

La L E T T R E

Documentation de la classe `letgut`, version 0.9.14

Introduction	3
Usage de la classe <code>letgut</code>	3
Compilation	3
Options	3
Options de la classe <code>article</code>	3
Options de la classe <code>letgut</code>	4
Options autres	6
Titre et titres courants	6
Importation d'articles	7
Structuration	7
Personnes et auteurs	8
Aide à la saisie et homogénéisation de la mise en forme	9
Packages et classes, logiciels, fichiers	9
Locutions étrangères, points de code Unicode	10
Termes et expressions (figés)	11
Horaires	14
Touches de clavier	14
Codes informatiques	15
Codes (La)TeX	15
Entrées et sorties dans un terminal	18
Caractères d'échappement et de raccourci pour les extraits de code	19
Nouveautés apparues sur le CTAN	19
Fiches de lecture	20
Acronymes	21
Séparateurs	24
Annonces	24
Exemple d'annonce	25
Rébus	25
Boîtes d'alertes	27
Fichier local de configuration	28
Aspects de la 1^{re} page	28
Les dinosaures, leur écosystème et <code>letgut</code>	28
Packages chargés par <code>letgut</code>	29
Packages utiles aux auteurs de la <i>Lettre</i>	29
Packages utiles au codage de la classe <code>letgut</code>	31
Exemple de déclaration de dialecte du langage TeX	31
Liste des acronymes prédéfinis par <code>letgut</code>	33
Références	34

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>

GU^Tenberg

Table des avertissements	34
Index	35
Adhésion à l'association	39
Tarifs 2026	39
Règlements.....	39

Broquillon

INTRODUCTION

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

Pour ce faire, une classe \LaTeX dédiée, maison, a peu à peu vu le jour¹ 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`².

USAGE DE LA CLASSE `LETGUT`

Compilation

Avertissement 1 – Lua \LaTeX (récent) et UTF-8 nécessaires

Les documents recourant à la classe `letgut` doivent :

- être compilés avec le moteur Lua \LaTeX ^a ;
- avoir (de ce fait) comme codage d'entrée l'UTF-8^b.

^a. Dans une version au minimum 1.13.2, le format `lualatex` devant être dans une version au minimum 2021.6.6.

^b. Y compris les fichiers sources auxiliaires tels que les `.bib`.

La section « [Les dinosaures, leur écosystème et `letgut`](#) », page 28, détaille un moyen de disposer d'une telle version de Lua \LaTeX sans risque de perturber une installation de \LaTeX 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.

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`.

Options de la classe `letgut`

Spécification

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.

Les options de la classe `letgut` peuvent être spécifiées en argument de la commande dédiée `\letgutsetup`.

`\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`^{P.5}) ;
 - éditorial³ (cf. clé `editorial`^{P.5}) ;
 - informations sur GUTenberg⁴ (cf. clé `informations`^{P.5}) ;
 - fichiers attachés au PDF produit (cf. clé `reverse-files-attachement`^{P.6})

<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 chargé⁵. Celui-ci met en lumière, par un changement de couleur, les lignes typographiquement imparfaites d'un fichier PDF produit par Lua[®]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>)

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 \LaTeX utilisée par la classe (plus de détails [ici](#).)

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`^{→ P. 5}.

`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`^{→ P. 4} n’est pas employée, de spécifier (selon le modèle « rgb ») une `<couleur>` de fond du papier autre que celle appliquée par défaut⁶.

`allcolorslinks=<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 défaut⁷.

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

Cette clé booléenne affiche automatiquement un (r)appel à cotisation en bas de 1^{re} 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^{re} 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.

6. C’est-à-dire celle-ci.

7. C’est-à-dire celle-ci.

M.-À.-J. : 2025-10-23

detailedtoc=*<niveau>* (init. *title*, par déf. *all*)

où *<niveau>* peut valoir *title*, *subtitle*, *section*, *subsection*, *subsubsection*, *paragraph* ou *subparagraph*.

Par défaut, une table des matières est automatiquement insérée en début de document, avec comme niveau de profondeur celui des titres des articles (saisi via la `\title`^{→P.7}), et seulement eux. La clé `detailedtoc`^{→P.6} permet de modifier le « niveau de profondeur » de cette table des matières, respectivement jusqu'aux titres, sous-titres, sections, sous-sections, sous-sous-sections, paragraphes, sous-paragraphes. Sont également acceptées les valeurs spéciales :

- *all* (alias de *subparagraph*);
- *none* qui inhibe l'affichage de la table des matières.

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 l'une des valeurs possibles de la clé `detailedtoc`, sauf *title*).

Nouv. : 2023-01-14

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`^{→P.4}) et en sortie écran (cf. `screen`^{→P.4}), 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.

Nouv. : 2024-10-07

watermark letter (init. *g*, par déf. *aucune*)

Nouv. : 2024-10-07

watermark scale (init. *1*, par déf. *aucune*)

La première page de la *Lettre* comporte, en filigrane et en gris clair, (entre autre) un (très grand) « g », composé avec la fonte « normale ».

- La clé `watermark letter` permet de spécifier une lettre autre qu'un « g » et/ou composé avec une fonte autre que « normale ».
- La clé `watermark scale` permet de spécifier échelle autre que 1 pour cette lettre (par exemple si celle-ci s'avère par défaut trop grande).

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*^{→P.5} et *date*^{→P.5}). Il figure alors automatiquement en 1^{re} page sous la forme « Numéro *⟨numéro⟩* – *⟨date⟩* ». Le titre courant est alors « La Lettre GUTenberg, *⟨date⟩* »;

utilisée⁸ : et ce, *avant* `\begin{document}`, le *⟨titre⟩* du document figure alors automatiquement en 1^{re} 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](#) de la présente page et [Personnes et auteurs](#) page suivante).

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.

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`.

```
\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*^{→P.6})

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

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 :

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

- `\title` est celle de plus haut niveau, introduisant l'«*intitulé*» de chaque article (automatiquement composé en grandes capitales et précédé de l'ornement «*¶*»);
- `\subtitle`⁹, 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 et ne peuvent donc être référencés par la commande habituelle `\ref` ou, mieux, la commande `\vref` fournie par le `varioref` chargé par la classe. Aussi pourra-t-on, pour faire référence à l'une de ces rubriques, recourir à la commande `\vrefsection`.

`\vrefsection{<label>}`

Cette commande permet d'aisément faire référence aux titres, sous-titres, sections, sous-sections, etc. (non numérotés) de la *Lettre*.

Exemple 1 – références croisées aux rubriques

1 On lira avec intérêt la section `\vrefsection{sec:acronymes}`.

On lira avec intérêt la section « **Acronymes** » page 21.

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 affichent⁹ 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;

9. Au fer à droite pour `\author`.

- 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!

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 (L^AT_EX, 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

```
\package[URL]{nom}
\package*[URL]{nom}[préfixe]
\class[URL]{nom}
\class*[URL]{nom}[préfixe]
```

Ces commandes affichent le *nom* d’un package ou d’une classe (L^AT_EX). Le *nom* 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 *URL*.

Pour que le *nom* affiché ne soit pas un lien hypertexte, il suffit d’employer un argument optionnel vide.

Les versions étoilées font précéder le *nom* d’un *préfixe* qui, par défaut, est respectivement « package » et « classe ».

Exemple 4 – packages et classes

code

```

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*[]{} est
5 \class[https://framagit.org/gutenberg/letgut/]{letgut}.
```

résultat

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 *⟨nom⟩* d'un logiciel qui est optionnellement un lien hypertexte vers *⟨URL⟩*. La version étoilée fait précéder le *⟨nom⟩* d'un *⟨préfixe⟩* qui, par défaut, est « logiciel ».

```

\file{⟨nom⟩}
\file*{⟨nom⟩}[⟨préfixe⟩]
```

Ces commandes affichent le *⟨nom⟩* d'un fichier. La version étoilée fait précéder le *⟨nom⟩* d'un *⟨préfixe⟩* qui, par défaut, est « fichier ».

Exemple 5 – logiciels et fichiers

code

```

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].
```

résultat

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`^{P.7} et `\subtitle`^{P.7}, les versions étoilées de ces commandes ont des effets indésirables (préfixes pas en grandes capitales et signets non conformes).

Locutions étrangères, points de code Unicode

```
\foreignloc{⟨locution⟩}
```

Cette commande est conçue pour afficher une *⟨locution⟩* étrangère.

```
\latinloc{⟨locution⟩}
```

Cette commande est conçue pour afficher une *⟨locution⟩* latine.

Exemple 6 – locutions étrangères

```

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

```

code

résultat

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

`\Ucode[⟨nom⟩]{⟨point de code⟩}`

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

Exemple 7 – Point de code d'un caractère Unicode

```

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

```

code

résultat

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

Termes et expressions (figés)

Nouv. : 2023-01-14

Nouv. : 2023-01-14

Nouv. : 2025-10-26

Nouv. : 2023-01-14

Nouv. : 2023-01-14

`\gutenberg`

`\gut`

`\assogut`

`\Assogut`

`\listegut`

`\lettres`

`\lettresgut`

`\cahier`

`\cahiers`

`\cahiergut`

`\cahiersgut`

`\letgut`

`\letgutcls`

`\knuth`

`\lampport`

`\tl`

`\tugboat`

Nouv. : 2025-10-26

`\dtk`

`\linux`

`\macos`

`\windows`

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

Par homogénéité avec les commandes `\class→P.9` et `\class*→P.9`, 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

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>\listegut</code>	liste GUT
<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>	letgut
<code>\letgutcls</code>	classe letgut
<code>\knuth</code>	Donald E. Knuth
<code>\lamport</code>	Leslie Lamport
<code>\tl</code>	T _E X Live
<code>\tugboat</code>	TUGboat
<code>\dtk</code>	la T _E Xnische Komödie
<code>\linux</code>	GNU/Linux
<code>\macos</code>	macOS
<code>\windows</code>	Windows

empêché l'utilisation de `\letgut` en argument de `\section`¹⁰.

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

Lorsqu'elle est employée en argument des commandes `\title`^{→P.7} et `\subtitle`^{→P.7}, la commande `\letgutcls` a un effet indésirable (préfixe pas en grandes capitales).

Nouv. : 2022-10-03

Nouv. : 2022-10-03

`\lettrenumber`[*entier relatif signé*]

`\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*¹¹ spécifié sinon.

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

Exemple 8 – emploi de la commande `\lettrenumber`

code

```

1 Si le numéro de la \lettre{ } en cours est 46,
2 celui de la \lettre :
3 \begin{enumerate}
```

10. Plus de détails [ici](#).

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

code (suite)

```

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}

```

résultat

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.

Nouv.: 2022-10-03

\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 signé¹¹;

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 et \lettregut

code

```

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*[.].

```

résultat (suite)

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.

Horaires

`\schedule{⟨heure⟩}{⟨minute⟩}`

Cette commande compose les horaires selon les règles typographiques propres au français.

Exemple 10 – composition d’horaire

code

1 Il est `\schedule{11}{53}`.

résultat

Il est 11 h 53.

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 11 – 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

```

	code (suite)
10	<code>\LKeyAt \LKeyScreenUp \LKeyScreenDown \LKeyCommand \LKeyOptionKey</code>
11	
12	<code>\LMouseN \LMouseL \LMouseM \LMouseR</code>
13	
14	<code>\LKey{exclam} \LKey{numbersign} \LKey{percent} \LKey{backslash}</code>
15	
16	<code>% Ajoutés par la classe 'letgut'</code>
17	<code>\LKey{à} \LKey{À} \LKey{â} \LKey{Â} \LKey{é} \LKey{É}</code>
18	<code>\LKey{è} \LKey{È} \LKey{ê} \LKey{Ê} \LKey{ë} \LKey{Ë}</code>
19	<code>\LKey{î} \LKey{Î} \LKey{ï} \LKey{Ï}</code>
20	<code>\LKey{ô} \LKey{Ô}</code>
21	<code>\LKey{ù} \LKey{Ù} \LKey{û} \LKey{Û} \LKey{ü} \LKey{Ü}</code>
22	<code>\LKey{ÿ} \LKey{Ÿ} \LKey{ç} \LKey{Ç}</code>
	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^AT_EX qui y est inséré et pour :

`ltx-code`^{→ P.15} seulement ce `<code>`;

`ltx-code-result`^{→ P.15} également le résultat, compilé en même temps que la *Lettre*;

`ltx-code-external-result`^{→ P.15} 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`^{→ P.15} et `ltx-code-external-result`^{→ P.15}), 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`^{→ P.15} et `ltx-code-external-result`^{→ P.15}, 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`^{→ P.17} ;

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`¹². Elles permettent notamment de surcharger les réglages par défaut, par exemple :
 - 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 12 –

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

- trois spécifiques à ces environnements :

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

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 *<n>* », un *<supplément>*.

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

Cette option, utile seulement pour l'environnement `ltx-code-external-result`^{P.15}, 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 <n> page <q>.*, par déf. *aucune*)

Cette option n'a d'effet que :

- avec l'un ou l'autre des environnements `ltx-code-result`^{P.15} et `ltx-code-external-result`^{P.15} ;
- 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 *<n>* page *<q>.* » automatiquement inséré au point d'insertion de l'exemple (cf. avertissement 9 page précédente)¹³.

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`^{P.7}, le langage supposé dans ces exemples de codes est T_EX, 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 T_EX et ses dialectes `primitive`, `common`, `plain`, `LaTeX`, `AllaTeX` et classes.

Il est néanmoins possible de spécifier d'autres langages et dialectes au moyen de la commande `\syntaxhl` 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 T_EX) ;
- une *<liste de dialectes>*, séparés par des virgules ;

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

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 `(L)TeX` et un exemple de déclaration de tel dialecte est fourni section « [Exemple de déclaration de dialecte du langage TeX](#) », page 31.

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`¹⁴.

Ainsi le code suivant :

Exemple 13 –

```
1 \terminal{lualatex}{
2 This is LuaHBTeX, Version 1.18.0 (TeX Live 2024)
3 restricted system commands enabled.
4 **
5 }
```

donne-t-il :

```
$ lualatex
```

```
This is LuaHBTeX, Version 1.18.0 (TeX Live 2024)
restricted system commands enabled.
**
```

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

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 14 –

```
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 [™] (U+2122)^a.

^a. Peu susceptible d'être utilisé dans du texte ordinaire.

Autrement dit, la classe `letgut` contient l'instruction ¹⁵ :

Exemple 15 –

```
1 \lstMakeShortInline™
```

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

Exemple 16 –

```
1 \lstDeleteShortInline™
```

Remarque 4 – Obtention du caractère [™]

Le caractère [™] s'obtient :

- sous GNU/Linux : `AltGr` + `8`^a ;
- sous macOS : `Alt` + `8` + `T` ;
- sous Windows : `Alt` + `0` + `1` + `5` + `3`.

^a. Touche `8` du clavier principal.

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}
```

¹⁵. À peu de choses près.

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 17 – 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] est une classe \hologo{LaTeX} destinée à la  
5   composition de la revue Matapli (conçue par Maxime Chupin,  
6   administrateur de l'\assogut{}).  
7 \end{ctannews}
```


code

LA LETTRE GUTENBERG, FÉVRIER 2026


résultat

1

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

 **matapli** est une classe L^AT_EX destinée à la composition de la revue Matapli (conçue par Maxime Chupin, administrateur de l'association 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}{⟨caractéristiques⟩  
  ⟨fiche de lecture⟩  
\end{bookreview}
```

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

obligatoires :

title=*titre* (init. vide, par déf. aucune)

Cette clé permet de spécifier le *titre* introductif de la fiche.

reviewer=*rapporteur* (init. vide, par déf. aucune)

Cette clé permet de spécifier le *rapporteur* de la fiche, à spécifier selon la syntaxe de la commande `\author`^{→ p.8}.

bibkey=*<clé>* (init. *vide*, par déf. *aucune*)

Cette clé permet de spécifier la *<clé>* 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 *<bibliographie>*.bib, structurées selon le format du package *biblatex* et chargées en préambule (par exemple dans le fichier local de configuration, cf. page 28) au moyen de la commande :

Exemple 18 –

```
1 \addbibresource{<bibliographie>.bib}
```

fortement conseillée :

frontcover=*<fichier>* (init. *vide*, par déf. *aucune*)

Cette clé permet de spécifier le *<fichier>* image de la couverture du document rapporté.

facultative :

price=*<prix>* (init. *vide*, par déf. *aucune*)

Cette clé permet le cas échéant de spécifier le *<prix>* 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^{er} et 2^e 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>}`¹⁶;
- `\acs{<court>}`¹⁷.

Toutefois, pour simplifier l’usage de ces acronymes, la classe `letgut` crée alors automatiquement une commande `\<court>`¹⁸ 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>`¹⁹ 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.7}, `\subtitle`^{→ P.7}, `\section`^{→ P.7}, `\subsection`^{→ P.7}, etc. et l’acronyme figure sous sa forme `<COURT>` dans les *bookmarks* (signets);
- un copié de `<COURT>` colle `<COURT>`.

Avertissement 16 – `\letgutacro`^{→ P.21} : uniquement en préambule

La définition d’acronymes au moyen de `\letgutacro`^{→ P.21} ne peut se faire qu’en préambule.

Ainsi, avec les définitions suivantes en préambule :

Exemple 19 – 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}
```

16. Acronyme automatiquement affiché sous sa forme complète à sa 1^{re} 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 33) ne surcharge-t-il pas la commande `\tikz` fournie notamment par le package `tikz`.

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

```

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 20 – 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 « \LaTeX 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 21 – définition d’acronyme avec lettres accentuées

```

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

```

a-t-on :

Exemple 22 – utilisation d’acronyme avec lettres accentuées

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

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 33.

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`.

Nouv. : 2023-01-14
M.-À-J. : 2023-05-21

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

Cet environnement est dédié aux `<annonce>`s. La spécification d’un `<titre>` (pouvant toutefois être vide) est obligatoire. S’il n’est pas vide, il figure alors par défaut dans la table des matières et dans les signets au même niveau que celui des titres d’articles.

Cet environnement admet des `<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 l’annonce ;
- deux qui lui sont propres :

Nouv. : 2023-05-21
M.-À-J. : 2023-11-08

`toc title=<titre alternatif>` (init. *vide*, par déf. *aucune*)

Cette clé permet de remplacer dans la table des matières et dans les signets le `<titre>` par un `<titre alternatif>`. Si cette option est utilisée avec un `<titre alternatif>` vide, l’annonce ne figure ni dans la table des matières, ni dans les signets.

Nouv. : 2023-05-21

`color=<couleur>` (init. *black*, par déf. *aucune*)

Cette clé permet une `<couleur>` d’ornement et de titre autre que celle appliquée par défaut.

Exemple 23 – annonces

code

```

1 \begin{announcement}[toc title={Exemple d'annonce},color=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`.

Nouv. : 2023-01-14

```

\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` ;

suivante : l'option `no solution`^{P.26} doit alors être employée.

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

- « page `<n>` » ou « ci-dessous » ;
- « dans la prochaine *Lettre* ».

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

Nouv. : 2023-01-14

```
\solution
```

Cette commande débute la solution d'un rébus à l'intérieur d'un environnement `rebus`.

En outre, cet environnement admet des `<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 le rébus ;

— deux qui lui sont propres :

Nouv. : 2023-01-14

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

Cette option permet d'adjoindre au titre d'un rébus, qui est par défaut et automatiquement « Rébus », un *<supplément>* alors en italique et entre parenthèses.

Nouv. : 2023-01-14

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 24 – 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 suivants²⁰.

➤ 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.

Nouv. : 2023-01-14

\rebussolution[*<solution>*][*<numéro>*][*<options>*]

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

en cours : (commande `\solution`^{P.25}) si ses 1^{er} et 2^e arguments optionnels *<solution>* et *<numéro>* sont inutilisés ;

précédente : si son 1^{er} argument optionnel *<solution>* est utilisé. Selon que 2^e 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

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

échéant, modifier la mise en forme par défaut de la boîte contenant la solution du rébus.

Ainsi le code :

Exemple 25 – solution de rébus

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

donne-t-il donne-t-il les résultats suivants²¹.

🔑 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 défaut²² ;
- contenant le ⟨texte⟩ (qui peut contenir plusieurs paragraphes).

Exemple 26 – boîtes d’alertes

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

code

résultat

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.

résultat (suite)

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^{re} 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^LTeX pour la présente classe `letgut` (cf. avertissement 1 page 3);
- éviter de perturber une installation existante de \LaTeX ;

on pourra recourir à `Docker`²³ dont cette section est un mode d'emploi :

- express;
- axé sur Linux, mais qui devant s'appliquer au moins en partie aux autres systèmes d'exploitation;
- axé sur la \TeX 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.

23. 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.

On commence par installer docker puis à lancer le service docker²⁴.

Ensuite, par exemple depuis un dossier où se trouve un fichier `mon-fichier.tex` (disons à compiler avec Lua \TeX), on lance la longue commande suivante (*qui doit être sur une seule ligne*²⁵) :

```
$ 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 \TeX 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 seulement²⁶ :

```
$ 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 (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`.

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.

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=match`, `group-minimum-digits=4`.

hologo collection de logos habituels (\LaTeX , $\text{\LaTeX 2}\epsilon$, 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`²⁷.

floatrow nombreuses possibilités de personnalisation de la disposition des flottants.

Options :

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

biblatex bibliographies sophistiquées.

Option : `sorting=none`.

acro création simple d’acronymes²⁹.

Options :

- `first-style=footnote` ;

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

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.

- `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`^{→ p. 4} 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 typographique-ment imparfaites avec Lua \LaTeX .

Option : `All`.

Packages utiles au codage de la classe `letgut`

`l3keys2e` traitement \LaTeX 2_ε 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`^{→ p. 19}.

`titlesec` personnalisation aisée des titres de sections, etc.

`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^{re} page de la *Lettre*.

Exemple de déclaration de dialecte du langage \LaTeX

Nous fournissons ci-dessous un exemple de déclaration de dialecte (ici le package \LaTeX `graphicx`) du langage \LaTeX (cf. section « *Coloration syntaxique* », page 17).

Exemple 27 –

```

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,bblx,bblly,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	<i>American Psychological Association</i> (association américaine de psychologie)
APMEP	Association des Professeurs de Mathématiques de l'Enseignement Public
ASCII	<i>American Standard Code for Information Interchange</i> (code américain normalisé pour l'échange d'information)
BBB	BigBlueButton
BSD	<i>Berkeley Software Distribution</i> (collection de logiciels de Berkeley)
BTS	Brevet de Technicien Supérieur
CA	Conseil d'Administration
CPGÉ	Classes Préparatoires aux Grandes Écoles
CSS	<i>Cascading Style Sheets</i> (feuilles de style en cascade)
CSV	<i>Comma-separated values</i> (valeurs séparées par des virgules)
CTAN	<i>Comprehensive T_EX Archive Network</i> (réseau complet d'archives T _E X)
CV	Curriculum Vitæ
DNS	<i>Domain Name System</i> (système de noms de domaine)
DOI	<i>Digital Object Identifier</i> (identifiant numérique d'objet)
DVI	<i>DeVice-Independent</i> (indépendant du type de périphérique)
DVIPS	<i>DVI-to-PS (translator)</i> ((convertisseur) DVI vers PS)
ECM	Exemple Complet Minimal
FAQ	<i>Frequently Asked Questions</i> (questions fréquemment posées)
GPL	<i>GNU General Public License</i> (licence publique générale GNU)
GRAPPA	Groupe de recherche en Apprentissage Automatique
HTML	<i>HyperText Markup Language</i> (langage de balises pour l'hypertexte)
IMAP	<i>Internet Message Access Protocol</i>
INSPÉ	Instituts Nationaux Supérieurs du Professorat et de l'Éducation
IPA	<i>International Phonetic Alphabet</i> (alphabet phonétique international)
IREM	Instituts de Recherche sur l'Enseignement des Mathématiques
ISO	<i>International Organization for Standardization</i> (organisation internationale de normalisation)
ISSN	<i>International Standard Serial Number</i> (numéro international normalisé des publications en série)
LDAP	<i>Lightweight Directory Access Protocol</i> (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	<i>New Font Selection Scheme</i> (nouveau schéma de sélection de fonte)
OFF	<i>Open Font Format</i> (format de fonte ouvert)
OFL	<i>SIL Open Font License</i>
ORCID	<i>Open Research and Contributor ID</i> (identifiant ouvert pour chercheur et contributeur)
OS	<i>Operating System</i> (système d'exploitation)
OTF	<i>Open Type Format</i>
PAO	Publication Assistée par Ordinateur
PDF	<i>Portable Document Format</i> (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_EX Directory Structure* (structure de répertoire de T_EX)

TIKZ *TikZ ist kein Zeichenprogramm* (TikZ n'est pas un programme de dessin)

TIPA *T_EX IPA or Tokyo IPA* (alphabet phonétique international)

TUG *T_EX User Group* (groupe (international) d'utilisateurs de T_EX)

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_EX, L^AT_EX et logiciels compagnons. URL : <https://www.gutenberg.eu.org/-Lettre-GUTenberg->.

🔗 TABLE DES AVERTISSEMENTS

1	LuaL ^A T _E X (récent) et UTF-8 nécessaires	3
2	Options de letgut : pas en argument optionnel de <code>\documentclass</code>	4
3	<code>\title</code> et <code>\author</code> ≠ avant et après <code>\begin{document}</code>	7
4	Commande <code>\maketitle</code> à ne pas employer	7
5	Importation de fichiers d'articles	7
6	Commandes pas toutes bienvenues en <code>\title</code> et <code>\subtitle</code>	10
7	Commande <code>\letgutcls</code> ^{→ P. 11} pas bienvenue en <code>\title</code> et <code>\subtitle</code>	12
8	Touche de clavier du symbole € manquant	15
9	Exemples de codes avec résultats : possiblement flottants	16
10	<code>reference text</code> ^{→ P. 17} avant <code>sidebyside</code>	17
11	Coloration syntaxique réduite par défaut	17
12	Dialectes colorés syntaxiquement seulement si définis	18
13	Caractère d'échappement des listings	19
14	Équivalent court de <code>\lstinline</code>	19
15	Fichier de bibliographie	21

16	<code>\letgutacro^{→P.21}</code> : uniquement en préambule	22
17	Commande <code>sudo</code> peut-être nécessaire	28

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>`, 22

Nombres

11pt, 3

A

acronyme, 21

`\alertbox`, 27

alerte, 27

all, 6

allcolorslinks, 5

announcement, 24

`\Assogut`, 11

`\assogut`, 11

auteur, 8

`\author`, 7, 8, 34

`\author*`, 8

B

bibkey, 21

bookreview, 20

C

`\cahier`, 11

`\cahiergut`, 11

`\cahiers`, 11

`\cahiersgut`, 11

`\class`, 9

`\class*`, 9

Clés

11pt, 3

allcolorslinks, 5

bibkey, 21

color, 24

date, 5

detailedtoc, 6

draft, 4

editorial, 5

final, 4

for-authors, 4

for-readers, 4

frontcover, 21

informations, 5

membership-reminder, 5

no solution, 26

number, 5

pagecolor, 5

paper, 4

price, 21

reference text, 17

result width, 17

reverse-files-attachement, 6

reviewer, 20

screen, 4

title, 20

title addon, 17, 26

toc title, 24

twoside, 3

watermark letter, 6

watermark scale, 6

color, 24

Commands

`\<court>`, 22

`\alertbox`, 27

`\assogut`, 11

`\Assogut`, 11

`\author`, 7, 8, 34

`\author*`, 8

`\cahier`, 11

`\cahiergut`, 11

`\cahiers`, 11

`\cahiersgut`, 11

`\class`, 9

`\class*`, 9

`\date`, 7

`\DeclareAcronym`, 21

[\DeclareTCBListing](#), 16, 18
[\documentclass](#), 4, 6, 34
[\dtk](#), 11
[\etocsetnexttocdepth](#), 6
[\file](#), 10
[\file*](#), 10
[\foreignloc](#), 10
[\francophony](#), 20
[\gut](#), 11
[\gutenberg](#), 11
[\input](#), 7
[\inputarticle](#), 7
[\inputarticle*](#), 7
[\item](#), 20
[\item*](#), 20
[\knuth](#), 11
[\lamport](#), 11
[\latinloc](#), 10
[\letgut](#), 11
[\letgutacro](#), 21
[\letgutcls](#), 11
[\letgutsetup](#), 4
[\lettre](#), 13
[\lettre*](#), 13
[\lettregut](#), 13
[\lettregut*](#), 13
[\lettrenumber](#), 12
[\lettrenumber*](#), 12
[\lettres](#), 11
[\lettresgut](#), 11
[\linux](#), 11
[\listegut](#), 11
[\localtableofcontents](#), 6
[\lstineline](#), 19, 34
[\macos](#), 11
[\maketitle](#), 7, 34
[\newtcblisting](#), 16, 18
[\package](#), 9
[\package*](#), 9
[\paragraph](#), 7
[\person](#), 8
[\person*](#), 8
[\rebussolution](#), 26
[\ref](#), 8
[\schedule](#), 14
[\section](#), 7, 12
[\separator](#), 24
[\setmainfont](#), 28
[\software](#), 10
[\software*](#), 10
[\solution](#), 25
[\subparagraph](#), 7
[\subsection](#), 7
[\subsubsection](#), 7
[\subtitle](#), 7, 10, 12, 34
[\syntaxhl](#), 17
[\terminal](#), 18
[\tikz](#), 22
[\title](#), 6, 7, 10, 12, 34
[\tl](#), 11
[\tugboat](#), 11
[\Ucode](#), 11
[\vref](#), 8
[\vrefsection](#), 8
[\windows](#), 11
 configuration, 4, 12
 Couleurs
 [letgut_allcolors_links](#), 5
 ctannews, 19

D
[\date](#), 7
[date](#), 5
[\DeclareAcronym](#), 21
[\DeclareTCBListing](#), 16, 18
[description](#), 19
[detailedtoc](#), 6
[\documentclass](#), 4, 6, 34
[draft](#), 4
[\dtk](#), 11

E
[editorial](#), 5
 Environnements
 [announcement](#), 24
 [bookreview](#), 20
 ctannews, 19
 [description](#), 19
 [ltx-code](#), 15
 [ltx-code-external-result](#), 15
 [ltx-code-result](#), 15
 [rebus](#), 25
 [tcblisting](#), 16, 18
[\etocsetnexttocdepth](#), 6

F
[\file](#), 10
[\file*](#), 10
[final](#), 4
[for-authors](#), 4
[for-readers](#), 4
[\foreignloc](#), 10
[\francophony](#), 20
[frontcover](#), 21

G

`\gut`, 11
`\gutenberg`, 11

I

`informations`, 5
`\input`, 7
`\inputarticle`, 7
`\inputarticle*`, 7
`\item`, 20
`\item*`, 20

K

`\knuth`, 11

L

`\lamport`, 11
`\latinloc`, 10
`\letgut`, 11
`letgut_allcolors_links`, 5
`\letgutacro`, 21
`\letgutcls`, 11
`\letgutsetup`, 4
`\lettre`, 13
`\lettre*`, 13
`\lettregut`, 13
`\lettregut*`, 13
`\lettrenumber`, 12
`\lettrenumber*`, 12
`\lettres`, 11
`\lettresgut`, 11
`\linux`, 11
`\listegut`, 11
`\localtableofcontents`, 6
`\lstinline`, 19, 34
`ltx-code`, 15
`ltx-code-external-result`, 15
`ltx-code-result`, 15

M

`\macos`, 11
`\maketitle`, 7, 34
`membership-reminder`, 5

N

`\newtcblisting`, 16, 18
`none`, 6
`no solution`, 26
`number`, 5

P

`\package`, 9
`\package*`, 9
`pagecolor`, 5

`paper`, 4
`\paragraph`, 7
`paragraph`, 6
`\person`, 8
`\person*`, 8
`personne`, 8
`price`, 21

R

`rebus`, 25
`\rebussolution`, 26
`\ref`, 8
`reference text`, 17
`result width`, 17
`reverse-files-attachement`, 6
`reviewer`, 20

S

`\schedule`, 14
`screen`, 4
`section`, 6
`\section`, 7, 12
`séparateur`, 24
`\separator`, 24
`\setmainfont`, 28
`\software`, 10
`\software*`, 10
`\solution`, 25
`subparagraph`, 6
`\subparagraph`, 7
`subparagraph`, 6
`\subsection`, 7
`subsection`, 6
`subsubsection`, 6
`\subsubsection`, 7
`\subtitle`, 7, 10, 12, 34
`subtitle`, 6
`\syntaxhl`, 17

T

`tcblisting`, 16, 18
`\terminal`, 18
`\tikz`, 22
`\title`, 6, 10, 34
`title`, 6
`\title`, 7
`title`, 6
`\title`, 7, 12
`title`, 20
`title addon`, 17, 26
`\tl`, 11
`toc title`, 24
`\tugboat`, 11

`twoside`, 3

U

`\Ucode`, 11

V

Valeurs

`all`, 6

`none`, 6

`paragraph`, 6

`section`, 6

`subparagraph`, 6

`subsection`, 6

`subsubsection`, 6

`subtitle`, 6

`title`, 6

`\vref`, 8

`\vrefsection`, 8

W

`watermark letter`, 6

`watermark scale`, 6

`\windows`, 11



Association GUTenberg
15 rue des Halles
BP 74
75001 Paris
France
secretariat@gutenberg-asso.fr

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

Cahiers : <https://cahiers.gutenberg-asso.fr/> et <https://www.numdam.org/journals/CG/>

Lettre : <https://lettre.gutenberg-asso.fr/>

Problèmes T_EXniques :

liste d'entraide : <https://gutenberg-asso.fr/~Listes-de-diffusion->

site de questions et réponses : <https://texnique.fr/>

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@gutenberg-asso.fr.
Adhérents, vous pouvez aussi échanger sur la vie de l'association sur la liste de diffusion adherents@gutenberg-asso.fr.

ADHÉSION À L'ASSOCIATION

GUTenberg étant reconnue d'intérêt général, vous recevrez en temps voulu un justificatif vous permettant de bénéficier d'une réduction fiscale de 66 % du montant de votre cotisation ou de votre don.

- 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.

Tarifs³⁰ 2026

Les membres de GUTenberg peuvent adhérer à l'association internationale, le groupe (international) d'utilisateurs de T_EX (rug), et recevoir son bulletin *TUGboat* à un tarif préférentiel³¹ :

Type d'adhésion	Prix
Personne physique	30 €
Personne physique + adhésion rug	100 € (= 30 € + 70 €)
Personne physique à tarif réduit	10 €
Personne physique à tarif réduit + adhésion rug	55 € (= 10 € + 45 €)
Association d'étudiants	65 €
Organisme ou association à but non lucratif	130 €
Personne morale à but lucratif	195 €

Règlements

Les règlements peuvent s'effectuer par :

- paiement en ligne sécurisé³² : <https://gutenberg-asso.fr/Adherer-en-ligne>
- bulletin et chèque : <https://gutenberg-asso.fr/Adherer-a-l-association>

La Lettre GUTenberg

Bulletin irrégulomestriel & apériodique de l'association GUTenberg

Directeur de la publication : P. Bideault

Comité de rédaction : P. Bideault, D. Bitouzé, C. Chevalier & B. Dumont

Adresse de la rédaction : Association GUTenberg

15 rue des Halles – BP 74 – 75001 Paris

ISSN : 2742-6149 (version numérique)

30. Dans ce tableau, une personne physique à tarif réduit est étudiant, demandeur d'emploi ou plus largement toute personne non redevable de l'impôt sur le revenu (sur présentation d'un justificatif) ; un organisme peut être doté ou non de la personnalité morale : laboratoire de recherche public, etc.

31. En tarif normal, 70 € (au lieu de 90 \$) ; en tarif réduit, 45 € (au lieu de 65 \$).

32. En carte bancaire.