

AsciiDoc環境構築 覚え書き

# 目次

1. asciidocについて .....	1
2. 環境構築手順 .....	2
2.1. chocolateyをインストール .....	2
2.2. ruby等のパッケージをインストール .....	2
2.3. asciidoc関連ツールをインストール .....	3
2.4. 作業ディレクトリを作成する .....	3
3. 実際にasciidocを書いてみる .....	5
3.1. attributeを書く .....	5
3.2. テストサンプルを書く .....	5
3.3. プレビューを行う .....	7
3.4. htmlやpdfに変換する .....	7
4. その他情報 .....	9

# 1. asciidocについて

asciidocとは？

- markdownのような軽量マークアップ言語の一つ
- プレーンテキストで体裁が整った文章を書くのが得意
- 可読性が高くmarkdownよりも表現力が高い

メリット

- テキスト形式なのでGitで管理しやすい
- インクルード機能により外部ファイルの読み込みができる
- html、pdfに変換できる

## 2. 環境構築手順

1. パッケージマネージャーのchocolateyをインストール
2. chocolateyのリポジトリに登録されているruby等のパッケージをインストール
3. ruby製のasciidoc関連ツールをインストール
4. 作業ディレクトリを作成

### 2.1. chocolateyをインストール

以下を参考にインストールします

<https://chocolatey.org/install> <https://qiita.com/konta220/items/95b40b4647a737cb51aa>  
<https://qiita.com/NaoyaOura/items/1081884068fe3ea79570>

コマンドプロンプト(管理者権限)で以下を実行

```
@"%SystemRoot%\System32\WindowsPowerShell\v1.0\powershell.exe" -NoProfile  
-InputFormat None -ExecutionPolicy Bypass -Command "iex ((New-Object  
System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))"  
&& SET "PATH=%PATH%;%ALLUSERSPROFILE%\chocolatey\bin"
```



必ず公式サイトからコマンドをコピーして実行すること  
インストールが成功すれば以下のコマンドで確認することができる  
`choco -v`

chocolateyとは？

- windows上で動作するソフトウェアをコマンドラインでパッケージ管理可能なツール

メリット

- chocolateyのリポジトリに登録されているパッケージを一発でインストールできる
- chocolateyでインストールしたソフトは一括アップデートできる

### 2.2. ruby等のパッケージをインストール

コマンドプロンプト(管理者権限)で以下を実行

```
cinst -y ruby ①  
cinst -y graphviz ②  
cinst -y jdk8 ③  
cinst -y vscode ④
```

- ① ruby
- ② Graphviz
- ③ Java
- ④ Visual Studio Code(テキストエディタ、以下`VScode`とする)



cinst = choco install の略コマンド  
 インストールが成功すれば以下のコマンドでverを確認できる

```
ruby -v
gem -v
```

GraphvizとJavaはPlantUML等で図のプレビューに必要なとなる  
 VScodeの拡張機能でasiidocをプレビューしながら編集が可能

## 2.3. asciidoc関連ツールをインストール

コマンドプロンプトで以下を実行

```
gem install asciidoctor ①
gem install --pre asciidoctor-pdf ②
gem install asciidoctor-pdf-cjk ③
gem install asciidoctor-diagram ④
gem install coderay ⑤
```

- ① asciidoc→htmlに変換用
- ② asciidoc→pdfに変換用
- ③ pdf変換のレイアウト崩れ対应用
- ④ PlantUML等の図の記述用
- ⑤ コードのシンタックスハイライト用



社内のネットワークから実施する場合はgemにproxyを指定する  
 例) gem install xxxx -p proxy <http://アドレス:ポート>  
 proxy確認方法は以下参考  
<https://pasokatu.hateblo.jp/entry/2017/07/04/111147>

## 2.4. 作業ディレクトリを作成する

ドキュメントを作成するための作業ディレクトリを作っておく

```
| -test          // ドキュメントを格納するフォルダ(任意の名前でOK)
| -dist          // pdfの出力先
| -fonts         // フォントファイルを格納
| -images        // イメージファイルを格納
| -style         // スタイルファイルを格納
```

以下のファイルを作業ディレクトリに格納する

- htmlのスタイルファイル  
 windowsの場合は以下に入っているのでcssファイルをコピーして格納

```
// ruby2.6でasciidoctorのverが2.0.10の場合
C:\tools\ruby26\lib\ruby\gems\2.6.0\gems\asciidoctor-
2.0.10\data\stylesheets\asciidoctor-default.css
```

- pdfのスタイルファイル

windowの場合は以下に入っているのでyamlファイルをコピーして格納

```
// ruby2.6でasciidoctor-pdfのverが1.5.0.beta.2の場合
C:\tools\ruby26\lib\ruby\gems\2.6.0\gems\asciidoctor-pdf-
1.5.0.beta.2\data\themes\default-theme.yml
```



public\_style.ymlにリネームし、下記サイトを参考に編集

<https://qiita.com/tamikura@github/items/5d3f62dae55617ee42bb>

- フォントファイル

windowsの場合は以下に入っているの中身を全てコピーして格納

```
// ruby2.6でasciidoctor-pdfのverが1.5.0.beta.2の場合
C:\tools\ruby26\lib\ruby\gems\2.6.0\gems\asciidoctor-pdf-
1.5.0.beta.2\data\fonts\*
```

- ドキュメントファイル

適当にメモ帳で以下の設定で作成して格納

```
拡張子 : .adoc
文字コード : UTF-8
```

作業フォルダ内はこんな感じになる

```
| -test
|   |-*.adoc
|   |-dist
|   |-fonts
|     |-LICENSE-mplus-testflight-58
|     |-LICENSE-noto-2015-06-05
|     |-mplus1mn-bold_italic-ascii.ttf
|     |-mplus1mn-bold-ascii.ttf
|     |-mplus1mn-italic-ascii.ttf
|     |-mplus1mn-regular-ascii-conums.ttf
|     |-mplus1p-regular-fallback.ttf
|     |-notoserif-bold_italic-subset.ttf
|     |-notoserif-bold-subset.ttf
|     |-notoserif-italic-subset.ttf
|     |-notoserif-regular-subset.ttf
|   |-images
|   |-style
|     |-default-theme.yml
|     |-public_style.yml
```

## 3. 実際にasciidocを書いてみる

VScodeで\*.adocファイルを開く

### 3.1. attributeを書く

とりあえず最低限の指定を行う

```
//日本語ドキュメント
:lang: ja
//文書タイプはbookにする
:doctype: book
//目次を自動生成する
:toc: left
//対象とする階層数を指定する
:toclevels: 3
//タイトルを変更する
:toc-title: 目次
//章見出し番号を出力する
:sectnums:
//章見出しのChapte.が表示されないようにする
:chapter-label:
//シンタックスハイライトを使用する
:source-highlighter: coderay
//画像をdata-uriとして埋め込む
:data-uri:
//イメージファイルを置くフォルダ
:imagesdir: ./images
//アイコンフォントを利用するフラグ
:icons: font
//pdf化時のフォントファイルを置くフォルダ
:pdf-fontsdir: ./fonts
//pdf化時のスタイルファイルを指定
:pdf-style: ./style/public_style.yml
//html化時のスタイルファイルを置くフォルダ
:stylesdir: ./style
//html化時のスタイルファイルを指定
:stylesheet: asciidoctor-default.css
```

### 3.2. テストサンプルを書く

attributeに続けて下記のテストサンプルを書く

= asciidocの使い方

== asciidocとは？

asciidocとは [blue]#軽量マークアップ言語# です

詳しくは<<can\_asciidoc,asciidocでできること>>を参照

[[can\_asciidoc]]

== asciidocでできること

.コードハイライト

[source, json]

```
{
  "hoge" : "fuga",
  "foo" : [1,2,3]
}
```

.結合+箇条書例

[cols="1,2a,3a"]

|====

|列1|列2|列3

3+|3列結合

.2+|2行縦結合|b-1|c-2

|b-2|

\* c-3

\* c-4

|====

[NOTE]

====

\* format="csv"ではできません

====

=== asciidoctorだとPlantUMLでシーケンス図作成

[plantuml]

----

actor ユーザー as user

user -> ログイン : ログインする

ログイン --> user:

----

公式リファレンス

[https://takumon.github.io/asciidoc-syntax-quick-reference-japanese-translation/#\\_%E8%84%9A%E6%B3%A8](https://takumon.github.io/asciidoc-syntax-quick-reference-japanese-translation/#_%E8%84%9A%E6%B3%A8)



### 3.3. プレビューを行う

VScodeの設定を行うことでプレビュー(ショートカット **Ctrl+K→V**)が可能

拡張機能をインストールします

[https://qiita.com/o\\_sol06/items/a07ebcb0b48295a4c3b3](https://qiita.com/o_sol06/items/a07ebcb0b48295a4c3b3)

画面上部の[表示]→[拡張機能]から **AsciiDoc** を検索しインストール

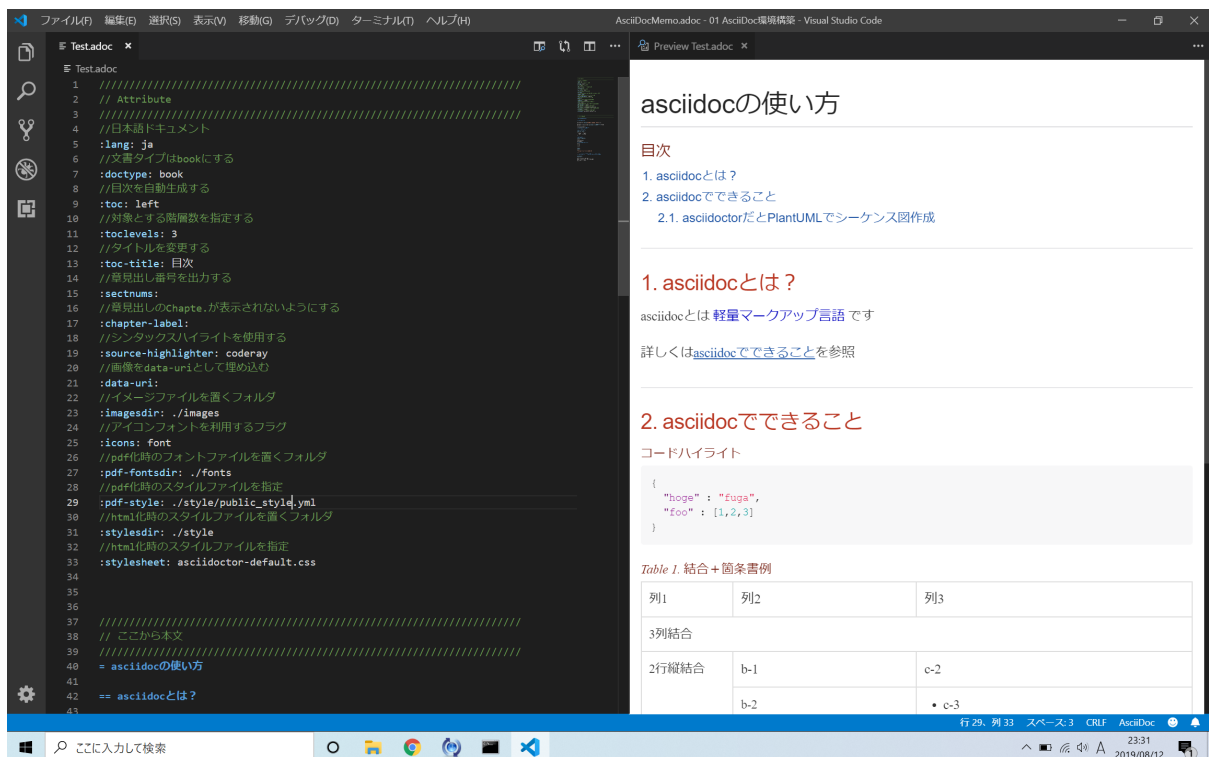
asciidocの設定を変更します

<https://qiita.com/hyt126/items/fdeff36f09bb221dfac0>

画面上部の[ファイル]→[基本設定]→[設定]から **asciidoc** を検索し、以下の設定を行う

```
asciidoc_command : asciidoc -n -r asciidoc-diagram -o-
asciidocpdf_command : asciidoc-pdf -n -r asciidoc-diagram -r
asciidoc-pdf-cjk -o-
use_asciidoc_js : false(チェックを外す)
```

参考までに「3.2.テストサンプル」のプレビュー結果を以下に示す



### 3.4. htmlやpdfに変換する

コマンドプロンプトで以下を実行

- htmlファイルに変換

```
asciidoc -r asciidoc-diagram -o dist/*.html *.adoc
```

- pdfファイルに変換

```
asciidoctor-pdf -r asciidoctor-diagram -r asciidoctor-pdf-cjk -o  
dist/*.pdf *.adoc
```

いちいちビルドコマンドを叩くのは面倒なのでビルドスクリプトのバッチファイルを作  
っておくとよい

make\_html.bat

```
@echo off +  
asciidoctor -r asciidoctor-diagram -o dist/*html *.adoc
```



make\_pdf.bat

```
@echo off +  
asciidoctor-pdf -r asciidoctor-diagram -r asciidoctor-pdf-cjk  
-o dist/*pdf *.adoc
```

## 4. その他情報

- pdfへ変換時に文字の色が変わらない 下記サイトの「2.2 asciidoctor-pdfのソースを修正する」を参考  
[https://blog.siwa32.com/asciidoctor\\_pdf\\_color/](https://blog.siwa32.com/asciidoctor_pdf_color/)
- pdfのフッター等の体裁や文言をカスタマイズ 下記サイトを参考  
<https://qiita.com/tamikura@github/items/5d3f62dae55617ee42bb>
- PlantUMLの内容を編集する度に画像ファイルが増えていくのを防ぎたい  
[plantuml,画像ファイル名]としてファイル名を指定するとよい

```
[plantuml,test]
----
class Animal {
    run()
}

class Cat extends Animal {
}
----
```

↓ プレビュー結果

