

[Back](#)

Export

April 20, 2021 by typora.io

- [General](#)

- [Quick Usage of Export](#)
- [Export with Previous Settings](#)
- [Open Export Settings](#)
- [Export Location](#)
- [Export Using Pandoc](#)
- [Config “After Export” Options](#)
- [Per-file Configurations](#)
- [Config Export Items](#)
 - [Reorder Export Items](#)
 - [Rename Export Item](#)
 - [Remove Export Item](#)
 - [Add New Export Item](#)

- [File Format](#)

- [HTML](#)
 - [Outline](#)
 - [Add Custom Contents](#)
 - [Export YAML meta](#)
 - [Append Contents per Document.](#)
 - [Change Theme](#)
 - [Example: Export using VLOOK](#)
 - [More HTML Options](#)
- [HTML \(without styles\)](#)
- [Image](#)
 - [Width and Font Size](#)
 - [Theme](#)
- [PDF](#)
 - [Paper Size](#)



- [Theme](#)
- [Page Break](#)
- [Header / Footer](#)
- [PDF Metadata](#)
- [Add Custom Content \(PDF\)](#)
 - [Example: Add Cover Page](#)
- [PDF \(LaTeX / Pandoc\)](#)
 - [PDF Engine](#)
 - [Template](#)
 - [Meta Data \(Other Options\)](#)
 - [Extra Arguments](#)
- [Word \(.docx\)](#)
 - [Style Reference](#)
- [OpenOffice \(.odt\)](#)
 - [Template](#)
- [RTF](#)
- [Epub](#)
 - [Custom CSS](#)
 - [Chapter Level](#)
 - [Cover Image](#)
 - [Meta Data](#)
- [LaTeX](#)
 - [Meta Data \(Other Options\)](#)
 - [Add Extra Content in LaTeX Header](#)
- [Wiki](#)
 - [Media Wiki](#)
 - [Other Wiki Format](#)
- [reStructuredText \(RST, ReST, or reST\)](#)
- [Textile](#)
- [OPML](#)
- [Other Markdown Spec](#)
- [Other Formats Using Pandoc](#)
 - [Target Format](#)
 - [Target File Extension](#)
 - [Raw Type](#)
 - [Extra Arguments](#)
 - [Get Exported AST for Pandoc](#)
- [Other Formats Using Custom Commands](#)
 - [Command](#)
 - [Show Save File Dialog](#)
 - [Target File Extensions](#)



- [Show Command Output after Export](#)
- [Other Pandoc Options](#)
 - [Append Extra Arguments](#)
- [Appendix](#)
 - [Install Pandoc](#)
 - [Variables](#)
 - [Unset Variables](#)
 - [Use Variables in Export Config](#)
 - [Trouble Shooting](#)
 - [Require Pandoc to Continue...](#)
 - [Export PDF not working on Windows](#)
 - [“Cannot Read Native” when export using pandoc](#)
 - [Other Pandoc Errors](#)
 - [Continues whitespace or line break are gone in exported PDF](#)
 - [Env: node: No such file or directory](#)

Note following documents are for Typora ≥ 0.10.0

General

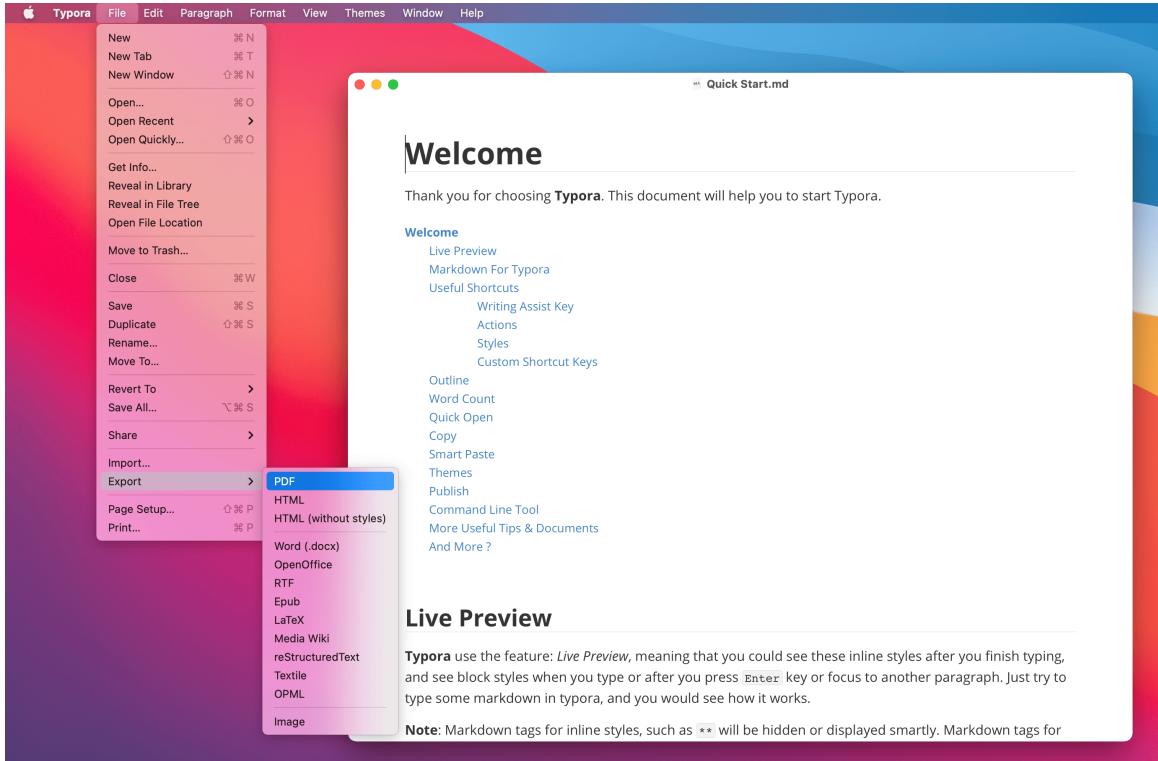
Quick Usage of Export

Typora supports you to export current document into PDF, HTML, HTML (without styles) and Image format.

More formats (like Word, RTF, etc) are supported via [pandoc](#), after you [install pandoc](#), those format are supported out-of-box.

You could access export command from **File → Export** command from menubar:





Export with Previous Settings

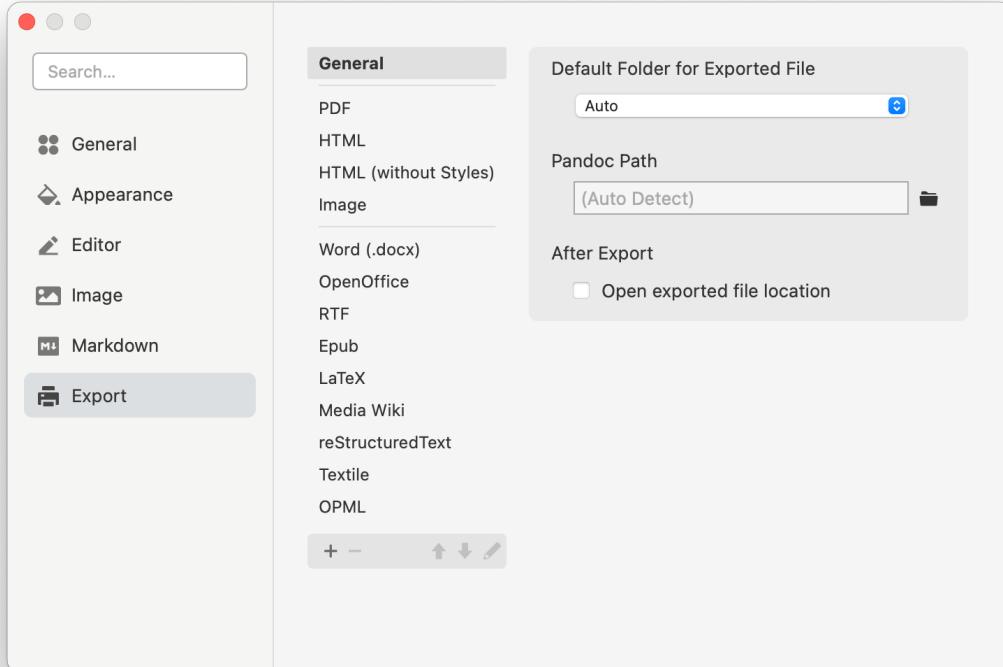
You can export current file again using previous successful export command by clicking **File → Export → Export with Previous** menu item.

If you want to export and overwrite previous exported file using previous command and newer file content, you can click **File → Export → Export and Overwrite with Previous**. One use case is that, you may export current markdown file into PDF file, but then you find some content is missing or some words are misspelled, after you correct the markdown file, instead of click PDF export and choose exported path again, you can just click **Export and Overwrite with Previous** so the newer content will be written to your previous exported PDF file. Please be careful since it will overwrite your previous exported file.

All “previous settings” prefer to previous export executed in current window and current document.

Open Export Settings

You could config export options in preferences panel:



Export Location

You could change Typora's default export location in `Export → General → Default Folder for Exported File`. Possible options are "auto", "same folder with current folder" and "custom location". This setting changes the default value, and when export, you can still change export file path in the save file dialog.

Export Using Pandoc

For formats other than HTML, PDF and Images, typora use [pandoc](#) for export. Pandoc is also used for importing non-markdown files, like word, rtf, etc.

So, it make them work, please [install pandoc](#) first. Quickest way is to follow their official instructions <https://pandoc.org/installing.html>.

After installed, please restart Typora. And if typora still says "Require Pandoc to Continue...", please open `Export → General` from preferences panel, and manually config pandoc path.

Config "After Export" Options

By default, when export succeed, typora will show a notification inside typora window. When fail, an error dialog with error message would pop up.

If you want to open folder location of exported file after export, you can check "Open exported file location" in preferences panel. Settings in specific export commands will override setting in "General" section for export.

If you want to open exported file automatically, you can check “**Open exported file**” for specific export commands in preferences panel.

Per-file Configurations

For supported configurations, you can define them in YAML front matters, please see [YAML front matters](#) for details.

Since the markdown file may be created by others, for security reasons, native supported formats (PDF / HTML) requires “Read and overwrite export settings from YAML front matters” to be enabled in export preferences panels, then settings in YAML front matters can be used and overwrite global export settings.

Config Export Items

Under **Export** tab in preferences panel, typora allows you to customize your export items, including re-order, re-name export items, add and remove export items based on your needs. The changes will be applied to menu items under **File** → **Export** menu.

The built-in options (PDF, HTML, HTML (without Styles), Image) are not allowed to reorder, rename or remove, for other options, you can do:

Reorder Export Items

You can select one item In list panel of Preferences panel → **Export**, then click on the **↑** or **↓** button to change orders.

Rename Export Item

You can select one item In list panel of Preferences panel → **Export**, then click on the “Edit” button to rename one export item.

Remove Export Item

You can select one item In list panel of Preferences panel → **Export**, then click on the “delete (-)” button to remove it from **File** → **Export** menu.

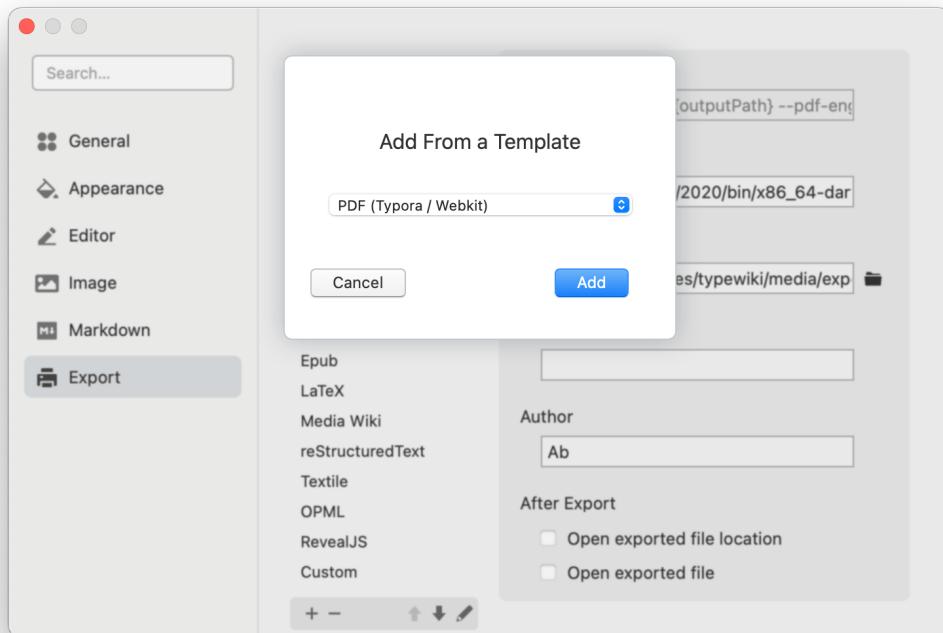
Add New Export Item

You could add export items based on Typora’s own exporter, pandoc, or custom command to extend Typora’s export function or just add preset export commands into **export** menu. You could do this by:

1. Click the “add (+)” button in the list panel



2. Select a template from the dialog:



3. Set a name and config its details.

Then you can use this from **File → Export** menu.

You could also assign it a shortcut key, following

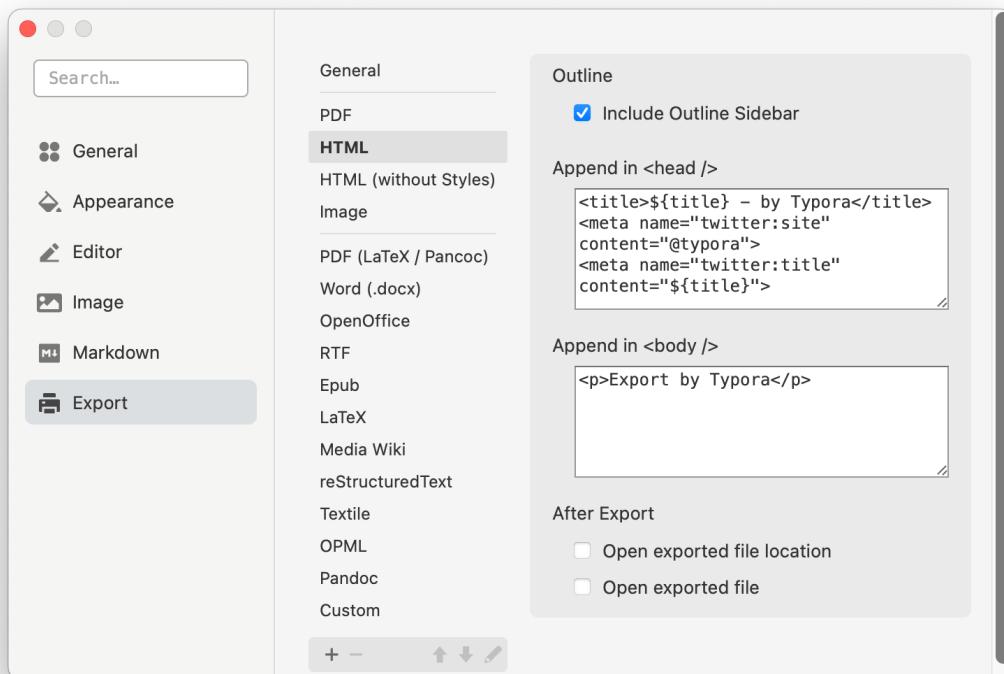
<https://support.typora.io/Shortcut-Keys/#change-shortcut-keys>.

File Format

HTML

You can select **Export → HTML** from the preferences panel to config export options for HTML file.





If you want to export a HTML without styles, you should choose `Export → HTML (without styles)`, you will get a clean HTML file so you can add your own styles.

Outline

You can include outline in exported HTML by enable this option in `Export → HTML` from preferences panel. Whether it is collapsible outline or flat outline depends on your current setting. You can switch them under `View` menu or under `appearance` section in preferences panel. The css theme you use may affect the outline behavior in exported file, so please disable this option if the outline in exported file do not work well in some 3rd party themes.

Add Custom Contents

You can add custom contents in `Append in <head />` and `Append in <body />`, which will be inserted into `<head />` part and `<body />` part in the exported html. For contents that cannot be added into `<head />`, even when it is added in `Append in <head />`, it will be prepend into `<body />` instead. Same for `Append in <body />`.

Export YAML meta

Variables like `${title}` or `${author}` in `<title>` and `<meta>` tag will be replaced by corresponding variables defined in YAML Front Matter.

For example, you can insert additional `<meta>` data or custom `<style>` into exported html using those options, e.g:

```
<meta charset="UTF-8">
<meta name="author" content="${author}">
<meta name="description" content="${description}">
```

To use those variables in somewhere else, you can add some javascript in `Append in <body />`, to read target meta tag, and modified export content as you want.

Append Contents per Document.

You could use `append-head` and `append-body` in [YAML Front Matter](#) to overwrite global settings, and / or add `append-head-extra` and `append-body-extra` to append extra content besides contents configured in the preferences panel. Those settings requires “Read and overwrite export settings from YAML front matters” option to be enabled for current HTML export item.

Change Theme

You can apply another theme in HTML export in the preferences panel. By default, current theme will be used.

Example: Export using [VLOOK](#)

[VLOOK™](#) is a **Theme Pack** and provide much enhancement for HTML export of Typora.

By using it, you could be able to export more fancy HTML file. Followings are setup process based on current export functionalities:

1. Adding a new export command for HTML format, named as “vlook”.
2. Install VLook themes under
<https://github.com/MadMaxChow/VLOOK/tree/master/released/theme>.
3. Select a VLook theme in Export → vlook → Theme.
4. Copy content from
<https://github.com/MadMaxChow/VLOOK/blob/master/released/plugin/plugin.txt>
to `Append in <body />` part.
5. Make file based on template
(<https://github.com/MadMaxChow/VLOOK/blob/master/released/VLOOK-Document-Template.md>)
6. Click Export → vlook from menu bar to export the content.

Similarly, you could config your own export styles by changing all those configs.

More HTML Options

You could use more options, like setting up html templates, export to slides, etc by adding [new export command based on Pandoc](#).

HTML (without styles)

You could export plain HTML (without styles) using export command `Export → HTML (without styles)` from menu bar.

You could add your own css styles, contents, or javascript by configuring `Append in <head />` and `Append in <body />` in preference panel. The settings are same with [add custom contents in HTML](#).

Image

Width and Font Size

Typora provides image export option for feeds in social media, so by default is 640px width with 24px font size. But you can change this in preferences panel.

Also, based on the purpose of this feature, overall performance and the final size of output images, if the output article is too long, the image export would fail.

Theme

You could choose another theme applied for exported images in preferences panel → Export → Image. By default, current theme will be used.

PDF

You can select `Export → PDF` from the preferences panel to config export options for PDF file, it provide options like:

Paper Size

Set paper size (A4, A5, etc) or custom size, and page margins.

Note: on Windows/Linux

```
@media print {  
    @page {  
        margin: 27mm 16mm 27mm 16mm;  
    }  
}
```



Will be respected when “Margin” is set to default (But please unset margin for `#write` or `body` which may also affect the final output), on macOS, they are ignored.

Theme

You could choose another theme applied for exported PDF in preferences panel → Export → PDF. By default, current theme will be used.

Please note that on Windows / Linux version, **only light themes are supported** for PDF Export, as well as for print.

If you only want to make small changes, **there's no need to specify new themes for PDF export**, you can simply add `@media print { /* your css code */}` in [Custom CSS](#).

Page Break

When “*Page Break Between Top Headings*” is enabled, page break will be inserted between sections divided by top level headings (`h1`), also, page break will be inserted before footnote definitions.

Note: You can input `<div style="page-break-after:always" />` in your document to manually add a page break at that position.

Header / Footer

You can add custom header and footer exported PDF. For example, you can set `${title} - ${author}` as header and `No. ${pageNo} / ${pageCount}` as footer, the exported PDF is like:



This is an example reference-style link.
The implicit link name shortcut allows you to omit the name of the link, in which case the link text itself is used as the name. Just use an empty set of square brackets — for example, to link the word "Google" to the google.com web site, you could simply write:

```
[Google]()
```

And then define the link:

```
[Google] http://google.com/
```

In Typora, clicking the link will expand it for editing, and command+click will open the hyperlink in your web browser.

URLs

Typora allows you to insert URLs as links, wrapped by <brackets>. For example <@typora.io> becomes @typora.io.

Typora will also automatically link standard URLs (for example: www.google.com) without these brackets.

Images

Images have similar syntax as links, but they require an additional : char before the start of the link. The syntax for inserting an image looks like this:

```
:[Alt text]/(path/to/img.jpg)
```

```
:[Alt text]/(path/to/img.jpg "Optional title")
```

You are able to use drag and drop to insert an image from an image file or your web browser. You can modify the markdown source code by clicking on the image. A relative path will be used if the image that is added using drag and drop is in same directory or sub-directory as the document you're currently editing.

If you're using markdown for building websites, you may specify a URL prefix for the image preview on your local computer with property `typora-root-url` in YAML Front Matter. For example, Enter `typora-root-url:/User/Abner/Website/typora.io/` in YAML Front Matter, and then `![alt]/(/blog/img/test.png)` will be treated as `![alt](file:///User/Abner/Website/typora.io/blog/img/test.png)` in Typora.

You can [download the sample PDF here](#).

Note: Please leave enough margin when you want to append page header or footer.

Note: You can also set per-document headers / footers using [YAML Front Matter](#), after enabled “Read and overwrite export settings from YAML front matters” option in preferences panel. Sample YAML front matters is like:

```
---  
header:  
footer: Written in Typora  
---
```

For md file contains above YAML Font Matter, header will be empty and footer will be “Written in Typora” in exported PDF file.

PDF Metadata

You can add metadata to exported PDF file via [YAML Front Matter](#), followings are supported PDF metadata:

```
---
```

title: Export in Typora

```

author: John Snow
creator: Typora inc.
subject: Tutorial
keywords: [Pandoc, Tutorial, Export]
---
```

Note: when “keywords” is absent, typora will also use “tags”, and you can use list with same space indent in YAML to represent an array, for instance:

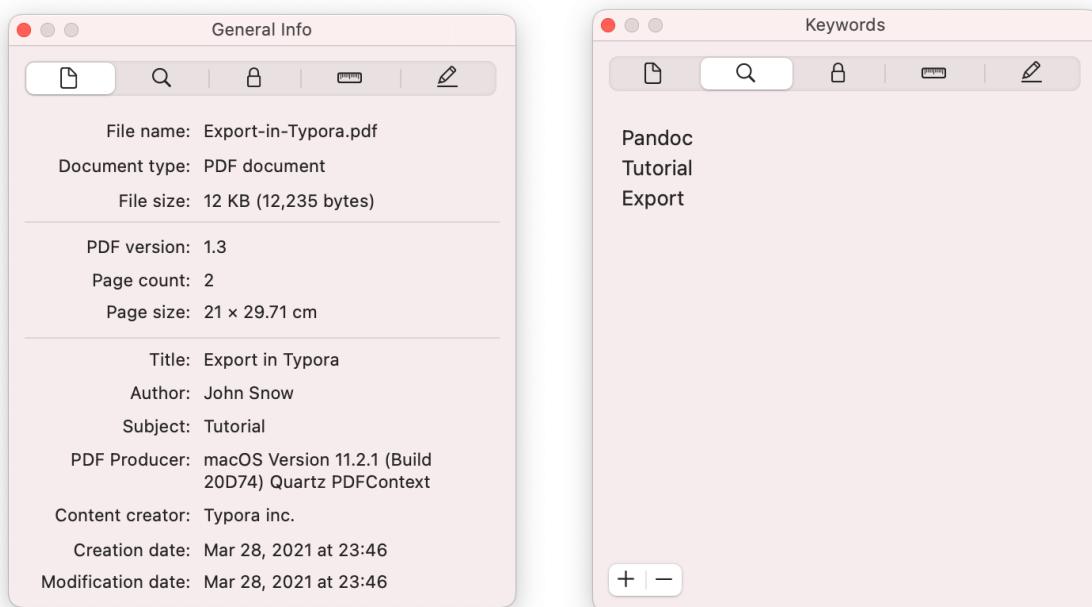
```

---
```

```

tags:
  - Pandoc
  - Tutorial
  - Export
---
```

Output is



Add Custom Content (PDF)

The PDF version is rendered from the HTML output, you could add HTML content (including scripts) in preferences panel → PDF → Append Extra Content, which will be appended in `<body>` tag when rendering generated HTML into PDF.

Example: Add Cover Page

It can be done by adding custom html contents into the `Append Extra Content` field, a simple case may be using the following content:



```
<meta name="title" content="${title}">
<div id='_export_cover' style="height:100vh;">
  <div id='_export_title' style="margin-top: 25%;text-align: center;font-size: 3rem;">
    </div>
</div>
<script>
var $cover = document.querySelector("#_export_cover");
var title =
document.querySelector("meta[name='title']").getAttribute("content");
if(!title || title == "${title}") {
  // no title
  $cover.remove();
} else {
  document.body.insertBefore($cover, document.body.childNodes[0])
  $cover.querySelector("#_export_title").textContent = title;
}
</script>
```

Another option is to use LaTex / Pandoc to generate PDF, you can see details below.

PDF (LaTeX / Pandoc)

Typora also supports you to generate PDF file using Pandoc and LaTeX (or other engines). By default, there is no such option in export menu, but you can add this by

1. Open Export section in preferences panel.
2. Click “Add” button on the export item list, select “PDF (LaTeX / Pandoc)” from template selector, and click “OK”.

PDF Engine

Pandoc provides serval engines for producing PDF file, including pdflatex, lualatex, xelatex, latexmk, tectonic, wkhtmltopdf, weasyprint, prince, context, and pdfroff. You will need to install one of those engines, and add it to your PATH, or specify the full path of the engine. You can config this in preferences panel.

Please refer to <https://pandoc.org/MANUAL.html#option-pdf-engine> for detail.

Note: On macOS, you could install MacTeX, open `/Applications/TeX/Docs` and `Utilities/Documents/What Is Installed.pdf` to check where binaries are installed to. On Intel macOS, the path for `pdflatex` after installing MacTeX-2020 might be `/usr/local/texlive/2020/bin/x86_64-darwin/pdftex`.

Template

You could also specify template option in preferences panel, for detailed explanations, please refer to [pandoc manual](#).

Meta Data (Other Options)

You could add meta data or extra options in [YAML Front Matter](#), please check <https://pandoc.org/MANUAL.html#variables-for-latex> for details, it support extra options including margins, paper size, font family, etc.

For example, if your select LaTeX as PDF engine, you can add footer and header via

```
---
```

```
header-includes: |
  \usepackage{fancyhdr}
  \pagestyle{fancy}
  \fancyhead[CO,CE]{This is fancy}
  \fancyfoot[CO,CE]{So is this}
  \fancyfoot[LE,RO]{\thepage}
```

```
--
```

Extra Arguments

You could append extra command line arguments for pandoc, for which arguments you could use, please refer <https://pandoc.org/MANUAL.html>.

Word (.docx)

You can export to word file via **Export → Word (.docx)** from menu bar. You can config it from **Export → Word (.docx)** in preferences panel.

Style Reference

For example, you can download [this sample reference doc file](#), then select this file in preferences panel for option “Style Reference”.

To create your own style template for docx output, you can generate one from command line. (<https://pandoc.org/MANUAL.html#option--reference-doc>)

```
pandoc -o custom-reference.docx --print-default-data-file reference.docx.
```

Then open `custom-reference.docx` in Word, modify the styles as you wish, and save the file. For best results, do not make changes to this file other than modifying the styles used by pandoc.

You can read more details [here](#).

OpenOffice (.odt)

You can export to OpenOffice file via **Export → OpenOffice** from menu bar. You can config it from **Export → OpenOffice** in preferences panel.

Template

You could find details on [pandoc manual](#), you could download the sample template [default.opendocument](#) from <https://github.com/jgm/pandoc-templates>.

RTF

RTF (Rich Text Format) is the format saved byTextEdit.app or WordPad.exe. You can export to RTF file via **Export → RTF** from menu bar. You can config it from **Export → RTF** in preferences panel.

Epub

You can export to RTF file via **Export → Epub** from menu bar. You can config it from **Export → Epub** in preferences panel.

Custom CSS

You can set custom css from preferences panel, if you don't know how to start with, you could check <https://github.com/mattharrison/epub-css-starter-kit> to make your own css style.

Chapter Level

Specify the heading level at which to split the EPUB into separate “chapter” files. The default is to split into chapters at level-1 headings. This option only affects the internal composition of the EPUB, not the way chapters and sections are displayed to users. Some readers may be slow if the chapter files are too large, so for large documents with few level-1 headings, one might want to use a chapter level of 2 or 3.

Cover Image

If you want to define a cover image fore pub output, please add a `cover-image: ${path}` in [YAML Front Matter](#). If it starts with `./` typora will convert to absolute path relative to current md file, or else, it should be absolute path, or pandoc will try to find one in its [data-dir](#).

Meta Data



You can use [YAML Front Matter](#) to set meta data on the output epub file, for example:

```
---
title: My Book
subtitle: An investigation of metadata
creator: Sarah Jones
identifier: doi:10.234234.234/33
publisher: My Press
rights: © 2007 John Smith, CC BY-NC
ibooks:
version: 1.3.4
---
```

You could find [more detail here](#).

LaTeX

Meta Data (Other Options)

Same as [Meta Data for PDF \(LaTeX / Pandoc\)](#), you could add meta data or extra options in [YAML Front Matter](#). Please check <https://pandoc.org/MANUAL.html#variables-for-latex>.

Add Extra Content in LaTeX Header

You can use `header-includes` or `latex_header` or `latex_header_extra` in [YAML Front Matter](#).

For example:

```
---
latex_header: |
  \usepackage[utf8]{inputenc}
  \usepackage[italian]{babel}
  \author{The Author}
  \date{2016}
  \title{The Title}
  \editor{The Editor}
classoption:
- twocolumn
- landscape
---
```

Wiki

Media Wiki

You can export to OpenOffice file via `Export → Media Wiki` from menu bar. You can config it from `Export → Media Wiki` in preferences panel.

Other Wiki Format

Please refer to [Other Formats Using Pandoc](#), you can use pandoc to convert markdown files into other wiki formats.

reStructuredText (RST, ReST, or reST)

You can export to OpenOffice file via `Export → reStructuredText` from menu bar. You can config it from `Export → reStructuredText` in preferences panel.

Textile

You can export to OpenOffice file via `Export → Textile` from menu bar. You can config it from `Export → Textile` in preferences panel.

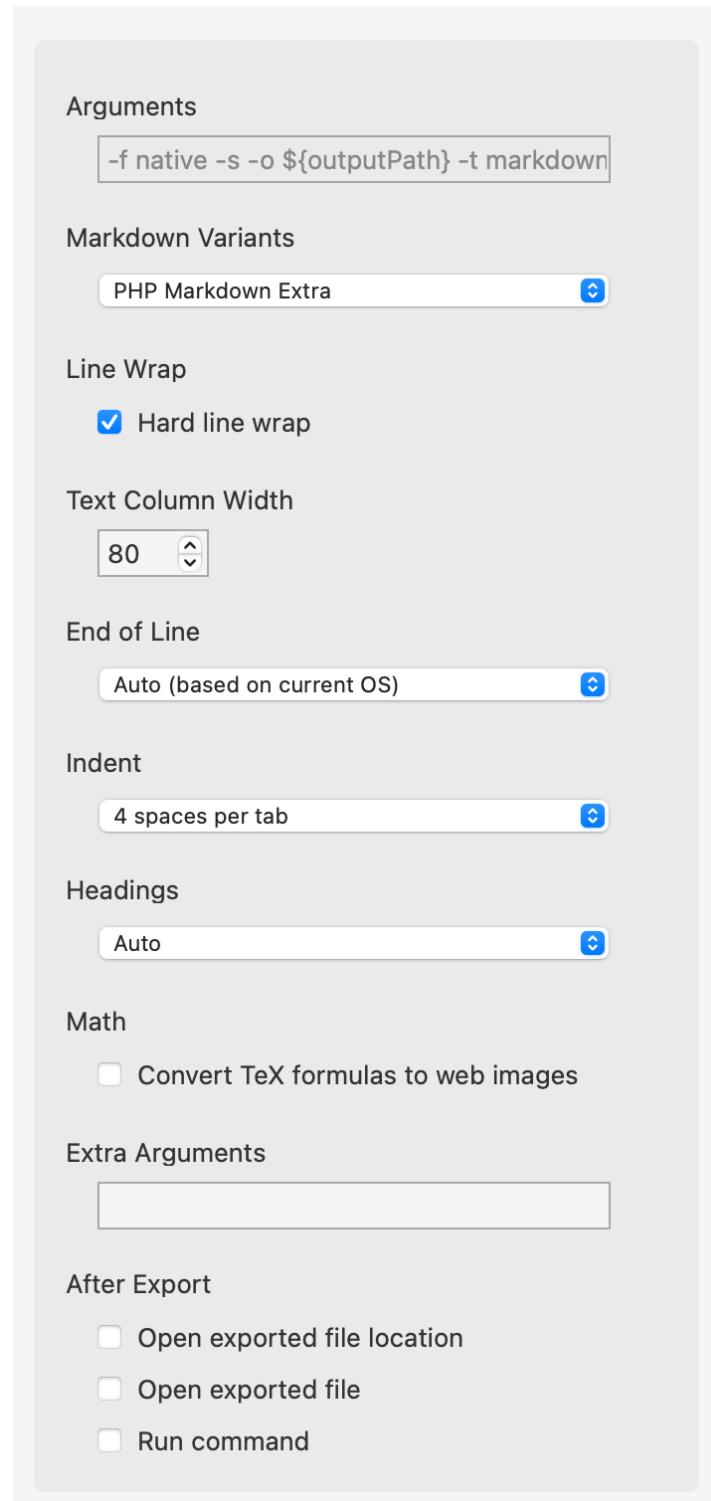
OPML

You can export to OpenOffice file via `Export → OPML` from menu bar. You can config it from `Export → OPML` in preferences panel.

Other Markdown Spec

You could follow [Other Formats Using Pandoc](#) to add new export target using template “Markdown (Other Spec)”. For details you and [check here](#).





You could also change options for the exported markdown file, including line wrap, line width, indent, unicode, etc.

Please note that “export” is not “reformat”, it will “represent” the same stuff, but its literal content may changed. For example:

```
[linke][typora]  
[typora]: http://www.typora.io
```

May become following content after export.

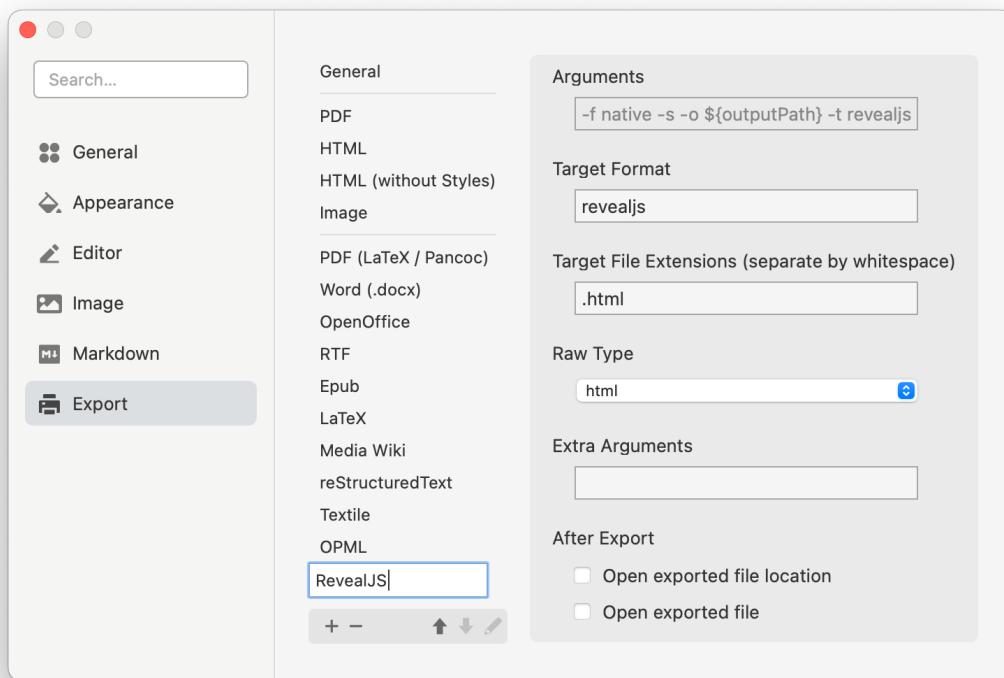
[[linke](#)](<http://www.typora.io>)

You can set line width, end of line, indent, only allow unicode characters, and other options as export configs.

Other Formats Using Pandoc

If you want to export current document to a format that pandoc supports but not included in Typora's export menu, you could config one such option in preferences panel.

Following is an example of providing reveals export option based on customized pandoc export option:



To start with, please add one export command choosing the “Pandoc” template, then add configurations based on your demand.

Target Format

You can leave this field empty, then pandoc will convert your document based on the exported filename. For example, if you export to `MyName.docx`, when pandoc will export the document to Word format automatically.

You could also specify a format, for possible options, please refer <https://pandoc.org/MANUAL.html#option--to>.

Target File Extension

When export, typora will pop up a save panel asking where to save the exported file. So please use this setting to set preferred file extensions, and please leave it empty when no preferred file extensions is required. You can set whitespace as separator, for example:

```
.latex .tex
```

Raw Type

For syntax that pandoc does not support natively, typora may use raw type as workaround.

For example, the native data model in pandoc cannot represent [toc], since there's no block type TOC in pandoc. In this case, if you set raw type as media wiki, typora will use __TOC__ syntax when export, if set raw type as latex, typora will use \\tableofcontents, etc.

Usually the raw type and target format should be same, and for html-based formats, like markdown, revealjs, etc.

And please note that the “Export” function cannot export all styles even when raw type is set.

Extra Arguments

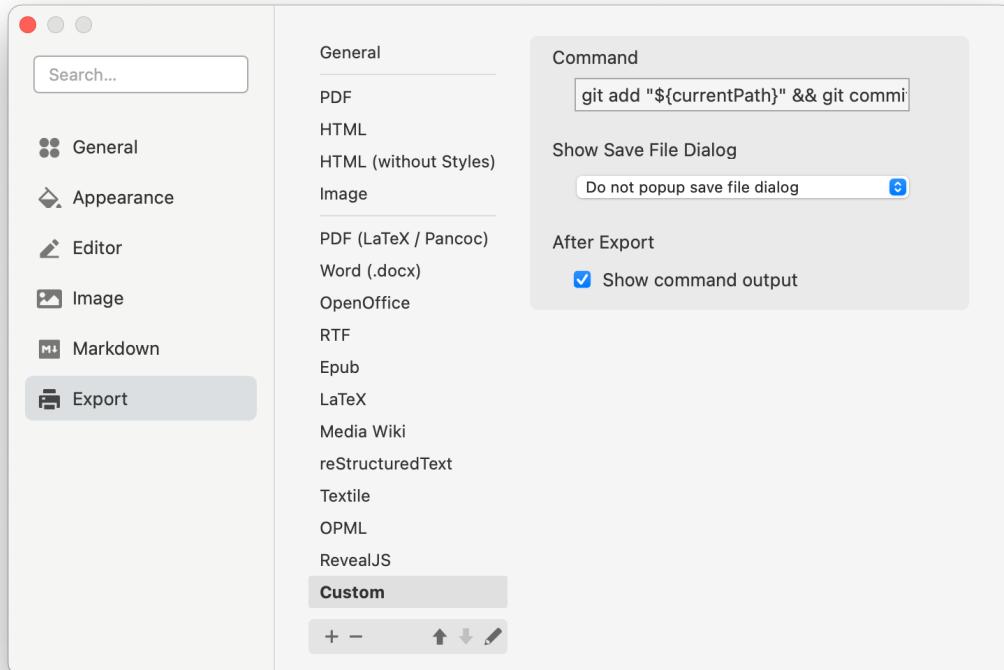
See [Append-extra Arguments](#).

Get Exported AST for Pandoc

To ensure the parse rule is same in Typora’s markdown and the output by pandoc. Typora will first convert into pandoc AST (Abstract Syntax Tree) saving in native format (see native under <https://pandoc.org/MANUAL.html#option--from>), then export to target format. You can get the native file by selecting “native” as target format in one custom pandoc export option.

Other Formats Using Custom Commands





You can also add export options based on command line commands. You will need to config:

Command

The command you want to trigger for this export option. For example, to submit current file to git server, you can use `git add "${currentPath}" && git commit -m "save" && git push`. The `${variable}` in this option will be replaced by variables defined in YAML front matter.

The command will be exported at the location of the folder of current file.

Show Save File Dialog

You can change this setting to tell typora if it needs to pop up a save file dialog, or pop up a folder select dialog, or no output path is required.

Target File Extensions

When `Select file path to save` is selected in “Show Save File Dialog”, you can use this filed to identify which file extensions are allowed for exported file target.

Show Command Output after Export

Besides the `Open exported file location` and `Open exported file location`, for export option based on custom commands, typora also provides a `Show command output` option. When enabled, typora will display dialog contains the command

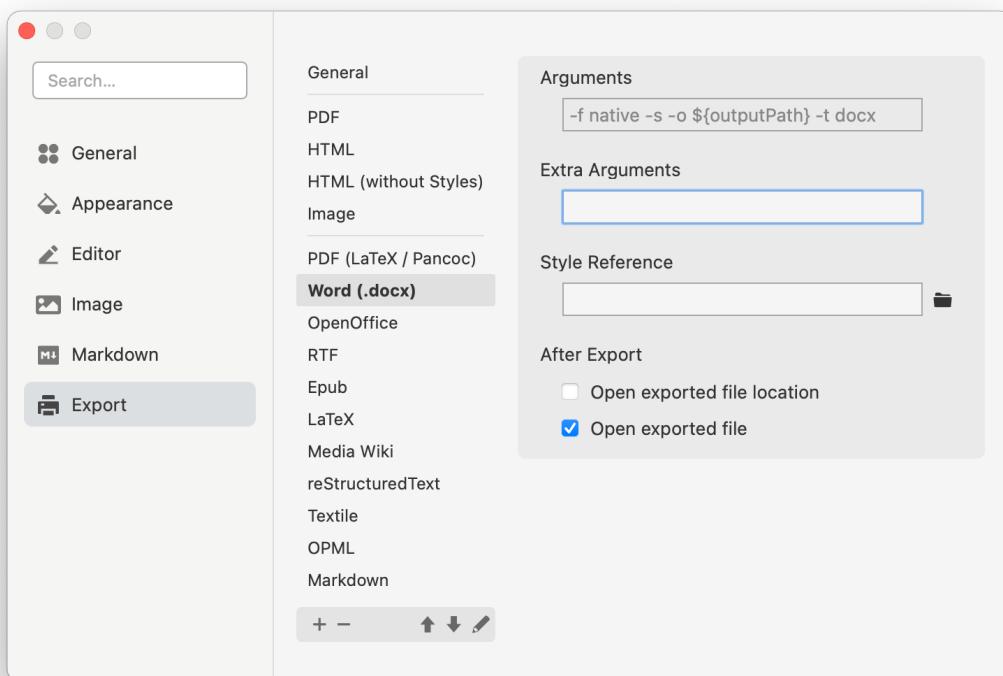


output (from `stdout`). Even it is not enabled, when export failed, the error message from command line will always be shown.

Other Pandoc Options

Append Extra Arguments

For pandoc-based export items, besides the options we put in the preferences panel, you can also use all available command line arguments in <https://pandoc.org/MANUAL.html> by appending them in the “Extra Arguments” field. For words like `${value}`, they will be replaced to `variables` defined in YAML front matter.



Appendix

Install Pandoc

Please refer [install pandoc](#).

Variables

You can use `${variables}` in custom header / footer text and custom export command, their values are:



Key	Value
<code> \${outputPath}</code>	Output file path after export. For example, if you export to location <code>/User/aaa/Documents/test.pdf</code> , then <code> \${outputPath}</code> will be replaced to that path.
<code> \${outputFileName}</code>	File name (without extension) of the saved exported file. It will be <code>test</code> in above case.
<code> \${outputFileFullName}</code>	File name (with extension) of the saved exported file. It will be <code>test.md</code> in above case.
<code> \${currentPath}</code>	Path of currently edited file. For example, if you are editing <code>/User/aaa/Document/readme.md</code> , then the value will be <code>/User/aaa/Document/readme.md</code> .
<code> \${currentFileName}</code>	Filename without extension of currently edited file. It will be <code>readme</code> in above case.
<code> \${currentFileFullName}</code>	Filename with extension of currently edited file. It will be <code>readme.md</code> in above case.
<code> \${today}</code>	Current date, for example: <code>2020-01-19</code>
<code> \${pageNo}</code>	Current page number. Only available for PDF format.
<code> \${pageCount} / \${totalPages}</code>	Total page counts. Only available for PDF format.
<code> \${title}</code>	Article title, should be defined in YAML Front Matter .
<code> \${author}</code>	Article author, defined in export options for PDF format, can be overwritten in YAML Front Matter .
<code> \${a.b}</code>	If <code>a</code> is an object defined in YAML Front Matter which contains <code>b</code> , then you can use <code>a.b</code> to access value for <code>b</code> .
Other variables	You can use keyword: <code>value</code> in YAML Front Matter , then uses <code> \${keyword}</code> variables in export configs.

Unset Variables

For example, if in preferences panel, “Author” is configured, you can “unset” this by `author: ''` in yaml.



Use Variables in Export Config

You can use \${variables} in custom header / footer text and custom export command.

Please note that if you use \${variables} in custom command line, typora may remove quotes ('") inside it and wrap it with quotes to avoid insecure commands.

You can also use them in custom contents (append-head, append-body, etc), to avoid XSS, only variables in <meta> tag will be replaced, but you can put your javascript to read them from <meta> tag, and append to where you want it.

Trouble Shooting

Require Pandoc to Continue...

This usually happened when Pandoc is not installed or typora cannot detect the pandoc path, please try following:

1. [Install pandoc](#)
2. Restart Typora after pandoc is newly installed.
3. If still not working, manually set pandoc path in **Export → General** from preferences panel.

Export PDF not working on Windows

1. Open Start.
2. Search for Services and click the top result to open the console.
3. Search for “Print Spooler” and start it, then click Apply and OK

Name	Description	Status	Startup Type	Log On As
Print Spooler	This service ...	Running	Manual	Local Syst...
Printer Extensions and Notifications	This service ...	Running	Manual	Local Syst...
PrintWorkflowUserSvc_40eee	Provides su...	Running	Manual	Local Syst...

See <https://github.com/typora/typora-issues/issues/3766>.

“Cannot Read Native” when export using pandoc

This is usually bug in typora side, please report it to us hi@typora.io with sample .md file, informations like which pandoc version you are using, target export format, and other necessary export configs.

Other Pandoc Errors

When export failed, the error dialog will show the pandoc command line parameters and error message. You can copy the parameters, replace “-f native”

with the markdown file path, and then input them in CMD (command line) or Terminal to test without Typora.

Continues whitespace or line break are gone in exported PDF

This relates to Markdown's spec for whitespace and line break. You can change in preferences in **Markdown → Whitespace / Line Break → Export / Print** to enable “Preserve whitespace and single line break” when export. For details, please check [Whitespace and Line Breaks](#) in typora.

Env: node: No such file or directory

If you use custom command like `node xxx`, please make sure `node` is in your system `PATH`. Or you can specify the `node` command with its full path. You can find your `node` path by input `which node` or `where node` in your shell to locate the `node` bin location.

Get Typora / Help Improve our documents

