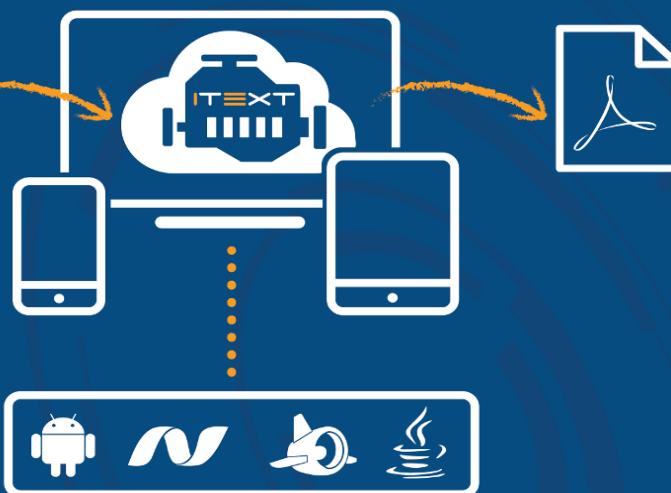




## THE WORLD'S MOST WIDELY USED PROGRAMMABLE PDF ENGINE

XML  
XHTML  
Web forms  
Databases



PDF/A  
PDF/UA  
Digital signatures  
PDF templates

# The Best iText Questions on stackoverflow

by Bruno Lowagie

# The Best iText Questions on StackOverflow

iText Software

This book is for sale at [http://leanpub.com/itext\\_so](http://leanpub.com/itext_so)

This version was published on 2015-06-23



This is a [Leanpub](#) book. Leanpub empowers authors and publishers with the Lean Publishing process. [Lean Publishing](#) is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

©2014 - 2015 iText Software

*This book is written by a developer for developers.*

*It is dedicated to all the developers who take pride in writing good code.*

# Contents

<b>introduction . . . . .</b>	<b>1</b>
Why StackOverflow? . . . . .	1
Acknowledgments . . . . .	2
How to use this book? . . . . .	3
<b>Questions about PDF in general . . . . .</b>	<b>4</b>
What is the difference between iText, JasperReports and Adobe LC? . . . . .	4
Does a PDF file have styles, headers and footers? . . . . .	5
How to extract a page number from a PDF file? . . . . .	5
How to disable the save button and hide the menu bar in Adobe Reader? . . . . .	7
How to close a PDF file to recreate it? (File in use problem) . . . . .	8
How to hide the Adobe floating toolbar when showing a PDF in browser? . . . . .	9
Why are PDF files different even if the content is the same? . . . . .	10
Why do PDFs change when processing them? . . . . .	12
How to find out if a PDF file compressed or not? . . . . .	14
How to protect a PDF with a username and password? . . . . .	15
How to encrypt PDF using a certificate? . . . . .	16
How to protect an already existing PDF with a password? . . . . .	17
How to allow page extraction when setting password security? . . . . .	18
How do I rotate a PDF page to an arbitrary angle? . . . . .	20
What is the size limit of pdf file? . . . . .	21
Can I change the page count by changing internal metadata? . . . . .	22
How to get the page number of an arbitrary PDF object? . . . . .	24
Where is the origin (x,y) of a PDF page? . . . . .	25
How should I interpret the coordinates of a rectangle in PDF? . . . . .	27
<b>Getting started . . . . .</b>	<b>29</b>
How to generate and design PDFs with iText or iTextSharp? . . . . .	29
Argument not specified in iTextSharp . . . . .	31
How to create a complex PDF document? . . . . .	32
How to align two paragraphs to the left and right on the same line? . . . . .	33
How to align a label versus its content? . . . . .	35
how to match the size of graphics with the size of a page? . . . . .	37
How to set the page size to Envelope size with Landscape orientation? . . . . .	39

## CONTENTS

How to use the full size of a page? . . . . .	40
How to create a document with unequal page sizes? . . . . .	41
How to change the line spacing of text? . . . . .	42
Does anyone have any idea how TabStop works? . . . . .	43
How to strike through text using iText? . . . . .	43
How to underline text with a dotted line? . . . . .	45
How to add a full line break? . . . . .	46
How to create a custom dashed line separator? . . . . .	47
How can I generate a PDF/UA compatible PDF with iText? . . . . .	49
<b>Fonts . . . . .</b>	<b>52</b>
How to use the font Verdana in PdfStamper? . . . . .	52
Why doesn't FontFactory.GetFont() work for all fonts? . . . . .	53
Why aren't my fonts getting registered? . . . . .	54
How to use Cyrillic characters in a PDF? . . . . .	56
Why is iText embedding a font even when I specify not to embed? . . . . .	59
How to identify a single font that can print all the characters in a string? . . . . .	60
How to change the font color and size when using FontSelector? . . . . .	61
How to create Persian content in PDF? . . . . .	62
How to solve an UnsupportedCharsetException when using itext-asian.jar? . . . . .	64
How to choose the optimal size for a font? . . . . .	65
How to restrict the number of characters on a single line? . . . . .	66
How to calculate the height of an element? . . . . .	70
How to calculate/set font line distance? . . . . .	71
Why can't I set the font of a Phrase? . . . . .	72
How to use two different colors in a single String? . . . . .	73
How to apply color to Strings in a Paragraph? . . . . .	74
How to introduce a custom font weight? . . . . .	76
How to make a single letter bold within a word? . . . . .	77
How to introduce superscript? . . . . .	81
How to allow a Unicode subscript symbol in a PDF without using setTextRise()? . . . . .	84
How to display indian rupee symbol? . . . . .	85
Why isn't the Rupee symbol showing? . . . . .	86
How to use System Font in iTextSharp with VB.net? . . . . .	88
Do I need a license for Windows fonts when using iText? . . . . .	93
How to get a font from an array? . . . . .	96
<b>Images . . . . .</b>	<b>97</b>
How to add multiple images into a single PDF? . . . . .	97
How to get the image DPI in PDF? . . . . .	98
Why aren't images added sequentially? . . . . .	99
How to add text to an image? . . . . .	100
How to draw lines on an image? . . . . .	103

## CONTENTS

How to preserve high resolution images in PDF? . . . . .	105
How to add JPEG images that are multi-stage filtered (DCTDecode and FlateDecode)? . . . . .	107
How to avoid an exception when importing a TIFF file? . . . . .	109
How to show an image with large dimensions across multiple pages? . . . . .	109
How to give an image rounded corners? . . . . .	111
How to convert colored images to black And white? . . . . .	112
How to make an image a qualified mask candidate in itext? . . . . .	115
How to create unique images with <code>Image.getInstance()</code> ? . . . . .	117
How to generate 2D barcode as vector image? . . . . .	120
How to change a background Image into a watermark by altering the opacity? . . . . .	123
<b>Absolute positioning of text . . . . .</b>	<b>125</b>
How to write a Zapfdingbats character at a specific location on a page? . . . . .	125
How to reduce redundant code when adding content at absolute positions? . . . . .	126
How to truncate text within a bounding box? . . . . .	128
How to continue an ordered list on a second page? . . . . .	128
Why does <code>ColumnText</code> ignore the horizontal alignment? . . . . .	131
How to fit a String inside a rectangle? . . . . .	132
Why does using <code>ColumnText</code> result in “The document has no pages” exception? . . . . .	134
How to divide a page in N parts so we can fill each with a different source? . . . . .	136
How to rotate a single line of text? . . . . .	140
How to rotate a paragraph? . . . . .	141
How to draw a rectangle around multiline text? . . . . .	143
What is causing syntax errors in a page created with iText? . . . . .	146
How to adapt the position of the first line in <code>ColumnText</code> ? . . . . .	148
<b>Absolute positioning of lines and shapes . . . . .</b>	<b>150</b>
How to add a border to a PDF page? . . . . .	150
How to create a PDF with a Cartesian grid? . . . . .	151
Why does the cell background color affect the color of other lines? . . . . .	152
How do I set parameters back to the default value? . . . . .	153
How to add a shading pattern to a custom shape? . . . . .	154
<b>Tables . . . . .</b>	<b>156</b>
How to right-align text in a <code>PdfPCell</code> ? . . . . .	156
How to use multiple fonts in a single cell? . . . . .	157
How to change width of single column of table? . . . . .	158
How to introduce a rowspan? . . . . .	159
How to maintain a paragraph’s indentation inside a table cell? . . . . .	160
What is the <code>PdfPTable.DefaultCell</code> property used for? . . . . .	162
How to draw a borderless table in iTextSharp? . . . . .	162
Why doesn’t <code>getDefautCell().setBorder(PdfPCell.NO_BORDER)</code> have any effect? . . . . .	163
How to define spacing and leading in <code>PdfPCells</code> ? . . . . .	165

## CONTENTS

How can I convert a CSV file to a table with a repeating header row? . . . . .	166
How to create a table based on a two-dimensional array? . . . . .	168
How to nest tables without extending the inner table? . . . . .	170
How to add an inner table surrounded by text? . . . . .	172
How to resize an Image to fit it into a PdfPCell? . . . . .	174
How to display barcodes in a matrix-like structure? . . . . .	176
How to split a row over multiple pages? . . . . .	177
How does a PdfPCell's height relate to the font size? . . . . .	177
How to add a table to the bottom of the last page? . . . . .	178
How do setMinimumSize() and setFixedSize() interact? . . . . .	179
How to get the rendered dimensions of text? . . . . .	180
How to tell iText how to clip text to fit in a cell? . . . . .	182
How to resize a PdfPTable to fit the page? . . . . .	183
What's an easy to print "first right, then down"? . . . . .	185
How to invoke a page break for nested tables? . . . . .	187
How to make a footer stick to bottom of every pdf page? . . . . .	189
How to add continue on next page / continued from next page to a table? . . . . .	190
<b>Table events . . . . .</b>	<b>194</b>
How to use a dotted line as a cell border? . . . . .	194
How to create a table with rounded corners? . . . . .	195
How to introduce rounded cells with a background color? . . . . .	196
How to set background image in PdfPCell in iText? . . . . .	198
How to get text and image in the same cell? . . . . .	199
Is it possible to attach multiple layout events to a PdfPCell? . . . . .	200
How to precisely position an image on top of a PdfPTable? . . . . .	202
Why is my cell event not triggered? . . . . .	204
<b>Page events . . . . .</b>	<b>206</b>
How to add text as a header or footer? . . . . .	206
How to add a rectangle to every page of a document? . . . . .	208
How can I add an image to all pages of my PDF? . . . . .	209
How to set a fixed background image for all my pages? . . . . .	210
Why do I get a System.StackOverflowException in the OnEndPage() event handler? . . . . .	212
How to introduce multiple PdfPageEventHelper instances? . . . . .	213
How to check for an event and remove it? . . . . .	214
How to add a text to the left and to the right in a header? . . . . .	214
How to add a table as a header? . . . . .	216
How to create a table with 2 rows that can be used as a footer? . . . . .	217
How to generate a report with dynamic header in PDF using itextsharp? . . . . .	218
How to add a page number in the header of a PDF/A Level A file? . . . . .	220
How to rotate a page while creating a PDF document? . . . . .	221
How to draw a line every 25 words? . . . . .	223

## CONTENTS

How to add a border to a paragraph? . . . . .	226
How to change the color of pages? . . . . .	230
<b>Parsing XML and XHTML . . . . .</b>	<b>232</b>
Why is it so difficult to convert XML to PDF? . . . . .	232
How to adjust the page height to the content height? . . . . .	233
How to set line spacing when using XML Worker? . . . . .	236
How to add external CSS while generating PDF? . . . . .	237
How to do HTML to XML conversion to generate closed tags? . . . . .	238
How to make a particular sub-string Bold when converting HTML to PDF? . . . . .	239
How to parse multiple HTML files into a single PDF . . . . .	240
How to add a rich Textbox (HTML) to a table cell? . . . . .	243
How to get particular html table contents to write it in PDF? . . . . .	245
Why doesn't CSS and RowSpan work? . . . . .	248
Why is XMLWorker parsing slow? . . . . .	250
How to export Vietnamese text to PDF using iText? . . . . .	252
Certain HTML entities (arrows) are not rendered in PDF . . . . .	254
How to set RTL direction for Hebrew when converting HTML to PDF? . . . . .	256
How to convert Arabic HTML to PDF? . . . . .	259
<b>Inspect a PDF with iText . . . . .</b>	<b>265</b>
Why do I get an "InvalidPdfException: PDF header signature not found"? . . . . .	265
How to Get PDF page width and height? . . . . .	266
Why is the page size of a PDF always the same, no matter if it's landscape or portrait? . . . . .	267
How to get the UserUnit from a PDF file? . . . . .	269
How can I read Roman page numbers? . . . . .	270
How do I get an object if I have its indirect reference? . . . . .	272
How to read PDFs created with an unknown random owner password? . . . . .	273
How to get values from comment annotations? . . . . .	274
How can I find the border color of a field? . . . . .	275
How to read bookmark titles? . . . . .	277
How to read bookmarks? . . . . .	278
How To find internal links in a PDF file? . . . . .	281
How to extract embedded streams? . . . . .	282
<b>Manipulating existing PDFs . . . . .</b>	<b>285</b>
How to update a PDF without creating a new PDF? . . . . .	285
How to add an image watermark to a PDF file? . . . . .	286
How to add a watermark to a page with an opaque image? . . . . .	287
How to watermark PDFs using text or images? . . . . .	289
How to extend the page size of a PDF to add a watermark? . . . . .	294
Why does PdfStamper create a file with 0 bytes? . . . . .	295
How to add blank pages to an existing PDF in java? . . . . .	296

## CONTENTS

Adding blank pages while concatenating several PDFs in iText using PdfSmartCopy . . . . .	297
How to position text relative to page? . . . . .	298
How can I crop the pages of an existing PDF document? . . . . .	299
How to rotate a page 90 degrees? . . . . .	299
How to crop out a part of PDF file? . . . . .	301
How to rotate and scale pages in an existing PDF? . . . . .	303
How to shrink pages in an existing PDF? . . . . .	304
How to fix the orientation of a PDF page in order to scale it? . . . . .	308
How to reuse a page from one PDF document into another PDF document? . . . . .	311
How to merge documents correctly? . . . . .	312
How to add a cover page to an existing PDF document? . . . . .	315
Why does the function to concatenate / merge PDFs cause issues in some cases? . . . . .	317
How to merge PDFs from <code>ByteArrayOutputStreams</code> ? . . . . .	318
How to merge PDFs and add bookmarks? . . . . .	319
How to create a TOC when merging documents? . . . . .	321
How to reorder pages in an existing PDF file? . . . . .	323
How to reorder the pages of a PDF file? . . . . .	325
How to set the OCG state of an existing PDF? . . . . .	327
How to change the order of Optional Content Groups? . . . . .	328
How to create a PDF with font information and embed the actual font while merging the files into a single PDF? . . . . .	330
How to decrypt a PDF document with the owner password? . . . . .	333
How do I add XMP metadata to each page of an existing PDF? . . . . .	337
How to move the content inside a rectangle inside an existing PDF? . . . . .	339
How to load a PDF from a stream and add a file attachment? . . . . .	342
<b>Interactive forms . . . . .</b>	<b>345</b>
How to fill out a pdf file programmatically? (AcroForm technology) . . . . .	345
How to fill out a pdf file programmatically? (Dynamic XFA) . . . . .	346
Is it safe to remove XFA? . . . . .	347
How to flatten a XFA PDF Form using iTextSharp? . . . . .	348
How to tell iText which fields to flatten first? . . . . .	350
How to merge forms from different files into one PDF? . . . . .	353
How to save an <code>.xfdf</code> file as a <code>.pdf</code> file? . . . . .	357
How to edit a PDF embedded in the browser and save the PDF directly to server? . . . . .	362
How to send a ‘success’ response back to Acrobat Reader from a java servlet? . . . . .	365
Why do I get an error saying that “use of extended features is no longer available”? . . . . .	366
How to fill XFA form using iText without breaking usage rights? . . . . .	368
How to deal with missing characters in the default font of a text field? . . . . .	369
How to make sure a check box is printed? . . . . .	370
How to check a check box? . . . . .	371
How to distribute the radio buttons of a radio field across multiple <code>PdfPCells</code> ? . . . . .	372
How to find out which fields are required? . . . . .	374

## CONTENTS

How to make a field <i>not</i> required? . . . . .	375
How to get the number of characters in a field? . . . . .	376
How to align AcroFields? . . . . .	377
How to change the text color of an AcroForm field? . . . . .	378
How to get the color properties of an AcroForm field? . . . . .	381
How to find the absolute position and dimension of a field? . . . . .	383
How to move an AcroForm field? . . . . .	384
How to continue field output on a second page? . . . . .	385
How to add a table on a form (and maybe insert a new page)? . . . . .	387
How to add an image to an AcroForm field? . . . . .	389
How to format a field as a percentage? . . . . .	390
How to add a hidden text field? . . . . .	391
How to send a file to the server through a PDF? . . . . .	392
How to add a text field to an existing pdf template . . . . .	392
How can I add a new AcroForm field to a PDF? . . . . .	393
How to underline a portion of text in a text field? . . . . .	394
Why are the AcroFields in my document empty? . . . . .	395
How to create a pushbuttonfield with double byte character text? . . . . .	397
How to define the data encoding when submitting a PDF form using AcroForm technology? . . . . .	399
<b>Actions and annotations . . . . .</b>	<b>401</b>
How to create hierarchical bookmarks? . . . . .	401
How can I add titles of chapters in ColumnText? . . . . .	404
How to create a link to a specific page number? . . . . .	409
How to insert a “linked rectangle” with iText? . . . . .	410
How to add a maps with a pointer to a PDF? . . . . .	411
How to create a clickable polygon or path? . . . . .	414
How do I insert a hyperlink to another page with iTextSharp in an existing PDF? . . . . .	415
How to add a printable or non-printable bitmap stamp to a PDF? . . . . .	416
How to create a pop-up a window to display images and text? . . . . .	418
How to stamp image on existing PDF and create an anchor? . . . . .	421
How to set the BaseUrl of an existing PDF document? . . . . .	423
How to set zoom level to pdf using iTextSharp? . . . . .	425
How to change the properties of an annotation? . . . . .	426
How to create a link to launch an external program? . . . . .	427
How to attach files to a PDF? . . . . .	428
How to delete attachments in PDF using iText? . . . . .	429
How to create a JavaScript action to open the attachments panel? . . . . .	433
How to add an onMouseOver javaScript action to a TextField? . . . . .	434
How to define multiple actions for a PushbuttonField? . . . . .	435
How to add an “In Reply To” annotation? . . . . .	436
How to get the author of a “free text” annotation? . . . . .	438
How to change the author name for comments? . . . . .	440

## CONTENTS

How to change the color of a circle annotation? . . . . .	441
<b>Extracting text from PDFs . . . . .</b>	<b>443</b>
How to remove text from a PDF? . . . . .	443
How to create and apply redactions? . . . . .	445
What are the extra characters in the font name of my PDF? . . . . .	448
How to extract text and anchor information from a PDF? . . . . .	449
How to read text from a specific position? . . . . .	451
How to use a text extraction strategy after applying a location extraction strategy? . . . . .	452
Why can't I extract text added using a Type3 font correctly from a PDF? . . . . .	454
How to get the co-ordinates of an image? . . . . .	455
How to check if a font is bold? . . . . .	456
Why is the text I extract from an English PDF page garbled? . . . . .	459
<b>General questions about iText . . . . .</b>	<b>461</b>
Unit Testing and Automated Testing Questions . . . . .	461
How to set initial view properties? . . . . .	462
Why do I get a "Could not find PdfGraphics2D" error? . . . . .	465
Why can't I compile the iText source code? . . . . .	466
Why do I get a BouncyCastle NoClassDefFoundError? . . . . .	467
How to change the spacing between words and characters? . . . . .	468
How to add / delete / retrieve information from a PDF using a custom property? . . . . .	470
How to create an uncompressed PDF file? . . . . .	473
How to detect if PDF has compressed xref table in Java? . . . . .	474
How to increase the accuracy of measurements in iTextSharp? . . . . .	475
How to prevent the resizing of pages in PDF? . . . . .	475
How to hyphenate text? . . . . .	476
How to get the current page count? . . . . .	477
When is the content flushed to a PDF File by itextsharp? . . . . .	478
How to convert PdfStamper to a byte array? . . . . .	479
How to create a new PDF file if a file name already exists? . . . . .	480
How can I server a PDF to a browser without storing a file on the server side? . . . . .	481
Why do I get a "getOutputStream() has already been called for this response" error in JSP? . . . . .	484
<b>Legal questions . . . . .</b>	<b>486</b>
What is the difference between Lowagie and iText? . . . . .	486
Can iText 2.1.7 or earlier be used commercially? . . . . .	487
Can I use iText without respecting the AGPL license? . . . . .	489
Is iText Java library free of charge or are there any fees to be paid? . . . . .	491
Can I bundle iText with my non-commercial software? . . . . .	492
When does Bruno Lowagie act like "kind of a dick"? . . . . .	495
<b>To be continued... . . . . .</b>	<b>501</b>

# introduction

A couple of years ago, I decided to self-publish new books about iText, as opposed to working with a publisher as I did before for the “iText in Action” books. This led to a book about digital signatures that is available for download<sup>1</sup> on the iText site, and a book called “The ABC of PDF”<sup>2</sup> published on LeanPub. The goal of “The ABC of PDF” was to start with a book that looks at PDF at the lowest level, examining the syntax of a PDF file and a PDF page, and then to continue writing a series of books that explain how to use iText on a higher level, answering questions such as:

- How to create a PDF from scratch?
- How to create PDF from HTML?
- How to fill out PDF forms?
- How to parse a PDF file?
- ...

However, in spite of the fact that more than 15,000 people downloaded “The ABC of PDF”, it turned out that people really wanted me to write a different kind of book. I’ve received many comments through LeanPub from people who were disappointed that the ABC-book didn’t explain *how to use iText*. They expected a book with more practical examples, instead of examples that helps them understand the PDF specification. Some people even used the feedback form to ask me technical questions. Unfortunately, I was unable to answer these questions, because the people posting them didn’t realize that I received these questions anonymously. Even if I knew the answers, I didn’t know who or where to send them to.

All of this faced me with a dilemma: do I stop writing “The ABC of PDF” and start writing one of the other books that were planned? If so, which part of iText is most important to iText users? The plan for the ABC was to write a book of about 150 pages, but much to my surprise, I was only half way when I finished writing page 150. Didn’t I have other writing priorities?

Then suddenly I had an idea: why not write a book with questions and answers? Why not create a book entitled “The Best iText Questions on StackOverflow?”

## Why StackOverflow?

I joined StackOverflow on August 24, 2012. Up until then, I had been answering many questions on the iText mailing-list. This mailing-list hosted on SourceForge used to be an important source

---

<sup>1</sup><http://itextpdf.com/book/digitalsignatures>

<sup>2</sup>[https://leanpub.com/itext\\_pdfabc/](https://leanpub.com/itext_pdfabc/)

of inspiration. I composed two “iText in Action” books for Manning Publications, simply by reorganizing the many answers and examples written in answer to question into a real book.

However, at some point I got tired of the mailing-list. When I referred to an example in one of my books, people would accuse me for trying to “trick them into buying my book.” The mailing-list was also used by people spreading false allegations, such as “iText is no longer open source.” One could explain that these people were wrong, for instance by [providing a link to the source code<sup>3</sup>](#), but there was no way to award people for providing good answers and to discourage people from posting bad answers. It felt as if the ungrateful were winning the debate.

Then I discovered [StackOverflow<sup>4</sup>](#) where people build a reputation getting reputation points when they ask good questions and provide good answers, losing points when they post bad questions or bad answers. I took me 2 years and almost 2 months to become a *Trusted User*, a status that requires 20,000 reputation points. Since I registered on StackOverflow, I have posted answers to more than 1,000 questions. Looking back at some of the more elaborate answers, I thought it would be a good idea to bundle those questions and answers that are of “book quality”.

## Acknowledgments

I have selected nothing but questions I have answered [myself<sup>5</sup>](#), but it goes without saying that I can’t answer every single question about iText personally. For instance: when I am travelling, I am off-line for many hours. As unanswered questions about iText give me stress, I am always happy to see that other people jump in when I’m away from my keyboard.

I want to thank Alexis Pigeon for editing many iText questions in order to clarify what is asked. I rely on Chris Haas for answering questions that require the C# skills that I am missing. I notice that I skip questions about digital signatures, because I know that Michael Klink’s answer will be much more accurate than mine.

I also want to thank the many people who accepted one of my answers, because that’s how one builds a reputation on StackOverflow. I know that some people down-vote me because my style can be harsh at times. Somebody once tweeted: “*Spent a lot of time today on StackOverflow and realized that Bruno Lowagie is kind of a dick.*” Ah well, I hope that the balance is positive.

Please understand that it is hard for me when people talk about “Lowagie” as if it’s a thing, not a person. Sometimes people start by saying that they are *using “Lowagie software”* and then they start cursing at me if I give them an answer they don’t like, for instance: please use a more recent version instead of a version that has been declared “End of Life” more than five years ago. So it goes... Not every developer realizes that I’m on their side and that their job is much easier if only their boss would purchase a commercial iText license so that they can use the most recent version.

---

<sup>3</sup><https://github.com/itext/itextpdf>

<sup>4</sup><http://stackoverflow.com>

<sup>5</sup><http://stackoverflow.com/users/1622493/bruno-lowagie>

## How to use this book?

I've tried organizing the questions and answers in different categories. This wasn't always simple. If somebody asks a question about adding an image to a table, should this question be categorized under "images" or under "tables"? If there's a question about XHTML content that needs to be added to a column, is that an "XML" or a "ColumnText" question? A book isn't a web site where you can easily introduce a taxonomy. That's why I took great care when creating the table of contents.

In many cases, I rephrased the original question so that you understand what a question is about at a glance, just by browsing the bookmarks. In some cases, I even had to rewrite the question.

All the questions are marked with a question mark icon like this:



this is a question

At times, I throw in a question of myself to clarify things. These questions are marked with an information icon:



This is an extra question added by myself

Sometimes, it was important to add a comment that was made on StackOverflow. I have marked comments like this:



This is a comment

I hope you enjoy this book, and that it helps you solving all your iText problems. If not, please post a question on [StackOverflow<sup>6</sup>](http://stackoverflow.com) and, who knows, maybe your question will be added to this book.

---

<sup>6</sup><http://stackoverflow.com>

# Questions about PDF in general

When posting a question on StackOverflow, people can tag their posts as iText or iTextSharp questions. This allows me to quickly find those questions by performing a simple query for [posts tagged as itext\\* questions<sup>7</sup>](#). This includes the tags `itext`, `itextsharp`, `itextpdf` and `itextg`.

However, not all questions tagged this way are iText-related. Sometimes, people using iText have questions that are about PDF in general.

## What is the difference between iText, JasperReports and Adobe LC?



Actually I want to know the difference or comparison between different PDF creation / generation techniques. For Example: iText, Adobe LC, Jasper Reports, etc.

I would like to know the exact advantage / disadvantage of using each of them.

Currently I am using Adobe LC ES2 and would like to also know the advantage of using Adobe software over other techniques.

Posted on StackOverflow on [Mar 19, 2013<sup>8</sup>](#) by [user2185226<sup>9</sup>](#)

That's a very broad question and I see that it already has a vote to close the question for this reason. Let me give the nutshell version of the answer. I could easily write a book on this topic (and maybe one day I will).

- iText is a library that can be used by developers to enhance their web and other applications with PDF functionality: create PDF, fill out PDF forms, examine and manipulate existing PDFs.
- JasperReports is a Business Intelligence / Reporting tool that uses an old iText version to create reports. It is distributed by JasperSoft / TIBCO. JasperReports only uses a limited part of the complete iText functionality. Creating PDF is just one of many features of JasperReports, and JasperSoft uses iText to implement that feature.

---

<sup>7</sup>[http://stackoverflow.com/questions/tagged/itext\\*](http://stackoverflow.com/questions/tagged/itext*)

<sup>8</sup><http://stackoverflow.com/questions/15492738/difference-between-itext-and-adobe-lc>

<sup>9</sup><http://stackoverflow.com/users/2185226/user2185226>

- Adobe LC is a suite of modules, some of which can only be provided by Adobe. For instance: no third party can “Reader enable” PDF documents because Reader enabling requires a private key that is proprietary to Adobe. However: iText competes with Adobe LC in some areas, for instance digital signing (read [the white paper from the Office of Legislative Counsel on digital signatures<sup>10</sup>](#)) and form filling (iText has an add-on called [XFA Worker<sup>11</sup>](#) that can convert your dynamic XFA forms into static PDF, e.g. PDF/A)

## Does a PDF file have styles, headers and footers?



Does a PDF file have styles, headers and footers information as is the case with docx files that have separate xml files with extra information?

Posted on StackOverflow on [Jan 21, 2014<sup>12</sup>](#) by [Prakhar<sup>13</sup>](#)

Regular PDFs don’t have styles, but different fonts (for instance Helvetica is one font, Helvetica-Bold is another font of the same family). They don’t have headers and footers, just like they don’t have paragraphs, section titles, table rows or table cells. Everything you see in a PDF page, is just a bunch of glyphs, paths and shapes drawn on a canvas.

However: if your PDF is a **Tagged PDF**, the PDF contains something that is known as the **StructTreeRoot**. This means that, apart from the presentation of the content, you also have a tree structure that stores the semantics of the content. This structure contains references to the content on the different pages, allowing you (for instance) to find out which lines belong together in a paragraph, which parts of the page are “artifacts” (such as a repeating page header or a footer with a page number), which content is organized as a table, etc...

Tagged PDF is a requirement for PDF/A Level A and PDF/UA documents. A majority of the PDF files you can find in the wild aren’t tagged (or aren’t tagged properly).

## How to extract a page number from a PDF file?



We explored many API’s like Tika, PdfBox and iText to extract page numbers from a PDF file, but we weren’t able to meet this requirement. In iText we tried `PdfPageLabels.getPageLabels(reader)` but the behavior of this method is not uniform.

Posted on StackOverflow on [Oct 31, 2014<sup>14</sup>](#) by [abhinav sharma<sup>15</sup>](#)

---

<sup>10</sup>[http://www.mnhs.org/preserve/records/legislative\\_records/docs\\_pdfs/CA\\_Authentication\\_WhitePaper\\_Dec2011.pdf](http://www.mnhs.org/preserve/records/legislative_records/docs_pdfs/CA_Authentication_WhitePaper_Dec2011.pdf)

<sup>11</sup>[http://itextpdf.com/product/xfa\\_worker](http://itextpdf.com/product/xfa_worker)

<sup>12</sup><http://stackoverflow.com/questions/21259333/does-pdf-has-styles-headers-and-foooters-information-as-docx>

<sup>13</sup><http://stackoverflow.com/users/1881995/prakhar>

<sup>14</sup><http://stackoverflow.com/questions/26673581/how-to-extract-page-number-from-pdf-file>

<sup>15</sup><http://stackoverflow.com/users/4202194/abhinav-sharma>

The reason why you don't find any software that is able to extract page numbers from a PDF is simple: the concept of a page number doesn't exist in PDF.

Allow me to predict your response.

*"Wait a minute!"* you say, *"When I open a PDF in Adobe Reader, I can clearly see a page number in the document!"*

Yes, you can see that page number with your eyes and your **human** intelligence, but to a **machine** that number is just some text drawn on a canvas. A machine consuming the document has no idea what all the glyphs and lines and shapes on a page are about. Hence, software can not give you the page number you see as a human. A machine *doesn't know where to look!*

If you know something about PDF, I can predict your next reply.

*"Wait a minute!"* you say, *"What about Tagged PDF? Doesn't Tagged PDF mean that the semantics of a document are stored along with the representation?"*

Yes, when a PDF is tagged a snippet of text knows that it is part of a title, or a paragraph, or a list,... But Tagged PDF is there to define the structure of the *real content*. Page numbers however, are not part of the real content. They are marked as *artifacts* along with headers, footers and other items on a page that are not considered being *real content*. There is no way to distinguish page numbers.

*"Then what are these page labels about?"* you ask.

Well, page labels are *optional*. They are present in some PDFs that are well conceived, but they will be absent in a large majority of the PDFs you'll find in the wild.

This is the long answer. The short answer is simple: **You are asking for something that is impossible** (in general, not only with iText, Tika, PdfBox, or any other tool you might try).

## How to disable the save button and hide the menu bar in Adobe Reader?



I am serving a PDF to a browser via a Servlet. I need to disable the “save” and “print” option in the Adobe Reader menu bar while other options like “scroll”, “find”,... should be preserved. In addition to this, I need to disable the file menu of the browser window in which it is rendered.

I have disabled print and file menu using below code

```
stamper.setEncryption(  
    null, null, PdfWriter.HideWindowUI, PdfWriter.STRENGTH40BITS);  
stamper.setViewerPreferences(PdfWriter.HideToolbar);
```

My problem boils down to this: how do I disable the “save” button in the Adobe Reader menu bar?

Posted on StackOverflow on Apr 5, 2014 <sup>16</sup> by seven-7<sup>17</sup>

We need to distinguish two different aspects: printing and saving.

You can encrypt a file and set the permissions in such a way that printing isn’t allowed. However: if you only encrypt a document with an owner password, it is very easy to decrypt the document and to remove the restrictions. Encrypting a document with an owner password only works on a psychological level. You indicate that the original producer of the document doesn’t want the document to be printed, but on a technical level, it’s easy to remove this form of protection.

If you want to avoid that an end users saves a PDF document, you are asking something that is impossible. The only way to avoid that an end user doesn’t have a copy of the PDF is by not sending him the PDF. A PDF can’t be opened in Adobe Reader without having the actual bytes on a local disk. Even if you would find a workaround to disable saving (for instance in the context of a web application), you’d always find the PDF somewhere in the temp files and people would be able to copy that file as many times as they want.

In your code snippet, you try hiding the toolbar (a viewer preference), but that’s not efficient. Whether or not this viewer preference will be respected entirely depends on the PDF viewer. For instance: in Adobe Reader X and later, you have a special widget that appears when you hover over the document. This widget allows users to save the document.

Even with Adobe Reader 9, hiding the toolbar isn’t sufficient: if the user chooses the appropriate menu item or hits the appropriate “hot key”, the toolbar would appear and they could happily click the Save button. In addition, they could have right-clicked and chosen “Save” as well.

<sup>16</sup><http://stackoverflow.com/questions/22880444/disable-save-button-in-adobe-pdf-reader-and-hide-menu-bar-in-ie-window>

<sup>17</sup><http://stackoverflow.com/users/3500945/seven-7>

What you need to do is not *to prevent saving*, but *to control the actual use* of the PDF and that's where Digital Rights Management (DRM) comes in. DRM however is usually expensive, it also requires a custom PDF viewer. In any case: it's out of the scope as far as iText is concerned.

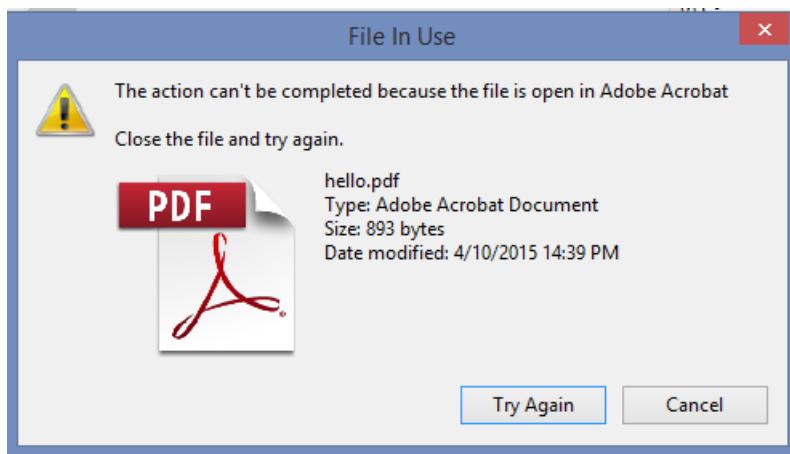
## How to close a PDF file to recreate it? (File in use problem)



I have the Java application that can create a PDF file. For example, I create a simple file from my program and I also wrote some code to open the file as soon as it's created correctly. However, if I want to run the code again, I must close the open file before I recreate it. If I don't close the file I have this error: `java.io.FileNotFoundException: Archivio_Etichette_12-4-2015.pdf (Impossibile accedere al file. Il file è utilizzato da un altro processo)` How can I fixed this problem?

Posted on StackOverflow on [Apr 12, 2015](#)<sup>18</sup> by [bircastri](#)<sup>19</sup>

You need to close the file. The problem is similar to trying to delete or rename a file that is open: if you're working on Windows, Windows will show this error:



“File in use” dialog

You are experiencing the exact same problem: in this case, I tried to rename a file called `hello.pdf` in Windows Explorer. However, this action could not be completed because the file was open in Adobe Acrobat. Tools such as Adobe Reader and Adobe Acrobat need random file access to the file and will therefore lock that file so that no other process can remove, rewrite, rename that file.

The solution is also shown in the dialog box: *Close the file and try again*. You are trying to do something that is impossible (and that is not related or limited to you using iText).

<sup>18</sup><http://stackoverflow.com/questions/29590481/close-the-pdf-file-the-re-create-it>

<sup>19</sup><http://stackoverflow.com/users/2405663/bircastri>

## Note

When working on an iText project, I experience the same problem you describe very often: I write some code, run it, look at the resulting PDF, change the code, run it, and then get the same exception you get. To avoid this, I often create files that have a timestamp in their name. E.g. `hello-20150411163400.pdf`, and then when I run the same code 30 seconds later `hello-20150411163430.pdf` and so on (the filename is created based on the current date and time). This way, I can avoid that exception.

## How to hide the Adobe floating toolbar when showing a PDF in browser?



I am generating a PDF document and displaying it in a Web browser (the current version of IE is the most important target). I want to suppress the floating toolbar (see below) that appears and disappears depending on mouse movement.



Adobe Reader floating toolbar

Is there a way to suppress this toolbar? I can control the PDF document (it's built using iText), as well as the URL.

Posted on StackOverflow on Oct 5, 2012 <sup>20</sup> by Mike Kantor<sup>21</sup>

What you're looking for isn't possible. Read the answer by Leonard Rosenthal<sup>22</sup> (Adobe's PDF architect) on the iText mailing list, where he says: "*there is no way to hide the toolbar (or the HUD) in the browser.*"

Setting the tool bar to false works for the tool bar, but you are referring to the "Heads Up Display" (HUD). Since version X of Adobe Reader, there is a new mode called "Read Mode", which is the default viewing mode when you open a PDF in a web browser. In "Read Mode" you can find a semi-transparent floating toolbar containing basic reading controls, such as page navigation, print and zoom: the HUD.

As documented by Adobe, there is no way to customize this feature, let me quote Adobe<sup>23</sup>:

<sup>20</sup><http://stackoverflow.com/questions/12750725/can-i-hide-the-adobe-floating-toolbar-when-showing-a-pdf-in-browser>

<sup>21</sup><http://stackoverflow.com/users/14607/mike-kantor>

<sup>22</sup><http://permalink.gmane.org/gmane.comp.java.lib.itext.general/60287>

<sup>23</sup><http://blogs.adobe.com/pdfdevjunkie/what-developers-need-to-know-about-acrobat-x>

the “Heads Up Display” (HUD) is not customizable. There are no APIs to HUD. You can’t use JavaScript to enter Read Mode, exit Read Mode or detect that the document is in Read Mode. Though it might seem like it, this wasn’t an oversight. There are some very sound engineering reasons why this is the case but I won’t go into those here.

Summarized: you’re asking something that isn’t supported in Adobe Acrobat / Reader. Unchecking “Display in Read Mode by Default” can be done from *Edit > Preferences > Internet* in Adobe Reader X but there is no way to disable “read mode” programmatically.

## Why are PDF files different even if the content is the same?



I’ve read the following paragraph in “iText in Action - Second Edition” (p 17).

“There’s usually more than one way to create PDF documents that look like identical twins when opened in a PDF viewer. And even if you create two identical PDF documents using the exact same code, there will be small differences between the two resulting files. That’s inherent to the PDF format.”

Can anyone please explain me what kind of differences the author’s talking about and the reason why the PDF format has this defect if I may say.

Posted on StackOverflow on Nov 18, 2013 <sup>24</sup> by programer<sup>25</sup>

Files that are created on a different moment, have a different value for the CreationDate and they have different file identifiers.

Two files, created on a different moment, should have a different ID. The file identifier is usually a hash created based on the date, a path name, the size of the file, part of the content of the PDF file (e.g. the entries in the information dictionary). I quote ISO-32000-1:

The calculation of the file identifier need not be reproducible; all that matters is that the identifier is likely to be unique. For example, two implementations of the preceding algorithm might use different formats for the current time, causing them to produce different file identifiers for the same file created at the same time, but the uniqueness of the identifier is not affected.

<sup>24</sup><http://stackoverflow.com/questions/20039691/reason-why-pdf-files-have-differences>

<sup>25</sup><http://stackoverflow.com/users/2811020/programer8>

File identifiers are mandatory when encrypting a document because they are used in the encryption process. As a result, encrypted PDF files with different file identifiers will have streams that are completely different. This is not a flaw, this is by design.

The ISO specification also allows other differences. For instance: when introducing a font subset, the name of the font is prefixed with a tag that consists of six upper-case letters. The choice of these letters is arbitrary and usually chosen at random. Two identical documents that are generated sequentially may result in different tags for the same font subset.

Another reason why two seemingly identical PDFs may differ internally concerns PDF dictionaries. The order of keys in a dictionary doesn't have any importance in PDF. Software that implements the specification to the letter, will for instance use a `HashMap` to store key/value pairs. Depending on the JVM, the same code can lead to two PDFs with dictionaries that are semantically identical, but of which the entries are sorted in a different way. This is not an error. This is completely compliant with ISO-32000-1.

The syntax that is used to display graphics and text on a page can be reorganized for whatever reason. See section 8.2 of ISO-32000-1 where it says:

The important point is that there is no semantic significance to the exact arrangement of graphics state operators. A conforming reader or writer of a PDF content stream may change an arrangement of graphics state operators to any other arrangement that achieves the same values of the relevant graphics state parameters for each graphics object.

When processing a PDF content stream a PDF processor may change an arrangement of graphics state operators to any other arrangement that achieves the same values of the relevant graphics state parameters for each graphics object. This can be done to optimize the page, to make it render more quickly, to make it easier to debug, to improve the compression, or for any other reason.

**Important:** the internal differences between two PDF files created using the same code, but on a different moment, may not result in a visual difference when opening the document in a PDF viewer or when printing the document on paper.

## Why do PDFs change when processing them?



In the next code snippet, I use PdfStamper, but I don't change anything. I just take the original metadata and I put it back unchanged:

```
public void manipulatePdf(String src, String dest)
    throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    Map<String, String> info = reader.getInfo();
    stamper.setMoreInfo((HashMap<String, String>) info);
    stamper.close();
    reader.close();
}
```

Although I didn't change anything to the `src` file, the `dest` file contains small differences. When I calculate a hash for both files, I get 2 different hash results. May I know why?

Posted on StackOverflow on [Nov 6, 2014](#)<sup>26</sup> by [brian](#)<sup>27</sup>

If you read ISO-32000-1, you should know that no two PDFs are equal *by design*. One of the most typical differences between two PDFs is the ID:

From ISO-32000-1:

ID: An array of two byte-strings constituting a file identifier.

From Section 14.4, entitled “file identifiers”:

The value of this entry shall be an array of two byte strings. The first byte string shall be a permanent identifier based on the contents of the file at the time it was originally created and shall not change when the file is incrementally updated. **The second byte string shall be a changing identifier based on the file's contents at the time it was last updated.** When a file is first written, both identifiers shall be set to the same value. If both identifiers match when a file reference is resolved, it is very likely that the correct and unchanged file has been found. If only the first identifier matches, a different version of the correct file has been found.

<sup>26</sup><http://stackoverflow.com/questions/26773266/when-changing-a-pdf-and-then-removing-the-change-the-hashes-of-the-restored-file>

<sup>27</sup><http://stackoverflow.com/users/4197487/brian>

If you create a PDF from scratch, the ID consists of two identical identifiers. When you update the PDF to add something, the first ID is preserved, the second ID is changed. If you update the PDF to remove that something, that second ID is again changed, but by definition, it should not be identical to the first ID, because you are at a different part of the workflow.

There aren't that many tools that create PDFs of which the identifiers are identical. That's because the PDF that is created from scratch is usually manipulated before the final version is saved to disk. Just create a PDF using Adobe Acrobat to reproduce this: you'll notice that the identifier pair consists of two different values. This makes that it is useless to ask: *can we create a situation where we make the second identifier identical to the first one?*

Moreover: it is inherent to PDF that the way objects are organized is random. Your use case using hashes goes against the PDF standard (see also the previous question).

### How to solve this problem?

In an earlier question, you indicated that you want to add custom metadata and then remove it. In my answer to this question, I explained how to add metadata to an existing PDF using a `PdfStamper` instance:

```
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
```

This creates a new PDF file in which objects are being reordered. You can use `PdfStamper` in *append mode* by changing this line into:

```
PdfStamper stamper = new PdfStamper(reader,
    new FileOutputStream(dest), '\0', true);
```

Now you are creating an *incremental update* of your PDF file.

### What is an incremental update?

Suppose that your **original PDF file** looks like this:

```
%PDF-1.4
% plenty of PDF objects and PDF syntax
%%EOF
```

When you use iText to manipulate such a file, you get an **altered PDF file**:

```
%PDF-1.4
% plenty of altered PDF objects and altered PDF syntax
%%EOF
```

During this process, objects can be renumbered, reorganized, etc... If you add something in a first go, and remove something in a second go, you can expect that the PDF looks the same to the human eye when opening the document in a PDF viewer, but you should not expect the PDF syntax to be identical.

However, when you use `PdfStamper` in *append mode* to perform an incremental update, you get an **incrementally updated PDF**:

```
%PDF-1.4
% plenty of PDF objects and PDF syntax
%%EOF
% updates for PDF objects and PDF syntax
%%EOF
```

In this case, the original bytes of the original PDF aren't changed. The file size gets bigger because you'll now have some redundant information (some objects will no longer be used, or you'll have an old version of some objects along with a new version), but the advantage of using an incremental update is that you can always go back to the original file.

It's sufficient to search for the second last appearance of `%%EOF` and to remove all the bytes that follow. You'll get a **truncated PDF file** like this:

```
%PDF-1.4
% plenty of PDF objects and PDF syntax
%%EOF
```

You can now take a hash of this **truncated PDF file** and compare it with the hash of the **original PDF file**. These hashes will be identical.

**Caveat:** beware of the whitespace characters that follow `%%EOF`. They can cause a minimal difference at the byte level that causes the hashes to be different.

## How to find out if a PDF file compressed or not?



We are using iText to decompress PDFs, but before doing so, we want to know if an existing PDF is already compressed or not. Is there any way we can check if a PDF is compressed or not?

Posted on StackOverflow on Dec 5, 2013 <sup>28</sup> by Vicky<sup>29</sup>

In PDF 1.0 (1993), a PDF file consisted of a mix of ASCII characters for the PDF syntax and binary code for objects such as images. A page stream would contain visible PDF operators and operands, for instance:

<sup>28</sup><http://stackoverflow.com/questions/20412002/is-there-any-way-we-can-find-pdf-file-is-compressed-or-not>

<sup>29</sup><http://stackoverflow.com/users/2898646/vicky>

```
56.7 748.5 m  
136.2 748.5 l  
S
```

This code tells you that a line has to be drawn (S) between the coordinate ( $x = 56.7$ ;  $y = 748.5$ ) because that's where the cursor is moved to with the m operator, and the coordinate ( $x = 136.2$ ;  $y = 748.5$ ) because a path was constructed using the l operator that adds a line.

Starting with PDF 1.2 (1996), one could start using filters for such content streams (page content streams, form XObjects). In most cases, you'll discover a /Filter entry with value /FlateDecode in the stream dictionary. You'll hardly find any "modern" PDFs of which the contents aren't compressed.

Up until PDF 1.5 (2003), all indirect objects in a PDF document, as well as the cross-reference stream were stored in ASCII in a PDF file. Starting with PDF 1.5, specific types of objects can be stored in an objects stream. The cross-reference table can also be compressed into a stream. iText's PdfReader has an isNewXrefType() method to check if this is the case. Maybe that's what you're looking for. Maybe you have PDFs that need to be read by software that isn't able to read PDFs of this type, but... you're not telling us.

Maybe we're completely misinterpreting the question. Maybe you want to know if you're receiving an actual PDF or a zip file with a PDF. Or maybe you want to data-mine the different filters used inside the PDF. In short: your question isn't very clear, and I hope this answer explains why you should clarify.

## How to protect a PDF with a username and password?



Let's say I have a private teaching forum, where each user has a username and an encrypted password (paid account), and there's a DOWNLOAD section where pdf files can be downloaded by users only. These files should be protected by the personal user name and password of these users. In other words, the password should be taken from the user's account and used to open the PDF file that is downloaded. This way, sharing the PDF file with others would force the "copier" to also share his user name and password...

Posted on StackOverflow on May 1, 2014<sup>30</sup> by ijassar<sup>31</sup>

You can't achieve what you want with PDF because encryption with a username and password doesn't exist in PDF.

There are two ways to encrypt a PDF document:

<sup>30</sup><http://stackoverflow.com/questions/23403011/pdf-protection-with-user-and-password>

<sup>31</sup><http://stackoverflow.com/users/3222120/ijassar>

1. Using certificates. You could ask your users to create a public/private key pair. You could then ask them to keep their private key private and ask them to give you their public key. When you encrypt your PDF using their public certificate, you can then encrypt the document with their public key. From that moment on, only the owner of the corresponding private key can read the document. However: the owner of the corresponding private key can also decrypt the document so that it can be shared.
2. Using passwords. You can define two passwords: a *user password* and an *owner password*. A document that is encrypted with an *owner password* can be opened by every one who receives the document. The owner password is there to define permissions (for instance: the document can be viewed, but not printed). Removing the restrictions without knowing the owner password is fairly easy. It used to be illegal when Adobe still owned the copyright on the PDF reference, but since PDF is now an ISO standard, it's not entirely clear if applying the spec to remove the owner password is allowed. If a document is encrypted using a user password, everybody who knows the user password can open the file. There is no username, *only* a user password.

Neither of both cases serve your purpose (read ISO-32000-1 for the full details). The only alternative is to buy an expensive DRM solution.

## How to encrypt PDF using a certificate?



We need to encrypt a PDF with a certificate. I've found something using iText some months ago, but I cannot find it any more.

Posted on StackOverflow on May 21, 2014<sup>32</sup> by user2946593<sup>33</sup>

Encrypting a PDF is done with a public certificate. Once a PDF is encrypted, only the person with the corresponding private certificate can open the PDF. In your scenario, this would mean that only the person who owns the smart card can open the document.

First you need to extract the public certificate from the smart card. The main question here is: do you want to do this in Java? If so, do you want to do this using PKCS#11? Using MSCAPI? Using a smart card API? I honestly don't think that's what you want to do. I think you want the owners of the smart card to extract their public certificate manually and to send it to you. If this assumption is wrong, you need to post another question: how to get a public certificate from a smart card.

Once you have this certificate, you can encrypt the PDF like this:

---

<sup>32</sup><http://stackoverflow.com/questions/23784155/crypt-embedded-attachments-in-pdf-with-java>

<sup>33</sup><http://stackoverflow.com/users/2946593/user2946593>

```

PdfReader reader = new PdfReader(src);
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
Certificate cert = getPublicCertificate("resources/encryption/public.cer");
stamper.setEncryption(new Certificate[]{cert},
    new int[]{PdfWriter.ALLOW_PRINTING}, PdfWriter.ENCRYPTION_AES_128);
stamper.close();
reader.close();

```

The public certificate is stored in the file `public.cer`. That's the file your end user extracted from the smart card.

## How to protect an already existing PDF with a password?



How to set password for an existing PDF?

Posted on StackOverflow on Dec 2, 2014 <sup>[34](#)</sup> by Sreekanth P<sup>[35](#)</sup>

It's as simple as this:

```

public void encryptPdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.setEncryption(USER, OWNER, PdfWriter.ALLOW_PRINTING,
        PdfWriter.ENCRYPTION_AES_128 | PdfWriter.DO_NOT_ENCRYPT_METADATA);
    stamper.close();
    reader.close();
}

```

Note that `USER` and `OWNER` are of type `byte[]`. You have different options for the permissions (look for constants starting with `ALLOW_`) and you can choose from different encryption algorithms.

---

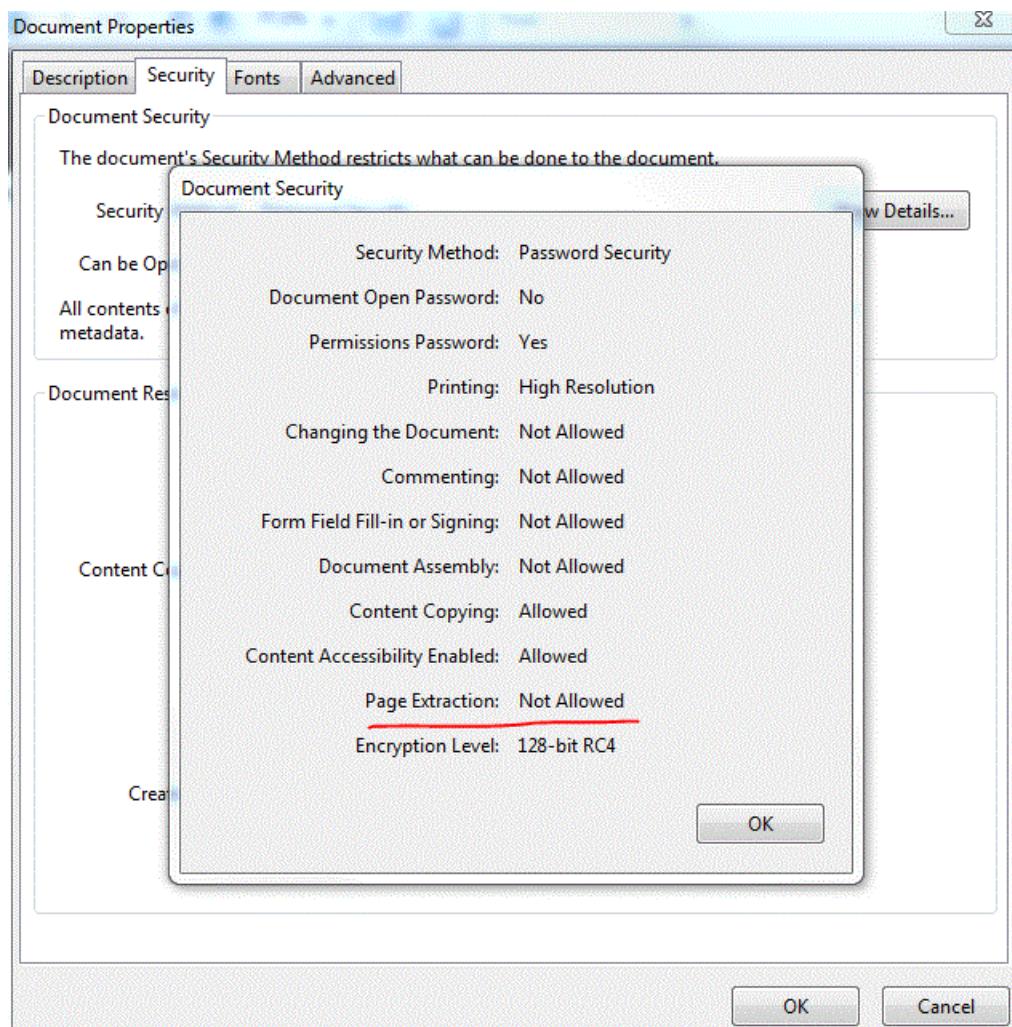
<sup>34</sup><http://stackoverflow.com/questions/27249386/password-protection-for-a-local-already-existing-pdf>

<sup>35</sup><http://stackoverflow.com/users/4315638/sreekanth-p>

## How to allow page extraction when setting password security?



I don't know if it is possible to create a PDF with password security enabled that also allows extraction of pages. I haven't found any property in iTextSharp which will allow enable page extraction. The following screen shot shows the property that I would like to enable.



### Permissions

Posted on StackOverflow on Sep 11, 2014 <sup>36</sup> by Ruben de la Fuente<sup>37</sup>

<sup>36</sup><http://stackoverflow.com/questions/25783530/allow-page-extraction-in-a-password-security-pdf-with-itextsharp>

<sup>37</sup><http://stackoverflow.com/users/2236139/ruben-de-la-fuente>

I've taken a look at the permission bits in the draft of ISO-32000-2 and I've compared them with the parameters (written in ALL\_CAPS) available in iText:

- bit 1: Not assigned
- bit 2: Not assigned
- bit 3: Degraded printing: ALLOW\_DEGRADED\_PRINTING
- bit 4: Modify contents: ALLOW MODIFY CONTENTS
- bit 5: Extract text / graphics: ALLOW\_COPY
- bit 6: Add / Modify text annotations: ALLOW\_MODIFY\_ANNOTATIONS
- bit 7: Not assigned
- bit 8: Not assigned
- bit 9: Fill in fields: ALLOW\_FILL\_IN
- bit 10: **Deprecated** ALLOW\_SCREEN\_READERS
- bit 11: Assembly: ALLOW\_ASSEMBLY
- bit 12: Printing: ALLOW\_PRINTING

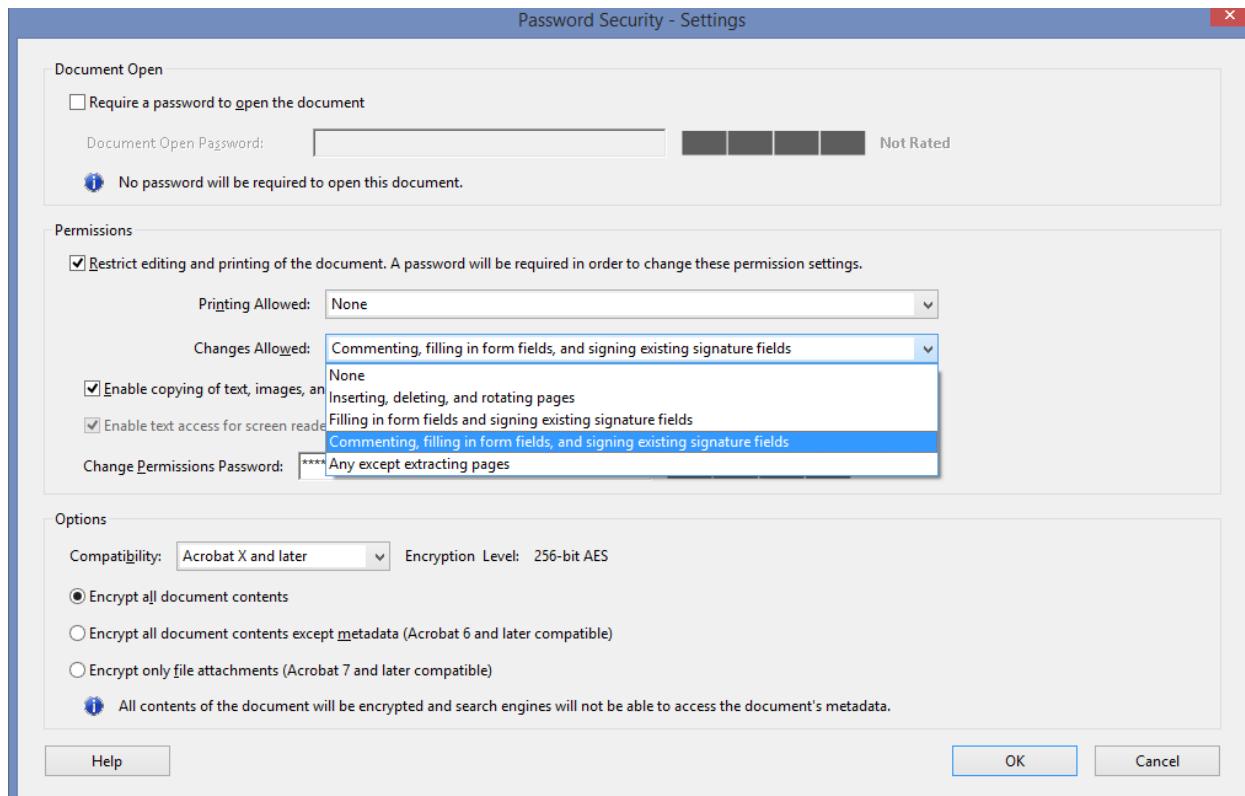
When I compare the spec with your screen shot, I assume that the permissions are as follows:

- Printing: ALLOW\_DEGRADED\_PRINTING or ALLOW\_PRINTING
- Changing the document: ALLOW\_MODIFY\_CONTENTS
- Commenting: ALLOW\_MODIFY\_ANNOTATIONS
- Form Field Fill-in or Signing: ALLOW\_FILL\_IN
- Document Assembly: ALLOW\_ASSEMBLY
- Content copying: ALLOW\_COPY
- Content Accessibility Enabled: ALLOW\_SCREENREADERS

I can't find any permission bit that refers to page extraction. I have tried setting all the flags that are documented in ISO-32000-2, but they didn't result in setting the Page Extraction to Allowed.

I have tried two things:

First I tried setting the bits that aren't assigned: bit 1, 2, 7, 8, 13, 14. This didn't change anything. Then I opened a test document in Acrobat and I tried finding a setting that would allow page extraction:



Screen shot

I couldn't find any.

As the permission isn't described in ISO-32000 and as there doesn't seem to be a way to set this permission in Acrobat, I'm inclined to believe that there is no way to set this permission. The only way to see "Allowed", is to open the document with the owner password.

Please update your question as soon as you find a way to set this permission with Acrobat. I'm using Acrobat XI Pro.

## How do I rotate a PDF page to an arbitrary angle?



I need to rotate a PDF page by an arbitrary angle, but it seems that the functionality to rotate pages is restricted to multiples of 90 degrees. The contents of the page are vector and text and I need to be able to zoom in on the contents later, which means that I cannot convert the page to an image because of the loss of resolution.

Posted on StackOverflow on Feb 1, 2015 <sup>38</sup> by sdmorris<sup>39</sup>

<sup>38</sup><http://stackoverflow.com/questions/28260880/how-do-i-rotate-a-pdf-page-to-an-arbitrary-angle>

<sup>39</sup><http://stackoverflow.com/users/3265237/sdmorris>

Please read ISO-32000-1 (this is the ISO standard for PDF), more specifically Table 30 (“Entries in a page object”). It defines the Rotate entry like this (literal copy/paste):

The number of degrees by which the page shall be rotated clockwise when displayed or printed. The value shall be a multiple of 90. Default value: 0.

Whenever an ISO standard uses the word *shall*, you are confronted a normative rule (as opposed to when the standard uses the word *should* in which case you’re facing a recommendation).

In short: you are asking something that is explicitly forbidden by the PDF specification. Meeting your requirement is impossible in PDF. Your page can have an orientation of 0, 90, 180 or 270 degrees. You will have to rotate the contents on the page instead of rotating the page.

## What is the size limit of pdf file?



I am using iTextSharp in a web application to generate PDF files. These PDF files contain .tiff images taken from a folder. If size of the contents of such a folder goes beyond 1GB then the browser gets closed automatically when receiving the PDF file. Is there a size limit when generating PDF files?

Posted on StackOverflow on Feb 2, 2015 <sup>40</sup> by sanket<sup>41</sup>

The answer to your question depends on three parameters:

1. The PDF version: before PDF 1.5 vs. PDF 1.5 and higher,
2. the PDF style: plain text cross-reference table vs cross reference stream, and
3. the iText(Sharp) version: before 5.3 vs 5.3 and higher).

Let’s start with the PDF version and the cross-reference table. In case you didn’t know: the cross-reference table defines the byte offsets of every object inside the PDF.

In PDF versions prior to PDF 1.5, the Cross-reference table is added in plain text, and each byte offset is defined using ten digits. Hence the size of a file will be limited to 10 to the tenth bytes (approximately 10 gigabytes).

**The maximum size of a PDF with a version older than PDF 1.5 is about 10 gigabytes.**

Starting with PDF 1.5, you can choose whether you want to create a cross-reference table in plain text, or whether you want to use a cross-reference stream. If you use a cross-reference stream, the “10 digits only” limitation disappears.

<sup>40</sup><http://stackoverflow.com/questions/28277273/what-is-the-size-limit-of-pdf-file-when-generated-using-c-sharp-code-with-images>

<sup>41</sup><http://stackoverflow.com/users/4519826/sanket>

**The maximum size of a PDF with a version PDF 1.5 or higher using a compressed cross-reference stream depends only on the limitations of the software processing the PDF.**

If you use iText(Sharp), you need to make sure that you create your files with a compressed cross-reference stream if you need files bigger than 10 gigabytes.

The version of iText also matters:

- All versions prior to iText 5.3 only supported files up to about 2 gigabytes because all byte offsets are stored as int values in those versions.
- From 5.3 on, iText supports files up to 1 terabyte because all byte offsets are stored as long values in those versions.

**The maximum size of a PDF created with iText versions before 5.3 is 2 gigabytes. The maximum size of a PDF created with iText versions 5.3 and higher is 1 terabyte.**

However: you also need to take into consideration that not all viewers can render a file that big. While the file may be completely compliant with ISO-32000-1, you may hit memory limits.

You are sending 1 gigabyte over the internet to a random PDF viewer plugin in a random browser on a random machine. If it's an old machine, it won't even have 1 gigabyte of memory. When the end user has a slow connection, the end user will have to wait forever to get the full file. There is no way you can predict how the browser and the PDF viewer will respond. My guess is that the browser shuts down because of an out-of-memory exception. This is not a problem that is caused by a limitation that is inherent to PDF, nor to a limitation of iText(Sharp). It is purely a limitation that is inherent to your design. You shouldn't send gigabytes to a browser. Write them to a shared drive in the cloud instead.

## Can I change the page count by changing internal metadata?



Some days ago I started to play with the internal structure of PDF document. While searching the internet, I've found some nice tools to edit the metadata, so far I haven't found how (and if) I can edit the page count in a way it won't affect the way the PDF is visualized. Can I change some metadata field, so that I can see a "fake" page count? If so, how is it done? Where can I read more about the internal structure of a PDF document?

Posted on StackOverflow on Feb 11, 2015 <sup>42</sup> by shalom shalom<sup>43</sup>

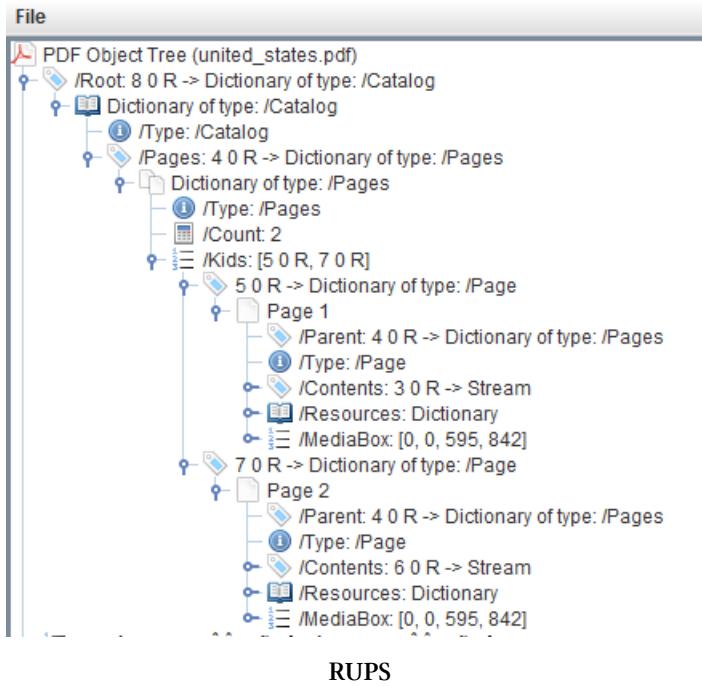
The page count isn't stored anywhere inside a PDF, hence you can not set some value to fool people into believing that there aren't as many pages as there are.

<sup>42</sup><http://stackoverflow.com/questions/28428410/pdf-change-internal-metadata-page-count>

<sup>43</sup><http://stackoverflow.com/users/2088560/shalom-shalom>

You can read more about the internal structure of a PDF document in ISO-32000-1, but if you prefer a more practical approach, please download [iText RUPS<sup>44</sup>](#) and take a look at the internal structure of a PDF.

I have opened a file with two pages in RUPS as an example:



As you can see, the `/Catalog` dictionary (aka the root dictionary) of the PDF file has an entry named `/Pages`. This is the root of the page tree.

This `/Pages` entry has `/Kids` that can either be another `/Pages` dictionary (a branch of the page tree) or a `/Page` dictionary (a leaf of the page tree). In this case, we see two `/Page` elements.

A PDF viewer will traverse through the page tree and count the number of `/Page` dictionaries to calculate the total page count.

If you want to change the page count, you can remove `/Page` dictionaries, but that will also remove the pages. You can add `/Page` dictionaries, but that will add pages.

The number of pages of a document isn't stored in metadata, hence your question to change the page count by changing some metadata is not a valid question.

---

<sup>44</sup>[http://itextpdf.com/product/itext\\_rups](http://itextpdf.com/product/itext_rups)

## How to get the page number of an arbitrary PDF object?



I am trying to find the page number of a PDF object using iText's Java API. The following code reads in the PDF file, and gets the object containing the open action. How do I get the page number of that object?

```
PdfReader soPdfItext = null;
try {
    soPdfItext = new PdfReader(new FileInputStream(f));
} catch (IOException e) { }
/* Get the catalog */
PdfDictionary soCatalog = soPdfItext.getCatalog();
/* Get the object referring to the open action */
PRIndirectReference soOpenActionReference =
    (PRIndirectReference) soCatalog.get(PdfName.OPENACTION);
/* Get the actual object containing the open action */
PdfObject soOpenActionObject =
    originalPdfItext.getPdfObject(soOpenActionReference.getNumber());
```

Now what? There is a class Document that contains a method getPageNumber(), but I'm not sure if a) it's relevant to what I want to do and b) if it is relevant, how to implement.

Posted on StackOverflow on Jun 15, 2015<sup>45</sup> by user271621<sup>46</sup>

There are no such things as page numbers in a PDF. Pages are part of a page tree. This page tree consists of /Pages elements (the branches of the tree) and /Page elements (the leaves of the tree). The page index is calculated by traversing the different branches and leaves of the tree. Optionally, a PDF also defines /PageLabels. If you know the page index and if you have the definition of the page labels, you can derive the page number.

You are extracting an PdfObject that represents an open action. It can be a PdfDictionary or a PdfArray.

### **PdfDictionary**

If the PdfObject is an instance of a PdfDictionary, then you need to look at the /S item of this dictionary to find out which type of action will be triggered.

- That action could be some JavaScript. If that JavaScript contains an action that jumps to a specific page, there might be a page number in that method.

---

<sup>45</sup><http://stackoverflow.com/questions/30855432/getting-page-number-of-arbitrary-pdf-object-using-itext>

<sup>46</sup><http://stackoverflow.com/users/5009319/user271621>

- That action could be a GoTo action, in which case you need to look at the /D entry for the *destination* (\*).

There are 20 possible types of actions, and actions can be *chained*, so it's up to you to loop through the action chain and to examine every possible action.

This is an example:

```
/OpenAction<</D[8 0 R/Fit]/S/GoTo>>
```

The << and >> indicate that the open action is described using a dictionary. The /S shows that you have a /GoTo action and /D describes the destination.

### PdfArray

If the PdfAction is an instance of a PdfArray, then this array is a *destination* (\*).

This is an example:

```
/OpenAction[6 0 R/XYZ 0 806 0]
```

### (\*) Destination

A destination is an array that consists of a variable number of elements. These are some examples:

```
[8 0 R/Fit]
[6 0 R/XYZ 0 806 0]
```

The first example is an array with two elements 8 0 R and /Fit. The second example is an array with four elements 6 0 R, /XYZ, 0, 806 and 0. You need the first element. It doesn't give you the page number (because there is no such thing as page numbers), but it gives you a reference to the /Page object. Based on that reference, you can deduce the page number by looping over the page tree and comparing the object number of a specific page with the object number in the destination.

## Where is the origin (x,y) of a PDF page?



I want to position text at some specific place in the document. However, I cannot figure out where to find the origin of the coordinate system of the page. What is the coordinate of the bottom left corner, top right corner, bottom right corner, top left corner? Where is this origin?

Posted on StackOverflow on May 19, 2015<sup>47</sup> by Giovanrich<sup>48</sup>

The dimensions of a page (aka the page boundaries) are defined in a page dictionary:

---

<sup>47</sup><http://stackoverflow.com/questions/30319228/where-is-the-origin-x-y-of-pdf-page>

<sup>48</sup><http://stackoverflow.com/users/4075832/giovanrich>

- `/MediaBox`: the boundaries of the physical medium (the page). This value is mandatory, so you'll find it in every PDF.
- `/CropBox`: the region that is visible when displayed or printed. The `/CropBox` is equal to or smaller than the `/MediaBox`. This value is optional; if it's missing, the `/CropBox` is equal to the `/MediaBox`.
- Other possible values are the `/BleedBox`, `/TrimBox` and `/ArtBox`. These have been defined for specific purposes, but they aren't used that often anymore. If they are missing, they default to the `/CropBox`. None of these values can outsize the `/CropBox`.

When you create a document with iText, you define the `/MediaBox` either explicitly or implicitly.

Explicitly:

```
Rectangle rect = new Rectangle(20, 20, 300, 600);
Document document = new Document(rect);
```

Implicitly:

```
Document document = new Document();
```

This single line is equivalent to:

```
Rectangle rect = new Rectangle(0, 0, 595, 842);
Document document = new Document(rect);
```

The four parameters passed to the `Rectangle` constructor (`llx, lly, urx, ury`) define a rectangle using the x and y coordinates of the lower-left and the upper-right corner.

In case of `new Rectangle(0, 0, 595, 842)`, the lower-left corner of the page coincides with the origin of the coordinate system `(0, 0)`. The upper-right corner of the page coincides with the coordinate `(595, 842)`.

All measurements are defined in user units, and by default, the user units *roughly* corresponds with the typographic point: 1 user unit = 1 point.

Note the word *roughly*: we use points to do calculations, but in the ISO standard, we are very cautious not to use point as a synonym for user unit. For instance: an A4 page measures 595 by 842 user units, but if you'd calculate the exact value in points, there would be a slight difference (some numbers after the point).

The lower-left corner of the page isn't always the origin of the coordinate system. If we define a page using `Rectangle(20, 20, 300, 600)`, the origin is 20 user units below and 20 user units to the left of the lower-left corner. It's also possible to use negative values to define a page size.

For instance: suppose that you want to create an A2 document consisting of 4 A4 pages, than you could define the page sizes like this:

```
Rectangle(-595, 0, 0, 842)  Rectangle(0, 0, 595, 842)
Rectangle(-595, -842, 0, 0) Rectangle(0, -842, 595, 0);
```

By defining the media box like this, you also pass information with respect to the relative position of the different pages. If you look at the 4 A4 pages as a unit, the origin of the coordinate system is the exact center of the A2 page.

**Important:**

All of the above assumes that you didn't introduce any coordinate transformations, e.g. using the concatCTM() or transform() method. These methods allow you to change the coordinate system, for instance change the angle between the x and y axis from 90 degrees (the default) to another angle. You can also scale an axis to get a different aspect ratio. While it's certainly fun to do this, it requires quite some Math.

## How should I interpret the coordinates of a rectangle in PDF?



I would like to know more about the coordinates of a Rectangle, in particular:

- lower left X
- lower left Y
- upper right X
- upper right Y

Every time, I get confused about how to make dimensions based on these coordinates to draw rectangle. If possible, can I get a graphical representation briefly about these coordinates positions?

Posted on StackOverflow on Jun 10, 2015 <sup>49</sup> by Nazeerbasha<sup>50</sup>

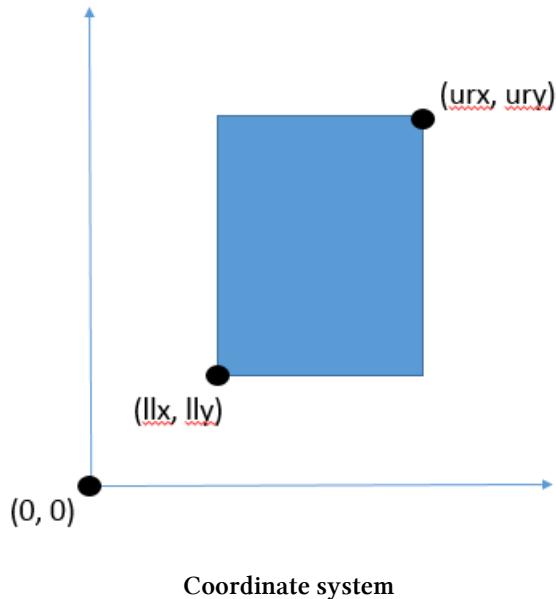
Before someone can explain what the lower-left X, lower-left Y, upper-right X and upper-right Y of a rectangle are about, you need to know about the coordinate system as explained in the answer to the question "Where is the origin (x,y) of a PDF page?"

The answer to that question contains all the information you need, except for the graphical representation you are asking for. This is a simple representation of the coordinate system:

---

<sup>49</sup><http://stackoverflow.com/questions/30748651/coordinates-of-rectangle-in-itextpdf>

<sup>50</sup><http://stackoverflow.com/users/4974002/nazeerbasha>



Coordinate system

The origin of the coordinate system is  $(0, 0)$ . Positive X values are to the right of the origin, positive Y values are above the origin.

I have drawn a Rectangle and indicates where you can find the lower-left corner (with coordinate  $(llx, lly)$ ) and the upper-right corner (with coordinate  $(urx, ury)$ ).

The sides of the rectangle are always in parallel with the X and the Y axis, hence you only need two coordinates to define the rectangle.

**Important:**

All of the above assumes that you didn't introduce any coordinate transformations, e.g. using the `concatCTM()` or `transform()` method. These methods allow you to change the coordinate system, for instance change the angle between the x and y axis from 90 degrees (the default) to another angle. You can also scale an axis to get a different aspect ratio.

# Getting started

The most popular iText example is the “Hello World” example, explaining the five steps to create a PDF from scratch using iText:

```
// step 1
Document document = new Document();
// step 2
PdfWriter.getInstance(document, new FileOutputStream(filename));
// step 3
document.open();
// step 4
document.add(new Paragraph("Hello World!"));
// step 5
document.close();
```

Obviously, iText is capable of doing much more than creating a PDF that shows the words “Hello World”, but let’s take a look at some basic questions to get started.

## How to generate and design PDFs with iText or iTextSharp?



I’m wondering what is the best/easiest way to design a PDF document? Is it remotely legit to actually design a whole PDF document with iTextSharp with code (i.e not loading external files)? I want the final result to look similar to a web page with various colors, borders, images and everything.

Or do you have to rely on other documents like .doc, .html files to achieve a good design? Originally I thought that I would use HTML markup to generate a PDF, but why even use a HTML markup or a template file to create the PDF design when I could just do it right within the PDF without having to rely on on various files that serves no real purpose.

Is it possible to generate and design big PDF documents using code and are there any more proper guides or similar with all the various commands to generate texts, images, borders and everything since I have no real clue about generating PDF with code.

Posted on StackOverflow on Oct 6, 2014 <sup>51</sup> by HenrikP<sup>52</sup>

---

<sup>51</sup><http://stackoverflow.com/questions/26218444/generate-and-design-pdf-with-itextsharp-or-similar>

<sup>52</sup><http://stackoverflow.com/users/2914876/henrikp>

The question is very broad, so I can only give you a very broad answer.

**Option 1:** you create your layout by using iText's high-level objects. There are countless applications out there that are using PdfPTable to generate complex reports. For instance: the time tables for a German Railway company are created from scratch through code; the invoices for a Belgian Telco company are created this way,... The advantage of this approach is that you can really fine-tune the layout. The disadvantage is that you need to change source code as soon as you want to change the layout.

**Option 2:** you create your layout by creating an AcroForm template. Every field in this template has a name and is visualized at exact positions (defined by its coordinates) on specific pages. The code to fill out such a form consists of only a handful of lines. Whenever you need to change the layout, you alter the AcroForm template. You do not need to change your code. The disadvantage is that AcroForms are very static. Compare it to a paper form: you can't insert a row in a paper form either.

**Option 3:** you create your data in XHTML format and your styles in CSS. A Belgian printing company responsible for creating invoices for its customers is streaming data into very simple HTML files involving a sequence of tables that never span more than a handful of pages. These files are then fed to iText's XML worker along with a CSS that is different for each of its customers. The advantage of this approach is that no extra programming is needed when a new customer joins. It's just a matter of creating a new CSS. The disadvantage is that you are limited by the HTML format. Elementary logic also tells you that you shouldn't expect URL2PDF: have you ever tried printing a website? Well, the bad quality of that print should give you an indication of the problems you'll encounter when trying to convert HTML to PDF. If you anticipate them, you can get good results. If you don't: it's a poor craftsman who blames his tools...

**Option 4:** define your template using the XML Forms Architecture (XFA). Such templates are usually created using Adobe LiveCycle Designer. An XSD is fed into LC Designer and the result is an empty form where the PDF format acts as a container for an XML stream. You can then use iText to inject your custom XML containing data that conforms with the XSD into the PDF and you can use XFA Worker to flatten such a form. XFA Worker is only available as a closed source product (givers need to set limits because takers rarely do).

**Option 5:** right now XML Worker is used to convert XHTML+CSS and XFA to PDF (ordinary PDF, PDF/A, PDF/UA). You could use the generic XML Worker engine to support your own XML format. The advantage would be a very powerful engine that you can tune to meet your exact needs. The disadvantage is that this involves a serious up-front development investment.

**Option 6:** use a third party tool to define the template and a third party server that uses iText under the hood to create PDFs based on the template. An example of such a third party tool is Scriptura developed by Inventive Designers. There are other tools, but Inventive Designers is a customer of iText and we know that they are using iText correctly whereas we don't have this guarantee from other vendors.

## Argument not specified in iTextSharp



As per suggestion by Bruno Lowagie, I am using the following code:

```
Dim table As New PdfPTable(3)
table.setWidthPercentage(100)
table.addCell(getCell("Text to the left", PdfPCell.ALIGN_LEFT))
table.addCell(getCell("Text in the middle", PdfPCell.ALIGN_CENTER))
table.addCell(getCell("Text to the right", PdfPCell.ALIGN_RIGHT))
document.add(table)

Public Function getCell(ByVal text As String, ByVal alignment As Integer)
    As PdfPCell
    Dim cell As New PdfPCell(New Phrase(text))
    cell.setPadding(0)
    cell.setHorizontalTextAlignment(alignment)
    cell.setBorder(PdfPCell.NO_BORDER)
    Return cell
End Function
```

But I am getting the errors that `cell.setPadding`, `cell.setHorizontalTextAlignment`, `cell.setBorder` all are not member of `iTextsharp.Text.pdf.PdfPCell`. Also `table.setWidthPercentage(100)` shows the error *argument not specified parameter*.

Posted on StackOverflow on Apr 11, 2015 <sup>[53](#)</sup> by Sunil Bhagwat<sup>[54](#)</sup>

I adapted your example like this:

```
Dim table As New PdfPTable(3)
table.WidthPercentage = 100
table.AddCell(GetCell("Text to the left", PdfPCell.ALIGN_LEFT))
table.AddCell(GetCell("Text in the middle", PdfPCell.ALIGN_CENTER))
table.AddCell(GetCell("Text to the right", PdfPCell.ALIGN_RIGHT))
document.Add(table)

Public Function GetCell(ByVal text As String, ByVal alignment As Integer)
    As PdfPCell
    Dim cell As New PdfPCell(New Phrase(text))
    cell.Padding = 0
    cell.HorizontalTextAlignment = alignment
```

---

<sup>53</sup><http://stackoverflow.com/questions/29575793/how-to-format-paragraph-string-to-show-content-left-right-or-middle-of-pdf-docu>

<sup>54</sup><http://stackoverflow.com/users/4754977/sunil-bhagwat>

```
cell.Border = PdfPCell.NO_BORDER  
Return cell  
End Function
```

This is commonly known:

- Methods in Java start with lower case; methods in .NET start with upper case, so when people ask you to use Java code as pseudo code and to convert Java to .NET, you need to change methods such as `add()` and `addCell()` into `Add()` and `AddCell()`.
- Member-variables in Java are changed and consulted using getters and setters; variables in .NET are changed and consulted using methods that look like properties. This means the you need to change lines such as `cell.setBorder(border);` and `border = cell.getBorder();` into `cell.Border = border` and `border = cell.Border.`

iText and iTextSharp are kept in sync, which means that, using the two rules explained above, a developer won't have any problem to convert iText code into iTextSharp code.

## How to create a complex PDF document?



I have an Java/Java EE based application wherein I have a requirement to create PDF certificates for various services that will be provided to the users. I am looking for a way to create PDF (no need for digital certificates for now). What is the easiest and convenient way of doing that? I have tried

1. XSL to PDF conversion
2. HTML to PDF conversion using itext.
3. the crude Java way (using `PdfWriter`, `PdfPTable`, etc.)

What is the best way out of these, or is there any other way which is easier and convenient?

Posted on StackOverflow on Jan 4, 2013 <sup>55</sup> by Ankit<sup>56</sup>

When you talk about Certificates, I think of standard sheets that look identical for every receiver of the certificate, except for:

- the name of the receiver,
- the course that was followed by the receiver,

<sup>55</sup><http://stackoverflow.com/questions/14151335/creating-complex-pdf-using-java>

<sup>56</sup><http://stackoverflow.com/users/810176/ankit>

- a date.

If this is the case, I would use any tool that allows you to create a fancy certificate (Acrobat, Open Office, Adobe InDesign,...) and create a static form (sometimes referred to as an AcroForm) containing three fields: name, course, date.

I would then use iText to fill in the fields like this:

```
PdfReader reader = new PdfReader(pathToCertificateTemplate);
PdfStamper stamper =
    new PdfStamper(reader, new FileOutputStream(pathToCertificate));
AcroFields form = stamper.getAcroFields();
form.setField("name", name);
form.setField("course", course);
form.setField("date", date);
stamper.setFormFlattening(true);
stamper.close();
reader.close();
```

Creating such a certificate from code is “the hard way”; creating such a certificate from XML is “a pain” (because XML isn’t well-suited for defining a layout), creating a certificate from (HTML + CSS) is possible with iText’s XML Worker, but all of these solutions have the disadvantage that it’s hard work to position every item correctly, to make sure everything fits on the same page, etc...

It’s much easier to maintain a template with fixed fields. This way, you only have to code once. If for some reason you want to move the fields to another place, you only have to change the template, you don’t have to worry about messing around in code, XML, HTML or CSS.

Please go to the section about interactive forms to learn more about this technology.

## How to align two paragraphs to the left and right on the same line?



I want to display two pieces of content (it may a paragraph or text) to the left and right side on a single line. My output should be like

Name : ABC

date : 2015-03-02

How can I do this?

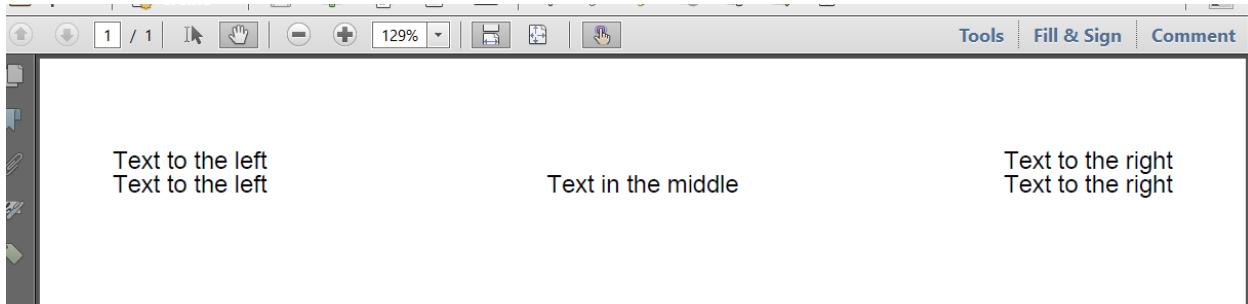
Posted on StackOverflow on Apr 11, 2015 <sup>57</sup> by Mohan Lal<sup>58</sup>

---

<sup>57</sup><http://stackoverflow.com/questions/29575142/how-to-align-two-paragraphs-or-text-in-left-and-right-in-a-single-line-in-pdf>

<sup>58</sup><http://stackoverflow.com/users/4539874/mohan-lal>

Please take a look at the [LeftRight<sup>59</sup>](#) example. It offers two different solutions for your problem:



Screen shot

### Solution 1: Use glue

By glue, I mean a special Chunk that acts like a separator that separates two (or more) other Chunk objects:

```
Chunk glue = new Chunk(new VerticalPositionMark());
Paragraph p = new Paragraph("Text to the left");
p.add(new Chunk(glue));
p.add("Text to the right");
document.add(p);
```

This way, you will have "Text to the left" on the left side and "Text to the right" on the right side.

### Solution 2: use a PdfPTable

Suppose that some day, somebody asks you to put something in the middle too, then using PdfPTable is the most future-proof solution:

```
PdfPTable table = new PdfPTable(3);
table.setWidthPercentage(100);
table.addCell(getCell("Text to the left", PdfPCell.ALIGN_LEFT));
table.addCell(getCell("Text in the middle", PdfPCell.ALIGN_CENTER));
table.addCell(getCell("Text to the right", PdfPCell.ALIGN_RIGHT));
document.add(table);
```

In your case, you only need something to the left and something to the right, so you need to create a table with only two columns: `table = new PdfPTable(2)`.

In case you wonder about the `getCell()` method, this is what it looks like:

---

<sup>59</sup><http://itextpdf.com/sandbox/objects/LeftRight>

```
public PdfPCell getCell(String text, int alignment) {
    PdfPCell cell = new PdfPCell(new Phrase(text));
    cell.setPadding(0);
    cell.setHorizontalAlignment(alignment);
    cell.setBorder(PdfPCell.NO_BORDER);
    return cell;
}
```

### Solution 3: Justify text

This is explained in the answer to this question: [How justify text using iTextSharp?](#)<sup>60</sup>

However, this will lead to strange results as soon as there are spaces in your strings. For instance: it will work if you have "Name:ABC". It won't work if you have "Name: Bruno Lowagie" as "Bruno" and "Lowagie" will move towards the middle if you justify the line.

## How to align a label versus its content?



I have a label (e.g. "A list of stuff") and some content (e.g. an actual list). When I add all of this to a PDF, I get:

A list of stuff: test A, test B, coconut, coconut, watermelons, apple, oranges,  
many more fruities, carshow, monstertrucks thing

I want to change this so that the content is aligned like this:

A list of stuff: test A, test B, coconut, coconut, watermelons, apple, oranges,  
many more fruits, carshow, monstertrucks thing, everything is  
starting on the same point in the line now

In other words: I want the content to be aligned so that every line starts at the same X position, no matter how many items are added to the list.

Posted on StackOverflow on [Feb 12, 2015](#) <sup>61</sup> by [Patrick Fritch](#)<sup>62</sup>

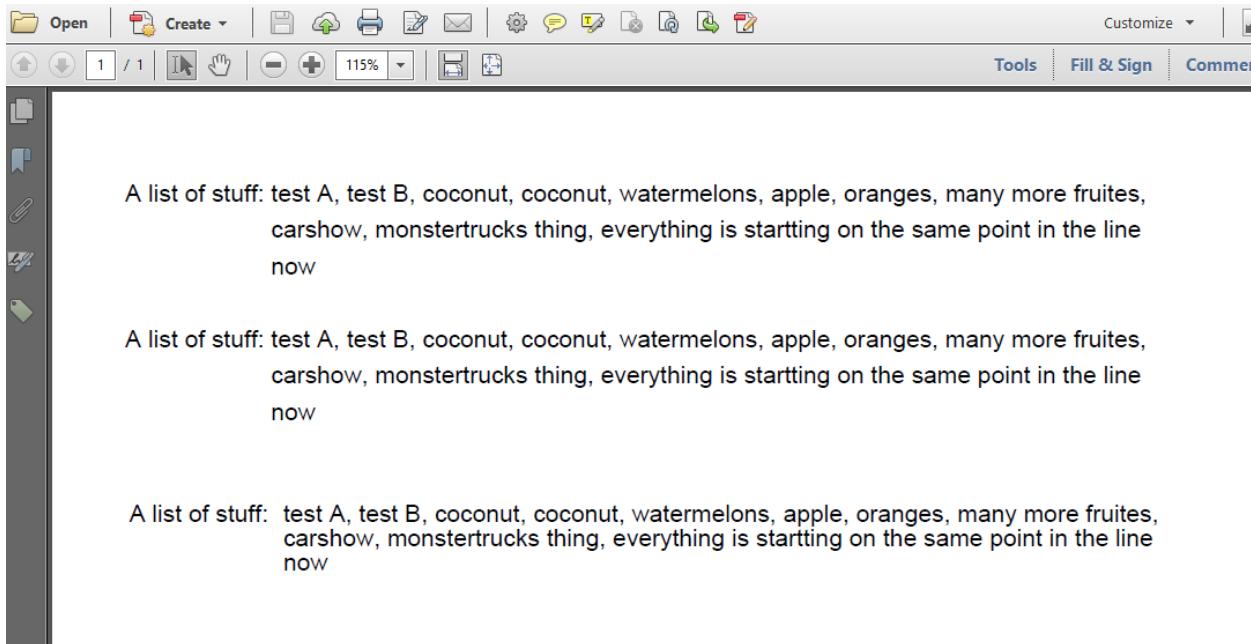
There are many different ways to achieve what you want: Take a look at the following screen shot:

---

<sup>60</sup><http://stackoverflow.com/questions/10145557/how-justify-text-using-itextsharp>

<sup>61</sup><http://stackoverflow.com/questions/28487918/how-to-make-it-start-from-the-same-point>

<sup>62</sup><http://stackoverflow.com/users/4542399/patrick-fritch>



### Indentation options

This PDF was created using the [IndentationOptions<sup>63</sup>](#) example.

In the first option, we use a `List` with the label ("A list of stuff: ") as the list symbol:

```
List list = new List();
list.setListSymbol(new Chunk(LABEL));
list.add(CONTENT);
document.add(list);
document.add(Chunk.NEWLINE);
```

In the second option, we use a paragraph of which we use the width of the `LABEL` as indentation, but we change the indentation of the first line to compensate for that indentation.

```
BaseFont bf = BaseFont.createFont();
Paragraph p = new Paragraph(LABEL + CONTENT, new Font(bf, 12));
float indentation = bf.getWidthPoint(LABEL, 12);
p.setIndentationLeft(indentation);
p.setFirstLineIndent(-indentation);
document.add(p);
document.add(Chunk.NEWLINE);
```

In the third option, we use a table with columns for which we define an absolute width. We use the previously calculated width for the first column, but we add 4, because the default padding (left and right) of a cell equals 2. (Obviously, you can change this padding.)

---

<sup>63</sup><http://itextpdf.com/sandbox/objects/IndentationOptions>

```

PdfPTable table = new PdfPTable(2);
table.getDefaultCell().setBorder(Rectangle.NO_BORDER);
table.setTotalWidth(new float[]{indentation + 4, 519 - indentation});
table.setLockedWidth(true);
table.addCell(LABEL);
table.addCell(CONTENT);
document.add(table);

```

There may be other ways to achieve the same result, and you can always tweak the above options. It's up to you to decide which option fits best in your case.

## how to match the size of graphics with the size of a page?



I am creating a graphical object using `PdfGraphics2D`, but it has a different size than my page. How do I match the size of my graphics with the size of my page?

```

Document document = new Document();
PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream(dest));
document.open();
PdfContentByte cb = writer.getDirectContent();
PdfTemplate pdfTemplate = cb.createTemplate(750,750);
Graphics2D g2 = pdfTemplate.createGraphics(750, 750);
Drawer drawer = new Drawer();
drawer.setSource(new File(jTextField1.getText()));
drawer.paintComponent(g2);
g2.dispose();
cb.addTemplate(pdfTemplate, 10, 10);
document.close();

```

When I open the saved file, I can't see the totality of the graphics; I just see about 60% of the graphics. How match the size of the graphics with the size of the page?

Posted on StackOverflow on Mar 23, 2015 <sup>64</sup> by jiji<sup>65</sup>

When you create a `Document` object like this:

---

<sup>64</sup><http://stackoverflow.com/questions/29209844/how-to-adapt-the-size-of-graphics-to-the-size-of-a-pdf-page-jav>

<sup>65</sup><http://stackoverflow.com/users/4597081/jiji>

```
Document document = new Document();
```

You are creating a document with size A4 that measures 595 x 842 user units (1 user unit = 1 point by default).

You also create a PdfTemplate that measures 750 x 750 user units:

```
PdfTemplate pdfTemplate = cb.createTemplate(750,750);
```

And you add this template with an offset of 10 user units:

```
cb.addTemplate(pdfTemplate, 10, 10);
```

Graphics that measure 750 by 750 user units do not match with pages that measure 595 by 842 user units. How do we solve this?

#### Option 1: adapt the page to your graphics:

I assume that you want a margin of 10 user units on every side, so the dimension of your page should be at least 770 x 770 user units (750 to accommodate the image plus 10 times two for the margins on either side). In that case, you should create your document like this:

```
Document document = new Document(new Rectangle(770, 770));
```

Now you are creating a document with a page that corresponds with the size of the graphic you want to add.

#### Option 2: adapt the size of your graphics to the size of your page:

Throw away this code:

```
PdfTemplate pdfTemplate = cb.createTemplate(750,750);
Graphics2D g2 = pdfTemplate.createGraphics(750, 750);
Drawer drawer = new Drawer();
drawer.setSource(new File(jTextField1.getText()));
drawer.paintComponent(g2);
g2.dispose();
cb.addTemplate(pdfTemplate, 10, 10);
```

Replace it with:

```

PdfTemplate pdfTemplate = cb.createTemplate(750, 750);
Graphics2D g2 = pdfTemplate.createGraphics(750, 750);
Drawer drawer = new Drawer();
drawer.setSource(new File(jTextField1.getText()));
drawer.paintComponent(g2);
g2.dispose();
Image img = Image.getInstance(pdfTemplate);
img.scaleToFit(575, 822);
img.setAbsolutePosition(10, 10);
document.add(img);

```

As you can see, we scale the image so that it measures 20 user units less in width (and height) than the page. That's because your code indicates that you want a margin of 10 user units for the image. Now we are sure that the image will fit the page (and even leave a margin of at least 10 points).

We define the absolute position at (10, 10) which were the coordinates you used in your addTemplate() method and we add the image to the document.

Don't worry: wrapping a PdfTemplate inside an Image object will not rasterize the image. If the template contained vector data, the vector data will be preserved.

## How to set the page size to Envelope size with Landscape orientation?



I create a PDF document using iTextSharp and this code:

```
Document pdfDoc = new Document(PageSize.A4.Rotate(), 10f, 10f, 100f, 0f);
```

I googled but I couldn't find the Envelope size. How do I set the page size as Envelope with Landscape orientation?

Posted on StackOverflow on Sep 17, 2014<sup>66</sup> by King\_Fisher<sup>67</sup>

You are creating an A4 document in landscape format with this line:

```
Document pdfDoc = new Document(PageSize.A4.Rotate(), 10f, 10f, 100f, 0f);
```

If you want to create a document in envelope format, you shouldn't create an A4 page, instead you should do this:

---

<sup>66</sup><http://stackoverflow.com/questions/25886909/how-do-i-set-page-size-as-envelope-landscape-in-itextsharp>

<sup>67</sup><http://stackoverflow.com/users/3141617/king-fisher>

```
Document pdfDoc = new Document(envelope, 10f, 10f, 100f, 0f);
```

In this line, envelope is an object of type Rectangle.

There is no such thing as **the** envelope size. There are different envelope sizes to choose from. Take a look at the [envelope size chart<sup>68</sup>](#).

For instance, if you want to create a page with the size of a [6-1/4 Commercial Envelope<sup>69</sup>](#), then you need to create a rectangle that measures 6 by 3.5 inch. The measurement system in PDF doesn't use inches, but user units. By default, 1 user unit = 1 point, and 1 inch = 72 points.

Hence you'd define the envelope variable like this:

```
Rectangle envelope = new Rectangle(432, 252);
```

Because:

6 inch x 72 points = 432 points (the width)  
3.5 inch x 252 points = 252 points (the height)

If you want a different envelope type, you have to do the Math with the dimensions of that envelope format.

## How to use the full size of a page?



I managed to set the size of the PDF document I'm creating to the size I need (~3cm x ~7cm), but the content inside the page is only using a third of the space. I need to use the whole available space.

Posted on StackOverflow on [Apr 17, 2015<sup>70</sup>](#) by [user1913644<sup>71</sup>](#)

You currently have something like:

```
Rectangle rect = new Rectangle(85, 200);
Document document = new Document(rect);
```

Try this instead:

<sup>68</sup><http://www.paper-papers.com/envelope-size-chart.html>

<sup>69</sup><http://www.paper-papers.com/6-14-Commercial-Envelopes-24lb-WHITE-WOVE-35-x-6.html>

<sup>70</sup><http://stackoverflow.com/questions/29711104/fit-content-on-pdf-size-with-itextsharp>

<sup>71</sup><http://stackoverflow.com/users/1913644/user1913644>

```
Rectangle rect = new Rectangle(85, 200);
Document document = new Document(rect, 0, 0, 0, 0);
```

The 0 values are the size of the margins (left, right, top, bottom). If you omit them, they are 36 user units by default (on either side). If your rectangle measures 85 by 200 user units, using the default margins leaves you 13 by 128 user units to add content. This is somewhat consistent with what you experience.

## How to create a document with unequal page sizes?



I want to create a PDF file that has unequal page sizes. I have these two rectangles:

```
Rectangle one = new Rectangle(70,140); Rectangle two = new Rectangle(700,400);
```

I am creating the PDF like this:

```
Document document = new Document();
PdfWriter writer= PdfWriter.getInstance(
    document, new FileOutputStream("MYpdf.pdf"));
```

when I create the document, I have the option to specify the page size, but I want different page sizes for different pages in my PDF. Is that possible? E.g. the first page will have rectangle one as the page size, and the second page will have rectangle two as the page size.

Posted on StackOverflow on Apr 16, 2014<sup>72</sup> by Harvey Slash<sup>73</sup>

I've created an [UnequalPages<sup>74</sup>](#) example for you that shows how it works:

```
Document document = new Document();
PdfWriter.getInstance(document, new FileOutputStream(dest));
Rectangle one = new Rectangle(70,140);
Rectangle two = new Rectangle(700,400);
document.setPageSize(one);
document.setMargins(2, 2, 2, 2);
document.open();
Paragraph p = new Paragraph("Hi");
document.add(p);
document.setPageSize(two);
```

---

<sup>72</sup><http://stackoverflow.com/questions/23117200/itext-create-document-with-unequal-page-sizes>

<sup>73</sup><http://stackoverflow.com/users/2670775/harvey-slash>

<sup>74</sup><http://itextpdf.com/sandbox/objects/UnequalPages>

```
document.setMargins(20, 20, 20, 20);
document.newPage();
document.add(p);
document.close();
```

It is important to change the page size (and margins) *before* the page is initialized. The first page is initialized when you `open()` the document, all following pages are initialized when a `newPage()` occurs. A new page can be triggered explicitly (using the `newPage()` method in your code) or implicitly (by iText, when a page was full and a new page is needed).

## How to change the line spacing of text?



I'm creating a PDF document consisting of text only, where all the text is the same point size and font family but each character could potentially be a different color. Everything seems to work fine, but the default space between the lines is slightly greater than I consider ideal. Is there a way to control this?

Posted on StackOverflow on Feb 16, 2014 <sup>[75](#)</sup> by user2079230<sup>[76](#)</sup>

According to the PDF specification, the distance between the baseline of two lines is called the *leading*. In iText, the default leading is 1.5 times the size of the font. For instance: the default font size is 12 pt, hence the default leading is 18.

You can change the leading of a Paragraph by using a constructor that accepts a `leading` parameter. You can also change the leading using one of the methods that sets the leading. For instance:

```
paragraph.SetLeading(fixed, multiplied);
```

The first parameter is the fixed leading: if you want a leading of 15 no matter which font size is used, you can choose `fixed = 15` and `multiplied = 0`.

The second parameter is a factor: for instance if you want the leading to be twice the font size, you can choose `fixed = 0` and `multiplied = 2`. In this case, the leading for a paragraph with font size 12 will be 24, for a font size 10, it will be 20, and so on.

You can also combine fixed and multiplied leading.

---

<sup>75</sup><http://stackoverflow.com/questions/21810133/changing-text-line-spacing>

<sup>76</sup><http://stackoverflow.com/users/2079230/user2079230>

## Does anyone have any idea how TabStop works?



I'm still trying to learn iText and have a few of the concepts down. However I can't figure out what TabStop is or how to use it. My particular problem is that I want to fill the end of all paragraphs with a bunch of dashes. I believe this is called a TabStop and I see the class in the iText API but I have no clue on how to use it.

Posted on StackOverflow on Mar 26, 2014 <sup>77</sup> by Jesus Mireles<sup>78</sup>

This is a short example that uses the TabStop class:

```
java.util.List<TabStop> tabStopsList = new ArrayList<TabStop>();
tabStopsList.add(new TabStop(100, new DottedLineSeparator()));
tabStopsList.add(new TabStop(200, new LineSeparator(),
    TabStop.Alignment.CENTER));
tabStopsList.add(new TabStop(300, new DottedLineSeparator(),
    TabStop.Alignment.RIGHT));
p = new Paragraph(new Chunk("Hello world", f));
p.setTabSettings(new TabSettings(tabStopsList, 50));
addTabs(p, f, 0, "la|la");
ct.addElement(p);
```

Hope this helps.

## How to strike through text using iText?



I have 2 numbers one above the other, but the first one must have an *Strikethrough*. I'm using a table and a cell to put both numbers in the table, is there a way to meet my requirement?

Posted on StackOverflow on Feb 19, 2015 <sup>79</sup> by Ischaves<sup>80</sup>

Please take a look at the [SimpleTable6<sup>81</sup>](#) example:

<sup>77</sup><http://stackoverflow.com/questions/22650016/anyone-have-any-idea-how-tabstop-works-in-itext-5-5-0>

<sup>78</sup><http://stackoverflow.com/users/2035553/jesus-mireles>

<sup>79</sup><http://stackoverflow.com/questions/28617095/strikethrough-in-cell-using-itext-in-android-java>

<sup>80</sup><http://stackoverflow.com/users/4585768/lschaves>

<sup>81</sup><http://itextpdf.com/sandbox/tables/SimpleTable6>

0123456789	<del>0123456789</del>
0123456789	<del>0123456789</del>

Screen shot

In the first row, we strike through a number using a STRIKETHRU font as explained by Paulo:

```
Font font = new Font(FontFamily.HELVETICA, 12f, Font.STRIKETHRU);
table.addCell(new Phrase("0123456789", font));
```

In this case, iText has made a couple of decisions for you: where do I put the line? How thick is the line?

If you want to make these decisions yourself, you can use the `setUnderline()` method:

```
chunk1.setUnderline(1.5f, -1);
table.addCell(new Phrase(chunk1));
Chunk chunk2 = new Chunk("0123456789");
chunk2.setUnderline(1.5f, 3.5f);
table.addCell(new Phrase(chunk2));
```

If you pass a negative value for the y-offset parameter, the Chunk will be underlined (see first column). You can also use this method to strike through text by passing a positive y-offset.

As you can see, we also defined the thickness of the line (1.5f). There is another `setUnderline()` method that also allows you to pass the following parameters:

- *color* - the color of the line or null to follow the text color
- *thickness* - the absolute thickness of the line
- *thicknessMul* - the thickness multiplication factor with the font size
- *yPosition* - the absolute y position relative to the baseline
- *yPositionMul* - the position multiplication factor with the font size
- *cap* - the end line cap. Allowed values are PdfContentByte.LINE\_CAP\_BUTT, PdfContentByte.LINE\_CAP\_ROUND and PdfContentByte.LINE\_CAP\_PROJECTING\_SQUARE

See the [API documentation for Chunk](#)<sup>82</sup> for more info.

---

<sup>82</sup><http://api.itextpdf.com/itext/com/itextpdf/text/Chunk.html>

## How to underline text with a dotted line?



I need to merge 2 paragraphs, the first is a sequence of dots, and the second is the text that I want write on dots:

```
Paragraph pdots1 = new Paragraph(  
    ".....", font10);  
Paragraph pnote= new Paragraph("Some text on the dots", font10);
```

I tried to play with: pnote.setExtraParagraphSpace(-15); But this messes up the next paragraphs.

Posted on StackOverflow on Mar 13, 2014 <sup>83</sup> by Accollativo<sup>84</sup>

It's not a good idea to use a String with dots when you need a dotted line. It's better to use a dotted line created using the DottedLineSeparator class. See for instance the [UnderlineWithDottedLine](#)<sup>85</sup> example.

```
Paragraph p = new Paragraph("This line will be underlined with a dotted line.");  
DottedLineSeparator dottedline = new DottedLineSeparator();  
dottedline.setOffset(-2);  
dottedline.setGap(2f);  
p.add(dottedline);  
document.add(p);
```

In this example (see [underline\\_dotted.pdf](#)<sup>86</sup> for the result), I add the line 2 points under the baseline of the paragraph (using the setOffset() method) and I define a gap of 2 points between the dots (using the setGap() method).

---

<sup>83</sup><http://stackoverflow.com/questions/22382717/write-two-itext-paragraphs-on-the-same-position>

<sup>84</sup><http://stackoverflow.com/users/1878854/accollativo>

<sup>85</sup><http://itextpdf.com/sandbox/objects/UnderlineWithDottedLine>

<sup>86</sup>[http://itextpdf.com/sites/default/files/underline\\_dotted.pdf](http://itextpdf.com/sites/default/files/underline_dotted.pdf)

## How to add a full line break?



I use itextsharp and I need to draw a dotted line break from the left to the right of the page (100% width) but I don't know how to do this. The document always has a margin to the left and the right.

<b>Người Gửi</b> Công ty cổ phần Takara 30 Đường 41, KDC Tân Quý Đông, P. Tân Phong, Quận 7, TPHCM Hotline: (08) 2228 9893	<b>Người Nhận</b> vo nam 9a tu xuong Quận 3, Hồ Chí Minh ĐT: 454535434	<b>Mã Số Đơn Hàng: 3067</b> Tổng Thanh Toán: 420.000đ 26/11/2013 2:57:23 CH HCM-Giao Nhanh (24-48h trừ CN & Lễ) Cash On Delivery (COD)
<b>TAKARA.VN</b>		

**Địa Chỉ Người Mua**  
vo nam - DT: 454535434 - Email: conghanhgiang@yahoo.com  
9a tu xuong

Adobe Reader floating toolbar

```
Chunk linebreak = new Chunk(new DottedLineSeparator());
doc.Add(linebreak);
```

Posted on StackOverflow on Nov 27, 2013 <sup>87</sup> by nam vo<sup>88</sup>

Please take a look at the example [FullDottedLine](#)<sup>89</sup>.

You're creating a `DottedLineSeparator` of which the width percentage is 100% by default. This 100% is the full available width *withing the margins of the page*. If you want the line to exceed the available width, you need a percentage that is higher than 100%.

In the example, the default page size (A4) and the default margins (36) are used. This means that the width of the page is 595 user units and the available width equals 595 - (2 x 36) user units. The percentage needed to span the complete width of the page equals 100 x (595 / 523).

Take a look at the resulting PDF file [full\\_dotted\\_line.pdf](#)<sup>90</sup> and you'll see that the line now runs through the margins.

<sup>87</sup><http://stackoverflow.com/questions/20233630/itextsharp-how-to-add-a-full-line-break>

<sup>88</sup><http://stackoverflow.com/users/1712232/nam-vo>

<sup>89</sup><http://itextpdf.com/sandbox/objects/FullDottedLine>

<sup>90</sup>[http://itextpdf.com/sites/default/files/full\\_dotted\\_line.pdf](http://itextpdf.com/sites/default/files/full_dotted_line.pdf)

## How to create a custom dashed line separator?



Using the `DottedLineSeparator` class, I am able to draw dotted line separator. Similarly, is it possible to draw continuous hyphens like ‘————’ as separator in a `PdfPCell`?

Posted on StackOverflow on Jan 3, 2015 <sup>91</sup> by Retalemime<sup>92</sup>

The `LineSeparator` class can be used to draw a solid horizontal line. Either as the equivalent of the `<hr>` tag in HTML, or as a separator between two parts of text on the same line. The `DottedLineSeparator` extends the `LineSeparator` class and draws a dotted line instead of a solid line. You can define the size of the dots by changing the line width and you get a method to define the gap between the dots.

You require a dashed line and it's very easy to create your own `LineSeparator` implementation. The easiest way to do this, is by extending the `DottedLineSeparator` class like this:

```
class CustomDashedLineSeparator extends DottedLineSeparator {
    protected float dash = 5;
    protected float phase = 2.5f;

    public float getDash() {
        return dash;
    }

    public float getPhase() {
        return phase;
    }

    public void setDash(float dash) {
        this.dash = dash;
    }

    public void setPhase(float phase) {
        this.phase = phase;
    }

    public void draw(PdfContentByte canvas,
                    float llx, float lly, float urx, float ury, float y) {
        canvas.saveState();
        canvas.setLineWidth(lineWidth);
```

<sup>91</sup><http://stackoverflow.com/questions/27752409/using-itext-to-draw-separator-line-as-continuous-hyphen-in-a-table-row>

<sup>92</sup><http://stackoverflow.com/users/2338248/retalemime>

```

    canvas.setLineDash(dash, gap, phase);
    drawLine(canvas, l1x, urx, y);
    canvas.restoreState();
}
}

```

As you can see, we introduce two extra parameters, a dash value and a phase value. The dash value defines the length of the hyphen. The phase value tells iText where to start (e.g. start with half a hyphen).

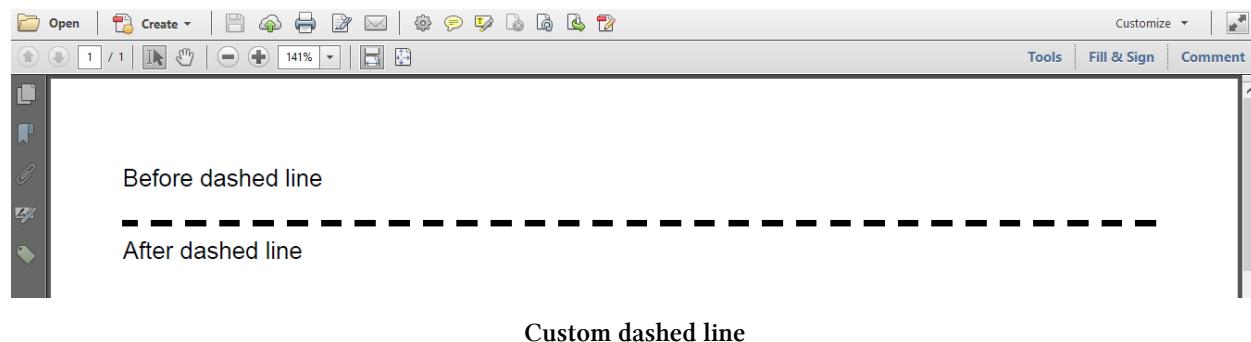
Please take a look at the [CustomDashedLine<sup>93</sup>](#) example. In this example, I use this custom implementation of the LineSeparator like this:

```

CustomDashedLineSeparator separator = new CustomDashedLineSeparator();
separator.setDash(10);
separator.setGap(7);
separator.setLineWidth(3);
Chunk linebreak = new Chunk(separator);
document.add(linebreak);

```

The result is a dashed line with hyphens of 10pt long and 3pt thick, with gaps of 7pt. The first dash is only 7.5pt long because we didn't change the phase value. In our custom implementation, we defined a phase of 2.5pt, which means that we start the hyphen of 10pt at 2.5pt, resulting in a hyphen with length 7.5pt.



You can use this custom LineSeparator in the same way you use the DottedLineSeparator, e.g. as a Chunk in a PdfPCell.

---

<sup>93</sup><http://itextpdf.com/sandbox/objects/CustomDashedLine>

## How can I generate a PDF/UA compatible PDF with iText?



We have a number of dynamically generated PDFs on our site that were created using iText 2.1.7. However, we also have a large number of users that have disabilities and use screen readers, like JAWS, to render our PDFs. We use the `setTagged()` method to tag the PDFs, but some elements of the PDF appear out of order. Some even become more jumbled after calling `setTagged()`!

I read about PDF/UA in a 2013 interview about iText with Bruno Lowagie, and this seems like something that might help with our problem. However, I have not been able to find a good example of how to generate a PDF/UA document. Can you provide an example?

Posted on StackOverflow on Jan 29, 2015 <sup>94</sup> by k-den<sup>95</sup>

Please take a look at the [PdfUA<sup>96</sup>](#) example. It explains step by step what is needed to be compliant with PDF/UA. A similar example was presented at the iText Summit in 2014 and at JavaOne. Watch the [iTExt Summit video tutorial<sup>97</sup>](#).

```
public void createPdf(String dest) throws IOException, DocumentException {
    Document document = new Document(PageSize.A4.rotate());
    PdfWriter writer =
        PdfWriter.getInstance(document, new FileOutputStream(dest));
    writer.setPdfVersion(PdfWriter.VERSION_1_7);
    //TAGGED PDF
    //Make document tagged
    writer.setTagged();
    =====
    //PDF/UA
    //Set document metadata
    writer.setViewerPreferences(PdfWriter.DisplayDocTitle);
    document.addLanguage("en-US");
    document.addTitle("English pangram");
    writer.createXmpMetadata();
    =====
    document.open();
    Paragraph p = new Paragraph();
    //PDF/UA
```

<sup>94</sup><http://stackoverflow.com/questions/28222277/how-can-i-generate-a-pdf-ua-compatible-pdf-with-itext>

<sup>95</sup><http://stackoverflow.com/users/809289/k-den>

<sup>96</sup><http://itextpdf.com/sandbox/pdfa/PdfUA>

<sup>97</sup>[https://www.youtube.com/watch?v=9b-ikCV\\_z8A](https://www.youtube.com/watch?v=9b-ikCV_z8A)

```

//Embed font
Font font =
    FontFactory.getFont(FONT, BaseFont.WINANSI, BaseFont.EMBEDDED, 20);
p.setFont(font);
=====
Chunk c = new Chunk("The quick brown ");
p.add(c);
Image i = Image.getInstance(FOX);
c = new Chunk(i, 0, -24);
//PDF/UA
//Set alt text
c.setAccessibleAttribute(PdfName.ALT, new PdfString("Fox"));
=====
p.add(c);
p.add(new Chunk(" jumps over the lazy "));
i = Image.getInstance(DOG);
c = new Chunk(i, 0, -24);
//PDF/UA
//Set alt text
c.setAccessibleAttribute(PdfName.ALT, new PdfString("Dog"));
=====
p.add(c);
document.add(p);
p = new Paragraph("\n\n\n\n\n\n\n\n\n\n\n\n", font);
document.add(p);
List list = new List(true);
list.addItem("quick", font);
list.addItem("brown", font);
list.addItem("fox", font);
list.addItem("jumps", font);
list.addItem("over", font);
list.addItem("the", font);
list.addItem("lazy", font);
list.addItem("dog", font);
document.add(list);
document.close();
}

```

You make the document tagged with the `setTagged` document, but that's not sufficient. You also need to set document data: the document title needs to be displayed and you need to indicate the language used in the document. XMP metadata is mandatory.

Furthermore you need to embed all fonts. When you have images, you need a alternate description.

In the example, we replace the words “dog” and “fox” by an image. To make sure that these images are “read out loud” correctly, we need to use the `setAccessibleAttribute()` method.

At the end of the example, I added a numbered list. In another question, you claim that the list is not read out loud correctly by JAWS. If you check the PDF file created with the above example, more specifically [pdfua.pdf](#)<sup>98</sup>, you’ll discover that JAWS reads the document as expected, with the numbers and the text in the right order.

The reason why “it doesn’t work” when you try this, is simple. You are using a version of iText that is 3 years older than the PDF/UA standard. Also: in the version you are using, you are responsible for creating the tag structure at the lowest PDF level when you use the `setTagged()` method. In more recent version, iText takes care of this at a high level. You need the latest iText version to achieve what you want.

---

<sup>98</sup><http://itextpdf.com/sites/default/files/pdfua.pdf>

# Fonts

Simple fonts, composite fonts, embedded fonts, encoding, ttf files, special characters, right to left writing systems,... This chapter is where all your questions about fonts belong.

## How to use the font Verdana in PdfStamper?



I want to use Verdana as a font while stamping a PDF file with iText PDF. The original file uses Verdana, which isn't an option in the class Basefont.

Here is the function to create my font right now:

```
def standardStampFont() {
    return BaseFont.createFont(BaseFont.HELVETICA, BaseFont.WINANSI, false)
}
```

I'd like to change that to the Verdana Font, but simply exchanging the Part BaseFont.HELVETICA with "Verdana" doesn't work.

Posted on StackOverflow on [Oct 16, 2014](#)<sup>99</sup> by [Alain Sarti](#)<sup>100</sup>

iText can support the Standard Type 1 fonts, because iText ships with AFM file (Adobe Font Metrics files). iText has no idea about the font metrics of other fonts (Verdana isn't a Standard Type 1 font). You need to provide the path to the Verdana font file.

```
BaseFont.createFont("c:/windows/fonts/verdana.ttf", BaseFont.WINANSI, BaseFont.EMBEDDED)
```

Note that I change `false` to `BaseFont.EMBEDDED` because the same problem you have on your side, will also occur on the side of the person who looks at your file: his PDF viewer can render Standard Type 1 fonts, but may not be able to render other fonts such as Verdana.

**Caveat:** The hard coded path "c:/windows/fonts/verdana.ttf" works for me on my local machine because the font file can be found using that path on my local machine. This code won't work on the server where I host the iText site, though (which is a Linux server that doesn't even have a c:/windows/fonts directory). I am using this hard coded path by way of example. You should make sure that the font is present and available when you deploy your application.

<sup>99</sup><http://stackoverflow.com/questions/26404418/how-to-use-verdana-font-in-stamper-itext-pdf>

<sup>100</sup><http://stackoverflow.com/users/1817618/alain-sarti>

## Why doesn't FontFactory.GetFont() work for all fonts?



If I say:

```
var georgia = FontFactory.GetFont("Georgia Regular", 10f);
```

it doesn't work. When I check the state of the variable `georgia`, it has its `Family` property set to the value `UNDEFINED` and its `FamilyName` property set to `Unknown`.

It only works if I actually load and register the font file and then get it like so:

```
FontFactory.Register("C:\\Windows\\Fonts\\georgia.ttf", "Georgia");
var georgia = FontFactory.GetFont("Georgia", 20f);
```

Why is that?

Posted on StackOverflow on Jun 3, 2014<sup>101</sup> by Water Cooler v2<sup>102</sup>

iText is written in Java, which means it's platform-independent. It ships with 14 AFM files containing the metrics of the 14 Standard Type 1 fonts (4 flavors of Helvetica, 4 flavors of Times Roman, 4 flavors of Courier, Symbol and ZapfDingbats).

As soon as you need other fonts, you need to register the font files by passing the path to the font directory or the path to an actual font. The font directory on Linux is different from the font directory on Windows (there is no "C:/Windows/fonts" on Linux). There's also a method `registerDirectories()` that looks at the operating system you're currently using and that registers all the 'usual suspects' (iText guesses the font path based on the OS). This method is expensive: it registers *all* fonts it finds and this costs time and memory.

Once fonts are registered, you can ask the `FontFactory` for the registered names. This is shown in the `FontFactoryExample`<sup>103</sup>. You'll notice the difference between the `getRegisteredFonts()` method and the `getRegisteredFamilies()` method.

**Additional note:** the original question is about iTextSharp, written in C#. iTextSharp is ported from Java and tries to stay as close as possible to the original version written in Java. Nevertheless, the same rationale applies: starting up an application would be much slower if iTextSharp would have to scan the fonts directory. In most applications, you only need a handful of fonts; registering all fonts available in the Windows fonts directory would be overkill.

<sup>101</sup><http://stackoverflow.com/questions/24007492/why-doesnt-fontfactory-getfontknown-font-name-floatsize-work>

<sup>102</sup><http://stackoverflow.com/users/303685/water-cooler-v2>

<sup>103</sup><http://itextpdf.com/examples/iiapdf?id=212>

## Why aren't my fonts getting registered?



I have a program using iTextSharp that includes the code

```
FontFactory.RegisterDirectories();
foreach (string fontname in FontFactory.RegisteredFonts) {
    Log.Info("**** Found registered font: " + fontname);
}
```

When I run it (using Mono on a CentOS box), the log shows only the core PostScript fonts:

- zapfdingbats
- times-roman
- times-italic
- helvetica-boldoblique
- courier-boldoblique
- helvetica-bold
- helvetica
- courier-oblique
- helvetica-oblique
- courier-bold
- times-bolditalic
- courier
- times-bold
- symbol

But I have 156 TTF files under my `/usr/share/fonts` directory tree (which is one of the directories mentioned in the code for the `RegisterDirectories` function). Why aren't these being registered?

Posted on StackOverflow on [Nov 29, 2013](#) <sup>104</sup> by [dan04](#)<sup>105</sup>

There are subtle differences between iText and iTextSharp.

In iText, `registerDirectories()` looks like this:

---

<sup>104</sup><http://stackoverflow.com/questions/13635212/why-arent-my-fonts-getting-registered>

<sup>105</sup><http://stackoverflow.com/users/287586/dan04>

```
public int registerDirectories() {
    int count = 0;
    String windir = System.getenv("windir");
    String fileseparator = System.getProperty("file.separator");
    if (windir != null && fileseparator != null) {
        count += registerDirectory(windir + fileseparator + "fonts");
    }
    count += registerDirectory("/usr/share/X11/fonts", true);
    count += registerDirectory("/usr/X/lib/X11/fonts", true);
    count += registerDirectory("/usr/openwin/lib/X11/fonts", true);
    count += registerDirectory("/usr/share/fonts", true);
    count += registerDirectory("/usr/X11R6/lib/X11/fonts", true);
    count += registerDirectory("/Library/Fonts");
    count += registerDirectory("/System/Library/Fonts");
    return count;
}
```

In iTextSharp however, the method looks like this:

```
public virtual int RegisterDirectories() {
    string dir = Path.Combine(
        Path.GetDirectoryName(Environment.GetFolderPath(
            Environment.SpecialFolder.System)), "Fonts");
    return RegisterDirectory(dir);
}
```

Java is platform independent, so we have to look for the ‘usual suspects’. C# is Windows specific, so we can depend on the environment to tell us where to find fonts. Your question tells us that Mono doesn’t support this, so you’ll have to use `FontFactory.RegisterDirectory("/usr/share/fonts")`.

## How to use Cyrillic characters in a PDF?



I have a problem when adding characters such as “Č” or “Ć” while generating a PDF. I’m mostly using paragraphs for inserting some static text into my PDF report. Here is some sample code I used:

```
var document = new Document();
document.Open();
Paragraph p1 = new Paragraph("Testing of letters Č,Ć,Š,Ž,Đ",
    new Font(Font.FontFamily.HELVETICA, 10));
document.Add(p1);
```

The output I get when the PDF file is generated, looks like this: “Testing of letters „Š,Ž,Đ” For some reason iTextSharp doesn’t seem to recognize these letters such as “Č” and “Ć”.

Posted on StackOverflow on Oct 29, 2014 <sup>106</sup> by perkes456<sup>107</sup>

### THE PROBLEM:

First of all, you don’t seem to be talking about Cyrillic characters, but about central and eastern European languages that use Latin script. Take a look at the difference between [code page 1250](#)<sup>108</sup> and [code page 1251](#)<sup>109</sup> to understand what I mean.

Second observation. You are writing code that contains special characters:

"Testing of letters Č,Ć,Š,Ž,Đ"

That is a bad practice. Code files are stored as plain text and can be saved using different encodings. An accidental switch from encoding (for instance: by uploading it to a versioning system that uses a different encoding), can seriously damage the content of your file. You should write code that doesn’t contain special characters, but that use a different notations. For instance:

"Testing of letters \u010c,\u0106,\u0160,\u017d,\u0110"

This will also make sure that the content doesn’t get altered when compiling the code using a compiler that expects a different encoding.

---

<sup>106</sup><http://stackoverflow.com/questions/26631815/cant-get-czech-characters-while-generating-a-pdf>

<sup>107</sup><http://stackoverflow.com/users/4008951/perkes456>

<sup>108</sup><http://en.wikipedia.org/wiki/Windows-1250>

<sup>109</sup><http://en.wikipedia.org/wiki/Windows-1251>

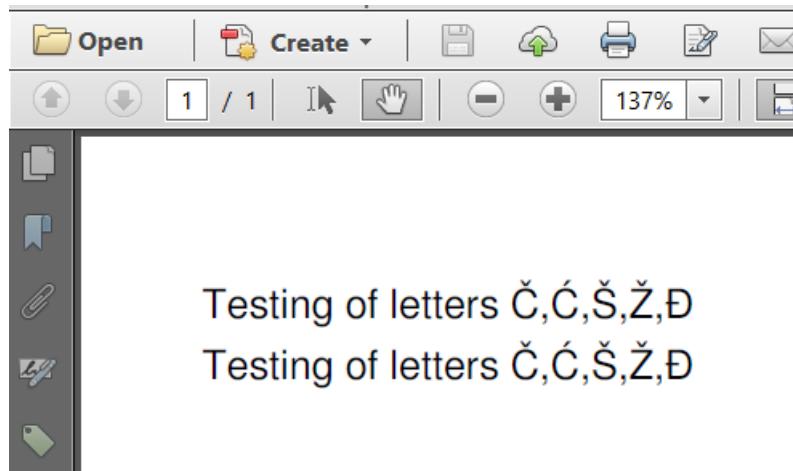
Your **third** mistake is that you assume that Helvetica is a font that knows how to draw these glyphs. That is a false assumption. You should use a font file such as Arial.ttf (or pick any other font that knows how to draw those glyphs).

Your **fourth** mistake is that you do not embed the font. Suppose that you use a font you have on your local machine and that is able to draw the special glyphs, then you will be able to read the text on your local machine. However, somebody who receives your file, but doesn't have the font you used on his local machine may not be able to read the document correctly.

Your **fifth** mistake is that you didn't define an encoding when using the font (this is related to your second mistake, but it's different).

### THE SOLUTION:

I have written a small example called [CzechExample<sup>110</sup>](#) that results in the following PDF: [czech.pdf<sup>111</sup>](#)



Czech characters in PDF

I have added the same text twice, but using a different encoding:

```
public static final String FONT = "resources/fonts/FreeSans.ttf";
public void createPdf(String dest) throws IOException, DocumentException {
    Document document = new Document();
    PdfWriter.getInstance(document, new FileOutputStream(DEST));
    document.open();
    Font f1 = FontFactory.getFont(FONT, "Cp1250", true);
    Paragraph p1 = new Paragraph(
        "Testing of letters \u010c,\u0106,\u0160,\u017d,\u0110", f1);
    document.add(p1);
    Font f2 = FontFactory.getFont(FONT, BaseFont.IDENTITY_H, true);
    Paragraph p2 = new Paragraph(
        "Testing of letters \u010c,\u0106,\u0160,\u017d,\u0110", f2);
    document.add(p2);
}
```

<sup>110</sup><http://itextpdf.com/sandbox/fonts/CzechExample>

<sup>111</sup><http://itextpdf.com/sites/default/files/czech.pdf>

```

    "Testing of letters \u010c,\u0106,\u0160,\u017d,\u0110", f2);
document.add(p2);
document.close();
}

```

To avoid your third mistake, I used the font FreeSans.ttf instead of Helvetica. You can choose any other font as long as it supports the characters you want to use.

To avoid your fourth mistake, I have set the `embedded` parameter to `true`.

As for your fifth mistake, I introduced two different approaches.

**In the first case**, I told iText to use code page 1250.

```
Font f1 = FontFactory.getFont(FONT, "Cp1250", true);
```

This will embed the font as a *simple font* into the PDF, meaning that each character in your `String` will be represented using *a single byte*. The advantage of this approach is simplicity; the disadvantage is that you shouldn't start mixing code pages. For instance: this line won't work for Cyrillic glyphs because you need "Cp1251" for Cyrillic.

**In the second case**, I told iText to use Unicode for horizontal writing:

```
Font f2 = FontFactory.getFont(FONT, BaseFont.IDENTITY_H, true);
```

This will embed the font as a *composite font* into the PDF, meaning that each character in your `String` will be represented using *more than one byte*. The advantage of this approach is that it is the recommended approach in the newer PDF standards (e.g. PDF/A, PDF/UA), and that you can mix Cyrillic with Latin, Chinese with Japanese, etc... The disadvantage is that you create more bytes, but that effect is limited by the fact that content streams are compressed anyway.

When I decompress the content stream for the text in the sample PDF, I see the following PDF syntax:

```

q
BT
36 806 Td
0 -18 Td
/F1 12 Tf
(Testing of letters Ě,Æ,Š, ,Đ) TJ
0 0 Td
0 -18 Td
/F2 12 Tf
( 9 J X Y N S L T K Q J Y Y J W X Đ Ě $ A Ō) TJ
0 0 Td
ET
Q

```

PDF syntax for Czech characters

As I explained, single bytes are used to store the text of the first line. Double bytes are used to store the text of the second line. You may be surprised that these characters look OK on the outside (when looking at the text in Adobe Reader), but don't correspond with what you see on the inside (when looking at the second screen shot), but that's how it works.

## CONCLUSION:

Many people think that creating PDF is trivial, and that tools for creating PDF should be a commodity. In reality, it's not always that simple ;-)

# Why is iText embedding a font even when I specify not to embed?



I am using Noto fonts to create a PDF and evaluating *embedding* versus *not embedding*.  
This is my code:

```
FontFactory.register("c:/temp/fonts/NotoSansCJKsc-Regular.otf", "my_nato_font");
Font myBoldFont = FontFactory.getFont("my_nato_font", BaseFont.IDENTITY_H, BaseFont.NOT_EMBEDDED);
```

When I create the pdf and do a CTRL + D, I can see that the fonts have been embedded.

However once I go with option

```
FontFactory.register("c:/temp/fonts/NotoSansCJKsc-Regular.otf", "my_nato_font");
Font myBoldFont = FontFactory.getFont("my_nato_font");
```

The size of file is reduced and the fonts are not embedded. Now I cannot see the Chinese characters which I have added to the PDF.

My questions

1. Why does NOT\_EMBEDDED option still embed the font ?
2. Since Noto fonts are open sourced by google and supported by adobe (see [Introducing Source Hans<sup>112</sup>](#)), I would assume that end user should be able to view the documents even with out the need to embed them. Is my understanding wrong?

Posted on StackOverflow on Mar 26, 2015 <sup>113</sup> by vsingh<sup>114</sup>

---

<sup>112</sup><http://blog.typekit.com/2014/07/15/introducing-source-han-sans/>

<sup>113</sup><http://stackoverflow.com/questions/29281505/why-is-itext-embedding-font-even-if-i-have-specified-not-to-embed>

<sup>114</sup><http://stackoverflow.com/users/156522/vsingh>

You are using Identity-H. In that case, the font shall be embedded. If the embedded parameter wouldn't be ignored, iText would be creating a PDF that is in violation with ISO-32000-1:

#### Section 9.7.5.2:

The Identity-H and Identity-V CMaps shall not be used with a non-embedded font.

By ignoring your NOT\_EMBEDDED setting, iText is protecting you from creating a PDF that isn't correct.

## How to identify a single font that can print all the characters in a string?



I have a string containing characters from different languages, for which I'd like to pick a single font (among the registered fonts) that has glyphs for all these characters. I would like to avoid a situation where different substrings in the string are printed using different fonts, if I already have one font that can display all these glyphs.

If there's no such single font, I would still like to pick a minimal set of fonts that covers the characters in my string. I'm aware of FontSelector, but it doesn't seem to try to find a minimal set of fonts for the given text.

Posted on StackOverflow on Feb 25, 2014 <sup>115</sup> by Kartick Vaddadi<sup>116</sup>

I see two questions in one:

**Is there a font that contains all characters for all languages?**

Allow me to explain why this is impossible:

- There are 1,114,112 code points in Unicode. Not all of these code points are used, but the possible number of different glyphs is huge.
- A simple font only contains 256 characters (1 byte per font), a composite font uses CIDs from 0 to 65,535.

65,535 is much smaller than 1,114,112, which means that it is technically impossible to have a single font that contains all possible glyphs.

**FontSelector doesn't find a minimal set of fonts!**

FontSelector doesn't look for a minimal set of fonts. You have to tell FontSelector which fonts you want to use and *in which order!* Suppose that you have this code:

---

<sup>115</sup><http://stackoverflow.com/questions/22005484/itext-how-do-i-identify-a-single-font-that-can-print-all-the-characters-in-a>

<sup>116</sup><http://stackoverflow.com/users/2893711/kartick-vaddadi>

```
FontSelector selector = new FontSelector();
selector.addFont(font1);
selector.addFont(font2);
selector.addFont(font3);
```

In this case, `FontSelector` will first look at `font1` for each specific glyph. If it's not there, it will look at `font2`, etc... Obviously `font1`, `font2` and `font3` will have different glyphs for the same character in common. For instance: `a`, `á` and `ä`. Which glyph will be used depends on the order in which you added the font.

**Bottom line:**

Select a wide range of fonts that cover all the glyphs you need and add them to a `FontSelector` instance. Don't expect to find one single font that contains all the glyphs you need.

## How to change the font color and size when using `FontSelector`?



I am using iText (Java) to write PDF which may contain Chinese characters. So I am using font selector to process the String and this works fine.

Now the problem is that if there are 2 strings

```
String str1 = "Hello Test1";
String str2 = "Hello Test2";
```

I need to write `str1` with `Font Color = Blue` and `size = 10`, whereas `str2` with `Font Color = Gray` and `size = 25`.

I am not able to figure out how to achieve this using `FontSelector`.

Posted on StackOverflow on Dec 13, 2012 <sup>[117](#)</sup> by user1900499<sup>[118](#)</sup>

Here you have a code snippet that adds the Times Roman text in Blue and the Chinese text in Red:

---

<sup>117</sup><http://stackoverflow.com/questions/13857273/itext-changing-the-font-color-and-size-when-using-fontselector>

<sup>118</sup><http://stackoverflow.com/users/1900499/user1900499>

```
FontSelector selector = new FontSelector();
Font f1 = FontFactory.getFont(FontFactory.TIMES_ROMAN, 12);
f1.setColor(BaseColor.BLUE);
Font f2 = FontFactory.getFont("MSung-Light",
    "UniCNS-UCS2-H", BaseFont.NOT_EMBEDDED);
f2.setColor(BaseColor.RED);
selector.addFont(f1);
selector.addFont(f2);
Phrase ph = selector.process(TEXT);
```

In your case you need two FontSelectors.

```
FontSelector selector1 = new FontSelector();
Font f1 = FontFactory.getFont(FontFactory.TIMES_ROMAN, 12);
f1.setColor(BaseColor.BLUE);
selector1.addFont(f1);
Phrase ph = selector1.process(str1);
FontSelector selector2 = new FontSelector();
Font f2 = FontFactory.getFont(FontFactory.TIMES_ROMAN, 12);
f2.setColor(BaseColor.GRAY);
selector2.addFont(f2);
Phrase ph = selector2.process(str2);
```

## How to create Persian content in PDF?



i have problem with UNICODE character showing in PDF file. There is a solution for this that it is not very efficient for me. The solution is like this.

```
document.add(new Paragraph("Unicode: \u0418", new Font(bfComic, 12)));
```

I want to retrieve data from database and show them to the user and my character is in Arabic language and some times in Farsi language. what solution do you suggest?

Posted on StackOverflow on Nov 8, 2014 <sup>[119](#)</sup> by gjmkdytthujk<sup>[120](#)</sup>

You are experiencing different problems:

### Encoding of the data:

You need to make sure that you retrieve data from the database using the correct encoding. For instance:

---

<sup>119</sup><http://stackoverflow.com/questions/26818555/how-to-create-persian-content-in-pdf-using-eclipse>

<sup>120</sup><http://stackoverflow.com/users/3932695/gjmkdytthujk>

```
String name1 = new String(rs.getBytes("given_name"), "UTF-8");
```

I use “UTF-8” in this example, because my database contains different names with special characters stored as “UTF-8”. I risk that these special characters are displayed as gibberish if I would retrieve the field like this:

```
String name2 = rs.getString("given_name")
```

### Encoding of the font:

You create your font like this:

```
Font font = new Font(bfComic, 12);
```

You don't show how you create bfComic, but I assume that this object is a BaseFont object using IDENTITY\_H as encoding.

### Writing from right to left / making ligatures

Although your code will work to show a single character, it won't work to show a sentence correctly.

Suppose that name1 is the Arabic version of the name “Lawrence of Arabia” and that we want to write this name to a PDF. This is done three times in the following screen shot:

Wrong: سنارول برع  
Wrong: رب ال س ان وران ل: راب  
العرب لورانس

Arabic text

The first line is wrong, because the characters are in the wrong order. They are written from left to right whereas they should be written from right to left. This is what will happen when you do:

```
document.add(name1);
```

Even if the encoding is correct, you're rendering the text incorrectly.

The second line is also wrong. The characters are now in the correct order, but no ligatures are made: ة followed by ة should be combined into a single glyph: ةة

You can only achieve this by adding the content to a ColumnText or PdfPCell object, and by setting the run direction to PdfWriter.RUN\_DIRECTION\_RTL. For instance:

```
pdfCell.setRunDirection(PdfWriter.RUN_DIRECTION_RTL);
```

Now the text will be rendered correctly.

This is explained in chapter 11 of my book. You can find a full example here: [Ligatures2<sup>121</sup>](#)

## How to solve an UnsupportedCharsetException when using iText-asian.jar?



My Code:

```
public static final String[] tempString =
    { "KozMinPro-Regular.otf", "UniJIS-UCS2-H", pharseString };
bf = BaseFont.createFont(tempString[0], tempString[1], BaseFont.NOT_EMBEDDED);
```

The exception:

```
java.nio.charset.UnsupportedCharsetException: UniJIS-UCS2-H
at java.nio.charset.Charset.forName(Unknown Source)
at com.itextpdf.text.pdf.PdfEncodings.convertToBytes(PdfEncodings.java:186)
at com.itextpdf.text.pdf.TrueTypeFont.<init>(TrueTypeFont.java:376)
at com.itextpdf.text.pdf.BaseFont.createFont(BaseFont.java:705)
at com.itextpdf.text.pdf.BaseFont.createFont(BaseFont.java:621)
at com.itextpdf.text.pdf.BaseFont.createFont(BaseFont.java:456)
at de.vogella.itext.write.Main.addTextJapanese(Main.java:145)
at de.vogella.itext.write.Main.addContent(Main.java:134)
at de.vogella.itext.write.Main.main(Main.java:254)
```

Posted on StackOverflow on Sep 14, 2014 <sup>122</sup> by Han Kun<sup>123</sup>

The cause of the exception that is thrown can be found in the following lines:

```
public static final String[] tempString =
    { "KozMinPro-Regular.otf", "UniJIS-UCS2-H", pharseString };
bf = BaseFont.createFont(tempString[0], tempString[1], BaseFont.NOT_EMBEDDED);
```

---

<sup>121</sup><http://itextpdf.com/examples/ii.php?id=210>

<sup>122</sup><http://stackoverflow.com/questions/25836370/how-to-use-itext-asian-library-in-eclipse>

<sup>123</sup><http://stackoverflow.com/users/4040263/han-kun>

Either you have a font program named `KozMinPro-Regular.otf`, or you want to use the font `KozMinPro-Regular`.

If you have a file named `KozMinPro-Regular.otf`, you don't need the `iText-Asian.jar`. Just use the font file with an encoding that is supported by that font program. `UniJIS-UCS2-H` is not supported by that OpenType font.

If you want to use CJK fonts (the fonts that are not embedded and require a font pack in Adobe Reader), you should use `KozMinPro-Regular` (without the `.otf`).

## How to choose the optimal size for a font?



Given a certain available width, how do I define the font size for a snippet of text, so that it matches the available width.

Posted on StackOverflow on Apr 12, 2013 <sup>[124](#)</sup> by Mark Edwards<sup>[125](#)</sup>

First you need a `BaseFont` object. For instance:

```
BaseFont bf = BaseFont.createFont(  
    BaseFont.COURIER, BaseFont.WINANSI, BaseFont.NOT_EMBEDDED);
```

Now you can ask the base font for the width of this String in ‘normalized 1000 units’ (these are units used in ‘Glyph space’; see ISO-32000-1 for more info):

```
float glyphWidth = bf.getWidth("WHAT IS THE WIDTH OF THIS STRING?");
```

Now we can convert these ‘normalized 1000 units’ to an actual size in points (actually user units, but let’s assume that 1 user unit = 1 pt for the sake of simplicity).

For instance: the width of the text “WHAT IS THE WIDTH OF THIS STRING?” when using Courier with size 16pt is:

```
float width = glyphWidth * 0.001f * 16f;
```

Your question is different: you want to know the font size for a given width. That’s done like this:

---

<sup>124</sup><http://stackoverflow.com/questions/15966472/choosing-optimal-font-in-the-itextpdf-table>

<sup>125</sup><http://stackoverflow.com/users/2273480/mark-edwards>

```
float fontSize = 1000 * width / glyphwidth;
```

Note that there are short cuts to get the width of a String in points: you can use `BaseFont.getWidthPoint(String text, float fontSize)` to get the width of `text` given a certain `fontSize`. Or, you can put the string in a `Chunk` and do `chunk.getWidthPoint()` (if you didn't define a `Font` for the `Chunk`, the default font Helvetica with the default font size 12 will be used).

## How to restrict the number of characters on a single line?



I am using iText to generate PDFs. In these PDFs, I need to add two Strings on one line.

The first String (`string1`) length is between 1 and 10. The length of the second String (`string2`) is unknown, but the combined length of `string1` and `string2` should not exceed 10 characters.

How can I add these strings to a single line that is underlined?

Posted on StackOverflow on Dec 22, 2013 <sup>[126](#)</sup> by Rajkumar Chilukuri<sup>[127](#)</sup>

Please take a look at the [UnderlineParagraphWithTwoParts](#)<sup>[128](#)</sup> example and allow me to explain which problems are solved in this example.

**First problem:** you want to make sure that 100 characters fit on one line. I've made this 101 characters because I assume that you want some space between `string1` and `string2` (if not, it should be easy to adapt the example).

As you don't know the content of `string1` and `string2` in advance, I've chosen a font of which all the glyphs have the same width: Courier (this is a *fixed width* or *monospaced* font). If you want to use a proportional font (such as Arial), you will have a very hard time calculating the font size, in the sense that you will have to calculate the font size separately for every `string1` and `string2` combination. This will result in a very odd-looking document where each line has a different font size.

This is the code to calculate the font size, based on the width of a single character in the font COURIER, the fact that we want to add 101 characters on one line that has a width equal to the space available between the right and left margin of the page:

---

<sup>126</sup><http://stackoverflow.com/questions/27591313/how-to-restrict-number-of-characters-in-a-phrase-in-pdf>

<sup>127</sup><http://stackoverflow.com/users/4192690/rajkumar-chilukuri>

<sup>128</sup><http://itextpdf.com/sandbox/objects/UnderlineParagraphWithTwoParts>

```

BaseFont bf = BaseFont.createFont(
    BaseFont.COURIER, BaseFont.WINANSI, BaseFont.NOT_EMBEDDED);
float charWidth = bf.getWidth(" ");
int charactersPerLine = 101;
float pageWidth = document.right() - document.left();
float fontSize = (1000 * (pageWidth / (charWidth * charactersPerLine)));
fontSize = ((int)(fontSize * 100)) / 100f;
Font font = new Font(bf, fontSize);

```

Note that I round the value of the float. If you don't do this, you may experience problems due to rounding errors inherent to using float values.

**Problem2:** Now we want to add two string to a single line and underline them. From your question, it isn't clear how you want to align these strings.

This is the simple situation, where `string1` and `string2` are separated by a single space:

```

public void addParagraphWithTwoParts2(
    Document document, Font font, String string1, String string2)
throws DocumentException {
    if (string1 == null) string1 = "";
    if (string1.length() > 10)
        string1 = string1.substring(0, 10);
    if (string2 == null) string2 = "";
    if (string1.length() + string2.length() > 100)
        string2 = string2.substring(0, 100 - string1.length());
    Paragraph p = new Paragraph(string1 + " " + string2, font);
    LineSeparator line = new LineSeparator();
    line.setOffset(-2);
    p.add(line);
    document.add(p);
}

```

This is a more complex situation, where you right-align `string2`:

```
public void addParagraphWithTwoParts1(
    Document document, Font font, String string1, String string2)
throws DocumentException {
    if (string1 == null) string1 = "";
    if (string1.length() > 10)
        string1 = string1.substring(0, 10);
    Chunk chunk1 = new Chunk(string1, font);
    if (string2 == null) string2 = "";
    if (string1.length() + string2.length() > 100)
        string2 = string2.substring(0, 100 - string1.length());
    Chunk chunk2 = new Chunk(string2, font);
    Paragraph p = new Paragraph();
    p.add(chunk1);
    p.add(GLUE);
    p.add(chunk2);
    LineSeparator line = new LineSeparator();
    line.setOffset(-2);
    p.add(line);
    document.add(p);
}
```

As you can see, we do not have to add any spaces, we just use a GLUE chunk that is defined like this:

```
public static final Chunk GLUE = new Chunk(new VerticalPositionMark());
```

I've created an example where string1 and string2 consist of numbers. This is what the result looks like:

# Test 1

# Test 1

# Test 1

## Test 2

## Test 2

0123456789 0123456789012345678901234567890123456789012345678901234567890123456789

## Test 3

Test 3

0123456

Test 3

Test 3

012

Test 4

## Test 4

---

Test 4

212245

Test 5

Test 3  
012345

---

Test 5

Test 3

## Screen shot

In this screen shot, you see examples where string2 is right aligned as well as examples where string2 is added right after string1 (but separated from it with a single space character).

## How to calculate the height of an element?



I am generating PDF files from XML data. I calculate the height of a paragraph element like this:

```
float paraWidth = 0.0f;
for (Object o : el.getChunks()) {
    paraWidth += ((Chunk) o).getWidthPoint();
}
float paraHeight = paraWidth/PageSize.A4.getWidth();
```

But this method does not work correctly.

Posted on StackOverflow on Jul 3, 2014 <sup>129</sup> by Valeri<sup>130</sup>

Your question is strange. According to the header of your question, you want to know the **height** of a string, but your code shows that you are asking for the **width** of a String.

Please take a look at the [FoobarFilmFestival<sup>131</sup>](#) example.

If bf is a BaseFont instance, then you can use:

```
float ascent = bf.getAscentPoint("Some String", 12);
float descent = bf.getDescentPoint("Some String", 12);
```

This will return the height above the baseline and the height below the baseline, when we use a font size of 12. As you probably know, the font size is an indication of the average height. It's not the actual height. It's just a number we work with.

The total height will be:

```
float height = ascent - descent;
```

Or maybe you want to know the number of lines taken by a Paragraph and multiply that with the leading. That's a different question, though.

---

<sup>129</sup><http://stackoverflow.com/questions/24549267/calculate-element-height-itext>

<sup>130</sup><http://stackoverflow.com/users/1430463/valeri>

<sup>131</sup><http://itextpdf.com/examples/iiapdf?id=61>

## How to calculate/set font line distance?



I am implementing a PdfPageEventHelper event to add a footer as shown in this code snippet:

```
ColumnText.showTextAligned(cb, Element.ALIGN_RIGHT,
    new Phrase(String.format(" %d ", writer.getPageNumber()), footerFont),
    document.right() - 2, document.bottom() - 20, 0);
```

I need to add 3 lines, but I don't find how to calculate the distance between the lines. Each line has different font size. What should I use as `xxx` value in `document.bottom() - xxx`?

Posted on StackOverflow on Sep 17, 2013 <sup>[132](#)</sup> by Dhaval<sup>[133](#)</sup>

The difference between two lines is the `leading`. You can pick your own leading, but it is custom to use 1.5 times the font size. You are drawing line by line yourself, using different font sizes, so you'll have to adjust the Y value based on that font size. Note that `ColumnText.showTextAligned()` uses the Y value as the baseline of the text you're adding, so if you have some text with font size of 12pt, you'd need to take into account a leading of 18pt. If you have a font size 8pt, you make sure you have 12pt.

That's the easy solution: based on "convention". If you really want to know how much horizontal space some *specific* takes, you need to calculate the ascender and the descender. If `bf` is your font (a `BaseFont` object), `text` is your text (a `String`) and `size` is your font size (a `float`), then the height of your text is equal to `height`:

```
float aboveBaseline = bf.getAscentPoint(text, size);
float underBaseline = bf.getDescentPoint(text, size);
float height = aboveBaseline - underBaseline;
```

When `y` is the Y-coordinate used in `showTextAligned()` make sure you keep the space between `y + aboveBaseline` and `y + underBaseline` free. This is the *accurate* solution.

Note that `document.bottom() - 20` looks somewhat strange. I would expect `document.bottom() + 20` as the Y-axis of the PDF coordinate system points upwards, not downwards.

---

<sup>132</sup><http://stackoverflow.com/questions/18849554/how-to-calculate-set-font-line-margin-itextpdf>

<sup>133</sup><http://stackoverflow.com/users/2182021/dhaval>

## Why can't I set the font of a Phrase?



I'm starting with iTextSharp and wondering if there is any reason why setting the font of a Phrase after the construction of the object doesn't work. Is there any reason, do I miss something?

```
iTextSharp.text.Font f = PdfFontFactory.GetComic();
f.SetStyle(PdfFontStyle.BOLD);
Color c = Color.DarkRed;
f.SetColor(c.R,c.G,c.B);
f.Size = 20;
Phrase titreFormules = new Phrase("Nos formules",f); //THIS WORKS
// titreFormules.Font = f; // THIS DOESN'T WORK!
document.Add(titreFormules);
document.Close();
```

Posted on StackOverflow on Sep 11, 2014 <sup>134</sup> by Alain BUFERNE<sup>135</sup>

When you use `setFont()`, you change the font of the Phrase for all the content that is added *after* the font was set.

- A Chunk is an atomic part of text in the sense that all the text in a Chunk has the same font family, font size, font color,...
- A Phrase is a collection of Chunk objects and as such a Phrase can contain different ‘atoms’ of text using different fonts.

In your example "Nos formules" will be written in Helvetica. You change the font *after* the Helvetica Chunk with the text "Nos formules" was added to the Phrase. As you didn't add anything else to `titreFormules`, the font "Comic" is never used.

---

<sup>134</sup><http://stackoverflow.com/questions/25793034/itextsharp-setting-phrase-font>

<sup>135</sup><http://stackoverflow.com/users/487313/alain-buferne>

## How to use two different colors in a single String?



I have string like below, and i can't split the string.

```
String result = "Developed By : Mr.XXXXX";
```

I can create a paragraph in iText and set font with color like this:

```
Font dataGreenFont = FontFactory.getFont("Garamond", 10, Color.GREEN);
preface.add(new Paragraph(result, dataGreenFont));
```

It sets the green color to the entire text `result` but I want to set the color only for `Mr . XXXXX` part. How do I do this?

Posted on StackOverflow on Jun 17, 2014 <sup>[136](#)</sup> by Ramakrishna<sup>[137](#)</sup>

A Paragraph consists of a series of Chunk objects. A Chunk is an atomic part of text in which all the glyphs are in the same font, have the same font size, color, etc...

Hence you need to split your String in two parts:

```
Font dataGreenFont = FontFactory.getFont("Garamond", 10, BaseColor.GREEN);
Font dataBlackFont = FontFactory.getFont("Garamond", 10, BaseColor.BLACK);
Paragraph p = new Paragraph();
p.Add(new Chunk("Developed By : ", dataGreenFont));
p.Add(new Chunk("Mr.XXXXX", dataBlackFont));
document.add(p);
```

---

<sup>136</sup><http://stackoverflow.com/questions/24256581/how-to-set-two-different-colors-for-a-single-string-in-itext>

<sup>137</sup><http://stackoverflow.com/users/3435914/ramakrishna>

## How to apply color to Strings in a Paragraph?



I am combining 2 strings to Paragraph this way,

```
String str2="";
String str1="";
ColumnText ct = new ColumnText(cb);
ct.setSimpleColumn(36, 600, 600, 800);
ct.addElement(new Paragraph(str1 + str2));
int status1 = ct.go();
```

The problem is I am getting the same font color for both str1 & str2. I want to have different font color and size for these Strings. How Can I do that on ColumnText/Paragraph?

Posted on StackOverflow on Dec 20, 2014 <sup>138</sup> by user1660325<sup>139</sup>

When you combine text into a Paragraph like this:

```
Paragraph p = new Paragraph("abc" + "def");
```

You implicitly tell iText that "abc" and "def" should be rendered using the same (default) font. As you probably know, a Paragraph is a collection of Chunk objects. In iText, a Chunk is like an atomic part of text in the sense that all the text in a Chunk has the same font, font size, font color, etc...

If you want to create a Paragraph with different font colors, you need to compose your Paragraph using different Chunk objects. This is shown in the [ColoredText<sup>140</sup>](#) example:

```
Font red = new Font(FontFamily.HELVETICA, 12, Font.NORMAL, BaseColor.RED);
Chunk redText = new Chunk("This text is red. ", red);
Font blue = new Font(FontFamily.HELVETICA, 12, Font.BOLD, BaseColor.BLUE);
Chunk blueText = new Chunk("This text is blue and bold. ", blue);
Font green = new Font(FontFamily.HELVETICA, 12, Font.ITALIC, BaseColor.GREEN);
Chunk greenText = new Chunk("This text is green and italic. ", green);
Paragraph p1 = new Paragraph(redText);
document.add(p1);
Paragraph p2 = new Paragraph();
p2.add(blueText);
p2.add(greenText);
document.add(p2);
```

---

<sup>138</sup><http://stackoverflow.com/questions/27578497/applying-color-to-strings-in-paragraph-using-itext>

<sup>139</sup><http://stackoverflow.com/users/1660325/user1660325>

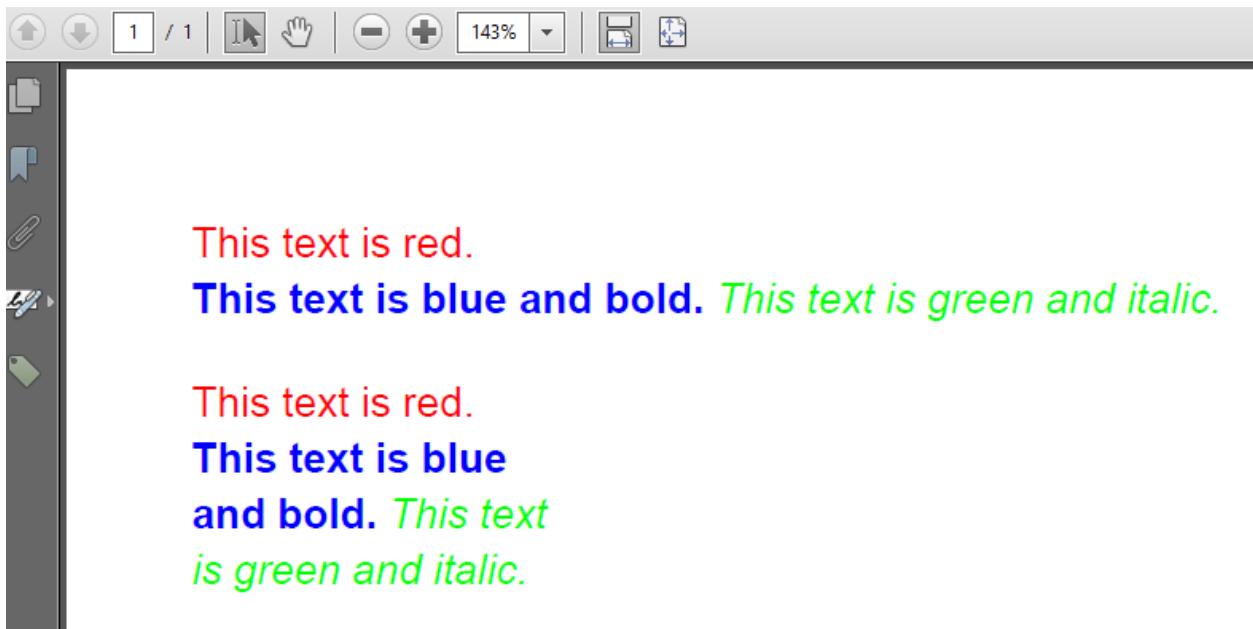
<sup>140</sup><http://itextpdf.com/sandbox/objects/ColoredText>

In this example, we create two paragraphs. One with a single Chunk in red. Another one that contains two Chunks with a different color.

In your question, you refer to `ColumnText`. The next code snippet uses `p1` and `p2` in a `ColumnText` context:

```
ColumnText ct = new ColumnText(writer.getDirectContent());
ct.setSimpleColumn(new Rectangle(36, 600, 144, 760));
ct.addElement(p1);
ct.addElement(p2);
ct.go();
```

As a result, the paragraphs are added twice: once positioned by iText, once positioned by ourselves by defining coordinates using a `Rectangle`:



Screen shot

## How to introduce a custom font weight?



Is there a way to set font weight in iTextSharp as a number? Now I can set bold font like this:

```
iTextSharp.text.Font font_main = new iTextSharp.text.Font(someBaseFont, 12, iTextSharp.text.Font.BOLD);
```

So, only one “boldness” is available. But I want something like this:

```
font_main.weight = 600;
```

Is it possible?

Posted on StackOverflow on Nov 10, 2014 <sup>[141](#)</sup> by [retif](#)<sup>[142](#)</sup>

When you pick a font family, for instance Arial, you also need to pick a font, e.g. Arial regular (`arial.ttf`), Arial Bold (`arialbd.ttf`), Arial Italic (`ariali.ttf`), Arial Bold-Italic (`arialbi.ttf`)...

When you switch styles the way you do in your code snippet, iText switches fonts. In other words: you don't use Arial regular to which you apply a style, you switch to another font in the same family: Arial Bold.

A font program such as the one stored in `arialbd.ttf`, contains the instructions to draw the outlines (aka path) of a series of glyphs. These glyphs are rendered by filling the path using a *fill color*. Knowing this is important to achieve your requirement.

You can control the boldness of your font by changing the line-width render-mode. This is especially useful in case you are working with a font family for which there is no bold version of the font available.

iText's `Chunk` object has a method called `setTextRenderMode()` that is demonstrated in the [SayPeace<sup>143</sup>](#) example. Allow me to show the corresponding method in iTextSharp:

```
Chunk bold = new Chunk("Bold", font);
bold.SetTextRenderMode(
    PdfContentByte.TEXT_RENDER_MODE_FILL_STROKE, 0.5f,
    GrayColor.GRAYBLACK
);
```

---

<sup>141</sup><http://stackoverflow.com/questions/26838812/custom-font-weight-in-itextsharp>

<sup>142</sup><http://stackoverflow.com/users/1688203/retif>

<sup>143</sup><http://itextpdf.com/examples/iaa.php?id=205>

The first method tells the `Chunk` that the glyphs should be filled (as usual) and stroked (not usual). The strokes should have a width of 0.5 user units (change this to get a different *boldness*), the final parameter defines the color of the text.

## Two extra remarks for the sake of completeness

1. There is a similar method to mimic an italic font: the `setSkew()` method. Using this method, you can introduce two angles. For instance `setSkew(0, 25)` will write the glyphs with an angle of 25 degrees which more or less corresponds with the angle used in italic fonts.
2. In the old days (a really long time ago), some PDF producers mimicked boldness by writing the same character over and over again at the same position. As a result, this character was rendered as if it were bold. This has some disadvantages: it is not documented in the ISO standard, so you can't expect that all viewers will support this in the same way. When you extract text from the PDF, you get multiple instances of the same character.

## How to make a single letter bold within a word?



Is it possible to bold out specific letters within a word in iText? Consider this sentence, "this is a word". Could I make all the "i" within that sentence bold? I want the output to be like this: "this **i**s a word"

So to get the first word right (this), do I really have to create three chunks for one word and than add all the chunks together? Or is there another approach?

With three chunks I mean something like this:

```
Chunk chunk1 = new Chunk("th");
Chunk chunk2 = new Chunk("i", bold);
Chunk chunk3 = new Chunk("s");
Paragraph p = new Paragraph();
p.add(chunk1); p.add(chunk2); p.add(chunk3)
```

There has to be a better way. I want to colorize some "strings" as well, so that each letter in the string will have a different color. Do I have to create chunks for each letter than combine them?

Posted on StackOverflow on Nov 22, 2014 <sup>144</sup> by user3711421 <sup>145</sup>

Take a look at the following example: [ColoredLetters](#)<sup>146</sup>

I've written this method that automates what you describe:

---

<sup>144</sup><http://stackoverflow.com/questions/27080565/itext-make-a-single-letter-bold-within-a-word>

<sup>145</sup><http://stackoverflow.com/users/3711421/user3711421>

<sup>146</sup><http://itextpdf.com/sandbox/objects/ColoredLetters>

```

private Chunk returnCorrectColor(char letter) {
    if (letter == 'b'){
        return B;
    }
    else if(letter == 'g'){
        return G;
    }
    return new Chunk(String.valueOf(letter), RED_NORMAL);
}

```

Note that I used some constants:

```

public static final Font RED_NORMAL =
    new Font(FontFamily.HELVETICA, 12, Font.NORMAL, BaseColor.RED);
public static final Font BLUE_BOLD =
    new Font(FontFamily.HELVETICA, 12, Font.BOLD, BaseColor.BLUE);
public static final Font GREEN_ITALIC =
    new Font(FontFamily.HELVETICA, 12, Font.ITALIC, BaseColor.GREEN);
public static final Chunk B = new Chunk("b", BLUE_BOLD);
public static final Chunk G = new Chunk("g", GREEN_ITALIC);

```

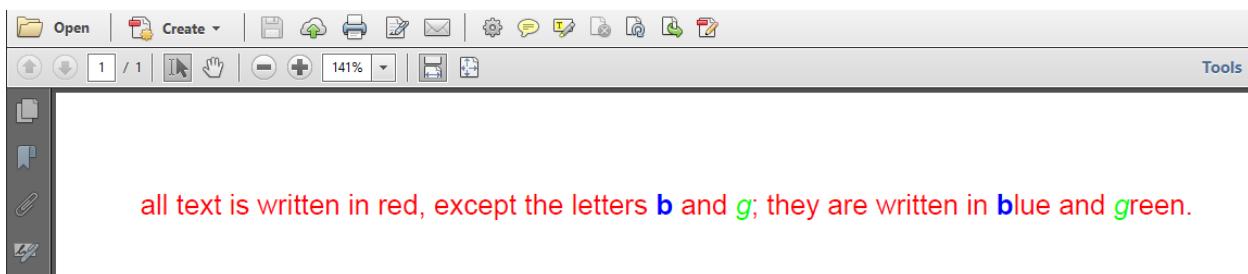
Now I can construct a Paragraph like this:

```

Paragraph p = new Paragraph();
String s = "all text is written in red, except the letters b and g; "
    + "they are written in blue and green.";
for (int i = 0; i < s.length(); i++) {
    p.add(returnCorrectColor(s.charAt(i)));
}

```

The result looks like this: [colored\\_letters.pdf](#)<sup>147</sup>



Screen shot

---

<sup>147</sup>[http://itextpdf.com/sites/default/files/colored\\_letters.pdf](http://itextpdf.com/sites/default/files/colored_letters.pdf)

An alternative would be to use different fonts, where fontB only contains the letter b and fontG only contains the letter g. You could then declare these fonts to the `FontSelector` object, along with a normal font. You could then process the font. For an example using this alternative, please read my answer to the question: "Changing the font color and size when using `FontSelector`."

### Updates based on additional questions in the comments

Could i specify the color as a HEX. for example something like `BaseColor(#ffcd03)`; I understand that you can specify it like `new BaseColor(0, 0, 255)`, however HEX would be more useful.

Actually, I like the HEX notation more myself, because it is easier for me to recognize certain colors when they are expressed in HEX notation. That's why I often use the HEX notation for integers in Java like this: `new BaseColor(0xFF, 0xCD, 0x03)`;

Additionally, there is a `decodeColor()` method in the `HtmlUtilities` class that can be used like this:

```
BaseColor color = HtmlUtilities.decodeColor("#ffcd03");
```

is there a way to just define the color and leave the font to the default. Something like this: `public static final Font RED_NORMAL = new Font(BaseColor.RED);`

You could create the font like this:

```
new Font(FontFamily.UNDEFINED, 12, Font.UNDEFINED, BaseColor.RED);
```

Now when you add a `Chunk` with such a font to a `Paragraph`, it should take the font defined at the level of the `Paragraph` (that's how I intended it when I first wrote iText. I should check if this is still true).

My goal is to write a program that colorizes existing PDF files without changing the structure of the PDF. Do you think this approach will work for this purpose (performance etc)?

No, this will not work. If you read the introduction of [chapter 6 of the second edition of iText in Action<sup>148</sup>](#), you'll understand that PDF is not a format that was created for editing. The internal syntax for the sentence shown in the screen shot looks like this:

---

<sup>148</sup><http://manning.com/lowagie2/samplechapter6.pdf>

```
BT
36 806 Td
0 -16 Td
/F1 12 Tf
1 0 0 rg
(all text is written in red, except the letters ) Tj
0 g
/F2 12 Tf
0 0 1 rg
(b) Tj
0 g
/F1 12 Tf
1 0 0 rg
( and ) Tj
0 g
/F3 12 Tf
0 1 0 rg
(g) Tj
0 g
/F1 12 Tf
1 0 0 rg
(; they are written in ) Tj
0 g
/F2 12 Tf
0 0 1 rg
(b) Tj
0 g
/F1 12 Tf
1 0 0 rg
(lue and ) Tj
0 g
/F3 12 Tf
0 1 0 rg
(g) Tj
0 g
/F1 12 Tf
1 0 0 rg
(reen.) Tj
0 g
0 0 Td
ET
```

As you can see, the Tf operator changes the font:

- /F1 is the regular font,
- /F2 is the bold font,
- /F3 is the italic font.

These names refer to fonts in the /Resources dictionary of the page, where you'll find the references to the actual font descriptors.

The rg operator changes the color. As we use red, green and blue, you recognize the operands 1 0 0, 0 1 0 and 0 0 1.

This looks fairly simple, because we used WINANSI encoding. One could parse the content stream, look for all the PDF string (between ( and ) or between < and >), and introduce Tf and rg operators.

However: the content stream won't always be as trivial as here. Custom encodings can be used, Unicode can be used,... Take a look at the second screen shot in my answer to the following question: <http://stackoverflow.com/questions/26631815/cant-get-czech-characters-while-generating-a-pdf>. You'll understand that your plan to change colors in an existing PDF is a non-trivial task. We have done such a project in the past for one of our customers. It was a project that took several weeks.

## How to introduce superscript?



I want to format a date like this: 14<sup>th</sup> may 2015. However, the **th** in 14 should be in superscript as is what happens when you use the "sup" tag.

This is my code:

```
ph102 = new Phrase(csq.EventDate.Value.Day
+ Getsuffix(csq.EventDate.Value.Day)
+ csq.EventDate.Value.ToString("MMM")
+ " " + csq.EventDate.Value.Year
+ " at " + eventTime.ToString("hh:mm tt"), textFont7);
PdfPCell cellt2 = new PdfPCell(ph102);
```

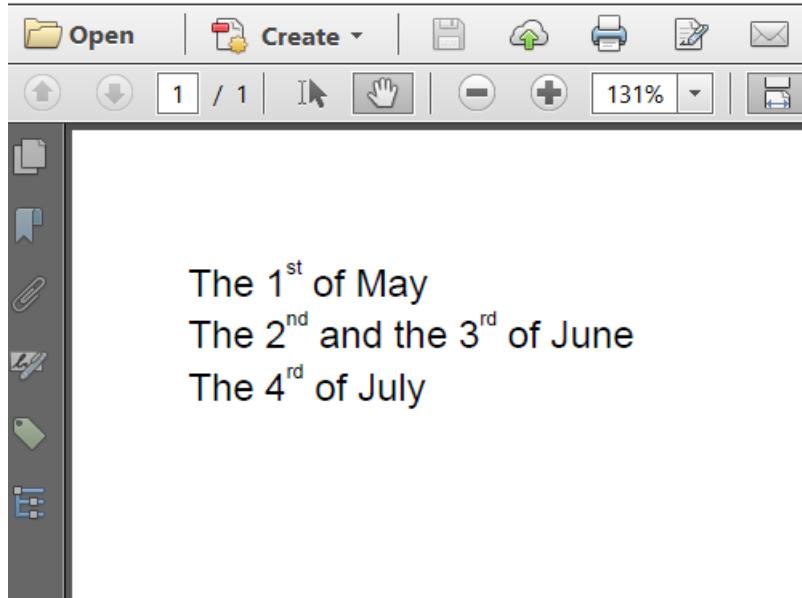
Posted on StackOverflow on May 19, 2015 <sup>149</sup> by Dhasarathan T <sup>150</sup>

When I read your question, I assume that you want something like this:

---

<sup>149</sup><http://stackoverflow.com/questions/30322665/how-to-access-the-sup-tag-in-itextsharp>

<sup>150</sup><http://stackoverflow.com/users/4002749/dhasarathan-t>



Superscript

In your code, you create a Phrase like this:

```
ph102 = new Phrase(csq.EventDate.Value.Day
    + Getsuffix(csq.EventDate.Value.Day)
    + csq.EventDate.Value.ToString("MMM")
    + " " + csq.EventDate.Value.Year
    + " at " + eventTime.ToString("hh:mm tt"), textFont7);
```

This complete Phrase is expressed in `textFont7` which can't work in your case, because you want to use a smaller font for the "st", "nd", "rd" or "th".

You need to do something like this (see [OrdinalNumbers<sup>151</sup>](#) for the full example):

```
public void createPdf(String dest) throws IOException, DocumentException {
    Document document = new Document();
    PdfWriter.getInstance(document, new FileOutputStream(dest));
    document.open();
    Font small = new Font(FontFamily.HELVETICA, 6);
    Chunk st = new Chunk("st", small);
    st.setTextRise(7);
    Chunk nd = new Chunk("nd", small);
    nd.setTextRise(7);
    Chunk rd = new Chunk("rd", small);
```

<sup>151</sup><http://itextpdf.com/sandbox/objects/OrdinalNumbers>

```
rd.setTextRise(7);
Chunk th = new Chunk("th", small);
th.setTextRise(7);
Paragraph first = new Paragraph();
first.add("The 1");
first.add(st);
first.add(" of May");
document.add(first);
Paragraph second = new Paragraph();
second.add("The 2");
second.add(nd);
second.add(" and the 3");
second.add(rd);
second.add(" of June");
document.add(second);
Paragraph fourth = new Paragraph();
fourth.add("The 4");
fourth.add(rd);
fourth.add(" of July");
document.add(fourth);
document.close();
}
```

This is the Java code to create the PDF in the screen shot. You'll have to adapt your code so that it works in C#. As you can see, you can not just concatenate your strings using the + operator. You need to compose your Phrase or Paragraph using different Chunk objects. What you call "sup" is done by using a smaller font and a text rise.

## How to allow a Unicode subscript symbol in a PDF without using setTextRise()?



I have been trying to create PDF files using iText. I have found that Superscript (0,4,5,6,7,8,9) and Subscript (0,1,2,3,4,5,6,7,8,9) are not allowed in the default font. Only Superscript 1,2,3 are allowed in the default setting. A String in my PDF contains subscript and superscript characters, is there a way I can display them in PDF?

I searched and found that:

1. the `setTextRise()` method is used to change the displacement of the text with respect to the current line, I cannot use it as I don't know which characters are to super/sub scripted.
2. Is there a setting to change the encoding/font of the text to UTF-8, so that it allows all the Unicode symbols?

Or some other way, because I have to show chemical symbols like H<sub>2</sub>SO<sup>4</sup>, O<sub>2</sub> etc.

Posted on StackOverflow on Jul 16, 2014 <sup>[152](#)</sup> by [nexus](#)<sup>[153](#)</sup>

Please take a look at the [SubSuperScript](#)<sup>[154](#)</sup> example I wrote in answer to your question. I'm writing H<sub>2</sub>SO<sup>4</sup> like this:

```
BaseFont bf = BaseFont.createFont(FONT, BaseFont.IDENTITY_H, BaseFont.EMBEDDED);
Font f = new Font(bf, 10);
Paragraph p = new Paragraph("H\u2082SO\u2074", f);
document.add(p);
```

I've found the values of the specific characters on [Wikipedia](#)<sup>[155](#)</sup>.

As you can see, iText is perfectly capable to show the special characters written in subscript or superscript: [sub\\_superscript.pdf](#)<sup>[156](#)</sup>

The main problem I had when I wrote this example was finding a font that supported the specific glyphs. That's what I explained in my earlier comment. I ended up using a font called "Carbo Regular".

---

<sup>152</sup><http://stackoverflow.com/questions/24772212/itext-pdf-writer-is-there-any-way-to-allow-unicode-subscript-symbol-in-the-pdf>

<sup>153</sup><http://stackoverflow.com/users/1441512/nexus>

<sup>154</sup><http://itextpdf.com/sandbox/objects/SubSuperScript>

<sup>155</sup>[http://en.wikipedia.org/wiki/Unicode\\_subscripts\\_and\\_superscripts](http://en.wikipedia.org/wiki/Unicode_subscripts_and_superscripts)

<sup>156</sup>[http://itextpdf.com/sites/default/files/sub\\_superscript.pdf](http://itextpdf.com/sites/default/files/sub_superscript.pdf)

## How to display indian rupee symbol?



I want to display Special Character India Rupee Symbol in iTextPDF,

My Code:

```
Font fontRupee = FontFactory.GetFont("Arial", "₹", true, 12);
Chunk chunkRupee = new Chunk(" ₹ 5410", font3);
```

Posted on StackOverflow on Jul 4, 2013 <sup>157</sup> by Avanish Singh<sup>158</sup>

It's never a good idea to store a Unicode character such as ₹ in your source code. Plenty of things can go wrong if you do so:

- Somebody can save the file using an encoding different from Unicode, for instance, the double-byte rupee character can be interpreted as two separate bytes representing two different characters.
- Even if your file is stored correctly, maybe your compiler will read it using the wrong encoding, interpreting the double-byte character as two separate characters.

From your code sample, it's evident that you're not familiar with the concept known as *encoding*. When creating a Font object, you pass the rupee symbol as encoding.

The correct way to achieve what you want looks like this:

```
BaseFont bf =
    BaseFont.CreateFont("c:/windows/fonts/arial.ttf",
        BaseFont.IDENTITY_H, BaseFont.EMBEDDED);
Font font = new Font(bf, 12);
Chunk chunkRupee = new Chunk(" ₹ 5410", font3);
```

Note that there are two possible Unicode values for the Rupee symbol (source [Wikipedia<sup>159</sup>](#)): \u20B9 is the value you're looking for; the alternative value is \u20A8 (which looks like this: Rs).

I've tested this with arialuni.ttf and arial.ttf. Surprisingly MS Arial Unicode was only able to render Rs; it couldn't render ₹. Plain arial was able to render both symbols. It's very important to check if the font you're using knows how to draw the symbol. If it doesn't, nothing will show up on your page.

<sup>157</sup><http://stackoverflow.com/questions/17465183/how-to-display-indian-rupee-symbol-in-itext-pdf-in-mvc3>

<sup>158</sup><http://stackoverflow.com/users/1637935/avinash-singh>

<sup>159</sup>[http://en.wikipedia.org/wiki/Indian\\_rupee\\_sign](http://en.wikipedia.org/wiki/Indian_rupee_sign)

## Why isn't the Rupee symbol showing?



I am using iText for creating PDF. I tried using arial.ttf and arialbd.ttf, but I don't have any luck with these to show the rupee symbol. I have read the answer to the question "How to display indian rupee symbol?", but it is not working for me.

This is the code I have used.

```
BaseFont rupee =BaseFont.createFont( "assets/arial .ttf", BaseFont.IDENTITY_H,Ba\
seFont.EMBEDDED);
createHeadings(cb,495,60,": " +edt_total.getText().toString(),12,rupee);

private void createHeadings(PdfContentByte cb, float x, float y, String text, in\
t size,BaseFont fb){
    cb.beginText();
    cb.setFontAndSize(fb, size);
    cb.setTextMatrix(x,y);
    cb.showText(text.trim());
    cb.endText();
}
```

Posted on StackOverflow on Oct 14, 2014<sup>160</sup>

The problem you describe is typical when:

1. you are using a font which doesn't have that glyph, or
2. you aren't using the right encoding.

I have written an example that illustrates this: [RupeeSymbol](#)<sup>161</sup>

```
public static final String DEST = "results/fonts/rupee.pdf";
public static final String FONT1 = "resources/fonts/PlayfairDisplay-Regular.ttf";
public static final String FONT2 = "resources/fonts/PT_Sans-Web-Regular.ttf";
public static final String FONT3 = "resources/fonts/FreeSans.ttf";
public static final String RUPEE =
    "The Rupee character \u20B9 and the Rupee symbol \u20A8";

public void createPdf(String dest) throws IOException, DocumentException {
    Document document = new Document();
```

---

<sup>160</sup><http://stackoverflow.com/questions/26360814/rupee-symbol-is-not-showing-in-android>

<sup>161</sup><http://itextpdf.com/sandbox/fonts/RupeeSymbol>

```
PdfWriter.getInstance(document, new FileOutputStream(DEST));
document.open();
Font f1 =
    FontFactory.getFont(FONT1, BaseFont.IDENTITY_H, BaseFont.EMBEDDED, 12);
Font f2 =
    FontFactory.getFont(FONT2, BaseFont.IDENTITY_H, BaseFont.EMBEDDED, 12);
Font f3 =
    FontFactory.getFont(FONT3, BaseFont.IDENTITY_H, BaseFont.EMBEDDED, 12);
Font f4 =
    FontFactory.getFont(FONT3, BaseFont.WINANSI, BaseFont.EMBEDDED, 12);
document.add(new Paragraph(RUPEE, f1));
document.add(new Paragraph(RUPEE, f2));
document.add(new Paragraph(RUPEE, f3));
document.add(new Paragraph(RUPEE, f4));
document.close();
}
```

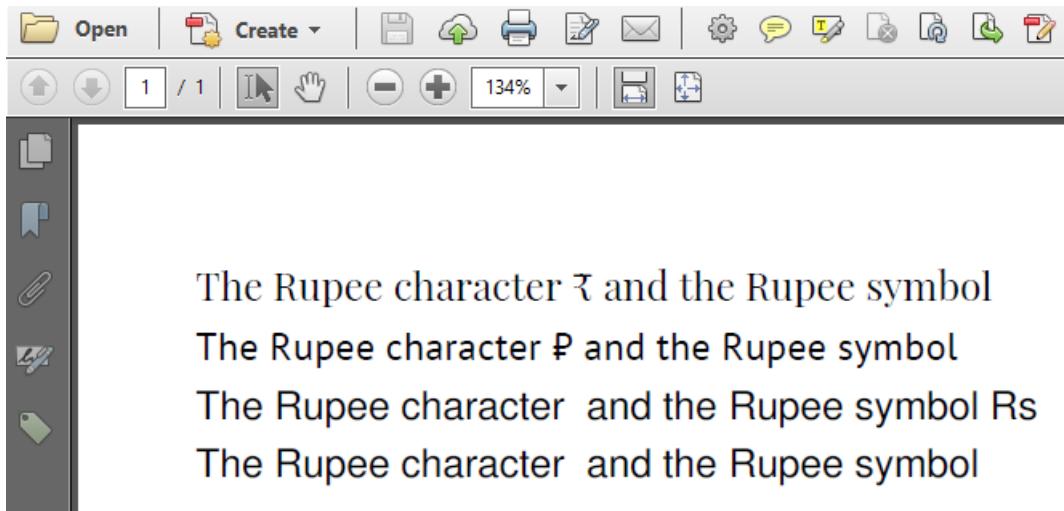
The RUPEE constant is a String that contains the Rupee character as well as the Rupee symbol: "The Rupee character ☰ and the Rupee symbol ☱". The characters are stored as Unicode values, because if we store the characters otherwise, they may not be rendered correctly. For instance: if you retrieve the values from a database as Winansi, you will end up with incorrect characters.

I test three different fonts (PlayfairDisplay-Regular.ttf, PT\_Sans-Web-Regular.ttf and FreeSans.ttf) and I use IDENTITY\_H as encoding three times. I also use WINANSI a fourth time to show that it goes wrong if you do.

The result is a file named [rupee.pdf](#)<sup>162</sup>:

---

<sup>162</sup><http://itextpdf.com/sites/default/files/rupee.pdf>



Rupee character vs Rupee Symbol

As you can see, the first two fonts know how to draw the Rupee character. The third one doesn't. The first two fonts don't know how to draw the Rupee symbol. The third one does. However, if you use the wrong encoding, none of the fonts draw the correct character or symbol.

In short: you need to find a font that knows how to draw the characters or symbols you need, then you have to make sure that you are using the correct encoding (for the String as well as the Font).

You can download the full sample code [here<sup>163</sup>](#)

## How to use System Font in iTextSharp with VB.net?



I'm using Itextsharp to convert text files to PDF documents dynamically using VB.net. However I need to use a system font that is not part of the iTextSharp library. I've seen some code examples using C#. However I'm a newbie in programming and my experience is all in Visual Basic. Can someone help me with writing the code to use a system font?

Posted on StackOverflow on [May 2, 2015<sup>164</sup>](#) by [Don Wilson<sup>165</sup>](#)

Suppose that you want to use Arial regular and you have the file `arial.ttf` in your `C:\windows\fonts` directory, then creating the `Font` object is as easy as this:

---

<sup>163</sup><http://itextpdf.com/sandbox/fonts/RupeeSymbol>

<sup>164</sup><http://stackoverflow.com/questions/29997695/how-to-use-system-font-in-itextsharp-with-vb-net>

<sup>165</sup><http://stackoverflow.com/users/2074355/don-wilson>

```
Dim arial As BaseFont = BaseFont.createFont("c:\windows\fonts\arial.ttf", BaseFont.IDENTITY_H, BaseFont.EMBEDDED)
font = New Font(arial, 16)
```

Using the font is easy too:

```
document.Add(New Paragraph("Hello World, Arial.", font))
```

This is almost a literal translation of the Java and C# examples that can be found in abundance. If this doesn't solve your problem, please show what you've tried and explain why it doesn't work.



Here's the code I'm going to attempt to use:

```
Dim fontpath As String = "C:\Windows\Fonts\
Dim bf As BaseFont = BaseFont.CreateFont(fontpath &
    "Arial monospaced for SAP.ttf", BaseFont.CP1252, BaseFont.EMBEDDED)
Dim ffont As New Font(bf, 7)
```

You claim that you have a file named Arial monospaced for SAP.ttf in the directory C:\Windows\Fonts\. I am 99% sure that this is not true. I have searched Google for such a font and I found a [webpage that says<sup>166</sup>](#):

Go to c:\windows\fonts and [it] should contain arimon\_\_.ttf and arimonbd.ttf

In other words, you need:

```
Dim arial As BaseFont = BaseFont.createFont(
    "c:\windows\fonts\arimon__.ttf", BaseFont.IDENTITY_H, BaseFont.EMBEDDED)
Dim arialbd As BaseFont = BaseFont.createFont(
    "c:\windows\fonts\arimonbd.ttf", BaseFont.IDENTITY_H, BaseFont.EMBEDDED)
```

This is *not* an iText problem. It is a problem of not understanding the difference between a file containing a font program and the name of that font program.



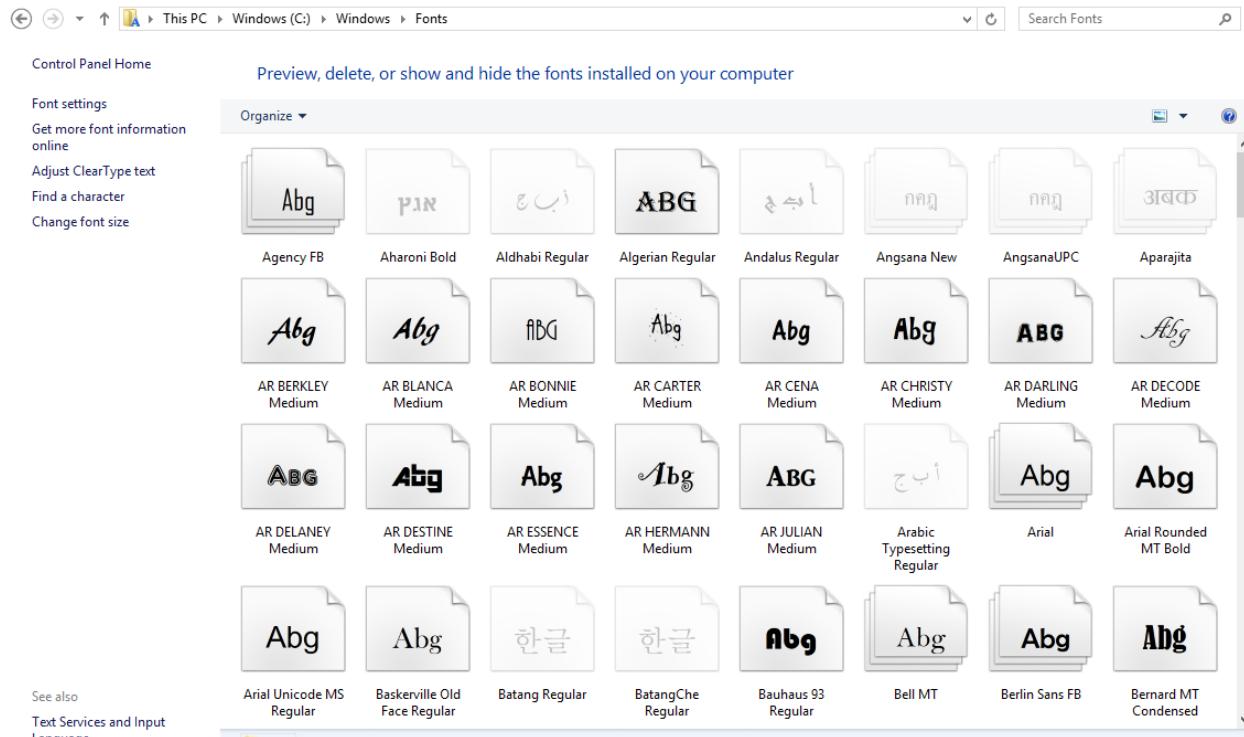
I'm looking at the system folder (C:\Windows\Fonts) and the file name is called just that.

---

<sup>166</sup><http://scn.sap.com/thread/1885079>

Allow me to guide you through the basics of Windows Explorer.

You are looking at your font directory like this:



Font directory

That view shows you the *names of the fonts*, NOT the *name of the font files*. Please select the view icon in the top right corner and change it to view the details. This is what you'll see:

The screenshot shows the Windows Control Panel Fonts window. The title bar reads "This PC > Windows (C:) > Windows > Fonts". The left sidebar includes links for Control Panel Home, Font settings, Get more font information online, Adjust ClearType text, Find a character, Change font size, and See also. The main area is titled "Preview, delete, or show and hide the fonts installed on your computer". It displays a table of fonts with columns: Name, Font style, Show/hide, Designed for, Category, and Designer/foundry. The "Name" column lists various fonts like Agency FB, Aharoni Bold, Aldhabi Regular, etc. The "Font style" column shows styles like Regular, Bold, Medium, etc. The "Show/hide" column indicates if the font is visible (Show) or hidden (Hide). The "Designed for" column specifies language support (Latin, Hebrew, Arabic, Thai, etc.). The "Category" column shows categories like Display, Text, and Glyph Systems. The "Designer/foundry" column lists companies like The Font Bureau, Inc., Kivun Computers Ltd., Microsoft Corporation, URW, and Glyp Systems.

### Detailed view font directory

Now right click on the header of the detailed list and select *Font file names*. This is what you'll see:

The screenshot shows the same Windows Control Panel Fonts window as above, but with a context menu open over the column headers. The menu is titled "Organize" and includes options: Size Column to Fit, Size All Columns to Fit, Name, Font style, Show/hide, Designed for, Category, Designer/foundry, Font embeddability, Font type, Family, Date created, Date modified, Size, Collection, Font file names, and Font version. The "Font file names" option is highlighted with a blue selection bar. The main table below shows the list of fonts with their corresponding font file paths in the "Font file names" column.

### Detailed view with file names

Use the path as shown in this overview and *your code will work*. If it doesn't work post a new

question and explain *exactly* what you're doing, so that we can help you.



I've reviewed your screen shots and have added the "Font File Name" column as suggested.

As you thought, there is not file name for "Arial Monospaced for SAP". To complicate matters, I don't see "arimon\_.ttf" or "arimonbd.ttf" listed either.

If you don't find arimon\_.ttf nor arimonbd.ttf under c:\windows\fonts, the fonts probably aren't there. If they aren't there, your code won't work. Another way to check their presence, is by clicking on Run under Windows (in the menu that opens when right-clicking the Start icon) and opening cmd. Then do cd c:\windows\fonts followed by dir ari\*. This will show you a list of all the font files that start with ari.

Take a look at the following screen shot that shows what happens when I do this on my machine:

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\iTText>cd c:\windows\fonts

c:\Windows\Fonts>dir ari*
Volume in drive C has no label.
Volume Serial Number is B277-1178

Directory of c:\Windows\Fonts

07/10/2014  01:19 AM      895,200 arial.ttf
07/10/2014  01:19 AM      857,344 arialbd.ttf
07/10/2014  01:19 AM      630,804 arialbi.ttf
07/10/2014  01:19 AM      635,860 ariali.ttf
10/02/2014  01:03 AM      175,956 ARIALN.TTF
10/02/2014  01:03 AM      180,740 ARIALNB.TTF
10/02/2014  01:03 AM      180,084 ARIALNBI.TTF
10/02/2014  01:03 AM      181,124 ARIALNI.TTF
11/18/2002  17:44 PM      23,275,812 ARIALUNI.TTF
08/22/2013  17:35 PM      167,476 ariblk.ttf
               10 File(s)   27,180,400 bytes
                  0 Dir(s)  260,835,086,336 bytes free

c:\Windows\Fonts>
```

Using cmd to find fonts

As you can see, there is no arimon\_.ttf or arimonbd.ttf in my c:\windows\fonts directory, hence code that tries to create a BaseFont or Font object using arimon\_.ttf or arimonbd.ttf would never work on my computer. If you create an application that requires these fonts, you'll have to ship these TTF files with your application (provided that you are allowed to do so).

## Do I need a license for Windows fonts when using iText?



I understand that iText does not come with any font programs and that you need to provide a font file. The PDFs I generate will be viewed in Adobe Reader and when I use Standard Type 1 fonts, Adobe Reader will have support for it, but my question is about licensing fonts.

1. Do I need to have a license of the fonts which I am using in iText? For example if I am using Arial, one of the most commonly used fonts (but which is not free), do I need to purchase the font?
2. If I am using any specific font (for example STSONG.ttf for Chinese characters), do I need to have a license for this ?

Posted on StackOverflow on Mar 20, 2014 <sup>167</sup> by vsingh<sup>168</sup>

Just like all software (including iText), fonts are licensed, but there are different types of licenses.

- You have fonts that have an open font license like [SIL](#)<sup>169</sup> which is the first that comes to mind, but there are many other licenses that allow you to use a font completely for free. **Caveat:** be careful with “free” fonts on the web. Many are just copies of protected fonts with their copyright notices removed.
- You have fonts that are proprietary in the sense that you can not redistribute them, but they allow you to embed the font in documents. For instance: the fonts that ship with MS Windows are proprietary. You are not allowed to copy them and ship them for free, however some fonts may be embedded within document files. Embedding allows fonts to travel with documents. Embedded fonts can only be used to print, preview and in some cases edit the document in which they are embedded. (See [Font redistribution and license issues](#)<sup>170</sup> in Microsoft’s FAQ)
- Some fonts require a full (paid) license even if you want to embed them. If I remember correctly, the fonts you can download as a font pack for Adobe Acrobat/Reader can only be used in the context of Acrobat. You are not allowed to embed them in a document using software that is not sold by Adobe.

I have different font examples in the [sandbox](#)<sup>171</sup> and for every font I use, I look at the [corresponding](#)

---

<sup>167</sup><http://stackoverflow.com/questions/29171445/does-one-need-to-have-a-license-for-fonts-if-we-are-using-ttf-files-in-itext>

<sup>168</sup><http://stackoverflow.com/users/156522/vsingh>

<sup>169</sup>[http://en.wikipedia.org/wiki/SIL\\_Open\\_Font\\_License](http://en.wikipedia.org/wiki/SIL_Open_Font_License)

<sup>170</sup><http://www.microsoft.com/typography/faq/faq11.htm>

<sup>171</sup><http://itextpdf.com/sandbox>

license<sup>172</sup>. If you check the [overview<sup>173</sup>](#), you'll see that I have fonts with a SIL license, an Apache license, and so on... I don't know about STSONG.ttf, so you'll have to check which license applies.

You were also asking explicitly about Arial. Arial<sup>174</sup> was created for the [Monotype Imaging Corporation<sup>175</sup>](#). The [Monotype Imaging EULA<sup>176</sup>](#) states:

You may embed the Font Software only into an electronic document that (i) is not a Commercial Product, (ii) is distributed in a secure format that does not permit the extraction of the embedded Font Software, and (iii) in the case where a recipient of an electronic document is able to Use the Font Software for editing, only if the recipient of such document is within your Licensed Unit.

However, you probably didn't obtain the font from Monotype. It was probably shipped with your legal copy of Microsoft Windows, and Monotype may have licensed its fonts to Windows under a license that is less strict.

Let's check which license applies by right clicking a font file such as arial.ttf in C:\windows\fonts and let's inspect the Properties:

---

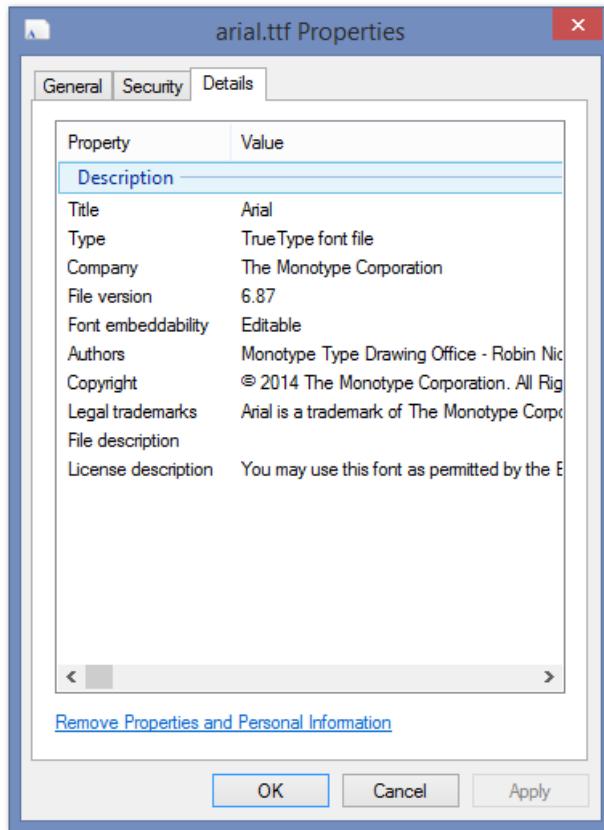
<sup>172</sup><https://github.com/itext/sandbox/tree/master/resources/fonts/licenses>

<sup>173</sup><https://github.com/itext/sandbox/blob/master/resources/fonts/licenses/overview.txt>

<sup>174</sup><http://en.wikipedia.org/wiki/Arial>

<sup>175</sup>[http://en.wikipedia.org/wiki/Monotype\\_Imaging](http://en.wikipedia.org/wiki/Monotype_Imaging)

<sup>176</sup><http://www.fonts.com/info/legal/eula/monotype-imaging>



Properties of arial.ttf

I am in luck: in my case, the Font embeddability says “Editable”. This means that I can embed the font in a document not only to print and (pre)view the document, but also to edit the document (e.g. edit form fields) as long as I use arial.ttf in the context of the Windows license that I bought. I can not copy arial.ttf to another computer.

Note that you shouldn't assume that was always the case. For instance if we look at [Arial Unicode<sup>177</sup>](#), we see that older versions only allowed print and preview embedding, whereas editable embedding was allowed only for version 1.00 and higher. You should really check the properties of each font you are planning to use.

If you need a font that can be distributed, you should look for an alternative font that has an open font license. I use FreeSans in the sandbox examples, but there are other [free alternatives<sup>178</sup>](#).

<sup>177</sup>[http://en.wikipedia.org/wiki/Arial\\_Unicode\\_MS](http://en.wikipedia.org/wiki/Arial_Unicode_MS)

<sup>178</sup>[http://en.wikipedia.org/wiki/Arial#Free\\_alternatives](http://en.wikipedia.org/wiki/Arial#Free_alternatives)

## How to get a font from an array?



Is it possible to create a Font object from a byte array instead of using file paths? The `FontFactory` class has two methods for registering fonts. Both use file/folder paths to register font.

Posted on StackOverflow on May 6, 2015 <sup>[179](#)</sup> by Robert<sup>[180](#)</sup>

Yes, you can create a `Font` from a `byte[]`, but in that case, you can't use the `FontFactory`. Instead you need to create a `BaseFont` instance using the `createFont` method, see <http://api.itextpdf.com/itext/com/itextpdf/text/BaseFont.html> for the different options.

Once you have a `BaseFont` instance, you can easily create a `Font` object:

Suppose that `fBytes` is a `byte[]`, then you'd have:

```
BaseFont bf = BaseFont.createFont(  
    "myFont", BaseFont.IDENTITY_H, BaseFont.EMBEDDED,  
    true, fBytes, null);  
Font font = new Font(bf, 12);
```

The method accepts two `byte[]` parameters because Type 1 fonts consist of two files: a metrics file (AFM or PFM) and a font binary (PFB). For all other fonts (TTF, OTF,...), the second `byte[]` parameter should be `null`.

There is currently no way to add such a font to the `FontFactory` as a “registered font”.

---

<sup>179</sup><http://stackoverflow.com/questions/30070613/register-font-from-byte-array-in-itext>

<sup>180</sup><http://stackoverflow.com/users/4587348/robert>

# Images

In this section, you'll find the questions related to raster images, such as JPG, PNG, GIF, and so on.

## How to add multiple images into a single PDF?



I have the following code but this code add only the last image into pdf.

```
Document document = new Document();
PdfWriter writer = PdfWriter.getInstance(document,
    new FileOutputStream(filePath));
document.open();
for (String imageIpath : imagePathsList) {
    Image image1 = Image.getInstance(imageIpath);
    image1.setAbsolutePosition(10f, 10f);
    image1.scaleAbsolute(600, 800);
    document.add(image1);
}
writer.close();
```

Would you give me a hint about how to update the code in order to add all the images into the exported pdf? The `imagePathsList` variable contains all the paths of images that that I want to add into a single pdf.

Posted on StackOverflow on Mar 29, 2015 <sup>181</sup> by aurelianr<sup>182</sup>

Take a look at the [MultipleImages<sup>183</sup>](#) example and you'll discover that there are two errors in your code:

1. You create a page with size 595 x 842 user units, and you add every image to that page regardless of the dimensions of the image.
2. You claim that only one image is added, but that's not true. You are adding *all* the images *on top of each other* on the *same page*. The last image covers all the preceding images.

Take a look at my code:

<sup>181</sup><http://stackoverflow.com/questions/29331838/add-multiple-images-into-a-single-pdf-file-with-itext-using-java>

<sup>182</sup><http://stackoverflow.com/users/1498697/aurelianr>

<sup>183</sup><http://itextpdf.com/sandbox/images/MultipleImages>

```

public void createPdf(String dest) throws IOException, DocumentException {
    Image img = Image.getInstance(IMAGES[0]);
    Document document = new Document(img);
    PdfWriter.getInstance(document, new FileOutputStream(dest));
    document.open();
    for (String image : IMAGES) {
        img = Image.getInstance(image);
        document.setPageSize(img);
        document.newPage();
        img.setAbsolutePosition(0, 0);
        document.add(img);
    }
    document.close();
}

```

I create a Document instance using the size of the first image. I then loop over an array of images, setting the page size of the next page to the size of each image *before* I trigger a `newPage()` [\*]. Then I add the image at coordinate  $(0, 0)$  because now the size of the image will match the size of each page.

[\*] The `newPage()` method only has effect if something was added to the current page. The first time you go through the loop, nothing has been added yet, so nothing happens. This is why you need set the page size to the size of the first image when you create the Document instance.

## How to get the image DPI in PDF?



I'm trying to get information about scanned images that are saved into PDF files through iText (using Java). I can get width and height (either through Matrix, or through BufferedImage). The idea was to use the answer here to calculate the DPI, but I am a bit lost. Are these values (width and height) in pixels or points? Is there any other way to achieve this? There are a lot of answers on how to scale and save an image to a PDF file, but I didn't find any on how to read the width/height/scale of an image and be confident about the result.

Posted on StackOverflow on Aug 28, 2014 <sup>184</sup> by Finik<sup>185</sup>

Let's split this problem into two separate problems. To calculate the DPI, you need two sets of values: a number of pixels and a distance in inch.

1. Number of pixels: you obtain the image and the image consists of pixels. You can retrieve the width and height of the image in pixels from the image. Let's say these values are  $w_{Px}$  and  $w_{Px}$ .

---

<sup>184</sup><http://stackoverflow.com/questions/25550000/getting-image-dpi-in-pdf-files-using-itext>

<sup>185</sup><http://stackoverflow.com/users/1300529/finik>

2. Distance in inch: you obtain the matrix which gives you values expressed in points. As 72 points equal 1 inch, you need to divide these values by 72. Let's say these values are `wInch` and `hInch`.

Now you can calculate the DPI in the x direction like this: `wPx / wInch` and the DPI in the y direction like this: `hPx / hInch`.

## Why aren't images added sequentially?



I am working on a pdf report that contains topics and images (charts). The document is formatted this way:

NR. TOPIC TITLE FOR TOPIC 1

CHART IMAGE for topic 1 (from bytearray)

NR. TOPIC TITLE FOR TOPIC 2

CHART IMAGE for topic 2 ...

Let's assume that I add this information in a loop, and that the loop runs 10 times. I expect 10 topic titles all directly followed by the image.

However, if the page end is reached and a new image should be added, I notice that the image is moved to the next page and the next topic title is printed on the previous page.

So on paper we have:

```
page 1: topic 1
        image topic 1
        topic 2
        image topic 2
        topic 3
        topic 4
page 2: image topic 3
        image topic 4
        topic 5
        image topic 5
```

So the order of the elements on paper, is NOT the same as the order that I used to put the element in the document via the 'document.add()' method. This is really strange. Anyone has any idea?

Posted on StackOverflow on Mar 26, 2014 <sup>186</sup> by [wim boone](#)<sup>187</sup>

---

<sup>186</sup><http://stackoverflow.com/questions/22664126/image-not-sequentialy-added-in-pdf-document-itextsharp-wrong-order-of-elements>

<sup>187</sup><http://stackoverflow.com/users/3149519/wim-boone>

If you have a `PdfWriter` instance (for instance `writer`), you need to force iText to use strict image sequence like this:

```
writer.setStrictImageSequence(true);
```

Otherwise, iText will postpone adding images until there's sufficient space on the page to add the image.

## How to add text to an image?

This is the original question:



### Pdf vertical postion method gives the next page position instead of current page

In my project I use iText to generate a PDF document. Suppose that the height of a page measures 500pt (1 user unit = 1 point), and that I write some text to the page, followed by an image. If the content and the image require less than 450pt, the text precedes the image. If the content and the image exceed 450pt, the text is forwarded to the next page. My question is: how can I obtain the remaining available space before writing an image?

Posted on StackOverflow on Nov 6, 2014 <sup>188</sup> by Madhesh<sup>189</sup>



This is a good example of a question that is phrased in a way that nobody can answer it correctly. After a lot of back and forth comments, it was finally clear that the OP wanted to add a text watermark to an image. Badly phrase questions can be very frustrating. Please take that in consideration when asking a question. In this case, I gave several wrong answers before it became clear what was actually asked.

First things first: when adding text and images to a page, iText sometimes changes the order of the textual content and the image. You can avoid this by using:

```
writer.setStrictImageSequence(true);
```

If you want to know the current position of the “cursor”, you can use the method `getVerticalPosition()`. Unfortunately, this method isn't very elegant: it requires a Boolean parameter that will add a newline (if `true`) or give you the position at the current line (if `false`).

I do not understand why you want to get the vertical position. Is it because you want to have a caption followed by an image, and you want the caption and the image to be at the same page?

---

<sup>188</sup><http://stackoverflow.com/questions/26814958/pdf-vertical-postion-method-gives-the-next-page-position-instead-of-current-page>

<sup>189</sup><http://stackoverflow.com/users/2894742/madhesh>

In that case, you could put your text and images inside a table cell and instruct iText not to split rows. In this case, iText will forward both text and image, in the correct order to the next page if the content doesn't fit the current page.

### Update:

Based on the extra information added in the comments, it is now clear that the OP wants to add images that are watermarked.

There are two approaches to achieve this, depending on the actual requirement.

#### Approach 1:

The first approach is explained in the [WatermarkedImages1<sup>190</sup>](#) example. In this example, we create a PdfTemplate to which we add an image as well as some text written on top of that image. We can then wrap this PdfTemplate inside an image and add that image together with its watermark using a single document.add() statement.

This is the method that performs all the magic:

```
public Image getWatermarkedImage(PdfContentByte cb, Image img, String watermark)
    throws DocumentException {
    float width = img.getScaledWidth();
    float height = img.getScaledHeight();
    PdfTemplate template = cb.createTemplate(width, height);
    template.addImage(img, width, 0, 0, height, 0, 0);
    ColumnText.showTextAligned(template, Element.ALIGN_CENTER,
        new Phrase(watermark, FONT), width / 2, height / 2, 30);
    return Image.getInstance(template);
}
```

This is how we add the images:

```
PdfContentByte cb = writer.getDirectContentUnder();
document.add(getWatermarkedImage(cb, Image.getInstance(IMAGE1), "Bruno"));
document.add(getWatermarkedImage(cb, Image.getInstance(IMAGE2), "Dog"));
document.add(getWatermarkedImage(cb, Image.getInstance(IMAGE3), "Fox"));
Image img = Image.getInstance(IMAGE4);
img.scaleToFit(400, 700);
document.add(getWatermarkedImage(cb, img, "Bruno and Ingeborg"));
```

As you can see, we have one very large image (a picture of my wife and me). We need to scale this image so that it fits the page. If you want to avoid this, take a look at the second approach.

#### Approach 2:

The second approach is explained in the [WatermarkedImages2<sup>191</sup>](#) example. In this case, we add each

---

<sup>190</sup><http://itextpdf.com/sandbox/images/WatermarkedImages1>

<sup>191</sup><http://itextpdf.com/sandbox/images/WatermarkedImages2>

image to a PdfPCell1. This PdfPCell will scale the image so that it fits the width of the page. To add the watermark, we use a cell event:

```
class WatermarkedCell implements PdfPCellEvent {
    String watermark;

    public WatermarkedCell(String watermark) {
        this.watermark = watermark;
    }

    public void cellLayout(PdfPCell cell, Rectangle position,
        PdfContentByte[] canvases) {
        PdfContentByte canvas = canvases[PdfPTable.TEXTCANVAS];
        ColumnText.showTextAligned(canvas, Element.ALIGN_CENTER,
            new Phrase(watermark, FONT),
            (position.getLeft() + position.getRight()) / 2,
            (position.getBottom() + position.getTop()) / 2, 30);
    }
}
```

This cell event can be used like this:

```
PdfPCell cell;
cell = new PdfPCell(Image.getInstance(IMAGE1), true);
cell.setCellEvent(new WatermarkedCell("Bruno"));
table.addCell(cell);
cell = new PdfPCell(Image.getInstance(IMAGE2), true);
cell.setCellEvent(new WatermarkedCell("Dog"));
table.addCell(cell);
cell = new PdfPCell(Image.getInstance(IMAGE3), true);
cell.setCellEvent(new WatermarkedCell("Fox"));
table.addCell(cell);
cell = new PdfPCell(Image.getInstance(IMAGE4), true);
cell.setCellEvent(new WatermarkedCell("Bruno and Ingeborg"));
table.addCell(cell);
```

You will use this approach if all images have more or less the same size, and if you don't want to worry about fitting the images on the page.

**Consideration:**

Obviously, both approaches have a different result because of the design choice that is made. Please compare the resulting PDFs to see the difference: [watermark\\_template.pdf<sup>192</sup>](#) versus [watermark\\_table.pdf<sup>193</sup>](#)

## How to draw lines on an image?



I am trying to draw lines on image that needs to be added to a document, just like we draw graphics on paint event of any control. How is this done?

Posted on StackOverflow on [Apr 10, 2015<sup>194</sup>](#) by [Harsh Kumar Singh<sup>195</sup>](#)

Please take a look at the [WatermarkedImages4<sup>196</sup>](#) example. It is based on the [WatermarkedImages1<sup>197</sup>](#) example. The only different between the two examples is that we add text in the example written in answer to [How to add text to an image?<sup>198</sup>](#) whereas we add lines in the example written in answer to your question.

We add images like this:

```
document.add(getWatermarkedImage(cb, Image.getInstance(IMAGE1)));
```

The getWatermarkedImage() method looks like this:

```
public Image getWatermarkedImage(PdfContentByte cb, Image img) throws DocumentException {
    float width = img.getScaledWidth();
    float height = img.getScaledHeight();
    PdfTemplate template = cb.createTemplate(width, height);
    template.addImage(img, width, 0, 0, height, 0, 0);
    template.saveState();
    template.setColorStroke(BaseColor.GREEN);
    template.setLineWidth(3);
    template.moveTo(width * .25f, height * .25f);
    template.lineTo(width * .75f, height * .75f);
    template.moveTo(width * .25f, height * .75f);
    template.lineTo(width * .25f, height * .25f);
```

<sup>192</sup>[http://itextpdf.com/sites/default/files/watermark\\_template.pdf](http://itextpdf.com/sites/default/files/watermark_template.pdf)

<sup>193</sup>[http://itextpdf.com/sites/default/files/watermark\\_table.pdf](http://itextpdf.com/sites/default/files/watermark_table.pdf)

<sup>194</sup><http://stackoverflow.com/questions/29561417/draw-lines-on-image-in-pdf-using-itextsharp>

<sup>195</sup><http://stackoverflow.com/users/4420141/harsh-kumar-singhi>

<sup>196</sup><http://itextpdf.com/sandbox/images/WatermarkedImages4>

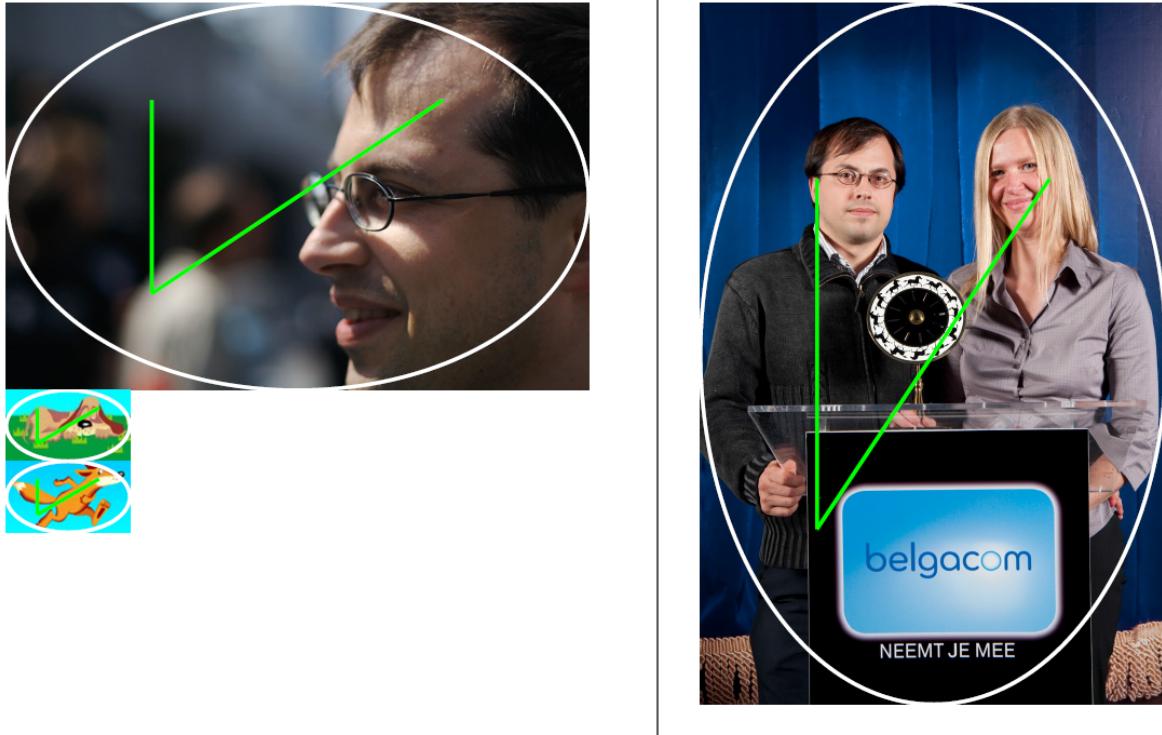
<sup>197</sup><http://itextpdf.com/sandbox/images/WatermarkedImages1>

<sup>198</sup><http://stackoverflow.com/questions/26814958/how-to-add-text-to-an-image>

```
template.stroke();
template.setColorStroke(BaseColor.WHITE);
template.ellipse(0, 0, width, height);
template.stroke();
template.restoreState();
return Image.getInstance(template);
}
```

As you can see, I add two green lines using `moveTo()`, `lineTo()` and `stroke()`. I also add a white ellipse using the `ellipse()` and `stroke()` method.

This results in a PDF that looks like this:



Some lines added to some images

As you can see, the shape of the ellipse and the position of the lines are different for the different images because I defined my coordinates based on the width and the height of the image.

## How to preserve high resolution images in PDF?



I'm trying to put high quality images into PDF (one per page). But if I set page size to a4, I have to resize my pictures, because they're too large. Then they lose their quality. Is there any way to put big image to a4 page without losing quality?

I'm using iTextSharp library, firstly I'm creating the document

```
document = new Document(PageSize.A4, 0, 0, 0, 0);
FileStream output = new FileStream(pdfPath + "document.pdf", FileMode.Create);
PdfWriter writer = PdfWriter.GetInstance(document, output);
document.Open();
```

then I'm adding each picture

```
document.Add(iTextSharp.text.Image.GetInstance(toSaveImage, System.Drawing.Imaging.ImageFormat.Tiff));
```

and closing the document

```
document.Close();
```

Posted on StackOverflow on [jun 6, 2013](#) <sup>199</sup> by [Com Piler](#)<sup>200</sup>

First let me clear a couple of misunderstandings:

- a PDF document as such doesn't have a resolution. There is no such thing as DPI in PDF. The resolution only comes into play when a PDF is rendered (to the screen, to paper,...) and that's why there may be a DPI in a PDF viewer (but that's something completely different).
- when you scale an `Image` object in iTextSharp, you don't lose any information: the number of pixels remains the same. Whereas PDF doesn't have a resolution, the images inside a PDF do. When you scale down (that is: you put the same number of pixels on a smaller canvas), the resolution increases; when you scale up, the resolution decreases.

Now for your question: you're not obliged to create A4 pages:

---

<sup>199</sup><http://stackoverflow.com/questions/16970106/c-sharp-high-resolution-images-in-pdf>

<sup>200</sup><http://stackoverflow.com/users/2411220/com-piler>

```
Image img =
    iTextSharp.text.Image.GetInstance(toSaveImage,
        System.Drawing.Imaging.ImageFormat.Tiff);
Rectangle pagesize = new Rectangle(img.ScaledWidth, img.ScaledHeight);
Document document = new Document(pagesize);
img.SetAbsolutePosition(0, 0);
document.Add(img);
```

I created the Document based on the scaled dimensions of the Image. Don't let the method names mislead you: ScaledWidth and ScaledHeight are the safest methods to use when getting the dimensions of an Image. Not only do they include any scaling operations, you may have done on the image, they also take into account the space needed for the image after rotating it.

Observe that I've set the absolute position to the lower-left corner. That's safer than setting the page margins to 0.

If you don't want to change the page size, then you have to use the `ScaleToFit()` method:

```
Image img =
    iTextSharp.text.Image.GetInstance(toSaveImage,
        System.Drawing.Imaging.ImageFormat.Tiff);
img.ScaleToFit(PageSize.A4);
```

Scale to fit will keep the aspect ratio of the image. If the aspect ratio of the image is different from the aspect ratio of the page, the page will have a margin.

## How to add JPEG images that are multi-stage filtered (DCTDecode and FlateDecode)?



Recently I am tasked with reducing the file sizes of PDF generated from blank office documents. The images are mostly blank, but they have a variety of company letterheads (in color), borders and footers. Some are generated by software (and therefore have very clean pixels), others are scanned by desktop scanners. Being “blank”, what I mean is center part of the page (two inches away from each margin) will be absolutely blank and white.

My boss want to keep these PDFs in color, but wouldn't mind making them fuzzier as long as they are not too ugly. I have tested with many file reduction schemes: different color compression methods (FlateDecode, LZWDecode, DCTDecode, ...), different JPEG quality settings, reducing the pixel width and height of the JPEG, and only stretch it up when the PDF is displayed, cutting up the image content into smaller patches of images,...

So far, I have found that the third method (reducing pixel dimensions) was more effective than reducing the JPEG quality settings (say, scaling image down 50% as opposed to dropping quality from 50 down to 20)

However, one approach that has eluded me in some of the sample PDFs we collected from other companies is that some had JPEG images that are **multi-stage filtered**. What I mean is that:

- The Filter name is an array of two: /DCTDecode, /FlateDecode
- when the PDF was generated, JPEG compression was applied first, and then it was Deflated. Upon viewing the PDF, the data was first Inflated and then JPEG decompressed into pixels.

I applied the first step of decompression (the FlateDecode) to the multi-stage filtered data, giving the original JPEG data. I used a hex viewer to inspect the JPEG data, and found that in the blank areas of the JPEG image, most of the bytes are repeated patterns. This explains why it would be advantageous to apply a secondary compression on top of an JPEG file - if only one knows that JPEG image is mostly blank.

Apparently those PDFs were also created by iText. However, it is not clear to me if there is an instance of the `iTextSharp.text.Image` class that supports this combination of two-stage filtering.

In case iText does not have built-in support for creating such two-stage compressed images, would it be possible to insert an image if I handle the two-stage compression and use something like `PdfImageObject`<sup>201</sup>?

Posted on StackOverflow on Feb 22, 2014 <sup>202</sup> by rwong<sup>203</sup>

<sup>201</sup><http://api.itextpdf.com/itext/com/itextpdf/text/pdf/parser/PdfImageObject.html>

<sup>202</sup><http://stackoverflow.com/questions/21958449/can-itextsharp-generate-pdf-with-jpeg-images-that-are-multi-stage-filtered-both>

<sup>203</sup><http://stackoverflow.com/users/377657/rwong>

I've created two examples, one that will work with old iText versions, one that will only work with the iText 5.5.1 and later versions:

```
Image img = Image.getInstance("some.jpg");
img.setCompressionLevel(PdfStream.BEST_COMPRESSION);
```

As you are passing a JPEG to iText, the filter will be /DCTDecode. If you then use the `setCompressionLevel()` method, the stream will be deflated too. You'll have two entries in the filter: /FlateDecode and /DCTDecode. This is shown in the [FlateCompressJPEG1Pass](#)<sup>204</sup> example.

I've also created the [FlateCompressJPEG2Passes](#)<sup>205</sup> example for people who are stuck with an older iText version:

```
PdfReader reader = new PdfReader(src);
// We assume that there's a single large picture on the first page
PdfDictionary page = reader.getPageN(1);
PdfDictionary resources = page.getAsDict(PdfName.RESOURCES);
PdfDictionary xobjects = resources.getAsDict(PdfName.XOBJECT);
PdfName imgName = xobjects.getKeys().iterator().next();
PRStream imgStream = (PRStream)xobjects.getAsStream(imgName);
imgStream.setData(PdfReader.getStreamBytesRaw(imgStream), true);
PdfArray array = new PdfArray();
array.add(PdfName.FLATEDECODE);
array.add(PdfName.DCTDECODE);
imgStream.put(PdfName.FILTER, array);
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
stamper.close();
reader.close();
```

This code sample post-processes an existing document, it requires more code and it's more error prone: in this short snippet, I assume that there's a single XObject and that this single object is an image. This may not be the case for your PDFs.

---

<sup>204</sup><http://itextpdf.com/sandbox/images/FlateCompressJPEG1Pass>

<sup>205</sup><http://itextpdf.com/sandbox/images/FlateCompressJPEG2Passes>

## How to avoid an exception when importing a TIFF file?



When I try to convert a specific tiff file to PDF, the following exception occurs:

java.lang.RuntimeException: Scanline must begin with EOL code word.

I can open the tiff file in any image viewer, so it's valid.

Posted on StackOverflow on [Apr 22, 2015](#) <sup>206</sup> by [hecatcat](#)<sup>207</sup>



Comment by Bruno: this statement is not *I can open the tiff file in any image viewer, so it's valid.* A TIFF file can be invalid and open in an image viewer anyway because the image viewer tolerates errors. iText can also tolerate errors, as explained in the answer provided by [Michaël Demey](#)<sup>208</sup> (from iText).

iText has a few fall backs when dealing with invalid or corrupt Tiff files. By default, these fallbacks aren't used, you'll need to explicitly use one of the `getInstance()` methods with the `recoverFromImageError` flag set to true if you want iText to try and parse the invalid Tiff files  
`getInstance(byte[] imgb, boolean recoverFromImageError)`<sup>209</sup>

If this Boolean is set to true, iText will only throw an error if it exhausted all of its options.

Another workaround could be to use the `TiffImage`<sup>210</sup> class, bypassing the `Image` class altogether. The `TiffImage` class also uses the `recoverFromImageError` flag, but it also has an additional flag called `direct` which might also solve your issues.

## How to show an image with large dimensions across multiple pages?



I have an image with large dimensions which I need to display in a PDF file. I cant scale to fit the image as this would make text on the node illegible. How could I split the image into multiple pages while retaining its original dimensions?

Posted on StackOverflow on [Nov 11, 2014](#) <sup>211</sup> by [Raja](#)<sup>212</sup>

<sup>206</sup><http://stackoverflow.com/questions/29787388/exception-when-converting-tiff-file-to-pdf-file-with-itext>

<sup>207</sup><http://stackoverflow.com/users/4817472/hecatcat>

<sup>208</sup><http://stackoverflow.com/users/512061/micha%C3%ABl-demey>

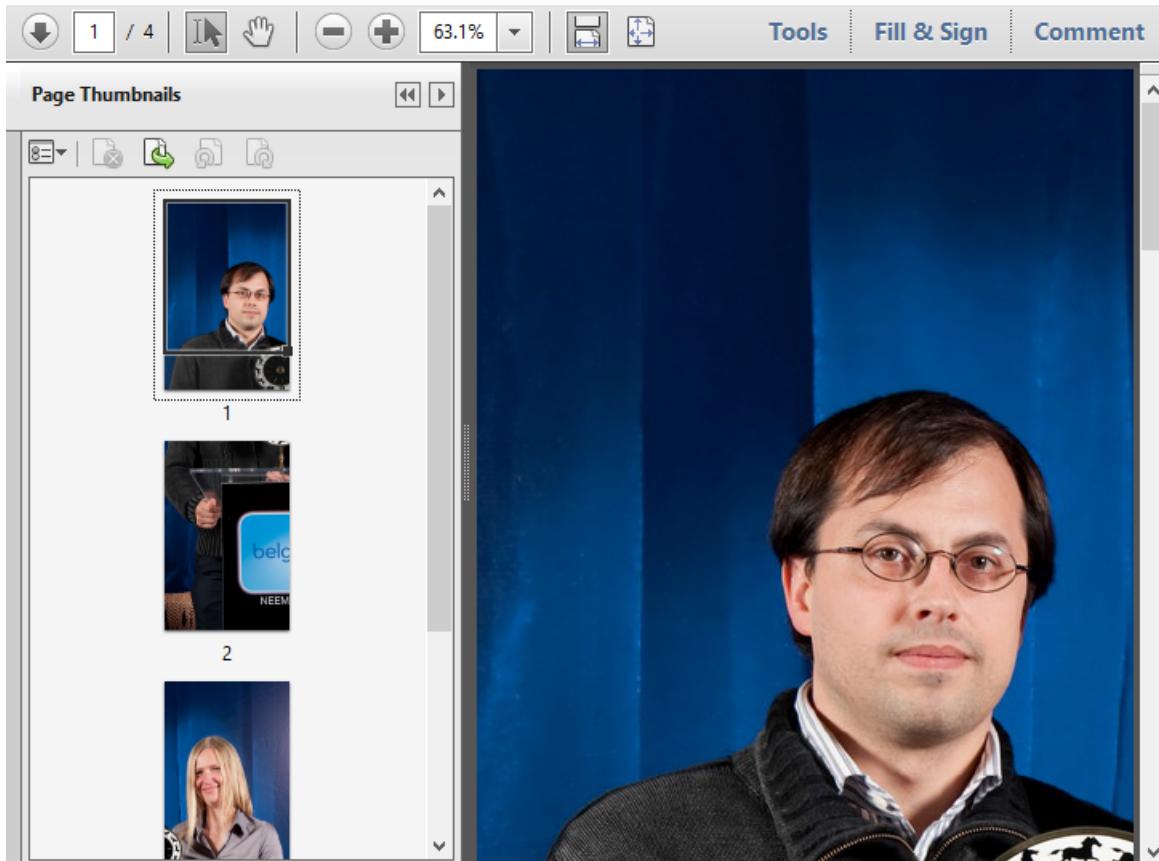
<sup>209</sup>[http://api.itextpdf.com/itext/com/itextpdf/text/Image.html#getInstance%28byte\[\],%20boolean%29](http://api.itextpdf.com/itext/com/itextpdf/text/Image.html#getInstance%28byte[],%20boolean%29)

<sup>210</sup><http://api.itextpdf.com/itext/com/itextpdf/text/pdf/codec/TiffImage.html>

<sup>211</sup><http://stackoverflow.com/questions/26859473/how-to-show-an-image-with-large-dimensions-across-multiple-pages-in-itextpdf>

<sup>212</sup><http://stackoverflow.com/users/2964266/raja>

Please take a look at the [TiledImage<sup>213</sup>](#) example. It takes an image at its original size and it tiles it over 4 pages: [tiled\\_image.pdf<sup>214</sup>](#)



Screen shot

To make this work, I first asked the image for its size:

```
Image image = Image.getInstance(IMAGE);
float width = image.getScaledWidth();
float height = image.getScaledHeight();
```

To make sure each page is as big as one fourth of the page, I define this rectangle:

```
Rectangle page = new Rectangle(width / 2, height / 2);
```

I use this rectangle when creating the Document instance and I add the same image 4 times using different coordinates:

---

<sup>213</sup><http://itextpdf.com/sandbox/images/TiledImage>

<sup>214</sup>[http://itextpdf.com/sites/default/files/tiled\\_image.pdf](http://itextpdf.com/sites/default/files/tiled_image.pdf)

```

Document document = new Document(page);
PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream(dest));
document.open();
PdfContentByte canvas = writer.getDirectContentUnder();
canvas.addImage(image, width, 0, 0, height, 0, -height / 2);
document.newPage();
canvas.addImage(image, width, 0, 0, height, 0, 0);
document.newPage();
canvas.addImage(image, width, 0, 0, height, -width / 2, -height / 2);
document.newPage();
canvas.addImage(image, width, 0, 0, height, -width / 2, 0);
document.close();

```

Now I have distributed the image over different pages, which is exactly what you are trying to achieve ;-)

## How to give an image rounded corners?



I'm using iTextSharp to export images into a PDF. Now I want to get the image width and height (while getting the image from disk) and make the edges of image curved (rounded corners), I get and add the image like this:

```

pdfDoc.Open();
//pdfDoc.Add(new iTextSharp.text.Paragraph("Welcome to dotnetfox"));
iTextSharp.text.Image gif =
    iTextSharp.text.Image.GetInstance(@"C:\images\logoall.bmp");
// gif.ScaleToFit(500, 100);
pdfDoc.Add(gif);

```

Posted on StackOverflow on Sep 20, 2014<sup>215</sup> by Annadate Piyush<sup>216</sup>

### Question 1: What is the size of an image?

You have an `Image` instance `gif`. The width of this image is `gif.ScaledWidth` and `jpg.ScaledHeight`. There are other ways to get the width and the height, but this way always gives you the size in user units that will be used in the PDF.

If you do not scale the image, `ScaledWidth` and `ScaledHeight` will give you the original size of the image in pixels. Pixels will be treated as user units by iText. In PDF, a user unit corresponds with a point by default (and 72 points correspond with 1 inch).

<sup>215</sup><http://stackoverflow.com/questions/25946116/how-to-smooth-the-edge-of-the-image-from-disk-exporting-to-pdf>

<sup>216</sup><http://stackoverflow.com/users/4057042/annadate-piyush>

**Question 2:** How do you display the image with rounded corners?

Some image formats (such as PNG) allow transparency. You could create an image in such a way that the effect of rounded corners is mimicked by making the corners transparent.

If this is not an option, you should apply a clipping path. This is demonstrated in the [ClippingPath<sup>217</sup>](#) example in chapter 10 of my book.

Ported to C#, the example would be something like this:

```
Image img = Image.GetInstance(some_path_to_an_image);
float w = img.ScaledWidth;
float h = img.ScaledHeight;
PdfTemplate t = writer.DirectContent.CreateTemplate(w, h);
t.Ellipse(0, 0, w, h);
t.Clip();
t.NewPath();
t.AddImage(img, w, 0, 0, h, 0, -600);
Image clipped = Image.GetInstance(t);
```

Of course: this clips the image into an ellipse as shown in the [resulting PDF<sup>218</sup>](#). You need to replace the `Ellipse()` method from the example with the `RoundRectangle()` method.

## How to convert colored images to black And white?



I'm trying to compress PDFs using iTextSharp. There are a lot of pages with color images stored as JPEGs (DCTDECODE)... so I'm converting them to black and white PNGs and replacing them in the document (the PNG is much smaller than a JPG for black and white format).

I've tried varieties of COLORSPACES and BITSPERCOMPONENTs, but always get "Insufficient data for an image", "Out of memory", or "An error exists on this page" upon trying to open the resulting PDF... so I must be doing something wrong.

Posted on StackOverflow on [Oct 27, 2014](#) <sup>219</sup> by [Jeff](#)<sup>220</sup>

### The Question:

You have a PDF with a colored JPG. For instance: [image.pdf<sup>221</sup>](#)

---

<sup>217</sup><http://itextpdf.com/examples/ii.php?id=191>

<sup>218</sup>[http://examples.itextpdf.com/results/part3/chapter10/clipping\\_path.pdf](http://examples.itextpdf.com/results/part3/chapter10/clipping_path.pdf)

<sup>219</sup><http://stackoverflow.com/questions/26580912/pdf-convert-to-black-and-white-pngs>

<sup>220</sup><http://stackoverflow.com/users/326518/jeff>

<sup>221</sup><http://itextpdf.com/sites/default/files/image.pdf>

If you look inside this PDF, you'll see that the filter of the image stream is /DCTDecode and the color space is /DeviceRGB.

Now you want to replace the image in the PDF, so that the result looks like this: [image\\_replaced.pdf](#)<sup>222</sup>

In this PDF, the filter is /FlateDecode and the color space is change to /DeviceGray.

In the conversion process, you want to user a PNG format.

### The Example:

I have prepared an example for you that makes this conversion: [ReplaceImage](#)<sup>223</sup>

I will explain this example step by step:

#### Step 1: finding the image

In my example, I know that there's only one image, so I'm retrieving the PRStream with the image dictionary and the image bytes in a quick and dirty way.

```
PdfReader reader = new PdfReader(src);
PdfDictionary page = reader.getPageN(1);
PdfDictionary resources = page.getAsDict(PdfName.RESOURCES);
PdfDictionary xobjects = resources.getAsDict(PdfName.XOBJECT);
PdfName imgRef = xobjects.getKeys().iterator().next();
PRStream stream = (PRStream) xobjects.getAsStream(imgRef);
```

I go to the /XObject dictionary with the /Resources listed in the page dictionary of page 1. I take the first XObject I encounter, assuming that it is an image, and I get that image as a PRStream object.

The code you shared is better than mine, but this part of the code isn't relevant to your question and it works in the context of my example, so let's ignore the fact that this won't work for other PDFs. What you really care about are steps 2 and 3.

#### Step 2: converting the colored JPG into a black and white PNG

Let's write a method that takes a PdfImageObject and that converts it into an Image object that is changed into gray colors and stored as a PNG:

---

<sup>222</sup>[http://itextpdf.com/sites/default/files/image\\_replaced.pdf](http://itextpdf.com/sites/default/files/image_replaced.pdf)

<sup>223</sup><http://itextpdf.com/sandbox/images/ReplaceImage>

```
public static Image makeBlackAndWhitePng(PdfImageObject image) throws IOException\\n, DocumentException {  
    BufferedImage bi = image.getBufferedImage();  
    BufferedImage newBi = new BufferedImage(bi.getWidth(), bi.getHeight(), BufferedImage.TYPE_USHORT_GRAY);  
    newBi.getGraphics().drawImage(bi, 0, 0, null);  
    ByteArrayOutputStream baos = new ByteArrayOutputStream();  
    ImageIO.write(newBi, "png", baos);  
    return Image.getInstance(baos.toByteArray());  
}
```

We convert the original image into a black and white image using standard `BufferedImage` manipulations: we draw the original image `bi` to a new image `newBi` of type `TYPE_USHORT_GRAY`.

Once this is done, you want the image bytes in the PNG format. This is also done using standard `ImageIO` functionality: we just write the `BufferedImage` to a byte array telling `ImageIO` that we want "png".

We can use the resulting bytes to create an `Image` object.

```
Image img = makeBlackAndWhitePng(new PdfImageObject(stream));
```

Now we have an iText `Image` object, but please note that the image bytes as stored in this `Image` object are no longer in the PNG format. As already mentioned in the comments, PNG is not supported in PDF. iText will change the image bytes into a format that is supported in PDF.

### Step 3: replacing the original image stream with the new image stream

We now have an `Image` object, but what we really need is to replace the original image stream with a new one and we also need to adapt the image dictionary as `/DCTDecode` will change into `/FlateDecode`, `/DeviceRGB` will change into `/DeviceGray`, and the value of the `/Length` will also be different.

You are creating the image stream and its dictionary manually. That's brave. I leave this job to iText's `PdfImage` object:

```
PdfImage image = new PdfImage(makeBlackAndWhitePng(new PdfImageObject(stream)), \\n  
    "", null);
```

`PdfImage` extends `PdfStream`, and I can now replace the original stream with this new stream:

```

public static void replaceStream(PRStream orig, PdfStream stream) throws IOException {
    orig.clear();
    ByteArrayOutputStream baos = new ByteArrayOutputStream();
    stream.writeContent(baos);
    orig.setData(baos.toByteArray(), false);
    for (PdfName name : stream.getKeys()) {
        orig.put(name, stream.get(name));
    }
}

```

The order in which you do things here is important. You don't want the `setData()` method to tamper with the length and the filter.

#### Step 4: persisting the document after replacing the stream

I guess it's not hard to figure this part out:

```

replaceStream(stream, image);
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
stamper.close();
reader.close();

```

## How to make an image a qualified mask candidate in iText?



I want to mask an image using another image, for example *A.jpg* should be masked with image *B.jpg*. But when I tried to make the image B a mask, I got the following `DocumentException`:

This image cannot be an image mask

Is there any other way to mask an image using another image in iText? I know how to clip an image but the mask I need can be quite complex, not only rectangles or circles that can be drawn with `ContentByte`.

Posted on StackOverflow on Mar 17, 2015 <sup>[224](#)</sup> by why1905<sup>[225](#)</sup>

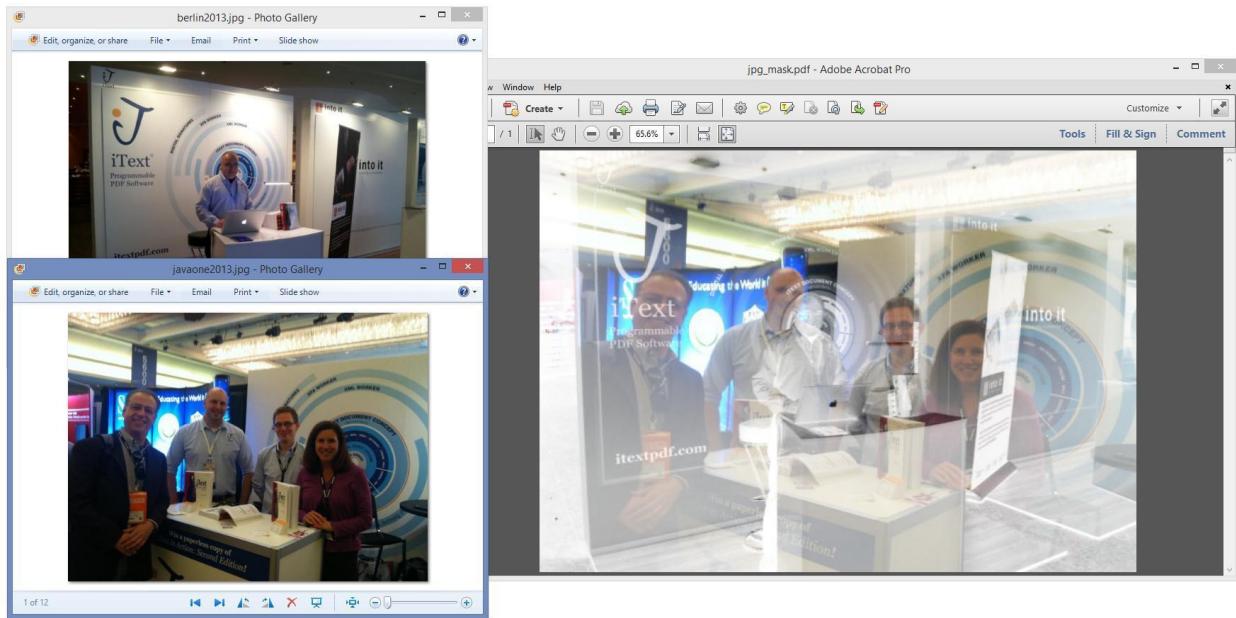
The reason why your code doesn't work is simple: image masks must be monochrome or grayscale; your JPG is a colored image.

---

<sup>224</sup><http://stackoverflow.com/questions/29096314/how-to-make-an-image-a-qualified-mask-candidate-in-itext>

<sup>225</sup><http://stackoverflow.com/users/4353055/why1905>

Please take a look at the [MakeJpgMask<sup>226</sup>](#) example. In this example, I took two normal JPG files and I used one as mask for the other, resulting in a rather spooky PDF: [jpg\\_mask.pdf<sup>227</sup>](#)



Screen shot

To achieve this, I needed to change one colored JPEG into a black and white image:

```
public void createPdf(String dest) throws IOException, DocumentException {
    Document document = new Document(PageSize.A4.rotate());
    PdfWriter writer =
        PdfWriter.getInstance(document, new FileOutputStream(dest));
    document.open();
    Image image = Image.getInstance(IMAGE);
    Image mask = makeBlackAndWhitePng(MASK);
    mask.makeMask();
    image.setImageMask(mask);
    image.scaleAbsolute(PageSize.A4.rotate());
    image.setAbsolutePosition(0, 0);
    document.add(image);
    document.close();
}

public static Image makeBlackAndWhitePng(String image)
    throws IOException, DocumentException {
```

<sup>226</sup><http://itextpdf.com/sandbox/images/MakeJpgMask>

<sup>227</sup>[http://itextpdf.com/sites/default/files/jpg\\_mask.pdf](http://itextpdf.com/sites/default/files/jpg_mask.pdf)

```
BufferedImage bi = ImageIO.read(new File(image));
BufferedImage newBi = new BufferedImage(
    bi.getWidth(), bi.getHeight(), BufferedImage.TYPE USHORT_GRAY);
newBi.getGraphics().drawImage(bi, 0, 0, null);
ByteArrayOutputStream baos = new ByteArrayOutputStream();
ImageIO.write(newBi, "png", baos);
return Image.getInstance(baos.toByteArray());
}
```

As you can see, we have converted `berlin2013.jpg` into a black and white image and we have used this as a mask for the colored `javaone2013.jpg` image.

## How to create unique images with `Image.getInstance()`?



I need to use the `GetInstance()` variant that accepts raw bitmap data:

```
Image.getInstance(int width, int height, int components, int bpc, byte[] data);
```

But if I call it repeatedly, even if the bitmap data is actually different, I get the first instance back instead of a new one. This is a very good feature with, for instance, path-based fixed images but not that good for on-the-fly image generation. How can I guarantee a new bitmap every time?

Posted on StackOverflow on [Nov 21, 2014](#)<sup>228</sup> by Gábor<sup>229</sup>

Please take a look at the [RawImages](#)<sup>230</sup> example. In this example, I create 8 images using the method you mention, one in color space gray, three in color space RGB, four in color space CMYK:

---

<sup>228</sup><http://stackoverflow.com/questions/27070222/how-to-create-unique-images-with-image-getinstance>

<sup>229</sup><http://stackoverflow.com/users/2076286/gábor>

<sup>230</sup><http://itextpdf.com/sandbox/images/RawImages>

```
Image gray = Image.getInstance(1, 1, 1, 8, new byte[] { (byte)0x80 });
gray.scaleAbsolute(30, 30);
Image red = Image.getInstance(1, 1, 3, 8, new byte[] { (byte)255, (byte)0, (byte)\0 });
red.scaleAbsolute(30, 30);
Image green = Image.getInstance(1, 1, 3, 8, new byte[] { (byte)0, (byte)255, (byte)0 });
green.scaleAbsolute(30, 30);
Image blue = Image.getInstance(1, 1, 3, 8, new byte[] { (byte)0, (byte)0, (byte)255, });
blue.scaleAbsolute(30, 30);
Image cyan = Image.getInstance(1, 1, 4, 8, new byte[] { (byte)255, (byte)0, (byt\0), (byte)0 });
cyan.scaleAbsolute(30, 30);
Image magenta = Image.getInstance(1, 1, 4, 8, new byte[] { (byte)0, (byte)255, (\byte)0, (byte)0 });
magenta.scaleAbsolute(30, 30);
Image yellow = Image.getInstance(1, 1, 4, 8, new byte[] { (byte)0, (byte)0, (byt\0), (byte)255, });
yellow.scaleAbsolute(30, 30);
Image black = Image.getInstance(1, 1, 4, 8, new byte[] { (byte)0, (byte)0, (byte)0, (byte)255 });
black.scaleAbsolute(30, 30);
```

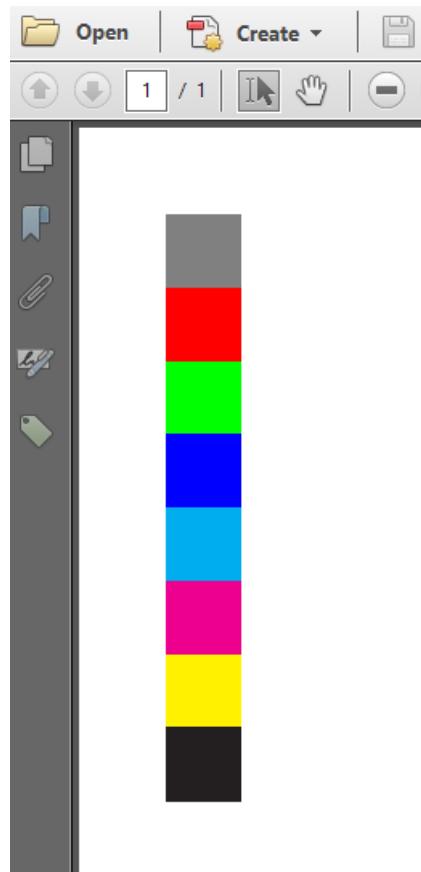
As you can see, each image is exactly one pixel in size, and I chose different byte[] values so that I get pixels in gray, red, green, blue, cyan, magenta, yellow and black. I also scale these image to a bigger size (otherwise it would be difficult to see them).

Now I add the images like this:

```
document.add(gray);
document.add(red);
document.add(green);
document.add(blue);
document.add(cyan);
document.add(magenta);
document.add(yellow);
document.add(black);
document.close();
```

The result does not correspond with what you claim in your question: [raw\\_images.pdf](#)<sup>231</sup>

<sup>231</sup>[http://itextpdf.com/sites/default/files/raw\\_images.pdf](http://itextpdf.com/sites/default/files/raw_images.pdf)



Raw images

There must be another error in your code, but since you don't share any code, nobody can answer your question.



Yes, I stand corrected. I used these images as soft masks and it turned out after much searching that the original images were replicated and forcing the same masks everywhere, not the on-the-fly generated masks themselves. Mea culpa and thanks.

## How to generate 2D barcode as vector image?



I am generating 2D barcodes using iText API, but these barcodes are placed into the PDF document as raster images, hence reducing the quality of the barcode on low resolution printers. As a result, we're unable to scan the barcode. This is our code:

```
BarcodePDF417 pdf417 = new BarcodePDF417();
String text = "BarcodePDF417 barcode";
pdf417.setText(text);
Image img = pdf417.getImage();
document.add(img);
```

I discovered the `placeBarcode()` method that is supposed to create a vector image. I tried using it like this:

```
Rectangle pageSize = new Rectangle(w * 72, h * 72);
Document doc = new Document(pageSize, 1f, 1f, 1f, 1f);
PdfWriter writer = PdfWriter.getInstance(doc, getOutputStream());
doc.open();
PdfContentByte cb = writer.getDirectContent();
BarcodePDF417 pf = new BarcodePDF417();
pf.setText("BarcodePDF417 barcode");
Rectangle rc = pf.getBarcodeSize();
pf.placeBarcode(cb, BaseColor.BLACK, rc.getHeight(), rc.getWidth());
doc.close();
```

This result in a page that is completely black.

Posted on StackOverflow on May 12, 2015 <sup>232</sup> by Dhorrairajj<sup>233</sup>

Please take a look at the [BarcodePlacement<sup>234</sup>](#) example. In this example, we create three PDF417 barcodes:

---

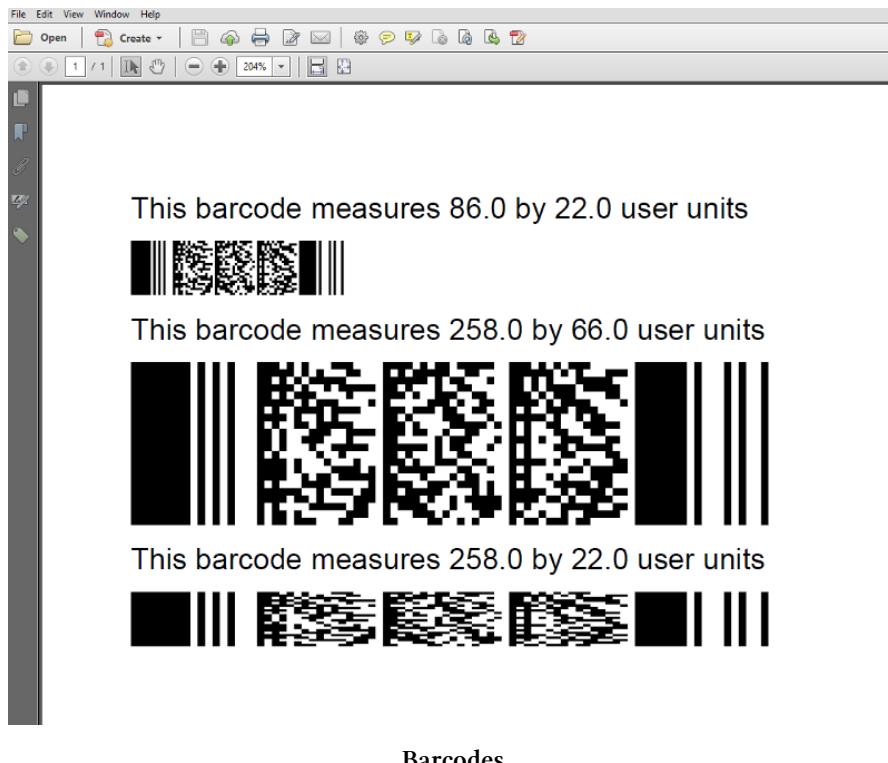
<sup>232</sup><http://stackoverflow.com/questions/30186774/2d-barcode-generation-issue-in-java>

<sup>233</sup><http://stackoverflow.com/users/3752645/dhorrairajj>

<sup>234</sup><http://itextpdf.com/sandbox/barcodes/BarcodePlacement>

```
PdfContentByte cb = writer.getDirectContent();
Image img = createBarcode(cb, "This is a 2D barcode", 1, 1);
document.add(new Paragraph(
    String.format("This barcode measures %s by %s user units",
        img.getScaledWidth(), img.getScaledHeight())));
document.add(img);
img = createBarcode(cb, "This is NOT a raster image", 3, 3);
document.add(new Paragraph(
    String.format("This barcode measures %s by %s user units",
        img.getScaledWidth(), img.getScaledHeight())));
document.add(img);
img = createBarcode(cb, "This is vector data drawn on a PDF page", 1, 3);
document.add(new Paragraph(
    String.format("This barcode measures %s by %s user units",
        img.getScaledWidth(), img.getScaledHeight())));
```

The result looks like this on the outside:



One particular barcode looks like this on the inside:

Key	Value
/Subtype	/Form
/Filter	/FlateDecode
/Type	/XObject
/Matrix	[1, 0, 0, 1, 0, 0]
/FormType	1
/Resources	Dictionary
/BBox	[0, 0, 258, 22]
/Length	2361

### Vector data

I'm adding this *inside* view to show that the 2D barcode is not added as a raster image (as was the case with the initial approach you've tried). It is a vector image consisting of a series of small rectangles. You can check this for yourself by taking a look at the [barcode\\_placement.pdf](#)<sup>235</sup> file.

Please don't be confused because I use an `Image` object. If you look at the `createBarcode()` method, you can see that the `Image` is, in fact, a vector image:

```
public Image createBarcode(PdfContentByte cb, String text,
    float mh, float mw) throws BadElementException {
    BarcodePDF417 pf = new BarcodePDF417();
    pf.setText("BarcodePDF417 barcode");
    Rectangle size = pf.getBarcodeSize();
    PdfTemplate template = cb.createTemplate(
        mw * size.getWidth(), mh * size.getHeight());
    pf.placeBarcode(template, BaseColor.BLACK, mh, mw);
    return Image.getInstance(template);
}
```

The height and the width passed to the `placeBarcode()` method, define the height and the width of the small rectangles that are drawn. If you look at the *inside* view, you can see for instance:

<sup>235</sup>[http://itextpdf.com/sites/default/files/barcode\\_placement.pdf](http://itextpdf.com/sites/default/files/barcode_placement.pdf)

```
0 21 3 1 re
```

This is a rectangle with  $x = 0$ ,  $y = 21$ , width 3 and height 1.

When you ask the barcode for its size, you get the number of rectangles that will be drawn. Hence the dimensions of the barcode is:

```
Rectangle size = pf.getBarcodeSize();
float width = mw * size.getWidth();
float height = mh * size.getHeight();
```

Your assumption that `size` is a size in user units is only correct if `mw` and `mh` are equal to 1.

I use these values to create a `PdfTemplate` instance and I draw the barcode to this *Form XObject*. Most of the times, it's easier to work with the `Image` class than working with `PdfTemplate`, so I wrap the `PdfTemplate` inside an `Image`.

I can then add this `Image` to the document just like any other image. The main difference with *ordinary* images, is that this image is a vector image.

## How to change a background Image into a watermark by altering the opacity?



I want to make my background image in iText transparent

Here is my code for the image:

```
string root = Server.MapPath("~/");
string parent = Path.GetDirectoryName(root);
string grandParent = Path.GetDirectoryName(parent);
string imagePath = parent + "/Images/logo.png";
iTextSharp.text.Image jpg = iTextSharp.text.Image.GetInstance(imagePath);
jpg.ScaleToFit(1700, 800);
jpg.Alignment = iTextSharp.text.Image.UNDERLYING;
jpg.SetAbsolutePosition(100, 250);
jpg.ScaleAbsoluteHeight(500);
jpg.ScaleAbsoluteWidth(500);
```

Any idea?

Posted on StackOverflow on Dec 2, 2014 <sup>236</sup> by dandy<sup>237</sup>

---

<sup>236</sup><http://stackoverflow.com/questions/27241731/change-background-image-in-itext-to-watermark-or-alter-opacity-c-sharp-asp-net>

<sup>237</sup><http://stackoverflow.com/users/4131886/dandy>

Please take a look at the [BackgroundTransparent<sup>238</sup>](#) example. It is a variation on the [BackgroundImage<sup>239</sup>](#) example.

In your code, you're adding the `Image` to the `Document` instance. That's OK, but if you want to make such an image transparent, you need to introduce a soft mask. That's not difficult, but there's an easier way to make your background transparent: add the image to the direct content, and introduce a `PdfGState` defining the opacity:

```
PdfContentByte canvas = writer.getDirectContentUnder();
Image image = Image.getInstance(IMAGE);
image.SetAbsolutePosition(0, 0);
canvas.SaveState();
PdfGState state = new PdfGState();
state.setFillOpacity(0.6f);
canvas.setGState(state);
canvas.addImage(image);
canvas.restoreState();
```

Compare [background\\_image.pdf<sup>240</sup>](#) with [background\\_transparent.pdf<sup>241</sup>](#) to see the difference.

My example is written in Java, but it's very easy to port this to C#:

```
PdfContentByte canvas = writer.DirectContentUnder;
Image image = Image.GetInstance(IMAGE);
image.SetAbsolutePosition(0, 0);
canvas.SaveState();
PdfGState state = new PdfGState();
state.FillOpacity = 0.6f;
canvas.SetGState(state);
canvas.AddImage(image);
canvas.RestoreState();
```

---

<sup>238</sup><http://itextpdf.com/sandbox/images/BackgroundTransparent>

<sup>239</sup><http://itextpdf.com/sandbox/images/BackgroundImage>

<sup>240</sup>[http://itextpdf.com/sites/default/files/background\\_image.pdf](http://itextpdf.com/sites/default/files/background_image.pdf)

<sup>241</sup>[http://itextpdf.com/sites/default/files/background\\_transparent.pdf](http://itextpdf.com/sites/default/files/background_transparent.pdf)

# Absolute positioning of text

In this section, we'll discuss problems that can occur when adding text at absolute positions.

## How to write a Zapfdingbats character at a specific location on a page?



I want to put a check mark using Zapfdingbats on a specific location in my PDF document.

What I achieved so far is this: I can show the check mark but it's on the side of the document and not on the specific X, Y coordinate that I want it to be.

Posted on StackOverflow on May 4, 2013 <sup>242</sup> by δοῦλος Ἰησοῦς Χριστός<sup>243</sup>

Let's start with a Font object that knows how to draw a Zapfdingbats character:

```
Font font = new Font(Font.FontFamily.ZAPFDINGBATS, 12);
```

Once you have a Font object, you can create a Phrase:

```
Phrase phrase = new Phrase(zapfstring, font);
```

Where `zapfstring` is a string containing any Zapfdingbats character you want.

To add this Phrase at an absolute position, you can use the `ShowTextAligned()` method and `PdfWriter`'s *direct content*:

```
PdfContentByte canvas = writer.DirectContent;
ColumnText.ShowTextAligned(canvas, Element.ALIGN_CENTER, phrase, 200, 500, 0);
```

Where 200 and 500 are an X and Y coordinate and 0 is an angle expressed in degrees. Instead of `ALIGN_CENTER`, you can also choose `ALIGN_RIGHT` or `ALIGN_LEFT`.

---

<sup>242</sup><http://stackoverflow.com/questions/16370428/how-to-write-in-a-specific-location-the-zapfdingbatslist-in-a-pdf-document-using>

<sup>243</sup><http://stackoverflow.com/users/2170392/>

## How to reduce redundant code when adding content at absolute positions?



This is part of a vb.net app that uses the itextsharp library:

```
Dim cb As PdfContentByte = writer.DirectContent
cb.BeginText()
cb.SetFontAndSize(Californian, 36)
cb.ShowTextAligned(PdfContentByte.ALIGN_CENTER,
    "CERTIFICATE OF COMPLETION", 396, 397.91, 0)
cbSetFontAndSize(Bold_Times, 22)
cb.ShowTextAligned(PdfContentByte.ALIGN_CENTER, name, 396, 322.35, 0)
cb.SetFontAndSize(Bold_Times, 16)
cb.ShowTextAligned(PdfContentByte.ALIGN_CENTER,
    _hours + " Hours", 297.05, 285.44, 0)
cb.SetFontAndSize(Bold_Times, 16)
cb.ShowTextAligned(PdfContentByte.ALIGN_CENTER,
    _dates, 494.95, 285.44, 0)
cb.SetFontAndSize(Bold_Times, 16)
cb.ShowTextAligned(PdfContentByte.ALIGN_CENTER, _class1, 396, 250.34, 0)
If Not String.IsNullOrWhiteSpace(_class2) Then
    cb.SetFontAndSize(Bold_Times, 16)
    cb.ShowTextAligned(PdfContentByte.ALIGN_CENTER, _class2, 396, 235.34, 0)
End If
cb.SetFontAndSize(Copper, 16)
cb.ShowTextAligned(PdfContentByte.ALIGN_CENTER,
    _conf_num + _prefix + " Annual Conference " + _dates, 396, 193.89, 0)
cb.SetFontAndSize(Bold_Times, 13)
cb.ShowTextAligned(PdfContentByte.ALIGN_CENTER, "Some Name", 396, 175.69, 0)
cb.SetFontAndSize(Bold_Times, 10)
cb.ShowTextAligned(PdfContentByte.ALIGN_CENTER,
    "Some Company Manager", 396, 162.64, 0)
cb.EndText()
```

Plenty of lines in this snippet look awfully redundant and in my opinion, this can't be the cleanest way to do things. Unfortunately, I can't figure out how to create a separate function to which I can simply pass some parameters, such as `string`, `x_Cord`, `y_Cord`, `tilt`. Such a function would then perform the necessary operations on the `PdfContentByte`.

Posted on StackOverflow on Nov 23, 2012 <sup>244</sup> by Skindeep2366<sup>245</sup>

<sup>244</sup><http://stackoverflow.com/questions/13523099/separating-redundant-code-from-pdf-generator-function>

<sup>245</sup><http://stackoverflow.com/users/969487/skindeep2366>

You're adding content "the hard way". If I were you, I'd write a separate class/factory/method that creates either a Phrase or a Paragraph with the content. For instance:

```
protected Font f1 = new Font(Californian, 36);
protected Font f2 = new Font(Bold_times, 16);

public Phrase getCustomPhrase(String name, int hours, ...) {
    Phrase p = new Phrase();
    p.add(new Chunk("...", f1));
    p.add(new Chunk(name, f2));
    ...
    return p;
}
```

Then I would use ColumnText to add the Phrase or Paragraph at the correct position. In the case of a Phrase, I'd use the ColumnText.showTextAligned() method. In the case of Paragraph, I'd use this construction:

```
ColumnText ct = new ColumnText(writer.DirectContent);
ct.setSimpleColumn(rectangle);
ct.addElement(getCustomParagraph(name, hours, ...));
ct.go();
```

The former (using a Phrase) is best if you only need to write one line that doesn't need to be wrapped, oriented in any direction you want.

The latter (using a Paragraph in *composite mode*) is best if you want to add text inside a specific rectangle (defined by the coordinates of the lower-left corner and the upper-right corner).

The approach you've taken works, but... it involves writing PDF syntax almost "manually". That's more difficult and therefore more error-prone. You already discovered that, otherwise you wouldn't ask the question ;-)

## How to truncate text within a bounding box?



I am writing content to a PdfContentByte object directly using:

```
PdfContentByte.showTextAligned()
```

I'd like to know how I can stop the text overflowing a given region when writing. If possible it would be great if iText could also place an ellipsis character where the text does not fit. I can't find any method on ColumnText that will help either. I do not wish the content to wrap when writing.

Posted on StackOverflow on [Nov 26, 2012](#)<sup>246</sup> by Brett Ryan<sup>247</sup>

Use this:

```
int status = ColumnText.START_COLUMN;
ColumnText ct = new ColumnText(cb);
ct.setSimpleColumn(rectangle);
status = ct.go();
```

Make sure that you define rectangle in a way so that only one line fits, use ColumnText.hasMoreText(status) to find out if you need to add an ellipsis character.

## How to continue an ordered list on a second page?



I am currently creating the document by creating ColumnText objects that are placed at specific coordinates. The content consists of variable data (text/barcodes/images) that is added to existing PDF documents/templates (think boiler plate). Most commonly, I have to place various sections of text in specific places.

I know how to create an ordered list, but I have come across a situation where the list begins with #1 on the first page and then #2-4 on the top of the second page. I use two different templates for p1 and p2.

However, when I start a new list for the part on the second page, iText starts numbering that new list from 1 instead of continuing with 2.

Posted on StackOverflow on [Mar 26, 2015](#)<sup>248</sup> by rcurrydev<sup>249</sup>

<sup>246</sup><http://stackoverflow.com/questions/13558135/how-can-i-truncate-text-within-a-bounding-box>

<sup>247</sup><http://stackoverflow.com/users/140037/brett-ryan>

<sup>248</sup><http://stackoverflow.com/questions/29277611/itextsharp-continuing-ordered-list-on-second-page-with-a-number-other-than-1>

<sup>249</sup><http://stackoverflow.com/users/1859781/rcurrydev>

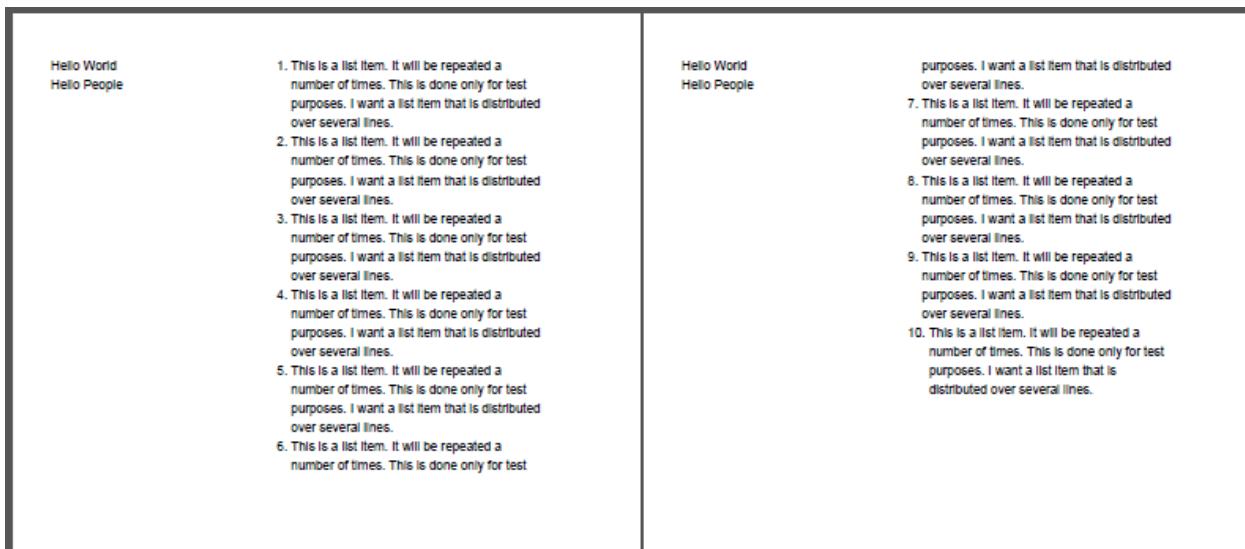
There are two answers to your question. The first one is to point you to the official documentation. There is a method `setFirst()`<sup>250</sup> that (I quote) *sets the number that has to come first in the list*.

You are using the C# port of iText, so if you want the list to start counting at 10, you need to do something like:

```
list.First = 10;
```

The second answer takes more time, but it is probably the better one. You don't need two List objects, one for the first page and one for the second page. It's better to add the List to a ColumnText object and then distribute the column over two pages.

Take a look at the [ListInColumn](#)<sup>251</sup> example. It takes an existing PDF (with the text "Hello World Hello People") and it adds a list using ColumnText: [list\\_in\\_column.pdf](#)<sup>252</sup>



Screen shot

This is how it's done:

<sup>250</sup>[http://api.itextpdf.com/itext/com/itextpdf/text/List.html#setFirst\(int\)](http://api.itextpdf.com/itext/com/itextpdf/text/List.html#setFirst(int))

<sup>251</sup>[<sup>252</sup>\[http://itextpdf.com/sites/default/files/list\\\_in\\\_column.pdf\]\(http://itextpdf.com/sites/default/files/list\_in\_column.pdf\)](http://itextpdf.com/sandbox/objects>ListInColumn</a></p>
</div>
<div data-bbox=)

```
PdfReader reader = new PdfReader(src);
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
List list = new List(List.ORDERED);
for (int i = 0; i < 10; i++) {
    list.add("...");}
ColumnText ct = new ColumnText(stamper.getOverContent(1));
ct.addElement(list);
Rectangle rect = new Rectangle(250, 400, 500, 806);
ct.setSimpleColumn(rect);
int status = ct.go();
if (ColumnText.hasMoreText(status)) {
    ct.setCanvas(stamper.getOverContent(2));
    ct.setSimpleColumn(rect);
    ct.go();}
stamper.close();
```

To add the content on the first page, I use:

```
ColumnText ct = new ColumnText(stamper.getOverContent(1));
```

You are probably using similar code.

The content is added using the line:

```
int status = ct.go();
```

If not all the content was added, I change the canvas to add the rest of the content on the second page:

```
ct.setCanvas(stamper.getOverContent(2));
```

The rest of the code is pretty standard.

I think the `setCanvas()` method is the missing piece in your puzzle, although in your case, you'll need:

```
ct.Canvas = stamper.GetOverContent(2);
```

## Why does ColumnText ignore the horizontal alignment?



I'm trying to get some rows of text on the left side and some on the right side. For some reason iText seems to ignore the alignment entirely. For example:

```
// create 200x100 column
ct = new ColumnText(writer.DirectContent);
ct.SetSimpleColumn(0, 0, 200, 100);
ct.AddElement(new Paragraph("entry1"));
ct.AddElement(new Paragraph("entry2"));
ct.AddElement(new Paragraph("entry3"));
ret = ct.Go();

ct.SetSimpleColumn(0, 0, 200, 100);
ct.Alignment = Element.ALIGN_RIGHT;
ct.AddElement(new Paragraph("entry4"));
ct.AddElement(new Paragraph("entry5"));
ct.AddElement(new Paragraph("entry6"));
ret = ct.Go();
```

I've set the alignment of the 2nd column to `Element.ALIGN_RIGHT` but the text appears printed on top of column one, rendering unreadable text. Like the alignment was still set to left

Posted on StackOverflow on [Aug 9, 2013](#)<sup>253</sup> by [Chuck](#)<sup>254</sup>

To understand what happens, you should learn about the concepts “*text mode*” and “*composite mode*”.

If you work in **text mode**, you can define the alignment at the level of the `ColumnText` object. In other words `ct.Alignment = Element.ALIGN_RIGHT;` will work in text mode.

If you work in **composite mode**, the alignment at the column level will be ignored in favor of the alignment of the elements added to the column. In your case, iText will ignore the `ALIGN_RIGHT` in favor of the alignment of the `Paragraph` objects added to the column. Looking at your code, I see that you didn't define an alignment for the paragraphs, so the default alignment `ALIGN_LEFT` is used.

How do you know if you're working in *text mode* or in *composite mode*?

By default, `ColumnText` uses *text mode* but it switches to *composite mode* (removing all previously added text) the moment you invoke the `AddElement()` method.

<sup>253</sup><http://stackoverflow.com/questions/18142623/itext-columntext-ignores-alignment>

<sup>254</sup><http://stackoverflow.com/users/1280511/chuck>

The concepts *text mode* and *composite mode* also applies to PdfPCell.

## How to fit a String inside a rectangle?



I'm trying to add some strings, images and tables into my pdf file (there have to be several pages) but when I try to use ColumnText (I use this because I want to place strings at absolute positions), I encounter a problem. When the column height is not sufficient to add the content of the strings, the content is incomplete. How can I avoid that content gets lost?

Here's the related code :

```

PdfContentByte cb = writer.getDirectContent();
Image imageust = Image.getInstance(imageUrl);
Image imageAlt = imageAlt = Image.getInstance(imageUrlAlt);
imageust.setAbsolutePosition(0f,
    document.getPageSize().getHeight() - imageust.getHeight()-10);
imageAlt.setAbsolutePosition(0f, 10f);
document.add(imageust);
document.add(imageAlt);
// now draw a line below the headline
cb.setLineWidth(1f);
cb.moveTo(0, 200);
cb.lineTo(200, 200);
cb.stroke();
// first define a standard font for our text
Font helvetica8BoldBlue = FontFactory.getFont(FontFactory.HELVETICA,16);
// create a column object
ColumnText ct = new ColumnText(cb);
// define the text to print in the column
Phrase myText = new Phrase("Very Very Long String!!!", helvetica8BoldBlue);
ct.setSimpleColumn(myText, 60, 750,
/* width*/document.getPageSize().getWidth() - 40, 100,
20, Element.ALIGN_LEFT);
ct.go();

```

Posted on StackOverflow on Nov 23, 2012<sup>255</sup> by Adnan Bal<sup>256</sup>

There are three options:

1. Either you provide a bigger rectangle, so that the content fits inside,

---

<sup>255</sup><http://stackoverflow.com/questions/13526043/itext-string-fitting>

<sup>256</sup><http://stackoverflow.com/users/1763734/adnan-bal>

2. or you reduce the content (e.g. smaller font, less text),...
3. or you keep the size of the rectangle, keep the font size, etc... but add the content that doesn't fit on the next page.

How do you know if the content doesn't fit?

You can add the content in simulation mode first, and test if all the content was 'consumed':

```
int status = ct.go(true);
boolean fits = !ColumnText.hasMoreText(status);
```

Based on the value of `fits`, you can decide to change the size of the rectangle or the content.

If you can distribute the content over different pages, you don't need simulation mode, you just need to insert a document.newPage();

```
ColumnText ct = new ColumnText(cb);
ct.setSimpleColumn(rect);
int status = ct.go();
while (ColumnText.hasMoreText(status)) {
    document.newPage();
    ct.setSimpleColumn(rect);
    status = ct.go();
}
```

In this example `rect` contains the coordinates of the rectangle.

## Why does using ColumnText result in “The document has no pages” exception?



Question 1: I want to use ColumnText to wrap text in a rectangle:

```
Document document = new Document(PageSize.A4.rotate());
ByteArrayOutputStream baos = new ByteArrayOutputStream();
PdfWriter writer = PdfWriter.getInstance(document, baos);
document.open();
ColumnText column = new ColumnText(writer.getDirectContent());
column.setSimpleColumn(new Phrase("text is very long ..."), 10, 10, 20, 20, 18, 
Element.ALIGN_CENTER);
column.go();
document.close();
```

When I run this code, I get the following error:

ExceptionConverter: java.io.IOException: The document has no pages.

Do you have any suggestions how to fix this problem?

Posted on StackOverflow on Sep 23, 2014 <sup>257</sup> by Han Kun<sup>258</sup>

I see that you have the following line:

```
column.go();
```

You did not use something like this:

```
int status = column.go();
```

If you did, and if you examined `status`, you would have noticed that the `column` object still contained some text.

What text? *All* the text.

There is a serious error in this line:

---

<sup>257</sup><http://stackoverflow.com/questions/25991936/using-columntext-results-in-the-document-has-no-pages-exception>

<sup>258</sup><http://stackoverflow.com/users/4040263/han-kun>

```
column.setSimpleColumn(new Phrase("text is very long ..."), 10, 10, 20, 20, 18, \
Element.ALIGN_CENTER);
```

You are trying to add the text "text is very long ..." into a rectangle with the following coordinates:

```
float l1x = 10;
float l1y = 10;
float urx = 20;
float ury = 20;
```

You didn't define a font, so the font is Helvetica with font size 12pt and you defined a leading of 18pt.

This means that you are trying to fit text that is 12pt height with an extra 6pt for the leading into a square that measures 10 by 10 pt. Surely you understand that this can't work!

As a result, nothing is added to the PDF and rather than showing an empty page, iText throws an exception saying: there are no pages! *You didn't add any content to the document!*

You can fix this, for instance by changing the incorrect line into something like this:

```
column.setSimpleColumn(new Phrase("text is very long ..."), 36, 36, 559, 806, 18\
, Element.ALIGN_CENTER);
```

An alternative would be:

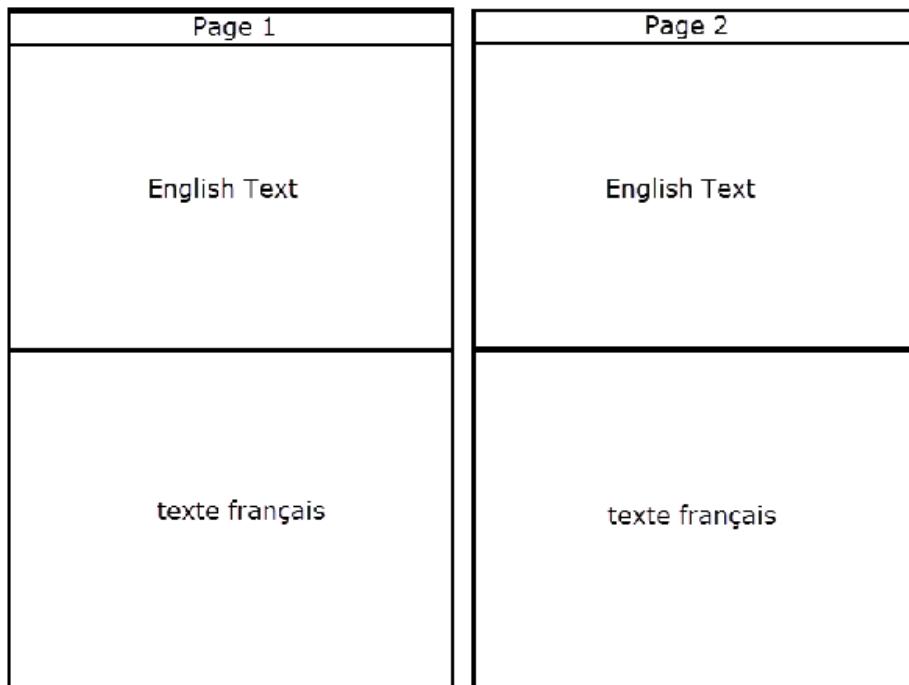
```
column.setSimpleColumn(rect);
column.addElement(paragraph);
```

In these two lines `rect` is a `Rectangle` object. The leading and the alignment are to be defined at the level of the `Paragraph` object (in this case, you don't use a `Phrase`).

## How to divide a page in N parts so we can fill each with a different source?



I need to create an User guide, where I've to put the content in 2 different languages but on the same page. So the first half of the page would be in English while the second part would be in French like this:



Example

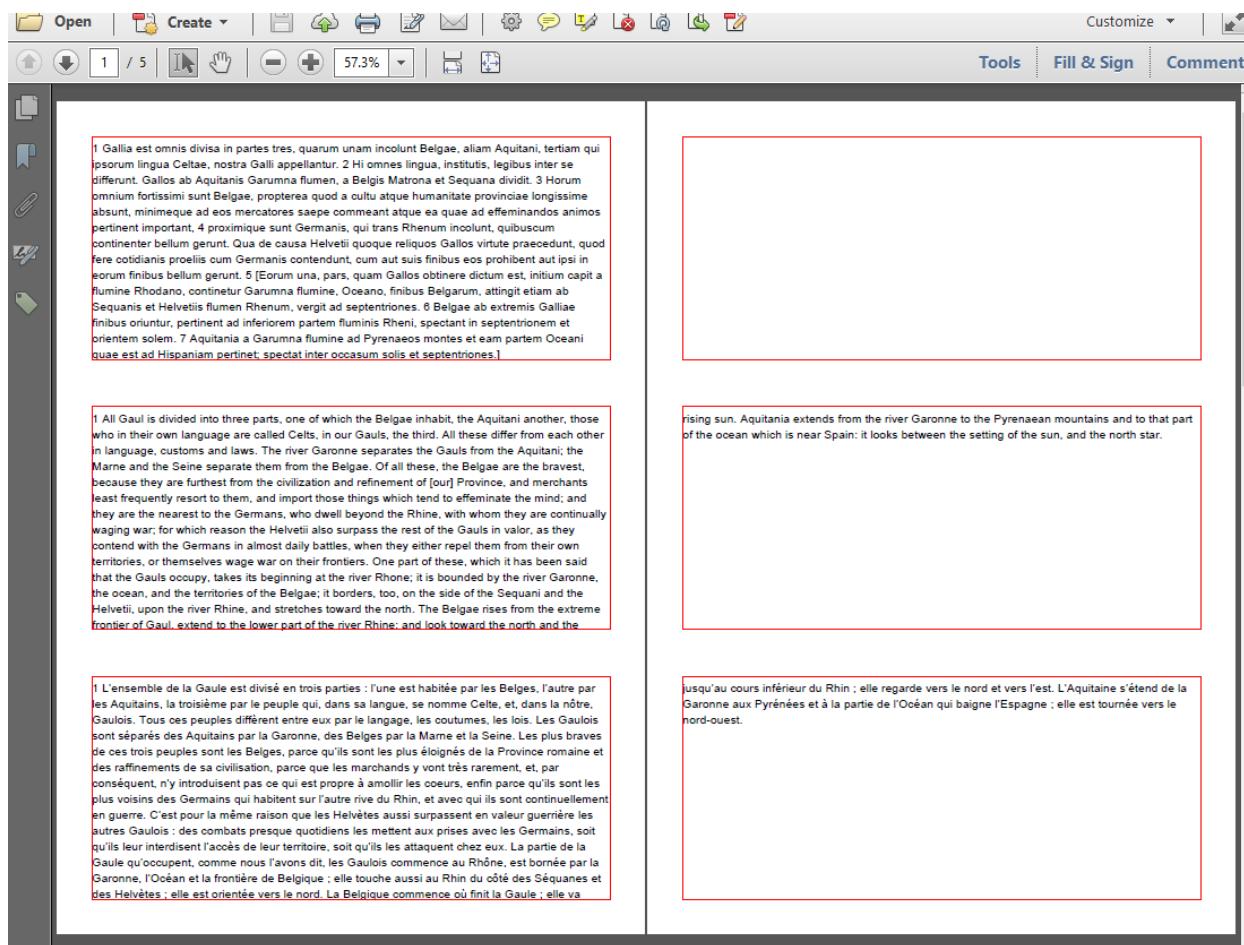
In the future they might ask for 3rd language also, but maximum 3. Right now each page would have 2 blocks. How can I achieve this using iTextPDF in java?

Posted on StackOverflow on Feb 13, 2012<sup>259</sup> by IT ppl<sup>260</sup>

If I understand your question correctly, you need to create something like this:

<sup>259</sup><http://stackoverflow.com/questions/28502315/divide-page-in-2-parts-so-we-can-fill-each-with-different-source>

<sup>260</sup><http://stackoverflow.com/users/1285273/it-ppl>



### Example

In this screen shot, you see the first part of the first book of Caesar's Commentaries on the Gallic War. *Gallia omnia est divisa in partes tres*, and so is each page in this document: the upper part shows the text in Latin, the middle part shows the text in English, the lower part shows the text in French. If you read the text, you'll discover that Belgians like me are considered being the bravest of all (although we aren't as civilized as one would wish). See [three\\_parts.pdf<sup>261</sup>](#) if you want to take a look at the PDF.

This PDF was created with the [ThreeParts<sup>262</sup>](#) example. In this example, I have 9 text files: [liber1\\_1\\_la.txt<sup>263</sup>](#), [liber1\\_1\\_en.txt<sup>264</sup>](#), [liber1\\_1\\_fr.txt<sup>265</sup>](#), [liber1\\_2\\_la.txt<sup>266</sup>](#), [liber1\\_2\\_en.txt<sup>267</sup>](#), [liber1\\_2\\_fr.txt<sup>268</sup>](#),

<sup>261</sup>[http://itextpdf.com/sites/default/files/three\\_parts.pdf](http://itextpdf.com/sites/default/files/three_parts.pdf)

<sup>262</sup><http://itextpdf.com/sandbox/events/ThreeParts>

<sup>263</sup>[http://itextpdf.com/sites/default/files/liber1\\_1\\_la.txt](http://itextpdf.com/sites/default/files/liber1_1_la.txt)

<sup>264</sup>[http://itextpdf.com/sites/default/files/liber1\\_1\\_en.txt](http://itextpdf.com/sites/default/files/liber1_1_en.txt)

<sup>265</sup>[http://itextpdf.com/sites/default/files/liber1\\_1\\_fr.txt](http://itextpdf.com/sites/default/files/liber1_1_fr.txt)

<sup>266</sup>[http://itextpdf.com/sites/default/files/liber1\\_2\\_la.txt](http://itextpdf.com/sites/default/files/liber1_2_la.txt)

<sup>267</sup>[http://itextpdf.com/sites/default/files/liber1\\_2\\_en.txt](http://itextpdf.com/sites/default/files/liber1_2_en.txt)

<sup>268</sup>[http://itextpdf.com/sites/default/files/liber1\\_2\\_fr.txt](http://itextpdf.com/sites/default/files/liber1_2_fr.txt)

`liber1_3_la.txt`<sup>269</sup>, `liber1_3_en.txt`<sup>270</sup>, and `liber1_3_fr.txt`<sup>271</sup>.

Liber is the latin word for book, so all files are snippets from the first book, more specifically sections 1, 2, and 3, in Latin, English and French.

This is how I defined the languages and the rectangles for each language:

```
public static final String[] LANGUAGES = { "la", "en", "fr" };
public static final Rectangle[] RECTANGLES = {
    new Rectangle(36, 581, 559, 806),
    new Rectangle(36, 308.5f, 559, 533.5f),
    new Rectangle(36, 36, 559, 261) };
```

In my code, I loop over the different sections, and I create a `ColumnText` object for each language:

```
PdfContentByte cb = writer.getDirectContent();
ColumnText[] columns = new ColumnText[3];
for (int section = 1; section <= 3; section++) {
    for (int la = 0; la < 3; la++) {
        columns[la] = createColumn(cb, section, LANGUAGES[la], RECTANGLES[la]);
    }
    while (addColumns(columns)) {
        document.newPage();
        for (int la = 0; la < 3; la++) {
            columns[la].setSimpleColumn(RECTANGLES[la]);
        }
    }
    document.newPage();
}
```

If you examine the body of the inner loop, you see that I first define three `ColumnText` objects, one for each language:

---

<sup>269</sup>[http://itextpdf.com/sites/default/files/liber1\\_3\\_la.txt](http://itextpdf.com/sites/default/files/liber1_3_la.txt)

<sup>270</sup>[http://itextpdf.com/sites/default/files/liber1\\_3\\_en.txt](http://itextpdf.com/sites/default/files/liber1_3_en.txt)

<sup>271</sup>[http://itextpdf.com/sites/default/files/liber1\\_3\\_fr.txt](http://itextpdf.com/sites/default/files/liber1_3_fr.txt)

```
public ColumnText createColumn(
    PdfContentByte cb, int i, String la, Rectangle rect)
throws IOException {
    ColumnText ct = new ColumnText(cb);
    ct.setSimpleColumn(rect);
    Phrase p = createPhrase(
        String.format("resources/text/liber1_%s_%s.txt", i, la));
    ct.addText(p);
    return ct;
}
```

In this case, I'm using `ColumnText` in *text mode*, and I read the text from the different files into a `Phrase` like this:

```
public Phrase createPhrase(String path) throws IOException {
    Phrase p = new Phrase();
    BufferedReader in = new BufferedReader(
        new InputStreamReader(new FileInputStream(path), "UTF8"));
    String str;
    while ((str = in.readLine()) != null) {
        p.add(str);
    }
    in.close();
    return p;
}
```

Once I have defined the `ColumnText` objects and added their content, I need to render the content to one or more pages until all the text is rendered from all columns. To achieve this, we use this method:

```
public boolean addColumns(ColumnText[] columns) throws DocumentException {
    int status = ColumnText.NO_MORE_TEXT;
    for (ColumnText column : columns) {
        if (ColumnText.hasMoreText(column.go()))
            status = ColumnText.NO_MORE_COLUMN;
    }
    return ColumnText.hasMoreText(status);
}
```

As you can see, I also create a new page for every new section I start. This isn't really necessary: I could add all the section to a single `ColumnText`, but depending on how the Latin text translated into English and French, you could end up with large discrepancies where section X of the Latin text starts on one page and the same section in English or French starts on another page. Hence my choice to start a new page, although it's not really necessary in this small proof of concept.

## How to rotate a single line of text?



I'm converting a document which is created with a online editor. I need to be able to rotate the text using it's central point and not (0,0).

Since Image rotates using the centre I assumed this would work but it doesn't. Anyone has a clue how to rotate text using the central point in itext?

```
float fw = bf.getWidthPoint(text, textSize);
float fh = bf.getAscentPoint(text, textSize)
    - bf.getDescentPoint(text, textSize);
PdfTemplate template = content.createTemplate(fw, fh);

Rectangle r = new Rectangle(0,0,fw, fw);
r.setBackgroundColor(BaseColor.YELLOW);
template.rectangle(r);

template.setFontAndSize(bf, 12);
template.beginText();
template.moveText(0, 0);
template.showText(text);
template.endText();

Image tmpImage = Image.getInstance(template);
tmpImage.setAbsolutePosition(Utilities.millimetersToPoints(x),
    Utilities.millimetersToPoints(
        pageHeight - (y + Utilities.pointsToMillimeters(fh)))); 
tmpImage.setRotationDegrees(0);
document.add(tmpImage);
```

Posted on StackOverflow on Aug 1, 2013<sup>272</sup> by user1236552<sup>273</sup>

Why are you using beginText(), moveText(), showText(), endText()? Those methods are to be used by developers who speak PDF syntax fluently.

There an easier way to achieve what you want. Just use the static showTextAligned() method that is provided in the ColumnText class. That way you don't even need to use beginText(), setFontAndSize(), endText():

---

<sup>272</sup><http://stackoverflow.com/questions/17998306/rotating-text-using-center-in-itext>

<sup>273</sup><http://stackoverflow.com/users/1236552/user1236552>

```
ColumnText.showTextAligned(template,  
    Element.ALIGN_CENTER, new Phrase(text), x, y, rotation);
```

Using the `showTextAligned()` method in `ColumnText` also has the advantage that you can use a phrase that contains chunks with different fonts.

## How to rotate a paragraph?



I have a web site where the users upload photos and create photobooks. Also, they can add text at absolute positions, rotations, and alignments. The text can have new lines.

I've been using the iText Library to automatize the creation of the Photobooks Hight Quality Pdfs that are printed latter on.

Adding the user uploaded images to the PDFs was really simple, the problem comes when I try to add the text.

In theory what I would need to do, is to define a paragraph of some defined width and height, set the users text, font, font style, alignment (center, left, right, justify), and finally set the rotation.

For what I've read about Itext, I could create a Paragraph, set the user properties, and use a `ColumnText` object to set the absolute position, width and height. However it's not possible to set the rotation of anything bigger than single line.

I cant use table cells either, because the rotation method only allow degrees that are multiples of 90.

Is there a way to add a paragraph with some rotation (say 20 degrees) without having to add the text line by line using the `ColumnText.showTextAligned()` method and all math that involves?

Posted on StackOverflow on Mar 14, 2013 <sup>274</sup> by BernalCarlos<sup>275</sup>

You can do this in a couple of steps:

- Create a `PdfTemplate` object; just a rectangle.
- Draw your `ColumnText` on this `PdfTemplate`; don't worry about the rotation, just fill the rectangle with whatever content you want to add to the column.
- Wrap the `PdfTemplate` inside an `Image` object; this is just for convenience, to avoid the math. This doesn't mean your text will be rasterized.
- Now apply a rotation and an absolute position to the `Image` and add it to your document.

<sup>274</sup><http://stackoverflow.com/questions/15414923/rotate-paragraphs-or-cells-some-arbitrary-number-of-degrees-itext>

<sup>275</sup><http://stackoverflow.com/users/1886508/bernalcarlos>

Your problem is now solved ;-)



If it helps anyone, this is the code i used to solve this problem (thanks to Bruno):

```
//Create the template that will contain the text
PdfContentByte canvas = pdfWriter.getDirectContent();
PdfTemplate textTemplate = canvas.createTemplate(imgWidth, imgHeight);

ColumnText columnText = new ColumnText(textTemplate);

columnText.setSimpleColumn(0, 0, imgWidth, imgHeight);
columnText.addElement(paragraph);

columnText.go();

//Create de image wrapper for the template
Image textImg = Image.getInstance(textTemplate);

//Assign the dimentions of the image, in this case, the text
textImg.setInterpolation(true);
textImg.scaleAbsolute(imgWidth, imgHeight);
textImg.setRotationDegrees((float) -textComp.getRotation());
textImg.setAbsolutePosition(imgXPos, imgYPos);

//Add the text to the pdf
pdfDocument.add(textImg);
```

## How to draw a rectangle around multiline text?



I am trying to draw a rectangle around multi-line text in iText. I use this code to draw the text:

```
ColumnText ct = new ColumnText(cb);
Phrase phrase = new Phrase("Some String\nOther string etc...\\n test");
ct.setSimpleColumn(myText.....);
ct.addElement(phrase);
ct.go();
```

I know how to draw a rectangle, but I am not able to draw a rectangle outlining this text.

Posted on StackOverflow on Mar 13, 2012<sup>276</sup> by user2439522<sup>277</sup>

It sounds as if you are missing only a single piece of the puzzle to meet your requirement. That piece is called `getYLine()`.

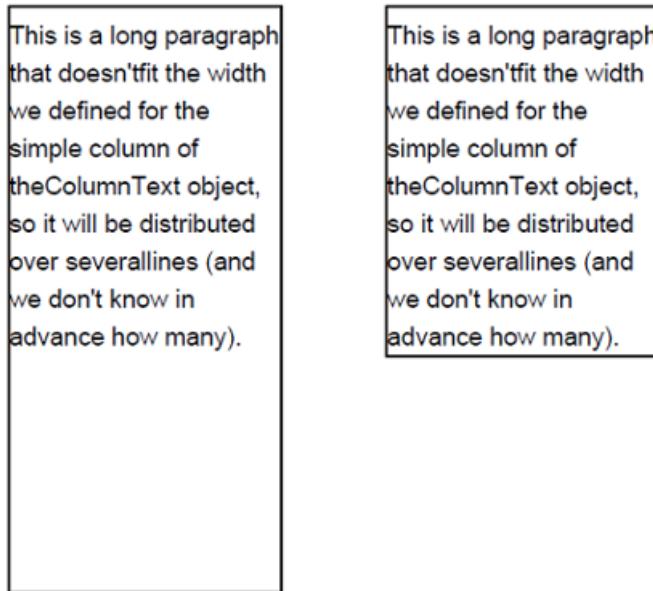
Please take a look at the [DrawRectangleAroundText<sup>278</sup>](#) example. This example draws the same paragraph twice. The first time, it adds a rectangle that probably looks like the solution you already have. The second time, it adds a rectangle the way you want it to look:

---

<sup>276</sup><http://stackoverflow.com/questions/29037981/how-to-draw-a-rectangle-around-multiline-text>

<sup>277</sup><http://stackoverflow.com/users/2439522/user2439522>

<sup>278</sup><http://itextpdf.com/sandbox/objects/DrawRectangleAroundText>



Screen shot

The first time, we add the text like this:

```
ColumnText ct = new ColumnText(cb);
ct.setSimpleColumn(120f, 500f, 250f, 780f);
Paragraph p = new Paragraph("This is a long paragraph that doesn't"
    + "fit the width we defined for the simple column of the"
    + "ColumnText object, so it will be distributed over several"
    + "lines (and we don't know in advance how many).");
ct.addElement(p);
ct.go();
```

You define your column using the coordinates:

```
l1x = 120;
l1y = 500;
urx = 250;
ury = 780;
```

This is a rectangle with lower left corner (120, 500), a width of 130 and a height of 380. Hence you draw a rectangle like this:

```
cb.rectangle(120, 500, 130, 280);
cb.stroke();
```

Unfortunately, that rectangle is too big.

Now let's add the text once more at slightly different coordinates:

```
ct = new ColumnText(cb);
ct.setSimpleColumn(300f, 500f, 430f, 780f);
ct.addElement(p);
ct.go();
```

Instead of using (300, 500) as lower left corner for the rectangle, we ask the ct object for its current Y position using the `getYLine()` method:

```
float endPos = ct.getYLine() - 5;
```

As you can see, I subtract 5 user units, otherwise the bottom line of my rectangle will coincide with the baseline of the final line of text and that doesn't look very nice. Now I can use the endPos value to draw my rectangle like this:

```
cb.rectangle(300, endPos, 130, 780 - endPos);
cb.stroke();
```

## What is causing syntax errors in a page created with iText?



I am getting the following error when I want to print a PDF generated with iTextSharp”

“An error exists on this page. Acrobat may not display the page correctly.  
please contact the person who created the pdf document to correct the  
problem”

The document prints fine, but why do I get this error? This is my code:

```
PdfContentByte cb = writer.DirectContent;
cb.BeginText();
Font NormalFont = FontFactory.GetFont("Arial", 12, Font.NORMAL, Color.BLACK);
iTextSharp.text.Image img = iTextSharp.text.Image.GetInstance("/banner.tiff");
img.SetAbsolutePosition(35, 760);
img.ScalePercent(50);
cb.AddImage(img);
cb.SetLineWidth(2);
cb.MoveTo(20, 740);
cb.LineTo(570, 740);
cb.Stroke();
cb.BeginText();
writeText(cb, drHead["EmpName"].ToString(), 25, 745, f_cb, 14);
writeText(cb, "Employee ID:", 450, 745, f_cn, 12);
writeText(cb, drHead["EmployeeID"].ToString(), 515, 745, f_cb, 12);
cb.EndText();
cb.BeginText();
writeText(cb, "XXXX:", 25, 725, f_cb, 8);
cb.EndText();
cb.SetLineWidth(2);
cb.MoveTo(20, 675);
cb.LineTo(570, 675);
cb.Stroke();
cb.EndText();
cb.BeginText();
writeText(cb, "XXXXXXXXXXXXXXXXXX", 20, 140, f_cb, 12);
cb.EndText();
cb.EndText();
```

Please help me to know what is the issue.

Posted on StackOverflow on Jan 23, 2014 <sup>279</sup> by Vandana<sup>280</sup>

---

<sup>279</sup><http://stackoverflow.com/questions/21301497/pdfs-generated-using-itextsharp-giving-error-at-the-time-of-first-print-command>

You have nested text blocks. That's illegal PDF syntax. I think recent versions of iTextSharp warn you about this, so I guess you're using an old version.

This is wrong:

```
cb.BeginText();
...
cb.BeginText();
...
cb.EndText();
...
cb.EndText();
```

This is right:

```
cb.BeginText();
...
cb.EndText();
...
cb.BeginText();
...
cb.EndText();
```

Moreover: ISO-32000-1 tells you that some operations are forbidden inside a text block.

This is wrong:

```
cb.BeginText();
...
cb.AddImage(img);
...
cb.EndText();
```

This is right:

---

<sup>280</sup><http://stackoverflow.com/users/2273006/vandana>

```
cb.BeginText();
...
cb.EndText();
...
cb.AddImage(img);
```

Finally, some operators are mandatory when creating a text block. For instance: you always need `setFontAndSize()` (I don't know what you're doing in `writeText()`, but I assume you're setting the font correctly).

In any case: you have chosen to use iTextSharp at the lowest level, writing PDF syntax almost manually. This assumes that you know ISO-32000-1 inside-out. If you don't, you should use some of the high-level objects, such as `ColumnText` to position content at absolute positions.

## How to adapt the position of the first line in `ColumnText`?



I'm dealing with a situation where I have a `Phrase` added to a `ColumnText` object.



Text alignment issue

The title in black is where iText is placing the text of the `Phrase` within the `ColumnText`. The title in pink is the desired placement. Is there anything I can do to tell iText not to put any space between the top of the highest glyphs (D and Z in this case) and the top of the `ColumnText` box?

Posted on StackOverflow on January 12, 2015<sup>281</sup> by Locriansax<sup>282</sup>

Please take a look at the [ColumnTextAscender<sup>283</sup>](#) example:

<sup>281</sup><http://stackoverflow.com/questions/27906725/itext-placement-of-phrase-within-columntext>

<sup>282</sup><http://stackoverflow.com/users/890192/locriansax>

<sup>283</sup><http://itextpdf.com/sandbox/objects/ColumnTextAscender>



### ColumnText positioning

In both cases, I have drawn a red rectangle:

```
rect.setBorder(Rectangle.BOX);
rect.setBorderWidth(0.5f);
rect.setBorderColor(BaseColor.RED);
PdfContentByte cb = writer.getDirectContent();
cb.rectangle(rect);
```

To the left, you see the text added the way you do. In this case, iText will add extra space commonly known as the *leading*. To the right, you see the text added the way you want to add it. In this case, we have told the ColumnText that it needs to use the Ascender value of the font:

```
Phrase p = new Phrase("This text is added at the top of the column.");
ColumnText ct = new ColumnText(cb);
ct.setSimpleColumn(rect);
ct.setUseAscender(true);
ct.addText(p);
ct.go();
```

Now the top of the text touches the border of the rectangle.

# Absolute positioning of lines and shapes

You can draw lines and shapes at absolute positions using low-level methods.

## How to add a border to a PDF page?



This is a snippet of my source code:

```
Rectangle rect= new Rectangle(36,108);
rect.enableBorderSide(1);
rect.enableBorderSide(2);
rect.enableBorderSide(4);
rect.enableBorderSide(8);
rect.setBorder(2);
rect.setBorderColor(BaseColor.BLACK);
document.add(rect);
```

Why am I not able to add border to my pdf page even after enabling borders for all sides?  
I've set border and its color too still I'm not able to add border.

Posted on StackOverflow on Jul 11, 2014 <sup>284</sup> by user3819936<sup>285</sup>

You didn't define a border width. You can fix this by adding:

```
rect.setBorder(Rectangle.BOX);
rect.setBorderWidth(2);
```

Note that I would remove the enableBorderSide() calls. You'll also notice that you've used the setBorder() method in the wrong way.

---

<sup>284</sup><http://stackoverflow.com/questions/24692950/add-border-to-pdf-page-using-itext>

<sup>285</sup><http://stackoverflow.com/users/3819936/user3819936>

## How to create a PDF with a Cartesian grid?



Could anyone please provide me a sample program that can dynamically create a grid or even dots starting at (0,0) at bottom left corner on a PDF of page size "Letter"? (max X = 8.5 Inches; max Y = 11 Inches)

Posted on StackOverflow on Jun 10, 2014 <sup>286</sup> by user3727496<sup>287</sup>

Please take a look at the [Grid<sup>288</sup>](#) example. In this example, I define the pagesize variable like this:

```
Rectangle pagesize = PageSize.LETTER;
```

I use this variable to create the Document instance, and I also use it in the loops that draw the grid:

```
PdfContentByte canvas = writer.getDirectContent();
for (float x = 0; x < pagesize.getWidth(); ) {
    for (float y = 0; y < pagesize.getHeight(); ) {
        canvas.circle(x, y, 1f);
        y += 72f;
    }
    x += 72f;
}
canvas.fill();
```

In this case, I increment x and y with 72 user units. This means that the distance between the dots will be 1 inch.

---

<sup>286</sup><http://stackoverflow.com/questions/24149640/creating-a-pdf-with-a-cartesian-grid-using-itext>

<sup>287</sup><http://stackoverflow.com/users/3727496/user3727496>

<sup>288</sup><http://itextpdf.com/sandbox/objects/Grid>

## Why does the cell background color affect the color of other lines?



I'm creating a PDF where I add some text to each page as well as 2 lines that are drawn using the following method:

```
private void DrawLines(Document pdfDoc, PdfContentByte cb) {
    cb.MoveTo(0, 562);
    cb.LineTo(pdfDoc.PageSize.Width, 562);
    cb.MoveTo(0, 561);
    cb.LineTo(pdfDoc.PageSize.Width, 561);
}
```

On one specific page, there's a table where I'm using the following code to change the background color for one particular cell:

```
header = new PdfPCell(new Phrase(market_data_list[i], grid_data_heading));
header.Colspan = 2;
header.HorizontalAlignment = iTextSharp.text.Element.ALIGN_CENTER;
header.BackgroundColor =new BaseColor(238,233,233);
market_table.AddCell(header); //adds cell to the table
```

I now get the cell with the background color I specified (grey), but the lines change from black to grey... I want to draw those lines in black!

Posted on StackOverflow on Oct 28, 2014<sup>289</sup> by Annadate Piyush<sup>290</sup> There are two problems with your code:

**Problem #1:** the method `DrawLines()` doesn't draw any lines.

It creates the paths for two lines, but the lines are not drawn by that method. You need to add the following line:

```
cb.Stroke();
```

Without that line, drawing the lines is postponed until the stroke operator is called. That may never happen, in which case, the lines are never drawn. In your case, it happens when other content is drawn. By that time, the stroke color may have changed, in which case, the color used to draw the paths you've constructed in your `DrawLines()` method is unpredictable.

---

<sup>289</sup><http://stackoverflow.com/questions/26605887/cell-background-color-affects-the-color-of-other-lines>

<sup>290</sup><http://stackoverflow.com/users/4057042/annadate-piyush>

### Problem #2: you don't use best practices.

The colors that will be used to draw lines and shapes in your code are unpredictable because you aren't careful with the graphics state stack. Best practices would be to save and restore the graphics state when changing colors, line widths, etc...

I would change your `DrawLines()` method like this:

```
private void DrawLines(Document pdfDoc, PdfContentByte cb) {
    cb.SaveState();
    cb.SetColorStroke(GrayColor.GRAYBLACK);
    cb.MoveTo(0, 562);
    cb.LineTo(pdfDoc.PageSize.Width, 562);
    cb.MoveTo(0, 561);
    cb.LineTo(pdfDoc.PageSize.Width, 561);
    cb.Stroke();
    cb.RestoreState();
}
```

Now you save the graphics state (`SaveState()`) before you change the color to Black (`SetColorStroke()`). You construct the paths for the lines (using the `LineTo()` and `MoveTo()` method) and you draw those lines (`Stroke()`). To make sure that the color change you applied doesn't affect other content you might be adding, you restore the graphics state stack to its previous state (`RestoreState()`).

## How do I set parameters back to the default value?



I am using absolute positioning when writing text in a PDF document using iTextSharp. It only have a single `BaseFont` instance for a regular font and there is no **Bold** version of that font. Therefore, it is not possible to set a **Bold** font with the `SetFontAndSize()` method.

I read in a post that this was an alternative way to set the font to bold:

```
pdfContentByte.SetCharacterSpacing(1);
pdfContentByte.SetRGBColorFill(66, 00, 00);
pdfContentByte.SetLineWidth((float)0.5);
pdfContentByte.SetTextRenderingMode(PdfContentByte.TEXT_RENDER_MODE_FILL_STROKE);
```

That works, but creates another problem. I don't know how to set these parameters back to my old default (none-bolded font).

Posted on StackOverflow on Apr 15, 2014 <sup>291</sup> by sdalby<sup>292</sup>

---

<sup>291</sup><http://stackoverflow.com/questions/23086454/setting-basefont-parameters-back-to-default-after-bold>

<sup>292</sup><http://stackoverflow.com/users/3173943/sdalby>

The answer is very simple: you need to *save* the state before you change the rendering mode, and *restore* the state after you've added the text:

```
pdfContentByte.SaveState();
pdfContentByte.SetCharacterSpacing(1);
pdfContentByte.SetRGBColorFill(66, 00, 00);
pdfContentByte.SetLineWidth((float)0.5);
pdfContentByte.SetTextRenderingMode(PdfContentByte.TEXT_RENDER_MODE_FILL_STROKE);
// add the text using the changed state
pdfContentByte.RestoreState();
```

The changes you make to the character spacing, color, line width and rendering mode will only be valid between the `SaveState()` and `RestoreState()` sequence.

## How to add a shading pattern to a custom shape?



I have drawn an equilateral triangle as follows using iText

```
canvas.setColorStroke(BaseColor.BLACK);
int x = start.getX();
int y = start.getY();
canvas.moveTo(x,y);
canvas.lineTo(x + side,y);
canvas.lineTo(x + (side/2), (float)(y+(side*Math.sin(convertToRadian(60)))));
canvas.closePathStroke();
```

I wish to multi color gradient in this shape i.e. fill it with shading comprising of `BaseColor.PINK` and `BaseColor.BLUE`. I just can't find a way to do this with iText?

Posted on StackOverflow on Oct 27, 2014 <sup>293</sup> by RaghavShukla<sup>294</sup>

I've created an example called [ShadedFill](#)<sup>295</sup> that fills the triangle you are drawing using a shading pattern that goes from pink to blue as shown in the [shaded\\_fill.pdf](#)<sup>296</sup> PDF:

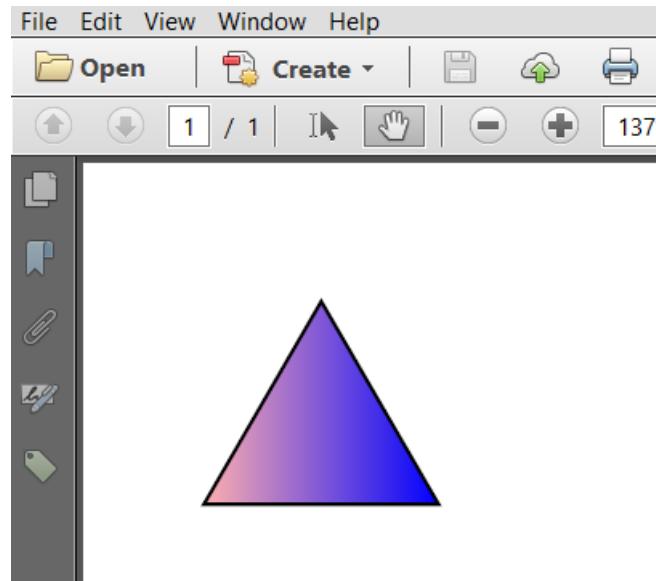
---

<sup>293</sup><http://stackoverflow.com/questions/26586093/how-to-add-a-shading-pattern-to-a-custom-shape>

<sup>294</sup><http://stackoverflow.com/users/2847083/raghaveshukla>

<sup>295</sup><http://itextpdf.com/sandbox/objects/ShadedFill>

<sup>296</sup>[http://itextpdf.com/sites/default/files/shaded\\_fill.pdf](http://itextpdf.com/sites/default/files/shaded_fill.pdf)



shaded\_fill.pdf

```
PdfContentByte canvas = writer.getDirectContent();
float x = 36;
float y = 740;
float side = 70;
PdfShading axial = PdfShading.simpleAxial(writer, x, y,
    x + side, y, BaseColor.PINK, BaseColor.BLUE);
PdfShadingPattern shading = new PdfShadingPattern(axial);
canvas.setShadingFill(shading);
canvas.moveTo(x,y);
canvas.lineTo(x + side, y);
canvas.lineTo(x + (side / 2), (float)(y + (side * Math.sin(Math.PI / 3))));
canvas.closePathFillStroke();
```

As you can see, you need to create a `PdfShading` object. I created an axial shading that varies from pink to blue from the coordinate `(x, y)` to the coordinate `(x + side, y)`. With this axial shading, you can create a `PdfShadingPattern` that can be used as a parameter of the `setShadingFill()` method to set the fill color for the canvas.

See [ShadedFill<sup>297</sup>](#) for the full source code.

<sup>297</sup><http://itextpdf.com/sandbox/objects/ShadedFill>

# Tables

The `PdfPTable` class is one of the most popular classes in the context of document creation. Let's take a look at some questions and answers regarding tables, rows and cells.

## How to right-align text in a `PdfPCell`?



I have a C# application that generates a PDF invoice. In this invoice is a table of items and prices. This is generated using a `PdfPTable` and `PdfPCells`.

I want to be able to right-align the price column but I cannot seem to be able to - the text always comes out left-aligned in the cell.

Here is my code for creating the table:

```
PdfPTable table = new PdfPTable(2);
table.TotalWidth = invoice.PageSize.Width;
float[] widths = { invoice.PageSize.Width - 70f, 70f };
table.SetWidths(widths);
table.AddCell(new Phrase("Item Name", tableHeadFont));
table.AddCell(new Phrase("Price", tableHeadFont));
SqlCommand cmdItems = new SqlCommand("SELECT...", con);
using (SqlDataReader rdrItems = cmdItems.ExecuteReader())
{
    while (rdrItems.Read())
    {
        table.AddCell(new Phrase(rdrItems["itemName"].ToString(), tableFont));
        double price = Convert.ToDouble(rdrItems["price"]);
        PdfPCell pcell = new PdfPCell();
        pcell.HorizontalAlignment = PdfPCell.ALIGN_RIGHT;
        pcelladdElement(new Phrase(price.ToString("0.00"), tableFont));
        table.AddCell(pcell);
    }
}
```

Can anyone help?

Posted on StackOverflow on Nov 28, 2012 <sup>298</sup> by colincameron<sup>299</sup>

---

<sup>298</sup><http://stackoverflow.com/questions/13607970/right-aligning-text-in-pdfcell>

<sup>299</sup><http://stackoverflow.com/users/970789/colincameron>

You're mixing **text mode** and **composite mode**.

In **text mode**, you create the PdfPCell with a Phrase as the parameter of the constructor, and you define the alignment at the level of the cell. However, you're working in **composite mode**. This mode is triggered as soon as you use the addElement() method. In **composite mode**, the alignment defined at the level of the cell is ignored (which explains your problem). Instead, the alignment of the separate elements is used.

How to solve your problem?

Either work in **text mode** by adding your Phrase to the cell in a different way. Or work in **composite mode** and use a Paragraph for which you define the alignment.

The advantage of **composite mode** over **text mode** is that different paragraphs in the same cell can have different alignments, whereas you can only have one alignment in **text mode**. Another advantage is that you can add more than just text: you can also add images, lists, tables,... An advantage of **text mode** is speed: it takes less processing time to deal with the content of a cell.

## How to use multiple fonts in a single cell?



I'm making a windows form for a friend that delivers packages. So I want to transfer his current paper form, into a .pdf with the library iTextSharp.

**What I need:**

I want the table to have a little headline, "Company name" for example, the text should be a little smaller than the text input from the windows form. Currently I'm using cells and was wondering if I can use 2 different font sizes within the same cell?

**What I have:**

```
table.AddCell("Static headline" + Chunk.NEWLINE + richTextBox1.Text);
```

**What I "want":**

```
var normalFont = FontFactory.GetFont(FontFactory.HELVETICA, 9);
var boldFont = FontFactory.GetFont(FontFactory.HELVETICA_BOLD, 12);
table.AddCell("Static headline", boldFont + Chunk.NEWLINE + richTextBox1.Text, n\
ormalFont);
```

Posted on StackOverflow on Feb 13, 2014 <sup>300</sup> by Frederik Kiel<sup>301</sup>

<sup>300</sup><http://stackoverflow.com/questions/21750597/c-sharp-itextsharp-multi-fonts-in-a-single-cell>

<sup>301</sup><http://stackoverflow.com/users/3300515/frederik-kiel>

You're passing a `String` and a `Font` to the `AddCell()` method. That's not going to work. You need the `AddCell()` method that takes a `Phrase` object or a `PdfPCell` object as parameter.

A `Phrase` is an object that consists of different `Chunks`, and the different `Chunks` can have different font sizes. For instance:

```
Phrase phrase = new Phrase();
phrase.Add(
    new Chunk("Some BOLD text", new Font(Font.FontFamily.TIMES_ROMAN, 12, Font.\nBOLD))
);
phrase.Add(new Chunk(" some normal text", new Font()));
table.AddCell(phrase);
```

A `PdfPCell` is an object to which you can add different objects, such as `Phrases`, `Paragraphs`, `Images`,...

```
PdfPCell cell = new PdfPCell();
cell.addElement(new Paragraph("Hello"));
cell.addElement(list);
cell.addElement(image);
```

In this snippet `list` is of type `List` and `image` is of type `Image`.

The first snippet uses **text mode**; the second snippet uses **composite mode**. Cells behave very differently depending on the mode you use.

## How to change width of single column of table?



hello I have created table in a PDF file using iText. The heading of my table columns are “Medicine Name”, “Doses”, and “time: This is what my columns look like:

Medicin	Doses	time
e name		

As you can see the word “Medicine” is split into “Medicin” and “e”. I want to avoid this by changing the width of the first column, but I don't know how to do that.

Posted on StackOverflow on Jun 10, 2014 <sup>302</sup> by DCoder<sup>303</sup>

---

<sup>302</sup><http://stackoverflow.com/questions/24141791/how-to-change-width-of-single-column-of-table-itext-android>

<sup>303</sup><http://stackoverflow.com/users/2843856/dcoder>

The [ColumnWidths<sup>304</sup>](#) example demonstrates different ways of changing the width of a column. This is one specific way:

```
PdfPTable table = new PdfPTable(3);
table.setWidths(new int[]{2, 1, 1});
```

Now the width of the first column is double the size of the second and third column.

## How to introduce a rowspan?



I am adding a table to a PDF file. I have 3 rows and 3 columns. I want the first column to appear only once as a single cell for all the rows. The result should be like where it says Deloitte in the column of company as shown in the image:

Company	Name	Gender
deloitte	sindhu	f
	shilpa	f
	srikan	m
	abc	f
	def	f
	geh	m

Table showing desired result

Posted on StackOverflow on Apr 5, 2014 <sup>305</sup> by sindhu jampani<sup>306</sup>

The [MyFirstTable<sup>307</sup>](#) example from my book does exactly what you need. Ported to C#, it looks like this:

---

<sup>304</sup><http://itextpdf.com/examples/ii.php?id=76>

<sup>305</sup><http://stackoverflow.com/questions/22878410/constant-column-for-multiple-rows>

<sup>306</sup><http://stackoverflow.com/users/3191619/sindhu-jampani>

<sup>307</sup><http://itextpdf.com/examples/ii.php?id=75>

```

PdfPTable table = new PdfPTable(3);
// the cell object
PdfPCell cell;
// we add a cell with colspan 3
cell = new PdfPCell(new Phrase("Cell with colspan 3"));
cell.Colspan = 3;
table.AddCell(cell);
// now we add a cell with rowspan 2
cell = new PdfPCell(new Phrase("Cell with rowspan 2"));
cell.Rowspan = 2;
table.AddCell(cell);
// we add the four remaining cells with addCell()
table.AddCell("row 1; cell 1");
table.AddCell("row 1; cell 2");
table.AddCell("row 2; cell 1");
table.AddCell("row 2; cell 2");

```

You can look at the resulting PDF [here<sup>308</sup>](#). In your case you'd need

```
cell.Rowspan = 6;
```

For the cell with value Deloitte.

## How to maintain a paragraph's indentation inside a table cell?



Some of my paragraph objects contain text that can be greater than the width of the cell in length. These paragraphs are also indented, but the problem is if a paragraph needs to take a new line due to text length, the indenting is not maintained on the new line, causing most of the text to be indented, then the remainder of the text containing no indenting on the new line.

Is there a way to determine if a paragraph will take a new line due to exceeding `PdfPCell` width, and to indent the remainder of the text if this is true?

Posted on StackOverflow on Jan 21, 2015 <sup>309</sup> by [jbailie1991<sup>310</sup>](#)

---

<sup>308</sup>[http://examples.itextpdf.com/results/part1/chapter04/first\\_table.pdf](http://examples.itextpdf.com/results/part1/chapter04/first_table.pdf)

<sup>309</sup><http://stackoverflow.com/questions/28073190/itext-maintain-indenting-if-paragraph-takes-new-line-in-pdfcell>

<sup>310</sup><http://stackoverflow.com/users/3914282/jbailie1991>

Please take a look at the [SimpleTable4<sup>311</sup>](#) example. In this example, I add an indented paragraph to a cell the wrong way (your way) and I add a paragraph to a cell the correct way (as explained in the documentation):

```
PdfPTable table = new PdfPTable(1);
Paragraph wrong = new Paragraph("This is wrong, because an object that was originally a paragraph is reduced to a phrase due to the fact that it's put into a cell that uses text mode.");
wrong.setIndentationLeft(20);
PdfPCell wrongCell = new PdfPCell(wrong);
table.addCell(wrongCell);
Paragraph right = new Paragraph("This is right, because we create a paragraph with an indentation to the left and as we are adding the paragraph in composite mode, all the properties of the paragraph are preserved.");
right.setIndentationLeft(20);
PdfPCell rightCell = new PdfPCell();
rightCell.addElement(right);
table.addCell(rightCell);
document.add(table);
```

This is the result:

This is wrong, because an object that was originally a paragraph is reduced to a phrase due to the fact that it's put into a cell that uses text mode.

This is right, because we create a paragraph with an indentation to the left and as we are adding the paragraph in composite mode, all the properties of the paragraph are preserved.

#### Cell in text mode, cell in composite mode

In the first row, we no longer have a Paragraph, we have a Phrase that uses the alignment and the leading of the PdfPCell.

In the second row, the Paragraph is preserved. If an alignment or a leading was defined at the level of the PdfPCell, it is ignored in favor of the alignment and the leading of the Paragraph. All the other properties that are defined at the level of the Paragraph (and that do not exist for Phrase objects) are preserved.

<sup>311</sup><http://itextpdf.com/sandbox/tables/SimpleTable4>

## What is the PdfPTable.DefaultCell property used for?



What is the DefaultCell property used for? The Java documentation for PdfPTable.getDefaultCell() reads:

*Gets the default PdfPCell that will be used as reference for all the addCell methods except addCell(PdfPCell).*

I don't understand this.

Posted on StackOverflow on Jun 3, 2014 <sup>312</sup> by Water Cooler v2<sup>313</sup>

When creating a PdfPTable, you add cells.

One way is to create a PdfPCell object and to add that cell with the addCell() method.

Another way is to use a short-cut: you don't create a PdfPCell, but you add a String or a Phrase to the table with the addCell() method. In this case, a PdfPCell is created internally using **default properties**. You can **change the default properties by changing the properties of the default cell**. The default cell is obtained using the getDefaultCell() method.

This is what the Javadoc information is about: this default PdfPCell will be used as reference for all the addCell() methods except addCell(PdfPCell). (Because when adding a PdfPCell, the properties of that PdfPCell will be used, *not* the properties of the *default* cell.)

## How to draw a borderless table in iTextSharp?



It appears as though the PdfPCell class does have a border property on it but not the PdfPTable class.

Is there some property on the PdfPTable class to set the borders of all its contained cells in one statement?

Posted on StackOverflow on Jun 3, 2014 <sup>314</sup> by Water Cooler v2<sup>315</sup>

Borders are defined at the level of the cell, not at the level of the table. Hence: if you want to remove the borders of the table, you need to remove the borders of each cell.

<sup>312</sup><http://stackoverflow.com/questions/24006618/what-is-the-pdfptable-defaultcell-property-used-for>

<sup>313</sup><http://stackoverflow.com/users/303685/water-cooler-v2>

<sup>314</sup><http://stackoverflow.com/questions/24006547/draw-a-borderless-table-in-itextsharp>

<sup>315</sup><http://stackoverflow.com/users/303685/water-cooler-v2>

By default, each cell has a border. You can change this default behavior by changing the border of each cell. For instance: if you create PdfPCell objects, you use:

```
cell.setBorder(Rectangle.NO_BORDER);
```

In case the cells are created internally, you need to change that property at the level of the default cell.

```
table.getDefaultCell().setBorder(Rectangle.NO_BORDER);
```

For special borders, for instance borders with rounded corners or a single border for the whole table, or double borders, you can use either cell events or table events, or a combination of both.

## Why doesn't `getDefaultCell().setBorder(PdfPCell.NO_BORDER)` have any effect?



I'm new with iText and I'm trying to build a table. For some reason `table.getDefaultCell().setBorder(PdfPCell.NO_BORDER)` has no effect: my table has still borders.

Here is my code:

```
PdfPTable table = new PdfPTable(new float[] { 1, 1, 1, 1, 1 });
table.getDefaultCell().setBorder(PdfPCell.NO_BORDER);
Font tfont = new Font(Font.FontFamily.UNDEFINED, 10, Font.BOLD);
table.setWidthPercentage(100);
PdfPCell cell;
cell = new PdfPCell(new Phrase("Menge", tfont));
table.addCell(cell);
cell = new PdfPCell(new Phrase("Beschreibung", tfont));
table.addCell(cell);
cell = new PdfPCell(new Phrase("Einzelpreis", tfont));
table.addCell(cell);
cell = new PdfPCell(new Phrase("Gesamtpreis", tfont));
table.addCell(cell);
cell = new PdfPCell(new Phrase("MwSt", tfont));
table.addCell(cell);
document.add(table);
```

Do you have any idea what I am doing wrong?

Posted on StackOverflow on Nov 30, 2014 <sup>[316](#)</sup> by hiasl<sup>[317](#)</sup>

---

<sup>316</sup><http://stackoverflow.com/questions/27212695/itext-5-getdefaultcell-setborderpdfcell-no-border-has-no-effect>

<sup>317</sup><http://stackoverflow.com/users/4308508/hiasl>

You are mixing two different concepts.

**Concept 1:** you define every PdfPCell manually, for instance:

```
PdfPCell cell = new PdfPCell(new Phrase("Menge", tfont));
cell.setBorder(Rectangle.NO_BORDER);
table.addCell(cell);
```

In this case, you define every aspect, every property of the cell on the cell itself.

**Concept 2:** you allow iText to create the PdfPCell implicitly, for instance:

```
table.addCell("Adding a String");
table.addCell(new Phrase("Adding a phrase"));
```

In this case, you can define properties at the level of the *default cell*. These properties will be used internally when iText creates a PdfPCell in your place.

**Conclusion:** either you define the border for all the PdfPCell instances separately, or you let iText create the PdfPCell instances in which case you can define the border at the level of the default cell. If you choose the second option, you can adapt your code like this:

```
PdfPTable table = new PdfPTable(new float[] { 1, 1, 1, 1, 1 });
table.getDefaultCell().setBorder(PdfPCell.NO_BORDER);
Font tfont = new Font(Font.FontFamily.UNDEFINED, 10, Font.BOLD);
table.setWidthPercentage(100);
table.addCell(new Phrase("Menge", tfont));
table.addCell(new Phrase("Beschreibung", tfont));
table.addCell(new Phrase("Einzelpreis", tfont));
table.addCell(new Phrase("Gesamtpreis", tfont));
table.addCell(new Phrase("MwSt", tfont));
document.add(table);
```

This decision was made by design, based on experience: it offers the most flexible to work with cells and properties.

## How to define spacing and leading in PdfPCells?



Is it possible to add space between the elements of a cell (rows) in C#? I'm creating a PDF in visual studio 2012 and wanted to set some space between the rows.

```
PdfPTable cellTable = new PdfPTable(1);
PdfPCell cell= new PdfPCell();
for(i=0; i < 5; i++) {
    var titleChunk = new Chunk(tittle[i], body);
    var descriptionChunk = new Chunk(" " description[i], body2);
    var phrase = new Phrase(titleChunk);
    phrase.Add(descriptionChunk);
    cell.AddElement(phrase);
}
cellTable.AddCell(cell);
```

Posted on StackOverflow on Nov 22, 2013 <sup>[318](#)</sup> by Micael Florêncio<sup>[319](#)</sup>

OK, I've made you an example named `LeadingInCell`<sup>[320](#)</sup>:

```
PdfPCell cell = new PdfPCell();
Paragraph p;
p = new Paragraph(16, "paragraph 1: leading 16");
cell.addElement(p);
p = new Paragraph(32, "paragraph 2: leading 32");
cell.addElement(p);
p = new Paragraph(10, "paragraph 3: leading 10");
cell.addElement(p);
p = new Paragraph(18, "paragraph 4: leading 18");
cell.addElement(p);
p = new Paragraph(40, "paragraph 5: leading 40");
cell.addElement(p);
```

As you can see in `leading_in_cell.pdf`<sup>[321](#)</sup>, you define the space between the lines using the first parameter of the `Paragraph` constructor. I've used different values to demonstrate how it works. The third paragraph sticks to the second one, because the leading of the third paragraph is only 10 pt. There's plenty of space between the fourth and the fifth paragraph, because the leading of the fifth paragraph is 40 pt.

<sup>318</sup><http://stackoverflow.com/questions/20145742/spacing-leading-pdfcells-elements>

<sup>319</sup><http://stackoverflow.com/users/2895166/micael-flor%C3%A9ncio>

<sup>320</sup><http://itextpdf.com/sandbox/tables/LeadingInCell>

<sup>321</sup>[http://itextpdf.com/sites/default/files/leading\\_in\\_cell.pdf](http://itextpdf.com/sites/default/files/leading_in_cell.pdf)

## How can I convert a CSV file to a table with a repeating header row?



I am generating a PDF file from CSV using iText. I need to format the file such that the header row (which occurs in the beginning of every page) should be in a different font and color. Just to be clear, I know how to set the font style/size/color. I'm having a tough time finding out how to do that *for the header rows* of my table.

Posted on StackOverflow on [Nov 10, 2014](#) <sup>322</sup> by [harsha](#)<sup>323</sup>

Your requirement is explained in large detail in our [tutorial video](#)<sup>324</sup>, more specifically in the [UnitedStates](#)<sup>325</sup> example. In this example, we take a CSV file containing the different states of the US: [united\\_states.csv](#)<sup>326</sup>

```
name;abbr;capital;most populous city;population;square miles;time zone 1;time zo\
ne 2;dst
ALABAMA;AL;Montgomery;Birmingham;4,708,708;52,423;CST (UTC-6);EST (UTC-5);YES
ALASKA;AK;Juneau;Anchorage;698,473;656,425;AKST (UTC-09) ;HST (UTC-10) ;YES
ARIZONA;AZ;Phoenix;Phoenix;6,595,778;114,006;MT (UTC-07); ;NO
ARKANSAS;AR;Little Rock;Little Rock;2,889,450;53,182;CST (UTC-6); ;YES
CALIFORNIA;CA;Sacramento;Los Angeles;36,961,664;163,707;PT (UTC-8); ;YES
```

And we parse these into a PDF with a repeating header: [united\\_states.pdf](#)<sup>327</sup>

The only difference, is that you want the text in the header to be bold. This requires only a minimal change to the code:

---

<sup>322</sup><http://stackoverflow.com/questions/26838145/set-format-of-header-row-in-itext>

<sup>323</sup><http://stackoverflow.com/users/3679708/harsha>

<sup>324</sup><http://www.youtube.com/watch?v=6YwDME0Fl1c>

<sup>325</sup><http://itextpdf.com/sandbox/tables/UnitedStates>

<sup>326</sup>[http://itextpdf.com/sites/default/files/united\\_states.csv](http://itextpdf.com/sites/default/files/united_states.csv)

<sup>327</sup>[http://itextpdf.com/sites/default/files/united\\_states.pdf](http://itextpdf.com/sites/default/files/united_states.pdf)

```
public void createPdf(String dest) throws IOException, DocumentException {
    Document document = new Document(PageSize.A4.rotate());
    PdfWriter.getInstance(document, new FileOutputStream(dest));
    document.open();
    PdfPTable table = new PdfPTable(9);
    table.setWidthPercentage(100);
    table.setWidths(new int[]{4, 1, 3, 4, 3, 3, 3, 3, 1});
    BufferedReader br = new BufferedReader(new FileReader(DATA));
    String line = br.readLine();
    process(table, line, new Font(FontFamily.HELVETICA, 16, Font.BOLD));
    table.setHeaderRows(1);
    while ((line = br.readLine()) != null) {
        process(table, line, new Font(FontFamily.HELVETICA, 12));
    }
    br.close();
    document.add(table);
    document.close();
}

public void process(PdfPTable table, String line, Font font) {
    StringTokenizer tokenizer = new StringTokenizer(line, ";");
    while (tokenizer.hasMoreTokens()) {
        table.addCell(new Phrase(tokenizer.nextToken(), font));
    }
}
```

Take a close look at how the `process()` method was changed: it now accepts a `font` parameter so that we can define a bigger, bolder font for the header.

## How to create a table based on a two-dimensional array?



I have a series of list in this list:

```
private List<List<String>> tableOverallList;
```

This object contains 8 values in each list. I need to place it in the table created. would like to have 2 Rows with 8 columns:

```
String[] tableTitleList = {" Title", " (Re)set", " Obs", " Mean",
    " Std.Dev", " Min", " Max", "Unit"};
List<String> tabTitleList = Arrays.asList(tableTitleList);
```

Please help me to place the 1st list of values inside the List tableOverallList in the 2nd Row. I will try managing with the rest of the list.

```
PdfPTable table = new PdfPTable(3); // 3 columns.
PdfPCell cell11 = new PdfPCell(new Paragraph("Cell 1"));
PdfPCell cell12 = new PdfPCell(new Paragraph("Cell 2"));
PdfPCell cell13 = new PdfPCell(new Paragraph("Cell 3"));
PdfPCell cell14 = new PdfPCell(new Paragraph("Cell 4"));
PdfPCell cell15 = new PdfPCell(new Paragraph("Cell 5"));
PdfPCell cell16 = new PdfPCell(new Paragraph("Cell 6"));
PdfPCell cell17 = new PdfPCell(new Paragraph("Cell 7"));
PdfPCell cell18 = new PdfPCell(new Paragraph("Cell 8"));
table.addCell(cell11);
table.addCell(cell12);
table.addCell(cell13);
table.addCell(cell14);
table.addCell(cell15);
table.addCell(cell16);
table.addCell(cell17);
table.addCell(cell18);
document.add(table);
```

Posted on StackOverflow on Jun 25, 2014 <sup>328</sup> by srinivasan<sup>329</sup>

That's really easy. So you have data in a nested list. For instance:

---

<sup>328</sup><http://stackoverflow.com/questions/24404686/i-need-to-create-a-table-and-assign-the-values-into-the-table-in-pdf-using-javaf>

<sup>329</sup><http://stackoverflow.com/users/3539828/srinivasan>

```
public List<List<String>> getData() {  
    List<List<String>> data = new ArrayList<List<String>>();  
    String[] tableTitleList = {"Title", "(Re)set", "Obs", "Mean",  
        "Std.Dev", "Min", "Max", "Unit"};  
    data.add((Arrays.asList(tableTitleList));  
    for (int i = 0; i < 10; ) {  
        List<String> dataLine = new ArrayList<String>();  
        i++;  
        for (int j = 0; j < tableTitleList.length; j++) {  
            dataLine.add(tableTitleList[j] + " " + i);  
        }  
        data.add(dataLine);  
    }  
    return data;  
}
```

This will return a set of data where the first record is a title row and the following 10 rows contain mock-up data. It is assumed that this is data you have.

Now when you want to render this data in a table, you do this:

```
PdfPTable table = new PdfPTable(8);  
table.setWidthPercentage(100);  
List<List<String>> dataset = getData();  
for (List<String> record : dataset) {  
    for (String field : record) {  
        table.addCell(field);  
    }  
}  
document.add(table);
```

The result will look like this:

Title	(Re)set	Obs	Mean	Std.Dev	Min	Max	Unit
Title 1	(Re)set 1	Obs 1	Mean 1	Std.Dev 1	Min 1	Max 1	Unit 1
Title 2	(Re)set 2	Obs 2	Mean 2	Std.Dev 2	Min 2	Max 2	Unit 2
Title 3	(Re)set 3	Obs 3	Mean 3	Std.Dev 3	Min 3	Max 3	Unit 3
Title 4	(Re)set 4	Obs 4	Mean 4	Std.Dev 4	Min 4	Max 4	Unit 4
Title 5	(Re)set 5	Obs 5	Mean 5	Std.Dev 5	Min 5	Max 5	Unit 5
Title 6	(Re)set 6	Obs 6	Mean 6	Std.Dev 6	Min 6	Max 6	Unit 6
Title 7	(Re)set 7	Obs 7	Mean 7	Std.Dev 7	Min 7	Max 7	Unit 7
Title 8	(Re)set 8	Obs 8	Mean 8	Std.Dev 8	Min 8	Max 8	Unit 8
Title 9	(Re)set 9	Obs 9	Mean 9	Std.Dev 9	Min 9	Max 9	Unit 9
Title 10	(Re)set 10	Obs 10	Mean 10	Std.Dev 10	Min 10	Max 10	Unit 10

Resulting table

You can find the full source code here: [ArrayToTable<sup>330</sup>](http://itextpdf.com/sandbox/tables/ArrayToTable) and this is the resulting PDF: [array\\_to\\_table.pdf<sup>331</sup>](http://itextpdf.com/sites/default/files/array_to_table.pdf)

If you only want 2 rows, a title row and a data row, change the following line:

```
for (int i = 0; i < 10; ) {
```

Into this:

```
for (int i = 0; i < 2; ) {
```

I've provided a more generic solution because it's more elegant to have generic code.

## How to nest tables without extending the inner table?



I am using iTextSharp in a .NET project to generate a PDF file. I have a PdfPTable with two cells in one row. I put another PdfPTable in each cell: the first one has two rows; the second one has three rows. The last row of the smaller table stretches to fill the space. How can I avoid that the smaller table stretches to fit the cell of the outer table? Additionally: how do I align the inner table to the bottom?

Posted on StackOverflow on Feb 11, 2015 <sup>332</sup> by lNg<sup>333</sup>

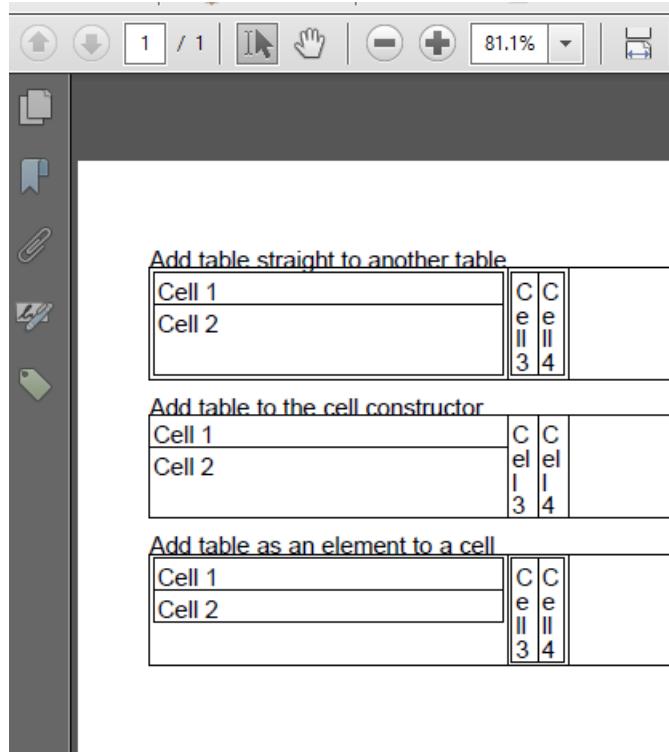
<sup>330</sup><http://itextpdf.com/sandbox/tables/ArrayToTable>

<sup>331</sup>[http://itextpdf.com/sites/default/files/array\\_to\\_table.pdf](http://itextpdf.com/sites/default/files/array_to_table.pdf)

<sup>332</sup><http://stackoverflow.com/questions/28444598/how-to-nest-tables-without-extending-the-inner-table>

<sup>333</sup><http://stackoverflow.com/users/477890/lNg>

Please take a look at the following screen shot:



Screen shot

This screen shot was taken from the result of the [NestedTables<sup>334</sup>](#) example. You are describing what happens when you add the table straight to another table (first table in the screen shot):

```
outerTable.addCell(innerTable);
```

Or maybe you are describing what happens when you add the table as a parameter for the constructor of a `PdfPCell` (second table in the screen shot):

```
PdfPCell cell = new PdfPCell(innerTable);
outerTable.addCell(cell);
```

Note that the difference between both is subtle: when you add the inner table straight to the outer table, the padding of the default cell is taken into account. This padding is 2 by default. When you pass the inner table as a parameter to create a `PdfPCell`, the padding of the cell is 0 by default.

If I understand your question correctly, you want the behavior shown in the third table in the screen shot. That table shows what happens when you add the table to a `PdfPCell` first:

---

<sup>334</sup><http://itextpdf.com/sandbox/tables/NestedTables>

```
PdfPCell cell = new PdfPCell();
cell.addElement(innerTable);
outerTable.addCell(cell);
```

You can align the inner table vertically like this:

```
cell.VerticalAlignment = Element.ALIGN_BOTTOM;
```

## How to add an inner table surrounded by text?



I am trying to add a table surrounded by text to an outer table in iText 5.5.4, but the inner table disappears and I can't seem to fix the problem.

Here is what I am expecting:

```
*****
* Hello World      *
* ++++++           * <-
* + Goodbye World + * <- these 3 lines never show up in the PDF
* ++++++           * <-
* Hello World      *
*****
```

Here is my code example:

```
final PdfPTable table2 = new PdfPTable(1);
table2.getDefaultCell().setBorderColor(BaseColor.RED);
table2.getDefaultCell().setBorderWidth(1);
table2.addCell("Goodbye World");
final PdfPTable table1 = new PdfPTable(1);
table1.setWidthPercentage(100);
table1.getDefaultCell().setBorderColor(BaseColor.BLACK);
table1.getDefaultCell().setBorderWidth(1);
Phrase phrase = new Phrase();
phrase.add(new Chunk("Hello World"));
phrase.add(table2); // <--- added but doesn't show up!
phrase.add(new Chunk("Hello World"));
table1.addCell(phrase);
document.add(table1);
```

Posted on StackOverflow on Feb 10, 2015 <sup>335</sup> by rveach<sup>336</sup>

---

<sup>335</sup><http://stackoverflow.com/questions/28418108/itext-how-to-add-an-inner-table-surrounded-by-text-to-a-table>

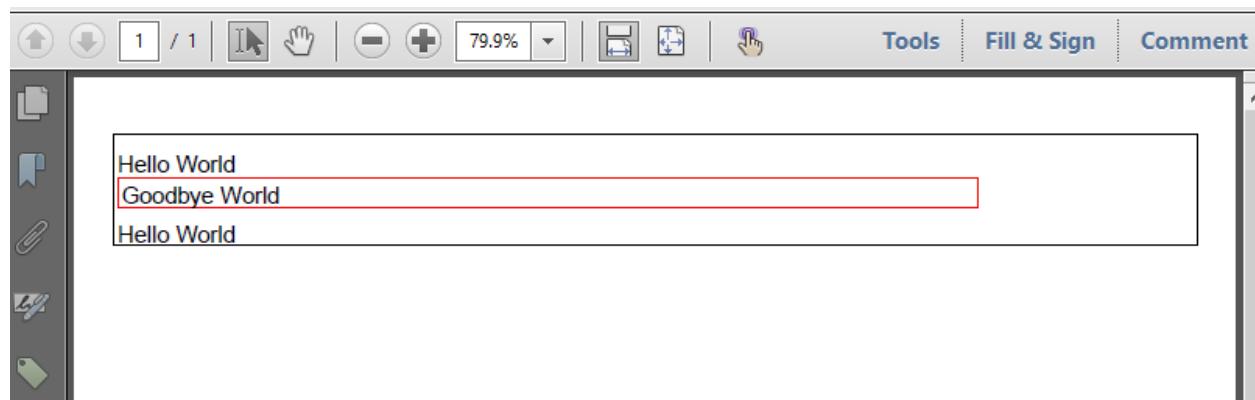
<sup>336</sup><http://stackoverflow.com/users/1016482/rveach>

You are using *text mode* (to be used when you only have text) in a situation where you should use *composite mode* (because you are adding a table to a cell).

Please take a look at the [NestedTableProblem<sup>337</sup>](#) example:

```
// table 2
final PdfPTable table2 = new PdfPTable(1);
table2.setHorizontalAlignment(Element.ALIGN_LEFT);
table2.getDefaultCell().setBorderColor(BaseColor.RED);
table2.getDefaultCell().setBorderWidth(1);
table2.addCell("Goodbye World");
// table 1
final PdfPTable table1 = new PdfPTable(1);
table1.setHorizontalAlignment(Element.ALIGN_LEFT);
table1.setWidthPercentage(100);
// contents
PdfPCell cell = new PdfPCell();
cell.setBorderColor(BaseColor.BLACK);
cell.setBorderWidth(1);
cell.addElement(new Chunk("Hello World"));
cell.addElement(table2);
cell.addElement(new Chunk("Hello World"));
table1.addCell(cell);
document.add(table1);
```

In this code snippet, the `cell` object is composed of different elements (that's what *composite mode* is about):



Screen shot

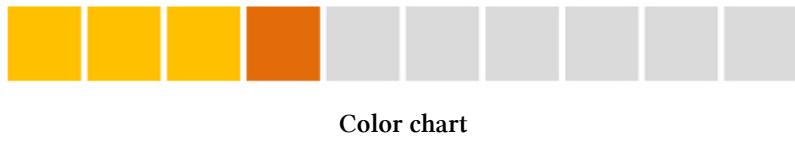
In your code snippet, you add several elements to a `Phrase` and you add this `Phrase` to a `PdfPCell` in *text mode*. As one element isn't ordinary text but a table, it can not be rendered.

<sup>337</sup><http://itextpdf.com/sandbox/tables/NestedTableProblem>

## How to resize an Image to fit it into a PdfPCell?

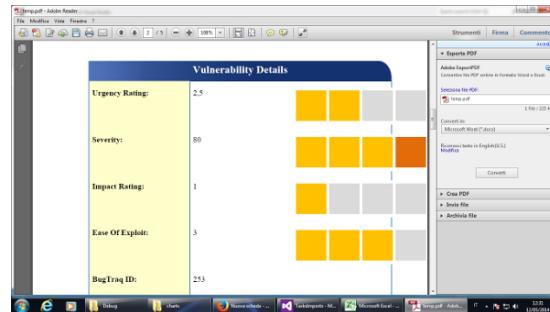


I am inserting some color charts inside my PDF. These charts look like this:



```
vulnerabilityDetailsTable.AddCell(new PdfPCell(img) {
    Border = PdfPCell.RIGHT_BORDER,
    BorderColor = new BaseColor(79, 129, 189),
    BorderWidth = 1,
    Padding = 5, MinimumHeight = 30, PaddingTop = 10 );
}
```

The problem is that the chart image is too big for the cell and I obtain this horrible result:



Resulting PDF

So I have the following 2 questions:

1. Can I resize the Image size into my code or do I have to do this with an image editor? If it's possible, where do I have to do this? Right after I load the image or when I put the image into my PDF table cell?
2. When I look at the image using my Windows Photo Viewer at 100% of the size, I see it much smaller than in the PDF. Why is this?

Posted on StackOverflow on Mar 12, 2014 <sup>338</sup> by Andrea Nobili<sup>339</sup>

There are different strategies for adding an Image to a PdfPCell.

<sup>338</sup><http://stackoverflow.com/questions/23607002/how-to-resize-an-itextsharp-text-image-size-into-my-code>

<sup>339</sup><http://stackoverflow.com/users/1833945/andreanobili>

You are using this:

```
// we wrap the image in a PdfPCell  
PdfPCell cell = new PdfPCell(img[0]);  
table.AddCell(cell);
```

As documented, this option doesn't scale the image (which is what you want). If you want to scale the image, you could use this:

```
// we wrap the image in a PdfPCell and let iText scale it  
cell = new PdfPCell(img[1], true);  
table.AddCell(cell);
```

By adding the boolean parameter `true`, you ask iText to scale the image.

Another option is to use `addCell()` like this:

```
// we add the image with addCell()  
table.AddCell(img[2]);
```

This will also scale the image, but use the properties of the default cell. If you don't change these properties, there will be a padding of 2 user units.

You also have the option to use *composite mode*:

```
cell = new PdfPCell();  
cell.AddElement(img[3]);  
table.AddCell(cell);
```

This will make sure the image is scaled to fill 100 percent of the cell width, unless you change the width percentage of the image, for instance:

```
img[3].WidthPercentage = 50;
```

This line will make sure that the width of the image is 50% of the available width of the cell.

Finally, you can scale the image before adding it to the cell.

## How to display barcodes in a matrix-like structure?



How can I display different bar codes in multiple columns in a PDF page using iText? I have to display 12 bar codes in the same PDF page in three columns each one containing 4 bar codes (in other words it's 4 by 3 matrix).

Posted on StackOverflow on Feb 24, 2014<sup>340</sup> by user3323182<sup>341</sup>

I've made a [Barcodes<sup>342</sup>](#) example that does exactly what you need. See the resulting pdf: [barcodes\\_table.pdf<sup>343</sup>](#)

There's nothing difficult about it. You just create a table with 4 column and you add 12 cell:

```
PdfPTable table = new PdfPTable(4);
table.setWidthPercentage(100);
for (int i = 0; i < 12; i++) {
    table.addCell(createBarcode(writer, String.format("%08d", i)));
}
```

The `createBarcode()` method creates a cell with a barcode:

```
public static PdfPCell createBarcode(PdfWriter writer, String code) throws DocumentException, IOException {
    BarcodeEAN barcode = new BarcodeEAN();
    barcode.setCodeType(Barcode.EAN8);
    barcode.setCode(code);
    PdfPCell cell = new PdfPCell(barcode.createImageWithBarcode(writer.getDirectContent(), BaseColor.BLACK, BaseColor.GRAY), true);
    cell.setPadding(10);
    return cell;
}
```

---

<sup>340</sup><http://stackoverflow.com/questions/21981602/display-barcodes-in-column-form-matrix-form-in-a-pdf-page-in-java-using-itext>

<sup>341</sup><http://stackoverflow.com/users/3323182/user3323182>

<sup>342</sup><http://itextpdf.com/sandbox/tables/Barcodes>

<sup>343</sup>[http://itextpdf.com/sites/default/files/barcode\\_table.pdf](http://itextpdf.com/sites/default/files/barcode_table.pdf)

## How to split a row over multiple pages?



I am using PdfPTable to create a table in PDF. I have a single row in the table. In my row, the last column has data which has a height that needs more space than remaining height of the page. So the row is getting started from the next page while the table headers are on the previous page and there is large blank space below the header on the first page. Can anybody suggest how I can split the row over multiple pages?

Posted on StackOverflow on Feb 27, 2014 <sup>[344](#)</sup> by Manish<sup>[345](#)</sup>

By default, table rows aren't split. iText will try to add a complete row to the current page, and if the row doesn't fit, it will try again on the next page. Only if it doesn't fit on the next page, it will split the row. This is the default behavior, so you shouldn't be surprised by what you see in your application.

You can change this default behavior. There's a method that will allow you to drop content that doesn't match (this is *not* what you want) and there's a method that will allow you to split rows when they don't fit the current page (this is what you want).

The method you need is used in the [HeaderFooter2 example](#)<sup>[346](#)</sup>:

```
PdfPTable table = getTable(...);
table.setSplitLate(false);
```

By default, the value of `setSplitLate()` is `true`: iText will split rows as late as possible. By changing this default to `false`, iText will split rows immediately.

## How does a PdfPCell's height relate to the font size?



In iText, what is the minimum height of a PdfPCell for which a given size of font text is visible. I mean is there any ratio between PdfPCell and font size? I am making a PDF with pages of size A3, A4 and A5. I need to know the relation between Font size and PdfPCell minimum height, so that the text will be visible.

Posted on StackOverflow on Sep 3, 2013 <sup>[347](#)</sup> by MGDroid<sup>[348](#)</sup>

Different concepts are at play when adding text to a cell.

---

<sup>344</sup><http://stackoverflow.com/questions/22063845/table-row-is-getting-started-from-the-new-page-in-itext-pdf>

<sup>345</sup><http://stackoverflow.com/users/1189254/manish>

<sup>346</sup><http://itextpdf.com/examples/ii.php?id=87>

<sup>347</sup><http://stackoverflow.com/questions/18590404/itext-pdfcell-height-in-relation-of-font-size>

<sup>348</sup><http://stackoverflow.com/users/1516775/mgdroid>

- *padding* is the extra space inside the borders of the cell. It's similar to the concept with the same name in HTML. You can change the padding of a cell with the `setPadding()` method.
- *leading* is the space between two lines. Even when there's only one line, this leading will be used to determine the baseline of the text. By default the leading is 1.5 times the font size. If you're working in *text mode*, the leading of a cell is set using the `setLeading()`; you can define a fixed leading (`fixedLeading`), or a leading that depends on the font size (`multipliedLeading`). This value is ignored if you're working in *composite mode*. In *composite mode*, the leading of the separate Elements added to the cell is used.
- *ascender* and *descender* are two value that are font-specific. The ascender is a value that tells you how much space is needed above the baseline of the text; the descender is a value that tells you how much space is needed below the baseline of the text. You can tell iText to take these values into account with the methods `setUseAscender()` and `setUseDescender()`.

So, if you want the cell to have a minimal size, you need to set the padding to 0, the leading to match the size of the font, and tell iText to use the ascender and descender value.

**DISCLAIMER:** in the past, we've received reports from customers that showed us that not all fonts contain the correct ascender and descender information. This needs to be fixed at the font level.

## How to add a table to the bottom of the last page?



I have created table in a PDF document using iText. This works fine, but I don't want to add the table at the current pointer in the page, I want to add the table at the bottom.

```
PdfPTable datatablebottom = new PdfPTable(8);
PdfPCell cell = new PdfPCell();
// add cells here
datatablebottom.setTotalWidth(PageSize.A4.getWidth()-70);
datatablebottom.setLockedWidth(true);
document.add(datatablebottom);
```

Posted on StackOverflow on Mar 9, 2014 <sup>349</sup> by user3434594<sup>350</sup>

You need to define the absolute width of your table:

```
datatable.setTotalWidth(document.right(document.rightMargin())
- document.left(document.leftMargin()));
```

Then you need to replace the line:

---

<sup>349</sup><http://stackoverflow.com/questions/23557956/how-to-create-table-in-pdf-document-last-page-bottom-using-itext-in-java>

<sup>350</sup><http://stackoverflow.com/users/3434594/user3434594>

```
document.add(datatablebottom);
```

with this one:

```
datatable.writeSelectedRows(0, -1,  
    document.left(document.leftMargin()),  
    datatable.getTotalHeight() + document.bottom(document.bottomMargin()),  
    writer.getDirectContent());
```

The `writeSelectedRows()` method draws the table at an absolute position. We calculate that position by asking the document for its left margin (the x value) and by adding the height of the table to the bottom margin of the document (the Y coordinate). We draw *all* rows (0 to -1).

## How do `setMinimumSize()` and `setFixedSize()` interact?



Is it OK to call `setMinimumSize(15)` on some cells in a row, and `setFixedSize(15)` on the other cells of the same row?

What I would like is for iText to increase the row height to accommodate the text in the cells whose minimum height is set, while letting text in cells set to a fixed height clip. Is that what iText does?

If not, how do I achieve this? While we're at it, am I correct that calling neither `setMinimumSize()` nor `setFixedSize()` is equivalent to calling `setMinimumSize(0)`, meaning that iText makes the cell as tall as it needs to be to accommodate the text?

Posted on StackOverflow on Feb 28, 2014 <sup>351</sup> by Kartick Vaddadi<sup>352</sup>

The value defined using `setFixedHeight()` always gets preference. If you use `setMinimumHeight()` and `setFixedHeight()` in the same row, and you define a minimum height along with a fixed height, the fixed height prevails.

- if the minimum height is set to 30pt and the fixed height is 60pt, the height will be 60pt, no matter how much content is added to the cell.
- if the minimum height is set to 60pt and the fixed height is 60pt, the height will be 60pt, no matter how much content is added to the cell.
- if the minimum height is set to 120pt and the fixed height is 60pt, the height will be 60pt, no matter how much content is added to the cell.

<sup>351</sup><http://stackoverflow.com/questions/22093745/itext-how-do-setminimumsize-and-setfixedsize-interact>

<sup>352</sup><http://stackoverflow.com/users/2893711/kartick-vaddadi>

If different fixed heights are defined, the highest value is taken. For instance: if you have a row where one cell has a fixed height (e.g 120 pt) that is higher than the fixed height of another cell (e.g. 60 pt), then the highest value (in this case 120) prevails.

All of this is demonstrated in the [FixedHeightCell<sup>353</sup>](#) example. Please take a look at the [resulting PDF<sup>354</sup>](#). In row D all the cells have a fixed height of 60 pt. In row E, most cells also have a fixed height of 60, but the cell in column 4 has a fixed height of 120, hence the height of the row is 120. Then there's row F, with a fixed height of 60 pt and a minimum height of 120 pt. Although we add text that doesn't fit the cell in column 2, the content is truncated.

## How to get the rendered dimensions of text?



I would like to find out information about the layout of text in a `PdfPCell`. I'm aware of `BaseFont.getWidthPointKerned()`, but I'm looking for more detailed information like:

1. How many lines would a string need if rendered in a cell of a given width (say, 30pt)? What would the height in points of the `PdfPCell` be?
2. Give me the prefix or suffix of a string that fits in a cell of a given width and height. That is, if I have to render the text "*Today is a good day to die*" in a specific font in a `PdfPCell` of width 12pt and height 20pt, what portion of the string would fit in the available space?
3. Where does iText break a given string when asked to render it in a cell of a given width?

Posted on StackOverflow on [Feb 28, 2014<sup>355</sup>](#) by [Kartick Vaddadi<sup>356</sup>](#)

iText uses the `ColumnText` class to render content to a cell. `TColumnText` measures the width of the characters and tests if they fit the available width. If not, the text is split. You can change the split behavior in different ways: by introducing hyphenation or by defining a custom split character.

I've written a small proof of concept to show how you could implement custom "truncation" behavior. See the [TruncateTextInCell<sup>357</sup>](#) example.

Instead of adding the content to the cell, I have an empty cell for which I define a cell event. I pass the long text "D2 is a cell with more content than we can fit into the cell." to this event.

In the event, I use a fancy algorithm: I want the text to be truncated in the middle and insert "..." at the place where I truncated the text.

<sup>353</sup><http://itextpdf.com/sandbox/tables/FixedHeightCell>

<sup>354</sup>[http://itextpdf.com/sites/default/files/fixed\\_height\\_cell.pdf](http://itextpdf.com/sites/default/files/fixed_height_cell.pdf)

<sup>355</sup><http://stackoverflow.com/questions/22093488/itext-how-do-i-get-the-rendered-dimensions-of-text>

<sup>356</sup><http://stackoverflow.com/users/2893711/kartick-vaddadi>

<sup>357</sup><http://itextpdf.com/sandbox/tables/TruncateTextInCell>

```
BaseFont bf = BaseFont.createFont();
Font font = new Font(bf, 12);
float availableWidth = position.getWidth();
int contentLength = content.length();
int leftChar = 0;
int rightChar = contentLength - 1;
availableWidth -= bf.getWidthPoint("...", 12);
while (leftChar < contentLength && rightChar != leftChar) {
    availableWidth -= bf.getWidthPoint(content.charAt(leftChar), 12);
    if (availableWidth > 0)
        leftChar++;
    else
        break;
    availableWidth -= bf.getWidthPoint(content.charAt(rightChar), 12);
    if (availableWidth > 0)
        rightChar--;
    else
        break;
}
String newContent =
    content.substring(0, leftChar) + "..." + content.substring(rightChar);
PdfContentByte canvas = canvases[PdfPTable.TEXTCANVAS];
ColumnText ct = new ColumnText(canvas);
ct.setSimpleColumn(position);
ct.addElement(new Paragraph(newContent, font));
ct.go();
```

As you can see, we get the available width from the `position` parameter and we check how many characters match, alternating between a character at the start and a character at the end of the content.

The result is shown in [the resulting PDF<sup>358</sup>](#): the content is truncated like this: "D2 is a c... the cell."

Your question about "how many lines" can be solved in a similar way. The `ColumnText` class has a `getLinesWritten()` method that gives you that information. You can find more info about positioning a `ColumnText` object in my answer to your other question: "Can I tell iText how to clip text to fit in a cell?"

---

<sup>358</sup>[http://itextpdf.com/sites/default/files/truncate\\_cell\\_content.pdf](http://itextpdf.com/sites/default/files/truncate_cell_content.pdf)

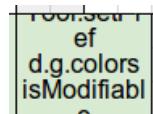
## How to tell iText how to clip text to fit in a cell?



When I call `setFixedHeight()` on a `PdfPCell`, and add more text than fits in the given height, iText seems to print the prefix of the string which fits.

Can I control this clipping algorithm? For example:

1. Print a suffix of the string rather than a prefix.
2. Mark a substring of the string as not to be removed. This is with footnote references. If I add text saying “Hello World [1]”, the [1] is a reference to a footnote and should not be removed. It’s okay to remove the other characters of the string, like “World”.
3. When there are multiple words in the string, iText seems to eliminate a word that doesn’t fit, while I would like it partially printed. That is, if the string is “Hello World”, and the cell has room only for “Hello Wo...”, I would like that to be printed, rather than just “Hello”, as iText prints.
4. Rather than printing characters in their entirety, print only part of them. Imagine printing the text to a PNG and chopping off the top and/or bottom part of the PNG to fit it in the space available. For example, notice that the top line and the bottom line are partially clipped here:



Clipped content in a cell

Are any of these possible? Does iText give me any control over how text is clipped?  
Thanks.

Posted on StackOverflow on Feb 28, 2014 <sup>[359](#)</sup> by [Kartick Vaddadi](#)<sup>[360](#)</sup>

I have written a proof of concept, [ClipCenterCellContent](#)<sup>[361](#)</sup>, where we try to fit the text "D2 is a cell with more content than we can fit into the cell." in a cell that is too small.

Just like in your other question “How do I get the rendered dimensions of text?”, we add the content using a cell event, but we now add it twice: once in simulation mode (to find out how much space is needed vertically) and once for real (using an offset).

This adds the content in simulation mode (we use the width of the cell and an arbitrary height):

---

<sup>359</sup><http://stackoverflow.com/questions/22095320/can-i-tell-itext-how-to-clip-text-to-fit-in-a-cell>

<sup>360</sup><http://stackoverflow.com/users/2893711/kartick-vaddadi>

<sup>361</sup><http://itextpdf.com/sandbox/tables/ClipCenterCellContent>

```

PdfContentByte canvas = canvases[PdfPTable.TEXTCANVAS];
ColumnText ct = new ColumnText(canvas);
ct.setSimpleColumn(new Rectangle(0, 0, position.getWidth(), -1000));
ct.addElement(content);
ct.go(true);
float spaceneeded = 0 - ct.getYLine();
System.out.println(String.format(
    "The content requires %s pt whereas the height is %s pt.",
    spaceneeded, position.getHeight()));

```

We now know the needed height and we can add the content for real using an offset:

```

float offset = (position.getHeight() - spaceneeded) / 2;
System.out.println(String.format(
    "The difference is %s pt; we'll need an offset of %s pt.",
    -2f * offset, offset));
PdfTemplate tmp = canvas.createTemplate(
    position.getWidth(), position.getHeight());
ct = new ColumnText(tmp);
ct.setSimpleColumn(0, offset, position.getWidth(), offset + spaceneeded);
ct.addElement(content);
ct.go();
canvas.addTemplate(tmp, position.getLeft(), position.getBottom());

```

In this case, I used a PdfTemplate to clip the content.

## How to resize a PdfPTable to fit the page?



I am generating a document that contains a table. I need to make sure that this table never exceeds one page in size, regardless of the amount of content in the cells. Is there a way to do this?

Posted on StackOverflow on Feb 19, 2013<sup>362</sup> by Jannis Alexakis<sup>363</sup>

If you want a table to fit a page, you should create the table before even thinking about page size and ask the table for its height as is done in the [TableHeight](#)<sup>364</sup> example. Note that the `getTotalHeight()` method returns 0 unless you define the width of the table. This can be done like this:

<sup>362</sup><http://stackoverflow.com/questions/14958396/resize-pdfptable-to-fit-page>

<sup>363</sup><http://stackoverflow.com/users/637791/jannis-alexakis>

<sup>364</sup><http://itextpdf.com/examples/iiapdf?id=80>

```
table.setTotalWidth(width);
table.setLockedWidth(true);
```

Now you can create a Document with size Rectangle(0, 0, width + margin \* 2, getTotalHeight() + margin \* 2) and the table should fit the document exactly when you add it with the writeSelectedRows() method.

If you don't want a custom page size, then you need to create a PdfTemplate with the size of the table and add the table to this template object. Then wrap the template object in an Image and use scaleToFit() to size the table down.

```
public static void main(String[] args) throws DocumentException, FileNotFoundException, IOException {
    String TARGET = "temp.pdf";
    Document document = new Document(PageSize.A4);
    PdfWriter writer
        PdfWriter.getInstance(document, new FileOutputStream(TARGET));
    document.open();
    PdfPTable table = new PdfPTable(7);

    for (int i = 0; i < 700; i++) {
        Phrase p = new Phrase("some text");
        PdfPCell cell = new PdfPCell();
        cell.addElement(p);
        table.addCell(cell);
    }

    table.setTotalWidth(PageSize.A4.getWidth());
    table.setLockedWidth(true);
    PdfContentByte canvas = writer.getDirectContent();
    PdfTemplate template = canvas.createTemplate(
        table.getTotalWidth(), table.getTotalHeight());
    table.writeSelectedRows(0, -1, 0, table.getTotalHeight(), template);
    Image img = Image.getInstance(template);
    img.scaleToFit(PageSize.A4.getWidth(), PageSize.A4.getHeight());
    img.setAbsolutePosition(
        0, (PageSize.A4.getHeight() - table.getTotalHeight()) / 2);
    document.add(img);
    document.close();
}
```

## What's an easy to print "first right, then down"?



I would like to print a PdfPTable containing many more rows and columns than fit in one page of the PDF.

I would like to print "first right, then down", which is similar to row-major order. Print the first page-full of rows, but all columns for those rows. Then print the next page-full of rows, and again all columns for those rows.

As I understand it, iText can handle a PdfPTable containing more rows than fit on a page, but can't handle more columns than fit on a page. So I split the columns myself. But that means that I end up printing "first down, then right", which is not what I want.

Is there an easy way to do what I want? Note that I do not know the row heights ahead of time, because I call `setMinimumHeight()` on the PdfPCell objects.

The only way I can think of, which is clumsy, is to first split columns, as before. Then, for each sequence of columns that fit on one page, add rows one by one to a PdfPTable as long as they fit. When they don't, that's where we need to split the rows. At the end of this, we end up with PdfPTable objects for each set of rows and columns. We need to keep all of them in memory and add them to the PDF Document in the right (row-major) order.

This seems rather inelegant, so I wanted to check if there's a cleaner solution.

Posted on StackOverflow on Feb 28, 2014<sup>365</sup> by Kartick Vaddadi<sup>366</sup>

If you have really large tables, the most elegant way to maintain the overview and to distribute the table over different pages, is by adding the table to one really large PdfTemplate object and then afterwards add clipped parts of that PdfTemplate object to different pages.

That's what I've done in the [TableTemplate](#)<sup>367</sup> example.

I create a table with 15 columns a total width of 1500pt.

```
PdfPTable table = new PdfPTable(15);
table.setTotalWidth(1500);
PdfPCell cell;
for (int r = 'A'; r <= 'Z'; r++) {
    for (int c = 1; c <= 15; c++) {
        cell = new PdfPCell();
        cell.setFixedHeight(50);
        cell.addElement(new Paragraph(
            String.valueOf((char) r) + String.valueOf(c)));
    }
}
```

<sup>365</sup><http://stackoverflow.com/questions/22093993/itext-whats-an-easy-to-print-first-right-then-down>

<sup>366</sup><http://stackoverflow.com/users/2893711/kartick-vaddadi>

<sup>367</sup><http://itextpdf.com/sandbox/tables/TableTemplate>

```
    table.addCell(cell);
}
}
```

Now I write this table to a Form XObject that measures 1500 by 1300. Note that 1300 is the total height of the table. You can get it like this: `table.getTotalHeight()` or you can multiply the 26 rows by the fixed height of 50 pt.

```
PdfContentByte canvas = writer.getDirectContent();
PdfTemplate tableTemplate = canvas.createTemplate(1500, 1300);
table.writeSelectedRows(0, -1, 0, 1300, tableTemplate);
```

Now that you have the Form XObject, you can reuse it on every page like this:

```
PdfTemplate clip;
for (int j = 0; j < 1500; j += 500) {
    for (int i = 1300; i > 0; i -= 650) {
        clip = canvas.createTemplate(500, 650);
        clip.addTemplate(tableTemplate, -j, 650 - i);
        canvas.addTemplate(clip, 36, 156);
        document.newPage();
    }
}
```

The result is [table\\_template.pdf](#)<sup>368</sup> where we have cells A1 to M5 on page 1, cells N1 to Z5 on page 2, cells A6 to M10 on page 3, cells N6 to Z10 on page 4, cells A11 to M15 on page 5 and cells N11 to Z15 on page 6.

---

<sup>368</sup>[http://itextpdf.com/sites/default/files/table\\_template.pdf](http://itextpdf.com/sites/default/files/table_template.pdf)

## How to invoke a page break for nested tables?



I have one PdfPTable with one column. One page fits 50 rows. If I add some text data to table (for an example, 300 rows), the report looks fine. When I nest a PdfPTable with a limited number of rows into a cell, all works fine too:

```
/-----
20 string rows
inner table (20 rows)
10 string rows
/-----
...additional rows
```

But, when my inner table has more rows, the table splits on the first page, and starts the inner table on the second page:

```
/-----
20 string rows
whitespace for 30 rows
/-----
inner table (50 rows from 90)
/-----
inner table (40 rows from 90)
..additional data
```

I need this:

```
/-----
20 string rows
inner table (30 rows from 90)
/-----
inner table (50 rows from 90)
/-----
inner table (10 rows from 80)
..additional data
```

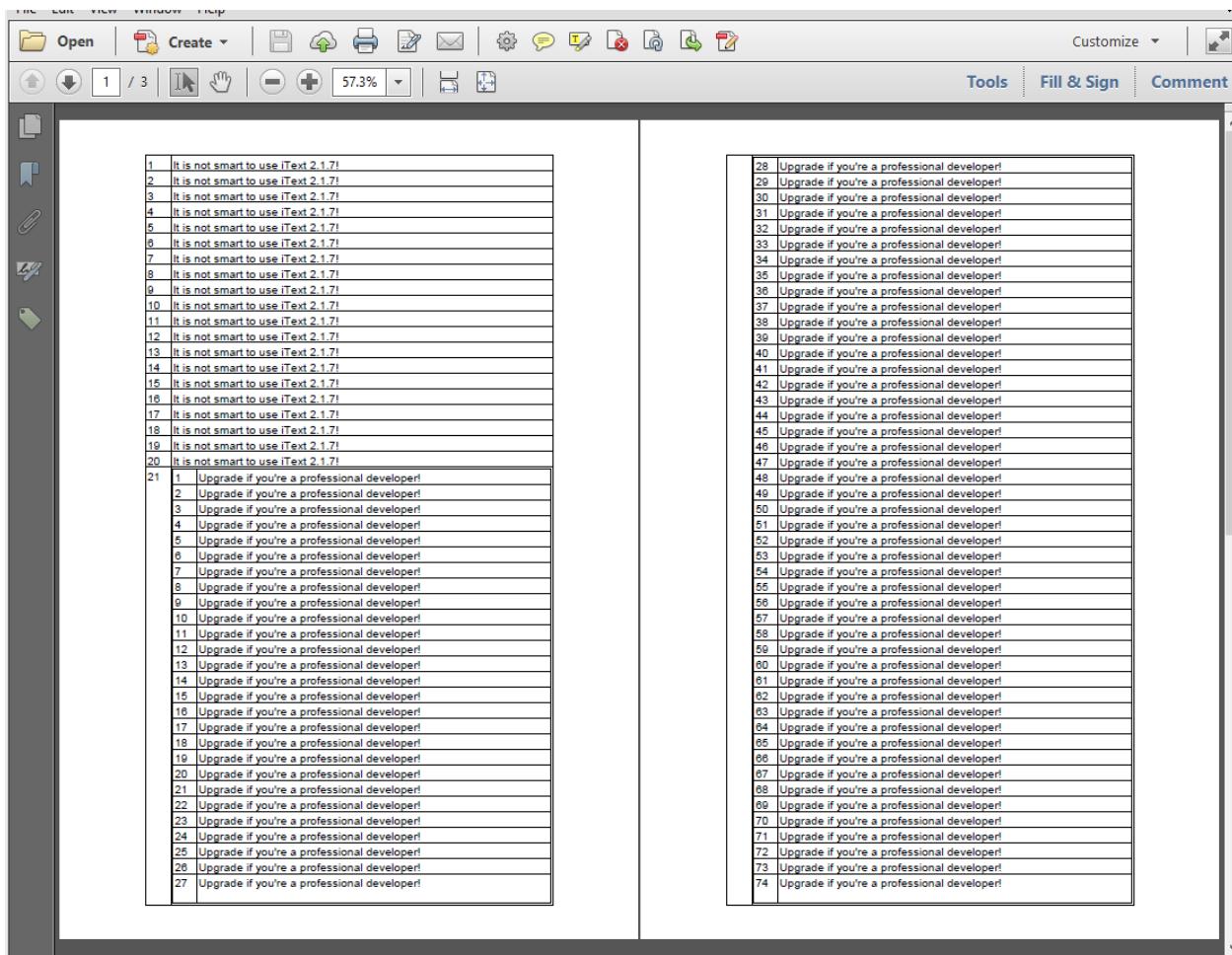
Posted on StackOverflow on Nov 28, 2012<sup>369</sup> by degr<sup>370</sup>

---

<sup>369</sup><http://stackoverflow.com/questions/28503491/large-table-in-table-cell-invoke-page-break>

<sup>370</sup><http://stackoverflow.com/users/2556867/degr>

Please take a look at the [NestedTables2<sup>371</sup>](#) example. In this example, we tell iText that it's OK to split cells as soon as a row doesn't fit on a page:



Nested table

This is not the default: by default, iText will try to keep a row intact by forwarding it to the next page. Only if the row doesn't fit the page, the row will be split.

Changing the default involves adding a single line to your code. That line is called: `table.setSplitLate(false)`;

This is the full method that created the PDF shown in the screen shot:

<sup>371</sup><http://itextpdf.com/sandbox/tables/NestedTables2>

```

public void createPdf(String dest) throws IOException, DocumentException {
    Document document = new Document();
    PdfWriter.getInstance(document, new FileOutputStream(dest));
    document.open();
    PdfPTable table = new PdfPTable(2);
    table.setSplitLate(false);
    table.setWidths(new int[]{1, 15});
    for (int i = 1; i <= 20; i++) {
        table.addCell(String.valueOf(i));
        table.addCell("It is not smart to use iText 2.1.7!");
    }
    PdfPTable innertable = new PdfPTable(2);
    innertable.setWidths(new int[]{1, 15});
    for (int i = 0; i < 90; i++) {
        innertable.addCell(String.valueOf(i + 1));
        innertable.addCell("Upgrade if you're a professional developer!");
    }
    table.addCell("21");
    table.addCell(innertable);
    for (int i = 22; i <= 40; i++) {
        table.addCell(String.valueOf(i));
        table.addCell("It is not smart to use iText 2.1.7!");
    }
    document.add(table);
    document.close();
}

```

## How to make a footer stick to bottom of every pdf page?



I'm using ItextSharp to create a PDF with content from my database. This content is visualized as a table. For this table, I define a footer that sticks to the bottom of every page except for the last page because there might not be enough content to "push" the footer down. How can I make sure that my footer stay at the bottom of every page no matter what?

Posted on StackOverflow on Feb 17, 2015 <sup>372</sup> by Carsten Løvbo Andersen<sup>373</sup>

Please take a look at the source code of `PdfPTable`, more specifically at the `SetExtendLastRow()`

---

<sup>372</sup><http://stackoverflow.com/questions/28561490/itextsharp-make-footer-stick-at-bottom-of-every-pdf-page>

<sup>373</sup><http://stackoverflow.com/users/2943218/carsten-løvbo-andersen>

method:

```
/*
 * When set the last row on every page will be extended to fill
 * all the remaining space to the bottom boundary; except maybe the
 * final row.
 *
 * @param extendLastRows true to extend the last row on each page;
 *           false otherwise
 * @param extendFinalRow false if you don't want to extend
 *           the final row of the complete table
 * @since iText 5.0.0
 */
public void SetExtendLastRow(bool extendLastRows, bool extendFinalRow) {
    extendLastRow[0] = extendLastRows;
    extendLastRow[1] = extendFinalRow;
}
```

If you want the last row to extend to the bottom of the last page, you need to set `extendFinalRow` to true (the default for `extendLastRows` and `extendFinalRow` is false).

## How to add continue on next page / continued from next page to a table?



I need to build a table to which I'm adding all the content dynamically by iterating over a list of records. Every record corresponds with a row in a `PdfPTable` and finally I add this `PdfPTable` to the Document.

When the table is split, I need to add a header on the next page that explains that this is the next page of a table that started on the previous page. Unfortunately, the `writer` nor the document give me a correct page number if my content moves on to the next page until the full table content is added to the document. As a result, I don't know when the content is moved on to next page during iteration process. `PdfWriter` is only showing me the page number before entering the loop. I need to get the updated page number whenever the table's content goes on to next page. I don't need this header to be on all pages. I need it only when the table is split and rows are carried over to the next page.

Posted on StackOverflow on Nov 28, 2012 <sup>374</sup> by degr<sup>375</sup>

I have written an example that produces something like this:

---

<sup>374</sup><http://stackoverflow.com/questions/28503491/large-table-in-table-cell-invokes-page-break>

<sup>375</sup><http://stackoverflow.com/users/2556867/degr>

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25
26	27	28	29	30
31	32	33	34	35
36	37	38	39	40
41	42	43	44	45
46	47	48	49	50
51	52	53	54	55
56	57	58	59	60
61	62	63	64	65
66	67	68	69	70
71	72	73	74	75
76	77	78	79	80
81	82	83	84	85
86	87	88	89	90
91	92	93	94	95
96	97	98	99	100
101	102	103	104	105
106	107	108	109	110
111	112	113	114	115
116	117	118	119	120
121	122	123	124	125
126	127	128	129	130
131	132	133	134	135
136	137	138	139	140
141	142	143	144	145
146	147	148	149	150
151	152	153	154	155
156	157	158	159	160

Continue on next page

Page 1

Table XYZ (Continued)				
161	162	163	164	165
166	167	168	169	170
171	172	173	174	175
176	177	178	179	180
181	182	183	184	185
186	187	188	189	190
191	192	193	194	195
196	197	198	199	200
201	202	203	204	205
206	207	208	209	210
211	212	213	214	215
216	217	218	219	220
221	222	223	224	225
226	227	228	229	230
231	232	233	234	235
236	237	238	239	240
241	242	243	244	245
246	247	248	249	250
251	252	253	254	255
256	257	258	259	260
261	262	263	264	265
266	267	268	269	270
271	272	273	274	275
276	277	278	279	280
281	282	283	284	285
286	287	288	289	290
291	292	293	294	295
296	297	298	299	300
301	302	303	304	305
306	307	308	309	310

Continue on next page

Page 2

Table XYZ (Continued)				
311	312	313	314	315
316	317	318	319	320
321	322	323	324	325
326	327	328	329	330
331	332	333	334	335
336	337	338	339	340
341	342	343	344	345
346	347	348	349	350

Page 3

As you can see, there is no header on the first page, there is only a header on the second and third page that says “Table XYZ (Continued)”. I even added a footer that says “Continue on next page”. As you can see, this footer only appears at the moment the table is split. It isn’t present on the last page. This wasn’t asked in the question, and it is easy to remove from my code.

You can find the complete code sample here: [SimpleTable5<sup>376</sup>](#)

These are the relevant parts:

```
PdfPTable table = new PdfPTable(5);
table.setWidthPercentage(100);
PdfPCell cell = new PdfPCell(new Phrase("Table XYZ (Continued)"));
cell.setColspan(5);
table.addCell(cell);
cell = new PdfPCell(new Phrase("Continue on next page"));
cell.setColspan(5);
table.addCell(cell);
table.setHeaderRows(2);
table.setFooterRows(1);
table.setSkipFirstHeader(true);
table.setSkipLastFooter(true);
for (int i = 0; i < 350; i++) {
```

<sup>376</sup><http://itextpdf.com/sandbox/tables/SimpleTable5>

```
    table.addCell(String.valueOf(i+1));
}
document.add(table);
```

If you only want a header:

```
PdfPTable table = new PdfPTable(5);
table.setWidthPercentage(100);
PdfPCell cell = new PdfPCell(new Phrase("Table XYZ (Continued)"));
cell.setColspan(5);
table.addCell(cell);
table.setHeaderRows(1);
table.setSkipFirstHeader(true);
for (int i = 0; i < 350; i++) {
    table.addCell(String.valueOf(i+1));
}

document.add(table);
```

The difference with your code, is that you add the following line to avoid that the header appears on the first page:

```
table.setSkipFirstHeader(true);
```

# Table events

We continue with some questions about tables of which the answer involves table or cell events.

## How to use a dotted line as a cell border?



I am trying to create a table with cells that have a dotted line for a border. How can I do this?

Posted on StackOverflow on Nov 21, 2013 <sup>377</sup> with user1913695<sup>378</sup>

I've made an example that solves your problem: [DottedLineCell](#)<sup>379</sup>; The resulting PDF is a document with two tables: [dotted\\_line\\_cell.pdf](#)<sup>380</sup>

For the first table, we use a table event:

```
class DottedCells implements PdfPTableEvent {  
    @Override  
    public void tableLayout(PdfPTable table, float[][] widths,  
                           float[] heights, int headerRows, int rowStart,  
                           PdfContentByte[] canvases) {  
        PdfContentByte canvas = canvases[PdfPTable.LINECANVAS];  
        canvas.setLineDash(3f, 3f);  
        float llx = widths[0][0];  
        float urx = widths[0][widths[0].length - 1];  
        for (int i = 0; i < heights.length; i++) {  
            canvas.moveTo(llx, heights[i]);  
            canvas.lineTo(urx, heights[i]);  
        }  
        for (int i = 0; i < widths.length; i++) {  
            for (int j = 0; j < widths[i].length; j++) {  
                canvas.moveTo(widths[i][j], heights[i]);  
                canvas.lineTo(widths[i][j], heights[i+1]);  
            }  
        }  
    }  
}
```

<sup>377</sup><http://stackoverflow.com/questions/20117321/dotted-line-for-cell-border>

<sup>378</sup><http://stackoverflow.com/users/1913695/user1913695>

<sup>379</sup><http://itextpdf.com/sandbox/tables/DottedLineCell>

<sup>380</sup>[http://itextpdf.com/sites/default/files/dotted\\_line\\_cell.pdf](http://itextpdf.com/sites/default/files/dotted_line_cell.pdf)

```

        }
        canvas.stroke();
    }
}

```

This is the most elegant way to draw the cell borders, as it uses only one `stroke()` operator for all the lines. Unfortunately, this solution isn't an option if you have tables with rowspans.

The second table uses a cell event:

```

class DottedCell implements PdfPCellEvent {
    @Override
    public void cellLayout(PdfPCell cell, Rectangle position,
        PdfContentByte[] canvases) {
        PdfContentByte canvas = canvases[PdfPTable.LINECANVAS];
        canvas.setLineDash(3f, 3f);
        canvas.rectangle(position.getLeft(), position.getBottom(),
            position.getWidth(), position.getHeight());
        canvas.stroke();
    }
}

```

With a cell event, a border is drawn around every cell. This means you'll have multiple `stroke()` operators and overlapping lines. However: this solution always works, also when the table has cells with a rowspan greater than one.

## How to create a table with rounded corners?



I have to create a table having rounded corners, something like it:

Empty warning - HelloMaven - May 14, 2014

Pag. 2/6

### Cell with rounded border

Can I do it with iTextSharp?

Posted on StackOverflow on May 14, 2014<sup>381</sup> by AndreaNobili<sup>382</sup>

<sup>381</sup><http://stackoverflow.com/questions/23650957/how-to-create-a-rounded-corner-table-using-itext-itextsharp>

<sup>382</sup><http://stackoverflow.com/users/1833945/andreanobili>

This is done using cell events.

Make sure that you don't add any "automated" borders to the cell, but draw the borders yourself in a cell event:

```
table.DefaultCell.Border = PdfPCell.NO_BORDER;
table.DefaultCell.CellEvent = new RoundedBorder();
```

The RoundedBorder class would then look like this:

```
class RoundedBorder : IPdfPCellEvent {
    public void CellLayout(PdfPCell cell, Rectangle rect, PdfContentByte[] canvas) {
        PdfContentByte cb = canvas[PdfPTable.BACKGROUNDCANVAS];
        cb.RoundRectangle(
            rect.Left + 1.5f,
            rect.Bottom + 1.5f,
            rect.Width - 3,
            rect.Height - 3, 4
        );
        cb.Stroke();
    }
}
```

You can of course fine-tune the values 1.5, 3 and 4 to get different effects.

## How to introduce rounded cells with a background color?



I have seen how to create cells with rounded borders, but is it possible to make a cell that will have no borders, but colored and rounded background.

Posted on StackOverflow on Nov 7, 2014 <sup>383</sup> by filipst<sup>384</sup>

To achieve that, you need cell events. I've provided different examples in my book. See for instance [calendar.pdf](#)<sup>385</sup>:

<sup>383</sup> <http://stackoverflow.com/questions/26798850/itextsharp-rounded-cell-background>

<sup>384</sup> <http://stackoverflow.com/users/2758774/filipst>

<sup>385</sup> <http://examples.itextpdf.com/results/part1/chapter05/calendar.pdf>



Colored cells

The Java code to create the white cells looks like this:

```
class CellBackground implements PdfPCellEvent {
    public void cellLayout(PdfPCell cell, Rectangle rect,
        PdfContentByte[] canvas) {
        PdfContentByte cb = canvas[PdfPTable.BACKGROUNDCANVAS];
        cb.roundRectangle(
            rect.getLeft() + 1.5f, rect.getBottom() + 1.5f, rect.getWidth() - 3,
            rect.getHeight() - 3, 4);
        cb.setCMYKColorFill(0x00, 0x00, 0x00, 0x00);
        cb.fill();
    }
}
```

In C#, the code looks like this:

```
class CellBackground : IPdfPCellEvent {
    public void CellLayout(
        PdfPCell cell, Rectangle rect, PdfContentByte[] canvas
    ) {
        PdfContentByte cb = canvas[PdfPTable.BACKGROUNDCANVAS];
        cb.RoundRectangle(
            rect.Left + 1.5f,
            rect.Bottom + 1.5f,
            rect.Width - 3,
            rect.Height - 3, 4
        );
        cb.SetCMYKColorFill(0x00, 0x00, 0x00, 0x00);
        cb.Fill();
    }
}
```

You can use:

```
CellBackground cellBackground = new CellBackground();
cell.CellEvent = cellBackground;
```

Now the `CellLayout()` method will be executed the moment the cell is rendered to a page.

## How to set background image in PdfPCell in iText?



I am currently using iText to generate PDF reports. I want to set a medium size image as a background in `PdfPCell` instead of using background color. Is this possible?

Posted on StackOverflow on Jun 11, 2014 <sup>386</sup> by user1283475<sup>387</sup>

You need to create your own implementation of the `PdfPCellEvent` interface, for instance:

```
class ImageBackgroundEvent implements PdfPCellEvent {

    protected Image image;

    public ImageBackgroundEvent(Image image) {
        this.image = image;
    }

    public void cellLayout(PdfPCell cell, Rectangle position,
        PdfContentByte[] canvases) {
        try {
            PdfContentByte cb = canvases[PdfPTable.BACKGROUNDCANVAS];
            image.scaleAbsolute(position);
            image.setAbsolutePosition(position.getLeft(), position.getBottom());
            cb.addImage(image);
        } catch (DocumentException e) {
            throw new ExceptionConverter(e);
        }
    }
}
```

Then you need to create an instance of this event and declare it to the cell that needs this background:

---

<sup>386</sup><http://stackoverflow.com/questions/24162974/how-to-set-background-image-in-pdfpcell-in-itext>

<sup>387</sup><http://stackoverflow.com/users/1283475/user1283475>

```
Image image = Image.getInstance(IMG1);
cell.setCellEvent(new ImageBackgroundEvent(image));
```

Take a look at the [ImageBackground<sup>388</sup>](#) example for the full code.

## How to get text and image in the same cell?



I have created a table which has 48 images. I put the images to the right side in the cell. Now I would like to put the number of the image in the same cell at the left side.

Here is my part of code :

```
for (int i = x; i <= 48; i += 4)
{
    int number = i + 4;
    imageFilePath = "images/image" + number + ".jpeg";
    jpg = iTextSharp.text.Image.GetInstance(imageFilePath);
    jpg.ScaleAbsolute(166.11023622047f, 124.724409448829f);
    jpg.BorderColor = BaseColor.BLUE;
    string numberofcard = i.ToString();
    cell = new PdfPCell(jpg);
    cell.FixedHeight = 144.56692913386f;
    cell.Border = 0;
    cell.HorizontalAlignment = Element.ALIGN_RIGHT;
    cell.VerticalAlignment = Element.ALIGN_MIDDLE;
    table5.AddCell(cell);
}
```

How could I insert the number of the image at the left corner of cell?

Posted on StackOverflow on [May 9, 2014<sup>389</sup>](#) by [NickName<sup>390</sup>](#)

You need to create a custom IPdfPCellEvent implementation.

---

<sup>388</sup><http://itextpdf.com/sandbox/tables/ImageBackground>

<sup>389</sup><http://stackoverflow.com/questions/23573083/text-and-image-in-same-cell-with-itextsharp>

<sup>390</sup><http://stackoverflow.com/users/1912983/nickname>

```

private class MyEvent : IPdfPCellEvent {
    string number;
    public MyEvent(string number) {
        this.number = number;
    }
    public void CellLayout(PdfPCell cell, Rectangle position, PdfContentByte[] canvases) {
        ColumnText.ShowTextAligned(
            canvases[PdfPTable.TEXTCANVAS],
            Element.ALIGN_LEFT,
            new Phrase(number),
            position.Left + 2, position.Top - 16, 0);
    }
}

```

As you can see, we add a `Phrase` with the content `number` at an absolute position. In this case: 2 user units apart from the left border of the cell and 16 user units below the top of the cell.

In your code snippet you then add:

```
cell.CellEvent = new my_event(n);
```

Where `n` is the string value of `number`. Now the number will be drawn every time a cell is rendered.

## Is it possible to attach multiple layout events to a PdfPCell?



I'm not sure if it's possible to set **multiple** events. I would like to separate different cell options in separate events based on my business logic. Sometimes I want to draw an ellipse in it, sometimes a square (or anything else). It would be nice if I could simply attach the events that I need.

Posted on StackOverflow on Mar 18, 2015 <sup>391</sup> by Robert<sup>392</sup>

Yes, you can add multiple cell events to a cell. This is the Java code of the `setCellEvent()` method:

---

<sup>391</sup><http://stackoverflow.com/questions/29120119/multiple-layout-events-in-itextsharp>

<sup>392</sup><http://stackoverflow.com/users/4587348/robert>

```

public void setCellEvent(PdfPCellEvent cellEvent) {
    if (cellEvent == null) {
        this.cellEvent = null;
    } else if (this.cellEvent == null) {
        this.cellEvent = cellEvent;
    } else if (this.cellEvent instanceof PdfPCellEventForwarder) {
        ((PdfPCellEventForwarder) this.cellEvent).addCellEvent(cellEvent);
    } else {
        PdfPCellEventForwarder forward = new PdfPCellEventForwarder();
        forward.addCellEvent(this.cellEvent);
        forward.addCellEvent(cellEvent);
        this.cellEvent = forward;
    }
}

```

If you pass `null`, then all existing events are removed from the cell. If no cell event was present, a new cell event is added. If there is already a cell event present, a `PdfPCellEventForwarder`<sup>393</sup> is created. This is a class that stores different cell events and that eventually will execute all these events one by one.



I found the answer thanks to your response. The `PdfPCellEventForwarder` class is publicly available in iTextSharp. An instance of this can be set to the `CellEvent` property of a `PdfPCell`. On the instance, one can call the `AddCellEvent` method.

iTextSharp (C#) is kept in sync with iText (Java), so the Java functionality also works for iTextSharp:

```

virtual public IPdfPCellEvent CellEvent {
    get {
        return this.cellEvent;
    }
    set {
        if (value == null) this.cellEvent = null;
        else if (this.cellEvent == null) this.cellEvent = value;
        else if (this.cellEvent is PdfPCellEventForwarder)
            ((PdfPCellEventForwarder)this.cellEvent).AddCellEvent(value);
        else {
            PdfPCellEventForwarder forward = new PdfPCellEventForwarder();
            forward.AddCellEvent(this.cellEvent);
            forward.AddCellEvent(value);
            this.cellEvent = forward;
        }
    }
}

```

---

<sup>393</sup><http://api.itextpdf.com/itext/com/itextpdf/text/pdf/events/PdfPCellEventForwarder.html>

```

        }
    }
}

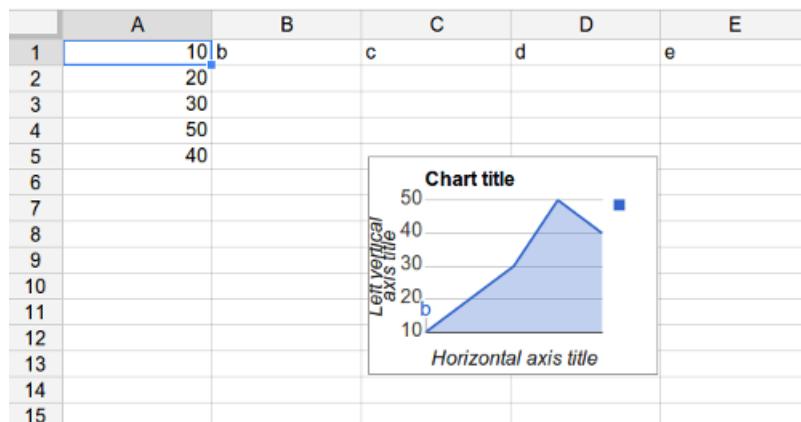
```

There is no need to create your own `PdfPCellEventForwarder` (although you may do so if you want to), iTextSharp will take care of creating a `PdfPCellEventForwarder` in your place if you add multiple events to a `PdfPCell`.

## How to precisely position an image on top of a PdfPTable?



I need to precisely position an image over a table, for example:



Precise positioning

Observe that the image overlays a specific set of rows and columns: it overlays row 5 and row 13 partially, and rows 8-12 completely; it also overlays columns C and D partially.

In other words, I want to say that the top-left of the image should be in the cell C5, and 4pt below and 6pt to the right of the top-left of C5.

How do I do this? I thought I'll add the text to the table, add the table to the Document, and then query the table to get the absolute positions of the rows and columns, and then add the image to the Document at that position, perhaps in direct content mode.

But the table may split across pages, because it may have hundreds of rows. In that case, I need to add the image to the right page, and at the right position.

Posted on StackOverflow on Feb 28, 2014 <sup>394</sup> by Kartick Vaddadi<sup>395</sup>

<sup>394</sup><http://stackoverflow.com/questions/22094289/itext-precisely-position-an-image-on-top-of-a-pdfptable>

<sup>395</sup><http://stackoverflow.com/users/2893711/kartick-vaddadi>

I've written some sample code that uses a table event. In the [AddOverlappingImage<sup>396</sup>](#) class, we create a table and declare a table event. This table event looks like this:

```
public class ImageContent implements PdfPTableEvent {  
    protected Image content;  
    public ImageContent(Image content) {  
        this.content = content;  
    }  
    public void tableLayout(PdfPTable table, float[][][] widths,  
                           float[] heights, int headerRows, int rowStart,  
                           PdfContentByte[] canvases) {  
        try {  
            PdfContentByte canvas = canvases[PdfPTable.TEXTCANVAS];  
            float x = widths[3][1] + 10;  
            float y = heights[3] - 10 - content.getScaledHeight();  
            content.setAbsolutePosition(x, y);  
            canvas.addImage(content);  
        } catch (DocumentException e) {  
            throw new ExceptionConverter(e);  
        }  
    }  
}
```

I pass an `Image` to the event class and in the `tableLayout()` method, I add the image to the 4th row and the 2nd column (the index starts counting at 0).

I hard-coded the cell position. Obviously, you should use some parameters, for instance a `List` of images and coordinates.

---

<sup>396</sup><http://itextpdf.com/sandbox/tables/AddOverlappingImage>

## Why is my cell event not triggered?



Please take a look at my code. I am expecting the cell event to be triggered for two cells, but it only triggers on the first cell. The difference appears to be that the cell event of the first cell is added to the cell *before* adding the cell to the table.

```
Document doc = new Document(PageSize.A4);
float twocm = Utilities.millimetersToPoints(20f);
doc.setMargins(twocm, twocm, twocm, twocm);
PdfWriter.getInstance(doc, new FileOutputStream(new File("test.pdf")));
try {
    doc.open();
    PdfPTable table = new PdfPTable(2);
    table.setTotalWidth(doc.right() - doc.left());
    table.setLockedWidth(true);
    table.setWidths( new int[] { 50, 50 } );
    PdfPCell leftCell = new PdfPCell();
    leftCell.addElement(new Paragraph("I am the left"));
    //Event added to cell before adding cell to table, it works.
    leftCell.setCellEvent(new MyCellEvent());
    PdfPCell rightCell = new PdfPCell();
    rightCell.addElement(new Paragraph("I am the right"));
    table.addCell(leftCell);
    table.addCell(rightCell);
    //Event added to cell after adding cell to table, doesn't work.
    rightCell.setCellEvent(new MyCellEvent());
    doc.add(table);
}
finally {
    doc.close();
}
static final class MyCellEvent implements PdfPCellEvent {
    @Override
    public void cellLayout(
        PdfPCell cell, Rectangle position, PdfContentByte[] canvases) {
        System.out.println("Cell event was called");
    }
}
```

The reason why this matters is that I need to do some measurements of the table (and other things) and pass values into the constructor of the cell renderer. I can't take the table measurements until *after* I've added the cell to the table; which is why I need to set the cell renderer after the cell has been added to the table.

Posted on StackOverflow on Apr 2, 2014 <sup>397</sup> by Richard<sup>398</sup>

---

<sup>397</sup><http://stackoverflow.com/questions/22813429/itext-cell-event-is-not-triggering>

The `cell` parameter of the `addCell()` method is final and changing `cell` after it has been added to the `PdfPTable` has no effect, because the object is being copied using the `PdfPCell` copy constructor of either the `PdfPHeaderCell` class or the `PdfPCell` class. We work on a copy because as soon as a cell is added to a table, we perform different operations on that cell. Should we do these operations on the original cell, then that cell can't be reused (and that is what some people do: they create a cell once and then reuse it).

There may be a workaround for your problem though. You could retrieve the copied cell from the row:

```
PdfPRow row = table.getRow(0);
PdfPCell cell = row.getCells()[1];
```

And then add the cell event to that `cell` instance. It's not elegant, but it might work.



Thanks Bruno, good explanation. And the work-around does work. cheers for that.

---

<sup>398</sup><http://stackoverflow.com/users/208687/richard>

# Page events

When creating a document from scratch, iText can trigger a series of events for instance when a new page starts or ends, when a paragraph is added, and so on...

## How to add text as a header or footer?



I'm creating a pdf with iText 5 and want to add a footer. I did everything like the book "iText in action" says. There are no errors but the footer doesn't show up. Can somebody tell me what I'm doing wrong?

```
class MyFooter extends PdfPageEventHelper {  
    public void onEndPage(PdfWriter writer, Document document) {  
        PdfContentByte cb = writer.getDirectContent();  
        ColumnText.showTextAligned(cb, Element.ALIGN_CENTER, footer(),  
            (document.right() - document.left()) / 2 + document.leftMargin(),  
            document.top() + 10, 0);  
    }  
    private Phrase footer() {  
        Font ffont = new Font(Font.FontFamily.UNDEFINED, 5, Font.ITALIC);  
        Phrase p = new Phrase("this is a footer");  
        return p;  
    }  
}  
...  
Document document = new Document(PageSize.A4);  
PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream(file));  
writer.setPageEvent(new MyFooter());
```

Posted on StackOverflow on January 5, 2015 <sup>399</sup> by Hiasl<sup>400</sup>

The problem you report can not be reproduced. I have taken your example and I create the [TextFooter<sup>401</sup>](#) example with this event:

<sup>399</sup><http://stackoverflow.com/questions/27780756/adding-footer-with-itext-doesnt-work>

<sup>400</sup><http://stackoverflow.com/users/4308508/hiasl>

<sup>401</sup><http://itextpdf.com/sandbox/events/TextFooter>

```
class MyFooter extends PdfPageEventHelper {  
    Font ffont = new Font(Font.FontFamily.UNDEFINED, 5, Font.ITALIC);  
  
    public void onEndPage(PdfWriter writer, Document document) {  
        PdfContentByte cb = writer.getDirectContent();  
        Phrase header = new Phrase("this is a header", ffont);  
        Phrase footer = new Phrase("this is a footer", ffont);  
        ColumnText.showTextAligned(cb, Element.ALIGN_CENTER,  
            header,  
            (document.right() - document.left()) / 2 + document.leftMargin(),  
            document.top() + 10, 0);  
        ColumnText.showTextAligned(cb, Element.ALIGN_CENTER,  
            footer,  
            (document.right() - document.left()) / 2 + document.leftMargin(),  
            document.bottom() - 10, 0);  
    }  
}
```

Note that I improved the performance by creating the `Font` and `Paragraph` instance only once. I also introduced a footer and a header. You claimed you wanted to add a footer, but in reality you added a header.

The `top()` method gives you the top of the page, so maybe you meant to calculate the y position relative to the `bottom()` of the page.

There was also an error in your `footer()` method:

```
private Phrase footer() {  
    Font ffont = new Font(Font.FontFamily.UNDEFINED, 5, Font.ITALIC);  
    Phrase p = new Phrase("this is a footer");  
    return p;  
}
```

You define a `Font` named `ffont`, but you don't use it. I think you meant to write:

```
private Phrase footer() {  
    Font ffont = new Font(Font.FontFamily.UNDEFINED, 5, Font.ITALIC);  
    Phrase p = new Phrase("this is a footer", ffont);  
    return p;  
}
```

Now when we look at the resulting PDF<sup>402</sup>, we clearly see the text that was added as a header and a footer to each page.

---

<sup>402</sup>[http://itextpdf.com/sites/default/files/page\\_footer.pdf](http://itextpdf.com/sites/default/files/page_footer.pdf)

## How to add a rectangle to every page of a document?



I'm using iText to create a PDF document. Right now I am trying to get a rectangle on every single page of the document but I'm not sure how to do this. I tried adding this at the end of my code:

```
PdfContentByte cb = writer.getDirectContent();
for (int pgCnt = 1; pgCnt <= writer.getPageNumber(); pgCnt++) {
    cb.saveState();
    cb.setColorStroke(new CMYKColor(1f, 0f, 0f, 0f));
    cb.setColorFill(new CMYKColor(1f, 0f, 0f, 0f));
    cb.rectangle(20,10,10,820);
    cb.fill();
    cb.restoreState();
}
```

but this only adds the rectangle on the last page and it kind of make sense because I'm not using the pgCnt anywhere. How can I specify that I want the rectangle on page number pgCnt, so I can add the rectangle on every page?

Posted on StackOverflow on Mar 19, 2013 <sup>403</sup> by Carla Stabille<sup>404</sup>

Please take a look at the entries for the keyword [Page events<sup>405</sup>](#) on the official iText site. You need to extend the [PdfPageEventHelper<sup>406</sup>](#) class and add your code to the `onEndPage()` method.

```
public void onEndPage(PdfWriter writer, Document document) {
    PdfContentByte cb = writer.getDirectContent();
    cb.saveState();
    cb.setColorStroke(new CMYKColor(1f, 0f, 0f, 0f));
    cb.setColorFill(new CMYKColor(1f, 0f, 0f, 0f));
    cb.rectangle(20,10,10,820);
    cb.fill();
    cb.restoreState();
}
```

Create an instance of your custom page event class, and declare it to the writer before opening the document:

<sup>403</sup><http://stackoverflow.com/questions/16638406/how-can-i-add-rectangle-on-every-page-of-a-document-using-itext>

<sup>404</sup><http://stackoverflow.com/users/1883606/carla-stabille>

<sup>405</sup><http://itextpdf.com/themes/keyword.php?id=204>

<sup>406</sup><http://api.itextpdf.com/itext/com/itextpdf/text/pdf/PdfPageEventHelper.html>

```
writer.setPageEvent(myPageEventInstance);
```

Now your rectangle will be drawn on every page, on top of the existing content. If you want the rectangle under the existing content: replace `getDirectContent()` with `getDirectContentUnder()`.

## How can I add an image to all pages of my PDF?



I have been trying to add an image to all pages using iTextSharp. The image needs to be OVER all content of every page. I have used the following code below all the otherdoc.add()

```
Document doc = new Document(iTextSharp.text.PageSize.A4, 10, 10, 30, 1);
PdfWriter writer = PdfWriter.GetInstance(doc, new FileStream(Server.MapPath("~/p\
df/" + fname), FileMode.Create));
doc.Open();
Image image = Image.GetInstance(Server.MapPath("~/images/draft.png"));
image.SetAbsolutePosition(12, 300);
writer.DirectContent.AddImage(image, false);
doc.Close();
```

The above code only inserts an image in the last page. Is there any way to insert the image in the same way in all pages?

Posted on StackOverflow on Feb 20, 2014 <sup>407</sup> by Neville Nazerane<sup>408</sup>

It's normal that the image is only added once; after all: you're adding it only once.

You should create a document in 5 steps and add an event in step 2:

```
// step 1
Document document = new Document();
// step 2
PdfWriter writer = PdfWriter.GetInstance(document, stream);
MyEvent event = new MyEvent();
writer.PageEvent = event;
// step 3
document.Open();
// step 4
// Add whatever content you want to add
// step 5
document.Close();
```

<sup>407</sup><http://stackoverflow.com/questions/21908651/add-an-image-in-all-pages-of-pdf>

<sup>408</sup><http://stackoverflow.com/users/991609/neville-nazerane>

You have to write the MyEvent class yourself:

```
protected class MyEvent : PdfPageEventHelper {  
  
    Image image;  
  
    public override void OnOpenDocument(PdfWriter writer, Document document) {  
        image = Image.GetInstance(Server.MapPath("~/images/draft.png"));  
        image.SetAbsolutePosition(12, 300);  
    }  
  
    public override void OnEndPage(PdfWriter writer, Document document) {  
        writer.DirectContent.AddImage(image);  
    }  
}
```

The OnEndPage() in class MyEvent will be triggered every time the PdfWriter has finished a page. Hence the image will be added on every page.

**Caveat:** it is important to create the image object outside the OnEndPage() method, otherwise the image bytes risk being added as many times as there are pages in your PDF (leading to a bloated PDF).

## How to set a fixed background image for all my pages?



On Button Click, I generate 4 pages on my PDF, i added this image to provide a background image

```
string imageFilePath = parent + "/Images/bg_image.jpg";  
iTextSharp.text.Image jpg = iTextSharp.text.Image.GetInstance(imageFilePath);  
jpg.ScaleToFit(1700, 1000);  
jpg.Alignment = iTextSharp.text.Image.UNDERLYING;  
jpg.SetAbsolutePosition(0, 0);  
document.Add(jpg);
```

It works only with 1 page, but when I generate a PDF that contains many records and have several pages, the bg image is only at the last page. I want to apply the background image to all of the pages.

Posted on StackOverflow on Nov 1, 2014 <sup>409</sup> by dandy<sup>410</sup>

<sup>409</sup><http://stackoverflow.com/questions/26688288/set-a-fix-background-image-for-all-my-pages-in-pdf-itext-asp-c-sharp>

<sup>410</sup><http://stackoverflow.com/users/4131886/dandy>

It is normal that the background is added only once, because you're adding it only once.

If you want to add content to every page, you should not do that *manually* because you don't know when a new page will be created by iText. Instead you should use a page event.

The idea is to create an implementation of the `PdfPageEvent` interface, for instance by extending the `PdfPageEventHelper` class and overriding the `OnEndPage()` method:

```
class TemplateHelper : PdfPageEventHelper {
    private Stationery instance;
    public TemplateHelper() { }
    public TemplateHelper(Stationery instance) {
        this.instance = instance;
    }
    /**
     * @see com.itextpdf.text.pdf.PdfPageEventHelper#onEndPage(
     *      com.itextpdf.text.pdf.PdfWriter, com.itextpdf.text.Document)
     */
    public override void OnEndPage(PdfWriter writer, Document document) {
        writer.DirectContentUnder.AddTemplate(instance.page, 0, 0);
    }
}
```

In this case, we add a `PdfTemplate`, but it is very easy to add an `Image` replacing the `Stationery` instance with an `Image` instance and replacing the `AddTemplate()` method with the `AddImage()` method.

Once you have an instance of your custom page event, you need to declare it to the `PdfWriter` instance:

```
writer.PageEvent = new TemplateHelper(this);
```

From that moment on, your `OnEndPage()` method will be executed each time a page is finalized.

**Warning:** as documented you *shall not* use the `OnStartPage()` method to add content in a page event!

If we adapt the above example to your requirement, the final result would look more or less like this:

```

class ImageBackgroundHelper : PdfPageEventHelper {
    private Image img;
    public ImageBackgroundHelper(Image img) {
        this.img = img;
    }
    /**
     * @see com.itextpdf.text.pdf.PdfPageEventHelper#onEndPage(
     *      com.itextpdf.text.pdf.PdfWriter, com.itextpdf.text.Document)
     */
    public override void OnEndPage(PdfWriter writer, Document document) {
        writer.DirectContentUnder.AddImage(img);
    }
}

```

Now you can use this event like this:

```

string imageFilePath = parent + "/Images/bg_image.jpg";
iTextSharp.text.Image jpg = iTextSharp.text.Image.GetInstance(imageFilePath);
jpg.ScaleToFit(1700, 1000);
jpg.SetAbsolutePosition(0, 0);
writer.PageEvent = new ImageBackgroundHelper(jpg);

```

Note that 1700 and 1000 seems quite big. Are you sure those are the dimensions of your page?

## Why do I get a System.StackOverflowException in the OnEndPage() event handler?



In the code below, you can see that I've overridden the `OnEndPage` event and tried to add a paragraph to the document. However, I get a `System.StackOverflowException` error when attempting to run the code. Does anyone have any idea why this is happening and how I can fix it?

```

public override void OnEndPage(PdfWriter writer, Document document)
{
    base.OnEndPage(writer, document);
    Paragraph p = new Paragraph("Paragraph");
    document.Add(p);
}

```

Posted on StackOverflow on Jun 18, 2014 <sup>411</sup> by Ognjen Koprivica<sup>412</sup>

---

<sup>411</sup><http://stackoverflow.com/questions/24285547/system-stackoverflowexception-in-onendpage-event-handler>

<sup>412</sup><http://stackoverflow.com/users/2319891/ognjen-koprivica>

It is **forbidden** to use `document.Add()` in a page event. The `document` object passed as a parameter is actually a `PdfDocument` object. It is not the `Document` you have created in your code, but an internal object that is used by iText. You should use it for read-only purposes only.

If you want to add content in the `OnEndPage()` method, you need the `writer`, for instance `writer.DirectContent`.

## How to introduce multiple PdfPageEventHelper instances?



I have to generate a large amount of different types of documents using the iTextSharp library, all have things in common, some have common headers, page counts, watermarks my initial thought was to have different `PdfPageEventHelper` subclasses for example `WatermarkPdfPageEventHelper`, `OrderHeaderPdfPageEventHelper`, `PageNumberPdfPageEventHelper`, and so on, and apply them when necessary to compose the documents but `PageEvent` is not really an event but an instance of only one `IPdfEvent`, what is the correct way to implement this?

Posted on StackOverflow on Mar 6, 2013 <sup>413</sup> by Marc<sup>414</sup>

Page events can be cumulative. In iText (Java), you can do this:

```
writer.setPageEvent(watermarkevent);
writer.setPageEvent(headerevent);
writer.setPageEvent(footerevent);
```

Internally, a `PdfPageEventForwarder` is created. This object will make sure that each event is triggered in the order you added them.

When you want to remove the events, you just need to do this:

```
writer.setPageEvent(null);
```

In your case, you could create your own `PdfPageEventForwarder` instances, creating different combinations of page events.

I'm pretty sure you can do the same thing in iTextSharp although there may be slight differences in the class and method names.

---

<sup>413</sup><http://stackoverflow.com/questions/15250459/using-multiple-pdfpageeventhelper>

<sup>414</sup><http://stackoverflow.com/users/329957/marc>

## How to check for an event and remove it?



Is there a way to check to see if a page event has already been added to a `PdfWriter` object? If so, can you also remove that page event?

Posted on StackOverflow on [Apr 15, 2014](#) <sup>415</sup> by IyaTaisho<sup>416</sup>

In the Java version of iText, there's a method `getPageEvent()` available in `PdfWriter`. There should be a `GetPageEvent()` or `PageEvent` in iTextSharp that you can use to find out if there is a page event present.

To remove an existing page event, you need to set the page event to `null`. Adding an extra page event won't replace the existing page event, but add an extra event that will be triggered along with the original event(s).

## How to add a text to the left and to the right in a header?



I need to add more than one header and footer to my document. In header I want title heading on left and some text in the center.

Likewise in footer I need to print my company name on the left side, the page number in the middle and some info regarding the contents in my table to the right.

Posted on StackOverflow on [Feb 20, 2013](#) <sup>417</sup> by Vignesh Vino<sup>418</sup>

Headers and footers should be added using 'page events'. Just create a class that extends `PdfPageEventHelper` and implement the `onEndPage()` method. People who read the documentation do not make the common mistake to use the `onStartPage()` method, but maybe you overlooked this, so I'm adding this as an extra caveat.

Add an instance of your class to the `PdfWriter` object with the `setPageEvent()` method.

I don't know if I understand what you mean by "multiple" headers. If you have more than one page event implementation, you can add them all using the `setPageEvent()` method and they will all be executed. If you want to switch from one page event implementation to another, you need to use `setPageEvent(null)` first.

<sup>415</sup><http://stackoverflow.com/questions/23087035/checking-for-an-event-in-itextsharp>

<sup>416</sup><http://stackoverflow.com/users/1631060/iyataisho>

<sup>417</sup><http://stackoverflow.com/questions/14977967/how-to-add-multiple-headers-and-footers-in-pdf-using-itext>

<sup>418</sup><http://stackoverflow.com/users/1862502/vignesh-vino>

Maybe you want the header to be different for different pages, just use a member-variable in your page event implementation and change it along the way. In one of the book examples named [MovieHistory2<sup>419</sup>](#), the text for the header is stored in a String array named `header`.

The position of the header depends on the page number:

```
public void onEndPage(PdfWriter writer, Document document) {
    Rectangle rect = writer.getBoxSize("art");
    switch(writer.getPageNumber() % 2) {
        case 0:
            ColumnText.showTextAligned(writer.getDirectContent(),
                Element.ALIGN_RIGHT, header[0],
                rect.getRight(), rect.getTop(), 0);
            break;
        case 1:
            ColumnText.showTextAligned(writer.getDirectContent(),
                Element.ALIGN_LEFT, header[1],
                rect.getLeft(), rect.getTop(), 0);
            break;
    }
    ColumnText.showTextAligned(writer.getDirectContent(),
        Element.ALIGN_CENTER, new Phrase(String.format("page %d", pagenumber)),
        (rect.getLeft() + rect.getRight()) / 2, rect.getBottom() - 18, 0);
}
```

For even page numbers, the header is added to the right; for odd page numbers to the left. The footer is centered as you can see.

You also mention a header table. If you want to use a table, please take a look at the [MovieCountries1<sup>420</sup>](#) example.

---

<sup>419</sup><http://itextpdf.com/examples/ii.php?id=103>

<sup>420</sup><http://itextpdf.com/examples/ii.php?id=104>

## How to add a table as a header?



I am working with **iTextSharp** trying to add an header and a footer to my generated PDF, but if I try to add an header that have width of 100% of my page I have some problem.

1. I have create a class named `PdfHeaderFooter` that extends the `iTextSharp PdfPageEventHelper` class,
2. into `PdfHeaderFooter` I have implemented the `OnStartPage()` method that generate the header:

```
public override void OnStartPage(PdfWriter writer, Document document) {  
    base.OnStartPage(writer, document);  
    PdfPTable tabHead = new PdfPTable(new float[] { 1F });  
    PdfPCell cell;  
    tabHead.WidthPercentage = 100;  
    cell = new PdfPCell(new Phrase("Header"));  
    tabHead.AddCell(cell);  
    tabHead.WriteSelectedRows(0, -1, 150, document.Top, writer.DirectContent);  
}
```

If I use something like `tabHead.TotalWidth = 300F`; instead of `tabHead.WidthPercentage = 100`; it works well, but if I try to set as 100% the width of the `tabHead` table (as I do in the previous example) when it call the `tabHead.WriteSelectedRows(0, -1, 150, document.Top, writer.DirectContent)` method it throws the following exception:

The table width must be greater than zero.

Why? What is the problem? How is it possible that the table have 0 size if I am setting it to 100%?

Posted on StackOverflow on Mar 12, 2014 <sup>[421](#)</sup> by [AndreaNobili](#)<sup>[422](#)</sup>

First things first: please do not add content to a PDF in the `OnStartPage()` event. Always use the `OnEndPage()` event.

As for your question: when using `writeSelectedRows()`, it doesn't make sense to set the width percentage to 100%. Setting the width percentage is meant for when you add a document using `document.add()` (which is a method you can't use in a page event). When using `document.add()`, iText calculates the width of the table based on the page size and the margins.

<sup>421</sup><http://stackoverflow.com/questions/23612105/how-to-set-a-100-header-in-itext-itextsharp-pdf>

<sup>422</sup><http://stackoverflow.com/users/1833945/andreamobili>

You are using `writeSelectedRows()`, which means you are responsible to define the size and the coordinates of the table.

If you want the table to span the complete width of the page, you need:

```
table.TotalWidth = document.Right - document.Left;
```

You're also using the wrong X-coordinate: you should use `document.Left` instead of `150`.

**Additional info:**

- The first two parameters define the start row and the end row. In your case, you start with row 0 which is the first row, and you don't define an end row (that's what `-1` means) in which case all rows are drawn.
- You omitted the parameters for the columns (there's a variation of the `writeSelectedRows()` that expects 7 parameters).
- Next you have the X and Y value of start coordinate for the table.
- Finally, you pass a `PdfContentByte` instance. This is the canvas on which you're drawing the table.

## How to create a table with 2 rows that can be used as a footer?



I want to add footer with 2 rows. The 1st row will have the document name with a background color. The 2nd row will have copyright notices. I tried to create this table using `ColumnText`, but I am not able to set the background color for the row (only the text is getting background color).

Posted on StackOverflow on Mar 2, 2014 <sup>423</sup> by Satesh S<sup>424</sup>

You can set the background of a cell using the `setBackgroundColor()` method and you can add a table at an absolute position using the `writeSelectedRows()` method.

Take a look at the [TableFooter<sup>425</sup>](#) example:

---

<sup>423</sup><http://stackoverflow.com/questions/22122340/creating-table-with-2-rows-in-pdf-footer-using-itext>

<sup>424</sup><http://stackoverflow.com/users/1716141/sathesh-s>

<sup>425</sup><http://itextpdf.com/sandbox/events/TableFooter>

```

PdfPTable table = new PdfPTable(1);
table.setTotalWidth(523);
PdfPCell cell = new PdfPCell(new Phrase("This is a test document"));
cell.setBackgroundColor(BaseColor.ORANGE);
table.addCell(cell);
cell = new PdfPCell(new Phrase("This is a copyright notice"));
cell.setBackgroundColor(BaseColor.LIGHT_GRAY);
table.addCell(cell);

```

If you have more than one cell in a row, you need to set the background for all cells. Note that I'm defining a total width for the table (523 is the width of the page minus the margins). The total width is needed because we'll add the table using `writeSelectedRows()`:

```
footer.writeSelectedRows(0, -1, 36, 64, writer.getDirectContent());
```

The resulting PDF looks like [this<sup>426</sup>](#). Make sure you define the margins of your page in such a way that the footer table doesn't overlap with the page content.

## How to generate a report with dynamic header in PDF using iTextsharp?



I'm generating a PDF report with iTextSharp, the header of the report has the information of the customer. In the body of the report contains a list of customer transactions.

My doubt is: How to generate a header dynamically for each client?

I need an example to generate a header dynamically to the report. Every new page Header need to have the data that client when changing client header should contain information on the new client.

Posted on StackOverflow on [Feb 7, 2014<sup>427</sup>](#) by [user3154466<sup>428</sup>](#)

I've written a small example in Java which you can easily adapt to C# as iText and iTextSharp share more or less the same syntax.

The example is called [VariableHeader<sup>429</sup>](#) and these are the most interesting snippets:

First I create a custom implementation of the `PdfPageEvent` interface (using `PdfPageEventHelper`). It's important to understand that you can't use the `onStartPage()` method, use the `onEndPage()` method instead.

---

<sup>426</sup>[http://itextpdf.com/sites/default/files/table\\_footer.pdf](http://itextpdf.com/sites/default/files/table_footer.pdf)

<sup>427</sup><http://stackoverflow.com/questions/21628429/itextsharp-how-to-generate-a-report-with-dynamic-header-in-pdf-using-itextsharp>

<sup>428</sup><http://stackoverflow.com/users/3154466/user3154466>

<sup>429</sup><http://itextpdf.com/sandbox/events/VariableHeader>

```
public class Header extends PdfPageEventHelper {  
  
    protected Phrase header;  
  
    public void setHeader(Phrase header) {  
        this.header = header;  
    }  
  
    @Override  
    public void onEndPage(PdfWriter writer, Document document) {  
        PdfContentByte canvas = writer.getDirectContentUnder();  
        ColumnText.showTextAligned(  
            canvas, Element.ALIGN_RIGHT, header, 559, 806, 0);  
    }  
}
```

As you can see, the text of the header is stored in a variable that can be changed using the `setHeader()` method we created.

The event is declared to the `PdfWriter` like this:

```
PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream(filename));  
Header event = new Header();  
writer.setPageEvent(event);
```

I change the header `Phrase` before invoking the `newPage()` method:

```
event.setHeader(new Phrase(String.format("THE FACTORS OF %s", i)));  
document.newPage();
```

In my simple example, I generate a document that lists the factors of all the numbers from 2 to 300: `variable_header.pdf`<sup>430</sup>. The header of each page says “THE FACTORS OF X” where X is the number of which the factors are shown on that page.

You can easily adapt this to show different customer names instead of numbers.

---

<sup>430</sup>[http://itextpdf.com/sites/default/files/variable\\_header.pdf](http://itextpdf.com/sites/default/files/variable_header.pdf)

## How to add a page number in the header of a PDF/A Level A file?



I am trying to create a document that conforms with PDF/A-1 level A. As long as I don't introduce a header or a footer using a page event, the code works fine. However, when I introduce text that is added at an absolute position, iText throws an exception telling me that the content I'm adding isn't tagged. How do I solve this problem?

Posted on StackOverflow on Dec 16, 2014 <sup>431</sup> by Praan<sup>432</sup>

I've written an example that creates a PDF/A document based on a CSV file. I'm adding a footer to this document using a simple page event implementation: [PdfA1A<sup>433</sup>](#)

The problem you are experiencing can be explained as follows: when you tell the `PdfWriter` that it needs to create Tagged PDF (using `writer.setTagged();`), iText will make sure that the appropriate tags are created when adding `Element` objects to the document. As long as you stick to using high-level objects, this will work.

However, the moment you introduce objects that are added at absolute positions, you take responsibility to correctly tag any content you add. In your case, you are adding a footer. Footers are not part of the "real content", hence they need to be marked as "artifacts".

In my example, I have adapted your page event implementation in a way that allows me to explain two different approaches:

In the first approach, I set the role at the level of the object:

```
Image total = Image.getInstance(t);
total.setRole(PdfName.ARTIFACT);
```

In the second approach, I mark content as an artifact at the lowest level:

```
PdfContentByte canvas = writer.getDirectContent();
canvas.beginMarkedContentSequence(PdfName.ARTIFACT);
table.writeSelectedRows(0, -1, 36, 30, canvas);
canvas.endMarkedContentSequence();
```

I use this second approach for `PdfPTable` because if I don't, I would have to tag all sub-elements of the table (every single row) as an artifact. If I didn't, iText would introduce `TR` elements inside an artifact (and that would be wrong).

<sup>431</sup><http://stackoverflow.com/questions/27500586/itext-page-number-in-header-within-pdf-a>

<sup>432</sup><http://stackoverflow.com/users/2790215/praan>

<sup>433</sup><http://itextpdf.com/sandbox/pdfa/PdfA1A>

## How to rotate a page while creating a PDF document?



I want to produce a PDF with pages in landscape. While I can set the size of the page to landscape using `document.setPageSize(PageSize.LETTER.rotate())`, it this doesn't achieve what I want. The content is still oriented left->right while I would like it to be bottom->top.

**Python (programming language)**  
From Wikipedia, the free encyclopedia

Python is a general-purpose, high-level programming language<sup>[1]</sup> whose design philosophy emphasizes code readability. Its syntax is said to be clear<sup>[2][3][4]</sup> and expressive.<sup>[5]</sup> Python has a large and comprehensive standard library.<sup>[6]</sup>

Python supports multiple programming paradigms, including object-oriented, imperative and functional programming styles. It features a fully dynamic system and automatic memory management. Python is often used as a scripting language, but is also used in a wide range of non-scripting contexts. Using third-party tools, Python code can be packaged into stand-alone executable programs. Python interpreters are available for many operating systems. Python is managed by the non-profit Python Software Foundation.

**Contents**

- 1 History
- 2 Features
- 3 Programming philosophy
- 4 Community and development
- 5 Name and logomarks
- 6 Usage
- 7 Syntax and semantics
- 7.1 Indentation
- 7.2 Data structures and control flow
- 7.3 Expressions
- 7.4 Methods
- 7.5 Typing
- 7.6 Mathematics
- 8 Implementations

**Python**  
multi-paradigm: object-oriented,  
imperative, functional,  
procedural, reflective

**Appeared in** 1991

**Designed by** Guido van Rossum

**Developer** Python Software Foundation

**Stable release** 3.3.0 /  
29 September 2012  
2.7.3 /  
11 April 2012

**Typing discipline** duck, dynamic, strong

**Major implementations** CPython, PyPy, IronPython, Jython

**Dialects** Cython, RPython, Stackless Python

**Influenced by** ABC<sup>[1]</sup> ALGOL 68<sup>[2]</sup> C<sup>[3]</sup>  
C++<sup>[4]</sup> Dylan<sup>[5]</sup> Haskell<sup>[6]</sup>  
Icon<sup>[7]</sup> Java<sup>[8]</sup> Lisp<sup>[9]</sup>  
Modula-2<sup>[10]</sup> Perl

**Influenced** Boo, Cobra, D, Falcon, Googy, JavaScript<sup>[11]</sup>, Ruby<sup>[12]</sup>

This is what I get

**Python (programming language)**  
From Wikipedia, the free encyclopedia

Python is a general-purpose, high-level programming language<sup>[1]</sup> whose design philosophy emphasizes code readability. Its syntax is said to be clear<sup>[2][3][4]</sup> and expressive.<sup>[5]</sup> Python has a large and comprehensive standard library.<sup>[6]</sup>

Python supports multiple programming paradigms, including object-oriented, imperative and functional programming styles. It features a fully dynamic system and automatic memory management. Python is often used as a scripting language, but is also used in a wide range of non-scripting contexts. Using third-party tools, Python code can be packaged into stand-alone executable programs. Python interpreters are available for many operating systems.

Python is the reference implementation of Python, is open source and is a community based development model, also known as an incubator. Python is managed by the non-profit Python Software Foundation.

**Contents**

- 1 History
- 2 Features
- 3 Programming philosophy
- 4 Community and development
- 5 Name and logomarks
- 6 Usage
- 7 Syntax and semantics
- 7.1 Indentation
- 7.2 Data structures and control flow
- 7.3 Expressions
- 7.4 Methods
- 7.5 Typing
- 7.6 Mathematics
- 8 Implementations

**Python**  
multi-paradigm: object-oriented,  
imperative, functional,  
procedural, reflective

**Appeared in** 1991

**Designed by** Guido van Rossum

**Developer** Python Software Foundation

**Stable release** 3.3.0 /  
29 September 2012  
2.7.3 /  
11 April 2012

**Typing discipline** duck, dynamic, strong

**Major implementations** CPython, PyPy, IronPython, Jython

**Dialects** Cython, RPython, Stackless Python

**Influenced by** ABC<sup>[1]</sup> ALGOL 68<sup>[2]</sup> C<sup>[3]</sup>  
C++<sup>[4]</sup> Dylan<sup>[5]</sup> Haskell<sup>[6]</sup>  
Icon<sup>[7]</sup> Java<sup>[8]</sup> Lisp<sup>[9]</sup>  
Modula-2<sup>[10]</sup> Perl

**Influenced** Boo, Cobra, D, Falcon, Googy, JavaScript<sup>[11]</sup>, Ruby<sup>[12]</sup>

This is what I want

I have been able to achieve the desired output by opening the PDF after it has been created and rotating it using iText, but I would like a solution that lets me rotate it immediately with iText after adding content to it.

Posted on StackOverflow on Jan 29, 2013 <sup>434</sup> by Natan Villaescusa<sup>435</sup>

<sup>434</sup><http://stackoverflow.com/questions/14591689/itext-rotate-page-content-while-creating-pdf>

<sup>435</sup><http://stackoverflow.com/users/264837/nathan-villaescusa>

Excellent question. If I was able to upvote it twice, I would!

You can achieve what you want with a PdfPageEvent:

```
public class RotateEvent extends PdfPageEventHelper {  
    public void onStartPage(PdfWriter writer, Document document) {  
        writer.addPageDictEntry(PdfName.ROTATE, PdfPage.SEASCAPE);  
    }  
}
```

You should use this RotateEvent right after you've defined the writer:

```
PdfWriter writer = PdfWriter.getInstance(document, os);  
writer.setPageEvent(new RotateEvent());
```

Note that I used SEASCAPE to get the orientation shown in your image.

All the possible values for the rotation are:

- PdfPage.PORTRAIT
- PdfPage.LANDSCAPE
- PdfPage.INVERTEDPORTRAIT
- PdfPage.SEASCAPE



That worked! I found that I didn't need to create the RotateEvent though, I just used `writer.addPageDictEntry(PdfName.ROTATE, PdfPage.SEASCAPE);` immediately after creating the PdfWriter since I am always creating a single page.

You're right, that works too, but each time the extra entry of the page dictionary was "used", it's gone. In your case, that doesn't matter because you're only creating one page!

## How to draw a line every 25 words?



I create a PDF using iText. The content consists of text and I need to draw a line to delimit this text every 25 words as shown in the following image:

01:31 NARRADORA

- 1 Durante más de cien años, los enfermos mentales de Gran Bretaña fueron  
aislados de la sociedad en grandes psiquiátricos victorianos. Esas personas nos  
dejaron sus testimonios.

01:42 JEAN

- 2 Recuerdo bien la primera vez que vi el hospital. Era grande y tenebroso, y tenía un reloj  
de torre. Y en ese momento pensé / "¿qué estoy haciendo acá, dónde me estoy  
metiendo?"

01:54 NARRADORA

- 3 En las últimas décadas de los psiquiátricos, los médicos trataron de curar y  
cuidar a los pacientes. Pero algunos de sus métodos fueron inhumanos.

02:03 HENRY

- 4 Era un desastre. / Hacían un agujero en el cráneo, llegaban al cerebro / y lo sacudían  
un poco. / Nada más.

02:12 NARRADORA

Example

I'm aware there's a way of that I can extract text and text positions once I have a finished PDF, but considering I'm writing the text to the PDF file, I guess there could be a way of adding these lines during the creation process without really having to find the text position, right?

Posted on StackOverflow on Feb 25, 2015 <sup>436</sup> by Matias Bello <sup>437</sup>

Please take a look at the [Every25Words<sup>438</sup>](#) example. In that example, I read a text file into a String with the `readFile()` method. I then split the text into words based on the occurrence of spaces, and

<sup>436</sup><http://stackoverflow.com/questions/28709603/draw-a-line-every-n-words-using-itextsharp>

<sup>437</sup><http://stackoverflow.com/users/4052286/matias-bello>

<sup>438</sup><http://itextpdf.com/sandbox/events/Every25Words>

I add each word one by one:

```
public void createPdf(String dest) throws IOException, DocumentException {
    Document document = new Document();
    PdfWriter writer =
        PdfWriter.getInstance(document, new FileOutputStream(dest));
    writer.setPageEvent(new WordCounter());
    writer.setInitialLeading(16);
    document.open();
    String[] words = readFile().split("\\s+");
    Chunk chunk = null;
    for (String word : words) {
        if (chunk != null) {
            document.add(new Chunk(" "));
        }
        chunk = new Chunk(word);
        chunk.setGenericTag("");
        document.add(chunk);
    }
    document.close();
}
```

The magic happens in this line:

```
writer.setPageEvent(new WordCounter());
...
chunk.setGenericTag("");
```

First we declare an instance of the WordCounter event. You may choose a better name for that class, as not only does it count words, it also draws a dashed line:

```
public class WordCounter extends PdfPageEventHelper {

    public int count = 0;

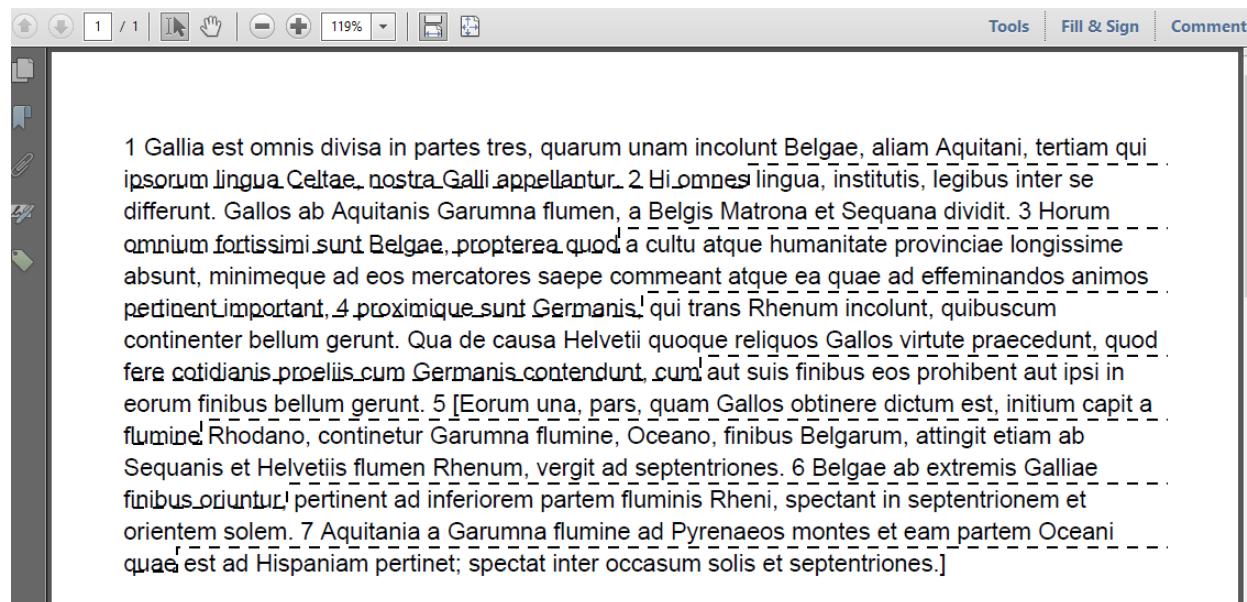
    @Override
    public void onGenericTag(
        PdfWriter writer, Document document, Rectangle rect, String text) {
        count++;
        if (count % 25 == 0) {
            PdfContentByte canvas = writer.getDirectContent();
```

```
    canvas.saveState();
    canvas.setLineDash(5, 5);
    canvas.moveTo(document.left(), rect.getBottom());
    canvas.lineTo(rect.getRight(), rect.getBottom());
    canvas.lineTo(rect.getRight(), rect.getTop());
    canvas.lineTo(document.right(), rect.getTop());
    canvas.stroke();
    canvas.restoreState();
}
}
}
```

Do you see what we do between the `saveState()` and `restoreState()` method? We define a dash pattern, we move to the left of the page, we construct a path to the right of the word, then we draw a short upwards line, to finish the path with a line to the right. Once the path is constructed, we stroke the line.

This `onGenericTag()` method will be triggered every time a `Chunk` is added on which we used the `setGenericTag` method.

This is what the result looks like [every25words.pdf](#)<sup>439</sup>:



Screen shot

<sup>439</sup><http://itextpdf.com/sites/default/files/every25words.pdf>

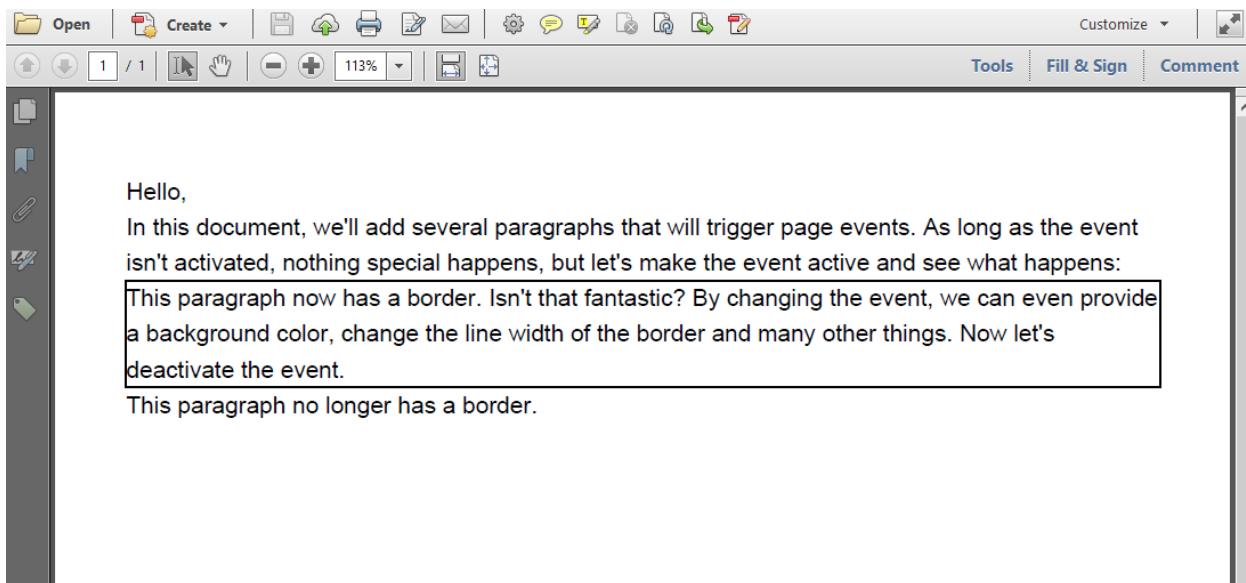
## How to add a border to a paragraph?



I've created a Paragraph using iText. Now I have to add a border to this paragraph, not to the whole document. How to do this?

Posted on StackOverflow on May 5, 2015 <sup>440</sup> by Amit Das<sup>441</sup>

Please take a look at the [BorderForParagraph<sup>442</sup>](#) example. It shows how to add a border for a paragraph like this:



A paragraph with a border

There is no method that allows you to create a border for a Paragraph, but you can create a PdfPageEvent implementation that allows you to draw a rectangle based on the start and end position of the Paragraph:

<sup>440</sup><http://stackoverflow.com/questions/30053684/how-to-add-border-to-paragraph-in-itext-pdf-library-in-java>

<sup>441</sup><http://stackoverflow.com/users/3978904/amit-das>

<sup>442</sup><http://itextpdf.com/sandbox/events/BorderForParagraph>

```
class ParagraphBorder extends PdfPageEventHelper {  
    public boolean active = false;  
    public void setActive(boolean active) {  
        this.active = active;  
    }  
  
    public float offset = 5;  
    public float startPosition;  
  
    @Override  
    public void onParagraph(  
        PdfWriter writer, Document document, float paragraphPosition) {  
        this.startPosition = paragraphPosition;  
    }  
  
    @Override  
    public void onParagraphEnd(  
        PdfWriter writer, Document document, float paragraphPosition) {  
        if (active) {  
            PdfContentByte cb = writer.getDirectContentUnder();  
            cb.rectangle(document.left(), paragraphPosition - offset,  
                document.right() - document.left(),  
                startPosition - paragraphPosition);  
            cb.stroke();  
        }  
    }  
}
```

As you can see, I introduced a boolean parameter named `active`. By default, I've set this parameter to `false`. I also create an `offset` (change this value to fine-tune the result) and a `startPosition` parameter.

Each time iText starts rendering a `Paragraph` object, the `startPosition` value is updated. Each time iText ends rendering a `Paragraph`, a rectangle is drawn if `active` is true (otherwise nothing happens).

We use this event like this:

```
public void createPdf(String dest) throws IOException, DocumentException {
    Document document = new Document();
    PdfWriter writer =
        PdfWriter.getInstance(document, new FileOutputStream(dest));
    ParagraphBorder border = new ParagraphBorder();
    writer.setPageEvent(border);
    document.open();
    document.add(new Paragraph("Hello,"));
    document.add(new Paragraph("In this document, we'll add several paragraphs that will trigger page events. As long as the event isn't activated, nothing special happens, but let's make the event active and see what happens:"));
    border.setActive(true);
    document.add(new Paragraph("This paragraph now has a border. Isn't that fantastic? By changing the event, we can even provide a background color, change the line width of the border and many other things. Now let's deactivate the event."));
    border.setActive(false);
    document.add(new Paragraph("This paragraph no longer has a border."));
    document.close();
}
```

As you can see, we declare the event to the `PdfWriter` using the `setPageEvent()` method. We activate the event like this:

```
border.setActive(true);
```

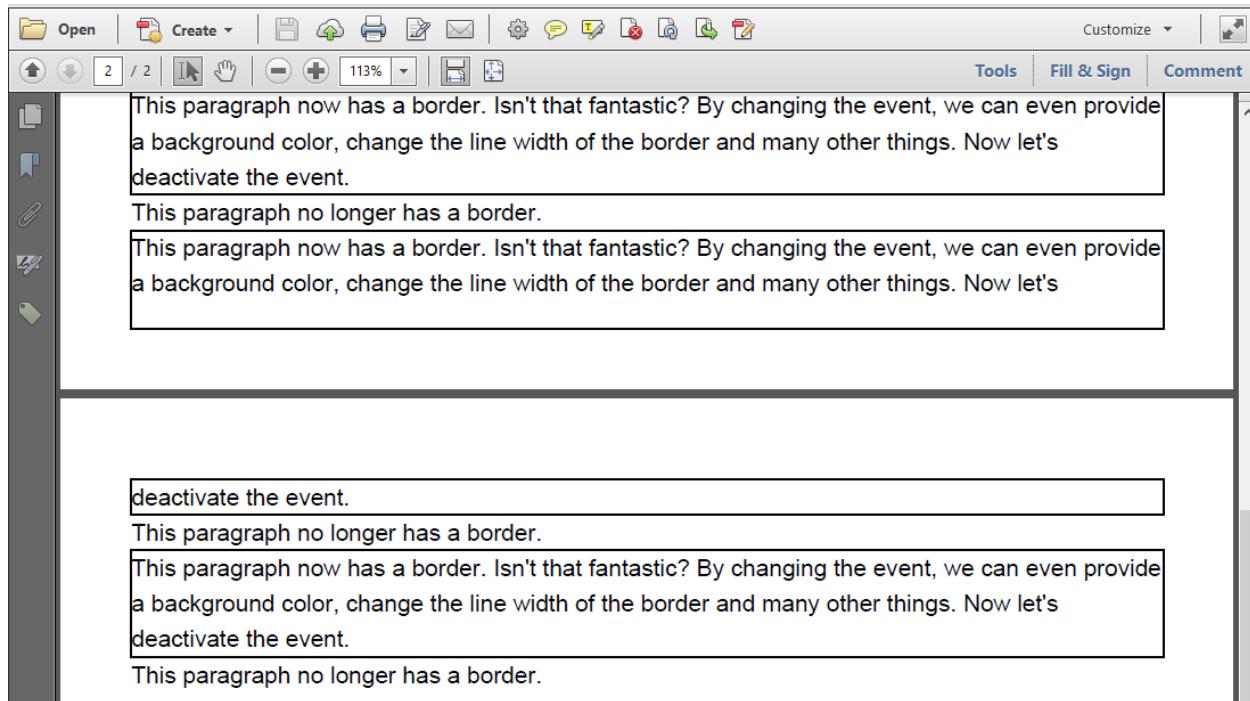
and we deactivate it like this:

```
border.setActive(false);
```

This is **only a proof of concept!** You will need to implement the `onStartPage()` and `onEndPage()` method if you want this to work for paragraphs that span more than one page. That's shown in `BorderForParagraph2`<sup>443</sup>:

---

<sup>443</sup><http://itextpdf.com/sandbox/events/BorderForParagraph2>



A paragraph distributed over 2 pages

The `onStartPage()` and `onEndPage()` implementation is a no-brainer:

```
class ParagraphBorder extends PdfPageEventHelper {  
    public boolean active = false;  
    public void setActive(boolean active) {  
        this.active = active;  
    }  
  
    public float offset = 5;  
    public float startPosition;  
  
    @Override  
    public void onStartPage(PdfWriter writer, Document document) {  
        startPosition = document.top();  
    }  
  
    @Override  
    public void onParagraph(PdfWriter writer, Document document,  
        float paragraphPosition) {  
        this.startPosition = paragraphPosition;  
    }  
}
```

```

@Override
public void onEndPage(PdfWriter writer, Document document) {
    if (active) {
        PdfContentByte cb = writer.getDirectContentUnder();
        cb.rectangle(document.left(), document.bottom() - offset,
                     document.right() - document.left(),
                     startPosition - document.bottom());
        cb.stroke();
    }
}

@Override
public void onParagraphEnd(PdfWriter writer, Document document,
                           float paragraphPosition) {
    if (active) {
        PdfContentByte cb = writer.getDirectContentUnder();
        cb.rectangle(document.left(), paragraphPosition - offset,
                     document.right() - document.left(),
                     startPosition - paragraphPosition);
        cb.stroke();
    }
}
}

```

## How to change the color of pages?



I'm creating report based on client activity. I'm creating this report with the help of the iText PDF library. I want to create the first two pages with a blue background color (for product name and disclaimer notes) and the remaining pages in white (without a background color). I colored two pages at the very beginning of report with blue using following code.

```

Rectangle pageSize = new Rectangle(PageSize.A4);
pageSize.setBackgroundColor(new BaseColor(84, 141, 212));
Document document = new Document( pageSize );

```

But when I move to 3rd page using `document.newpage()`, the page is still in blue. I can't change the color of 3rd page. I want to change the color of 3rd page onward to white. How can I do this using iText?

Posted on StackOverflow on May 13, 2015 <sup>444</sup> by Arun jk<sup>445</sup>

---

<sup>444</sup><http://stackoverflow.com/questions/30211043/change-the-color-of-pdf-pages-alternatively-using-itext-pdf-in-java>

<sup>445</sup><http://stackoverflow.com/users/4560055/arun-jk>

Please take a look at the [PageBackgrounds<sup>446</sup>](#) example. In this example, I create a blue background for page 1 and 2, and a grey background for all the subsequent even pages. See [page\\_backgrounds.pdf<sup>447</sup>](#)

How is this achieved? Well, using the same technique as used in my answer to this related question: [How to draw a border for PDF pages?<sup>448</sup>](#)

I create a page event like this:

```
public class Background extends PdfPageEventHelper {  
    @Override  
    public void onEndPage(PdfWriter writer, Document document) {  
        intpagenumber = writer.getPageNumber();  
        if (pagenumber % 2 == 1 && pagenumber != 1)  
            return;  
        PdfContentByte canvas = writer.getDirectContentUnder();  
        Rectangle rect = document.getPageSize();  
        canvas.setColorFill(pagenumber < 3 ?  
            BaseColor.BLUE : BaseColor.LIGHT_GRAY);  
        canvas.rectangle(rect.getLeft(), rect.getBottom(),  
            rect.getWidth(), rect.getHeight());  
        canvas.fill();  
    }  
}
```

As you can see, I first check for the page number. If it's an odd number and if it's not equal to 1, I don't do anything.

However, if I'm on page 1 or 2, or if the page number is even, I get the content from the `writer`, and I get the dimension of the page from the `document`. I then set the fill color to either blue or light gray (depending on the page number), and I construct the path for a rectangle that covers the complete page. Finally, I fill that rectangle with the fill color.

Now that we've got our custom `Background` event, we can use it like this:

```
PdfWriter writer =  
    PdfWriter.getInstance(document, new FileOutputStream(filename));  
Background event = new Background();  
writer.setPageEvent(event);
```

Feel free to adapt the `Background` class if you need a different behavior.

---

<sup>446</sup><http://itextpdf.com/sandbox/events/PageBackgrounds>

<sup>447</sup>[http://itextpdf.com/sites/default/files/page\\_backgrounds.pdf](http://itextpdf.com/sites/default/files/page_backgrounds.pdf)

<sup>448</sup><http://stackoverflow.com/questions/25749828/how-to-draw-border>

# Parsing XML and XHTML

There are a lot of questions about `HTMLWorker` on StackOverflow. Many of these questions remain unanswered as `HTMLWorker` has been abandoned in favor of XML Worker. `HTMLWorker` was initially meant as a parser for a small selection of HTML tags. People started using it as if it were a full-blown HTML to PDF converter and then complained because `HTMLWorker` doesn't support CSS parsing. The `HTMLWorker` code grew organically up until a point where it was no longer maintainable.

We started another project, called XML Worker. It can be used to convert XHTML to PDF. It's not an URL to PDF converter in the sense that it won't "print your web site to PDF". In HTML, you can encounter content at the end of the file that needs to be added at the start of the document. When this happens, one would expect that the start of the document is the first page. That isn't possible with iText as iText flushes finished pages to the `OutputStream` as soon as possible and there is no way to return to a previous page to add the extra content.

XML Worker is meant to create simple reports using an easy language such as HTML (and some CSS). It won't resolve ASP pages, nor execute JavaScript. It will only deal with finished XHTML.

## Why is it so difficult to convert XML to PDF?



Could anybody explain to me why is it so complicated to create a pdf file from xml sheet? Acrobat can create XML File but when I want to do this other way round it suddenly gets complicated. I would like to find some simple application which would allow me to create a pdf file out of xml. Is it possible?

Posted on StackOverflow on Jun 13, 2013 <sup>449</sup> by DDEX<sup>450</sup>

XML is a bunch of ingredients, PDF is the finished meal.

He or she who knows how to cook can create a wide variety of meals using the same ingredients. With a potato, he can create soup, mashed potatoes, crisps, french fries,... There's an almost endless list of possibilities.

He or she who can't cook, will stare at the potato and wonder: *How on earth can I turn this ugly vegetable into a nice croquette?*

The answer is: you need a recipe. That recipe could be an XSL:FO file, the XHTML specification, a DocBook implementation, an XFA template,... Without that recipe, you'll never be able to turn your XML into PDF.

---

<sup>449</sup><http://stackoverflow.com/questions/17081907/why-is-it-so-difficult-to-convert-xml-to-pdf>

<sup>450</sup><http://stackoverflow.com/users/1934834/ddex>

## How to adjust the page height to the content height?



I'm using iTextPDF + FreeMarker for my project. Basically I load and fill an HTML template with FreeMarker and then render it to pdf with iTextPDF's XMLWorker. The code works fine, but I have a problem with variable heights.

Let's say that we have data like this:

```
errorId = "ERROR-01"; systemId = "SYSTEM-01";
description = "A SHORT DESCRIPTION OF THE ISSUE"
```

Then the produced document looks like this:

```
TIMESTAMP: 2014/11/28 11:06:09
ERROR ID: ERROR-01
SYSTEM ID: SYSTEM-01
DESCRIPTION: A SHORT DESCRIPTION
OF THE ISSUE.
```

One page of data

But if the data looks like this:

```
errorId = "ERROR-01"; systemId = "SYSTEM-01";
description = "A SHORT DESCRIPTION OF THE ISSUE.
THIS IS MULTILINE AND IT SHOULD STAY ALL IN THE SAME PDF PAGE."
```

The produced document is:

```
TIMESTAMP: 2014/11/28 11:14:11
ERROR ID: ERROR-01
SYSTEM ID: SYSTEM-01
DESCRIPTION: A SHORT DESCRIPTION
OF THE ISSUE. THIS

IS MULTILINE AND IT
SHOULD STAY ALL IN
THE SAME PDF PAGE.
```

Two pages of data

As you can see, I now have two pages. I would like to have only one page that changes its height according to the content height.

Posted on StackOverflow on Nov 28, 2014 <sup>451</sup> by BackSlash<sup>452</sup>

<sup>451</sup><http://stackoverflow.com/questions/27186661/adjust-page-height-to-content-height>

<sup>452</sup><http://stackoverflow.com/users/1759845/backslash>

You can not change the page size after you have added content to that page. One way to work around this, would be to create the document in two passes: first create a document to add the content, then manipulate the document to change the page size. That would have been my first reply if I had time to answer immediately.

Now that I've taken more time to think about it, I've found a better solution that doesn't require two passes. Please take a look at [HtmlAdjustPageSize<sup>453</sup>](#)

For testing purposes, I used static String values for HTML and CSS:

```
public static final String HTML = "<table>" +
"<tr><td class=\"ra\">TIMESTAMP</td><td><b>2014-11-28 11:06:09</b></td></tr>" +
"<tr><td class=\"ra\">ERROR ID</td><td><b>ERROR-01</b></td></tr>" +
"<tr><td class=\"ra\">SYSTEM ID</td><td><b>SYSTEM-01</b></td></tr>" +
"<tr><td class=\"ra\">DESCRIPTION</td><td><b>TEST WITH A VERY," +
" VERY LONG DESCRIPTION LINE THAT NEEDS MULTIPLE LINES</b></td></tr>" +
"</table>";
public static final String CSS =
    "table {width: 200pt;} .ra { text-align: right;}";
public static final String DEST = "results/xmlworker/html_page_size.pdf";
```

You can see that I took HTML that looks more or less like the HTML you are dealing with.

I parse this HTML and CSS to an ElementList:

```
ElementList el = XMLWorkerHelper.parseToElementList(HTML, CSS);
```

I am now going to use this el twice:

1. I'll add the list to a ColumnText in simulation mode. This ColumnText isn't tied to any document or writer yet. The sole purpose to do this, is to know how much space I need vertically.
2. I'll add the list to a ColumnText for real. This ColumnText will fit exactly on a page of a size that I define using the results obtained in simulation mode.

Some code will clarify what I mean:

---

<sup>453</sup><http://itextpdf.com/sandbox/xmlworker/HtmlAdjustPageSize>

```

// I define a width of 200pt
float width = 200;
// I define the height as 10000pt (which is much more than I'll ever need)
float max = 10000;
// I create a column without a `writer` (strange, but it works)
ColumnText ct = new ColumnText(null);
ct.setSimpleColumn(new Rectangle(width, max));
for (Element e : el) {
    ct.addElement(e);
}
// I add content in simulation mode
ct.go(true);
// Now I ask the column for its Y position
float y = ct.getYLine();

```

The above code is useful for only one things: getting the y value that will be used to define the page size of the Document and the column dimension of the ColumnText that will be added for real:

```

Rectangle pagesize = new Rectangle(width, max - y);
// step 1
Document document = new Document(pagesize, 0, 0, 0, 0);
// step 2
PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream(file));
// step 3
document.open();
// step 4
ct = new ColumnText(writer.getDirectContent());
ct.setSimpleColumn(pagesize);
for (Element e : el) {
    ct.addElement(e);
}
ct.go();
// step 5
document.close();

```

Please download the full [HtmlAdjustPageSize.java<sup>454</sup>](#) code and change the value of `HTML`. You'll see that this leads to different page sizes.

---

<sup>454</sup><http://itextpdf.com/sites/default/files/HtmlAdjustPageSize.java>

## How to set line spacing when using XML Worker?



I am using XML Worker to parse an HTML string into a PDF Document, and cannot find a way to control the line spacing of the PDF being generated.

Posted on StackOverflow on Mar 26, 2015 <sup>455</sup> by Jon Z <sup>456</sup>

If you want to have a different line-height for different paragraphs, you have to define a different value for the `line-height` attribute in your CSS. I have made a very simple example with some very simple inline CSS:

The screenshot shows the iTextSharp XML Worker interface. At the top, there's a toolbar with various icons for file operations like Open, Save, Print, and Insert. Below the toolbar is a menu bar with 'Customize', 'Tools', 'Fill & Sign', and 'Com...'. The main area is divided into two sections: a preview window on the left and a code editor on the right. The preview window shows a page with three paragraphs of Latin text. The first paragraph has a line-height of 16pt, the second has 25pt, and the third has 13pt. The code editor shows the corresponding XML source code:

```

<p style="line-height: 16pt">Non eram nescius, Brute, cum, quae summis ingenii exquisitaque doctrina philosophi Graeco sermone tractavissent, ea Latinis litteris mandaremus, fore ut hic noster labor in varias reprehensiones incurreret. nam quibusdam, et iis quidem non admodum indoctis, totum hoc displicet philosophari. quidam autem non tam id reprehendunt, sed etiam tamquam operam ponendam in eo non: Graecis litteris, contemnentes Latinas, qui se dicant in postremo aliquos futuros suspicor, qui me ad alias litteras elegans, personae tamen et dignitatis esse negent.
```

Contra quos omnis dicendum breviter existimo. Quamvis satis responsum est eo libro, quo a nobis philosophia accusata et vituperata ab Hortensio. qui liber cum et titillat iudicare arbitrarer, plura suscepit veritus ne movere horum. Qui autem, si maxime hoc placeat, moderatius tamen in temperantiam postulant in eo, quod semel admissum est propter modum iustioribus utamur illis, qui omnino avocant infinitus modum constituant in reque eo meliore, quo magis sive enim ad sapientiam perveniri potest, non paranda [sapientia] est; sive hoc difficile est, tamen nec modus querendis defatigatio turpis est, cum id, quod queritur cum scribimus, quis est tam invitus, qui ab eo nos abducatur.

```

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
lineheight.html
1  <p style="line-height: 16pt">Non eram nescius, Brute, cum, quae summis ingenii exquisitaque doctrina philosophi Graeco sermone tractavissent, ea Latinis litteris mandaremus, fore ut hic noster labor in varias reprehensiones incurreret. nam quibusdam, et iis quidem non admodum indoctis, totum hoc displicet philosophari. quidam autem non tam id reprehendunt, sed etiam tamquam operam ponendam in eo non: Graecis litteris, contemnentes Latinas, qui se dicant in postremo aliquos futuros suspicor, qui me ad alias litteras elegans, personae tamen et dignitatis esse negent.
```

Contra quos omnis dicendum breviter existimo. Quamvis satis responsum est eo libro, quo a nobis philosophia accusata et vituperata ab Hortensio. qui liber cum et titillat iudicare arbitrarer, plura suscepit veritus ne movere horum. Qui autem, si maxime hoc placeat, moderatius tamen in temperantiam postulant in eo, quod semel admissum est propter modum iustioribus utamur illis, qui omnino avocant infinitus modum constituant in reque eo meliore, quo magis sive enim ad sapientiam perveniri potest, non paranda [sapientia] est; sive hoc difficile est, tamen nec modus querendis defatigatio turpis est, cum id, quod queritur cum scribimus, quis est tam invitus, qui ab eo nos abducatur.

Some XML and the resulting PDF

As you can see, the line-height of the paragraph starting with *Non eram nescius* is 16pt. As I use the default font which is 12 pt Helvetica. The paragraph looks fine.

For the paragraph that starts with *Contra quos omnis*, I use a line-height of 25pt and you see that there are big gaps between the lines.

For the paragraph that starts with *Sive enim ad*, I use a line-height of 13pt which is only 1 pt more than the font height. The lines are very close together for this paragraph.

<sup>455</sup><http://stackoverflow.com/questions/29290668/set-line-spacing-when-using-xmlworker-to-parse-html-to-pdf-itextsharp-c-sharp>

<sup>456</sup><http://stackoverflow.com/users/4346553/jon-z>

It doesn't matter where you define the `line-height`. Your options are to define it inline in the tag, in the `<head>` section of your HTML or in an external CSS file that is either referenced from the header of your HTML or loaded into XML Worker separately. Whatever you like most is OK.



Thanks Bruno, I ended up using an external CSS file that sets proper `line-height`, and loading that into XML Worker. Initially I had some trouble with consecutive paragraph tags all rendering on top of each other, but I think there was something improper with my HTML string, and not an issue with `ITextSharp`.

## How to add external CSS while generating PDF?



Currently i am using following code to generate PDF in a JSP file:

```
response.setContentType("application/force-download");
response.setHeader("Content-Disposition", "attachment;filename=reports.pdf");
Document document = new Document();
document.setPageSize.PageSize.A1;
PdfWriter writer = null;
writer = PdfWriter.getInstance(document, response.getOutputStream());
document.open();
ByteArrayInputStream bis
= new ByteArrayInputStream(htmlSource.toString().getBytes());
XMLWorkerHelper.getInstance().parseXHtml(writer, document, bis);
document.close();
```

With this code, I'm able to generate PDF, but I would like to add a CSS file while generating PDF.

Posted on StackOverflow on Jul 16, 2014 <sup>457</sup> by Yella Goud<sup>458</sup>

Please take a look at the [ParseHtmlTable1<sup>459</sup>](#) example. In this example, we have HTML stored in a `StringBuilder` object and some CSS stored in a `String`. In my example, I convert the `sb` object and the `CSS` object to an `InputStream`. If you have files with the HTML and the CSS, you could easily use a `FileInputStream`.

Once you have an `InputStream` for the HTML and the CSS, you can use this code:

<sup>457</sup><http://stackoverflow.com/questions/24777549/how-to-add-external-css-while-generating-pdf>

<sup>458</sup><http://stackoverflow.com/users/2436481/yella-goud>

<sup>459</sup><http://itextpdf.com/sandbox/xmlworker/ParseHtmlTable1>

```

// CSS
CSSResolver cssResolver = new StyleAttrCSSResolver();
CssFile cssFile = XMLWorkerHelper.getCSS(new ByteArrayInputStream(CSS.getBytes())\n));
cssResolver.addCss(cssFile);
// HTML
HtmlPipelineContext htmlContext = new HtmlPipelineContext(null);
htmlContext.setTagFactory(Tags.getHtmlTagProcessorFactory());
// Pipelines
PdfWriterPipeline pdf = new PdfWriterPipeline(document, writer);
HtmlPipeline html = new HtmlPipeline(htmlContext, pdf);
CssResolverPipeline css = new CssResolverPipeline(cssResolver, html);
// XML Worker
XMLWorker worker = new XMLWorker(css, true);
XMLParser p = new XMLParser(worker);
p.parse(new ByteArrayInputStream(sb.toString().getBytes()));

```

Or, if you don't like all that code:

```

ByteArrayInputStream bis =
    new ByteArrayInputStream(htmlSource.toString().getBytes());
ByteArrayInputStream cis =
    new ByteArrayInputStream(cssSource.toString().getBytes());
XMLWorkerHelper.getInstance().parseXHtml(writer, document, bis, cis);

```

## How to do HTML to XML conversion to generate closed tags?



When I try converting html to pdf using iText and XML Worker, I'm asked to give the closing tag for <hr> and <br> tags. It works if I do this manually, but I don't want to add each closing tag manually. How can I do this in an automated way?

Posted on StackOverflow on Oct 30, 2014 <sup>460</sup> by Kannu Verma<sup>461</sup>

You are experiencing this problem because you are feeding HTML to iText's XML Worker. XML Worker requires XML, so you need to convert your HTML into XHTML.

There is an example on how to do this on the official iText site: [D00\\_XHTML<sup>462</sup>](#)

---

<sup>460</sup><http://stackoverflow.com/questions/26652029/how-to-do-xml-to-html-conversion-to-generate-closed-tags>

<sup>461</sup><http://stackoverflow.com/users/4197576/kannu-verma>

<sup>462</sup>[http://itextpdf.com/sandbox/xmlworker/D00\\_XHTML](http://itextpdf.com/sandbox/xmlworker/D00_XHTML)

```

public static void tidyUp(String path) throws IOException {
    File html = new File(path);
    byte[] xhtml = Jsoup.parse(html, "US-ASCII").html().getBytes();
    File dir = new File("results/xml");
    dir.mkdirs();
    FileOutputStream fos = new FileOutputStream(new File(dir, html.getName()));
    fos.write(xhtml);
    fos.close();
}

```

In this example, we get a path to an ordinary HTML file (similar to what you have). We then use the [Jsoup<sup>463</sup>](#) library to parse the HTML into an XHTML byte array. In this example, we use that byte array to write an XHTML file to disk. You can use the byte array directly as input for XML Worker.

## How to make a particular sub-string Bold when converting HTML to PDF?



I am using iText in Java to convert a HTML to PDF. I want a particular paragraph which has some words as Bold and some as Bold+Underlined to be passed as a string to the Java code and to be converted to PDF using the iText library. I am unable to find a suitable method for this. How should I do this?

Posted on StackOverflow on Jun 4, 2014 <sup>464</sup> by [Anuranjan](#)<sup>465</sup>

If you want to convert XHTML to PDF, you need iText + XML Worker.

The most simple examples looks like this:

```

public void createPdf(String file) throws IOException, DocumentException {
    // step 1
    Document document = new Document();
    // step 2
    PdfWriter writer =
        PdfWriter.getInstance(document, new FileOutputStream(file));
    // step 3
    document.open();
    // step 4

```

---

<sup>463</sup><http://jsoup.org/>

<sup>464</sup><http://stackoverflow.com/questions/24036662/how-to-make-a-particular-sub-string-bold-while-printing-a-string-in-pdf-using-it>

<sup>465</sup><http://stackoverflow.com/users/3706786/anuranjan>

```

XMLWorkerHelper.getInstance().parseXHtml(writer, document,
    new FileInputStream(HTML));
// step 5
document.close();
}

```

Note that the HTML file is passed as a `FileInputStream` in this case. You want to pass a `String`. This means you'll have to do something like this:

```

XMLWorkerHelper.getInstance().parseXHtml(writer, document,
    new StringReader("<p>The <b>String</b> I want to render to PDF</p>"));

```

There are [more complex examples in the Sandbox<sup>466</sup>](#) in case you need support for images, special fonts, and so on.

## How to parse multiple HTML files into a single PDF



I want to use iText to convert a series of html file to PDF. For instance: I have these files: page1.html, page2.html, page3.html,... Now I want to create a single PDF file, where page1.html is the first page, page2.html is the second page, and so on... I know how to convert a single HTML file to a PDF, but I don't know how to combine these different PDFs resulting from this operation into a single PDF.

Posted on StackOverflow on January 6, 2015 <sup>467</sup> by [kyzh101](#)<sup>468</sup>

There are two answers and answer #2 is generally better than answer #1, but I'm giving both options because there may be specific cases where answer #1 is better.

**Test data:** I have created 3 simple HTML files, each containing some info about a State in the US:

- [page1.html<sup>469</sup>](#): California
- [page2.html<sup>470</sup>](#): New York
- [page3.html<sup>471</sup>](#): Massachusetts

---

<sup>466</sup><http://itextpdf.com/sandbox/xmlworker>

<sup>467</sup><http://stackoverflow.com/questions/27814701/itextsharp-how-to-use-htmlworker-to-covert-html-to-pdf-with-pagination>

<sup>468</sup><http://stackoverflow.com/users/4319442/kyzh101>

<sup>469</sup><http://itextpdf.com/sites/default/files/page1.html>

<sup>470</sup><http://itextpdf.com/sites/default/files/page2.html>

<sup>471</sup><http://itextpdf.com/sites/default/files/page3.html>

We are going to use XML Worker to parse these three files and we want a single PDF file as a result.

**Answer #1:** see [ParseMultipleHtmlFiles1<sup>472</sup>](#) for the full code sample and [multiple\\_html\\_pages1.pdf<sup>473</sup>](#) for the resulting PDF.

You say that you already succeeded in converting one HTML file into one PDF files. It is assumed that you did it like this:

```
public byte[] parseHtml(String html) throws DocumentException, IOException {
    ByteArrayOutputStream baos = new ByteArrayOutputStream();
    // step 1
    Document document = new Document();
    // step 2
    PdfWriter writer = PdfWriter.getInstance(document, baos);
    // step 3
    document.open();
    // step 4
    XMLWorkerHelper.getInstance().parseXHtml(writer, document,
        new FileInputStream(html));
    // step 5
    document.close();
    // return the bytes of the PDF
    return baos.toByteArray();
}
```

This is not the most efficient way to parse an HTML file (there are other examples on the web site), but it's the simplest way.

As you can see, this method parse an HTML into a PDF file and returns that PDF file in the form of a byte[]. As we want to create a single PDF, we can feed this byte array to a PdfCopy instance, so that we can concatenate multiple documents.

Suppose that we have three documents:

```
public static final String[] HTML = {
    "resources/xml/page1.html",
    "resources/xml/page2.html",
    "resources/xml/page3.html"
};
```

We can loop over these three documents, parse them one by one to a byte[], create a PdfReader instance with the PDF bytes, and add the document to the PdfCopy instance using the addDocument() method:

---

<sup>472</sup><http://itextpdf.com/sandbox/xmlworker/ParseMultipleHtmlFiles1>

<sup>473</sup>[http://itextpdf.com/sites/default/files/multiple\\_html\\_pages1.pdf](http://itextpdf.com/sites/default/files/multiple_html_pages1.pdf)

```

public void createPdf(String file) throws IOException, DocumentException {
    Document document = new Document();
    PdfCopy copy = new PdfCopy(document, new FileOutputStream(file));
    document.open();
    PdfReader reader;
    for (String html : HTML) {
        reader = new PdfReader(parseHtml(html));
        copy.addDocument(reader);
        reader.close();
    }
    document.close();
}

```

This solves your problem, but suppose that you need to use a special font that needs to be embedded. In that case, every separate PDF file will contain a subset of that font. Different files will require different font subsets, and `PdfCopy` (nor `PdfSmartCopy` for that matter) can merge font subsets. This could result in a bloated PDF file with way too many font subsets of the same font. How can we avoid this? That's explained in answer #2.

**Answer #2:** See [ParseMultipleHtmlFiles2<sup>474</sup>](#) for the full code sample and [multiple\\_html\\_pages2.pdf<sup>475</sup>](#) for the resulting PDF. You already see the difference in file size: 4.61 KB versus 5.05 KB (and we didn't even introduce embedded fonts).

In this case, we don't parse the HTML to a PDF file the way we did in the `parseHtml()` method from answer #1. Instead, we parse the HTML to an `iText ElementList` using the `parseToElementList()` method. This method requires two `Strings`. One containing the HTML code, the other one containing CSS values. We use a utility method to read the HTML file into a `String`. As for the CSS value, we could pass `null` to `parseToElementList()`, but in that case, default styles will be ignored. You'll notice that the `<h1>` tag we introduced in our HTML will look completely different if you don't pass the `default.css` that is shipped with XML Worker. This is the code:

```

public void createPdf(String file) throws IOException, DocumentException {
    Document document = new Document();
    PdfWriter.getInstance(document, new FileOutputStream(file));
    document.open();
    String css = readCSS();
    for (String htmlfile : HTML) {
        String html = Utilities.readFileToString(htmlfile);
        ElementList list = XMLWorkerHelper.parseToElementList(html, css);
        for (Element e : list) {
            document.add(e);
        }
    }
}

```

---

<sup>474</sup><http://itextpdf.com/sandbox/xmlworker/ParseMultipleHtmlFiles2>

<sup>475</sup>[http://itextpdf.com/sites/default/files/multiple\\_html\\_pages2.pdf](http://itextpdf.com/sites/default/files/multiple_html_pages2.pdf)

```

        }
        document.newPage();
    }
    document.close();
}

```

We create a single Document and a single PdfWriter instance. We parse the different HTML files into ElementLists one by one, and we add all the elements to the Document.

As you want a new page, each time a new HTML file is parsed, I introduced a `document.newPage()`. If you remove this line, you can add the three HTML pages on a single page (which wouldn't be possible if you would opt for answer #1).

## How to add a rich Textbox (HTML) to a table cell?



I have a rich text box with the following content:

```

<p>Overview&#160;line1</p>
<p>Overview&#160;line2</p>
<p>Overview&#160;line3</p>
<p>Overview&#160;line4</p>
<p>Overview&#160;line4</p>
<p>Overview&#160;line5</p>

```

When I add the HTML content of this rich text box to a PdfPCell, I expect to see:

```

Overview line1
Overview line2
Overview line3
Overview line4
Overview line5

```

The problem is that the text appears as HTML instead of being rendered.

Posted on StackOverflow on Nov 19, 2014 <sup>476</sup> by Kate<sup>477</sup>

Please take a look at the [HtmlContentForCell<sup>478</sup>](#) example.

In this example, we have the HTML you mention:

---

<sup>476</sup><http://stackoverflow.com/questions/27015644/add-a-rich-textbox-to-pdf-using-itextsharp>

<sup>477</sup><http://stackoverflow.com/users/4222385/kate>

<sup>478</sup><http://itextpdf.com/sandbox/htmlworker/HtmlContentForCell>

```
public static final String HTML = "<p>Overview&#160;line1</p>"  
    + "<p>Overview&#160;line2</p><p>Overview&#160;line3</p>"  
    + "<p>Overview&#160;line4</p><p>Overview&#160;line4</p>"  
    + "<p>Overview&#160;line5&#160;.</p>";
```

We also create a font for the `<p>` tag:

```
public static final String CSS = "p { font-family: Cardo; }";
```

In your case, you may want to replace Cardo with Arial.

Note that we registered the regular version of the Cardo font:

```
FontFactory.register("resources/fonts/Cardo-Regular.ttf");
```

If you need bold, italic and bold-italic, you also need to register those fonts of the same Cardo family.  
(In case of arial, you'd register arial.ttf, arialbd.ttf, ariali.ttf and arialbi.ttf).

Now we can parse this HTML and CSS into a list of `Element` objects with the `parseToElementList()` method. We can use these objects inside a cell:

```
PdfPTable table = new PdfPTable(2);  
table.addCell("Some rich text:");  
PdfPCell cell = new PdfPCell();  
for (Element e : XMLWorkerHelper.parseToElementList(HTML, CSS)) {  
    cell.addElement(e);  
}  
table.addCell(cell);  
document.add(table);
```

See `html_in_cell.pdf`<sup>479</sup> for the resulting PDF.

---

<sup>479</sup>[http://itextpdf.com/sites/default/files/html\\_in\\_cell.pdf](http://itextpdf.com/sites/default/files/html_in_cell.pdf)

## How to get particular html table contents to write it in PDF?



I have an HTML file that looks like this when I render it in a browser:

Sr. No.	Text Data	Number Data
1	This is text data 0	0
2	This is text data 1	1
3	This is text data 2	4
4	This is text data 3	9
5	This is text data 4	16
6	This is text data 5	25
7	This is text data 6	36
8	This is text data 7	49
9	This is text data 8	64
10	This is text data 9	81

some text

[Export to Excel](#)

HTML table

When I export the String with this HTML to PDF using a Paragraph object, I get the following output:

```
<TABLE id=exportTableSelector border=2 align=left>
<THEAD>
<TR bgColor=lightgreen>
<TH>Sr. No.</TH>
<TH>Text Data</TH>
<TH>Number Data</TH></TR></THEAD>
<TBODY>
<TR bgColor=lightblue>
<TD align=middle>1</TD>
<TD align=middle>This is text data 0</TD>
<TD align=middle>0</TD></TR>
<TR bgColor=lightblue>
```

HTML output

This is not what I want. I want the PDF to look like the HTML in the browser.

Posted on StackOverflow on Jul 2, 2014 <sup>480</sup> by user3152748<sup>481</sup>

Please take a look at the examples [ParseHtmlTable1](#)<sup>482</sup> and [ParseHtmlTable2](#)<sup>483</sup>. They create the

---

<sup>480</sup><http://stackoverflow.com/questions/24530852/how-to-get-particular-html-table-contents-to-write-it-in-pdf-using-itext>

<sup>481</sup><http://stackoverflow.com/users/3152748>

<sup>482</sup><http://itextpdf.com/sandbox/xmlworker/ParseHtmlTable1>

<sup>483</sup><http://itextpdf.com/sandbox/xmlworker/ParseHtmlTable2>

following PDFs: [html\\_table\\_1.pdf<sup>484</sup>](#) and [html\\_table\\_2.pdf<sup>485</sup>](#).

The table is created like this:

```
StringBuilder sb = new StringBuilder();
sb.append("<table border=\"2\">");
sb.append("<tr>");
sb.append("<th>Sr. No.</th>");
sb.append("<th>Text Data</th>");
sb.append("<th>Number Data</th>");
sb.append("</tr>");
for (int i = 0; i < 10; ) {
    i++;
    sb.append("<tr>");
    sb.append("<td>");
    sb.append(i);
    sb.append("</td>");
    sb.append("<td>This is text data ");
    sb.append(i);
    sb.append("</td>");
    sb.append("<td>");
    sb.append(i);
    sb.append("</td>");
    sb.append("</tr>");
}
sb.append("</table>");
```

I've taken the liberty to define the CSS like this:

```
tr { text-align: center; }
th { background-color: lightgreen; padding: 3px; }
td {background-color: lightblue; padding: 3px; }
```

Now we parse the CSS and the HTML like this:

---

<sup>484</sup>[http://itextpdf.com/sites/default/files/html\\_table\\_1.pdf](http://itextpdf.com/sites/default/files/html_table_1.pdf)

<sup>485</sup>[http://itextpdf.com/sites/default/files/html\\_table\\_2.pdf](http://itextpdf.com/sites/default/files/html_table_2.pdf)

```

CSSResolver cssResolver = new StyleAttrCSSResolver();
CssFile cssFile = XMLWorkerHelper.getCSS(new ByteArrayInputStream(CSS.getBytes())\
));
cssResolver.addCss(cssFile);
// HTML
HtmlPipelineContext htmlContext = new HtmlPipelineContext(null);
htmlContext.setTagFactory(Tags.getHtmlTagProcessorFactory());
// Pipelines
ElementList elements = new ElementList();
ElementHandlerPipeline pdf = new ElementHandlerPipeline(elements, null);
HtmlPipeline html = new HtmlPipeline(htmlContext, pdf);
CssResolverPipeline css = new CssResolverPipeline(cssResolver, html);
// XML Worker
XMLWorker worker = new XMLWorker(css, true);
XMLParser p = new XMLParser(worker);
p.parse(new ByteArrayInputStream(sb.toString().getBytes()));

```

Now the `elements` list contains a single element: your table:

```
return (PdfPTable)elements.get(0);
```

You can add this table to your PDF document. This is what the result looks like:



This is my HTML table:

Sr. No.	Text Data	Number Data
1	This is text data 1	1
2	This is text data 2	2
3	This is text data 3	3
4	This is text data 4	4
5	This is text data 5	5
6	This is text data 6	6
7	This is text data 7	7
8	This is text data 8	8
9	This is text data 9	9
10	This is text data 10	10

Table presented in PDF

## Why doesn't CSS and RowSpan work?



I am trying to convert Html to PDF. I found that iTextSharp does not support CSS well. In fact, I think `HtmlWorker` does not support it all, nor does it seem to support `RowSpan` either.

INVOICE				
AIRNET NETWORKS asdadadadaada asd asd a ads adsadsadsadasd		INVOICE	DATE	aDATEsd
Order	XXXX	Ref XXXXXX	Delivery	XXXX
Ref XXXXXX	Due Date	XXXX	Ref XXXXXX	CUSTOMER NAME asd asd adadaadadada adadaadsasdad ada asd adad
Customer Care No:	544646454,88877978975		Email Id	airnet@gmail.com

Smo	Particulars	Quantity	Rate/Month	Total Rupees
1	1 MBPS Plan	1	600	692
1	1 MBPS Plan	1	600	692
asdadasd		Total		692

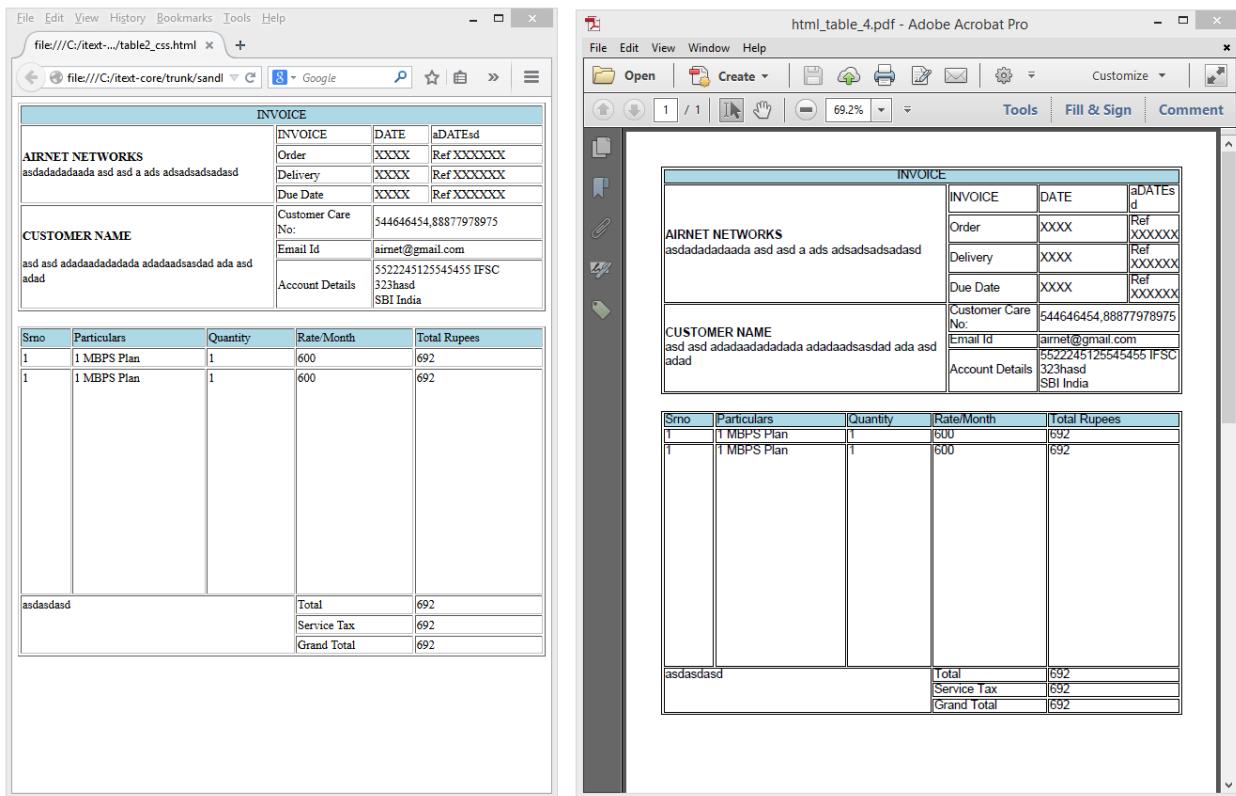
Unwanted result

Posted on StackOverflow on Oct 31, 2014 <sup>486</sup> by arbaaz<sup>487</sup>

Please take a look at the following screen shot:

<sup>486</sup><http://stackoverflow.com/questions/26679003/rowspan-does-not-work-in-itextsharp>

<sup>487</sup><http://stackoverflow.com/users/2064292/arbaaz>



#### HTML to PDF conversion

To the left, you see an HTML file rendered in a browser. To the right, you see that HTML file rendered to PDF using iText (the Java version).

There are three reasons why your application doesn't work:

1. Your HTML doesn't make sense. I had to clean it up (change `<br>` into `<br />`, introduce the correct CSS, correct the column-count for some rows,...) and make it XHTML before it rendered correctly in a browser. You can find the HTML that was used in the screen shot here: [table2\\_css.html<sup>488</sup>](#)
2. You are using `HTMLWorker` instead of `XML Worker`, and you are right: `HTMLWorker` has no support for CSS. Saying CSS doesn't work in iTextSharp is wrong. It doesn't work when you use `HTMLWorker`, but that's documented: the CSS you need works in `XML Worker`.
3. You are probably using an old version of iTextSharp, and you are right: CSS and table support wasn't as good as in older versions of iTextSharp when compared to the most recent version.

Apart from iTextSharp, you also need to download [XML Worker<sup>489</sup>](#). Most of the examples on the web site are written in Java, but you should have no problem converting them to C#. The example I

<sup>488</sup>[http://itextpdf.com/sites/default/files/table2\\_css.html](http://itextpdf.com/sites/default/files/table2_css.html)

<sup>489</sup>[http://itextpdf.com/product/xml\\_worker](http://itextpdf.com/product/xml_worker)

used to make the PDF in the screen shot ([html\\_table\\_4.pdf<sup>490</sup>](#)) can be found here: [ParseHtmlTable4<sup>491</sup>](#)

## Why is XMLWorker parsing slow?



I am parsing HTML string using iTextSharp XMLWorker in my WPF application using the below code:

```
var css = "";
using (var htmlMS = new MemoryStream(System.Text.Encoding.UTF8.GetBytes(html)))
{
    //Create a stream to read our CSS
    using (var cssMS = new MemoryStream(System.Text.Encoding.UTF8.GetBytes(css)))
    {
        //Get an instance of the generic XMLWorker
        var xmlWorker = XMLWorkerHelper.GetInstance();
        //Parse our HTML using everything setup above
        xmlWorker.ParseXHtml(writer, doc, htmlMS, cssMS,
            System.Text.Encoding.UTF8, fontProv);
    }
}
```

The parsing works fine but it is really slow, it takes around 2 seconds to parse the HTML. So for a 50 page PDF it takes around 2 minutes. I am using inline styling to in my HTML string. Is this the natural behaviour or it can be optimized?

Posted on StackOverflow on Jan 22, 2014 <sup>492</sup> by user2877090<sup>493</sup>

The question suggests that the HTML parsing is slowing everything down. I'm convinced that the bottleneck occurs even before the first snippet of HTML is parsed.

You are using the most basic handful of lines of code to create your PDF from HTML as demonstrated in the [ParseHtml<sup>494</sup>](#) example:

---

<sup>490</sup>[http://itextpdf.com/sites/default/files/html\\_table\\_4.pdf](http://itextpdf.com/sites/default/files/html_table_4.pdf)

<sup>491</sup><http://itextpdf.com/sandbox/xmlworker/ParseHtmlTable4>

<sup>492</sup><http://stackoverflow.com/questions/21275800/itextsharp-xmlworker-parsing-really-slow>

<sup>493</sup><http://stackoverflow.com/users/2877090/user2877090>

<sup>494</sup>[http://itextpdf.com/sandbox/xmlworker/D02\\_ParseHtml](http://itextpdf.com/sandbox/xmlworker/D02_ParseHtml)

```

public void createPdf(String file) throws IOException, DocumentException {
    // step 1
    Document document = new Document();
    // step 2
    PdfWriter writer = PdfWriter.getInstance(document,
        new FileOutputStream(file));
    // step 3
    document.open();
    // step 4
    XMLWorkerHelper.getInstance().parseXHtml(writer, document,
        new FileInputStream(HTML));
    // step 5
    document.close();
}

```

This code is simple, but it performs a lot of operations internally such as registering font directories. This consumes plenty of time. You can avoid this, by using your own `FontProvider` as is done in the `ParseHtmlFonts495` example.

```

public void createPdf(String file) throws IOException, DocumentException {
    // step 1
    Document document = new Document();
    // step 2
    PdfWriter writer = PdfWriter.getInstance(document,
        new FileOutputStream(file));
    writer.setInitialLeading(12.5f);
    // step 3
    document.open();
    // step 4
    // CSS
    CSSResolver cssResolver = new StyleAttrCSSResolver();
    CssFile cssFile = XMLWorkerHelper.getCSS(new FileInputStream(CSS));
    cssResolver.addCss(cssFile);
    // HTML
    XMLWorkerFontProvider fontProvider =
        new XMLWorkerFontProvider(XMLWorkerFontProvider.DONTLOOKFORFONTS);
    fontProvider.register("resources/fonts/Cardo-Regular.ttf");
    fontProvider.register("resources/fonts/Cardo-Bold.ttf");
    fontProvider.register("resources/fonts/Cardo-Italic.ttf");
    fontProvider.addFontSubstitute("lowagie", "cardo");
    CssAppliers cssAppliers = new CssAppliersImpl(fontProvider);

```

---

<sup>495</sup>[http://itextpdf.com/sandbox/xmlworker/D06\\_ParseHtmlFonts](http://itextpdf.com/sandbox/xmlworker/D06_ParseHtmlFonts)

```

HtmlPipelineContext htmlContext = new HtmlPipelineContext(cssAppliers);
htmlContext.setTagFactory(Tags.getHtmlTagProcessorFactory());
// Pipelines
PdfWriterPipeline pdf = new PdfWriterPipeline(document, writer);
HtmlPipeline html = new HtmlPipeline(htmlContext, pdf);
CssResolverPipeline css = new CssResolverPipeline(cssResolver, html);
// XML Worker
XMLWorker worker = new XMLWorker(css, true);
XMLParser p = new XMLParser(worker);
p.parse(new FileInputStream(HTML));
// step 5
document.close();
}

```

In this case, we instruct iText DONTLOOKFORFONTS, thus saving an enormous amount of time. Instead of having iText looking for fonts, we tell iText which fonts we're going to use in the HTML.

## How to export Vietnamese text to PDF using iText?



I'm facing a problem when trying to export a Vietnamese document as PDF using iText. I put Vietnamese words in .xml file like this

```

<td fontfamily="Helvetica" fontstyle="0" fontsize="9" align="0" colspan="48" lin\
eoccupied="1">
    T\u1ED5 ch\u1EE9c tham gia
</td>

```

I convert this into Unicode, exporting the String to PDF using encoding UTF-8, but the program fails to display the Vietnamese characters '\u1ED5' and '\u1EE9' and the output becomes "T chc tham gia".

Posted on StackOverflow on Feb 28, 2014 <sup>496</sup> by NTLC<sup>497</sup>

There are 3 [XML Worker examples<sup>498</sup>](#) involving Asian languages on the official iText web site. They parse an XHTML file containing Chinese characters, but it should be easy to adapt them to Vietnamese examples.

You can find the HTML files were going to parse here:

---

<sup>496</sup><http://stackoverflow.com/questions/22085316/how-to-export-vietnamese-text-to-pdf-using-itext>

<sup>497</sup><http://stackoverflow.com/users/1040917/ntlc>

<sup>498</sup><http://itextpdf.com/sandbox/xmlworker/>

- <http://itextpdf.com/sites/default/files/hero.html>
- <http://itextpdf.com/sites/default/files/hero2.html>

Both files contain the following text:

𠂇 (Broken Sword), 𠂇𠂇 (Flying Snow), 𠂇 (Moon), 𠂇 (the King), and 𠂇 (Sky).

In the first case, a font is defined using CSS:

```
<span style="font-size:12.0pt; font-family:MS Mincho">𠂇</span>
```

In the second case, no specific font is defined:

```
<body><p>𠂇 (Broken Sword), 𠂇𠂇 (Flying Snow), 𠂇 (Moon), 𠂇 (the King), and 𠂇\\(Sky). </p></body>
```

These files contain UTF-8 characters, so we're going to parse them like this:

```
XMLWorkerHelper.getInstance().parseXHtml(writer, document,  
    new FileInputStream(HTML), Charset.forName("UTF-8"));
```

The first thing you need, is a font that supports Vietnamese characters. That's something iText can't help you with. In your HTML file, you've defined Helvetica, but that's a standard Type1 font that is never embedded when using iText and that doesn't know how to draw Vietnamese glyphs. That's never going to work.

The first example [D07\\_ParseHtmlAsian<sup>499</sup>](#) will automatically search for a font named MS Mincho. If it finds that font (for instance because you have msmincho.ttc in your Windows fonts directory), the font will show up in your PDF. See [hero.pdf<sup>500</sup>](#). If it doesn't find a font with that name, then the glyphs won't be visible, because you didn't provide any font program for those glyphs.

The second example [D07bis\\_ParseHtmlAsian<sup>501</sup>](#) offers a workaround in case you don't have MS Mincho anywhere. In that case, you have to use an `XMLWorkerFontProvider` and register a font that can be used instead of MS Mincho. For instance: we use a font stored in the file cfmingeb.ttf and assign the alias MS Mincho:

---

<sup>499</sup>[http://itextpdf.com/sandbox/xmlworker/D07\\_ParseHtmlAsian](http://itextpdf.com/sandbox/xmlworker/D07_ParseHtmlAsian)

<sup>500</sup><http://itextpdf.com/sites/default/files/hero.pdf>

<sup>501</sup>[http://itextpdf.com/sandbox/xmlworker/D07bis\\_ParseHtmlAsian](http://itextpdf.com/sandbox/xmlworker/D07bis_ParseHtmlAsian)

```
XMLWorkerFontProvider fontProvider = new XMLWorkerFontProvider(XMLWorkerFontProvider.DONTLOOKFORFONTS);
fontProvider.register("resources/fonts/cfmingeb.ttf", "MS Mincho");
```

The resulting file `asian.pdf`<sup>502</sup> is slightly different from what we expect, but now we can at least see the Chinese glyphs.

In the third example, the HTML file doesn't tell us anything about the font that needs to be used. We'll define the font using CSS like this:

```
CSSResolver cssResolver = new StyleAttrCSSResolver();
CssFile cssFile = XMLWorkerHelper.getCSS(
    new ByteArrayInputStream("body {font-family:tsc fm ing s tt}").getBytes());
cssResolver.addCss(cssFile);
```

Now, all the text in the body will use the font TSC FMing S TT (stored in the file `cfmingeb.ttf`). You can see the difference in the resulting PDF `asian2.pdf`<sup>503</sup>.

## Certain HTML entities (arrows) are not rendered in PDF



I tried to render &rArr; &hArr; &euro; &copy; &rarr; from HTML to PDF with the XMLWorker and iText. Only the copyright and euro symbol appear in the PDF. I used the default font and Arial but without success. Is there a way to get those entities rendered? Is there an alternative way to render arrows in text?

Posted on StackOverflow on Mar 18, 2015 <sup>504</sup> by Code Clown<sup>505</sup>

As you can see in the `ParseHtml3`<sup>506</sup> example, it works for me:

---

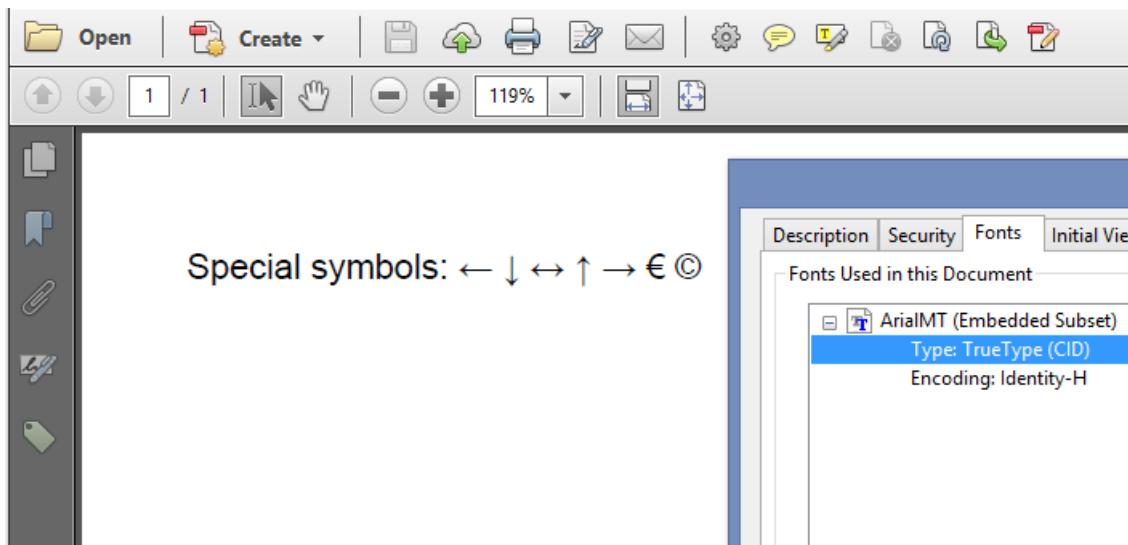
<sup>502</sup><http://itextpdf.com/sites/default/files/asian.pdf>

<sup>503</sup><http://itextpdf.com/sites/default/files/asian2.pdf>

<sup>504</sup><http://stackoverflow.com/questions/29121766/certain-html-entities-arrows-are-not-rendered-in-pdf-itext>

<sup>505</sup><http://stackoverflow.com/users/378386/code-clown>

<sup>506</sup><http://itextpdf.com/sandbox/xmlworker/ParseHtml3>



Screen shot

This is my code to create the PDF:

```
public void createPdf(String file) throws IOException, DocumentException {
    // step 1
    Document document = new Document();
    // step 2
    PdfWriter writer =
        PdfWriter.getInstance(document, new FileOutputStream(file));
    // step 3
    document.open();
    // step 4
    String str = "<html><head></head>" +
        "<body style=\"font-size:12.0pt; font-family:Arial\">" +
        "<p>Special symbols: " +
        "&larr; &darr; &harr; &uarr; &rarr; &euro; &copy;</p>" +
        "</body></html>";

    XMLWorkerHelper worker = XMLWorkerHelper.getInstance();
    InputStream is = new ByteArrayInputStream(str.getBytes());
    worker.parseXHtml(writer, document, is);
    // step 5
    document.close();
}
```

Note that all entities are written in lower-case, whereas you have &rArr; (which doesn't work for me either).

## How to set RTL direction for Hebrew when converting HTML to PDF?



I'm trying to convert an HTML file with Hebrew characters (UTF-8) to PDF by using iText, but I'm getting all letters in reverse order. As far I understand, I can set RTL only for ColumnText and PdfCell objects. So here's my doubt: is it possible to convert Hebrew HTML to PDF? This is my HTML:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
 <title>Title of document</title>
</head>
<body style="font-size:12.0pt; font-family:Arial">
    סלול מלוּע
</body>
</html>
```

When I convert this HTML to PDF using XML Worker, I get this result:

Wrong order

These is “Hello World” in Hebrew written from left to right. It should be written from right to left.

Posted on StackOverflow on Jun 15, 2015 <sup>507</sup> by Anatoly<sup>508</sup>

Please take a look at the [ParseHtml10<sup>509</sup>](#) example. In this example, we have take the file [hebrew.html<sup>510</sup>](#):

<sup>507</sup><http://stackoverflow.com/questions/30847907/cant-set-rtl-direction-for-hebrew-letters-while-converting-from-xhtml-to-pdf>

<sup>508</sup><http://stackoverflow.com/users/947111/anatoly>

<sup>509</sup><http://itextpdf.com/sandbox/xmlworker/ParseHtml10>

<sup>510</sup><http://itextpdf.com/sites/default/files/hebrew.html>

```
<html>

<head>
    <title>Hebrew text</title>
</head>

<body style="font-size:12.0pt; font-family:Arial">
<div dir="rtl" style="font-family: Noto Sans Hebrew">הָרָקָעַ</div>
</body>

</html>
```

And we convert it to PDF using this code:

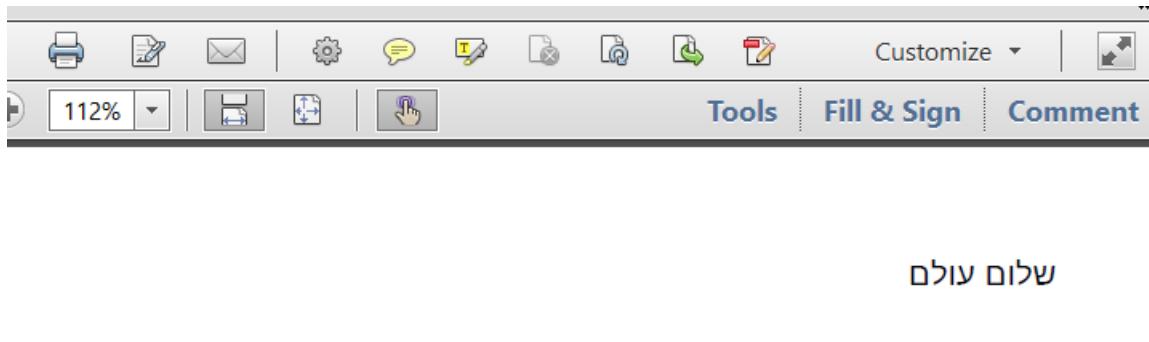
```
public void createPdf(String file) throws IOException, DocumentException {
    // step 1
    Document document = new Document();
    // step 2
    PdfWriter writer =
        PdfWriter.getInstance(document, new FileOutputStream(file));
    // step 3
    document.open();
    // step 4
    // Styles
    CSSResolver cssResolver = new StyleAttrCSSResolver();
    XMLWorkerFontProvider fontProvider =
        new XMLWorkerFontProvider(XMLWorkerFontProvider.DONTLOOKFORFONTS);
    fontProvider.register("resources/fonts/NotoSansHebrew-Regular.ttf");
    CssAppliers cssAppliers = new CssAppliersImpl(fontProvider);
    HtmlPipelineContext htmlContext = new HtmlPipelineContext(cssAppliers);
    htmlContext.setTagFactory(Tags.getHtmlTagProcessorFactory());

    // Pipelines
    PdfWriterPipeline pdf = new PdfWriterPipeline(document, writer);
    HtmlPipeline html = new HtmlPipeline(htmlContext, pdf);
    CssResolverPipeline css = new CssResolverPipeline(cssResolver, html);

    // XML Worker
    XMLWorker worker = new XMLWorker(css, true);
    XMLParser p = new XMLParser(worker);
    p.parse(new FileInputStream(HTML), Charset.forName("UTF-8"));
    // step 5
```

```
document.close();  
}
```

The result looks like [hebrew.pdf](#)<sup>511</sup>:



Text from right to left

What are the hurdles you need to take?

- You need to wrap your text in an element such as a `<div>` or a `<td>`.
- You need to add the attribute `dir="rtl"` to define the direction.
- You need to make sure that you're using a font that knows how to display Hebrew. I used a NOTO font for Hebrew. This is one of the fonts distributed by Google in their program to provide fonts for every possible language.

**Important:** this solution requires at least iText and XML Worker 5.5.5, because support for the `dir` attribute was introduced in 5.5.4 and improved in 5.5.5.

<sup>511</sup><http://itextpdf.com/sites/default/files/hebrew.pdf>

## How to convert Arabic HTML to PDF?



I am having difficulties displaying Arabic Characters from HTML content in PDF. The characters are generated as “?” characters. I am able to display Arabic text from a String value, but not from HTML. I want to display the PDF with two column, left side English and the right side Arabic Text.

```
BaseFont bf = BaseFont.createFont(  
    "C:\\arial.ttf", BaseFont.IDENTITY_H, BaseFont.EMBEDDED);  
Font font = new Font(bf, 8);  
PdfPCell contentArCell = new PdfPCell();  
contentArCell.setRunDirection(PdfWriter.RUN_DIRECTION_RTL);  
String htmlContentAr =  
    "<p><span> ﻢﻟآ ﻢﻟآ ﻢﻟآ ﻢﻟآ</p><p/> ﻢﻟآ ﻢﻟآ ﻢﻟآ ...</span></p>";  
for (Element e : XMLWorkerHelper.parseToElementList(htmlContentAr, null)) {  
    for (Chunk c:e.getChunks()) {  
        c.setFont(new Font(bf));  
    }  
    contentArCell.addElement(e);  
}
```

Posted on StackOverflow on May 13, 2015 <sup>512</sup> by user2579223<sup>513</sup>

Please take a look at the [ParseHtml7<sup>514</sup>](#) and [ParseHtml8<sup>515</sup>](#) examples. They take HTML input with Arabic characters and they create a PDF with the same Arabic text:

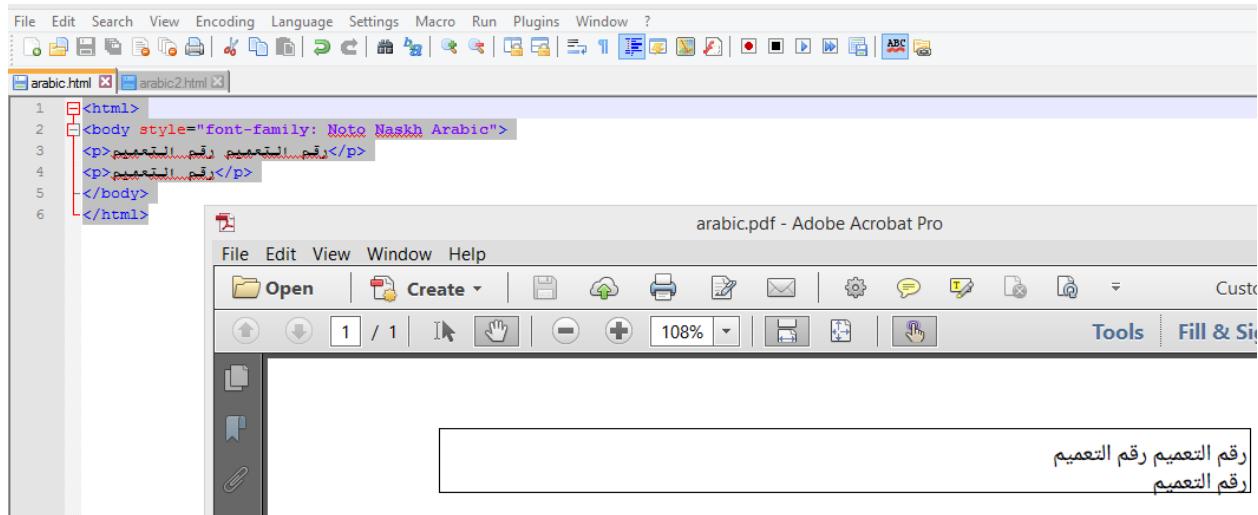
---

<sup>512</sup><http://stackoverflow.com/questions/30214147/arabic-characters-from-html-content-to-pdf-using-itext>

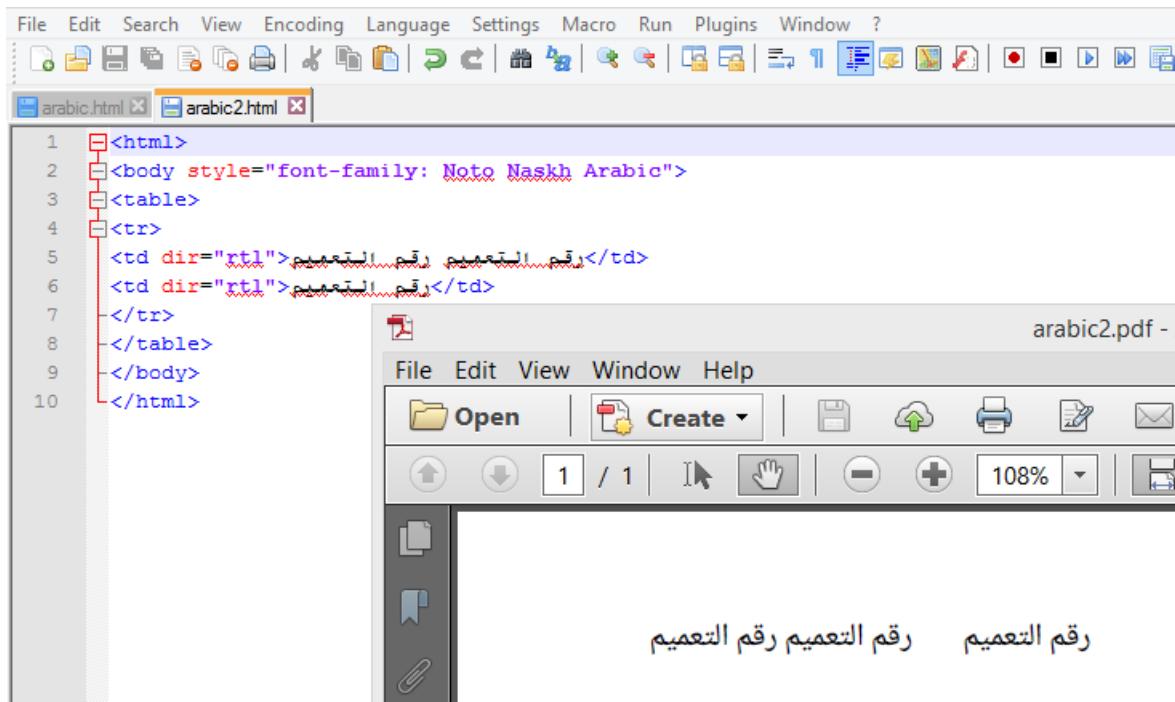
<sup>513</sup><http://stackoverflow.com/users/2579223/user2579223>

<sup>514</sup><http://itextpdf.com/sandbox/xmlworker/ParseHtml7>

<sup>515</sup><http://itextpdf.com/sandbox/xmlworker/ParseHtml8>



A PDF table with HTML content



An HTML table in PDF

Before we look at the code, allow me to explain that it's not a good idea to use non-ASCII characters in source code. For instance: this is not done:

```
htmlContentAr = "<p>﷽ ﻦﺴـtasdfـورـيـا</p>";
```

You never know how a Java file containing these glyphs will be stored. If it's not stored as UTF-8, the characters may end up looking like something completely different. Versioning systems are known

to have problems with non-ASCII characters and even compilers can get the encoding wrong. If you really want to stored hard-coded String values in your code, use the UNICODE notation.

For the examples shown in the screen shots, I saved the following files using UTF-8 encoding:

This is what you'll find in the file arabic.html:

```
<html>
<body style="font-family: Noto Naskh Arabic">
<p>﷽ ﷺ ﷺ ﷺ ﷺ </p>
<p>﷽ ﷺ ﷺ ﷺ </p>
</body>
</html>
```

This is what you'll find in the file arabic2.html:

```
<html>
<body style="font-family: Noto Naskh Arabic">
<table>
<tr>
<td dir="rtl">۰۰۰ ۰۰۰۰۰۰ ۰۰۰ ۰۰۰۰۰۰</td>
<td dir="rtl">۰۰۰ ۰۰۰۰۰۰</td>
</tr>
</table>
</body>
</html>
```

The second part of your problem concerns the font. It is important that you use a font that knows how to draw Arabic glyphs. It is hard to believe that you have arial.ttf right at the root of your C: drive. That's not a good idea. I would expect you to use C:/windows/fonts/arialuni.ttf which certainly knows Arabic glyphs.

Selecting the font isn't sufficient. Your HTML needs to know which font family to use. Because most of the examples in the documentation use Arial, I decided to use a NOTO font. I really like these fonts because they are nice and (almost) every language is supported. For instance, I used NotoNaskhArabic-Regular.ttf which means that I need to define the font familie like this:

style="font-family: Noto Naskh Arabic"

I defined the style in the body tag of my XML, it's obvious that you can choose where to define it: in an external CSS file, in the styles section of the `<head>`, at the level of a `<td>` tag,... That choice is entirely yours, but you have to define somewhere which font to use.

Of course: when XML Worker encounters font-family: Noto Naskh Arabic, iText doesn't know where to find the corresponding NotoNaskhArabic-Regular.ttf unless we register that font. We can do this, by creating an instance of the FontProvider interface. I chose to use the XMLWorkerFontProvider, but you're free to write your own FontProvider implementation:

```
XMLWorkerFontProvider fontProvider = new XMLWorkerFontProvider(XMLWorkerFontProvider.DONTLOOKFORFONTS);
fontProvider.register("resources/fonts/NotoNaskhArabic-Regular.ttf");
```

There is one more hurdle to take: Arabic is written from right to left. I see that you want to define the run direction at the level of the PdfPCell and that you add the HTML content to this cell using an ElementList. That's why I first wrote a similar example, named ParseHtml7<sup>516</sup>:

```
public void createPdf(String file)
    throws IOException, DocumentException {
    // step 1
    Document document = new Document();
    // step 2
    PdfWriter writer =
        PdfWriter.getInstance(document, new FileOutputStream(file));
    // step 3
    document.open();
    // step 4
    // Styles
    CSSResolver cssResolver = new StyleAttrCSSResolver();
    XMLWorkerFontProvider fontProvider =
        new XMLWorkerFontProvider(XMLWorkerFontProvider.DONTLOOKFORFONTS);
    fontProvider.register("resources/fonts/NotoNaskhArabic-Regular.ttf");
    CssAppliers cssAppliers = new CssAppliersImpl(fontProvider);
    // HTML
    HtmlPipelineContext htmlContext = new HtmlPipelineContext(cssAppliers);
    htmlContext.setTagFactory(Tags.getHtmlTagProcessorFactory());
    // Pipelines
    ElementList elements = new ElementList();
    ElementHandlerPipeline pdf = new ElementHandlerPipeline(elements, null);
    HtmlPipeline html = new HtmlPipeline(htmlContext, pdf);
    CssResolverPipeline css = new CssResolverPipeline(cssResolver, html);

    // XML Worker
    XMLWorker worker = new XMLWorker(css, true);
    XMLParser p = new XMLParser(worker);
    p.parse(new FileInputStream(HTML), Charset.forName("UTF-8"));

    PdfPTable table = new PdfPTable(1);
    PdfPCell cell = new PdfPCell();
    cell.setRunDirection(PdfWriter.RUN_DIRECTION_RTL);
```

---

<sup>516</sup><http://itextpdf.com/sandbox/xmlworker/ParseHtml7>

```

for (Element e : elements) {
    cell.addElement(e);
}
table.addCell(cell);
document.add(table);
// step 5
document.close();
}

```

There is no table in the HTML, but we create our own PdfPTable, we add the content from the HTML to a PdfPCell with run direction LTR, and we add this cell to the table, and the table to the document.

Maybe that's your actual requirement, but why would you do this in such a convoluted way? If you need a table, why don't you create that table in HTML and define some cells are RTL like this:

```
<td dir="rtl">...</td>
```

That way, you don't have to create an ElementList, you can just parse the HTML to PDF as is done in the [ParseHtml8<sup>517</sup>](#) example:

```

public void createPdf(String file)
    throws IOException, DocumentException {
    // step 1
    Document document = new Document();
    // step 2
    PdfWriter writer =
        PdfWriter.getInstance(document, new FileOutputStream(file));
    // step 3
    document.open();
    // step 4
    // Styles
    CSSResolver cssResolver = new StyleAttrCSSResolver();
    XMLWorkerFontProvider fontProvider =
        new XMLWorkerFontProvider(XMLWorkerFontProvider.DONTLOOKFORFONTS);
    fontProvider.register("resources/fonts/NotoNaskhArabic-Regular.ttf");
    CssAppliers cssAppliers = new CssAppliersImpl(fontProvider);
    HtmlPipelineContext htmlContext = new HtmlPipelineContext(cssAppliers);
    htmlContext.setTagFactory(Tags.getHtmlTagProcessorFactory());

    // Pipelines

```

---

<sup>517</sup><http://itextpdf.com/sandbox/xmlworker/ParseHtml8>

```

PdfWriterPipeline pdf = new PdfWriterPipeline(document, writer);
HtmlPipeline html = new HtmlPipeline(htmlContext, pdf);
CssResolverPipeline css = new CssResolverPipeline(cssResolver, html);

// XML Worker
XMLWorker worker = new XMLWorker(css, true);
XMLParser p = new XMLParser(worker);
p.parse(new FileInputStream(HTML), Charset.forName("UTF-8"));
// step 5
document.close();
}

```

There is less code needed in this example, and when you want to change the layout, it's sufficient to change the HTML. You don't need to change your Java code.

One more example: in [ParseHtml9<sup>518</sup>](#), I create a table with an English name in one column ("Lawrence of Arabia") and the Arabic translation in the other column "لورانس العرب". Because I need different fonts for English and Arabic, I define the font at the <td> level:

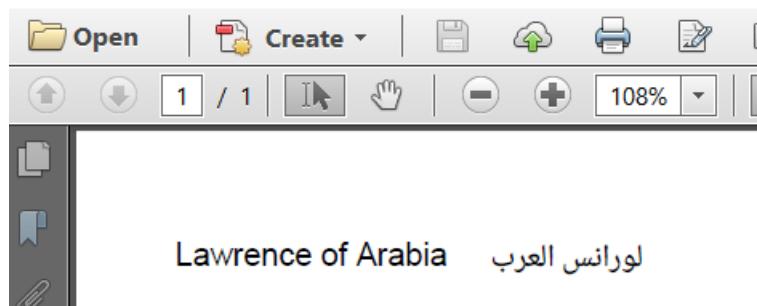
```


|                    |              |
|--------------------|--------------|
| Lawrence of Arabia | لورانس العرب |
|--------------------|--------------|


```

For the first column, the default font is used and no special settings are needed to write from left to right. For the second column, I define an Arabic font and I set the run direction to "rtl".

The result looks like this:



English next to Arabic

That's much easier than what you're trying to do in your code.

---

<sup>518</sup><http://itextpdf.com/sandbox/xmlworker/ParseHtml9>

# Inspect a PDF with iText

iText can tell you more about a PDF. What is the size of a page? Which measurement unit is used. All of these questions can be answered with a simple example using PdfReader.

## Why do I get an “`InvalidPdfException: PDF header signature not found`”?



I have some code that reads `pdf` files. The code fails at the line :

```
iTextSharp.text.pdf.PRTokeniser.CheckPdfHeader() at  
iTextSharp.text.pdf.PdfReader.ReadPdf()
```

I know from other entries that this issue is coming from some invalid formatting in the PDF. However I'm not in a position to tell my users to redo their PDFs. Is there some other way around this issue, that can allow reading of the pdf despite this problem?

Posted on StackOverflow on Sep 10, 2012 <sup>519</sup> by David Choi<sup>520</sup>

If a file doesn't start with %PDF- then there's nothing to fix: the file isn't a PDF file.

However, there may be another problem: maybe you're trying to access a file that has zero length due to some problem while creating the InputStream. Another context in which I've seen this happen, is a PDF loaded from a server, where the server returned a 404 message in HTML instead of a PDF file ;-)

Whenever that exception happens, you should store the bytes somewhere, and examine them. Without those bytes, nobody will be able to give you useful advice.

---

<sup>519</sup><http://stackoverflow.com/questions/12357126/invalidpdfexception-pdf-header-signature-not-found>

<sup>520</sup><http://stackoverflow.com/users/1426199/david-choi>

## How to Get PDF page width and height?



I have a PDF, and I want to get the width and height of each page using iTextSharp. This is what I have so far:

```
string source=@"D:\pdf\test.pdf";
PdfReader reader = new PdfReader(source);
```

Posted on StackOverflow on Aug 13, 2013 <sup>521</sup> by Mohamed Kamal<sup>522</sup>

Do you want the MediaBox?

```
Rectangle mediabox = reader.GetPageSize(page);
```

Do you want the rotation?

```
int rotation = reader.GetPageRotation(page);
```

Do you want the combination of both?

```
Rectangle pagesize = reader.GetPageSizeWithRotation(page);
```

Do you want the CropBox?

```
Rectangle cropbox = reader.GetCropBox(page);
```

These are some methods that will give you information about the dimensions of a page. Most of them return an object of type Rectangle that has methods such as getWidth() and getHeight() to get the width and the height of the page. Other useful methods are getLeft() and getRight() as well as getTop() and getBottom(). These four methods return the x and y coordinates that define the boundaries of your page.

---

<sup>521</sup><http://stackoverflow.com/questions/18202660/how-to-get-pdf-page-width-and-height>

<sup>522</sup><http://stackoverflow.com/users/2677532/mohamed-kamal>

## Why is the page size of a PDF always the same, no matter if it's landscape or portrait?



I have a PdfReader instance that contains some pages in landscape mode and other page in portrait. I need to distinguish them to perform some operations... However, if I call the getPageSize(), the value is always the same. Why isn't the value different for a page in landscape ? I've tried to find other methods to retrieve page width / orientation but nothing worked.

Here is my code :

```
for(int i = 0; i < pdfreader.getNumberOfPages(); i++) {  
    document = PdfStamper.getOverContent(i).getPdfDocument();  
    document.getPageSize().getWidth; //this will always be the same  
}
```

Posted on StackOverflow on Apr 28, 2014 <sup>523</sup> by Jean-François Savard<sup>524</sup>

There are two convenience methods, named getPageSize() and getPageSizeWithRotation().

Let's take a look at an example:

```
PdfReader reader =  
    new PdfReader("src/main/resources/pages.pdf");  
show(reader.getPageSize(1));  
show(reader.getPageSize(3));  
show(reader.getPageSizeWithRotation(3));  
show(reader.getPageSize(4));  
show(reader.getPageSizeWithRotation(4));
```

In this example, the show() method looks like this:

---

<sup>523</sup><http://stackoverflow.com/questions/23353146/pagesize-of-pdf-always-the-same-between-landscape-and-portrait-with-itextpdf>

<sup>524</sup><http://stackoverflow.com/users/2683146/jean-françois-savard>

```
public static void show(Rectangle rect) {
    System.out.print("llx: ");
    System.out.print(rect.getLeft());
    System.out.print(", lly: ");
    System.out.print(rect.getBottom());
    System.out.print(", urx: ");
    System.out.print(rect.getRight());
    System.out.print(", lly: ");
    System.out.print(rect.getTop());
    System.out.print(", rotation: ");
    System.out.println(rect.getRotation());
}
```

This is the output:

```
llx: 0.0, lly: 0.0, urx: 595.0, lly: 842.0, rotation: 0
llx: 0.0, lly: 0.0, urx: 595.0, lly: 842.0, rotation: 0
llx: 0.0, lly: 0.0, urx: 842.0, lly: 595.0, rotation: 90
llx: 0.0, lly: 0.0, urx: 842.0, lly: 595.0, rotation: 0
llx: 0.0, lly: 0.0, urx: 842.0, lly: 595.0, rotation: 0
```

Page 3 (see line 4 in code sample 3.8) is an A4 page just like page 1, but it's oriented in landscape. The /MediaBox entry is identical to the one used for the first page [0 0 595 842], and that's why getPageSize() returns the same result.

The page is in landscape, because the \Rotate entry in the page dictionary is set to 90. Possible values for this entry are 0 (which is the default value if the entry is missing), 90, 180 and 270.

The getPageSizeWithRotation() method takes this value into account. It swaps the width and the height so that you're aware of the difference. It also gives you the value of the /Rotate entry.

Page 4 also has a landscape orientation, but in this case, the rotation is mimicked by adapting the /MediaBox entry. In this case the value of the /MediaBox is [0 0 842 595] and if there's a /Rotate entry, its value is 0.

That explains why the output of the getPageSizeWithRotation() method is identical to the output of the getPageSize() method.

When I read your question, I see that you are looking for the rotation. This can be done with the getRotation() method.

**The major mistake you are making** can be found in this line:

```
document = PdfStamper.getOverContent(i).getPdfDocument();
```

This line doesn't make sense. You should not use the internal PdfDocument instance. If you want to know the size and orientation of a page, you should ask PdfReader for those values.

## How to get the UserUnit from a PDF file?



I have a bunch of PDF files that I read into a byte array one by one. I then pass these byte arrays to a PdfReader instance. Now I want know the dimensions of each page in pixels. From what I've read so far it seems by PDF files work in points, a point being a configurable unit stored in some kind of dictionary in an element called /UserUnit.

Loading my PDF file into a PdfReader, what do I need to do to get this user unit for each page (apparently it can vary from page to page) so I can then get the page dimensions in pixels.

At present I have this code, which grabs the dimensions for each page in "points". I guess I just need the /UserUnit value, and can then multiply these dimensions by that to get pixels or something similar.

```
PdfReader reader = new iTextSharp.text.pdf.PdfReader(file_content);
for (int i = 1; i <= reader.NumberOfPages; i++) {
    Rectangle dim = reader.GetPageSize(i);
    int[] xy = new int[] { (int)dim.Width, (int)dim.Height };
    page_data[objectid + '-' + i] = xy;
}
```

Posted on StackOverflow on Jan 29, 2013 <sup>525</sup> by Shawson<sup>526</sup>

Allow me to quote from my book, iText in Action - Second Edition, page 9:



### What is the measurement unit in PDF documents?

Most of the measurements in PDFs are expressed in user space units. ISO-32000-1 (section 8.3.2.3) tells us "the default for the size of the unit in default user space (1/72 inch) is approximately the same as a point (pt), a unit widely used in the printing industry. It is not exactly the same; there is no universal definition of a point."

In short, 1 in. = 25.4 mm = 72 user units (which roughly corresponds to 72 pt).

On the next page, I explain that it's possible to change the default value of the user unit, and I add an example on how to create a document with pages that have a different user unit.

Now for your question: suppose you have an existing PDF, how do you find which user unit was used? Before we answer this, we need to take a look at ISO-32000-1.

In section 7.7.3.3, entitled "Page Objects", you'll find the description of the /UserUnit entry in Table 30, "Entries in a page object":

<sup>525</sup><http://stackoverflow.com/questions/14586315/how-to-get-the-userunit-property-from-a-pdf-file-using-itextsharp-pdfreader>

<sup>526</sup><http://stackoverflow.com/users/192305/shawson>

(Optional; PDF 1.6) A positive number that shall give the size of default user space units, in multiples of 1/72 inch. The range of supported values shall be implementation-dependent. Default value: 1.0 (user space unit is 1/72 inch).

This key was introduced in PDF 1.6; you won't find it in older files. It's optional, so you won't always find it in every page dictionary. In my book, I also explain that the maximum value of the `UserUnit` key is 75,000.

**Now how to retrieve this value with iTextSharp?**

You already have `Rectangle dim = reader.GetPageSize(i);` which returns the `MediaBox`. This may not be the size of the visual part of the page. If there's a `CropBox` defined for the page, viewers will show a much smaller size than what you have in `xy` (but you probably knew that already).

What you need now is the page dictionary, so that you can retrieve the value of the `UserUnit` key:

```
PdfDictionary pageDict = reader.GetPageN(i);
PdfNumber userUnit = pageDict.GetAsNumber(PdfName.USERUNIT);
```

Most of the times `userUnit` will be `null`, but if it isn't you can use `userUnit.FloatValue`.

## How can I read Roman page numbers?



In Adobe Reader, the first pages of an ebook can have page numbers in the Roman number format. I would like to read these page numbers (instead of the indexed page number) with iText, but I don't know which properties (labels or annotations) I should use.

Posted on StackOverflow on Mar 20, 2015 <sup>527</sup> by T N<sup>528</sup>

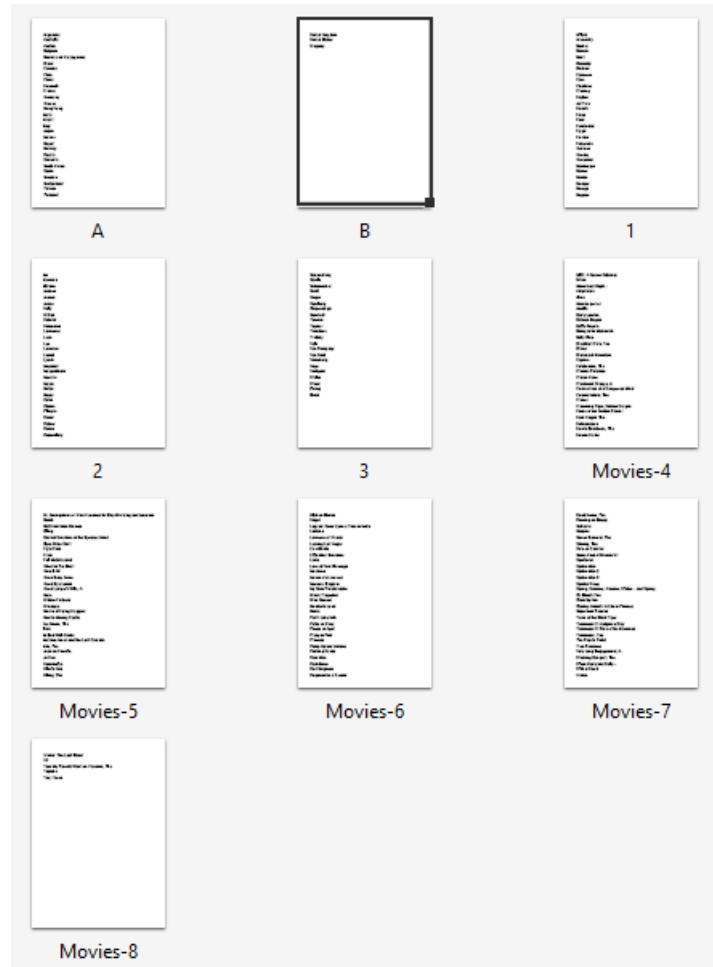
You are looking for the [PageLabelExample](#)<sup>529</sup>. In this example, we have a PDF, [page\\_labels.pdf](#)<sup>530</sup> that has pages numbered like this:

<sup>527</sup><http://stackoverflow.com/questions/29164682/itextsharp-read-roman-page-number-of-page>

<sup>528</sup><http://stackoverflow.com/users/4173702/t-n>

<sup>529</sup><http://itextpdf.com/examples/ii.php?id=234>

<sup>530</sup>[http://examples.itextpdf.com/results/part4/chapter13/page\\_labels.pdf](http://examples.itextpdf.com/results/part4/chapter13/page_labels.pdf)



Pages with page labels

In the `listPageLabels()` method, we create a txt file with all the page labels:

```
public void listPageLabels(String src, String dest) throws IOException {
    // no PDF, just a text file
    PrintStream out = new PrintStream(new FileOutputStream(dest));
    PdfReader reader = new PdfReader(src);
    String[] labels = PdfPageLabels.getPageLabels(reader);
    for (int i = 0; i < labels.length; i++) {
        out.println(labels[i]);
    }
    out.flush();
    out.close();
    reader.close();
}
```

The result looks like this:

```

A
B
1
2
3
Movies-4
Movies-5
Movies-6
Movies-7
Movies-8

```

If you want an iTextSharp example, take a look at this method:

```

/*
 * Reads the page labels from an existing PDF
 * @param src the existing PDF
 */
public string ListPageLabels(byte[] src) {
    StringBuilder sb = new StringBuilder();
    String[] labels = PdfPageLabels.GetPageLabels(new PdfReader(src));
    for (int i = 0; i < labels.Length; i++) {
        sb.Append(labels[i]);
        sb.AppendLine();
    }
    return sb.ToString();
}

```

## How do I get an object if I have its indirect reference?



I am using iTextSharp to analyze a form enabled PDF. I know how to navigate to the radio button control. I would like to analyze the individual radio buttons.

I have the `PdfArray` of the “kids” for the radio button. Each item in that array is a `PdfIndirectReference`. How do I get the actual object when all I have is the `PdfIndirectReference`?

Posted on StackOverflow on Jul 12, 2013 <sup>531</sup> by epotter<sup>532</sup>

Suppose that `array` is the `PdfArray` object, then you have a complete series of methods to get its elements. You’re probably using the `Get()` method, but you should use the `GetDirectObject()` one of the `GetAsX()` methods. For instance:

---

<sup>531</sup><http://stackoverflow.com/questions/17619141/using-itextsharp-how-do-i-get-an-object-if-i-have-its-indirectreference>

<sup>532</sup><http://stackoverflow.com/users/26339/epotter>

```
PdfDictionary d = array.GetAsDict(0);
PdfArray a = array.GetAsArray(1);
PdfObject o = array.GetDirectObject(2);
```

Please start reading [this book<sup>533</sup>](#) for more info.

## How to read PDFs created with an unknown random owner password?



Requirement is to process a batch of PDFs one at a time and on success encrypt each of them with an user password. However, these PDFs were encrypted previously with randomly generated dynamic owner password (*not know to any one*) to prevent any edits.

I use iText for encryption as shown below:

```
byte[] userPass = "user".getBytes();
byte[] ownerPass = "owner".getBytes();
PdfReader reader = new PdfReader("Misc.pdf");

PdfStamper stamper = new PdfStamper(reader,
    new FileOutputStream("Processed_Encrypted.pdf"));
stamper.setEncryption(userPass, ownerPass,
    PdfWriter.ALLOW_PRINTING, PdfWriter.ENCRYPTION_AES_128
    | PdfWriter.DO_NOT_ENCRYPT_METADATA);
stamper.close();
reader.close();
```

But this code throws an `com.itextpdf.text.exceptions.BadPasswordException`:  
PdfReader not opened with owner password

Can some one guide on how to resolve this error / bypass owner password?

Here I would like to make clear that we legally own these PDFs, so no crime / hacking is committed.

Posted on StackOverflow on [Apr 11, 2013<sup>534</sup>](#) by [Bond - Java Bond<sup>535</sup>](#)

`PdfReader` has an undocumented static boolean variable named `unethicalReading`. For obvious reasons, this variable is set to `false` by default. You could set this variable to true like this:

---

<sup>533</sup>[https://leanpub.com/itext\\_pdfabc/](https://leanpub.com/itext_pdfabc/)

<sup>534</sup><http://stackoverflow.com/questions/15955620/issue-while-setting-user-password-for-pdfs-created-with-unknown-random-owner-pas>

<sup>535</sup><http://stackoverflow.com/users/1910582/bond-java-bond>

```
PdfReader.unethicalreading = true;
```

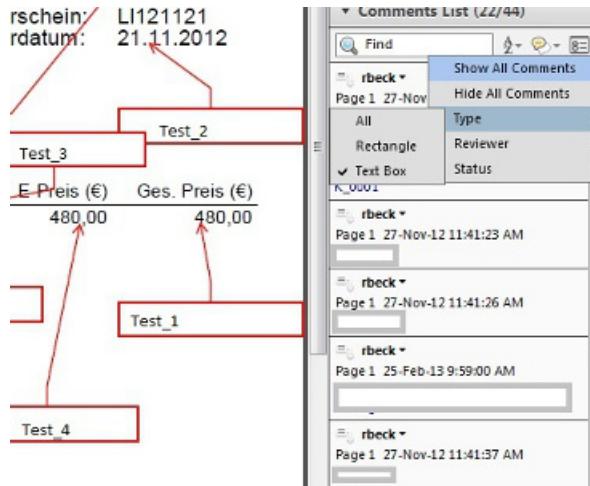
From now on, PdfReader will ignore the presence of an owner password. It will only throw an exception if a user password is in place.

Use this at your own risk.

## How to get values from comment annotations?



I have a pdf document, inside are comments lists inside rectangles and text boxes:



Screen shot

I want to get values from Text Boxes with c# and itextsharp.

Posted on StackOverflow on Apr 5, 2013 <sup>536</sup> by Alex<sup>537</sup>

The text boxes and rectangles you're referring to are called Annotations. Annotations are defined as dictionaries and they are listed per page.

In other words: you need to create a PdfReader instance and get the ANNOTS from each page:

<sup>536</sup><http://stackoverflow.com/questions/15830050/how-to-get-values-from-textbox-comments-in-pdf-document>

<sup>537</sup><http://stackoverflow.com/users/1180365/alex>

```

PdfReader reader = new PdfReader("your.pdf");
for (int i = 1; i <= reader.NumberOfPages; i++) {
    PdfArray array = reader.GetPageN(i).GetAsArray(PdfName.ANOTS);
    if (array == null) continue;
    for (int j = 0; j < array.Size; j++) {
        PdfDictionary annot = array.GetAsDict(j);
        PdfString text = annot.GetAsString(PdfName.CONTENTS);
        ...
    }
}

```

In the above code sample, I have a `PdfDictionary` named `annot`, from which I can extract the `Contents`. You may be interested in some other entries too (for instance the name of the annotation, if any). Please inspect all the keys that are available in the `annot` object in case the `Contents` entry isn't what you're looking for.

Replace the dots with whatever you want to do with the text. `PdfString` has different method that will reveal its contents.

## How can I find the border color of a field?



Is there anyway of finding the border color of a specific field in my PDF? I could get `AcroField.Item`, but I don't see an option to get border color from there.

Posted on StackOverflow on Feb 4, 2015 <sup>538</sup> by user2296988<sup>539</sup>

Please take a look at this PDF: [text\\_fields.pdf](#)<sup>540</sup>. This PDF was created using the [TextFields](#)<sup>541</sup> example. The following code snippet was used to set the border of the field with name `text_2`:

```

text.setBorderStyle(PdfBorderDictionary.STYLE_SOLID);
text.setBorderColor(BaseColor.BLUE);
text.setBorderWidth(2);

```

Now when we look inside the PDF using [iText RUPS](#)<sup>542</sup>, and we take a look at the field dictionary / widget annotation for this field, we see the following structure:

---

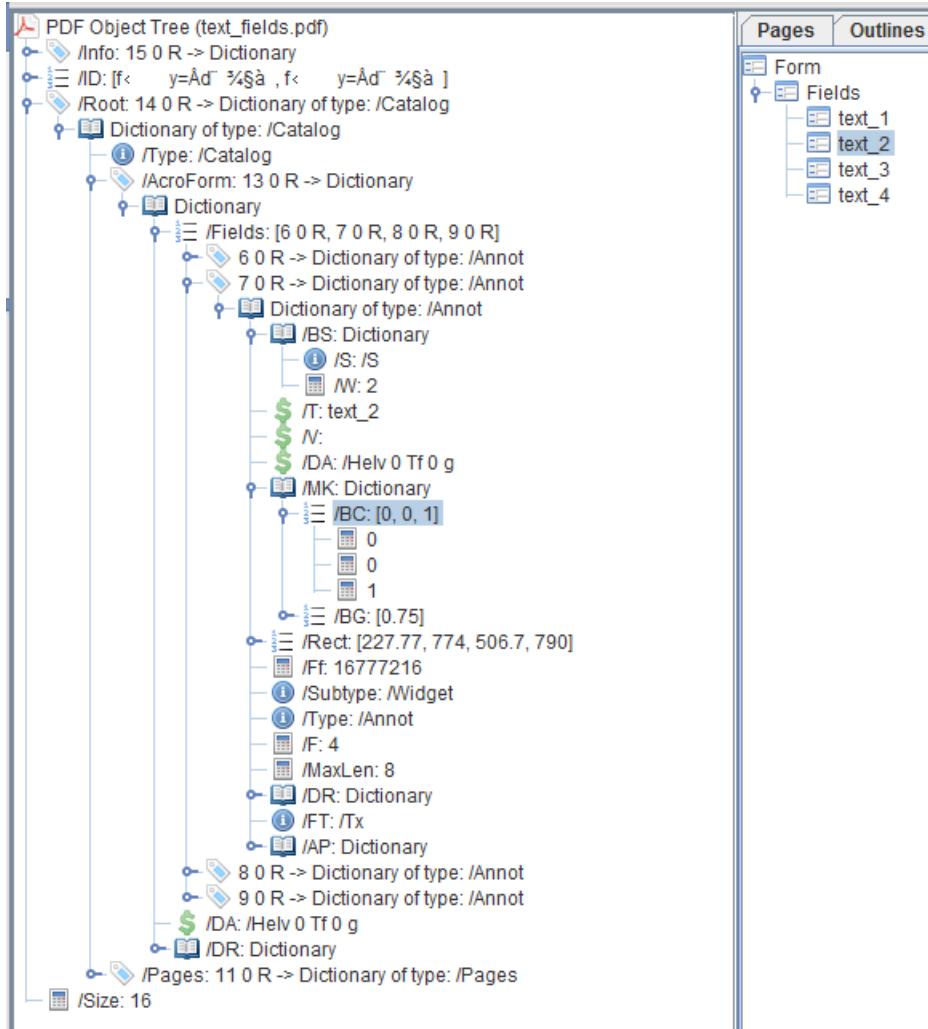
<sup>538</sup><http://stackoverflow.com/questions/28312620/can-i-find-bordercolor-of-a-field-in-pdf-using-itext>

<sup>539</sup><http://stackoverflow.com/users/2296988/user2296988>

<sup>540</sup>[http://examples.itextpdf.com/results/part2/chapter08/text\\_fields.pdf](http://examples.itextpdf.com/results/part2/chapter08/text_fields.pdf)

<sup>541</sup><http://itextpdf.com/examples/ii.php?id=157>

<sup>542</sup>[http://itextpdf.com/product/itext\\_rups](http://itextpdf.com/product/itext_rups)



Internal structure of a form field

We see a /BS dictionary that defines a solid border style (the value for the /S key is /S) and a border width (/W) with value 2.

We also see that the border color (/BC) entry of the /MK entry is an array with three values: [ 0 0 1 ]. This means that the border color is an RGB color where the value for Red is 0, the value for Green is 0, and the value for Blue is 1. This is consistent with us setting the color to `BaseColor.BLUE` when we created the file.

You say that you have the `AcroField.Item` object for a field. Now you need to get the merged field / widget annotation dictionary and follow the path shown by iText RUPS:

```
AcroFields.Item item = acroFields.getFieldItem(fldName);
PdfDictionary merged = item.getMerged(0);
PdfDictionary mk = merged.getAsDict(PdfName.MK);
PdfArray bc = mk.getAsArray(PdfName.BC);
```

The values stored in the array bc will inform you about the background color. If the array has only one value, you have a *gray* color, if there are three, you have an *RGB* color, if there are four, you have a *CMYK* color.

**Warning:** some values may not be present (e.g. there may be no /BC entry). In that case you can get NullPointerExceptions.

## How to read bookmark titles?



I am working with a single PDF containing multiple documents. Each document has a bookmark. I need to read the bookmark names for a reconciliation application that I am building. The code below is not working for me.

```
PdfReader reader = new PdfReader("C:\\\\Work\\\\Input.pdf");
List<HashMap<String, Object>> bookmarks = SimpleBookmark.getBookmark(reader);
for(int i = 0; i < bookmarks.size(); i++){
    HashMap<String, Object> bm = bookmarks.get(i);
    String title = ((String)bm.get("Title"));
}
```

I am trying to place the bookmark name in the *title* string.

Posted on StackOverflow on Feb 20, 2015 <sup>543</sup> by jcalder<sup>544</sup>

You are not taking into account that bookmarks are stored in a tree structure with branches and leaves (in the PDF specification, it's called the *outline tree*).

Your code works for the top-level, but if you want to see *all* the titles, you need to use a recursive method that also looks at the "Kids".

Take a look at [this code sample<sup>545</sup>](#):

---

<sup>543</sup><http://stackoverflow.com/questions/28634172/java-reading-pdf-bookmark-names-with-itext>

<sup>544</sup><http://stackoverflow.com/users/4588607/jcalder>

<sup>545</sup><http://itextpdf.com/sandbox/interactive/FetchBookmarkTitles>

```

public void inspectPdf(String filename) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(filename);
    List<HashMap<String, Object>> bookmarks = SimpleBookmark.getBookmark(reader);
    for (int i = 0; i < bookmarks.size(); i++) {
        showTitle(bookmarks.get(i));
    }
    reader.close();
}

public void showTitle(HashMap<String, Object> bm) {
    System.out.println((String)bm.get("Title"));
    List<HashMap<String, Object>> kids =
        (List<HashMap<String, Object>>)bm.get("Kids");
    if (kids != null) {
        for (int i = 0; i < kids.size(); i++) {
            showTitle(kids.get(i));
        }
    }
}
}

```

The `showTitle()` method is recursive. It calls itself if an examined bookmark entry has kids. With this code snippet, you can walk through all the branches and leaves of the outline tree.

## How to read bookmarks?



I'm making a tool that scans PDF files and searches for text in PDF bookmarks. How do I load a list of bookmarks from an existing PDF file?

Posted on StackOverflow on Jan 2, 2015 <sup>546</sup> by excitive<sup>547</sup>

It depends on what you understand when you say “bookmarks”.

**You want the outlines (the entries that are visible in the bookmarks panel):**

The [CreateOnlineTree<sup>548</sup>](#) examples shows you how to use the `SimpleBookmark` class to create an XML file containing the complete outline tree (in PDF jargon, bookmarks are called outlines).

Java:

---

<sup>546</sup><http://stackoverflow.com/questions/27739820/reading-pdf-bookmarks-in-vb-net-using-itextsharp>

<sup>547</sup><http://stackoverflow.com/users/931406/excitive>

<sup>548</sup><http://itextpdf.com/examples/iiapdf?id=139>

```

PdfReader reader = new PdfReader(src);
List<HashMap<String, Object>> list = SimpleBookmark.getBookmark(reader);
SimpleBookmark.exportToXML(list,
    new FileOutputStream(dest), "ISO8859-1", true);
reader.close();

```

C#:

```

PdfReader reader = new PdfReader(pdfIn);
var list = SimpleBookmark.GetBookmark(reader);
using (MemoryStream ms = new MemoryStream()) {
    SimpleBookmark.ExportToXML(list, ms, "ISO8859-1", true);
    ms.Position = 0;
    using (StreamReader sr = new StreamReader(ms)) {
        return sr.ReadToEnd();
    }
}

```

The list object can also be used to examine the different bookmark elements one by one programmatically (this is all explained in the official documentation).

**You want the named destinations (specific places in the document you can link to by name):**

Now suppose that you meant to say named destinations, then you need the SimpleNamedDestination class as shown in the [LinkAnnotations<sup>549</sup>](#) example:

Java:

```

PdfReader reader = new PdfReader(src);
HashMap<String, String> map = SimpleNamedDestination.getNamedDestination(reader, false);
SimpleNamedDestination.exportToXML(map, new FileOutputStream(dest),
    "ISO8859-1", true);
reader.close();

```

C#:

---

<sup>549</sup><http://itextpdf.com/examples/ii.php?id=131>

```
PdfReader reader = new PdfReader(src);
Dictionary<string,string> map = SimpleNamedDestination
    .GetNamedDestination(reader, false);
using (MemoryStream ms = new MemoryStream()) {
    SimpleNamedDestination.ExportToXML(map, ms, "ISO8859-1", true);
    ms.Position = 0;
    using (StreamReader sr = new StreamReader(ms)) {
        return sr.ReadToEnd();
    }
}
```

The `map` object can also be used to examine the different named destinations one by one programmatically. Note the Boolean parameter that is used when retrieving the named destinations. Named destinations can be stored using a PDF name object as name, or using a PDF string object. The Boolean parameter indicates whether you want the former (`true` = stored as PDF name objects) or the latter (`false` = stored as PDF string objects) type of named destinations.

Named destinations are predefined targets in a PDF file that can be found through their name. Although the official name is named destinations, some people refer to them as bookmarks too (but when we say bookmarks in the context of PDF, we usually want to refer to outlines).

## How To find internal links in a PDF file?



I am using ItextSharp for searching internal links in a PDF file. This is already done with External Links.

```

//Get the current page
PdfDictionary PageDictionary = R.GetPageN(page);
//Get all of the annotations for the current page
PdfArray Annots = PageDictionary.GetAsArray(PdfName.ANNOTS);
//Make sure we have something
if ((Annots == null) || (Annots.Length == 0)) {
    Console.WriteLine("nothing");
}
//Loop through each annotation
if (Annots != null) {
    foreach (PdfObject A in Annots.ArrayList) {
        //Convert the itext-specific object as a generic PDF object
        PdfDictionary AnnotationDictionary =
            (PdfDictionary)PdfReader.GetPdfObject(A);
        //Make sure this annotation has a link
        if (!AnnotationDictionary.Get(PdfName.SUBTYPE).Equals(PdfName.LINK))
            continue;
        //Make sure this annotation has an ACTION
        if (AnnotationDictionary.Get(PdfName.A) == null)
            continue;
        //Get the ACTION for the current annotation
        PdfDictionary AnnotationAction =
            AnnotationDictionary.GetAsDict(PdfName.A);
        // Test if it is a URI action (There are tons of other types of actions,
        // some of which might mimic URI, such as JavaScript,
        // but those need to be handled separately)
        if (AnnotationAction.Get(PdfName.S).Equals(PdfName.URI)) {
            PdfString Destination = AnnotationAction.GetAsString(PdfName.URI);
            string url1 = Destination.ToString();
        }
    }
}

```

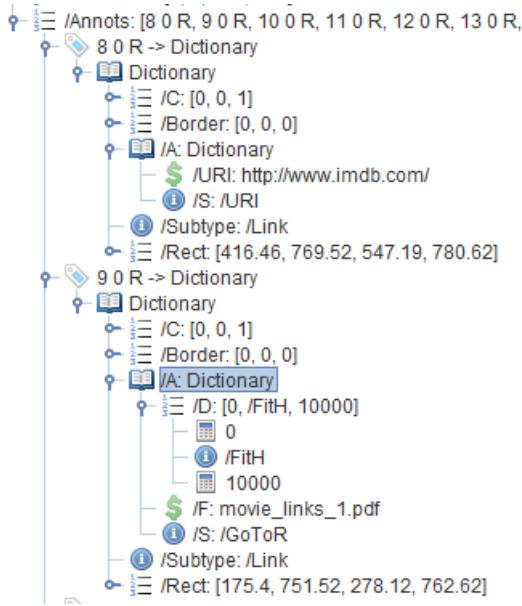
Posted on StackOverflow on Feb 22, 2014 <sup>550</sup> by Ashwani<sup>551</sup>

---

<sup>550</sup><http://stackoverflow.com/questions/21952616/how-to-find-cross-referencesinternal-links-in-pdf-file-using-itextsharp-lib>

<sup>551</sup><http://stackoverflow.com/users/3340285/ashwani>

You've already done most of the work. Please take a look at the following screen shot:



Internal view of the PDF

You see the /Annots array of a page. You are already parsing that array in your code and you skip all annotations that aren't of the /Subtype /Link or don't have an /A key, which is excellent.

Currently you're only looking for values of /S that are of type /URI. You say you're already done with external links, but that's not true: you should also look for entries where /S is /GoToR (remote goto). If you want internal links, you need to look for /S values equal to /GoTo, /GoToE, and (in the future) /GoToDp. Maybe you also want to remove the /JavaScript actions, because they can also be used to jump to a specific page.

## How to extract embedded streams?



I have embedded a byte array into a PDF file, more specifically an AVI file in a RichMedia annotation. Now I am trying to extract that same array. How can I do this?

Posted on StackOverflow on May 17, 2015 <sup>552</sup> by Itai Soudry<sup>553</sup>

I have written a brute force method to extract *all* streams in a PDF and store them as a file without an extension:

<sup>552</sup><http://stackoverflow.com/questions/30286601/extracting-an-embedded-object-from-a-pdf>

<sup>553</sup><http://stackoverflow.com/users/3759717/itai-soudry>

```

public static final String SRC = "resources/pdfs/image.pdf";
public static final String DEST = "results/parse/stream%s";

public static void main(String[] args) throws IOException {
    File file = new File(DEST);
    file.getParentFile().mkdirs();
    new ExtractStreams().parse(SRC, DEST);
}

public void parse(String src, String dest) throws IOException {
    PdfReader reader = new PdfReader(src);
    PdfObject obj;
    for (int i = 1; i <= reader.getXrefSize(); i++) {
        obj = reader.getPdfObject(i);
        if (obj != null && obj.isStream()) {
            PRStream stream = (PRStream)obj;
            byte[] b;
            try {
                b = PdfReader.getStreamBytes(stream);
            }
            catch(UnsupportedPdfException e) {
                b = PdfReader.getStreamBytesRaw(stream);
            }
            FileOutputStream fos = new FileOutputStream(String.format(dest, i));
            fos.write(b);
            fos.flush();
            fos.close();
        }
    }
}
}

```

Note that I get all PDF objects that are streams as a `PRStream` object. I also use two different methods:

- When I use `PdfReader.getStreamBytes(stream)`, iText will look at the *filter*. For instance: page content streams consists of PDF syntax that is compressed using `/FlateDecode`. By using `PdfReader.getStreamBytes(stream)`, you will get the *uncompressed* PDF syntax.
- Not all filters are supported in iText. Take for instance `/DCTDecode` which is the filter used to store JPEGs inside a PDF. Why and how would you “decode” such a stream? You wouldn’t, and that’s when we use `PdfReader.getStreamBytesRaw(stream)` which is also the method you need to get your AVI-bytes from your PDF.

This example already gives you the methods you’ll certainly need to extract PDF streams. Now it’s

up to you to find the path to the stream you need. That calls for [iText RUPS<sup>554</sup>](#). With iText RUPS you can look at the internal structure of a PDF file. In your case, you need to find the annotations as is done in this question: [How to change the zoom factor in link annotations?](#)<sup>555</sup>

You loop over the page dictionaries, then loop over the `/Annots` array of this dictionary (if it's present), but instead of checking for `/Link` annotations (which is what was asked in the question I refer to), you have to check for `/RichMedia` annotations and from there examine the assets until you find the stream that contains the AVI file. RUPS will show you how to dive into the annotation dictionary.

---

<sup>554</sup>[http://itextpdf.com/product/itext\\_rups](http://itextpdf.com/product/itext_rups)

<sup>555</sup><http://stackoverflow.com/questions/22828782/all-links-of-existing-pdf-change-the-action-property-to-inherit-zoom-itext-lib>

# Manipulating existing PDFs

In this chapter, we're going to solve some problems when working with existing PDFs that need to be split into different files, merged or stamped. Usually, we are going to use a combination of `PdfReader` to read the document and `PdfStamper`, `PdfCopy` or `PdfSmartCopy` to create a new PDF. Note that we'll skip filling out interactive forms for now. We'll deal with AcroForm and XFA technology in the next chapter.

## How to update a PDF without creating a new PDF?



I need to change the value of a field in an existing PDF file. I am using `PdfReader`, `PdfStamper` and `AcroFields` and that's working fine. But, in doing so, it is required to create a new PDF and I would like the change to be reflected in the existing PDF itself. If I am setting the destination `filename` to be the same as the original `filename`, then my application fails.

Posted on StackOverflow on Apr 18, 2013 <sup>556</sup> by tk2013<sup>557</sup>

You can't read a file and write to it simultaneously. Think of how Microsoft Word works: you can't open a Word document and write directly to it. Word always creates a temporary file, writes the changes to it, then replaces the original file with it, and then throws away the temporary file.

You can do that too:

- read the original file with `PdfReader`;
- create a temporary file for `PdfStamper`, and when you're done,
- replace the original file with the temporary file.

Or:

- read the original file into a `byte[]`,
- create `PdfReader` with this `byte[]`, and
- use the path to the original file for `PdfStamper`.

The latter option is more dangerous, as you'll lose the original file if you do something that causes an exception in `PdfStamper`. If I were you, I'd create a temporary file.

<sup>556</sup><http://stackoverflow.com/questions/16081831/using-itextsharp-stamper-required-to-update-in-the-same-pdf>

<sup>557</sup><http://stackoverflow.com/users/2239456/tk2013>

## How to add an image watermark to a PDF file?



I'm using C# and iTextSharp to add a watermark to my PDF files:

```
Document document = new Document();
PdfReader pdfReader = new PdfReader(strFileLocation);
PdfStamper pdfStamper = new PdfStamper(pdfReader, new FileStream(strFileLocation\Out, FileMode.Create, FileAccess.Write, FileShare.None));
iTextSharp.text.Image img = iTextSharp.text.Image.GetInstance(WatermarkLocation);
img.SetAbsolutePosition(100, 300);
PdfContentByte waterMark;
for (int pageIndex = 1; pageIndex <= pdfReader.NumberOfPages; pageIndex++) {
    waterMark = pdfStamper.GetOverContent(pageIndex);
    waterMark.AddImage(img);
}
pdfStamper.FormFlattening = true;
pdfStamper.Close();
```

It works fine, but my problem is that in some PDF files no watermark is added although the file size increased, any idea?

Posted on StackOverflow on Jul 8, 2013 <sup>558</sup> by Abady<sup>559</sup>

The fact that the file size increases is a good indication that the watermark is added. The main problem is that you're adding the watermark outside the visible area of the page. See my answer to the question *How to position text relative to page using iText?*

You need something like this:

```
Rectangle pagesize = reader.getCropBox(pageIndex);
if (pagesize == null)
    pagesize = reader.getMediaBox(pageIndex);
img.SetAbsolutePosition(
    pagesize.GetLeft(),
    pagesize.GetBottom());
```

That is: if you want to add the image in the lower-left corner of the page. You can add an offset, but make sure the offset in the x direction doesn't exceed the width of the page, and the offset in the y direction doesn't exceed the height of the page.

---

<sup>558</sup><http://stackoverflow.com/questions/17522965/how-to-add-a-watermark-to-a-pdf-file>

<sup>559</sup><http://stackoverflow.com/users/1450667/abady>

## How to add a watermark to a page with an opaque image?



Is it possible to add a watermark that will be transparent on top of an image? I currently use:

```
PdfContentByte canvas = pdfStamper.getUnderContent(i);
ColumnText.showTextAligned(
    canvas, Element.ALIGN_CENTER, watermark, 298, 421, 45);
```

It works fine, but if the content consists of an image, the watermark is hidden by that image.

Posted on JIRA (our closed ticketing system for customers) on Mar 19, 2015

If you have opaque shapes in your PDF (such as images, but also colored shapes), you need to add the Watermark on top of the existing content:

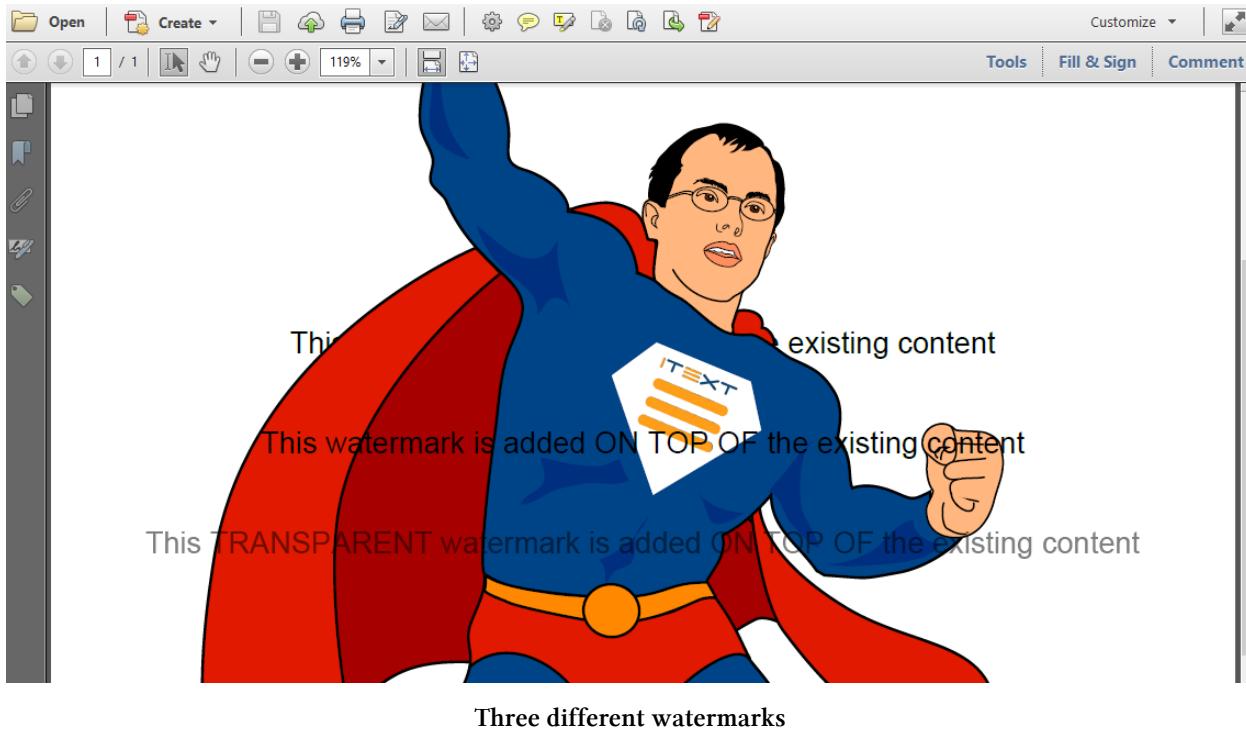
```
PdfContentByte canvas = pdfStamper.getOverContent(i);
```

Now the text will cover the images, but it may hide some important information. If you want to avoid this, you need to introduce transparency.

I have written a simple example that shows how this is done. It is called [TransparentWatermark<sup>560</sup>](#). Let's take a look at the result:

---

<sup>560</sup><http://itextpdf.com/sandbox/stamper/TransparentWatermark>



First I add the text “This watermark is added UNDER the existing content” under the existing content. Part of the text is hidden (as you indicate in your question). Then I add the text “This watermark is added ON TOP OF the existing content” on top of the existing content. This may be sufficient, unless you fear that some crucial information will get lost by covering the existing content. In that case, take a look at how I add the text “This TRANSPARENT watermark is added ON TOP OF the existing content”:

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    int n = reader.getNumberOfPages();
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    PdfContentByte under = stamper.getUnderContent(1);
    Font f = new Font(FontFamily.HELVETICA, 15);
    Phrase p = new Phrase(
        "This watermark is added UNDER the existing content", f);
    ColumnText.showTextAligned(under, Element.ALIGN_CENTER, p, 297, 550, 0);
    PdfContentByte over = stamper.getOverContent(1);
    p = new Phrase("This watermark is added ON TOP OF the existing content", f);
    ColumnText.showTextAligned(over, Element.ALIGN_CENTER, p, 297, 500, 0);
    p = new Phrase(
        "This TRANSPARENT watermark is added ON TOP OF the existing content", f);
    over.saveState();
```

```
PdfGState gs1 = new PdfGState();
gs1.setFillOpacity(0.5f);
over.setGState(gs1);
ColumnText.showTextAligned(over, Element.ALIGN_CENTER, p, 297, 450, 0);
over.restoreState();
stamper.close();
reader.close();
}
```

Some extra tips and tricks:

- Always use `saveState()` and `restoreState()` when you change the graphics state. If you don't you may get undesirable effects such as other content that is affected by the changes you make (e.g. you don't want all the content to become transparent).
- The default rendering mode of text is "fill", hence I change the fill opacity.
- In this case, I defined a fill opacity of 50% (0.5f). Choose any value between 0.0f and 1.0f if you want to change the transparency of the text.

## How to watermark PDFs using text or images?



I have a bunch of PDF documents in a folder and I want to augment them with a watermark. What are my options from a Java serverside context? Preferably the watermark will support transparency. Both vector and raster is desirable.

Posted on StackOverflow on Apr 10, 2015 <sup>561</sup> by Lennart Rolland<sup>562</sup>

Please take a look at the [TransparentWatermark2<sup>563</sup>](#) example. It adds transparent text on each odd page and a transparent image on each even page of an existing PDF document.

This is how it's done:

---

<sup>561</sup><http://stackoverflow.com/questions/29560373/how-to-watermark-pdfs-using-text-or-images>

<sup>562</sup><http://stackoverflow.com/users/1035897/lennart-rolland>

<sup>563</sup><http://itextpdf.com/sandbox/stamper/TransparentWatermark2>

```

public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    int n = reader.getNumberOfPages();
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    // text watermark
    Font f = new Font(FontFamily.HELVETICA, 30);
    Phrase p = new Phrase("My watermark (text)", f);
    // image watermark
    Image img = Image.getInstance(IMG);
    float w = img.getScaledWidth();
    float h = img.getScaledHeight();
    // transparency
    PdfGState gs1 = new PdfGState();
    gs1.setFillOpacity(0.5f);
    // properties
    PdfContentByte over;
    Rectangle pagesize;
    float x, y;
    // loop over every page
    for (int i = 1; i <= n; i++) {
        pagesize = reader.getPageSizeWithRotation(i);
        x = (pagesize.getLeft() + pagesize.getRight()) / 2;
        y = (pagesize.getTop() + pagesize.getBottom()) / 2;
        over = stamper.getOverContent(i);
        over.saveState();
        over.setGState(gs1);
        if (i % 2 == 1)
            ColumnText.showTextAligned(over, Element.ALIGN_CENTER, p, x, y, 0);
        else
            over.addImage(img, w, 0, 0, h, x - (w / 2), y - (h / 2));
        over.restoreState();
    }
    stamper.close();
    reader.close();
}

```

As you can see, we create a `Phrase` object for the text and an `Image` object for the image. We also create a `PdfGState` object for the transparency. In our case, we go for 50% opacity (change the `0.5f` into something else to experiment).

Once we have these objects, we loop over every page. We use the `PdfReader` object to get information about the existing document, for instance the dimensions of every page. We use the `PdfStamper`

object when we want to stamp extra content on the existing document, for instance adding a watermark on top of each single page.

When changing the graphics state, it is always safe to perform a `saveState()` before you start and to `restoreState()` once you're finished. Your code will probably also work if you don't do this, but believe me: it can save you plenty of debugging time if you adopt the discipline to do this as you can get really strange effects if the graphics state is out of balance.

We apply the transparency using the `setGState()` method and depending on whether the page is an odd page or an even page, we add the text (using `ColumnText` and an  $(x, y)$  coordinate calculated so that the text is added in the middle of each page) or the image (using the `addImage()` method and the appropriate parameters for the transformation matrix).

Once you've done this for every page in the document, you have to `close()` the stamper and the reader.

#### Caveat:

You'll notice that pages 3 and 4 are in landscape, yet there is a difference between those two pages that isn't visible to the naked eye. Page 3 is actually a page of which the size is defined as if it were a page in portrait, but it is rotated by 90 degrees. Page 4 is a page of which the size is defined in such a way that the width > the height.

This can have an impact on the way you add a watermark, but if you use `getPageSizeWithRotation()`, iText will adapt. This may not be what you want: maybe you want the watermark to be added differently.

Take a look at [TransparentWatermark3<sup>564</sup>](#):

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    int n = reader.getNumberOfPages();
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.setRotateContents(false);
    // text watermark
    Font f = new Font(FontFamily.HELVETICA, 30);
    Phrase p = new Phrase("My watermark (text)", f);
    // image watermark
    Image img = Image.getInstance(IMG);
    float w = img.getScaledWidth();
    float h = img.getScaledHeight();
    // transparency
    PdfGState gs1 = new PdfGState();
    gs1.setFillOpacity(0.5f);
```

<sup>564</sup><http://itextpdf.com/sandbox/stamper/TransparentWatermark3>

```
// properties
PdfContentByte over;
Rectangle pagesize;
float x, y;
// loop over every page
for (int i = 1; i <= n; i++) {
    pagesize = reader.getPageSize(i);
    x = (pagesize.getLeft() + pagesize.getRight()) / 2;
    y = (pagesize.getTop() + pagesize.getBottom()) / 2;
    over = stamper.getOverContent(i);
    over.saveState();
    over.setGState(gs1);
    if (i % 2 == 1)
        ColumnText.showTextAligned(over, Element.ALIGN_CENTER, p, x, y, 0);
    else
        over.addImage(img, w, 0, 0, h, x - (w / 2), y - (h / 2));
    over.restoreState();
}
stamper.close();
reader.close();
}
```

In this case, we don't use `getPageSizeWithRotation()` but simply `getPageSize()`. We also tell the stamper *not* to compensate for the existing page rotation: `stamper.setRotateContents(false)`;

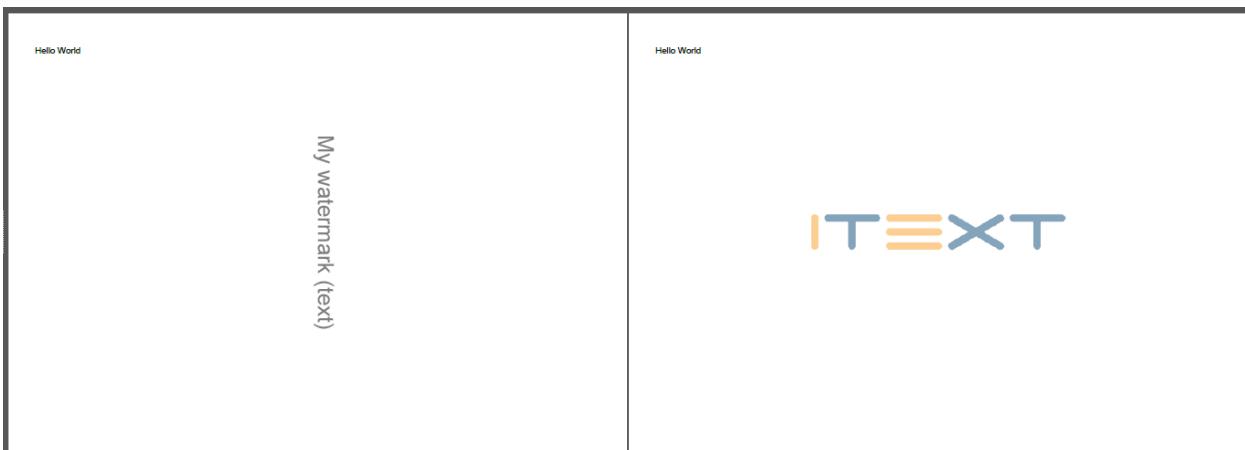
Take a look at the difference in the resulting PDFs:

In the first screen shot (showing page 3 and 4 of the resulting PDF of `TransparentWatermark2`), the page to the left is actually a page in portrait rotated by 90 degrees. iText however, treats it as if it were a page in landscape just like the page to the right.



Watermark without rotation

In the second screen shot (showing page 3 and 4 of the resulting PDF of `TransparentWatermark3`), the page to the left is a page in portrait rotated by 90 degrees and we add the watermark as if the page is in portrait. As a result, the watermark is also rotated by 90 degrees. This doesn't happen with the page to the right, because that page has a rotation of 0 degrees.



Watermark with rotation

This is a subtle difference, but I thought you'd want to know.

If you want to read this answer in French, please read [Comment créer un filigrane transparent en PDF?](#)<sup>565</sup>

---

<sup>565</sup><http://www.developpez.net/forums/blogs/133351-blowagie/b432/creer-filigrane-transparent-pdf/>

## How to extend the page size of a PDF to add a watermark?



I want to add a watermark to a document that doesn't cover any of the existing text (not even if the watermark is transparent). Instead, I want to add a small margin on the left side of each page and add the watermark there. Is this possible?

Posted on StackOverflow on Apr 21, 2015 <sup>[566](#)</sup> by Eduardo<sup>[567](#)</sup>

I will break up the question in two parts and I'll skip the part about the actual watermarking as this is already explained elsewhere on StackOverflow and in this book. I will focus on the extra requirement to add an extra margin to the right.

Take a look at the [primes.pdf](#)<sup>[568](#)</sup> document. This is the source file we are going to use in the [AddExtraMargin](#)<sup>[569](#)</sup> example with the following result: [primes\\_extra\\_margin.pdf](#)<sup>[570](#)</sup>. As you can see, a half an inch margin was added to the left of each page.

This is how it's done:

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    int n = reader.getNumberOfPages();
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    // properties
    PdfContentByte over;
    PdfDictionary pageDict;
    PdfArray mediabox;
    float llx, lly, ury;
    // loop over every page
    for (int i = 1; i <= n; i++) {
        pageDict = reader.getPageN(i);
        mediabox = pageDict.getAsArray(PdfName.MEDIABOX);
        llx = mediabox.getAsNumber(0).floatValue();
        lly = mediabox.getAsNumber(1).floatValue();
        ury = mediabox.getAsNumber(3).floatValue();
        mediabox.set(0, new PdfNumber(llx - 36));
```

---

<sup>566</sup><http://stackoverflow.com/questions/29775893/watermark-in-a-pdf-with-itext>

<sup>567</sup><http://stackoverflow.com/users/1454121/eduardo>

<sup>568</sup><http://itextpdf.com/sites/default/files/primes.pdf>

<sup>569</sup><http://itextpdf.com/sandbox/stamper/AddExtraMargin>

<sup>570</sup>[http://itextpdf.com/sites/default/files/primes\\_extra\\_margin.pdf](http://itextpdf.com/sites/default/files/primes_extra_margin.pdf)

```

        over = stamper.getOverContent(i);
        over.saveState();
        over.setFillColor(new GrayColor(0.5f));
        over.rectangle(l1x - 36, l1y, 36, ury - l1x);
        over.fill();
        over.restoreState();
    }
    stamper.close();
    reader.close();
}

```

The PdfDictionary we get with the getPageN() method is called the *page dictionary*. It has plenty of information about a specific page in the PDF. We are only looking at one entry: the /MediaBox. This is only a proof of concept. If you want to write a more robust application, you should also look at the /CropBox and the /Rotate entry. Incidentally, I know that these entries don't exist in primes.pdf, so I am omitting them here.

The *media box* of a page is an array with four values that represent a rectangle defined by the coordinates of its lower-left and upper-right corner (usually, I refer to them as l1x, l1y, urx and ury).

In my code sample, I change the value of l1x by subtracting 36 user units. If you compare the page size of both PDFs, you'll see that we've added half an inch.

We also use these coordinates to draw a rectangle that covers the extra half inch. Now switch to the other watermark examples to find out how to add text or other content to each page.

## Why does PdfStamper create a file with 0 bytes?



When attempting to encrypt a pdf, I get a zero size file as a result. Any ideas what is causing this? If I don't try to encrypt, then I get the file okay.

```

File f = new File("C://secure_abc.pdf");
FileOutputStream out = new FileOutputStream(f);
PdfReader reader = new PdfReader("C://abc.pdf");
System.out.println("reader.getFileLength(): "+reader.getFileLength());
PdfStamper stamp = new PdfStamper(reader, out);
stamp.setEncryption(null, null,
    PdfWriter.ALLOW_PRINTING, PdfWriter.ENCRYPTION_AES_128);

```

Posted on StackOverflow on Jul 8, 2014 <sup>571</sup> by integratedsolns<sup>572</sup>

---

<sup>571</sup><http://stackoverflow.com/questions/24641319/encrypting-pdf-with-itext-produces-blank-file>

<sup>572</sup><http://stackoverflow.com/users/2505603/integratedsolns>

Please add the following line at the very end:

```
stamp.close();
```

You created a zero-length file when you do this:

```
FileOutputStream out = new FileOutputStream(f);
```

But no bytes are written to that output stream up until you close the PdfStamper instance.

Also make sure to use bouncy castle libraries, iText has dependencies on the lib for everything that concerns encryption or digital signing.

## How to add blank pages to an existing PDF in java?



I have a normal PDF file, I want to insert blank pages at the end of this PDF using iText, without changing the existing PDF contents.

Posted on StackOverflow on May 24, 2013 <sup>573</sup> by khAn<sup>574</sup>

Please try this short code snippet:

```
PdfReader reader = new PdfReader(src);
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
stamper.insertPage(reader.getNumberOfPages() + 1,
    reader.getPageSizeWithRotation(1));
stamper.close();
reader.close();
```

If `src` refers to a document with 10 pages, the code above will add an extra blank 11th page, using the same page size as the first page.

---

<sup>573</sup><http://stackoverflow.com/questions/16731168/how-to-add-blank-pages-in-exist-pdf-in-java>

<sup>574</sup><http://stackoverflow.com/users/1249325/khan>

## Adding blank pages while concatenating several PDFs in iText using PdfSmartCopy



I'm trying to concatenate a huge number of PDF files to create a print-ready file, so that I don't have to print them one by one. The print-ready file I want needs to be duplex, so for the documents I have which are only 1 page, I need to add a blank page for the "back" of the document, otherwise the following document is messed up.

Is there any way to add blank pages while concatenating files using PdfSmartCopy? I know PdfWriter can easily add blank pages, but it isn't made for merging a large number of files, which is why I'm not using it. I've read the answer to the question "*How to add blank pages in exist PDF in java?*" However, in this case I can't use PdfStamper either because I require a small file size and it seems PdfSmartCopy is the only viable option for that, unless I have missed something.

Posted on StackOverflow on Jan 30, 2014 <sup>575</sup>

In my answer to the question you refer to, I explained how to insert a blank page into an existing PDF using PdfStamper. This doesn't help you, because you're using PdfSmartCopy, which is an extension of PdfCopy. When using PdfCopy (or its subclass PdfSmartCopy), you can use the addPage( ) method like this:

```
copy.addPage(PageSize.A4, 0);
```

In this case, a page with size A4 will be added. In your case, you'd want to make sure that the blank page has the same dimensions as the other pages in your documents, so you'll do something like this:

```
copy.addPage(reader.getPageSize(1), reader.getPageRotation(1));
```

The Rectangle value will now correspond with the size of the first page in the reader; the int value will correspond with the rotation of the first page of your existing document.

---

<sup>575</sup><http://stackoverflow.com/questions/21454267/adding-blank-pages-while-concatenating-several-pdfs-in-itext-using-pdfsmar copy>

## How to position text relative to page?



How do I set the position of text so that it is centred vertically relative to its page size? I want to position it say for example x number of points from right and centred vertically. The text of course is rotated 90 degrees.

```
int n = reader.getNumberOfPages();
PdfImportedPage page;
PdfCopy.PageStamp stamp;
for (int j = 0; j < n; ) {
    ++j;
    page = writer.getImportedPage(reader, j);
    stamp = writer.createPageStamp(page);
    Rectangle crop = reader.getCropBox(1);
    // add overlay text
    Phrase phrase = new Phrase("Overlay Text");
    ColumnText.showTextAligned(stamp.getOverContent(), Element.ALIGN_CENTER,
        phrase, crop.getRight(72f), crop.getHeight() / 2, 90);
    stamp.alterContents();
    writer.addPage(page);
}
```

The code above gives me inconsistent positions of text. In some pages, only a portion of the “Overlay text” is visible. Please help, I don’t know how to properly use mediabox and cropbox and I’m new to iText.

Posted on StackOverflow on Jul 13, 2013 <sup>[576](#)</sup> by euler<sup>[577](#)</sup>

Regarding the inconsistent position: that should be fixed by adding the vertical offset:

```
crop.getRight(72f), crop.getBottom() + crop.getHeight() / 2
```

Do you see? You took the right border with a margin of 1 inch as x coordinate, but you forgot to take into account the y coordinate of the bottom of the page (it’s not always 0). Normally, this should fix the positioning problem.

---

<sup>576</sup><http://stackoverflow.com/questions/17439520/how-to-position-text-relative-to-page-using-itext>

<sup>577</sup><http://stackoverflow.com/users/1919069/euler>

## How can I crop the pages of an existing PDF document?



I have an existing document of which the pages are too big. How can I crop the pages?

Posted on StackOverflow on Apr 12, 2014 <sup>578</sup> by choudhary<sup>579</sup>

Take a look at the [RotatePages<sup>580</sup>](#) example from my book. In the `manipulatePdf()` method, I loop over the pages, I take the page dictionary, and I change the `/Rotate` key to rotate the page. That's not what you need, but the principle is similar.

You need to get the `/MediaBox` and `/CropBox` value from the page dictionary:

```
PdfArray mediabox = pageDict.getAsArray(PdfName.MEDIABOX);
PdfArray cropbox = pageDict.getAsArray(PdfName.CROPBOX);
```

In many cases, `cropbox` will be `null` in which case you can safely ignore it and use the `mediabox` value instead.

The `cropbox` value (or if `null`, `mediabox`) is an array with 4 values. These values represent two coordinates: one for the lower-left corner of the page, the other one for the upper-right corner of the page. If you want to crop a page, you need to change these coordinates and either replace the existing `cropbox` value (if one already exists) or add a new `cropbox` value (if there is none).

```
pageDict.put(PdfName.CROPBOX, new PdfArray(new float[]{llx, lly, urx, ury}));
```

Where `llx`, `lly` are the x and y coordinate of the lower-left corner and `urx`, `ury` are the x and y coordinate of the upper-right corner.

## How to rotate a page 90 degrees?



I am trying to use a PDF for stamping and need to rotate it 90 degrees to lay it on correctly?

Anyone know how to do this?

Posted on StackOverflow on Nov 19, 2014 <sup>581</sup> by Casey<sup>582</sup>

Please take a look at the [Rotate90Degrees<sup>583</sup>](#) example.

In this example, we use `PdfReader` to get an instance of the document, and we change the `/Rotate` value in every page dictionary (if there is no such entry, we add one with value 90):

---

<sup>578</sup><http://stackoverflow.com/questions/23029178/cropping-a-pdf-document-using-itext-returns-undesired-output>

<sup>579</sup><http://stackoverflow.com/users/2251538/choudhary>

<sup>580</sup><http://itextpdf.com/examples/iiapdf?id=232>

<sup>581</sup><http://stackoverflow.com/questions/27020542/rotating-pdf-90-degrees-using-itextsharp-in-c-sharp>

<sup>582</sup><http://stackoverflow.com/users/2762516/casey>

<sup>583</sup><http://itextpdf.com/sandbox/stamper/Rotate90Degrees>

```
PdfReader reader = new PdfReader(src);
int n = reader.getNumberOfPages();
PdfDictionary page;
PdfNumber rotate;
for (int p = 1; p <= n; p++) {
    page = reader.getPageN(p);
    rotate = page.getAsNumber(PdfName.ROTATE);
    if (rotate == null) {
        page.put(PdfName.ROTATE, new PdfNumber(90));
    }
    else {
        page.put(PdfName.ROTATE, new PdfNumber((rotate.intValue() + 90) % 360));
    }
}
```

Once this is done, we use a PdfStamper to persist the change:

```
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
stamper.close();
reader.close();
```

This is an iText example. If you want an iTextSharp example, you'll discover that it is very easy to port the Java to C# as the terminology is identical. You just need to change some lower cases into upper cases like this:

```
PdfDictionary page = reader.GetPageN(1);
page.Put(PdfName.ROTATE, new PdfNumber(90));
```

## How to crop out a part of PDF file?



I have to remove the top part of each page from a PDF. I managed to do this with CROPBOX, but the problem is that this will make the pages smaller by removing the top part.

Can someone help me to implement this so the page size remains the same.

This is the current code I'm using to crop the page.

```
PdfRectangle rect = new PdfRectangle(55, 0, 1000, 1000);
PdfDictionary pageDict;
for (int curentPage = 2;
    curentPage <= pdfReader.getNumberOfPages(); curentPage++) {
    pageDict = pdfReader.getPageN(curentPage);
    pageDict.put(PdfName.CROPBOX, rect);
}
```

Posted on StackOverflow on Nov 6, 2014 <sup>584</sup> by user2837742<sup>585</sup>

In your code sample, you are *cropping* the pages. This reduces the visible size of the page.

Based on your description, you don't want *cropping*. Instead you want *clipping*.

I've written an example that clips the content of all pages of a PDF by introducing a margin of 200 user units (that's quite a margin). The example is called [ClipPdf<sup>586</sup>](#) and you can see a clipped page here: [hero\\_clipped.pdf<sup>587</sup>](#) (the iText superhero has lost arms, feet and part of his head in the clipping process.)

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    int n = reader.getNumberOfPages();
    PdfDictionary page;
    PdfArray media;
    for (int p = 1; p <= n; p++) {
        page = reader.getPageN(p);
        media = page.getAsArray(PdfName.CROPBOX);
        if (media == null) {
```

<sup>584</sup><http://stackoverflow.com/questions/26773942/itext-crop-out-a-part-of-pdf-file>

<sup>585</sup><http://stackoverflow.com/users/2837742/user2837742>

<sup>586</sup><http://itextpdf.com/sandbox/stamper/ClipPdf>

<sup>587</sup>[http://itextpdf.com/sites/default/files/hero\\_clipped.pdf](http://itextpdf.com/sites/default/files/hero_clipped.pdf)

```
        media = page.getAsArray(PdfName.MEDIABOX);
    }
    float l1x = media.getAsNumber(0).floatValue() + 200;
    float l1y = media.getAsNumber(1).floatValue() + 200;
    float w = media.getAsNumber(2).floatValue()
        - media.getAsNumber(0).floatValue() - 400;
    float h = media.getAsNumber(3).floatValue()
        - media.getAsNumber(1).floatValue() - 400;
    String command = String.format(
        "\nq %.2f %.2f %.2f %.2f re W n\nq\n",
        l1x, l1y, w, h);
    stamper.getUnderContent(p).setLiteral(command);
    stamper.getOverContent(p).setLiteral("\nQ\nQ\n");
}
stamper.close();
reader.close();
}
```

Obviously, you need to study this code before using it. Once you understand this code, you'll know that this code will only work for pages that aren't rotated. If you understand the code well you should have no problem adapting the example for rotated pages.

### Some pointers

The `re` operator constructs a rectangle. It takes four parameters (the values preceding the operator) that define a rectangle: the x coordinate of the lower-left corner, the y coordinate of the lower-left corner, the width and the height.

The `W` operator sets the clipping path. We have just drawn a rectangle; this rectangle will be used to clip the content that follows.

The `n` operator starts a new path. It discards the paths we've constructed so far. In this case, it prevents that the rectangle we have drawn (and that we use as clipping path) is actually drawn.

The `q` and `Q` operators save and restore the graphics state stack, but that's rather obvious.

## How to rotate and scale pages in an existing PDF?



I'm using iTextSharp to import specific pages from a PDF, possibly rotate, resize or otherwise alter that page, and exporting it into a new PDF. To this end I'm using the `PdfWriter` class. The problem I'm running into is that when using the writer class, it doesn't appear to be including the visible annotations from the source page (for instance: the comment are missing).

I am looking for one of two solutions:

- Preferably I'd like to continue using the `PdfWriter` class, but need it to copy/duplicate any visible annotations from the source pages and include them in the output, or
- I need some example code of using the `PdfCopy` class to resize a page. There is a `PdfCopy.SetPageSize()` function (which doesn't work, which I suspect means I'm doing it wrong), but I have basically no idea how to properly scale the source page when it needs to be resized.

Posted on StackOverflow on Feb 19, 2014<sup>588</sup> by Finch042<sup>589</sup>

I've created a small `ScaleRotate`<sup>590</sup> example that looks like this.

```
PdfReader reader = new PdfReader(src);
int n = reader.getNumberOfPages();
PdfDictionary page;
for (int p = 1; p <= n; p++) {
    page = reader.getPageN(p);
    if (page.getAsNumber(PdfName.USERUNIT) == null)
        page.put(PdfName.USERUNIT, new PdfNumber(2.5f));
    page.remove(PdfName.ROTATE);
}
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
stamper.close();
reader.close();
```

Given an original PDF `pages.pdf`<sup>591</sup> with rotated pages, cropped pages and annotations, we scale and rotate some pages, resulting in `pages_altered.pdf`<sup>592</sup>.

<sup>588</sup><http://stackoverflow.com/questions/21871027/rotating-in-itextsharp-while-preserving-comment-location-orientation>

<sup>589</sup><http://stackoverflow.com/users/1943542/finch042>

<sup>590</sup><http://itextpdf.com/sandbox/stamper/ScaleRotate>

<sup>591</sup><http://itextpdf.com/sites/default/files/pages.pdf>

<sup>592</sup>[http://itextpdf.com/sites/default/files/pages\\_altered.pdf](http://itextpdf.com/sites/default/files/pages_altered.pdf)

We introduce a **UserUnit** of 2.5 for all pages that aren't scaled yet. If you change the **UserUnit** to 0.5, you'll see that it won't have any effect in Adobe Reader. The ISO standard for PDF says that the range that can be used for the user unit is implementation-independent. Version 1.7 of the PDF specification originally written by Adobe says: "Acrobat 7.0 supports a maximum UserUnit value of 75,000." Nothing is said about the minimum value, but experience tells us that the minimum value supported by Adobe Reader is 1, meaning you can't scale down.

As for the rotation, you can change the rotation of a page by changing the **/Rotate** key in the page dictionary. In the example, I removed the key, changing all pages shown in landscape (of which the value for **/Rotate** is 90) into portrait (the default value for **/Rotate** is 0). You'll notice that this doesn't have any effect on page 4. Page 4 isn't rotated. It looks like a page in landscape because the dimensions of the page are created in such a way that the width is greater than the height.

Summarized: it's a piece of cake to rotate pages in an existing PDF, so is scaling the pages up to a bigger size. If you want to downscale pages, you can only use **PdfWriter** (which throws away all annotations) and you need to copy the annotations separately *after* transforming all the **/Rect** values of these annotations. This is not a trivial task. It took one of our customers several weeks to achieve this correctly. Be prepared to spend an equal amount of time if that's what you want.

**DISCLAIMER:** the **UserUnit** value isn't supported by all viewers. Implementations may vary depending on the viewer that is used. The feature was introduced in PDF 1.6, meaning that the functionality won't work in any viewer supporting only older PDF versions.

## How to shrink pages in an existing PDF?



I am using **PdfWriter**, **PdfImportedPage** and the **addTemplate()** method to shrink pages. However, when I do so, I lose the rotation of the pages and I lose all interactive features. Is there another way I can shrink pages?

Posted on StackOverflow on Aug 18, 2014 <sup>593</sup> by Butani Vijay<sup>594</sup>

Scaling a PDF to make pages larger is easy. This is shown in the [ScaleRotate<sup>595</sup>](#) example. It's only a matter of changing the default version of the user unit. By default, this unit is 1, meaning that 1 user unit equals to 1 point. You can increase this value to 75,000. Larger values of the user unit, will result in larger pages.

Unfortunately, the user unit can never be smaller than 1, so you can't use this technique to shrink a page. If you want to shrink a page, you need to introduce a transformation (resulting in a new CTM).

This is shown in the [ShrinkPdf<sup>596</sup>](#) example. In this example, I take a PDF named `hero.pdf`<sup>597</sup> that

<sup>593</sup><http://stackoverflow.com/questions/25356302/shrink-pdf-pages-with-rotation-using-rectangle-in-existing-pdf>

<sup>594</sup><http://stackoverflow.com/users/2178702/butani-vijay>

<sup>595</sup><http://itextpdf.com/sandbox/stamper/ScaleRotate>

<sup>596</sup><http://itextpdf.com/sandbox/stamper/ShrinkPdf>

<sup>597</sup>[http://itextpdf.com/sites/default/files/hero\\_0.pdf](http://itextpdf.com/sites/default/files/hero_0.pdf)

measures 8.26 by 11.69 inch, and we shrink it by 50%, resulting in a PDF named `hero_shrink.pdf`<sup>598</sup> that measures 4.13 by 5.85 inch.

To achieve this, we need a dirty hack:

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    int n = reader.getNumberOfPages();
    PdfDictionary page;
    PdfArray crop;
    PdfArray media;
    for (int p = 1; p <= n; p++) {
        page = reader.getPageN(p);
        media = page.getAsArray(PdfName.CROPBOX);
        if (media == null) {
            media = page.getAsArray(PdfName.MEDIABOX);
        }
        crop = new PdfArray();
        crop.add(new PdfNumber(0));
        crop.add(new PdfNumber(0));
        crop.add(new PdfNumber(media.getAsNumber(2).floatValue() / 2));
        crop.add(new PdfNumber(media.getAsNumber(3).floatValue() / 2));
        page.put(PdfName.MEDIABOX, crop);
        page.put(PdfName.CROPBOX, crop);
        stamper.getUnderContent(p).setLiteral("\nq 0.5 0 0 0.5 0 0 cm\nq\n");
        stamper.getOverContent(p).setLiteral("\nQ\nQ\n");
    }
    stamper.close();
    reader.close();
}
```

We loop over every page and we take the crop box of each page. If there is no crop box, we take the media box. These boxes are stored as arrays of 4 numbers. In this example, I assume that the first two numbers are zero and I divide the next two values by 2 to shrink the page to 50% (if the first two values are not zero, you'll need a more elaborate formula).

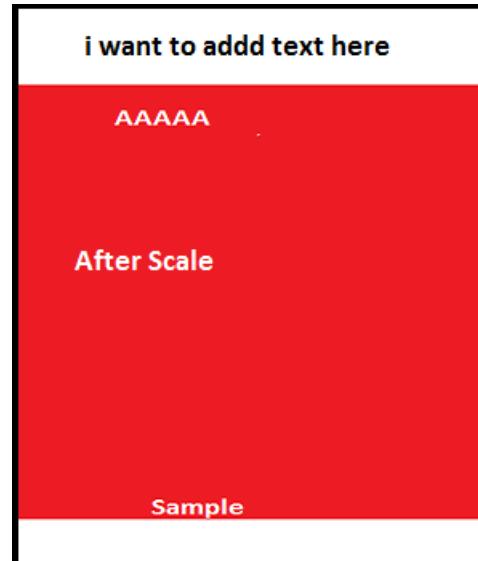
Once I have the new array, I change the media box and the crop box to this array, and I introduce a CTM that scales all content down to 50%. I need to use the `setLiteral()` method to fool iText.

---

<sup>598</sup> [http://itextpdf.com/sites/default/files/hero\\_shrink.pdf](http://itextpdf.com/sites/default/files/hero_shrink.pdf)



Thanks Bruno Lowagie. Above code scale whole page actually I want to make room for adding header and footer means by scaling(shrink) page as shown in this screenshot:



Screen shot

I have made a second example, named `ShrinkPdf2`<sup>599</sup> that allows you to introduce a different scaling percentage:

```

PdfReader reader = new PdfReader(src);
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
int n = reader.getNumberOfPages();
float percentage = 0.8f;
for (int p = 1; p <= n; p++) {
    float offsetX = (reader.getPageSize(p).getWidth() * (1 - percentage)) / 2;
    float offsetY = (reader.getPageSize(p).getHeight() * (1 - percentage)) / 2;
    stamper.getUnderContent(p).setLiteral(
        String.format("\nq %s 0 0 %s %s cm\nq\n",
            percentage, percentage, offsetX, offsetY));
    stamper.getOverContent(p).setLiteral("\nQ\nQ\n");
}
stamper.close();
reader.close();

```

In this example, I don't change the page size (no changes to the media or crop box), I only shrink the

<sup>599</sup><http://itextpdf.com/sandbox/stamper/ShrinkPdf2>

content (in this case to 80%) and I center the shrunken content on the page, leaving bigger margins to the top, bottom, left and right.

As you can see, this is only a matter of applying the correct Math. This second example is even more simple than the first, so I introduced some extra complexity: now you can easily change the percentage (in the first example I hardcoded 50%. in this case I introduced the percentage variable and I defined it as 80%).

Note that I apply the scaling in the X direction as well as in the Y direction to preserve the aspect ratio. Looking at your screen shots, it looks like you only want to shrink the content in the Y direction. For instance like this:

```
String.format("\nq 1 0 0 %s 0 %s cm\nq\n", percentage, offsetY)
```

Feel free to adapt the code to meet your need, but... the result will be ugly: all text will look funny and if you apply it to photos of people standing up vertically, you'll make them look fat.

## How to fix the orientation of a PDF page in order to scale it?



I need to scale down the pages of an existing PDF document. This works fine if the pages aren't rotated, but I don't succeed in rotating pages correctly, for instance if they have a rotation of 270 degrees:

```
for(int i=1; i<=reader.getNumberOfPages(); i++){
    int rotation = reader.getPageRotation(i);
    float pageWidth = reader.getPageSizeWithRotation(i).getWidth();
    float pageHeight = reader.getPageSizeWithRotation(i).getHeight();
    doc.newPage();
    PdfImportedPage page = writer.getImportedPage(reader, i);
    if (rotation == 0) {
        cb.addTemplate(page, 1f, 0, 0, scaleHeight, x, y);
    } else if (rotation == 90) {
        cb.addTemplate(page, 0, -1f, 1f, 0, 0, pageHeight);
    } else if (rotation == 180) {
        cb.addTemplate(page, 1f, 0, 0, -1f, pageWidth, pageHeight);
    } else if (rotation == 270) {
        cb.addTemplate(page, 0, -1f, 1f, 0, 0, pageHeight);
    }
}
```

I have two questions:

1. How do I properly rotate the page contents?
2. How can I scale the rotated contents correctly?

Posted on StackOverflow on Mar 19, 2014 <sup>600</sup> by user3854377<sup>601</sup>

Please take a look at the [ScaleDown<sup>602</sup>](#) example that can be used to scale down pages, keeping the original orientation.

This example uses a page event named it ScaleEvent:

---

<sup>600</sup><http://stackoverflow.com/questions/29152313/fix-the-orientation-of-a-pdf-in-order-to-scale-it>

<sup>601</sup><http://stackoverflow.com/users/3854377/user3854377>

<sup>602</sup><http://itextpdf.com/sandbox/events/ScaleDown>

```
public class ScaleEvent extends PdfPageEventHelper {  
    protected float scale = 1;  
    protected PdfDictionary pageDict;  
    public ScaleEvent(float scale) {  
        this.scale = scale;  
    }  
    public void setPageDict(PdfDictionary pageDict) {  
        this.pageDict = pageDict;  
    }  
    @Override  
    public void onStartPage(PdfWriter writer, Document document) {  
        writer.addPageDictEntry(PdfName.ROTATE, pageDict.getAsNumber(PdfName.ROTATE));  
        writer.addPageDictEntry(PdfName.MEDIABOX, scaleDown(pageDict.getAsArray(PdfName.MEDIABOX), scale));  
        writer.addPageDictEntry(PdfName.CROPBOX, scaleDown(pageDict.getAsArray(PdfName.CROPBOX), scale));  
    }  
}
```

When you create the event, you pass a value `scale` that will define the scale factor. I apply the scale to the width and the height, feel free to adapt it if you only want to scale the height or the width only.

The information about page size and rotation is stored in the page dictionary. Obviously the `ScaleEvent` needs the values of the original document, and that's why we'll pass a `pageDict` for every page we copy.

Every time a new page is created, we will copy/replace:

- the `/Rotate` value. Remove this line if you want to remove the rotation,
- the `/MediaBox` value. This defines the full size of the page.
- the `/CropBox` value. This defines the visible size of the page.

As we want to scale the page, we use the following `scaleDown()` method:

```

public PdfArray scaleDown(PdfArray original, float scale) {
    if (original == null)
        return null;
    float width = original.getAsNumber(2).floatValue()
        - original.getAsNumber(0).floatValue();
    float height = original.getAsNumber(3).floatValue()
        - original.getAsNumber(1).floatValue();
    return new PdfRectangle(width * scale, height * scale);
}

```

Suppose that I want to reduce the page width and height to 50% of the original width and height, then I create the event like this:

```

PdfReader reader = new PdfReader(src);
float scale = 0.5f;
ScaleEvent event = new ScaleEvent(scale);
event.setPageDict(reader.getPageN(1));

```

I can define a Document with any page size I want as the size will be changed in the ScaleEvent anyway. For this to work I need to declare the event to the PdfWriter instance before opening the document:

```

Document document = new Document();
PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream(dest));
writer.setPageEvent(event);
document.open();

```

Now it's only a matter of looping over the pages:

```

int n = reader.getNumberOfPages();
Image page;
for (int p = 1; p <= n; p++) {
    page = Image.getInstance(writer.getImportedPage(reader, p));
    page.setAbsolutePosition(0, 0);
    page.scalePercent(scale * 100);
    document.add(page);
    if (p < n) {
        event.setPageDict(reader.getPageN(p + 1));
    }
    document.newPage();
}
document.close();

```

I am wrapping the imported page inside an `Image` because I personally think that the methods available for the `Image` class are easier to use than defining the parameters of the `addTemplate()` method. If you want to use `addTemplate()` instead of `Image`, feel free to do so; the result will be identical (contrary to what you wrote in a comment, wrapping a page inside an image will not cause any loss of “resolution” as all the text remains available as vector data).

Note that I update the `pageDict` for every new page.

This code takes the file [orientations.pdf<sup>603</sup>](#) measuring 8.26 by 11.69 in and transforms it into the file [scaled\\_down.pdf<sup>604</sup>](#) measuring 4.13 by 5.85 in.

## How to reuse a page from one PDF document into another PDF document?



I'm trying to add an image that I have on the file system as a PDF into a PDF that I'm creating on the fly. I tried to use the `Image` class, but it seems that it does not work with PDFs (only JPEG, PNG or GIF). How can I create an element from an existing PDF, so that I can place it in my new PDF?

Posted on StackOverflow on [Apr 11, 2014<sup>605</sup>](#) by [Andres Olarte<sup>606</sup>](#)

The most basic solution is to create a `PdfReader` instance and to import the page into the `PdfWriter` instance, from which point on you can use the `PdfImportedPage` instance, either directly, or wrapped inside an `Image` object:

```
PdfReader reader = new PdfReader(existingPdf);
PdfImportedPage page = writer.getImportedPage(reader, i);
PdfImage img = Image.getInstance(page);
reader.close();
```

<sup>603</sup><http://itextpdf.com/sites/default/files/orientations.pdf>

<sup>604</sup>[http://itextpdf.com/sites/default/files/scaled\\_down.pdf](http://itextpdf.com/sites/default/files/scaled_down.pdf)

<sup>605</sup><http://stackoverflow.com/questions/23022471/creating-a-pdf-image-in-itext>

<sup>606</sup><http://stackoverflow.com/users/1030209/andres-olarte>

## How to merge documents correctly?

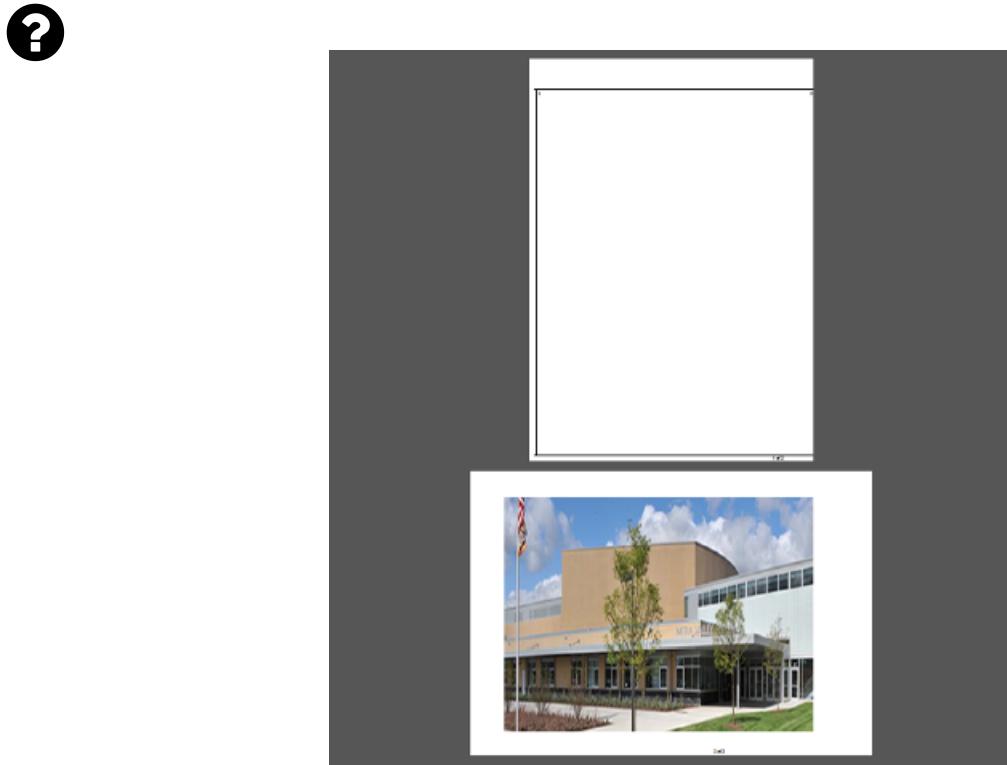


I would like to add a link to an existing pdf that jumps to a coordinate on another page.

I have the following problem when printing the PDF file after merge, the PDF documents get cut off. Sometimes this happens because the documents aren't 8.5 x 11 whereas the page size might be 11 x 17.

Is there some way to detect the page size and then use that same page size for those documents? Or, if not, is it possible to have it fit to page? This is my code:

```
Document document = new Document();
PdfWriter writer = PdfWriter.getInstance(document, outputStream);
document.open();
BaseFont bf = BaseFont.createFont(BaseFont.HELVETICA,
    BaseFont.CP1252, BaseFont.NOT_EMBEDDED);
PdfContentByte cb = writer.getDirectContent();
PdfImportedPage page;
int currentPageNumber = 0;
int pageOfCurrentReaderPDF = 0;
Iterator<PdfReader> iteratorPDFReader = readers.iterator();
while (iteratorPDFReader.hasNext()) {
    PdfReader pdfReader = iteratorPDFReader.next();
    while (pageOfCurrentReaderPDF < pdfReader.getNumberOfPages()) {
        Rectangle r = pdfReader.getPageSize(
            pdfReader.getPageN(pageOfCurrentReaderPDF + 1));
        if(r.getWidth()==792.0 && r.getHeight()==612.0)
            document.setPageSize.PageSize.A4.rotate());
        else
            document.setPageSize.PageSize.A4);
        document.newPage();
        pageOfCurrentReaderPDF++;
        currentPageNumber++;
        page = writer.getImportedPage(pdfReader, pageOfCurrentReaderPDF);
        cb.addTemplate(page, 0, 0);
        cb.beginText();
        cb.setFontAndSize(bf, 9);
        cb.showTextAligned(PdfContentByte.ALIGN_CENTER, ""
            + currentPageNumber + " of " + totalPages, 520, 5, 0);
        cb.endText();
    }
    pageOfCurrentReaderPDF = 0;
}
document.close();
```



Screen shot

Posted on StackOverflow on Feb 12, 2014 <sup>607</sup> by Sumit Vaidya<sup>608</sup>

Using `PdfWriter` to merge documents is a bad idea. This has been explained on StackOverflow many times!

Merging documents is done using `PdfCopy` (or `PdfSmartCopy`).

If you need an example, see for instance the [FillFlattenMerge2](#)<sup>609</sup> example:

```
Document document = new Document();
PdfCopy copy = new PdfSmartCopy(document, new FileOutputStream(dest));
document.open();
PdfReader reader;
String line = br.readLine();
// loop over readers
    // add the PDF to PdfCopy
    reader = new PdfReader(baos.toByteArray());
    copy.addDocument(reader);
```

<sup>607</sup><http://stackoverflow.com/questions/21731439/pdf-page-cutting-through-itext-api>

<sup>608</sup><http://stackoverflow.com/users/2853641/sumit-vaidya>

<sup>609</sup><http://itextpdf.com/sandbox/acroforms/reporting/FillFlattenMerge2>

```

    reader.close();
// end loop
document.close();

```

In your case, you also need to add page numbers, you can do this in a second go, as is done in the [StampPageXofY<sup>610</sup>](#) example:

```

PdfReader reader = new PdfReader(src);
int n = reader.getNumberOfPages();
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
PdfContentByte pagecontent;
for (int i = 0; i < n; ) {
    pagecontent = stamper.getOverContent(++i);
    ColumnText.showTextAligned(pagecontent, Element.ALIGN_RIGHT,
        new Phrase(String.format("page %s of %s", i, n)), 559, 806, 0);
}
stamper.close();
reader.close();

```

Or you can add them while merging, as is done in the [MergeWithToc<sup>611</sup>](#) example.

```

Document document = new Document();
PdfCopy copy = new PdfCopy(document, new FileOutputStream(filename));
PageStamp stamp;
document.open();
int n;
int pageNo = 0;
PdfImportedPage page;
Chunk chunk;
for (Map.Entry<String, PdfReader> entry : filesToMerge.entrySet()) {
    n = entry.getValue().getNumberOfPages();
    for (int i = 0; i < n; ) {
        pageNo++;
        page = copy.getImportedPage(entry.getValue(), ++i);
        stamp = copy.createPageStamp(page);
        chunk = new Chunk(String.format("Page %d", pageNo));
        if (i == 1)
            chunk.setLocalDestination("p" + pageNo);
        ColumnText.showTextAligned(stamp.getUnderContent(),

```

---

<sup>610</sup><http://itextpdf.com/sandbox/stamper/StampPageXofY>

<sup>611</sup><http://itextpdf.com/sandbox/merge/MergeWithToc>

```

        Element.ALIGN_RIGHT, new Phrase(chunk),
        559, 810, 0);
stamp.alterContents();
copy.addPage(page);
}
}
document.close();
for (PdfReader r : filesToMerge.values()) {
    r.close();
}
reader.close();

```

I strongly advise against using `PdfWriter` to merge documents! As documented, you are throwing away all annotations by adding `PdfImportedPage` instances to a document using `addTemplate()`. This is typically not what you want. You're only making it harder on yourself if you insist on using that class. I don't understand why so many people use the wrong approach to merge documents. I blame the unofficial documentation for the popularity of this wrong approach.

## How to add a cover page to an existing PDF document?



I need to add an existing cover page (a single-page PDF) to another existing PDF document.  
How can this be done?

Posted on StackOverflow on Apr 10, 2015 <sup>[612](#)</sup> by Doug LN<sup>[613](#)</sup>

Suppose that we have a document named `pages.pdf`<sup>[614](#)</sup> and that we want to add the cover `hero.pdf`<sup>[615](#)</sup> as the cover of this document.

**Approach 1:** Use `PdfCopy`

Take a look at the `AddCover1`<sup>[616](#)</sup> example:

---

<sup>612</sup><http://stackoverflow.com/questions/29563942/how-to-add-a-cover-pdf-in-a-existing-itext-document>

<sup>613</sup><http://stackoverflow.com/users/4103270/doug-ln>

<sup>614</sup><http://itextpdf.com/sites/default/files/pages.pdf>

<sup>615</sup>[http://itextpdf.com/sites/default/files/hero\\_0.pdf](http://itextpdf.com/sites/default/files/hero_0.pdf)

<sup>616</sup><http://itextpdf.com/sandbox/merge/AddCover1>

```

PdfReader cover = new PdfReader("hero.pdf");
PdfReader reader = new PdfReader("pages.pdf");
Document document = new Document();
PdfCopy copy = new PdfCopy(document, new FileOutputStream("pages_with_cover.pdf")\ 
));
document.open();
copy.addDocument(cover);
copy.addDocument(reader);
document.close();
cover.close();
reader.close();

```

The result is a document where you first have the cover and then the rest of the document: [pages\\_-with\\_cover.pdf](#)<sup>617</sup>

### Approach 2: Use PdfStamper

Take a look at the [AddCover2](#)<sup>618</sup> example:

```

PdfReader cover = new PdfReader("hero.pdf");
PdfReader reader = new PdfReader("pages.pdf");
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream("cover_with_pag\ 
es.pdf"));
stamper.insertPage(1, cover.getPageSizeWithRotation(1));
PdfContentByte page1 = stamper.getOverContent(1);
PdfImportedPage page = stamper.getImportedPage(cover, 1);
page1.addTemplate(page, 0, 0);
stamper.close();
cover.close();
reader.close();

```

In this case, we take the original document `pages.pdf` and we insert a new page 1 with the same page size as the cover. We then get this `page1` and we add the first page of `hero.pdf` to this first page. The result is [cover\\_with\\_pages.pdf](#)<sup>619</sup>

### What is the difference between the two approaches?

With `PdfCopy`, you may lose some of the properties that are defined at the document level (e.g. the page layout setting), but you keep the interactive features of both files. You may need to set some parameters in case you want to preserve tagging, form fields, etc... but for simple PDFs, you don't need all of that.

---

<sup>617</sup>[http://itextpdf.com/sites/default/files/pages\\_with\\_cover.pdf](http://itextpdf.com/sites/default/files/pages_with_cover.pdf)

<sup>618</sup><http://itextpdf.com/sandbox/merge/AddCover2>

<sup>619</sup>[http://itextpdf.com/sites/default/files/cover\\_with\\_pages.pdf](http://itextpdf.com/sites/default/files/cover_with_pages.pdf)

With `PdfStamper`, you preserve the properties that are defined at the document level of `pages.pdf`, but you lose all the interactive features of the cover page (if any). You may need to tweak the example if you want to define the cover as an artifact and if the original cover page has odd page boundaries, but it would lead us too far to discuss this in this simple answer.

## Why does the function to concatenate / merge PDFs cause issues in some cases?



I'm using the following code to merge PDFs together using iText:

```
public static void concatenatePdf(List<File> listOfPdfFiles, File outputFile)
    throws DocumentException, IOException {
    Document document = new Document();
    FileOutputStream outputStream = new FileOutputStream(outputFile);
    PdfWriter writer = PdfWriter.getInstance(document, outputStream);
    document.open();
    PdfContentByte cb = writer.getDirectContent();
    for (File inFile : listOfPdfFiles) {
        PdfReader reader = new PdfReader(inFile.getAbsolutePath());
        for (int i = 1; i <= reader.getNumberOfPages(); i++) {
            document.newPage();
            PdfImportedPage page = writer.getImportedPage(reader, i);
            cb.addTemplate(page, 0, 0);
        }
    }
    document.close();
}
```

This usually works great! But once and a while, it's rotating some of the pages by 90 degrees? Anyone ever have this happen?

Posted on StackOverflow on Apr 14, 2014 <sup>620</sup> by Nicholas DiPiazza<sup>621</sup>

There are errors once in a while because you are using the wrong method to concatenate documents. You should not use `PdfWriter` to concatenate (or merge) PDF documents. That is wrong because:

- You completely ignore the page size of the pages in the original document (you assume they are all of size A4),

---

<sup>620</sup><http://stackoverflow.com/questions/23062345/function-that-can-use-itext-to-concatenate-merge-pdfs-together-causing-some>

<sup>621</sup><http://stackoverflow.com/users/1174024/nicholas-dipiazza>

- You ignore page boundaries such as the crop box (if present),
- You ignore the rotation value stored in the page dictionary,
- You throw away all interactivity that is present in the original document, and so on.

Concatenating PDFs is done using PdfCopy, see for instance:

```
Document document = new Document();
PdfCopy copy = new PdfSmartCopy(document, new FileOutputStream(dest));
document.open();
PdfReader reader;
String line = br.readLine();
// loop over readers
    // add the PDF to PdfCopy
    reader = new PdfReader(baos.toByteArray());
    copy.addDocument(reader);
    reader.close();
// end loop
document.close();
```

If you are merging documents that contain fields, you need to add the following line:

```
copy.SetMergeFields();
```

## How to merge PDFs from ByteArrayOutputStream?



I have two PDF files, each one in a `ByteArrayOutputStream`. I want to merge the two PDFs, and I want to use iText, but I don't understand how to do this because it seems that I need an `InputStream` (and I only have `ByteArrayOutputStreams`).

Posted on StackOverflow on Jan 13, 2014 <sup>622</sup> by 3vi<sup>623</sup>

The `ByteArrayOutputStream` object has a `toByteArray()`<sup>624</sup> method that returns a `byte[]`. The `PdfReader` class has a `constructor`<sup>625</sup> that takes a `byte[]` as parameter. Once you have a `PdfReader` instance of both files, you can use these instances with `PdfCopy` or `PdfSmartCopy` to merge the files. Use the [Concatenate](#)<sup>626</sup> example for inspiration.

<sup>622</sup><http://stackoverflow.com/questions/21093981/merge-pdf-from-outputstream>

<sup>623</sup><http://stackoverflow.com/users/1310687/3vi>

<sup>624</sup><http://docs.oracle.com/javase/6/docs/api/java/io/ByteArrayOutputStream.html#toByteArray%28%29>

<sup>625</sup><http://api.itextpdf.com/itext/com/itextpdf/text/pdf/PdfReader.html#PdfReader%28byte%5B%5D%29>

<sup>626</sup><http://itextpdf.com/examples/iiapdf?id=123>

## How to merge PDFs and add bookmarks?



I am merging multiple PDFs together into one PDF and I need to build bookmarks for the final PDF. For example, I have three PDFs: doc1.pdf, doc2.pdf and doc3.pdf, doc1 and doc2 belong to Group1, doc3 belongs to Group2. I need to merge them and have to build nested bookmarks for the resulting PDFs like this:

```
Group1
  doc1
  doc2
Group2
  doc3
```

Posted on StackOverflow on May 15, 2014<sup>627</sup> by user3642526<sup>628</sup>

I've made a [MergeWithOutlines](#)<sup>629</sup> example that concatenates three existing PDFs using PdfCopy (I assume that you already know that part).

While doing so, I create an outlines object like this:

```
ArrayList<HashMap<String, Object>> outlines = new ArrayList<HashMap<String, Object>>();
```

and I add elements to this outlines object:

```
HashMap<String, Object> helloworld = new HashMap<String, Object>();
helloworld.put("Title", "Hello World");
helloworld.put("Action", "GoTo");
helloworld.put("Page", String.format("%d Fit", page));
outlines.add(helloworld);
```

When I want some hierarchy, I introduce kids:

---

<sup>627</sup><http://stackoverflow.com/questions/23688308/merge-pdfs-and-add-bookmark-with-itext-in-java>

<sup>628</sup><http://stackoverflow.com/users/3642526/user3642526>

<sup>629</sup><http://itextpdf.com/sandbox/merge/MergeWithOutlines>

```
ArrayList<HashMap<String, Object>> kids = new ArrayList<HashMap<String, Object>>\n()\n;\nHashMap<String, Object> link1 = new HashMap<String, Object>();\nlink1.put("Title", "link1");\nlink1.put("Action", "GoTo");\nlink1.put("Page", String.format("%d Fit", page));\nkids.add(link1);\nhelloworld.put("Kids", kids);
```

If you want an entry without a link, remove the lines that put an Action and a Page.

Once you're finished, add the outlines to the copy object:

```
copy.setOutlines(outlines);
```

Look at the resulting PDF<sup>630</sup> and you'll see the outlines in the bookmarks panel.

---

<sup>630</sup>[http://itextpdf.com/sites/default/files/merge\\_with\\_outlines.pdf](http://itextpdf.com/sites/default/files/merge_with_outlines.pdf)

## How to create a TOC when merging documents?



I need to create a TOC (not bookmarks) at the beginning of this document with clickable links to the first pages of each of the source PDFs.

```
Document document = new Document();
PdfCopy copy = new PdfCopy(document, outputStream);
PdfCopy.PasteStamp stamp;
document.open();
PdfReader reader;
List<InputStream> pdfs = streamOfPDFFiles;
List<PdfReader> readers = new ArrayList<PdfReader>();
Iterator<InputStream> iteratorPDFs = pdfs.iterator();
for (; iteratorPDFs.hasNext(); pdfCounter++) {
    InputStream pdf = iteratorPDFs.next();
    reader = new PdfReader(pdf);
    readers.add(reader);
    pdf.close();
}
int currentPageNumber = 0;
Iterator<PdfReader> readerIterator = readers.iterator();
PdfImportedPage page;
int count = 1;
while (readerIterator.hasNext()) {
    reader = readerIterator.next();
    count++;
    int number_of_pages = reader.getNumberOfPages();
    for (int pageNum = 0; pageNum < number_of_pages;) {
        currentPageNumber++;
        page = copy.getImportedPage(reader, ++pageNum);
        ColumnText.showTextAligned(stamp.getUnderContent(),
            Element.ALIGN_RIGHT, new Phrase(
                String.format("%d", currentPageNumber),
                new Font(FontFamily.TIMES_ROMAN, 3)), 50, 50, 0);
        stamp.alterContents();
        copy.addPage(page);
    }
}
document.close();
```

Posted on StackOverflow on Feb 4, 2014<sup>631</sup> by Butani Vijay<sup>632</sup>

---

<sup>631</sup><http://stackoverflow.com/questions/21548552/create-index-filetoc-for-merged-pdf-using-itext-library-in-java>

<sup>632</sup><http://stackoverflow.com/users/2178702/butani-vijay>

You're asking for something that should be trivial, but that isn't. Please take a look at the [MergeWithToc<sup>633</sup>](#) example. You'll see that your code to merge PDFs is correct, but in my example, I added one extra feature:

```
chunk = new Chunk(String.format("Page %d", pageNo));
if (i == 1)
    chunk.setLocalDestination("p" + pageNo);
ColumnText.showTextAligned(stamp.getUnderContent(),
    Element.ALIGN_RIGHT, new Phrase(chunk), 559, 810, 0);
```

For every first page, I define a *named destination* as a local destination. We use p followed by the page number as its name.

We'll use these named destinations in an extra page that will serve as a TOC:

```
PdfReader reader = new PdfReader(SRC3);
page = copy.getImportedPage(reader, 1);
stamp = copy.createPageStamp(page);
Paragraph p;
PdfAction action;
PdfAnnotation link;
float y = 770;
ColumnText ct = new ColumnText(stamp.getOverContent());
ct.setSimpleColumn(36, 36, 559, y);
for (Map.Entry<Integer, String> entry : toc.entrySet()) {
    p = new Paragraph(entry.getValue());
    p.add(new Chunk(new DottedLineSeparator()));
    p.add(String.valueOf(entry.getKey()));
    ct.addElement(p);
    ct.go();
    action = PdfAction.gotoLocalPage("p" + entry.getKey(), false);
    link = new PdfAnnotation(copy, 36, ct.getYLine(), 559, y, action);
    stamp.addAnnotation(link);
    y = ct.getYLine();
}
ct.go();
stamp.alterContents();
copy.addPage(page);
```

In my example, I assume that the TOC fits on a single page. You'll have to keep track of the y value and create a new page if its value is lower than the bottom margin.

---

<sup>633</sup><http://itextpdf.com/sandbox/merge/MergeWithToc>

If you want the TOC to be the first page, you need to reorder the pages in a second go. This is shown in the [MergeWithToc2<sup>634</sup>](#) example:

```
reader = new PdfReader(baos.toByteArray());
n = reader.getNumberOfPages();
reader.selectPages(String.format("%d, 1-%d", n, n-1));
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(filename));
stamper.close();
```

## How to reorder pages in an existing PDF file?



I am using the iText pdf library. Does any one know how I can move pages in an existing PDF? Actually I want to move a couple of the last pages to the start of the document.

It is done more or less using the following code, but I don't understand how it works.

```
reader = new PdfReader(baos.toByteArray());
n = reader.getNumberOfPages();
reader.selectPages(String.format("%d, 1-%d", n, n-1));
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(filename));
stamper.close();
```

Can any one explain this code in detail?

Posted on StackOverflow on [Feb 6, 2014](#) <sup>635</sup> by [Butani Vijay](#)<sup>636</sup>

The `selectPages()` method is explained in chapter 6 of “iText in Action - Second Edition” (see page 164). In the context of code snippet 6.3 and 6.11, it is used to reduce the number of pages being read by `PdfReader` for consumption by `PdfStamper` or `PdfCopy`. However, it can also be used to reorder pages. First allow me to explain the syntax.

There are different flavors of the `selectPages()` method:

You can pass a `List<Integer>` containing all the page numbers you want to keep. This list can consist of increasing page numbers, 1, 2, 3, 4,... If you change the order, for instance: 1, 3, 2, 4,... `PdfReader` will serve the pages in that changed order.

You can also pass a `String` (which is what is done in your snippet) using the following syntax:

---

<sup>634</sup><http://itextpdf.com/sandbox/merge/MergeWithToc2>

<sup>635</sup><http://stackoverflow.com/questions/21600040/pdf-page-re-ordering-using-itext>

<sup>636</sup><http://stackoverflow.com/users/2178702/butani-vijay>

```
[!] [o] [odd] [e] [even] start[-end]
```

You can have multiple ranges separated by commas, and the ! modifier removes pages from what is already selected. The range changes are incremental; numbers are added or deleted as the range appears. The start or the end can be omitted; if you omit both, you need at least o (odd; selects all odd pages) or e (even; selects all even pages).

In your case, we have:

```
String.format("%d, 1-%d", n, n-1)
```

Suppose we have a document with 10 pages, then n equals 10 and the result of the formatting operation is: "10, 1-9". In this case, PdfReader will present the last page as the first one, followed by pages 1 to 9.

Now suppose that you have a TOC that starts on page 8, and you want to move this TOC to the first pages, then you need something like this: 8-10, 1-7, or if toc equals 8 and n equals 10:

```
String.format("%d-%d, 1-%d", toc, n, toc -1)
```

For more info about the `format()` method, see the API docs for [String<sup>637</sup>](#) and the [Format String syntax<sup>638</sup>](#).

---

<sup>637</sup><http://docs.oracle.com/javase/6/docs/api/java/lang/String.html#format%28java.lang.String,%20java.lang.Object...%29>

<sup>638</sup><http://docs.oracle.com/javase/6/docs/api/java/util/Formatter.html#syntax>

## How to reorder the pages of a PDF file?



I am generating Table of Contents at last,I want to move Table of Contents at Beginning. Suppose that I have 16 pages in my PDF and that the TOC starts from page 13 and ends on page 15. I want to move the TOC to the second page, so that the first page remains page 1 and the last page remains page 16. This code doesn't give me what I want:

```
public void changePagesOrder() {
    try {
        PdfReader sourcePDFReader = new PdfReader(RESULT1);
        int n = sourcePDFReader.getNumberOfPages();
        System.out.println("no of pages in pdf files..." + n);
        int totalNoPages=n;
        int tocStartsPage=13;
        sourcePDFReader.selectPages(String.format("%d-%d, 2-%d",
            tocStartsPage, totalNoPages-1, tocStartsPage -2));
        PdfStamper stamper = new PdfStamper(sourcePDFReader,
            new FileOutputStream(RESULT2));
        stamper.close();
    }
    catch(Exception ex) { }
}
```

Posted on StackOverflow on Jun 18, 2015<sup>639</sup> by Surya Rawat<sup>640</sup>

Your formula is wrong. You have:

```
sourcePDFReader.selectPages(String.format("%d-%d, 2-%d",
    tocStartsPage, totalNoPages-1, tocStartsPage -2));
```

But that puts your TOC at page 1. That is not what you want according to your description.

You want something like this:

---

<sup>639</sup><http://stackoverflow.com/questions/30911216/how-to-re-arrange-pages-in-pdf-using-itext>

<sup>640</sup><http://stackoverflow.com/users/2656897/surya-rawat>

```
PdfReader reader = new PdfReader(baos.toByteArray());
int startToc = 13;
int n = reader.getNumberOfPages();
reader.selectPages(String.format("1,%s-%s, 2-%s, %s",
    startToc, n-1, startToc - 1, n));
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
stamper.close();
```

This code was tested using the [ReorderPage<sup>641</sup>](#) example on a PDF with 16 pages, having the text Page 1, Page 2, ..., Page 16 as content. The result was the following PDF: (reordered.pdf)[<http://itextpdf.com/sites/default/>]

The pages are now in this order: page 1, page 13, page 14, page 15, page 2, page 3, page 4, page 5, page 6, page 7, page 8, page 9, page 10, page 11, page 12, page 16. This is the order you described in your question.



It's working! Thanks a lot. I just want to say, I am working on iText for the last two months and I found it very interesting. One small extra question: I want to understand how the string formatter is working in this case.

You want to know how `String.format()` worked in this case. Let's look at what we want to achieve first. We have pages in this order:

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16

We want to reorder them like this:

1, 13, 14, 15, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 16

This means that we need this pattern:

1, 13-15, 2-12, 16

This is a hard-coded pattern, where two variables are important:

- The start of the TOC: page 13 (`startToc`)
- The final page: 16 (`n`)

From these variables, we derive two more variables:

- The last page of the TOC. This is the last page minus one, or  $16 - 1 = 15$  (`n - 1`)
- The last page before the TOC:  $13 - 1 = 12$  (`startToc - 1`)

We can now rewrite the pattern like this:

---

<sup>641</sup><http://itextpdf.com/sandbox/stamper/ReorderPages>

```
1, startToc-(n - 1), 2-(startToc - 1), n
```

We need to make this a String, so that's why we use `String.format()`:

```
String.format("1,%s-%s, 2-%s, %s", startToc, n-1, startToc - 1, n)
```

The first occurrence of `%s` is replaced by the first parameter after the `String`, the second occurrence of `%s` is replaced by the second parameter after the `String`, and so on...

If `startToc = 13` and `n = 16`, this results in:

```
1, 13-15, 2-12, 16
```

## How to set the OCG state of an existing PDF?



I need to turn off a specific layer (OCG object) in an existing PDF that was created in Autocad and that sets the default value to “on”. How can I change this to “off”?

Posted on StackOverflow on May 1, 2014 <sup>[642](#)</sup> by HoustonVBnovice<sup>[643](#)</sup>

I've created an example that turns off the visibility of a specific layer. See [ChangeOCG](#)<sup>[644](#)</sup>.

The concept is really simple. You already have a `PdfReader` object and you want to apply a change to a file. As documented, you create a `PdfStamper` object. As you want to change an OCG layer, you use the `getPdfLayers()` method and you select the layer you want to change by name. (In my example, the layer I want to turn off is named “Nested layer 1”). You use the `setOn()` method to change its status, and you're done:

```
PdfReader reader = new PdfReader(src);
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
Map<String, PdfLayer> layers = stamper.getPdfLayers();
PdfLayer layer = layers.get("Nested layer 1");
layer.setOn(false);
stamper.close();
reader.close();
```

This is Java code. Please read it as if it were pseudo-code and adapt it to your language of choice.

---

<sup>642</sup><http://stackoverflow.com/questions/23415419/using-itextsharp-to-set-ocg-state-of-existing-pdf>

<sup>643</sup><http://stackoverflow.com/users/3594000/houstonvb novice>

<sup>644</sup><http://itextpdf.com/sandbox/stamper/ChangeOCG>

## How to change the order of Optional Content Groups?



I have a PDF file with a hierarchy of layers (aka OCG). Using the following code snippet

```

1 var ocProps = reader.Catalog.GetAsDict(PdfName.OCPROPERTIES);
2 var occd = ocProps.GetAsDict(PdfName.D);
3 var order = occd.GetAsArray(PdfName.ORDER);

```

I can query the current order from the source file. But I have no idea how to modify this data in order to copy it into a new file with the following snippet.

```

var reader = new PdfReader(input);
var document = new Document(reader.GetPageSizeWithRotation(1));
var pdfCopyProvider = new PdfCopy(document,
    new System.IO.FileStream(output, System.IO.FileMode.Create));
document.Open();
// TBD do OCG modification ...
var importedPage = pdfCopyProvider.GetImportedPage(reader, 1);
pdfCopyProvider.AddPage(importedPage);
document.Close();

```

Nonetheless, the ogc information is copied to the new pdf file by default.

Any hint on this is welcome.

Posted on StackOverflow on Apr 24, 2014<sup>645</sup> by Holger<sup>646</sup>

You're already very close to the solution. See the [ChangeOCGOrder<sup>647</sup>](#) example to find out how to change [ocg.pdf<sup>648</sup>](#) into [ocg\\_reordered.pdf<sup>649</sup>](#).

You already had something like this:

```

PdfDictionary catalog = reader.getCatalog();
PdfDictionary ocProps = catalog.getAsDict(PdfName.OCPROPERTIES);
PdfDictionary occd = ocProps.getAsDict(PdfName.D);
PdfArray order = occd.getAsArray(PdfName.ORDER);

```

This is good: you're looking at the right place!

Now you need something like this:

<sup>645</sup><http://stackoverflow.com/questions/23280083/itextsharp-change-order-of-optional-content-groups>

<sup>646</sup><http://stackoverflow.com/users/1330968/holger>

<sup>647</sup><http://itextpdf.com/sandbox/stamper/ChangeOCGOrder>

<sup>648</sup><http://itextpdf.com/sites/default/files/ocg.pdf>

<sup>649</sup>[http://itextpdf.com/sites/default/files/ocg\\_reordered.pdf](http://itextpdf.com/sites/default/files/ocg_reordered.pdf)

```
PdfObject nestedLayers = order.getObject(0);
PdfObject nestedLayerArray = order.getObject(1);
PdfObject groupedLayers = order.getObject(2);
PdfObject radiogroup = order.getObject(3);
order.set(0, radiogroup);
order.set(1, nestedLayers);
order.set(2, nestedLayerArray);
order.set(3, groupedLayers);
```

In my example, the ORDER array contains 4 elements. I get these four elements, and I change the order of the entries in the original array.

Note that I could also have done something like:

```
order.addFirst(order.remove(3));
```

That would have the same effect as the 8 lines of code above, but the 8 lines help you understand the mechanism.



Thanks, that works! In addition to your answer, I publish the C# version of the manipulatePdf method:

```
public void ManipulatePdf(string source, string destination)
{
    var reader = new PdfReader(source);
    var ocProps = reader.Catalog.GetAsDict(PdfName.OCPROPERTIES);
    var occd = ocProps.GetAsDict(PdfName.D);
    var order = occd.GetAsArray(PdfName.ORDER);

    var nestedLayers = (PdfObject)order[0];
    var nestedLayerArray = (PdfObject)order[1];
    var groupedLayers = (PdfObject)order[2];
    var radiogroup = (PdfObject)order[3];

    order[0] = radiogroup;
    order[1] = nestedLayers;
    order[2] = nestedLayerArray;
    order[3] = groupedLayers;

    var stamper = new PdfStamper(reader, new System.IO.FileStream(destination,
        System.IO.FileMode.Create));
```

```

    stamper.Close();
    reader.Close();
}

```

## How to create a PDF with font information and embed the actual font while merging the files into a single PDF?



I create pdfs and concatenate them into a single pdf. My resulting pdf is a lot bigger than I had expected in file size. As it turns out, my PDF has a ton of duplicate fonts, and this is the reason why it's so big. I would like to create PDFs which only embed font information, not the full font. Then when I merge these PDFs into a single document, I want to insert actual font needed by the PDF.

Posted on StackOverflow on Feb 24, 2014<sup>650</sup> by pixerce<sup>651</sup>

I've created the [MergeAndAddFont<sup>652</sup>](#) example to explain the different options.

We'll create PDFs using this code snippet:

```

public void createPdf(String filename, String text, boolean embedded, boolean su\\
bset)
    throws DocumentException, IOException {
    // step 1
    Document document = new Document();
    // step 2
    PdfWriter.getInstance(document, new FileOutputStream(filename));
    // step 3
    document.open();
    // step 4
    BaseFont bf = BaseFont.createFont(FONT, BaseFont.WINANSI, embedded);
    bf.setSubset(subset);
    Font font = new Font(bf, 12);
    document.add(new Paragraph(text, font));
    // step 5
    document.close();
}

```

---

<sup>650</sup><http://stackoverflow.com/questions/21979200/how-to-create-pdf-with-font-information-and-embed-actual-font-when-merging-them>

<sup>651</sup><http://stackoverflow.com/users/3345194/pixerce>

<sup>652</sup><http://itextpdf.com/sandbox/fonts/MergeAndAddFont>

We use this code to create 3 test files, 1, 2, 3 and we'll do this 3 times: A, B, C.

**The first time**, we use the parameters `embedded = true` and `subset = true`, resulting in the files `testA1.pdf`<sup>653</sup> with text "abcdefgh" (3.71 KB), `testA2.pdf`<sup>654</sup> with text "ijklmnopq" (3.49 KB) and `testA3.pdf`<sup>655</sup> with text "rstuvwxyz" (3.55 KB). The font is embedded and the file size is relatively low because we only embed a subset of the font.

Now we merge these files using the following code, using the `smart` parameter to indicate whether we want to use `PdfCopy` or `PdfSmartCopy`:

```
public void mergeFiles(String[] files, String result, boolean smart)
    throws IOException, DocumentException {
    Document document = new Document();
    PdfCopy copy;
    if (smart)
        copy = new PdfSmartCopy(document, new FileOutputStream(result));
    else
        copy = new PdfCopy(document, new FileOutputStream(result));
    document.open();
    PdfReader[] reader = new PdfReader[3];
    for (int i = 0; i < files.length; i++) {
        reader[i] = new PdfReader(files[i]);
        copy.addDocument(reader[i]);
    }
    document.close();
    for (int i = 0; i < reader.length; i++) {
        reader[i].close();
    }
}
```

When we merge the document, be it with `PdfCopy` or `PdfSmartCopy`, the different subsets of the same font will be copied as separate objects in the resulting PDF `testA_merged1.pdf`<sup>656</sup> / `testA_merged2.pdf`<sup>657</sup> (both 9.75 KB).

This is the problem you are experiencing: `PdfSmartCopy` can detect and reuse identical objects, but the different subsets of the same font aren't identical and iText can't merge different subsets of the same font into one font.

**The second time**, we use the parameters `embedded = true` and `subset = false`, resulting in the

---

<sup>653</sup><http://itextpdf.com/sites/default/files/testA1.pdf>

<sup>654</sup><http://itextpdf.com/sites/default/files/testA2.pdf>

<sup>655</sup><http://itextpdf.com/sites/default/files/testA3.pdf>

<sup>656</sup>[http://itextpdf.com/sites/default/files/testA\\_merged1.pdf](http://itextpdf.com/sites/default/files/testA_merged1.pdf)

<sup>657</sup>[http://itextpdf.com/sites/default/files/testA\\_merged2.pdf](http://itextpdf.com/sites/default/files/testA_merged2.pdf)

files `testB1.pdf`<sup>658</sup> (21.38 KB), `testB2.pdf`<sup>659</sup> (21.38 KB) and `testA3.pdf`<sup>660</sup> (21.38 KB). The font is fully embedded and the file size of a single file is a lot bigger than before because the full font is embedded.

If we merge the files using `PdfCopy`, the font will be present in the merged document redundantly, resulting in the bloated file `testB_merged1.pdf`<sup>661</sup> (63.16 KB). This is definitely not what you want!

However, if we use `PdfSmartCopy`, iText detects an identical font stream and reuses it, resulting in `testB_merged2.pdf`<sup>662</sup> (21.95 KB) which is much smaller than we had with `PdfCopy`. It's still bigger than the document with the subsetted fonts, but if you're concatenating a huge amount of files, the result will be better if you embed the complete font.

The third time, we use the parameters `embedded = false` and `subset = false`, resulting in the files `testC1.pdf`<sup>663</sup> (2.04 KB), `testC2.pdf`<sup>664</sup> (2.04 KB) and `testC3.pdf`<sup>665</sup> (2.04 KB). The font isn't embedded, resulting in an excellent file size, but if you compare with one of the previous results, you'll see that the font looks completely different.

We merge the files using `PdfSmartCopy`, resulting in `testC_merged1.pdf`<sup>666</sup> (2.6 KB). Again, we have an excellent file size, but again we have the problem that the font isn't visualized correctly.

To fix this, we need to embed the font:

```
private void embedFont(String merged, String fontfile, String result)
    throws IOException, DocumentException {
    // the font file
    RandomAccessFile raf = new RandomAccessFile(fontfile, "r");
    byte fontbytes[] = new byte[(int)raf.length()];
    raf.readFully(fontbytes);
    raf.close();
    // create a new stream for the font file
    PdfStream stream = new PdfStream(fontbytes);
    stream.flateCompress();
    stream.put(PdfName.LENGTH1, new PdfNumber(fontbytes.length));
    // create a reader object
    PdfReader reader = new PdfReader(merged);
    int n = reader.getXrefSize();
    PdfObject object;
    PdfDictionary font;
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(result));
```

<sup>658</sup><http://itextpdf.com/sites/default/files/testB1.pdf>

<sup>659</sup><http://itextpdf.com/sites/default/files/testB2.pdf>

<sup>660</sup><http://itextpdf.com/sites/default/files/testB3.pdf>

<sup>661</sup>[http://itextpdf.com/sites/default/files/testB\\_merged1.pdf](http://itextpdf.com/sites/default/files/testB_merged1.pdf)

<sup>662</sup>[http://itextpdf.com/sites/default/files/testB\\_merged2.pdf](http://itextpdf.com/sites/default/files/testB_merged2.pdf)

<sup>663</sup><http://itextpdf.com/sites/default/files/testC1.pdf>

<sup>664</sup><http://itextpdf.com/sites/default/files/testC1.pdf>

<sup>665</sup><http://itextpdf.com/sites/default/files/testC1.pdf>

<sup>666</sup>[http://itextpdf.com/sites/default/files/testC\\_merged1.pdf](http://itextpdf.com/sites/default/files/testC_merged1.pdf)

```

PdfName fontname = new PdfName(
    BaseFont.createFont(fontfile, BaseFont.WINANSI, BaseFont.NOT_EMBEDDED)
    .getPostscriptFontName());
for (int i = 0; i < n; i++) {
    object = reader.getPdfObject(i);
    if (object == null || !object.isDictionary())
        continue;
    font = (PdfDictionary)object;
    if (PdfName.FONTDESCRIPTOR.equals(font.get(PdfName.TYPE))
        && fontname.equals(font.get(PdfName.FONTNAME))) {
        PdfIndirectObject objref = stamper.getWriter().addToBody(stream);
        font.put(PdfName.FONTFILE2, objref.getIndirectReference());
    }
}
stamper.close();
reader.close();
}

```

Now, we have the file [testC\\_merged2.pdf<sup>667</sup>](#) (22.03 KB) and that's actually the answer to your question. As you can see, the second option is better than this third option.

**Caveats:** This example uses the Gravitas One font as a *simple font*. As soon as you use the font as a *composite font* (you tell iText to use it as a composite font by choosing the encoding IDENTITY-H or IDENTITY-V), you can no longer choose whether or not to embed the font, whether or not to subset the font. As defined in ISO-32000-1, iText will always embed composite fonts and will always subset them.

This means that you can't use the above solutions when you need special fonts (Chinese, Japanese, Korean). In that case, you shouldn't embed the fonts, but use so-called CJK fonts. They CJK fonts will use font packs that can be downloaded by Adobe Reader.

## How to decrypt a PDF document with the owner password?



I need to be able to remove the security/encryption from some PDF documents, preferably with the iText library. This used to be possible, but after a [change in version 5.3.5<sup>668</sup>](#) of the library that solution no longer works.

Posted on StackOverflow on January 9, 2015 <sup>669</sup> by [Daniel Pratt<sup>670</sup>](#)

<sup>667</sup>[http://itextpdf.com/sites/default/files/testC\\_merged2.pdf](http://itextpdf.com/sites/default/files/testC_merged2.pdf)

<sup>668</sup><http://itextpdf.com/changelog/535>

<sup>669</sup><http://stackoverflow.com/questions/27867868/how-can-i-decrypt-a-pdf-document-with-the-owner-password>

<sup>670</sup><http://stackoverflow.com/users/76123/daniel-pratt>

In order to test code to encrypt a PDF file, we need a sample PDF that is encrypted. We'll create such a file using the [EncryptPdf<sup>671</sup>](#) example.

```
public void manipulatePdf(String src, String dest)
    throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.setEncryption("Hello".getBytes(), "World".getBytes(),
        PdfWriter.ALLOW_PRINTING,
        PdfWriter.ENCRYPTION_AES_128 | PdfWriter.DO_NOT_ENCRYPT_METADATA);
    stamper.close();
}
```

With this code, I create an encrypted file [hello\\_encrypted.pdf<sup>672</sup>](#) that I will use in the first example demonstrating how to decrypt a file.

Your original question sounds like “How can I decrypt a PDF document with the owner password?”

That is easy. The [DecryptPdf<sup>673</sup>](#) example shows you how to do this:

```
public void manipulatePdf(String src, String dest)
    throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src, "World".getBytes());
    System.out.println(new String(reader.computeUserPassword()));
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.close();
    reader.close();
}
```

We create a `PdfReader` instance passing the owner password as the second parameter. If we want to know the user password, we can use the `computeUserPassword()` method. Should we want to encrypt the file, than we can use the owner password we know and the user password we computed and use the `setEncryption()` method to reintroduce security.

However, as we didn't do this, all security is removed, which is exactly what you wanted. This can be checked by looking at the [hello.pdf<sup>674</sup>](#) document.

One could argue that your question falls in the category of “It doesn't work” questions that can only be answered with an “it works for me” answer. One could vote to close your question because you didn't provide a code sample that can be used to reproduce the problem, whereas anyone can provide a code sample that proves you wrong.

---

<sup>671</sup><http://itextpdf.com/sandbox/security/EncryptPdf>

<sup>672</sup>[http://itextpdf.com/sites/default/files/hello\\_encrypted.pdf](http://itextpdf.com/sites/default/files/hello_encrypted.pdf)

<sup>673</sup><http://itextpdf.com/sandbox/security/DecryptPdf>

<sup>674</sup><http://itextpdf.com/sites/default/files/hello.pdf>

Fortunately, I can read between the lines, so I have made another example.

Many PDFs are encrypted without a user password. They can be opened by anyone, but encryption is added to enforce certain restrictions (e.g. you can view the document, but you can not print it). In this case, there is only an owner password, as is shown in the [EncryptPdfWithoutUserPassword](#)<sup>675</sup> example:

```
public void manipulatePdf(String src, String dest)
    throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.setEncryption(null, "World".getBytes(),
        PdfWriter.ALLOW_PRINTING,
        PdfWriter.ENCRYPTION_AES_128 | PdfWriter.DO_NOT_ENCRYPT_METADATA);
    stamper.close();
    reader.close();
}
```

Now we get a PDF that is encrypted, but that can be opened without a user password: [hello\\_-encrypted2.pdf](#)<sup>676</sup>

We still need to know the owner password if we want to manipulate the PDF. If we don't pass the password, then iText will **rightfully** throw an exception:

```
Exception in thread "main" com.itextpdf.text.exceptions.BadPasswordException:
    Bad user password
        at com.itextpdf.text.pdf.PdfReader.readPdf(PdfReader.java:681)
        at com.itextpdf.text.pdf.PdfReader.<init>(PdfReader.java:181)
        at com.itextpdf.text.pdf.PdfReader.<init>(PdfReader.java:230)
        at com.itextpdf.text.pdf.PdfReader.<init>(PdfReader.java:207)
        at sandbox.security.DecryptPdf.manipulatePdf(DecryptPdf.java:26)
        at sandbox.security.DecryptPdf.main(DecryptPdf.java:22)
```

But what if we don't remember that owner password? What if the PDF was produced by a third party and we do not want to respect the wishes of that third party?

In that case, you can deliberately be unethical and change the value of the static `unethicalReading` variable. This is done in the [DecryptPdf2](#)<sup>677</sup> example:

---

<sup>675</sup><http://itextpdf.com/sandbox/security/EncryptPdfWithoutUserPassword>

<sup>676</sup>[http://itextpdf.com/sites/default/files/hello\\_encrypted2.pdf](http://itextpdf.com/sites/default/files/hello_encrypted2.pdf)

<sup>677</sup><http://itextpdf.com/sandbox/security/DecryptPdf2>

```
public void manipulatePdf(String src, String dest)
    throws IOException, DocumentException {
    PdfReader.unethicalreading = true;
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.close();
    reader.close();
}
```

This example will not work if the document was encrypted with a user *and* an owner password, in that case, you will have to pass at least one password, either the “owner password” or the “user password” (the fact that you have access to the PDF using only the “user” password is a side-effect of unethical reading). If only an *owner* password was introduced, iText does not need that owner password to manipulate the PDF if you change the `unethicalreading` flag.

However: there used to be a bug in iText that also removed the owner password(s) in this case. That is not the desired behavior. In the first `PdfDecrypt` example, we saw that we can retrieve the user password (if a user password was present), but there is no way to retrieve the owner password. It is truly secret. With the older versions of iText you refer to, the owner password was removed from the file after manipulating it, and that owner password was lost for eternity.

I have fixed this bug and the fix is in release 5.3.5. As a result, the owner password is now preserved. You can check this by looking at [hello2.pdf](http://itextpdf.com/sites/default/files/hello2.pdf)<sup>678</sup>, which is the file we decrypted in an “unethical” way. (If there was an owner and a user password, both are preserved.)

Based on this research, I am making the assumption that your question is incorrect. You meant to ask: “How can I decrypt a PDF document *without* the owner password?” or “How can I decrypt a PDF with the *user* password?”

It doesn’t make sense to *unfix* a bug that I once fixed. We will not restore the (wrong) behavior of the old iText versions, but that doesn’t mean that you can’t achieve what you want. You’ll only have to fool iText into thinking that the PDF wasn’t encrypted.

This is shown in the `DecryptPdf3`<sup>679</sup> example:

---

<sup>678</sup><http://itextpdf.com/sites/default/files/hello2.pdf>

<sup>679</sup><http://itextpdf.com/sandbox/security/DecryptPdf3>

```

class MyReader extends PdfReader {
    public MyReader(String filename) throws IOException {
        super(filename);
    }
    public void decryptOnPurpose() {
        encrypted = false;
    }
}
public void manipulatePdf(String src, String dest)
throws IOException, DocumentException {
    MyReader.unethicalreading = true;
    MyReader reader = new MyReader(src);
    reader.decryptOnPurpose();
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.close();
    reader.close();
}

```

Instead of PdfReader, we are now using a custom subclass of PdfReader. I have named it MyReader and I have added an extra method that allows me to set the encrypted variable to false.

I still need to use unethicalreading and right after creating the MyReader instance, I have to fool this reader into thinking that the original file wasn't encrypted by using the decryptOnPurpose() method.

This results in the file [hello3.pdf<sup>680</sup>](#) which is a file that is no longer encrypted with an owner password. This example can even be used to remove all passwords from a file that is encrypted with a user and an owner password as long as you have the user password.

## How do I add XMP metadata to each page of an existing PDF?



So far all the examples seem to involve adding metadata while creating a new Document. I want to take an existing PDF and add a GUID to each page's XMP using the stamper.

Posted on StackOverflow on [Feb 10, 2015](#)<sup>681</sup> by [Rahul](#)<sup>682</sup>

The most common use of XMP in the context of PDF is when you add an XMP stream for the whole document that is referred to from the root dictionary of the PDF (aka the Catalog).

---

<sup>680</sup><http://itextpdf.com/sites/default/files/hello3.pdf>

<sup>681</sup><http://stackoverflow.com/questions/28427100/how-do-i-add-xmp-metadata-to-each-page-of-an-existing-pdf-using-itextsharp>

<sup>682</sup><http://stackoverflow.com/users/11100/rahul>

However, if you consult the PDF specification, you notice that XMP can be used in the context of many other objects inside a PDF, the page level being one of them. If you look at the spec, you will discover that `/Metadata` is an optional key in the page dictionary. It expects a reference to an XMP stream.

If you would use iText to create a PDF document from scratch, you wouldn't find a specific method to add this metadata, but you could use the generic `addPageDictEntry()` that is available in `PdfWriter`. You would pass `PdfName.METADATA` as key and a reference to a stream that is already added to the `PdfWriter` as value.

Your question isn't about creating a PDF from scratch, but about modifying an existing PDF. In that case, you also need the page dictionary. It is very easy to obtain these dictionaries:

```
PdfReader reader = new PdfReader(src);
int n = reader.getNumberOfPages();
PdfDictionary page;
for (int p = 1; p <= n; p++) {
    page = reader.getPageN(p);
    // do stuff with the page dictionary
}
```

This snippet was taken fro the [Rotate90Degrees<sup>683</sup>](#) example. In that example, we look at the `/Rotate` entry which is a number:

```
PdfNumber rotate = page.getAsNumber(PdfName.ROTATE);
```

You need to look for the `/Metadata` entry, which is a stream:

```
PRStream stream = (PRStream) page.getAsStream(PdfName.METADATA);
```

Maybe this stream is `null`, in this case, you need to add a `/Metadata` entry as is shown in the [AddXmpToPage<sup>684</sup>](#) example:

---

<sup>683</sup><http://itextpdf.com/sandbox/stamper/Rotate90Degrees>

<sup>684</sup><http://itextpdf.com/sandbox/stamper/AddXmpToPage>

```
// We create some XMP bytes
ByteArrayOutputStream baos = new ByteArrayOutputStream();
XmpWriter xmp = new XmpWriter(baos, new PdfDictionary());
xmp.close();
// We add the XMP bytes to the writer
PdfIndirectObject ref = stamper.getWriter().addToBody(new PdfStream(baos.toByteArray()));
// We add a reference to the XMP bytes to the page dictionary
page.put(PdfName.METADATA, ref.getIndirectReference());
```

If there is an XMP stream, you want to keep it and add something to it.

This is how you get the XMP bytes:

```
byte[] xmpBytes = PdfReader.getStreamBytes(stream);
```

You perform your XML magic on these bytes, resulting in a new byte[] named newXmpBytes. You replace the original bytes with these new bytes like this:

```
stream.setData(newXmpBytes);
```

All these operations are done on the existing file that resides in the PdfReader object. You now have to persist the changes like this:

```
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
stamper.close();
reader.close();
```

## How to move the content inside a rectangle inside an existing PDF?



My goal is to move text in a PDF around, which is within a certain rectangular area. All content matching the area in the original PDF should be shifted somewhat in the x and y direction. I have no influence on how the PDFs are created and I can accept a solution that only vaguely works.

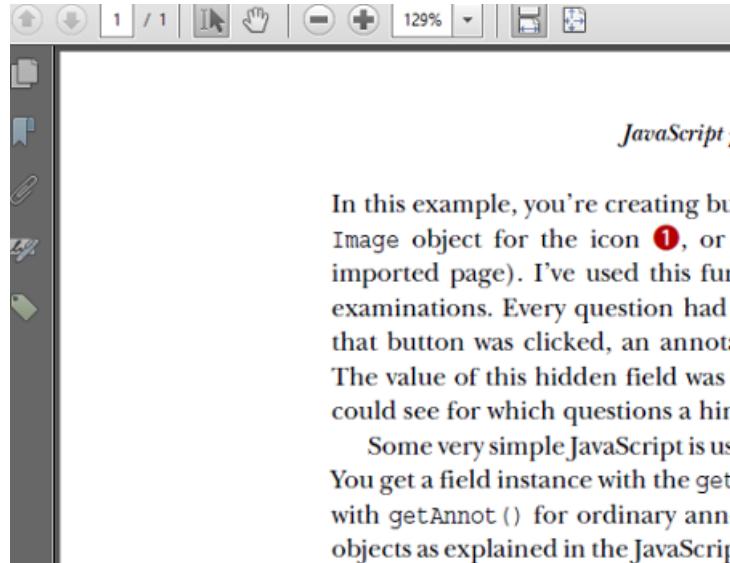
Posted on StackOverflow on Feb 6, 2015<sup>685</sup> by tom\_ink<sup>686</sup>

---

<sup>685</sup><http://stackoverflow.com/questions/28368317/itext-or-itextsharp-move-text-in-an-existing-pdf>

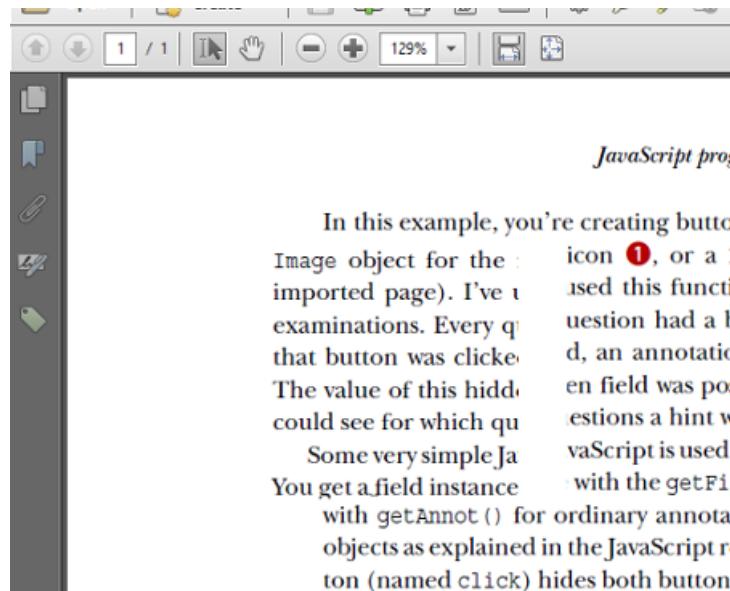
<sup>686</sup><http://stackoverflow.com/users/4537574/tom-imk>

Suppose that you have a PDF document like this:



Original PDF

I also have the coordinates of a rectangle: new Rectangle(100, 500, 200, 600); and an offset: move everything in that rectangle 10 points to the left and 2 points to the bottom, like this:



Adapted PDF

That is fairly easy to achieve. Take a look at the [CutAndPaste](#)<sup>687</sup> example:

---

<sup>687</sup><http://itextpdf.com/sandbox/merge/CutAndPaste>

```
public void manipulatePdf(String src, String dest)
    throws IOException, DocumentException {
    // Creating a reader
    PdfReader reader = new PdfReader(src);
    // step 1
    Rectangle pageSize = reader.getPageSize(1);
    Rectangle toMove = new Rectangle(100, 500, 200, 600);
    Document document = new Document(pageSize);
    // step 2
    PdfWriter writer
        = PdfWriter.getInstance(document, new FileOutputStream(dest));
    // step 3
    document.open();
    // step 4
    PdfImportedPage page = writer.getImportedPage(reader, 1);
    PdfContentByte cb = writer.getDirectContent();
    PdfTemplate template1 = cb.createTemplate(pageSize.getWidth(), pageSize.getHeight());
    template1.rectangle(0, 0, pageSize.getWidth(), pageSize.getHeight());
    template1.rectangle(toMove.getLeft(), toMove.getBottom(),
        toMove.getWidth(), toMove.getHeight());
    template1.eoClip();
    template1.newPath();
    template1.addTemplate(page, 0, 0);
    PdfTemplate template2 = cb.createTemplate(pageSize.getWidth(), pageSize.getHeight());
    template2.rectangle(toMove.getLeft(), toMove.getBottom(),
        toMove.getWidth(), toMove.getHeight());
    template2.clip();
    template2.newPath();
    template2.addTemplate(page, 0, 0);
    cb.addTemplate(template1, 0, 0);
    cb.addTemplate(template2, -20, -2);
    // step 4
    document.close();
    reader.close();
}
```

## How to load a PDF from a stream and add a file attachment?



I'm using a MS SQL Report Server web service to generate reports in the PDF format:

```
byte[] Input;
ReportServer report = new ReportServer(
    serverUrl + @"/ReportExecution2005.asmx", reportPath);
Input = report.RenderToPDF(reportParamNames, reportParamValues);
```

This service returns a byte array with pdf file and I need this byte array load to iTextSharp:

```
using (MemoryStream ms = new MemoryStream(Input)) {
    Document doc = new Document();
    PdfWriter writer = PdfWriter.GetInstance(doc, ms);
    doc.Open();
    ...
}
```

This seems ok, but then I am trying to add an attachment to this PDF:

```
PdfFileSpecification pfs = PdfFileSpecification.FileEmbedded(writer, xmlInputFil\
e, xmlFileDisplayName, null);
writer.AddFileAttachment(pfs);
```

This seems OK too, but when I save stream to file, the resulting PDF is not correct.

Note that the file attachment will always be an XML file, which I need to create in memory and never will be in file system. How can I do this with iTextSharp?

Posted on StackOverflow on Jan 29, 2015 <sup>688</sup> by Davecz<sup>689</sup>

I read:

This service returns a byte array with pdf file and I need this byte array load to iTextSharp:

---

<sup>688</sup><http://stackoverflow.com/questions/28210389/loading-pdf-from-stream-and-add-file-attachment>

<sup>689</sup><http://stackoverflow.com/users/1654683/davecz>

```
using (MemoryStream ms = new MemoryStream(Input))
{
    Document doc = new Document();
    PdfWriter writer = PdfWriter.GetInstance(doc, ms);
    doc.Open();
    ...
}
```

this seems ok

This is **not** OK. You want to add an attachment to an existing PDF file, however, you are using `Document` and `PdfWriter` which are classes to create a **new PDF document from scratch**.

Please read the [documentation<sup>690</sup>](#). There is a handy table (6.1) that gives you an overview of the different classes and when they are to be used.

I quote the description of the `PdfReader` and `PdfStamper` class:

`PdfReader`: Reads PDF files. You pass an instance of this class to one of the other PDF manipulation classes.

`PdfStamper`: Manipulates one (and only one) PDF document. Can be used to add content at absolute positions, to add extra pages, or to fill out fields. All interactive features are preserved, except when you explicitly remove them (for instance, by flattening a form).

We have established that you are doing it wrong: you should use `PdfReader` and `PdfStamper` instead of `Document` and `PdfWriter`. Now let's take a look at [some examples<sup>691</sup>](#):

<sup>690</sup><http://manning.com/lowagie2/samplechapter6.pdf>

<sup>691</sup><http://itextpdf.com/book/chapter.php?id=6>

```
PdfReader reader = new PdfReader(pdf_bytes);
using (var ms = new MemoryStream()) {
    using (PdfStamper stamper = new PdfStamper(reader, ms)) {
        PdfFileSpecification pfs = PdfFileSpecification.FileEmbedded(
            stamper.Writer, xmlInputFile, xmlFileDialogName, null);
        stamper.AddFileAttachment(pfs);
    }
    reader.Close();
    return ms.ToArray();
}
```

As you can see, we create a `PdfReader` instance using the bytes that were kept in memory. We then use `PdfStamper` to create a new `MemoryStream` of which we use the bytes.

# Interactive forms

Is your form based on AcroForm technology or is it based on the XML Forms Architecture? That's a common counter-question you'll be confronted with when asking a question about forms. In any case, these answers should help you solving the most common problems with respect to forms.

## How to fill out a pdf file programmatically? (AcroForm technology)



What techniques available to fill a pdf form automatically using external data and save them. I have to use data from a database to fill a template PDF and save a copy of it on disk with that data. Language and platform is not issue but it would be good if it can run on windows and Linux.

Posted on StackOverflow on Jun 24, 2010 <sup>692</sup> by affan<sup>693</sup>

If your form is based on AcroForm technology, you can use iText to fill it out like this:

```
PdfReader reader = new PdfReader(src);
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
AcroFields form = stamper.getAcroFields();
form.setField(key, value);
stamper.setFormFlattening(true);
stamper.close();
reader.close();
```

In this snippet, `src` is the source of a PDF file (could be a path to a file, could be a `byte[]`) and `dest` is the path to the resulting PDF. The `key` corresponds with the name of a field in your template. The `value` corresponds with the value you want to fill in. If you want the form to keep its interactivity, you need to remove the line `stamper.setFormFlattening(true);` otherwise all form fields will be removed, resulting in a *flat* PDF.

---

<sup>692</sup><http://stackoverflow.com/questions/3108704/how-to-fill-out-a-pdf-file-programmatically>

<sup>693</sup><http://stackoverflow.com/users/109769/affan>

## How to fill out a pdf file programmatically? (Dynamic XFA)



I have a dynamic XFA Form that I can fill out manually using Adobe Acrobat on my computer. Using iTextSharp I can read what the XFA XML data is and see the structure of the data. I am essentially trying to mimic that with iText using the following code:

```
PdfReader pdfReader = new PdfReader(sourceFilePath);
using (MemoryStream ms = new MemoryStream()) {
    using (PdfStamper stamper = new PdfStamper(pdfReader, ms)) {
        XfaForm xfaForm = new XfaForm(pdfReader);
        XmlDocument doc = new XmlDocument();
        doc.Load(replacementXmlFilePath);
        xfaForm.DomDocument = doc;
        xfaForm.Changed = true;
        XfaForm.SetXfa(xfaForm, stamper.Reader, stamper.Writer);
    }
    var bytes = ms.ToArray();
    File.WriteAllBytes(destinationFilePath, bytes);
}
```

For some reason this code doesn't work.

Posted on StackOverflow on May 11, 2013<sup>694</sup> by jon333<sup>695</sup>

This question was answered by the person who posted the question:



I found the issue. The replacement DomDocument needs to be the entire merged XML of the new document, not just the data or datasets portion.

I upvoted this answer, because it's not incorrect, but now that I think it's better to use the [example from the book](#)<sup>696</sup>:

<sup>694</sup><http://stackoverflow.com/questions/16502427/how-can-i-set-xfa-data-in-a-static-xfa-form-in-itextsharp-and-get-it-to-save>

<sup>695</sup><http://stackoverflow.com/users/511518/jon333>

<sup>696</sup><http://itextpdf.com/examples/iiapdf?id=165>

```

public byte[] ManipulatePdf(String src, String xml) {
    PdfReader reader = new PdfReader(src);
    using (MemoryStream ms = new MemoryStream()) {
        using (PdfStamper stamper = new PdfStamper(reader, ms)) {
            AcroFields form = stamper.AcroFields;
            XfaForm xfa = form.Xfa;
            xfa.FillXfaForm(XmlReader.Create(new StringReader(xml)));
        }
        return ms.ToArray();
    }
}

```

As you can see, it's not necessary to replace the whole XFA XML. If you use the `FillXfaForm` method, the data is sufficient.

## Is it safe to remove XFA?



Are there any issues that can come up of removing the XFA format from a PDF form? If I don't drop XFA, I can see the fields pre-filled on Acrobat Reader but not in Acrobat Pro. Other viewers like Ubuntu Document Viewer, present the file correctly. I don't mind dropping XFA but I'm just checking if there might be issues that I am not aware of.

Posted on StackOverflow on Apr 14, 2015 <sup>[697](#)</sup> by user3501223<sup>[698](#)</sup>

There are three types of forms in PDF:

- Forms using AcroForm technology. In this case, each field corresponds with one or more widgets with fixed positions on specific pages. The form is described using nothing but PDF syntax.
- Dynamic forms using the XML Forms Architecture (XFA). In this case, the PDF file is nothing but a container for an XML file that describes the whole form. We refer to this as *dynamic* XFA, because the form can expand or shrink based on the data that is added: a 1-page form can turn into a 100-page form by adding more data.
- Hybrid forms that combine AcroForm and XFA technology. In this case, the form is described twice: once using PDF objects; once using XML. Obviously, such a form is not dynamic: the AcroForm part still defines widget annotations that are defined at absolute positions on specific pages. The form can't adapt to its data.

---

<sup>697</sup><http://stackoverflow.com/questions/29633873/pdfkit-and-removing-the-xfa-format>

<sup>698</sup><http://stackoverflow.com/users/3501223/user3501223>

If you have a *dynamic XFA form*, dropping the XML will remove the complete form. There won't be anything left.

However, it seems that you are confronted with a hybrid form that consists of both AcroForm and XFA syntax. Hybrid forms are a pain because they often lead to confusion. For instance: a viewer that is not XFA aware, will show you the data as stored in the AcroForm. A viewer that is XFA aware, can give preference to the data as stored in the XFA form. What's the problem, you might ask? Aren't both forms equivalent?

Ideally, both versions of the form are indeed equivalent, but:

- If the form isn't filled out correctly, the AcroForm can be different from the XFA form.
- XFA has more functionality than AcroForm technology. For instance: a text field in an XFA form can be *justified* (similar to `<p align="justify">` in HTML). However, this option doesn't exist in an AcroForm text field (you can only have left, center or right alignment). Hence if you have text that is justified in an XFA form, but you only look at the AcroForm, then the text won't be justified (because justified text doesn't exist in an AcroForm text field).

This is a long answer to explain that, if you have a hybrid form, it is *in most cases* OK to throw away the XFA part. You may have small differences, but if you are OK with what the form looks like in Ubuntu Document Viewer (a viewer that doesn't support XFA), then you should be fine.

## How to flatten a XFA PDF Form using iTextSharp?



I assume I need to flatten a XFA form in order to display properly on the UI of an application that uses Nuance's CSDK. When I process it now I get a generic message "*Please Wait.... If this message is not eventually replaced.*"

Posted on StackOverflow on Nov 21, 2014 <sup>699</sup> by Chuck<sup>700</sup>

You didn't insert a code sample to show how you are currently flattening the XFA form. I assume your code looks like this:

---

<sup>699</sup><http://stackoverflow.com/questions/27067250/how-can-i-flatten-a-xfa-pdf-form-using-itextsharp>

<sup>700</sup><http://stackoverflow.com/users/2413934/chuck>

```
var reader = new PdfReader(existingFileStream);
var stamper = new PdfStamper(reader, newFileStream);
var form = stamper.AcroFields;
var fieldKeys = form.Fields.Keys;
foreach (string fieldKey in fieldKeys) {
    form.SetField(fieldKey, "X");
}
stamper.FormFlattening = true;
stamper.Close();
reader.Close();
```

This code will work when trying to flatten forms based on AcroForm technology, **but it can't be used for forms based on the XML Forms Architecture (XFA)!**

A dynamic XFA form usually consists of a single page PDF (the one with the “Please wait...” message) that is shown in viewer that do not understand XFA. The actual content of the document is stored as an XML file.

The process of flattening an XFA form to an ordinary PDF requires that you parse the XML and convert the XML syntax into PDF syntax. This can be done using iText’s [XFA Worker](#)<sup>701</sup>.

Suppose that you have a file in memory (`ms`) that consists of an XFA form that is filled out, then you can use the following code to flatten that form:

```
Document document = new Document();
PdfWriter writer =
    PdfWriter.GetInstance(document, new FileStream(dest, FileMode.Create));
XFAFlattener xfaf = new XFAFlattener(document, writer);
ms.Position = 0;
xfaf.Flatten(new PdfReader(ms));
document.Close();
```

You can download a zip file containing the DLLs for XFA Worker as well as an example from the [iText web site](#)<sup>702</sup>.

---

<sup>701</sup>[http://itextpdf.com/product/xfa\\_worker](http://itextpdf.com/product/xfa_worker)

<sup>702</sup><http://itextsupport.com/download/xfa-net.zip>

## How to tell iText which fields to flatten first?



I have a PDF with text form fields at are layered one on top of the other. When I fill the fields via iText and flatten the form, the form field that I had created on top of the other form field is now on the bottom.

For instance, I have a text field named “number\_field” and that is underneath a second text field that is titled “name\_field”. When I set the value for those fields via iText (so ‘10’ for “number\_field” and ‘John’ for “name\_field”), the number\_field is now on top of the name\_field.

How do I change the order on the page of these fields with iText?

Posted on StackOverflow on Apr 14, 2015 <sup>[703](#)</sup> by Matt<sup>[704](#)</sup>

The easiest way to solve this problem, is to fill out the form in two passes! This is shown in the [FillFormFieldOrder<sup>705</sup>](#) example. We fill out the form `src` resulting in the flattened form `dest` like this:

```
public void manipulatePdf(String src, String dest) throws DocumentException, IOException {
    go2(go1(src), dest);
}
```

As you can see, we execute the `go1()` method first:

```
public byte[] go1(String src) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    ByteArrayOutputStream baos = new ByteArrayOutputStream();
    PdfStamper stamper = new PdfStamper(reader, baos);
    AcroFields form = stamper.getAcroFields();
    form.setField("sunday_1", "1");
    form.setField("sunday_2", "2");
    form.setField("sunday_3", "3");
    form.setField("sunday_4", "4");
    form.setField("sunday_5", "5");
    form.setField("sunday_6", "6");
    stamper.setFormFlattening(true);
    stamper.partialFormFlattening("sunday_1");
```

<sup>703</sup><http://stackoverflow.com/questions/29633035/change-acrofields-order-in-existing-pdf-with-itext>

<sup>704</sup><http://stackoverflow.com/users/4787547/matt>

<sup>705</sup><http://itextpdf.com/sandbox/acroforms/FillFormFieldOrder>

```

stamper.partialFormFlattening("sunday_2");
stamper.partialFormFlattening("sunday_3");
stamper.partialFormFlattening("sunday_4");
stamper.partialFormFlattening("sunday_5");
stamper.partialFormFlattening("sunday_6");
stamper.close();
reader.close();
return baos.toByteArray();
}

```

This fills out all the `sunday_x` fields and uses partial form flattening to flatten *only those fields*. The `go1()` method takes `src` as parameter and returns a `byte[]` will the partially flattened form.

The `byte[]` will be used as a parameter for the `go2()` method, that takes `dest` as its second parameter. Now we are going to fill out the `sunday_x_notes` fields:

```

public void go2(byte[] src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    AcroFields form = stamper.getAcroFields();
    form.setField("sunday_1_notes", "It's Sunday today, let's go to the sea");
    form.setField("sunday_2_notes", "It's Sunday today, let's go to the park");
    form.setField("sunday_3_notes", "It's Sunday today, let's go to the beach");
    form.setField("sunday_4_notes", "It's Sunday today, let's go to the woods");
    form.setField("sunday_5_notes", "It's Sunday today, let's go to the lake");
    form.setField("sunday_6_notes", "It's Sunday today, let's go to the river");
    stamper.setFormFlattening(true);
    stamper.close();
    reader.close();
}

```

As you can see, we now flatten all the fields. The result looks like [this<sup>706</sup>](#):

---

<sup>706</sup>[http://itextpdf.com/sites/default/files/calendar\\_filled.pdf](http://itextpdf.com/sites/default/files/calendar_filled.pdf)

Child Name:

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
It's Sunday today, let's go to the sea 1						
It's Sunday today, let's go to the park 2						
It's Sunday today, let's go to the beach 3						
It's Sunday today, let's go to the woods 4						
It's Sunday today, let's go to the lake 5						
It's Sunday today, let's go to the river 6						

Center Phone:      Center Contact:      Center Address:

Screen shot

Now, you no longer have to worry about the order of the fields, not in the /Fields array, not in the /Annots array. The fields are filled out in the exact order you want to. The notes cover the dates now, instead of the other way round.

## How to merge forms from different files into one PDF?



I currently have a PdfReader and a PdfStamper that I am using to fill out a form. I now have to add another PDF to the end of the form I have been filling out, but when I do this, I lose all the interactive fields. This is what I currently have:

```
public static void addSectionThirteenPdf(
    PdfStamper stamper, Rectangle pageSize, int pageIndex){
    PdfReader reader = new PdfReader("section13.pdf");
    AcroFields fields = reader.getAcroFields();
    fields.renameField("field", "field" + pageIndex);
    stamper.insertPage(pageIndex, pageSize);
    stamper.replacePage(reader, 1, pageIndex);
}
```

The way that I am creating the original document is like this.

```
PdfReader pdfTemplate = new PdfReader(pathToPdf);
PdfStamper stamper = new PdfStamper(pdfTemplate, output);
stamper.setFormFlattening(true);
AcroFields fields = stamper.getAcroFields();
fields.setField(key, value);
stamper.close();
```

Is there a way to merge using the first piece of code and merge both of the forms together?

Posted on StackOverflow on Feb 17, 2015 <sup>707</sup> by user3585563<sup>708</sup>

Depending on what you want exactly, different scenarios are possible, but in any case: you are doing it wrong. You should use either PdfCopy or PdfSmartCopy to merge documents.

The different scenarios are explained in the following video tutorial<sup>709</sup>.

You can find most of the examples in the iText sandbox<sup>710</sup>.

### Merging different forms (having different fields)

If you want to merge *different* forms without flattening them, you should use PdfCopy as is done in the MergeForms<sup>711</sup> example:

<sup>707</sup><http://stackoverflow.com/questions/28566190/itext-merge-documents-with-acrofields>

<sup>708</sup><http://stackoverflow.com/users/3585563/user3585563>

<sup>709</sup><https://www.youtube.com/watch?v=6YwDME0Fl1c>

<sup>710</sup><http://itextpdf.com/sandbox>

<sup>711</sup><http://itextpdf.com/sandbox/acroforms/MergeForms>

```

public void createPdf(String filename, PdfReader[] readers)
    throws IOException, DocumentException {
    Document document = new Document();
    PdfCopy copy = new PdfCopy(document, new FileOutputStream(filename));
    copy.setMergeFields();
    document.open();
    for (PdfReader reader : readers) {
        copy.addDocument(reader);
    }
    document.close();
    for (PdfReader reader : readers) {
        reader.close();
    }
}
}

```

In this case, `readers` is an array of `PdfReader` instances containing *different* forms (with different field names), hence we use `PdfCopy` and we make sure that we don't forget to use the `setMergeFields()` method, or the fields won't be copied.

### Merging identical forms (having identical fields)

In this case, we need to rename the fields, because we probably want different values on different pages. In PDF a field can only have a single value. If you merge identical forms, you have multiple visualizations of the same field, but *each visualization will show the same value* (because in reality, there is only one field).

Let's take a look at the [MergeForms2<sup>712</sup>](#) example:

```

public void manipulatePdf(String src, String dest)
    throws IOException, DocumentException {
    Document document = new Document();
    PdfCopy copy = new PdfSmartCopy(document, new FileOutputStream(dest));
    copy.setMergeFields();
    document.open();
    List<PdfReader> readers = new ArrayList<PdfReader>();
    for (int i = 0; i < 3; ) {
        PdfReader reader = new PdfReader(renameFields(src, ++i));
        readers.add(reader);
        copy.addDocument(reader);
    }
    document.close();
    for (PdfReader reader : readers) {

```

---

<sup>712</sup><http://itextpdf.com/sandbox/acroforms/MergeForms2>

```

        reader.close();
    }
}

public byte[] renameFields(String src, int i) throws IOException, DocumentException {
    ByteArrayOutputStream baos = new ByteArrayOutputStream();
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, baos);
    AcroFields form = stamper.getAcroFields();
    Set<String> keys = new HashSet<String>(form.getFields().keySet());
    for (String key : keys) {
        form.renameField(key, String.format("%s_%d", key, i));
    }
    stamper.close();
    reader.close();
    return baos.toByteArray();
}

```

As you can see, the `renameFields()` method creates a new document in memory. That document is merged with other documents using `PdfSmartCopy`. If you'd use `PdfCopy` here, your document would be bloated (as we'll soon find out).

### Merging flattened forms

In the [FillFlattenMerge1<sup>713</sup>](#), we fill out the forms using `PdfStamper`. The result is a PDF file that is kept in memory and that is merged using `PdfCopy`. While this example is fine if you'd merge different forms, this is actually an example on **how not to do it** (as explained in the [video tutorial<sup>714</sup>](#)).

The [FillFlattenMerge2<sup>715</sup>](#) shows how to merge identical forms that are filled out and flattened correctly:

```

public void manipulatePdf(String src, String dest)
    throws DocumentException, IOException {
    Document document = new Document();
    PdfCopy copy = new PdfSmartCopy(document, new FileOutputStream(dest));
    document.open();
    ByteArrayOutputStream baos;
    PdfReader reader;
    PdfStamper stamper;
    AcroFields fields;

```

---

<sup>713</sup><http://itextpdf.com/sandbox/acroforms/reporting/FillFlattenMerge1>

<sup>714</sup><https://www.youtube.com/watch?v=6YwDME0Fl1c>

<sup>715</sup><http://itextpdf.com/sandbox/acroforms/reporting/FillFlattenMerge2>

```
 StringTokenizer tokenizer;
BufferedReader br = new BufferedReader(new FileReader(DATA));
String line = br.readLine();
while ((line = br.readLine()) != null) {
    // create a PDF in memory
    baos = new ByteArrayOutputStream();
    reader = new PdfReader(SRC);
    stamper = new PdfStamper(reader, baos);
    fields = stamper.getAcroFields();
    tokenizer = new StringTokenizer(line, ";");
    fields.setField("name", tokenizer.nextToken());
    fields.setField("abbr", tokenizer.nextToken());
    fields.setField("capital", tokenizer.nextToken());
    fields.setField("city", tokenizer.nextToken());
    fields.setField("population", tokenizer.nextToken());
    fields.setField("surface", tokenizer.nextToken());
    fields.setField("timezone1", tokenizer.nextToken());
    fields.setField("timezone2", tokenizer.nextToken());
    fields.setField("dst", tokenizer.nextToken());
    stamper.setFormFlattening(true);
    stamper.close();
    reader.close();
    // add the PDF to PdfCopy
    reader = new PdfReader(baos.toByteArray());
    copy.addDocument(reader);
    reader.close();
}
br.close();
document.close();
}
```

Now it's up to you to decide which scenario fits your specific requirement.

## How to save an .xfdf file as a .pdf file?



I am currently using Excel to populate a PDF file with form fields. It all works but it exports it as an .xfdf file. Does anyone know how I can save it as a .pdf instead?

```

FullFileName = FilePath & "Requisition - " & Trim(MyRecord.CompanyName) & _
    " - " & Year & "-" & Month & "-" & Day & "-" & Hour & "-" & Minute & _
    " - " & Seconds & ".xfdf"
Open FullFileName For Output As #1
Print #1, "<?xml version=""1.0"" encoding=""ISO-8859-1""?>"
Print #1, "<xfdf xmlns=""http://ns.adobe.com/xfdf/"" xml:space=""preserve"">"
Print #1, "<fields>"
Print #1, "<field name=""Submitted by""><value>" +
    Trim(MyRecord.RequestedBy) + "</value></field>"
Print #1, "<field name=""Job Name""><value>" + "Auto Cards" +
    "</value></field>"
Print #1, "<field name=""Shipping address""><value>" +
    Trim(MyRecord.StreetNumber) & " " & Trim(MyRecord.StreetName) & _
    ", Unit " & Trim(MyRecord.Suite) & ", " & Trim(MyRecord.City) & _
    ", " & Trim(MyRecord.Province) & ", " & Trim(MyRecord.PostalCode) +
    "</value></field>"
Print #1, "<field name=""Special Instructions""><value>Print</value></field>"
Print #1, "</fields>"
Print #1, "<f href=""..\Requisition_Fillable.pdf""/>"
Print #1, "</xfdf>"
Close #1      ' Close file.

```

Posted on StackOverflow on May 28, 2015 <sup>[716](#)</sup> by [santaaimonce](#)<sup>[717](#)</sup>

XFDF is the XML flavor of FDF. FDF stands for Forms Data Format. It is a file format to store data without the actual content of a PDF. FDF uses PDF syntax to store this data. XFDF uses XML.

You can't "convert" an FDF or XFDF file to PDF because there is too much information missing. You can import an (X)FDF file into a PDF that has corresponding AcroForm fields.

Please take a look at the [ImportXFDF<sup>718</sup>](#) example. For that example, I wrote a simple XFDF file: [data.xfdf<sup>719</sup>](#):

<sup>716</sup><http://stackoverflow.com/questions/30508966/saving-xfdf-as-pdf>

<sup>717</sup><http://stackoverflow.com/users/3335518/santaaimonce>

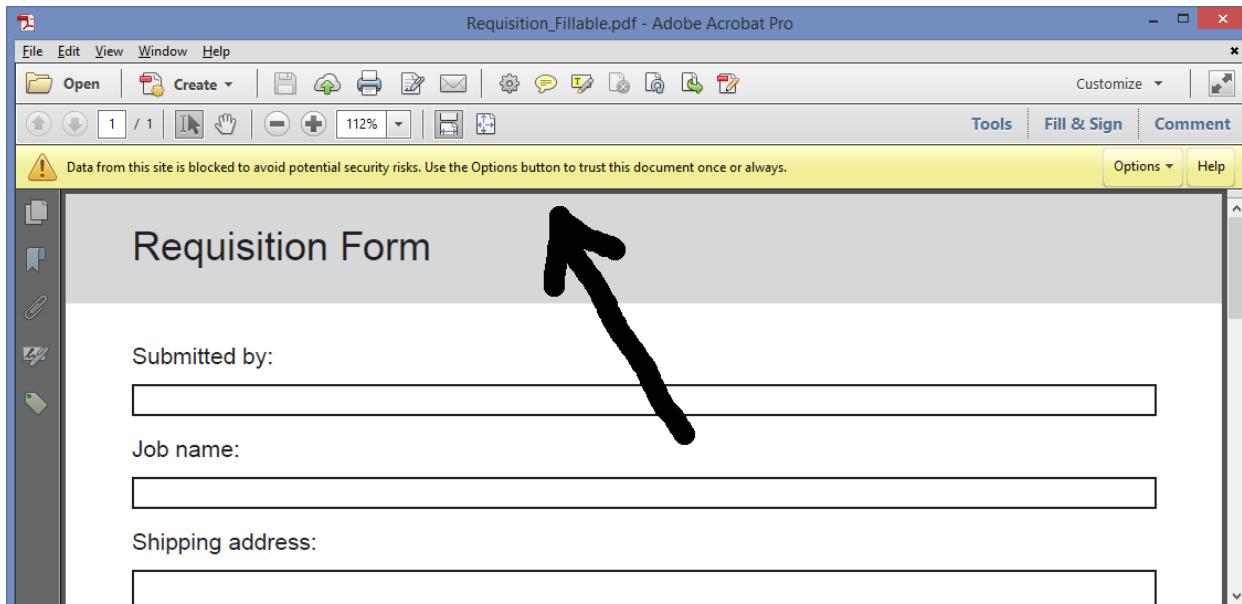
<sup>718</sup><http://itextpdf.com/sandbox/acroforms/ImportXFDF>

<sup>719</sup><http://itextpdf.com/sites/default/files/data.xfdf>

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xfdf xmlns="http://ns.adobe.com/xfdf/" xml:space="preserve">
<fields>
<field name="Submitted by"><value>Santaaimonce</value></field>
<field name="Job name"><value>Developer</value></field>
<field name="Shipping address"><value>Alphabet Street
    Minneapolis
    Minnesota</value></field>
<field name="Special instructions"><value>Use iText to fill out
    interactive form using data stored in XFDF file.
    If you are a C# developer, use iTextSharp.</value></field>
</fields>
<f href="Requisition_Fillable.pdf"/>
</xfdf>
```

This information is not sufficient to create a PDF, but as you can see, there is a reference to a template PDF file: [Requisition\\_Fillable.pdf](#)<sup>720</sup>.

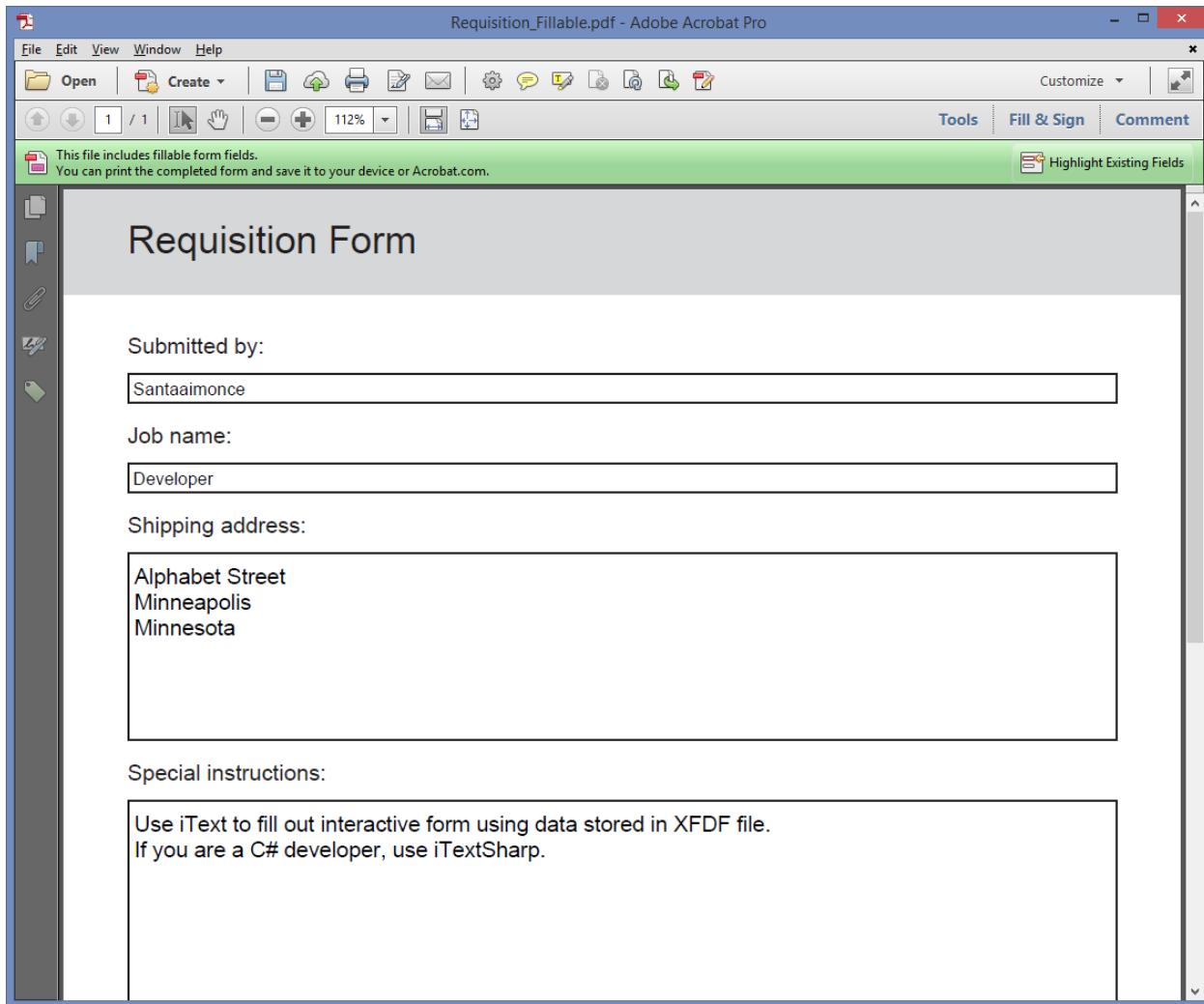
If you put `data.xfdf` and `Requisition_Fillable.pdf` in the same directory and you open `data.xfdf`, you will see the following message:



Import data

Adobe Reader opened the XFDF file, found the reference to the PDF template, opened the template and now asks you if you want to import the data. If you click on the Options button and you allow the import, you will get this result:

<sup>720</sup>[http://itextpdf.com/sites/default/files/Requisition\\_Fillable.pdf](http://itextpdf.com/sites/default/files/Requisition_Fillable.pdf)



#### Imported data

The problem you are experiencing is simple: you don't have the file Requisition\_Fillable.pdf anywhere. You may also want to avoid that people are able to change the data that is filled in.

You can solve this by merging the XFDF file and its template programmatically:

```
protected void fillXfdf(String SRC, String XFDF, String DEST) throws IOException\\
, DocumentException {
    // We receive the XML bytes
    XfdfReader xfdf = new XfdfReader(new FileInputStream(XFDF));
    // We get the corresponding form
    PdfReader reader = new PdfReader(SRC);
    // We create an OutputStream for the new PDF
    FileOutputStream baos = new FileOutputStream(DEST);
    // Now we create the PDF
```

```
PdfStamper stamper = new PdfStamper(reader, baos);
// We alter the fields of the existing PDF
AcroFields fields = stamper.getAcroFields();
fields.setFields(xfdf);
// take away all interactivity
stamper.setFormFlattening(true);
// close the stamper
stamper.close();
reader.close();
}
```

Mind the following line:

```
stamper.setFormFlattening(true);
```

This will *flatten* the form: the fields will no longer be fields, they will just be like any other content. You end up with a regular PDF: `xfdf_merged.pdf`<sup>721</sup>

---

<sup>721</sup>[http://itextpdf.com/sites/default/files/xfdf\\_merged.pdf](http://itextpdf.com/sites/default/files/xfdf_merged.pdf)

The screenshot shows a Adobe Acrobat Pro window titled "xfdf\_merged.pdf - Adobe Acrobat Pro". The main area displays a "Requisition Form" with several fields filled in:

- Submitted by: Santaaimonce
- Job name: Developer
- Shipping address:  
Alphabet Street  
Minneapolis  
Minnesota
- Special instructions:  
Use iText to fill out interactive form using data stored in XFDF file.  
If you are a C# developer, use iTextSharp.

Flattened form data

Obviously this will only work if the form (in this case `Requisition_Fillable.pdf`) corresponds with the data in the XFDF stream. For instance: my template will work with your data for the fields "Submitted by" and "Shipping address". It won't work for the fields "Job Name" and "Special Instructions", because those fields don't exist in my template. In my template those fields are named "Job name" and "Special instructions"; do you see the lower cases instead of the upper case?

If you want to use my template, you'll have to change those two field names, either in your XFDF file or in the PDF template. Also: my example is written in Java. As you are working in VB, you'll have to use the .NET version of iText, which is called iTextSharp. An experienced VBA developer shouldn't have any problem porting my example from Java to VB. Just think of the Java as if it were pseudo-code.

**DISCLAIMER:** the above example requires iText. I am the CEO of the iText Group and the example was taken from my book "iText in Action - Second Edition".

## How to edit a PDF embedded in the browser and save the PDF directly to server?



I have this workflow.

1. Load a PDF containing form fields into the browser,
2. A user fills out the form,
3. A user clicks a 'Submit' button to save the pdf.

What I would like to do in #3 is to collect all the data associated with the form fields and save the data to a database table on the server. I don't want a user to save the PDF to his/her local computer and upload it to the server. I would like to make it more user-friendly.

I am going to use Java/JSP/Servlet in the server-side. I looked into the iText which seems to be popular/well-known for handling a PDF file, but iText seems to be used for generating/editing a PDF. I am not sure if there is any way to have a feature being able to edit a PDF embedded in the browser and save to the database.

Is there any software providing some kind feature which I can inject some kind of script which can capture a user's submit? I know that PDF is not a front-end scripting language, but I am just asking. I was going to create a HTML form which looks like this PDF and populate it into the PDF when a user clicks 'submit' button, but as I said, I would like to make it more user-friendly.

I'd appreciated if there is anybody who has seen this type of feature or has done it gives me some resources or tip.

Posted on StackOverflow on Mar 18, 2015 <sup>722</sup> by user826323<sup>723</sup>

My assumption: you have an interactive PDF document with AcroForm fields similar to [submit\\_me.pdf](#)<sup>724</sup>:

<sup>722</sup><http://stackoverflow.com/questions/29113588/edit-pdf-embedded-in-the-browser-and-save-the-pdf-directly-to-server>

<sup>723</sup><http://stackoverflow.com/users/826323/user826323>

<sup>724</sup>[http://itextpdf.com:8180/book/submit\\_me.pdf](http://itextpdf.com:8180/book/submit_me.pdf)

This file includes fillable form fields.  
You can print the completed form and save it to your device or Acrobat.com.

Fill & Sign | Comment | Highlight Existing Fields

Your name:		
Login:		
Your motivation:		

**POST**      **FDF**      **XFDF**      **RESET**

A PDF form shown in a browser

The main difference, is that I have different buttons on the form:

- POST will post the data I fill out as if the PDF was an HTML form,
- FDF will post the data to the server in the Forms Data Format, a format that is very similar to PDF, but that only contains the data pairs, not the actual form,
- XFDF will post the data to the server in the XML Forms Data Format, which is the XML version of the Forms Data Format.
- RESET will reset all the fields to their initial value.

The [SubmitForm<sup>725</sup>](#) example shows how the buttons were added to an existing form. Note that there are more options than the ones described in this example.

For instance:

- Instead of using the POST method, you can also use GET by tweaking some of the parameters,
- You can also submit the complete PDF (data + form) to the server, but this will only work if the end user has Adobe Acrobat as plug-in. This won't work if the plug-in is merely Adobe Reader,
- ...

To show you what to expect when you commit a form, I wrote the [ShowData<sup>726</sup>](#) servlet. This servlet returns the bytes that are sent to the server.

In case of a POST:

<sup>725</sup><http://itextpdf.com/examples/ii.php?id=168>

<sup>726</sup><http://itextpdf.com/examples/ii.php?id=169>

```
personal.loginname=jdoe&personal.name=John+Doe&personal.password=test&personal.r\
eason=reason&post.x=29&post.y=7
```

Note that I also defined the button in such a way that the coordinate of my click is passed to the server. You probably don't need this.

In case of FDF:

```
%FDF-1.2
%âäïÓ
1 0 obj
<</Fields[<</T(FDF)>><</Kids[<</T(loginname)/V(jdoe)>><</T(name)/V(John Doe)>><</T(password)/V(test)>><</T(reason)/V(Reason)>>]>>/T(personal)>>] /ID[<EF0089E\16ED50F11CB6057A700B9046E><1205D069D1D6AE37665B6FF7EEA65414>] >>/Type/Catalog>>
endobj
trailer
<</Root 1 0 R>>
%%EOF
```

In case of XFDF:

```
<?xml version="1.0" encoding="UTF-8"?>
<xfdf xmlns="http://ns.adobe.com/xfdf/" xml:space="preserve"
><f href="http://itextpdf.com:8180/book/submit_me.pdf"
/><fields
><field name="XFDF"
/><field name="personal"
><field name="loginname"
><value
>jdoe</value
></field
><field name="name"
><value
>John Doe</value
></field
><field name="password"
><value
>test</value
></field
><field name="reason"
><value
>Reason</value
>
```

```

></field
></field
></fields
><ids original="EF0089E16ED50F11CB6057A700B9046E" modified="1205D069D1D6AE37665B\
6FF7EEA65414"
/></xfdf
>

```

In an ideal world, this would be your solution. It is described in ISO-32000-1 which is the world-wide standard for PDF. However: many people started using crappy PDF viewers that do not support this functionality, so if you want to use this solution, you'll have to make sure that people use a decent PDF viewer as their browser plug-in.

## How to send a ‘success’ response back to Acrobat Reader from a java servlet?



I am generating an editable PDF form at the server side and sending it to the browser. At the client side, I am saving the PDF and adding the fields and submitting it via Acrobat Reader. Now once the servlet has read the form fields, I wan to send a success response back to the Acrobat Reader letting the user know that the form has been successfully submitted.

How do I send a response back to the Acrobat Reader as a java servlet response?

Posted on StackOverflow on Jul 21, 2014 <sup>[727](#)</sup> by user3207455<sup>[728](#)</sup>

What do you want the end user to see?

Some options:

1. Usually, I send back simple HTML saying “Thank you for submitting your form.” When the form is archived, I add a link to the filled out form so that people can check what they’ve filled in.
2. You could return the filled out form in PDF, but add some document-level JavaScript that opens an alert that says “This form was submitted on 2014-07-21”. If you don’t like the JavaScript (it can be annoying to see such an alert), why not stamp that message on the filled out form.
3. When I don’t want anything to happen, I send a response code “204 No Content” to the browser. In that case, nothing happens. That’s not what you want, but maybe you can return “200 OK” or “202 Accepted”. The end user won’t see much, but the browser will know what happened. More info can be found on the [W3C site](#)<sup>[729](#)</sup>.

<sup>727</sup><http://stackoverflow.com/questions/24862698/sending-success-response-back-to-acrobat-reader-from-java-servlet>

<sup>728</sup><http://stackoverflow.com/users/3207455/user3207455>

<sup>729</sup><http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>

In many cases, it's common to send a mail with either the link mentioned in (1) or the actual filled-out PDF.

Note that option (1) is only valid when you submit the form from a browser, using Adobe Reader / Acrobat as a plug-in. When using Adobe Reader / Acrobat as a standalone application, you'll have to use option (2) or (3), because Adobe Reader / Acrobat can't interpret HTML and will throw an exception saying that it has received content that can't be read.



Thanks for your response. But if I have to send a javascript alert as a response to the stand-alone acrobat reader, what would be the content-type for the response? Would it be application/pdf or application/vnd.fdf?

AFAIK both are possible. Note that you can also use an annotation instead of JavaScript.

## Why do I get an error saying that “use of extended features is no longer available”?



I am stamping text on existing PDF document using this code:

```
PdfReader reader = new PdfReader("/Users/simple-user/Downloads/acroform.pdf");
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(
    "/Users/simple-user/Downloads/acroform2.pdf"));
BaseFont bf = BaseFont.createFont(BaseFont.HELVETICA, BaseFont.CP1252,
    BaseFont.NOT_EMBEDDED);
PdfContentByte over = stamper.getOverContent(1);
over.beginText();
over.setFontAndSize(bf, 10);
over.setTextMatrix(107, 107);
over.showText("page updated");
over.endText();
stamper.close();
```

This works for all PDFs, except for PDFs with forms. When I open a document after adding this text, I get a lengthy error message about reader extensions.

Posted on StackOverflow on Jun 8, 2015 <sup>730</sup> by IowA<sup>731</sup>

Your diagnosis is wrong. The *problem* is not related to the presence of AcroForms. The *problem* is related to whether or not your document is **Reader Enabled**. Reader-enabling can only be done

<sup>730</sup><http://stackoverflow.com/questions/30707967/pdf-with-acroform-editing-using-itext>

<sup>731</sup><http://stackoverflow.com/users/1955569/iowa>

using Adobe software. It is a process that requires a digital signature using a private key from Adobe. When a valid signature is present, specific functionality (as defined in the usage rights when signing) is unlocked in Adobe Reader.

You change the content of such a PDF, hence you break the signature. Breaking this signature is what causes the ugly error message:

This document enabled extended features in Adobe Acrobat Reader DC. The document has been changed since it was created and use of extended features is no longer available. Please contact the author for the original version of this document.

There are two ways to avoid this error message:

1. Remove the usage rights. This will result in a form that is no longer Reader enabled. For instance: if the creator of the document allowed that the filled out form could be saved locally, this will no longer be possible after removing the usage rights.
2. Fill out the form in append mode. This will result in a bigger file size, but Reader enabling will be preserved.

Removing usage rights is done like this:

```
PdfReader reader = new PdfReader(old_file);
if (reader.hasUsageRights()) {
    reader.removeUsageRights();
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(new_file));
    stamper.close();
}
reader.close();
```

Using iText in append mode is done like this:

```
PdfReader reader = new PdfReader(src);
PdfStamper stamper =
    new PdfStamper(reader, new FileOutputStream(dest), '\0', true);
stamper.close();
reader.close();
```

Note the extra parameters in PdfStamper. They cause iText to preserve the original bytes of the file and avoid breaking the signature.

**Extra remark:** you are adding text the hard way. You'd make it easier on yourself if you'd use the ColumnText.showTextAligned() method to add text.

## How to fill XFA form using iText without breaking usage rights?



This is my code:

```
using (FileStream pdf = new FileStream("C:/test.pdf", FileMode.Open))
using (FileStream xml = new FileStream("C:/test.xml", FileMode.Open))
using (FileStream filledPdf = new FileStream("C:/test_f.pdf", FileMode.Create))
{
    PdfReader pdfReader = new PdfReader(pdf);
    PdfStamper stamper = new PdfStamper(pdfReader, filledPdf);
    stamper.AcroFields.Xfa.FillXfaForm(xml);
    stamper.Close();
    pdfReader.Close();
}
```

This code throws no exception and everything seems to be OK, but if I open filled pdf, Adobe Reader says something like this:

This document enabled extended features. This document was changed since it was created and using extended features isn't possible anymore.

If I choose xml manually by clicking 'Import data' from Adobe Reader, form is filled properly, so I guess there is no error in xml.

Posted on StackOverflow on Oct 29, 2014 <sup>[732](#)</sup> by paldir<sup>[733](#)</sup>

You are not creating the `PdfStamper` object correctly. Use:

```
PdfStamper stamper = new PdfStamper(pdfReader, filledPdf, '\0', true)
```

In your code, you are not using `PdfStamper` in append mode. This means that iText will reorganize the different objects in your PDF. Usually that isn't a problem.

However: your PDF is Reader-enabled, which means that your PDF is digitally signed using a private key owned by Adobe. By reorganizing the objects inside the PDF, that signature is broken. This is made clear by the message you already mentioned:

This document enabled extended features. This document was changed since it was created and using extended features isn't possible anymore.

---

<sup>732</sup><http://stackoverflow.com/questions/26629498/how-to-fill-xfa-form-using-itext>

<sup>733</sup><http://stackoverflow.com/users/4148435/paldir>

To avoid breaking the signature, you need to use `PdfStamper` in append mode. Instead of reorganizing the original content, iText will now keep the original file intact and append new content after the end of the original file.

## How to deal with missing characters in the default font of a text field?



We are using iText to create PDFs using a custom font and I am running into an issue with the unicode character 2120 (SM, Service Mark). The problem is the glyph is not in the custom font. Is there a way I can specify a fallback font for a field in a PDF? We tried adding a text field with Verdana so that the form had the secondary font embedded in it but that didn't seem to help.

Posted on StackOverflow on Nov 7, 2013 <sup>734</sup> by El Hombre<sup>735</sup>

First you just need to add substitutions fonts to the form:

```
AcroFields form = stamper.getAcroFields();
form.addSubstitutionFont(bf1);
form.addSubstitutionFont(bf2);
```

Now the font defined in the form has preference, but if that font can't show a specific glyph, it will look at `bf1`, then at `bf2` and so on.

---

<sup>734</sup><http://stackoverflow.com/questions/19845688/missing-character-in-custom-font>

<sup>735</sup><http://stackoverflow.com/users/912570/el-hombre>

## How to make sure a check box is printed?



I am new to iTextSharp, and would appreciate some help. I am creating some checkboxes, an example of the code is below:

```
PdfFormField checkbox1 = PdfFormField.CreateCheckBox(writer);
checkbox1.SetWidget(new Rectangle(524, 600, 540, 616),
    PdfAnnotation.HIGHLIGHT_INVERT);
checkbox1.ValueAsName = ("Off");
checkbox1.AppearanceState = ("Off");
checkbox1.FieldName = ("UsersNo");
checkbox1.SetAppearance(PdfAnnotation.APPEARANCE_NORMAL, "Off", chkOff);
checkbox1.SetAppearance(PdfAnnotation.APPEARANCE_NORMAL, "On", chkOn);
writer.AddAnnotation(checkbox1);
```

Everything looks great and is working well, until it comes to printing the PDF. When I click on *File* and *Print*, the check boxes don't show in the print preview, and also don't print.

Posted on StackOverflow on Mar 23, 2015 <sup>[736](#)</sup> by user1176737<sup>[737](#)</sup>

There are two ways to create a check box.

1. There is the easy way, using the RadioCheckField class.
2. There is the hard way, using the PdfFormField class.

For some reason you have chosen the hard way, and you are now complaining that the visibility is set to “Show on screen, not in print” instead of “Show on screen and in print”.

- The former (“Show on screen, not in print”) is the default visibility setting when you create a check box the hard way. It corresponds with no flags being set.
- The latter (“Show on screen and in print”) is the default when creating a check box the easy way. In this case, the following flag is set automatically for your convenience.

As you have chosen the hard way to create a check box, you need to add the line that add the “Print flag” to your code yourself:

```
checkbox1.Flags = PdfAnnotation.FLAGS_PRINT;
```



Perfect, works a treat, many thanks for that Bruno.

---

<sup>736</sup><http://stackoverflow.com/questions/29210183/itextsharp-checkbox-wont-print>

<sup>737</sup><http://stackoverflow.com/users/1176737/user1176737>

## How to check a check box?



I've tried so many different ways, but I can't get the check box to be checked! Here's what I've tried:

```
var reader = new iTextSharp.text.pdf.PdfReader(originalFormLocation);
using (var stamper = new iTextSharp.text.pdf.PdfStamper(reader, ms)) {
    var formFields = stamper.AcroFields;
    formFields.SetField("IsNo", "1");
    formFields.SetField("IsNo", "true");
    formFields.SetField("IsNo", "On");
}
```

None of them work. Any ideas?

Posted on StackOverflow on Oct 31, 2013 <sup>738</sup> by user948060<sup>739</sup>

You shouldn't “guess” for the possible values. You need to use a value that is stored in the PDF. Try the [CheckBoxValues<sup>740</sup>](#) example to find these possible values:

```
public String getCheckboxValue(String src, String name) throws IOException {
    PdfReader reader = new PdfReader(SRC);
    AcroFields fields = reader.getAcroFields();
    // CP_1 is the name of a check box field
    String[] values = fields.getAppearanceStates("IsNo");
    StringBuffer sb = new StringBuffer();
    for (String value : values) {
        sb.append(value);
        sb.append('\n');
    }
    return sb.toString();
}
```

Or take a look at the PDF using [RUPS<sup>741</sup>](#). Go to the widget annotation and look for the normal (/N) appearance (AP) states. In my example they are /Off and /Yes:

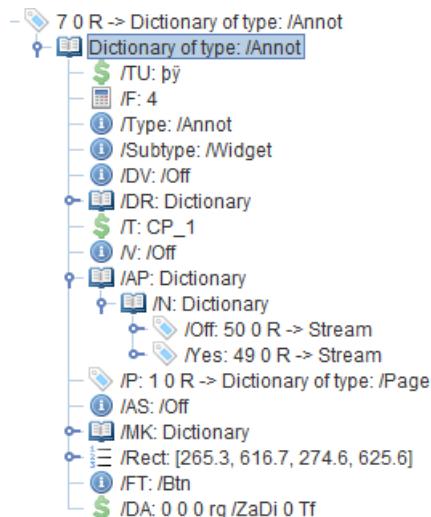
---

<sup>738</sup><http://stackoverflow.com/questions/19698771/checking-off-pdf-checkbox-with-itextsharp>

<sup>739</sup><http://stackoverflow.com/users/948060/user948060>

<sup>740</sup><http://itextpdf.com/sandbox/acroforms/CheckBoxValues>

<sup>741</sup>[http://itextpdf.com/product/itext\\_rups](http://itextpdf.com/product/itext_rups)



screen shot

## How to distribute the radio buttons of a radio field across multiple PdfPCells?



I'd like to make a PdfPTable with multiple rows. In each row I'd like to have a Radio button in the first cell and descriptive text in the second cell. I'd like all the radio buttons to be a part of the same radio group. I've used PdfPCell's `setCellEvent()` method and I've created my own custom cell events to render text fields and check boxes in PdfPTables. However, I can't seem to figure out how to do it with radio buttons / radio groups. Is this possible with iText? Does anyone have an example?

Posted on StackOverflow on Apr 1, 2015 <sup>[742](#)</sup> by corestruct00<sup>[743](#)</sup>

Please take a look at the [CreateRadioInTable](#)<sup>[744](#)</sup> example.

In this example, we create a PdfFormField for the radio group and we add it after constructing and adding the table:

<sup>742</sup><http://stackoverflow.com/questions/29393050/itext-radiogroup-radio-buttons-across-multiple-pdfpcells>

<sup>743</sup><http://stackoverflow.com/users/3980635/corestruct00>

<sup>744</sup><http://itextpdf.com/sandbox/acroforms/CreateRadioInTable>

```

PdfFormField radiogroup = PdfFormField.createRadioButton(writer, true);
radiogroup.setFieldName("Language");
PdfPTable table = new PdfPTable(2);
// add cells
document.add(table);
writer.addAnnotation(radiogroup);

```

When we create the cells for the radio buttons, we add an event, for instance:

```
cell.setCellEvent(new MyCellField(radiogroup, "english"));
```

The event looks like this:

```

class MyCellField implements PdfPCellEvent {
    protected PdfFormField radiogroup;
    protected String value;
    public MyCellField(PdfFormField radiogroup, String value) {
        this.radiogroup = radiogroup;
        this.value = value;
    }
    public void cellLayout(PdfPCell cell, Rectangle rectangle, PdfContentByte[] \ 
canvases) {
        final PdfWriter writer = canvases[0].getPdfWriter();
        RadioCheckField radio = new RadioCheckField(writer, rectangle, null, val\ 
ue);
        try {
            radiogroup.addKid(radio.getRadioField());
        } catch (final IOException ioe) {
            throw new ExceptionConverter(ioe);
        } catch (final DocumentException de) {
            throw new ExceptionConverter(de);
        }
    }
}

```



Taking this a bit further:

If you're nesting a table of radio buttons (radio group) in another table you'll have to change the following from Bruno's example:

instead of:

```
document.add(table);
writer.addAnnotation(radiogroup);
```

use (assuming you created a parent table and a PdfPCell in that table called parentCell)

```
parentCell.addElement(table);
parentCell.setCellEvent(new RadioGroupCellEvent(radioGroup));
```

with a parent cell event like so

```
public class RadioGroupCellEvent implements PdfPCellEvent {
    private PdfFormField radioGroup;

    public RadioGroupCellEvent(PdfFormField radioGroup) {
        this.radioGroup = radioGroup;
    }

    @Override
    public void cellLayout(PdfPCell cell, Rectangle position, PdfContentByte[] canvases) {
        PdfWriter writer = canvases[0].getPdfWriter();
        writer.addAnnotation(radioGroup);
    }
}
```

## How to find out which fields are required?



I'm starting with new functionality in my android application which will help to fill certain PDF forms. I found out that the best solution will be to use iText library. I can read file, and read AcroFields from document but is there any possibility to find out that specific field is marked as required?

Posted on StackOverflow on Feb 17, 2013 <sup>[745](#)</sup> by weedget<sup>[746](#)</sup>

Please take a look at section 13.3.4 of “iText in Action - Second Edition”, entitled “AcroForms revisited”. Listing 13.15 shows a code snippet from the [InspectForm](#)<sup>[747](#)</sup> example that checks whether or not a field is a password or a multi-line field.

With some minor changes, you can adapt that example to check for required fields:

---

<sup>745</sup><http://stackoverflow.com/questions/21838945/finding-out-required-fields-to-fill-in-pdf-file>

<sup>746</sup><http://stackoverflow.com/users/2371671/weedget>

<sup>747</sup><http://itextpdf.com/examples/iiapdf?id=237>

```

for (Map.Entry<String,AcroFields.Item> entry : fields.entrySet()) {
    out.write(entry.getKey());
    item = entry.getValue();
    dict = item.getMerged(0);
    flags = dict.getAsNumber(PdfName.FF);
    if (flags != null && (flags.intValue() & BaseField.REQUIRED) > 0)
        out.write(" -> required\n");
}

```

## How to make a field *not* required?



I am trying to make a field not required using iText. I know that I can make a field required using something like below statement:

```
formFields.setFieldProperty(field, "setfflags", PdfFormField.FF_REQUIRED, null);
```

Is there anything similar to make an existing field not required?

Posted on StackOverflow on Dec 1, 2014<sup>748</sup> by user2296988<sup>749</sup>

By default, the “required field” flag is not set when creating a form field. As you indicate in your question, you can use the `setFieldProperty()` method to set this flag:

```
formFields.setFieldProperty(field, "setfflags", PdfFormField.FF_REQUIRED, null);
```

Based on your question, I assume that you have an existing PDF where you have fields for which this flag is already set. You can remove this flag (remove the bit from the bit set), by changing “`setfflags`” into “`clrfflags`”:

```
formFields.setFieldProperty(field, "clrfflags", PdfFormField.FF_REQUIRED, null);
```

In this context `clr` stands for *clear* and `fflags` stands for *field flags* (not to be confused with `clrfflags` which will clear flags in the widget annotation).

---

<sup>748</sup><http://stackoverflow.com/questions/27235259/can-i-make-a-field-not-required-in-pdf-using-itext>

<sup>749</sup><http://stackoverflow.com/users/2296988/user2296988>

## How to get the number of characters in a field?



I am trying to find out which formats a date of birth should be in a empty field in a PDF using iText. I can stamp the field with a value but then I need to know how many digits are expected for the date of birth. I figured that if I could get the length of the field, I could know which format to use, because the formats can be:

YYYYMMDDNNNN (14 digits)

YYYYMMDD (10 digits)

YYMMDD (8 digits)

The field I'm talking about has a fixed number of digits, if I stamp the field with too many numbers, then the numbers that fall outside the field disappear. I can share my PDF if necessary.

Posted on StackOverflow on Jul 15, 2014 <sup>750</sup> by Johan Nordli<sup>751</sup>

You're in luck. I've examined your PDF and the field for the date of birth has a /MaxLen entry which means that you can find out the maximum length of the field. In the following screen shot, you can see the properties of the field / annotation dictionary for field Falt\_41 (which can be used to add the *Sökandens personnummer*):

The screenshot shows the RUPS (iText RUPS) interface. The left pane displays the structure of a PDF field, specifically for field Falt\_41. The dictionary includes entries for `/T: Falt_99`, `/P: 1 0 R -> Dictionary of type: /Page`, `/AP: Dictionary`, `/N: 64 0 R -> Stream`, `157 0 R -> Dictionary`, `Dictionary`, `/MaxLen: 12`, `/Ff: 29360128`, `/FT: /Tx`, `/DA: /CourierNew 10.67 Tf 0.00 g`, `/T: Falt_41`, `/Kids: [559 0 R, 124 0 R]`, `135 0 R -> Dictionary of type: /Annot`, `/DR: 155 0 R -> Dictionary`, `/DA: /Helv 0 Tf 0 g`, `/SigFlags: 2`, `/Metadata: 87 0 R -> Stream`, and `/Pages: 352 0 R -> Dictionary of type: /Pages`. The right pane shows a list of pages, with page Falt\_41 selected. The top bar indicates the version: iText RUPS iText® 5.5.1 ©2000-2014 iText Group NV (AGPL-version).

RUPS screen shot

<sup>750</sup><http://stackoverflow.com/questions/24754825/itext-get-number-of-characters-in-a-field>

<sup>751</sup><http://stackoverflow.com/users/2294456/johan-nordli>

As you can see, this field can contain a maximum of 12 characters. Moreover, the /Ff (fields flags) value is 29360128 or binary value: 11100000000000000000000000. This means that the following flags are active: **do not spell check**, **do not scroll**, and **comb**. The **comb** flag makes that whatever you enter, the characters will be evenly distributed over the available width. In this case, every character will take 1/12 of the available width.

Now how do you retrieve the /MaxLen value? This is some code written from memory:

```
AcroFields form = reader.getAcroFields();
AcroFields.Item item = form.getFieldItem("Falt_41");
PdfDictionary field = item.getMerged(0);
PdfNumber maxlen = field.getAsNumber(PdfName.MAXLEN);
```

Now you can get the int value of maxlen.

**Important note:** not every field has a /MaxLen value.

## How to align AcroFields?



I'm using iText to populate the data to PDF templates, which are created in OpenOffice. The form is populating fine; I'm getting proper PDF, but I want to change the alignment of some fields. using the following code.

```
fields.setFieldProperty(fieldName, "fflags", PdfFormField.Q_LEFT, null);
```

Posted on StackOverflow on Jun 19, 2014<sup>752</sup> by Rajesh<sup>753</sup>

The quadding isn't part of the field flags as you wrongly assumed. It's an entry of the widget annotation dictionary. This is how you change it:

---

<sup>752</sup><http://stackoverflow.com/questions/24301578/align-acrofields-in-java>

<sup>753</sup><http://stackoverflow.com/users/1653759/rajesh>

```
AcroFields form = stamper.getAcroFields();
AcroFields.Item item;
item = form.getFieldItem("fieldLeft");
item.getMerged(0).put(PdfName.Q, new PdfNumber(PdfFormField.Q_LEFT));
item = form.getFieldItem("fieldCenter");
item.getMerged(0).put(PdfName.Q, new PdfNumber(PdfFormField.Q_CENTER));
item = form.getFieldItem("fieldRight");
item.getMerged(0).put(PdfName.Q, new PdfNumber(PdfFormField.Q_RIGHT));
```

## How to change the text color of an AcroForm field?



I have a PDF with named fields I created with Adobe LifeCycle. I am able to fill the fields using iTextSharp but when I change the text color for a field it does not change. I really don't know why this is so. Here is my code:

```
form.SetField("name", "Michael Okpara");
form.SetField("session", "2014/2015");
form.SetField("term", "1st Term");
form.SetFieldProperty("name", "textcolor", BaseColor.RED, null);
form.RegenerateField("name");
```

Posted on StackOverflow on Nov 21, 2014<sup>754</sup> by Okolie Solomon<sup>755</sup>

If your form is created using Adobe LifeCycle, then there are two options:

- **You have a pure XFA form.** XFA stands for the XML Forms Architecture and your PDF is nothing more than a container of an XML stream. There is hardly any PDF syntax in the document and there are no AcroForm fields. I don't think this is the case, because you are still able to fill out the fields (which wouldn't work if you had a pure XFA form).
- **You have a hybrid form.** In this case, the form is described twice inside the PDF file: once using an XML stream (XFA) and once using PDF syntax (AcroForm). iText will fill out the fields in both descriptions, but the XFA description gets preference when rendering the document. Changing the color of a field (or other properties) would require changing the XML and iText(Sharp) can not do that.

If I may make an educated guess, I would say that you have a hybrid form and that you are only changing the text color of the AcroForm field without changing the text color in the XFA field (which is really hard to achieve).

Please try adding this line:

---

<sup>754</sup><http://stackoverflow.com/questions/27057187/setting-acrofield-text-color-in-itextsharp>

<sup>755</sup><http://stackoverflow.com/users/2491068/okolie-solomon>

```
form.RemoveXfa();
```

This will remove the XFA stream, resulting in a form that only keeps the AcroForm description.

I have written a small example named [RemoveXFA<sup>756</sup>](#) using the form you shared to demonstrate this:

```
public void manipulatePdf(String src, String dest)
    throws DocumentException, IOException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    AcroFields form = stamper.getAcroFields();
    form.removeXfa();
    Map<String, AcroFields.Item> fields = form.getFields();
    for (String name : fields.keySet()) {
        if (name.indexOf("Total") > 0)
            form.setProperty(name, "textcolor", BaseColor.RED, null);
        form.setField(name, "X");
    }
    stamper.close();
    reader.close();
}
```

In this example, I remove the XFA stream and I look over all the remaining AcroFields. I change the textcolor of all the fields with the word “Total” in their name, and I fill out every field with an “X”.

The result looks like this: [reportcard.pdf<sup>757</sup>](#)

---

<sup>756</sup><http://itextpdf.com/sandbox/acroforms/RemoveXFA>

<sup>757</sup><http://itextpdf.com/sites/default/files/reportcard.pdf>

The screenshot shows the Adobe Acrobat interface with a toolbar at the top and a sidebar on the left. A green banner at the top states: "This file includes fillable form fields. You can print the completed form and save it to your device or Acrobat.com." The main content area contains the following text and table:

**XYZ GROUP OF SCHOOLS**

NAME: X  
SESSION: X  
TERM: X  
CLASS: X

	1 <sup>st</sup> TEST	2 <sup>nd</sup> TEST	3 <sup>rd</sup> TEST	EXAM	TOTAL	POSITION
X	X	X	X	X	X	X
X	X	X	X	X	X	X
X	X	X	X	X	X	X
X	X	X	X	X	X	X
X	X	X	X	X	X	X
v	v	v	v	v	v	v

Resulting PDF

All the fields show the letter “X”, but the fields in the TOTAL column are written in red.

## How to get the color properties of an AcroForm field?



I am using iText to read a PDF file. I have 20 AcroForm text fields in my pdf with different color fill properties. I'm not able to read these properties. Is there any way we can get color information from the fields?

EDIT: I created the AcroForm fields in the PDF using the following Adobe Javascript

```
var oFld = this.addField("nameOfField", "button", 0, fldRect);
if (oFld != null) {
    oFld.buttonSetCaption("");
    oFld.borderColor = border.s;
    oFld.fillColor = color.gray;
    oFld.textColor = color.white;
    oFld.lineWidth = 1;
}
```

Posted on StackOverflow on Nov 25, 2013 <sup>758</sup> by MaheshVarma<sup>759</sup>

Chapter 8<sup>760</sup> of my book discusses AcroForm fields from a rather high level. If you want to dig deeper, you need chapter 13<sup>761</sup>. On page 449, table 13.11 lists the different `AcroFields.Item` methods. As you know a form field is described using a form dictionary. The visual representation(s) of a field is (or are) described using one or more widget annotations. You're looking for a property of the appearance, so you need the annotation dictionary.

You also know that the field dictionary and the widget dictionary are often merged when one field corresponds with one widget annotation, and that's why the `AcroFields.Item` class has a method called `getMerged()`. For every widget annotation of a specific field, it returns the merged properties of the field and the widget annotation.

That's the theory. Let's look at an example: [InspectForm](#)<sup>762</sup>

---

<sup>758</sup><http://stackoverflow.com/questions/20186868/how-to-get-acrofield-properties-using-itext>

<sup>759</sup><http://stackoverflow.com/users/2416228/maheshvarma>

<sup>760</sup><http://itextpdf.com/book/chapter.php?id=8>

<sup>761</sup><http://itextpdf.com/book/chapter.php?id=13>

<sup>762</sup><http://itextpdf.com/examples/iaa.php?id=237>

```
Map<String,AcroFields.Item> fields = form.getFields();
AcroFields.Item item;
PdfDictionary dict;
int flags;
for (Map.Entry<String,AcroFields.Item> entry : fields.entrySet()) {
    out.write(entry.getKey());
    item = entry.getValue();
    dict = item.getMerged(0);
    // inspect dict
}
```

In the example, we inspect the field flags (/FF), which are properties of the field dictionary. You are interested in appearance characteristics, so I guess you'll want to inspect the /MK entry, which is (ISO-32000-1 Table 188):

An appearance characteristics dictionary (see Table 189) that shall be used in constructing a dynamic appearance stream specifying the annotation's visual presentation on the page. The name MK for this entry is of historical significance only and has no direct meaning.

You'll need to look at table 189 to find out the specific attributes you want:

**R integer (Optional):** The number of degrees by which the widget annotation shall be rotated counterclockwise relative to the page. The value shall be a multiple of 90. Default value: 0.

**BC array (Optional):** An array of numbers that shall be in the range 0.0 to 1.0 specifying the colour of the widget annotation's border. The number of array elements determines the colour space in which the colour shall be defined: 0 No colour; transparent 1 DeviceGray 3 DeviceRGB 4 DeviceCMYK

**BG array (Optional):** An array of numbers that shall be in the range 0.0 to 1.0 specifying the colour of the widget annotation's background. The number of array elements shall determine the colour space, as described for BC.

**CA** *text string (Optional; button fields only)*: The widget annotation's normal caption, which shall be displayed when it is not interacting with the user. Unlike the remaining entries listed in this Table, which apply only to widget annotations associated with pushbutton fields (see Pushbuttons in 12.7.4.2, "Button Fields"), the CA entry may be used with any type of button field, including check boxes (see Check Boxes in 12.7.4.2, "Button Fields") and radio buttons (Radio Buttons in 12.7.4.2, "Button Fields").

**RC** *text string (Optional; pushbutton fields only)*: The widget annotation's rollover caption, which shall be displayed when the user rolls the cursor into its active area without pressing the mouse button.

**AC** *text string (Optional; pushbutton fields only)*: The widget annotation's alternate (down) caption, which shall be displayed when the mouse button is pressed within its active area.

When you ask for the fill color, I assume that you're referring to the background color, which means you'll have to look at the BC entry for the colorspace, and at the BG entry for the actual color value.

## How to find the absolute position and dimension of a field?



Given an field name, is it possible to find the absolute position and dimension of that particular field (`getLeft()`, `getTop()`, `getWidth()`, `getHeight()`)?

And is the opposite possible: if I know the position, can I get the name of the field?

Posted on StackOverflow on Apr 16, 2014 <sup>763</sup> by ahairshi<sup>764</sup>

### First part of your question:

Suppose that you have an `AcroFields` instance (`form`), either retrieved from a `PdfReader` (read only) or a `PdfStamper` instance, then you can get the field position of the first widget that corresponds with a specific field name like this:

<sup>763</sup><http://stackoverflow.com/questions/23104809/find-field-absolute-position-and-dimension-by-acrokey>

<sup>764</sup><http://stackoverflow.com/users/472182/ahairshi>

```
Rectangle rectangle = form.getFieldPositions(name).get(0).position;
```

Note that one field can correspond with multiple widgets. For instance, to get the second widget, you need:

```
Rectangle rectangle = form.getFieldPositions(name).get(1).position;
```

Of course: you probably also want to know the page number:

```
int page = form.getFieldPositions(name).get(0).page;
```

### Second part of your question

Fields correspond with widget annotations. If you know the page number of the widget annotation, you could get the page dictionary and inspect the entries of the /Annots array. You'll have to loop over the different annotations, inspecting each annotation's /Rect entry. Once you find a match, you need to crawl the content for the field that corresponds with the annotation. That's more work than can be provided in a code sample on this site.

## How to move an AcroForm field?



I have a process that inserts a table of content into an existing Acroform, and I am able to track where I need to start that content. However, I have existing Acrofields below that point that will need to be moved up or down, based on the height of the tables I insert. With that, how can I change the position of an Acrofield?

Posted on StackOverflow on Nov 7, 2013 <sup>[765](#)</sup> by [cro717](#) <sup>[766](#)</sup>

First some info about fields and their representation on one or more pages. A PDF form can contain a number of fields. Fields have unique names in the sense that one specific field with one specific name has one and one value. Fields are defined using a *field dictionary*.

Each field can have zero, one or more representations in the document. These visual representations are called *widget annotations* and they are defined using an *annotation dictionary*.

Knowing this, your question needs to be rephrased: *how do I change the location of a specific widget annotation of a specific field?*

I've made a sample in Java named [ChangeFieldPosition](#) <sup>[767](#)</sup> in answer to this question. It will be up to you to port it to C#.

You already have the `AcroFields` instance:

---

<sup>765</sup><http://stackoverflow.com/questions/22258598/itextsharp-move-acrofield>

<sup>766</sup><http://stackoverflow.com/users/3381321/cro717>

<sup>767</sup><http://itextpdf.com/sandbox/acroforms/ChangeFieldPosition>

```
AcroFields form = stamper.getAcroFields();
```

What you need now is the `Item` instance for the specific field (in my example: for the field with name "timezone2"):

```
Item item = form.getFieldItem("timezone2");
```

The position is a property of the widget annotation, so you need to ask the `item` for its widget. In the following line, I get the annotation dictionary for the first widget annotation (with index 0):

```
PdfDictionary widget = item.getWidget(0);
```

In most cases there will be only one widget annotation: each field has only one visual representation.

The position of the annotation is an array with four values: `llx`, `lly`, `urx` and `ury`. We can get this array like this:

```
PdfArray rect = widget.getAsArray(PdfName.RECT);
```

In the following line I change the x-value of the upper-right corner (index 2 corresponds with `urx`):

```
rect.set(2, new PdfNumber(rect.getAsNumber(2).floatValue() - 10f));
```

As a result the width of the field is shortened by 10pt.

## How to continue field output on a second page?



I have generated a PDF from a template. The PDF has a field in the middle of it that is of variable length. I am trying to work it so that if the field's content overflows, then the program will use a second instance template as a second page and continue in the same field there. Is this possible?

Posted on StackOverflow on Nov 10, 2014<sup>768</sup> by Carlos Mendieta<sup>769</sup>

This will only work if you flatten the form. I've written a proof of concept where I have a PDF form `src` that has a field named "body":

<sup>768</sup><http://stackoverflow.com/questions/26853894/continue-field-output-on-second-page-with-itextsharp>

<sup>769</sup><http://stackoverflow.com/users/1539292/carlos-mendieta>

```
public void manipulatePdf(String src, String dest)
    throws DocumentException, IOException {
    PdfReader reader = new PdfReader(src);
    Rectangle pagesize = reader.getPageSize(1);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    Paragraph p = new Paragraph();
    p.add(new Chunk("Hello "));
    p.add(new Chunk("World", new Font(FontFamily.HELVETICA, 12, Font.BOLD)));
    AcroFields form = stamper.getAcroFields();
    Rectangle rect = form.getFieldPositions("body").get(0).position;
    int status;
    PdfImportedPage nextPage = null;
    ColumnText column = new ColumnText(stamper.getOverContent(1));
    column.setSimpleColumn(rect);
    int pagecount = 1;
    for (int i = 0; i < 100; ) {
        i++;
        column.addElement(new Paragraph("Hello " + i));
        column.addElement(p);
        status = column.go();
        if (ColumnText.hasMoreText(status)) {
            nextPage = loadPage(nextPage, reader, stamper);
            triggerNewPage(stamper, pagesize, nextPage, column,
                           rect, ++pagecount);
        }
    }
    stamper.setFormFlattening(true);
    stamper.close();
    reader.close();
}

public PdfImportedPage loadPage(
    PdfImportedPage page, PdfReader reader, PdfStamper stamper) {
    if (page == null) {
        return stamper.getImportedPage(reader, 1);
    }
    return page;
}

public void triggerNewPage(PdfStamper stamper, Rectangle pagesize,
    PdfImportedPage page, ColumnText column, Rectangle rect, int pagecount)
    throws DocumentException {
```

```

stamper.insertPage(pagecount, pagesize);
PdfContentByte canvas = stamper.getOverContent(pagecount);
canvas.addTemplate(page, 0, 0);
column.setCanvas(canvas);
column.setSimpleColumn(rect);
column.go();
}

```

As you can see, we create a PdfImportedPage instance and we insert a new page with this page as background. We add the content at the position defined by the field using ColumnText.

## How to add a table on a form (and maybe insert a new page)?



I have two parts to my java project.

- I need to populate the fields of a pdf
- I need to add a table below the populated section on the blank area of the page (and this table needs to be able to roll over to the next page).

I am able to do these things separately (populate the PDF and create a table). But I cannot effectively merge them. I have tried doing doc.add(table) which will result in the table being on the next page of the pdf, which I don't want.

I essentially just need to be able to specify where the table starts on the page (so it wouldn't overlap the existing content) and then stamp the table onto the existing pdf.

Posted on StackOverflow on Feb 18, 2015 <sup>770</sup> by Jennifer<sup>771</sup>

Please take a look at the [AddExtraTable<sup>772</sup>](#) example. It's a simplification of the [AddExtraPage<sup>773</sup>](#) example written in answer to the question [How to continue field output on a second page?<sup>774</sup>](#)

That question is almost an exact duplicate of your question, with as only difference the fact that your requirement is easier to achieve.

I simplified the code like this:

---

<sup>770</sup><http://stackoverflow.com/questions/28590487/adding-table-to-existing-pdf-on-the-same-page-itext>

<sup>771</sup><http://stackoverflow.com/users/4580758/jennifer>

<sup>772</sup><http://itextpdf.com/sandbox/acroforms/AddExtraTable>

<sup>773</sup><http://itextpdf.com/sandbox/acroforms/AddExtraPage>

<sup>774</sup><http://stackoverflow.com/questions/26853894/how-to-continue-field-output-on-a-second-page>

```
public void manipulatePdf(String src, String dest)
    throws DocumentException, IOException {
    PdfReader reader = new PdfReader(src);
    Rectangle pagesize = reader.getPageSize(1);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    AcroFields form = stamper.getAcroFields();
    form.setField("Name", "Jennifer");
    form.setField("Company", "iText's next customer");
    form.setField("Country", "No Man's Land");
    PdfPTable table = new PdfPTable(2);
    table.addCell("#");
    table.addCell("description");
    table.setHeaderRows(1);
    table.setWidths(new int[]{ 1, 15 });
    for (int i = 1; i <= 150; i++) {
        table.addCell(String.valueOf(i));
        table.addCell("test " + i);
    }
    ColumnText column = new ColumnText(stamper.getOverContent(1));
    Rectangle rectPage1 = new Rectangle(36, 36, 559, 540);
    column.setSimpleColumn(rectPage1);
    column.addElement(table);
    int pagecount = 1;
    Rectangle rectPage2 = new Rectangle(36, 36, 559, 806);
    int status = column.go();
    while (ColumnText.hasMoreText(status)) {
        status =
            triggerNewPage(stamper, pagesize, column, rectPage2, ++pagecount);
    }
    stamper.setFormFlattening(true);
    stamper.close();
    reader.close();
}
public int triggerNewPage(PdfStamper stamper,
    Rectangle pagesize, ColumnText column, Rectangle rect, int pagecount)
    throws DocumentException {
    stamper.insertPage(pagecount, pagesize);
    PdfContentByte canvas = stamper.getOverContent(pagecount);
    column.setCanvas(canvas);
    column.setSimpleColumn(rect);
    return column.go();
}
```

As you can see, the main differences are:

1. We create a `rectPage1` for the first page and a `rectPage2` for page 2 and all pages that follow. That's because we don't need a full page on the first page.
2. We don't need to load a `PdfImportedPage`, instead we're just adding blank pages of the same size as the first page.

Possible improvements: I hardcoded the `Rectangle` instances. It goes without saying that `rect1Page` depends on the location of your original form. I also hardcoded `rect2Page`. If I had more time, I would calculate `rect2Page` based on the `pagesize` value.

## How to add an image to an AcroForm field?



I'm trying to fill out a PDF form using the `AcroFields` class. I'm able to add text data perfectly, but I'm having issues adding images. How is this done?

Posted on StackOverflow on [Apr 17, 2013](#)<sup>775</sup> by [Anil M](#)<sup>776</sup>

The “official” way to do this, is to have a Button field as placeholder for the image, and to replace the “icon” of the button like this:

```
PushButtonField ad = form.getNewPushButtonFromField(imageFieldName);
ad.setLayout(PushbuttonField.LAYOUT_ICON_ONLY);
ad.setProportionalIcon(true);
ad.setImage(Image.getInstance("E:/signature/signature.png"));
form.replacePushButtonField("advertisement", ad.getField());
```

See [ReplaceIcon.java](#)<sup>777</sup> for the full code sample.

---

<sup>775</sup><http://stackoverflow.com/questions/16052047/add-image-to-acrofield-in-itext>

<sup>776</sup><http://stackoverflow.com/users/1716809/anil-m>

<sup>777</sup><http://itextpdf.com/examples/iiapdf?id=156>

## How to format a field as a percentage?



I use iText from C# code. I use AcroFields to populate a form with data, but I lose my formatting:

```
Stream os = new FileStream(PDFPath, FileMode.CreateNew);
PdfReader reader = new PdfReader(memIO);
PdfStamper stamper = new PdfStamper(reader, os, '9', true);
AcroFields fields = stamper.AcroFields;
fields.SetField("Pgo", 1.0 "Percentage");
```

What am I doing wrong?

Posted on StackOverflow on Dec 30, 2014<sup>778</sup> by Arne<sup>779</sup>

You say you are losing all your formatting. This leads to the assumption that your form contains JavaScript that formats fields when a human user enters a value.

When you fill out a form using iText, the JavaScript methods that are triggered when a form is filled out manually, aren't executed. Hence 1.0 will not be changed into 100%.

I assume that this is a typo:

```
fields.SetField("Pgo", 1.0 "Percentage");
```

This line doesn't even compile. Your IDE should show an error when typing this line. Even if I would add the missing comma between the second and third parameter (assuming that 1.0 and "Percentage" are two separate parameters), you'd get an error saying that there is no `setField()` method that accepts a String, a double and a String.

There are two `setField()` methods:

- `setField(String name, String value)`<sup>780</sup>: the first parameter is the key (e.g. "Pgo"), the second one is the value (e.g. the String value "1.0").
- `setField(String name, String value, String display)`<sup>781</sup>: the first parameter is the key (e.g. "Pgo"), the second one is the value (e.g. the String value "1.0"), and the third parameter is what should be displayed (e.g. "100%").

For instance, when you do this:

<sup>778</sup><http://stackoverflow.com/questions/27695621/itextpdf-acrofields-format-as-percentage>

<sup>779</sup><http://stackoverflow.com/users/2685789/arne>

<sup>780</sup>[http://api.itextpdf.com/itext/com/itextpdf/text/pdf/AcroFields.html#setField\(java.lang.String,%20java.lang.String\)](http://api.itextpdf.com/itext/com/itextpdf/text/pdf/AcroFields.html#setField(java.lang.String,%20java.lang.String))

<sup>781</sup>[http://api.itextpdf.com/itext/com/itextpdf/text/pdf/AcroFields.html#setField\(java.lang.String,%20java.lang.String,%20java.lang.String,%20java.lang.String\)](http://api.itextpdf.com/itext/com/itextpdf/text/pdf/AcroFields.html#setField(java.lang.String,%20java.lang.String,%20java.lang.String,%20java.lang.String))

```
fields.SetField("Pgo", "1.0", "100%");
```

iText will display 100%, but the value of the field will be 1.0.

If somebody clicks the field, 1.0 could appear, so that the end user can change the field, for instance into 0.5. It will depend on the presence of formatting JavaScript whether or not this manually entered 0.5 will be displayed as 50% or not.

## How to add a hidden text field?



I'd like to create a text field with iTextSharp that is not visible. Here's code I'm using to create a text field:

```
TextField field = new TextField(writer, new Rectangle(x, y - h, x + w, y), name);
field.BackgroundColor = new BaseColor(bgcolor[0], bgcolor[1], bgcolor[2]);
field.BorderColor = new BaseColor(
    bordercolor[0], bordercolor[1], bordercolor[2]);
field.BorderWidth = border;
field.BorderStyle = PdfBorderDictionary.STYLE_SOLID;
field.Text = text;
writer.AddAnnotation(field.GetTextField());
```

Posted on StackOverflow on May 21, 2013 <sup>782</sup> by mikessu<sup>783</sup>

In Java, the `TextField` class has a method named `setVisibility()`<sup>784</sup> inherited from its parent, the `BaseField` class. Possible values are:

- `BaseField.VISIBLE`,
- `BaseField.HIDDEN`,
- `BaseField.VISIBLE_BUT_DOES_NOT_PRINT`, and
- `BaseField.HIDDEN_BUT_PRINTABLE`.

As you're using iTextSharp, you should look for the `Visibility` property.

```
field.Visibility = BaseField.HIDDEN;
```

<sup>782</sup><http://stackoverflow.com/questions/16665562/how-do-i-make-text-field-to-be-hidden>

<sup>783</sup><http://stackoverflow.com/users/2138736/mikessu>

<sup>784</sup><http://api.itextpdf.com/itext/com/itextpdf/text/pdf/BaseField.html#setVisibility%28int%29>

## How to send a file to the server through a PDF?



I'm dynamically creating some PDF files through my application, and I need to use several components (TextField, CheckBox, RadioButtons etc), and then submit the values to server. One of the requirements says that the user needs to be able to select and send files along with the other values. I did not find a specific component to this, and so I'm asking for some help with this situation.

Is there a way to create some kind of Input File, File Chooser, etc, and attach it on the generated PDF file? And then send this selected file to server?

Posted on StackOverflow on Jul 18, 2014<sup>785</sup> by user2230377<sup>786</sup>

I've prepared an example that explains how to create such a field when creating a document from scratch: [FileSelectionExample<sup>787</sup>](#)

A file selection field is created just like any other text field, except that you have to set a flag using the `setOptions()` method. If you want a file chooser to appear, you also have to add a JavaScript action:

```
TextField file = new TextField(
    writer, new Rectangle(36, 788, 559, 806), "myfile");
file.setOptions(TextField.FILE_SELECTION);
PdfFormField upload = file.getTextField();
upload.setAdditionalActions(PdfName.U, PdfAction.javaScript(
    "this.getField('myfile').browseForFileToSubmit();", writer));
writer.addAnnotation(upload);
```

In the full example, I also added a second field and after selecting the file with the `browseForFileToSubmit()` method, I set the focus to that other field. I'm doing this because the file name only becomes visible in the file selection field after that field loses focus.

## How to add a text field to an existing pdf template



I want create a PDF template from another template. The resulting PDF should still be a template that I can fill out with data. I tried using `PdfStamper` but the resulting PDF is not template.

Posted on StackOverflow on Sep 11, 2013<sup>788</sup> by cgnbkm<sup>789</sup>

<sup>785</sup><http://stackoverflow.com/questions/24830060/send-file-to-server-through-itext-pdf>

<sup>786</sup><http://stackoverflow.com/users/2230377/user2230377>

<sup>787</sup><http://itextpdf.com/sandbox/acroforms/FileSelectionExample>

<sup>788</sup><http://stackoverflow.com/questions/18738582/how-can-i-add-textfield-to-the-existing-pdf-template>

<sup>789</sup><http://stackoverflow.com/users/2711046/cgnbkm>

Let's distinguish two situations, depending on the nature of your PDF template:

#### You are talking about an XFA template:

In this case, the PDF is merely a container for an XML stream that defines your form. The only way to change it, is by editing the XML. This is best done *manually* using Adobe LiveCycle Designer, but if you really want to do it *programmatically*, you can extract the XML from the PDF using iText, manipulate the XML using any type of XML editing software, and finally put back the XML into the PDF using iText. The programmatical solution is very difficult as it requires you to be familiar with the XFA syntax and the specs for XFA consist of several hundreds of pages.

#### You are talking about an AcroForm template

In this case, the root dictionary has an /AcroForm dictionary of which one of the entries is a /Fields array that isn't empty. You can create a PdfReader instance for this template and pass the reader object to PdfStamper. You then create the extra fields you need (text fields, button fields,...) and add them to the stamper using the addAnnotation() method.

If this doesn't answer your question, please clarify, as it's generally not accepted to limit your question to saying "*I tried using PdfStamper but the resulting PDF is not template.*", you should at least show what you've tried, otherwise you risk that your question will be closed.

## How can I add a new AcroForm field to a PDF?



I have used iText to fill data into existing AcroForm fields in a PDF.

I am now looking for a solution to add new AcroForm fields to a PDF. Is this possible with iText? If so, how can I do this?

Posted on StackOverflow on Nov 30, 2014 <sup>790</sup> by Peter<sup>791</sup>

This is documented in the official documentation, more specifically in the [SubmitForm<sup>792</sup>](#) example. Additionally, I've written a simple example called [AddField<sup>793</sup>](#). It adds a button field at a specific position defined by new Rectangle(36, 700, 72, 730).

<sup>790</sup><http://stackoverflow.com/questions/27206327/add-new-acroform-field-to-a-pdf>

<sup>791</sup><http://stackoverflow.com/users/230996/peter>

<sup>792</sup><http://itextpdf.com/examples/ii.php?id=168>

<sup>793</sup><http://itextpdf.com/sandbox/acroforms/AddField>

```

public void manipulatePdf(String src, String dest)
    throws DocumentException, IOException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    PushbuttonField button = new PushbuttonField(
        stamper.getWriter(), new Rectangle(36, 700, 72, 730), "post");
    button.setText("POST");
    button.setBackgroundColor(new GrayColor(0.7f));
    button.setVisibility(PushbuttonField.VISIBLE_BUT_DOES_NOT_PRINT);
    PdfFormField submit = button.getField();
    submit.setAction(PdfAction.createSubmitForm(
        "http://itextpdf.com:8180/book/request", null,
        PdfAction.SUBMIT_HTML_FORMAT | PdfAction.SUBMIT_COORDINATES));
    stamper.addAnnotation(submit, 1);
    stamper.close();
}

```

As you can see, you need to create a `PdfFormField` object (using helper classes such as `PushbuttonField`, `TextField`,...) and then use `PdfStamper`'s `addAnnotation()` method to add the field to a specific page.

## How to underline a portion of text in a text field?



I have an application that uses iText to fill PDF form fields. One of these fields has some text with tags. For example:

`<U>This text</U>` should be underlined.

I need the text enclosed in

Posted on StackOverflow on Feb 18, 2015 <sup>794</sup> by Galma88<sup>795</sup>

Ordinary text fields do not support rich text. If you want the fields to remain interactive, you will need RichText fields. These are fields that are flagged in a way that they accept an RV value.

If it is OK for you to flatten the form (i.e. remove all interactivity), please take a look at the `FillWithUnderline`<sup>796</sup> example:

---

<sup>794</sup><http://stackoverflow.com/questions/28579382/underline-portion-of-text-using-itextsharp>

<sup>795</sup><http://stackoverflow.com/users/3667920/galma88>

<sup>796</sup><http://itextpdf.com/sandbox/acroforms/FillWithUnderline>

```

public void manipulatePdf(String src, String dest)
    throws DocumentException, IOException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.setFormFlattening(true);
    AcroFields form = stamper.getAcroFields();
    FieldPosition pos = form.getFieldPositions("Name").get(0);
    ColumnText ct = new ColumnText(stamper.getOverContent(pos.page));
    ct.setSimpleColumn(pos.position);
    ElementList elements = XMLWorkerHelper.parseToElementList(
        "<div>Bruno <u>Lowagie</u></div>", null);
    for (Element element : elements) {
        ct.addElement(element);
    }
    ct.go();
    stamper.close();
}

```

In this example, we don't fill out the field, but we get the fields position (a page number and a rectangle). We then use `ColumnText` to add content at this position. As we are inputting HTML, we use XML Worker to parse the HTML into iText objects that we can add to the `ColumnText` object.

## Why are the `AcroFields` in my document empty?



I have a PDF form with filled out fields. I can change the values and save them. But when I try to read the `AcroFields`, they are empty.

```

var reader = new PdfReader((DataContext as PDFContext).Datei);
AcroFields form = reader.AcroFields;
txt.Text = GetFormFieldNamesWithValues(reader);

private static string GetFormFieldNamesWithValues(PdfReader pdfReader) {
    return string.Join("\r\n", pdfReader.AcroFields.Fields
        .Select(x => x.Key + "=" +
        pdfReader.AcroFields.GetField(x.Key)).ToArray());
}

```

How can I read the fields?

Posted on StackOverflow on Apr 7, 2014<sup>797</sup> by Gregor Glinka<sup>798</sup>

---

<sup>797</sup><http://stackoverflow.com/questions/22909979/itextsharp-acrofields-are-empty>

<sup>798</sup><http://stackoverflow.com/users/3506318/gregor-glinka>

Clearly your PDF is broken. The fields are defined as *widget annotations* on the page level, but they aren't referenced in the `/AcroForm` fields set on the *document root level*.

You can fix your PDF using this code sample:

```
PdfReader reader = new PdfReader(src);
PdfDictionary root = reader.getCatalog();
PdfDictionary form = root.getAsDict(PdfName.ACROFORM);
PdfArray fields = form.getAsArray(PdfName.FIELDS);
PdfDictionary page;
PdfArray annots;
for (int i = 1; i <= reader.getNumberOfPages(); i++) {
    page = reader.getPageN(i);
    annots = page.getAsArray(PdfName.ANNOTS);
    for (int j = 0; j < annots.size(); j++) {
        fields.add(annots.getAsIndirectObject(j));
    }
}
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
stamper.close();
reader.close();
```

Or if you need this in C#:

```
PdfReader reader = new PdfReader(src);
PdfDictionary root = reader.Catalog;
PdfDictionary form = root.GetAsDict(PdfName.ACROFORM);
PdfArray fields = form.GetAsArray(PdfName.FIELDS);
PdfDictionary page;
PdfArray annots;
for (int i = 1; i <= reader.NumberOfPages; i++) {
    page = reader.GetPageN(i);
    annots = page.GetAsArray(PdfName.ANNOTS);
    for (int j = 0; j < annots.Size; j++) {
        fields.Add(annots.GetAsIndirectObject(j));
    }
}
PdfStamper stamper = new PdfStamper(reader, new FileStream(dest, FileMode.Create));
stamper.Close();
reader.Close();
```

You should inform the creators of the tool that was used to produce the form that their PDFs aren't compliant with the PDF reference.

## How to create a pushbuttonfield with double byte character text?



I need a pushbuttonfield with double byte characters (DBCS) as the text. Currently if I set the button text as any double byte text, the text won't show up.

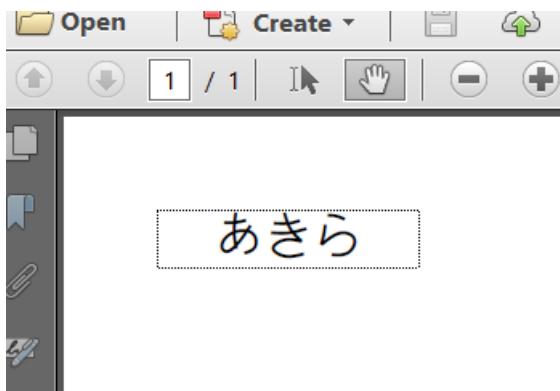
This works:

```
PushbuttonField myButton = new PushbuttonField(writer, rect, "JapaneseButton");
myButton.setText("japanese text here");
```

But if I replace "japanese text here" with real Japanese text, the text doesn't show.

Posted on StackOverflow on Mar 12, 2015<sup>799</sup> by Jason<sup>800</sup>

Please take a look at the [CreateJapaneseButton<sup>801</sup>](#) example. It creates a PDF with a button that looks like this:



Button with Japanese text

Your code snippet is very short, but you probably have all the code you need, except the line that sets the font:

<sup>799</sup> <http://stackoverflow.com/questions/29016333/pushbuttonfield-with-double-byte-character>

<sup>800</sup> <http://stackoverflow.com/users/4420935/jason>

<sup>801</sup> <http://itextpdf.com/sandbox/acroforms/CreateJapaneseButton>

```
public static final String JAPANESE = "\u3042\u304d\u3089";
public static final String FONT = "resources/fonts/FreeSans.ttf";

public void createPdf(String dest) throws IOException, DocumentException {
    Document document = new Document();
    PdfWriter writer =
        PdfWriter.getInstance(document, new FileOutputStream(dest));
    document.open();
    PushbuttonField button = new PushbuttonField(
        writer, new Rectangle(36, 780, 144, 806), "japanese");
    BaseFont bf = BaseFont.createFont(
        FONT, BaseFont.IDENTITY_H, BaseFont.EMBEDDED);
    button.setFont(bf);
    button.setText(JAPANESE);
    writer.addAnnotation(button.getField());
    document.close();
}
```

If you don't set the font, the default font is used. The default font is the Standard Type 1 font Helvetica. This font doesn't know how to write Japanese. I used the font freesans.ttf. If you don't have that font, you can try arialuni.ttf or any other font that supports Japanese.

## How to define the data encoding when submitting a PDF form using AcroForm technology?



When I create a PDF form (for instance using Acrobat) that contains text fields in AcroForm format (PDF dictionaries, no XFA), and I submit the data to a server, how can I specify/retrieve the encoding that will be used?

For instance. When I submit the Chinese glyphs '测试' (test), I receive the following headers and content on the server-side:

```
accept: application/x-ms-application, image/jpeg, application/xaml+xml, image/gi\
f, image/pjpeg, application/x-ms-xbap, application/vnd.ms-excel, application/vnd\
.ms-powerpoint, application/msword, /*\
content-type: application/x-www-form-urlencoded
content-length: 23
acrobat-version: 10.1.4
user-agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64; Trident/4.\.
0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Cent\
er PC 6.0; MDDC; .NET4.0C; AskTbCLA/5.15.1.22229)
accept-encoding: gzip, deflate
connection: Keep-Alive
Song=%b2%e2%ca%d4&Test=
```

There's no reference to an encoding, except x-www-form-urlencoded. The two glyphs are represented as four bytes: B2 E2 CA D4. After some investigation, I know that B2E2 is the GBK value for the first glyph, and CAD4 the GBK value for the second glyph, but I can't derive this from the request header.

Where can I find information about the data encoding used when submitting PDF? Is it always GBK? Can I change the data encoding by setting a specific key in a dictionary in the PDF? For instance: can I make sure the PDF always sends Unicode characters instead of GBK?

Note that I've already experimented by changing the default font (and encoding) of the text field. I've also searched ISO-32000-1 for encodings in fields, but all I found was a way to define non-Latin characters for check boxes, and some info about the encoding of an FDF file. None of which answered my questions.

Posted on StackOverflow on Sep 26, 2012 <sup>802</sup> by Bruno Lowagie<sup>803</sup>

I didn't find anything in ISO-32000-1, but studying the Acrobat JavaScript reference, I found the cCharset parameter that is available for the submitForm() method. That parameter defines:

<sup>802</sup><http://stackoverflow.com/questions/12604171/data-encoding-when-submitting-a-pdf-form-using-acroform-technology>

<sup>803</sup><http://stackoverflow.com/users/1622493/bruno-lowagie>

The encoding for the values submitted. String values are utf-8, utf-16, Shift-JIS, BigFive, GBK, and UHC. If not passed, the current Acrobat behavior applies. For XML-based formats, utf-8 is used. For other formats, Acrobat tries to find the best host encoding for the values being submitted. XFDF submission ignores this value and always uses utf-8.

In other words: in my case GBK was used because it fits best to submit Chinese characters. However, one could force UTF-8 by using the `submitForm()` JavaScript method using the appropriate value.

Based on this question, I have asked the ISO committee to fix this problem in ISO-32000-2. As a result, an extra possible entry was added to the table entitled *Additional entries specific to a submit-form action* in section 12.7.6.2:

**CharSet** string: (Optional; inheritable) Possible values include: *utf-8, utf-16, Shift-JIS, BigFive, GBK, or UHC*.

Starting with PDF 2.0, this problem will no longer exist.

# Actions and annotations

All things interactive are discussed here. Except for forms, we've already covered these.

## How to create hierarchical bookmarks?



I have data in `ArrayList` like below :

```
ArrayList<DTONodeDetail> tree = new ArrayList<DTONodeDetail>();
tree.add(getDTO(1, "Root", 0));
tree.add(getDTO(239, "Node-1", 1));
tree.add(getDTO(242, "Node-2", 239));
tree.add(getDTO(243, "Node-3", 239));
tree.add(getDTO(244, "Node-4", 242));
tree.add(getDTO(245, "Node-5", 243));
```

I am able to display above data in `tree` structure like below :

```
Root
-----Node-1
-----Node-2
-----Node-4
-----Node-3
-----Node-5
```

How can I create hierarchical bookmark (as shown above) in PDF using iText?

Posted on StackOverflow on Feb 26, 2015 <sup>804</sup> by Butani Vijay<sup>805</sup>

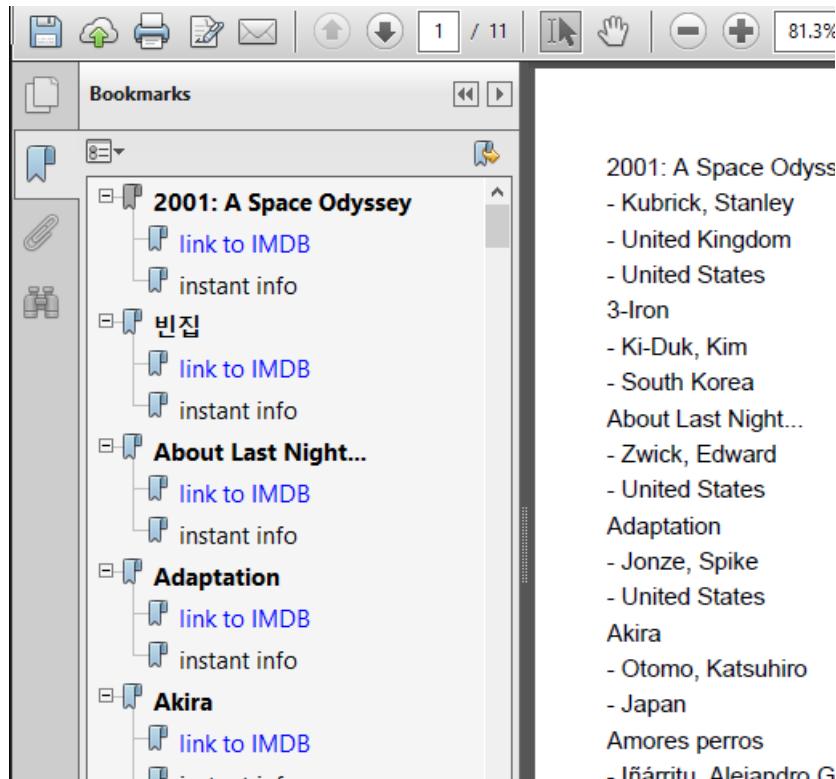
In PDF terminology, bookmarks are referred to as *outlines*. Please take a look at the [CreateOutline<sup>806</sup>](#) example from my book to find out how to create an *outline tree* as shown in this PDF: [outline\\_tree.pdf<sup>807</sup>](#)

<sup>804</sup><http://stackoverflow.com/questions/28734490/hierarchical-bookmark-generation-in-pdf-using-itext-library>

<sup>805</sup><http://stackoverflow.com/users/2178702/butani-vijay>

<sup>806</sup><http://itextpdf.com/examples/iiapdf?id=139>

<sup>807</sup>[http://examples.itextpdf.com/results/part2/chapter07/outline\\_tree.pdf](http://examples.itextpdf.com/results/part2/chapter07/outline_tree.pdf)



Outline tree shown in the bookmarks panel

We start with the root of the tree:

```
PdfOutline root = writer.getRootOutline();
```

Then we add a branch:

```
PdfOutline movieBookmark = new PdfOutline(root,
    new PdfDestination(
        PdfDestination.FITH, writer.getVerticalPosition(true)),
    title, true);
```

To this branch, we add a leaf:

```
PdfOutline link = new PdfOutline(movieBookmark,
    new PdfAction(String.format(RESOURCE, movie.getImdb())),
    "link to IMDB");
```

And so on.

The key is to use `PdfOutline`<sup>808</sup> and to pass the parent outline as a parameter when constructing a child outline.



Can I do this in an existing pdf? I mean without creating new PDF, I want to add bookmarks to an existing pdf.

There's also an example called `BookmarkedTimeTable`<sup>809</sup> where we create the outline tree in a completely different way:

```
ArrayList<HashMap<String, Object>> outlines =
    new ArrayList<HashMap<String, Object>>();
HashMap<String, Object> map = new HashMap<String, Object>();
outlines.add(map);
```

In this case, `map` is the root object to which we can add branches and leaves. To create a hierarchy, you just need to add kids like this:

First level:

```
HashMap<String, Object> calendar = new HashMap<String, Object>();
calendar.put("Title", "Calendar");
```

Second level:

```
HashMap<String, Object> day = new HashMap<String, Object>();
day.put("Title", "Monday");
ArrayList<HashMap<String, Object>> days =
    new ArrayList<HashMap<String, Object>>();
days.add(day);
calendar.put("Kids", days);
```

Once we're finished, we add the outline tree to the `PdfStamper` like this:

```
stamper.setOutlines(outlines);
```

Note that `PdfStamper` is the class we need when we manipulate an existing PDF (as opposed to `PdfWriter` which is to be used when we create a PDF from scratch).



I am unable to add third level using this. I mean suppose in your example I want to add child to the date, and have a branch like Calendar -> 2014-03-02 -> Monday

Third level:

---

<sup>808</sup><http://api.itextpdf.com/itext/com/itextpdf/text/pdf/PdfOutline.html>

<sup>809</sup><http://itextpdf.com/examples/ii.php?id=140>

```
HashMap<String, Object> hour = new HashMap<String, Object>();
hour.put("Title", "10 AM");
ArrayList<HashMap<String, Object>> hours = new ArrayList<HashMap<String, Object>>();
hours.add(hour);
day.put("Kids", hours);
```

And so on...

## How can I add titles of chapters in ColumnText?



I need to make a PDF like this:

зубная паста Silca/Сника	
ан Silca Нетанс White 100ml/Denc/12	36.00 12230
ан Silca Arctic White в гутт 100ml/Denc/12	70.01 12264
ан Silca Best Care-Aloe в гутт 100ml/Denc/12	55.57 12300
ан Silca Citrus Fresh+Whitening в гутт 100ml/Denc/12	70.01 12322
ан Silca Complete Generative в гутт 100ml/Denc/12	72.66 12231
ан Silca Family в гутт 100ml/Denc/12	80.71 12233
ан Silca Herbal Complete в гутт 100ml/Denc/12	72.65 12238
ан Silca Herbal White в гутт 100ml/Denc/12	70.01 12230
ан Silca Naturale Efficiale в гутт 100ml/Denc/12	72.65 12299
ан Silca MED Продекс уход Временный энз., 100ml/Denc/12	100.00 13416
ан Silca MED Продекс уход Тотал, 100ml/Denc/12	100.00 13417
ан Silca Мультиактив в фит 100ml/Denc/12	78.57 13486
ан Silca Сенситивесид туба 130ml/Denc/24	28.81 80025
ан Silca Сенситивесид туба 130ml/Denc/24	28.81 80024
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80023
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80022
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80021
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80020
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80019
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80018
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80017
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80016
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80015
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80014
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80013
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80012
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80011
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80010
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80009
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80008
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80007
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80006
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80005
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80004
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80003
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80002
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80001
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 80000
ан Silca Сенситивесид туба 6 л-р дент. флюид ПРОМО/Denc/24	41.90 7_PRL
ан Silca Brilliant NEW/Denc/12	68.56 172
ан Silca Comfort/Denc/12	88.46 208
ан Silca Fresh/Denc/12	50.00 95
ан Silca Soft Мягкий жесткий/Denc/12	30.15 501WS
ан Silca Z3 Макси/Denc/12	87.13 174PL
ан Silca dent Hard жесткий/Denc/12	30.16 603
ан Silca dent Medium средний жесткий/Denc/12	30.16 622
ан Silca dent Дорожная на флямбере/Denc/12	28.19 604
ан Silca med MultiPaste/Denc/12	95.71 181
ан Silca med Travel-Compact/Denc/12	95.71 190
ан Silca med X-Актив/Denc/12	95.71 182
ан Silca Классический/Denc/12	60.00 203Н PL
ан Silca Средний жесткий/Denc/12	60.00 203М PL
ан Silca Классический гладкий/Denc/12	67.13 207 PL
ан Silca X-гладкий/Denc/12	95.71 173 PL
ан Silca Puff Baby 1-6 лет/Denc/12	65.75 164
ан Silca Puff Junior 6-12 лет/Denc/12	73.57 160
ан Silca Puff Kids 3-6 лет/Denc/12	73.57 161
ан Silca Dent soft мягкая 2-7 лет/Denc/12	27.50 605
ан Silca Краска для зубов/Denc/12	69.25 169
<b>Суперпасты WhiteGlo/LongaVita/Splat</b>	
ан White Glo Отбеливание для детей возрастом 10-14 лет/гелевая/жидкость	121.40 66007
ан White Glo Отбеливание для взрослых 15+ лет/гелевая/жидкость	123.75 74811
ан White Glo Отбеливание для детей возрастом 10-14 лет/гелевая/жидкость	121.40 66007
ан Longa Vita зефир/конфеты Magic Light/жевательная	206.66 94929
ан White Glo оранжевый дуэтный напиток/жидкость	80.37 86624
ан Longa Vita забавные зверяки мятные 3-лист/пакетик/4441	130.31 65898F-B6D
<b>Жидкое</b>	
Капли для глаз Ежевика/жидкость	328.98 81137163/12084
Капли для глаз Банан/жидкость	104.93 81237150
Капли для глаз-2 в 1 яблоко/жидкость	281.91 81120444
Банка для сыворотки Машин-2 яблоко/жидкость	195.25 81176003
Банка для сыворотки Машин-2 яблоко/жидкость	195.25 81175026
Банка для сыворотки Машин-2 яблоко/жидкость	227.52 81427057
Продукт для увлажнения кожи 200ml/Gel	227.52 81464971
Гель для рук ProSkin Protective Skin душевые косметики 200ml/Gel	227.52 81416327/81451
Гель для рук ProSkin Protective Skin душевые косметики 200ml/Gel	227.52 81454955/84855
Гель для рук ProSkin Protective Skin душевые косметики 200ml/Gel	227.52 84857294
Гель для рук ProSkin Protective Skin душевые косметики 200ml/Gel	193.27 81364197/84834
Гель для рук Sensitive Moist Увлажняющий гель 200ml/Gel	193.27 80214416
Гель для рук Sensitive Moist Увлажняющий гель 75ml/Gel	115.80 80213161
Гель для рук Protection Защита с мандариновым маслом 200ml/Gel	193.27 81370797
Гель для рук SERIES Extra Comfort крем-гель 200ml/Gel	193.27 84857351/84834
Гель для рук SERIES Moisturizing увлажняющий 200ml/Gel	193.27 81407255
Гель для рук SERIES Sensitive душевые косметики 200ml+банный	251.00 81306531
Гель для рук SERIES Sensitive душевые косметики 200ml/Gel	193.27 84857355
Гель для сан.дев. Avoada Testor лосьон 200ml/Gel	197.42 84854954
Гель для сан.дев. душевые душистые косметики 200ml/Gel	197.42 81464962
Гель для рук Gillette Mild Sensitive душевые косметики 100ml/Gel	193.27 81386777/84834
Косметика Gillette Fusion 3 шампунь 200ml/Gel	446.01 81260909/81304
Косметика Gillette Fusion 4 шампунь 200ml/Gel	609.92 81362393/81372

### Example

And then add TOC to this document. I create the document using `ColumnText` and `PdfPTable` objects, but I can't add a `Chapter` to create a TOC. I can only add a `Chapter` to the document body, but in that case, the title of the `Chapter` isn't shown in the `ColumnText`.

Posted on StackOverflow on Nov 23, 2012 <sup>810</sup> by Skindeep2366<sup>811</sup>

<sup>810</sup><http://stackoverflow.com/questions/13523099/separating-redundant-code-from-pdf-generator-function>

<sup>811</sup><http://stackoverflow.com/users/969487/skindeep2366>

Your question isn't clear in the sense that you don't tell us if you want a TOC like this:

The screenshot shows a software application window with a toolbar at the top. On the left, there is a 'Bookmarks' panel containing a tree view of outlines. The root node is 'Numbers from 1 to 100', which has several sub-nodes such as 'Numbers from 1 to 10', 'Numbers from 11 to 20', etc. To the right of the bookmarks panel are five tables, each labeled with a range of numbers and containing the word 'Test' in every cell.

Numbers from 1 to 10	
1	Test
2	Test
3	Test
4	Test
5	Test
6	Test
7	Test
8	Test
9	Test
10	Test

Numbers from 11 to 20	
11	Test
12	Test
13	Test
14	Test
15	Test
16	Test
17	Test
18	Test
19	Test
20	Test

Numbers from 21 to 30	
21	Test
22	Test

Numbers from 41 to 50	
42	Test
43	Test
44	Test
45	Test
46	Test
47	Test
48	Test
49	Test
50	Test

Numbers from 51 to 60	
51	Test
52	Test
53	Test
54	Test
55	Test
56	Test
57	Test
58	Test
59	Test
60	Test

Numbers from 61 to 70	
61	Test
62	Test
63	Test
64	Test

Numbers from 81 to 90	
85	Test
86	Test
87	Test
88	Test
89	Test
90	Test

Numbers from 91 to 100	
91	Test
92	Test
93	Test
94	Test
95	Test
96	Test
97	Test
98	Test
99	Test
100	Test

Numbers from 101 to 110	
101	Test
102	Test
103	Test
104	Test
105	Test
106	Test
107	Test

**Outline tree**

If this is the case, you are using the wrong terminology, as what you see in the Bookmarks panel can be referred to as Outlines or bookmarks.

If you say you want a TOC, you want something like this:

The screenshot shows a software application window with a toolbar at the top. On the left, there is a 'Table of Contents' panel containing a list of sections. The list includes: 'Numbers from 1 to 10', 'Numbers from 11 to 20', 'Numbers from 21 to 30', 'Numbers from 31 to 40', 'Numbers from 41 to 50', 'Numbers from 51 to 60', 'Numbers from 61 to 70', 'Numbers from 71 to 80', 'Numbers from 81 to 90', 'Numbers from 91 to 100', 'Numbers from 101 to 110', 'Numbers from 111 to 120', 'Numbers from 121 to 130', 'Numbers from 131 to 140', 'Numbers from 141 to 150', 'Numbers from 151 to 160', and 'Numbers from 161 to 170'. To the right of the table of contents is a large, mostly empty document area.

**Table of Contents**

I mention both, because you talk about the Chapter (a class you should no longer use) and that class creates bookmarks/outlines, not a TOC.

I have created a PDF file that has both, bookmarks and a TOC: [columns\\_with\\_toc.pdf<sup>812</sup>](#). Please take a look at the [CreateTOCinColumn<sup>813</sup>](#) example to find out how it's done.

Just like you, I create a `ColumnText` object with titles and tables:

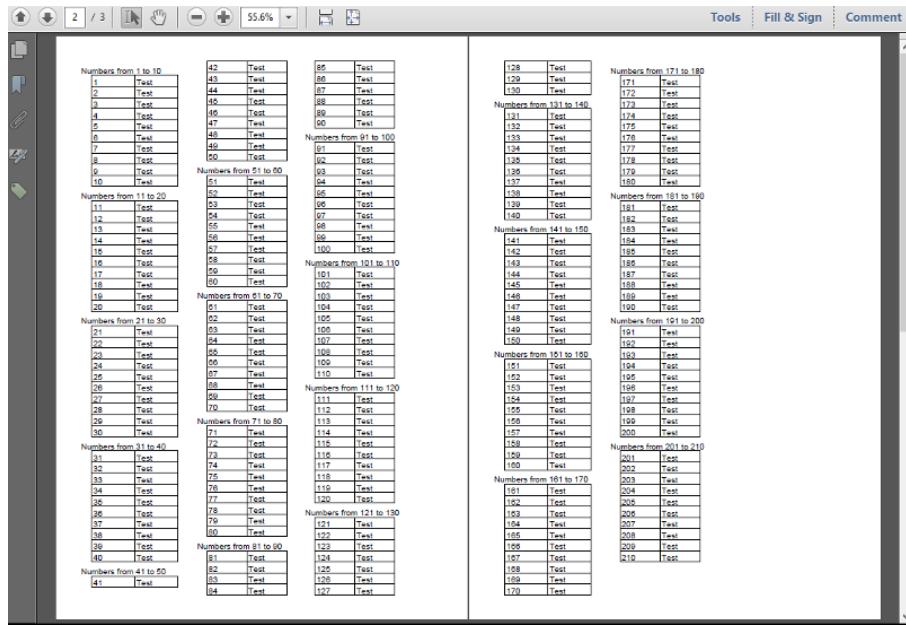
```
ColumnText ct = new ColumnText(writer.getDirectContent());
int start;
int end;
for (int i = 0; i <= 20; ) {
    start = (i * 10) + 1;
    i++;
    end = i * 10;
    String title = String.format("Numbers from %s to %s", start, end);
    Chunk c = new Chunk(title);
    c.setGenericTag(title);
    ct.addElement(c);
    ct.addElement(createTable(start, end));
}
int column = 0;
do {
    if (column == 3) {
        document.newPage();
        column = 0;
    }
    ct.setSimpleColumn(COLUMNS[column++]);
} while (ColumnText.hasMoreText(ct.go()));
```

The result looks like this:

---

<sup>812</sup>[http://itextpdf.com/sites/default/files/columns\\_with\\_toc.pdf](http://itextpdf.com/sites/default/files/columns_with_toc.pdf)

<sup>813</sup><http://itextpdf.com/sandbox/events/CreateTOCinColumn>



### Columns with titles and tables

In spite of the rules for posting a question on StackOverflow, you didn't post a code sample, but there is at least one difference between your code and mine:

```
c.setGenericTag(title);
```

In this line, we declare a generic tag. This tag is used by the TOCEntry class that looks like this:

```
public class TOCCreation extends PdfPageEventHelper {

    protected PdfOutline root;
    protected List<TOCEntry> toc = new ArrayList<TOCEntry>();

    public TOCCreation() {
    }

    public void setRoot(PdfOutline root) {
        this.root = root;
    }

    public List<TOCEntry> getToc() {
        return toc;
    }

    @Override
```

```

public void onGenericTag(
    PdfWriter writer, Document document, Rectangle rect, String text) {
    PdfDestination dest = new PdfDestination(
        PdfDestination.XYZ, rect.getLeft(), rect.getTop(), 0);
    new PdfOutline(root, dest, text);
    TOCEntry entry = new TOCEntry();
    entry.action = PdfAction.gotoLocalPage(
        writer.getPageNumber(), dest, writer);
    entry.title = text;
    toc.add(entry);
}
}

```

As you can see, we create a PdfDestination based on the position of the title:

```

PdfDestination dest =
    new PdfDestination(PdfDestination.XYZ, rect.getLeft(), rect.getTop(), 0);

```

If you want bookmarks, you can create a PdfOutline like this:

```

new PdfOutline(root, dest, text);

```

If you want a TOC, you can store a String and a PdfAction in a List:

```

TOCEntry entry = new TOCEntry();
entry.action = PdfAction.gotoLocalPage(writer.getPageNumber(), dest, writer);
entry.title = text;
toc.add(entry);

```

Now that we understand the TOCCreation class, we take a look at how to use it:

```

PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream(dest));
TOCCreation event = new TOCCreation();
writer.setPageEvent(event);
document.open();
event.setRoot(writer.getRootOutline())

```

We create an event object, pass it to the writer and after we've opened the document, we pass the root of the outline tree to the event. The bookmarks will be created automatically, the TOC won't. If you want to add the TOC, you need something like this:

```

document.newPage();
for (TOCEntry entry : event.getToc()) {
    Chunk c = new Chunk(entry.title);
    c.setAction(entry.action);
    document.add(new Paragraph(c));
}

```

You now have a list of titles which you can click to jump to the corresponding table.

## How to create a link to a specific page number?



I know how to target any text of any PDF page using code:

```

Anchor click = new Anchor("Click to go to Target");
click.Reference = "#target";
Paragraph p1 = new Paragraph();
p1.Add(click);
doc.Add(p1);
Anchor target = new Anchor("Target");
target.Name = "target";
doc.Add(target);

```

My question is how to target a page based on its number. For example if targeted page number is 6, clicking on the Anchor text should take to 6th page.

Posted on StackOverflow on Feb 20, 2014<sup>814</sup> by Yogesh<sup>815</sup>

Instead of an Anchor, you need a Chunk. To this Chunk you need to add a PdfAction. The action needs to be a `gotoLocalPage()`<sup>816</sup> action.

For instance:

```

Chunk chunk = New Chunk("Go to page 5");
PdfAction action = PdfAction.GotoLocalPage(5, New PdfDestination(0), writer);
chunk.SetAction(action);

```

---

<sup>814</sup><http://stackoverflow.com/questions/21907184/itextsharp-how-to-target-pdf-page-number>

<sup>815</sup><http://stackoverflow.com/users/532384/yogesh>

<sup>816</sup><http://api.itextpdf.com/itext/com/itextpdf/text/pdf/PdfAction.html#gotoLocalPage%28int,%20com.itextpdf.text.pdf.PdfDestination,%20com.itextpdf.text.pdf.PdfWriter%29>

## How to insert a “linked rectangle” with iText?



I want to insert a hyperlink into an existing PDF at a position I know in advance: I already have the coordinates of a rectangle on a given page. I want to link this rectangle to another page of the same PDF (which I also know in advance). How do I achieve this?

Posted on StackOverflow on Nov 7, 2013 <sup>817</sup> by Hans Stricker<sup>818</sup>

Please take a look at the [AddLinkAnnotation](#)<sup>819</sup> example.

As you (should) already know (but you didn't show what you've already tried, which is kind of mandatory on StackOverflow), you can use PdfStamper to manipulate an existing PDF. Adding a rectangular link on one page to another page, is as simple as adding a *link annotation* to that page:

```
PdfReader reader = new PdfReader(src);
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
Rectangle linkLocation = new Rectangle(523, 770, 559, 806);
PdfDestination destination = new PdfDestination(PdfDestination.FIT);
PdfAnnotation link = PdfAnnotation.createLink(stamper.getWriter(),
    linkLocation, PdfAnnotation.HIGHLIGHT_INVERT,
    3, destination);
link.setBorder(new PdfBorderArray(0, 0, 0));
stamper.addAnnotation(link, 1);
stamper.close();
```

The link object is created using:

- the writer instance tied to the stamper,
- the rectangle (the position you say you know in advance),
- a highlighting option (pick one: HIGHLIGHT\_NONE, HIGHLIGHT\_INVERT, HIGHLIGHT\_OUTLINE, HIGHLIGHT\_PUSH, HIGHLIGHT\_TOGGLE),
- the page you want to link to,
- a destination.

Once you have an instance of PdfAnnotation, you can add it to a specific page using the addAnnotation() method.

---

<sup>817</sup><http://stackoverflow.com/questions/22194844/inserting-a-linked-rectangle-with-itext>

<sup>818</sup><http://stackoverflow.com/users/363429/hans-stricker>

<sup>819</sup><http://itextpdf.com/sandbox/annotations/AddLinkAnnotation>

## How to add a maps with a pointer to a PDF?



I am using java and iText to create a pdf. Is it possible to add a map with a pointer on it so the user will know where the starting point is?

Posted on StackOverflow on Nov 6, 2014 <sup>820</sup> by user2487493<sup>821</sup>

What do you mean by “a map with a pointer so the user knows where the starting point is”? If you have a map in your PDF, you could add an annotation that looks like an arrow. Is that what you’re looking for?

Since you didn’t answer my counter-question added in comment, I’m providing two examples. If these are not what you’re looking for, you really should clarify your question.

### Example 1: add a custom shape as extra content on top of a map

This is demonstrated in the [AddPointer<sup>822</sup>](#) example:

```
PdfContentByte canvas = writer.getDirectContent();
canvas.setColorStroke(BaseColor.RED);
canvas.setLineWidth(3);
canvas.moveTo(220, 330);
canvas.lineTo(240, 370);
canvas.arc(200, 350, 240, 390, 0, (float) 180);
canvas.lineTo(220, 330);
canvas.closePathStroke();
canvas.setColorFill(BaseColor.RED);
canvas.circle(220, 370, 10);
canvas.fill();
```

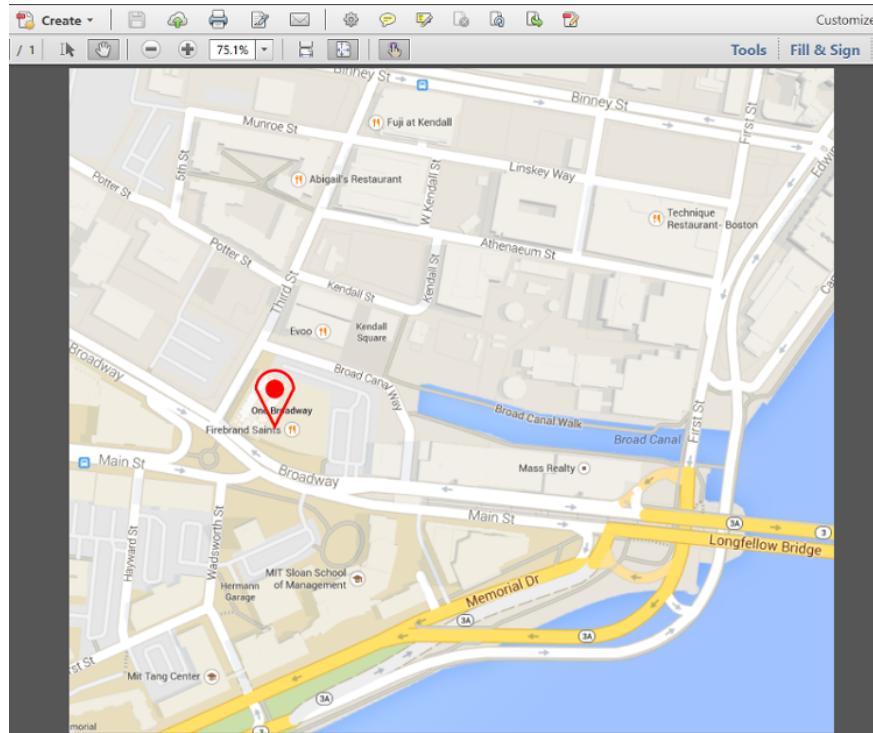
If we know the coordinates of the pointer, we can draw lines and curves that result in a the red pointer shown here (see the red pin near the Cambridge Innovation Center):

---

<sup>820</sup><http://stackoverflow.com/questions/26752663/adding-maps-at-itext-java>

<sup>821</sup><http://stackoverflow.com/users/2487493/user2487493>

<sup>822</sup><http://itextpdf.com/sandbox/objects/AddPointer>



Map with a pin

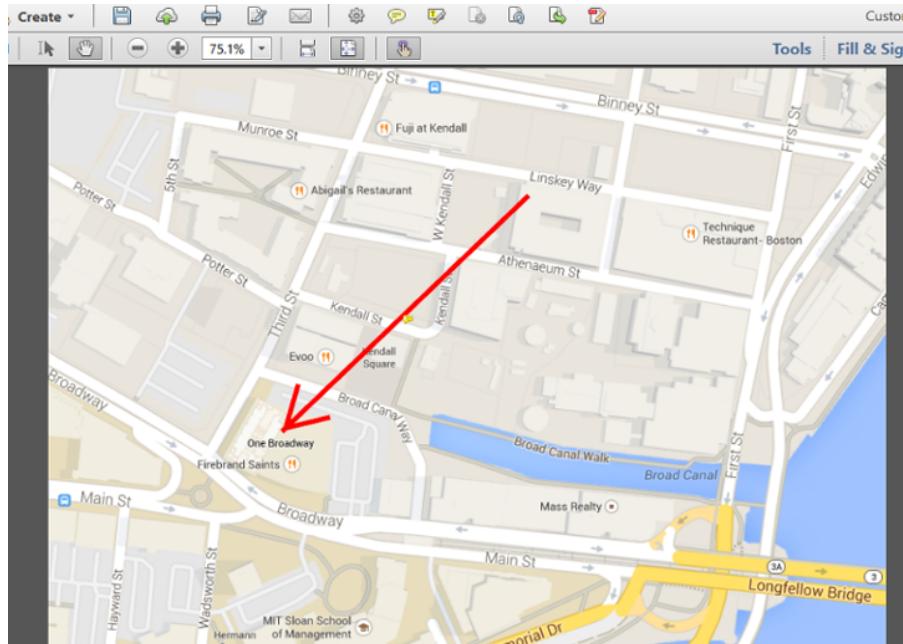
### Example 2: add a line annotation on top of a map

This is demonstrated in the [AddPointerAnnotation<sup>823</sup>](#) example:

```
Rectangle rect = new Rectangle(220, 350, 475, 595);
PdfAnnotation annotation = PdfAnnotation.createLine(writer, rect,
    "Cambridge Innovation Center", 225, 355, 470, 590);
PdfArray le = new PdfArray();
le.add(new PdfName("OpenArrow"));
le.add(new PdfName("None"));
annotation.setTitle("You are here:");
annotation.setColor(BaseColor.RED);
annotation.setFlags(PdfAnnotation.FLAGS_PRINT);
annotation.setBorderStyle(
    new PdfBorderDictionary(5, PdfBorderDictionary.STYLE_SOLID));
annotation.put(new PdfName("LE"), le);
annotation.put(new PdfName("IT"), new PdfName("LineArrow"));
writer.addAnnotation(annotation);
```

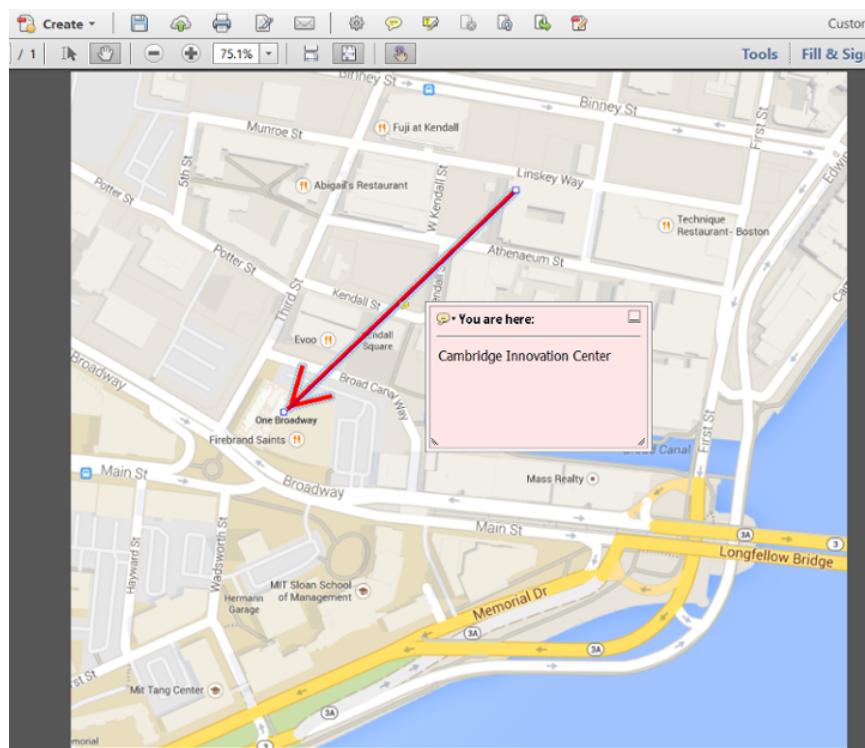
The result is an annotation (which isn't part of the real content, but part of an interactive layer on top of the real content):

<sup>823</sup><http://itextpdf.com/sandbox/annotations/AddPointerAnnotation>



Map with an annotation

It is interactive in the sense that extra info is shown when the user clicks the annotation:



Map with an annotation that has been opened

Many other options are possible, but once again: your question wasn't entirely clear.

## How to create a clickable polygon or path?



Is anyone out there able to create a clickable annotation that has an irregular shape?

I know I can create a rectangular one like this

```
float x1 = 100, x2 = 200, y1 = 150, y2 = 200;
iTextSharp.text.Rectangle r = new iTextSharp.text.Rectangle(x1, y1, x2, y2);
PdfName pfn = new PdfName(lnk.LinkID.ToString());
PdfAction ac = new PdfAction(lnk.linkUrl, false);
PdfAnnotation anno = PdfAnnotation.CreateLink(stamper.Writer, r, pfn, ac);
int page = 1;
stamper.AddAnnotation(anno, page);
```

Is there any way to do that with a graphics path?

Posted on StackOverflow on [Nov 22, 2014](#) <sup>824</sup> by [ksliman](#)<sup>825</sup>

The secret ingredient you are looking for is called **QuadPoints** ;-)

Allow me to explain how QuadPoints are used by showing you the [AddPolygonLink](#)<sup>826</sup> example.

First let's take a look at some code that constructs and draws a path:

```
canvas.moveTo(36, 700);
canvas.lineTo(72, 760);
canvas.lineTo(144, 720);
canvas.lineTo(72, 730);
canvas.closePathStroke();
```

I am using this code snippet only to show the irregular shape that we'll make clickable.

You already know how to create a clickable link with a rectangular shape:

```
Rectangle linkLocation = new Rectangle(36, 700, 144, 760);
PdfDestination destination = new PdfDestination(PdfDestination.FIT);
PdfAnnotation link = PdfAnnotation.createLink(stamper.getWriter(),
linkLocation, PdfAnnotation.HIGHLIGHT_INVERT,
1, destination);
```

This corresponds with the code snippet you already provided in your question.

Now let's introduce some QuadPoints:

---

<sup>824</sup><http://stackoverflow.com/questions/27083206/itextshape-clickable-polygon-or-path>

<sup>825</sup><http://stackoverflow.com/users/2328086/ksliman>

<sup>826</sup><http://itextpdf.com/sandbox/annotations/AddPolygonLink>

```
PdfArray array = new PdfArray(new int[]{72, 730, 144, 720, 72, 760, 36, 700});
link.put(PdfName.QUADPOINTS, array);
```

According to ISO-32000-1, QuadPoints are:

An array of  $8 \times n$  numbers specifying the coordinates of  $n$  quadrilaterals in default user space that comprise the region in which the link should be activated. The coordinates for each quadrilateral are given in the order

$x_1 \ y_1 \ x_2 \ y_2 \ x_3 \ y_3 \ x_4 \ y_4$

specifying the four vertices of the quadrilateral in counterclockwise order. For orientation purposes, such as when applying an underline border style, the bottom of a quadrilateral is the line formed by  $(x_1, y_1)$  and  $(x_2, y_2)$ .

If this entry is not present or the conforming reader does not recognize it, the region specified by the Rect entry should be used. QuadPoints shall be ignored if any coordinate in the array lies outside the region specified by Rect.

Note that I defined the linkLocation parameter in such a way that the irregular shape fits inside that rectangle.

**Caveat:** you can try this functionality by testing this example: [link\\_polygon.pdf<sup>827</sup>](#), but be aware that while this will work when viewing the file in Adobe Reader, this may not work with inferior PDF viewers that did not implement the QuadPoints functionality.

## How do I insert a hyperlink to another page with iTextSharp in an existing PDF?



I would like to add a link to an existing pdf that jumps to a coordinate on another page.

I am able to add a rectangle using this code:

```
PdfContentByte overContent = stamper.GetOverContent(1);
iTextSharp.text.Rectangle rectangle = new Rectangle(10,10,100,100,0);
rectangle.BackgroundColor = BaseColor.BLUE;
overContent.Rectangle(rectangle);
stamper.Close();
```

How can I do similar to create a link that is clickable?

Posted on StackOverflow on [Nov 19, 2012<sup>828</sup>](#) by [dazzler77<sup>829</sup>](#)

---

<sup>827</sup>[http://itextpdf.com/sites/default/files/link\\_polygon.pdf](http://itextpdf.com/sites/default/files/link_polygon.pdf)

<sup>828</sup><http://stackoverflow.com/questions/13448853/how-do-i-insert-a-hyperlink-to-another-page-with-itextsharp-in-an-existing-pdf>

<sup>829</sup><http://stackoverflow.com/users/261787/dazzler77>

You are already adding the Rectangle, now you need to add the annotation:

```
PdfAnnotation annotation = PdfAnnotation.CreateLink(
    stamper.Writer, rectangle, PdfAnnotation.HIGHLIGHT_INVERT,
    new PdfAction("http://itextpdf.com/"))

);
stamper.AddAnnotation(annotation, page);
```

In this code sample `page` is the number of the page where you want to add the link and `rectangle` is the `Rectangle` object defining the coordinates on that page.

## How to add a printable or non-printable bitmap stamp to a PDF?



I would like to add a bitmap stamp to a PDF file, that would be either printable or non-printable depending on the actual Acrobat Reader print settings. I.e.

- when a user selects the option “Document” in Adobe Reader’s Print Dialog box, it would not be printed, but
- when “Document and stamps” is selected, then the bitmap would print

Right now I can create either printable or non-printable bitmap, but I am unable to create a bitmap that would be both printable and non-printable depending on users choice.

Posted on StackOverflow on Nov 19, 2012<sup>830</sup> by Vojtěch Dohnal<sup>831</sup>

Creating stamp annotations is described in Chapter 7 of *iText in Action - Second Edition*<sup>832</sup>, more specifically in the `TimeTableAnnotations3`<sup>833</sup> example:

```
PdfAnnotation annotation = PdfAnnotation.createStamp(stamper.getWriter(),
    rect, "Press only", "NotForPublicRelease");
annotation.setFlags(PdfAnnotation.FLAGS_PRINT);
```

If you look at the print preview, you can see that these annotations don’t show up if you print the Document without stamps:

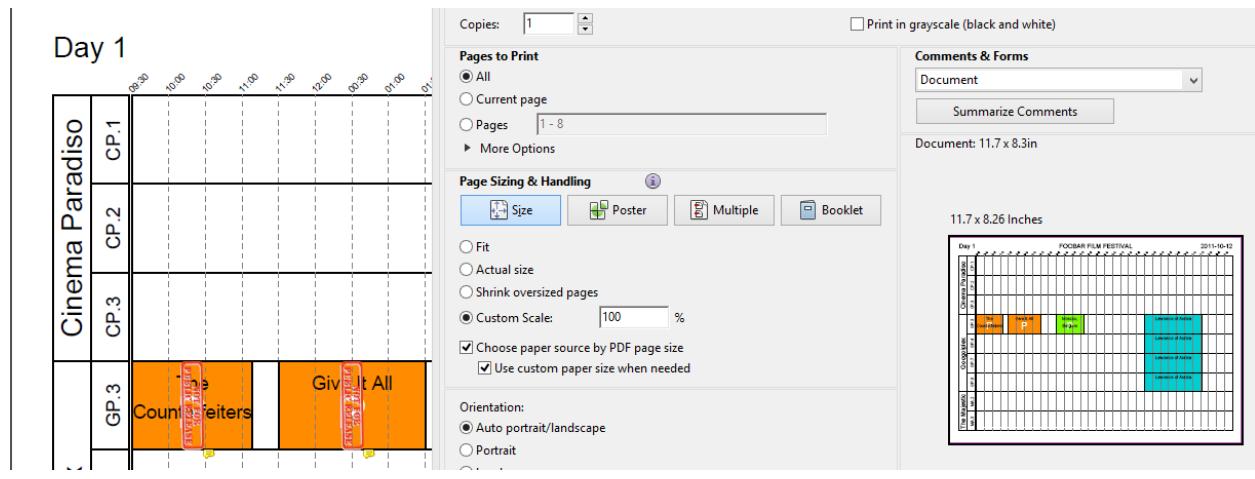
---

<sup>830</sup><http://stackoverflow.com/questions/29229629/how-to-add-a-printable-or-non-printable-bitmap-stamp-to-a-pdf>

<sup>831</sup><http://stackoverflow.com/users/2224701>

<sup>832</sup><http://itextpdf.com/book>

<sup>833</sup><http://itextpdf.com/examples/ii.php?id=151>



### Print Preview

In C#, the code is very similar to the Java code:

```
PdfAnnotation annotation = PdfAnnotation.CreateStamp(
    stamper.Writer, rect, "Press only", "NotForPublicRelease"
);
annotation.Flags = PdfAnnotation.FLAGS_PRINT;
```

Note that a PDF viewer should have predefined icons for at least the following names:

- Approved,
- Experimental,
- NotApproved,
- AsIs,
- Expired,
- NotForPublicRelease,
- Confidential,
- Final,
- Sold,
- Departmental,
- ForComment,
- TopSecret,
- Draft,
- ForPublicRelease.

What these icons look like will depend from viewer to viewer.

You want to use an image instead of one of these predefined stamps. This is shown in the [AddStamp<sup>834</sup>](#) example. We need to create an appearance for the stamp annotation and add it like this:

---

<sup>834</sup><http://itextpdf.com/sandbox/annotations/AddStamp>

```

public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    Image img = Image.getInstance(IMG);
    float w = img.getScaledWidth();
    float h = img.getScaledHeight();
    Rectangle location = new Rectangle(36, 770 - h, 36 + w, 770);
    PdfAnnotation stamp = PdfAnnotation.createStamp(
        stamper.getWriter(), location, null, "ITEXT");
    img.setAbsolutePosition(0, 0);
    PdfContentByte cb = stamper.getOverContent(1);
    PdfAppearance app = cb.createAppearance(w, h);
    app.addImage(img);
    stamp.setAppearance(PdfName.N, app);
    stamp.setFlags(PdfAnnotation.FLAGS_PRINT);
    stamper.addAnnotation(stamp, 1);
    stamper.close();
    reader.close();
}

```

## How to create a pop-up a window to display images and text?



I want to add and pop-up window in a C# project to display an image and text by clicking on an annotation:

```

iTextSharp.text.pdf.PdfAnnotation annot =
    iTextSharp.text.pdf.PdfAnnotation.CreateLink(
        stamper.Writer, c.rect, PdfAnnotation.HIGHLIGHT_INVERT,
        PdfAction.JavaScript("app.alert('action!')", stamper.Writer));

```

The above code is used to display an alert and I want to customize it to my needs. I'm not familiar with javascript. Are there any other options ?

Posted on StackOverflow on May 31, 2014 <sup>835</sup> by Buddhima Naween Rathnayake<sup>836</sup>

You need a couple of annotations to achieve what you want.

---

<sup>835</sup><http://stackoverflow.com/questions/23968916/pop-up-a-window-from-itextsharp-annotation-to-display-images-and-text>

<sup>836</sup><http://stackoverflow.com/users/3297553/buddhima-naween-rathnayake>

Let me start with a simple text annotation:

Suppose that:

- writer is your PdfWriter instance,
- rect1 and rect2 are rectangles that define coordinates,
- title and contents are string objects with the content you want to show in the text annotation,

Then you need this code snippet to add the popup annotations:

```
// Create the text annotation
PdfAnnotation text = PdfAnnotation.CreateText(writer, rect1, title, contents, false, "Comment");
text.Name = "text";
text.Flags = PdfAnnotation.FLAGS_READONLY | PdfAnnotation.FLAGS_NOVIEW;
// Create the popup annotation
PdfAnnotation popup = PdfAnnotation.CreatePopup(writer, rect2, null, false);
// Add the text annotation to the popup
popup.Put(PdfName.PARENT, text.IndirectReference);
// Declare the popup annotation as popup for the text
text.Put(PdfName.POPUP, popup.IndirectReference);
// Add both annotations
writer.AddAnnotation(text);
writer.AddAnnotation(popup);
// Create a button field
PushButtonField field = new PushButtonField(writer, rect1, "button");
PdfAnnotation widget = field.Field;
// Show the popup onMouseEnter
PdfAction enter = PdfAction.JavaScript(JS1, writer);
widget.SetAdditionalActions(PdfName.E, enter);
// Hide the popup onMouseExit
PdfAction exit = PdfAction.JavaScript(JS2, writer);
widget.SetAdditionalActions(PdfName.X, exit);
// Add the button annotation
writer.AddAnnotation(widget);
```

Two constants aren't explained yet:

JS1:

```
"var t = this.getAnnot(this.pageNum, 'text');
t.popupOpen = true;
var w = this.getField('button');
w.setFocus();"
```

JS2:

```
"var t = this.getAnnot(this.pageNum, 'text');
t.popupOpen = false;"
```

You can find a full example [here<sup>837</sup>](#).

If you also want an image, please take a look at this example: [advertisement.pdf<sup>838</sup>](#)

Here you have an advertisement that closes when you click “close this advertisement”. This is also done using JavaScript. You need to combine the previous snippet with the code of the [Advertisement<sup>839</sup>](#) example.

The key JavaScript methods you’ll need are: `getField()` and `getAnnot()`. You’ll have to change the properties to show or hide the content.

---

<sup>837</sup><http://itextpdf.com/examples/ii.php?id=152>

<sup>838</sup><http://examples.itextpdf.com/results/part2/chapter07/advertisement.pdf>

<sup>839</sup><http://itextpdf.com/examples/ii.php?id=143>

## How to stamp image on existing PDF and create an anchor?



I have an existing document, onto which I would like to stamp an image at an absolute position. I am able to do this, but I would also like to make this image clickable: when a user clicks on the image I would like the PDF to go to the last page of the document. Here is my code:

```
PdfReader readerOriginalDoc = new PdfReader("src/main/resources/test.pdf");
PdfStamper stamper = new PdfStamper(
    readerOriginalDoc, new FileOutputStream("NewStamper.pdf"));
PdfContentByte content = stamper.getOverContent(1);
Image image = Image.getInstance("src/main/resources/images.jpg");
image.scaleAbsolute(50, 20);
image.setAbsolutePosition(100, 100);
image.setAnnotation(new Annotation(0, 0, 0, 0, 3));
content.addImage(image);
stamper.close();
```

Any idea how to do this?

Posted on StackOverflow on Nov 17, 2014 <sup>840</sup> by BYJU SUKUMARAN<sup>841</sup>

You are using a technique that only works when creating documents from scratch.

Please take a look at the [AddImageLink<sup>842</sup>](#) example to find out how to add an image and a link to make that image clickable to an existing document:

```
public void manipulatePdf(String src, String dest)
    throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    Image img = Image.getInstance(IMG);
    float x = 10;
    float y = 650;
    float w = img.getScaledWidth();
    float h = img.getScaledHeight();
    img.setAbsolutePosition(x, y);
    stamper.getOverContent(1).addImage(img);
```

---

<sup>840</sup><http://stackoverflow.com/questions/26983703/itext-how-to-stamp-image-on-existing-pdf-and-create-an-anchor>

<sup>841</sup><http://stackoverflow.com/users/4263226/byju-sukumaran>

<sup>842</sup><http://itextpdf.com/sandbox/annotations/AddImageLink>

```
Rectangle linkLocation = new Rectangle(x, y, x + w, y + h);
PdfDestination destination = new PdfDestination(PdfDestination.FIT);
PdfAnnotation link = PdfAnnotation.createLink(stamper.getWriter(),
    linkLocation, PdfAnnotation.HIGHLIGHT_INVERT,
    reader.getNumberOfPages(), destination);
link.setBorder(new PdfBorderArray(0, 0, 0));
stamper.addAnnotation(link, 1);
stamper.close();
}
```

You already have the part about adding the image right. Note that I create parameters for the position of the image as well as its dimensions:

```
float x = 10;
float y = 650;
float w = img.getScaledWidth();
float h = img.getScaledHeight();
```

I use these values to create a Rectangle object:

```
1 Rectangle linkLocation = new Rectangle(x, y, x + w, y + h);
```

This is the location for the link annotation we are creating with the PdfAnnotation class. You need to add this annotation separately using the addAnnotation() method.

You can take a look at the result here: [link\\_image.pdf](http://itextpdf.com/sites/default/files/link_image.pdf)<sup>843</sup> If you click on the i icon, you jump to the last page of the document.

---

<sup>843</sup>[http://itextpdf.com/sites/default/files/link\\_image.pdf](http://itextpdf.com/sites/default/files/link_image.pdf)

## How to set the BaseUrl of an existing PDF document?



We're having trouble setting a BaseUrl using iTextSharp. We have used Adobes Implementation for this in the past, but we got some severe performance issues. So we switched to iTextSharp, which is aprox 10 times faster.

Adobe enabled us to set a base url for each document. We really need this in order to deploy our documents on different servers. But we cant seem to find the right code to do this.

This code is what we used with Adobe:

```
public bool SetBaseUrl(object js, string baseUrl)
{
    try
    {
        object result = js.GetType().InvokeMember("baseUrl", BindingFlags.SetProperty,
null, js, new Object[] {baseUrl});
        return result != null;
    }
    catch
    {
        return false;
    }
}
```

A lot of solutions describe how you can insert links in new or empty documents. But our documents already exist and do contain more than just text. We want to overlay specific words with a link that leads to one or more other documents. Therefore, its really important to us that we can insert a link without accessing the text itself. Maybe lay a box on top of these words and set its position (since we know where the words are located in the document)

We have tried different implementations, using the `setAction` method, but it doesn't seem to work properly. The result was in most cases, that we saw out box, but there was no link inside or associated with it. (the cursor didn't change and nothing happened, when I clicked inside the box)

Posted on StackOverflow on Jul 4, 2014 <sup>844</sup> by ChriSi<sup>845</sup>

I've made you a couple of examples.

First, let's take a look at [BaseUrl1<sup>846</sup>](#). In your comment, you referred to JavaScript, so I created a

---

<sup>844</sup><http://stackoverflow.com/questions/24568386/set-baseurl-of-an-existing-pdf-document>

<sup>845</sup><http://stackoverflow.com/users/3279180/chriSi>

<sup>846</sup><http://itextpdf.com/sandbox/interactive/BaseURL1>

document to which I added a snippet of document-level JavaScript:

```
writer.addJavaScript("this.baseURL = \"http://itextpdf.com/\\";");
```

This works perfectly in Adobe Acrobat, but when you try this in Adobe Reader, you get the following error:



NotAllowedError: Security settings prevent access to this property or method.  
Doc.baseURL:1:Document-Level:0000000000000000

This is consistent with the JavaScript reference for Acrobat where it is clearly indicated that special permissions are needed to change the base URL.

So instead of following your suggested path, I consulted ISO-32000-1 and I discovered that you can add a URI dictionary to the catalog with a Base entry. So I wrote a second example, [BaseUrl2<sup>847</sup>](#), where I add this dictionary to the root dictionary of the PDF:

```
PdfDictionary uri = new PdfDictionary(PdfName.URI);
uri.put(new PdfName("Base"), new PdfString("http://itextpdf.com/"));
writer.getExtraCatalog().put(PdfName.URI, uri);
```

Now the baseUrl works in both Acrobat and Reader.

Assuming that you want to add a baseUrl to existing documents, I wrote [BaseUrl3<sup>848</sup>](#). In this example, we add the same dictionary to the root dictionary of an existing PDF:

```
PdfReader reader = new PdfReader(src);
PdfDictionary uri = new PdfDictionary(PdfName.URI);
uri.put(new PdfName("Base"), new PdfString("http://itextpdf.com/"));
reader.getCatalog().put(PdfName.URI, uri);
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
stamper.close();
```

Using this code, you can change a link that points to “index.php” ([base\\_url.pdf<sup>849</sup>](#)) into a link that points to “[http://itextpdf.com/index.php](#)” ([base\\_url\\_3.pdf<sup>850</sup>](#)).

Now you can replace your Adobe license with a less expensive iTextSharp license ;-)

---

<sup>847</sup><http://itextpdf.com/sandbox/interactive/BaseURL2>

<sup>848</sup><http://itextpdf.com/sandbox/interactive/BaseURL3>

<sup>849</sup>[http://itextpdf.com/sites/default/files/base\\_url.pdf](http://itextpdf.com/sites/default/files/base_url.pdf)

<sup>850</sup>[http://itextpdf.com/sites/default/files/base\\_url\\_3.pdf](http://itextpdf.com/sites/default/files/base_url_3.pdf)

## How to set zoom level to pdf using iTextSharp?



I need to set the zoom level 75% to pdf file using iTextSharp. I am using following code to set the zoom level.

```
PdfReader reader = new PdfReader("input.pdf".ToString());
Document doc = Document(reader.GetPageSize(1));
doc.OpenDocument();
PdfWriter writer = PdfWriter.GetInstance(doc,
    new FileStream("Zoom.pdf", FileMode.Create));
PdfDestination pdfDest = new PdfDestination(
    PdfDestination.XYZ, 0, doc.PageSize.Height, 0.75f);
doc.Open();
PdfAction action = PdfAction.GotoLocalPage(1, pdfDest, writer);
writer.SetOpenAction(action);
doc.Close();
```

But I am getting the error “the page 1 was request but the document has only 0 pages” in the doc.Close();

Posted on StackOverflow on Jun 6, 2014 <sup>851</sup> by mail2vguna<sup>852</sup>

You need to use PdfStamper instead of PdfWriter. Please take a look at the [AddOpenAction](#)<sup>853</sup> example:

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    PdfDestination pdfDest = new PdfDestination(
        PdfDestination.XYZ, 0, reader.getPageSize(1).getHeight(), 0.75f);
    PdfAction action = PdfAction.gotoLocalPage(1, pdfDest, stamper.getWriter());
    stamper.getWriter().setOpenAction(action);
    stamper.close();
    reader.close();
}
```

The result is a PDF that opens with a zoom factor of 75%<sup>854</sup>.

<sup>851</sup><http://stackoverflow.com/questions/24087786/how-to-set-zoom-level-to-pdf-using-itextsharp>

<sup>852</sup><http://stackoverflow.com/users/3710891/mail2vguna>

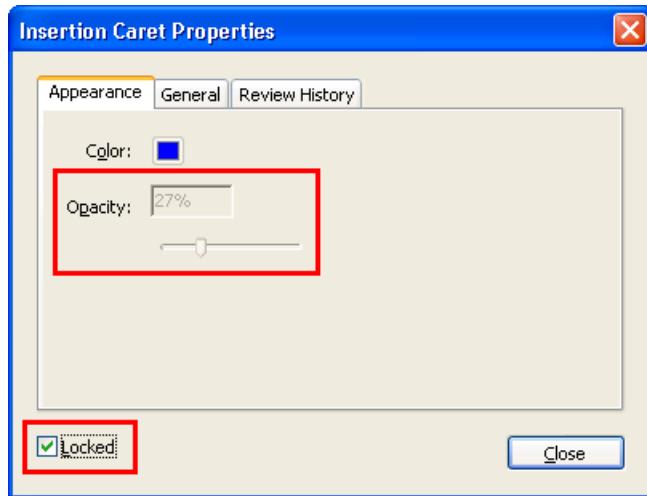
<sup>853</sup><http://itextpdf.com/sandbox/stamper/AddOpenAction>

<sup>854</sup>[http://itextpdf.com/sites/default/files/hello\\_open75p.pdf](http://itextpdf.com/sites/default/files/hello_open75p.pdf)

## How to change the properties of an annotation?



Hi I am adding Caret annotation in already existing PDF using iTextSharp in C#,  
 Now I want to made some changes in the Annotation properties, such as **Opacity** of color and **Locked**,



screen shot Acrobat

Posted on StackOverflow on Oct 23, 2013 <sup>855</sup> by Thirusanguraja Venkatesan<sup>856</sup>

Suppose that you have a `PdfAnnotation` object. This is a class that extends `PdfDictionary`.

To lock the annotation defined by this annotation dictionary, you need to set the `PdfAnnotation.FLAGS_LOCKED` flag, for instance with the `setFlags()` method:

```
annot.setFlags(PdfAnnotation.FLAGS_LOCKED);
```

Note that using this method will override the flags that were already defined before.

As for the opacity, that's define by the `ca` entry of the annotation dictionary.

```
annot.put(PdfName.ca, new PdfNumber(0.27));
```

My snippets are written in Java. You'll need to apply small changes to the methods if you want to use them in C# code.

<sup>855</sup><http://stackoverflow.com/questions/19538354/change-pdf-annotation-properties-using-itextsharp-c-sharp>

<sup>856</sup><http://stackoverflow.com/users/2027957/thirusanguraja-venkatesan>

## How to create a link to launch an external program?



I am using iTextSharp to create PDF files. Can I use iTextSharp to create a link in pdf that will allow me to launch a program.

The users of this file will have no problem trusting this file. It is part of the functionalities they requested to launch a certain program.

Posted on StackOverflow on [Feb 24, 2013](#) <sup>857</sup> by [user1019042](#)<sup>858</sup>

You are looking for a [Launch action](#)<sup>859</sup>. I'm the author of the book about iText, and I usually don't talk about this functionality because it's considered being a security hazard (which you point out in your comment: the user really has to trust the PDF).

In iTextSharp, you'd create a launch action like this:

```
Paragraph p = new Paragraph()
    new Chunk( "Click to open test.txt in Notepad." )
    .SetAction(
        new PdfAction(
            "c:/windows/notepad.exe",
            "test.txt", "open",
            Path.Combine(Utility.ResourceText, ""))
    )
);
document.Add(p);
```

Looking at the code, you immediately see a second problem: PDF is supposed to be platform dependent, but we're introducing two dependencies in this code sample:

1. in this sample we only provide a launch action for a PDF opened on Windows (one could add extra properties for other OSs).
2. we assume that the executable is present on the path we defined. That can be a huge problem if you want this PDF to work in every environment.

You'll have to talk this through with your customer to see if they can meet these extra requirements: OS and location of the executable.

---

<sup>857</sup><http://stackoverflow.com/questions/15048712/itextsharp-create-link-in-pdf-file-to-launch-a-program>

<sup>858</sup><http://stackoverflow.com/users/1019042/user1019042>

<sup>859</sup><http://itextpdf.com/themes/keyword.php?id=279>

## How to attach files to a PDF?



I am struggling with attaching files to a PDF that I am generating at runtime.

I manage to attach a file to the PDF, but I can't seem to reference them as links within the PDF.

iText in Action suggests I can do this as annotations or document level attachments. I don't find the book easy to follow or the code easy to understand though. There also seems to be scarce help on this around in the way of articles on the internet but apologies if I have missed anything.

Posted on StackOverflow on May 22, 2013<sup>860</sup> by DuncanOppaz<sup>861</sup>

I'm sorry you didn't like my book.

Did you read chapter 16? You want to embed a file as a document-level attachment like this:

```
PdfFileSpecification fs = PdfFileSpecification.FileEmbedded(writer, ... );
fs.AddDescription("specificname", false);
writer.AddFileAttachment(fs);
```

Inside the document, you want to create a link to that opens the PDF document described with the keyword "specificname". This is done through an action:

```
PdfTargetDictionary target = new PdfTargetDictionary(true);
target.EmbeddedFileName = "specificname";
PdfDestination dest = new PdfDestination(PdfDestination.FIT);
dest.AddFirst(new PdfNumber(1));
PdfAction action = PdfAction.GotoEmbedded(null, target, dest, true);
```

You can use this action for an annotation, a Chunk, etc... For instance:

```
Chunk chunk = new Chunk(" (see info)");
chunk.SetAction(action);
```

It is a common misconception to think that this will work for any attachment. However, ISO-32000-1 is very clear about the GotoE(mbedded) functionality:

<sup>860</sup><http://stackoverflow.com/questions/16687631/attaching-files-to-a-pdf>

<sup>861</sup><http://stackoverflow.com/users/1988003/duncanoppaz>

**12.6.4.4 Embedded Go-To Actions** An embedded go-to action (PDF 1.6) is similar to a remote go-to action but allows jumping to or from a PDF file that is embedded in another PDF file (see 7.11.4, “Embedded File Streams”).

If you meant to ask “I want to attach any file (such as a Docx, jpg,... file) to my PDF and add an action to the PDF that opens such a file upon clicking a link,” then you’re asking something that isn’t supported in the PDF specification.

Feel free to read ISO-32000-1. If you didn’t understand my book, you’ll have to do an extra effort trying to read the PDF standard...

## How to delete attachments in PDF using iText?



I am new to PDF and I use the following code to embed the file to PDF. However, I want to write another program to delete the embedded files. May I know how can I do it?

```
public void addAttachments(String src, String dest, String[] attachments)
    throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    for (int i = 0; i < attachments.length; i++) {
        addAttachment(stamper.getWriter(), new File(attachments[i]));
    }
    stamper.close();
}
protected void addAttachment(PdfWriter writer, File src) throws IOException {
    PdfFileSpecification fs = PdfFileSpecification.fileEmbedded(
        writer, src.getAbsolutePath(), src.getName(), null);
    writer.addFileAttachment(
        src.getName().substring(0, src.getName().indexOf('.')), fs);
}
```

Posted on StackOverflow on Oct 30, 2014<sup>862</sup> by brian<sup>863</sup>

Let me start by rewriting your code to add an embedded file.

---

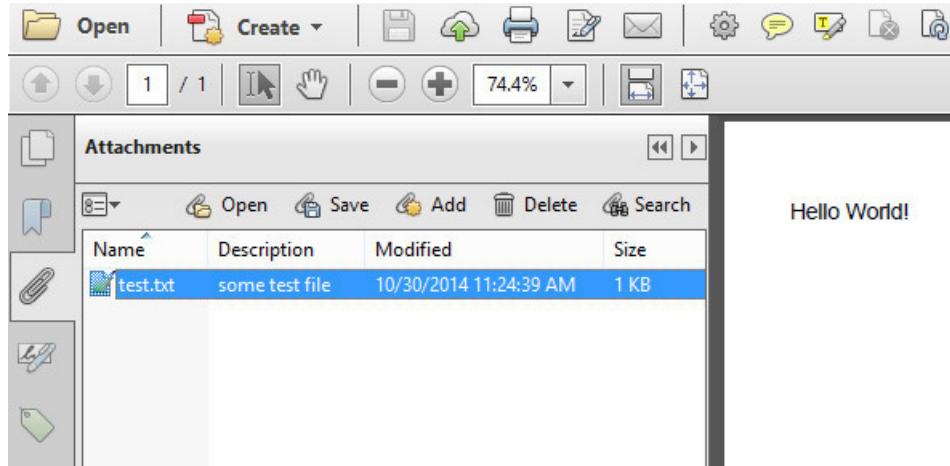
<sup>862</sup><http://stackoverflow.com/questions/26648462/how-to-delete-attachment-of-pdf-using-itext>

<sup>863</sup><http://stackoverflow.com/users/4197487/brian>

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    PdfFileSpecification fs = PdfFileSpecification.fileEmbedded(
        stamper.getWriter(), null, "test.txt", "Some test".getBytes());
    stamper.addFileAttachment("some test file", fs);
    stamper.close();
}
```

You can find the full code sample here: [AddEmbeddedFile<sup>864</sup>](#)

Now when we look at the Attachments panel of the resulting PDF, we see an attachment test.txt with description “some test file”:

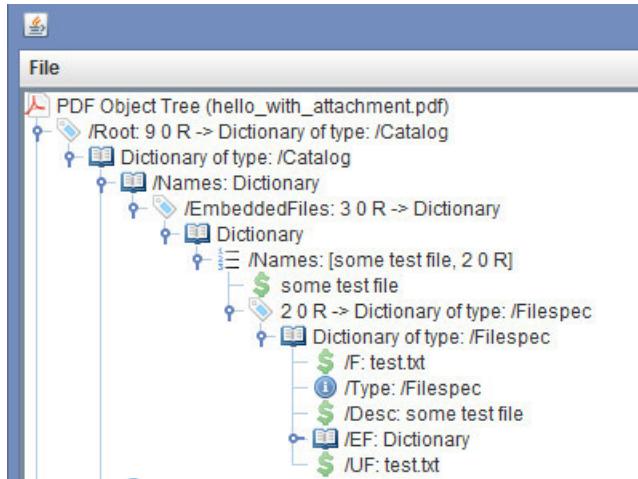


Attachment shown in Adobe Acrobat

After you have added this file, you now want to remove it. To do this, please use RUPS and take a look inside:

---

<sup>864</sup><http://itextpdf.com/sandbox/annotations/AddEmbeddedFile>



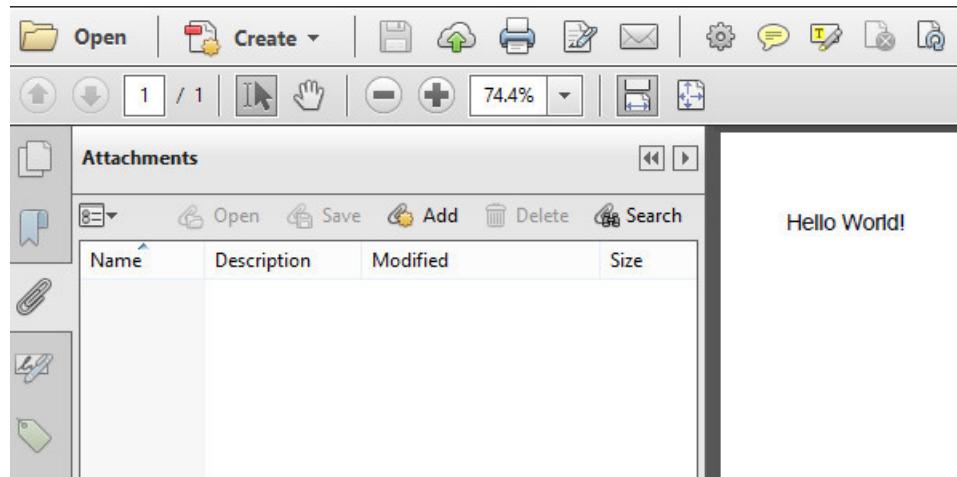
Attachment shown in iText RUPS

This gives us a hint on where to find the embedded file. Take a look at the code of the [RemoveEmbeddedFile](#)<sup>865</sup> example to see how we navigate through the object-oriented file format that PDF is:

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfDictionary root = reader.getCatalog();
    PdfDictionary names = root.getAsDict(PdfName.NAMES);
    PdfDictionary embeddedFiles = names.getAsDict(PdfName.EMBEDDEDFILES);
    PdfArray namesArray = embeddedFiles.getAsArray(PdfName.NAMES);
    namesArray.remove(0);
    namesArray.remove(0);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.close();
}
```

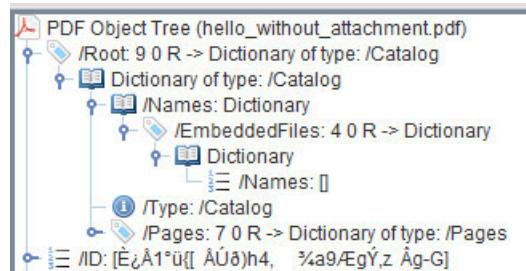
As you can see, we start at the root of the document (aka the catalog) and we walk via `Names` and `EmbeddedFiles` to the `Names` array. As I know that the embedded file I want to remove is the first in the array, I remove the name and value by removing the element with index 0 twice. This first removes the description, then the reference to the file. The attachment is now gone:

<sup>865</sup><http://itextpdf.com/sandbox/annotations/RemoveEmbeddedFile>



Attachment is gone in Adobe Acrobat

As there was only one embedded file in my example, I now see an empty array when I look inside the PDF:



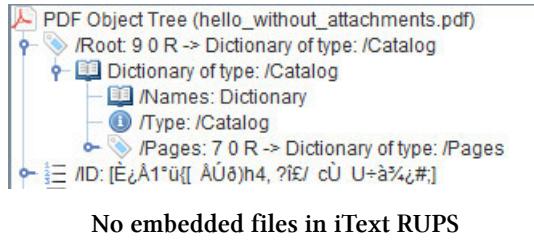
Empty array in iText RUPS

If you want to remove all the embedded files at once, the code is even easier. That is shown in the [RemoveEmbeddedFiles<sup>866</sup>](#) example:

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfDictionary root = reader.getCatalog();
    PdfDictionary names = root.getAsDict(PdfName.NAMES);
    names.remove(PdfName.EMBEDDEDFILES);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.close();
}
```

Now we don't even look at the entries of the `EmbeddedFiles` dictionary. There is no longer such an entry in the `Names` dictionary:

<sup>866</sup><http://itextpdf.com/sandbox/annotations/RemoveEmbeddedFiles>



## How to create a JavaScript action to open the attachments panel?



Is there an action which opens the attachments panel? Not right away, but when the user presses some text?

I know the `writer.setViewerPreferences(PdfWriter.PageModeUseAttachments)` but I don't want it to open right away.

Posted on StackOverflow on Sep 12, 2012 <sup>[867](#)</sup> by Michael A <sup>[868](#)</sup>

This can be done using a JavaScript action:

```
Chunk c = new Chunk("Show / Hide attachment panel");
c.setAction(PdfAction.javaScript(
    "app.execMenuItem('ShowHideFileAttachment');", writer));
document.add(new Paragraph(c));
```

Note that this won't work on all viewers (app = Adobe Reader) and it won't work if people disable Javascript.

<sup>867</sup><http://stackoverflow.com/questions/12383187/action-to-open-attachments-panel>

<sup>868</sup><http://stackoverflow.com/users/498727/michael-a>

## How to add an onMouseOver javaScript action to a TextField?



How can I add javascripts to iTextsharp created textfields? I've tried following:

```
TextField field = new iTextSharp.text.pdf.TextField(writer,
    new iTextSharp.text.Rectangle(x, y - h, x + w, y), name);
field.BackgroundColor =
    new BaseColor(bgcolor[0], bgcolor[1], bgcolor[2]);
field.BorderColor =
    new BaseColor(bordercolor[0], bordercolor[1], bordercolor[2]);
field.BorderWidth = border;
fieldBorderStyle = PdfBorderDictionary.STYLE_SOLID;
field.Text = text;
// PROBLEM:
// field.AddJavaScript = PdfAction.JavaScript(
// "this.getField(\"Total_0\").value = ( this.getField(\"Quantity_0\").value
// * this.getField(\"Price_0\").value ) / this.getField(\"Multiplier_0\").value;
",
// writer);
writer.AddAnnotation(field.GetTextField());
```

The problem is: `iTextSharp.text.pdf.TextField` does not contain a definition for `AddJavaScript`. So then how can I add javascript that activates when mouse cursor is over text field or when text field has been edited? My purpose is to calculate value from other text fields to this one with Javascript.

Posted on StackOverflow on Sep 20, 2013 <sup>869</sup> by mikessu<sup>870</sup>

What you're looking for is called an *additional action*. For instance: you have an *entry* action, defined using `PdfName.E` and an *exit* action, defined by `PdfName.X`. The entry action is triggered when the mouse enters the rectangle that defines the field; the exit action is triggered when the mouse exist the rectangle that defines the field.

In your code, you're skipping a step and that's probably why you didn't find the function you need:

```
PdfFormField ffield = field.GetTextField();
ffield.SetAdditionalActions(PdfName.E, PdfAction("app.alert('action!')"));
writer.AddAnnotation(ffield);
```

---

<sup>869</sup><http://stackoverflow.com/questions/18914946/adding-onmouseover-javascript-to-textfield>

<sup>870</sup><http://stackoverflow.com/users/2138736/mikessu>

This snippet will cause an alert to appear when the mouse enters the text field. Other options are `PdfName.D` (`mouseDown`), `PdfName.U` (`mouseUp`), `PdfName.K` (keystroke by user), `PdfName.V` (validate, because the value of the field has changed), etc.

## How to define multiple actions for a PushbuttonField?



In a PDF, I can set up a button to:

1. Submit the FDF data to my RestAPI
2. Redirect to a webpage (to tell user 'Thanks') and that works perfectly.

But I need to do this using code... I am using iTextSharp in the following way:

- `pb` is my pushbutton...
- `sURL` is the URL I need the data to go to...

This is my code:

```
PdfFormField pff = pb.Field;
pff.SetAdditionalActions(PdfName.A, PdfAction.CreateSubmitForm(sURL, null, PdfAction.SUBMIT_XFDF));
af.ReplacePushButtonField("submit", pff);
```

This does replace the button with the submit action. How do I keep the page redirect or how do I set it in code? I have also tried `PdfName_AA` and `PdfName_U` based on examples and I am not certain what the different names are for. Unfortunately, none of them solve my problem.

Apparently, the `SetAdditionalActions()` replaces the existing actions. I have also tried using annotations and just adding a button that doesn't exist.

Posted on StackOverflow on Apr 18, 2014<sup>871</sup> by JohnKochJr<sup>872</sup>

You are looking for a concept called *chained actions*:

---

<sup>871</sup><http://stackoverflow.com/questions/23161573/itextsharp-multiple-actions-for-pushbuttonfield>

<sup>872</sup><http://stackoverflow.com/users/3396436/johnkochjr>

```
Chunk chunk = new Chunk("print this page");
PdfAction action = PdfAction.javaScript(
    "app.alert('Think before you print!');", stamper.getWriter());
action.next(PdfAction.javaScript(
    "printCurrentPage(this.pageNum);", stamper.getWriter()));
action.next(new PdfAction("http://www.panda.org/savepaper/"));
chunk.setAction(action);
```

In this case, we have an action that shows an alert, prints a page and redirects to an URL. These actions are chained to each other using the `next()` method.

In C#, this would be:

```
Chunk chunk = new Chunk("print this page");
PdfAction action = PdfAction.JavaScript(
    "app.alert('Think before you print!');", stamper.Writer);
action.Next(PdfAction.JavaScript(
    "printCurrentPage(this.pageNum);", stamper.Writer));
action.Next(new PdfAction("http://www.panda.org/savepaper/"));
chunk.SetAction(action);
```

## How to add an “In Reply To” annotation?



I am trying to add a sticky note reply to in pdf using iTextSharp. I am able to create a new annotation in the pdf. But I cannot link it as child of an already existing annotation. I copied most of the properties in parent to its child. I copied it by analyzing the properties of a reply, by manually adding a reply from Adobe Reader. What I am missing is the property /IRT. It needs a reference to the parent popup. Like /IRT 16 0 R. How do I find this reference?

Posted on StackOverflow on Feb 11, 2015<sup>873</sup> by Jose Tuttu<sup>874</sup>

Please take a look at the [AddInReplyTo<sup>875</sup>](#) example.

We have a file named [hello\\_sticky\\_note.pdf<sup>876</sup>](#) that looks like this:

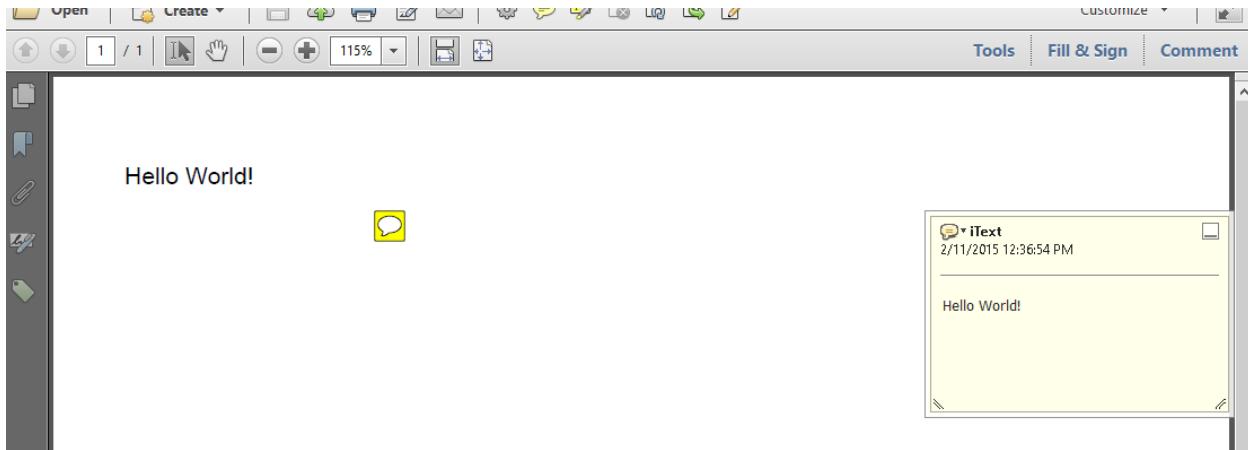
---

<sup>873</sup><http://stackoverflow.com/questions/28450668/how-to-add-in-reply-to-annotation-using-itextsharp>

<sup>874</sup><http://stackoverflow.com/users/2310478/jose-tuttu>

<sup>875</sup><http://itextpdf.com/sandbox/annotations/AddInReplyTo>

<sup>876</sup>[http://itextpdf.com/sites/default/files/hello\\_sticky\\_note.pdf](http://itextpdf.com/sites/default/files/hello_sticky_note.pdf)



PDF with a sticky note

In my example, I know that this annotation is the first entry in the `/Annots` array (the annotation with index 0), so this is how I'm going to add an "in reply to" annotation:

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfDictionary page = reader.getPageN(1);
    PdfArray annots = page.getAsArray(PdfName.ANNOTS);
    PdfDictionary sticky = annots.getAsDict(0);
    PdfArray stickyRect = sticky.getAsArray(PdfName.RECT);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    PdfWriter writer = stamper.getWriter();
    Rectangle stickyRectangle = new Rectangle(
        stickyRect.getAsNumber(0).floatValue(),
        stickyRect.getAsNumber(1).floatValue(),
        stickyRect.getAsNumber(2).floatValue(),
        stickyRect.getAsNumber(3).floatValue()
    );
    PdfAnnotation replySticky = PdfAnnotation.createText(
        writer, stickyRectangle, "Reply", "Hello PDF", true, "Comment");
    replySticky.put(PdfName.IRT, annots.getAsIndirectObject(0));
    stamper.addAnnotation(replySticky, 1);
    stamper.close();
}
```

I get the original annotation (in my code, it's named `sticky`) and I get the position of that annotation (`stickyRect`). I create a `stickyRectangle` object and I use that `stickyRectangle` to create a new `PdfAnnotation` named `replySticky`. That's what you already have. Now I add the missing part, that is the reference that you were asking for:

```
replySticky.put(PdfName.IRT, annots.getAsIndirectObject(0));
```

The resulting PDF looks like [hello\\_in\\_reply\\_to.pdf<sup>877</sup>](#):

![PDF with a sticky note and a reply to this note][2]



The /NM property is missing for the reply. I thought, it would be generated automatically.  
Is it reliable to save NM property in an external DB?

The /NM entry is defined as follows:

(Optional) The annotation name, a text string uniquely identifying it among all the annotations on its page.

You can choose which ever String you want for the Reply as long as there is no other annotation with the same name on the page:

```
replySticky.put(PdfName.NM, new PdfString("MYID0123456789"));
```

You can save this name in an external DB along with the ID of the PDF and the page number of the page to which the annotation is added.

## How to get the author of a “free text” annotation?



Is it possible to get the author of a free text annotation using iText? I can retrieve the /Type and /Contents, but I can't find a way to get the author

Posted on StackOverflow on [Feb 28, 2015<sup>878</sup>](#) by [john renfrew<sup>879</sup>](#)

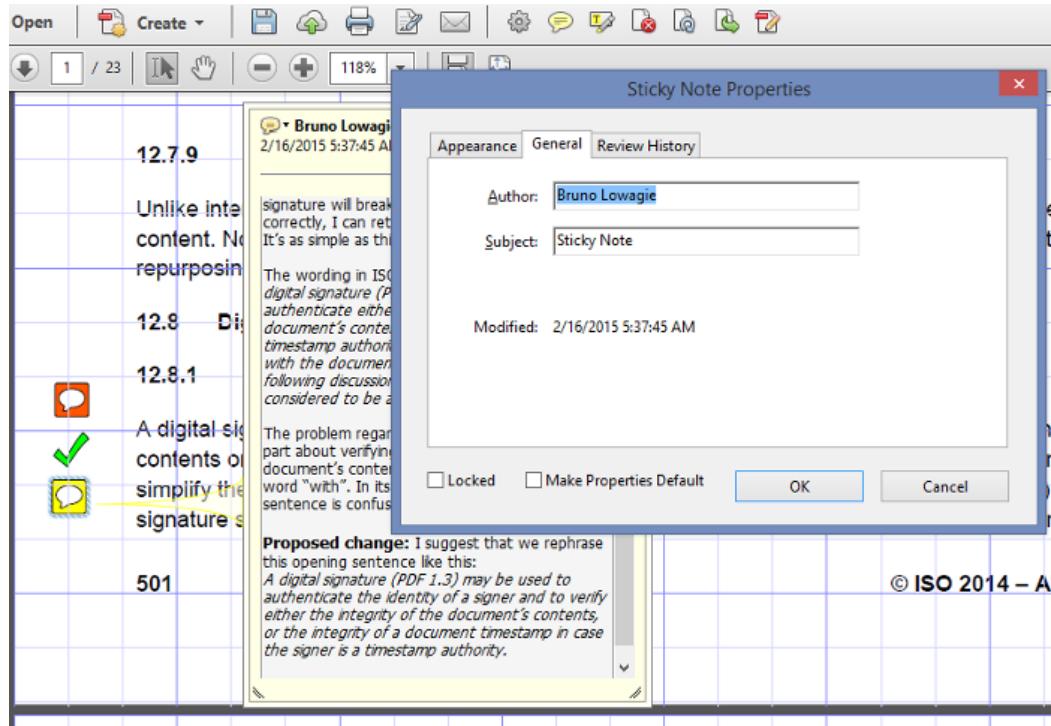
If the author value is present, you'll find it in the /T entry. Please consult ISO-32000-1 table 170 “Additional entries specific to markup annotations”. It defines a key named /T that is described as an optional *text string* that *shall be displayed in the title bar of the annotation's pop-up window when open and active. This entry shall identify the user who added the annotation.* The value you seek is not available for every type of annotations, only for markup annotations.

<sup>877</sup>[http://itextpdf.com/sites/default/files/hello\\_in\\_reply\\_to.pdf](http://itextpdf.com/sites/default/files/hello_in_reply_to.pdf)

<sup>878</sup><http://stackoverflow.com/questions/28788217/how-to-get-the-author-of-a-pdf-annotation>

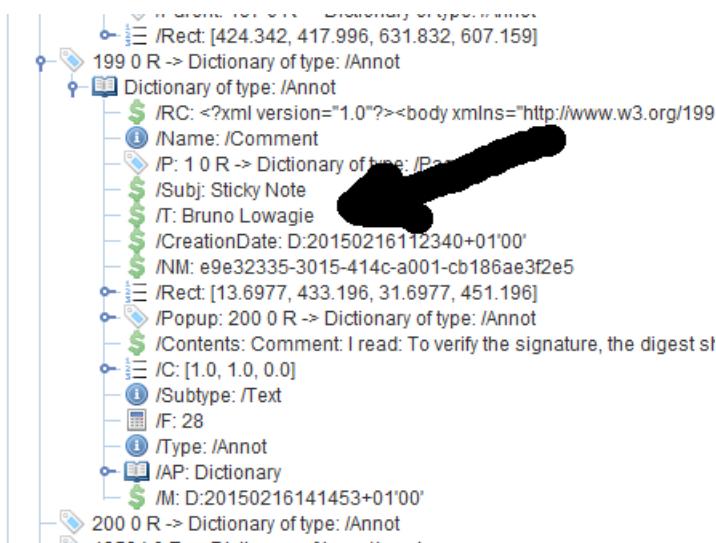
<sup>879</sup><http://stackoverflow.com/users/370092/john-renfrew>

So when you have this in Adobe Acrobat:



Annotations in a PDF

You'll find this inside the PDF:



Annotations in iText RUPS

You already have the /Contents and /Type entry, now you should also look for the /T entry. If it is missing, the author of the annotation can not be retrieved.

## How to change the author name for comments?



Does anyone have a script which will change the name of the author whenever any markup such as underlining, highlighting, etc, is applied? You can do this manually through the properties box, but this is time consuming and difficult.

Posted on LinkedIn on Mar 19, 2015<sup>880</sup> by Paul Tredoux

I have created a small [ChangeAuthor<sup>881</sup>](#) example that changes the author from “iTExt” to “Bruno Lowagie” using iTExt in Java.

It takes a [PDF with annotations created by “iTExt”<sup>882</sup>](#) and turns it into [a PDF with annotations created by “Bruno Lowagie”<sup>883</sup>](#).

The code is really simple: we loop over the annotations of a page and we change the T entry (if present and if equal to “iTExt”). We use PdfStamper to persist the changed PDF.

```
public void manipulatePdf(String src, String dest)
    throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfDictionary pageDict = reader.getPageN(1);
    PdfArray annots = pageDict.getAsArray(PdfName.ANNOTS);
    if (annots != null) {
        PdfDictionary annot;
        for (int i = 0; i < annots.size(); i++) {
            changeAuthor(annots.getAsDict(i));
        }
    }
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.close();
}

public void changeAuthor(PdfDictionary annot) {
    if (annot == null) return;
    PdfString t = annot.getAsString(PdfName.T);
    if (t == null) return;
    if ("iTExt".equals(t.toString()))
        annot.put(PdfName.T, new PdfString("Bruno Lowagie"));
}
```

---

<sup>880</sup><https://www.linkedin.com/groups/Script-Change-Author-Name-Comments-159987.S.5984062085800144899>

<sup>881</sup><http://itextpdf.com/sandbox/interactive/ChangeAuthor>

<sup>882</sup>[http://itextpdf.com/sites/default/files/page229\\_annotations.pdf](http://itextpdf.com/sites/default/files/page229_annotations.pdf)

<sup>883</sup>[http://itextpdf.com/sites/default/files/page229\\_changed\\_author.pdf](http://itextpdf.com/sites/default/files/page229_changed_author.pdf)

## How to change the color of a circle annotation?



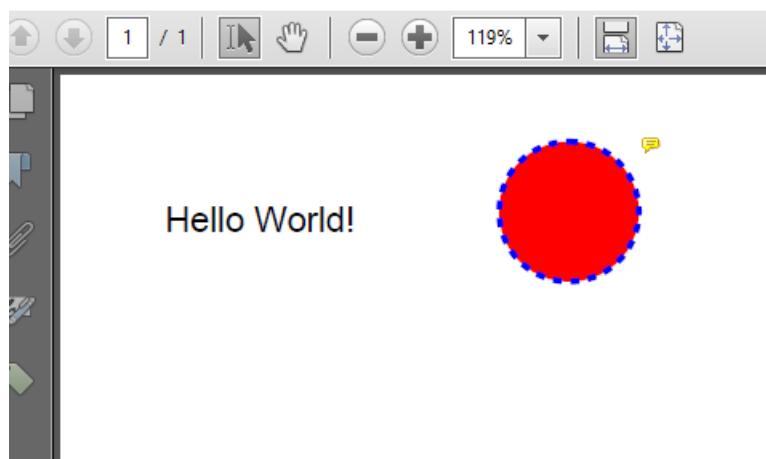
I am adding a SquareCircle annotation to an already existing PDF using iTextSharp in C#. Now I want to change the Fill Color annotation property, but I don't know how. When opening the PDF in Acrobat, the fill color property is in the appearance tab of the annotation properties.



Appearance tab

Posted on StackOverflow on Mar 26, 2015 <sup>884</sup> by Lupetto Burlone<sup>885</sup>

Please take a look at the [CircleAnnotation<sup>886</sup>](#) example. It creates a circle annotation with a blue border and red as the interior color:



Colored circle annotation

The code to add this annotation looks like this:

<sup>884</sup><http://stackoverflow.com/questions/29275194/how-to-change-the-color-of-a-circle-annotation>

<sup>885</sup><http://stackoverflow.com/users/4715828/lupetto-burlone>

<sup>886</sup><http://itextpdf.com/sandbox/annotations/CircleAnnotation>

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    Rectangle rect = new Rectangle(150, 770, 200, 820);
    PdfAnnotation annotation = PdfAnnotation.createSquareCircle(
        stamper.getWriter(), rect, "Circle", false);
    annotation.setTitle("Circle");
    annotation.setColor(BaseColor.BLUE);
    annotation.setFlags(PdfAnnotation.FLAGS_PRINT);
    annotation.setBorder(new PdfBorderArray(0, 0, 2, new PdfDashPattern()));
    annotation.put(PdfName.IC, new PdfArray(new int[]{1, 0, 0}));
    stamper.addAnnotation(annotation, 1);
    stamper.close();
}
```

I based this example on an example from the [official documentation<sup>887</sup>](#), more specifically the [MovieTemplates<sup>888</sup>](#) example.

The only thing I added was the line that sets the interior color:

```
annotation.put(PdfName.IC, new PdfArray(new int[]{1, 0, 0}));
```

If you need a C# example: the examples I wrote for my book were ported to C#, you can find them here: [chapter 7: C# examples<sup>889</sup>](#) It shouldn't be a problem to change the `put()` into `Put()` to make it work for you.

#### Caveat:

some PDF viewers (such as Chrome PDF viewer) are not full PDF viewers. They don't support every type of annotation. For instance, if you open [hello\\_circle.pdf<sup>890</sup>](#) in Chrome, you won't see the annotation. That is not a problem caused by the PDF (nor iTextSharp), that is a viewer problem.

---

<sup>887</sup><http://itextpdf.com/book>

<sup>888</sup><http://itextpdf.com/examples/ii.php?id=151>

<sup>889</sup><http://tinyurl.com/itextsharpIIA2C07>

<sup>890</sup>[http://itextpdf.com/sites/default/files/hello\\_circle.pdf](http://itextpdf.com/sites/default/files/hello_circle.pdf)

# Extracting text from PDFs

iText can parse PDFs to extract the content of a page. As there are many different ways to create a PDF file, and as the text on a page usually isn't more than a bunch of characters drawn on a page, it's not trivial to extract text correctly.

## How to remove text from a PDF?



Is it possible to remove all text occurrences contained in a specified area (red color rectangle area) of a pdf document?



I want the text to be removed, not merely covered.

Posted on StackOverflow on [jan 12, 2015](#) <sup>891</sup> by [amar](#)<sup>892</sup>

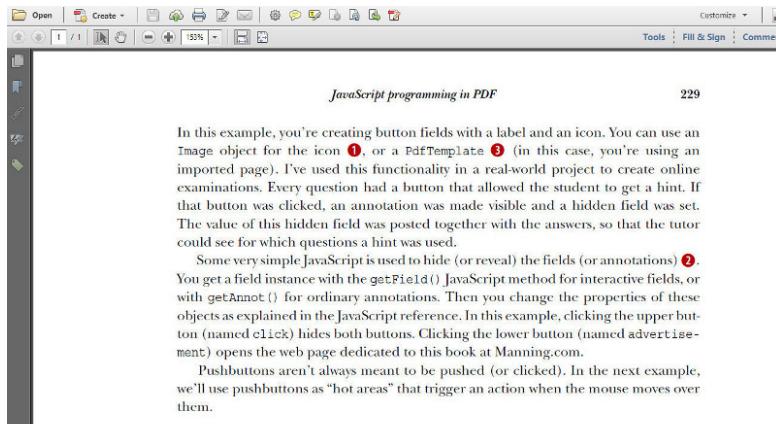
Please take a look at the [RemoveContentInRectangle](#)<sup>893</sup> example.

Let's say we have the following page:

<sup>891</sup><http://stackoverflow.com/questions/27905740/remove-text-occurrences-contained-in-a-specified-area-with-itext>

<sup>892</sup><http://stackoverflow.com/users/4316823/amar>

<sup>893</sup><http://itextpdf.com/sandbox/parse/RemoveContentInRectangle>



Original PDF

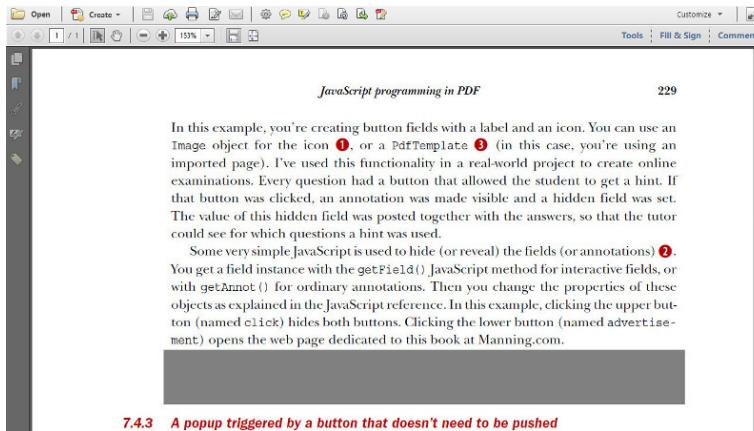
Now we want to remove all the text in the rectangle defined by the coordinates:  $llx = 97$ ,  $lly = 405$ ,  $urx = 480$ ,  $ury = 445$ ] (where  $ll$  stands for lower-left and  $ur$  stands for upper-right).

We can now use the following code:

```
public void manipulatePdf(String src, String dest)
    throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    List<PdfCleanUpLocation> cleanUpLocations =
        new ArrayList<PdfCleanUpLocation>();
    cleanUpLocations.add(new PdfCleanUpLocation(
        1, new Rectangle(97f, 405f, 480f, 445f), BaseColor.GRAY));
    PdfCleanUpProcessor cleaner =
        new PdfCleanUpProcessor(cleanUpLocations, stamper);
    cleaner.cleanUp();
    stamper.close();
    reader.close();
}
```

As you see, we define a list of `PdfCleanUpLocation` objects. To this list, we add a `PdfCleanUpLocation` passing the page number, a `Rectangle` defining the area we want to clean up, and a color that will show the area where content has been removed.

We then pass this list of `PdfCleanUpLocations` to the `PdfCleanUpProcessor` along with the `PdfStamper` instance. We invoke the `cleanUp()` method and when we close the `PdfStamper` instance, we get the following result:



Text in gray area has been completely removed

You can inspect this file: you will no longer be able to select any text in the gray area. All the text inside that rectangle has been removed. Note that this code sample will only work if you add the `iText-xtra.jar` to your CLASSPATH (`iText-xtra` is shipped with `iText core`). It will only work with versions equal to or higher than `iText 5.5.4`.

## How to create and apply redactions?



Is there any way to implement PDF redaction using iText? Working with the Acrobat SDK API I found that redactions also just seem to be annotations with the subtype “Redact”.

With the Acrobat SDK the code would like simply like this: `AcroPDAnnot annot = page.AddNewAnnot(-1, "Redact", rect) as AcroPDAnnot;` Unfortunately, I haven't be able to apply them though as `annot.Perform(avDoc)` does not seem to work.

In iTextSharp I can create simple text annotations like this `PdfAnnotation annotation = PdfAnnotation.CreateText(stamper.Writer, rect, "Title", "Content", false, null);` but there's no method to create Redact annotations.

The only other option I found so was was to create black rectangles as explained [here<sup>894</sup>](#), but that doesn't remove the text (it can still be selected). I want to create redaction annotations and eventually apply redaction.

Posted on StackOverflow on Jun 4, 2014 <sup>895</sup> by [silent<sup>896</sup>](#)

### Answer 1: Creating redaction annotations

iText is a toolbox that gives you the power to create any object you want. You are using a *convenience method* to create a Text annotation. That's scratching the surface.

<sup>894</sup><http://stackoverflow.com/questions/21308383/how-to-redact-pdf-using-itextsharp>

<sup>895</sup><http://stackoverflow.com/questions/24037282/how-to-create-and-apply-redactions>

<sup>896</sup><http://stackoverflow.com/users/1537195/silent>

You can use iText to create any type of annotation you want, because the `PdfAnnotation` class extends the `PdfDictionary` class.

Take a look at the [GenericAnnotations<sup>897</sup>](#) is the example that illustrates this functionality.

If we port this example to C#, we have:

```
PdfAnnotation annotation = new PdfAnnotation(writer, rect);
annotation.Title = "Text annotation";
annotation.Put(PdfName.SUBTYPE, PdfName.TEXT);
annotation.Put(PdfName.OPEN, PdfBoolean.PDFFALSE);
annotation.Put(PdfName.CONTENTS,
    new PdfString(string.Format("Icon: {0}", text))
);
annotation.Put(PdfName.NAME, new PdfName(text));
writer.AddAnnotation(annotation);
```

This is a *manual* way to create a text annotation. You want a Redact annotations, so you'll need something like this:

```
PdfAnnotation annotation = new PdfAnnotation(writer, rect);
annotation.Put(PdfName.SUBTYPE, new PdfName("Redact"));
writer.AddAnnotation(annotation);
```

You can use the `Put()` method to add all the other keys you need for the annotation.



As I finally got around to create a working example I wanted to share it here. It does not apply the redactions in the end but it creates valid redactions which are properly shown within Acrobat and can then be applied manually.

```
using (Stream stream = new FileStream(
    fileName, FileMode.Open, FileAccess.Read, FileShare.ReadWrite)) {
    PdfReader pdfReader = new PdfReader(stream);
    using (PdfStamper stamper = new PdfStamper(
        pdfReader, new FileStream(newFileName, FileMode.OpenOrCreate))) {
        int page = 1;
        iTextSharp.text.Rectangle rect =
            new iTextSharp.text.Rectangle(500, 50, 200, 300);
        PdfAnnotation annotation = new PdfAnnotation(stamper.Writer, rect);
        annotation.Put(PdfName.SUBTYPE, new PdfName("Redact"));
```

---

<sup>897</sup><http://itextpdf.com/examples/ii.php?id=147>

```

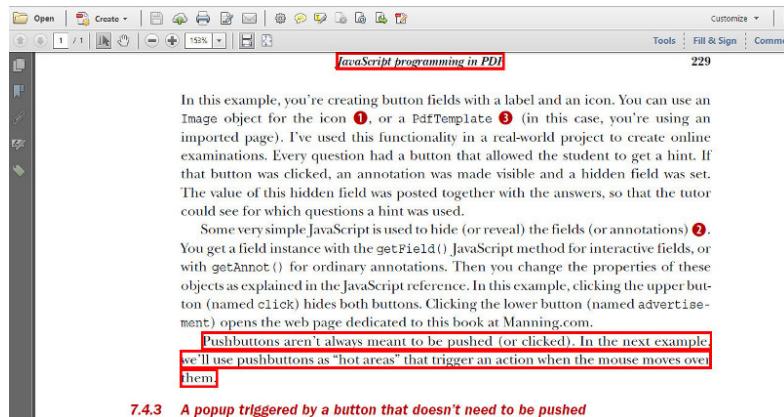
annotation.Title = "My Author";
annotation.Put(new PdfName("Subj"), new PdfName("Redact"));
float[] fillColor = { 0, 0, 0 };
annotation.Put(new PdfName("IC"), new PdfArray(fillColor));
float[] fillColorRed = { 1, 0, 0 };
annotation.Put(new PdfName("OC"), new PdfArray(fillColorRed));
stamper.AddAnnotation(annotation, page);
}
}

```

### Answer 2: Applying redaction

The second question requires the iText-xtra.jar (an extra jar shipped with iText) and you need at least iText 5.5.4. The approach to add opaque rectangles doesn't apply redaction: the redacted content is merely covered, not removed. You can still select the text and copy/paste it. If you're not careful, you risk ending up with a so-called [PDF Blackout Folly](#)<sup>898</sup>. See for instance the NSA / AT&T scandal.

Suppose that you have a file to which you added some redaction annotations: [page229\\_redacted.pdf](#)<sup>899</sup>



### A PDF with redaction annotations

We can now use this code to remove the content marked by the redaction annotations:

<sup>898</sup><http://slashdot.org/story/06/06/22/138210/more-pdf-blackout-follies>

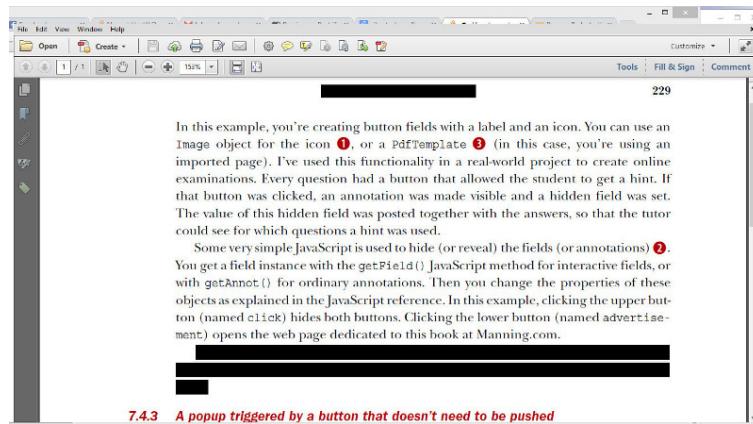
<sup>899</sup>[http://itextpdf.com/sites/default/files/page229\\_redacted.pdf](http://itextpdf.com/sites/default/files/page229_redacted.pdf)

```

public void manipulatePdf(String src, String dest)
    throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    PdfCleanUpProcessor cleaner = new PdfCleanUpProcessor(stamper);
    cleaner.cleanUp();
    stamper.close();
    reader.close();
}

```

This results in the following PDF: [page229\\_apply\\_redacted.pdf](#)<sup>900</sup>



A redacted PDF

As you can see, the red rectangle borders are replaced by filled black rectangles. If you try to select the original text, you'll notice that is is no longer present.

## What are the extra characters in the font name of my PDF?



while extracting font name from PDF, I get some junk characters followed by plus sign and then the font name with font style. I want to remove the junk characters. I get those junk characters only for a few PDF file, for example: MMLPEO+RemingtonNoiseless

```
string curFont = renderInfo.GetFont().PostscriptFontName;
```

Posted on StackOverflow on May 16, 2013 <sup>901</sup> by pdp<sup>902</sup>

<sup>900</sup>[http://itextpdf.com/sites/default/files/page229\\_apply\\_redacted.pdf](http://itextpdf.com/sites/default/files/page229_apply_redacted.pdf)

<sup>901</sup><http://stackoverflow.com/questions/16580270/extra-characters-while-extracting-font-family-from-pdf>

<sup>902</sup><http://stackoverflow.com/users/2185037/pdp>

The “junk” characters indicate that the font isn’t embedded completely. You’ll find names such as ABC123+RemingtonNoiseless, XYZ456+RemingtonNoiseless, etc... meaning that there may be different subsets of the same font inside the PDF.

For an explanation have a look at section 9.6.4 *Font Subsets* of the PDF specification ISO 32000-1:2008<sup>903</sup>:

For a font subset, the PostScript name of the font — the value of the font’s **BaseFont** entry and the font descriptor’s **FontName** entry — shall begin with a tag followed by a plus sign (+). The tag shall consist of exactly six uppercase letters; the choice of letters is arbitrary, but different subsets in the same PDF file shall have different tags.

EXAMPLE EOODIA+Poetica is the name of a subset of Poetica®, a Type 1 font.

In other words: these characters aren’t merely “junk”. If you want to remove them, that’s a no-brainer, just use the appropriate string manipulation method, but be aware that removing them throws away information that may be useful in some contexts.

## How to extract text and anchor information from a PDF?



I am looking for a method to extract the text as well as anchor information using iText.

For example: the PDF content is “You can visit our website, XYZ, and do something” where XYZ is a clickable link. The output when extracting this content should be: “You can visit our website, XYZ ([www.google.com](http://www.google.com)) and do something”.

Basically I am trying to generate a text file with target links information.

Posted on StackOverflow on Jul 10, 2014 <sup>904</sup> by user985395<sup>905</sup>

The static text you can see in an PDF file is stored in content streams using PDF syntax as described in Adobe’s Imaging Model.

The interactive features you can see in a PDF file are stored outside the content stream of a page in so called Annotation dictionary using the Carousel Object System (COS).

You are probably making the assumption that when you see a clickable word XYZ, there is something like `<a href="http://xyz.com/">XYZ</a>` inside the PDF.

<sup>903</sup>[http://www.adobe.com/content/dam/Adobe/en/devnet/acrobat/pdfs/PDF32000\\_2008.pdf](http://www.adobe.com/content/dam/Adobe/en/devnet/acrobat/pdfs/PDF32000_2008.pdf)

<sup>904</sup><http://stackoverflow.com/questions/24668550/extract-text-and-anchor-information-from-pdf-using-itext>

<sup>905</sup><http://stackoverflow.com/users/985395/user985395>

There isn't.

There will be something like:

```
/F1 12 Tf  
(XYZ )Tj
```

somewhere in the content stream that contains the /Contents of a page.

When you inspect the /Annots of a page, you will find something like:

```
<<  
/A<<  
/S/URI  
/URI(http://xyz.com)  
>>  
/Subtype/Link  
/C[0 0 1]  
/Border[0 0 0]  
/Rect[36 803.52 98.03 814.62]  
>>
```

as an object in your PDF file.

If you want to extract all the links and the corresponding text from a document, you need to loop over all the page dictionaries, get the /Annots, check which annotations are of subtype /Link, get the action (/A), and the coordinates (/Rect).

To know which text corresponds with the text, you need to use iText text parser classes with a “region text” strategy and extract the text at the positions defined by the /Rect entry.

## How to read text from a specific position?



I have a problem using iTextSharp when reading data from PDF File. What I want to achieve is to read only specific part of PDF page (I want to only retrieve Address Information, which is located at constant position). I have seen using iTextSharp for reading all pages such as following:

```
StringBuilder text = new StringBuilder();
if (File.Exists(fileName)) {
    PdfReader pdfReader = new PdfReader(fileName);
    for (int page = 1; page <= pdfReader.NumberOfPages; page++) {
        ITextExtractionStrategy strategy = new SimpleTextExtractionStrategy();
        string currentText = PdfTextExtractor.GetTextFromPage(pdfReader, page, \
strategy);
        currentText = Encoding.UTF8.GetString(ASCIIEncoding.Convert(Encoding.De\
fault, Encoding.UTF8, Encoding.Default.GetBytes(currentText)));
        text.Append(currentText);
    }
    pdfReader.Close();
}
return text.ToString();
```

But how can I only restrict it to a specific location?

Posted on StackOverflow on Jun 12, 2014 <sup>[906](#)</sup> by Robert J. <sup>[907](#)</sup>

You are using a `SimpleTextExtractionStrategy` instead of a `LocationTextExtractionStrategy`. If `rect` is a rectangle based on the coordinates of your address, you need:

```
RenderFilter[] filter = {new RegionTextRenderFilter(rect)};
ITextExtractionStrategy strategy;
StringBuilder sb = new StringBuilder();
for (int i = 1; i <= reader.NumberOfPages; i++) {
    strategy = new FilteredTextRenderListener(new LocationTextExtractionStrategy\
(), filter);
    sb.AppendLine(PdfTextExtractor.GetTextFromPage(reader, i, strategy));
}
```

Now you'll get all the text snippets that intersect with the `rect` (so part of the text may be outside `rect`, iText doesn't cut text snippets in pieces).

---

<sup>906</sup><http://stackoverflow.com/questions/24185066/itextsharp-read-from-specific-position>

<sup>907</sup><http://stackoverflow.com/users/1539189/robert-j>

Note that you can get the MediaBox of a page using:

```
Rectangle mediabox = reader.GetPageSize(pagenum);
```

The coordinate of the lower-left corner is `x = mediabox.Left` and `y = mediabox.Bottom`; the coordinate of the upper-right corner is `x = mediabox.Right` and `y = mediabox.Top`.

The values of `x` increase from left to right; the values of `y` increase from bottom to top. The unit of the measurement system in PDF is called “user unit”. By default one user unit coincides with one point (this can change, but you won’t find many PDFs with a different `UserUnit` value). In normal circumstances, 72 user units = 1 inch.

## How to use a text extraction strategy after applying a location extraction strategy?



I used the following code to get data in PDF from a particular location.

```
Rectangle rect = new Rectangle(0,0,250,250);
RenderFilter filter = new RegionTextRenderFilter(rect);
fontBasedTextExtractionStrategy strategy = new fontBasedTextExtractionStrategy();
strategy = new FilteredTextRenderListener(new LocationTextExtractionStrategy(), \
filter); //Throws Error.
```

I want to get the bold text present in that location. Would creating a new method or class called `FontBasedTextExtractionStrategy` instead of a simple `TextExtractionStrategy` help?

Posted on StackOverflow on Jul 1, 2014<sup>908</sup> by Raka<sup>909</sup>

Please take a look at the [ParseCustom<sup>910</sup>](#) example. In this example, we create a custom `RenderFilter` (not a `TextExtractionStrategy`):

---

<sup>908</sup><http://stackoverflow.com/questions/24506830/can-we-use-text-extraction-strategy-after-applying-location-extraction-strategy>

<sup>909</sup><http://stackoverflow.com/users/3759100/raka>

<sup>910</sup><http://itextpdf.com/sandbox/parse/ParseCustom>

```
class FontRenderFilter extends RenderFilter {  
    public boolean allowText(TextRenderInfo renderInfo) {  
        String font = renderInfo.getFont().getPostscriptFontName();  
        return font.endsWith("Bold") || font.endsWith("Oblique");  
    }  
}
```

This text will filter all text so that only text of which the Postscript font name ends with Bold or Oblique.

This is how you use this filter:

```
public void parse(String filename) throws IOException {  
    PdfReader reader = new PdfReader(filename);  
    Rectangle rect = new Rectangle(36, 750, 559, 806);  
    RenderFilter regionFilter = new RegionTextRenderFilter(rect);  
    FontRenderFilter fontFilter = new FontRenderFilter();  
    TextExtractionStrategy strategy = new FilteredTextRenderListener(  
        new LocationTextExtractionStrategy(), regionFilter, fontFilter);  
    System.out.println(PdfTextExtractor.getTextFromPage(reader, 1, strategy));  
    reader.close();  
}
```

As you can see, we create a `FilteredTextRenderListener` that takes two filters, a `RegionTextRenderFilter` and our self-made filter based on the font.

## Why can't I extract text added using a Type3 font correctly from a PDF?



I have PDF file in Arabic that has text with font Type3 when I extract text using PDFBox some characters are empty and their font equals null? I want to know what is the problem.

```
protected void processTextPosition(TextPosition text) {
    String character=text.getCharacter(); // is empty
    String font=text.getFont().getBaseFont(); // equal null

}
```

The stream produced with iText looks like this: ( dJ v{d WcG)Tj

Why do I get the characters in this format?

Question marks appear in my stream as “SOH-STX-ETX-EOT”, not as one character. The character inside the PDF is shown as ‘d’ and ‘J’!

Posted on StackOverflow on Feb 9, 2014 <sup>911</sup> by Ayman Younis<sup>912</sup>

A Type 3 font is a user-defined font. For instance: a user can define that the character ‘P’ corresponds with the symbol for “The Artist Formerly Known As Prince” which is a glyph, but not a letter from any known alphabet:



The TAFKAP symbol

A glyph in a Type 3 font is a series of lines and shapes, and there's no way for a program such as iText or PDFBox to determine which character was meant. It is only normal that you get a question mark.

One of the following reasons applies for a PDF that contains Type 3 fonts:

1. The font was used to introduce symbols that don't exist in any font.
2. The font was used to obfuscate the content of the PDF so that its content can't be extracted.

---

<sup>911</sup><http://stackoverflow.com/questions/21659463/text-extraction-is-empty-and-unknown-for-text-has-type3-font-using-pdfbox-itext>

<sup>912</sup><http://stackoverflow.com/users/3215541/ayman-younis>

### 3. The PDF wasn't created in an elegant way.

If the Type 3 font was used for normal characters, you'll need to use OCR to convert the content to normal text.

## How to get the co-ordinates of an image?



In my project, I want to find co-ordinates of the images in my PDFs. I tried searching iText, but I was not successful. Using these co-ordinates and extracted image, I want to verify whether the extracted image is the same as an image present in database, and the co-ordinates of image are same as present in database.

Posted on StackOverflow on Jun 5, 2014 <sup>913</sup> by novice\_3<sup>914</sup>

When you say that you've tried with iText, I assume that you've used the [ExtractImages<sup>915</sup>](#) example as the starting point for your code. This example uses the helper class [MyImageRenderListener<sup>916</sup>](#), which implements the RenderListener interface.

In that helper class the renderImage() method is implemented like this:

```
public void renderImage(ImageRenderInfo renderInfo) {
    try {
        String filename;
        FileOutputStream os;
        PdfImageObject image = renderInfo.getImage();
        if (image == null) return;
        filename = String.format(path, renderInfo.getRef().getNumber(), image.get
tFileType());
        os = new FileOutputStream(filename);
        os.write(image.getImageAsBytes());
        os.flush();
        os.close();
    } catch (IOException e) {
        System.out.println(e.getMessage());
    }
}
```

<sup>913</sup><http://stackoverflow.com/questions/24055187/get-co-ordinates-of-image-in-pdf>

<sup>914</sup><http://stackoverflow.com/users/3682400/novice-3>

<sup>915</sup><http://itextpdf.com/examples/iiapdf?id=284>

<sup>916</sup><http://itextpdf.com/examples/iiapdf?id=283>

It uses the `ImageRenderInfo` object to obtain a `PdfImageObject` instance and it creates an image file using that object.

If you inspect the `ImageRenderInfo`<sup>917</sup> class, you'll discover that you can also ask for other info about the image. What you need, is the `getImageCTM()`<sup>918</sup> method. This method returns a `Matrix`<sup>919</sup> object. This matrix can be interpreted using ordinary high-school algebra. The values `I31` and `I32` give you the X and Y position. In most cases `I11` and `I22` will give you the width and the height (unless the image is rotated).

If the image is rotated, you'll have to consult your high-school schoolbooks, more specifically the ones discussing analytic geometry.

## How to check if a font is bold?



I have an application, that extracts headings out of pdf files. The documents that the application is supposed to work with, all have more or less coherent structure and formatting. In fact, telling if a text chunk is bold or not, is very important. Recently I came across a bunch of files, where some chunks visually appear bold, but do not have "bold" piece in string representation of font. I know that there is one more way of making text appear bold, for instance by changing the render mode. However in my case calling `GetTextRenderMode()` does not help either, as it returns 0 as if it were normal text. Are there any other ways of making text appear bold, and is it possible to detect it using iTextSharp?

Posted on StackOverflow on Jan 21, 2015 <sup>920</sup> by user2082616<sup>921</sup>

You are making the assumption that the font inside your PDF file knows if it's bold or not. Let's take a look inside and check if your assumption is correct.

This is what the subset JOJJAH of the font TT116t00 looks like when you look at the internals of the PDF file you have shared:

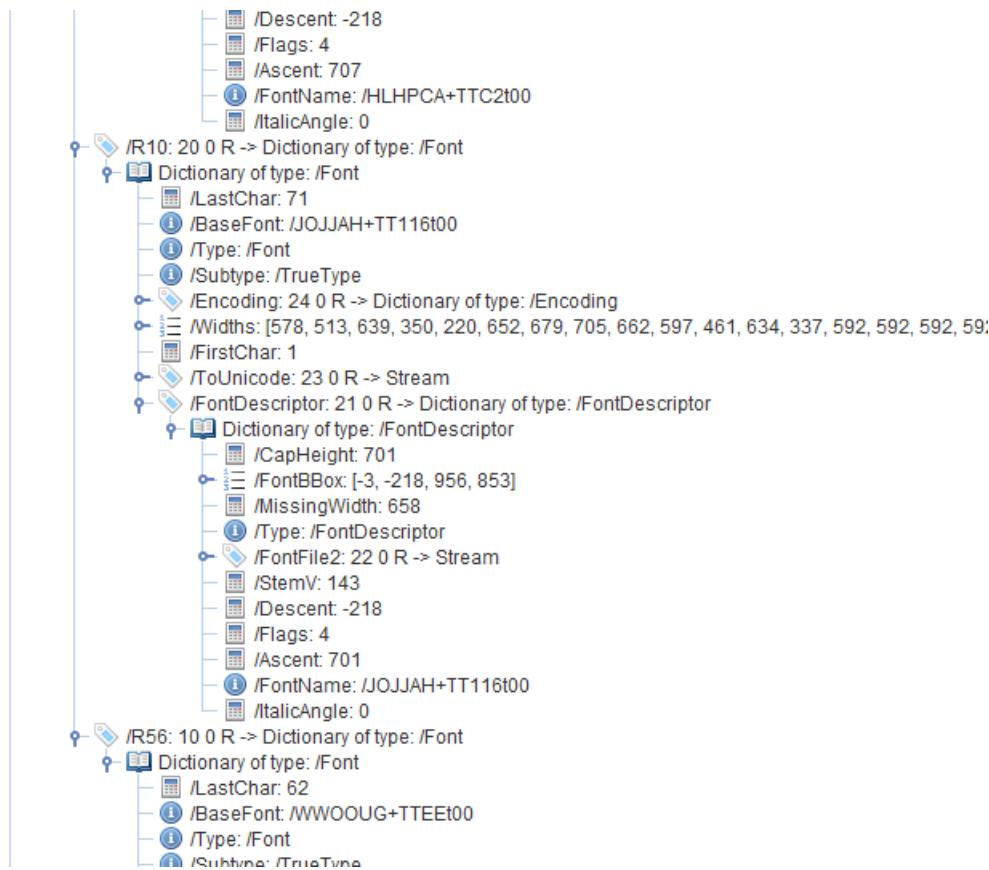
<sup>917</sup><http://api.itextpdf.com/itext/com/itextpdf/text/pdf/parser/ImageRenderInfo.html>

<sup>918</sup><http://api.itextpdf.com/itext/com/itextpdf/text/pdf/parser/ImageRenderInfo.html#getImageCTM%28%29>

<sup>919</sup><http://api.itextpdf.com/itext/com/itextpdf/text/pdf/parser/Matrix.html>

<sup>920</sup><http://stackoverflow.com/questions/28065269/what-are-the-ways-of-checking-if-piece-of-text-in-pdf-document-is-bold-using-it>

<sup>921</sup><http://stackoverflow.com/users/2082616/user2082616>



Internal view showing how a font is stored inside a PDF

We see that the font is of subtype `/TrueType`, we see that the `/ItalicAngle` is 0, and... we see that the 3rd bit of the `/Flags` is set. Let's check the PDF reference to find out what this tells us:

### 5.7.1 Font Descriptor Flags

The value of the **Flags** entry in a font descriptor is an unsigned 32-bit integer containing flags specifying various characteristics of the font. Bit positions within the flag word are numbered from 1 (low-order) to 32 (high-order). Table 5.20 shows the meanings of the flags; all undefined flag bits are reserved and must be set to 0. Figure 5.13 shows examples of fonts with these characteristics.

TABLE 5.20 Font flags

BIT POSITION	NAME	MEANING
1	FixedPitch	All glyphs have the same width (as opposed to proportional or variable-pitch fonts, which have different widths).
2	Serif	Glyphs have serifs, which are short strokes drawn at an angle on the top and bottom of glyph stems. ( <i>Sans serif</i> fonts do not have serifs.)
3	Symbolic	Font contains glyphs outside the Adobe standard Latin character set. This flag and the Nonsymbolic flag cannot both be set or both be clear (see below).
4	Script	Glyphs resemble cursive handwriting.
6	Nonsymbolic	Font uses the Adobe standard Latin character set or a subset of it (see below).
7	Italic	Glyphs have dominant vertical strokes that are slanted.

---

SECTION 5.7

459

*Font Descriptors*

BIT POSITION	NAME	MEANING
17	AllCap	Font contains no lowercase letters; typically used for display purposes, such as for titles or headlines.
18	SmallCap	Font contains both uppercase and lowercase letters. The uppercase letters are similar to those in the regular version of the same typeface family. The glyphs for the lowercase letters have the same shapes as the corresponding uppercase letters, but they are sized and their proportions adjusted so that they have the same size and stroke weight as lowercase glyphs in the same typeface family.
19	ForceBold	See below.

PDF Reference 1.7 section 5.7.1

I quote:

The font contains glyphs outside the Adobe standard Latin character set.

The glyphs look bold, because the glyphs are drawn in a way that they appear bold. You see the font

as bold because you are human. However, when a machine looks at the font, it doesn't have a clue that the font is bold. A machine just follows the instructions stored in the /FontFile2 stream.

In short: iTextSharp doesn't have any indications that the font is bold.

## Why is the text I extract from an English PDF page garbled?



I'm trying to extract and print English text out of a PDF on the console. Extraction is done through iText's PdfTextExtractor class. The text I'm getting is not understandable. The following code snippet represents my string extractor:

```
Document document = new Document();
PdfWriter writer = PdfWriter.getInstance(document,
    new FileOutputStream(OUTPUTFILE));
document.open();
PdfReader reader = new PdfReader(input);
int n = reader.getNumberOfPages();
PdfImportedPage page;
// Go through all pages
for (int i = 1; i <= n; i++) {
    String str=PdfTextExtractor.getTextFromPage(reader, i);
    System.out.println(str);
}
document.close();
```

The output I'm getting on console is not understandable even though the text in the PDF is in English:

t cotenn dna o mntoafinir yales r ni et h layhcsip Amgteu end y Retila m eysts  
e erefcern emsyst o f et h se. ru I n tioi, dnda etseh orpvedi eddda e ulav o  
se vdcie ollaw na s tiouquibu cacess o t latoutenxc e rpap dna t ilagid otten  
tofoi. nmirna ni soitaoli n mor f chea e. roth s iTh s i a cel ra csea  
ewerh " eth lweoh is ermo nath eth ms u fo sti rtasp ".

Can anybody please help me out what could be the possible solution for bringing text in English language as it is like in source PDF.

Posted on StackOverflow on May 16, 2014 <sup>922</sup> by codechefvaibhavkashyap<sup>923</sup>

<sup>922</sup><http://stackoverflow.com/questions/23693706/english-text-extracted-using-itextpdf-is-not-understandable>

<sup>923</sup><http://stackoverflow.com/users/1247963/codechefvaibhavkashyap>

If you want the text to be ordered based on its position on the page, you need to introduce a specific strategy, such as the `LocationTextExtractionStrategy`:

```
for (int i = 1; i <= reader.getNumberOfPages(); i++) {  
    String str = PdfTextExtractor.getTextFromPage(reader,  
        i, new LocationTextExtractionStrategy());  
}
```

The `LocationTextExtractionStrategy` sometimes results in odd sentences, more specifically if the letters ‘dance’ on the page (the baseline of the glyphs differs for text on the same line). In that case, you can try the `SimpleTextExtractionStrategy` which will return the text in the order in which it appears in the PDF syntax content stream.

# General questions about iText

These are some questions about iText in general. They aren't always about a technical problem, but they can be about a basic concept that is explained in more detail in one of the later chapters.

## Unit Testing and Automated Testing Questions



I have been searching for some unit tests for the program iText with no luck. Is anyone aware of any such tests? Also, does anyone know if the developers use any automatic testing tools on iText, such as Jenkins?

Posted on StackOverflow on Feb 21, 2014 <sup>924</sup> by user3338813<sup>925</sup>

Internally, we use Jenkins as well as TeamCity.

We have two types of tests:

1. The tests that are added when new core functionality is added. You can find these where Maven expects them: each Maven project has a `src` directory with 2 sub directories: `main` and `test`. For instance: if you look at iText core, you'll find the released stuff [here](#)<sup>926</sup> and the tests [here](#)<sup>927</sup>. Most of these tests are built on top of our `testutils`<sup>928</sup>.
2. The tests that are added when we get questions on SO or when we create code samples for the books. For these we use a generic test classes such as [GenericTest](#)<sup>929</sup> and [SandboxSampleWrapper](#)<sup>930</sup>. The wrapper class makes creating a test a no-brainer. All you need to do to turn a sample into a test is adding the `@WrapToTest` annotation. Well, actually there's more involved: you need to follow a specific pattern when writing a sample: always use `SRC` and `DEST` for source PDFs and resulting PDFs, always use a `createPdf()` or `manipulatePdf()` method, and always give the `cmp` file the same name as the `DEST` file prefixed with `cmp_`.

In both cases, you'll find PDF files of which the name starts with `cmp_`, see for instance the [cmpfiles folder](#)<sup>931</sup> for the examples. In both cases, you'll find references to Ghostscript and a compare tool (you'll need to configure these).

<sup>924</sup><http://stackoverflow.com/questions/21944424/itext-unit-testing-and-automated-testing-questions>

<sup>925</sup><http://stackoverflow.com/users/3338813/user3338813>

<sup>926</sup><http://sourceforge.net/p/itext/code/HEAD/tree/trunk/itext/src/main/>

<sup>927</sup><http://sourceforge.net/p/itext/code/HEAD/tree/trunk/itext/src/test/>

<sup>928</sup><http://sourceforge.net/p/itext/code/HEAD/tree/trunk/itext/src/main/java/com/itextpdf/testutils/>

<sup>929</sup><http://sourceforge.net/p/itext/code/HEAD/tree/trunk/sandbox/src/test/java/sandbox/GenericTest.java>

<sup>930</sup><http://sourceforge.net/p/itext/code/HEAD/tree/trunk/sandbox/src/test/java/sandbox/SandboxSampleWrapper.java>

<sup>931</sup><http://sourceforge.net/p/itext/code/HEAD/tree/trunk/sandbox/cmpfiles/>

## How to set initial view properties?



I want to set the properties available under the **Initial View** tab in Adobe Acrobat for an existing PDF programmatically.

### Document Options:

- Show = Bookmarks Panel and Page
- Page Layout = Continuous
- Magnification = Fit Width
- Open to Page number = 1

### Window Options:

- Show = Document Title

I tried to achieve this using the following code:

```
PdfStamper stamper =
    new PdfStamper(reader, new FileStream(dPDFFile, FileMode.Create));
stamper.AddViewerPreference(PdfName.DISPLAYDOCTITLE, new PdfBoolean(true));
```

the above code is used to set the document title show, but the following code is **not working**:

```
// For page layout
stamper.AddViewerPreference(PdfName.PAGELAYOUT, new PdfName("OneColumn"));
// For Bookmarks Panel and Page:
stamper.AddViewerPreference(PdfName.PageMode, new PdfName("UseOutlines"));
```

Finally I also want to set the language to English. The PS Script to do this looks like this [ {Catalog} <</Lang(EN)>> /PUT pdfmark. How is it done in PDF?

Posted on StackOverflow on Jun 23, 2014 <sup>[932](#)</sup> by Thirusanguraja Venkatesan<sup>[933](#)</sup>

When you have a `PdfWriter` instance named `writer`, you can set the Viewer preferences like this:

---

<sup>932</sup><http://stackoverflow.com/questions/24370273/set-initial-view-pdf-document-properties-using-itextsharp-with-c-sharp>

<sup>933</sup><http://stackoverflow.com/users/3768102/thirusanguraja-venkatesan>

```
writer.ViewerPreferences = viewerpreference;
```

In this case, the `viewerpreference` is a value that can have one of the following values:

- `PdfWriter.PageLayoutSinglePage`
- `PdfWriter.PageLayoutOneColumn`
- `PdfWriter.PageLayoutTwoColumnLeft`
- `PdfWriter.PageLayoutTwoColumnRight`
- `PdfWriter.PageLayoutTwoPageLeft`
- `PdfWriter.PageLayoutTwoPageRight`

See the [PageLayoutExample<sup>934</sup>](#) for more info.

You can also change the page mode as is shown in the [ViewerPreferencesExample<sup>935</sup>](#). In which case the different values are “OR”-ed:

- `PdfWriter.PageModeFullScreen`
- `PdfWriter.PageModeUseThumbs`
- `PdfWriter.PageLayoutTwoColumnRight | PdfWriter.PageModeUseThumbs`
- `PdfWriter.PageModeFullScreen | PdfWriter.NonFullScreenPageModeUseOutlines`
- `PdfWriter.FitWindow | PdfWriter.HideToolbar`
- `PdfWriter.HideWindowUI`

Currently, you’ve only used the `PrintPreferences` example from the official documentation:

```
writer.AddViewerPreference(PdfName.PRINTSCALING, PdfName.NONE);
writer.AddViewerPreference(PdfName.NUMCOPIES, new PdfNumber(3));
writer.AddViewerPreference(PdfName.PICKTRAYBYPDFSIZE, PdfBoolean.PDFTRUE);
```

But in some cases, it’s just easier to use:

```
writer.ViewerPreferences = viewerpreference;
```

As for setting the language, this is done like this:

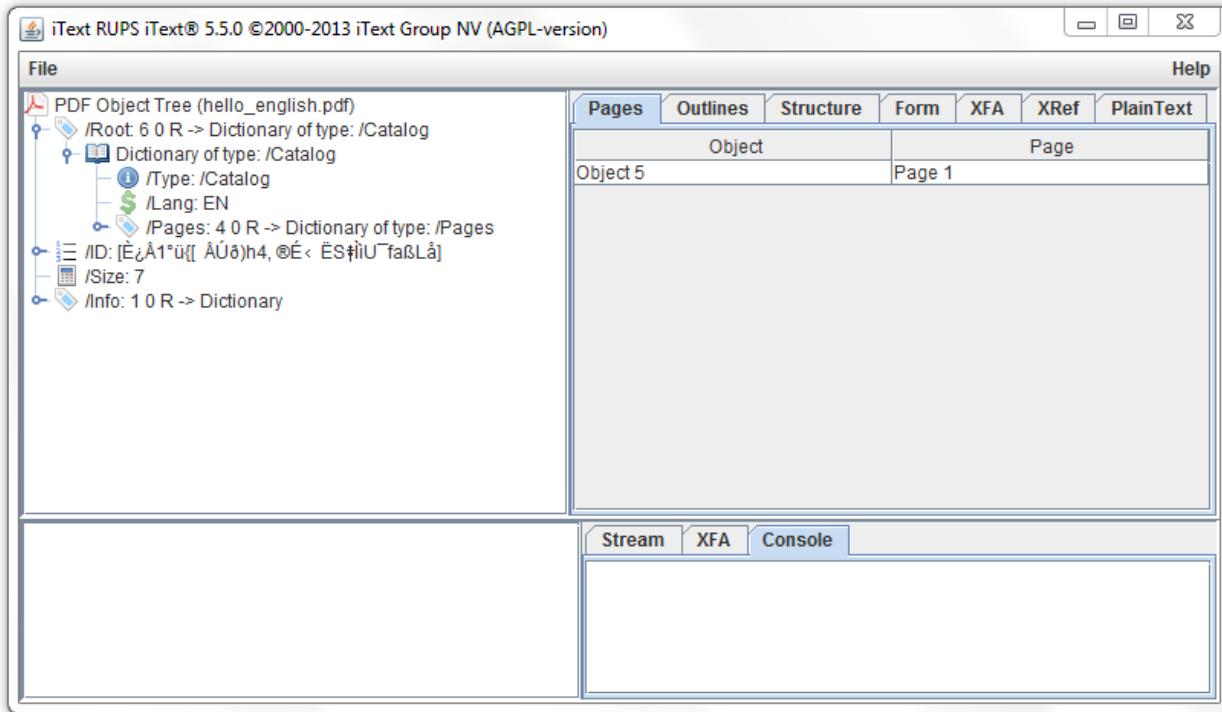
---

<sup>934</sup><http://itextpdf.com/examples/ii.php?id=228>

<sup>935</sup><http://itextpdf.com/examples/ii.php?id=229>

```
stamper.getWriter().extraCatalog.put(PdfName.LANG, new PdfString("EN"));
```

The result is shown in the following screen shot:



RUPS: looking at the internal structure of a PDF

As you can see, there is now a Lang entry with value EN added to the catalog.

**Taken from the additional answer by Chris Haas:**

The items *Magnification = Fit Width* and *Open to Page number = 1* are also part of the */Catalog* but in a special key called */OpenAction*. You can set this using `Writer.setOpenAction()`.

In your case you're looking for:

```
//Create a destination that fit's width (fit horizontal)
var D = new PdfDestination(PdfDestination.FITH);
//Create an open action that points to a specific page using this destination
var OA = PdfAction.GotoLocalPage(1, D, stamper.getWriter());
//Set the open action on the writer
stamper.getWriter().setOpenAction(OA);
```

## Why do I get a “Could not find PdfGraphics2D” error?



I have come across a runtime exception “Could not find class com.itextpdf.awt.PdfGraphics2D”. I wanted to create a PDF document from android device. For that I used iText library. This my code for creating PDF:

```
Document document = new Document();
PdfWriter.getInstance(document, outStream);
document.open();
document.add(new Paragraph(data));
document.close();
```

The code works fine. It is creating PDF successfully. but it gives me a runtime exception:

```
06-14 10:09:20.491: W/dalvikvm(764):
    Unable to resolve superclass of Lcom/itextpdf/awt/PdfGraphics2D; (1251)
06-14 10:09:20.491: W/dalvikvm(764):
    Link of class 'Lcom/itextpdf/awt/PdfGraphics2D;' failed
06-14 10:09:20.491: E/dalvikvm(764):
    Could not find class 'com.itextpdf.awt.PdfGraphics2D',
        referenced from method com.itextpdf.text.pdf.PdfContentByte.createGraphics
06-14 10:09:20.491: W/dalvikvm(764):
    VFY: unable to resolve new-instance 480
        (Lcom/itextpdf/awt/PdfGraphics2D;) in Lcom/itextpdf/text/pdf/PdfContentByte;
06-14 10:09:25.280: E/dalvikvm(764):
    Could not find class 'org.bouncycastle.cert.X509CertificateHolder',
        referenced from method com.itextpdf.text.pdf.PdfReader.readDecryptedDocObj
06-14 10:09:25.280:
    W/dalvikvm(764): VFY: unable to resolve new-instance 1612
        (Lorg/bouncycastle/cert/X509CertificateHolder;) in Lcom/itextpdf/text/pdf/Pd\
fReader;
```

I have done clean and build, added jar to libs folder and make it selected on order and export and i done lot of research for past 2 days. but nothing helped me. Based upon my knowledge there should be these possibilities: (1) the external jar isn't loaded properly, or (2) the class PdfGraphics2D extends java.awt.Graphics2D which is not available on Android.

Posted on StackOverflow on Jun 14, 2013 <sup>[936](#)</sup> by R9J<sup>[937](#)</sup>

---

<sup>936</sup><http://stackoverflow.com/questions/17102533/could-not-find-class-com-itextpdf-awt-pdfgraphics2d>

<sup>937</sup><http://stackoverflow.com/users/1912085/r9j>

You've discovered that `PdfGraphics2D` extends `java.awt.Graphics2D`, and as you already know `Graphics2D` is a forbidden class on Android.

You've also encountered problems related to BouncyCastle.

This tells me that you're using the Java version of iText instead of the [Android port](#)<sup>938</sup>. In the Android port, we replaced BouncyCastle by SpongyCastle (as recommended when using encryption on Android) and we removed all references to forbidden classes (for instance in the `awt` and `nio` packages).

Please switch to using the Android port of iText. It is called [iTextG](#)<sup>939</sup>.

## Why can't I compile the iText source code?



I get an error after running `mvn install -P all -e` to compile and install iText. I've also done `mvn clean` on before hand. The error says:

```
testFlatteningGenerateAppearances2(com.itextpdf.text.pdf.FlatteningTest):  
Path to GhostScript is not specified. Please use -DgsExec
```

Do I need to install Ghostscript?

Posted on StackOverflow on Sep 17, 2014 <sup>940</sup> by user3769040<sup>941</sup>

Add the parameter `maven.test.skip=true` or `skipTests=true` in the command line. The tests create plenty of PDF files (new documents) that need to be compared with reference PDF files (existing documents stored with the tests). Ghostscript is used to create images of those PDF files. These images are compared on a pixel basis to check if the new files have the same appearance of the reference files.

If you want to build iText *with* the tests, you need Ghostscript. If you want to build iText merely to get a jar, you can skip the tests and you don't need Ghostscript.

---

<sup>938</sup><http://itextpdf.com/product/itextg>

<sup>939</sup><http://itextpdf.com/product/itextg>

<sup>940</sup><http://stackoverflow.com/questions/25881776/cant-compile-itext-source-code-unmodified>

<sup>941</sup><http://stackoverflow.com/users/3769040/user3769040>

## Why do I get a BouncyCastle NoClassDefFoundError?



I want to encrypt my PDF but there seems to be an error:

```
PdfReader reader = new PdfReader("my.pdf");
PdfStamper stamper = new PdfStamper(reader, new FileOutputStream("new.pdf"));
stamper.setEncryption("reader_password".getBytes(), "permission_password".getByt
es(),
    PdfWriter.ALLOW_COPY | PdfWriter.ALLOW_PRINTING, PdfWriter.STANDARD_ENCRYPTI
ON_128);
stamper.close();
```

Here's the error I get:

```
Exception in thread "main" java.lang.NoClassDefFoundError: org/bouncy-
castle/asn1/ASN1Encodable
```

Posted on StackOverflow on May 19, 2014<sup>942</sup> by wizclark99<sup>943</sup>

When you look at the POM file for iText, you see the following dependencies:

```
<dependency>
    <groupId>org.bouncycastle</groupId>
    <artifactId>bcpkix-jdk15on</artifactId>
    <version>1.49</version>
    <type>jar</type>
    <scope>compile</scope>
    <optional>true</optional>
</dependency>
<dependency>
    <groupId>org.bouncycastle</groupId>
    <artifactId>bcpkix-jdk15on</artifactId>
    <version>1.49</version>
    <type>jar</type>
    <scope>compile</scope>
    <optional>true</optional>
</dependency>
```

---

<sup>942</sup><http://stackoverflow.com/questions/23728847/pdf-encryption-error-in-java-itext>

<sup>943</sup><http://stackoverflow.com/users/3650927/wizclark99>

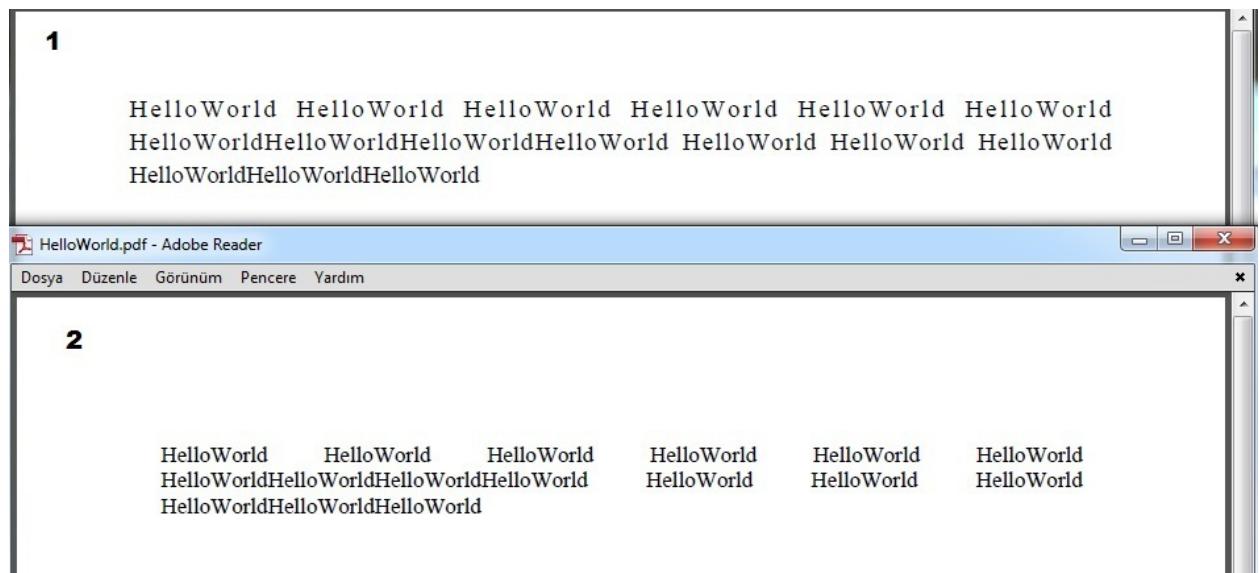
This means that you need the bcprov and the bcpkix jars version 1.49 from [Bouncycastle<sup>944</sup>](#).

If you are not using iText 5.5.x, please check the POM file as older version of iText might require older versions of BouncyCastle.

## How to change the spacing between words and characters?



I want to set the text alignment to justified, but I don't want iText to add any extra space between the characters (see figure 1). I prefer space between words as shown in figure 2.



Different spacing when justifying text

---

<sup>944</sup><http://bouncycastle.org/java.html>



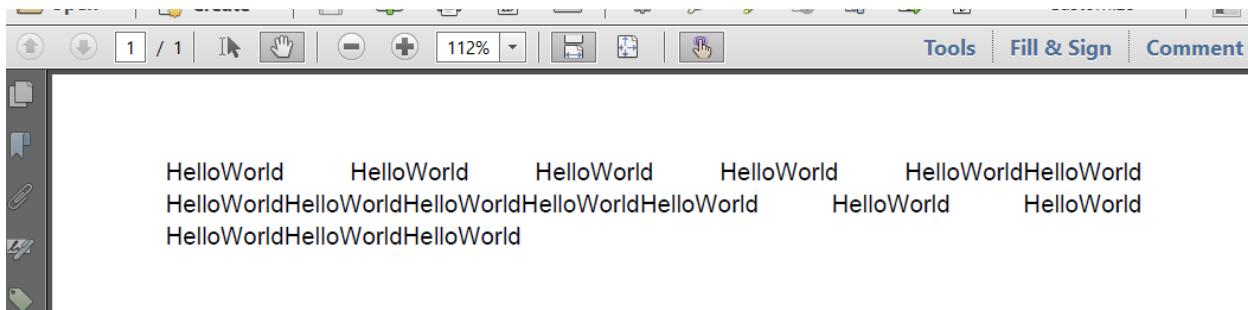
With this code, I get the result shown in figure 1.

```
public static void main(String[] args) throws DocumentException, IOException {
    Document document = new Document();
    PdfWriter.getInstance(document, new FileOutputStream("abc.pdf"));
    BaseFont bf1 = BaseFont.createFont(
        BaseFont.TIMES_ROMAN, "iso-8859-9", BaseFont.EMBEDDED);
    Font font1 = new Font(bf1);
    document.open();
    Paragraph paragraph2 = new Paragraph();
    paragraph2.setAlignment(Element.ALIGN_JUSTIFIED);
    paragraph2.setFont(font1);
    paragraph2.setIndentationLeft(20);
    paragraph2.setIndentationRight(20);
    paragraph2.add("HelloWorld HelloWorld HelloWorld HelloWorld HelloWorld" +
        "HelloWorld HelloWorldHelloWorldHelloWorldHelloWorld" +
        "HelloWorld HelloWorld HelloWorld HelloWorldHelloWorldHelloWorld");
    document.add(paragraph2);
    document.close();
}
```

How can I change this code to get a result like in figure 2?

Posted on StackOverflow on Jun 16, 2015 <sup>[945](#)</sup> by Osman Yilmaz<sup>[946](#)</sup>

Please take a look at the [SpaceCharRatioExample<sup>947</sup>](#) to find out how to create a PDF that looks like [space\\_char\\_ratio.pdf<sup>948</sup>](#):



Spacing between words

When you justify a paragraph, iText will add extra space between the words and between the characters. By default, iText will add 2.5 times more space between words than between characters.

<sup>945</sup><http://stackoverflow.com/questions/30869268/java-itext-alignment>

<sup>946</sup><http://stackoverflow.com/users/4999271/osman-yilmaz>

<sup>947</sup><http://itextpdf.com/sandbox/objects/SpaceCharRatioExample>

<sup>948</sup>[http://itextpdf.com/sites/default/files/space\\_char\\_ratio.pdf](http://itextpdf.com/sites/default/files/space_char_ratio.pdf)

You can change this default like using the `setSpaceCharRatio()` method:

```
public void createPdf(String dest) throws IOException, DocumentException {
    // step 1
    Document document = new Document();
    // step 2
    PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream(dest));
    // step 3
    document.open();
    // step 4
    writer.setSpaceCharRatio(PdfWriter.NO_SPACE_CHAR_RATIO);
    Paragraph paragraph = new Paragraph();
    paragraph.setAlignment(Element.ALIGN_JUSTIFIED);
    paragraph.setIndentationLeft(20);
    paragraph.setIndentationRight(20);
    paragraph.add("HelloWorld HelloWorld HelloWorld HelloWorld HelloWorld" +
        "HelloWorld HelloWorldHelloWorldHelloWorldHelloWorld" +
        "HelloWorld HelloWorld HelloWorld HelloWorldHelloWorldHelloWorld");
    document.add(paragraph);
    // step 5
    document.close();
}
```

In this case, we tell iText not to add any space between the characters, only between the words: `PdfWriter.NO_SPACE_CHAR_RATIO`. Well, as a matter of fact we tell iText to add 10,000,000 times more space between the words than between the characters, which is almost the same thing.

## How to add / delete / retrieve information from a PDF using a custom property?



I want to store 2 strings (`string1` and `string2`) as a custom entry (key and value) that is stored inside the PDF in the custom property area. You can find this custom property area under File > Properties > Custom > Custom properties in Adobe Reader, where you can find “Name” and “Value” in pairs. I want to store `string1` in the “Value” and store `string2` in the “Name”. Later, I want to retrieve/delete the strings in the custom property area.

Posted on StackOverflow on Nov 4, 2014 <sup>949</sup> by brian<sup>950</sup>

---

<sup>949</sup><http://stackoverflow.com/questions/26726485/add-delete-retrieve-information-from-a-pdf-using-a-custom-property>

<sup>950</sup><http://stackoverflow.com/users/4197487/brian>

In order to better understand the problem, I've used Acrobat to add a custom metadata entry named "Test" with value "test" and when you look inside that file, you can see that this key/value pair turns up on two places (marked with a red dot).

1. It is present in the Info dictionary, which is the *traditional* place to store metadata.
2. It is present in the XMP metadata stream as a tag named Test with prefix pdfx (for custom tags).

The screenshot shows the iText RUPS interface. At the top, it says "iText RUPS iText® 5.5.3 ©2000-201". Below that is a "File" menu. The main area has a "PDF Object Tree" tab selected, showing the structure of the PDF file "hello.pdf". The tree includes nodes for the Root object, Catalog, Metadata stream, Pages, and Info dictionary. The Info dictionary node is expanded, showing keys like /ID, /Info, /Producer, /ModDate, and /CreationDate. A red circle highlights the "test" value under the /Info key. Below the tree is a table showing the contents of the Info dictionary:

Key	Value	
/Type	/Metadata	<span style="color: red;">X</span>
/Subtype	/XML	<span style="color: red;">X</span>
/Length	3223	<span style="color: red;">X</span>
		<span style="color: green;">+</span>

Below the table is the "Stream" tab, which displays the XML code of the XMP metadata stream. A red circle highlights the "pdfx:Test" tag. The XML code includes various XMP namespaces and tags, such as xpacket, exxmpmeta, rdf, and dc.

```

<?xpacket begin="i>" id="W5M0MpCehiHzreSzNTczkc9d"?>
<exxmpmeta xmlns:x="adobe:ns:meta" x:xmptk="Adobe XMP Core 5.4-c005">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<rdf:Description rdf:about="">
  xmlns:xmp="http://ns.adobe.com/xap/1.0/"
  xmlns:pdf="http://ns.adobe.com/pdf/1.3/"
  xmlns:pdfx="http://ns.adobe.com/pdfx/1.3/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:xmpMM="http://ns.adobe.com/xap/1.0/mm/"
<xmp:CreateDate>2013-09-08T12:21:50+02:00</xmp:CreateDate>
<xmp:ModifyDate>2014-11-04T12:24:26+01:00</xmp:ModifyDate>
<xmp:MetadataDate>2014-11-04T12:24:26+01:00</xmp:MetadataDate>
<pdf:Producer>iText® 5.4.3 ©2000-2013 1T3XT BVBA (AGPL-version)</pdf:Producer>
<pdfx:Test>test</pdfx:Test>
<dc:format>application/pdf</dc:format>
<xmpMM:DocumentID>uuid:ba4481f1-db04-4121-ab84-e97cd66589</xmpMM:DocumentID>
<xmpMM:InstanceID>uuid:d2960451-941d-4d33-9da9-32828824980</xmpMM:InstanceID>
</rdf:Description>
</rdf:RDF>

```

Custom metadata viewed in iText RUPS

Adding an extra value to the Info dictionary is easy when using iText. Updating the XMP metadata is also possible, but you'll have to create the XMP stream yourself and maybe it's overkill in your case. Maybe your PDF only has an Info dictionary and no XMP.

Moreover: you say that the purpose of having that key is to retrieve its value and to delete the custom entry afterwards. In that case, it's sufficient to add the extra entry in the Info dictionary.

Depending on whether you want to add a custom entry to the Info dictionary to a PDF created from scratch or to an existing PDF you need one of the following examples.

In [CustomMetaEntry<sup>951</sup>](#), we add a standard metadata entry for the title and a custom entry, *Test*.

```
public void createPdf(String dest) throws IOException, DocumentException {
    Document document = new Document();
    PdfWriter.getInstance(document, new FileOutputStream(dest));
    document.addTitle("Some example");
    document.add(new Header("Test", "test"));
    document.open();
    Paragraph p = new Paragraph("Hello World");
    document.add(p);
    document.close();
}
```

As you can see, iText had `addX()` methods to add Title, Author,... metadata. However, if you want to add a custom entry, you need to use the `add()` method to add a `Header` instance. You need to add the metadata *before* opening the document.

If you want to add entries to the info dictionary of an existing PDF, take a look at the [MetadataPdf<sup>952</sup>](#) example:

```
public void manipulatePdf(String src, String dest) throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    Map<String, String> info = reader.getInfo();
    info.put("Title", "Hello World stamped");
    info.put("Subject", "Hello World with changed metadata");
    info.put("Keywords", "iText in Action, PdfStamper");
    info.put("Creator", "Silly standalone example");
    info.put("Author", "Also Bruno Lowagie");
    stamper.setMoreInfo(info);
    stamper.close();
    reader.close();
}
```

---

<sup>951</sup><http://itextpdf.com/sandbox/objects/CustomMetaEntry>

<sup>952</sup><http://itextpdf.com/examples/iaa.php?id=216>

In this example, we get the info dictionary from a `PdfReader` instance using the `getInfo()` method. This also answers how to retrieve the custom data from a PDF. If the `Map` contains an entry with key `Test`, you can get its value like this:

```
String test = info.get("Test");
```

You can now add extra pairs of `Strings` to this `Map`. In the example, we add standard keys for metadata, but you can also use custom keys.

Removing an entry from an existing PDF file is done in the same way as adding an entry. It's sufficient to add a `null` value. For instance:

```
info.put("Test", null);
```

This will remove a custom entry named `Test` in case such a value was present in your Info dictionary.

## How to create an uncompressed PDF file?



During development testing, I'd prefer to create uncompressed, non-binary PDF files with iTextSharp so that I can check their internals easily. I can always convert the finished PDF with various utilities but that takes an extra step that would be comfortable to avoid. There are some places (e.g. `PdfImage`) where I can decide about compression level but I couldn't find anything about the compression used to output the individual PDF objects into the stream. Do you think this is possible with iTextSharp?

Posted on StackOverflow on May 15, 2014<sup>953</sup> by Gábor<sup>954</sup>

You have two options:

### [1.] Disable compression for all your documents at once:

Please take a look at the `Document` class. You'll find:

```
public static bool Compress = true;
```

If you change this `static bool` to `false`, PDF syntax streams won't be compressed.

Of course: images will be de facto compressed. For instance: if you add a JPEG to your document, you are adding an image that is already compressed, and iText won't *uncompress* it.

### [2.] Disable compression for a single document at a time:

Please take a look at the `PdfWriter` class. You'll find:

<sup>953</sup><http://stackoverflow.com/questions/23689760/how-to-create-uncompressed-pdf-file>

<sup>954</sup><http://stackoverflow.com/users/2076286/gábor>

```
protected internal int compressionLevel = PdfStream.DEFAULT_COMPRESSION;
```

If you change the value of `compressionLevel` to `PdfStream.NO_COMPRESSION` before opening the 'document', your PDF syntax streams won't be compressed.

## How to detect if PDF has compressed xref table in Java?



Is there an API available in iText to detect whether the PDF file has a compressed xref table?

The `PdfReader` class of this library has some useful API with regards to xref, but none serve my purpose.

The requirement is to:

1. Check if the PDF has xref compressed table.
2. If 1 is true -> then "Uncompress" xref table.
3. Send the byte stream for further processing.
4. Once the processing completes "Compress" back the xref table to its original form

Any pointers in this regard would be appreciated.

Posted on StackOverflow on Jan 15, 2013 <sup>955</sup> by Bond - Java Bond<sup>956</sup>

As it turns out, this is already supported in iText. You need to create a `PdfReader` instance and then use `isNewXrefType()`.

To uncompress the XRef table of a PDF document, you can use this method:

```
public void uncompressXRef(String src, String dest)
    throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.close();
    reader.close();
}
```

To recompress the XRef table, use this method:

---

<sup>955</sup><http://stackoverflow.com/questions/14342041/how-to-detect-if-pdf-has-compressed-xref-table-in-java>

<sup>956</sup><http://stackoverflow.com/users/1910582/bond-java-bond>

```

public void recompressXRef(String src, String dest)
    throws IOException, DocumentException {
    PdfReader reader = new PdfReader(src);
    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));
    stamper.getWriter().setFullCompression();
    stamper.close();
    reader.close();
}

```

## How to increase the accuracy of measurements in iTextSharp?



I want to resize a pdf to a specific size, but when I use scaling it loses accuracy because a float rounds the value. How can I avoid this and make measurements more accurate?

Posted on StackOverflow on Jul 21, 2014<sup>957</sup> by Stephen De Clerk<sup>958</sup>

The ByteBuffer class has a public static variable named HIGH\_PRECISION. By default, it is set to false. You can set it to true so that you get 6 decimal places when rounding a number:

```
iTextSharp.text.pdf.ByteBuffer.HIGH_PRECISION = true;
```

That will cost you some performance (but maybe you'll hardly notice that) and the resulting file will have more bytes (but measurements will be more accurate).

## How to prevent the resizing of pages in PDF?



I create a PDF where I define absolute measurements. However, when I print this PDF, the measurements aren't correct. I want to have a PDF file that contains no differences between the actual size vs "fit to page" when printing.

Posted on StackOverflow on Jun 6, 2014<sup>959</sup> by Cristi B.<sup>960</sup>

Please add the following line to your code. This line will make sure that the page isn't scaled when printed:

---

<sup>957</sup><http://stackoverflow.com/questions/24861632/resize-pdf-to-specific-width-and-height-with-itexsharp>

<sup>958</sup><http://stackoverflow.com/users/2640851/stephen-de-klerk>

<sup>959</sup><http://stackoverflow.com/questions/24084271/itext-resize-table-at-actual-size-vs-fit-to-page>

<sup>960</sup><http://stackoverflow.com/users/2064134/cristi-b>

```
writer.addViewerPreference(PdfName.PRINTSCALING, PdfName.NONE);
```

This is the only parameter you can set to influence the print setting. You can't tell a viewer that it needs to fit the page to the size of the printer page. You can only instruct the viewer that the actual size needs to be taken, by setting the PRINTSCALING to NONE.

## How to hyphenate text?



I generate an PDF file with iText in Java. My table columns have fixed widths. Text that is too long for one line is wrapped in the cell. But hyphenation is not used. The word "Leistungsscheinziffer" is shown as:

```
Leistungssc //Break here  
heinziffer
```

My code where I use hyphenation:

```
final PdfPTable table = new PdfPTable(sumCols);
table.getDefaultCell().setBorder(Rectangle.NO_BORDER);
table.getDefaultCell().setPadding(4f);
table.setWidths(widthsCols);
table.setWidthPercentage(100);
table.setSpacingBefore(0);
table.setSpacingAfter(5);
final Phrase result = new Phrase(text, font);
result.setHyphenation(new HyphenationAuto("de", "DE", 2, 2));
final PdfPCell cell = new PdfPCell(table.getDefaultCell());
cell.setPhrase(result);
table.addCell(cell);
...
```

Hyphenation is activated and the following test results "Lei-stungs-schein-zif-fer"

```
Hyphenator h = new Hyphenator("de", "DE", 2, 2);
Hyphenation s = h.hyphenate("Leistungsscheinziffer");
System.out.println(s);
```

Posted on StackOverflow on Nov 21, 2013<sup>961</sup> by user3017202<sup>962</sup>

---

<sup>961</sup><http://stackoverflow.com/questions/20119709/itext-hyphen-in-table-cell>

<sup>962</sup><http://stackoverflow.com/users/3017202/user3017202>

Normally hyphenation is set on the Chunk level:

```
Chunk chunk = new Chunk("Leistungsscheinziffer");
chunk.setHyphenation(new HyphenationAuto("de", "DE", 2, 2));
table.addCell(new Phrase(chunk));
```

If you want to set the hyphenation on the Phrase level, you can do so, but this will only work for all subsequent Chunks that are added. It won't work for the content that is already stored inside the Phrase. In other words: you need to create an empty Phrase object and then add one or more Chunk objects:

```
Phrase phrase = new Phrase();
phrase.setHyphenation(new HyphenationAuto("de", "DE", 2, 2));
phrase.add(new Chunk("Leistungsscheinziffer"));
```

I've made an example based on your code ([HyphenationExample<sup>963</sup>](#)); the word "Leistungsscheinziffer" is hyphenated in the resulting PDF: [hyphenation\\_table.pdf<sup>964</sup>](#).

**Note:** for this example to work, you need to add the `iText-hyph-xm1.jar` to your CLASSPATH. This jar contains the hyphenation rules for a wide array of languages.

## How to get the current page count?



I have to create a PDF document from a grid view. When I try to count the total number of pages, I don't get the current amount.

Dim pdfDoc As New Document(PageSize.A2, 50, 50, 50, 50) ..... pdfDoc.PageCount

Posted on StackOverflow on [Jul 21, 2014](#) <sup>965</sup> by [Please help me](#)<sup>966</sup>

Your error is based on a misunderstanding of the basic concepts of iTextSharp.

A document is created in 5 steps:

1. Create a document. This document doesn't know anything about the **presentation** of the document, only about its **content**.
2. Create a writer. You are creating a `PdfWriter` that will translate the **content** into a **presentation**, more specifically into a PDF document with one or more pages.

---

<sup>963</sup><http://itextpdf.com/sandbox/tables/HyphenationExample>

<sup>964</sup>[http://itextpdf.com/sites/default/files/hyphenation\\_table.pdf](http://itextpdf.com/sites/default/files/hyphenation_table.pdf)

<sup>965</sup><http://stackoverflow.com/questions/25540686/total-page-count-for-itextsharp>

<sup>966</sup><http://stackoverflow.com/users/3962482/please-help-me>

3. Open the document.
4. Add content.
5. Close the document.

You are asking the document object for the current page number, yet the document isn't aware of its presentation. It doesn't even know that a PDF is produced.

You should ask the writer that is responsible for creating the PDF how many pages were already created; `writer.PageNumber` will return that number.

## When is the content flushed to a PDF File by iTextsharp?



I assumed that `Document.Add()` flushed content to the PDF file (the file stream) immediately, but it looks like that's not the case.

Posted on StackOverflow on Feb 18, 2014<sup>967</sup> by sameer<sup>968</sup>

PDF is a Page Description Language. Every page is an autonomous set of objects. The content is stored in one or more streams. There is no such thing as a paragraph or a table etc in a PDF. It's just a sequence of lines, shapes and glyphs drawn on a page.

When you add content to a document using the `Add()` method, this content is converted into PDF syntax that is appended to the content stream of a page. As soon as the page is full, this content stream and the corresponding page dictionary are written to the output stream and flushed.

Not sooner!

Several objects, such as fonts, the cross-reference table, Form XObjects,... are kept into memory, because they can change during the document creation process.

In some cases you can release these objects early. For instance: there a "release template" method to write Form XObject to the output stream immediately. Image XObjects are always written immediately.

<sup>967</sup><http://stackoverflow.com/questions/21854409/when-is-the-content-flushed-to-a-pdf-file-by-itextsharp>

<sup>968</sup><http://stackoverflow.com/users/1327093/sameer>

## How to convert PdfStamper to a byte array?



In my application, I need to read the existing PDF and add barcodes to this PDF and pass it to output stream. Here the existing pdf is like a template. I am using iText jar for adding barcode.

I want to know the possibilities of converting PdfStamper object to byte array or PdfContentByte to byte array. Can anyone help on this?

Posted on StackOverflow on Oct 27, 2014 <sup>969</sup> by user1636102<sup>970</sup>

I assume that you want to write to a ByteArrayOutputStream instead of to a FileOutputStream. There are different examples on how to do that on the iText web site.

See for instance the [FormServlet<sup>971</sup>](#) example where it says:

```
// We create an OutputStream for the new PDF
ByteArrayOutputStream baos = new ByteArrayOutputStream();
// Now we create the PDF
PdfStamper stamper = new PdfStamper(reader, baos);
```

Then later in the example, we do this:

```
// We write the PDF bytes to the OutputStream
OutputStream os = response.getOutputStream();
baos.writeTo(os);
```

If you want a byte[], you can simply do this:

```
byte[] pdfBytes = baos.toByteArray();
```

I hope your question wasn't about writing a PdfContentByte stream to a byte[] because that wouldn't make sense: a content stream doesn't contain any resources such as fonts, images, form XObjects, etc...

<sup>969</sup><http://stackoverflow.com/questions/26385446/how-to-convert-pdfstamper-to-byte-array>

<sup>970</sup><http://stackoverflow.com/users/1636102/user1636102>

<sup>971</sup><http://itextpdf.com/examples/iiapdf?id=170>

## How to create a new PDF file if a file name already exists?



I need to create simple receipts which contain user details such as a name, a fare and a number. Whenever a new user inputs data in the main form, the following code just overwrites the data of the first user in the PDF file:

```
Document document = new Document();
PdfWriter.getInstance(document,
    new FileOutputStream("C://sample.pdf"));
document.open();
// adding content as Paragraph objects
document.close();
```

How can I create a new PDF file without appending or overwriting the original data of the first user? For instance, how can I create sample.pdf, sample2.pdf, sample3.pdf, and so on?

Posted on StackOverflow on Mar 27, 2015 <sup>972</sup> by Gelly<sup>973</sup>

Almost all the methods you might need to achieve what you want can be found in the Java API documentation for the [File class](#)<sup>974</sup>

You want to create a **unique** file that starts with sample and ends with pdf. To achieve this, you can use the [createTempFile\(\) method](#)<sup>975</sup>. This question was already answered on StackOverflow 6 years ago: [What is the best way to generate a unique and short file name in Java?](#)<sup>976</sup>

Suppose that you really want to have **incremental numbers** in your file name, e.g. sample0001.pdf, sample0002.pdf, sample0003.pdf and so on, then you can use the [list\(\) method](#)<sup>977</sup>. This returns an array of String values with the names of all files in a directory. I suggest that you use a [FilenameFilter](#) so that you only get the PDF files starting with sample. You could then sort these names to find the name with the highest number. See [How to list latest files in a directory using FileNameFilter](#)<sup>978</sup> to find out how to create such a filter. Once you have the file name with the highest number, it's only a matter of String manipulation to create a new file name. Use that file name (or that File instance) when you define the OutputStream. As you can see, this answer doesn't mention iText anywhere and although the extension of the files we create or list is .pdf, it has nothing to do with PDF or PDF generation either. It's a pure Java question.

<sup>972</sup><http://stackoverflow.com/questions/29302545/create-a-new-pdf-file-if-name-exists-already-in-itext>

<sup>973</sup><http://stackoverflow.com/users/4511680/gelly>

<sup>974</sup><http://docs.oracle.com/javase/7/docs/api/java/io/File.html>

<sup>975</sup>[http://docs.oracle.com/javase/7/docs/api/java/io/File.html#createTempFile\(java.lang.String,%20java.lang.String,%20java.io.File\)](http://docs.oracle.com/javase/7/docs/api/java/io/File.html#createTempFile(java.lang.String,%20java.lang.String,%20java.io.File))

<sup>976</sup><http://stackoverflow.com/questions/825678/what-is-the-best-way-to-generate-a-unique-and-short-file-name-in-java>

<sup>977</sup>[http://docs.oracle.com/javase/7/docs/api/java/io/File.html#list\(java.io.FilenameFilter\)](http://docs.oracle.com/javase/7/docs/api/java/io/File.html#list(java.io.FilenameFilter))

<sup>978</sup><http://stackoverflow.com/questions/21236105/how-to-list-latest-files-in-a-directory-using-filenamefilter>

## How can I server a PDF to a browser without storing a file on the server side?



I have two methods. One that generates a PDF at the server side and another that serves the PDF to the client side"

```

public void generatePDF() throws Exception {
    Document doc = new Document();
    File file = new File("C://New folder//itext_Test.pdf");
    FileOutputStream pdfFileout = new FileOutputStream(file);
    PdfWriter.getInstance(doc, pdfFileout);
    doc.open();
    Paragraph para = new Paragraph("Test");
    doc.add(para);
    doc.close();
}

public void downloadPDF(HttpServletRequest request, HttpServletResponse response)
    throws IOException{
    response.setContentType("application/pdf");
    response.setHeader("Content-disposition", "attachment;filename=" + "testPDF.pdf");
    try {
        File f = new File("C://New folder//itext_Test.pdf");
        FileInputStream fis = new FileInputStream(f);
        DataOutputStream os = new DataOutputStream(response.getOutputStream());
        response.setHeader("Content-Length", String.valueOf(f.length()));
        byte[] buffer = new byte[1024];
        int len = 0;
        while ((len = fis.read(buffer)) >= 0) {
            os.write(buffer, 0, len);
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
}

```

How can I serve the PDF file to the client without storing the file on the server side and allow the client side to directly download the file that is generated?

Posted on StackOverflow on May 26, 2015 <sup>979</sup> by Mishal Harish<sup>980</sup>

---

<sup>979</sup><http://stackoverflow.com/questions/30456901/download-pdf-file-using-java-without-storing-file-in-server-side>

<sup>980</sup><http://stackoverflow.com/users/4315788/mishal-harish>

You can use any `OutputStream` when creating a PDF file, so in theory, you could use a `response.getOutputStream()`. See for instance the [Hello<sup>981</sup>](#) Servlet from Chapter 9 of “iText in Action - Second Edition”:

```
public class Hello extends HttpServlet {
    protected void doGet(HttpServletRequest request, HttpServletResponse response) {
        throws ServletException, IOException {
            response.setContentType("application/pdf");
            try {
                // step 1
                Document document = new Document();
                // step 2
                PdfWriter.getInstance(document, response.getOutputStream());
                // step 3
                document.open();
                // step 4
                document.add(new Paragraph("Hello World"));
                document.add(new Paragraph(new Date().toString()));
                // step 5
                document.close();
            } catch (DocumentException de) {
                throw new IOException(de.getMessage());
            }
        }
    }
}
```

However, some browsers experience problems when you send bytes directly like this. It's safer to create the file in memory using a `ByteArrayOutputStream` and to tell the browser how many bytes it can expect in the content header:

```
public class PdfServlet extends HttpServlet {
    protected void service(HttpServletRequest request, HttpServletResponse response) {
        throws ServletException, IOException {
            try {
                // Get the text that will be added to the PDF
                String text = request.getParameter("text");
                if (text == null || text.trim().length() == 0) {
                    text = "You didn't enter any text.";

```

---

<sup>981</sup><http://itextpdf.com/examples/iiia.php?id=167>

```
        }
        // step 1
        Document document = new Document();
        // step 2
        ByteArrayOutputStream baos = new ByteArrayOutputStream();
        PdfWriter.getInstance(document, baos);
        // step 3
        document.open();
        // step 4
        document.add(new Paragraph(String.format(
            "You have submitted the following text using the %s method:",
            request.getMethod())));
        document.add(new Paragraph(text));
        // step 5
        document.close();

        // setting some response headers
        response.setHeader("Expires", "0");
        response.setHeader("Cache-Control",
            "must-revalidate, post-check=0, pre-check=0");
        response.setHeader("Pragma", "public");
        // setting the content type
        response.setContentType("application/pdf");
        // the contentlength
        response.setContentLength(baos.size());
        // write ByteArrayOutputStream to the ServletOutputStream
        OutputStream os = response.getOutputStream();
        baos.writeTo(os);
        os.flush();
        os.close();
    }
    catch(DocumentException e) {
        throw new IOException(e.getMessage());
    }
}
}
```

For the full source code, see [PdfServlet<sup>982</sup>](#). You can try the code here: <http://itextpdf.com:8180/book/>

---

<sup>982</sup><http://itextpdf.com/examples/ii.php?id=173>

## Why do I get a “getOutputStream( ) has already been called for this response” error in JSP?



I'm using JDBC to fetch data from database and then I use iText to create a PDF file which can be downloaded on client machine. The application is coded in HTML/JSP and runs on Apache Tomcat.

I use the `response.getOutputStream` to create an output PDF file immediately. However, I get the following error:

**getOutputStream() has already been called for this response**

How can I generate a dynamic PDF file which can be downloaded by client machine?

Posted on StackOverflow on Jun 13, 2013 <sup>983</sup> by Sahil Sharma<sup>984</sup>

When you write JSP, you probably like white space and indentation, for instance:

```
<% //a line of code %>
<%
    // some more code
%>
<% // another line of code %>
<%
    response.getOutputStream();
%>
```

This will always cause the exception "getOutputStream() has already been called for this response" regardless if you're using iText or not. The `getOutputStream()` method was called the moment you introduced your first white space character in your JSP script.

To fix this, you need to remove all white space:

```
<% //a line of code %><%
    // some more code
%><% // another line of code %><%
    response.getOutputStream();
%>
```

---

<sup>983</sup><http://stackoverflow.com/questions/17083318/how-to-insert-image-in-pdf-using-itext-and-download-to-client-machine>

<sup>984</sup><http://stackoverflow.com/users/2367475/sahil-sharma>

Not a single character is accepted outside the <% and %> markers. As explained in the better JSP manuals, you shouldn't use JSP to create binary files. Why not? Because JSP introduces white space characters at arbitrary places in your binary file. That results in corrupt files. Use Servlets instead!

# Legal questions

Although StackOverflow is a forum where developers post *technical* questions and *technical* questions *only*, we notice that some developers also want to know more about the legal aspects of using open source, more specifically: is it legal to use iText for free? Is there a license fee involved?

## What is the difference between Lowagie and iText?



What is the difference between lowagie and iText. Is this just version difference or up-gradation to library. Which one recommended to be used.

Posted on StackOverflow on [Nov 22, 2012](#) <sup>985</sup> by [Adeeb Cheulkar](#)<sup>986</sup>

I am Lowagie, the lowagie you refer to. I'm the original author of iText and the author of the "iText in Action" books.

As explained in the [Sales FAQ](#)<sup>987</sup>, you should use the latest version of iText.

The differences between old versions of iText (iText 2.x.y dates from July 2009 or earlier) and newer versions of iText can be found in the [changelogs](#)<sup>988</sup>.

The 5.0.0 version had the following substantial changes:

- iText and iTextSharp started using the same version numbers
- the iText.jar is compiled using Java 5 (instead of with the JDK 1.4).
- The F/OSS license has been upgraded from MPL/LGPL to AGPL.
- The package names have changed from com.lowagie to com.itextpdf.
- The toolbox and RTF support have been removed: they are now in a separate project at SourceForge.

Numerous bugs have been fixed since July 2009. Functionality that makes your PDFs future-proof such as updates regarding new digital signature standards and new standards such as PDF/UA, PDF/A-2 and PDF/A-3 is only available in the more recent iText versions.

<sup>985</sup><http://stackoverflow.com/questions/13515210/difference-between-lowagie-and-itext>

<sup>986</sup><http://stackoverflow.com/users/1771109/adeeb-cheulkar>

<sup>987</sup><http://itextpdf.com/salesfaq>

<sup>988</sup><http://itextpdf.com/changelog>

## Can iText 2.1.7 or earlier be used commercially?



Can iText 2.1.7 (MPL/GPL) licence be used in commercial projects? I am not a legal guy but lots of discussion threads suggest that there is no issue using the earlier version (2.1.7) of iText in commercial projects as that version is bounded with terms & conditions governed by [MPL<sup>989</sup>](#)/GPL license.

However, if we look at [iText's official website<sup>990</sup>](#), it says as the licence has been upgraded to AGPL licence, one has to buy the software before commercially using it. See the topic entitled *Why shouldn't I use iText 2.x (or iTextSharp 4.x)?*

**LEGAL REASONS:** Older versions of iText under the free model may contain code fragments that infringe other people's copyrights or intellectual property rights. iText Software Group has done a significant investment in identifying and eliminating all those cases as of version 5.1. which is one of the reason why it is now a paying commercial version. We do not recommend the use of versions prior to 5.1 for commercial projects as your company could be liable for copyright or IP infringements.

Of course, this seems a warning only. Discouragement of not using iText with earlier version due to Technical reasons could be understood but Legal reasons are not worth. What about the commercial projects who have been using iText 2.1.7 before the licence upgrade happened in iText? Would they now have to change their whole project planning because iText has now changed his mind to not to distribute it commercially? Of course iText might have done significant investment in upgrading the version technically but what about the investment one might have done in his commercial project using iText 2.1.7 or earlier?

Please someone who understands legal implications of both the licences clarify this confusion. iText can use such warning to encourage its sale but is there anything substantial in such warning? Can one use iText with version 2.1.7 or earlier commercially? Comments from Mr. Bruno Lowagie, the original author of iText are highly appreciated.

Posted on StackOverflow on [Sep 6, 2014<sup>991</sup>](#) by [Devendra Sharma<sup>992</sup>](#)

The first iText company was founded in 2008. The purpose of this company was to put all the Intellectual Property of the code into one legal entity. This was achieved by identifying [1.] every third party project from which code was borrowed, as well as [2.] every individual developer who contributed code.

<sup>989</sup><https://www.mozilla.org/MPL/1.1/>

<sup>990</sup><http://itextpdf.com/salesfaq>

<sup>991</sup><http://stackoverflow.com/questions/25696851/can-itext-2-1-7-or-earlier-can-be-used-commercially>

<sup>992</sup><http://stackoverflow.com/users/2881228/devendra-sharma>

[1.] Some code snippets were borrowed from projects with an ambiguous license. For instance: we had a snippet that was released under Sun's Example License (which allowed us to use the code), but in the comment section of the class, it said that the code was proprietary to SUN (which prevented us to use the code). Which of both prevailed? Being an ignorant developer at that time, I thought the Example License was the one I could use, just like some people claim that you can use iText 2.1.7 today. Lawyers however, disagreed: they said that the most strict license was the valid one.

We solved these problems by (1) asking permission to use code with ambiguous licenses, (2) refactoring code if we didn't get permission, (3) removing code we couldn't refactor.

We did the same with contributions from individual developers.

[2.] The IP from individual developers was transferred to iText Group NV (formerly known as 1T3XT BVBA) by asking every developer who contributed 20 lines of code or more to sign a Contributor License Agreement.

Two problems arose:

1. Individual developers could not be reached. For example: we dropped the RTF package completely because we couldn't find a couple of the core developers of the RTF functionality.
2. In a couple of cases, we had to negotiate about the CLA. For example: one company didn't like the CLA. Instead, this company released the contribution of its employees under an MIT license, so that we could use it anyway. Another organization was really slow in agreeing with the CLA. It took us until September 2009 before we received formal approval. Only after this approval, we switched to the AGPL. I can't disclose the document (it was different from the CLA), nor the name of the organization (I hope I don't break the NDA just by writing this). I can only say that we only had full coverage of the code base after that document was signed.

Ignorant developers claim that the LGPL/MPL header "protects" them, but what if some proprietary code was accidentally added to a class with such a header? Does this make that proprietary code "available under the MPL/LGPL"? If it did, it would be sufficient to take proprietary code, add an MPL/LGPL header and publish it. Doing this on purpose would be illegal. Doing this out of ignorance can be pardoned if there is a willingness to fix the issue.

In the early years of open source, it did occur that proprietary code got mixed into an open source project by accident. At iText, we have invested a lot of time and effort into cleaning up the code base. Since that exercise, we are very disciplined with respect to code contributions. This is one of the core tasks of a professional open source company.

After we fixed all the issues, we removed all copies of those old iText versions from our servers to make sure we were in the clear. If a company decides to use some rogue version of iText 2.1.7 that is outside of our control, this company does so **willingly and knowingly**, in other words: **at its own risk!** There is no way such a company can claim: "*We didn't know there was a possible IP issue with the code.*"

If you want to use iText 2.1.7, you need to do the exercise we have done between 2007-2009 at your own expense. This will cost you more than the price of a license. For instance: the individual

developers gave permission to iText Group NV to do business with iText, but will they give that permission to you? How will you identify those individual developers?

Moreover: iText 2.1.7 dates from July 2009, meaning that it is more than 5 years old. Many bugs have been fixed since that date. Should you knowingly introduce those bugs into the code base of your customer, then your customer may claim that you had an alternative: you could have used a more recent version of iText...

As for your question “*what about the investment one might have done in his commercial project using iText 2.1.7 or earlier?*” That investment must have been done at least 3 years ago, because we’ve been informing people that they should upgrade for at least that long. Upgrading to a recent version is an investment that should be categorized as a maintenance cost. It should be an affordable cost because whoever has been using iText 2.1.7 for that long in a commercial project has been making money thanks to iText for that long. Claiming that “iText has *now* changed its mind” is not correct unless *now* is marked as a synonym of *5 years ago* in your dictionary.

## Can I use iText without respecting the AGPL license?



If someone uses the iText library in an android application for PDF without purchasing license, how will iText know he did that? Since he will not be releasing the source code how will they know he used their library? how could they take legal action against him? Again I am not planning in doing this. I was just curious.

Posted on StackOverflow on Oct 31, 2014 <sup>993</sup> by Shiva<sup>994</sup>

iText contains some visible fingerprints (e.g. the producer line) as well as invisible fingerprints (which we obviously do not share).

On a regular basis, we discover PDFs that were created using iText, but that can’t be linked to a paying customer. In that case, we contact the person distributing the PDF.

**Example one:** a large publishing house in Germany was distributing PDFs created with iText. We contacted that company in a friendly way and explained that we wanted to talk to them about their use of iText. At first, they were surprised: they didn’t know they were using iText. So we showed them the PDFs and as it turned out, the company had hired an external integrator to build an application. That integrator had introduced iText without purchasing a license. The first thing that happened was funny: suddenly the producer line was removed (which is not allowed), but we could still prove that iText was used (because of the secret fingerprints). So we explained the publishing house that we didn’t really appreciate that. As a result, the integrator was offered the choice: either they complied and bought a commercial license, or the publishing house would never hire them again.

<sup>993</sup><http://stackoverflow.com/questions/26673601/if-someone-uses-the-itext-library-in-an-android-application-for-pdf-without-purc>

<sup>994</sup><http://stackoverflow.com/users/3139715/shiva>

**Morale of this story:** it is very bad for your reputation and for your business if you are revealed as being a fraud. If we go to your customer and it turns out that you fooled him, you risk losing that customer.

**Example two:** a medium-sized company was using iText and other open source software without worrying about the licenses. At some point of time, this company was about to be acquired. The acquiring company obviously went through a due diligence process and discovered the mess. We were contacted by the medium-sized company because they needed a commercial license ASAP... In at least one case, the company didn't get acquired because of the mess.

**Morale of this story:** do the right thing. If you make money using our software, it is only fair that you pay for your use of our software.

**Example three:** a developer at a company using iText informed us that his management deliberately chose to use iText in an illegal way. He provided us with proof, and we contacted the legal department of that company. No legal action was needed: the company purchased a license and is now a very happy customer because they are now really benefiting from the commercial relationship we have with them.

**Morale of this story:** in most cases, it isn't even necessary to go to court. When faced with the evidence, it is less expensive to come to an agreement than to risk losing a case (and your reputation).

These three stories have one thing in common: **honor**.

I have come close to going to court a couple of times, but eventually, the problem solved itself because the people involved understood that it wasn't in their interest getting sued.

Also: it is counter-productive to sue somebody into paying a license. It is much better to create a win-win situation where the company using iText benefits from their business relationship with the iText Group. Especially now that we're winning awards (such as the [BelCham Award<sup>995</sup>](#) and the [Fast 50 Award<sup>996</sup>](#)), good PR is very important. The more iText Group grows, the more companies realize that it's important to work with us. The more companies realize this, the more we grow ;-)

I hope this helps. I also hope this explains why I can be very harsh towards people with nick. Most of the times there's a reason why people want to remain anonymous and that reason isn't always a good one.

**Update:** Our site tracks usage statistics. In many cases, we can see which companies are visiting our site; in some cases, we can even track individuals. A couple of years ago, a company was denying that they were using iText, but they were visiting <http://itextpdf.com> 200 times a year! Confronted with those numbers, they admitted that they had lied to us. When we look at the global scale, we see that India is #2 and China is #4 in visits, but both companies have a very low ranking in sales. That's why we've now [opened an office in Asia<sup>997</sup>](#). One of our sales people is now visiting India, Malaysia, and other countries in the East to talk to companies about their use of iText.

---

<sup>995</sup><http://itextpdf.com/events/winner-most-promising-company-of-the-year-2014-by-belcham>

<sup>996</sup><http://itextpdf.com/events/deloitte-technology-fast-50-Belgium-2014>

<sup>997</sup><http://itextpdf.com/itext-goes-east-go-global-singapore>

## Is iText Java library free of charge or are there any fees to be paid?



Are iText Java libraries to generate PDF documents free, or do we have to pay for them?

Posted on StackOverflow on Jan 9, 2015 <sup>998</sup> by Deven Shah<sup>999</sup>

I am the original developer of iText and the CEO of the iText Group. I'm also a Mentor at the Founder Institute<sup>1000</sup>. Please take a look at my slides for the session about [Startup Legal and IP for the Founder Institute](#)<sup>1001</sup>.

iText is software and therefore copyright law applies:

Copyright law allows an author to prohibit others from reproducing, adapting, or distributing copies of the author's work.

According to the copyright law, you do not have the right to use software that you didn't write yourself.

However, iText is distributed using a copyleft license:

Copyleft gives every person who receives a copy of a work permission to reproduce, adapt or distribute the work as long as any resulting copies or adaptations are also bound by the same copyleft licensing scheme.

This means that anyone can use iText for free as long as the conditions for its use are met.

To know more about the conditions, you need to take a look at the license. In this case: the AGPL. This means that:

- You can not distribute a closed source application that is based on iText without distributing the full source code of your own application.
- You can not use iText in a web application without making the full source code of your web application available through that web application.

<sup>998</sup><http://stackoverflow.com/questions/27867400/is-itext-java-library-free-of-charge-or-have-any-fees-to-be-paid>

<sup>999</sup><http://stackoverflow.com/users/4438096/deven-shah>

<sup>1000</sup><http://fi.co/>

<sup>1001</sup><http://www.slideshare.net/blowagie/startup-legal-and-ip>

This is why people often refer to the AGPL as a viral license: all the software that touches an AGPL library such as iText needs to be free too.

Obviously, there are plenty of companies who do not want to ship their source code. That's why iText Software also provides iText under another license. This license is a commercial license. You have to pay for it.

**To answer your question:** iText can be used for free in situations where you also distribute your software for free. As soon as you want to use iText in a closed source, proprietary environment, you have to pay for your use of iText.

## Can I bundle iText with my non-commercial software?



I'm trying to create a java based plugin for an IDE (non eclipse), for which I need PDF generation. I started using iText, but I read somewhere that I cannot bundle this as part of my plug-in for distribution. I'm finding it hard to comprehend license agreements.

Posted on StackOverflow on Apr 13, 2015 <sup>1002</sup> by user907737<sup>1003</sup>

If you want to distribute iText with your application, you should take into account that:

- if you distribute your source code as AGPLv3 software, you can use iText under the AGPLv3 at no charge (there's no money involved).
- if you distribute your source code under another type of open source license (e.g. the EPL, the Apache Software License,...), then you can't use iText because such a license isn't compatible with the AGPLv3 (there may be exceptions, such as the GPLv3 *under conditions*).
- If you distribute your source code under a closed source license (e.g. a commercial license), then you can only use iText if you purchase a commercial iText license.

I teach this kind of stuff, so allow me to copy/paste some slides from my [Startup Legal and IP class](#)<sup>1004</sup>.

---

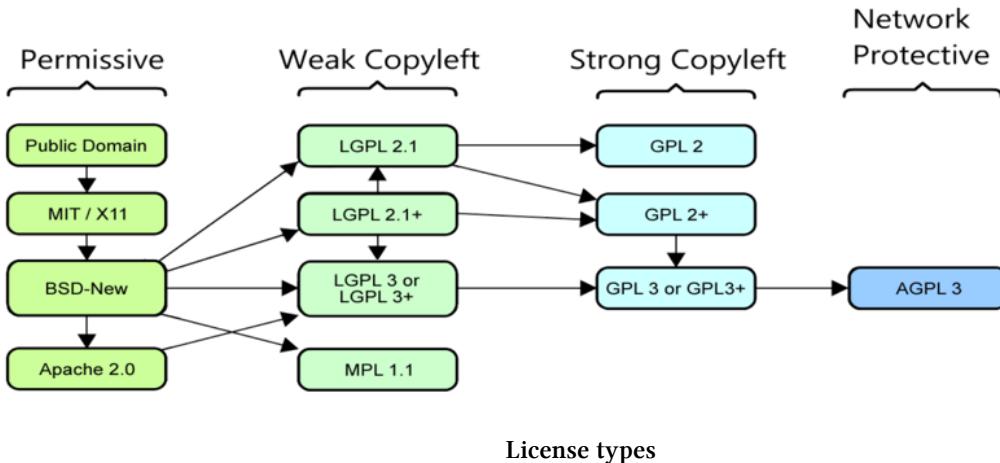
<sup>1002</sup><http://stackoverflow.com/questions/29604663/java-pdf-creation-options-free-to-bundle>

<sup>1003</sup><http://stackoverflow.com/users/907737/user907737>

<sup>1004</sup><http://www.slideshare.net/blowagie/startup-legal-and-ip>

Quote: <http://zguide.zeromq.org/page:all#header-136>

- ☰ A permissive license makes most people see you as lunch,
- ☰ A proprietary license makes most people see you as an enemy,
- ☰ A copyleft license makes most people your ally.



License types

The quote was borrowed from [the ZeroMQ guide<sup>1005</sup>](#); the figure is an adaptation of a [schema drawn by David Wheeler<sup>1006</sup>](#).

Based on personal experiences in the business, I'd say it boils down to this:

- if you publish your code under a permissive license, companies using your software will see your work as a free lunch. They will use it, but there is very little chance you'll be rewarded for your work. On the contrary: they will demand a solution if they have a problem with your software. (This is what happened to me: my son was in hospital with Cancer for almost a year and I got harassed by a company that expected *me* to solve *their* technical problem for free.)
- if you publish your code as a closed source product, companies may see you as somebody stuck in the 20th century, because *everything needs to be open source today* and you're not very friendly if you only have a closed source offering.
- if you publish your code under a copyleft license, then companies are obliged to cooperate, either by enhancing the software (the way you intend to enhance it), or by contributing financially which allows the open source vendor to invest in further development.

iText used to be available under the MPL/LGPL, but switched to the AGPL in 2009. All the MPL/LGPL versions have been removed from our servers (see [this question<sup>1007</sup>](#) for more info).

What's the difference between LGPL, GPL and AGPL? This is explained in this overview:

<sup>1005</sup><http://zguide.zeromq.org/page:all#header-136>

<sup>1006</sup><http://www.dwheeler.com/essays/floss-license-slide.html>

<sup>1007</sup><http://stackoverflow.com/questions/25696851/can-itext-2-1-7-or-earlier-can-be-used-commercially>

# What is distribution?

Suppose that iText was an engine and the application using iText was a car or a bus:

License: 	MPL / LGPL	GPL	AGPL
Car distribution (e.g. OEM)			
Commercial use?	OK for gratis commercial use	Commercial license needed	Commercial license needed
Bus service (e.g. SaaS)			
Commercial use?	OK for gratis commercial use	OK for gratis commercial use	Commercial license needed
Free/Proprietary	<b>Before iText 5:</b> Improvements engine: LGPL Car or bus: can be proprietary	Car: must be GPL Bus: can be proprietary	<b>Since iText 5:</b> Car or bus: must be AGPL Or: buy commercial license

## What is distribution?

Suppose that we'd be talking about hardware, more specifically about an engine.

- If the engine is AGPL, then you can use that engine for free to build a car, and you can distribute this car commercially: you can ask money for selling the car and you don't have to pay for your use of the engine. Your main obligation is that you share all the improvements you've done to the engine.
- If the engine is GPL, then you can only use that engine for free inside a car if you also distribute the car for free under the same license. The moment you start selling cars under other conditions, you can no longer use the engine for free. However, if you use that engine inside a bus and instead of selling the bus, you make money selling bus tickets, you don't need to pay for your use of the engine. This is what many SaaS companies do: they do not distribute software executables, they offer Software as a Service. This is commonly referred to as the "SaaS loophole" of the GPL.
- The AGPL closes the SaaS loophole: with the AGPL, offering Software as a Service is also considered as "distribution", hence you need to distribute your complete application as AGPL for free. You can usually escape from this obligation by purchasing a commercial license.

So the answer is really simple: you want to offer your plug-in for community use. If you choose to release your plug-in under the AGPL, then you can use iText and we'll even promote your plug-in (e.g. by writing a blog post about it on the iText site). If you choose to release your plug-in under a license that isn't compatible with the AGPL, you shouldn't use iText.

## When does Bruno Lowagie act like “kind of a dick”?



Although the author has removed his Tweet, I once received the following notification from Twitter:

The screenshot shows a Twitter notification from Bruno Lowagie (@bruno1970) stating "You were mentioned in a Tweet!" followed by a tweet from a user whose profile picture is blurred. The tweet reads: "Spent a lot of time today on Stackoverflow and realized @bruno1970 is kind of a dick." It was posted at 10:28 PM - 15 Dec 14. A "Follow" button is visible next to the user's name.

### Twitter insult

This isn't really a legal question, because there is freedom of speech, but sometimes I regret that I even bother to look at new questions.

The following insult was written by a StackOverflow user with a reputation of 1 (meaning that he didn't contribute anything substantial to StackOverflow yet):

A screenshot of a StackOverflow comment from a user with a reputation of 1. The comment reads: "you are a pretentious pedantic that not bring anything to my solution. no one right way to program and would know if you were "the original developer of iText" and you have to live with his hurt ego. – Eleazar Ceballos 12 hours ago". Below it, another user replies: "Great quote! I'll add it to my book ;‐) – Bruno Lowagie just now edit x".

### Insult by a StackOverflow user with a reputation of 1 (can't get any lower)

When something like this happens, I don't feel like answering any other question for a while, and people who do have a genuine question often don't get the answer they deserve, because some questions remain unanswered if I don't spend any time on them.

What usually triggers such a situation?

There are three types of questions that trigger a harsh comment or answer.

#### [1.] Questions that sound like “I have tried X and it doesn’t work!”

Questions like this can be OK because software remains software, and it's always possible that something doesn't work because of a bug. However, it is important that you phrase such a question in a way that isn't offensive. I sometimes reuse this quote I found on StackOverflow:

3 `doesn't work` is not a problem description that a software developer should give when asking a question on an internet forum. That's a problem description most commonly employed by standard users that do not understand or care about how computers work. – Darin Dimitrov yesterday /

### It doesn't work

There is a clear rule when you claim that something doesn't work: you have to provide proof. You can't just declare that something doesn't work and expect every one to believe you. The best way to convince people is by writing a Short, Self Contained, Correct (Compilable) Example, better known as an [SSCCE<sup>1008</sup>](#). Without such an example, it is very hard for third parties to check whether or not your allegation is correct. Remember that you are asking a favor: you have a problem and you want *other people* to solve your problem *for free*. By not providing some sample code that can be used to reproduce the problem, you are creating a threshold that may be too high for people willing to spend *some time* on your question, but not *too much time*. If the problem can be reproduced by compiling and running the SSCCE, you have a better chance at getting people's attention. Many developers on StackOverflow love solving mysteries. Give them a problem they can reproduce and they'll give you the time they might otherwise spend on a silly Sudoku.

If the problem can't be reproduced by compiling and running the SSCCE, people get frustrated. Especially when you say things such as: "*I have read the documentation and the examples don't work.*" That can easily be interpreted as "*The documentation is badly written,*" resulting in a reply: "*Maybe the documentation is badly read. Did you even try to understand the examples?*"

You're a developer, not a copy/paste artist. Don't use code you've found somewhere in the wild and expect it to work if you don't understand what the code is for. On several occasions, I have written documentation showing *how not to do something* versus showing *how to do something correctly*. You should know which example is which before you start copy/pasting code. Therefore it is important to *read the documentation!* Some people have no idea how much time and effort is spent on writing documentation, and have no respect at all for the author. Obviously, you are not one of them, as you are reading this book.

### [2.] Questions that sound like “I have searched the whole internet and I didn’t find a solution!”

Never start a question with a lie that can easily be disproven by a link to the appropriate documentation. It makes for a bad first impression and people will not be inclined to help you.

The main question is: "*Did you do an effort?*" Whatever you claim about doing research, people can usually tell from the way you phrase your question if you took the time to solve the question yourself, or if you were just lazy and threw the question on StackOverflow. A great way to answer questions like this, is by using the service [let me Google that for you<sup>1009</sup>](#), but it's forbidden to link to that site on StackOverflow. I do see [whathaveyoutried.com<sup>1010</sup>](#) in a comment once in a while. When I started my career, I was asked to read "[How To Ask Questions The Smart Way<sup>1011</sup>](#)" by Eric Raymond. I loved it! You should read it too if you haven't already.

<sup>1008</sup><http://sscce.org/>

<sup>1009</sup><http://lmgtfy.com/>

<sup>1010</sup><http://whathaveyoutried.com/>

<sup>1011</sup><http://www.catb.org/esr/faqs/smarter-questions.html>

On a side note, I want to mention that it is possible to mark questions as *duplicate* on StackOverflow. However, you can only refer to questions that have either been accepted or that have received at least one up-vote. When I read the 1,000+ answers I have posted on StackOverflow to compile this book, I noticed that many of my answers weren't accepted and received zero up-votes. This is sad, isn't it? It only takes a single click to indicate that an answer was helpful, yet that's already too high a price for receiving a good answer.

### [3.] Questions that sound like “I can’t upgrade to the latest iText version because we can only use the open source version”

Free software is licensed software. iText is free software. That doesn't mean that iText is *for free*: iText has always been distributed with a copyleft license.

Some people only read the first part in the definition of “copyleft”. They read about receiving “permission to reproduce, adapt or distribute” iText, but they forget about the second part: “as long as any resulting copies or adaptations are also bound by the same copyleft licensing scheme.”

iText versions pre-dating iText 5 were versions using a *weak copyleft* license (more specifically the MPL/LGPL). This means that you can use such a version in an application that isn't bound by the copyleft license. Your only obligation is to distribute the changes you make to iText under the MPL, the LGPL, or the MPL/LGPL. In theory, those versions can still be used inside applications that are not free. In practice, you should no longer use those old versions for both technical as well as legal reasons (as explained in one of the previous questions).

Starting with iText 5, iText has increased software freedom, in the sense that the license was changed to a very strong (some use the word “viral”) copyleft license, more specifically the AGPL. When you use iText in an application that you distribute or when you use iText in a web application that allows people to directly use iText (through a SaaS application, on a web site,...), you have to distribute all the source code that touches iText *under the same license* and *under the same license only*.

Some people argue: “*We do not modify iText, hence we are not bound by the AGPL.*”

That assumption is **incorrect**. Writing a web application that uses iText is considered being a modification in the context of the AGPL and putting this application on a web server for people to access is considered being distribution.

Nothing is more annoying than hearing people claim that iText is no longer open source. I'm sorry if I get rude when I hear such lies. iText is still open source software: you can download the source code and take a look inside. The new version of iText leads to more freedom in the sense that almost every snippet of code that touches iText becomes free software too.

If you have customers or an employer who wants a closed source version of your product, you or your employer should buy a *commercial* license from iText Software. That's only normal: we are continuously investing in research and development. We also spend a lot of time and energy answering questions to help out developers. We are here to *help* developers. Help us help you.

Don't say: “*I didn't make the decision to use an old iText version. Management made that choice and I have no responsibility whatsoever in this matter.*”

Say: “*I didn’t make that decision, but I will help iText Software to get in touch with the people that did.*” I can understand that you aren’t as assertive an employee as I used to be before I decided to start my own company, but the least you can do, is to help iText Software explain to your management, CTO to CTO, why using the old iText version is a bad idea.

I was a member of a panel at Devoxx (the largest developer conference in Europe) in 2014. During this panel session, David Blevins ([tomtribe.com](http://tomtribe.com)<sup>1012</sup>) made a useful suggestion. A report of this panel session can be found on [Geertjan Wielenga’s blog](http://geertjanwielenga.nl/reflecting-on-open-source-software-java-9-and-startup-strategies-at-devoxx-2014)<sup>1013</sup>. I quote:

“*We have this fairytale idea that open source is an infinite resource, but it’s not,*” Blevins said. An approach that was suggested is for developers to approach their organization’s CTO and say to them: “*Let’s outsource all our software development to people we don’t know*” and then, when the CTO looks surprised and annoyed at the suggestion, say: “*That’s what we’re doing already, shouldn’t we get to know the organizations behind the software we’re using?*” The bottom line reached by the end of the discussions was that organizations need to take responsibility. Figure out who is behind the open source software you’re using and then [feed the fish](http://tomtribe.com)<sup>1014</sup> in one way or another.

Be a good developer. Know that there’s no such thing as a free lunch. Make the choice up-front:

- Are you going to establish a business relationship with iText software (so that we can help you with all your PDF problems)?
- Are you going to open source your own code under the AGPL (which excludes you from offering your software under a closed source license)?
- Or, are you going to waste your time trying to work around the limitations of an obsolete version of iText?

If you choose the first or the second option, you contribute to the project you are using and benefiting from. That’s what makes the world turn round and keeps the eco-system alive.

If you choose the third option, you are not helping the open source community at all and you shouldn’t expect any sympathy from any one. I know that there are some iText forks around. I call such a fork a *gork*, because God Only Really Knows what’s inside. Usually, these gorks are based on iText 2.1.7 and they are thrown on the internet without the quality control we have in place at iText Software. You’ll notice that the people who made these gorks don’t maintain them and don’t answer any questions. If something goes wrong with such a version, don’t blame iText software! Blame the person who published this *gork*!

If you can’t afford a commercial license, it is better to use an official version of iText under the AGPL than to use an unofficial version from a developer you do not know and who is not going to help you when you’re in trouble.

<sup>1012</sup><http://www.tomtribe.com/>

<sup>1013</sup><http://www.theserverside.com/news/2240234582/Reflecting-on-open-source-software-Java-9-and-startup-strategies-at-Devoxx-2014>

<sup>1014</sup><http://www.tomtribe.com/blog/2013/11/feed-the-fish/>

Once there was a developer who needed functionality introduced in a 2012 standard (PDF/UA, initially released on July 15, 2012). However: he insisted on using a rogue version of iText dating from 2009. Ignoring that the version of iText was three years older than the standard he wanted to support (*a technical argument*), he demanded a solution *for free*.

I explained that the development cost to support PDF/UA was fairly high and that he probably wouldn't receive any other answer. If he didn't want to pay, his best option was to use the AGPL version:

---

[pdfua.pdf](#) proves that my answer to that question adequately solves your problem. I have removed your remark about "price" (that is a *technical site*). One can always use iText under the AGPL if one can not afford buying a license: [Is iText Java library free of charge or are there any fees to be paid?](#) There is no reason why you should use a rogue version of iText of which God Only Really Knows what's inside. – [Bruno Lowagie](#) 20 hours ago

#### If you don't want to pay, use iText under the AGPL

Although my answer was correct (pdfua.pdf was exactly what he needed), I received a down-vote and the following reply:

[Making the source code available to meet the terms of the AGPL will cost time, manpower, and money. It would not be free since we do not meet the terms of the AGPL already. Therefore in order to use AGPL software I would have to feedback significant justification to my \(non-profit\) project to get these resources. I see sidestepping here, not justification.](#) – [REDACTED] 15 hours ago

#### Open sourcing the own code also costs

Note that the same person also wrote this:

---

Thank you for your response. Do you think you could link to the actual question on Stack Overflow that you are referring to rather than an advertisement for your ebook? – [REDACTED] yesterday

[REDACTED] "an advertisement for my ebook"? The ebook is available for free! Why would you complain about a link to free documentation? – [Bruno Lowagie](#) yesterday

---

I don't consider anything that requires registration for download to be truly free. – [REDACTED] 15 hours ago

#### The free ebook is not appreciated

Seriously? People like this remind me of the proverb: *Givers have to set limits because takers rarely do*. This person went on, claiming there was a conflict of interest on my side and complaining that price *is* a technical constraint:

[It seems to me that it is a conflict of interest for someone who profits from sales of iText to remove a remark about price, and that you would recuse yourself from such edits in the interest of protecting your appearance of integrity. That said, I do think that resources do become a technical constraint on what libraries you can use, and this is a reality for most programmers. Most of us have to justify our resource usage.](#) – [REDACTED] 15 hours ago

#### Pricing, a technical issue?

To which I replied:

You are looking to get value for nothing. Even to get a free ebook, you are not willing to register. You don't have any moral objections using code that I wrote for free, but when I explain that there's no such thing as a free lunch, you talk about a conflict of interest and me having to worry about my integrity. Justify that... – [Bruno Lowagie](#) 46 mins ago

### You are looking to get value for nothing

Some people say that I have an *attitude problem*, but I say: it's a *gratitude problem*. If that makes me a dick, so be it.

# To be continued...

All the answers and the many code samples I have provided on StackOverflow were written in the hope that they are helpful. I leave it up to the reader of this “Best of” selection to decide whether or not “*I’m kind of a dick*” as the people who down-voted some of my answers claim. I just love answering questions, and where love is involved, there’s also pain, for instance the pain if the love isn’t returned. Some people seem to make a sport out of it to beg for an answer and then to thank me by saying: we’re never going to be a customer of iText Software. Somehow that doesn’t compute. I hope you understand.

Obviously, a book like this is never finished. New questions about iText are posted every day. I expect that this book will grow over the years. Some answers may become obsolete, some new functionality will require more clarification. This clarification may be provided in the form of an answer to a new question, or as a topic in one of the other upcoming books:

- The ABC of PDF<sup>1015</sup>
- Create your PDFs with iText<sup>1016</sup>
- Update your PDFs with iText<sup>1017</sup>
- Sign your PDFs with iText<sup>1018</sup>

All of these books are available for free. No donation is expected.

---

<sup>1015</sup>[https://leanpub.com/itext\\_pdfabce](https://leanpub.com/itext_pdfabce)

<sup>1016</sup>[https://leanpub.com/itext\\_pdfcreate](https://leanpub.com/itext_pdfcreate)

<sup>1017</sup>[https://leanpub.com/itext\\_pdfupdate](https://leanpub.com/itext_pdfupdate)

<sup>1018</sup>[https://leanpub.com/itext\\_pdftsign](https://leanpub.com/itext_pdftsign)