

## TP : Introduction à l'édition de documents avec LaTeX

**Objectifs** : à l'issue de ce TP, vous devrez savoir

- les opérations de base de l'édition de document en LaTeX
- compiler un document avec ShareLaTeX et en local

**Liens** : il est fortement recommandé de se documenter tout au long du TP

- tutorial LaTeX de l'ENS: <http://www.tuteurs.ens.fr/logiciels/latex/>

**Préambule** : Les élèves qui n'ont pas terminé le précédent TP sur Git doivent le faire en début de séance. Il faut faire jusqu'à l'exercice 5.

### Exercice 1 : ShareLaTeX

- Question 1 – Connexion à ShareLaTeX
  - L'école dispose d'un serveur ShareLaTeX permettant d'éditer des documents en LaTeX via une application web collaborative : <https://latex.telecomnancy.univ-lorraine.fr>
  - Si ce n'est pas déjà fait, il faut préalablement définir son mot-de-passe ShareLaTeX à partir de cette page de l'intranet : [https://www-intranet.telecomnancy.univ-lorraine.fr/doku.php/faq/initialisation\\_du\\_compte\\_share\\_latex](https://www-intranet.telecomnancy.univ-lorraine.fr/doku.php/faq/initialisation_du_compte_share_latex)
  - Se connecter à la plateforme avec son adresse email @telecomnancy.eu et le mot-de-passe précédemment défini
  - Créer un nouveau projet vierge « tpLatex »
- Question 2 - Définir un modèle
  - Les modèles permettent de créer de nombreux types de documents au style pré-formaté (rapport, article scientifique, livre, CV, etc.). En Gestion de Projet, vous utiliserez un style très simple permettant d'écrire des comptes-rendus de réunion.
  - Importer le modèle disponible sur arche « meetingmins.cls » dans ShareLaTeX (bouton « Upload » dans la colonne de gauche), ainsi que le fichier d'exemple « minutes.tex ».
  - Compiler le fichier « minutes.tex » et observer le résultat.
    - la documentation du style est disponible dans le fichier « meetingmins.pdf » et illustre les commandes propres au style : `\setcommitee`, `\setmembers`, `\setdate`, etc.
    - Autres exemples de templates à utiliser durant votre scolarité :
      - Rapport de PIDR: [https://www-intranet.telecomnancy.univ-lorraine.fr/lib/exe/fetch.php/2a/pidr/templateieee\\_pidr.zip](https://www-intranet.telecomnancy.univ-lorraine.fr/lib/exe/fetch.php/2a/pidr/templateieee_pidr.zip)
      - Rapport de stage, de PI : <https://github.com/TELECOMNancy/tn-report>
- Question 3 - Structure d'un document LaTeX
  - Revenir sur le fichier initial « main.tex » pour la suite du TP
  - Observer le fichier « main.tex » :
    - `\documentclass` définit le style du document
    - `\usepackage[option]{nom_du_package}` permet d'utiliser des fonctionnalités particulières, les packages doivent être déclarés avant le corps du document
    - `\title`, `\author`, `\date`, définissent des méta-données du document. Elles sont affichées dans le corps du document par la commande `\maketitle`
    - `\begin{document}` et `\end{document}` encadrent le corps du document
    - le symbole % permet de commenter une ligne
    - le symbole `\` permet d'annuler l'interprétation d'un caractère spécial de Latex
      - afficher un symbole % s'écrit donc `\%`

## Exercice 2 : Commandes LaTeX de base

- Question 1 - Structurer un document
    - Créer une section et deux sous-sections :

```
\section{Introduction}
\subsection{Première sous-section}
\subsubsection{Première sous-sous-section}
```
    - Les questions suivantes de cet exercice feront l'objet d'une sous-section dédiée dans votre document Latex
    - Note :
      - selon le style utilisé, les sections peuvent être numérotées ou non
      - d'autres niveaux hiérarchiques existent dans d'autres styles :
        - (\chapter) ou \part > \section > \subsection > \subsubsection > \paragraph
        - ... et permettent de constituer automatiquement la table des matières
  - Question 2 - Ajouter une liste à puces
    - Pour créer une liste à puces, il faut une balise d'ouverture `\begin{itemize}`, une balise de fermeture `\end{itemize}`, ainsi qu'une balise `\item` par élément de la liste.
    - Créer une liste à deux niveaux :

```
\begin{itemize}
\item des poires ;
\begin{itemize}
\item des Williams
\item des Guyot
\end{itemize}
\item des carottes ;
\item des choux.
\end{itemize}
```
    - Note : une source d'erreur de compilation fréquente consiste à ouvrir une liste sans la fermer (ou inversement). Remplacer « itemize » par « enumerate » permet de créer des listes numérotées
  - Question 3 - Ajouter une image et la référencer
    - Nous allons ajouter une image grâce à `\includegraphics`, la centrer `\begin{center}` et lui donner un titre `\caption{titre}`, et un label `\label{...}` grâce à l'environnement figure `\begin{figure}` :
      - importer l'image du logo de l'école dans ShareLaTeX
      - déclarer l'utilisation du package suivant : `\usepackage{graphicx}` (*rappel : les packages se déclarent en préambule, c'est-à-dire, avant \begin{document}*)

```
\begin{figure}
\centering
\includegraphics{logo_TNCY.png}
\caption{Le logo de TELECOM Nancy}
\label{fig:logo_tncy}
\end{figure}
```
  - Note : les balises `caption` et `label` doivent être à l'intérieur des balises délimitant l'environnement `figure` pour s'y rattacher
  - La figure est trop grosse : réduire son échelle grâce à l'option `scale` de la commande `\includegraphics` : `\includegraphics[scale=0.1]{...}`
  - Ajouter un texte décrivant l'image et la référencer avec `\ref{fig:logo_tncy}` :
    - Exemple : « Le logo de l'école est illustré par la Figure `\ref{fig:logo_tncy}`. »
- Question 4 - Ajouter un tableau
  - L'environnement `tabular` `\begin{tabular}{motif_colonnes}` permet de créer des tableaux dont les colonnes suivent le motif défini, par exemple le motif « `||c|c|` » signifie que trois colonnes sont

définies, la première alignée à gauche et les deux autres centrées

- `\hline` permet de tracer les lignes horizontales du tableau, le symbole `&` de séparer les colonnes et le symbole `\\` de terminer la ligne
- Comme pour les images, les tableaux sont souvent insérés dans un environnement « table » afin de bénéficier d'une légende et d'une référence par label
- Entrer l'exemple de tableau suivant :

```
\begin{table}
  \begin{center}
    \begin{tabular}{|l|c|c|}
      \hline
        colonne 1 & colonne 2 & colonne 3 \\
      \hline
        1.1 & 1.2 & 1.3 \\
        2.1 & 2.2 & 2.3 \\
      \hline
    \end{tabular}
  \end{center}
  \caption{Exemple simple de tableau}
  \label{tab:exemple}
\end{table}
```

- Question 5 - Ajouter des formules mathématiques

- Les formules mathématiques peuvent être insérées directement dans du texte entre deux symboles `$`, par exemple :
  - Einstein a établi la célèbre formule  $E = mc^2$ .
- L'environnement équation `\begin{equation}` permet de numéroter les formules et de les référencer :

```
Einstein a établi la célèbre formule \ref{eq-Einstein}.
\begin{equation}
  \label{eq-Einstein}
  E = mc^2
\end{equation}
```

- Beaucoup de symboles scientifiques sont disponibles en environnement mathématiques : `\sum`, `\frac`, `\pi`, `\forall`, `\in`, etc.

- Question 6 - Ajouter des références bibliographiques

- Nous avons vu que Latex gère les références internes via les labels qui peuvent être associés aux sections, figures, équations, etc.
- La bibliographie est traitée à part : elle est définie dans un fichier spécifique `.bib`, à son propre style, est compilée séparément
- Créer un nouveau fichier « `biblio.bib` » dans ShareLateX et y ajouter la référence suivante :

```
@Book{GOT4,
  author = {George R.R. Martin},
  title = {{A Feast for Crows (A Song of Ice and Fire)}},
  year = {2005},
  isbn = {0-553-80150-3},
  pages = {784},
  publisher = {Bantam}
}
```

- De nombreux types de documents peuvent être cités : `book`, `article`, `proceedings`, `techreport`, `misc`, etc. , et permettent de formater l'affichage de la référence selon les champs requis
- Dans le document principal, ajouter la bibliographie en fin de document :

```
\bibliographystyle{plain}
\bibliography{biblio.bib}
```

- et citer la référence dans le texte avec `\cite{GOT4}` :

- Exemple : « G.R.R Martin a écrit, dans le livre `\cite{GOT4}`: "“I prefer my history dead. Dead history is writ in ink, the living sort in blood." »

### Exercice 3 : Du rapport à la présentation

- Question 1 – Structure d'une présentation LaTeX
  - LaTeX permet également de faire des présentations et ainsi recycler rapidement les contenus du rapport. Créer un fichier `presentation.tex` reprenant le contenu de `main.tex` mais avec un corps de document vide (entre `\begin{document}` et `\end{document}`)
  - Changer le `\documentclass` par **beamer**, qui est celui des présentations
  - Ajouter un style en préambule : `\usetheme{Berlin}` (de nombreux autres styles existent)
  - Créer la slide de titre avec `\maketitle`
  - Ajouter votre première slide de contenu : `\begin{frame}{Titre} ... \end{frame}`
  - Pour les présentations longues, les frames peuvent être organisées en sections, ss-sections...
  - Votre code doit ressembler à :

<pre>\documentclass{beamer} \usepackage[utf8]{inputenc} \usepackage{graphicx} \usetheme{Berlin}  \title{tpLatex} \author{Thibault Cholez} \date{October 2017}</pre>	<pre>\begin{document} \maketitle  \begin{frame}{Frame Title}  \end{frame}  \bibliographystyle{plain} \bibliography{biblio.bib}  \end{document}</pre>
---	--

- Question 2 – Ajouter un bloc d'information
  - L'environnement `block` est très utile pour structurer les slides (souvent combiné avec des `itemize`) :  

```
\begin{block}{Premier élément}
    blabla
\end{block}
```
- Question 3 – Intégrer les éléments du document à la présentation
  - Toutes les commandes vues précédemment sont valables en beamer : intégrer les éléments vus dans l'exercice 2 à la présentation, à raison d'une slide par question

### Exercice 4 : LaTeX en local et gestion de version

- Question 1 – Téléchargement des fichiers depuis ShareLatex
  - Revenir à la liste de vos projets ShareLatex
  - Sélectionner le projet de ce tp et le télécharger : tous les fichiers sont regroupés dans une archive
- Question 2 – Versionner les fichiers
  - Créer un nouveau répertoire « `tp_latex` » dans votre dépôt git du précédent TP et y extraire le contenu de l'archive
  - Vérifier que vos fichiers LaTeX compilent bien localement :
    - `pdflatex main.tex` → le fichier pdf correspondant doit être généré dans le même répertoire et correspondre à ce qu'affichait ShareLatex
    - Si la bibliographie n'est pas construite : `bibtex main.aux`, puis à nouveau `pdflatex`
  - Indexer ce répertoire : `git add tp_latex`, enregistrer et pousser cette modification sur le dépôt distant (sur le serveur gitlab de l'école). Grâce au `.gitignore` pour TeX positionné à la fin du TP git (Exercice 5, Question 1), seuls les fichiers sources (`.tex`, `.bib`) doivent être versionnés par git !
  - Vérifier que les fichiers sont bien présents sur le dépôts distant via l'interface web de gitlab. **La présence des fichiers sera vérifiée pour l'attribution des crédits du module.**
  - Note : Ces deux manières de pratiquer LaTeX à plusieurs (via ShareLaTeX, ou via un dépôt git) sont valides, utilisez celle que vous préférez pour la suite
- Question 3 – Pour aller plus loin
  - Renseignez-vous sur d'autres environnements LaTeX, comme l'environnement `listing` qui permet d'insérer du code source