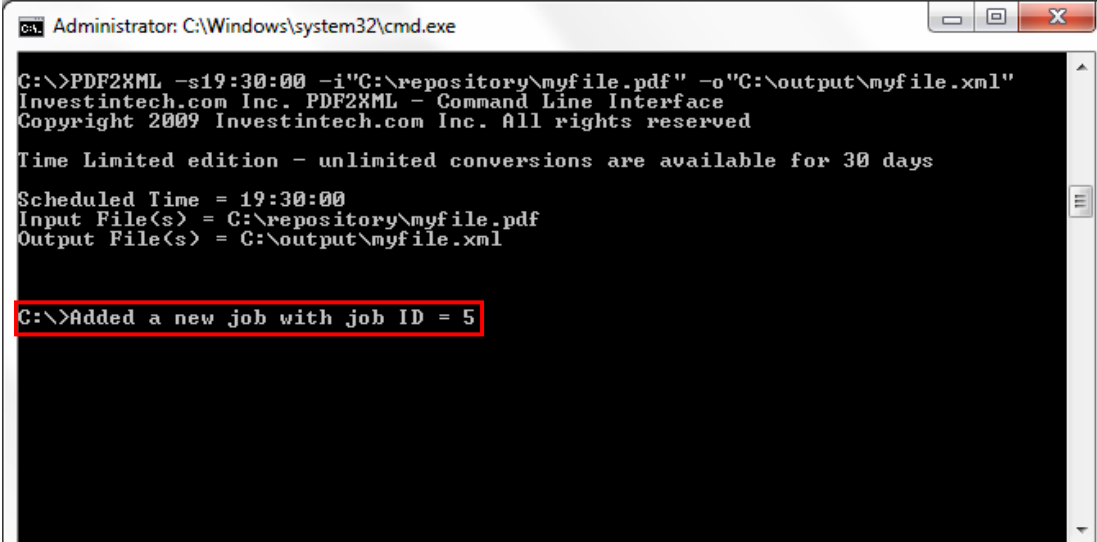
```
PDF2XML -s19:30:00 -i"C:\repository\myfile.pdf" -o"C:\output\myfile.xml"
```



The screenshot shows a Windows command prompt window titled "Administrator: C:\Windows\system32\cmd.exe". The command entered is: `C:\>PDF2XML -s19:30:00 -i"C:\repository\myfile.pdf" -o"C:\output\myfile.xml"`. The time parameter `-s19:30:00` is highlighted with a red box.

Figure 15: Scheduling the conversion (step 1/2)

After issuing the command, press **Enter** on your keyboard and the information similar to the following will be displayed.



The screenshot shows the same Windows command prompt window after the command has been executed. The output is: `C:\>PDF2XML -s19:30:00 -i"C:\repository\myfile.pdf" -o"C:\output\myfile.xml"` followed by: `Investintech.com Inc. PDF2XML - Command Line Interface`, `Copyright 2009 Investintech.com Inc. All rights reserved`, `Time Limited edition - unlimited conversions are available for 30 days`, `Scheduled Time = 19:30:00`, `Input File(s) = C:\repository\myfile.pdf`, and `Output File(s) = C:\output\myfile.xml`. The final line, `C:\>Added a new job with job ID = 5`, is highlighted with a red box.

Figure 16: Scheduling the conversion (step 2/2)

The PDF2XML CL will confirm the scheduled time and add the scheduled job to the spool.

Note:

- If you get an “Access is denied” message, try running the command prompt as an administrator (one way to do it is by entering **cmd** in the run box and hitting **Ctrl+Shift+Enter** instead of just **Enter**).
- You can use the Windows Task Scheduler to cancel the job if necessary.
- You can close the command prompt after scheduling the conversion; this will not cancel it.
- The following conditions should be met for the conversion to take place at a scheduled time:
 - The PC has to be turned on.
 - There has to be an input file.
 - Output folder has to be specified.
 - If there is already a file with the same name within the output folder, the **-d** switch should be used to force the deletion of the existing file (otherwise PDF2XML will abort the conversion) – see 4.37.

3.6 Performing Partial Conversion of the PDF File

The PDF2XML CL tool allows you to convert parts of the PDF by using the **-p** switch to specify the range of pages.

The switch is followed by the specification of the range of pages i.e. **-p12-16** which will convert pages 12-16 only. Below is the example of the command:

```
PDF2XML -p12-16 -i"C:\repository\myfile.pdf" -o"C:\output\myfile.xml"
```

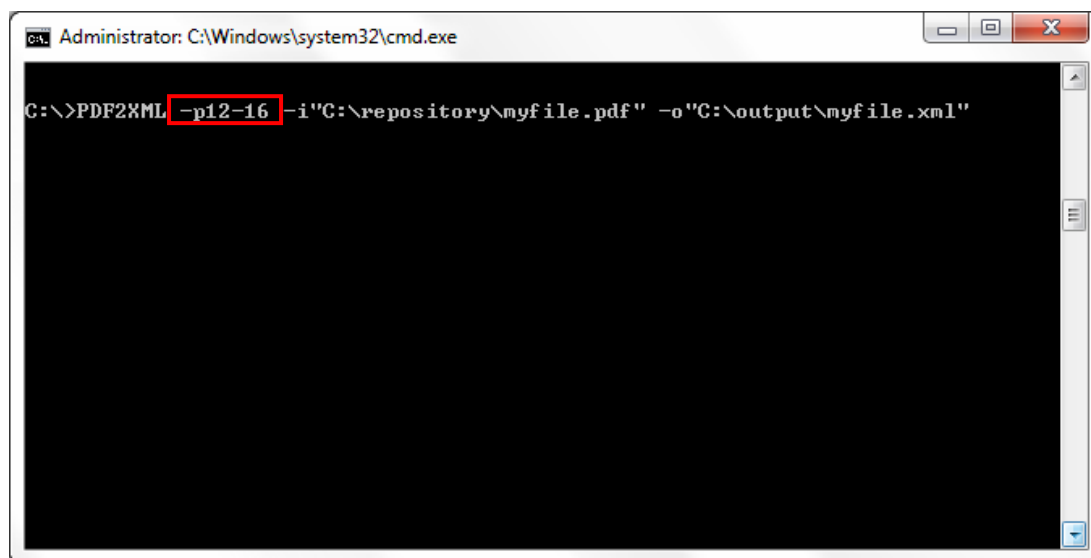


Figure 17: Specifying the page range (step 1/2)

After typing the command, press **Enter** on the keyboard and the specified page range will be converted.

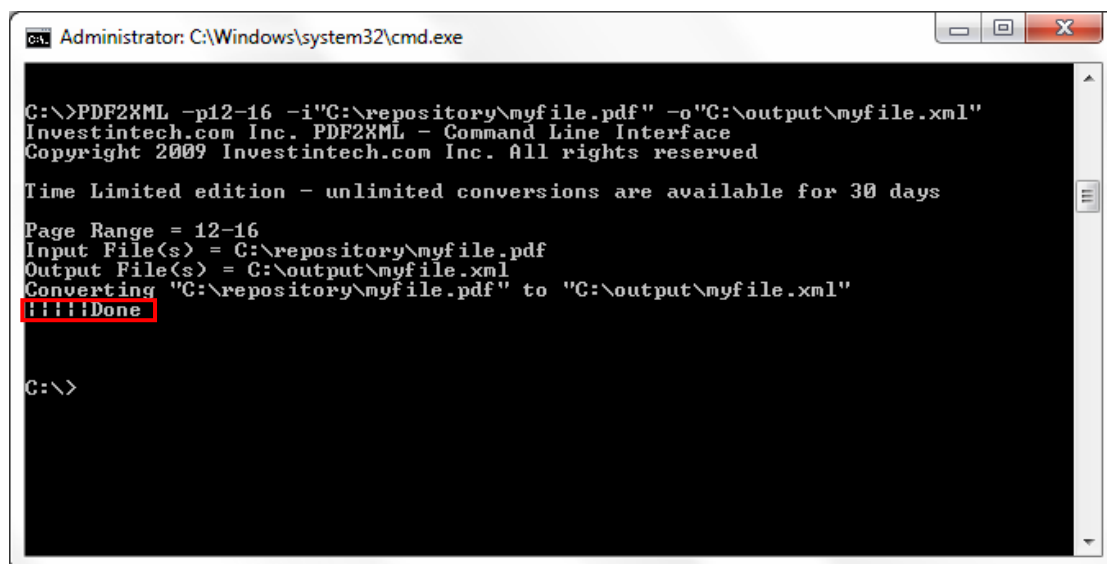


Figure 18: Specifying the page range (step 2/2)

The following other options for converting a range of pages are available:

- **PDF2XML -p5-9,15** – use this format to convert range of pages plus a single page (e.g. pages 5-9 and page 15)
- **PDF2XML -p2,4,7,8,9** – use this format to convert specific single pages
- **PDF2XML -p12-16o** – use this format to only convert **odd pages** within the specified range of pages

- **PDF2XML -p12-16e** – use this format to only convert **even pages** within the specified range of pages

3.7 Specifying the Password for the Protected PDF files

PDF2XML allows you to convert password protected files i.e. PDF files which require password before they could be opened.

The **-w** switch is used for specifying the password. The whole PDF2XML command for opening a password protected PDF file might look like this:

```
PDF2XML -i"C:\repository\myfile.pdf" -o"C:\output\myfile.xml" -wxL7#-%H
```

This command will convert a PDF file called **myfile.pdf** to an XML file called **myfile.xml** with a password specified using the **-w** switch and a password **xL7#-%H**.

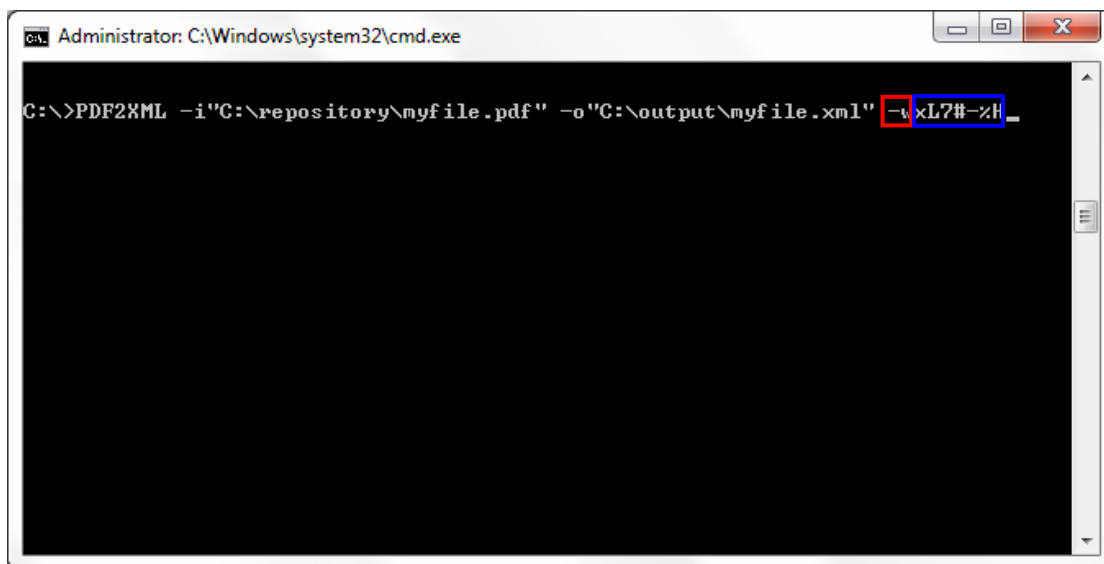


Figure 19: Specifying the Password for the Protected PDF files (step 1/2)

Press **ENTER** on the keyboard and PDF2XML CL will begin the conversion which involves supplying the password in order to open the PDF. Once the conversion has been completed, the confirmation line will be displayed: **Done**.

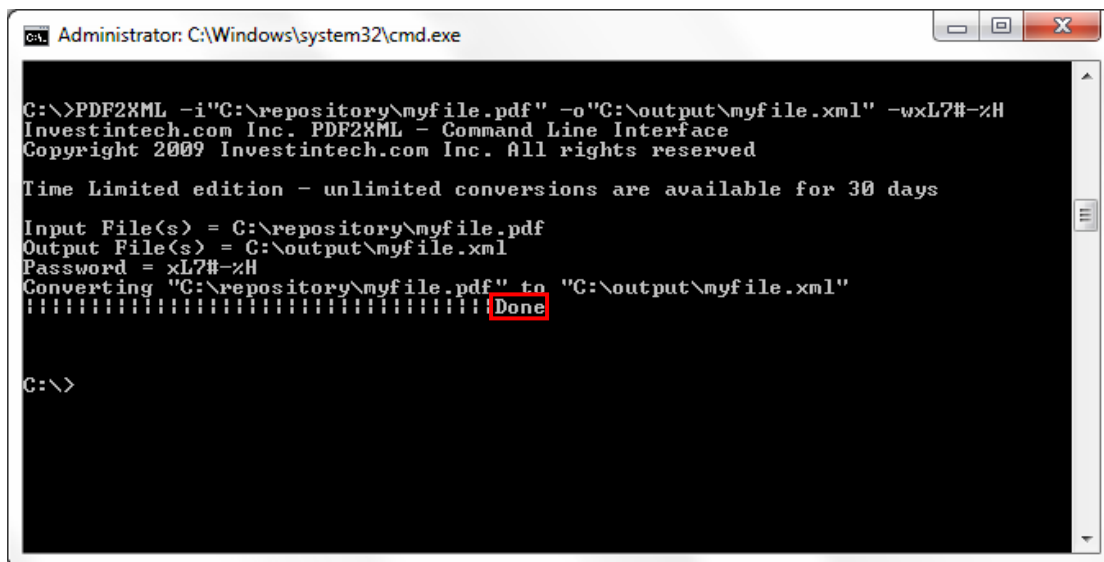


Figure 20: Specifying the Password for the Protected PDF files (step 2/2)

4 CONVERTING MULTIPLE FILES

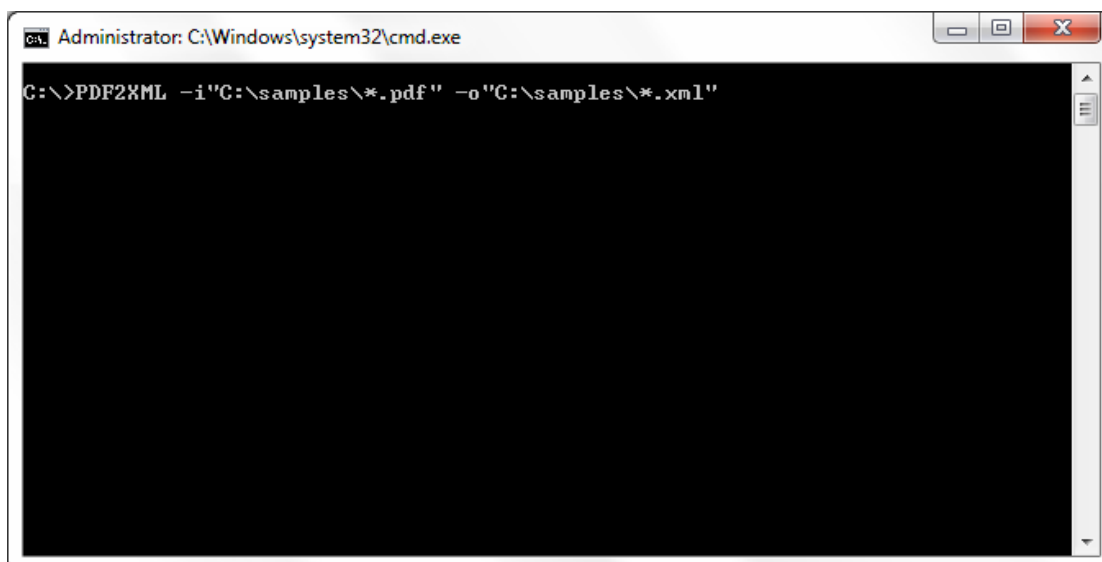
PDF2XML CL allows you to convert multiple files simultaneously. There are several methods you can use – please consult the following sections for further information.

4.1 Converting Multiple Files Using Wildcards

PDF2XML allows you to convert multiple files using wildcards. In the example below, the issued command:

```
PDF2XML -i"C:\samples\*.pdf" -o"C:\samples\*.xml"
```

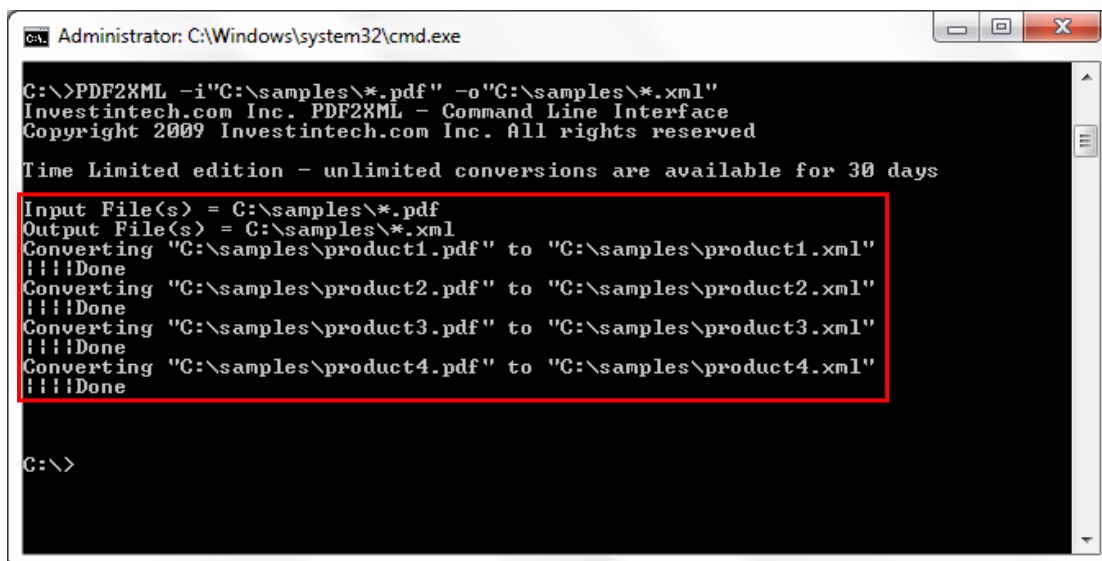
will convert all PDF files within the “samples” directory keeping the original file names.



```
Administrator: C:\Windows\system32\cmd.exe
C:\>PDF2XML -i"C:\samples\*.pdf" -o"C:\samples\*.xml"
```

Figure 21: Converting Multiple Files Using Wildcards (step 1/2)

Press **Enter** after typing the command and all files within the directory will be converted to an XML format.



```
Administrator: C:\Windows\system32\cmd.exe
C:\>PDF2XML -i"C:\samples\*.pdf" -o"C:\samples\*.xml"
Investintech.com Inc. PDF2XML - Command Line Interface
Copyright 2009 Investintech.com Inc. All rights reserved
Time Limited edition - unlimited conversions are available for 30 days
Input File(s) = C:\samples\*.pdf
Output File(s) = C:\samples\*.xml
Converting "C:\samples\product1.pdf" to "C:\samples\product1.xml"
||||Done
Converting "C:\samples\product2.pdf" to "C:\samples\product2.xml"
||||Done
Converting "C:\samples\product3.pdf" to "C:\samples\product3.xml"
||||Done
Converting "C:\samples\product4.pdf" to "C:\samples\product4.xml"
||||Done
C:\>
```

Figure 22: Converting Multiple Files Using Wildcards (step 2/2)

Note:

- as usual you can use wildcards to convert files by specifying partial file names; this way you can convert multiple files within a directory which have common letters within the file name (for replacing single characters use the question mark [?]) and for replacing multiple characters use the asterisk [*]).

For example:

- convert all files which begin with the word “report” by issuing the following command:

```
PDF2XML -i"C:\samples\report*.pdf" -o"C:\samples\*.xml"
```

- or all files starting with letter “p” by issuing:

```
PDF2XML -i"C:\samples\p*.pdf" -o"C:\samples\*.xml"
```

4.2 Converting Files Recursively

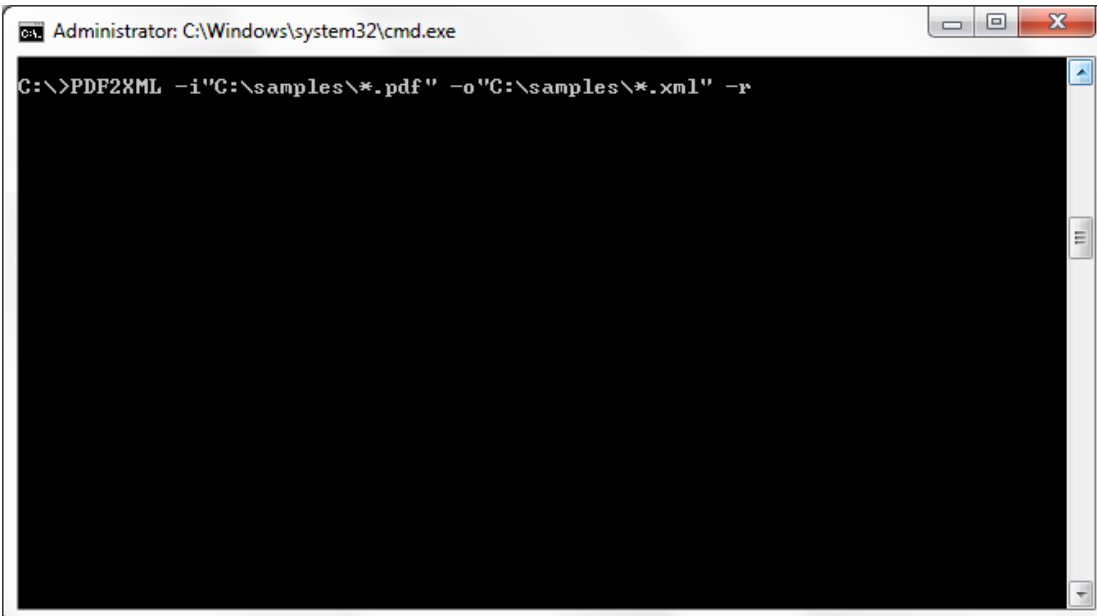
PDF2XML allows you to convert multiple files recursively i.e. by instructing the program to go through the current folder and all subfolders and convert all available PDF files automatically.

Note: the converted files will be saved to the same folders as the source files.

In order to convert the files recursively, use the `-r` switch. This switch is used without any additional arguments, but the absolute path to both input and output folders has to be specified. In the example below, the issued command:

```
PDF2XML -i"C:\samples\*.pdf" -o"C:\samples\*.xml" -r
```

will convert all PDF files within the current folder and all subfolders keeping the original file names and placing the converted files into same folders as the source files.

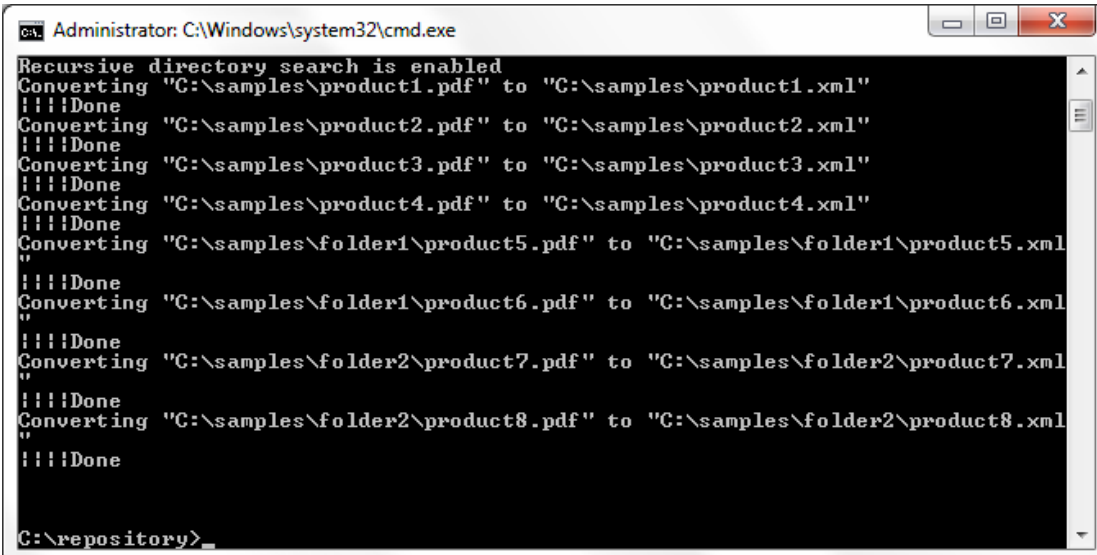


```
Administrator: C:\Windows\system32\cmd.exe
C:\>PDF2XML -i"C:\samples\*.pdf" -o"C:\samples\*.xml" -r
```

Figure 23: Converting the Files Recursively (step 1/2)

Press **Enter** after issuing the command and PDF2XMLCL will begin the conversion.

Note: the absolute path to both input and output folders has to be specified.



```
Administrator: C:\Windows\system32\cmd.exe
Recursive directory search is enabled
Converting "C:\samples\product1.pdf" to "C:\samples\product1.xml"
!!!Done
Converting "C:\samples\product2.pdf" to "C:\samples\product2.xml"
!!!Done
Converting "C:\samples\product3.pdf" to "C:\samples\product3.xml"
!!!Done
Converting "C:\samples\product4.pdf" to "C:\samples\product4.xml"
!!!Done
Converting "C:\samples\folder1\product5.pdf" to "C:\samples\folder1\product5.xml"
!!!Done
Converting "C:\samples\folder1\product6.pdf" to "C:\samples\folder1\product6.xml"
!!!Done
Converting "C:\samples\folder2\product7.pdf" to "C:\samples\folder2\product7.xml"
!!!Done
Converting "C:\samples\folder2\product8.pdf" to "C:\samples\folder2\product8.xml"
!!!Done
C:\repository>
```

Figure 24: Converting the Files Recursively (step 2/2)

In the above example, the **-R** switch has been used to recursively convert all PDF files within the **C:\Samples** folder including the **folder1** and **folder2** subfolders.

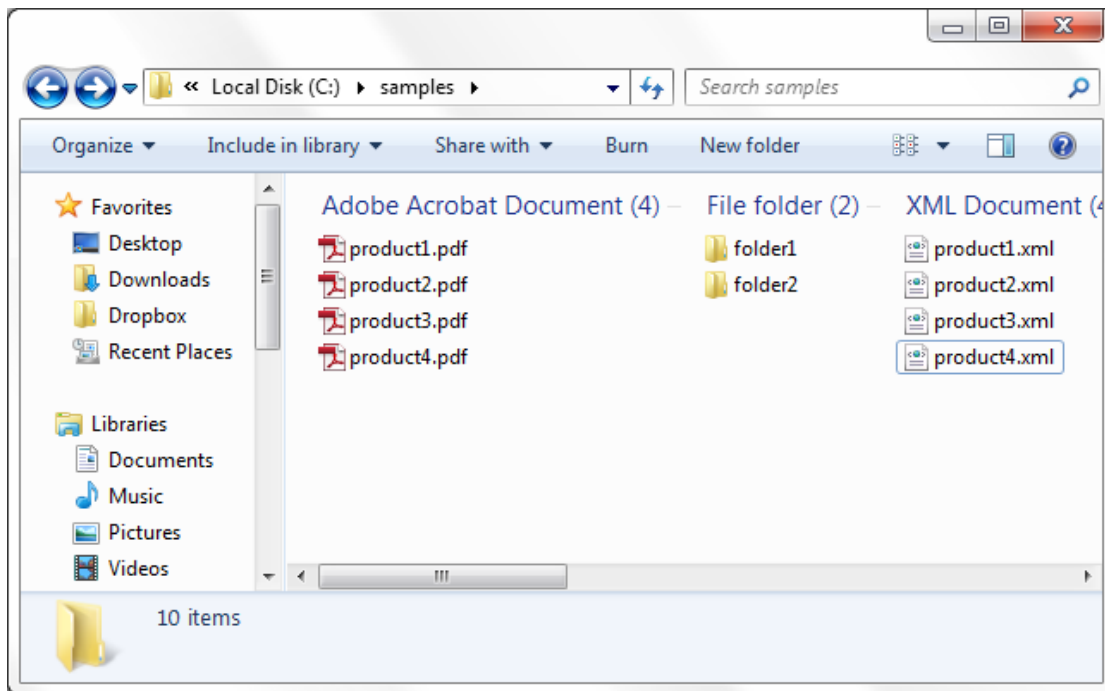


Figure 25: Example of the converted files

4.3 Force Deletion of Files with the Same Name

Normally, PDF2XML aborts the conversion if there is already a file with the same name in the output directory.

However, PDF2XML allows you to use a switch to replace the file with the same name i.e. force the deletion of such file. The switch which used for this purpose is **-d**. There are no additional arguments.

The following command has been issued in the example below:

```
PDF2XML -i"C:\repository\myfile.pdf" -o"C:\repository\myfile.xml" -d
```

This will convert a file called **myfile.pdf** to a file called **myfile.xml** even if the target directory already contains the file by the same name.

In other words, the pdf file will be converted to xml and the existing file will be deleted in case the deletion is possible (the file is not read-only, already opened in another program, etc).

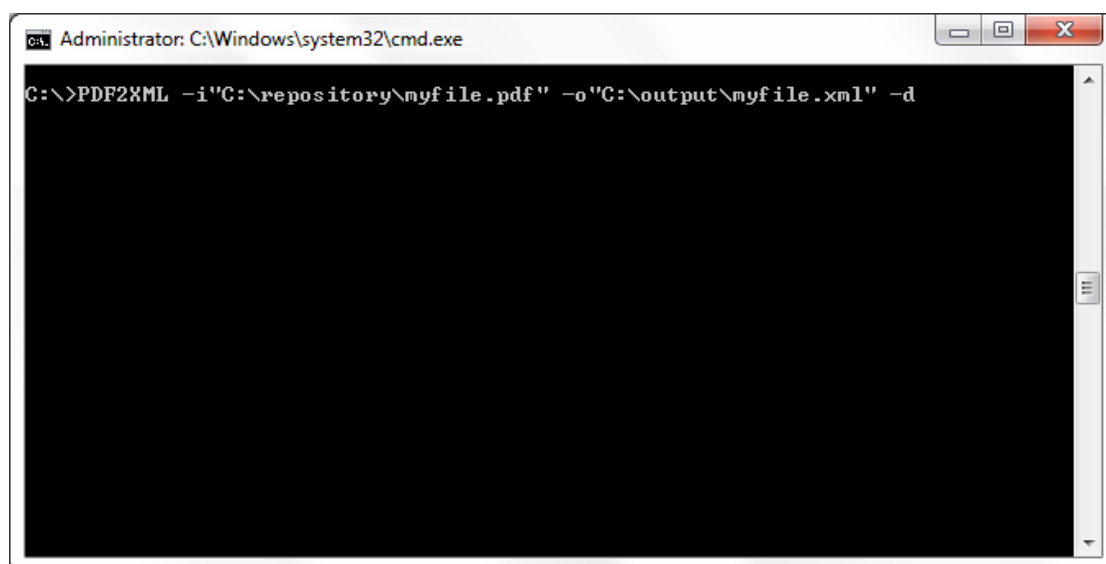
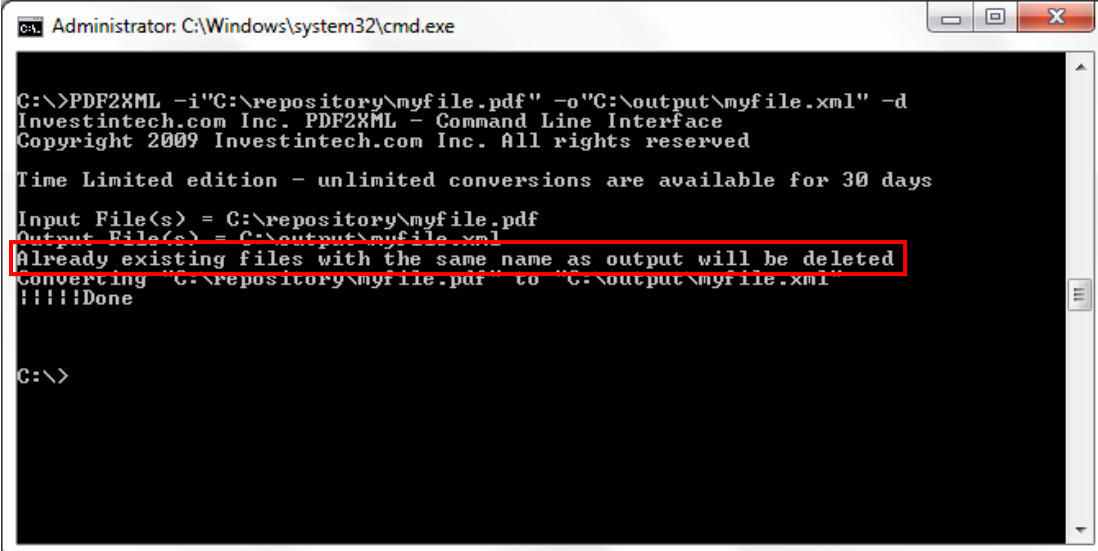


Figure 26: Force Deletion of Files with the Same Name (step 1/2)

The following screen will be displayed after issuing the command.



```
Administrator: C:\Windows\system32\cmd.exe

C:\>PDF2XML -i"C:\repository\myfile.pdf" -o"C:\output\myfile.xml" -d
Investintech.com Inc. PDF2XML - Command Line Interface
Copyright 2009 Investintech.com Inc. All rights reserved

Time Limited edition - unlimited conversions are available for 30 days

Input File(s) = C:\repository\myfile.pdf
Output File(s) = C:\output\myfile.xml
Already existing files with the same name as output will be deleted
Converting C:\repository\myfile.pdf to C:\output\myfile.xml
!!!!Done

C:\>
```

Figure 27: Force Deletion of Files with the Same Name (step 2/2)

Note:

- Once you issue the command with the `-d` switch PDF2XML will inform you that the existing files with the same name will be deleted.
- It is important to note that you will not be asked for confirmation issuing this command.

5 SUPPORT

5.1 Customer Service and Technical Support

Investintech.com Inc. strives to provide the best possible technical support to its customers and prospective customers. If you would like personal assistance, please feel free to call us or send e-mail to Customer Service or Technical Support.

We are available by phone during regular business hours and in the vast majority of cases, we will return our e-mail with an answer the same day it is received.

- **Telephone:** +1 416 920 5884 +1 416 920 2539
- **E-mail:** cs@investintech.com techsupport@investintech.com
- **Hours:** Our Business Hours are Monday to Friday 9am-6pm (Eastern Time GMT-5:00).
- **Fax and Mailing Address**
 - Mailing Address:
 - Investintech.com Inc.
 - 301 - 425 University Avenue
 - Toronto, ON
 - M5G 1T6 Canada
 - Fax: +1 416 920 5848

5.2 How to Submit an Issue

Unlike many other companies, we appreciate the feedback, the ideas and the tech support queries we receive.

We respond to all queries we receive and we do our best to resolve any problems related to our products. To get the fastest possible response/resolution to your issue, please do the following:

- Briefly describe your problem in the body of your e-mail
- Attach the problematic source file (in most cases the PDF document). We make this request solely for problem solving purposes. We do not look at the document's contents and we will not share your file with any third parties. Furthermore, we will destroy your document when we are done with it.
- Send the above to techsupport@investintech.com

6 TROUBLESHOOTING

There are a few cases in which PDF2XML CL is unable to perform the requested file conversion. Please consult the table below if you experience errors while working with PDF2XML CL.

Error/Question	Cause	Solution
Unable to convert – did not find input document file name expression.	<ol style="list-style-type: none"> 1. Misspelled or omitted the input file parameter. 2. It is also possible you do not have permissions to access input file. 3. In case the file path contains spaces, it has to be specified using quotes (e.g. "C:\Program Files\Folder\myfile.pdf") – this rule applies to both input and output file paths 	<ol style="list-style-type: none"> 1. Use -i<filename.pdf> expression to specify PDF document file(s) which should be converted. 2. Alternatively, contact your system administrator to give you permissions for reading the input file. 3. If there are spaces in the file path, use quotes.
Unable to convert – did not find output document file name expression.xml	<ol style="list-style-type: none"> 1. Misspelled or omitted input document file parameter. 2. It is also possible you do not have permissions to access input file 3. In case the file path contains spaces, it has to be specified using quotes (e.g. "C:\Program Files\Folder\myfile.pdf") – this rule applies to both input and output file paths 	<ol style="list-style-type: none"> 1. Use -o<filename.xml> to specify file(s) that will contain converted information. 2. Try a path for which you have the write permissions. 3. If there are spaces in the file path, use quotes.
Empty converted document when using page range	Page range is invalid e.g. -p50-40	Use ascending order in page range e.g. -p40-50
Task not scheduled when using schedule parameter.	Schedule time specified incorrectly.	Use a formatted 24 hour time with valid optional recurring sub parameters e.g. -s:30:00 -s06:30:00:/next:1 -3:59:59:/every:M,T,W,Th,F

Error/Question	Cause	Solution
Input file is damaged or not a valid PDF file!	Input file was found but it is either not PDF file or it is damaged and/or corrupted.	<p>If Acrobat Reader can open the file without any error message please send the file to our technical support and we will look into it.</p> <p>If Adobe Acrobat (full version) can open the file with an error message, please save the file back – this way Adobe Acrobat will repair the file. Then try conversion again.</p>
You can only use the tool from its root folder.	The program hasn't been added to the path.	Restart the computer and you will be able to use the tool from any folder on your file system.
Program not responding	Designated input file may be too large for the system to handle.	<p>Our software does not put any restrictions on the file sizes it can open and convert.</p> <p>The cause of the non-response may be system or memory related.</p> <p>Older computers with less virtual memory may have a more difficult time opening and converting a very large PDF file.</p> <p>If you think this is the case you can send us the file to techsupport@investintech.com</p>
The tool doesn't work properly if started from a service application.	The Service application doesn't have required permission rights on the file system.	<p>Change the account which is used for running the Windows Service to a Local System Account (or Administrator Account).</p> <p>Also, if you don't want to switch accounts, make sure that the service account has enough permissions for accessing the file system.</p>

Error/Question	Cause	Solution
<p>There are no reported errors, but the resulting files are missing.</p>	<p>To protect against some types of viruses, malwares, etc. – Windows Vista and Windows 7 do not allow users to store data within certain System folders (for example: C:\Windows, C:\Program Files, etc, or their subfolders).</p> <p>Files are automatically and transparently moved into the VirtualStore folder.</p> <p>So, if they are saved by the user into C:\Windows, they are automatically moved into:</p> <p>C:\Users\”User name”\AppData\Local\VirtualStore\Windows\...</p> <p>Similarly, if they are saved by the user into: C:\Program Files, they are automatically moved into:</p> <p>C:\Users\”User name”\AppData\Local\VirtualStore\Program Files\...</p>	<p>Look for the files within the VirtualStore folder.</p>
<p>The message “Access is denied” is displayed when trying to schedule a conversion.</p>	<p>Inadequate permissions.</p>	<p>Try running the command prompt as an administrator.</p>

Table 2: Troubleshooting

Note: In case the problem you are having is not described in the table, please feel free to contact us at techsupport@investintech.com and we will work to remedy the situation as quickly as possible.