

# Initiation à la rédaction de documents avec L<sup>A</sup>T<sub>E</sub>X

---

Nicolas Gutehrlé

# Sommaire

## 1. Présentation

- 1.1 Qu'est-ce que  $\text{\LaTeX}$
- 1.2 Compilation

## 2. Écrire un document

- 2.1 Préambule des documents
- 2.2 Packages
- 2.3 Structure
- 2.4 Police et caractères

## 3. Environnements

- 3.1 Listes
- 3.2 Tableaux
- 3.3 Images
- 3.4 Références

## 4. Bibliographie et index

- 4.1 Le fichier BibTeX
- 4.2 Compilateurs
- 4.3 BibLaTeX
- 4.4 Natbib
- 4.5 Index

## 5. Autre

- 5.1 Inclure des fichiers

## 6. Pour aller plus loin

# Présentation

## Qu'est-ce que $\text{\LaTeX}$

- $\text{\LaTeX}$ (se prononce "lataik") est un langage à balises compilé permettant la génération de texte. Il est utilisé par des chercheurs, éditeurs, professeurs, étudiants, ...
- Créé par Leslie Lamport en 1983,  $\text{\LaTeX}$  repose sur  $\text{\TeX}$ , qui a été créé par Donal Knuth en 1977
- Les fichiers  $\text{\LaTeX}$  se reconnaissent à leur extension : **.tex**

# Pourquoi utiliser $\text{\LaTeX}$

- Adapté à l'écriture scientifique : génération d'équations mathématiques, affichage de code, croisement des références, génération rapide de la bibliographie, ...
- Permet de générer de nombreux types de documents : articles, livres, posters, présentation, ... et dans différents formats (PDF, HTML, XML, ...)
- Sépare l'écriture de la mise en page du texte
- Libre, gratuit, stable et rapide

## Les inconvénients de $\text{\LaTeX}$

- Son apprentissage est plus complexe que pour un outil de traitement de texte plus classique
- $\text{\LaTeX}$  n'est pas **WYSIWYG (What You See Is What You Get)** : pour visualiser le document avec sa mise en page, il faut d'abord le compiler
- Ne s'adapte pas à tous les usages : pour rédiger un document court ou que l'on va partager avec des non-initiés à  $\text{\LaTeX}$ , il est préférable d'utiliser différents outils (LibreOffice, Word, Pages, ...)

# Avec quoi écrire $\text{\LaTeX}$

Pour rédiger des documents en  $\text{\LaTeX}$ , il vous faudra :

- un compilateur
- un éditeur de texte

# Compilation

Les fichiers `.tex` contiennent des commandes  $\text{\LaTeX}$  qui indiquent comment mettre en page des éléments textuels. Cependant, pour générer le document final à partir de ces commandes, il nous faut un compilateur. Il en existe plusieurs :

**Tex Live** : pour Linux et autres systèmes UNIX

**MiKTeX** : pour Windows

**MacTex** : distribution de Tex Live pour les systèmes macOS



# Compilation

## Remarques

- Le compilateur génère plusieurs fichiers de sorties : .aux, .log, .dvi, .ps, .bbl, .blg, .ind, .idx, .ilg, .toc, .pdf
- Si votre fichier `.tex` est mal écrit, la compilation ne pourra pas aboutir et le compilateur affichera un message d'erreur.

# Erreurs de compilation

- si une commande indiquée n'existe pas

```
1 TraditionalBuilder: Engine: pdflatex. Invoking latexmk... done.  
2 Errors:  
3 premier-doc.tex:8: Undefined control sequence. [\ssection]
```

- une accolade n'a pas été fermée

```
1 TraditionalBuilder: Engine: pdflatex. Invoking latexmk... done.  
2 Errors:  
3 TeX STOPPED: File ended while scanning use of \textit{en  
    italique et \textbf {en gras} ... Encore du texte. \par  
    \begin {c  
4 TeX reports the error was in file: premier-doc.tex
```

# Erreurs de compilation

- un environnement n'a pas été fermé

```
1 TraditionalBuilder: Engine: pdflatex. Invoking latexmk... done.  
2 Errors:  
3 premier-doc.tex:18: LaTeX Error: \begin{center} on input line  
    14 ended by \end{document}. [\end{document}]
```

- un caractère non-autorisé a été utilisé (ex : &, %)

```
1 TraditionalBuilder: Engine: pdflatex. Invoking latexmk... done.  
2 Errors:  
3 premier-doc.tex:11: Misplaced alignment tab character &.  
    [Encore&]
```

# Editeur $\text{\LaTeX}$

On peut écrire en  $\text{\LaTeX}$  avec n'importe quel éditeur de texte, mais il en existe des dédiés tels que :

- TexMaker
- TexStudio
- Overleaf (en ligne) : <https://www.overleaf.com/learn>

# Ecrire un document

# Votre premier document

```
1 \documentclass[a4paper, 12pt]{article}
2
3 \usepackage[utf8]{inputenc}
4
5 \title{Mon premier document}
6 \author{Nicolas Gutehrlé}
7 \date{Novembre 2021}
8
9 \begin{document}
10
11 \maketitle
12
13
14 \section{Introduction}
15
16 % Ceci est un commentaire. Ils commencent par un le symbole %
17 Ceci est un introduction à \LaTeX. Grâce aux commandes, vous pouvez
   modifier la mise en page du document, par exemple écrire en
   \textit{italique} ou en \textbf{gras}.
18
19 % Ci-dessous, l'environnement "itemize", qui permet d'écrire des listes
   d'éléments
20 \begin{itemize}
21   \item Premier élément de la liste
22   \item Second élément de la liste
23   \item ...
24 \end{itemize}
25
26 \end{document}
```

# Votre premier document

## Mon premier document

Nicolas Gutehrlé

Novembre 2021

### 1 Introduction

Ceci est une introduction à  $\text{\LaTeX}$ . Grâce aux commandes, vous pouvez modifier la mise en page du document, par exemple écrire en *italique* ou en **gras**.

- Premier élément de la liste
- Second élément de la liste
- ...

# Elements d'un document

Un document `.tex` est divisé en deux grandes parties, chacune contenant des éléments spécifiques :

## Le préambule :

- Classe du document
- Packages
- Commandes

## Le document :

- Commandes
- Environnements



## Préambule

Le préambule ne contient pas le document lui-même, mais les données relatives à ce document :

```
1 \documentclass[a4paper, 12pt]{article}
2
3 \usepackage[utf8]{inputenc}
4
5 \title{Mon premier document}
6 \author{Nicolas Gutehrlé}
7 \date{Novembre 2021}
```

Figure – Préambule du document

# Classes de documents

La première ligne du préambule (et par conséquent, du fichier **.tex**) indique la classe du document, c'est à dire son type :

```
\documentclass[options]{type de document}
```

**Types de document** : book, report, article, ...

**Options** : a4paper, a5paper, ... 10pt, 11pt, 12pt,  
twocolumn, landscape, oneside, twoside,  
...

```
\documentclass{book}
\author{...}
\title{...}

\begin{document}
\maketitle
\chapter{...}
\section{...}
...
\subsection{...}
\end{document}
```

A. Dracopis

**Rule:** here is some text without a meaning. The text should show how a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and your sentence like *the student follows*. Q&R - Now read a blind text like this you get information about the printed text, here the letters are written and the composition of the text. This text should contain all letters of the alphabet and it should be written like the original language. There is no need for a special sentence, but the length of each-sentence can

### 1.3 Heading on level 1 (section)

### 3.1.1 Heading on level 2 (information)

[illegible]

### 1.2.1 Example for list (strings)

- First item in a list
  - First item in a list
    - First item in a list
      - First item in a list
    - Second item in a list
  - Second item in a list

### 3.4.2. Examples for Not Commensurate

Example for list (XPNumericals)

1. First item in a list
  - (a) First item in a list
    - i. First item in a list
      - A. First item in a list
      - B. Second item in a list
    - ii. Second item in a list
  - (b) Second item in a list
- i. Second item in a list



IEEE	ACM	LLNCS
<code>\documentclass{IEEEtran}</code>	<code>\d..class{sig-alternate}</code>	<code>\documentclass{llncs}</code>

20/96

# Packages

Les **packages** ajoutent de nouvelles fonctions à votre document **.tex**. Pour les intégrer, il faut ajouter dans le préambule la commande `\usepackage{}`, en ajoutant le nom du package entre les accolades. Parmi les packages important que l'on peut citer :

**babel** : gère plusieurs encodage, et permet d'écrire dans différentes langues (français, arabe, turque, japonais, ...)

**graphicx** : permet d'insérer et de gérer des images

**ansmath** : permet de produire des équations mathématiques

# Packages

## Code $\text{\LaTeX}$

```
\documentclass{article}  
\usepackage{tabularx}  
\usepackage{graphicx}  
\usepackage{amsmath}  
\usepackage{url}  
\usepackage{verbatim}
```

## Commandes spécifiques au préambule

Certaines commandes ne peuvent apparaître que dans le préambule. On y indique généralement les métadonnées du documents telles que :

**titre** : avec la commande `\title{}`

**nom de l'auteur** : avec la commande `\author{}`

**date** : avec la commande `\date{}`



# Document

```
9 ▾ \begin{document}
10
11 \maketitle
12
13
14 ▾ \section{Introduction}
15
16 % Ceci est un commentaire. Ils commencent par un le symbole %
17 Ceci est un introduction à \LaTeX. Grâce aux commandes, vous pouvez
    modifier la mise en page du document, par exemple écrire en
    \textit{italique} ou en \textbf{gras}.
18
19 % Ci-dessous, l'environnement "itemize", qui permet d'écrire des listes
    d'éléments
20 ▾ \begin{itemize}
21     \item Premier élément de la liste
22     \item Second élément de la liste
23     \item ...
24 \end{itemize}
25
26 \end{document}
```

# Structure du document

Commande	Niveau
<code>\section{}</code>	1
<code>\subsection{}</code>	2
<code>\subsubsection{}</code>	3
<code>\paragraph{}</code>	4
<code>\subparagraph{}</code>	5
<code>\part{}</code> **	-1
<code>\chapter{}</code> **	0

\*\* : valable uniquement pour les documents de classe book et report

# Exemple

```
\documentclass[a4paper,10pt]{book}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[french]{babel}

\begin{document}
\part{Le début}

Ceci est le texte de la partie.

\chapter{Titre ici}

Texte

\section[Un titre court]{Un titre très long qui prendrait
plusieurs lignes dans la table des matières}

Texte

\section*{Une section sans numéro}

Texte

\end{document}
```

## Exemple

# Chapitre 1

## Titre ici

Texte

### 1.1 Un titre très long qui prendrait plusieurs lignes dans la table des matières

Texte

### Une section sans numéro

Texte

## Remarques

- Les styles des titres peuvent être modifiés avec le package `titlesec`
- On peut donner un titre plus court entre crochets, qui servira pour la table des matières, les entêtes et pieds de page, etc.
- Le package `babel` permet d'écrire des textes en différentes langues, dont le français.

## Afficher le titre

Pour afficher le titres et autres informations indiquées dans le préambule, il faut employer la commande :

```
\maketitle
```

juste après `\begin{document}`

# Table des matières

Pour générer un sommaire, il faut employer la commande suivante :

Code  $\text{\LaTeX}$

```
\tableofcontents
```

# Table des matières

## Sortie $\text{\LaTeX}$

1. Présentation
  - 1.1 Qu'est-ce que  $\text{\LaTeX}$
  - 1.2 Compilation
2. Ecrire un document
  - 2.1 Préambule des documents
  - 2.2 Packages
  - 2.3 Structure
  - 2.4 Police et caractères
3. Environnements
  - 3.1 Listes
  - 3.2 Tableaux
  - 3.3 Images
  - 3.4 Références
4. Bibliographie et index
  - 4.1 Le fichier BibTex



# Table des matières

## Remarques

- Les styles des titres dans la table des matières peuvent être personnalisés avec le package `tocloft`.
- Si on a besoin de mini-tables par chapitre, voir le package `minitoc`

## E spacements

Tous les espaces (simples ou multiples), tabulations, simples retours à la ligne sont convertis en simples espaces dans le document mis en page. Par conséquent, les deux textes suivants :

### Exemple

La pluie fait des claquettes sur le trottoir à minuit.

La pluie fait des claquettes  
sur le trottoir à minuit

# E spacements

Sont identiques :

## Exemple

La pluie fait des claquettes sur le trottoir à minuit.

La pluie fait des claquettes sur le trottoir à minuit

## E spacements

Pour créer un nouveau paragraphe, il faut donc soit :

- insérer au moins un double retour à la ligne
- utiliser la commande `\paragraph{}`

### Exemple

La pluie fait des claquettes

sur le trottoir à minuit

La pluie fait des claquettes  
sur le trottoir à minuit

# Espacements

## Remarques

- On peut ajouter des commentaires, des lignes vides, etc. autant que nécessaire pour la lisibilité du fichier .tex. Les lignes vides n'apparaissent pas dans le document final.
- Pour insérer des sauts de lignes :
  - `\newline` ou `\\` : saut de ligne
  - `\linebreak` : saut de ligne et texte justifié
  - `\pagebreak` : saut de page
- Le symbole `~` produit un espace insécable

## Taille des polices

Texte

\Huge

Texte

\huge

Texte

\LARGE

Texte

\Large

Texte

\large

Texte

\normalsize

Texte

\small

Texte

\footnotesize

Texte

\scriptsize

Texte

\tiny

## Taille des polices

### Exemple

\Huge La pluie \normalsize fait des \Large claquettes

Sur \small le trottoir à \tiny minuit

\footnotesize Parfois, je m'y arrête

**La pluie** fait des claquettes

Sur le trottoir à minuit

Parfois, je m'y arrête

## Effets de caractères

**Texte en gras**

`\textbf{}`

*Texte en italique*

`\textit{}`

Texte souligné

`\underline{}`



# Effets de caractères

## Exemple

La pluie fait des `\textbf{claquettes}`

`\underline{Sur le trottoir}` à minuit

Parfois, `\textit{je m'y arrête}`

La pluie fait des **claquettes**

Sur le trottoir à minuit

Parfois, *je m'y arrête*

## Autres polices

### Remarques

Un très grand nombre de police d'écriture est disponible. Vous pouvez consulter le lien suivant pour en avoir un aperçu :

<https://tug.org/FontCatalogue/>

## Caractères spéciaux

### Important

Plusieurs caractères tel que la barre oblique servent d'opérateurs en  $\text{\LaTeX}$ , et ne peuvent donc pas être utilisés tel quel. Pour cela, il faut les **échapper** avec le caractère `\`

## Caractères spéciaux

Caractère	Expression Latex
<code>\</code>	<code>\textbackslash</code>
<code>{</code>	<code>\{</code>
<code>}</code>	<code>\}</code>
<code>%</code>	<code>\%</code>
<code>~</code>	<code>\~</code>
<code>^</code>	<code>\^</code>
<code>\$</code>	<code>\\$</code>
<code>_</code>	<code>\_</code>
<code>&amp;</code>	<code>\&amp;</code>
<code>#</code>	<code>\#</code>

# Polices

Par défaut, la police utilisée dans un fichier  $\text{\LaTeX}$  est de la famille Roman, avec empattements. Les commandes ci-dessous permettent de modifier la police utilisée :

- `\textXX{}` Change la police du texte contenu dans les accolades
- `\XXfamily` Change la police du texte jusqu'à la fin du document, ou qu'une autre police est spécifiée

# Polices

Dans les commandes présentées ci-dessus, il faut remplacer XX par l'une des valeurs suivantes :

**rm** : police à empattements (style "roman")

**sf** : police sans empattements (style "sans serif")

**tt** : police à chasse fixe (style "teletype")

# Polices

## Code $\text{\LaTeX}$

En `\texttt{typographie}`, les `empattements` sont les petites extensions qui terminent les extrémités des caractères dans certaines polices d'écriture, dites `\rmfamily` avec empattement (au singulier~; serif en anglais)}, que l'on oppose aux polices `\sffamily` sans empattement (sans serif)}.

# Polices

## Sortie

En typographie, les empattements sont les petites extensions qui terminent les extrémités des caractères dans certaines polices d'écriture, dites avec empattement (au singulier ; serif en anglais), que l'on oppose aux polices sans empattement (sans serif).



# Environnements

# Environnements

Les environnements permettent de formater un ou plusieurs blocs de texte. Ils commencent par la commande `\begin{}` et se terminent par la commande `\end{}`. Ces environnements peuvent posséder leurs propres éléments de syntaxe.

Les plus notables sont :

- document** : formate le document selon la classe choisie dans le préambule

- table** : formate le texte sous forme de tableau

- figure** : formate une image

- itemize, enumerate, description** : formate le texte sous forme de liste

# Listes

LaTeX propose plusieurs environnements, chacun produisant un type différent de liste :

**itemize** : liste à points

**enumerate** : liste numérotée

**description** : liste indexée

# itemize

itemize crée une liste à points :

Code  $\text{\LaTeX}$

```
\begin{itemize}
  \item Jaune: bonheur
  \item Bleu: calme
  \item Orange: créativité
\end{itemize}
```

- Jaune : bonheur
- Bleu : calme
- Orange : créativité

## itemize

On peut indiquer à l'aide crochets le symbole qui servira de puce

Code  $\text{\LaTeX}$

```
\begin{itemize}
  \item[--] Jaune: bonheur
  \item[1] Bleu: calme
  \item[a] Orange: créativité
\end{itemize}
```

- Jaune : bonheur
- 1 Bleu : calme
- a Orange : créativité

## enumerate

`enumerate` crée une liste numérotée :

Code  $\text{\LaTeX}$

```
\begin{enumerate}
  \item Jaune: bonheur
  \item Bleu: calme
  \item Orange: créativité
\end{enumerate}
```

1. Jaune : bonheur
2. Bleu : calme
3. Orange : créativité

## description

`description` crée une liste indexée. Pour cela, il faut indiquer les noms entre crochets :

Code  $\text{\LaTeX}$

```
\begin{description}
  \item[Jaune]: bonheur
  \item[Bleu]: calme
  \item[Orange]: créativité
\end{description}
```

**Jaune** : bonheur

**Bleu** : calme

**Orange** : créativité

# Tableaux

Pour générer un tableau, il faut employer deux environnements : **table** et **tabular**.

**tabular** contient le tableau lui-même, avec la gestion des colonnes et du contenu. Il est contenu dans l'environnement **table** si celui-ci est utilisé.

**table** est nécessaire pour gérer la mise en page du tableau.



# Tableaux

## Code $\text{\LaTeX}$

```
\begin{table}[h]
  \centering
  \begin{tabular}{|c|c|c|}
    \hline
      Colonne 1 & Colonne 2 & Colonne 3 \\
    \hline
      cell1 & cell2 & cell3 \\
      cell4 & cell5 & cell6 \\
    \hline
  \end{tabular}
  \caption{Simple tableau}
  \label{tab:tableau}
\end{table}
```

# Tableaux

Sortie  $\text{\LaTeX}$

Colonne 1	Colonne 2	Colonne 3
cell1	cell2	cell3
cell4	cell5	cell6

Table – Simple tableau

# Tableaux

- `\centering` : permet de forcer le positionnement du tableau dans la page. Ici, le tableau doit être centré
- `{|c|c|c|}` : chaque lettre représente une colonne. Le symbole `|` permet de séparer chaque colonne par une barre verticale. Les trois principales lettres utilisées sont :
  - `l` : aligner le texte sur la gauche
  - `c` : centrer le texte
  - `r` : aligner le texte sur la droite
- `\hline` : permet d'insérer une barre horizontale

## Tableaux

- `cell1 & cell2 & cell3` : Le contenu de chaque colonne, séparé par le caractère `&`
- `\\` : permet d'insérer un saut de ligne
- `[h]` : permet de forcer le positionnement du tableau dans le document. Ici, le tableau doit être au même endroit que le code  $\LaTeX$

# Tableaux

## Remarques

Les éléments `\caption{}` et `\label{}` sont communs à plusieurs environnements, notamment `table` et `figure` :

- `\caption{}` : contient le titre du tableau, de l'image, ...
- `\label{}` : est un identifiant associé au tableau, à l'image, ... Il n'apparaîtra pas dans le PDF, mais peut être utilisé pour référencer l'élément.

# Tableaux

## Remarques

- Il s'agit de la syntaxe de base pour créer un simple tableau, mais il est possible d'en créer des bien plus complexes :

<https://www.overleaf.com/learn/latex/Tables>

- La syntaxe des tableaux peut très rapidement devenir illisible et prône à l'erreur. Il est donc recommandé d'utiliser un générateur de tableau, comme celui-ci :

[https://www.tablesgenerator.com/latex\\_tables](https://www.tablesgenerator.com/latex_tables)

## Tableau avec plusieurs lignes dans les cellules

Ceci nécessite le package tabularx

Code  $\text{\LaTeX}$

```
\begin{tabularx}{\textwidth}{|l|c|X|}  
\hline  
début & milieu & texte très long ici \\ \hline  
1 & 2 & Voici un exemple d'une ligne longue qui devrait  
dépasser la largeur du tableau. \\  
4 & 5 & 6 \\ \hline  
\textit{Total} & \textit{7} & \textit{9} \\ \hline  
\end{tabularx}
```

## Tableau avec plusieurs lignes dans les cellules

Sortie  $\text{\LaTeX}$

début	milieu	texte très long ici
1	2	Voici un exemple d'une ligne longue qui devrait dépasser la largeur du tableau.
4	5	
<i>Total</i>	7	



## Cellule sur plusieurs colonnes

Ceci nécessite le package `multirow`

Code  $\text{\LaTeX}$

```
\begin{tabular}{|l|c|r|}  
\hline  
début & milieu & fin \\ \hline  
1 & \multicolumn{2}{c|}{$2+3$} \\ \hline  
\multirow{2}{*}{4} & 5 & 6 \\  
& 7 & 8 \\ \hline  
\textit{Total} & \textit{7} & \textit{9} \\ \hline  
\end{tabular}
```

## Cellule sur plusieurs colonnes

Sortie  $\text{\LaTeX}$

début	milieu	fin
1	$2 + 3$	
$2 * 4$	5	6
	7	8
<i>Total</i>	7	9

# Images

Pour insérer des images, il faut utiliser trois outils :

- le package `graphicx`
- l'environnement `figure`
- la commande `\includegraphics`

# Images

Dans le préambule :

```
\usepackage{graphicx}
```

Où l'on insère l'image :

```
\begin{figure}  
  \centering  
  \includegraphics[width=.5\textwidth]{img/fricative.jpg}  
  \caption{Image}  
  \label{fig:fricatives}  
\end{figure}
```

## Images



Figure – Image

# Images

- les éléments `\centering`, `\caption{}` et `\label{}` sont identiques à ceux utilisés pour les tableaux
- l'élément `\includegraphics[ ]{}` demande deux arguments :
  - `[ ]` est optionnel. On y indique la taille de l'image
  - `{}` est obligatoire. On y indique le nom du fichier image à ouvrir.

## Images avec différentes échelles et angles

Code  $\text{\LaTeX}$

```
\includegraphics[scale=0.5]{img/fricative.jpg}
```



## Images avec différentes échelles et angles

Code  $\text{\LaTeX}$

```
\includegraphics[width=5cm, height=4cm]{img/fricative.jpg}
```





## Images avec différentes échelles et angles

Code  $\text{\LaTeX}$

```
\includegraphics[angle=180]{img/fricative.jpg}
```



## Références

Tout élément ayant un `\label{}` peut être référencé à l'aide de la commande `\ref{}`. On peut également indiquer le numéro de page correspondant à l'aide de la commande `\pageref{}` :

### Code $\text{\LaTeX}$

Dans ce cours, le tableau `\ref{tab:tableau}`  
(page `\pageref{tab:tableau}`)  
et l'image `\ref{fig:fricatives}`  
(page `\pageref{fig:fricatives}`)  
sont des exemples d'environnements.

# Références

## Sortie $\text{\LaTeX}$

Dans ce cours, le tableau 1 (page 58) et l'image 3 (page 69) sont des exemples d'environnements.

# Bibliographie et index

# Bibliographie

LaTeX dispose de deux outils principaux pour générer une bibliographie et intégrer des références bibliographiques au texte :

- un fichier au format `.bib` contenant les ressources bibliographiques
- un package

# Bibliographie

Le fichier `.bib` contient les entrées bibliographiques au format BibTex. Celles-ci ont la structure suivante :

- le type de document, indiqué par le signe `@`
- une accolade ouvrante
- un identifiant, venant juste après l'accolade
- différents champs, certains optionnels (`OPT`) ou alternatifs (`ALT`), séparés par des virgules
- une accolade fermante

# Bibliographie

```
@book{
  historical-macroscope,
  title={Exploring Big Historical Data: The Historian's
  Macroscope},
  author={Graham, S. and Milligan, I. and Weingart, S.},
  isbn={9781783266111},
  year={2016},
  publisher={World Scientific Publishing Company}
}
```

Figure – Exemple d'une entrée au format BibTex

# Bibliographie

- En général, les sites bibliographiques (Google Scholar, HAL, Zenodo, ACL...) permettent d'exporter une référence au format BibTex
- De même, les outils de gestion bibliographique tels que Zotero permettent cet export
- Si besoin, il existe des générateurs de références BibTex en ligne, ex : <https://www.citethisforme.com/citation-generator/bibtex>
- De même, les éditeurs  $\text{\LaTeX}$  tels que TexMaker permettent de générer ces références



# Compilateurs

Il existe trois principaux packages bibliographiques :

- BibLaTeX (successeur de Bibtex)
- Natbib
- BibTex

Ces packages diffèrent notamment dans les styles bibliographiques et de références disponibles

# BibLaTeX

Les quatre commandes principales de BibLaTeX sont :

Dans le préambule :

```
\usepackage[OPTIONS]{biblatex}
```

```
\addbibresource{FICHER.bib}
```

Dans le document :

```
\cite{IDENTIFIANT}
```

```
\printbibliography
```

## BibLaTeX

Les trois options suivantes sont à indiquer lors de l'import de BibLaTeX :

`backend=biber`

le compilateur à utiliser. `biber` est la valeur par défaut et est recommandé

`style=alphabetic`

changer le style bibliographique

`sorting=ynt`

changer la manière d'ordonner les éléments de la référence

# BibLaTeX

- Styles bibliographiques : [https://www.overleaf.com/learn/latex/Biblatex\\_bibliography\\_styles](https://www.overleaf.com/learn/latex/Biblatex_bibliography_styles)
- Ordonnancement : [https://www.overleaf.com/learn/latex/Bibliography\\_management\\_with\\_biblatex#Reference\\_guide](https://www.overleaf.com/learn/latex/Bibliography_management_with_biblatex#Reference_guide)

# Natbib

Les quatre commandes principales de Natbib sont :

Dans le préambule :

```
\usepackage[OPTIONS]{natbib}
```

Dans le document :

```
\cite{IDENTIFIANT}
```

```
\bibliographystyle{STYLE}
```

```
\bibliography{FICHER SANS L'EXTENSION .BIB}
```

# Natbib

- Styles bibliographiques : [https://www.overleaf.com/learn/latex/Natbib\\_bibliography\\_styles](https://www.overleaf.com/learn/latex/Natbib_bibliography_styles)
- Styles de citation : [https://www.overleaf.com/learn/latex/Natbib\\_citation\\_styles](https://www.overleaf.com/learn/latex/Natbib_citation_styles)

# Index

TeX permet de générer des index à l'aide du package `imakeidx`.

L'import du package doit être suivi de la commande `\makeindex{}`

Code TeX

```
\usepackage{imakeidx}  
\makeindex
```

# Index

Pour ajouter un élément à l'index, on emploie la commande `\index{}`. Pour afficher l'index, on emploie la commande `\printindex`

Code  $\text{\LaTeX}$

```
Ceci est un test de la commande \index{Index}  
\printindex
```

Ceci est un test de la commande Index



## Inclure des fichiers

- La commande `\include{CHEMIN VERS LE FICHIER}` permet d'inclure des fichiers `.tex` les uns dans les autres
- **Attention**, les fichiers `.tex` qui sont intégrés ne doivent pas contenir de préambule, ni l'environnement `\begin{document}`

## Inclure des fichiers

Exemple d'un fichier  $\text{\LaTeX}$  qui sera intégré

```
\section{Titre de section}
```

Ce fichier .tex ne contient pas de préambule, et n'utilise pas l'environnement `\begin{document}`

Exemple du fichier  $\text{\LaTeX}$  principal

```
...  
\begin{document}  
    \include{autrefichier.tex}  
\end{document}
```

## Inclure des fichiers

- De même, la commande `\includepdf[OPTIONS]{CHEMIN VERS FICHIER PDF}` permet d'intégrer des fichiers PDF à votre document. Pour l'utiliser, il faut importer le package `pdfpages`
- l'option `pages=` indique quelles pages du pdf doivent être intégrées. Pour intégrer toutes les pages, on utilise le symbole -

Pour aller plus loin

## Pour aller plus loin

- Marges, en-têtes et notes de bas de page
- Inclure des fichiers .tex et .pdf
- Présentations avec **beamer**
- Utiliser une feuille de style existante (template)
- Texte sur plusieurs colonnes

## Pour aller plus loin

- Ecrire des formules mathématiques, théorèmes, ...
- Algorithmes
- Couleurs
- Créer ses propres commandes et environnements
- Créer des schémas (avec pstricks)
- Typographie avancée (lettrines, ligatures, ...)
- Révisions (avec latexdiff)
- ...

# Ressources

**Overleaf** (en anglais) : <https://www.overleaf.com/learn>

**Wikibooks** : <https://fr.wikibooks.org/wiki/LaTeX>

**CTAN** : <https://ctan.org/>

**xm1math** : <https://www.xm1math.net/doculatem/index.html>

...

Merci pour votre attention !



# Contact

Nicolas Gutehrle

[nicolas.gutehrle@univ-fcomte.fr](mailto:nicolas.gutehrle@univ-fcomte.fr)

[nicolas.gutehrle@gmail.com](mailto:nicolas.gutehrle@gmail.com)