

1 Readme

% ju – <https://bw1.eu> – 3-11-18 – Readme.md

1.1 Hinweis

Projekt getestet unter Win10

1.2 Software

Pandoc: <https://pandoc.org/installing.html>

Latex: <https://www.tug.org/texlive/acquire-netinstall.html>

```
# Shell: TeXlive update  
tlmgr update --all
```

Editor: <https://code.visualstudio.com/download>

```
# Editor visual studio code  
# Datei / einstellungen / User settings  
{  
  "powershell.powerShellExePath": "C:\\PowerShell6\\6\\pwsh.exe",  
  "php.executablePath": "C:/xampp/php/php.exe",  
  "python.pythonPath": "C:\\Python37\\python.exe",  
  "editor.tabSize": 2,    // tabulator  
  "files.eol": "\\n",    // LF Linux  
  "files.encoding": "utf8", //coding  
  "files.autoSave": "afterDelay", //speichern  
  "files.autoSaveDelay": 2000, //2s  
  "editor.fontSize": 17,  
  "workbench.colorTheme": "Visual Studio Dark",  
  "window.zoomLevel": 0,  
}
```

Git: <https://git-scm.com/downloads>

```
# Shell: Git version
git --version
```

Imagemagick: <https://www.imagemagick.org/script/download.php#windows>

1.3 Repository piMonk von Github downloaden

```
# Shell: Kopie downloaden
$ git clone https://github.com/ju-bw/piMonk.git .
```

1.4 neues Repository auf github anlegen

```
# https://github.com/new
# github: Create a new repository
# Repository name = piMonk
# Shell: Git Befehle
# ".gitconfig", ".gitignore" konfigurieren und erstellen
git init
git add .
git commit -am "Projekt start"
git remote add origin https://github.com/ju-bw/piMonk.git
git push -u origin master
git status
git pull
git push
git log --oneline # less beenden mit <Shift+q>
git log --graph --oneline
git log --graph --pretty=format:"%cn; %h; %ad; %s" --date=relative >
    log.txt
```

1.5 Markdown Dokumente / Notizen

Markdown Dokumente / Notizen im Ordner "md/neu.md" erstellen.

Beachte das *min.* zwei Markdowndateien vorhanden sein müssen.

Powershellscript "docKonverter-v02.ps1" erstellt LaTeX - pdf und html files.

```
# Editor - Powershellscript "docKonverter-v02.ps1" anpassen
### Projekt
# anpassen
$thema = "piMonk" # Thema
$bildformat = "svg" # Bildformate: svg, jpg, png
$codeformat = "sh" # Codeformate: c, cpp, sh, py, ps1
$language = "Powershell" # Latex-Code: C, [LaTeX]TeX, Bash, Python,
Powershell
```

PowerShell: Script ausfuehren

```
$ ./docKonverter-v02.ps1
ju -- https://bw1.eu -- 26-0kt-18
```

Auswahlmenue

=====

- (0) schnell PDF erstellen
- (1) artikel.pdf
- (2) book.pdf
- (3) **print.pdf**
- (4) alle Abbildungen.tex
- (5) alle Quellcodedateien.tex
- (6) backup - "../\$archiv/\$timestampArchiv-\$thema.zip"
- (7) git - Repository auf github notwendig!
- (8) imgWeb.ps1 # ext. Script - Bilder optimieren (Latex/Web)
- (9) html

- (10) pandoc & suchen/ersetzen - Achtung: **min.** zwei Markdown Dateien
notwendig !!!
- (11) Projekt reset
- (12) Projekt neu
- (13) Beenden

Eingabe - [Zahl]:

1.6 Bilder optimieren

JPG Bilder in den Ordner "imgOriginal/" kopieren.

Powershellscript "#imgWeb.ps1" optimiert Fotos für das Web und die PDF Datei.

```
# Shell: Script ausfuehren
```

```
$ ./imgWeb.ps1
```