

Introduction brouillonne et rapide à L^AT_EX

Où les auteurs tenteront désespérément de présenter LaTeX en 45 minutes

1. Qu'est-ce que L^AT_EX ?

L^AT_EX est un système de composition de documents. On écrit un fichier source en texte brut, généralement avec l'extension `.tex`, puis on le *compile* pour produire un document mis en page, souvent un PDF.

Contrairement à un traitement de texte classique, on ne positionne pas chaque élément à la souris : on décrit la **structure** (titre, section, formule, tableau, etc.) et L^AT_EX se charge de la présentation.

Points forts : formules mathématiques, numérotation automatique, références croisées, documents longs, mise en page cohérente. **Limite** : il faut apprendre quelques commandes et interpréter les messages de compilation.

2. Overleaf : utiliser LaTeX en ligne

Overleaf (<https://www.overleaf.com>) est un éditeur LaTeX accessible directement depuis un navigateur Internet, sans installation de logiciel.

Il permet d'écrire son code et de visualiser le PDF obtenu dans une même fenêtre. La compilation est effectuée sur les serveurs d'Overleaf.

Ses principaux avantages sont la sauvegarde en ligne, le partage de documents et la possibilité de travailler à plusieurs sur un même projet.

Une formule gratuite est disponible, avec certaines limitations.

3. Le fonctionnement en trois étapes

1. Créer ou modifier un fichier `.tex` dans un éditeur de texte adapté ou non.
2. Compiler avec un moteur comme LaTeX, pdfLaTeX, XeLaTeX ou LuaLaTeX.
3. Lire le PDF obtenu et, si nécessaire, corriger le code puis recompiler.

Exemple minimal, complet et compilable :

```
\documentclass[a4paper,12pt]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[french]{babel}

\begin{document}
Bonjour ! Voici mon premier document en LaTeX.
\end{document}
```

`\documentclass` choisit la classe du document. Les lignes `\usepackage` chargent des extensions. La partie avant `\begin{document}` est le **préambule**; le contenu visible est placé entre `\begin{document}` et `\end{document}`.

4. Les commandes et les environnements

Une commande commence par une barre oblique inverse : \. Ses arguments obligatoires sont généralement entre accolades {} ; les options facultatives, entre crochets [].

```
\textbf{Texte en gras}  
\textit{Texte en italique}  
\section{Titre de section}
```

Un **environnement** encadre une partie du document :

```
\begin{center}  
Ce texte est centré.  
\end{center}
```

Le symbole % introduit un commentaire, ignoré à la compilation. Les accolades permettent aussi de limiter la portée d'un changement de présentation.

5. Écrire et organiser du texte

Une ligne vide dans le code source commence un nouveau paragraphe. Un simple retour à la ligne dans le fichier ne crée pas forcément de saut de ligne dans le PDF.

```
\section{Introduction}  
Premier paragraphe avec \textbf{un mot important}.  
  
Deuxième paragraphe avec \textit{une idée à souligner}.  
\subsection{Un sous-titre}  
Le texte continue ici.
```

Les commandes \section, \subsection et \subsubsection créent une hiérarchie et une numérotation automatiques. On évite d'écrire soi-même « I. », « A. », etc.

Caractères particuliers : certains caractères ont un rôle technique : %, \$, &, #, _, {, }. Pour les afficher, on les précède généralement de \, par exemple \% pour %.

6. Monsieur, comment on saute des lignes et comment on fait des espaces ?

En voilà une bonne question ! Réponse rapide :

- \\ permet de faire un retour à la ligne sans commencer un nouveau paragraphe. On évite d'enchaîner les \\ pour sauter des lignes : nous ne sommes pas des brutes !
- Pour commencer un nouveau paragraphe, il suffit de laisser une ligne vide dans le code source. On peut aussi utiliser la commande \par.
- \vspace{3mm} permet d'ajouter un espace vertical de 3 mm entre deux éléments. Attention : cette commande ne provoque pas à elle seule un retour à la ligne.
- \smallskip, \medskip et \bigskip permettent d'ajouter respectivement un petit, un moyen et un grand espace vertical.

- `\hspace{0.5cm}` permet d'ajouter un espace horizontal de 0,5 cm.
- `\quad` et `\qquad` permettent d'ajouter des espaces horizontaux prédéfinis. Le second est deux fois plus grand que le premier.
- `\hfill` insère un espace horizontal extensible. Il permet notamment de placer un élément à gauche et un autre à droite sur une même ligne.
- `\newpage` permet de passer à la page suivante.

7. Les sciences : le grand atout de $\text{\LaTeX}$

Pour une formule dans une phrase, utiliser `$...$` ou `\(...\)` : par exemple $a^2 + b^2 = c^2$.

Pour une formule isolée et centrée, utiliser `$$ ... $$` ou `\[...]` :

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

Dans le texte : `$x^2 + 2x + 1$`.

`$$ \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} $$`

Quelques commandes fondamentales (il y en a des milliers) :

Code	Résultat	Usage
<code>x^2</code>	x^2	Exposant
<code>a_{n+1}</code>	a_{n+1}	Indice
<code>\frac{a}{b}</code>	$\frac{a}{b}$	Fraction
<code>\sqrt{x}</code>	$\sqrt{x}$	Racine carrée
<code>\alpha, \pi</code>	α, π	Lettres grecques
<code>\leq, \neq</code>	$\leq, \neq$	Relations
<code>\in, \mathbb{R}</code>	$\in, \mathbb{R}$	Ensembles (<code>amssymb</code>)
<code>\Delta T</code>	ΔT	Variation de température
<code>\vec{F}</code>	$\vec{F}$	Vecteur force
<code>\ce{CO2}</code>	CO_2	Chimie : dioxyde de carbone

Pour aligner plusieurs étapes : charger `amsmath`, puis utiliser `align*` (sans numéros) :

```
\begin{align*}
3(x+2)-4 &= 3x+6-4 \\
&= 3x+2.
\end{align*}
```

$$\begin{aligned} 3(x+2) - 4 &= 3x + 6 - 4 \\ &= 3x + 2. \end{aligned}$$

Le caractère `&` indique l'endroit de l'alignement et `\\` sépare les lignes.

Personnellement, j'utilise plutôt des tableaux, même lorsque j'écris des mathématiques. C'est l'objet de la section suivante.

8. Tableaux et listes

8.1 Tableaux

Un tableau simple utilise l'environnement `tabular`. Les lettres `l`, `c` et `r` alignent les colonnes à gauche, au centre ou à droite.

```
\begin{tabular}{lcr}
Nom & Note & Mention \\
Alice & 15 & Bien \\
Samir & 18 & Très bien
\end{tabular}
```

Nom	Note	Mention
Alice	15	Bien
Samir	18	Très bien

Le caractère & sépare les colonnes et \\ termine une ligne.

```
\begin{tabular}{|c||c|c|}
\hline
A & B & C \\ \hline
1 & 2 & 3 \\ \hline
\end{tabular}
```

A	B	C
1	2	3

Dans `{|c||c|c|}`, les symboles `|` et `||` indiquent respectivement une séparation verticale simple et double entre les colonnes. `\hline` trace une ligne horizontale.

Pour des tableaux complexes, on peut employer `tabularx`.

8.2 Listes

L'environnement `enumerate` permet de créer une liste numérotée.

```
\begin{enumerate}
\item Première étape
\item Deuxième étape
\item Troisième étape
\end{enumerate}
```

1. Première étape
2. Deuxième étape
3. Troisième étape

8.3 Listes à puces

L'environnement `itemize` permet de créer des listes à puces. On peut également choisir le symbole utilisé devant chaque élément, par exemple `\item[{$\circ$}]`, `\item[{$\star$}]` ou `\item[{$\bullet$}]`.

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

— Pomme
— Poire

```
\begin{itemize}
\item[{$\bullet$}] Pomme
\item[{$\bullet$}] Poire
\end{itemize}
```

• Pomme
• Poire

9. Insérer une image

Pour insérer une image dans un document LaTeX, il faut charger le package `graphicx` dans le préambule :

```
\usepackage{graphicx}
```

On peut ensuite insérer une image avec la commande `\includegraphics` :

```
\includegraphics[width=0.6\textwidth]{figure.png}
```

L'option `width=0.6\textwidth` signifie que l'image occupera 60 % de la largeur du texte.

On peut également préciser sa hauteur ou modifier son échelle :

```
\includegraphics[height=5cm]{figure.jpg}  
\includegraphics[scale=0.8]{figure.pdf}
```

Quels formats d'images peut-on utiliser ?

LaTeX accepte plusieurs formats, selon le moteur de compilation choisi :

- `.png` : images matricielles, notamment captures d'écran et illustrations ;
- `.jpg` : photographies ;
- `.pdf` : images vectorielles, particulièrement adaptées aux figures mathématiques ;
- `.eps` : images vectorielles au format Encapsulated PostScript, souvent utilisées pour les figures scientifiques et géométriques.

Attention au moteur de compilation !

- Avec `pdfLaTeX`, on utilise directement les fichiers `.png`, `.jpg` et `.pdf`.
- Avec la chaîne de compilation `LaTeX + dvips + ps2pdf`, on peut utiliser directement les fichiers `.eps`.
- Avec `pdfLaTeX`, un fichier `.eps` doit généralement être converti en `.pdf`. Cette conversion peut parfois être effectuée automatiquement si les outils nécessaires sont disponibles.

Exemple avec une image EPS

Si l'on utilise la chaîne de compilation `LaTeX + dvips + ps2pdf`, on peut écrire :

```
\usepackage[dvips]{graphicx}  
  
% Dans le document :  
\begin{center}  
\includegraphics[width=8cm]{figure.eps}  
\end{center}
```

Où placer les fichiers images ?

Le plus simple est de placer les images dans le même dossier que le fichier `.tex`.

On peut aussi les ranger dans un sous-dossier :

```
\includegraphics[width=7cm]{images/figure.eps}
```

On peut aussi créer directement des dessins vectoriels avec le package `TikZ`, mais nous n'en parlerons pas ici.

10. Un premier exercice en autonomie

Créez un fichier source contenant : un titre, deux sections, une liste numérotée, la formule $\Delta = b^2 - 4ac$, un tableau à deux colonnes et une figure simple. Compilez-le, puis modifiez une section et observez la renumérotation automatique.