

# Introduction à $\text{\LaTeX}$

## Mise en page et modifications manuelles

Pascal Bessonneau

05/2016

# Plan

Listes à puce, insérer du code, ...

Justification, taille et couleur du texte

Ajout d'espacement et autres...

Mise en page

Babel...

## Les listes à *puces*

Les commandes de mise en page utilisent la syntaxe par blocs expliquée précédemment.

Code

```
\begin{itemize}
  \item Numéro 1
  \begin{itemize}
    \item Numéro 1
    \item Numéro 2
  \end{itemize}
  \item Numéro 2
\end{itemize}
```

Ce qui donne...

# Les listes à *puces*

- Numéro 1
  - Numéro 1
  - Numéro 2
- Numéro 2

## Les listes à *puces*

La syntaxe qui permet de changer le type de puce est assez simple.

Code

```
\begin{itemize}
  \item[\CheckmarkBold] Numéro 1
  \begin{itemize}
    \item[\ding{51}] Numéro 1
    \item[\ding{56}] Numéro 2
  \end{itemize}
  \item[\XSolidBrush] Numéro 2
\end{itemize}
```

Ce qui donne...

## Les listes à *puces*

- ✓ Numéro 1
  - ✓ Numéro 1
  - ✗ Numéro 2
- ✗ Numéro 2

## Les listes à puces

Les quelques lignes précédentes pose le problème de trouver la liste des symboles de  $\text{\LaTeX}$ .

Comme évoqué précédemment, le lecteur trouvera son bonheur dans la liste des caractères spéciaux réunis dans le document *The Comprehensive  $\text{\LaTeX}$  Symbol List*.

L'ouvrage FramaBook  $\text{\LaTeX}$  de l'association *Framasoft* est également une source d'inspiration.

# Les listes descriptives

$\text{T}_{\text{E}}\text{X}$  Le langage original par Knuth

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  La version évoluée et plus simple.

$\text{ConT}_{\text{E}}\text{X}$  Une version très intéressante



# Les listes descriptives

Code

```
\begin{description}
  \item[\TeX] Le langage original par Knuth
  \item[\LaTeX] La version évoluée et plus simple.
  \item[Con\TeX] Une version très intéressante
\end{description}
```

# Les listes incrémentées

Code

```
\begin{enumerate}
  \item Numéro 1
  \begin{enumerate}
    \item Numéro 1
    \item Numéro 1
  \end{enumerate}
  \item Numéro 2
\end{enumerate}
```

Ce qui donne...

# Les listes incrémentées

1. Numéro 1
  1. Numéro 1
  2. Numéro 1
2. Numéro 2

## Les listes incrémentées

Il y a la possibilité de changer le style de numérotation :

**Brutale** en utilisant l'environnement

paquet *enumitem* Les commandes disponibles dans ce paquet sont notamment :

`\alph{number}` lettres en minuscules

`\Alph{number}` lettres en majuscules

`\arabic{number}` les nombres

`\roman{number}` des lettres « romaines » en minuscules et des numéros

`\Roman{number}` des lettres « romaines » en majuscules et des numéros

## Les listes incrémentées

Code

```
\begin{enumerate}[label=\arabic{*})]  
  \item Numéro 1  
  \begin{enumerate}[label=\alph{*})]  
    \item Numéro 1  
    \item Numéro 1  
  \end{enumerate}  
  \item Numéro 2  
\end{enumerate}
```

# Les listes incrémentées

paquet *enumerate* qui a une syntaxe plus naturelle mais qui présente moins de possibilités...

Code

```
\begin{enumerate}[a)]  
\item ...  
\end{enumerate}
```

# Les listes incrémentées

- 1) Numéro 1
  - a) Numéro 1
  - b) Numéro 1
- 2) Numéro 2

## Les listes incrémentées

Pour commencer (ou plutôt reprendre) une liste numérotée en cours de route, il faut jouer sur la valeur du compteur. Les compteurs décrits plus loin sont des variables incrémentées par  $\text{\LaTeX}$  et/ou par l'utilisateur.

Code

```
\begin{enumerate}[1)]  
  \setcounter{enumi}{5}  
  \item Numéro 1  
  \begin{enumerate}[a)]  
    \setcounter{enumii}{3}  
    \item Numéro 1  
    \item Numéro 1  
  \end{enumerate}  
  \item Numéro 2  
\end{enumerate}
```



## Les listes incrémentées

6) Numéro 1

d) Numéro 1

e) Numéro 1

7) Numéro 2

## Encore plus de listes...

Les paquets *inparaenum*, *enumitem*, *easylist*, ... permettent de nombreuses fonctionnalités autour des listes.

Par exemple, *paralist* permet d'ajouter une liste numérotée à l'intérieur d'un texte.

## Justifions du texte...

Toujours par blocs...

Code

```
\begin{flushright}
    Ce texte est à droite...\n
\end{flushright}
\begin{center}
    Ce texte est centré...\n
\end{center}
\begin{flushleft}
    Ce texte est à gauche...\n
\end{flushleft}
```

Ce qui donne...

## Justifions du texte...

Ce texte est à droite...

Ce texte est centré...

Ce texte est à gauche...

## Justification du texte

Vous pouvez souhaiter justifier du texte comme *Word* avec l'option "ligne pleine" : c'est-à-dire avec le moins d'espace possible et aucun mot ne dépassant et aucune césure. Dans ce cas, il faut intégrer le code ci-dessous dans le préambule où dans un bloc.

Code

```
\tolerance=1  
\emergencystretch=\maxdimen  
\hyphenpenalty=10000  
\hbadness=10000
```

## Mettons en évidence...

Commande	Déclarations	Output
<code>\textrm{...}</code>	<code>{\rmfamily ...}</code>	roman
<code>\textsf{...}</code>	<code>{\sffamily ...}</code>	sans sérif
<code>\texttt{...}</code>	<code>{\ttfamily ...}</code>	machine à écrire
<code>\textup{...}</code>	<code>{\upshape ...}</code>	droit
<code>\textit{...}</code>	<code>{\itshape ...}</code>	<i>italique</i>
<code>\textsl{...}</code>	<code>{\slshape ...}</code>	<i>penché</i>
<code>\textsc{...}</code>	<code>{\scshape ...}</code>	PETITES CAPITALES
<code>\textmd{...}</code>	<code>{\mdseries ...}</code>	medium
<code>\textbf{...}</code>	<code>{\bfseries ...}</code>	<b>gras</b>

TABLE: tiré du FramaBook sur L<sup>A</sup>T<sub>E</sub>X

## Mettons en évidence...

Cette fois c'est une commande avec arguments...

Code

```
Ce texte est en mode normal\\  
\emph{Ce texte est en italique}\\  
\texttt{Ce texte est en machine à écrire}\\  
\textit{Ce texte est en italique}\\  
\textsc{Ce texte est en petites capitales}\\  
\textbf{Ce texte est en gras}\\
```

ce qui donne...

## Mettons en évidence...

Ce texte est en mode normal

*Ce texte est mis en évidence*

Ce texte est en machine à écrire

*Ce texte est en italique*

CE TEXTE EST EN PETITES CAPITALES

**Ce texte est en gras**



## Mettons en évidence...

Code

```
{ \bfseries Cette syntaxe diminue notablement la  
lisibilité du code.  
}
```

**Donc à utiliser avec précaution.**

## Mettons en couleurs...

Dans ce cas, on appelle une macro dans le paquet *color*. Du coup, l'appel se fait par la commande suivie de plusieurs accolades, chaque couple d'accolade abritant un argument :

Code

```
\textcolor{red}{...texte...}
```

L'appel se fait avec la couleur en premier argument puis le texte à coloriser dans le second argument.

## Les tailles de polices...

Pour les tailles de police, c'est une commande sans argument qui va changer l'interprétation du texte qui la suit :

Code

```
\Huge Lorem ipsum \LARGE dolor sit amet, consectetur  
\normalsize adipiscing elit. \tiny Nunc in dui nec arcu  
\normalsize facilisis cursus id nec lorem.
```

## Les tailles de polices...

Ce qui donne...

**Lorem ipsum** dolor sit amet,  
**consectetur** adipiscing elit. Nunc in dui nec arcu facilisis cursus id nec  
lorem.

## Les tailles de polices...

**TABLE:** Changement de taille (Tiré du Framabook sur  $\text{\LaTeX}$ )

<code>\Huge</code>	<b>immense</b>	<code>\normalsize</code>	normal
<code>\huge</code>	<b>énorme</b>	<code>\small</code>	petit
<code>\LARGE</code>	<b>très très gros</b>	<code>\footnotesize</code>	plus petit
<code>\Large</code>	<b>très gros</b>	<code>\scriptsize</code>	encore plus petit
<code>\large</code>	<b>gros</b>	<code>\tiny</code>	minuscule

# Espacement vertical

Pour forcer  $\text{\LaTeX}$  à insérer un espace entre deux paragraphes, il faut utiliser la commande `\vspace`.

Code

```
Du texte...\  
\vspace{1cm}  
Le reste du texte...
```

Du texte...

Le reste du texte...

# Espacement horizontal

Pour forcer  $\text{\LaTeX}$  à insérer un espace entre deux mots, il faut utiliser la commande `\hspace`.

Code

```
Du texte...\hspace{1cm}Le reste du texte...
```

Du texte...      Le reste du texte...

## Insérer une ligne

Pour insérer une ligne horizontale, il suffit d'appeler la commande `\hrule`.

Code

```
Du texte...\n      \vspace{0.1cm}\n\hrule\n\vspace{0.1cm}\nLe reste du texte...
```

Du texte...

---

Le reste du texte...



# Insérer une ligne

Dans le même genre d'idées...

Code

```
Du texte \dotfill du texte \dotfill du texte\\
\vspace{0.1cm}
Du texte \hrulefill du texte \hrulefill du texte\\
```

Du texte .....du texte .....du texte

Du texte \_\_\_\_\_du texte \_\_\_\_\_du texte

## Insérer des sauts de page

Pour insérer un saut de page manuel, il suffit d'ajouter `\newpage`. La commande `\clearpage` est légèrement différentes : elle indique au compilateur qu'un saut de page est nécessaire mais elle impose aussi au compilateur de vider le tampon contenant les flottants. Par conséquent tous les tableaux et toutes les figures » en attente » sont placés dans le document au niveau de la commande. La commande `\clearpage` s'utilise typiquement quand on veut un saut de page entre le chapitre 2 et le chapitre 3. De plus, les figures du chapitre 2 ne doivent pas apparaître dans le chapitre 3.

## Redéfinir les marges et l'espace imprimable...

Cela peut être réalisé à l'aide de commandes  $\text{\LaTeX}$  directement :

Code

```
\setlength{\oddsidemargin}{0cm}  
\setlength{\textwidth}{16cm}  
\setlength{\textheight}{24cm}  
\setlength{\topmargin}{-1cm}  
\setlength{\marginparsep}{0.2cm}
```

Le lecteur exigeant sur les marges trouvera plus de renseignement en consultant la page du package *geometry* qui permet beaucoup de manipulations sur les différentes marges des documents  $\text{\LaTeX}$ .

## Redéfinir les marges et l'espace imprimable...

Au lieu de définir les valeurs exactes, on peut redéfinir par rapport à la valeur par défaut :

Code

```
\addtolength{grandeur}{ajout positif ou négatif}
```

# Redéfinir les marges et l'espace imprimable...

<code>\pdfpageheight</code>	hauteur du PDF
<code>\pdfpagewidth</code>	largeur du PDF
<code>\topmargin</code>	longueur de la marge en haut
<code>\evensidemargin</code>	marge gauche sur les pages impaires
<code>\oddsidemargin</code>	marge gauche sur les pages paires
<code>\headheight</code>	hauteur du pied de page
<code>\headsep</code>	distance entre le bas de l'en tête et le texte.
<code>\topskip</code>	distance entre le haut du texte entre le haut du texte et la première ligne de texte.
<code>\textheight</code>	hauteur du texte principal
<code>\textwidth</code>	hauteur du texte principal
<code>\footskip</code>	distance entre le bas du texte et le bas du pied de page
<code>\parskip</code>	distance entre deux paragraphes
<code>\parindent</code>	longueur de l'indentation des paragraphes.

**TABLE:** Options de mise en page

## Redéfinir les en-têtes et pieds de page

Pour redéfinir les en-têtes et pieds de page, le plus simple est d'utiliser le package *fancyhdr*.

Code

```
\usepackage{fancyhdr}  
\usepackage{lastpage}
```

Vous utiliserez souvent le paquet *lastpage* avec car il permet d'imprimer le nombre total de pages du document.

## Redéfinir les en-têtes et pieds de page

Pour le document texte de la formation, dans le préambule vous pourrez trouver ces commandes pour les en-têtes et pieds de page :

Code

```
\pagestyle{fancy}  
\fancyfoot[L]{\small Formation \LaTeX, DEPP}  
\fancyfoot[c]{}  
\fancyfoot[R]{\small \thepage/\pageref{LastPage}}  
\fancyhead[L]{}  
\fancyhead[c]{}  
\fancyhead[R]{}
```

## Redéfinir les en-têtes et pieds de page

Vous l'aurez compris le paquet offre de personnaliser la gauche, le centre et la droite des en-têtes et pieds de page avec les commandes L, C et R.

On appelle le nombre total de pages avec la commande `\pageref{LastPage}` et le numéro de la page courante par la commande `\thepage`.



## Redéfinir la taille d'un paragraphe

Parfois il est nécessaire de limiter la largeur d'un paragraphe ou de forcer  $\text{\LaTeX}$  à considérer un paragraphe comme un bloc. La commande `\parbox` est là pour ça.

Lorem ipsum dolor sit amet,  
consectetur adipiscing elit. In-  
teger sapien mi, pretium ut  
consectetur a, congue eu odio.  
Vestibulum et odio vel sem pul-  
vinar ultrices ultrices eget nulla.  
Nam in ipsum nibh. Integer  
euismod turpis et diam molestie  
accumsan. Suspendisse potenti.  
Curabitur fringilla, urna conval-  
lis dapibus placerat, elit nibh vi-  
verra leo, ut suscipit quam dui  
vitae felis. . .

## Redéfinir la taille d'un paragraphe

Code

```
\begin{center}  
  \parbox{5cm}{  
    Lorem ipsum dolor ...  
  }  
\end{center}
```

## Faire du multicolonne

Pour un article ou pour tout autre document, il peut être intéressant d'avoir plusieurs colonnes.

Tous les fonctionnalités sont bien décrites dans ce document sur le site Gutenberg.

Code

```
\begin{multicols}{3}  
Lorem ipsum dolor sit amet...  
\end{multicols}
```



## L'intérêt des *minipage*

Code

```
\begin{tabular}{cc}
  \begin{minipage}[l]{0.4\textwidth}
    Lorem ...
  \end{minipage}\vspace{0.3cm}
  &
  \begin{minipage}[l]{0.4\textwidth}
    Lorem ...
  \end{minipage}\vspace{0.3cm}\\
  \begin{minipage}[l]{0.4\textwidth}
    \includegraphics[width=\textwidth]
      {Graphiques/t_distribution}
  \end{minipage}
  &
  \begin{minipage}[l]{0.4\textwidth}
    Lorem ...
  \end{minipage}\\
\end{tabular}
```

## Un mot sur Babel

Pour écrire que M<sup>me</sup> s'est mariée pour la 1<sup>re</sup> fois, c'est possible grâce à l'extension *french* de Babel que vous trouverez dans le préambule des documents qui vous sont fournis.

1<sup>o</sup>, 2<sup>o</sup>, etc... et d'autres abbréviations françaises sont possibles. Pour une revue de détail, voir l'ouvrage *L<sup>A</sup>T<sub>E</sub>X par la pratique* de Christian ROLLAND qui donne aussi des clefs sur la typographie française qui sont loin d'être toutes respectées dans ce document.

### Code

```
Pour écrire que M\up{me} s'est mariée pour la 1\ere fois  
\primo, \secundo, ...  
Christian \bsc{Rolland}
```