

# **style-tex3R**

Vincent Crombez & Frédéric Léothaud

# Table des matières

# Introduction

La `style-tex3R` fait partie d'un projet comprenant :

- Une classe, un style et des templates de documents pratiques
- Un environnement LaTeX portable
- Une extension (basée sur `latex-workshop`)

Il est préférable d'aller d'abord lire la documentation de `classe-tex3R` qui explique l'ensemble des commandes disponibles.

L'objectif de ce guide est de guider la personnalisation du style à appliquer aux environnements et commandes de la `classe-tex3R`, afin d'avoir une apparence propre selon les goûts de chacun, tout en gardant la possibilité de transférer son document à un autre utilisateur de la `classe-tex3R`, qui lui le compilera avec le style de son choix.

# 1 Créer un nouveau style

Afin d'utiliser son style personnalisé, il suffit de créer une copie du fichier `style-tex3R.sty` dans le même dossier, de le renommer `monstyle.sty` et de travailler sur ce dernier. Le fichier `style-tex3R` se trouve dans l'emplacement suivant :

```
TeX3R-ClasseStyle
├── documentation
├── templates
├── tex
│   ├── fonts
│   ├── lualatex
│   └── tex3R
│       ├── classe-tex3R.cls
│       └── style-tex3R.sty
├── .gitignore
└── README.md
```

Une fois le fichier `monstyle.sty` créé il est nécessaire d'ouvrir MiKTeX Console et de faire Tasks > Refresh file name database.

Ensuite, pour appeler `monstyle` dans un nouveau document, il suffit de changer le préambule comme suit :

```
\documentclass[fiche]{classe-tex3R}
\usepackage{monstyle}

\begin{document}

%Le texte du document

\end{document}
```

## 2 Principe de modification

`classe-tex3R` et `style-tex3R` sont codés majoritairement en `lua`. L'idée derrière cela est d'avoir une part minimale de choses à modifier dans le style, dans des chaînes de caractères `lua`, qui seront ensuite insérées et utilisées par les commandes de la classe. Ceci étant, les choses pouvant être modifiées sont toujours entre `[[ ]]` ou entre `' '` qui sont les délimiteurs des chaînes de caractères en `lua`.

Cette expérience simplifiée évite d'avoir à aller voir dans de nombreuses documentations pour connaître la syntaxe exacte pour chaque paramètre.

Par la suite, nous détaillerons chacune des fonctions `lua`, et à quoi correspondent les différentes parties modifiables.

## 3 FormatUtilisateur

```
PAGE.enumi = [[ \textbf{\arabic*.\~} ]]  
PAGE.enumii = [[ \textbf{\alph*.\~} ]]  
PAGE.itemi = [[ \textbullet ]]  
PAGE.itemii = [[ $\circ$ ]]
```