

# Matière : Logiciels libres et open source - Cours 04 : Initiation à Latex

Il existe plusieurs catégories de logiciels pour gérer du texte.

## **Éditeur de texte : (Bloc-notes, Notepad++ ...)**

- Destiné à la création et à l'édition de fichiers textes sans mise en forme des polices, des couleurs, de justification ...
- Les fichiers produits sont légers et exploitables par d'autres éditeurs de texte.

## **Traitement de texte : (Microsoft Word, OpenOffice ...)**

- Produit une composition du texte à partir des données textuelles.
- Le format de sauvegarde retient les informations de composition : mise en forme, couleurs, polices ...
- Les fichiers sont plus gros.
- Les fichiers sont souvent moins exploitables par d'autres logiciels.

## **Logiciel de mise en page : (Adobe InDesign, QuarkXpress ...)**

- Réalise une composition de très grande qualité typographique à partir de textes et d'images.
- Logiciels haut de gamme, donc souvent chers.
- Apprentissage difficile, souvent réservés à des professionnels

## **1. Latex**

LaTeX est un logiciel de composition de documents gratuit, professionnel et multiplateforme qui permet de produire des documents typographiques de haute qualité. Il facilite l'intégration de formules mathématiques, de graphiques, de diagrammes, etc., dans un document.

Le logiciel ou le langage de composition de documents textuels Latex est devenu la méthode privilégiée d'écriture de documents (articles, rapports, livres, etc.) dans la communauté scientifique, en particulier du côté des sciences exactes et appliquées.

## **2. Latex, LibreOffice Writer ou Microsoft Office Word ?**

## 2.1. Présentation rapide de chaque outil

Logiciel	Type	Utilisation principale	Public cible
Microsoft Word	Logiciel propriétaire (Microsoft)	Rédaction bureautique (rapports, lettres, CV, etc.)	Grand public, entreprises
LibreOffice Writer	Logiciel libre (suite LibreOffice)	Rédaction bureautique (rapports, lettres, CV, etc.)	Utilisateurs de logiciels libres
LaTeX	Langage de composition de documents	Rédaction scientifique, technique, académique	Chercheurs, étudiants, ingénieurs

## 2.2. Mode de fonctionnement

Aspect	MS Word / LibreOffice Writer	LaTeX
Interface	WYSIWYG ( <i>What You See Is What You Get</i> ) : on voit le rendu final à l'écran.	Basé sur du <b>code</b> : on écrit du texte avec des balises de formatage (ex. <code>\section{}</code> , <code>\textbf{}</code> ) et on compile le document pour obtenir un PDF.
Format de fichier	.docx, .odt, etc.	.tex (puis PDF généré)
Mise en page	Manuelle : l'utilisateur ajuste la forme (polices, marges, styles).	Automatique : LaTeX gère la mise en page selon des règles typographiques très strictes.
Facilité d'utilisation	Intuitive pour les débutants.	Nécessite un apprentissage initial du langage.

## 2.3. Principales différences

Critère	Word / Writer	LaTeX
Apparence	L'utilisateur voit et modifie directement la mise en forme.	L'utilisateur écrit le contenu, LaTeX s'occupe de la mise en page.
Stabilité des grands documents	Word/Writer devient lent ou instable avec de gros fichiers (thèses, rapports de 200+ pages).	LaTeX gère parfaitement les grands documents.
Formules mathématiques	Éditeur basique (Word) ou limité (Writer).	Gestion parfaite des équations complexes avec la syntaxe mathématique LaTeX.
Collaboration et versionnage	Difficile à suivre dans Word (suivi des modifications limité).	Compatible avec Git (comme du code), donc parfait pour le travail collaboratif, et il offre aussi une version en ligne ( <a href="https://www.overleaf.com/">https://www.overleaf.com/</a> )
Compatibilité	Formats propriétaires (Word) → problèmes de compatibilité parfois.	Format texte brut → lisible sur toutes les plateformes.

Critère	Word / Writer	LaTeX
Personnalisation / automatisation	Limitée, dépend de l'interface.	Très flexible : on peut automatiser presque tout via macros et packages.

## 2.4. Avantages et limites

### Avantages majeurs :

#### 1. Qualité typographique professionnelle :

LaTeX respecte les règles de mise en page (espacement, césure, polices, alignement, marges) comme une maison d'édition.

#### 2. Parfait pour les documents scientifiques et techniques :

Intégration native des **formules mathématiques**, **figures**, **tableaux**, **bibliographies**, **citations** et **références**.

#### 3. Structure claire :

Le contenu est séparé de la présentation → idéal pour maintenir des documents cohérents.

#### 4. Automatisation puissante :

Génération automatique de la table des matières, des listes de figures, d'index, des bibliographies, etc.

#### 5. Durabilité et compatibilité :

Les fichiers `.tex` sont du texte brut — lisibles et portables dans le temps (pas de problème de compatibilité entre versions).

#### 6. Versionnement facile :

On peut suivre les changements avec **Git** comme dans un projet de code source.

### Limites de LaTeX

- Nécessite **un apprentissage** du langage et de la structure (balises, compilation, packages).
- Peu adapté à la rédaction rapide de documents simples (lettres, CV, etc.).
- Pas d'aperçu direct (on doit compiler pour voir le rendu).

### 3. Composition d'un document

La rédaction d'un document avec LATEX consiste à écrire le code source du document. Il s'agit simplement d'un fichier texte, dont l'extension est .tex. Une fois ce fichier écrit, on le compile à l'aide du compilateur pdflatex pour obtenir un document final, au format PDF.

### 4. Installation

Pour pouvoir rédiger des documents en LATEX, deux éléments essentiels sont à avoir sur ton ordinateur :

- Une distribution : c'est une collection de logiciels (des compilateurs, des convertisseurs, etc.) et de packages (fournissant des extensions à LATEX) ;
- Un éditeur : est un programme qui permet de rédiger le code source des documents et propose des outils pour aider à la rédaction du document, notamment un correcteur orthographique.

Les distributions les plus courantes : MiKTeX, TeXLive.

Les éditeurs les plus courants : WinEdt, TeXstudio.

Il existe des logiciels LaTeX qui offrent ces deux éléments simultanément : la distribution et l'éditeur, par exemple : TeX Studio.

### 5. Vocabulaire

- **Préambule** : Le préambule est la première partie de tout document LATEX. Il permet de spécifier plusieurs informations très importantes, comme par exemple le type de document à créer (rapport, livre...) ou encore les différents packages (extensions et fichiers de style) que l'on souhaite utiliser dans le document. On renseigne également dans le préambule le titre du document, l'auteur, la date ...
- **Fichier source** : Le fichier source (ou document LATEX), contient le texte du document et les commandes de mise en page. C'est un fichier texte, écrit avec n'importe quel éditeur de texte, avec l'extension .tex.
- **Commande** : Ce sont les instructions de base de LATEX qui permettent de changer des paramètres de mise en forme du texte, d'insérer un tableau ou une image. Les commandes sont identifiées par des noms qui commencent tous par un \ (backslash).

- **Environnement** : Les environnements sont des commandes spéciales qui modifient un gros blocs de texte. Ils permettent de mémoriser une mise en forme de texte et de l’appliquer ainsi à plusieurs endroits du documents.

Les environnements commencent presque toujours par `\begin{environnement}` et se terminent par `\end{environnement}`, où le nom de l’environnement remplace environnement.

La partie du document entre le `\begin{ }` et `\end{ }` est dans l’environnement choisi.

## 6. Structure d’un fichier LaTeX

Tous les fichiers LATEX doivent commencer par la commande `\documentclass`. Cette commande indique à LATEX de quel type de document il s’agit, d’un livre, rapport, article, ...

Ensuite viennent les commandes qui vont changer le style du document ou charger d’éventuelles extensions avec la commande `\usepackage`. Cette partie est appelée le préambule.

Une fois tout celà fait, le corps du document peut enfin commencer. Le corps du document se trouve dans l’environnement document. Il commence donc par la commande `\begin{document}`.

Maintenant, vous pouvez saisir le texte de votre document et une fois celà terminé, il suffit d’utiliser la commande `\end{document}` qui signale la fin de l’environnement document et donc la fin du corps de votre document.

<div style="display: flex; align-items: center;"> <div style="writing-mode: vertical-rl; transform: rotate(180deg); font-weight: bold; margin-right: 5px;">Préambule</div> <div style="font-family: monospace;"> <code>\documentclass{report}</code>  <math>\left\{ \begin{array}{l} \dots \\ \dots \\ \dots \end{array} \right.</math>  <code>\begin{document}</code> </div> </div>		<div style="display: flex; align-items: center;"> <div style="writing-mode: vertical-rl; transform: rotate(180deg); font-weight: bold; margin-right: 5px;">Corps du document</div> <div style="font-family: monospace;"> <math>\left\{ \begin{array}{l} \dots \\ \dots \\ \dots \end{array} \right.</math>  <code>\end{document}</code> </div> </div>		<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">Exemple :</div> <div style="font-family: monospace;"> <code>\documentclass[12pt,a4paper]{report}</code>  <code>\usepackage[utf8]{inputenc}</code>  <code>\usepackage[francais]{babel}</code>  <code>\setlength{\parskip}{0.5em}</code>  <code>\renewcommand{\baselinestretch}{1.1}</code>  <code>\begin{document}</code>  Département de Mathématiques   Première Année Master Proba  <code>\end{document}</code> </div> </div>
--	--	---	--	--

Tout fichier TEX est bâti selon le même canevas constitué :

- d’une commande `\documentclass` : obligatoire et unique, placer à la première ligne. Son rôle est de spécifier la “classe” de document qu’on souhaite rédiger.
- d’un préambule : c’est l’ensemble (éventuellement vide) de ce qui figure entre les commandes `\documentclass` et `\begin{document}`.

- d’un environnement document : celui-ci est obligatoire.
- d’un corps de document : c’est tout ce qui se trouve à l’intérieur de l’environnement document, donc entre les commandes `\begin{document}` et `\end{document}`.

## 6.1 Espaces

Le changement de ligne simple équivaut à une espace. Par contre, si on laisse une ou plusieurs lignes vides, cela indique un changement de paragraphe. On peut laisser plusieurs lignes vides, cela équivaut à une seule ligne vide. De même, si l’on met plusieurs espace, c’est comme s’il n’y en avait qu’une seule.

### – Commandes pour les espaces horizontaux fixes :

`\,` espace fine.

`\:` espace moyenne.

`\;` espace large.

`\quad` espace plus large (cadratin).

`\qquad` espace encore plus large (double cadratin).

`\hspace*{2cm}` exemple d’espace horizontal de largeur donnée.

### – Commandes pour les espaces horizontaux ”élastiques”

`\hfill` remplit l’espace restant sur la ligne courante.

`\dotfill` remplit l’espace restant sur la ligne courante par des pointillés.

`\hrulefill` remplit l’espace restant sur la ligne courante par un filet horizontal.

### – Commandes pour les espaces verticaux fixes

`\smallskip` petit espace vertical.

`\medskip` espace vertical moyen.

`\bigskip` grand espace vertical.

`\vspace*{2cm}` exemple d’espace vertical de hauteur donnée.

### – Commandes pour les espaces verticaux ”élastiques”

`\vfill` remplit l’espace vertical restant sur la page courante.

### – Autres espacements

`\newline` impose un changement de ligne à l’endroit du document où la commande est utilisée.

`\newpage` impose un changement de page à l'endroit du document où la commande est utilisée.

## 6.2. Classe de document

Syntaxe : `\documentclass[option]{class}` où : option est un argument optionnel, et classe est un argument obligatoire.

Exemples de quelques classes :

- **article** pour des articles destinés à la publication et ne contenant que quelques pages.
- **report** pour des documents un peu plus longs contenant plusieurs chapitres, comme des mémoires de thèse.
- **book** pour de véritables livres, de plusieurs centaines de pages.
- **slides** pour faire des présentations.
- **lettre** pour faire des lettres au format français.
- **memoir** pour écrire des mémoires, par exemple de fin d'étude.

Le choix de la classe va déterminer un certain nombre de paramètres par défaut, comme par exemple les marges.

### 6.2.1. Classe book

La classe book est prévue pour faire un livre. Les spécificités du livre sont les suivantes :

- il dispose d'une page de titre séparée, suivie d'une page blanche.
- il peut se décomposer en parties, chapitres, sections, sous-sections, sous-sous-sections, paragraphes et sous-paragraphes.
- les parties et chapitres commencent sur une page impaire.
- les marges sont assez grandes pour permettre une lecture aisée (par rapport à la quantité du texte).

### 6.2.2. Classe report

La classe report est similaire à la classe book, mais les chapitres ne commencent pas nécessairement en page impaire, et l'on ne peut pas utiliser certaines fonctions comme `\frontmatter`, `\mainmatter` et `\backmatter`. Par contre, elle dispose d'un environnement `abstract` permettant la mise en forme automatique d'un résumé.

### 6.2.3. Classe article

Par rapport à un livre, un article :

- a son titre sur la même page que le début du texte.
- a des marges plus étroites (mise en page dans un journal).
- a moins de subdivisions de texte : il n’y a pas de chapitre.

### **6.3. Quelques valeurs de l’argument option :**

10pt, 11pt, 12pt	taille de police (valeur par défaut = 10pt ).
a4paper, a5paper ...	taille de page (valeur par défaut = letterpaper = 11in×8.5in).
onecolumn, twocolumn	multiples colonnes (valeur par défaut = onecolumn).
Landscape	mode paysage.
Fleqn	alignement à gauche des formules mathématiques.