

目次

1	事前準備	1
1.1	Rmd ファイルについて	1
1.2	YAML フィールドの指定	2
1.3	ファイルの結合	2
1.4	出力設定	2
2	second chapter	3

1 事前準備

- github で新規リポジトリを作成する。
- RStudio から New Project -> VersionControl -> Git と選択し、Repository URL に先程作成したリポジトリの URL を指定、新規プロジェクトを作成する。
- bookdown パッケージをインストールしていなければインストールしておく。

```
install.packages("bookdown")
```

1.1 Rmd ファイルについて

- Rmd ファイルはワーキングディレクトリに配置する。
- デフォルトでは、Rmd ファイルは名前順に結合される。例えば 01-intro.Rmd の次は 02-literature.Rmd といった具合に。
- アンダースコア _ から始まるファイルは無視される。
- index.Rmd というファイルがあった場合は、常に最初のファイルとして使用される。
- ファイル名に関するこれらの挙動は _bookdown.yml というファイルを book のディレクトリに配置し、設定を記入することで上書きできる。
 - 例: Rmd ファイルの名前と処理順序を指定したければ、rmd_files フィールドに指定する。

```
rmd_files: ["index.Rmd", "second.Rmd", "third.Rmd"]
```

1.2 YAML フィールドの指定

`_bookdown.yml` でも指定できるが、最初の Rmd ファイルの YAML メタデータで Pandoc 関連の設定を指定することもできる。このファイルの指定例を示す。

```
---
title: "bookdown の説明"
author: " かつどん"
date: "2018-03-05"
site: bookdown::bookdown_site
documentclass: book
rmd_files: ["index.Rmd", "second.Rmd"]
output:
  bookdown::gitbook: default
---
```

1.3 ファイルの結合

- Rmd ファイルをマージしてから knit する (M-K アプローチ) か、それぞれ knit してからマージする (K-M アプローチ) か、2 つの方法がある。
- 大きな違いはコードチャンクが同時に評価されるか、ファイルごとに評価されるかという点である。
- knitr はチャンクラベルの重複を許していないため、M-K アプローチでは全てのファイルに渡ってチャンクラベルが重複しないように注意する必要がある。K-M アプローチでは一つのファイル内でチャンクラベルが重複しなければ良い。
- K-M アプローチでは Rmd ファイルをサブディレクトリに入れることはできないが、M-K アプローチなら可能である。
- デフォルトは M-K アプローチであり、変更したければ `render_book()` 呼び出し時に `new_session = TRUE` を指定するか、`_bookdown.yml` に `new_session: yes` を指定する。

1.4 出力設定

- 出力に関する設定は、最初の Rmd ファイルに YAML メタデータとして記載する他に、`_output.yml` というファイルを作成してそこに記述する方法もある。
- Rmd ファイルに記述する場合は、`output` フィールドに記載するが (先程の例にも含まれている)、`_output.yml` に記述する場合は `output` フィールドを省略し、すべての設定をトップレベルに記述する。また、`---` は不要である。以下に例を示す。

```
bookdown:::gitbook:  
  lib_dir: assets  
  split_by: section  
  config:  
    toolbar:  
      position: static  
bookdown:::pdf_book:  
  keep_tex: yes  
bookdown:::html_book:  
  css: toc.css
```

2 second chapter

this is second chapter.