

LaTeX環境構築

TeX LiveとVisual Studio Codeを用いたLaTeX環境の構築ガイド

必要なもののインストール

どちらも最新版で大丈夫なはず。TeX Liveは2019版・2020版で動作確認済。

- [TeX Live](#)
- [Visual Studio Code](#)

TeX Liveはインストールするときセキュリティソフトに引っかかりがちだけど基本問題ない(はず)。

インストール後はコマンドとかでそれぞれ動作するかを確認。

```
# texの動作確認 -> 確認後はctrl+cで抜けられる
tex

# VSCodeの確認 こっちはわざわざコマンドで確認しなくてもOK
code
```

Visual Studio Codeに拡張機能をインストール

以下の拡張機能をインストール。インストール後VSCodeの再起動を要求される場合もある。

- [LaTeX Workshp](#)

VSCodeの日本語化拡張機能もインストールしていなければ入れておくと便利かも。

- [Japanese Language Pack for Visual Studio Code](#)

settings.jsonにオプションを追記

VSCodeのsettings.jsonにいくつかオプションを追加することで自分好みのLaTeX環境を目指す。

- [settings.jsonの開き方](#)

拡張子どおりjsonファイルなのでjsonの記法に従えば問題ない。

以下の中から必要と思ったものをsettings.jsonにコピペする。以下の設定は正直どれもかなり便利なので特に問題なければ全部コピペがオススメ。これ以外の設定も勿論あるので、ネットで良さげなのがあればどんどん追加して自分好みにカスタマイズしよう。//はコメント文。

```
{
  // -----
```

```

// latex
// -----

// tabをスペース2個で定義
"[latex]": {
  "editor.tabSize": 2,
  // "files.encoding": "shiftjis"
},

// editor.wordSeparators: 単語単位での移動を行う場合の区切り文字を指定
"editor.wordSeparators": ". / \\ ( ) \\ \" ' - : , . ; < > ~ ! @ # $ % ^ & * | + = [ ] { } ` ~ ? 、 。 「 」 【 】 『 』 ( ) ! ?
てにをはがのともへでや",

// latex-workshop.latex.tools: Tool の定義
"latex-workshop.latex.tools": [
  {
    "name": "latexmk",
    "command": "latexmk",
    "args": [
      "-f",
      "-gg",
      "-synctex=1",
      "-interaction=nonstopmode",
      "-file-line-error",
      "%DOC%"
    ]
  },
  {
    "name": "ptex2pdf",
    "command": "ptex2pdf",
    "args": [
      "-l",
      "-ot",
      "-kanji=utf8 -synctex=1",
      "-interaction=nonstopmode",
      "-file-line-error",
      "%DOC%"
    ]
  },
  {
    "name": "ptex2pdf (uplatex)",
    "command": "ptex2pdf",
    "args": [
      "-l",
      "-u",
      "-ot",
      "-kanji=utf8 -synctex=1",
      "-interaction=nonstopmode",
      "-file-line-error",
      "%DOC%"
    ]
  },
  {
    "name": "pdflatex",

```

```

        "command": "pdflatex",
        "args": [
            "-synctex=1",
            "-interaction=nonstopmode",
            "-file-line-error",
            "%DOC%"
        ]
    },
    {
        "name": "bibtex",
        "command": "bibtex",
        "args": [
            "%DOCFILE%"
        ]
    },
    {
        "name": "pbibtex",
        "command": "pbibtex",
        "args": [
            "-kanji=utf8",
            "%DOCFILE%"
        ]
    },
],

```

// latex-workshop.latex.recipes: Recipe の定義

```

"latex-workshop.latex.recipes": [
    {
        "name": "ptex2pdf*2",
        "tools": [
            "ptex2pdf",
            "ptex2pdf"
        ]
    },
    {
        "name": "latexmk",
        "tools": [
            "latexmk"
        ]
    },
    {
        "name": "pdflatex*2",
        "tools": [
            "pdflatex",
            "pdflatex"
        ]
    },
    {
        "name": "pdflatex -> bibtex -> pdflatex*2",
        "tools": [
            "pdflatex",
            "bibtex",
            "pdflatex",
            "pdflatex"
        ]
    }
]

```

```

    ],
    {
      "name": "ptex2pdf -> pbibtex -> ptex2pdf*2",
      "tools": [
        "ptex2pdf",
        "pbibtex",
        "ptex2pdf",
        "ptex2pdf"
      ]
    },
    {
      "name": "ptex2pdf (uplatex) *2",
      "tools": [
        "ptex2pdf (uplatex)",
        "ptex2pdf (uplatex)"
      ]
    },
    {
      "name": "ptex2pdf (uplatex) -> pbibtex -> ptex2pdf (uplatex) *2",
      "tools": [
        "ptex2pdf (uplatex)",
        "pbibtex",
        "ptex2pdf (uplatex)",
        "ptex2pdf (uplatex)"
      ]
    },
    {
      "name": "ptex2pdf",
      "tools": [
        "ptex2pdf", //タイプセットに使うtoolの名前
      ]
    },
  ],

```

// latex-workshop.latex.magic.args: マジックコメント付きの LaTeX ドキュメントをビルドする設定
 // 参考リンク: <https://blog.miz-ar.info/2016/11/magic-comments-in-tex/>

```

"latex-workshop.latex.magic.args": [
  "-f",
  "-gg",
  "-pv",
  "-synctex=1",
  "-interaction=nonstopmode",
  "-file-line-error",
  "%DOC%"
],

```

// latex-workshop.latex.clean.fileTypes: クリーンアップ時に削除されるファイルの拡張子

```

"latex-workshop.latex.clean.fileTypes": [
  "*.aux",
  "*.bbl",
  "*.blg",
  "*.idx",
  "*.ind",

```

```

        "*.lof",
        "*.lot",
        "*.out",
        "*.toc",
        "*.acn",
        "*.acr",
        "*.alg",
        "*.glg",
        "*.glo",
        "*.gls",
        "*.ist",
        "*.fls",
        "*.log",
        "*.fdb_latexmk",
        "*.synctex.gz",
        // for Beamer files
        "_minted*",
        "*.nav",
        "*.snm",
        "*.vrb",
    ],

    // latex-workshop.view.pdf.viewer: PDF ビューアの開き方
    "latex-workshop.view.pdf.viewer": "tab",

    // latex-workshop.latex.clean.onFailBuild.enabled: 一時ファイルのクリーンアップを行うかどうか
    "latex-workshop.latex.autoClean.run": "onBuilt",

    // latex-workshop.latex.autoBuild.onSave.enabled: .tex ファイルの保存時に自動的にビルドを行うかどうか
    "latex-workshop.latex.autoBuild.run": "onFileChange",
}

```

動作確認

texファイルがコンパイルが通るかをチェック。

とりあえずテストしたいだけであれば以下をコピーしてtest.texとでも名前をつけて保存してみよう。

```

\documentclass{jsarticle}

\begin{document}

\title{\LaTeX コンパイルテスト用}
\maketitle

test

\section{テストセクション}

```

```
\begin{itemize}
  \item test1
  \item test2
\end{itemize}

\end{document}
```

PDFに変換したいときは、VSCode左のメニューからTeXマークをクリックして **COMMANDS** -> **Build LaTeX project** 内の **Recipe** から、ソースのtexファイルに応じて適切なレシピを選択する。大抵の日本語論文は **ptex2pdf*2** でなんとかなるはず。コンパイル後のPDFは、基本ソースのtexファイルと同じディレクトリに生成される。

コンパイルするときの注意点として

- コンパイルを通したいtexファイルが共有フォルダ上などにあると通らない
- コンパイル時に出力PDFがChromeやEdgeで開かれていると上書きできず通らない

また、論文によっては適切なRecipeを選択する必要がある。卒修論であれば上で設定したもののみで大丈夫なはず。

ファイル保存時にコンパイルする設定であれば、Recipeの定義で一番上にあるコンパイルのレシピが自動で実行される (上の設定だと **"name": "ptex2pdf*2"** のレシピ)。別のレシピを既定にしたければ、settings.jsonのRecipe内で順番入れかえるだけでOK。

作成されたPDFはVSCode内で開けるので、そこで開いて確認しつつtexファイルを編集するのがオススメ。VSCode内で開いていればコンパイル時に自動で更新してくれるので確認も楽。