

Outline Text

aa

08.24.26 Revision

Walton Mendelson

12on14.us

Cover: the “a” on the left shows a Bezier curve around the “a”: it is a container for the fill color. It is non-printing.

© 2026 Walton Mendelson
All rights reserved.

Photoshop, InDesign, and Illustrator are registered trademarks of Adobe Inc. CreateSpace (CS) and Kindle Direct Publishing (KDP) are registered trademarks of Amazon.com Inc. Affinity is a registered trademark of Canva. IngramSpark (IS) is a registered trademark of Ingram. Word and Microsoft fonts are copyrighted by or trademarks of Microsoft Corporation. Pantone is a trademark of Pantone LLC. All trademarks, these and others mentioned in the paper, are the property of their respective owners.

This PDF, One-Off Press, and the author are independent of any vendor, product, company, or person mentioned in this book. No product, company, service, or person mentioned or quoted in this book has in any way, explicitly or implicitly, endorsed, authorized, approved, or sponsored this paper.

Table of Contents

Fix KDP “Unsupported Font” Errors	5
Outline Text	7
Affinity	8
InDesign	8
Default Method for Acrobat DC	9
Compression/Optimization	10
Alternate Methods in Acrobat	12
Ghostscript	13
Downloading	13
Command Prompt	16
Ghostscript on Mac	18
Building the Command	20
Putting it Together	23
Running It	24
LightPDF—Alternative to Acrobat	25
Check Fonts	26

ERROR

If your book has errors, it won't meet our quality standards. To publish your book, you will need to fix the errors.

Unsupported fonts are used in your file. Please create a new interior file with "LucidaSans", "LucidaConsole", "LucidaCasual", "LucidaSansUnicode", "LucidaCalligraphy-Italic" replaced. [Learn more about fonts.](#)

Review

Fonts are not properly embedded in your file. We've embedded them for you, but this might slightly change your book's appearance. Please check that your book still looks as intended. [Learn how to embed fonts.](#)

3 4

[Previous Issue](#)

2/2

[Next Issue](#)**Two Page View**

Thumbnail View

Page Range:

4-5

/ 2



Suspendisse aliquam dui lorem, eget interdum tortor posuere id. Aenean vitae sem est. Sed consequat mauris id enim gravida commodo. Nam ac lacus vestibulum quam auctor aliquet accumsan a dui. Fusce at mi ut ex vulputate consectetur. Integer lacinia enim at neque lacinia, condimentum facilisis mauris placerat. Donec ut maximus risus, nec blandit diam. Morbi eu nibh magna. Quisque nisi nisl, semper vitae turpis hendrerit, tincidunt dignissim lacus. Vivamus lacinia, ipsum vitae scelerisque cursus, arcu nisi posuere arcu, sed sodales ligula lorem vel nulla.

Lucida Sans Unicode . . . Vivamus pellentesque id justo id euismod. Curabitur eget sapien erat. Aliquam ligula enim, interdum et malesuada in, condimentum vitae nisi. Praesent ex lectus, bibendum sit amet ultrices sit amet, accumsan a sem. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Praesent feugiat est arcu, facilisis interdum arcu scelerisque non. Fusce commodo erat nec enim eleifend, vel porta ante finibus. Sed leo felis, gravida sed vulputate at, varius in nibh. Aenean blandit nulla et est fermentum congue. Aliquam vehicula ex dapibus scelerisque posuere.

Cras quam tortor, gravida nec fringilla et, molestie vel justo. Ut convallis faucibus mi, in elementum arcu feugiat vehicula. Integer eget porta mi. Proin eu condimentum turpis, ac dignissim nisl. Fusce id quam quis enim tristique pharetra ac et nisi. Vivamus pretium dolor at egestas consectetur. Nunc rhoncus libero quis cursus auctor. Donec porttitor lacus pretium egestas aliquam. Duis lobortis molestie ornare. Aliquam gravida

A screenshot of a test book in KDP's Previewer. The book was created to test KDP's Unsupported Fonts Error policy in February, 2026.

Fix KDP “Unsupported Font” Errors

Convert Text to Outline

As of January 2026, KDP is blocking books that contain “unsupported fonts.” KDP does not provide a way to determine which fonts are supported. If your book is blocked, KDP offers two options:

- Change fonts (limited to a short list of recommended families¹)
- Convert text to outline

Changing fonts requires reformatting the entire book. Converting text to outline is the fastest and most reliable solution. It has been used in professional printing since the mid-1980s to eliminate font-related issues.

Pros

- Eliminates font-related errors
- Preserves layout and typesetting
- Reliable for KDP submissions

Common concerns (frequently mentioned online)

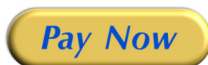
- Text may appear bolder or jaggy
 - In practice, differences are negligible (see page 7)
- Some formatting may be affected (e.g., bullets, underlines)
 - Rare in well-prepared PDFs. I have never seen it in almost twenty years of doing it.
- File size will increase
 - Generally not a problem given KDP’s 650MB limit; however, see **Example**, page 13
- Text cannot be edited in the PDF
 - Editing should always be done in the source file

Do it yourself—or have it done

The following pages show how to convert text to outline using Affinity, InDesign, Acrobat, Ghostscript (free), and LightPDF.

If you prefer not to deal with the process, I offer a **conversion service for \$25**:

- 1: Make the payment through PayPal² (you do not need an account)



- 2: Return to this page and use the upload link³ to upload your file.



- 3: Send me an [email](#) with your book’s title.

Allow 1–3 days—however, the turnaround is usually a few hours if the file is received during business hours.

Please review the file carefully before uploading it.

¹ Customer Support says: Centaur, Garamond, Hightower Text, Palatino Linotype, Amazon Endure, Constantia, and Cambria, from *Paperback Fonts* https://kdp.amazon.com/en_US/help/topic/G202145450

² Paypal: https://www.paypal.com/donate/?hosted_button_id=3RGSTYT754C2J

³ Hightail: <https://spaces.hightail.com/uplink/12on14-review-AcdT13u>

Outline Text

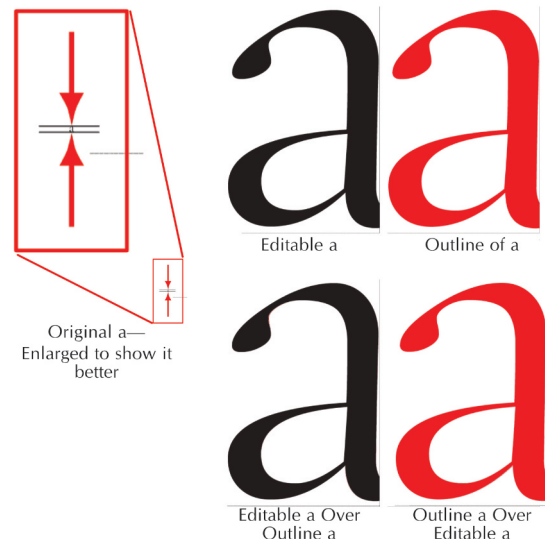
Outlining text—also called “converting text to curves” or “text as curves”—means converting editable text into graphic objects. In the early days of digital printing and desktop publishing, font handling was often unreliable, especially across different systems. Converting text to outlines was a common solution to minimize text-related printing problems.

Today, almost every expert will tell you not to use outlined text. Here’s a composite from the Internet: “This is one of those urban myths . . . Certain attributes will be lost . . . The truth is that you should almost never have to outline fonts . . . It’s a crazy thing to do . . . I’ve heard the reasons, and they all make me sad . . . If your printer wants you to outline text, you should find another printer . . . Printers who want outlined text are using antiquated RIPs . . . are ignorant . . .”

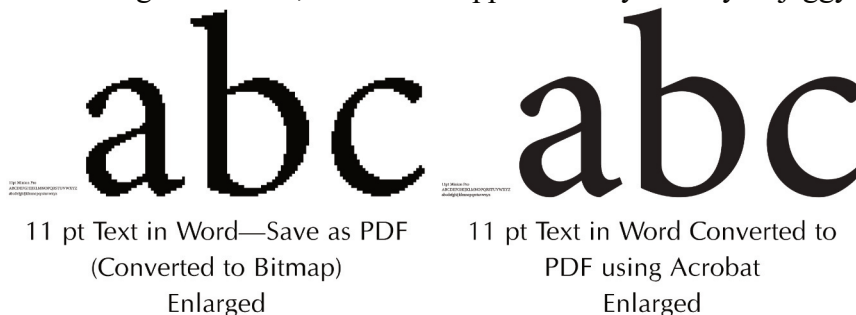
I doubt that Amazon KDP is old-fashioned or ignorant, or uses antiquated software. These companies print millions upon millions of books a year. Their presses are designed to run 24/7. One press can print 20,000 book pages an hour, the equivalent of sixty-six 300-page books. Yes, sometimes stuff happens, and historically, font handling has been one of the biggest problems. KDP is not asking for outlined fonts, except as a fix for a problem that is ill-defined. My point is simply that, in some cases, converting text to outlines solves real production problems.

Two common objections to converting text to outline are that the outlined text gets thicker, and outline text can be “chunky.”

I have converted text to outline scores of times for clients and many more times for tests. And I have seen the text appear to thicken. I hit undo/redo to assess the problem: thicker-thinner, thicker-thinner. The eye is very sensitive to subtle shifts, even when the differences are minimal. I took an “a” from a block of text that appeared to thicken. I compared the editable version to the outline version. Here, the text has been enlarged 14,000%. You can just see a slight red fringe outside the black, editable “a” when it is placed over the red outline “a”.



The other problem, the outline text getting chunky, is clearly not a problem here. Conversions using Acrobat, InDesign, and Affinity are all like the previous example: sharp and smooth. However, until quite recently, Word wouldn’t embed OpenType Fonts (e.g., most Adobe fonts). When Word converts a file to PDF/A, it rasterizes (Word calls it “bitmaps”) those fonts. When enlarged to 600%, those fonts appear visibly chunky or jaggy.



I have seen five types of problems that converting text to outlines can fix:

- Font export issues (PDF): Some fonts do not export reliably to PDF unless converted to curves. Until recently, Word did not properly embed certain OpenType fonts (e.g., Adobe fonts).
- Complex text situations (printing): Books containing dense typography, type specimen material, mathematical formulas, or other complex layout elements may not print correctly or may be rejected as “too complex” or “corrupt.”
- Font substitution during print processing: All PDFs received by print-on-demand companies such as KDP and IS are processed before printing. During that preparation, errors can occur. I have seen books in which some text printed as boxes (as if glyphs were missing), or fonts changed unexpectedly. In one instance, several running heads in a 600-page book were replaced with Hebrew glyphs. The substitution occurred in the same locations across scattered retail copies, but not in the proof copy the client approved.
- Occasional font corruption during printing:

<i>BarlowCondensed—light italic</i>	<i>Sphinx of black quartz, judge my vow.</i>
ĚňňĚv ĚĚĚĚĚĚĚĚ ĚĚĚĚĚĚ *	ânfiaewęż t őt ě nōňňřř ōůžķ ě x ůĚv °
ĚňňĚv ĚĚĚĚĚĚĚĚ ĚĚĚĚ *	ânfiaewęż t őt ě nōňňřř ōůžķ ě x ůĚv °

This same example, converted to outline. It prints perfectly:

<i>BarlowCondensed—light italic</i>	<i>Sphinx of black quartz, judge my vow.</i>
<i>BarlowCondensed—regular *</i>	<i>Sphinx of black quartz, judge my vow.</i>
<i>BarlowCondensed—italic*</i>	<i>Sphinx of black quartz, judge my vow.</i>

- KDP’s “Unsupported Font Error,” as noted on page 5, is fixable by: 1) changing fonts, and depending on your book, having to reformat it; or 2) converting text to outline.

Depending on what program you are working in, you have different options:

Affinity Publisher: export to PDF. Select either 1) the Flatten preset, or 2) *Advanced > Embed Fonts > Text as Curves*. Flatten produces a file 2–3 times bigger than Text as Curves. This is the safest and easiest of all methods, if you have Affinity. File size might be a problem.

InDesign: Since 2013 I have tried this and every other set of instructions for converting text to outline from inside InDesign. I do not remember ever getting it to work. Several CreativePro articles suggest this won’t work in some versions of InDesign. Nonetheless, here are the basics:

1. Create a “convert all text to outlines” transparency flattener preset.
2. Place some non-printing transparency on each spread . . . you need to ensure that the transparency you put on the page isn’t just any transparent object, but rather a transparent object that interacts with some text. This can be done easily with master pages.
3. Export to an Acrobat 4 compatible PDF, or printing-and-distilling, or exporting an EPS file. [creativepro.com/converting-text-to-outlines-the-right-way-updated/]

I recommend exporting from InDesign to PDF then converting in Acrobat or Ghostscript.

Word: Using Word’s Save As PDF with PDF/A selected will rasterize OpenType (.otf) fonts into small bitmapped images. You cannot make it bitmap TrueType Fonts (.ttf).

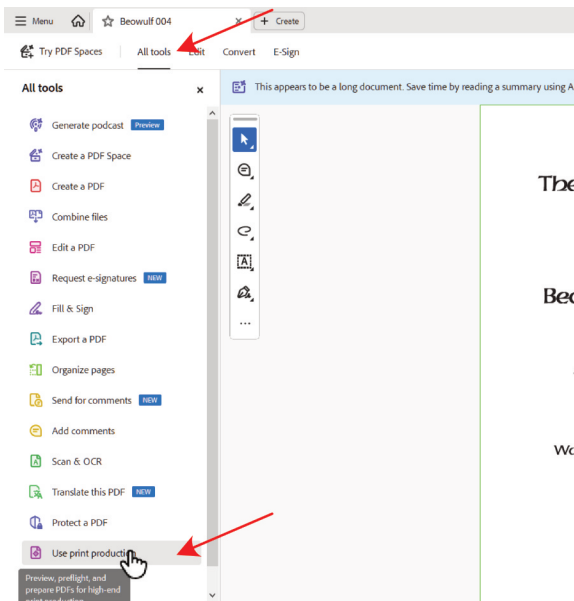
Adobe Acrobat DC (not Reader): has three methods (how well they work is based on my experience and recollection, and may be idiosyncratic).

Each of the following methods starts with a print-ready PDF: that is, a PDF made from your finished—edited and formatted—file. Always work with a copy.

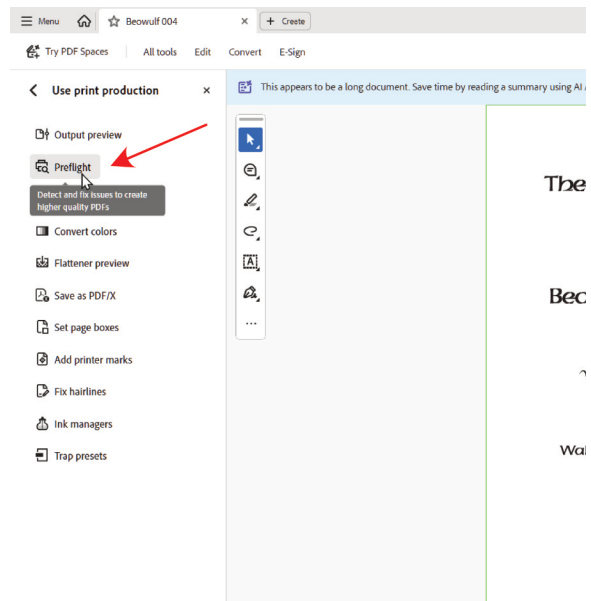
Default Method for Acrobat DC

PDF Fixups—Acrobat DC—when this first came out it was not reliable; however, for several years now, it now works consistently. It is quick and easy. It's also the one I have recommended to most people, because if you already own Acrobat and you don't want to learn to use Command Prompt, it's a good choice. File optimization or compression is a different matter. If you want a more reliable result, if you need compression, and you don't mind pasting some code into the Command Prompt terminal, see Ghostscript (pg 13).

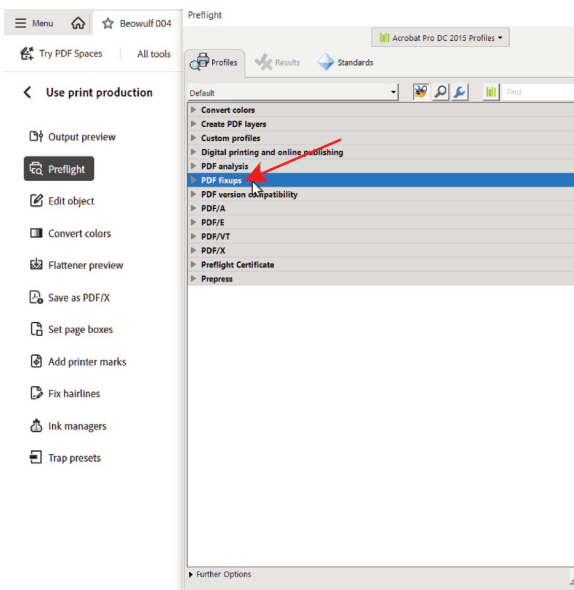
- Open the PDF in Acrobat DC.
- Click on *All Tools* > *Use print production*, then click on *Preflight*
- PDF Fixups* > *Convert fonts to outlines* > *Analyze and fix*.



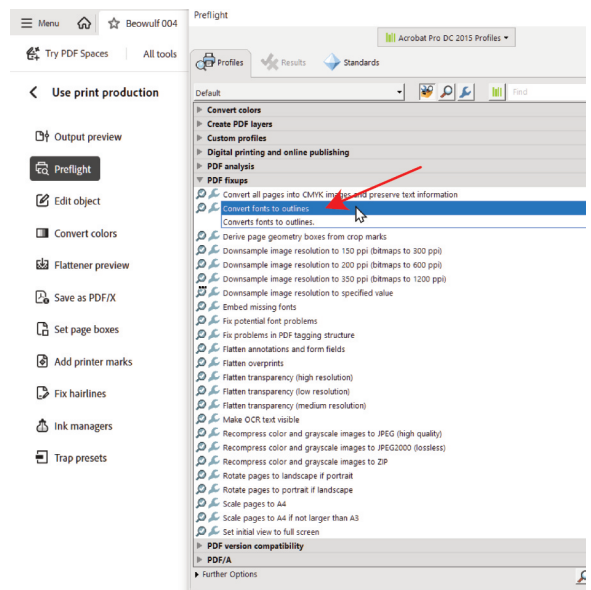
Click on *All Tools* *Use print production*



Click on *Preflight*



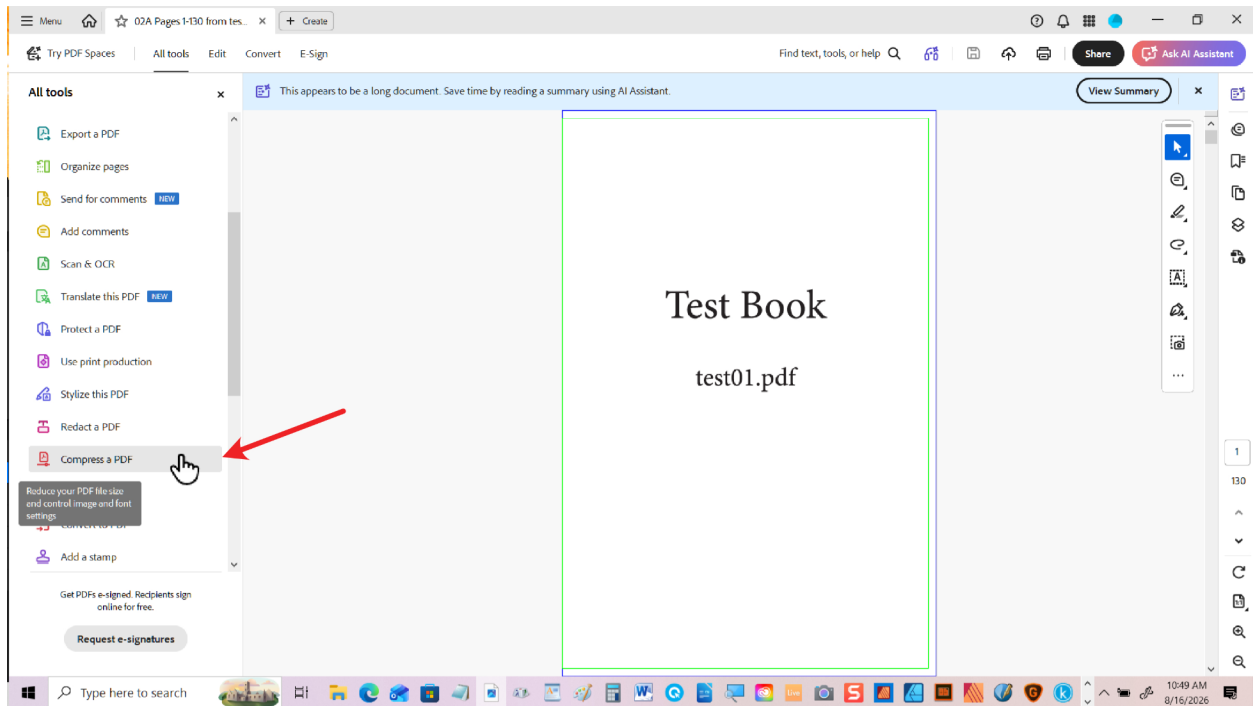
Open *PDF fixups*



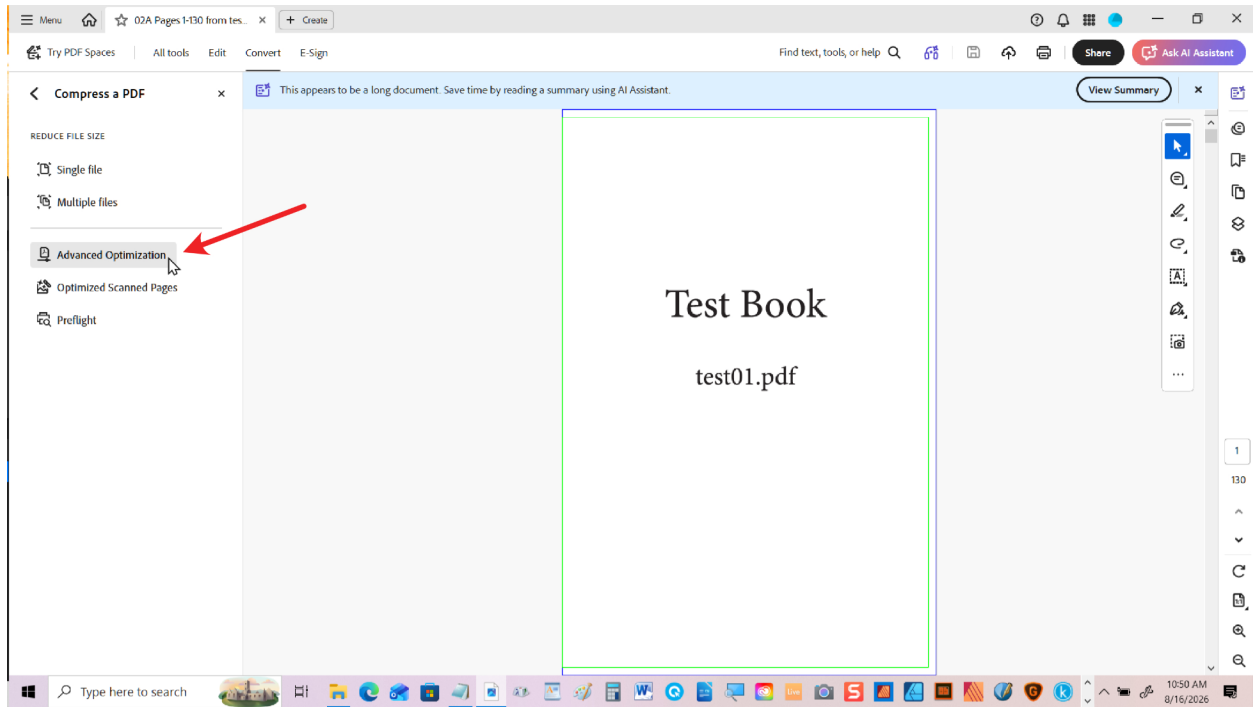
Select *Convert fonts to outlines*; click on *Analyze and fix*

At this point your text has been converted. You can confirm this by checking that there are no fonts embedded in the PDF, see page 26. If the file size is OK, then check your file carefully to see that everything looks correct. Upload this file, if it is correct. If the file is too big, go to step “d.” If that does not reduce the file size sufficiently or Acrobat freezes, try **Ghostscript**. While you can apply Ghostscript to the outlined PDF, it’s better to start with the PDF before you converted the text to outline.

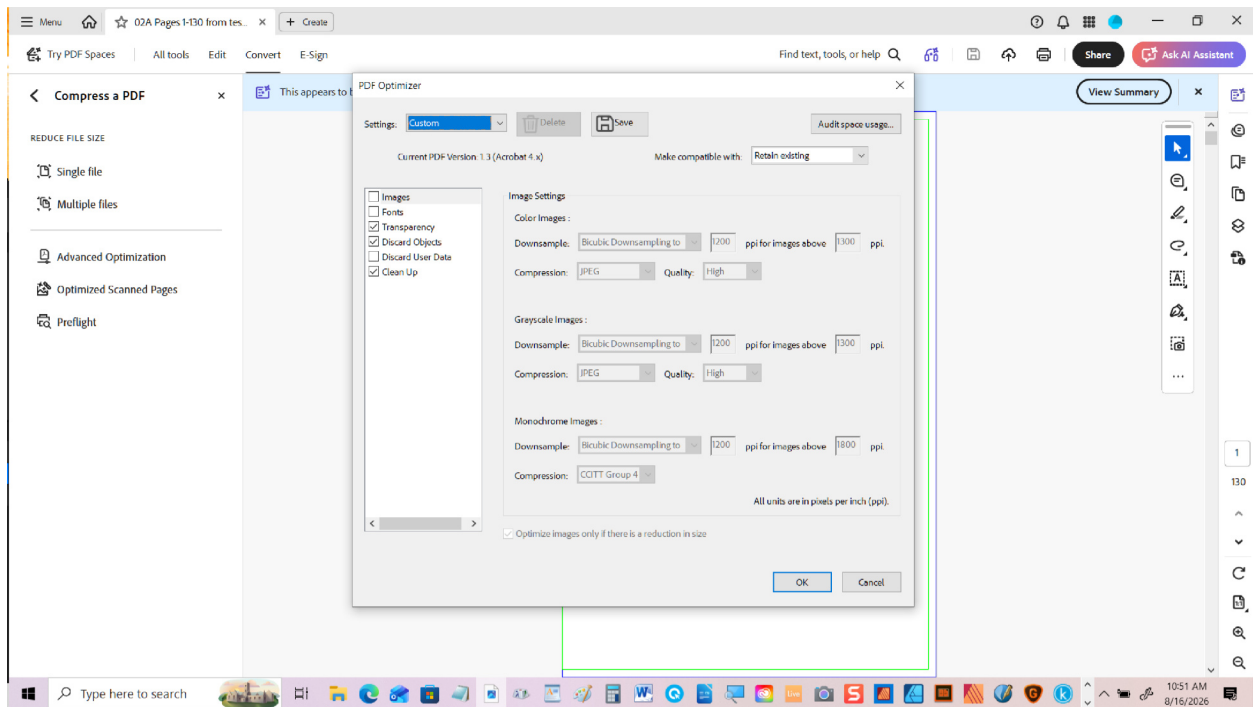
d. *Tools > Compress a PDF > Advanced Optimization > Clean Up > [tick] “Use Flate to encode streams that are not encoded.”*



You might have to click on *Show More Tools*, but look for *Compress a PDF*

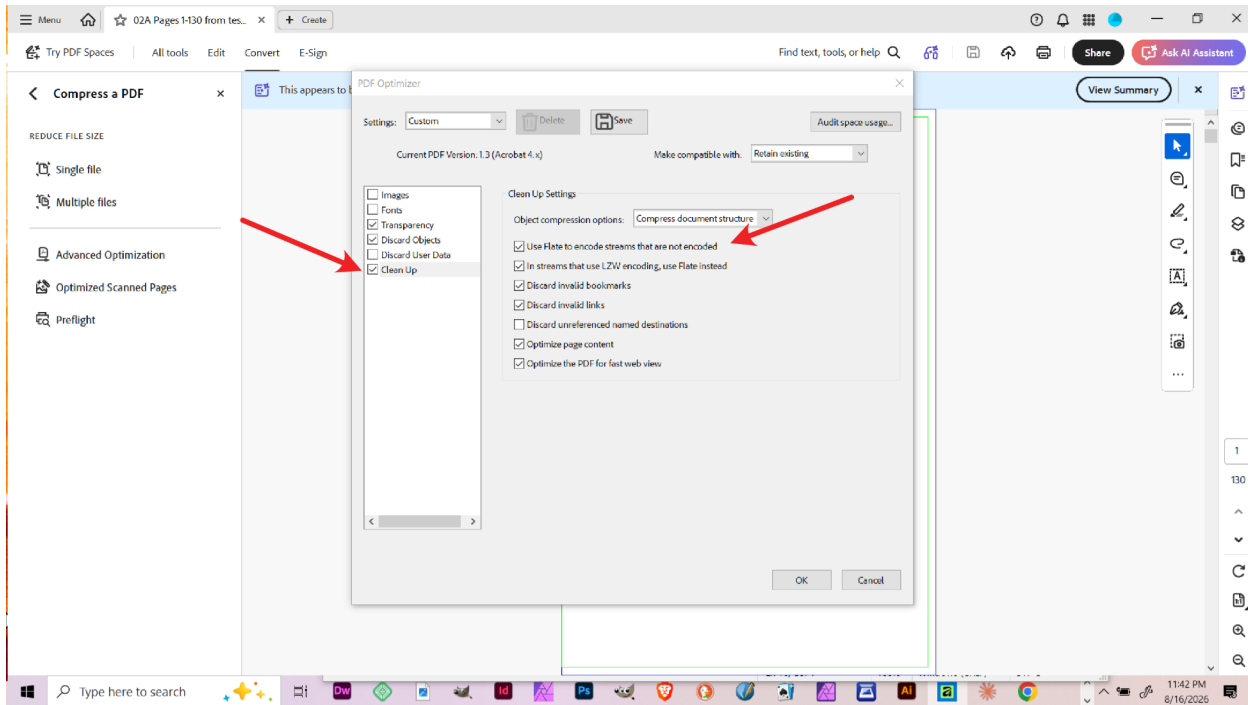


Click on *Advanced Optimization*



This is the *PDF Optimizer* panel. Make sure that *Images* is unticked and that *Clean Up* is ticked.

Acrobat—Compression & Alternate Conversion Methods



Double-click on *Clean Up*, then make sure *Use Flate to encode streams . . .* is ticked.

Alternate Methods in Acrobat

Flattener—by itself, I may have gotten it to work once or twice.

- a. Open the PDF in Acrobat
- b. *Tools > Print Production > Flattener Preview*
 - i. Under Preview Settings, tick *Convert All Text to Outlines*
 - ii. Under Apply to PDF select All pages, Current Page, or Page range (the preview will go blank)
 - iii. Click Apply

Adobe's position is that the flattener needs something to flatten. Hence the next method:

Watermark & Flattener—this works most of the time, and until the last few years, was my preferred method when I had to convert text to outline.

- a. Open the PDF in Acrobat
- b. *Tools > Edit PDF > Watermark > Add*⁶
 - i. Type anything in Text field—even an 8 pt period is okay
 - ii. Select a color (I usually choose the lightest gray)
 - iii. Slide the opacity down to 1%
 - iv. Click OK
- c. *Tools > Print Production > Flattener Preview*
 - i. Under Preview Settings, tick *Convert All Text to Outlines*
 - ii. Under Apply to PDF select All pages, Current Page, or Page range (the preview window will go blank)
 - iii. Click Apply

⁶ This watermark can cause a slightly noticeable dark area if placed over a background. If this could be a problem, test things, and, if possible, move to a print-safe area outside the background.

Ghostscript

I was reluctant to try, let alone recommend, Ghostscript because it looks daunting: installing new software from a somewhat geeky site; using Command Prompt, which harkens back to MS DOS days; and code. But with step-by-step instructions, it is the best method:

Example: In one step Ghostscript can convert text to outline and compress the PDF. I have a client's book: 170,000 words, 372 pages, 36 images. Using Acrobat, the outlined file was 653MB. It had to be broken into smaller sections to be compressed, then recombined into one file. The final PDF was 350MB. With crashes and reboots, it took over four hours. That same book, running Ghostscript was 374MB—slightly larger than Acrobat's 350MB, and it took 7–14 minutes instead of four hours.

Ghostscript is free. It runs on Windows 32- and 64-bit systems and on Macs through Homebrew (free). Linux users install from their package manager (`sudo apt install ghostscript`), not from that download page—the packaged version will be built for your distribution and will update with the rest of your system.

You will have to copy and paste the code from here to something like Notepad, or *any other plain text editor*. You will run Ghostscript in Command Prompt.

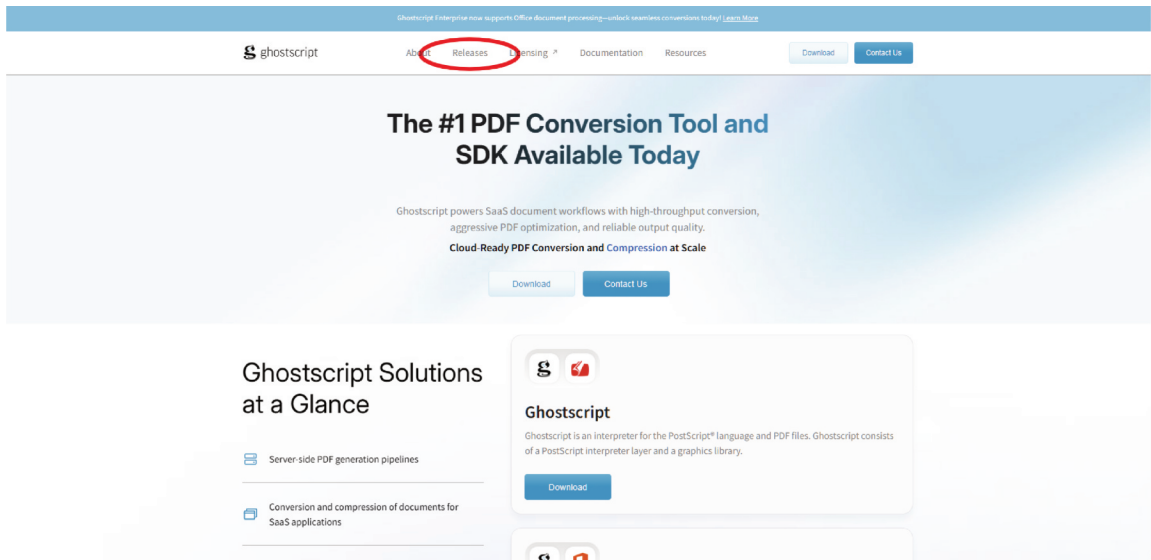
As an aside, if you're nervous about trying this, almost nothing can go wrong. Ghostscript reads your PDF and writes a new one; it doesn't touch the original. If you mistype the code, nothing happens: The one exception I could find: if you give the output file the same name and folder as the input file, Ghostscript will overwrite your print-ready PDF. Send the output to a different folder—which is why I have you make one, and give it a different name.

Downloading & Using Ghostscript

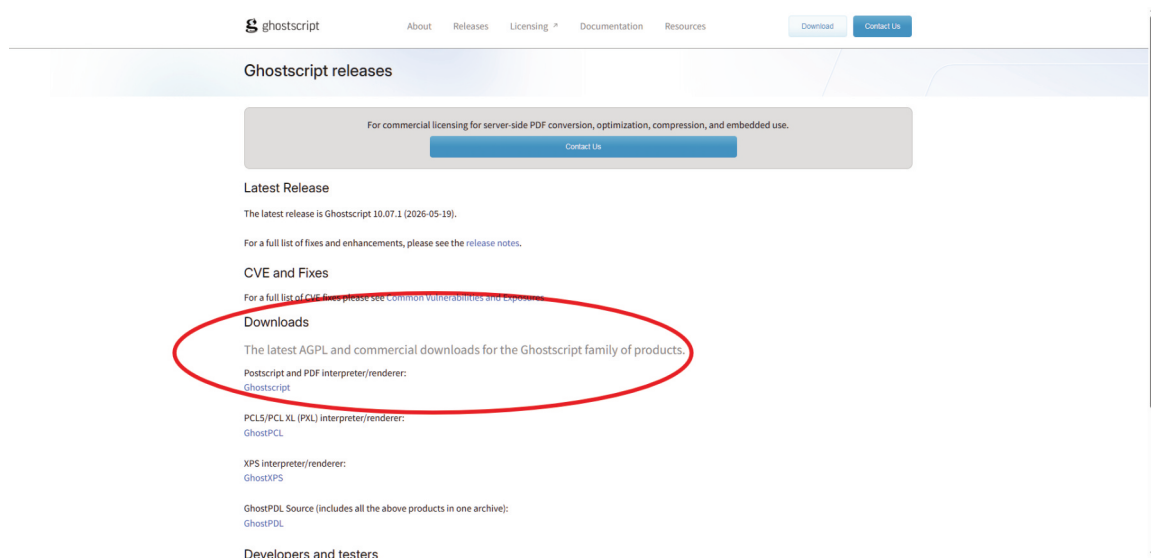
General instructions:

- Download Ghostscript from <https://www.ghostscript.com/> click on Releases. Better, go straight to Releases: <https://www.ghostscript.com/releases/index.html>
- Click on AGPL/Postscript and PDF interpreter/renderer.
- On the download page download Linux if that's your system or, for most people on Windows, Windows 64.
- Like most downloads, select where you want it filed, then go to that folder and double-click the file icon. It will self-install.
- I like to create a target folder where the processed PDF will go. Note exactly the folder name and path.
- Note the name and path for the print-ready PDF that you want to convert.
- From Building the Command, pages 20–23, select the correct sections of code and paste them into a text editing program like Notepad. Follow the instructions.
- Open Command Prompt and paste the code in. Press Enter.

Windows (and Linux)

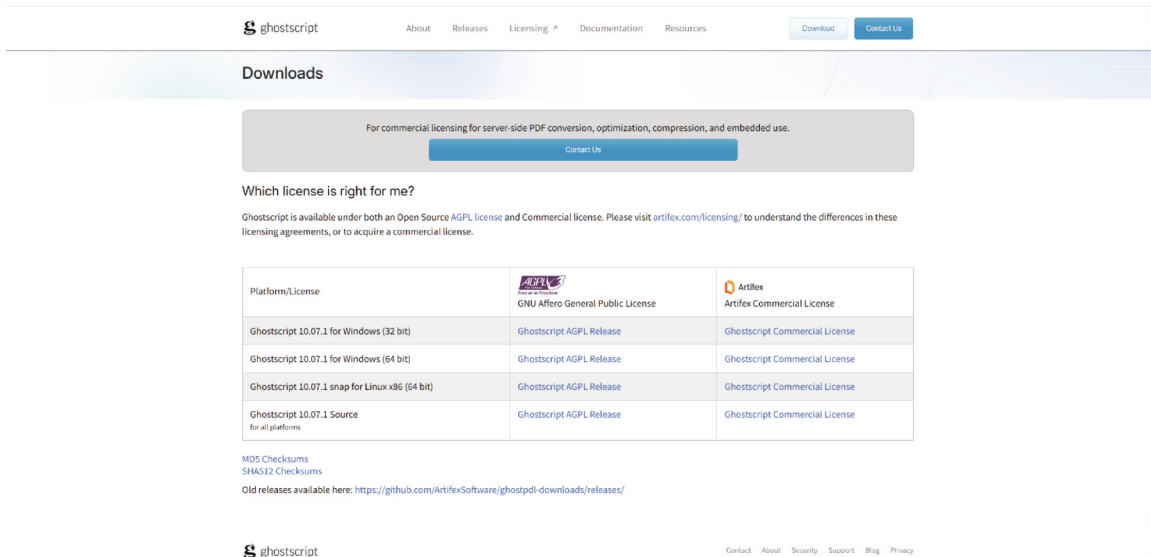


If you go to the home page, you need to click on Releases:



Here is the Releases page:

Currently the actual releases are here <https://www.ghostscript.com/releases/gsdnld.html>:
Download to wherever you can find the file. Then double-click on it and it will self-install.

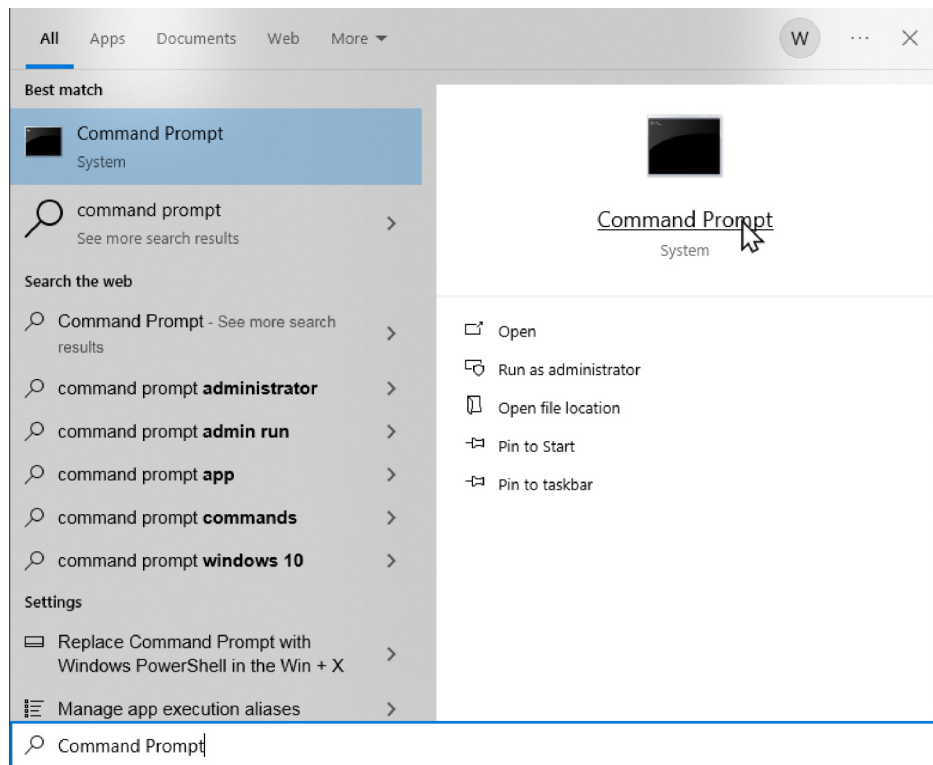


If you double-click on Ghostscript where it is installed, it will bring up this screen:

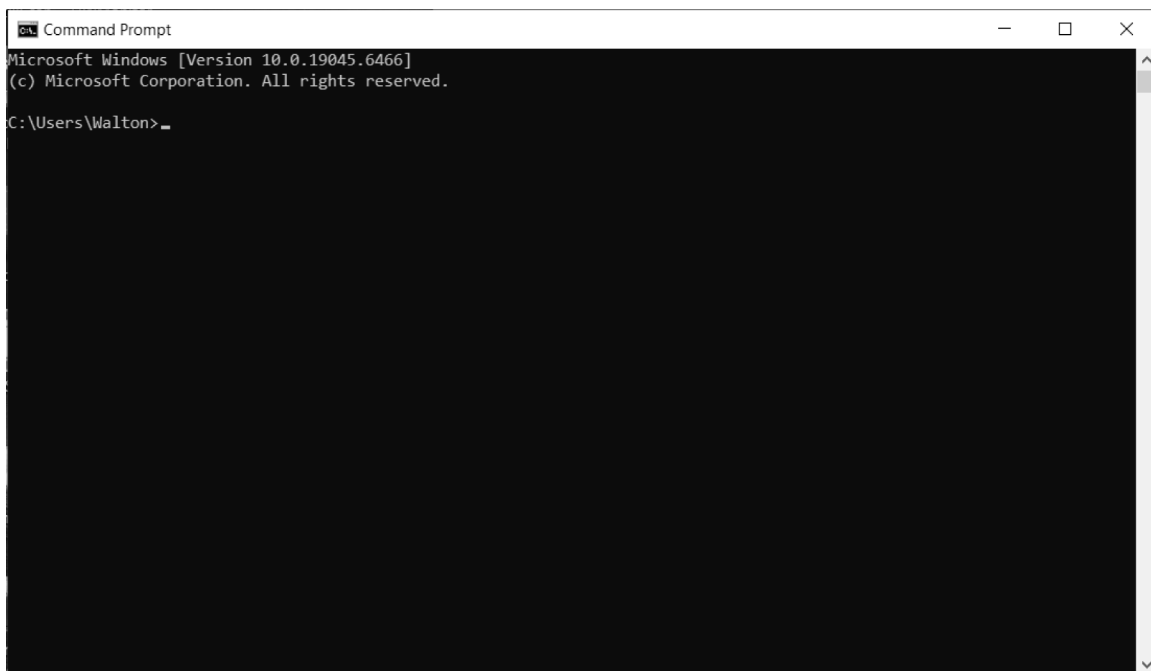


Do not use it. It won't work. This is Ghostscript's own interactive console. It runs Ghostscript commands in a different form than the one in this paper, and the command on page 17 will not work there. Close it and open Command Prompt instead.

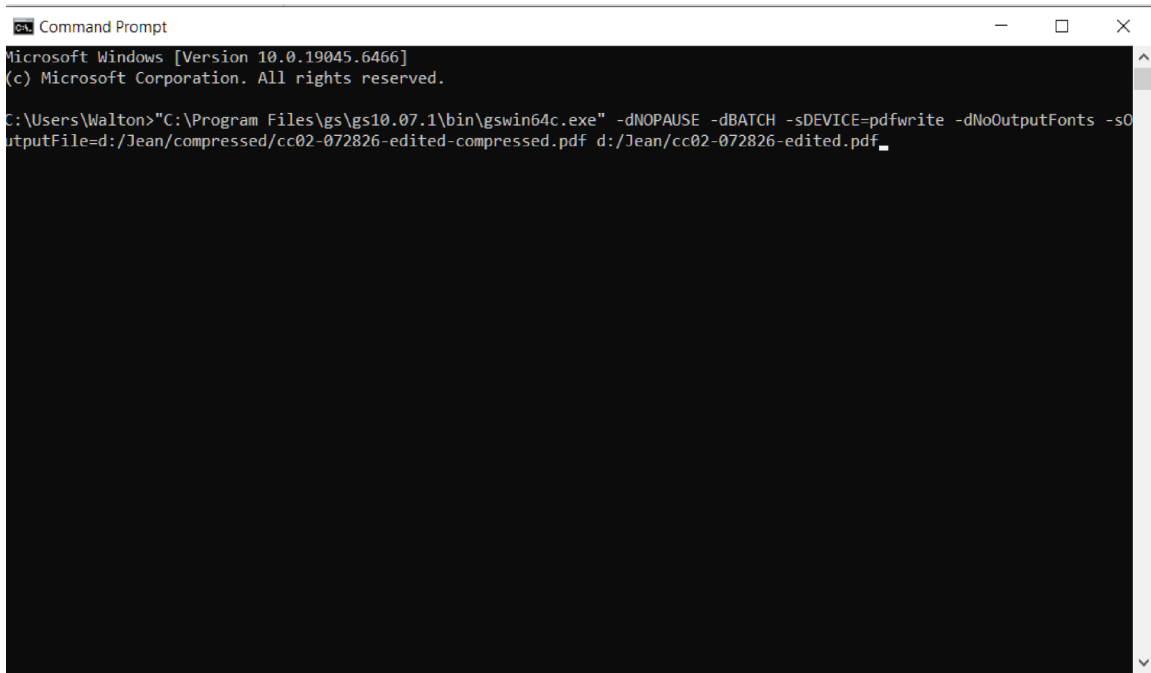
In Windows, type in Command Prompt in the search window:



This will open the Command Prompt:



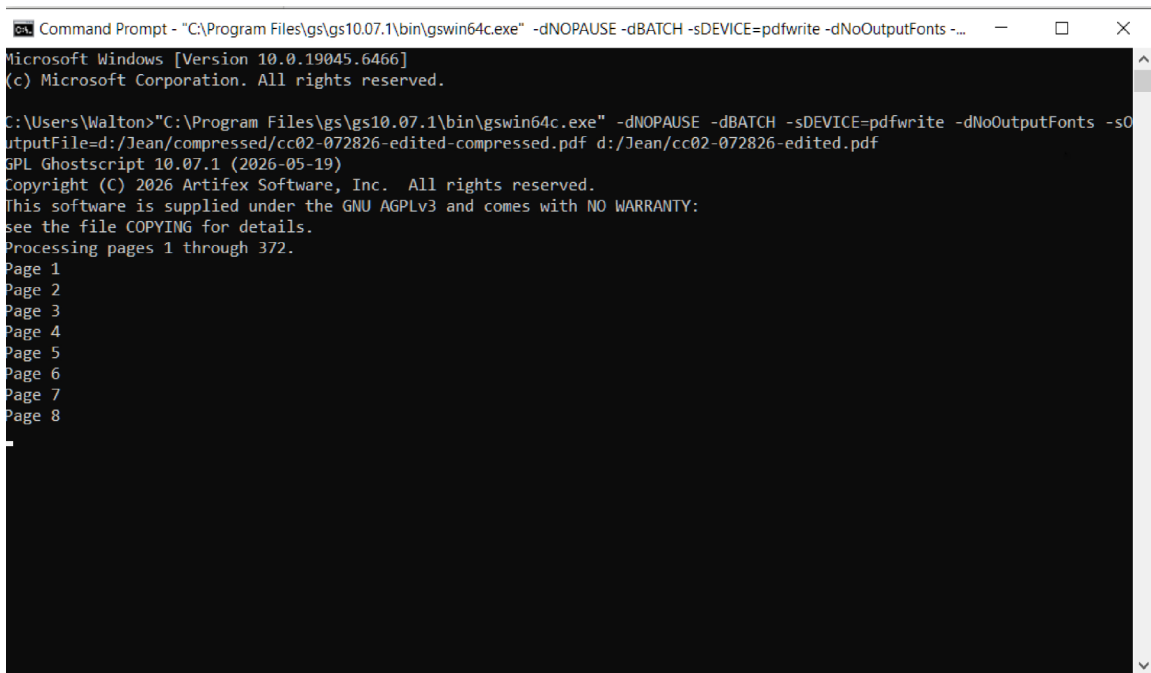
Paste the code into the Command Prompt at the flashing cursor:



```
Command Prompt
Microsoft Windows [Version 10.0.19045.6466]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Walton>"C:\Program Files\gs\gs10.07.1\bin\gswin64c.exe" -dNOPAUSE -dBATCH -sDEVICE=pdfwrite -dNoOutputFonts -sOutputFile=d:/Jean/compressed/cc02-072826-edited-compressed.pdf d:/Jean/cc02-072826-edited.pdf
```

After you paste in the code, press Enter. This will initiate the conversion process, page by page:

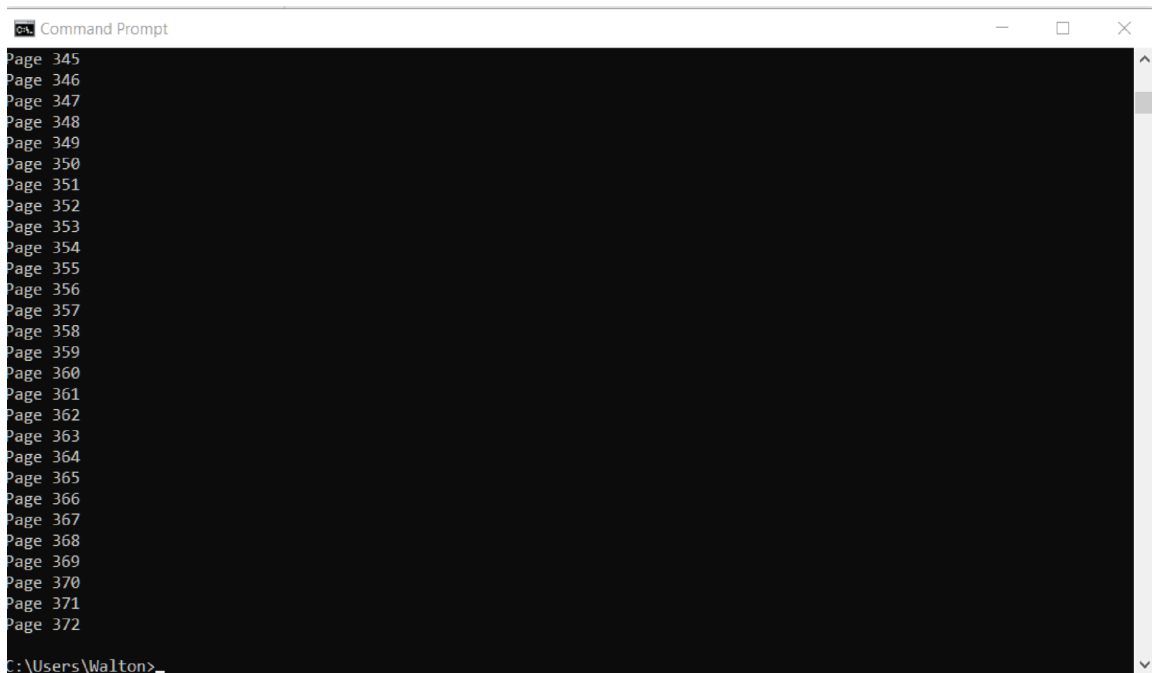


```
Command Prompt - "C:\Program Files\gs\gs10.07.1\bin\gswin64c.exe" -dNOPAUSE -dBATCH -sDEVICE=pdfwrite -dNoOutputFonts -...
Microsoft Windows [Version 10.0.19045.6466]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Walton>"C:\Program Files\gs\gs10.07.1\bin\gswin64c.exe" -dNOPAUSE -dBATCH -sDEVICE=pdfwrite -dNoOutputFonts -sOutputFile=d:/Jean/compressed/cc02-072826-edited-compressed.pdf d:/Jean/cc02-072826-edited.pdf
GPL Ghostscript 10.07.1 (2026-05-19)
Copyright (C) 2026 Artifex Software, Inc. All rights reserved.
This software is supplied under the GNU AGPLv3 and comes with NO WARRANTY:
see the file COPYING for details.
Processing pages 1 through 372.
Page 1
Page 2
Page 3
Page 4
Page 5
Page 6
Page 7
Page 8

```

When the process is done, it will stop. The completed text to outline conversion and file compression PDF will be in the output folder you made.



```
Page 345
Page 346
Page 347
Page 348
Page 349
Page 350
Page 351
Page 352
Page 353
Page 354
Page 355
Page 356
Page 357
Page 358
Page 359
Page 360
Page 361
Page 362
Page 363
Page 364
Page 365
Page 366
Page 367
Page 368
Page 369
Page 370
Page 371
Page 372
C:\Users\Walton>
```

Ghostscript on Mac

Installing Homebrew First

Windows Ghostscript installs from a downloaded installer. Mac has no equivalent — Artifex doesn't ship one; the two routes I know of are Homebrew and MacPorts; these instructions cover Homebrew. Getting Ghostscript on a Mac means installing a separate tool first, Homebrew, which then fetches Ghostscript for you. Two installs, not one.

I have not run this on Mac. These steps are based Homebrew's and Ghostscript's documentation, not from a test. If you run this please let me know how it works.

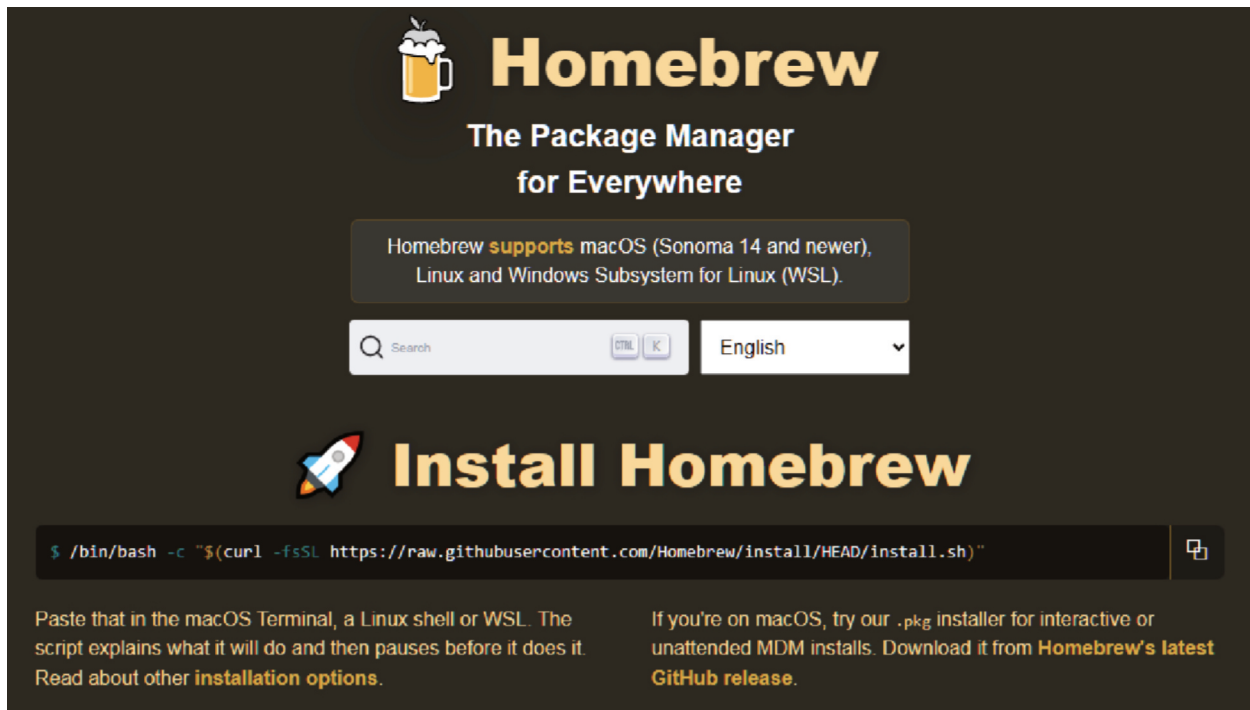
Before You Start

You'll need your Mac's administrator password. Homebrew's installer asks for it partway through.

The Walkthrough

1. Open Terminal. Press Command and Space together to open Spotlight search, type Terminal, press Enter. Terminal is a Mac's equivalent of Command Prompt.
2. Install Homebrew. Go to `brew.sh` in a web browser. On that page is a single line of text starting with `/bin/bash -c`. Copy it exactly, paste it into Terminal, press Enter.

The install line is the block of text under “Install Homebrew”:



3. Follow the prompts. Homebrew explains what it’s about to do and asks you to press Enter to continue, then asks for your Mac password. Typing a password in Terminal shows nothing on screen — not even dots. That’s normal, not a freeze. Type it and press Enter anyway.
4. Wait. This step downloads and installs Homebrew itself. Several minutes, depending on your connection.
5. Check for a final instruction. On some Macs — recent ones with Apple’s own chips, not older Intel Macs — Homebrew finishes by printing a command it wants you to run, something with the words eval and shellenv in it. If you see one, copy it exactly as printed, paste it into Terminal, press Enter. If you don’t see one, skip this step.
6. Close Terminal and open it again. This makes sure Homebrew is fully recognized before you continue.
7. Install Ghostscript. Type `brew install ghostscript` and press Enter. Wait for it to finish — this step downloads Ghostscript itself.
8. Confirm it worked. Type `gs --version` and press Enter. A version number means it’s installed.

On a Mac, the program is just `gs`—not `gswin64c.exe`. Wherever the Windows walkthrough in this paper uses the full quoted Windows path, a Mac reader types `gs` instead. Same switches, same everything else.

If Something Goes Wrong

“command not found: brew” after installing. Step 5 was needed and got skipped, or Terminal wasn’t closed and reopened. Go back to step 5.

Terminal seems frozen after you type your password. It isn't. Terminal never shows anything while you type a password, including dots. Press Enter and continue.

Building the Command

Converting text to outline is not a problem. Although the documentation implies that images are safe, I added code that specifically instructs Ghostscript to leave JPGs alone. I wanted to be sure, though, so I built a test book and checked for myself. Nine of the ten test images—JPG, PNG, TIFF, and every color mode: sRGB, CMYK, grayscale—came through this command byte-for-byte unchanged.⁴ The one exception was the indexed-color image, an old GIF-style palette format that's rare in book work.⁵ If your manuscript has indexed-color art, this is the one open question left—everything else here is confirmed.

Every Ghostscript command has four parts, *always* in the same order:

1. **The program**—where Ghostscript lives on your machine.
2. **The switches**—what you want it to do. For text-to-outline plus compression, this never changes.
3. **The output file**—where the finished PDF goes, and what you want to call it.
4. **The input file**—your print-ready PDF, wherever it currently sits.

Put one space between parts 1 and 2, and one space between parts 3 and 4. There is no space between parts 2 and 3: the switch list ends in `-sOutputFile=` and the output path is typed directly onto the end of it, with nothing in between. A space there will cause the command to fail.

1: The program

Windows (64-bit)—most people: `"C:\Program Files\gs\gs10.07.1\bin\gswin64c.exe"`

Note: The quotation marks are part of the code, do not remove them. They must be straight marks ("), not curly ones. If you retype the path by hand, your word processor may substitute curly quotes; assemble the command in Notepad, never in Word.

Windows (32-bit)—rare on a modern PC: `"C:\Program Files\gs\gs10.07.1\bin\gswin32c.exe"`

Note: The quotation marks are part of the code, do not remove them. They must be straight marks ("), not curly ones. If you retype the path by hand, your word processor may substitute curly quotes; assemble the command in Notepad, never in Word.

⁴Confirmed with a byte-for-byte hash comparison (SHA-256), original vs. Ghostscript output, run against a 90,000-word test book. The original PDF actually contained seventeen embedded image objects, not ten—InDesign's PDF/X-3 export split the GIF into sections and duplicated an image before Ghostscript ever saw the file. Eleven of those seventeen are JPEG/JPX; all eleven came out identical. PDF/X-3 was used because it lets RGB and CMYK coexist in one file (Affinity, tried as an alternative, tags one color profile to the whole file); that choice hasn't been tested against a plain, non-PDF/X-3 export.

⁵Comparing the original indexed-color image (a GIF) to the same image in the processed book, both look the same to the eye. Measured directly, though, the processed version's color palette has one extra color, and its file size is very slightly different due to the way the GIF was processed when the file was exported to PDF (see note 4, this page)—a real difference, but not a visible one. Why it happens, I don't know.

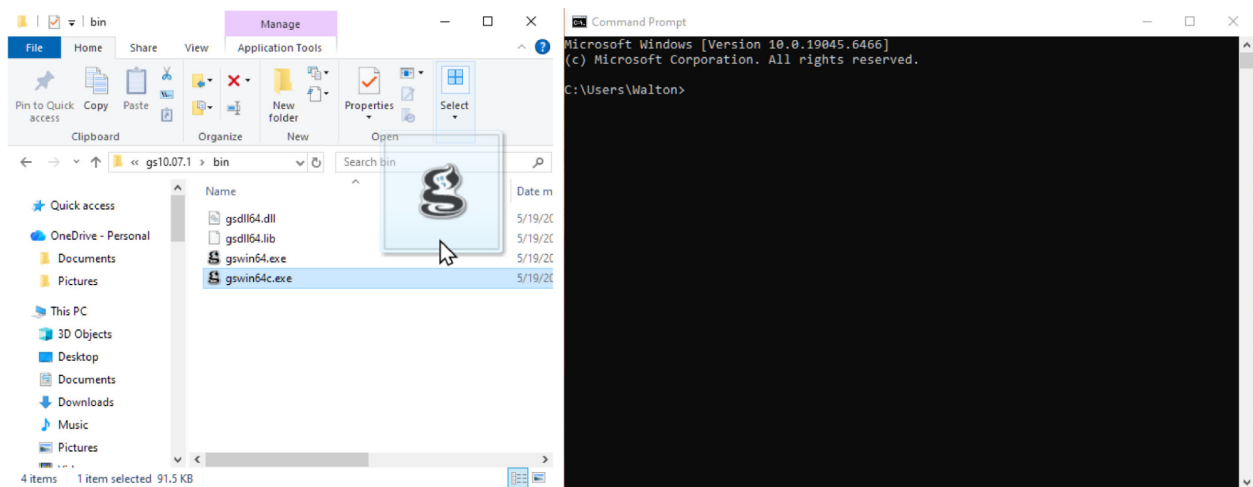
Mac (once installed through Homebrew, p. 18–19): `gs`

Linux (once installed through your package manager): `gs`

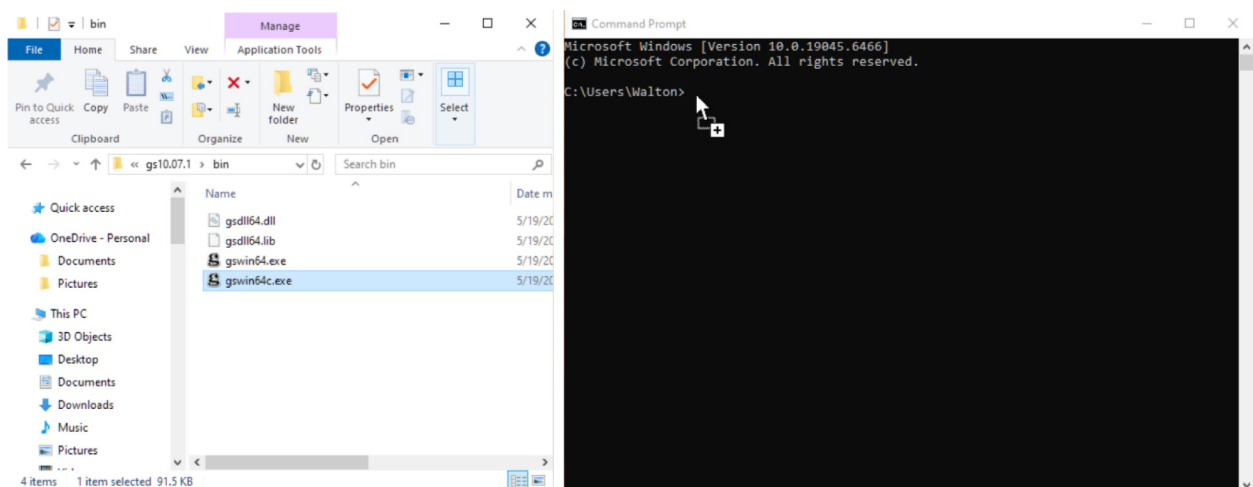
Windows is the only platform where this part changes machine to machine—the version number (10.07.1 above) is whatever you installed, and typing it wrong doesn't give you a clear error, just “the system cannot find the path specified.” Two ways to get it right:

- **Drag it in, don't type it.** [Walkthrough below]: locate `gswin64c.exe` in File Explorer and drag it into Command Prompt. Windows fills in the full, correctly quoted path for you—no version number to remember or mistype.
- **Look it up,** if you'd rather type it by hand. Open Command Prompt and run: `dir "C:\Program Files\gs" /b` This lists the folder name Ghostscript installed under. It will return something like `gs10.07.1`. Use whatever it returns in place of `gs10.07.1` in the path above—the whole folder name, not just the number.

Click and drag the icon:

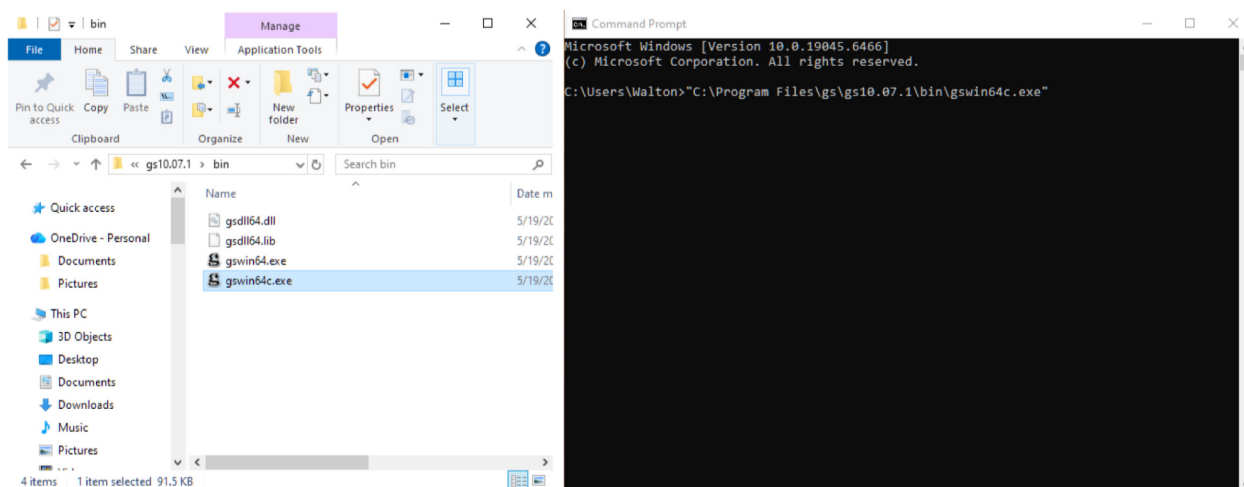


Drag the icon into the Command Prompt terminal: note the cursor changed:

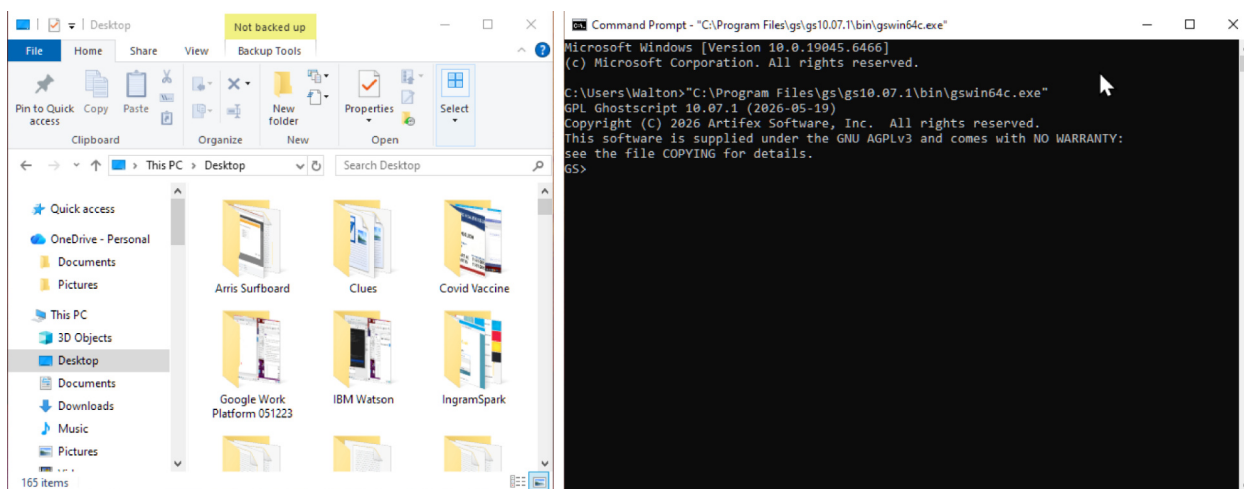


Ghostscript—Building the Command

When the icon is released, the path to GS is typed in:



Press Enter (above) and this shows the full recognition by Command Prompt of GS. If you press Enter here by mistake, you get the `gs` prompt form shown on page 15. Type `quit` and press Enter, or close the window and open Command Prompt again.



Mac and Linux never have this problem. Once installed, `gs` just runs—no version number involved, ever. This is documented Homebrew and Linux-package-manager behavior; I have not confirmed it on an actual Mac or Linux machine.

2: the switches (fixed)

```
-dNOPAUSE -dBATCH -sDEVICE=pdfwrite -dNoOutputFonts -dPassThroughJPEGImages=true -dPassThroughJPXImages=true -sOutputFile=
```

Copy this exactly. It's identical on Windows, Mac, and Linux.

The two `PassThrough` switches keep JPEG and JPEG2000 images out of Ghostscript's recompression path—confirmed above by hash comparison to come out byte-identical to the original.

3 & 4: your files

- **Output**—the folder you made for the finished PDF (p. 13), plus whatever you want the new file called.
- **Input**—your print-ready PDF, wherever it currently sits.

Two rules that will save you an afternoon:

- **Keep names and paths simple.** Letters, numbers, hyphens, underscores. No spaces, no parentheses, no punctuation beyond that. `MyBook-outlined.pdf`, not `My Book (final!).pdf`.
- **Forward slash for your files, backslash only inside the program's own path.** Windows normally uses backslashes (`\`) everywhere, but Ghostscript wants forward slashes (`/`) in the input and output paths you type. Only the quoted program path in Part 1 uses backslashes. Mixing the two up is one of the more common ways this fails.

Putting it together

Paste all four parts into Notepad. Then copy the whole line, paste it into Command Prompt in one shot, and press Enter. Don't type it directly into Command Prompt.

Here is what a finished command looks like. **Do not copy these verbatim**—the paths shown are mine, not yours. Substitute your own output folder in Part 3 and your own print-ready PDF in Part 4.

Example (Windows): `"C:\Program Files\gs\gs10.07.1\bin\gswin64c.exe" -dNOPAUSE -dBATCH -sDEVICE=pdfwrite -dNoOutputFonts -dPassThroughJPEGImages=true -dPassThroughJPXImages=true -sOutputFile=d:/MyBook/outlined/MyBook-outlined.pdf d:/MyBook/MyBook.pdf`

Example (Mac): `gs -dNOPAUSE -dBATCH -sDEVICE=pdfwrite -dNoOutputFonts -dPassThroughJPEGImages=true -dPassThroughJPXImages=true -sOutputFile=/Users/YourName/MyBook/outlined/MyBook-outlined.pdf /Users/YourName/MyBook/MyBook.pdf`

Example (Linux): `gs -dNOPAUSE -dBATCH -sDEVICE=pdfwrite -dNoOutputFonts -dPassThroughJPEGImages=true -dPassThroughJPXImages=true -sOutputFile=/home/yourname/MyBook/outlined/MyBook-outlined.pdf /home/yourname/MyBook/MyBook.pdf`

Confirming Ghostscript is installed

Windows: `dir "C:\Program Files\gs" /b` A folder name coming back means it's installed, and gives you your version number in the same step.

Mac/Linux: `gs --version` A version number means it's installed.

Running it

Once you press Enter, if you don't get an error message, the screen will list page numbers as Ghostscript works through the file. When it stops, the finished PDF is in the output folder you made.

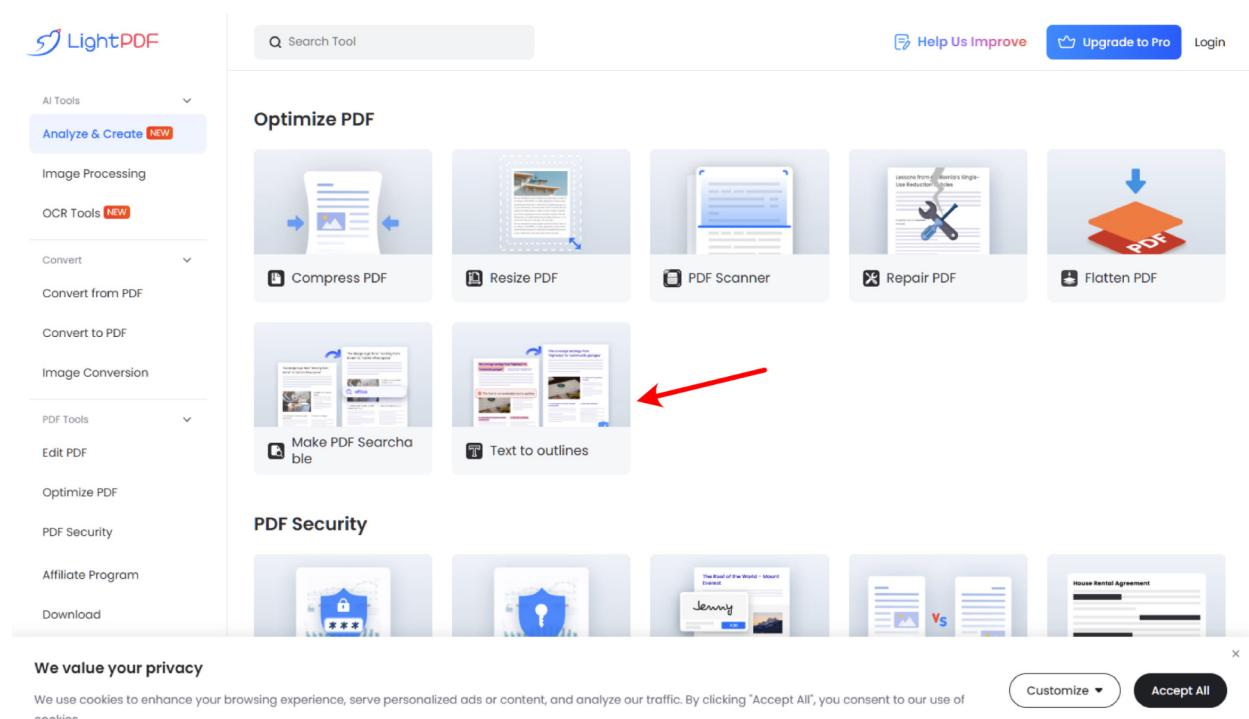
Confirm it worked. Open the output PDF in Acrobat and check *Menu > Document Properties > Fonts*. A successful outline leaves that list empty—nothing left to embed or substitute.

LightPDF—Alternative to Acrobat

LightPDF, <https://lightpdf.com>, can convert text to outline. If I read it correctly, you can convert up to three 10MB files for free for three days. As of May, 2026, there is a \$19/month plan. You can use it online or download the program. You can use LightPDF to merge PDFs. It can create PDF/A⁷ files but not PDF/X. It will not convert to grayscale, nor remove transparency.

I converted one file using LightPDF. It worked well. Files within a 10–30MB size can be merged.

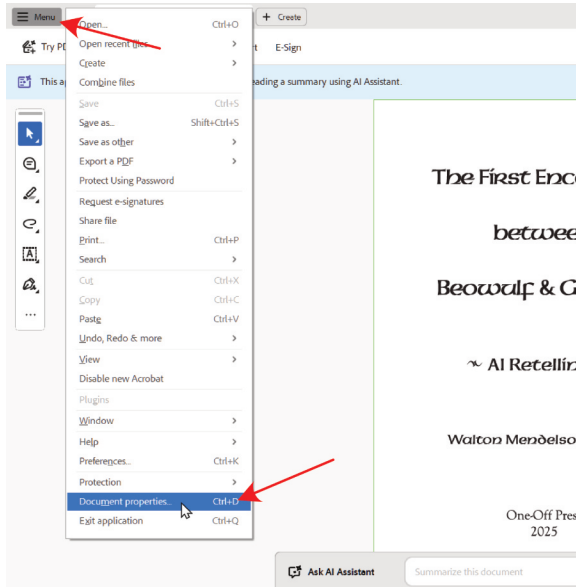
If you use the online conversion feature keep in mind that you would be uploading your file to someone else's server. If that's a concern use the download version or one of the offline methods described earlier.



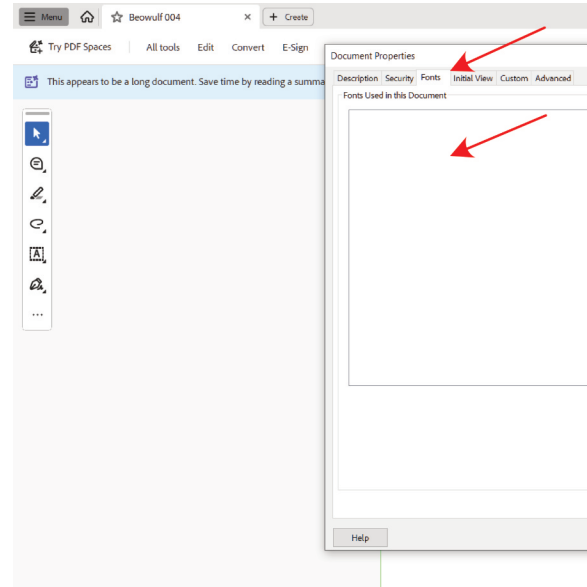
⁷I do not know if this is PDF/A-1, A-2, or A3. PDF/A-1 does not permit transparency; whereas A-2 and A-3 do.

Check Fonts

Carefully double-check the PDF *before* uploading. Look first at *Menu > Document Properties > Fonts*. No fonts should be listed.



Click on *Menu > Document Properties*



Click on *Fonts* tab: the Fonts box should be empty

I recommend scrolling through the entire file to confirm that everything converted correctly. If you discover problems, you will not be able to fix them in the PDF itself. Return to the source file (Word, Affinity, InDesign, etc.), make corrections there, export to PDF, and repeat the outline conversion.