

Narration&Reference How to install Pandoc in Ubuntu on WSL

Base File Name: NarrationReference_WSL_Ubuntu_Pandoc_ja

2018.11.8

How to install Pandoc in Ubuntu on WSL

by *Shuichi Ohtsu*

Operating environment

In this video, I will show you how to install Pandoc on Ubuntu installed on *WSL* (Windows Subsystem for Linux) .

Pandoc is a tool for converting documents written in one markup format to another format.

It is a tool developed by Haskell by Professor John McFarlane of the University of California, Berkeley.

There are quite a lot of supported formats, including Markdown, HTML, LaTeX, EPUB, PDF, etc.

Here, we will assume that you will convert the Markdown format file to PDF format and install it.

Pandoc's superiority is that you can directly write LaTeX syntax into Markdown format files.

Of course, in order to realize this, LaTeX must also be installed, so I will introduce this method as well.

In addition, for bibliography and cross-reference functions we are going to install additional tools, Pandoc-citeproc and Pandoc-crossref.

Since this video is supposed to create PDF files in Japanese, if you want to create PDF files of other languages, you do not need a language package or you need to incorporate a different language package , That point in advance please acknowledge.

For an overview and grammar of Pandoc, please refer to Reference of *Narration & Reference* file.

Install Pandoc

First of all, execute `sudo apt update` and `sudo apt upgrade` beforehand to update the system.

Then type `sudo apt install pandoc` to install Pandoc.

Install Pandoc-Citeproc

Next we are going to install Pandoc-citeproc.

Type `sudo apt install pandoc-citeproc`.

Install TeX

Next we are going to install TeX.

Type `sudo apt install texlive-luatex`.

Install TeX language package etc.

Next, install the packages related to each language. In this case, Japanese is assumed, but in other languages, install the corresponding language packages if necessary, or skip this process.

Type `sudo apt install texlive-xetex texlive-lang-japanese`.

Install Haskell

Next, install Haskell. This depends on Pandoc itself being developed by Haskell.

Type `sudo apt install haskell-platform`.

Install Pandoc-Crossref

Next, install the tool Cabal to install Pandoc-crossref.

Type `sudo apt install cabal-install`.

Next, we will obtain cabal related update information.

Type `sudo cabal update`.

Next, install pandoc-crossref using cabal.

Type `sudo cabal install --global pandoc-crossref`.

Please note that *global* option has been added here.

Since pandoc-crossref is used as a filter, this specification is required.

This process takes a considerable amount of time.

Start Visual Studio Code

Next, we are going to create a folder for Pandoc.

First, type `cd /mnt/c` from the command line and move to Windows C drive.

Next, type `ls` and display the contents of C drive.

It is OK.

Next, create a directory for Pandoc in a specific directory.

Here we created a directory named `Pandoc` in the `__myprg` directory.

Go to this Pandoc directory and type `code .` to start the Visual Studio Code.

Download of bibliography list style file

Since this time we will output up to the bibliography list, we need to download the style in advance.

First, open GitHub's style site in the browser.

That URL is <https://github.com/citation-style-language/styles/>.

Click on `chicago-author-date.csl` in this to display the data and copy all the rows.

Return to VS Code, create a new `chicago-author-date.csl` file and paste it into it.

Save this file.

Create Bib file

Next we are going to create a *bib* file for bibliography.

In this example, I created a new file `myref01.bib`.

We are going to write the literature information here.

For the notation, see *Reference*.

Creating a sample Markdown file

Next, we are going to create sample data for testing in Markdown format.

Here I created it as `sample01.md` file.

Creating the Node.js file

Next, we are going to compile with Pandoc, but since it is necessary to specify so many parameters, I decided to write a script for Node.js.

Create a file called `makepdf01.js` and describe the processing process here.

In addition, since this source was uploaded to GitHub, please see it for details.

Starting the Node.js file

Next, run `makepdf01.js` from the command line.

Type `node makepdf01.js sample01`.

When compilation is completed, a new `sample01.pdf` file is generated.

On the VS Code, start the file explorer and display this PDF file.

Confirmation of generated PDF file

A PDF file was created and a table of contents was also created.

Each link is also functional.

A bibliography list is also generated.

Direct invocation of Pandoc from the command line

Up to this point, compiling with Pandoc using Node.js was possible, but even if Node.js is not installed, you can also compile directly with Pandoc by specifying many parameters on the command line.

In that case,

```
pandoc sample02.md -o sample02.pdf -V documentclass=ltjarticle --pdf-engine=lualatex --toc --bibliography=myref01.bib --csl chicago-author-date.csl --filter pandoc-crossref.
```

We are going to try to actually enter.

The `pdf-engine` name was wrong. Fix this and restart it.

This time, the extension of `sample02` was insufficient. Fix this and restart it.

It is OK.

`sample02.pdf` has been generated.

Confirmation of generated PDF file

We are going to display this file.

It is OK.

Thank you for your attention.

Reference

- "Sample data and program (Ohtsu/pandoc-o2-makepdf)",
<https://github.com/Ohtsu/pandoc-o2-makepdf>
- "citation-style-language/styles",
<https://github.com/citation-style-language/styles/>
- "Pandoc ユーザーズガイド 日本語版",
<http://sky-y.github.io/site-pandoc-jp/users-guide/>
- "Pandoc User's Guide",
<https://pandoc.org/MANUAL.html>
- "WSL (Windows Subsystem on Linux) で pandoc メモ",
<https://qiita.com/miyamiya/items/4d2e93ad7895e302c27e>
- "プログラミングPandoc",
https://www.amazon.co.jp/dp/4274067815/ref=sxbs_sxwds-stvp_1?pf_rd_m=AN1VRQENFRJN5&pf_rd_p=14895845-6b63-47e2-b967-96bf0ca66fcb&pd_rd_wg=ZJDGq&pf_rd_r=0649V2CWECEG2NZ0KXHKB&pf_rd_s=desktop-sx-bottom-slot&pf_rd_t=301&pd_rd_i=4274067815&pd_rd_w=zZKAt&pf_rd_i=Pandoc&pd_rd_r=7e5f7e6c7-4c7f-9e7e-a4ef2c2531f2&ie=UTF8&qid=1541321932&sr=1
- "化学系だけど Markdown でレポートを書いて Pandoc を使った",
<http://pinkmagenta.hatenablog.jp/entry/2017/12/20/124911>

- "Pandocで相互参照",
<http://pinkmagenta.hatenablog.jp/entry/2017/12/20/124911>
- "MarkdownとPandocを使って論文っぽい文章を書く",
<https://inody1991.tumblr.com/post/134742076815/markdown%E3%81%A8pandoc%E3%82%9C>
- "Windows 10 Installation Guide",
<https://docs.microsoft.com/en-us/windows/wsl/install-win10>
- "Windows 10でLinuxプログラムを利用可能にするWSLをインストールする(バージョン1803以降対応版)",
<http://www.atmarkit.co.jp/ait/articles/1608/08/news039.html>
- "Windows Subsystem for Linuxをインストールしてみよう！",
<https://qiita.com/AruneKo/items/c79810b0b015bebf30bb>
- "「Windows Subsystem for Linux(WSL)」セットアップガイド【スクリーンショットつき解説】",
<https://linuxfan.info/wsl-setup-guide>
- "Angular5, Angular6, Angular7 Custom Library: Step-by-step guide",
<https://www.udemy.com/angular5-custom-library-the-definitive-step-by-step-guide/>
- "Angular5, Angular6, Angular7用 カスタムライブラリの作成: 完全ステップ・バイ・ステップ・ガイド",
<https://www.udemy.com/angular5-1/>