

# Guide d'utilisation de $\text{\LaTeX}$

Le KI'022



## Préambule

Ce document a pour but de présenter les fonctionnalités de base et les plus utiles de  $\text{\LaTeX}$ , ce langage étant devenu la norme pour produire des documents scientifiques aujourd'hui. Si vous ne connaissez absolument rien à ce langage pas de panique ! Ce guide est fait pour vous, la première section est consacrée à vous donner les bases qui vous permettront d'écrire tous vos rapports en  $\text{\LaTeX}$ . Ensuite, si cela vous intéresse, le guide explore plus en détails les différentes fonctionnalités offertes par  $\text{\LaTeX}$ .

Mais d'abord je vais essayer de répondre à la question " $\text{\LaTeX}$  ?" où plutôt, "Pourquoi me donner du mal à apprendre un langage compliqué si je peux faire plus ou moins la même chose avec Word ?". Le premier truc qu'on vous dira, c'est que  $\text{\LaTeX}$  permet de faire des jolies maths, et c'est probablement ce qui vous servira le plus mais il y a d'autres manières de répondre à cette question :

- $\text{\LaTeX}$  vous permet de vous concentrer uniquement sur le **contenu** de votre document et non sur la forme qui est gérée automatiquement.
- $\text{\LaTeX}$  demande un certain temps d'adaptation, surtout si vous êtes habitués à Word mais ensuite cela paraîtra naturel.
- Sans parler des maths,  $\text{\LaTeX}$  procure une grande facilité pour ce qui est de référencer ses figures, ses graphiques, faire une bibliographie ou une table des matières. Il sera plus facile et surtout plus rapide de produire un rapport scientifique avec  $\text{\LaTeX}$  car tout ce que j'ai cité se fera (presque) tout seul.

Bonne lecture !

# Table des matières

<b>1</b>	<b>Commencer <math>\text{\LaTeX}</math> rapidement</b>	<b>3</b>
1.1	L'installation . . . . .	3
1.2	Les bases d'un document $\text{\LaTeX}$ . . . . .	3
1.3	Les commandes classiques de mise en page . . . . .	4
1.4	Les images, tableaux et autres flottants . . . . .	5
1.5	Ecrire des mathématiques . . . . .	7
1.6	Complément . . . . .	8
1.6.1	Inclure du code . . . . .	8
1.6.2	Liens utiles . . . . .	8
1.6.3	Logiciel conseillé . . . . .	8
<b>2</b>	<b>Formatage de texte</b>	<b>9</b>
<b>3</b>	<b>Design de pages</b>	<b>9</b>
<b>4</b>	<b>Les listes</b>	<b>9</b>
<b>5</b>	<b>Tableaux et images</b>	<b>9</b>
<b>6</b>	<b>Références</b>	<b>9</b>
<b>7</b>	<b>Maths avancées</b>	<b>9</b>
<b>8</b>	<b>Choisir sa police</b>	<b>9</b>
<b>9</b>	<b>Pour aller plus loin</b>	<b>9</b>
<b>10</b>	<b>Les erreurs les plus communes</b>	<b>9</b>

# 1 Commencer L<sup>A</sup>T<sub>E</sub>X rapidement

## 1.1 L'installation

Avant d'écrire un document LaTeX, il vous faut une distribution de LaTeX. Le plus simple, rapide et efficace est d'utiliser le site *Overleaf*. Il suffit de se connecter et de créer un nouveau document et...c'est tout. Il n'y a pas plus simple, le seul désavantage<sup>1</sup> étant que vous allez avoir besoin d'une connexion internet, ce qui sera rarement un soucis.

De plus Overleaf vous permettra de travailler en simultané sur un même document, chose impossible sur un éditeur classique.

Vous trouverez quand même en annexe des conseils pour ceux qui voudraient avoir LaTeX sur leur PC.

## 1.2 Les bases d'un document L<sup>A</sup>T<sub>E</sub>X

Pour débiter un document L<sup>A</sup>T<sub>E</sub>X il faut utiliser un certain nombre de commandes et inclure des paquets<sup>2</sup>. Voici ce à quoi chacun de vos documents devrait ressembler (commandes obligatoires en rouge et obligatoires pour un document en français en bleu) :

```
\documentclass[options]{type}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[french]{babel}

\usepackage[options]{nom}

\title{titre}
\author{auteur}
\date{date}

\begin{document}
\maketitle
\taleofcontents

\section{Titre de la section}  (\section* pour éviter la numérotation)
\subsection{Titre de la sous-section}
Votre texte

\subsection{Titre de la sous-section}
Votre texte

\end{document}
```

Vous pouvez déjà essayer de reproduire ce schéma (sans le `\usepackage` en noir pour le moment) avec `\documentclass[11pt]{article}`, remplacez juste ce qui est en italique par le titre, auteur, date, etc...et vous devriez avoir un document qui ressemble à ceci :

---

1. Il y en a d'autres, mais peu conséquents par rapport à la facilité d'utilisation procurée par Overleaf

2. Le template fourni lors de la formation regroupe les paquets les plus utilisés en pratique.

# Un premier document

Le KI

4 août 2020

## Table des matières

<b>1</b>	<b>Première section</b>	<b>1</b>
1.1	Première sous-section . . . . .	1
1.2	Deuxième sous-section . . . . .	1

### 1 Première section

#### 1.1 Première sous-section

Du texte

#### 1.2 Deuxième sous-section

Du texte

Pas mal non ? En quelques lignes vous avez déjà un document prêt à être rendu.

## 1.3 Les commandes classiques de mise en page

Pour modifier le texte, il y a les commandes :

Commande	Rendu	Raccourci	Selon les éditeurs (Overleaf ou autre), il peut y avoir plusieurs raccourcis que ceux-ci, néanmoins il est toujours possible de les utiliser façon "Word", c'est à dire sélectionner le texte à mettre en italique par exemple puis d'appuyer sur Ctrl+I.
<code>\textbf{texte}</code>	→ <b>texte</b>	Ctrl + B	
<code>\textit{texte}</code>	→ <i>texte</i>	Ctrl + I	
<code>\texttt{texte}</code>	→ <b>texte</b>		
<code>\underline{texte}</code>	→ <u>texte</u>		
<code>\emph{texte}</code>	→ <i>texte</i>		
<code>\textsc{texte}</code>	→ TEXTE		
<code>\fbox{texte}</code>	→ <span style="border: 1px solid black; padding: 2px;">texte</span>		

Les listes à puces (respectivement numérotées) se font avec les commandes suivantes :

<code>\begin{itemize}</code>	ou	<code>\begin{enumerate}</code>
<code>\item texte</code>		<code>\item texte</code>
<code>\item texte</code>		<code>\item texte</code>
<code>\end{itemize}</code>		<code>\end{enumerate}</code>

Pour des listes avec des • plutôt que des tirets, mettre la commande `\frenchbsetup{StandardLists=true}` en début de document (avant le `\begin{document}`). On peut également "empiler" plusieurs listes en cascade, elles seront indentées au fur et à mesure.

→ En LaTeX, le `\begin{}` suivi d'un `\end{}` s'appelle un environnement, cela permet de formater ce que vous allez mettre entre ces deux commandes. **Faire attention à bien mettre le `\begin` et le `\end` sinon ça ne marchera pas !**

## Pour les sauts de lignes et paragraphes :

Vous l'avez peut-être remarqué mais en LaTeX, pour revenir à la ligne vous pouvez pas juste appuyer sur la touche **Enter**, il vous faut soit laisser une ligne blanche, mettre un double backslash `\\` à la fin de votre ligne (les deux combinés donne un saut de ligne) ou utiliser la commande `\newline`. En fait le retour à la ligne vous permettra de faire un nouveau paragraphe, celui-ci sera indenté par défaut (décalé vers la droite) pour palier ce problème il suffit de mettre `\noindent` en début de ligne ou, pour le faire pour tout le document, mettre la commande `\setlength{\parindent}{0pt}` en début de document (avant le `\begin{document}`).

Pareillement, les espaces ne se font pas en appuyant 40 fois sur la touche espace, il faut utiliser des commandes telles que `\quad` et `\qquad`.

Les citations se font de la façon suivante :

```
\begin{quote}
texte
\end{quote}
```

La commande `\footnote{texte}` crée une note en bas de page.

La position du texte sur la page se modifie avec les environnements `flushleft`, `center`, `flushright`. C'est certainement le 2e qui vous sera le plus utile :

Et pour des citations de plusieurs lignes :

```
\begin{quotation}
texte
\end{quotation}
```

```
\begin{center}
texte
\end{center}
```

Voilà pour le texte pur et simple, j'en profite pour dire qu'il ne faut pas mémoriser toutes les commandes, par exemple pour les listes, il existe plein d'options pour modifier le type de liste ou même en faire plusieurs en cascade etc...Mais il est inutile de retenir tout cela, une simple recherche sur internet vous donnera ce que vous voulez savoir ! Vous trouverez aussi dans ce guide une description des options les plus utiles dans la section [Section name].

## 1.4 Les images, tableaux et autres flottants

Un des points les plus importants dans un document est l'inclusion d'images, leur référencement et les disposer intelligemment pour ne pas perturber la lecture. Sur ce point, il peut être très compliqué de vouloir faire exactement ce que l'on a en tête en LaTeX car contrairement aux logiciels Office, on ne peut pas déplacer l'image avec la souris. Pas exactement certes mais pour la plupart de vos documents, LaTeX va gérer très bien tout seul votre placement d'images.

Commençons pas le plus basique, le `\includegraphics[scale=échelle]{chemin d'accès à l'image}`. Disons que vous ayez une image `image.png` dans le même dossier que votre fichier `.tex` (sur Overleaf vous pouvez directement importer vos images). Pour mettre cette image dans votre document il y a juste besoin d'écrire deux choses :

- `\usepackage{graphicx}` en début de document.
- `\includegraphics[scale=x]{image.png}` (avec *x* le facteur d'échelle entre 0 et 1) là où vous voulez insérer votre image.

Bien sur cette commande fonctionne dans les environnements cités ci-dessus, par exemple si vous voulez centrer votre image :

```
\begin{center}  
\includegraphics[scale= $x$ ]{image.png}  
\end{center}
```

D'autres paramètres sont possibles plutôt que le scale, voir section [Section name]

## Les flottants

La méthode présentée ci-dessus permet de dire à LaTeX précisément où doivent être insérées vos images. Mais il peut être parfois plus judicieux de laisser LaTeX organiser vos images, pour cela on utilise des flottants (non ce n'est pas pareil que des flottants en info...). Les flottants tiennent leur nom du fait que LaTeX va les déplacer à sa convenance pour faire le meilleur document possible (selon lui libre à vous de changer ensuite).

Prenons l'exemple de l'image précédente, nous allons maintenant la mettre dans un environnement supplémentaire nommé **figure** :

```
\begin{figure}[h]  
\centering (Equivalent à un center)  
\includegraphics[scale= $x$ ]{image.png}  
\caption{légende}  
\end{figure}
```

Sans le `\centering`, l'image sera à gauche.

L'argument "h" indique que l'on veut que notre image soit ici ("here"), c'est à dire là où on écrit le texte. On peut aussi mettre b ou t pour "top" et "bottom" par exemple, ainsi notre image se retrouvera en haut ou en bas de la page peut importe où l'on a écrit la commande.



FIGURE 1 – Le logo du KI

Voici une image incorporée dans un document avec les commandes ci-dessus. Il y a plein de manière de modifier la légende et l'image à votre convenance, comme l'encadrer, l'inclure dans du texte (package `wrapfig`) et d'autres. Plus de détails sont donnés dans la section [section name].

→ Attention cependant, LaTeX peut décider de ne pas respecter votre argument entre crochet [] si cela conduit l'image à déborder de la page ou à trop éloigner le titre d'une section de son texte.

Un autre exemple de flottants : Les tableaux, ceux-ci ont une syntaxe bien précise mais il existe certains sites comme [tablesgenerator.com](http://tablesgenerator.com) qui vous simplifie la vie, vous avez juste à rentrer vos données et il vous génère le code automatiquement.

## 1.5 Ecrire des mathématiques

On aborde ici une grosse partie de ce que vous allez faire en LaTeX, au début cela va être un peu laborieux, car vous passerez votre temps à regarder la magnifique fiche mémo faite par le KI (disponible à l'adresse [latex.enpc.org](http://latex.enpc.org)) avec entre autre les commandes mathématiques. Mais ces commandes sont assez naturels et vous les mémoriserez assez vite.

Avant toute chose, assurez vous d'avoir mis `\usepackage{amsmath}` en préambule.

Pour écrire des maths, c'est très simple, il suffit de se placer dans un environnement mathématique, obtenu avec `\( \)` ou `\[ \]`.

Ainsi le code suivant :

Le théorème de Pythagore `\( x^2 + y^2 = z^2 \)`  
a été prouvé faux pour des puissances supérieures.  
Cela veut dire que l'équation suivante n'a pas de  
solutions entières : `\[ x^n + y^n = z^n \]`

Devient sur le pdf :

Le théorème de Pythagore  $x^2 + y^2 = z^2$  a été  
prouvé faux pour des puissances supérieures.  
Cela veut dire que l'équation suivante n'a pas  
de solutions entières :

$$x^n + y^n = z^n$$

Observez la différences entre les deux, le `\( \)` permet d'écrire des maths sur la même ligne que le texte en cours alors que `\[ \]` centre vos maths en sautant une ligne (**inline** vs **display** mode). Vous pouvez aussi utiliser le `$` plutôt `\(` et `\)` pour un environnement inline.

Si vous voulez numéroter vos équations alors il faut utiliser l'environnement `equation` :

$$x^2 + y^2 = z^2 \tag{1}$$

Vous pouvez inclure toutes sortes de symboles dans un environnement mathématiques comme des lettres grecques  $\alpha$   $\beta$   $\gamma$ , des signes logiques et quantificateurs  $\forall$   $\exists$   $\cup$   $\sum$ . La plupart de ces symboles sont répertoriés dans la fiche mémo distribué lors de la formation ou disponible [latex.enpc.org](http://latex.enpc.org).

### Puissance et indice

Pour écrire des puissances et des indices à coté d'une variable ou d'un symbole, il faut utiliser `^` et `_`.

Le code suivant :

Devient sur le pdf :

`\[ a_1^2 + a_2^2 = a_3^2 \]`

$$a_1^2 + a_2^2 = a_3^2$$

En utilisant des `{ }` lorsque les exposants/indices ne sont pas un seul symbole :

`\[ x^{2\alpha} - 1 = y_{ij} + y_{ij} \]`

$$x^{2\alpha} - 1 = y_{ij} + y_{ij}$$

Plus compliqué :

`\[ \sum_{i=1}^{\infty} \frac{1}{n^s} = \prod_p \frac{1}{1 - p^{-s}} \]`

$$\sum_{i=1}^{\infty} \frac{1}{n^s} = \prod_p \frac{1}{1 - p^{-s}}$$

## 1.6 Complément

### 1.6.1 Inclure du code

Il est également possible d'inclure du code dans un fichier  $\text{\LaTeX}$  grâce au paquet `\usepackage{listings}` et aux commandes suivantes :

```
\lstset{language= nom de langage }
```

```
\begin{lstlisting}
```

*Le code, tel qu'il serait écrit dans un éditeur de texte classique. Exemple de code Python :*

```
for i in range (0 : N) :
```

```
X[i] = methode(X[i - 1])
```

```
\end{lstlisting}
```

Cela va inclure le code ainsi :

```
// New function.
```

```
int function(int x) {
```

```
    return x * 10;
```

```
}
```

```
for (int i = 0; i < 10; i++) {
```

```
    int j = function(i);
```

```
    std::cout << j << std::endl;
```

```
}
```

Il existe de nombreuses options pour ajouter la coloration syntaxique, encadrer le code, numéroté les lignes, mais les liens fournis en 1.6.2 expliquent tous ces détails.

### 1.6.2 Liens utiles

Voici quelque lien utile pour chercher des commandes, des options ou des précisions supplémentaires :

- <https://www.google.com>
- [tex.stackexchange.com](https://tex.stackexchange.com)
- <https://fr.wikibooks.org/wiki/LaTeX>
- <https://openclassrooms.com/courses/redigez-des-documents-de-qualite-avec-latex>
- <https://fr.sharelatex.com>
- <https://upont.enpc.fr/ponthub/autres>

### 1.6.3 Logiciel conseillé

ShareLaTeX est une solution pratique pour s'éviter l'installation de  $\text{\LaTeX}$  et faire des projets à plusieurs<sup>3</sup>, néanmoins, il peut être utile d'avoir  $\text{\LaTeX}$  sur son propre ordinateur et donc d'avoir un logiciel de traitement de texte adapté : Sublime Text<sup>4</sup> et TexMaker sont de bons logiciel testés et approuvés par le KI'018.

---

3. Deux maximum sur la version gratuite.

4. Sublime Text doit être configuré pour gérer  $\text{\LaTeX}$ .



- 2    Formatage de texte
- 3    Design de pages
- 4    Les listes
- 5    Tableaux et images
- 6    Références
- 7    Maths avancées
- 8    Choisir sa police
- 9    Pour aller plus loin
- 10   Les erreurs les plus communes