

**Investintech.com Inc.
Software Development Kit:
PDF Create SDK User's Guide**

February 27, 2009
<http://www.investintech.com>

Copyright 2008, 2009 Investintech.com, Inc. All rights reserved

Adobe® is registered by Adobe Systems Incorporated
Acrobat® is registered by Adobe Systems Incorporated
Postscript® is registered by Adobe Systems Incorporated
Access™ is registered by Microsoft Corporation
Excel® is registered by Microsoft Corporation
Visual Basic® is registered by Microsoft Corporation
Visual C++® is registered by Microsoft Corporation
Visual C#® is registered by Microsoft Corporation
Visual J#® is registered by Microsoft Corporation
Visual Studio® is registered by Microsoft Corporation
Win32® is registered by Microsoft Corporation
Windows® is registered by Microsoft Corporation
Windows NT® is registered by Microsoft Corporation
Windows Vista™ is registered by Microsoft Corporation

Contents

Preface	1
About This Documentation	1
Typographical Conventions Used in This Document	1
Getting More Information	2
Customer Service and Technical Support	2
Fax and Mailing Address.....	2
Investintech PDFCreate SDK	3
What is the Investintech PDFCreate SDK?.....	4
Installation Instructions for PDFCreate SDK.....	4
Starting the Installation.....	4
Using the Investintech PDF Create SDK in Microsoft Visual C++ .NET 2003	12
Implicit Linking.....	12
Linking .lib file with project	13
Using methods from DLL file.....	14
Investintech PDF Create DLL Methods.....	14
Interface	15
Parameter Type.....	15
File Names	15
Error Handling.....	16
Common Sample Source Code.....	16
<i>PDF creation from File</i>	17
PDF_Create_CBR	17
Prototype	17
Description	17
Calling Convention	17
Parameters	17
Returns	17
Example	17
PDF_Create_CBT.....	17
Prototype	17
Description	18
Calling Convention	18
Parameters	18
Returns	18
Example	18
PDF_Create_CLR.....	18
Prototype	18
Description	18
Calling Convention	19
Parameters	19
Returns	19
Example	19
PDF_Create_CLT.....	19
Prototype	19
Description	19
Calling Convention	19

Parameters.....	19
Returns.....	20
Example.....	20
PDF_Create_SBR.....	20
Prototype.....	20
Description.....	20
Calling Convention.....	20
Parameters.....	20
Returns.....	21
Example.....	21
PDF_Create_SBT.....	21
Prototype.....	21
Description.....	21
Calling Convention.....	21
Parameters.....	21
Returns.....	21
Example.....	21
PDF_Create_SLR.....	22
Prototype.....	22
Description.....	22
Calling Convention.....	22
Parameters.....	22
Returns.....	22
Example.....	22
PDF_Create_SLT.....	23
<i>Prototype</i>	23
Description.....	23
Calling Convention.....	23
Parameters.....	23
Returns.....	23
Example.....	23
VB6 callable code.....	24
PDF_Create_VB6.....	24
Prototype.....	24
Description.....	24
Calling Convention.....	24
Parameters.....	24
Returns.....	24
Example.....	24
Investintech PDF Create COM Server Methods.....	25
CreatePDF.....	25
Prototype.....	25
Description.....	25
Parameters.....	25
Returns.....	25
Example.....	25
Index	41

Preface

Welcome to Investintech.com Inc. Software Development Kit (SDK). This SDK provides technologies for creating PDF files from many common file formats. This user's guide provides the basic information needed to use all parts of the SDK. This SDK consists of:

- Investintech PDF Create SDK, a Dynamic-Link Library for creating PDF files.
- Investintech PDF Create COM, a COM component for creating PDF files.

This document provides a comprehensive description of the SDK.

About This Documentation

This documentation consists of:

- Investintech PDF Create DLL, describing the usage and methods of the DLL designed to create PDF files.
- Investintech PDF Create COM, describing the usage and methods of the COM component designed to create PDF files.

Typographical Conventions Used in This Document

The following table describes typographical and naming conventions used in this document:

<i>Font</i>	<i>Used for</i>	<i>Examples</i>
monospaced	Expected input from the user	pdfcreate -isource.doc -odest.pdf
	Paths and filenames	c:\Program Files\Investintech\PDFCrea te\pdfcreatedll.dll
	Source code	MessageBox("ABC");
monospaced blue	Source code keywords	If
monospaced green	Source code comments	//this is a comment
monospaced bold	Method name	The AboutBox() method
'Single quoted bold'	Elements of the user interface like buttons, icons etc.	click on ' Start ' button, press ' Enter '
blue underlined	Internet links	http://www.investintech.com

Getting More Information

Investintech.com Inc. conducts research in data conversion technology and develops data conversion products utilizing results of the research. More information about our company is available at the company's website <http://www.investintech.com>.

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.

	<i>Customer Service</i>	<i>Technical Support</i>
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.
600 - 425 University Avenue
Toronto, ON
M5G 1T6 Canada

Fax: +1 416 920 5848

Investintech.com Inc. Software Development Kit User's Guide

Investintech PDFCreate SDK

What is the Investintech PDFCreate SDK?

The Investintech PDFCreate SDK is a collection of methods compiled, linked and stored in a dynamic-link library (DLL) file. These methods provide a set of methods for converting files from common file formats to PDF format.

Installation Instructions for PDFCreate SDK

Below you will find step-by-step instructions to install PDFCreate SDK on your system. You may want to print this instruction sheet for reference before beginning to install PDFCreate SDK.

System Requirements

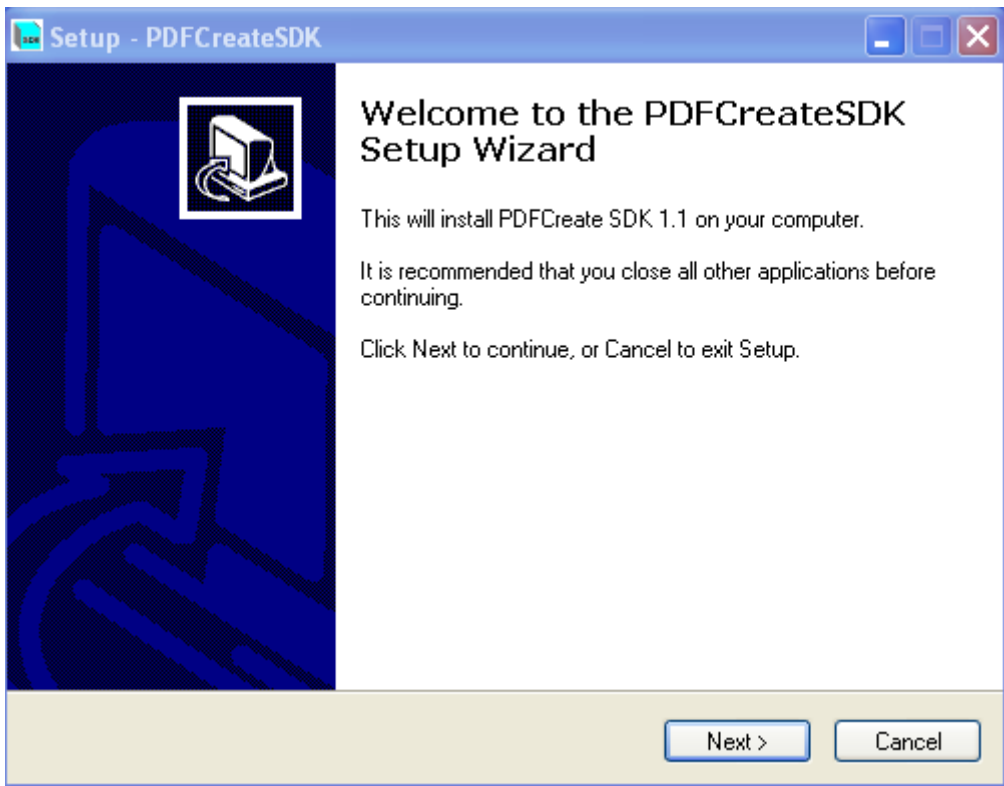
The minimum computer system resources required to install and use PDFCreate SDK are:

- Microsoft Windows XP
- 64 MB of RAM
- 32 MB of available hard disk space
- A software development environment, such as Microsoft Visual Studio or Borland Delphi

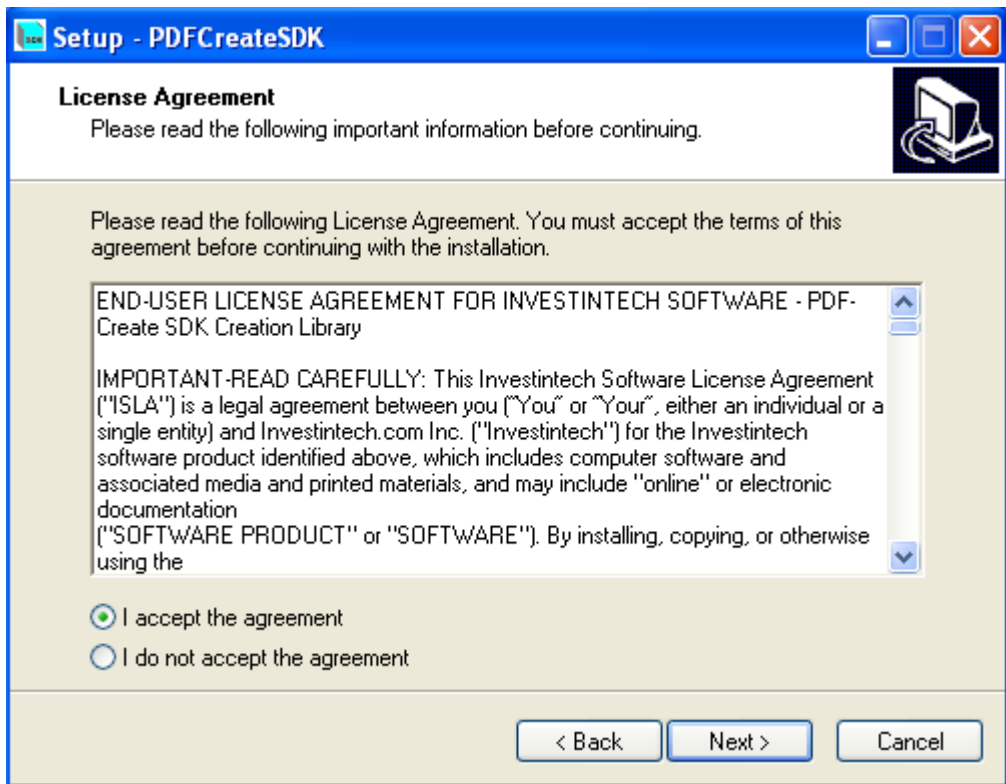
Current mainstream desktop machines greatly exceed the minimum required resources and you are recommended to use the best available machine in order to maximize your development experience.

Starting the Installation

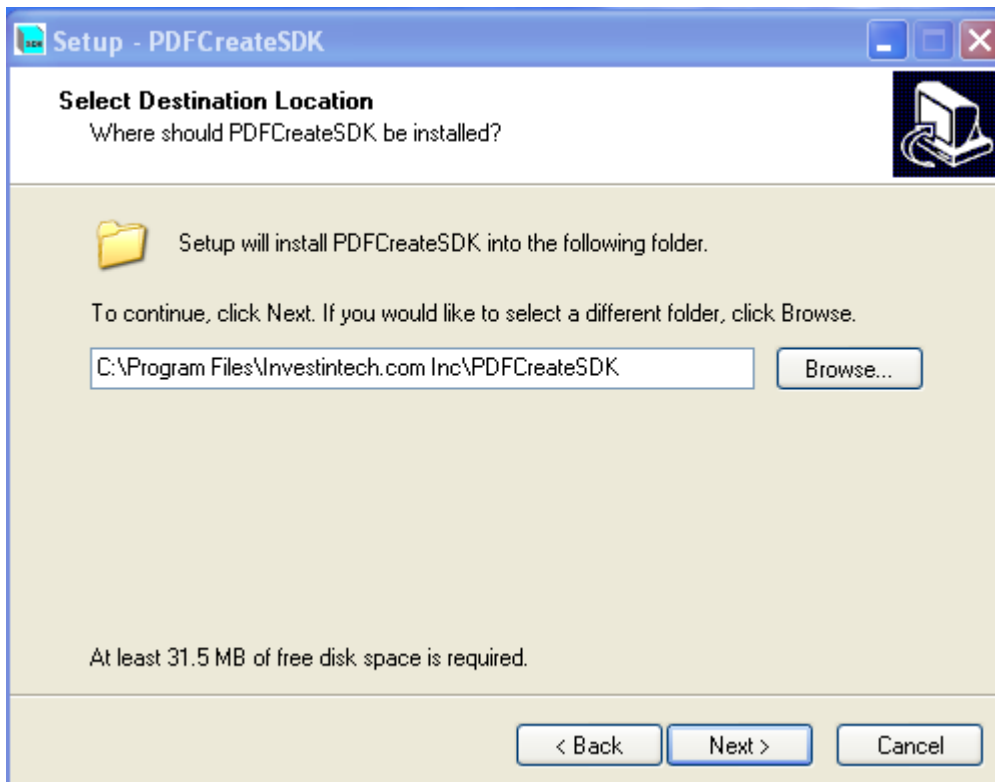
Start the installation by launching the `InstallPDFCreateSDK.exe` file, e.g. by double clicking it in the Windows Explorer. After that you should see a window that looks like this:



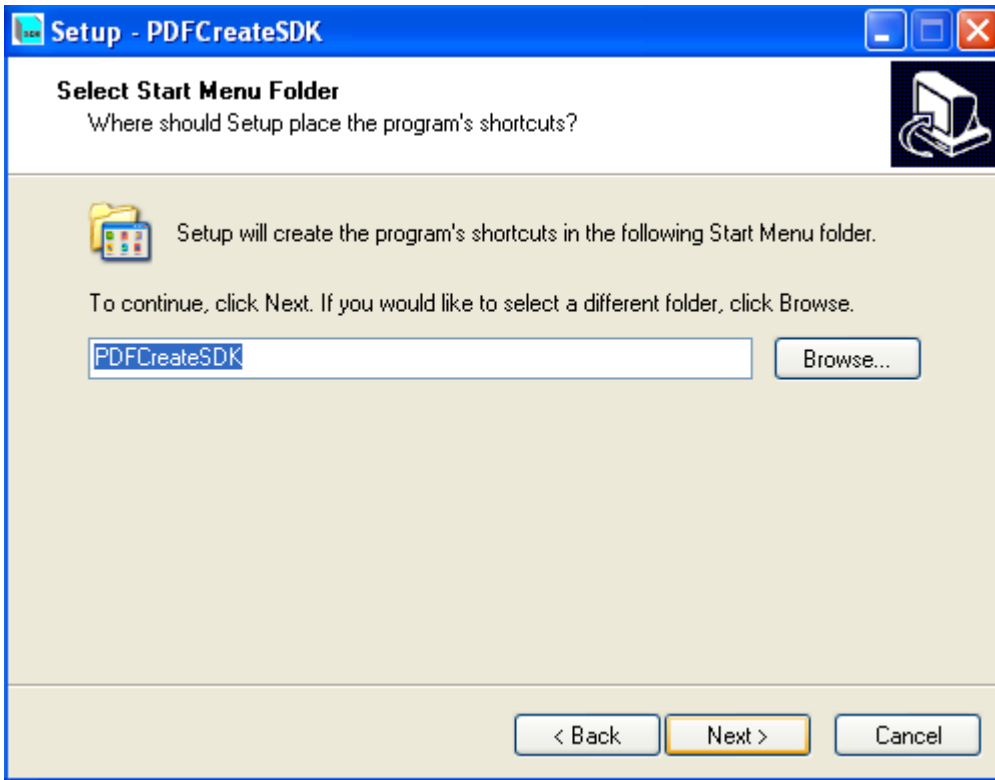
Click the 'Next' button to continue installation. You should see the next window:

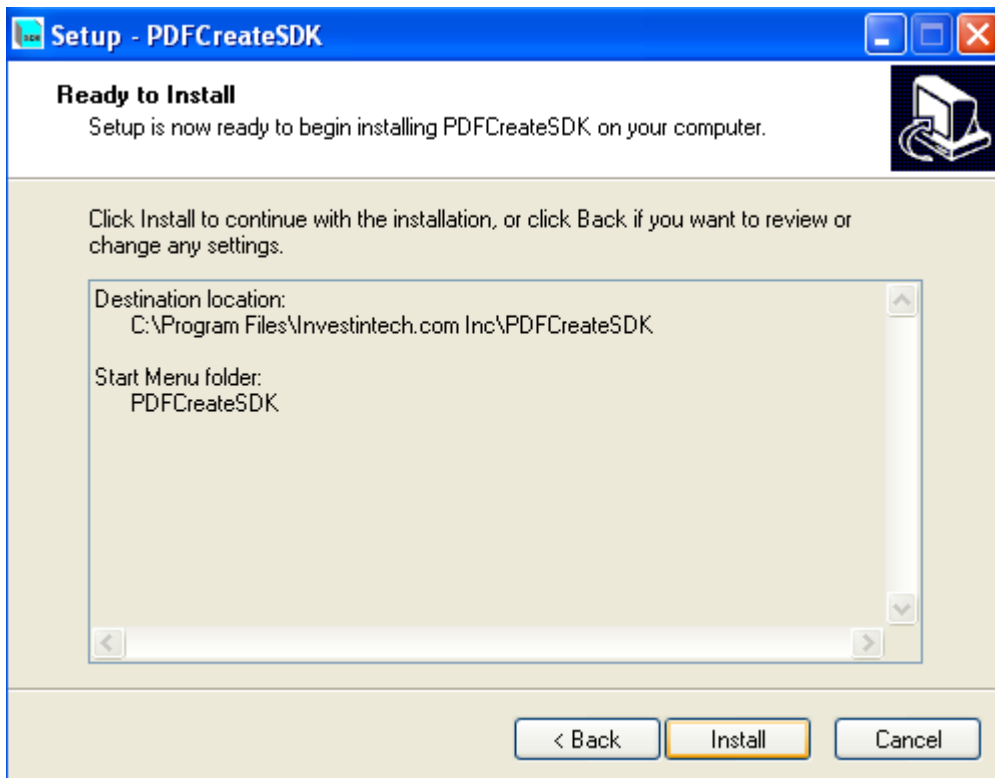


Please read the Investintech Software License Agreement carefully and thoroughly. If you wish to continue with installation, choose **'I agree with the above terms and conditions'** and click **'Next'**. Otherwise, click **'Exit'** and installation will terminate. If you clicked **'Next'** you should see the next window:



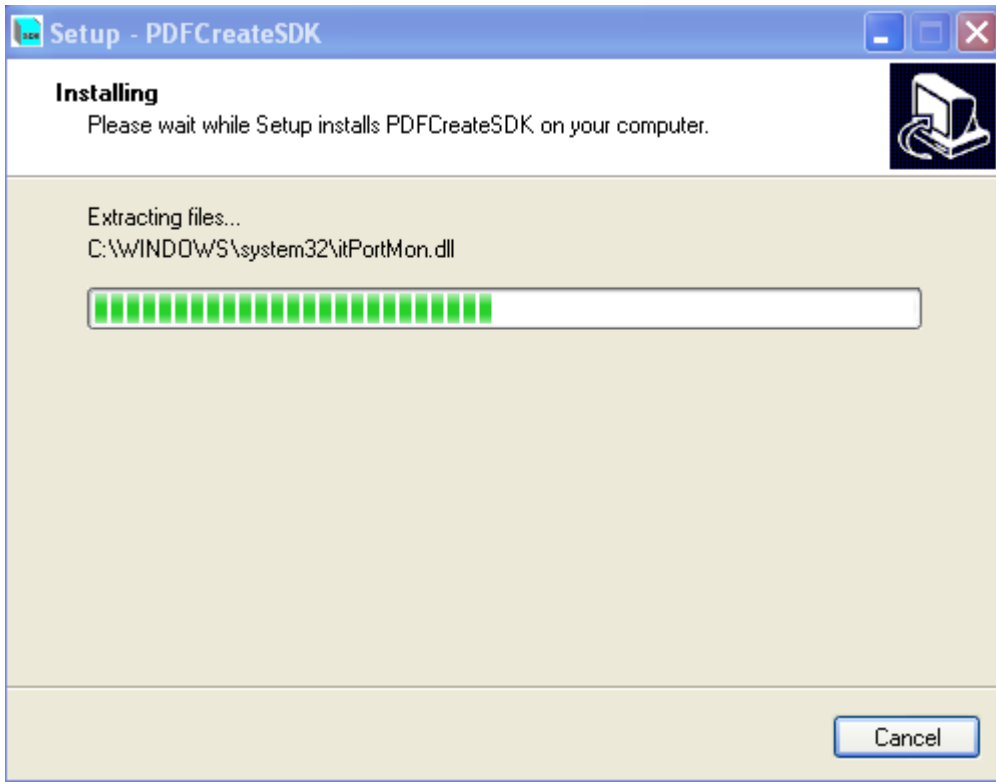
On this screen you can choose the installation directory for the PDF Create SDK software. The default installation folder will be contained in the `Program Files` folder on your system disk. It is possible to change the location of installation folder either by typing its path directly or by clicking the 'Browse...' button. If the destination folder (directory) you have chosen does not exist, the installation program will create it. Click the '**Next**' button to continue the installation.



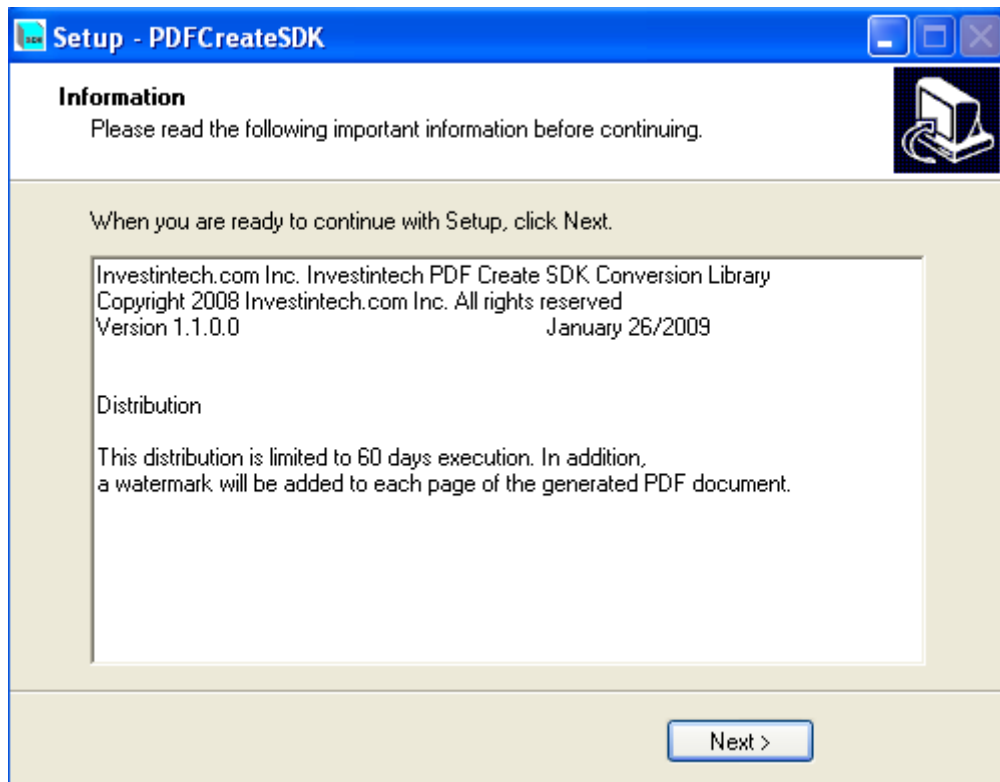


Clicking the **'Install'** button will start the installation.

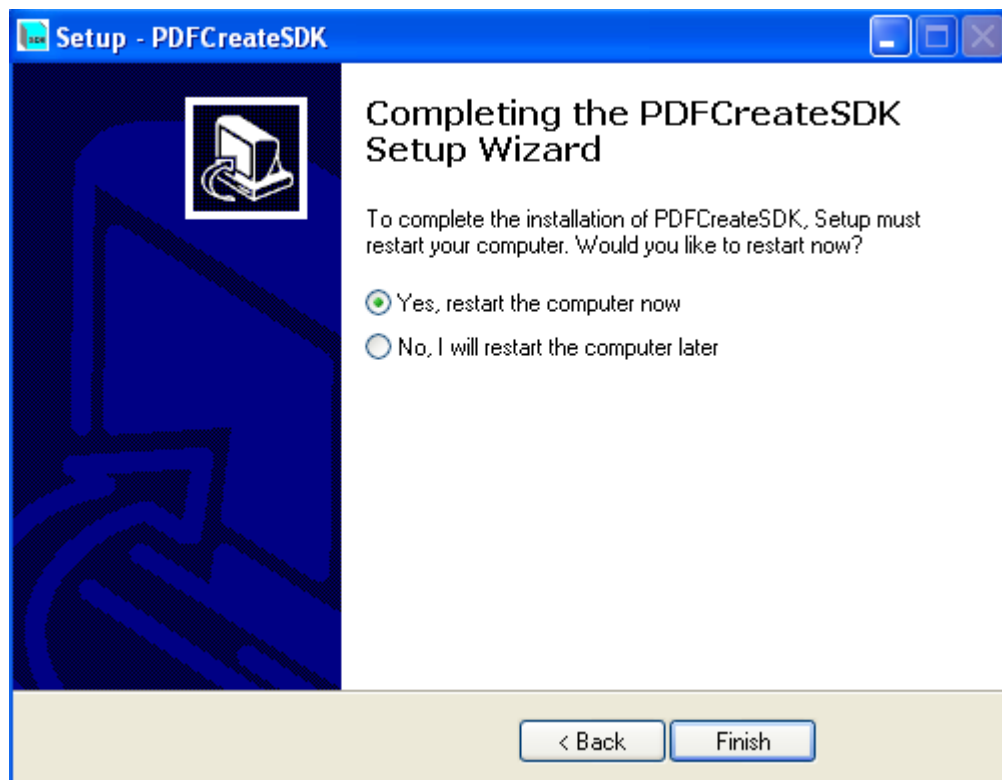
Investintech PDFCreate SDK



The installation will copy the files that comprise PDF Create SDK to the directory you specified earlier. After a very short period of time, usually a few seconds, you should see the next window:



This windows describes the distribution of PDF Create SDK being installed. Click 'Next' to continue.



When this window appears, the installation process is complete. Click **Finish** to end the installation program.

Using the Investintech PDF Create SDK in Microsoft Visual C++ .NET 2005

In this section you will learn how to use the Investintech PDF Create DLL. All image and examples from now on will be based on Microsoft Visual C++ .NET 2005.

In order to use methods contained in DLL you have to link an executable file to DLL. Executable file can be either EXE or DLL file. There are two possible ways of linking: implicit and explicit linking. In this documentation implicit linking will be shown.

For more implementation details, please see the sample code located in the samples\lib folder of the PDF_Create_SDK installation folder.

Implicit Linking

Implicit linking is load-time linking. The following is needed for executables to implicitly link to Investintech Create PDF DLL:

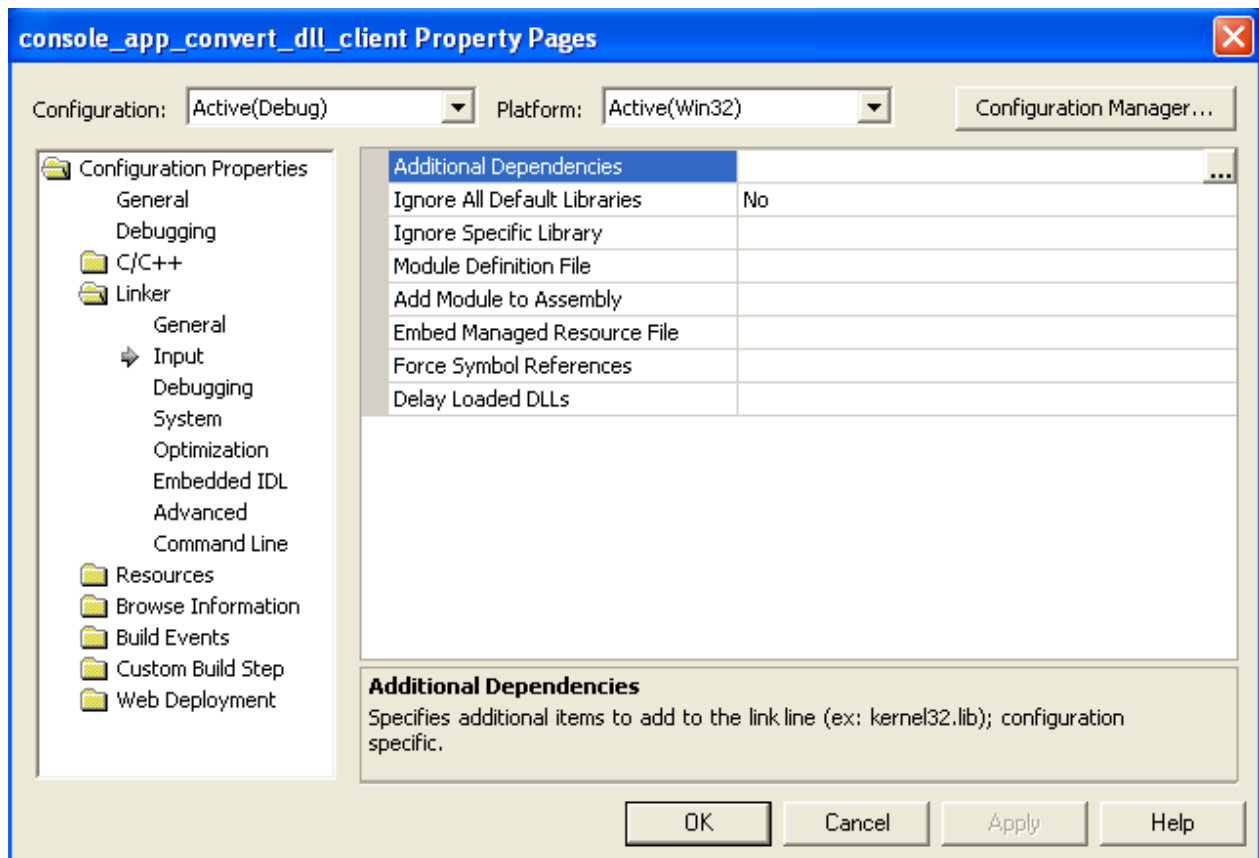
- PDF_Create_SDK.h, the header file which contains declarations of exported methods
- PDF_Create_SDK.lib, an import library used by the linker
- PDF_Create_SDK.dll, the Dynamic Link Library file

These files are located in the PDF Create SDK installation directory.

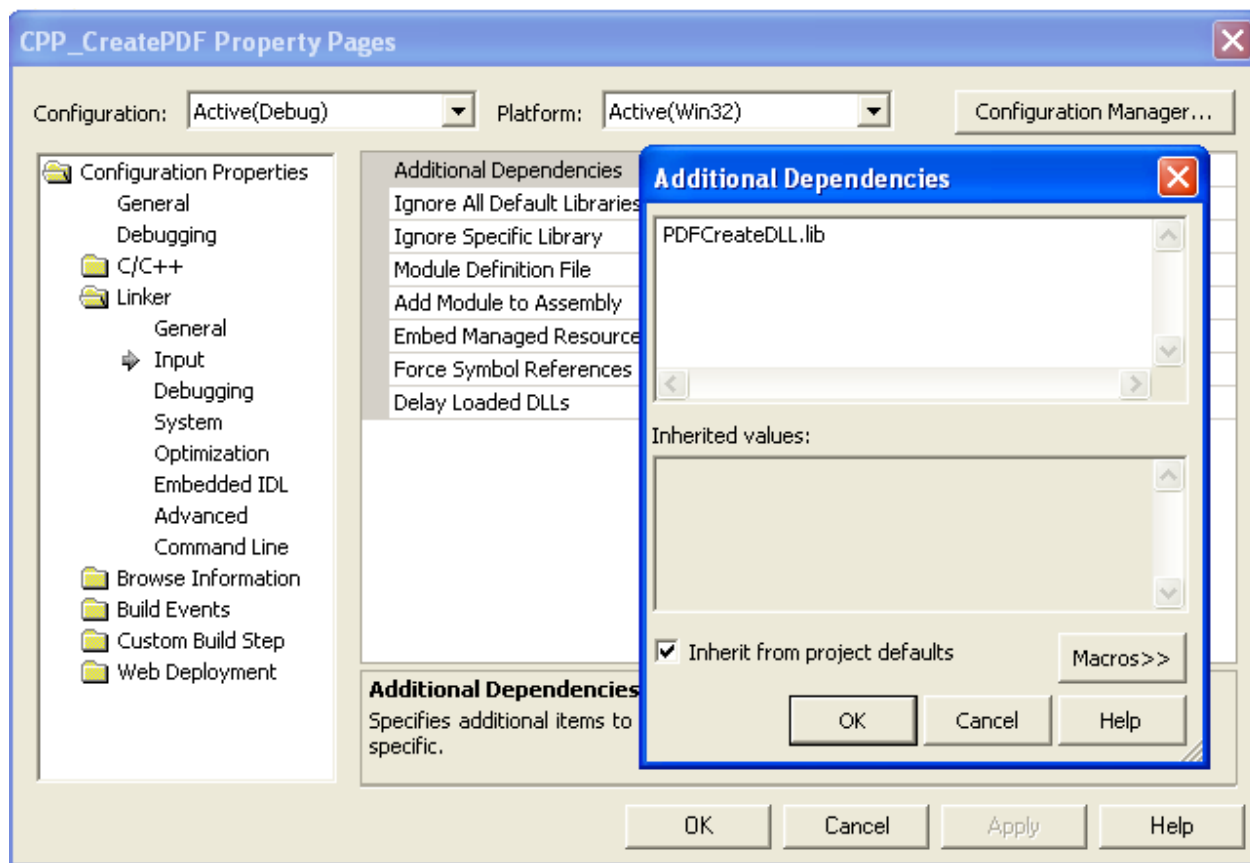
Linking .lib file with project

Below you will find step-by-step instructions on how to link PDF_Create_SDK.lib file with your project.

Start Visual Studio and select '**File > Open > Project**'. Select the project in which you want to use PDF_Create_SDK.dll and click '**OK**'. Select '**Project > Properties**'. You should see the window like next one:



Click the '**Linker**' folder and open '**Input**' property page. Modify the '**Additional Dependencies**' property by clicking the '...' button.



The window titled “Additional Dependencies” should open: type `PDF_Create_SDK.lib` and click the ‘OK’ button. Click ‘OK’ to close “Property Pages” window. You have successfully linked `PDF_Create_SDK.lib` with your project.

Using methods from DLL file

After linking with `PDF_Create_SDK.lib` file you can use methods contained in `PDF_Create_SDK.dll` file. Simply include the `PDF_Create_SDK.h` file in all source files that need to use those methods.

PDF Create SDK Methods

In this section you will find details about each method such as: prototype; description; calling conventions; parameters; return value; and sample use of the methods.

Interface

The interface of the Investintech PDF Create SDK consists of its exposed methods. The interface contains an exposed method for each combination of: calling convention; parameter type; and error handling.

The calling convention is either the “cdecl” calling convention or the “stdcall” calling convention.

The “cdecl” calling convention is normally used for calling C/C++ functions. It is characterized by the following points:

- arguments are passed in order from right to left
- the calling function is responsible for popping the arguments from the stack at the conclusion of function execution
- an underscore character (`_`) is prefixed to function names
- no case translation is performed on the function name

The “stdcall” calling convention is typically used to call Windows operating system API functions. It is characterized by the following points:

- arguments are passed in order from right to left
- arguments are passed by value, unless a pointer or reference type is passed.
- The called function is responsible for popping its own arguments from the stack.
- an underscore (`_`) is prefixed to the name. The name is followed by the at sign (`@`) followed by the number of bytes (in decimal) in the argument list. For instance, the function declared as `void foo(int bar)` is decorated as follows: `_foo@4`
- no case-translation is performed on the function name

Parameter Type

The parameter type is either ANSI (C char *) or UNICODE (BSTR).

File Names

Files may be specified by supplying relative or absolute file names. A relative file name is relative to the application default directory containing the application executable file (e.g. “`..\in_parent_dir.pdf`”, “`in_application_dir.pdf`”, “`subdir\in_sub_directory.pdf`”). An absolute file name includes the directory path to the file beginning at the top of the directory tree (e.g. “`c:\pdfs\absolute_file.pdf`”).

It is also possible to use UNC notation (e.g. [\\myserver\public\file.pdf](#)) provided your user account has sufficient access permissions.

Error Handling

The error handling strategy is either to throw an exception (part of the C++ throw-catch error handling construct) or return in integer/boolean type result code (the C style error handling strategy).

The Investintech Conversion DLL identifies the characteristics of a function by an encoded suffix. The suffix is defined by the rules:

- C or S - cdecl or stdcall calling convention
- B or L - BSTR or LPST parameter type
- R or T - return error code or throw exception

Common Sample Source Code

All sample code snippets share the following initialization code snippet:

```
//.h file must be included if we want to use conversion methods
#include <afx.h>
#include <afxwin.h>           // MFC core and standard components
#include "PDF_CREATE_SDK.h"

CString str_inputFile = _T("c:\\home\\in.rtf"); //input file name
CString str_outputFile = _T("c:\\home\\out.pdf"); //output file
name
CString str_ticketFile = _T("c:\\home\\topdf.tikd"); //ticket
file name

//conversion from CString to BSTR
BSTR bstr_inputFile = str_inputFile.AllocSysString();
BSTR bstr_outputFile = str_output.AllocSysString();
BSTR bstr_ticketFile = str_ticketFile.AllocSysString();
```

PDF creation from File

The following methods convert a file to a PDF document file.

PDF_Create_CBR

Prototype

```
bool PDF_Create_CBR(BSTR inFile, BSTR outFile, BSTR ticketFile)
```

Description

The PDF_Create_CBR() method creates a PDF document file from the user specified file. The original file is not modified.

Calling Convention

The function uses the “cdecl” calling convention.

Parameters

inFile	string containing the absolute or relative filename of the source file.
outFile	string containing the absolute or relative filename of the PDF file that will be created.
ticketFile	string containing the absolute or relative filename of the ticket file containing creation parameters.

Returns

true if PDF file is successfully created, false otherwise.

Example

```
bool ret = PDF_Create_CBR(bstr_inputFile, bstr_outputFile,  
bstr_ticketFile);
```

PDF_Create_CBT

Prototype

```
bool PDF_Create_CBT(BSTR inFile, BSTR outFile)
```

Description

The PDF_Create_CBT() method creates a PDF document file from the user specified file. The original file is not modified.

Calling Convention

The function uses the “cdecl” calling convention.

Parameters

inFile	string containing the absolute or relative filename of the source file.
outFile	string containing the absolute or relative filename of the PDF file that will be created.
ticketFile	string containing the absolute or relative filename of the ticket file containing creation parameters.

Returns

true if PDF file is successfully created, otherwise throws an exception that should be handled by the calling application.

Example

```
try
{
    bool ret = PDF_Create_CBT(bstr_inputFile, bstr_outputFile, bstr_ticketFile);
}
catch(...)
{
    // TODO: Handle exception
}
```

PDF_Create_CLR

Prototype

```
bool PDF_Create_CLR(LPSTR inFile, LPSTR outFile, LPSTR
ticketFile)
```

Description

The PDF_Create_CLR() method creates a PDF document file from the user specified file. The original file is not modified.

Calling Convention

The function uses the “cdecl” calling convention.

Parameters

inFile	string containing the absolute or relative filename of the source file.
outFile	string containing the absolute or relative filename of the PDF file that will be created.
ticketFile	string containing the absolute or relative filename of the ticket file containing creation parameters.

Returns

true if PDF file is successfully created, false otherwise.

Example

```
bool ret = PDF_Create_CLR(str_inputFile.GetBuffer(), str_outputFile.GetBuffer(),  
str_ticketFile.GetBuffer() );
```

PDF_Create_CLT

Prototype

```
bool PDF_Create_CLT(LPSTR inFile, LPSTR outFile, LPSTR  
ticketFile)
```

Description

The PDF_Create_CLT() method creates a PDF document file from the user specified file. The original file is not modified.

Calling Convention

The function uses the “cdecl” calling convention.

Parameters

inFile	string containing the absolute or relative filename of the source file.
outFile	string containing the absolute or relative filename of the PDF file that will be created.
ticketFile	string containing the absolute or relative filename of the ticket file containing

creation parameters.

Returns

true if PDF file is successfully created, otherwise throws an exception that the calling application should handle.

Example

```
try
{
    bool ret = PDF_Create_CLT(str_inputFile.GetBuffer(), str_outputFile.GetBuffer(),
str_ticketFile.GetBuffer() );
}
catch(...)
{
    // TODO: Handle exception
}
```

PDF_Create_SBR

Prototype

```
UINT PDF_Create_SBR(BSTR inFile, BSTR outFile, BSTR ticketFile)
```

Description

The PDF_Create_SBR () method creates a PDF document file from the user specified file. The original file is not modified.

Calling Convention

The function uses the “stdcall” calling convention.

Parameters

inFile	string containing the absolute or relative filename of the source file.
outFile	string containing the absolute or relative filename of the PDF file that will be created.
ticketFile	string containing the absolute or relative filename of the ticket file containing creation parameters.

Returns

Non-zero integer value if PDF file is successfully created, zero integer value otherwise.

Example

```
UINT ret = PDF_Create_SBR(bstr_inputFile, bstr_outputFile, bstr_ticketFile);
```

PDF_Create_SBT

Prototype

```
UINT PDF_Create_SBT(BSTR inFile, BSTR outFile, BSTR ticketFile)
```

Description

The PDF_Create_SBT() method creates a PDF document file from the user specified file. The original file is not modified.

Calling Convention

The function uses the “stdcall” calling convention.

Parameters

inFile	string containing the absolute or relative filename of the source file.
outFile	string containing the absolute or relative filename of the PDF file that will be created.
ticketFile	string containing the absolute or relative filename of the ticket file containing creation parameters.

Returns

Non-zero integer value if PDF file is successfully created, otherwise throws an exception that the calling application should handle.

Example

```
try  
{  
    UINT ret = PDF_Create_SBT(bstr_inputFile, bstr_outputFile, bstr_ticketFile);  
}  
catch(...)
```

Investintech PDFCreate SDK

```
{  
    // TODO: Handle exception  
}
```

PDF_Create_SLR

Prototype

```
UINT PDF_Create_SLR(LPSTR inFile, LPSTR outFile, LPSTR  
ticketFile)
```

Description

The PDF_Create_SLR () method creates a PDF document file from the user specified file. The original file is not modified.

Calling Convention

The function uses the “stdcall” calling convention.

Parameters

inFile	string containing the absolute or relative filename of the source file.
outFile	string containing the absolute or relative filename of the PDF file that will be created.
ticketFile	string containing the absolute or relative filename of the ticket file containing creation parameters.

Returns

Non-zero integer value if PDF file is successfully created, zero integer value otherwise.

Example

```
UINT retCode = PDF_Create_SLR(str_inputFile, str_outputFile, str_ticketFile);
```

PDF_Create_SLT

Prototype

```
UINT PDF_Create_SLT(LPSTR inFile, LPSTR outFile, LPSTR  
ticketFile)
```

Description

The PDF_Create_SLT() method creates a PDF document file from the user specified file. The original file is not modified.

Calling Convention

The function uses the “stdcall” calling convention.

Parameters

inFile	string containing the absolute or relative filename of the source file.
outFile	string containing the absolute or relative filename of the PDF file that will be created.
ticketFile	string containing the absolute or relative filename of the ticket file containing creation parameters.

Returns

Non-zero integer value if PDF file is successfully converted to HTML, otherwise throws an exception that should be handled by the calling application.

Example

```
try  
{  
    UINT retCode = PDF_Create_SLT(str_inputFile, str_outputFile, str_TicketFile);  
}  
catch(...)  
{  
    //TODO: handle error  
}
```

VB6 callable code

VB6 client applications must use these functions in order to avoid runtime errors related to stack and string references.

PDF_Create_VB6

Prototype

```
UINT PDF_Create_VB6(BSTR inFile, BSTR outFile, BSTR ticketFile)
```

Description

The PDF_Create_VB6() method creates a PDF document file from the user specified file. The original file is not modified.

Calling Convention

The function uses the “stdcall” calling convention.

Parameters

<code>inFile</code>	string containing the absolute or relative filename of the source file.
<code>outFile</code>	string containing the absolute or relative filename of the PDF file that will be created.
<code>ticketFile</code>	string containing the absolute or relative filename of the ticket file containing creation parameters.

Returns

Non-zero integer value if PDF file is successfully created, zero integer value otherwise.

Example

```
UINT ret = PDF_Create_VB6(bstr_inputFile, bstr_outputFile, bstr_ticketFile);
```

PDF Create COM Server Methods

This section provides a description of the conversion methods exposed by the PDF Create COM Server. The COM library PDF_Create_COMLib defines the CoClass PDFCreate that implements the interface IPDFCreate. The interface consists of the following methods. For more information, please refer to the sample code in the samples\COM folder of the PDF_Create_SDK installation folder.

CreatePDF

Prototype

```
HRESULT CreatePDF([in] BSTR sourceFile, [in] BSTR destFile,  
[in] BSTR ticketFile, [out,retval] VARIANT_BOOL *successFlag)
```

Description

The CreatePDF() method creates a PDF document file from the user specified file. The original file is not modified.

Parameters

inFile	string containing the absolute or relative filename of the source file.
outFile	string containing the absolute or relative filename of the PDF file that will be created.
ticketFile	string containing the absolute or relative filename of the ticket file containing creation parameters.
SuccessFlag	Pointer to boolean variable that will contain the result of the operation: true is the conversion operation is successful; else false.

Returns

HRESULT enumerated value indicating the result of the COM operation.

Example

```
HRESULT result = CreatePDF(BSTR_sourceFile BSTR_destinationFile, BSTR_ticketFile,  
&boolFlag);
```


Appendix A - Tickets

This section provides a description of the ticket file and the Ticket editor, a GUI tool that may be used to edit ticket files.

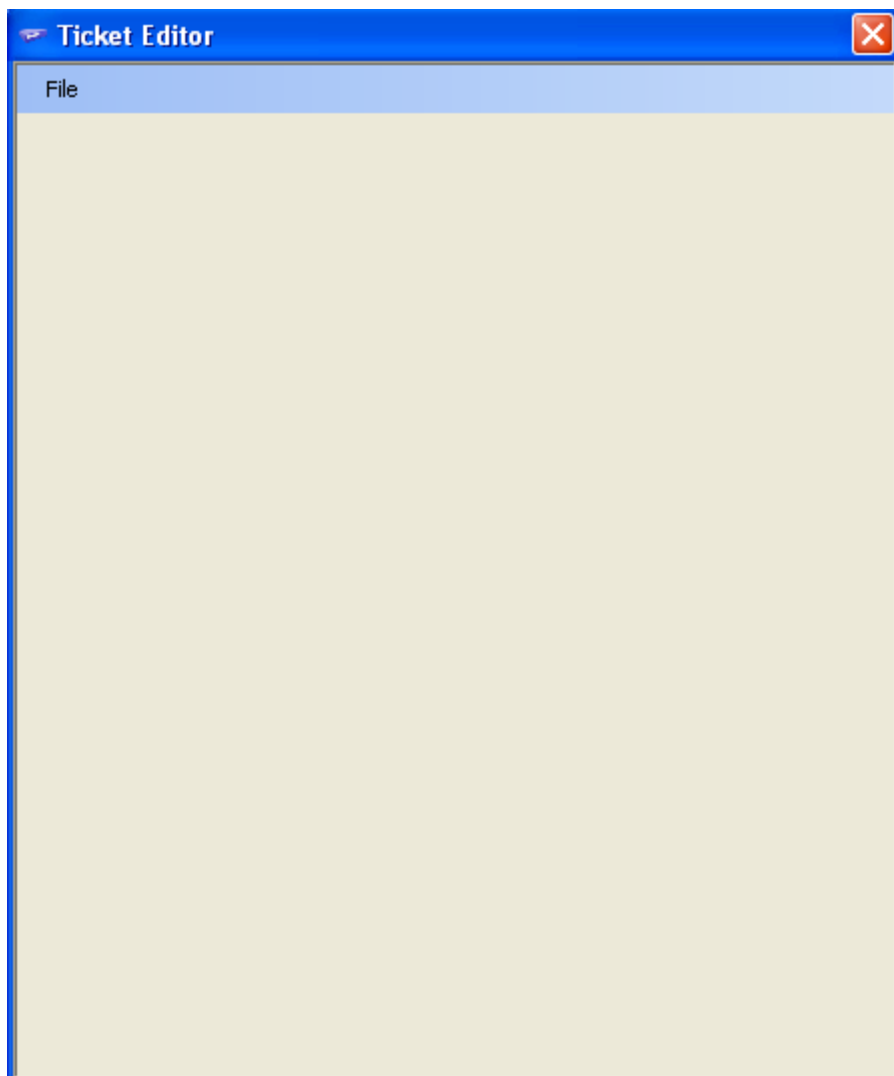
Ticket File

A ticket file contains several sets of related parameters used to create a PDF file. Each PDF creation method contained in the DLL or COM component of the SDK requires you to specify a ticket file in addition to the source file and the PDF file to be created. The SDK provides a default ticket file named ToPDF.tikd that resides in the TicketEditor folder of the SDK base installation folder. You may use the default ticket file without change or you may modify its parameters to fine tune your conversions. You may use the TicketEditor tool to edit a ticket file. The following section describes how you may use the TicketEditor to create the needed ticket file.

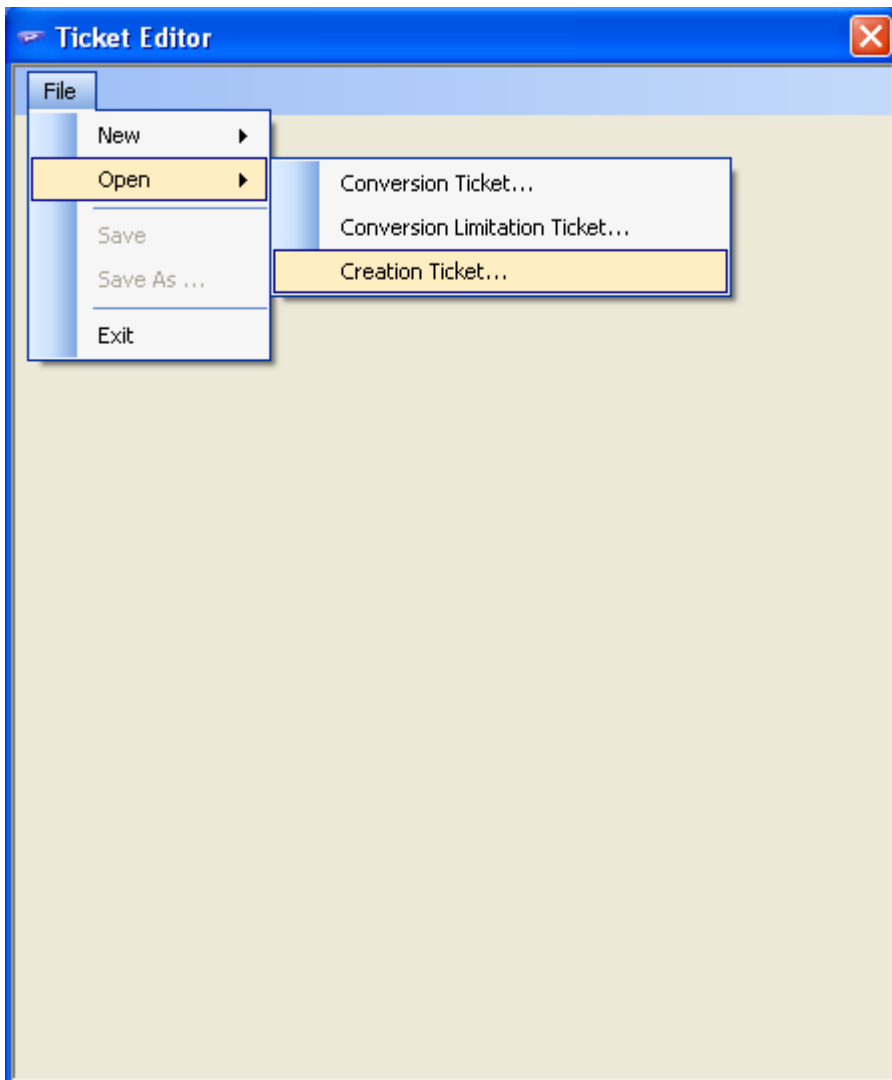
Ticket Editor

You may launch the Ticket Editor tool by navigating the path:
Start | All Programs | PDFCreateSDK | Ticket Editor

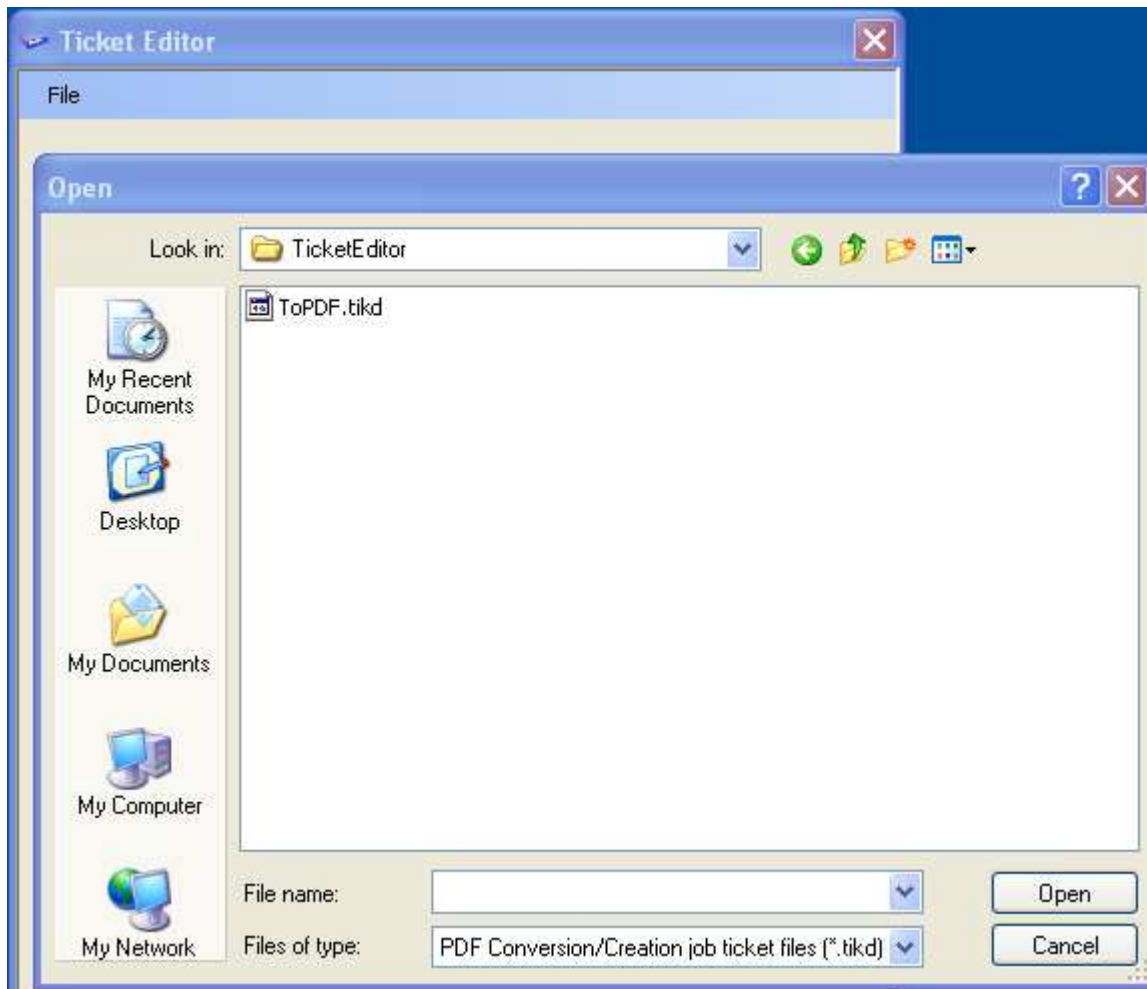
When the main windows appears:



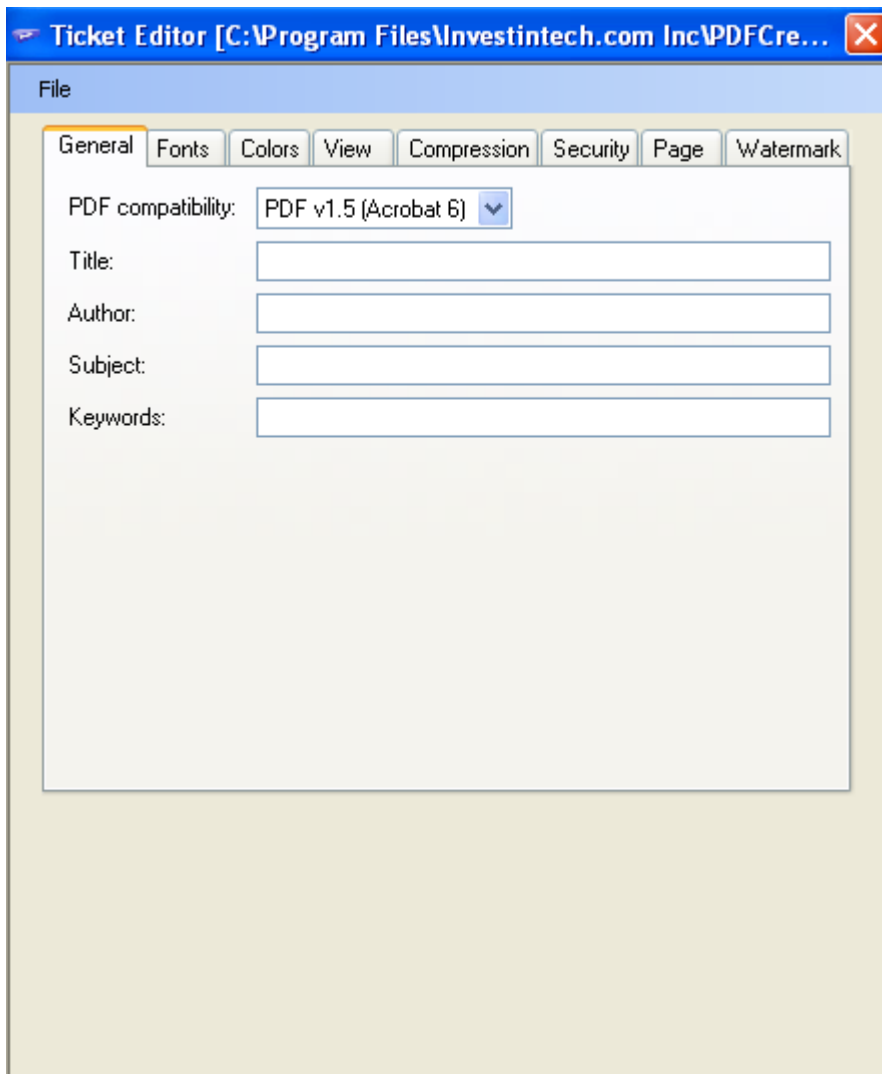
You may open a ticket file by navigating the tool's menu to:
File | Open | Creation Ticket...



Select the default ticket file, ToPDF.tikd that resides in the TicketEditor folder.



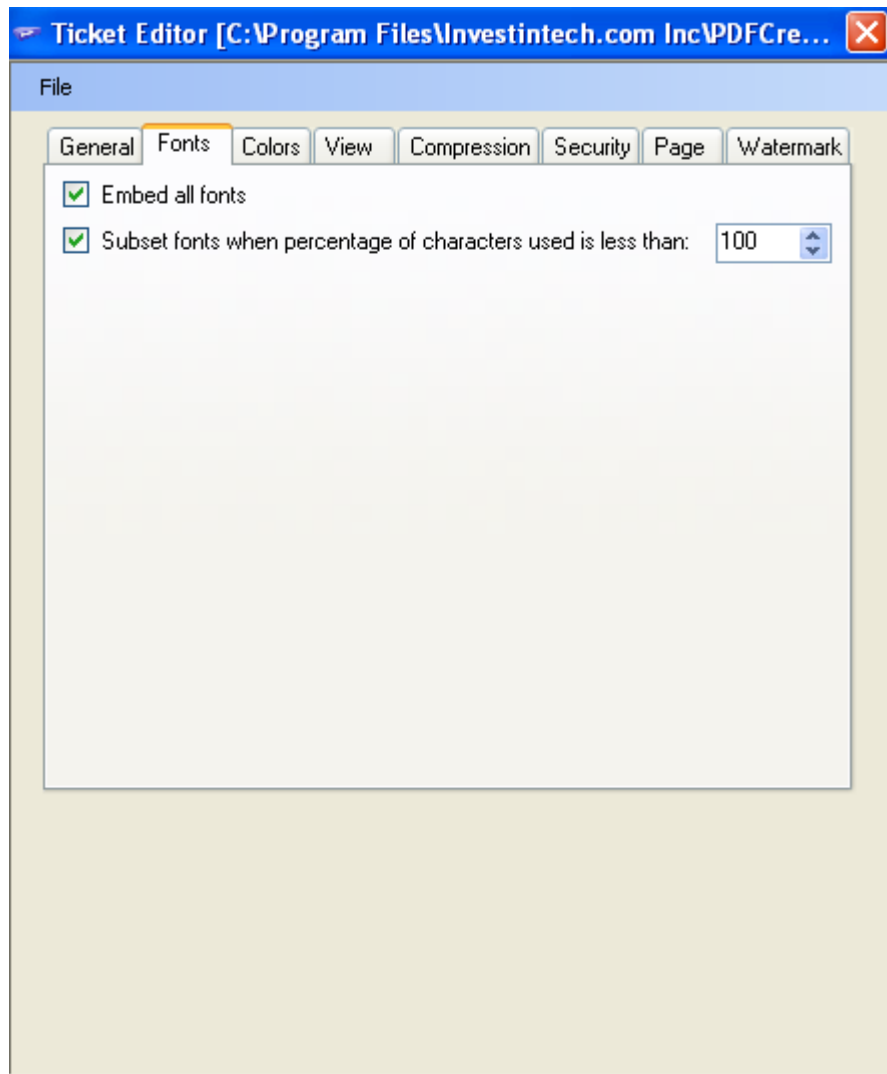
The first set of parameters is displayed on the General tab. It includes the target PDF version as well as PDF document meta data.



PDF compatibility – sets the compatibility of a created document to various versions of PDF standard. The older the standard is set, more compatible document will be.

Fields **Author, Subject, Keywords** – this fields are optional, here user can set a name of the author of the PDF document, its subject or keywords (might be useful if a document is intended for web publication, for example).

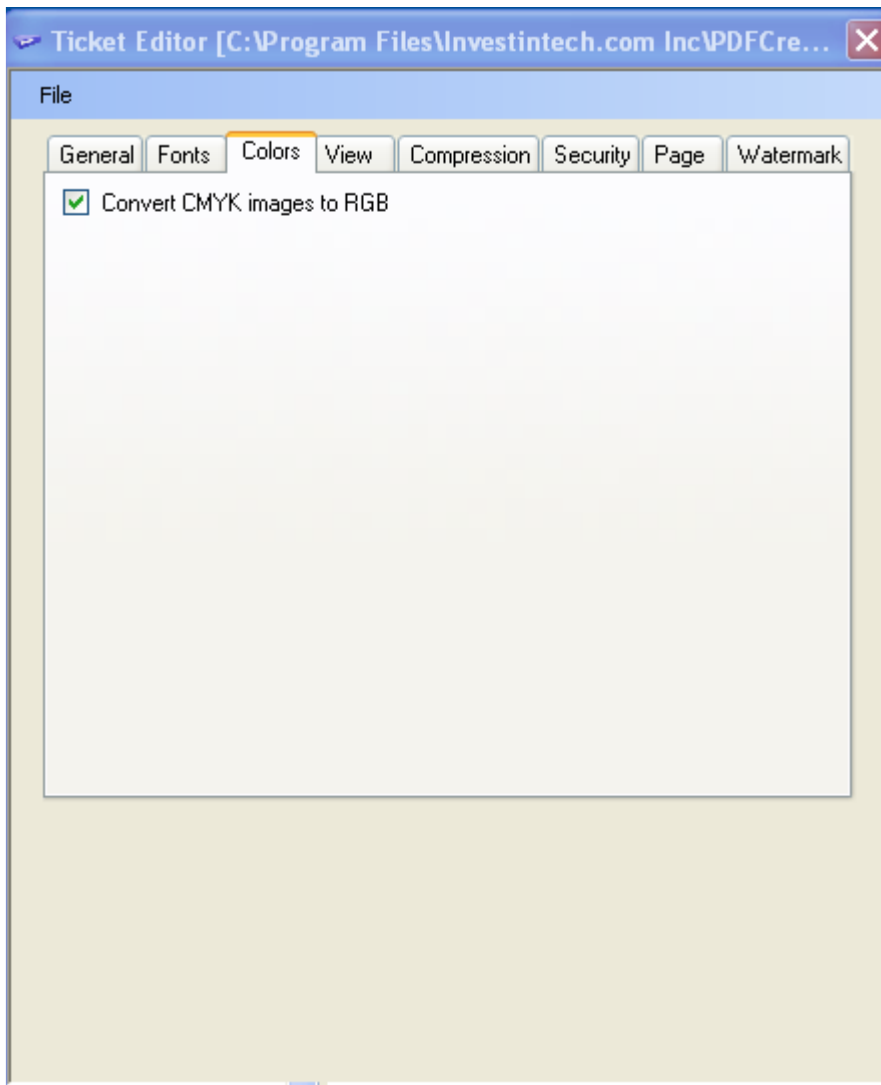
The second set of parameters is contained in the Fonts tab.



Embed all fonts – this options forces embedding of all fonts used in created document, as opposed to referencing definitions of commonly used fonts. Selecting this option increases portability of a document, but also its size.

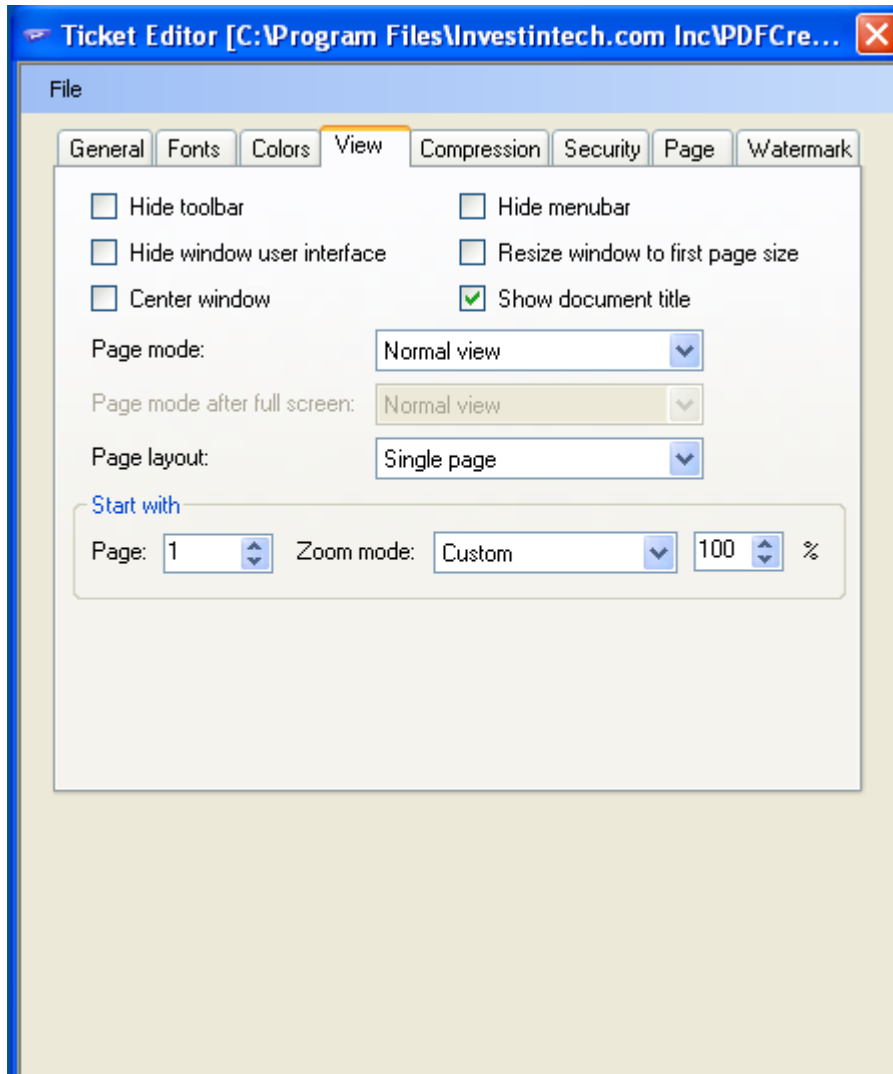
Subset fonts when percentage of character used is less than – this option is actually a sub option of “Embed all fonts” option. Characters defined in a font, but not used in this actual document should not be defined in an embedded font.

The third set of parameters is contained in the Colors tab.



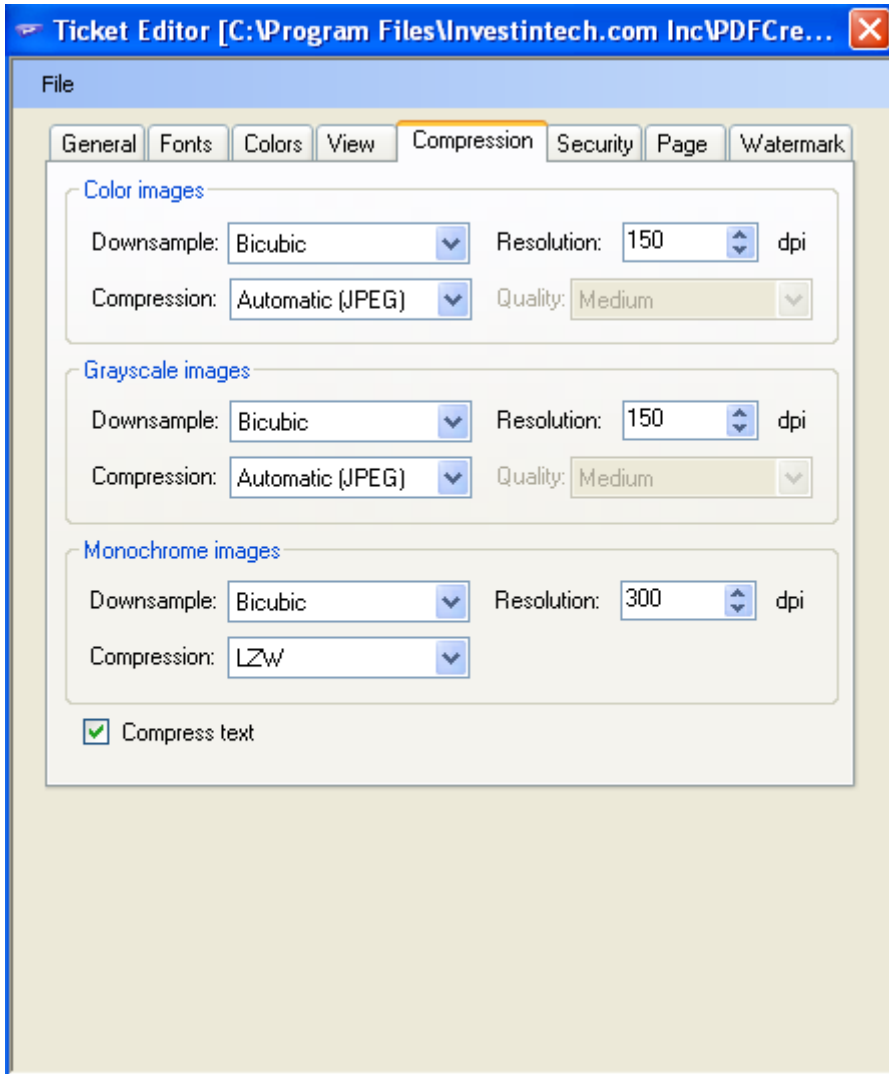
Convert CMYK images to RGB – this option is used to convert color palettes used in document’s images from CMYK palette to RGB palette.

The next set of parameters is contained in the View tab.



“View” tab of new PDF Creation JT dialog is used to set parameters for opening a created document, that are to be embedded in a created document. All of these parameters are only default viewing parameters (e.g. applied when generated PDF file is double-clicked), changeable in PDF viewer once a file has been opened.

The next set of parameters is contained in the Compression tab.



Color images compression

Subsample – 3 sampling methods, and also sub sampling off

Resolution – 10-2400 dpi,

Compression – various bitmap compression algorithms (image formats)

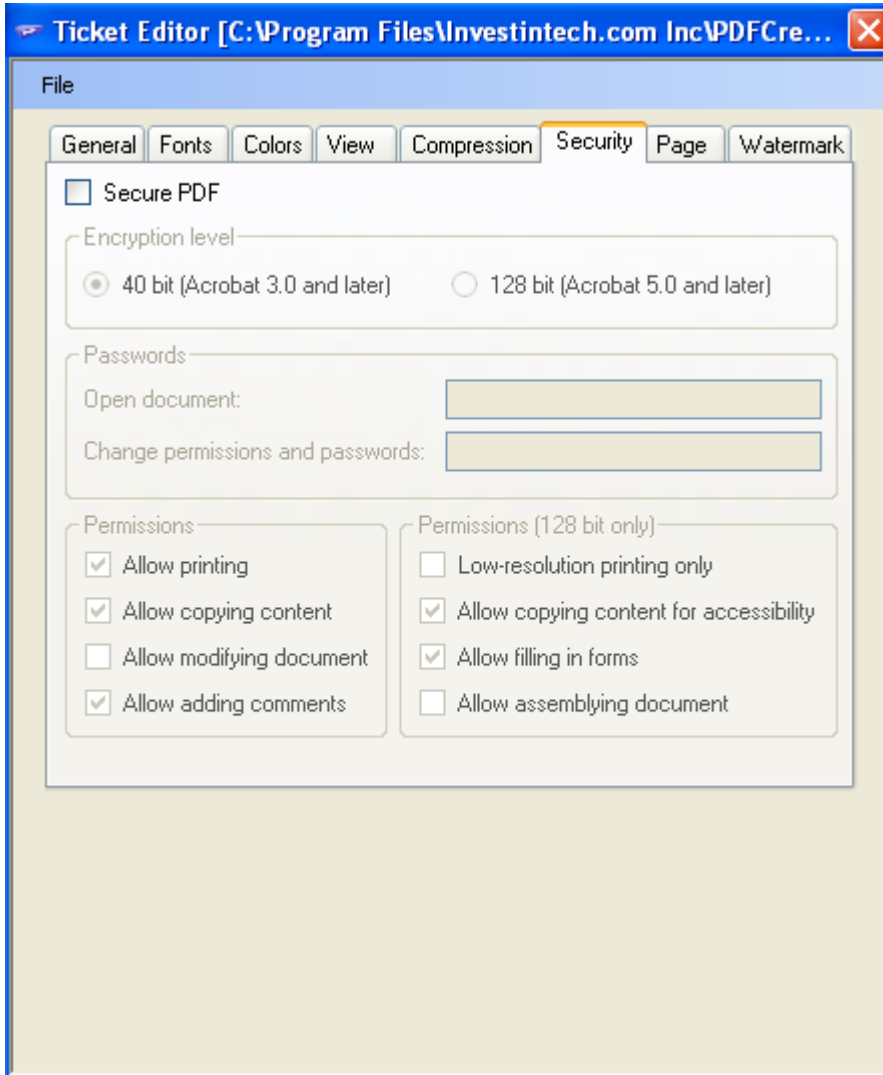
Quality – applicable only when compression is set to JPEG

Grayscale images compression - similar to color images compression

Monochrome images compression - similar to color images compression, but compression options are different and there's no quality option.

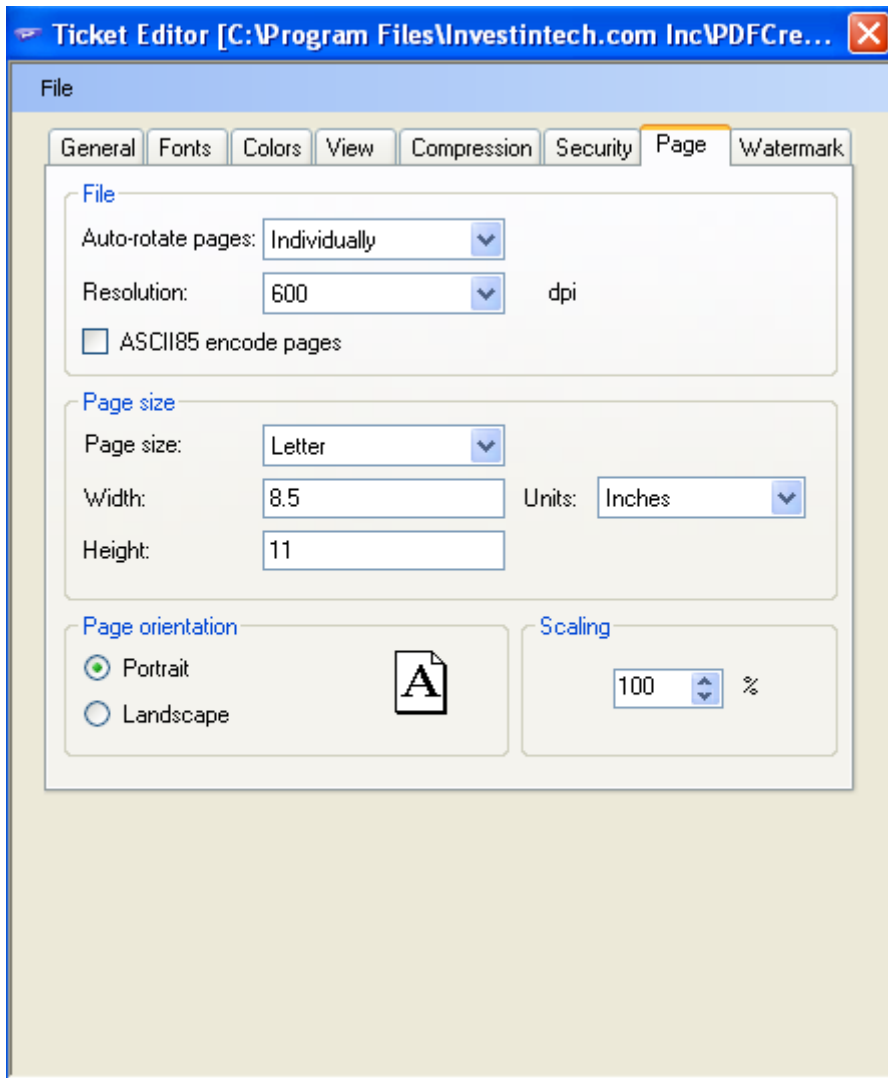
Text compression – sets text compression on or off

The next set of parameters is contained in the Security tab.



“Security” tab contains options for creating a secure PDF, with sub options for Encryption level, Passwords and Permissions.

The next set of parameters is contained in the Page level tab.



Page Rotation options: off (manually), individually (Page by page) and collectively by File (Whole file)

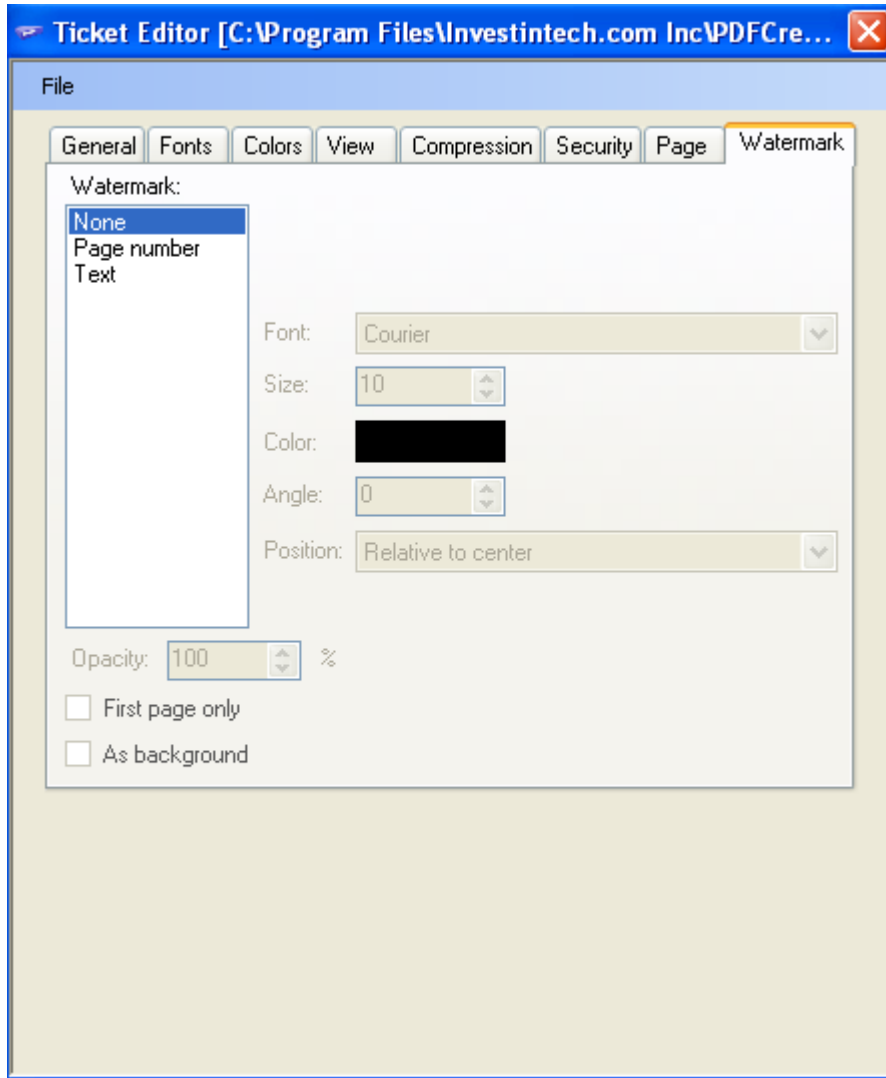
Output PDF resolution setting: 8 options ASCII 85 encoding selection – ASCII 85 is an encoding standard. Used in Postscript, ASCII85 is more space efficient than MIME base 64.

Page size selection: standard or custom, both can be showed and operated in 3 kinds of unit: Inches, MMs or Points

Page content scaling: from 1%-100%

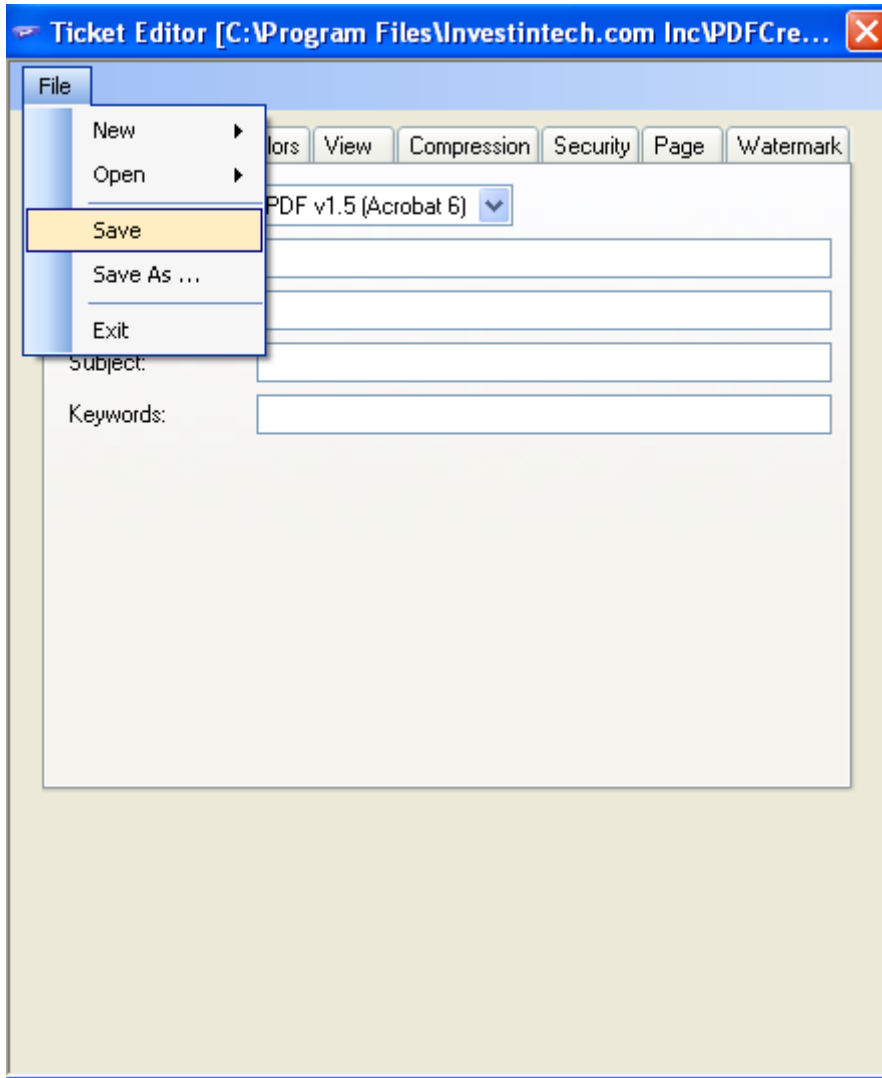
Page orientation: use can select only when Auto-Rotate pages sets to off

The next set of parameters is contained in the Watermark tab.



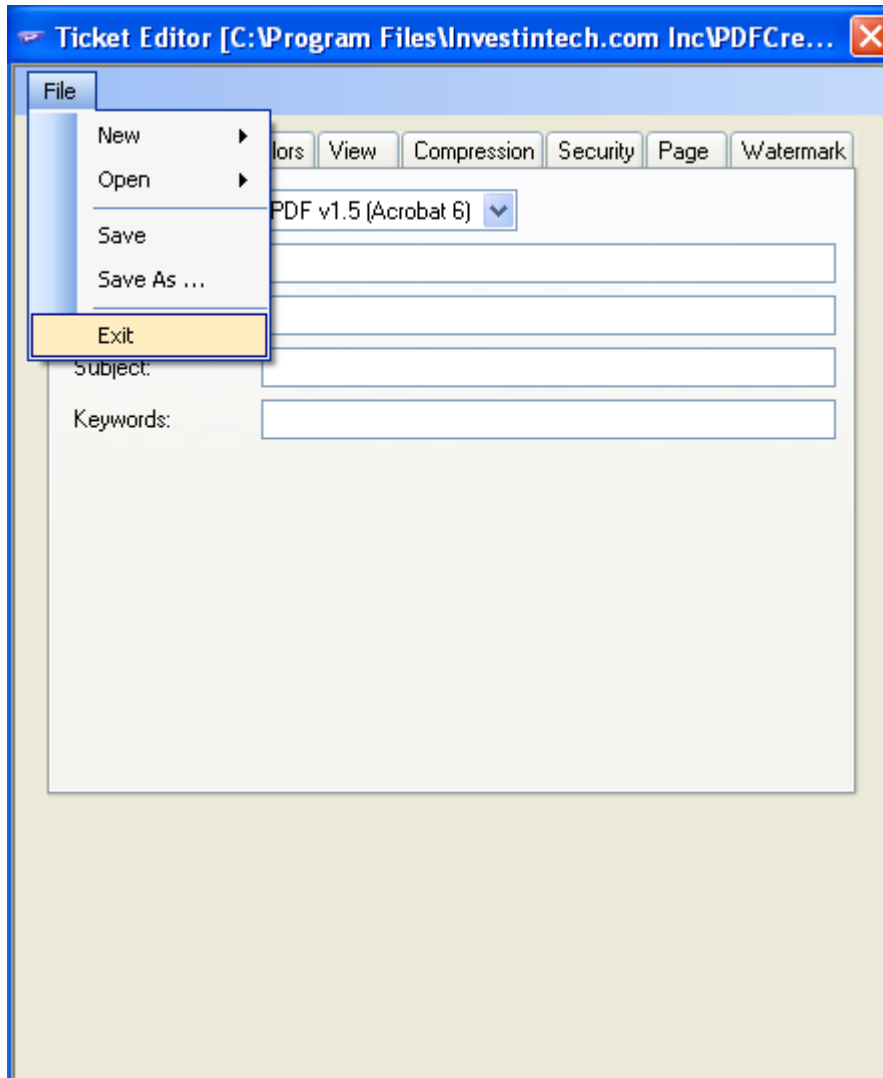
- watermark type: page number and text.
- Supports text type watermarks - page #, header, footer, ...
- Watermark font settings
- Opacity setting
- Select if only apply to first page
- Select if watermark is to be applied as background

To save your changes,



Investintech PDFCreate SDK

To exit the Ticket Editor,



Index

COM		PDF_Create_SLR	22
Methods	25	PDF_Create_SLT	23
Customer Service	2	PDF_Create_VB6	24
Investintech	2	String Parameter Type	15
Investintech.com Inc.	2	PDF Create SDK	4
Methods		Installation Instructions	4
Calling Conventions	15	System Requirements	4
Common Source Code	16	PDF Create SDK	
Error Handling	16	Implicit Linking	12
File Names	15	PDF_Create_SDK.dll	13
PDF_Create_CBR	17	PDF_Create_SDK.h	13
PDF_Create_CBT	17	PDF_Create_SDK.lib	13
PDF_Create_CLR	18	Technical Support	2
PDF_Create_CLT	19	Ticket Editor	27
PDF_Create_SBR	20	Ticket File	27
PDF_Create_SBT	21		