

## Documentation de la classe `letgut`

Association GUTenberg

Version 0.9.6 en date du 14 janvier 2023  
<https://framagit.org/gutenberg/letgut>

|  |    |
|--|----|
| Introduction .....   | 2  |
| Usage de la classe <code>letgut</code> .....                 | 2  |
| Convocation AG ordinaire .....                               | 23 |
| Aspects de la 1 <sup>re</sup> page .....                     | 26 |
| Les dinosaures, leur écosystème et <code>letgut</code> ..... | 26 |
| Packages chargés par <code>letgut</code> .....               | 27 |
| Liste des acronymes prédéfinis par <code>letgut</code> ..... | 31 |
| Références .....   | 32 |
| Table des avertissements .....                               | 33 |
| Index .....  | 34 |
| Adhésion à l'association .....                               | 37 |

**Avez-vous pensé à régler votre cotisation ?**

Si vous avez oublié, ce n'est ni trop tard ni difficile :  
<https://www.gutenberg-asso.fr/?Adherer-en-ligne>

## INTRODUCTION

L'association GUTenberg publie la *Lettre* GUTenberg, son bulletin irrégulomestriel, depuis février 1993 [1].

Pour ce faire, une classe (L)A<sub>T</sub>E<sub>X</sub> dédiée, maison, a peu à peu vu le jour<sup>1</sup> mais, au gré des nouveaux besoins et des personnes qui ont assuré la publication de la *Lettre*, son développement a été quelque peu erratique; il n'aurait notamment pas été possible de publier son code en l'état. En outre, sa documentation était inexistante.

Le CA de l'association élu en novembre 2020 a souhaité fournir une classe mieux structurée, davantage pérenne et documentée, à même d'être publiée sur le CTAN. C'est désormais chose faite avec la présente classe `letgut`<sup>2</sup>.

## USAGE DE LA CLASSE LETGUT

### Compilation

#### Avertissement 1 – Lua<sub>L</sub>A<sub>T</sub>E<sub>X</sub> (récent) et UTF-8 nécessaires

Les documents recourant à la classe `letgut` doivent :

- être compilés avec le moteur Lua<sub>L</sub>A<sub>T</sub>E<sub>X</sub><sup>a</sup>;
- avoir (de ce fait) comme codage d'entrée l'UTF-8<sup>b</sup>.

<sup>a</sup>. Dans une version au minimum 1.13.2, le format `luaLatex` devant être dans une version au minimum 2021.6.6.

<sup>b</sup>. Y compris les fichiers sources auxiliaires tels que les `.bib`.

La section « *Les dinosaures, leur écosystème et `letgut`* », page 26, détaille un moyen de disposer d'une telle version de Lua<sub>L</sub>A<sub>T</sub>E<sub>X</sub> sans risque de perturber une installation de (L)A<sub>T</sub>E<sub>X</sub> déjà existante.

### Options

#### Options de la classe `article`

La classe `letgut` est basée sur la classe `article`, chargée avec les options `twoside` et `11pt` qui ne peuvent donc être surchargées. Les autres options peuvent être employées mais sont déconseillées par souci d'homogénéité de la mise en page des numéros successifs de la *Lettre* GUTenberg.

#### Options de la classe `letgut`

##### Spécification

Les options de la classe `letgut` peuvent être spécifiées de deux façons :

1. en argument de la commande dédiée `\letgutsetup`<sup>P.3</sup>;
2. à la compilation, au moyen de la commande (à lancer dans un terminal) :

```
1 $ lualatex '\PassOptionsToClass{<options>}{letgut} \input{<fichier>}'
```

1. Notamment grâce au concours de Jacques André, Daniel Flipo et Maxime Chupin.  
2. À cette occasion, la classe a été légèrement renommée de `let-gut` en `letgut`.

### Avertissement 2 – Options de `letgut` : pas en argument optionnel de `\documentclass`

On évitera de passer les options de la classe `letgut` en argument optionnel de la commande `\documentclass` et ce, de sorte à éviter les conflits d'options avec les différents packages chargés.

```
\letgutsetup{<options>}
```

Cette commande permet de spécifier les `<options>` de la classe `letgut`.

### Liste des options

|                          |  |
|--------------------------|--|
| <code>for-readers</code> | (init. <code>true</code> , par déf. <code>true</code> )  |
| <code>for-authors</code> | (init. <code>false</code> , par déf. <code>true</code> ) |

Ces clés booléennes, contraires l'une de l'autre, activent les versions respectivement « pour lecteurs » (par défaut) et « pour auteurs » de la *Lettre*. Celles-ci mettent en page la *Lettre* respectivement :

- telle que les lecteurs finaux la liront ;
- telle que les auteurs d'articles de la *Lettre* peuvent le souhaiter au cours de leur rédaction, notamment sans :
  - titre ;
  - (r)appel à cotisation (cf. clé `membership-reminder`<sup>→p.4</sup>) ;
  - éditorial<sup>3</sup> (cf. clé `editorial`<sup>→p.4</sup>) ;
  - informations sur GUTenberg<sup>4</sup> (cf. clé `informations`<sup>→p.4</sup>) ;
  - fichiers attachés au PDF produit (cf. clé `reverse-files-attachement`<sup>→p.5</sup>)

|                    |  |
|--------------------|--|
| <code>draft</code> | (init. <code>true</code> , par déf. <code>true</code> )  |
| <code>final</code> | (init. <code>false</code> , par déf. <code>true</code> ) |

Ces clés booléennes, contraires l'une de l'autre, activent les versions respectivement « brouillon » (par défaut) et « final » de la *Lettre*.

En version « brouillon », et seulement dans cette version :

1. le package `lua-typo` est chargé<sup>5</sup>. Celui-ci met en lumière, par un changement de couleur, les lignes typographiquement imparfaites d'un fichier PDF produit par Lua<sup>®</sup>TeX ;
2. le mot « Brouillon » figure en filigrane sur chaque page ;
3. les boîtes trop pleines (*Overfull* `\hbox`) sont mises en évidence comme avec les classes standards.

|                     |  |
|---------------------|--|
| <code>screen</code> | (init. <code>true</code> , par déf. <code>true</code> )  |
| <code>paper</code>  | (init. <code>false</code> , par déf. <code>true</code> ) |

Ces clés booléennes, contraires l'une de l'autre, activent les formats de sortie respectivement « écran » (par défaut) et « papier » de la *Lettre*.

En version « écran », et seulement dans cette version, la couleur de fond du papier est par défaut non pas le blanc mais celle spécifiée (et modifiable) par `pagecolor`<sup>→p.4</sup>.

3. Ou avertissement s'il n'est pas trouvé.

4. Ou avertissement si elles ne sont pas trouvées.

5. En fait, cela est momentanément désactivé car le package `luacolor`, chargé en sous-main par `lua-typo`, ne fonctionne actuellement pas bien avec une fonctionnalité du noyau <sup>®</sup>TeX utilisée par la classe (plus de détails [ici](#).)

`number=<numéro>` (init. *vide*, par déf. *aucune*)

Cette clé permet de spécifier le `<numéro>` de la *Lettre*.

`date=<année>-<mois> ou <texte>` (init. *année et mois en cours*, par déf. *aucune*)

Cette clé permet de spécifier la date de la *Lettre*. Celle-ci est affichée sous la forme :

- « `<Mois> <année>` » dans les cas où l'option :
  - n'est pas employée (les mois et année en cours sont alors utilisés) ;
  - est employée sous la forme `date=<année>-<mois>` où `<année>` et `<mois>` doivent être des nombres entiers positifs. Les garde-fous suivants sont mis en place :
    - si `<année>` n'est pas celle en cours ou la suivante, elle est remplacée par l'année en cours ;
    - si `<mois>` n'est pas entre 1 et 12, il est remplacé par le mois en cours ;
- « `<texte>` » si l'option est employée sous la forme `date=<texte>`.

`pagecolor={<couleur>}` (init. `letgut_pagecolor`, par déf. *aucune*)

Cette clé permet, si (et seulement) l'option `paper`<sup>→ p.3</sup> n'est pas employée, de spécifier (selon le modèle « rgb ») une `<couleur>` de fond du papier autre que celle appliquée par défaut<sup>6</sup>.

`allcolorlinks=<couleur>` (init. `letgut_allcolors_links`, par déf. *aucune*)

Cette clé permet de spécifier (selon le modèle « named ») une `<couleur>` pour (tous) les liens hypertextes autre que celle `letgut_allcolors_links` par défaut<sup>7</sup>.

`membership-reminder` (init. *true*, par déf. *true*)

Cette clé booléenne affiche automatiquement un (r)appel à cotisation en bas de 1<sup>re</sup> page de la *Lettre*.

`editorial` (init. *true*, par déf. *true*)

Cette clé booléenne importe automatiquement en tout début de la *Lettre* (néanmoins après le titre et le sommaire) le fichier `editorial.tex` contenant l'éditorial. Si aucun fichier `editorial.tex` n'est trouvé, un avertissement est émis lors de la compilation et une boîte d'alerte est affichée en 1<sup>re</sup> page.

`informations` (init. *true*, par déf. *true*)

Cette clé booléenne importe automatiquement en dernière page de la *Lettre* le fichier `informations-gut.tex` contenant toutes les informations sur GUTenberg. Si aucun fichier `informations-gut.tex` n'est trouvé, un avertissement est émis lors de la compilation et une boîte d'alerte est affichée en dernière page.

`detailedtoc` (init. *false*, par déf. *true*)

Par défaut, la table des matières automatiquement insérée en début de document a pour niveau de profondeur celui des titres (saisis via la `\title`<sup>→ p.6</sup>), et seulement les titres des articles. La clé booléenne `detailedtoc` a pour effet que cette table des matières est détaillée au maximum (son niveau de profondeur allant jusqu'aux sous-paragraphes), ce qui peut être utile lors de la conception d'un numéro de la *Lettre*.

6. C'est-à-dire celle-ci .

7. C'est-à-dire celle-ci.

**Remarque 1 – Tables des matières locales**

Chaque article peut contenir une table des matières locale, affichée au moyen de la commande `\localtableofcontents` (fournie par le package `etoc` chargé en sous-main par la classe `letgut`). Le niveau de profondeur est par défaut celui des sections mais cela peut être modifié en la faisant précéder de la commande `\etocsetnexttocdepth` (dont l'argument est par exemple `subsection`).

**`reverse-files-attachement`** (init. *pas de valeur*, par déf. *aucune*)

Si, et seulement si, la *Lettre* est à la fois en version pour les lecteurs (cf. `for-readers`<sup>→P.3</sup>) et en sortie écran (cf. `screen`<sup>→P.3</sup>), chacun des fichiers nécessaires (et suffisants) à la compilation d'un de ses articles est :

- attaché au PDF produit ;
- accessible en cliquant sur l'hyperlien correspondant en forme de trombone :  ;

L'option `reverse-files-attachement` inverse ce comportement par défaut.

**Options autres**

D'autres options peuvent être passées à la classe `letgut`. Il est ainsi possible de faire usage de langues du package `babel`, autres que le français et l'anglais déjà chargées par `letgut`, en les stipulant en option de `\documentclass` et en les employant selon la syntaxe de ce package.

**Titre et titres courants**

Si la commande `\title{<titre>}` est

**non utilisée** : le titre du document est construit à partir du `<numéro>` et de la `<date>` spécifiés (cf. clés `number`<sup>→P.4</sup> et `date`<sup>→P.4</sup>). Il figure alors automatiquement en 1<sup>re</sup> page sous la forme « Numéro `<numéro>` – `<date>` ». Le titre courant est alors « La *Lettre* GUTenberg, `<date>` » ;

**utilisée**<sup>8</sup> : et ce, *avant* `\begin{document}`, le `<titre>` du document figure alors automatiquement en 1<sup>re</sup> page sous sa forme habituelle et est suivie d'un changement de page. Le titre courant est alors « `<titre>`, `<date>` ».

**Avertissement 3 – `\title` et `\author` ≠ avant et après `\begin{document}`**

Les commandes `\title` et `\author` ne se comportent pas de la même façon avant et après `\begin{document}` (cf. sections *Structuration* page suivante et *Personnes et auteurs* page 7).

**Avertissement 4 – Commande `\maketitle` à ne pas employer**

La commande `\maketitle` est à ne pas employer car elle l'est en sous-main par la classe.

8. Ainsi que les habituelles commandes `\author` et `\date`.

## Importation d'articles

### Avertissement 5 – Importation de fichiers d'articles

Si le contenu d'un article est stocké dans un `<fichier enfant>.tex`, on l'importera dans un fichier parent recourant à la classe `letgut` :

- *non pas* au moyen de la commande ordinaire `\input` ;
- *mais* au moyen de la commande `\inputarticle`<sup>→ p.6</sup>.

```
\inputarticle{<fichier enfant>}
\inputarticle*{<fichier enfant>}
```

Cette commande permet d'importer le contenu d'un article stocké dans un `<fichier enfant>.tex`.

En version étoilée, les fichiers nécessaires (et suffisants) à la compilation de l'article ne sont pas attachés au PDF produit (cf. clé `reverse-files-attachement`<sup>→ p.5</sup>)


En plus de l'importation proprement dite, cette commande procède à un certain nombre de réinitialisations.

```
\inputarticle{<fichier enfant>}
```

## Structuration

```
\title[<intitulé alternatif>]{<intitulé>}
\subtitle[<intitulé alternatif>]{<intitulé>}
\section[<intitulé alternatif>]{<intitulé>}
\subsection[<intitulé alternatif>]{<intitulé>}
\subsubsection[<intitulé alternatif>]{<intitulé>}
\paragraph[<intitulé alternatif>]{<intitulé>}
\subparagraph[<intitulé alternatif>]{<intitulé>}
```

Ces commandes permettent de structurer le contenu de la *Lettre* :

- `\title` est celle de plus haut niveau, introduisant l'`<intitulé>` de chaque article (automatiquement composé en grandes capitales et précédé de l'ornement );
- `\subtitle`, de niveau suivant et facultative, introduisant un `<intitulé>` d'éventuel sous-titre d'article (automatiquement composé en grandes capitales). Ceci peut être utile par exemple pour distinguer des parties indépendantes d'un même article ;
- celles de niveaux suivants sont les habituelles commandes de structuration fournies par la classe `article`.

### Remarque 2 – Structure non numérotée

Les titres, sous-titres, sections, sous-sections, etc. de la *Lettre* ne sont pas numérotés. Aussi pourra-t-on, pour faire référence à l'une de ces rubriques, recourir aux commandes :

- `\nameref` pour en citer l'`<intitulé>` ;
- `\vpageref` pour en citer la page ;
- `\enquote` pour, le cas échéant, faire figurer l'`<intitulé>` entre guillemets ; ces trois commandes étant directement utilisables puisque fournies par les

packages respectivement `hyperref`, `varioref` et `csquotes`, chargés en sous-main par la classe `letgut`.

#### Exemple 1 : références croisées aux rubriques

```
1 On lira avec intérêt la section \enquote{\nameref{sec:acronymes}}
2 \vpageref{sec:acronymes}.
```

On lira avec intérêt la section « Acronymes » page 20.

## Noms de personnes et d'auteurs d'articles

```
\person{<données>}
\person{<données1> and <données2>[ and ...]}
\person*{<données1> and <données2>[ and ...]}
\author{<données>}
\author{<données1> and <données2>[ and ...]}
\author*{<données1> and <données2>[ and ...]}
```

Ces commandes affichent<sup>9</sup> les *<données>* (noms et éventuels prénoms et titres) d'une ou plusieurs personnes ou d'un ou plusieurs auteurs d'articles, ces *<données>* étant spécifiées :

**pour un individu unique** selon l'un des formats suivants :

- *<nom>*
- *<nom>*, *<prénom>*
- *<nom>*, *<prénom>*, *<titre>*

**pour des individus multiples :**

- selon le même schéma que pour un individu unique ;
- les *<données>* de chacun des individus étant séparées par le mot clé `and`.

Les version étoilées de ces commandes trient alphabétiquement les listes de personnes ou d'auteurs.

Indépendamment de la casse utilisée en entrée, pour chaque *<nom>* et *<prénom>*, chacune des initiales et des premières lettres après un espace ou un tiret est affiché en grande capitale.

#### Exemple 2 : personnes

```
1 On peut dire merci à
2 \person{Knuth, Donald E., dieu and Lamport, Leslie} !
```

On peut dire merci à Donald E. Knuth (dieu) & Leslie Lamport !

9. Au fer à droite pour `\author`.

Exemple 3 : auteur

```

1 \begin{displayquote} % Fourni par `csquotes' chargé par `letgut'
2   Wait, wait, I never said that.
3   \author{knuth, DONALD e.}
4 \end{displayquote}

```

Wait, wait, I never said that.

Donald E. Knuth

## Aide à la saisie et homogénéisation de la mise en forme

Les articles de la *Lettre* sont émaillés de concepts (packages ou classes  $\LaTeX$ , logiciels, etc.) et de termes et expressions (« GUTenberg », « *Lettre* », etc.) employés de façon récurrente. Aussi des commandes spécifiques sont-elles prévues de façon à en faciliter la saisie et à en homogénéiser la mise en forme.

### Packages et classes, logiciels, fichiers, locutions étrangères, points de code Unicode

```

\package[⟨URL⟩]{⟨nom⟩}
\package*[⟨URL⟩]{⟨nom⟩}[⟨préfixe⟩]
\class[⟨URL⟩]{⟨nom⟩}
\class*[⟨URL⟩]{⟨nom⟩}[⟨préfixe⟩]

```

Ces commandes affichent le  $\langle nom \rangle$  d'un package ou d'une classe  $\LaTeX$ . Le  $\langle nom \rangle$  affiché est un lien hypertexte si et seulement si l'argument optionnel est :

**non employé** la cible étant alors `https://ctan.org/pkg/⟨nom⟩`;

**employé** mais non vide, la cible étant alors  $\langle URL \rangle$ .

Pour que le  $\langle nom \rangle$  affiché ne soit pas un lien hypertexte, il suffit d'employer un argument optionnel vide.

Les versions étoilées font précéder le  $\langle nom \rangle$  d'un  $\langle préfixe \rangle$  qui, par défaut, est respectivement « package » et « classe ».

Exemple 4 : packages et classes

```

1 La \class*{letgut} s'appuie entre autres sur le \package*{etoc}
2 (qui permet de personnaliser les tables des matières).
3
4 Une des classes s'appuyant sur le \package*[]{etoc} est
5 \class[https://framagit.org/gutenberg/letgut/]{letgut}.

```

La classe `letgut` s'appuie entre autres sur le package `etoc` (qui permet de personnaliser les tables des matières).

Une des classes s'appuyant sur le package `etoc` est `letgut`.

```

\software[⟨URL⟩]{⟨nom⟩}
\software*[⟨URL⟩]{⟨nom⟩}[⟨préfixe⟩]

```



Ces commandes affichent le  $\langle nom \rangle$  d'un logiciel qui est optionnellement un lien hypertexte vers  $\langle URL \rangle$ . La version étoilée fait précéder le  $\langle nom \rangle$  d'un  $\langle préfixe \rangle$  qui, par défaut, est « logiciel ».

```
\file{ $\langle nom \rangle$ }
\file*{ $\langle nom \rangle$ }[ $\langle préfixe \rangle$ ]
```

Ces commandes affichent le  $\langle nom \rangle$  d'un fichier. La version étoilée fait précéder le  $\langle nom \rangle$  d'un  $\langle préfixe \rangle$  qui, par défaut, est « fichier ».

#### Exemple 5 : logiciels et fichiers

```
1 Le \file*{test.tex} a été ouvert dans le
2 \software*[https://www.gnu.org/software/emacs/]{Emacs}, plus
3 précisément dans \software*{Emacs}[l'éditeur de texte].
```

Le fichier `test.tex` a été ouvert dans le logiciel `Emacs`, plus précisément dans l'éditeur de texte `Emacs`.

#### Avertissement 6 – Commandes pas toutes bienvenues en `\title` et `\subtitle`

Lorsqu'elles sont employées en argument des commandes `\title`<sup>p.6</sup> et `\subtitle`<sup>p.6</sup>, les versions étoilées de ces commandes ont des effets indésirables (préfixes pas en grandes capitales et signets non conformes).

```
\foreignloc{ $\langle locution \rangle$ }
```

Cette commande est conçue pour afficher une  $\langle locution \rangle$  étrangère.

```
\latinloc{ $\langle locution \rangle$ }
```

Cette commande est conçue pour afficher une  $\langle locution \rangle$  latine.

#### Exemple 6 : locutions étrangères

```
1 Attention aux \foreignloc{load-time options} !
2 Mais... \latinloc{errare humanum est}.
```

Attention aux *load-time options*! Mais... *errare humanum est*.

```
\Ucode[ $\langle nom \rangle$ ]{ $\langle point de code \rangle$ }
```

Cette commande est conçue pour afficher le  $\langle point de code \rangle$  et éventuellement le  $\langle nom \rangle$  d'un caractère Unicode sous la forme « U+ $\langle point de code \rangle$  ( $\langle nom \rangle$ ) ».

#### Exemple 7 : Point de code d'un caractère Unicode

```
1 Unicode a prévu le caractère \Ucode[symbole numéro]{2116}.
```

résultat (suite)

Unicode a prévu le caractère U+2116 (SYMBOLE NUMÉRO).

## Termes et expressions

```

\gutenberg
\gut
\assogut
\Assogut
\lettres
\lettresgut
\cahier
\cahiers
\cahiergut
\cahiersgut
\letgut
\letgutcls
\knuth
\lamport
\tl
\tugboat
\linux
\macos
\windows

```

Ce que ces commandes affichent est répertorié dans le tableau 1.

TABLEAU 1 – Effet des commandes de raccourcis

|                          |                            |
|--------------------------|----------------------------|
| <code>\gutenberg</code>  | GUTenberg                  |
| <code>\gut</code>        | GUTenberg                  |
| <code>\assogut</code>    | association GUTenberg      |
| <code>\Assogut</code>    | Association GUTenberg      |
| <code>\lettres</code>    | <i>Lettres</i>             |
| <code>\lettresgut</code> | <i>Lettres</i> GUTenberg   |
| <code>\cahier</code>     | <i>Cahier</i>              |
| <code>\cahiergut</code>  | <i>Cahier</i> GUTenberg    |
| <code>\cahiers</code>    | <i>Cahiers</i>             |
| <code>\cahiersgut</code> | <i>Cahiers</i> GUTenberg   |
| <code>\letgut</code>     | <code>letgut</code>        |
| <code>\letgutcls</code>  | classe <code>letgut</code> |
| <code>\knuth</code>      | Donald E. Knuth            |
| <code>\lamport</code>    | Leslie Lamport             |
| <code>\tl</code>         | TeX Live                   |
| <code>\tugboat</code>    | <i>TUGboat</i>             |
| <code>\linux</code>      | GNU/Linux                  |
| <code>\macos</code>      | macOS                      |
| <code>\windows</code>    | Windows                    |

Par homogénéité avec les commandes `\class→P.8` et `\class*→P.8`, on aurait pu souhaiter que les terme et expression « `letgut` » et « classe `letgut` » soient produits par `\letgut` et `\letgut*`. Mais cette dernière commande, étoilée, a dû être remplacée par

une commande non étoilée (`\letgutcls`), sans quoi un problème technique aurait empêché l'utilisation de `\letgut` en argument de `\section`<sup>10</sup>.

#### Avertissement 7 – Commande `\letgutcls` pas bienvenue en `\title` et `\subtitle`

Lorsqu'elle est employée en argument des commandes `\title`<sup>P.6</sup> et `\subtitle`<sup>P.6</sup>, la commande `\letgutcls` a un effet indésirable (préfixe pas en grandes capitales).

Nouv. : 2022-10-03

`\lettrenumber` [*entier relatif signé*]

Nouv. : 2022-10-03

`\lettrenumber*` [*entier relatif signé*]

Cette commande affiche le numéro de la *Lettre* :

**en cours** si l'argument optionnel n'est pas employé ;

**décalé de celui en cours** de l'*entier relatif signé*<sup>11</sup> spécifié sinon.

En version étoilée, la chaîne « n° » précède ce numéro.

#### Exemple 8 : emploi de la commande `\lettrenumber`

```

1 Si le numéro de la \lettre{} en cours est 46,
2 celui de la \lettre :
3 \begin{enumerate}
4 \item en cours est \lettrenumber ;
5 \item en cours est le \lettrenumber* ;
6 \item suivante est \lettrenumber[+1] ;
7 \item précédente est le \lettrenumber*[-1].
8 \end{enumerate}
```

code

Si le numéro de la *Lettre* en cours est 46, celui de la *Lettre* :

1. en cours est 46 ;
2. en cours est le n° 46 ;
3. suivante est 47 ;
4. précédente est le n° 45.

résultat

`\lettre`

Nouv. : 2022-10-03

`\lettre` [*argument optionnel*]

Nouv. : 2022-10-03

`\lettre*` [*argument optionnel*]

`\lettregut`

Nouv. : 2022-10-03

`\lettregut` [*argument optionnel*]

Nouv. : 2022-10-03

`\lettregut*` [*argument optionnel*]

Ces commandes affichent les chaînes de caractères respectivement « *Lettre* » et « *Lettre* GUTenberg », le cas échéant suivies du numéro de la *Lettre* :

**en cours** si l'*argument optionnel* est un point (.) ;

**décalé de celui en cours** de ce qui est spécifié si l'*argument optionnel* est un entier relatif signé<sup>11</sup> ;

10. Plus de détails [ici](#).

11. C.-à-d. un « plus » ou un « moins » (+ ou -) suivi d'un nombre entier.

**spécifié** si l'*(argument optionnel)* est autre.

En version étoilée, la chaîne « n° » précède le numéro (seulement si l'*(argument optionnel)* est employé).

#### Exemple 9 : emplois des commandes `\lettre`<sup>p.11</sup> et `\lettregut`<sup>p.11</sup>

```

1 Si le numéro de la \lettre{ } en cours est 46, on a :
2 \begin{enumerate}
3 \item \lettre
4 \item \lettre[.]
5 \item \lettre[+10]
6 \item \lettre[-10]
7 \item \lettre[43]
8 \item \lettre[coucou]
9 \item \lettre*[.]
10 \item \lettre*[+10]
11 \item \lettre*[-10]
12 \item \lettre*[43]
13 \item \lettre*
14 \end{enumerate}
15 On fait usage de ces commandes dans la \lettregut*[.].

```

code

Si le numéro de la *Lettre* en cours est 46, on a :

1. *Lettre*
2. *Lettre* 46
3. *Lettre* 56
4. *Lettre* 36
5. *Lettre* 43
6. *Lettre* coucou
7. *Lettre* n° 46
8. *Lettre* n° 56
9. *Lettre* n° 36
10. *Lettre* n° 43
11. *Lettre*

On fait usage de ces commandes dans la *Lettre* GUTenberg n° 46.

résultat

## Touches de clavier

Afin de disposer d'un moyen simple, riche et élégant pour composer des touches de clavier, la classe `letgut` s'appuie sur le package `biolinum` et notamment sa commande `\LKey`. Cette dernière a été légèrement étendue de façon à faciliter la saisie pour toutes les touches des diacritiques utilisés en français.

**Exemple 10 : touches de clavier**

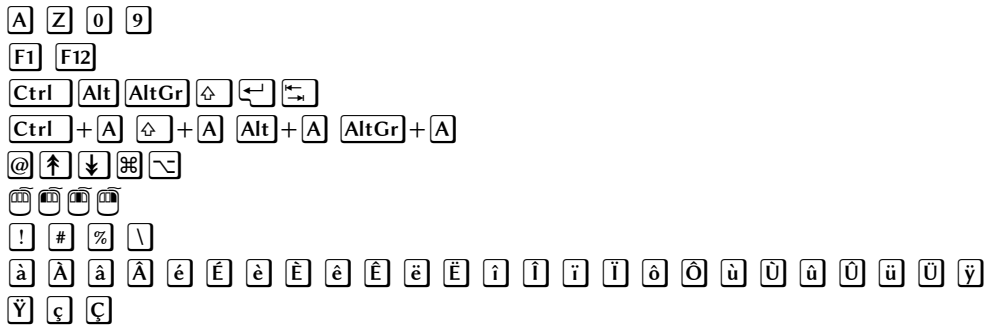
code

```

1 % De base (échantillon) :
2 \LKey{A} \LKey{Z} \LKey{0} \LKey{9}
3
4 \LKeyF{1} \LKeyF{12}
5
6 \LKeyCtrl \LKeyAlt \LKeyAltGr \LKeyShift \LKeyEnter \LKeyTab
7
8 \LKeyCtrlX{A} \LKeyShiftX{A} \LKeyAltX{A} \LKeyAltGrX{A}
9
10 \LKeyAt \LKeyScreenUp \LKeyScreenDown \LKeyCommand \LKeyOptionKey
11
12 \LMouseN \LMouseL \LMouseM \LMouseR
13
14 \LKey{exclam} \LKey{numbersign} \LKey{percent} \LKey{backslash}
15
16 % Ajoutés par la classe `letgut`
17 \LKey{à} \LKey{À} \LKey{â} \LKey{Â} \LKey{é} \LKey{É}
18 \LKey{è} \LKey{È} \LKey{ê} \LKey{Ê} \LKey{ë} \LKey{Ë}
19 \LKey{î} \LKey{Î} \LKey{ï} \LKey{Ï}
20 \LKey{ô} \LKey{Ô}
21 \LKey{ù} \LKey{Ù} \LKey{û} \LKey{Û} \LKey{ü} \LKey{Ü}
22 \LKey{ÿ} \LKey{Ÿ} \LKey{ç} \LKey{Ç}

```

résultat



**Avertissement 8 – Touche de clavier du symbole € manquant**

La touche de clavier du symbole € n'est pas fournie par le package `biolinum`.

## Codes informatiques

Cette section est consacrée aux outils spécifiques à la classe `letgut` permettant de faire figurer du code informatique dans la *Lettre*.

### Codes $\LaTeX$

#### Exemples de codes $\LaTeX$ , possiblement avec résultats

Afin de présenter aisément et de façon homogène les exemples de codes  $\LaTeX$ , possiblement avec leurs résultats, la classe `letgut` fournit les environnements « verbatim »

suivants.

```
\begin{ltx-code}[<options>]
  <code>
\end{ltx-code}
\begin{ltx-code-result}[<options>]
  <code>
\end{ltx-code-result}
\begin{ltx-code-external-result}[<options>]{<fichier>}
  <code>
\end{ltx-code-external-result}
```

Ces environnements affichent le `<code>` (L<sup>A</sup>T<sub>E</sub>X qui y est inséré et pour :

`ltx-code` seulement ce `<code>`;

`ltx-code-result` également le résultat, compilé en même temps que la *Lettre*;

`ltx-code-external-result` également le résultat, compilé indépendamment de la *Lettre* et dont le `<fichier>` image est spécifié.

### Remarque 3 – Mise en page des exemples de codes

1. Les exemples de codes (avec ou sans résultats) sont par défaut automatiquement coupés en frontière de page.
2. Les exemples de codes avec résultats (environnements `ltx-code-result` et `ltx-code-external-result`), présentent ces codes et résultats :
  - l'un sous l'autre par défaut;
  - l'un à gauche de l'autre si l'option `sidebyside` est employée.

### Avertissement 9 – Exemples de codes avec résultats : possiblement flottants

Si l'option `sidebyside` est passée à l'un ou l'autre des environnements `ltx-code-result` et `ltx-code-external-result`, l'exemple :

- présente ses code et résultat en regard ce qui rend impossible sa coupure en frontière de page;
- est alors automatiquement flottant.

Dans le cas où cet exemple (n° `<n>`) s'avère se trouver sur une page (`<q>`) autre que celle (`<p>`) de son point d'insertion, deux références croisées sont automatiquement insérées :

**une « avant »** : au point d'insertion de l'exemple pour indiquer qu'il est à consulter plus loin ; son texte, *par défaut* « Cf. exemple `<n>` page `<q>`. », peut être surchargé au moyen de l'option `reference text`<sup>P.15</sup> ;

**une « arrière »** : à la fin du titre de l'exemple ; son texte est « (cf. page `<p>`) ».

Ces trois environnements admettent des `<options>` :

- (toutes) celles acceptées par l'environnement `tcblisting` et la commande `\newtcblisting` fournis par la bibliothèque `listings` du package `tcolorbox`<sup>12</sup>. Elles permettent notamment de surcharger les réglages par défaut, par exemple :

12. Et, aussi, les commandes `\DeclareTCBListing` et assimilées fournies par la bibliothèque `xparse` de ce package.

- de faire figurer l'éventuel résultat, non pas sous le code comme c'est le cas par défaut, mais en regard (à droite) au moyen de l'option `sidebyside`;
- de supprimer les numéros de ligne au moyen de l'option

## Exemple 11

```
1 listing options={numbers=none}
```

- trois spécifiques à ces environnements :

`title addon=<supplément>` (init. *vide*, par déf. *aucune*)

Cette option permet d'ajouter au titre de ces exemples, qui sont par défaut et automatiquement « Exemple  $\langle n \rangle$  », un  $\langle \text{supplément} \rangle$ .

`result width=<longueur>` (init. `\linewidth`, par déf. *aucune*)

Cette option, utile seulement pour l'environnement `ltx-code-external-result`<sup>P.14</sup>, permet de spécifier une largeur autre que celle initiale pour le fichier image du résultat, compilé indépendamment de la *Lettre*.

`reference text=<texte>` (init. Cf. exemple  $\langle n \rangle$  page  $\langle q \rangle$ ., par déf. *aucune*)

Cette option n'a d'effet que :

- avec l'un ou l'autre des environnements `ltx-code-result`<sup>P.14</sup> et `ltx-code-external-result`<sup>P.14</sup>;
- lorsque l'option `sidebyside` leur est passée;
- lorsque l'exemple s'avère se trouver sur une page autre que celle de son point d'insertion.

Elle permet alors de surcharger le texte « Cf. exemple  $\langle n \rangle$  page  $\langle q \rangle$ . » automatiquement inséré au point d'insertion de l'exemple (cf. avertissement 9 page précédente)<sup>13</sup>.

Avertissement 10 – `reference text` avant `sidebyside`

Pour qu'elle soit prise en compte, l'option `reference text` doit être passée *avant* l'option `sidebyside`.

## Coloration syntaxique

Par défaut, en début de document et de chaque fichier importé au moyen de `\inputarticle`<sup>P.6</sup>, le langage supposé dans ces exemples de codes est  $\text{T}\text{E}\text{X}$ , chargé (seulement) avec ses « dialectes » :

- `primitive`, `common`, `plain`, `LaTeX`, `AllaTeX` fournis par le package `listings`;
- classes fournis par classe `letgut` (répertoriant les classes disponibles sur le CTAN).

Une conséquence notable est la suivante :

## Avertissement 11 – Coloration syntaxique réduite par défaut

La coloration syntaxique dans les exemples de codes n'est par défaut active que pour le langage  $\text{T}\text{E}\text{X}$  et ses dialectes `primitive`, `common`, `plain`, `LaTeX`, `AllaTeX` et classes.

13. Il est par exemple possible de s'affranchir de ce texte en recourant à `reference text={}`.

Il est néanmoins possible de spécifier d'autres langages et dialectes au moyen de la commande `\syntaxhl`<sup>P.16</sup> suivante, à insérer avant le début de l'exemple de code concerné.

```
\syntaxhl{<liste de dialectes>}
\syntaxhl[<langage>]{<liste de dialectes>}
```

Cette commande permet de spécifier :

- un `<langage>` (par défaut `TEX`);
  - une `<liste de dialectes>`, séparés par des virgules;
- auxquels on souhaite que s'applique la coloration syntaxique.

#### Avertissement 12 – Dialectes colorés syntaxiquement seulement si définis

Ceci suppose que ces langages et dialectes sont définis (et saisis selon la syntaxe du package `listings`) dans le fichier `letgut-1stlang.sty` situé :

- soit dans le dossier de la *Lettre* en cours;
- soit dans le dossier parent de celui de la *Lettre* en cours;
- soit dans un dossier de la `TDS`

Pour le langage `TEX`, ces dialectes sont essentiellement les classes et les packages `(LA)TEX` et un exemple de déclaration de tel dialecte est fourni section « [Exemple de déclaration de dialecte du langage T<sub>E</sub>X](#) », page 30.

## Entrées et sorties dans un terminal

Afin de présenter aisément et de façon homogène des exemples de commandes entrées et éventuellement de leurs sorties correspondantes, la classe `letgut` fournit la commande à arguments « *verbatim* » suivante.

```
\terminal[<prompt>][<options>]{<stdin>}{<stdout>}
```

Cette commande affiche les codes en entrée (`<stdin>`) et en sortie (`<stdout>`), chacun des deux étant possiblement vide.

Le `<prompt>`, ou « invite de commande », est par défaut le symbole `$` affiché en rouge.

Il est possible de surcharger les réglages par défaut de cette commande au moyen d'`<options>` qui sont (toutes) celles acceptées par l'environnement `tcblisting` et la commande `\newtcblisting` fournis par la bibliothèque `listings` du package `tcolorbox`<sup>14</sup>.

Ainsi le code suivant :

#### Exemple 12

```
1 \terminal{time rg foobar -g "*.sty" "/home/bitouze/texlive/2022"}{
2 /home/bitouze/texlive/2022/texmf-dist/tex/latex/skeyval/skeyval.sty
3 445:% \usepackage[option1,option2]{foobar}
4 447:% \expandafter\show\csname foobar.sty.poxkeys\endcsname
5
6 /home/bitouze/texlive/2022/texmf-dist/tex/latex/thmtools/thm-restate.sty
7 197:%% support for keyval-style: restate=foobar
8
9 /home/bitouze/texlive/2022/texmf-dist/tex/latex/pinlabel/pinlabel.sty
```

14. Et, aussi, les commandes `\DeclareTCBListing` et assimilées fournies par la bibliothèque `xparse` de ce package.



```

10 284:\edef\foobar{[width=\@p@swidth sp,height=\@p@sheight sp]{\@p@dffile}}%
11 286:\@message{\string\@includegraphics@{foobar}}%
12 287:\rlap{\smash{\expandafter\@includegraphics@{foobar}}}%
13
14 /home/bitouze/texlive/2022/texmf-dist/tex/latex/theoremref/theoremref.sty
15 129: its label by ``\thlabel{foobar}``. For example,
16
17 /home/bitouze/texlive/2022/texmf-dist/tex/latex/qrbill/qrbill.sty
18 12:%% Marei Peischl (peiTeX) and Alex Antener (foobar LLC).
19 rg -g 0,25s user 0,23s system 320% cpu 0,150 total
20 }

```

donne-t-il :

```

1 $ time rg foobar -g "*.sty" "/home/bitouze/texlive/2022"
1 /home/bitouze/texlive/2022/texmf-dist/tex/latex/skeyval/skeyval.sty
2 445:%% \usepackage[option1,option2]{foobar}
3 447:%% \expandafter\show\csname foobar.sty.poxkeys\endcsname
4
5 /home/bitouze/texlive/2022/texmf-dist/tex/latex/thmtools/thm-restate.sty
6 197:%% support for keyval-style: restate=foobar
7
8 /home/bitouze/texlive/2022/texmf-dist/tex/latex/pinlabel/pinlabel.sty
9 284:\edef\foobar{[width=\@p@swidth sp,height=\@p@sheight sp]{\@p@dffile}}%
10 286:\@message{\string\@includegraphics@{foobar}}%
11 287:\rlap{\smash{\expandafter\@includegraphics@{foobar}}}%
12
13 /home/bitouze/texlive/2022/texmf-dist/tex/latex/theoremref/theoremref.sty
14 129: its label by ``\thlabel{foobar}``. For example,
15
16 /home/bitouze/texlive/2022/texmf-dist/tex/latex/qrbill/qrbill.sty
17 12:%% Marei Peischl (peiTeX) and Alex Antener (foobar LLC).
18 rg -g 0,25s user 0,23s system 320% cpu 0,150 total

```

## Caractères d'échappement et de raccourci pour les extraits de code

### Avertissement 13 – Caractère d'échappement des listings

La classe `letgut` définit `£` comme caractère d'échappement dans  $\LaTeX$  au sein d'un listing.

Au besoin, on pourra désactiver ce caractère actif au moyen de

#### Exemple 13

```
1 \lstset{escapechar=}
```

### Avertissement 14 – Équivalent court de `\lstinline`

Les extraits de code peuvent être saisis au moyen de la commande `\lstinline` du package `listings` mais, pour simplifier la tâche, la classe `letgut` définit comme équivalent court de `\lstinline` le caractère unicode <sup>™</sup> (U+2122) <sup>a</sup>.

<sup>a</sup>. Peu susceptible d'être utilisé dans du texte ordinaire.

Autrement dit, la classe `letgut` contient l'instruction <sup>15</sup> :

15. À peu de choses près.

## Exemple 14

```
1 \lstMakeShortInline™
```

Au besoin, on pourra désactiver ce caractère actif au moyen de :

## Exemple 15

```
1 \lstDeleteShortInline™
```

## Remarque 4 – Obtention du caractère ™

Le caractère ™ s’obtient :

- sous GNU/Linux : `⌘`+`AltGr`+`8`<sup>a</sup> ;
- sous macOS : à l’aide du visualiseur de caractères<sup>b</sup> ;
- sous Windows : `Alt`+`0`+`1`+`5`+`3`.

a. Touche `8` du clavier principal.

b. Et possiblement de raccourcis claviers personnels.

## Nouveautés apparues sur le CTAN

Afin de pouvoir plus aisément lister les nouveautés (packages et classes  $\LaTeX$ , etc.) apparues sur le CTAN, la classe `letgut` fournit le nouvel environnement de liste `ctannews`, similaire à l’environnement `description`.

```
\begin{ctannews}
  <liste des nouveautés>
\end{ctannews}
```

Cet environnement permet de dresser la `<liste des nouveautés>` apparues sur le CTAN.

Chaque `<nouveauté>` est introduite au moyen de la commande `\item` suivante.

```
\item[<nom>]
\item*[<nom>]
```

Cette commande affiche le `<nom>` de la `<nouveauté>` comme ce serait le cas pour le « label » d’une liste de description, ce qui permet ensuite de décrire la `<nouveauté>` en question. Le `<nom>` est en outre un lien hypertexte vers sa page sur le CTAN (<https://ctan.org/pkg/<nom>>).


La version étoilée `\item*` est dédiée aux nouveautés œuvres de contributeurs francophones et le logo de la francophonie, alors automatiquement situé en regard dans la marge, les signale comme telles.

## Exemple 16 : nouveautés


```
1 \begin{ctannews}
2 \item[nl-interval] vise à simplifier le processus de
3   représentation graphique des intervalles de l'axe réel.
4 \item*[matapli] classe \hologo{LaTeX} destinée à la composition
5   de la revue Matapli (conçue par \person{Chupin, Maxime},
6   secrétaire adjoint de \gutenberg{}).
7 \end{ctannews}
```

résultat (suite)

`nl-interval` : vise à simplifier le processus de représentation graphique des intervalles de l'axe réel.

 `matapli` : classe L<sup>A</sup>T<sub>E</sub>X destinée à la composition de la revue Matapli (conçue par Maxime CHUPIN, secrétaire adjoint de GUTenberg).

### `\francophony`

Cette commande affiche le logo de la francophonie, ainsi : .

## Fiches de lecture

Les fiches de lecture d'un livre sont créées au moyen de l'environnement `bookreview` suivant.

```
\begin{bookreview}{\langle caractéristiques \rangle}
  \langle fiche de lecture \rangle
\end{bookreview}
```

Cet environnement permet de mettre en page une `\langle fiche de lecture \rangle` caractérisée par les `\langle caractéristiques \rangle` suivantes qui sont, selon les cas :

### obligatoires :

`title=\langle titre \rangle` (init. *vide*, par déf. *aucune*)

Cette clé permet de spécifier le `\langle titre \rangle` introductif de la fiche.

`reviewer=\langle rapporteur \rangle` (init. *vide*, par déf. *aucune*)

Cette clé permet de spécifier le `\langle rapporteur \rangle` de la fiche, à spécifier selon la syntaxe de la commande `\author`<sup>P.7</sup>.

`bibkey=\langle clé \rangle` (init. *vide*, par déf. *aucune*)

Cette clé permet de spécifier la `\langle clé \rangle` identifiant l'entrée d'un fichier `.bib` contenant les données bibliographiques du document rapporté.

### Avertissement 15 – Fichier de bibliographie

Ces données bibliographiques doivent se trouver dans un fichier `\langle bibliographie \rangle.bib`, structurées selon le format du package `biblatex` et chargées en préambule (par exemple dans le fichier local de configuration, cf. page 26) au moyen de la commande :

#### Exemple 17

```
1 \addbibresource{\langle bibliographie \rangle.bib}
```

### fortement conseillée :

`frontcover=\langle fichier \rangle` (init. *vide*, par déf. *aucune*)

Cette clé permet de spécifier le `\langle fichier \rangle` image de la couverture du document rapporté.

### facultative :

`price=\langle prix \rangle` (init. *vide*, par déf. *aucune*)

Cette clé permet le cas échéant de spécifier le `\langle prix \rangle` du document rapporté.

## Acronymes

Nombreux sont les articles de la *Lettre* susceptibles de contenir des acronymes peut-être pas connus de tous. Aussi est-il opportun que, lors de leur première occurrence, ceux-ci soient explicités.

Pour automatiser cela, la classe `letgut` s'appuie sur le package `acro`; toutefois, pour à la fois simplifier la création desdits acronymes et étendre (légèrement) les fonctionnalités offertes par `acro`, elle fournit la commande dédiée `\letgutacro`.

```
\letgutacro[options]{<COURT>}{<long>}[<traduction française>]
```

Cette commande permet de créer un nouvel acronyme en spécifiant :

- sa forme courte `<COURT>`, *obligatoirement en grandes capitales*;
- sa forme longue `<long>`.

En outre, le 1<sup>er</sup> et 2<sup>e</sup> arguments *optionnels* permettent de, respectivement :

- passer à la commande `\DeclareAcronym` (de création d'acronymes du package `acro` agissant en sous-main) des `<options>` qui lui sont propres, permettant ainsi de surcharger les options par défaut passées à cette commande par `\letgutacro`;
- signaler que l'acronyme provient de l'anglais et d'en spécifier la `<traduction française>` (éventuellement vide si celle-ci n'est pas pertinente).

L'acronyme ainsi créé a pour identifiant `<court>`, c'est-à-dire la version *en bas de casse* de `<COURT>`, et peut donc être employé au moyen des commandes fournies par le package `acro`, par exemple :

- `\ac{<court>}`<sup>16</sup>;
- `\acs{<court>}`<sup>17</sup>.

Toutefois, pour simplifier l'usage de ces acronymes, la classe `letgut` crée alors automatiquement une commande `\<court>`<sup>18</sup> qui agit comme :

- `\ac{<court>}` en version non étoilée;
- `\acs{<court>}` en version étoilée.

Ainsi, l'acronyme utilisé via `\<court>` figure, pour ses occurrences :

**première** : sous la forme `<COURT>`<sup>19</sup> suivi d'une note de bas de page contenant « `<long>`. »;

**suivantes** : sous la forme `<COURT>`.

En outre :

- cette commande peut être utilisée sans restriction en argument des commandes `\title→P.6`, `\subtitle→P.6`, `\section→P.6`, `\subsection→P.6`, etc. et l'acronyme figure sous sa forme `<COURT>` dans les *bookmarks* (signets);
- un copié de `<COURT>` colle `<COURT>`.

16. Acronyme automatiquement affiché sous sa forme complète à sa 1<sup>re</sup> occurrence, sous sa forme courte à ses occurrences suivantes.

17. Acronyme affiché sous sa forme courte seulement.

18. Sauf si elle existe déjà, auquel cas la création d'une telle commande est silencieusement escamotée. Ainsi par exemple, l'acronyme `\letgutacro[...]{TIKZ}{...}[...]` fourni par `letgut` (cf. page 31) ne surcharge-t-il pas la commande `\tikz` fournie notamment par le package `tikz`.

19. C'est-à-dire `<court>` en petites capitales.

**Avertissement 16 – `\letgutacro` : uniquement en préambule**

La définition d'acronymes au moyen de `\letgutacro` ne peut se faire qu'en préambule.

Ainsi, avec les définitions suivantes en préambule :

**Exemple 18 : définition d'acronymes**

```

1 % Acronyme français
2 \letgutacro{LD}{\LaTeX{}} dépendant}
3
4 % Acronyme anglais avec traduction française
5 \letgutacro{TLMGR}{\TeX~Live Manager}[gestionnaire \TeX~Live]
6
7 % Acronyme anglais sans traduction française
8 \letgutacro{CD}{Compact Disc}[]
9
10 % Acronyme anglais avec surcharge :
11 % - `short=no-op` : l'acronyme est « no-op » (en bas de casse)
12 % et le nom de la commande sous-jacente ne peut être \no-op
13 % (tiret interdit)
14 % - `short-format=\upshape` : les petites capitales sont
15 % désactivées
16 \letgutacro[short=no-op,short-format=\upshape]
17 {NOOP}
18 {No Operation}
19 [instruction nulle]
20
21 % Acronyme en allemand :
22 \letgutacro[
23   short=\emph{Ti\emph{k}Z},
24   short-format=\em,
25   foreign-babel=german,
26   foreign-locale=allemand]
27 {TIKZ}
28 {Ti\emph{k}Z ist \emph{kein} Zeichenprogramm}
29 [Ti\emph{k}Z \emph{n'}est \emph{pas} un programme de dessin]

```

a-t-on :

**Exemple 19 : utilisation d'acronymes**

code

```

1 On dispose désormais pour \enquote{\LaTeX{}} dépendant}
2 d'un acronyme qu'on peut utiliser par exemple
3 \begin{itemize}
4 \item ainsi : \ac{ld} ou \acs{ld} ;
5 \item ou bien ainsi : \ld{} ou \ld*{}.
6 \end{itemize}
7
8 On peut également employer les acronymes :

```

```

9  \begin{itemize}
10 \item \cd{}, \cd{} ;
11 \item \tlmgr{}, \tlmgr{} ;
12 \item \noop{}, \noop{} ;
13 \item \ac{tikz}, \ac{tikz}. % Noter le non emploi de '\tikz'
14 \end{itemize}

```

résultat

On dispose désormais pour « L<sup>A</sup>T<sub>E</sub>X dépendant » d'un acronyme qu'on peut utiliser par exemple

- ainsi : `LD` ou `LD`;
- ou bien ainsi : `LD` ou `LD`.

On peut également employer les acronymes :

- `CD`, `CD`;
- `TLMGR`, `TLMGR`;
- `no-op`, `no-op`;
- `TIKZ`, `TIKZ`.

Les noms des commandes sous-jacentes ne doivent contenir que des lettres, mais celles-ci peuvent être accentuées. Ainsi, avec la définition suivante en préambule :

Exemple 20 : définition d'acronyme avec lettres accentuées

```

1  \letgutacro{CPGÉ}{Classes Préparatoires aux Grandes Écoles}

```

a-t-on :

Exemple 21 : utilisation d'acronyme avec lettres accentuées

```

1  On peut également employer l'acronyme \cpgé{}, \cpgé{}.
```

code

résultat

On peut également employer l'acronyme `CPGÉ`, `CPGÉ`.

La classe `letgut` fournit un fichier `letgut-acronyms.tex` dans lequel sont définis plusieurs acronymes anglais et français, directement utilisables. Ceux-ci sont répertoriés page 31.

## Séparateurs

Il est parfois utile d'accentuer la séparation entre les articles de la *Lettre*. Ceci peut se faire au moyen de la commande `\separator` qui insère un filet horizontal.

```

\separator

```

Cette commande permet d'accentuer la séparation entre deux articles.

## Annonces

Afin de présenter aisément et de façon homogène les annonces à paraître dans la *Lettre*, la classe `letgut` fournit l'environnement `announcement` <sup>→ p. 23</sup>.

```
\begin{announcement}[<couleur>]{<titre>}
  <annonce>
\end{announcement}
```

Cet environnement est dédié aux *<annonce>*s. La spécification :

- d'un *<titre>* (pouvant toutefois être vide) est obligatoire ;
- d'une *<couleur>* (du tire et des ornements) autre que celle (noire) par défaut est possible.

### Exemple 22 : annonces

```
1 \begin{announcement}[red!35!black]{Convocation \acs{ag} ordinaire}
2   Les adhérents de l'\assogut{} sont invités à participer à
3   l'\textbf{assemblée générale \emph{ordinaire}} de l'association
4   le \textbf{dimanche 11 décembre 2022}.
5 \end{announcement}
```

résultat



## CONVOCATION AG ORDINAIRE

Les adhérents de l'association GUTenberg  
sont invités à participer à l'**assemblée  
générale *ordinaire*** de l'association  
le **dimanche 11 décembre 2022**.

## Rébus

Afin de présenter aisément et de façon homogène les rébus à paraître dans la *Lettre*, la classe `letgut` fournit l'environnement `rebus`.

```
\begin{rebus}[<options>]
  <rébus>
\end{rebus}
```

Cet environnement affiche un *<rébus>* et s'emploie différemment selon que la solution est prévue de figurer dans la *Lettre* :

**en cours :** le rébus et sa solution doivent alors être séparés par la commande `\solution` <sup>→ p. 24</sup> ;

**suivante :** l'option `no solution` <sup>→ p. 24</sup> doit alors être employée.

La boîte contenant le rébus mentionne que sa solution se trouve, respectivement :

- « page  $\langle n \rangle$  » ou « ci-dessous » ;
- « dans la prochaine *Lettre* ».

La commande séparant le rébus et sa solution prévue de figurer dans la *Lettre* en cours est :

#### `\solution`

Cette commande débute la solution d'un rébus à l'intérieur d'un environnement `rebus`<sup>23</sup>.

En outre, cet environnement admet des  $\langle options \rangle$  :

- (toutes) celles acceptées par les environnements `tcolorbox` et assimilés du package `tcolorbox`, destinées à, le cas échéant, modifier la mise en forme par défaut de la boîte contenant le rébus ;
- deux qui lui sont propres :

`title addon= $\langle supplément \rangle$`  (init. *vide*, par déf. *aucune*)

Cette option permet d'ajouter au titre d'un rébus, qui est par défaut et automatiquement « Rébus », un  $\langle supplément \rangle$  alors en italique et entre parenthèses.

`no solution` (init. *false*, par déf. *true*)

Cette option est à spécifier pour un rébus dont la solution ne doit figurer que dans la prochaine *Lettre*.

Ainsi le code :

#### Exemple 23 : rébus

```

1 \begin{rebus}[no solution]
2   Rébus \emph{sans} solution dans le présent numéro.
3 \end{rebus}
4 \begin{rebus}
5   Rébus \emph{avec} solution dans le présent numéro.
6   \solution
7   Solution du rébus.
8 \end{rebus}
```

donne-t-il les résultats suivants<sup>20</sup>.

#### ☞ Rébus

Rébus *sans* solution dans le présent numéro.

*Solution dans la prochaine Lettre*

#### ☞ Rébus

Rébus *avec* solution dans le présent numéro.

*Solution page ci-contre*

La classe `letgut` fournit bien sûr le moyen d'afficher les solutions des rébus de la *Lettre* soit en cours, soit précédente.

`\rebussolution[ $\langle solution \rangle$ ][ $\langle numéro \rangle$ ][ $\langle options \rangle$ ]`

20. Éventuellement pas immédiatement à la suite car les boîtes contenant les rébus sont flottantes.



Cette commande affiche la ou les solutions du ou des rébus qui figurent dans la *Lettre* :

**en cours** : (commande `\solution`<sup>p.24</sup>) si ses 1<sup>er</sup> et 2<sup>e</sup> arguments optionnels `<solution>` et `<numéro>` sont inutilisés ;

**précédente** : si son 1<sup>er</sup> argument optionnel `<solution>` est utilisé. Selon que 2<sup>e</sup> argument optionnel `<numéro>` est spécifié ou pas, il est indiqué que la `<solution>` est celle d'un rébus figurant dans la *Lettre* :

- `<numéro>` ;
- précédente.

Cette commande admet comme `<options>` (toutes) celles acceptées par les environnements `tcolorbox` et assimilés du package `tcolorbox`, destinées à, le cas échéant, modifier la mise en forme par défaut de la boîte contenant la solution du rébus.

Ainsi le code :

#### Exemple 24 : solution de rébus

```
1 \rebussolution
2 \rebussolution[Solution du rébus]
3 \rebussolution[Solution du rébus][49]
```

donne-t-il donne-t-il les résultats suivants<sup>21</sup>.

#### ☞ **Solution du rébus page précédente**

Solution du rébus.

#### ☞ **Solution du rébus de la *Lettre* précédente**

Solution du rébus

#### ☞ **Solution du rébus de la *Lettre* 49**

Solution du rébus

## Boîtes d'alertes

```
\alertbox[<couleur>]{<texte>}
```

Cette commande insère une boîte d'alerte :

- optionnellement de `<couleur>` de fond (à spécifier selon le modèle « *named* ») autre que celle par défaut<sup>22</sup> ;
- contenant le `<texte>` (qui peut contenir plusieurs paragraphes).

#### Exemple 25 : boîtes d'alertes

```
1 \alertbox{%
2   Adhérez, adhérez, il en restera toujours quelque chose !%
```

21. Éventuellement pas immédiatement à la suite car les boîtes contenant les solutions des rébus sont flottantes.

22. C'est-à-dire celle-ci.

```

3 }
4 \alertbox[yellow]{%
5   Adhérez, adhérez !
6
7   Il en restera toujours quelque chose...%
8 }

```

code (suite)

---

résultat

**Adhérez, adhérez, il en restera toujours quelque chose!**

**Adhérez, adhérez!**  
**Il en restera toujours quelque chose...**

## Fichier local de configuration

Chaque numéro de la *Lettre* nécessite certaines configurations locales : configuration dédiée au numéro en question, packages particuliers utilisés dans les articles, configurations propres du package `listings`, etc. Afin de ne pas encombrer le fichier `.tex` principal de la *Lettre*, un fichier de configuration locale nommé `localconf.tex` est, si présent dans le répertoire courant, automatiquement inclus à la fin du préambule.

## ASPECTS DE LA 1<sup>re</sup> PAGE

La première page de la *Lettre* comporte :

- une bannière sous forme d'un très grand « L » en noir sur lequel figurent de la couleur de fond de la page, en gras, dans sa partie :
  - verticale, « La » puis, chacune sur une ligne, les lettres du mot « Lettre » en grandes capitales;
  - horizontale, « GUTenberg ».
- un très grand « g », en filigrane et en gris clair.

La fonte de ces deux éléments est la principale utilisée (spécifiée au moyen de `\setmainfont`).

Pour ce faire, la classe charge le package maison `letgut-banner` qui n'est pas décrit ici.

## LES DINOSAURES, LEUR ÉCOSYSTÈME ET LETGUT

Pour à la fois :

- disposer d'une version suffisamment récente de Lua<sup>L</sup>A<sup>T</sup>E<sup>X</sup> pour la présente classe `letgut` (cf. avertissement 1 page 2);
- éviter de perturber une installation existante de (L<sup>A</sup>)T<sub>E</sub>X;

on pourra recourir à Docker<sup>23</sup> dont cette section est un mode d'emploi :

<sup>23</sup>. Cerise sur le gâteau : un temps de compilation éventuellement réduit de façon significative. Ainsi, celui de la présente documentation est-il sur la machine de Denis Bitouzé d'un peu plus de 13 s avec Docker et de plus de 30 s par le biais habituel.

- `express` ;
- axé sur Linux, mais qui devant s'appliquer au moins en partie aux autres systèmes d'exploitation ;
- axé sur la  $\text{T}_{\text{E}}\text{X}$  Live.

#### Avertissement 17 – Commande `sudo` peut-être nécessaire

Les commandes `docker` ci-après ne sont pas précédées de `sudo` mais, selon les systèmes d'exploitation, elles peuvent devoir l'être.

On commence par installer `docker` puis à lancer le service `docker`<sup>24</sup>.

Ensuite, par exemple depuis un dossier où se trouve un fichier `mon-fichier.tex` (disons à compiler avec  $\text{Lua}_{\text{L}}\text{T}_{\text{E}}\text{X}$ ), on lance la longue commande suivante (*qui doit être sur une seule ligne*<sup>25</sup>) :

```
1 $ docker run -i --rm --name latex -v "$PWD":/usr/src/app -w /usr/
   src/app registry.gitlab.com/islandoftex/images/texlive:
   latest-with-cache lualatex mon-fichier
```

La toute première fois, cela provoque le téléchargement de plusieurs fichiers, dont celui assez lourd de l'image d'une version allégée de la  $\text{T}_{\text{E}}\text{X}$  Live 2021 (délestée des sources et des documentations) puis lance la compilation demandée.

Pour simplifier les compilations ultérieures, on aura intérêt à créer dans son fichier `.bashrc` (ou `.zshrc`, etc.) un ou plusieurs alias de la forme :

```
1 alias docker-texlive='docker run -i --rm --name latex -v
   "$PWD":/usr/src/app -w /usr/src/app
   registry.gitlab.com/islandoftex/images/texlive:latest-with-cache'
2 alias docker-pdflatex='docker-texlive pdflatex'
3 alias docker-xelatex='docker-texlive xelatex'
4 alias docker-lualatex='docker-texlive lualatex'
5 alias docker-biber='docker-texlive biber'
6 alias docker-makeglossaries='docker-texlive makeglossaries'
7 alias docker-latexmk-pdf='docker-texlive latexmk -pdf'
8 alias docker-latexmk-xe='docker-texlive latexmk -pdfxe'
9 alias docker-latexmk-lua='docker-texlive latexmk -pdflua'
```


pour pouvoir compiler au moyen de seulement<sup>26</sup> :

```
1 $ docker-latexmk-lua mon-fichier
```

## PACKAGES CHARGÉS PAR LETGUT

La classe `letgut` charge en sous-main un certain nombre de packages utiles, voire nécessaires, à son codage. Elle en charge également certains pas indispensables, mais considérés comme « incontournables » pour que les auteurs de la *Lettre* puissent

24. Pour Ubuntu, cf. par exemple [ce guide](#).

25. Pour la copier d'un seul bloc, il devrait suffire de *copier* (et non de *cliquer sur*) l'icône suivante 

26. En lançant préalablement `source ~/.bashrc` (ou assimilé) afin de pouvoir en bénéficier dans un terminal déjà ouvert.

(aisément) composer un « joli » document. Nous en dressons ci-après la liste en les regroupant selon ces deux catégories.

## Packages utiles aux auteurs de la *Lettre*

`fontspec` : fontes `OTF`.

`microtype` : raffinements subliminaux vers la perfection typographique.

**Options** : `stretch=30`, `shrink=25`, `letterspace=150`.

`graphicx` : prise en charge améliorée des graphiques.

`array` : extension des environnements `array` et `tabular`.

`fancyvrb` : notamment pour permettre l'usage de commandes « verbatim » dans les notes de bas de page.

`booktabs` : tableaux de qualité.

`csquotes` : facilités de citations, en ligne et hors-texte, sensibles au contexte.

`amsmath` : nombreux outils utiles pour la composition mathématique.

`mathtools` : étend les fonctionnalités et corrige certaines déficiences d'`amsmath` et de `LATEX`.

**Option** : `fleqn`.

`siunitx` : aide à la saisie et à l'affichage cohérent des nombres, unités et quantités.

**Options** : `locale=FR`, `mode=text`.

`hologo` : collection de logos habituels (`LATEX`, `LATEX 2ε`, etc.) avec support pour les signets.

`xcolor` : accès facile, indépendant du pilote, à plusieurs types de teintes, de nuances, de tons et de mélanges de couleurs arbitraires.

**Option** : `table`.

`ninecolors` : sélection de couleurs avec contraste `WCAG` approprié.

`tabularray` : mise en page de tableaux et de matrices offrant une séparation complète des contenus et styles. Ce package très récent (sorti le 14 mai 2021) est utilisé dans le code de la classe pour la création de boîtes d'alertes « légères », c.-à-d. ne nécessitant notamment pas le chargement (indirect) du package `tikz` qui augmente significativement le temps de compilation ; mais il pourrait (devrait) être utile également aux auteurs de la *Lettre*.

`babel` : support multilingue.

**Options** : `english`, `french` ;

**Configuration** `\renewcommand*\frenchtablename{Tableau}`.

`varioref` : références de pages intelligentes.

**Options** : `nospace`, `french`.

`eurosym` : symbole et montants en €.

**Option** : `right`.

`listings` : composition des listings informatiques.

**Options** : `basicstyle=`, `frame=single`, `belowskip=0pt`<sup>27</sup>.

`floatrow` : nombreuses possibilités de personnalisation de la disposition des flottants.

27. Cette dernière option du fait d'un problème actuel impliquant les packages `parskip` et `listings`.

**Options :**

- `objectset=justified`;
- `style=__letgut_ruled`<sup>28</sup> et `margins=hangleft` pour les figures ;
- `capposition=top` pour les tableaux.

**biblatex** : bibliographies sophistiquées.

**Option :** `sorting=none`.

**acro** : création simple d'acronymes<sup>29</sup>.

**Options :**

- `first-style=footnote`;
- `format/short=\scshape`;
- `format/foreign=\em`;
- `foreign/display`;
- `locale/display`;
- `locale/format=\upshape`.

**hyperref** : prise en charge étendue de l'hypertexte.

**Options :**

- `draft` si l'option de classe `paper`<sup>→ p.3</sup> est utilisée ;
- `colorlinks`, `allcolors=letgut_allcolors_links` sinon.

**hyccap** : ajustement des ancrs des légendes.

**Option :** `all`.

**cleveref** : détermination automatique du format des références en fonction du type de référence.

**Option :** `french`.

**lua-typo** : mise en lumière, par un changement de couleur, des lignes typographiquement imparfaites avec Lua $\LaTeX$ .

**Option :** `All`.

## Packages utiles au codage de la classe `letgut`

**l3keys2e** : traitement  $\LaTeX 2_{\epsilon}$  des options de classe en utilisant les clés  $\LaTeX 3$ .

**parskip** : mise en page de paragraphes séparés par un blanc vertical au lieu (ou en plus) d'un retrait.

**fancyhdr** : contrôle étendu des en-têtes et des pieds de page.

**geometry** : interface flexible et complète pour les dimensions des documents.

**Options :** `a4paper`, `asymmetric`.

**etoc** : tables des matières entièrement personnalisables.

**enumitem** : contrôle de la mise en page de `itemize`, `enumerate`, `description` et permet de cloner les environnements standards et créer de nouveaux environnements. Utilisé pour la création de l'environnement `ctannews`<sup>→ p.18</sup>.

**titlesec** : personnalisation aisée des titres de sections, etc.

28. Style propre à la classe.

29. Pour la gestion des acronymes, il était initialement prévu de recourir au package `glossaries-extra` mais celui-ci augmente significativement le temps de compilation.

**placeins** : contrôle du placement des flottants permettant de s'assurer que ceux d'une section (donc d'un article dans le cas de la *Lettre*) apparaissent avant la commande `\section` suivante.

**Options** : `section`, `above`.

**accsupp** : notamment remplacement de texte lors des copiés-collés, utilisé en particulier pour que les acronymes, composés en petites capitales par `letgut`, une fois copiés, soient collés en grandes capitales.

**letgut-banner** : bannière de la 1<sup>re</sup> page de la *Lettre*.

## Exemple de déclaration de dialecte du langage T<sub>E</sub>X

Nous fournissons ci-dessous un exemple de déclaration de dialecte (ici le package `\LaTeX` `graphicx`) du langage T<sub>E</sub>X (cf. section « [Coloration syntaxique](#) », page 15).

### Exemple 26

```

1 \lst@definlanguage[graphicx]{TeX}[LaTeX]{TeX}{%
2   % Control sequences names
3   moretexcs={%
4     DeclareGraphicsExtensions,DeclareGraphicsRule,graphicspath,%
5     includegraphics*,includegraphics,reflectbox,resizebox*,%
6     resizebox,rotatebox,scalebox,%
7   },%
8   % Keywords of class 1 : keywords that contain other characters
9   % (since of the same class as the ones specified as
10  % 'otherkeywords')
11  morekeywords={%
12  },%
13  % Keywords of class 2 : environments names
14  morekeywords=[2]{%
15  },%
16  % Keywords of class 3 : mandatory arguments (not environments)
17  % & optional arguments which are keys (in key=value)
18  morekeywords=[3]{%
19    draft,final,hiresbb,demo,setpagesize,nosetpagesize,dvips,xdvi,%
20    dvipdf,dvipdfm,dvipdfmx,xetex,pdftex,luatex,dvisvgm,dvipsone,%
21    dviwindo,emtex,dviwin,oztex,textures,pctexps,pctexwin,pctexhp,%
22    pctex32,truetex,tcidvi,vtex,debugshow,hiderotate,hidescale,%
23    alt,%
24    %
25    bb,bbllx,bbllx,bburx,bbury,natwidth,natheight,hiresbb,pagebox,%
26    viewport,trim,angle,origin,width,height,totalheight,%
27    keepaspectratio,scale,clip,type,ext,read,command,quiet,page,%
28    interpolate,decodearray,origin,x,y,units,%
29  },%
30  % Keywords of class 4 : values of keys (in key=value)
31  morekeywords=[4]{%
32    mediabox,cropbox,bleedbox,trimbox,artbox,true,false,%
33  },%
34  % Keywords of class 5 : arguments specifications (after ":"
35  % in expl3 syntax)

```

```

36     morekeywords=[5]{%
37     },%
38     % Keywords of class 6 : current package name (and possibly
39     % derived packages)
40     morekeywords=[6]{%
41         graphicx,%
42     },%
43     % otherkeywords={},%
44     alsoletter={23},%
45     % alsodigit={},%
46     sensitive,%
47 }[keywords,tex,comments]%

```

## ☞ LISTE DES ACRONYMES PRÉDÉFINIS PAR LETGUT

**AG** Assemblée Générale

**APA** *American Psychological Association* (association américaine de psychologie)

**APMEP** Association des Professeurs de Mathématiques de l'Enseignement Public

**ASCII** *American Standard Code for Information Interchange* (code américain normalisé pour l'échange d'information)

**BBB** BigBlueButton

**BSD** *Berkeley Software Distribution* (collection de logiciels de Berkeley)

**BTS** Brevet de Technicien Supérieur

**CA** Conseil d'Administration

**CPGÉ** Classes Préparatoires aux Grandes Écoles

**CSS** *Cascading Style Sheets* (feuilles de style en cascade)

**CSV** *Comma-separated values* (valeurs séparées par des virgules)

**CTAN** *Comprehensive T<sub>E</sub>X Archive Network* (réseau complet d'archives T<sub>E</sub>X)

**CV** Curriculum Vitæ

**DNS** *Domain Name System* (système de noms de domaine)

**DOI** *Digital Object Identifier* (identifiant numérique d'objet)

**DVI** *DeVice-Independent* (indépendant du type de périphérique)

**DVIPS** *DVI-to-PS (translator)* ((convertisseur) DVI vers PS)

**ECM** Exemple Complet Minimal

**FAQ** *Frequently Asked Questions* (questions fréquemment posées)

**GPL** *GNU General Public License* (licence publique générale GNU)

**GRAPPA** Groupe de recherche en Apprentissage Automatique

**HTML** *HyperText Markup Language* (langage de balises pour l'hypertexte)

**IMAP** *Internet Message Access Protocol*

**INSPÉ** Instituts Nationaux Supérieurs du Professorat et de l'Éducation

**IPA** *International Phonetic Alphabet* (alphabet phonétique international)

**IREM** Instituts de Recherche sur l'Enseignement des Mathématiques

**ISO** *International Organization for Standardization* (organisation internationale de normalisation)

- ISSN** *International Standard Serial Number* (numéro international normalisé des publications en série)
- LDAP** *Lightweight Directory Access Protocol* (protocole d'accès à répertoire léger, système garantissant l'authentification d'utilisateurs de plusieurs services dispersés)
- MEEF** Métiers de l'Enseignement, de l'Éducation et de la Formation
- NDLR** Note de la Rédaction
- NFSS** *New Font Selection Scheme* (nouveau schéma de sélection de fonte)
- OFF** *Open Font Format* (format de fonte ouvert)
- OFL** *SIL Open Font License*
- ORCID** *Open Researcher and Contributor ID* (identifiant ouvert pour chercheur et contributeur)
- OS** *Operating System* (système d'exploitation)
- OTF** *Open Type Format*
- PAO** Publication Assistée par Ordinateur
- PDF** *Portable Document Format* (format de document portable)
- PGF** *Portable Graphics Format* (format de graphiques portable)
- PS** *PostScript*
- RGPD** Règlement Général sur la Protection des Données
- RTF** *Rich Text Format* (format de texte enrichi)
- SHS** Sciences Humaines et Sociales
- SMAI** *Société de Mathématiques Appliquées et Industrielles*
- SMTP** *Simple Mail Transfer Protocol* (protocole simple de transfert de courrier)
- SVG** *Scalable Vector Graphics* (graphiques vectoriels extensibles)
- TDS** *T<sub>E</sub>X Directory Structure* (structure de répertoire de T<sub>E</sub>X)
- TIKZ** *TikZ ist kein Zeichenprogramm* (TikZ n'est pas un programme de dessin)
- TIPA** *T<sub>E</sub>X IPA or Tokyo IPA* (alphabet phonétique international)
- TUG** *T<sub>E</sub>X User Group* (groupe (international) d'utilisateurs de T<sub>E</sub>X)
- UCA** *Unicode Collation Algorithm* (algorithme de tris d'Unicode)
- URL** *Uniform Resource Locator* (localisateur uniforme de ressource ou, plus clairement, format des adresses du Web)
- UTF** *Universal (Character Set) Transformation Format* (format de transformation (du jeu) universel (de caractères codés))
- VPS** *Virtual Private Server* (serveur dédié virtuel)
- WCAG** *Web Content Accessibility Guidelines* (directives sur l'accessibilité du contenu Web)
- WYSIWYG** *What You See Is What You Get* (ce que vous voyez est ce que vous obtenez)
- XML** *eXtensible Markup Language* (langage de balisage extensible)
- YAML** *YAML Ain't Markup Language* (YAML n'est pas un langage de balisage (acronyme récursif))

## RÉFÉRENCES

- [1] ASSOCIATION GUTENBERG. *Lettre GUTenberg*. Groupe francophone des Utilisateurs de T<sub>E</sub>X, L<sup>A</sup>T<sub>E</sub>X et logiciels compagnons. URL : <https://www.gutenberg.eu.org/-Lettre-GUTenberg->.



## 🔗 TABLE DES AVERTISSEMENTS

|    |  |    |
|----|--|----|
| 1  | Lua $\LaTeX$ (récent) et UTF-8 nécessaires . . . . .   | 2  |
| 2  | Options de <code>letgut</code> : pas en argument optionnel de <code>\documentclass</code> . . . . .                        | 3  |
| 3  | <code>\title</code> et <code>\author</code> $\neq$ avant et après <code>\begin{document}</code> . . . . .                  | 5  |
| 4  | Commande <code>\maketitle</code> à <i>ne pas</i> employer . . . . .  | 5  |
| 5  | Importation de fichiers d'articles . . . . .   | 6  |
| 6  | Commandes pas toutes bienvenues en <code>\title</code> et <code>\subtitle</code> . . . . .                                 | 9  |
| 7  | Commande <code>\letgutcls</code> <sup>p. 10</sup> pas bienvenue en <code>\title</code> et <code>\subtitle</code> . . . . . | 11 |
| 8  | Touche de clavier du symbole € manquant . . . . .  | 13 |
| 9  | Exemples de codes avec résultats : possiblement flottants . . . . .  | 14 |
| 10 | <code>reference text</code> <sup>p. 15</sup> avant <code>sidebyside</code> . . . . .                                       | 15 |
| 11 | Coloration syntaxique réduite par défaut . . . . .   | 15 |
| 12 | Dialectes colorés syntaxiquement seulement si définis . . . . .  | 16 |
| 13 | Caractère d'échappement des listings . . . . .   | 17 |
| 14 | Équivalent court de <code>\lstinline</code> . . . . .  | 17 |
| 15 | Fichier de bibliographie . . . . .   | 19 |
| 16 | <code>\letgutacro</code> <sup>p. 20</sup> : uniquement en préambule . . . . .  | 21 |
| 17 | Commande <code>sudo</code> peut-être nécessaire . . . . .  | 27 |

## INDEX

Afin de différencier leurs natures, les entrées de cet index sont affichées en couleurs (variées) lorsqu'elles correspondent à des :

- commandes ;
- environnements ;
- clés ;
- valeurs de clé.

### Symboles

`\<court>`, 20

### Nombres

11pt, 2

### A

acronyme, 20

`\alertbox`, 25

alerte, 25

`allcolorlinks`, 4

`announcement`, 23

`\Assogut`, 10

`\assogut`, 10

auteur, 7

`\author`, 5, 7, 33

`\author*`, 7

### B

`bibkey`, 19

`bookreview`, 19

### C

`\cahier`, 10

`\cahiergut`, 10

`\cahiers`, 10

`\cahiersgut`, 10

`\class`, 8

`\class*`, 8

### Clés

11pt, 2

`allcolorlinks`, 4

`bibkey`, 19

`date`, 4

`detailedtoc`, 4

`draft`, 3

`editorial`, 4

`final`, 3

`for-authors`, 3

`for-readers`, 3

`frontcover`, 19

`informations`, 4

`membership-reminder`, 4

`no solution`, 24

`number`, 4

`pagecolor`, 4

`paper`, 3

`price`, 19

`reference text`, 15

`result width`, 15

`reverse-files-attachement`, 5

`reviewer`, 19

`screen`, 3

`title`, 19

`title addon`, 15, 24

`twoside`, 2

### Commands

`\Assogut`, 10

`\DeclareAcronym`, 20

`\DeclareTCBListing`, 14, 16

`\Ucode`, 9

`\<court>`, 20

`\alertbox`, 25

`\assogut`, 10

`\author`, 5, 7, 33

`\author*`, 7

`\cahier`, 10

`\cahiergut`, 10

`\cahiers`, 10

`\cahiersgut`, 10

`\class`, 8

`\class*`, 8

`\date`, 5

`\documentclass`, 3, 5, 33

`\enquote`, 6

`\etocsetnexttocdepth`, 5

`\file`, 9

`\file*`, 9

`\foreignloc`, 9

`\francophony`, 19

`\gut`, 10

`\gutenberg`, 10

`\input`, 6

`\inputarticle`, 6  
`\inputarticle*`, 6  
`\item`, 18  
`\item*`, 18  
`\knuth`, 10  
`\lampo`, 10  
`\latinloc`, 9  
`\letgut`, 10  
`\letgutacro`, 20  
`\letgutcls`, 10  
`\letgutsetup`, 3  
`\lettre`, 11  
`\lettre*`, 11  
`\lettreletgut`, 11  
`\lettreletgut*`, 11  
`\lettrenumber`, 11  
`\lettrenumber*`, 11  
`\lettres`, 10  
`\lettresletgut`, 10  
`\linux`, 10  
`\localtableofcontents`, 5  
`\lstinline`, 17, 33  
`\macos`, 10  
`\maketitle`, 5, 33  
`\nameref`, 6  
`\newtcblisting`, 14, 16  
`\package`, 8  
`\package*`, 8  
`\paragraph`, 6  
`\person`, 7  
`\person*`, 7  
`\rebusresolution`, 24  
`\section`, 6, 11  
`\separator`, 22  
`\setmainfont`, 26  
`\software`, 8  
`\software*`, 8  
`\solution`, 24  
`\subparagraph`, 6  
`\subsection`, 6  
`\subsubsection`, 6  
`\subtitle`, 6, 9, 11, 33  
`\syntaxhl`, 16  
`\terminal`, 16  
`\tikz`, 20  
`\title`, 5, 6, 9, 11, 33  
`\tl`, 10  
`\tugboat`, 10  
`\vpageref`, 6  
`\windows`, 10  
configuration, 3, 11  
Couleurs  
`letgut_allcolors_links`, 4  
`ctannews`, 18  
**D**  
`date`, 4  
`\date`, 5  
`\DeclareAcronym`, 20  
`\DeclareTCBListing`, 14, 16  
`description`, 18  
`detailedtoc`, 4  
`\documentclass`, 3, 5, 33  
`draft`, 3  
**E**  
`editorial`, 4  
`\enquote`, 6  
Environnements  
`announcement`, 23  
`bookreview`, 19  
`ctannews`, 18  
`description`, 18  
`ltx-code`, 14  
`ltx-code-external-result`, 14  
`ltx-code-result`, 14  
`rebus`, 23  
`tcblisting`, 14, 16  
`\etocsetnexttocdepth`, 5  
**F**  
`\file`, 9  
`\file*`, 9  
`final`, 3  
`for-authors`, 3  
`for-readers`, 3  
`\foreignloc`, 9  
`\francophony`, 19  
`frontcover`, 19  
**G**  
`\gut`, 10  
`\gutenberg`, 10  
**I**  
`informations`, 4  
`\input`, 6  
`\inputarticle`, 6  
`\inputarticle*`, 6  
`\item`, 18  
`\item*`, 18  
**K**  
`\knuth`, 10  
**L**  
`\lampo`, 10

`\latinloc`, 9  
`\letgut`, 10  
`letgut_allcolors_links`, 4  
`\letgutacro`, 20  
`\letgutcls`, 10  
`\letgutsetup`, 3  
`\lettre`, 11  
`\lettre*`, 11  
`\lettregut`, 11  
`\lettregut*`, 11  
`\lettrenumber`, 11  
`\lettrenumber*`, 11  
`\lettres`, 10  
`\lettresgut`, 10  
`\linux`, 10  
`\localtableofcontents`, 5  
`\lstinline`, 17, 33  
`ltx-code`, 14  
`ltx-code-external-result`, 14  
`ltx-code-result`, 14

**M**

`\macos`, 10  
`\maketitle`, 5, 33  
`membership-reminder`, 4

**N**

`\nameref`, 6  
`\newtcblisting`, 14, 16  
`no solution`, 24  
`number`, 4

**P**

`\package`, 8  
`\package*`, 8  
`pagecolor`, 4  
`paper`, 3  
`\paragraph`, 6  
`\person`, 7  
`\person*`, 7  
`personne`, 7  
`price`, 19

**R**

`rebus`, 23  
`\rebussolution`, 24  
`reference text`, 15  
`result width`, 15  
`reverse-files-attachement`, 5  
`reviewer`, 19

**S**

`screen`, 3  
`\section`, 6, 11  
séparateur, 22  
`\separator`, 22  
`\setmainfont`, 26  
`\software`, 8  
`\software*`, 8  
`\solution`, 24  
`\subparagraph`, 6  
`\subsection`, 6  
`\subsubsection`, 6  
`\subtitle`, 6, 9, 11, 33  
`\syntaxhl`, 16

**T**

`tcblisting`, 14, 16  
`\terminal`, 16  
`\tikz`, 20  
`\title`, 33  
`title`, 19  
`\title`, 5, 6, 9, 11, 33  
`title addon`, 15, 24  
`\tl`, 10  
`\tugboat`, 10  
`twoside`, 2

**U**

`\Ucode`, 9

**V**

`\vpageref`, 6

**W**

`\windows`, 10

# GUTenberg

Association GUTenberg  
15 rue des Halles – BP 74  
75001 Paris  
France

secretariat[at]gutenberg[dot]eu[dot]org

**Site Internet :** <https://www.gutenberg-asso.fr/>

**Cahiers :** <http://www.numdam.org/journals/CG/>

**Problèmes techniques :**

la liste `gut` : <https://www.gutenberg-asso.fr/-Listes-de-diffusion->

le site `TEX`nique de questions et réponses : <https://texnique.fr/>

la foire aux questions : <https://faq.gutenberg-asso.fr/>

Cette association est la vôtre : faites-nous part de vos idées, de vos envies, de vos préoccupations à l'adresse [secretariat\[at\]gutenberg\[dot\]eu\[dot\]org](mailto:secretariat@gutenberg.eu.org).

## ADHÉSION À L'ASSOCIATION

- Les adhésions sont à renouveler en début d'année pour l'année civile.
- Les administrations peuvent joindre un bon de commande revêtu de la signature de la personne responsable ; les étudiants doivent joindre un justificatif.

### Tarifs 2023

Les membres de GUTenberg peuvent adhérer à l'association internationale, le `rug`, et recevoir son bulletin *TUGboat* à un tarif préférentiel :

**tarif normal :** 65 € (au lieu de 85 \$)

**tarif étudiant :** 40 € (au lieu de 55 \$)

| Type d'adhésion  | Prix  |
|--|-------|
| Membre individuel                                      | 30 €  |
| Membre individuel + adhésion <code>rug</code>          | 95 €  |
| Membre individuel étudiant/demandeur d'emploi          | 15 €  |
| Membre individuel étudiant + adhésion <code>rug</code> | 55 €  |
| Association d'étudiants                                | 65 €  |
| Organisme à but non lucratif                           | 130 € |
| Organisme à but lucratif                               | 229 € |

## Règlements

Les règlements peuvent s'effectuer par :

- **virement bancaire**<sup>30</sup> (IBAN : FR76 1870 7000 3003 0191 3568 475)

**Veillez à bien indiquer vos nom et prénom dans les références du virement !**

- Paypal<sup>30</sup> : <https://www.gutenberg-asso.fr/?Adherer-en-ligne>
- bulletin et chèque<sup>30</sup> : <https://www.gutenberg-asso.fr/?Adherer-a-l-association>

### La Lettre GUTenberg

Bulletin irrégulomestriel de l'association GUTenberg

**Directeur de la publication :** Patrick Bideault

**Comité de rédaction :** Patrick Bideault, Denis Bitouzé,  
Céline Chevalier & Maxime Chupin

**Adresse de la rédaction :** Association GUTenberg  
15 rue des Halles – BP 74  
75001 Paris

ISSN : 2742-6149 (version numérique)

30. Nous vous remercions de privilégier le virement bancaire.