

# Fiche L<sup>A</sup>T<sub>E</sub>X

AMÉLIE LEDEIN

LOUIS LEMONNIER

YOAN GERAN

2022-2023

## Un langage à base de commandes

L<sup>A</sup>T<sub>E</sub>X est un langage à base de **commandes**. Une commande commence par un antislash et peut prendre des arguments obligatoires (entre accolades) et optionels (entre crochets). Par exemple, avec `\section{Test}`, on crée une section nommée « Test ». L<sup>A</sup>T<sub>E</sub>X se chargera de mettre en forme la section, de gérer la numérotation, etc.

De plus, L<sup>A</sup>T<sub>E</sub>X dispose également d'**environnements**. Il s'agit de placer un bout de code entre `\begin{nom_environnement}` et `\end{nom_environnement}`. Par exemple, on a un environnement `itemize` pour faire des listes.

```
\begin{itemize}
  \item Un.
  \item Deux.
\end{itemize}
```

Ce qui se lit mieux que si on avait juste une commande `\itemize` qui prendrait par exemple en paramètre les différents éléments de la liste.

## Structure d'un document L<sup>A</sup>T<sub>E</sub>X

Un **exemple**<sup>1</sup> est disponible à la page **page 3**. Il permet de voir la structure générale d'un document L<sup>A</sup>T<sub>E</sub>X et présente quelques commandes et environnements utiles.

- Un document commence par la commande `\documentclass` qui permet de choisir le type de document que l'on veut produire (article, live, diapositives, etc.).
- La commande `usepackage` permet de charger des *packages*. Les *packages* sont des bouts de code qui nous offrent des fonctionnalités supplémentaires (par exemple, `babel` permet de gérer la langue d'un document).
- Le corps du document est placé dans l'environnement `document`.

---

1. [https://lmf.cnrs.fr/downloads/AmelieLedein/latex\\_example\\_2022.tex](https://lmf.cnrs.fr/downloads/AmelieLedein/latex_example_2022.tex)

## Compilation et éditeur

Avec `pdflatex nom.tex`, on compile `nom.tex` pour créer `nom.pdf`. Pour compiler le code de l'exemple, il y aura également besoin de l'option `-shell-escape` (nécessaire pour le *package* `minted`). La commande sera donc la suivante.

```
pdflatex -shell-escape nom.tex
```

N'importe quel éditeur permet d'écrire du L<sup>A</sup>T<sub>E</sub>X et votre éditeur préféré a sûrement des outils pour vous aider à écrire votre code (auto-complétion, prévisualisation de code, intégration des outils pour gérer la bibliographie, etc.).

Vous pouvez également choisir un éditeur spécialisé pour l'écriture du L<sup>A</sup>T<sub>E</sub>X. TexStudio par exemple est disponible sous Windows, Linux et MacOS et a une interface assez facile à utiliser.

Notez que vous pouvez également choisir d'utiliser un éditeur en ligne comme [Overleaf](#), ce qui vous permettra notamment de modifier votre code depuis n'importe quel ordinateur. Néanmoins, une bonne idée pour avoir son code à peu près partout est d'utiliser git avec par exemple Gitlab pour vous aider à gérer votre projet.

## Classes et packages utiles

- La classe `scrartcl` pour remplacer la classe `article`.
- La classe `beamer` pour faire des diapositives (thème `metropolis` sympa).
- Le *package* `mathtools` pour des outils supplémentaires pour rédiger des maths.
- Les *packages* `amssymb` et `stmaryrd` pour plus de fontes et symboles mathématiques.
- Les *packages* `newpxtext` et `newpxmath` pour de meilleures polices.
- Le *package* `geometry` pour gérer les marges de son document.
- Le *package* `hyperref` pour des liens hypertextes, des urls, etc.
- Le *package* `graphics` pour inclure des images dans un document.
- Le *package* `minted` pour inclure du code.
- Le *package* `xcolor` pour ajouter des couleurs à son document.
- Le *package* `biblatex` avec l'outil Biber pour gérer sa bibliographie.
- Le *package* `tabularray` pour faciliter la création de beaux tableaux.
- Le *package* `ebproof` pour faire des arbres de dérivation.
- Le *package* `tikz` pour faire des dessins.
- Le *package* `pgfplots` pour tracer des graphiques.

```
\documentclass[12pt,french]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{minted}
\usepackage{babel}
\usepackage[babel=true]{microtype}

\setminted{fontsize=\small, frame=single}

\title{Titre}
\author{Auteur}
\date{\today}

\begin{document}
\maketitle

\section{Une section}

\subsection{Une sous-section}
On peut avoir du texte \textbf{en gras}, \textit{en italique},
en \textsc{petites capitales}, \texttt{sans empattements}, etc.
Des maths en ligne \$1 + \exp(i\pi) = 0\$ et hors-texte
\[
\sum_{n = 1}^{+\infty} \frac{1}{n^2} = \frac{\pi^2}{6}.
\]

On change de paragraphe en laissant au moins une ligne vide.
On a des environnements pour faire des listes, inclure du code, etc.

\begin{itemize}
\item Un.
\item Deux.
\end{itemize}

\begin{enumerate}
\item Un.
\item Deux.
\end{enumerate}

\begin{minted}{ocaml}
let rec iter_list f = function
| [] -> ()
| x::l -> f x; iter_list f l
\end{minted}
\end{document}
```