

Mon premier document LaTeX

Dimitri Donzé, Vincent Huber, Igor Milhit

10 mai 2023

Table des matières

1	LaTeX et ses principes	4
1.1	Définition et principes de base	4
1.2	Logiciels à installer	5
1.2.1	logiciels multi-usage	6
1.3	Outils en ligne	6
1.3.1	Overleaf	6
1.3.2	Papeeria	6
1.3.3	LaTeXBase	7
1.4	Structure d'un document LaTeX	7
2	TeXmaker	8
2.1	Parties principales d'une fenêtre TeXmaker	8
2.2	Paramètres et personnalisation	8
3	Commandes principales	9
3.1	Les types de documents	9
3.2	Packages	10
3.3	Notions de base	11
3.4	Structure du corps du document (Chapitrage, etc.)	11
3.5	Mise en page	12
3.6	Mise en forme du texte	13
3.7	Images	15
3.7.1	Les éléments flottants	15
3.7.2	L'enveloppement des éléments flottants	15
3.8	Tableaux	16
3.9	Équations Mathématiques	19
3.9.1	Lettres grecques et autres symboles	22
3.9.2	Affichage et alignement des équations	22
3.9.3	Aides dans TeXmaker	23
3.9.4	Générateurs d'équations en ligne	23
3.10	Citer ses sources et créer une bibliographie	23
3.10.1	Structure d'une référence bib	25
3.10.2	Différences entre le format BibTeX et BibLaTeX	25
3.10.3	Création du fichier bib depuis Zotero	25
3.10.4	Préciser les packages et les options dans TeXmaker	25
3.10.5	Principes de citation et de création de bibliographie	27
4	Beamer : les fichiers de présentation LaTeX	28

5	Outils complémentaires	29
5.1	Templates - Modèles	29
5.2	MathPix	29
5.3	CTAN Comprehensive TeX Archive Network	29

Chapitre 1

LaTeX et ses principes

1.1 Définition et principes de base

LaTeX est un langage informatique permettant de rédiger et mettre en page des textes. Il est donc utilisé dans certains domaines comme un traitement de texte où l'utilisateur se concentre avant tout sur la structure du document (GAUTRELET, 2020 ; TAILLET, 2022). Un éditeur LaTeX n'étant pas WYSIWYG, il sera nécessaire à l'utilisateur/trice de taper à la fois le texte proprement dit, mais également les commandes structurant et mettant en page le contenu (GERIN et al., 2008 ; MAGUIS, 2010).

Ce contrôle dans la structure et la mise en forme du document demande généralement un temps d'apprentissage et d'adaptation non négligeables lors de la création de ses premiers documents. Pour autant, votre utilisation fréquente et sa popularité dans certaines disciplines gommeront la majorité de ses difficultés. Ainsi, certaines institutions proposent des templates déjà préparés pour les différents types de documents rédigés au sein de celles-ci. De plus, il est relativement aisé de reprendre la structure d'un ancien document pour la réutiliser dans un nouveau ; le rédacteur ou la rédactrice n'aura ainsi plus qu'à se concentrer sur le contenu du document, la mise en forme ayant été définie *a priori* (GERIN et al., 2008).

LaTeX affiche également divers avantages en comparaison des éditeurs de texte WYSIWYG tels que WORD. Ce contrôle permet une personnalisation accrue des besoins du rédacteur/trice. D'une manière générale presque tout dans LaTeX est personnalisable ou paramétrable (MAGUIS, 2010). Les avantages de LaTeX sont notamment (MAGUIS, 2010 ; TAILLET, 2022) :

- Hiérarchisation des différentes parties d'un document (parties, chapitres, sections, etc.). Les différentes parties sont alors numérotées automatiquement
- Gestion typographique des éléments mathématiques (équations...) et des caractères spéciaux (lettres grecques, symboles, etc.).
- Gestion des *éléments flottants* tels que les images, graphiques, tableaux, etc.
- Facilité de faire référence à un élément antérieur en le labelisant.
- Le format du document source en texte brut rendant le contenu lisible par de nombreux logiciels mais rendant les instructions prévisibles et reproductibles.

Éditeur et distribution LaTeX

Concrètement, LaTeX fonctionne avec deux outils complémentaires.

1. Un éditeur LaTeX
2. Une distribution LaTeX

Certains outils, notamment en ligne ([overleaf](#), [papeeria](#)...), incluent les deux ensemble, alors que si vous travaillez en local, il sera peut-être nécessaire d'installer un élément de chaque.

- [TeXworks](#)

Linux

- Distribution
 - [TeX Live](#)
- Éditeurs
 - [TeXmaker](#). TeXmaker est un éditeur libre et multiplateforme. C'est le logiciel que nous utiliserons comme exemple dans ce polycopié et durant le cours.
 - [Kile](#)
 - [TeXworks](#)

1.2.1 logiciels multi-usage

Possibilité de configurer des éditeurs généralistes pour l'adapter à des langages comme LaTeX pour avoir les mêmes types d'options que les logiciels spécialisés ci-dessus. VS Code
Visual Studio

1.3 Outils en ligne

Les outils en ligne peuvent être une alternative très intéressante aux suites à installer. Dans ce cas-ci, l'éditeur et la distribution sont intégrés dans un outil à utiliser directement dans le navigateur web. Il y est nécessaire de se créer un compte afin de gérer ses projets. L'intérêt est donc un outil souvent plus facile d'accès (rien à installer, une interface WYSIWYG...), plus souple et peut être utiliser sur plusieurs ordinateurs à la fois. De plus, il est généralement possible de partager des projets à plusieurs, ce qui peut être très utile lors de la rédaction en groupe.

L'inconvénient principal est que ses fonctions avancées sont généralement payantes. Il est possible de créer un compte gratuit mais les fonctions y sont limitées suite à une période d'essai. De plus, les serveurs y sont parfois surchargés, limitant l'accès à vos projets. Pour l'instant, l'UNIGE ne propose aucune licence institutionnelle pour l'un de ces produits. Par contre, certains laboratoires/bureaux ont leur propre licence. Nous vous conseillons de vous informer auprès de vos collègues. La plupart de ces outils offrent la possibilité de sauvegarder localement ses fichiers permettant de travailler hors ligne.

1.3.1 Overleaf

[Overleaf](#) est l'un des outils en ligne les plus connus. Vous avez la possibilité de passer facilement d'une vue en commandes LaTeX en éditeur de texte WYSIWYG en un seul clic. La fonction payante est très complète (intégration de Zotero, Mendeley et de GitHub, Projets à plusieurs...) mais coûte cher (plus de 100CHF par année au minimum !!) alors que la version gratuite offre des options limitées mais qui peuvent suffire tout du moins pour se faire la main. Par exemple, le nombre de projets n'y est pas limité. De plus, vous avez accès à de très nombreux modèles de documents prédéfinis que vous pourrez utiliser ou desquels vous inspirer.



1.3.2 Papeeria

Papeeria propose également une version gratuite et une version payante de son outil. La version gratuite propose l'édition à plusieurs mais seul un projet peut être en mode privé. C'est-à-dire que vos autres documents seront visibles sur la plateforme. La version payante est moins onéreuse qu'Overleaf (5\$ par mois) mais n'autorise que 10 projets privés. Papeeria ne semble pas avoir d'éditeur WYSIWYG non plus.

1.3.3 LaTeXBase

LaTeXBase ne propose pas de travaux privés en version gratuite. Cependant, la version payante ne coûte que 4\$ par mois pour les étudiant-es.

1.4 Structure d'un document LaTeX

Un document LaTeX est composé de divers éléments qu'il est important de distinguer et de maîtriser afin de les compléter ou modifier correctement. D'une manière générale, un document LaTeX est composé de deux parties (GAUTRELET, 2020) :

- Le préambule : celui-ci contiendra les commandes définissant le type de document (livre, rapport, présentation, etc.), sa forme générale (marges, format de feuille, orientation, etc.), les extensions (packages), le lien avec le fichier de références bibliographiques, ainsi que des informations complémentaires comme le titre, les auteurs, etc.
- Le corps du document : le corps du document fait suite au préambule. il est composé du texte proprement dit qui sera affiché dans le fichier PDF, dans une structure bien définie (chapitres, sections, sous-sections, etc.).

Voici un exemple de structure de document LaTeX adapté de (GAUTRELET, 2020) :

```
\documentclass[llpt,a4paper]{article} %classe du document

%extensions alias packages
\usepackage[utf8]{inputenc} %encodage du texte
\usepackage[french]{babel} %langage --> règles
typographiques du français, francisation des titres, etc.
\usepackage[T1]{fontenc} %encodage de sortie
\usepackage{amsmath} %extension pour les mathématiques
\usepackage{amsfonts} %extension pour les mathématiques
\usepackage{amssymb} %extension pour les mathématiques
\usepackage{graphicx} %extension pour les images
\usepackage{lmodern} %extension pour les lettre grecques
\usepackage[left=2cm,right=2cm,top=2cm,bottom=2cm]{geometry}
%mise en page / marges
%gestion du titre et auteurs/trices
\title{Mon document LaTeX} %définition du titre
\author{Vincent Huber} %définition de l'auteur

\begin{document} %début du corps du document
\maketitle %affiche du titre

\section{Partie 1} %début de la lère section importante du
travail
\subsection{Sous-partie} %début de la première sous-section
de la partie 1
Le texte rédigé ici s'affichera comme tel dans mon PDF

\end{document} %fin du corps du document. Normalement il n'y
a rien qui suit cette commande
```

Partie
« préambule »

Partie « Corps
du document »

Chapitre 2

TeXmaker

2.1 Parties principales d'une fenêtre TeXmaker

METTRE CAPTURE D'ÉCRAN AVEC DESCRIPTION DES DIFFÉRENTES PARTIES DE LA FENÊTRE.

2.2 Paramètres et personnalisation

Chapitre 3

Commandes principales

Les commandes développées ci-dessous ne sont bien sûr pas exhaustives. Il ne s'agit là que de propositions et d'exemples afin de fournir les bases avec lesquelles débiter LaTeX. Nous vous recommandons de vous documenter sur une liste plus complètes des options possibles.

De plus, les éditeurs LaTeX proposent généralement pour écrire les commandes automatiquement sans avoir à les taper soi-même. Il s'agit généralement de boutons à cliquer ou d'options à choisir dans les menus. Nous les préciserons au fur et à mesure pour **TeXmaker**

3.1 Les types de documents

La première ligne du fichier LaTeX définit le type de document créé. Elle se présente comme ceci :

```
\documentclass{type de document}
```

Où l'on précise entre les {} le type de document en question, par exemple :

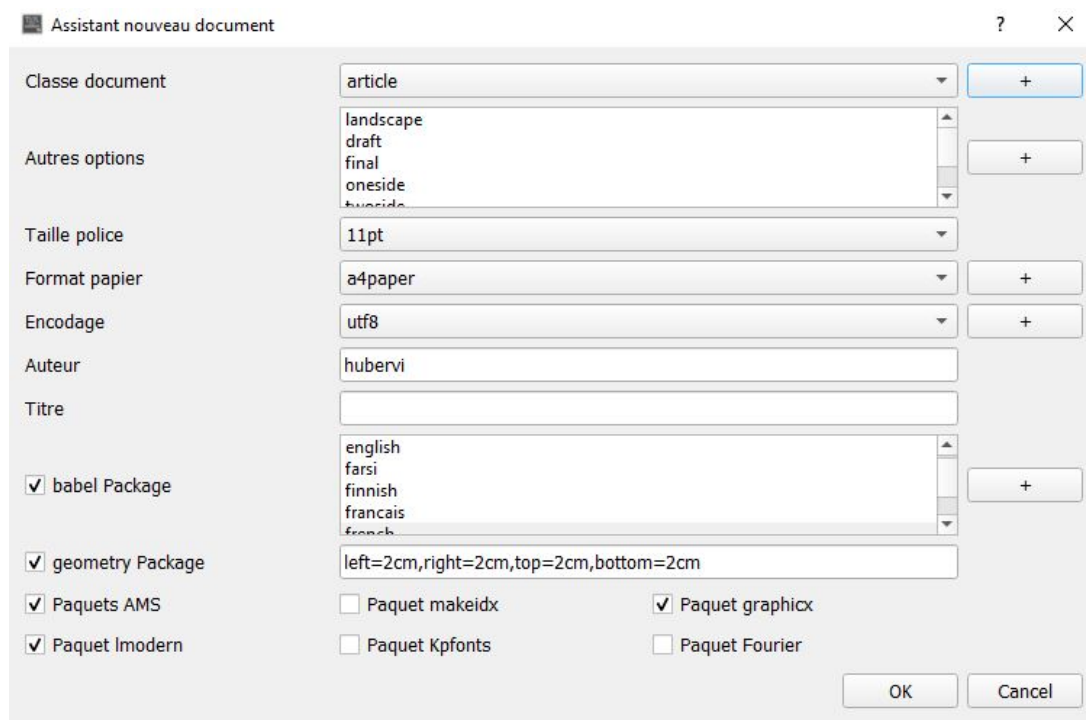
- `article`
- `report`
- `book`
- `memoir`
- `letter`
- `etc.`

Ce choix de départ peut avoir une importance car il va prédéterminer certaines disponibilités de commandes, certains choix typographiques ou de structure du document. Par exemple, il est prévu que les documents de la classe *article* soient moins longs que les documents de la classe *report* ou *book*. Ainsi, le niveau *Chapter* dans la structuration n'y est pas présent contrairement aux deux autres. De même, le texte débute sur la même page que le titre principal contrairement aux deux autres. Des changements par défaut dans la mise en page y sont également présent.

Il est tout de même possible pour l'utilisateur/trice de préciser certaines préférences de mise en page dès cette commande en les mettant entre [] après le `\documentclass`. On peut ainsi préciser la taille de la police par défaut en pt, le format de la page, si le texte est sera sur une ou plusieurs colonnes... Exemple :

```
\documentclass[10pt,a4paper]{article}
```

TeXmaker vous permet de définir toutes ces options très facilement dès le départ. Pour cela, à partir d'une nouveau fichier LaTeX vierge, cliquez sur *Assistants* → *Assistant nouveau document* → choisissez vos préférences pour chaque critère. À noter que vous pouvez déjà à ce niveau-là sélectionner certains packages que vous prévoyez d'utiliser.



3.2 Packages

Les packages sont des extensions stockées dans la distribution LaTeX (MiKTeX dans notre cas). Par défaut, votre éditeur LaTeX local n'est que très limité en termes de personnalisation ; pour résumer, vous ne pourriez écrire que du texte brut. Selon vos besoins, vous appellerez donc les packages afin de personnaliser les options de votre documents. Les distributions que nous avons conseillées d'installer comportent de très nombreux packages mais sachez que si certains s'avèrent à manquer, vous pourrez toujours les ajouter à votre distribution. Notez également que lorsque vous définissez une classe de document dans TeXMaker, celui-ci va directement ajouter les packages de base qui lui sembleront nécessaires.

Les packages s'appellent en faisant la commande suivante : `\usepackage {nom du package}`.

Vous pourrez donc les ajouter les uns à la suite des autres dans le préambule du document. Ces extensions entrent en jeu dans presque tous les paramètres de votre document. Par exemple, elles peuvent concerner :

- La mise en page (marges, entêtes, pieds de page) avec `\usepackage{geometry}` et `{fancyhdr}`
- L'encodage des polices (pour les caractères spéciaux, les accents, etc.) :
 - Pour l'encodage à l'entrée (écrire avec des accents dans le code source) : `\usepackage[utf8]{inputenc}`
 - Pour l'affichage des caractères à la sortie : `\usepackage[T1]{fontenc}`
- Coloriser la police de texte, les cellules de tableaux, etc. : `\usepackage{xcolor}`
- Utilisation des règles typographiques liées à une langue. Par exemple, ajouter un espace après : , titres des chapitres et des sections etc. : `\usepackage[french]{babel}`
- Inclusion et mise en page des images : `\usepackage{graphicx}` et `{wrapfig}`
- Pratiques des Mathématiques : `\usepackage{amsmath}`, `{amssymb}`, `{amsfonts}` et `{mathtools}`
- Gestion des liens urls : `\usepackage{hyperref}`
- Gestion avancée des citations : `\usepackage{csquotes}`
- Citations et bibliographie à partir d'un fichier de références bibliographiques .bib : `\usepackage[backend=]{}` ou `\usepackage[backend=bibtex]{natbib}`
- ETC.

3.3 Notions de base

Les exemples de commandes qui suivent se trouvent dans le corps du document, c'est-à-dire qu'elles s'appellent après la commande `\begin{document}`.

Toutes les commandes s'appellent de cette manière : `\[options] {nom de la commande}`. Exemples : `\includegraphics[width=1\width]{image.png}` pour ajouter une image faisant la largeur du texte, ou `\textit{texte en italique}` pour écrire une phrase en italique.

Il est possible d'annoter/commenter son code LaTeX en mettant au préalable un `$` suivi du commentaire.

3.4 Structure du corps du document (Chapitrage, etc.)

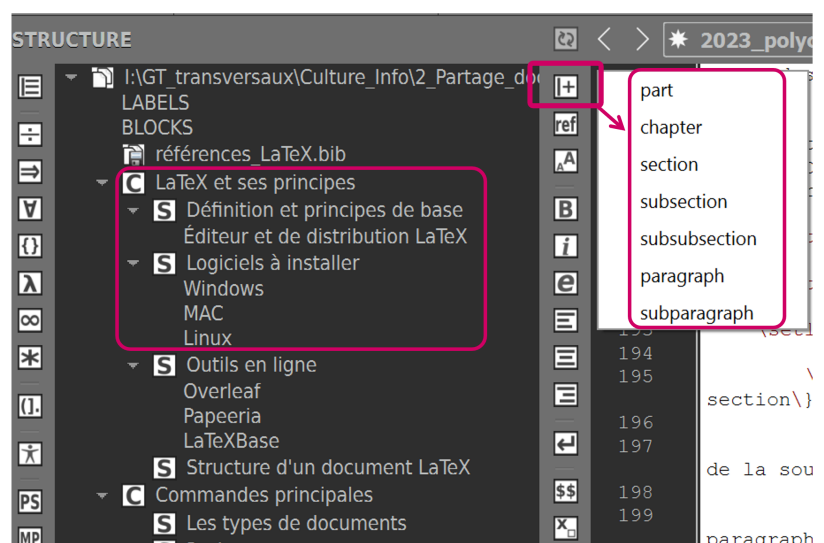
Selon la classe de document que vous aurez sélectionnée dans le préambule, vous aurez accès à diverses commandes afin de structurer le corps de votre document. Cette hiérarchisation est pratique pour vous, afin de vous retrouver aisément dans votre document mais elle est également utile pour l'éditeur LaTeX, car elle permettra de mettre en page automatiquement le texte selon cette hiérarchie, numéroté chaque partie et créera également automatiquement une table des matières. Ce système a d'ailleurs été repris dans WORD avec les "styles" (MAGUIS, 2010). La hiérarchie et les commandes des différentes parties se définit comme ceci :

Partie `\part{nom de la partie}`
Chapitre `\chapter{nom du chapitre}`
Section `\section{nom de la section}`
Sous-section `\subsection{nom de la sous-section}`
Sous-sous-section `\subsubsection{nom de la sous-sous-section}`
Paragraphe `\paragraph{nom du paragraphe}`
Sous-paragraphe `\subparagraph{nom du sous-paragraphe}`

Notez que le niveau `\chapter` n'est pas disponible pour les classes de document `article` et `letter`.

Dans TeXmaker, vous n'avez pas forcément besoin de créer ces commandes à la main. Il vous suffit de cliquer sur le bouton "+" dans la barre de gauche et de choisir le niveau hiérarchique souhaité. Nous vous conseillons également de naviguer dans votre document LaTeX via la structure hiérarchique de l'éditeur afin de vous y retrouver plus facilement.

Il est également possible de préciser à divers endroits du corps de document, les parties générales dans lesquelles nous nous trouvons, qui peuvent exiger une mise en page ou une numérotation particulière. Par exemple, `\frontmatter` pour un avant-propos, `\mainmatter` pour la partie principale du document, `\appendix` pour les annexes, `\backmatter` notamment pour la bibliographie.



Enfin, sachez que vous pouvez créer un nouveau chapitre/section, etc. sans numéro. Pour cela, il suffit d'ajouter une * après la précision du type de partie. EX : `\section* {nom de la section}`

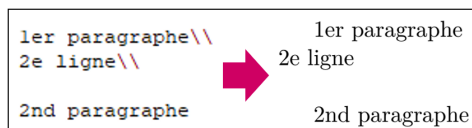
3.5 Mise en page

TeXmaker n'étant pas un éditeur WYSIWYG, vos personnalisations dans votre texte brut doivent être faites sous forme de commandes pour qu'elles soient appliquées.

Sauts de ligne, de paragraphe et de page Par exemple, taper sur la touche *enter* de votre clavier aura certes comme effet de faire un retour à la ligne dans votre éditeur de texte mais cela n'aura aucune incidence sur la mise en page du texte après compilation.

Autrement dit, il est nécessaire de préciser une commande *retour à la ligne* pour que celui-ci soit fait dans la mise en page du document final. Ainsi, un retour à la ligne se fait en ajoutant `\\` au bout de votre ligne.

Un changement de paragraphe se fait de manière plus simple en ajoutant une ligne vierge entre deux phrase dans votre éditeur ou en ajoutant la commande `\par` à la fin de votre paragraphe. Le saut de page, lui, se fait en utilisant la commande `\newpage`.



Alignement du texte LaTeX justifie par défaut les paragraphes. Ce sera donc à vous de préciser si vous préférez avec le texte aligné à gauche, droite ou encore centré. Pour cela, vous pouvez entrer les commandes

`\begin{commande}`

`\end{commande}`

Où vous préciserez `{flushleft}` pour aligner le texte à gauche, `{flushright}` à droite et `{center}` pour centrer le texte.

Notez enfin que vous avez dans la barre de commandes de TeXmaker les boutons vous permettant d'appeler facilement ces commandes

Marges Les marges sont gérées par défaut par LaTeX **selon le type de classe de document que vous avez choisi**. Vous pouvez néanmoins les personnaliser à votre aise, en appelant l'extension `geometry` dans laquelle vous précisez vos préférences.

Exemple : `\usepackage[left=2cm, right=2cm, top=1.5cm, bottom=1cm]{geometry}`

Interlignes Le package `setspace` à appeler conjugué aux commandes `\begin{onehalfspace}` ou `\begin{doublespace}` suivi de la commande `\end{doublespace}` permet de préciser des parties du texte où l'interligne sera réduit de moitié ou, au contraire, doublé.

Notes de bas de page, en-têtes et pieds de page l'insertion d'une note de bas de page se fait via la commande `\footnote{texte apparaissant en note de bas de page}` qui s'insérera au milieu de votre texte. Par défaut, l'appel de note de bas de page est un numéro en exposant. Mais sachez que vous pouvez également le personnaliser (astérisk, chiffre romain, etc.).

Pour personnaliser les en-têtes et pieds de page, utilisez le package `\usepackage {fancyhdr}`. Par la suite vous appelez la commande `\pagestyle{fancy}` pour que ce soit celui-ci qui soit pris en compte et non pas celui par défaut. Puis, précisez la commande `\fancyhf{}` pour initialiser les contenus. Enfin, vous précisez vos préférences concernant respectivement l'en-tête `\fancyhead[position]{texte à apparaître dans l'en-tête}` et le pied de page `\fancyfoot[position]{texte à apparaître`

dans le pied de page} où l'on utilise les termes L , C ou R pour préciser la position du texte. Pour résumer, voilà ce que cela donne :

```
\usepackage {fancyhdr}
\pagestyle{fancy}

\begin{document}

\fancyhf{}
\fancyhead[C]{Mon rapport sur LaTeX}
\fancyfoot[R]{\thepage}
```

Colonnes Si vous désirez écrire sur plusieurs colonnes, appelez l'extension `\usepackage{multicol}` puis vous créez un environnement avec le nombre de colonne désiré avec la commande `\begin{multicols}{nombre de colonnes}` puis `\end{multicols}`. Exemple :

```
\usepackage {fancyhdr}

\begin{document}
Texte sur une ligne entière

\begin{multicols}{2}
Texte qui apparaîtra sur deux colonnes
\end{multicols}
Texte apparaissant sur une ligne entière
```

Les citations L'extension `\usepackage[babel=true]{csquotes}` permet d'adapter vos types de guillemets aux règles de la langue utilisée (extension babel). Pour créer une citation, utilisez la commande `\enquote{texte de la citation}`.

3.6 Mise en forme du texte

Taille du texte La taille par défaut du texte est définie dès la commande `\documentclass [12pt]{classe}` à la première ligne de votre document LaTeX. A partir de là, vous pouvez accentuer ou diminuer une partie du texte via la commande `\commande_taille{texte dont la taille est modifiée}` où vous précisez la commande telle que `\small` pour petit, `\Large` pour grand, `\huge` pour très grand, etc. (Voir document de commandes pour plus de détails).

Police de caractère LaTeX fonctionne également avec des packs de polices : `\usepackage{nom_du_pack}`. Par exemple, vous pouvez appeler le pack de police `lmodern` (la police classique de LaTeX) mais également `newcent`, `charter`, etc. A vous de vous renseigner sur la police qui vous sied le mieux !

Gras, italique, etc. Voici quelques exemples de préférences de style :

```
\textbf {texte_à_modifier} = texte en gras
\textit {} = texte en italique
\texttt {} = texte style machine à écrire
\emph {} = texte avec emphase
Etc.
```

Liste Une liste basique se fait en créant un environnement sous cette forme : Ouverture d'un environnement `itemize`, lister les éléments avec la commande `\item`, puis fermeture de l'environnement `itemize`

```
\begin{itemize}
\item Pomme
\item Poire
\Etc.
\end{itemize}
```

Sachez que vous pouvez imbriquer les environnements `itemize` avec de créer des listes à plusieurs niveaux. Exemple :

```
\begin{itemize}
\item Fruits
\begin{itemize}
\item Pomme
\item Poire
\end{itemize}
\item Légumes
\begin{itemize}
\item Carotte
\item Courgette
\end{itemize}
\end{itemize}
```

Notez enfin que par défaut, le type de marqueur pour la liste est déterminé par la langue choisie dans le package Babel. Vous pouvez néanmoins les modifier individuellement pour chaque liste. Il existe également la possibilité de faire des liste numérotées. Celles-ci fonctionnent de manière identique, mais avec un environnement `enumerate`.

Couleurs Par défaut, les documents produits par LaTeX sont en noir et blanc. Vous pouvez gérer l'ajout et la personnalisation de couleurs en appelant soit l'extension `\usepackage{color}`, soit l'extension `\usepackage [option]{xcolor}`, où dans le dernier cas, vous devez sélectionner une option (par exemple `dvipsnames`). La première extension est plus limitée en nombre de couleurs disponibles mais est un peu plus facile à utiliser que le second. A partir de là, vous pouvez utiliser des commandes pour **coloriser le texte en lui-même** ou **coloriser le fond, créant un effet de surlignage**.

La commande pour coloriser le texte est `\textcolor {couleur}{Texte colorisé}`.

La commande pour coloriser le fond est `\colorbox {couleur}{Fond colorisé}`.

Vous pouvez bien sûr imbriquer ces deux commandes pour avoir un texte d'une certaine couleur dans un fond d'une autre couleur.

Notez enfin que si vous ne trouvez pas votre bonheur dans les couleurs à disposition, vous pouvez en définir à la main en amont avec la commande `\definecolor {nom de la nouvelle couleur}{code de couleur}{composition numérique}`. Exemple : `\definecolor {rose}{rgb}{1, 0.4, 0.6}`

Hyperliens Pour que votre document LaTeX gère les hyperliens qui seront cliquables, il vous faut appeler le package `\usepackage{hyperref}`. Notez que ces liens peuvent être externes, notamment vers des pages web, mais peuvent être également internes, renvoyant à d'autres parties du document en question. Une fois ce package chargé, toutes les références appelées notamment avec les commandes `\label` et `\ref`, les éléments de la table des matières, et les notes de bas de page deviendront

cliquables (TAILLET, 2022).

Vous pouvez alors utiliser la commande `\url{lien url}` pour afficher un lien cliquable, (exemple : <https://www.wikipedia.org/>, ou la commande `\href{lien url}{texte à afficher à la place du lien}` pour pointer vers une page web via un mot/texte. [EXEMPLE](#).

Notez également que, bien que ces liens soient cliquables, ils ne se démarquent pas du texte classique visuellement. Si vous désirez que le lien soit souligné et/ou qu'il s'affiche en bleu par exemple, il vous faudra inclure votre commande de lien dans une commande de forme correspondante.

3.7 Images

Pour ajouter des images dans votre document, appelez dans un premier temps le package `graphicx`. Celui-ci permet de gérer les images au format pdf, jpeg et png.

Dans un second temps, vous pourrez appeler la commande `\includegraphics [option]{adresse + nom_du_fichier sur votre ordinateur}` où les options concernent notamment l'emplacement de l'image sur la page, sa largeur (`width=...`) en cm ou vis-à-vis de la page, sa hauteur (`height=...`), sa mise à l'échelle (`scale=...`) ou son angle de rotation (`angle=...`). Par exemple :

```
\includegraphics [height=2cm, angle=20]{mon_image}
```

Ou

```
\includegraphics [width=0.5\textwidth]{mon_image}
```

3.7.1 Les éléments flottants

Pour personnaliser davantage la mise en page de vos images, notamment leur positionnement sur la page, l'enrobage ou l'ajout d'une légende, il faut enrober votre commande `\includegraphics` dans un environnement `\begin {figure}[paramètre de placement]`

`\end {figure}` où vous préciserez une des options suivantes :

- **h** = place l'élément flottant (l'image) à l'endroit où il apparaît dans le code de l'éditeur
- **p** = place l'élément flottant sur une page particulière réservée aux flottants
- **t** = place l'élément flottant en haut de la page
- **b** = place l'élément flottant en bas de la page

En plaçant le caractère **!** avant le paramètre souhaité, vous forcez LaTeX à disposer l'image comme vous le désirez même si cela va à l'encontre de ses paramètres par défaut.

Exemple

```
\begin {figure}[t]
\includegraphics [scale=0.5]{image.png}
\end {figure}
```

Légendes

Pour ajouter une légende à votre image, en précisant la commande `\caption{Texte}` dans l'environnement. Selon où vous placez la commande par rapport à la commande `includegraphics`, votre légende apparaîtra au-dessus ou en dessous de votre image.

3.7.2 L'enveloppement des éléments flottants

Vous pouvez envelopper votre image flottante de texte pour que celle-ci soit "inclue" dans un paragraphe de texte, via l'environnement `wrapfig`.

```
\begin{wrapfigure}{alignement}{largeur}
```

```
\end{wrapfigure}
```

Où l'argument `alignement` va définir si la figure sera collée à gauche (L) ou à droite (R) du texte.

Exemple

```
\begin{wrapfigure}{R}{0.5\textwidth}
\includegraphics [width=0.5\textwidth]{image.png}
\caption {Légende de mon image}
\end{wrapfigure}
```

3.8 Tableaux

La création de tableaux dans LaTeX comprend deux étapes. L'élaboration du tableau en lui-même d'une part, puis son placement en tant qu'élément flottant d'autre part. Les tableaux sont en réalité des structures où l'information est défini en lignes et colonnes. Il

Tableau sans bordure

Un tableau simple sans bordure se fait via l'environnement `tabular`. Dans cet environnement, il faut dans un premier temps définir comment sont justifié des éléments dans les cellules pour chaque colonne. Ainsi, nous spécifierons `r` pour un alignement à droite, `l` à gauche et `c` pour un alignement centré, et ce pour chaque colonne. Ensuite, il suffit de rentrer de ligne en ligne le contenu de chaque cellule, séparées par le caractère "&" afin de changer de colonne. Chaque ligne se terminera par `\\`. Exemple :

```
\begin{tabular}{ccc}
1 & 2 & 3\\
4 & 5 & 6\\
\end{tabular}
```

Ce qui donne comme tableau :

1	2	3
4	5	6

Tableau avec bordures

Le tableau avec bordures pour colonnes et/ou ligne fonctionne avec le même principe mais nous devons définir quelle(s) colonne(s) et quelle(s) ligne(s) doivent avoir des bordures.

Colonnes

Préciser une bordure de colonne se fait dans la commande de l'environnement, dans la partie `{ccc}`. Dans celle-ci, il faudra ajouter un "|" entre chaque précision de justification. Ainsi, `{ccc}` deviendra `{|c|c|c|}`. Nous pouvons préciser des largeur de colonnes en ajoutant une indication pour chacune dans la commande ci-dessus. Exemple : `{|c{2cm}|c{1cm}|c{3cm}|}`

Lignes

Contrairement aux colonnes, l'indication des bordures de lignes ne se fait pas dans la commande d'environnement mais directement entre les lignes de texte du tableau, avec la commande `\hline`. Il

suffit alors de l'insérer dans chaque ligne où nous désirons une bordure.

Exemple pour colonnes et lignes :

```
\begin{tabular}{|c|c|c|}
\hline
1 & 2 & 3\\ [1cm]
\hline
4 & 5 & 6\\
\hline
\end{tabular}
```

1	2	3
4	5	6

Ce qui donne comme résultat :

Notez également, dans l'exemple ci-dessus, comme il est possible de préciser la hauteur d'une ligne en ajoutant une taille après les `\\` de la ligne en question.

Environnement flottant

Créer un tableau flottant, c'est-à-dire un tableau qu'il est possible de déplacer dans le document à sa guise, se fait en insérant l'environnement `tabular` dans un environnement qui est, lui, flottant, et dont le nom est `table`. Le fonctionnement de ce dernier est similaire à celui de `figure` vu précédemment pour les images. Il faut donc préciser dans cet environnement-ci sa position sur la page avec les options `center`, `left` ou `right` avant de créer le tableau lui-même ainsi que la commande `caption` pour ajouter une légende. Exemple :

```
\begin{table}
\begin{center}
\begin{tabular}{|c|c|c|}
\hline
1 & 2 & 3\\ [1cm]
\hline
4 & 5 & 6\\
\hline
\end{tabular}
\caption{Ce beau tableau flottant m'apprend à compter}
\end{table}
```

Ce qui va donner :

1	2	3
4	5	6

TABLE 3.1 – Ce beau tableau flottant m'apprend à compter

Autres paramétrages de tableau

Fusion de colonnes ou de lignes

Colonne la commande de fusion de colonne est la suivante :

`\multicolumn{nombre de colonnes fusionnées}{|alignement|}{texte de la cellule}`. Cette commande **doit être précisée pour chaque ligne du tableau** lorsque l'on précise le contenu de chaque cellule de la ligne (donc dans la partie où nous trouvons en théorie `1 & 2 & 3 \\`). Exemple :

```
\begin{tabular}{|c|c|c|}  
\hline  
1 & \multicolumn{2}{c|}{2} \\  
\hline  
3 & 4 & 5 \\  
\hline  
\end{tabular}
```

Ce qui donne :

1	2	
3	4	5

Ligne La fusion des lignes se fait via la commande `multirow` contenue dans le package du même nom. En théorie, cela donne ça : `\multirow{nombre de lignes fusionnées}{largeur de la colonne comprenant les lignes fusionnées}{texte *potentiel" de la cellule fusionnée}`.

La commande se place dans la cellule de la première ligne à fusionner. Les lignes suivantes, selon le nombre de lignes fusionnées précisé, resteront vides dans le script (GAUTRELET, 2020). Exemple :

```
\usepackage{multirow}  
...  
\begin{tabular}{|c|c|c|}  
\hline  
\multirow{2}{*}{1} & 2 & 3 \\  
& 4 & 5 \\  
\hline  
6 & 7 & 8 \\  
\hline  
\end{tabular}
```

Donnant le résultat :

1	2	3
4	5	
6	7	8

Où les deux premières lignes sont fusionnées. Vous remarquez que dans la première cellule, le chiffre 1 prend la place de toute la cellule fusionnée. Cela est le cas parce que nous l'avons mis entre `{}` dans la commande de la fusion, alors que les chiffres 2, 3, 4 et 5 sont indiqués de manière normales dans la ligne. Pour autant ces dernières cellules sont également fusionnées, mais le texte, lui, est distinct.

Coloriser le tableau

La colorisation de cellules, lignes ou colonnes peut être utile pour mettre en évidence un élément en particulier ou des entêtes au tableau. On commence par appeler l'extension `colortbl`

Cellule Pour modifier la couleur de fond d'une cellule, il faut utiliser la commande `\cellcolor{couleur}` qui sera placée juste avant le texte, dans la cellule du tableau.

Pour changer la couleur de police d'une cellule, le principe est le même en faisant cette fois-ci la commande `\color{couleur}`. Voici un exemple des deux commandes conjuguées :

```
\usepackage{colortbl}
...
\begin{tabular}{|c|c|c|}
\hline
1 & \cellcolor{yellow}2 & 3\\
\hline
4 & \color{red}5 & 6\\
\hline
\end{tabular}
```

Donnant le résultat :

1	2	3
4	5	6

Ligne Pour changer la couleur de fond de toute une ligne, il faut utiliser la commande suivante en début de ligne : `\rowcolor{couleur}`. Exemple :

```
\{blue} 1 & 2 & 3 \\\
```

Colonne Le changement de couleur de fond d'une colonne se fait dans les options du tableau, juste avant l'argument correspondant à la colonne :

```
>\columncolor{couleur}}
```

Et pour changer la couleur de la police :

```
>\color{couleur}}
```

Exemple :

```
\begin{tabular}{|>\columncolor{magenta}}c|c|>\color{cyan}}c|}
\hline
1 & 2 & 3\\
\hline
4 & 5 & 6\\
\hline
\end{tabular}
```

Donnant comment résultat :

1	2	3
4	5	6

3.9 Équations Mathématiques

L'écriture des mathématiques se fait également via des environnements. Nous allons indiquer une zone dans laquelle le compilateur lira une formule mathématique à la place de texte normal. Sachez qu'il vous est possible de suivre les commandes proposées dans les sections suivantes afin de créer à la main ces équations. **Mais certaines options dans TeXmaker ainsi que des outils en ligne ou à installer vous permettent de créer des commandes d'équations LaTeX complexes à partir de formules mathématiques que vous créez en WYSIWYG.** Nous vous conseillons de les utiliser afin de vous faciliter le travail. Voir plus bas pour certains de ces outils.

Intégration d'une formule dans une phrase Si l'on veut intégrer une formule au milieu de texte, il suffit d'utiliser `$...$`. Exemple : Ceci est une équation $(a+b)^2 = a^2 + 2ab + b^2$ au milieu d'une

phrase.

Mettre en évidence une formule Pour mettre en évidence une équation, il faut l'écrire entre `\[... \]` ou `$$... $$`. Cette commande fera aller à la ligne et centrera la formule :

$$(a + b)^2 = a^2 + 2ab + b^2$$

Equations numérotées L'environnement `\begin {equation} ... \end {equation}`. Cela permet de numéroté les équations, dans le but d'y faire référence plus tard dans votre document. Exemple :

$$f(x) = a(c)^x \quad (3.1)$$

Ajouter du texte dans une équation Pour ajouter une info texte au milieu d'une équation mathématique, insérer la commande `\text { }` à l'intérieur de votre commande de formule.

Commandes de base des fonctions mathématiques

En premier lieu, il est nécessaire d'appeler les packages `amsmath` et `amssym`. Un autre package nommé `mathtools` permet d'étendre encore davantage les options. Nous vous conseillons donc d'appeler pour tous vos documents ces trois extensions afin d'avoir des options exhaustives. D'une manière générale, les commandes se feront sous la forme suivante où l'on précise un exposant et un indice :

`\commande_{en-dessous}^{au dessus}`

Exposant et indice L'exposant s'exprime via l'accent circonflexe $\hat{}$. L'indice, lui, s'écrit via l'underscore $_$: `(U_n) = 4^{\hat{n}} + 3.14^{\{n+1\}}` donnant comme équation $U_n = 4^n + 3.14^{n+1}$

Fraction Une fraction se fait avec la commande `\frac {numérateur}{dénominateur}`. Ex : `\frac {2}{5}` donnant $\frac{2}{5}$

Fractions en cascade Avec le package `amsmath`, vous pouvez créer des fractions imbriquées en utilisant la commande `\cfrac`. Exemple. `\frac{2}{5 + \cfrac{6}{3}}`, donnant $\frac{2}{5 + \frac{6}{3}}$

Racine `\sqrt[3]{4}` $\rightarrow \sqrt[3]{4}$

Limite `\lim_{1 \rightarrow n} x+42` $\rightarrow \lim_{1 \rightarrow n} x + 42$

Sommes `\sum_{k=0}^p U_k` $\rightarrow \sum_{k=0}^p U_k$

Intégrales `\int_0^{\infty} f(x)dx` $\rightarrow \int_0^{\infty} f(x)dx$.

Notez qu'il est possible d'imbriquer les commandes d'intégrales afin de créer des intégrales doubles ou triples.

Produit `\prod_{k=2}^{47} k+1` $\rightarrow \prod_{k=2}^{47} k + 1$

Accolades (ensembles) Créer un environnement `\cases`, p. ex :

```
$
\begin{cases}
f(x) = 3x+2\\
f(x) = 4x^4 + 12x^3 - 2x^2 + x-5
\end{cases}
$
```

Donnant le résultat : $\begin{cases} f(x) = 3x + 2 \\ g(x) = 4x^4 + 12x^3 - 2x^2 + x - 5 \end{cases}$

Fonctions

Les fonctions telles que cos, sin, tan, etc. se font via une commande simple propre à chacune.

Exemple : `\arctan^{42}(x)` $\rightarrow \arctan^{42}(x)$

Voir les feuilles annexes pour toutes les fonctions usuelles.

Matrices

Une matrice se crée via un environnement classique selon une commande propre à chaque forme de matrice (parenthèses, crochets, etc.) et se remplit comme un tableau en utilisant des "& " pour séparer les éléments des différentes colonnes. Exemple :

```
$
\begin{pmatrix}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{pmatrix}
$
```

Donnant $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$

```
\begin{vmatrix}
1 & 2 & 3 \\
4 & 5 & 6
\end{vmatrix}
```

```
\begin{Vmatrix}
1 & 2 & 3 \\
4 & 5 & 6
\end{Vmatrix}
```

```
\begin{bmatrix}
1 & 2 & 3 \\
4 & 5 & 6
\end{bmatrix}
```

```
\begin{Bmatrix}
1 & 2 & 3 \\
4 & 5 & 6
\end{Bmatrix}
```

Vecteurs et barres

Pour une flèche apparaissant au dessus d'une seule lettre (ex. \vec{v}), il est possible d'utiliser la commande `$ \vec{lettre} $`, mais si la chaîne de caractères sous la flèche est plus longue, il vous faudra utiliser la commande `\overrightarrow{mot}` (ex. \overrightarrow{AB}).

On retrouve un binôme de commandes analogue pour placer une barre au-dessus des lettres, avec respectivement les commandes `$ \bar{lettre} $` (ex. \bar{x}) et `\overline{mot}` (ex. \overline{AB}).

3.9.1 Lettres grecques et autres symboles

Globalement, les lettres grecques (ex. α , λ , Ω ...) sont représentées par des commandes portant le nom de la lettre (ex. `\alpha`, `\lambda`, `\Omega`...). Pour avoir la lettre en majuscule, il faut mettre l'initiale en majuscule (ex. `\Omega`). Pour avoir une liste complète, voir feuille de commandes.

Le principe est le même pour les autres symboles mathématiques tels que \neq , \approx , ∞ , etc. Chacun a sa commande propre.

3.9.2 Affichage et alignement des équations

Nous avons vu précédemment qu'il était possible soit d'inclure une équation dans une ligne classique, au milieu du texte, ou de la mettre en évidence sur une ligne à part entière. Ces options fonctionnent correctement telles quelles mais peuvent avoir quelques désagréments. Il existe donc des options *displaystyle* des commandes mathématiques pour améliorer cela. Premièrement, lorsqu'une équation est comprise dans une ligne de texte, celle-ci a par défaut la hauteur définie dans l'interligne et pas plus. Ce qui fait que selon le type de formule mathématique, par exemple une fraction ou une fraction en cascade, l'affichage de l'équation sera très (trop ?) compact, et donc difficile à lire. Pour corriger ce problème, il est possible d'utiliser la commande `\dfrac{}{}` pour "display frac" offrira un affichage plus aéré, à la place de `\frac{}{}`, ce qui donne $\frac{2}{5}$ au lieu de $\frac{2}{5}$. Notez qu'à la place de `\dfrac`, vous auriez pu utiliser la commande `\displaystyle \frac{}{}`, cela serait revenu au même !

De même, la commande `\displaystyle` peut être utilisée lorsque vous affichez plusieurs équations les unes en dessous des autres, même lorsqu'elles sont mises en évidence. Ainsi, le code suivant :

```
$$
\begin{cases}
\displaystyle f(x) = \sqrt{\ell^2 + x^2} & \text{lorsque } x \geq 0 \\
\displaystyle f(x) = \frac{1}{1+x} & \text{sinon}
\end{cases}
$$
```

Donne une équation aérée :

$$\begin{cases} f(x) = \sqrt{\ell^2 + x^2} & \text{lorsque } x \geq 0 \\ f(x) = \frac{1}{1+x} & \text{sinon} \end{cases}$$

Plutôt que la suivante sans le `displaystyle` qui est "ramassée" :

$$\begin{cases} f(x) = \sqrt{\ell^2 + x^2} & \text{lorsque } x \geq 0 \\ f(x) = \frac{1}{1+x} & \text{sinon} \end{cases}$$

Cette personnalisation de l'affichage des équations se poursuit concernant l'alignement de celle-ci. Par défaut, les équations sont alignées au centre de la page. Néanmoins, l'environnement `eqnarray` pour les équations numérotées et `eqnarray*` non numérotées permettent de centrer les équations tout en alignant les éléments de celles-ci. L'alignement se fait via les caractères "&" qui servent de repaire comme pour les colonnes des tableaux. L'environnement `eqnarray` permet également de faire un *regroupement* d'équations, formant ainsi un seul bloc plutôt qu'une suite d'équations indépendantes les unes des autres. Exemple :

$$\begin{array}{lcl} y & = & x - y + z \\ x & = & y \\ z & = & y \end{array}$$

Le résultat ci-dessus paraît plus propre et plus lisible que des listes d'équations centrée au milieu de la page comme celles-dessous :

$$y = x - y + z$$

$$x = y$$

$$z = y$$

3.9.3 Aides dans TeXmaker

Vous n'aurez pas à connaître les commandes ci-dessus par coeur. TeXmaker propose un panel de commandes préformattées pour les formules mathématiques ainsi que pour les lettres et symboles directement dans la barre d'outils sur la gauche. Il vous suffit de cliquer sur l'onglet correspondant pour rechercher et utiliser l'objet voulu :

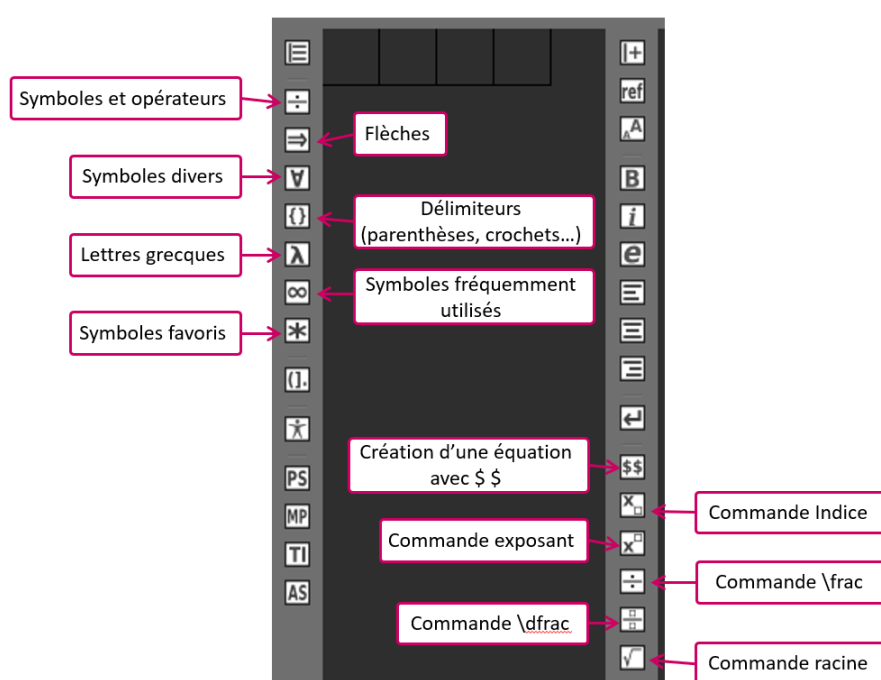


FIGURE 3.1 – Raccourcis pour symboles et commandes

3.9.4 Générateurs d'équations en ligne

Comme nous l'avons dit précédemment, il ne vous sera pas forcément nécessaire de créer vos équations en commandes et environnement LaTeX. Il existe des générateurs d'équation en ligne gratuits + ou - complets vous permettant de créer vos formules et dont le code LaTeX se créera en parallèle. Par exemple, vous pourrez tester [ce LaTeX Equation Editor](#), [cet autre LaTeX Equation Editor](#) ou encore [celui-ci](#). Le principe est le même : vous utilisez les options pré-formatées pour créer votre équation, et les commandes LaTeX vous sont présentées, prêtes à être copiées-collées dans votre éditeur. Exemple :

3.10 Citer ses sources et créer une bibliographie

La citation de références et la création de bibliographie dans un fichier LaTeX se font à l'aide d'un fichier aux formats **BibTeX** ou **BibLaTeX** (les deux ont l'extension .bib) qui contient les références

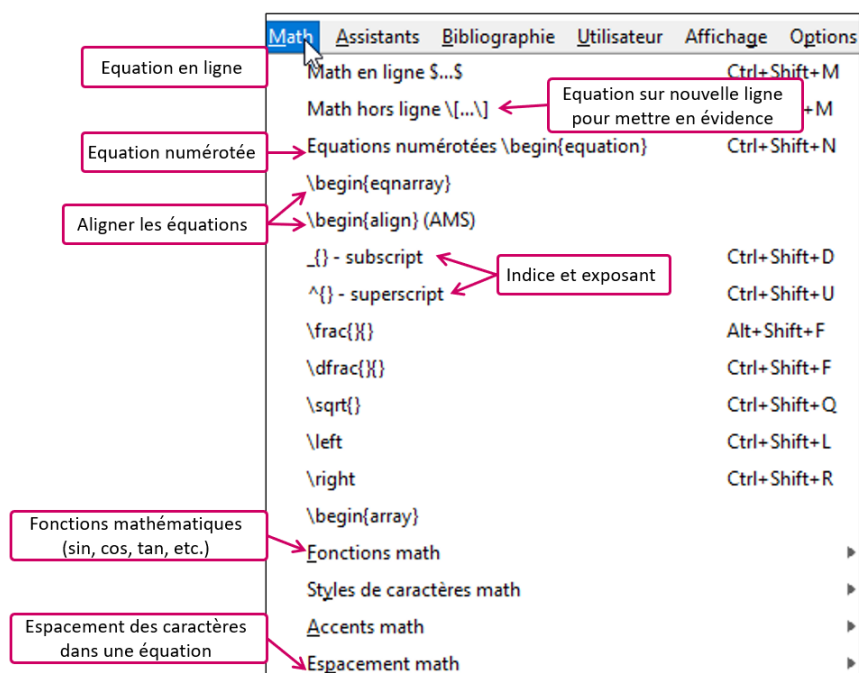


FIGURE 3.2 – Raccourcis et commandes dans le menu

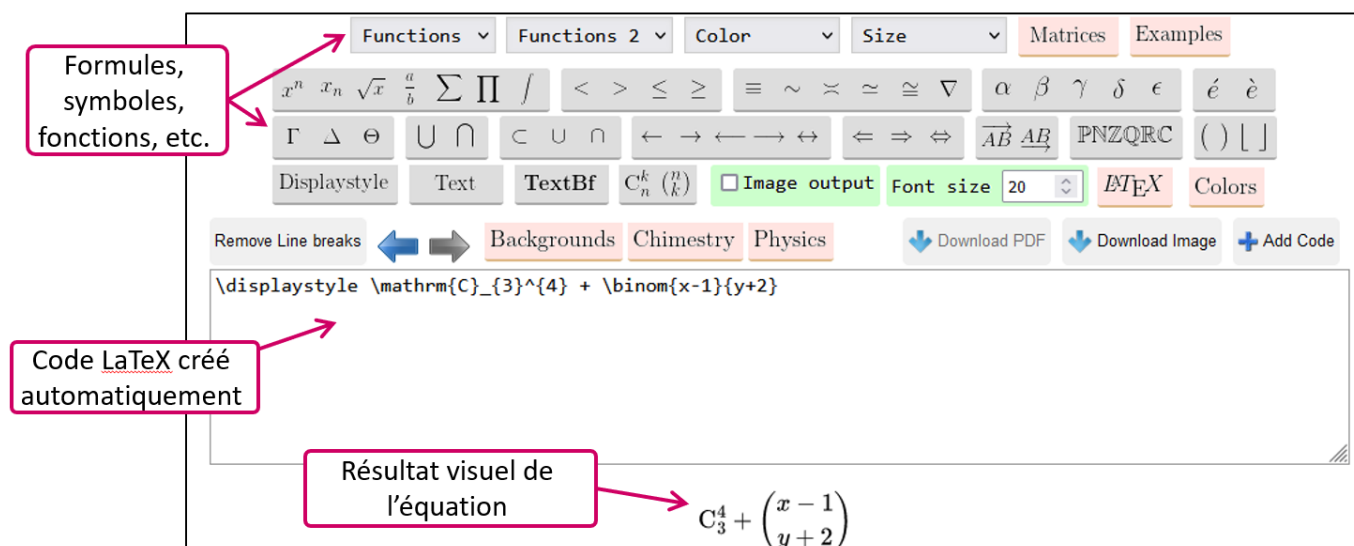
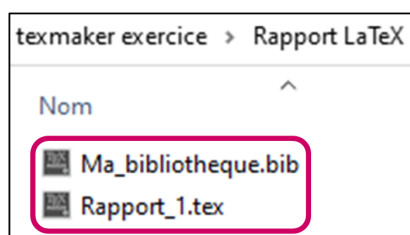


FIGURE 3.3 – Exemple d'un éditeur d'équation en ligne proposé

que lon souhaite citer. Ce fichier .bib, que lon va á appeler á chaque fois que lon cite une référence, est sauvegardé dans le même dossier que le fichier LaTeX utilisé. Ainsi, le dossier doit, au minimum, comporter ces deux fichiers (dautres éléments sy ajouteront éventuellement). La personnalisation

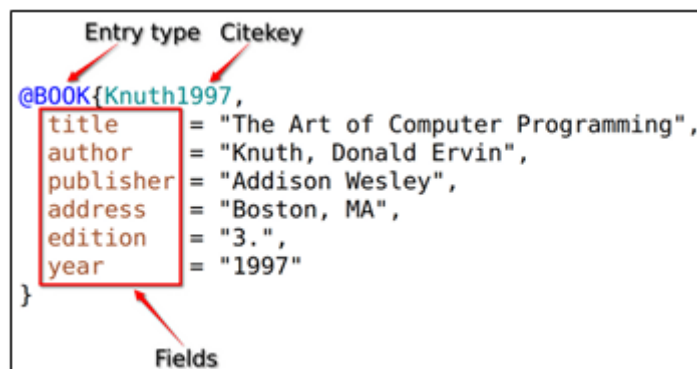


et les fonctions avancées des méthodes de citation dans LaTeX sont nombreuses mais exigerait un document propre à ce sujet. Nous nous intéresserons ici qu'aux principes de base. Veuillez donc vous

référer au polycopié complet sur ce sujet pour avoir davantage d'informations.

3.10.1 Structure d'une référence bib

Les références dans le fichier .bib ressemblent à cela : La première ligne correspond au type



```
@BOOK{Knuth1997,  
  title    = "The Art of Computer Programming",  
  author   = "Knuth, Donald Ervin",  
  publisher = "Addison Wesley",  
  address  = "Boston, MA",  
  edition  = "3.",  
  year     = "1997"  
}
```

The diagram shows a BibTeX entry for a book. Annotations include: 'Entry type' pointing to '@BOOK', 'Citekey' pointing to '{Knuth1997', and 'Fields' pointing to the list of key-value pairs. The key-value pairs are enclosed in a red box.

de document dont il s'agit (@book, @article, @phdthesis, etc.) accompagné de la clé de citation (Citekey), c'est-à-dire l'identifiant unique de la référence qui sera appelé dans le fichier LaTeX lorsque l'on désirera citer cette référence. Les lignes suivantes correspondent aux champs d'information de la référence. On y trouve notamment le champs title, author, journal, volume (volume d'une revue), number (info additionnelle, p.ex. issue numéro d'une revue), pages, date, etc.

3.10.2 Différences entre le format BibTeX et BibLaTeX

Un fichier bibliographique .bib peut être créé dans deux formats : BibTeX ou BibLaTeX. Il est important de choisir le bon type de format dès le départ car chacun des deux aura ses avantages et inconvénients et devra être utilisé avec des packages et des moteurs de compilation propre à chacun. Voici ci-dessous, en résumé, le processus pour chacun des deux :

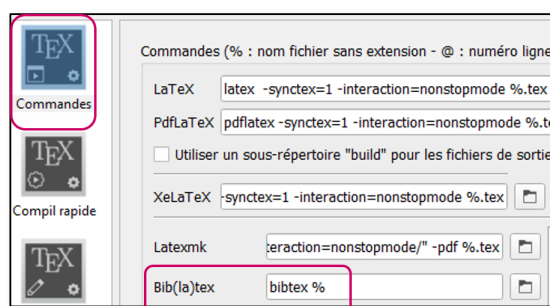
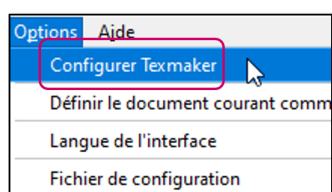
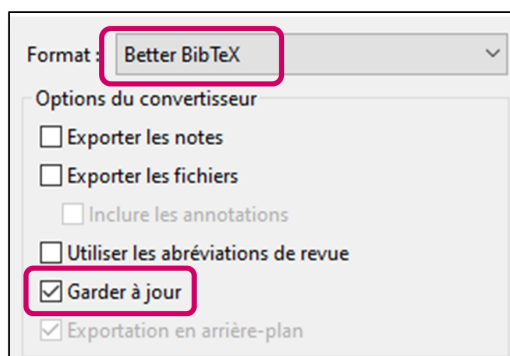
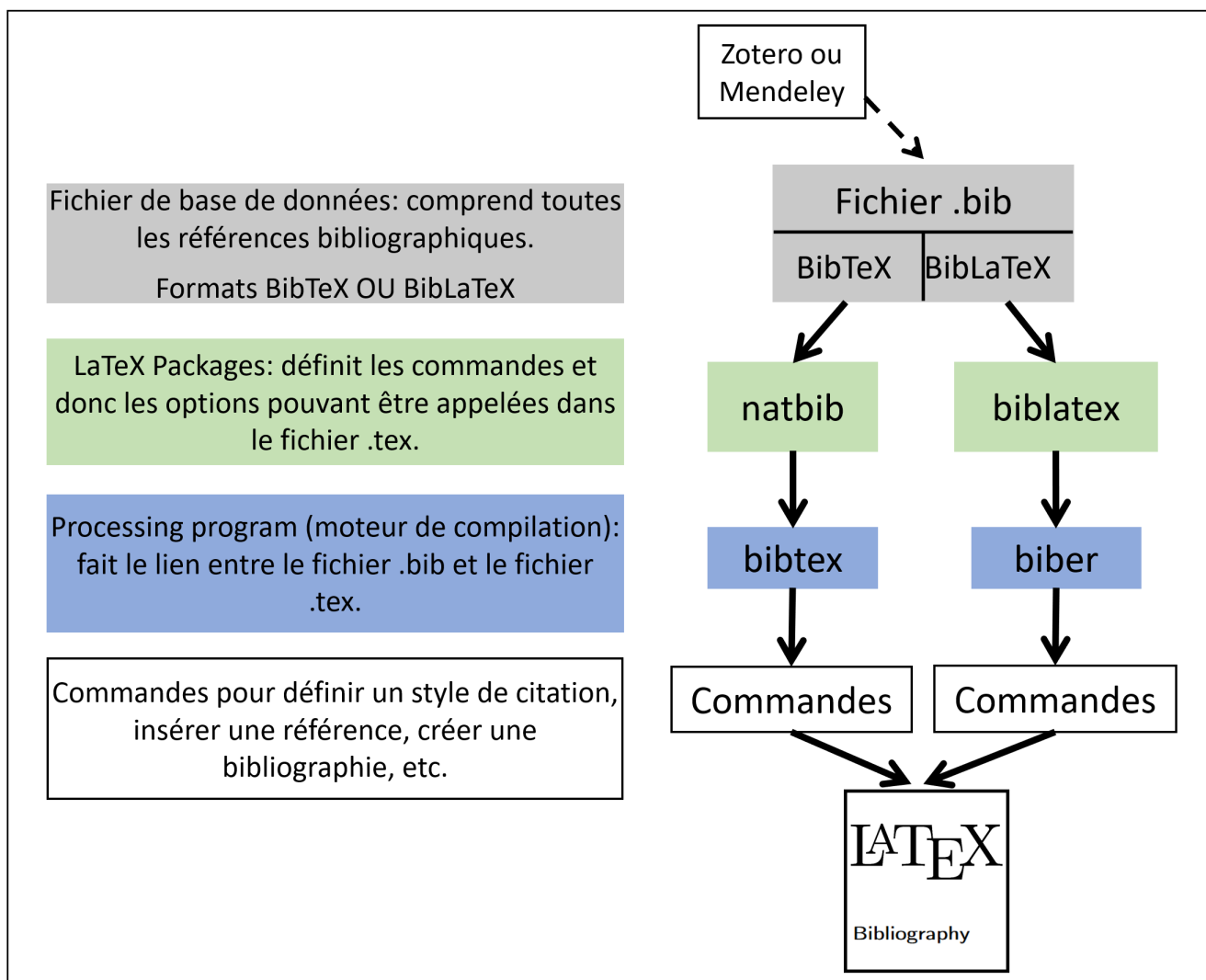
3.10.3 Création du fichier bib depuis Zotero

Une fois que l'on a décidé quel format de fichier bibliographique privilégier, il suffit d'aller dans Zotero pour sélectionner les références ou une collection entière et faire un clic-droit *Exporter la collection / les documents* choisir le format BibTeX ou BibLaTeX. Ce procédé très simple a néanmoins un inconvénient, qui est que le fichier créé est statique, c'est-à-dire qu'il ne sera pas mis à jour automatiquement si la collection est complétée dans Zotero. Il faudra recréer et écraser le fichier .bib pour qu'il soit à jour. Pour corriger cet inconvénient, nous vous conseillons d'installer le plugin *BetterbibTeX pour Zotero* qui permet de créer des fichiers bibliographiques dynamiques :

3.10.4 Préciser les packages et les options dans TeXmaker

Il est nécessaire ensuite de préciser ses options dans TeXmaker. Depuis l'onglet *options* puis *Configurer TeXmaker*, dans l'onglet *commandes*, sélectionnez soit *bibtex %*, soit *biber %* selon si vous avez choisi BibTeX ou BibLaTeX : Vous appelez par la suite un des deux packages :

- `\usepackage[backend=bibtex]{natbib}` pour le format BibTeX
- `\usepackage[backend=biber]{biblatex}` pour le format BibLaTeX



3.10.5 Principes de citation et de création de bibliographie

A partir de là, lorsque vous voudrez citer une référence, vous appellerez la clés de citation avec une commande de citation telle que `\cite{citation key}` ou `\parencite {citation key}` par exemple.

Enfin, pour écrire automatiquement la bibliographie, il vous suffit d'utiliser une des deux commandes `\bibliography` ou `\printbibliography` à l'endroit désiré. A partir de là, lorsque vous voudrez citer une référence, vous appellerez la clés de citation avec une commande de citation telle que `\cite{citation key}` ou `\parencite {citation key}` par exemple.

Enfin, pour écrire automatiquement la bibliographie, il vous suffit d'utiliser une des deux commandes `\bibliography` ou `\printbibliography` à l'endroit désiré.

Chapitre 4

Beamer : les fichiers de présentation LaTeX

Chapitre 5

Outils complémentaires

5.1 Templates - Modèles

Les templates sont des modèles de fichiers LaTeX pré-formatés en termes de structures et de contenu qu'il est possible de télécharger et d'utiliser tels quels ou de s'en inspirer pour créer des documents adaptés à ses besoins. Ces templates peuvent être proposés par des institutions officielles (par exemple template officiel pour un mémoire de Master), par des éditeurs, par des portails web, etc. En voici quelques exemples :

- [Templates proposés par Overleaf](#)
- [Modèle non officiel de thèses à l'EPFL](#)
- [Modèle de thèse à l'Université de Fribourg en trois langues disponibles](#)
- [Template d'article de la revue PloS One](#)

Notez qu'il n'existe pas, à notre connaissance, de modèles officiels créés par et pour l'UNIGE. Nous vous recommandons néanmoins de rechercher et regarder des modèles existants ailleurs car cela peut grandement vous faciliter le processus de rédaction et vous faire gagner un temps important. De plus, cela vous permettra également de vous inspirer d'options LaTeX plus avancées afin de rendre vos documents plus professionnels.

5.2 MathPix

[MathPix](#) est un outil en ligne permettant de créer en un clic une équation mathématique en LaTeX à partir de textes ou documents. Concrètement MathPix vous permet de faire une capture d'écran d'une formule mathématique et celui-ci va reconnaître automatiquement son contenu et va créer une équation LaTeX correspondante. Cela permet de gagner un temps important si d'aventure vous devez créer des équations longues et complexes à créer à la main. La version gratuite de MathPix permet seulement 20 captures d'écran avec un compte institutionnel mais une version payante à 5CHF/mois est également proposée. Notez qu'il est également possible de lui faire créer un document LaTeX à partir d'un fichier pdf de 20 pages pour la version gratuite et 500 pages pour la version payante !

5.3 CTAN Comprehensive TeX Archive Network

[CTAN](#) est un portail permettant de télécharger et installer un grand nombre d'outils liés à LaTeX, notamment des packages supplémentaires non inclus dans votre suite, des macros pré-fabriquées, polices, etc.

Bibliographie

GAUTRELET, E. (2020). *Introduction à LaTeX*. Dunod.

GERIN, L., PRIVAT, R., & PRIVAT, Y. (2008). Petit guide pour les d ebutants en LATEX.

MAGUIS, N.-A. (2010). *R edigez des documents de qualit e avec LATEX : l'outil des professionnels pour publier m emoires, th eses, rapports, articles scientifiques...* Simple IT.

TAILLET, R. (2022). *Bien d ebuter en LaTeX*. De Boeck Sup erieur.