

# Conférence L<sup>A</sup>T<sub>E</sub>X n° 4<sup>1</sup>

Tableaux, unités et quantités, listings informatiques

Denis BITOUZÉ

[denis.bitouze@univ-littoral.fr](mailto:denis.bitouze@univ-littoral.fr)

<https://mt2e.univ-littoral.fr/Members/denis-bitouze/pub/latex>

Laboratoire de Mathématiques Pures et Appliquées Joseph Liouville

<https://lmpa.univ-littoral.fr/>

&

BUT Métiers de la Transition et de l'Efficacité Énergétiques de Dunkerque

<https://mt2e.univ-littoral.fr/>

Le présent cours L<sup>A</sup>T<sub>E</sub>X est :

- conçu pour des utilisateurs de tous niveaux :
  - débutant
  - intermédiaire
  - avancé
- émaillé de passages <sup>1</sup> de niveau plus avancé, alors signalés par des couleurs de fond spécifiques :

niveau intermédiaire : **jaune pâle**

niveau avancé : **rouge pâle**

---

1. Parfois sur plusieurs transparents consécutifs

Conférence  
L<sup>A</sup>T<sub>E</sub>X n° 4

D. Brouzé

Tableaux

Nombres,  
angles,  
quantités

Listings

- 1 Construction de tableaux
- 2 Écriture des nombres, angles, unités et quantités
- 3 Mise en forme de listings informatiques

## Conférence L<sup>A</sup>T<sub>E</sub>X v<sup>2</sup>.<sub>4</sub>

D. BITOUZÉ

### Tableaux

- Tour d'horizon
- Principe
- Files
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- Tableaux flottants
- Tableaux plus professionnels
- Autres fonctionnalités
- Package très prometteur
- Package **array**

Nombres,  
angles,  
quantités

Listings

- 1 Construction de tableaux
- 2 Écriture des nombres, angles, unités et quantités
- 3 Mise en forme de listings informatiques

Tableaux à composer : package `array` conseillé. Il :

- corrige certaines imperfections
- propose des fonctionnalités intéressantes

## Remarque

Dans la suite, `array` supposé systématiquement chargé

## Code source

```
\usepackage{array}
```

## 1 Construction de tableaux

- Tour d'horizon
- Principe
- Jouer avec les filets
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- Tableaux flottants
- Tableaux plus professionnels
- Autres fonctionnalités
- Package très prometteur
- Package array

~~¡Ay ay ay qué dolor!~~

Conférence  
 $\text{\LaTeX}$  n° 4

D. BRUZÉ

Tableaux

Tour d'horizon

Principe

Files

Fusion de cellules

Entrées sur

plusieurs lignes

Assistants

Tableaux et  
paragraphes

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

## Remarque

Construction de tableaux avec  $\text{\LaTeX}$  :

- riche, puissante
- ~~fun, simple~~ : assistants<sup>1</sup>

## Exemple (construction assistée)

Soit le tableau :


1	2	3	4
centrée	centrée	à gauche	à droite

Construisons-le avec l'assistant de TeXstudio

1. D'éditeurs orientés  $\text{\LaTeX}$  ou en ligne



- Visiter **Assistants** > **Tableau...**
- Spécifier le nombre de colonnes et de lignes
- Sélectionner une colonne/ligne pour la configurer <sup>1</sup>
- Remplir le tableau <sup>2</sup>
- Cliquer sur OK

- 
1. P. ex. alignement horizontal des cellules d'une colonne
  2.  pour circuler entre les cellules, comme dans un formulaire Web



Assistant tableau

	c	c	l	r
1	1	2	3	4
2	centrée	centrée	à gauche	à droite

Nombre de colonnes: 4 - +

Nombre de lignes: 2 - +

**Colonnes**

Colonne: 4 - +

Alignement: Alignée à droite

Bordure gauche: |

Appliquer à toutes les colonnes

Bordure droite (dernière colonne): |

**Lignes**

Ligne: 1 - +

Bordure supérieure

Fusionner les colonnes: 1 - + → 2 - +

Appliquer à toutes les lignes

Bordure inférieure (dernière ligne)

Ajouter une marge verticale à chaque ligne

Ok Annuler

The screenshot shows the TeXstudio interface with a LaTeX document titled 'TableauxIDE.tex'. The editor displays the following code:

```

1 \documentclass[french]{article}
2 \usepackage[T1]{fontenc}
3 \usepackage[utf8]{inputenc}
4 \usepackage{lmodern}
5 \usepackage[4paper]{geometry}
6 \usepackage{array}
7 \usepackage{babel}
8 \begin{document}
9 \begin{tabular}{|c|c|l|l|}
10 \hline
11 1 & 2 & 3 & 4 \\
12 \hline
13 centrée & centrée & à gauche & à droite \\
14 \hline
15 \end{tabular}
16 \end{document}
17

```

The rendered output on the right shows a table with four columns and two rows:

1	2	3	4
centrée	centrée	à gauche	à droite

The status bar at the bottom indicates 'Page 1 sur 1', '139%' zoom, and 'INSÉRER' mode.



The screenshot shows the TeXstudio interface with a LaTeX document titled 'TableauxIDE.tex'. The code in the editor is as follows:

```

1 \documentclass[french]{article}
2 \usepackage[T1]{fontenc}
3 \usepackage[utf8]{inputenc}
4 \usepackage{lmodern}
5 \usepackage[4paper]{geometry}
6 \usepackage{array}
7 \usepackage{babel}
8 \begin{document}
9 \begin{table}[c][c][r]
10 \hline
11 1 & 2 & 3 & 4 \\
12 \centered & \centered & à gauche & à droite \\
13 \end{table}
14 \end{document}

```

The rendered output on the right shows a table with four columns and two rows. The first row contains the numbers 1, 2, 3, and 4. The second row contains the text 'centré', 'centré', 'à gauche', and 'à droite'. The table is centered on the page.

## Remarque

**Assistants** : commodes, mais

- **pas** toujours **adaptés** à certains **tableaux** + **complexes**
- **commandes**  $\text{\LaTeX}$  propres aux **tableaux**, à **connaître** :
  - **syntaxe**
  - **possibilités** qu'elles offrent

Ainsi, le code généré par les assistants peut être :

- **compris**
- **modifié** (au besoin)
- **étendu** (au besoin)

## 1 Construction de tableaux

- Tour d'horizon
- **Principe**
- Jouer avec les filets
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- Tableaux flottants
- Tableaux plus professionnels
- Autres fonctionnalités
- Package très prometteur
- Package **array**

## Exemple

### Code source (& : caractère spécial)

```

1 \begin{tabular}{|c|c|l|r|}
2 \hline
3 1 & 2 & 3 & 4 \\
4 \hline
5 centrée & centrée & à gauche & à droite \\
6 \hline
7 \end{tabular}

```

### Résultat

1	2	3	4
centrée	centrée	à gauche	à droite

### Exemple

#### Code source

```

1 \begin{tabular}{|c|c|l|r|}
2   \hline
3   1 & 2 & 3 & 4 \\
4   \hline
5   centrée & centrée & à gauche & à droite \\
6   \hline
7 \end{tabular}

```

#### Résultat

1	2	3	4
centrée	centrée	à gauche	à droite

## Syntaxe

```
\begin{tabular}{\langle motif du tableau \rangle  
  \langle contenu du tableau \rangle  
\end{tabular}
```



### Syntaxe

```
\begin{tabular}{<motif du tableau>}
  <contenu du tableau>
\end{tabular}
```

où *<motif du tableau>* spécifie :

- le **nombre de colonnes**
- l'**alignement** dans celles-ci

C'est une combinaison de :

**spécificateurs de colonne**<sup>1</sup> : les + courants étant les caractères :

- l : pour une colonne alignée à gauche
- c : pour une colonne centrée
- r : pour une colonne alignée à droite

**séparateurs de colonne**<sup>2</sup> : le + courant étant le caractère |<sup>3 4</sup>

1. Au moins un
2. Optionnels
3. Dit *pipe* (**AltGr**+**]** sur un clavier de PC)
4. Génère un *filet* vertical sur toute la hauteur du tableau

### Syntaxe

```
\begin{tabular}{<motif du tableau>}
  <contenu du tableau>
\end{tabular}
```

où *<contenu du tableau>* est composé ligne à ligne au moyen de :

- texte et/ou code L<sup>A</sup>T<sub>E</sub>X valide
- **&** pour séparer deux cellules consécutives <sup>1</sup>
- **\\** pour passer d'une ligne à la suivante
- **\hline** <sup>2</sup> pour un filet horizontal sur toute la largeur du tableau

---

1. Dans un tableau à *n* colonnes, chaque ligne compte *n* – 1 caractères **&**

2. Optionnel

## Tableaux

Tour d'horizon

Principe

### Filets

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphes

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

## 1 Construction de tableaux

- Tour d'horizon
- Principe
- **Jouer avec les filets**
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- Tableaux flottants
- Tableaux plus professionnels
- Autres fonctionnalités
- Package très prometteur
- Package array

## Exemple

### Code source

```

1 \begin{tabular}{|c|c|cc|}
2   \hline
3   1 & 2 & 3 & 4 \\
4   \hline
5   \hline
6   5 & 6 & 7 & 8 \\
7   9 & 10 & 11 & 12 \\
8   \hline
9 \end{tabular}

```

### Résultat

1	2	3	4
5	6	7	8
9	10	11	12

## Exemple

A	B	C	D
E	F	G	H
I	J	K	L
M	N	O	P

Syntaxe (filet horizontal s'étendant des colonnes  $\langle m \rangle$  à  $\langle n \rangle^1$ )

`\cline{ $\langle m \rangle$ – $\langle n \rangle$ }`

### Exemple

#### Code source

```

1 \begin{tabular}{|c|c|c|c|}
2   \hline                               % total
3   A & B & C & D \\
4   \cline{1-3}                           % partiel
5   E & F & G & H \\
6   \cline{2-3}                           % partiel
7   I & J & K & L \\
8   \cline{2-4}                           % partiel
9   M & N & O & P \\
10  \hline                               % total
11 \end{tabular}

```

#### Résultat

A	B	C	D
E	F	G	H
I	J	K	L
M	N	O	P

## 1 Construction de tableaux

- Tour d'horizon
- Principe
- Jouer avec les filets
- **Fusion de cellules**
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- Tableaux flottants
- Tableaux plus professionnels
- Autres fonctionnalités
- Package très prometteur
- Package array

## Exemple (construction assistée)

Soit le tableau :

1	Coucou	4	
5	6	7	8

Construisons-le avec l'assistant de TeXstudio





## Tableaux

Tour d'horizon

Principe

Filiats

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphe

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

Procéder comme au transparent 8 en :

- plaçant le curseur sur la ligne des cellules à fusionner
- cochant « Fusionner les colonnes »<sup>1</sup>
- ajustant les n<sup>os</sup> de colonnes de la plage de cellules à fusionner

---

1. Dans le volet inférieur droit

## Exemple

### Code source

```

1 \begin{tabular}{|c|c|c|c|}\hline
2   1 & \multicolumn{2}{c}{Coucou} & 4 \\ \hline
3   5 & 6 & 7 & 8 \\ \hline
4 \end{tabular}

```

### Résultat

1	Coucou		4
5	6	7	8

## Syntaxe

```
\multicolumn{<n>}{<motif>}{<texte>}
```

où :

**<n>** : nb de **cellules fusionnées horizontalement**

**<motif>** : **alignement horiz.** dans la cellule résultante :

- **spécificateur de colonne**<sup>1</sup>
- **séparateur(s) de colonne**<sup>2</sup> éventuel(s)

**<texte>** : contenu de la cellule résultante

- 
1. **Unique** : généralement caractère **l**, **c** ou **r**
  2. Généralement caractère |

## Exemple

### Code source

```
\usepackage{multirow}

1 \begin{tabular}{|c|c|c|c|}\hline
2   \multirow{2}{*}{Coucou} & 2 & 3 & 4 \\ \cline{2-4}
3                           & 6 & 7 & 8 \\ \hline
4 \end{tabular}
```

### Résultat

Coucou	2	3	4
	6	7	8

## Remarque

Fusion verticale de cellules :

- ~~assurée~~ par l'assistant de tableaux de TeXstudio
- assurée par d'autres assistants (en ligne)

## Syntaxe

```
\usepackage{multirow}
```

```
\multirow{<n>}{<dimension>}{<texte>}
```

où :

*<n>* : nb de **cellules fusionnées verticalement**

*<dimension>* : largeur de la cellule résultante :

- soit \* pour un ajustement automatique<sup>1</sup>
- soit une largeur explicite<sup>2</sup>

*<texte>* : contenu de la cellule résultante

- 
1. Conseillé généralement
  2. P. ex. `0.2\linewidth` ou `3.2cm`

## Exemple

### Code source

```
\usepackage{multirow}

1 \begin{tabular}{|c|c|c|}\hline
2   \multirow{2}{*}{Pays} & \multicolumn{2}{c}{PIB} \\
3   & 1988 & 1998   \\
4   Allemagne & 100 & 126   \\
5   France & 100 & 121   \\
6 \end{tabular}
```

### Résultat

Pays	PIB	
	1988	1998
Allemagne	100	126
France	100	121

## Tableaux

Tour d'horizon

Principe

Filets

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphe

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

## 1 Construction de tableaux

- Tour d'horizon
- Principe
- Jouer avec les filets
- Fusion de cellules
- **Entrées sur plusieurs lignes**
- Assistants
- Tableaux et paragraphes
- Tableaux flottants
- Tableaux plus professionnels
- Autres fonctionnalités
- Package très prometteur
- Package **array**



# Entrées sur plusieurs lignes (aïe!)

Motivation : exemple (semainier de stage étudiant)

Conférence  
L<sup>A</sup>T<sub>E</sub>X n° 4

D. BRUZÉ

## Tableaux

Tour d'horizon

Principe

Files

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphes

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

## Exemple

### Code source

```
1 \begin{tabular}{|l|l|l|} \hline
2 \No & Semaine & Activités \\ \hline
3 1 & 01/04--05/04 & Accueil dans l'entreprise, suivi d'un chantier \\ \hline
4 2 & 08/04--12/04 & Création d'un outil de calcul, établissement de devis \\ \hline
5 3 & 15/04--19/04 & Rédaction du rapport de stage, établissement de devis \\ \hline
6 \end{tabular}
```

### Résultat

N°	Semaine	Activités
1	01/04–05/04	Accueil dans l'entreprise, suivi d'un chantier
2	08/04–12/04	Création d'un outil de calcul, établissement de devis
3	15/04–19/04	Rédaction du rapport de stage, établissement de devis

### Attention!

Les cellules des colonnes de types

• l

• c

• r

sont composées sur **1 seule ligne!**

### Remarque

Mais on peut (bien sûr!) s'en sortir

# Entrées sur plusieurs lignes (ouf!)

Colonne de type p (alignement en haut)

Conférence

LaTeX n° 4

D. BIROUZÉ

## Tableaux

Tour d'horizon

Principe

Files

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphes

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

## Exemple

### Code source

```

1 \begin{tabular}{|l|l|p{5cm}|} \hline
2 \No & Semaine & Activités \\\hline
3 1 & 01/04--05/04 & Accueil dans l'entreprise, suivi d'un chantier \\\hline
4 2 & 08/04--12/04 & Création d'un outil de calcul, établissement de devis \\\hline
5 3 & 15/04--19/04 & Rédaction du rapport de stage, établissement de devis \\\hline
6 \end{tabular}

```

### Résultat

N°	Semaine	Activités
1	01/04–05/04	Accueil dans l'entreprise, suivi d'un chantier
2	08/04–12/04	Création d'un outil de calcul, éta- blissement de devis
3	15/04–19/04	Rédaction du rapport de stage, éta- blissement de devis

# Entrées sur plusieurs lignes (ouf!)

Colonne de type m (alignement au milieu)

Conférence  
L<sup>A</sup>T<sub>E</sub>X n° 4

D. BIROUZÉ

## Tableaux

Tour d'horizon

Principe

Files

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphes

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

## Exemple

### Code source

```
1 \begin{tabular}{|l|l|m{5cm}|} \hline
2 \No & Semaine & Activités \\\hline
3 1 & 01/04--05/04 & Accueil dans l'entreprise, suivi d'un chantier \\\hline
4 2 & 08/04--12/04 & Création d'un outil de calcul, établissement de devis \\\hline
5 3 & 15/04--19/04 & Rédaction du rapport de stage, établissement de devis \\\hline
6 \end{tabular}
```

### Résultat

N°	Semaine	Activités
1	01/04–05/04	Accueil dans l'entreprise, suivi d'un chantier
2	08/04–12/04	Création d'un outil de calcul, éta- blissement de devis
3	15/04–19/04	Rédaction du rapport de stage, éta- blissement de devis

# Entrées sur plusieurs lignes (ouf!)

Colonne de type b (alignement en bas)

Conférence  
L<sup>A</sup>T<sub>E</sub>X n° 4

D. BRIZÉ

## Tableaux

Tour d'horizon

Principe

Files

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphes

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

## Exemple

### Code source

```

1 \begin{tabular}{|l|l|b{5cm}|} \hline
2 \No & Semaine & Activités \\\hline
3 1 & 01/04--05/04 & Accueil dans l'entreprise, suivi d'un chantier \\\hline
4 2 & 08/04--12/04 & Création d'un outil de calcul, établissement de devis \\\hline
5 3 & 15/04--19/04 & Rédaction du rapport de stage, établissement de devis \\\hline
6 \end{tabular}

```

### Résultat

N°	Semaine	Activités
1	01/04–05/04	Accueil dans l'entreprise, suivi d'un chantier
2	08/04–12/04	Création d'un outil de calcul, éta- blissement de devis
3	15/04–19/04	Rédaction du rapport de stage, éta- blissement de devis

Pour des renvois à la ligne automatiques, spécif. de col. :

### Syntaxe

```
p{⟨largeur⟩} % align. vertical : haut du parag.
m{⟨largeur⟩} % align. vertical : milieu du parag.
b{⟨largeur⟩} % align. vertical : bas du parag.
```

où la *⟨largeur⟩* de la colonne peut être exprimée en cm

### Attention!

Spécificateurs `m` et `b`  $\implies$  package `array`

## Tableaux

Tour d'horizon

Principe

Filets

Fusion de cellules

Entrées sur  
plusieurs lignes

### Assistants

Tableaux et  
paragraphe

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

## 1 Construction de tableaux

- Tour d'horizon
- Principe
- Jouer avec les filets
- Fusion de cellules
- Entrées sur plusieurs lignes
- **Assistants**
- Tableaux et paragraphes
- Tableaux flottants
- Tableaux plus professionnels
- Autres fonctionnalités
- Package très prometteur
- Package array



### Tableaux

- Tour d'horizon
- Principe
- Filiés
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
  - Tableaux et paragraphes
  - Tableaux flottants
  - Tableaux plus professionnels
- Autres fonctionnalités
- Package très prometteur
- Package **array**

Nombres,  
angles,  
quantités

Listings

Création :   (déjà vu)

Gestion :   puis


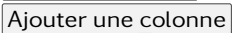
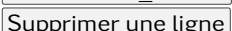
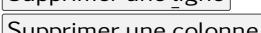
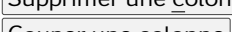
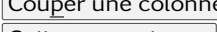

- 
- 
- 
- 
- 
- 
- 
- etc.



Tableau créé avec un tableur<sup>1</sup> : exportable en code L<sup>A</sup>T<sub>E</sub>X :

(Open|Libre)Office : macro Calc2LaTeX<sup>2</sup>

Excel : macro *excel2latex*

- 1 Installation *facile*!
- 2 À tester soi-même<sup>3</sup>

- 
1. Formules comprises
  2. Actuellement **bogué** : contributions bienvenues!
  3. Je ne dispose pas d'Excel

Il existe plusieurs assistants en ligne, notamment :

- *Tables Generator*
- *Table Editor*
- $\text{\LaTeX}$  *Complex Table Editor*

## Exemples de fonctionnalités :

- `import` de tableaux <sup>1</sup>
- `fusion` horizontale/verticale de cellules
- `lignes/colonnes` aisément :
  - `insérées`
  - `supprimées`
  - `déplacées` et `échangées`
- `lignes partielles` aisément `définies`
- code pour `tableaux flottants` (cf. plus loin)

---

1. Par copié-collé, p. ex. depuis un tableur

## 1 Construction de tableaux

- Tour d'horizon
- Principe
- Jouer avec les filets
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- **Tableaux et paragraphes**
- Tableaux flottants
- Tableaux plus professionnels
- Autres fonctionnalités
- Package très prometteur
- Package array

## Remarque

Les tableaux sont intégrés aux paragraphes

## Exemple

### Code source

```

1 Bonjour les amis,
2 \begin{tabular}{c}      % pas d'option
3 je suis un\\
4 tableau sur 2 lignes.
5 \end{tabular}
6 Étonnant, non?
```

### Résultat

Bonjour les amis, 



 Étonnant, non?

## Remarque

Les tableaux sont intégrés aux paragraphes

## Exemple

### Code source

```

1 Bonjour les amis,
2 \begin{tabular}[t]{c} %      option 't' (top)
3 je suis un\\
4 tableau sur 2 lignes.
5 \end{tabular}
6 Étonnant, non?
```

### Résultat

Bonjour les amis,            je suis un            Étonnant, non ?  
tableau sur 2 lignes.

## Remarque

Les tableaux sont intégrés aux paragraphes

## Exemple

### Code source

```
1 Bonjour les amis,
2 \begin{tabular}[b]{c} %      option 'b' (bottom)
3 je suis un\\
4 tableau sur 2 lignes.
5 \end{tabular}
6 Étonnant, non?
```

### Résultat

je suis un

Bonjour les amis,    tableau sur 2 lignes.    Étonnant, non ?

### Tableaux

Tour d'horizon

Principe

Files

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

**Tableaux et  
paragraphes**

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

Attention!

*It's not a bug, it's a feature!*



### Remarque

#### Tableaux :

- néanmoins **isolables dans** des **paragraphes** dédiés
- mais alors **pas bien détachés** des autres paragraphes

### Exemple (tableaux isolés, mais pas bien détachés )

#### Code source

```

1 Coucou, comment va ?
2
3 \begin{tabular}{|r|}
4 \hline Ça va, merci. \\ \hline
5 Et vous ? \\ \hline
6 \end{tabular}
7
8 Grâce à \LaTeX{}, ça va !

```

#### Résultat

Coucou, comment va ?

Ça va, merci.
---------------

Et vous ?
-----------

Grâce à L<sup>A</sup>T<sub>E</sub>X, ça va !

## Remarque

Tableaux :

- néanmoins **isolables dans** des **paragraphes** dédiés
- mais alors **pas** bien **détachés** des autres paragraphes...  
**mais aérables**

Exemple (tableaux isolés, mais pas bien détachés... mais aérables)

Code source (**sous-optimal!**)

```

1 Coucou, comment va ?
2 \begin{center} % Seules diff.
3   \begin{tabular}{|r|}
4     \hline Ça va, merci. \\ \hline
5     Et vous ?           \\ \hline
6   \end{tabular}
7 \end{center} % Seules diff.
8 Grâce à \LaTeX{}, ça va !

```

Résultat (**sous-optimal!**)

Coucou, comment va ?

Ça va, merci.
Et vous ?

Grâce à L<sup>A</sup>T<sub>E</sub>X, ça va !

## Attention (mieux!!)

Pour détacher un tableau d'un §, le faire flotter. Ainsi :

positionné de façon optimale

explicité par une légende

référéncé dans le texte <sup>1</sup>

listé dans l'éventuelle liste des tableaux <sup>2</sup>

## Remarque

Ce qui suit est analogue au cours sur les images flottantes

1. Normalement, obligatoire!
2. Commande `\listoftables`, analogue de `\listoffigures`

## Tableaux

Tour d'horizon

Principe

Filets

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragrapes

### Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

## 1 Construction de tableaux

- Tour d'horizon
- Principe
- Jouer avec les filets
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- **Tableaux flottants**
- Tableaux plus professionnels
- Autres fonctionnalités
- Package très prometteur
- Package array

## Syntaxe (tableaux flottants)

```
\begin{table}[\langle préférences de placement \rangle]
  \centering
  \langle un tableau \rangle % \begin{tabular}{...}...
  \caption{\langle légende \rangle}
  \label{\langle identifiant \rangle}
\end{table}
```

où :

- *\langle préférences de placement \rangle* : suite de lettres<sup>1</sup> parmi **h** (*here*), **t** (*top*), **b** (*bottom*), **p** (*page*<sup>2</sup>)
- **\centering** : centrage horizontal du **tableau**
- *\langle légende \rangle* : ajoutée sous le **tableau**
- *\langle identifiant \rangle* : en vue de références croisées

1. Ordre indifférent
2. Page dédiée aux flottants

## Code source (tableaux « flottants »)

```
1 \begin{table}[ht]
2   \centering
3   \begin{tabular}{|c|c|l|r|}
4     \hline
5     1          & 2          & 3          & 4          \\
6     \hline
7     centrée & centrée & à gauche & à droite \\
8     \hline
9   \end{tabular}
10  \caption{Exemple de tableau \LaTeX{}}
11  \label{beau-tableau}
12 \end{table}
13 Le tableau~\ref{beau-tableau} est beau.
```

1	2	3	4
centrée	centrée	à gauche	à droite


TABLE 1 – Exemple de tableau  $\LaTeX$

Le tableau 1 est beau.

Séquence du ou des quelques transparents suivants :

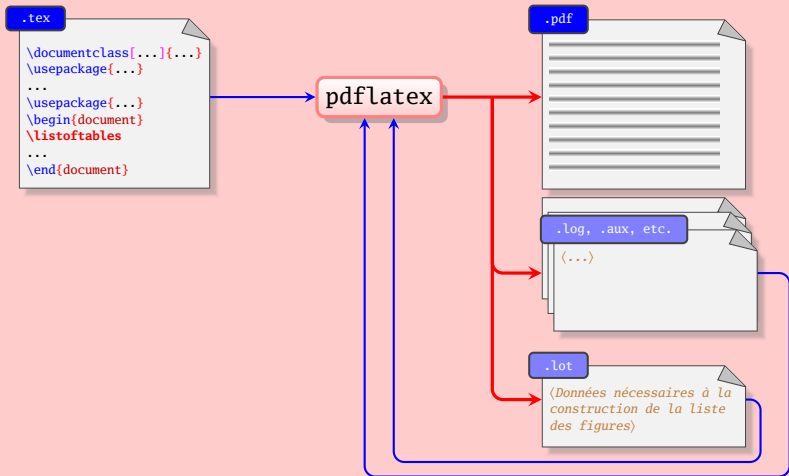
- de niveau avancé, significativement plus élevé
- traite de détails omissibles en 1<sup>re</sup> approche
- peut, sur chacun d'eux, être :

passée au moyen du bouton 

réentamée au moyen du bouton 

## Remarque

Présent transparent : ~~pas répété~~ avant la ou les prochaines séquences de transparents de niveau avancé (signalés par leur fond de couleur rouge pâle)





### Remarque

Un **tableau flottant** avec :

- une **légende**
- un **label** (en vue de références croisées)

**peut contenir** des **sous-tableaux**, chacun avec :

- une **légende propre**
- un **label propre** (en vue de références croisées)

Pour cela, il suffit de recourir :

- à l'environnement **subtable**<sup>1</sup>
- fourni par le package **subcaption**
- à l'**intérieur** de l'environnement **table**

1. Analogue aux sous-figures : + de détails dans le cours sur les images 57

## Tableaux

Tour d'horizon

Principe

Filets

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphe

Tableaux flottants

**Tableaux plus  
professionnels**

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

## 1 Construction de tableaux

- Tour d'horizon
- Principe
- Jouer avec les filets
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- Tableaux flottants
- **Tableaux plus professionnels**
- Autres fonctionnalités
- Package très prometteur
- Package array

## Remarque

Tableaux : **lisibles**  $\implies$  avec **filets verticaux**? **Non!**

Comparer la lisibilité des 2 tableaux suivants :

1

$D$	$P_u$	$u_u$	$\beta$	$G_f$
5 in	269.8 lbs	0.000674 in	1.79	0.04089 psi·jn
10 in	421.0 lbs	0.001035 in	3.59	
20 in	640.2 lbs	0.001565 in	7.18	

2

$D$ (in)	$P_u$ (lbs)	$u_u$ (in)	$\beta$	$G_f$ (psi·jn)
5	269.8	0.000674	1.79	0.04089
10	421.0	0.001035	3.59	0.04089
20	640.2	0.001565	7.18	0.04089

## Remarque

Tableaux plus « professionnels » avec :

- le package **booktabs**<sup>1</sup>
- ses commandes :
  - `\toprule`
  - `\midrule`
  - `\bottomrule`

---

1. Et aussi le package **ctable**, plus complexe mais plus puissant

## Exemple

### Code source

```
\usepackage{booktabs}

1 \begin{tabular}{ccc}
2         & & & \\
3 Libre & & & \\
4 Gratuit & & & \\
5 \end{tabular}
```

### Résultat

	Word	L <sup>A</sup> T <sub>E</sub> X
Libre	Non	Oui
Gratuit	Non	Oui



### Remarque

Export à la **booktabs** possible avec :

- **Tables Generator**<sup>1</sup>
- **L<sup>A</sup>T<sub>E</sub>X Complex Table Editor**<sup>2</sup>

---

1. Cf. sa liste déroulante « Default table style »

2. Cf. son bouton « Auto-Booktabs »

## Tableaux

Tour d'horizon

Principe

Filets

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphe

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings


## 1 Construction de tableaux


- Tour d'horizon
- Principe
- Jouer avec les filets
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- Tableaux flottants
- Tableaux plus professionnels
- **Autres fonctionnalités**
- Package très prometteur
- Package array



Séquence du ou des quelques transparents suivants :

- de niveau intermédiaire, un peu plus élevé
- traite de détails omissibles en 1<sup>re</sup> approche
- peut, sur chacun d'eux, être :

passée au moyen du bouton 

réentamée au moyen du bouton 

## Remarque

Présent transparent : ~~pas répété~~ avant la ou les prochaines séquences de transparents de niveau intermédiaire (signalés par leur fond de couleur **jaune** pâle)

## Tableaux

Tour d'horizon

Principe

Files

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphe

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

On **survole** maintenant quelques :

- **fonctionnalités spécifiques** aux **tableaux**
- **packages** associés

## Tableaux de largeur déterminée : packages <sup>1</sup>

- **tabularx**
- **tabulary**

Ci-après, exemples avec le package **tabularx**

---

1. Notamment

# Entrées sur plusieurs lignes (aïe!)

Motivation : exemple (semainier de stage étudiant, **rappel**)

Conférence  
L<sup>A</sup>T<sub>E</sub>X n° 4

D. Brouzé

## Tableaux

- Tour d'horizon
- Principe
- Files
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- Tableaux flottants

Tableaux plus professionnels

Autres fonctionnalités

Package très prometteur

Package **array**

Nombres, angles, quantités

Listings

## Exemple

### Code source

```
%
1 \begin{tabular}{|l|l|l|} \hline
2 \No & Semaine & Activités \\ \hline
3 1 & 01/04--05/04 & Accueil dans l'entreprise, suivi d'un chantier \\ \hline
4 2 & 08/04--12/04 & Création d'un outil de calcul, établissement de devis \\ \hline
5 3 & 15/04--19/04 & Rédaction du rapport de stage, établissement de devis \\ \hline
6 \end{tabular}
```

### Résultat

N°	Semaine	Activités
1	01/04–05/04	Accueil dans l'entreprise, suivi d'un chantier
2	08/04–12/04	Création d'un outil de calcul, établissement de devis
3	15/04–19/04	Rédaction du rapport de stage, établissement de devis

# Entrées sur plusieurs lignes (ouf!)

Colonne de type p (alignement en haut, **rappel**)

Conférence  
L<sup>A</sup>T<sub>E</sub>X n° 4

D. BIROUZÉ

## Tableaux

- Tour d'horizon
- Principe
- Files
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- Tableaux flottants
- Tableaux plus professionnels

## Autres fonctionnalités

- Package très prometteur
- Package **array**

Nombres, angles, quantités

Listings

## Exemple

### Code source

```
%
1 \begin{tabular}{|l|l|p{5cm}|} \hline
2 \No & Semaine & Activités \\\hline
3 1 & 01/04--05/04 & Accueil dans l'entreprise, suivi d'un chantier \\\hline
4 2 & 08/04--12/04 & Création d'un outil de calcul, établissement de devis \\\hline
5 3 & 15/04--19/04 & Rédaction du rapport de stage, établissement de devis \\\hline
6 \end{tabular}
```

### Résultat

N°	Semaine	Activités
1	01/04–05/04	Accueil dans l'entreprise, suivi d'un chantier
2	08/04–12/04	Création d'un outil de calcul, établissement de devis
3	15/04–19/04	Rédaction du rapport de stage, établissement de devis

# Entrées sur plusieurs lignes (ouf!)

Tableau de largeur déterminée (package `tabularx`)

Conférence  
L<sup>A</sup>T<sub>E</sub>X n° 4

D. Brouzé

## Tableaux

- Tour d'horizon
- Principe
- Files
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- Tableaux flottants
- Tableaux plus professionnels

## Autres fonctionnalités

- Package très prometteur
- Package `array`

Nombres, angles, quantités

Listings

## Exemple

### Code source

```
\usepackage{tabularx}

1 \begin{tabularx}{\linewidth}{|l|l|X|} \hline
2 \No & Semaine & & Activités \\\hline
3 1 & 01/04--05/04 & & Accueil dans l'entreprise, suivi d'un chantier \\\hline
4 2 & 08/04--12/04 & & Création d'un outil de calcul, établissement de devis \\\hline
5 3 & 15/04--19/04 & & Rédaction du rapport de stage, établissement de devis \\\hline
6 \end{tabularx}
```

### Résultat

N°	Semaine	Activités
1	01/04–05/04	Accueil dans l'entreprise, suivi d'un chantier
2	08/04–12/04	Création d'un outil de calcul, établissement de devis
3	15/04–19/04	Rédaction du rapport de stage, établissement de devis



Les tableaux **trop larges** : **pivotables**. Cf. packages :

- **graphicx**<sup>1</sup>
- **rotfloat**
- **floatrow** (plus complexe)
- **ctable** (plus complexe)

Ci-après, exemples avec les packages **graphicx** et **rotfloat**

---

1. Grâce à sa commande `\rotatebox{90}{...}`



## Code source (tableau trop large)

```
%
```

```
1 %  
2 %  
3 \begin{tabular}{|l|l|l|l|l|l|l|l|l|l|}  
4 \hline  
5 bla 1 & bla 2 & bla 3 & bla 4 & bla 5  
6 & bla 6 & bla 7 & bla 8 & bla 9 & bla 10 \\  
7 ble 1 & ble 2 & ble 3 & ble 4 & ble 5  
8 & ble 6 & ble 7 & ble 8 & ble 9 & ble 10 \\  
9 \hline  
10 \end{tabular}  
11 %  
12 %
```

bla 1	bla 2	bla 3	bla 4	bla 5	bla 6	bla 7	bla 8	bla 9
ble 1	ble 2	ble 3	ble 4	ble 5	ble 6	ble 7	ble 8	ble 9

## Code source (tableau trop large (mais pivoté))

```
\usepackage{graphicx}
```

```
1 %  
2 \rotatebox{90}{% <-  
3 \begin{tabular}{|l|l|l|l|l|l|l|l|l|l|}  
4 \hline  
5 bla 1 & bla 2 & bla 3 & bla 4 & bla 5  
6 & bla 6 & bla 7 & bla 8 & bla 9 & bla 10 \\  
7 bla 1 & bla 2 & bla 3 & bla 4 & bla 5  
8 & bla 6 & bla 7 & bla 8 & bla 9 & bla 10 \\  
9 \hline  
10 \end{tabular}  
11 } % <-  
12 %
```

bla 1	bla 2	bla 3	bla 4	bla 5	bla 6	bla 7	bla 8	bla 9	bla 10
ble 1	ble 2	ble 3	ble 4	ble 5	ble 6	ble 7	ble 8	ble 9	ble 10

## Code source (tableau trop large (mais pivoté et flottant))

```
\usepackage{rotfloat}
```

```
1 \begin{sidewaystable} % <-  
2 \centering % <-  
3 \begin{tabular}{|l|l|l|l|l|l|l|l|l|l|}  
4 \hline  
5 bla 1 & bla 2 & bla 3 & bla 4 & bla 5  
6 & bla 6 & bla 7 & bla 8 & bla 9 & bla 10 \\  
7 ble 1 & ble 2 & ble 3 & ble 4 & ble 5  
8 & ble 6 & ble 7 & ble 8 & ble 9 & ble 10 \\  
9 \hline  
10 \end{tabular}  
11 \caption{Coucou} % <-  
12 \end{sidewaystable} % <-
```

bla 1	bla 2	bla 3	bla 4	bla 5	bla 6	bla 7	bla 8	bla 9	bla 10
ble 1	ble 2	ble 3	ble 4	ble 5	ble 6	ble 7	ble 8	ble 9	ble 10

TABLE 1 – Coucou

Tableaux sur plusieurs pages : cf. packages :

- 1 **longtable**
- 2 **xltabular**<sup>1</sup>

Ci-après, exemple avec le package **longtable**

**Attention!**

Imbrication d'environnements **longtable** : impossible

---

1. Fusion des fonctionnalités de **longtable** et **tabularx**



### Code source

```

\usepackage{fontawesome5}
\usepackage[table]{xcolor}

1 \rowcolors{2}{lightgray}{}
2 \begin{tabular}{ccc}
3     & Word & & \LaTeX{} \\ \hline
4 Libre & \faTimes & & \faCheck \\
5 Gratuit & \faTimes & & \faCheck \\
6 Efficace & \faGrinSquint & & \faCheck \\
7 Gentil & \faGrinSquint & & \faCheck \\
8 \end{tabular}

```

### Résultat

	Word	L <sup>A</sup> T <sub>E</sub> X
Libre	✘	✓
Gratuit	✘	✓
Efficace	☹	✓
Gentil	☺	✓

Pour des tableaux avec lignes colorées :

- package **xcolor**, chargé avec l'option **table**
- commande **\rowcolors** (à insérer avant le tableau) :

### Syntaxe

```
\rowcolors{%
  <n° de la 1re ligne devant être colorée>%
} {%
  <couleur lignes impaires>%
} {%
  <couleur lignes paires>%
}
```

## Tableaux

Tour d'horizon

Principe

Filets

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphe

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

**Package très  
prometteur**

Package `array`

Nombres,  
angles,  
quantités

Listings

## 1 Construction de tableaux

- Tour d'horizon
- Principe
- Jouer avec les filets
- Fusion de cellules
- Entrées sur plusieurs lignes
- Assistants
- Tableaux et paragraphes
- Tableaux flottants
- Tableaux plus professionnels
- Autres fonctionnalités
- **Package très prometteur**
- Package `array`



## tabularray :

- package récent, très prometteur
- permet de séparer au maximum :
  - données
  - mise en formedu tableau
- offre beaucoup de fonctionnalités

### Code source

```

1 \begin{tblr}{
2   hlines = {blue1},vlines = {blue1},
3   cell{1,6}{odd} = {teal7},cell{1,6}{even} = {green7},
4   cell{2,4}{1,4} = {red7},
5   cell{3,5}{1,4} = {purple7},
6   cell{2}{2} = {r=4,c=2}{c,azure7,fg=white},
7 }
8 Alpha & Beta & Gamma & Delta & \\\
9 Epsilon & Zeta & Eta & Theta & \\\
10 Iota & Kappa & Lambda & Mu & \\\
11 Nu & Xi & Omicron & Pi & \\\
12 Rho & Sigma & Tau & Upsilon & \\\
13 Phi & Chi & Psi & Omega & \\\
14 \end{tblr}

```

Alpha	Beta	Gamma	Delta
Epsilon	Zeta		Theta
Iota	Zeta		Mu
Nu	Zeta		Pi
Rho	Zeta		Upsilon
Phi	Chi	Psi	Omega

## Tableaux

Tour d'horizon

Principe

Filets

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphe

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package **array**

Nombres,  
angles,  
quantités

Listings

## 1 Construction de tableaux



● **Package array**

Tableaux à composer : package `array` conseillé. Il :

- corrige certaines imperfections
- propose des fonctionnalités intéressantes

## Remarque

Dans la suite, `array` supposé systématiquement chargé

## Code source

```
\usepackage{array}
```

## Attention!

Package `array` : permet d'éviter de pénibles répétitions

Pour une **colonne donnée**, on peut spécifier du **code  $\LaTeX$ <sup>1</sup>** :

- à insérer **automatiquement**
- à **chacun** des **début** et/ou **fin** de **cellule**

---

1. Et/ou du texte

# Package `array` : éviter les saisies répétitives

Mise en forme appliquée à toute une colonne : pénible

Conférence  
L<sup>A</sup>T<sub>E</sub>X n° 4

D. BRUZÉ

## Tableaux

Tour d'horizon

Principe

Filets

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragraphes

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package `array`

Nombres,  
angles,  
quantités

Listings

## Exemple (saisies répétitives : pénible)

### Code source

```

1 \begin{tabular}{l l}
2 \textbf{Français} & \textcolor{red}{English} \\
3 \hline
4 \textbf{un} & \textcolor{red}{one} \\
5 \textbf{deux} & \textcolor{red}{two} \\
6 \textbf{trois} & \textcolor{red}{three} \\
7 \textbf{quatre} & \textcolor{red}{four} \\
8 \textbf{cinq} & \textcolor{red}{five} \\
9 \textbf{six} & \textcolor{red}{six} \\
10 \textbf{sept} & \textcolor{red}{seven} \\
11 \textbf{huit} & \textcolor{red}{eight} \\
12 \textbf{neuf} & \textcolor{red}{nine} \\
13 \textbf{dix} & \textcolor{red}{ten} \\
14 \end{tabular}

```

### Résultat

Français	English
un	one
deux	two
trois	three
quatre	four
cinq	five
six	six
sept	seven
huit	eight
neuf	nine
dix	ten

# Package `array` : éviter les saisies répétitives

Mise en forme appliquée à toute une colonne : ~~pas pénible~~!

Conférence  
L<sup>A</sup>T<sub>E</sub>X n° 4

D. BRUZÉ

## Tableaux

Tour d'horizon

Principe

Filets

Fusion de cellules

Entrées sur  
plusieurs lignes

Assistants

Tableaux et  
paragrophes

Tableaux flottants

Tableaux plus  
professionnels

Autres  
fonctionnalités

Package très  
prometteur

Package `array`

Nombres,  
angles,  
quantités

Listings

## Exemple (saisies ~~pas répétitives~~ : ~~pas pénible~~)

### Code source

```
\usepackage{array}

1 \begin{tabular}{>{\bfseries}l>{\color{red}}l}
2   Français & English \\
3 \hline
4   un      & one    \\
5   deux   & two   \\
6   trois  & three \\
7   quatre & four  \\
8   cinq  & five  \\
9   six   & six   \\
10  sept  & seven \\
11  huit  & eight \\
12  neuf  & nine  \\
13  dix   & ten   \\
14 \end{tabular}
```

### Résultat

Français	English
un	one
deux	two
trois	three
quatre	four
cinq	five
six	six
sept	seven
huit	eight
neuf	nine
dix	ten

Dans le motif du tableau :

Syntaxe (insertion en **début** de cellule)

`>{\langle en début \rangle}\langle spécif. de colonne \rangle`

Syntaxe (insertion en **fin** de cellule)

`\langle spécif. de colonne \rangle <{\langle en fin \rangle}`

Syntaxe (insertion en **début** et en **fin** de cellule)

`>{\langle en début \rangle}\langle spécif. de colonne \rangle <{\langle en fin \rangle}`



## Attention!

En argument de `>{...}` et `<{...}` :

commandes d'action *sur* le texte<sup>1</sup>?

⇒ **uniquement** sous forme de **bascules**

P. ex. :

- `\bfseries` et pas ~~`\textbf`~~
- `\color{...}` et pas ~~`\textcolor{...}`~~

---

1. ≠ commandes de **production de** texte

## Code source

```

\usepackage{array}
%
1 %
2 \begin{tabular}{l l}
3   Test \no 1 & $\sqrt{2}$ \\
4   Test \no 2 & $\sqrt{3}$ \\
5   Test \no 3 & $\sqrt{5}$ \\
6   Test \no 4 & $\sqrt{7}$ \\
7 \end{tabular}

```

## Résultat

Test n° 1	$\sqrt{2}$
Test n° 2	$\sqrt{3}$
Test n° 3	$\sqrt{5}$
Test n° 4	$\sqrt{7}$

## Code source

```

\usepackage{array}
%
1 %
2 \begin{tabular}{>{\Test \no           }l>{\$}l<{\$}}
3     1 & \sqrt{2}  \\
4     2 & \sqrt{3}  \\
5     3 & \sqrt{5}  \\
6     4 & \sqrt{7}  \\
7 \end{tabular}

```

## Résultat

Test n° 1	$\sqrt{2}$
Test n° 2	$\sqrt{3}$
Test n° 3	$\sqrt{5}$
Test n° 4	$\sqrt{7}$

## Code source

```

\usepackage{array}
\usepackage[table]{xcolor}

1 \rowcolors{1}{}{} % Nécessaire, même si lignes non colorées
2 \begin{tabular}{>{Test \no\the\rownum}l>{$}l<{$}}
3         & \sqrt{2} \\
4         & \sqrt{3} \\
5         & \sqrt{5} \\
6         & \sqrt{7} \\
7 \end{tabular}

```

## Résultat

Test n° 1	$\sqrt{2}$
Test n° 2	$\sqrt{3}$
Test n° 3	$\sqrt{5}$
Test n° 4	$\sqrt{7}$

Conférence  
L<sup>A</sup>T<sub>E</sub>X n° 4

D. BRIZÉ

Tableaux

Nombres,  
angles,  
quantités

Nombres

Angles

Quantités

Listes, produits et  
pages

Unités seules

Tableaux de  
nombres

Unités binaires

Raffinements

Listings

- 1 Construction de tableaux
- 2 Écriture des nombres, angles, unités et quantités
- 3 Mise en forme de listings informatiques

## Attention!

Les auteurs<sup>1</sup> de documents :

- scientifiques
- techniques

écrivant de façon conforme<sup>2</sup> les :

- nombres
- angles
- unités
- quantités

sont moins nombreux qu'on ne l'imagine!

- 
1. Même professionnels!
  2. Aux préconisations du Système International d'Unités (SI)

## Remarque

Des packages L<sup>A</sup>T<sub>E</sub>X aident à se conformer aux normes du si

On étudie ici le package plus abouti : `siunitx`

## Remarque

Dans la suite, `siunitx` supposé systématiquement chargé

## Code source

```
\usepackage{siunitx}
```

## 2 Écriture des nombres, angles, unités et quantités

- **Nombres**
- Angles
- Écriture des quantités
- Listes, produits et pages de nombres et de quantités
- Écriture des unités seulement
- Tableaux de nombres
- Unités binaires
- Raffinements



## Syntaxe

```
\num[<options>]{<nombre>}
```

Code	Résultat
<code>\num{1}</code>	1
<code>\num{12}</code>	12
<code>\num{123}</code>	123
<code>\num{1234}</code>	1234
<code>\num{12345}</code>	12 345

Code	Résultat
<code>\num{1e6}</code>	$1 \times 10^6$
<code>\num{1e15}</code>	$1 \times 10^{15}$
<code>\num{e24}</code>	$10^{24}$
<code>\num{3.45e-4}</code>	$3.45 \times 10^{-4}$
<code>\num{-3e5}</code>	$-3 \times 10^5$

Code	Résultat
<code>\num{0.1}</code>	0.1
<code>\num{0.12}</code>	0.12
<code>\num{0.123}</code>	0.123
<code>\num{0.1234}</code>	0.1234
<code>\num{0.12345}</code>	0.123 45

Code	Résultat
<code>\num{0,1}</code>	0.1
<code>\num{0,12}</code>	0.12
<code>\num{0,123}</code>	0.123
<code>\num{0,1234}</code>	0.1234
<code>\num{0,12345}</code>	0.123 45

### Remarque

Séparateur décimal en :

entrée = « . » ou « , »

⇒ sortie = « . »<sup>1</sup>

### Attention!

Option `locale=FR` ⇒ séparateur décimal en sortie = « , »

---

1. Par défaut

## Code source

```
\usepackage[locale=FR]{siunitx}
```

Code	Résultat	Code	Résultat
<code>\num{0.1}</code>	0,1	<code>\num{0,1}</code>	0,1
<code>\num{0.12}</code>	0,12	<code>\num{0,12}</code>	0,12
<code>\num{0.123}</code>	0,123	<code>\num{0,123}</code>	0,123
<code>\num{0.1234}</code>	0,1234	<code>\num{0,1234}</code>	0,1234
<code>\num{0.12345}</code>	0,123 45	<code>\num{0,12345}</code>	0,123 45

## Remarque

Dans toute la suite, l'option `locale=FR` sera utilisée

## Remarque

Modification du comportement par défaut de `siunitx` :

- au moyen d'`\langle options \rangle`
- pouvant être passées *via* :
  - `\usepackage[\langle options \rangle]{siunitx}`
  - `\sisetup{\langle options \rangle}`<sup>1</sup>

## Attention!

`\sisetup` peut être utilisé :

- dans le corps du document
- **en préambule**<sup>2</sup>

---

1. Bascule  
2. Préférable, par souci d'homogénéité

Mise en forme des **incertitudes** :

**in**habituelle par défaut :

Code	Résultat
<code>\num{1.234(5)}</code>	1,234(5)
<code>\num{1.234(5)e6}</code>	1,234(5) × 10 <sup>6</sup>

~~in~~habituelle grâce à l'option **separate-uncertainty** :

Code source

```
\sisetup{separate-uncertainty}
```

Code	Résultat
<code>\num{1.234(5)}</code>	1,234 ± 0,005
<code>\num{1.234(5)e6}</code>	(1,234 ± 0,005) × 10 <sup>6</sup>



## 2 Écriture des nombres, angles, unités et quantités

- Nombres
- **Angles**
- Écriture des quantités
- Listes, produits et plages de nombres et de quantités
- Écriture des unités seulement
- Tableaux de nombres
- Unités binaires
- Raffinements

## Syntaxe

```
\ang[<options>]{<nombre décimal>}
```

```
\ang[<options>]{<degrés>;<minutes>;<secondes>}
```

## Attention!

Angle en degrés, minutes, secondes

⇒ séparateurs = points-virgules (;)

Code	Résultat
<code>\ang{10}</code>	10°
<code>\ang{12.3}</code>	12,3°
<code>\ang{1;2;3}</code>	1°2'3"
<code>\ang{;;1}</code>	1"
<code>\ang{-10}</code>	-10°
<code>\ang{-10;3;}</code>	-10°3'

## 2 Écriture des nombres, angles, unités et quantités

- Nombres
- Angles
- **Écriture des quantités**
- Listes, produits et pages de nombres et de quantités
- Écriture des unités seulement
- Tableaux de nombres
- Unités binaires
- Raffinements

## Définition

Quantité<sup>1</sup> = *nombre* × *unité*

Ainsi :

$$10 \text{ m} = 10 \times 1 \text{ m} = 10 \times \text{m}$$

$$3,7 \text{ kg} = 3,7 \times 1 \text{ kg} = 3,7 \times \text{kg}$$

---

1. Non adimensionnelle (c.-à-d. ayant une unité)

## Syntaxe (écriture d'une quantité)

$$\backslash qty[\langle options \rangle] \{ \langle nombre \rangle \} \{ \langle unité \rangle \}$$

Dans la ou les  $\langle unité(s) \rangle$  :

## Syntaxe (mise en exposant et en indice)

$$\wedge \{ \langle exposant \rangle \}$$

$$_ \{ \langle indice \rangle \}$$

## Syntaxe (produits d'unités)

$$\langle unité \rangle . \langle unité \rangle$$

## Attention (ancienne syntaxe)!

À partir de `siunitx` v. 3.0.0<sup>1</sup>, syntaxe :

obsolète : `\SI`<sup>2</sup>

nouvelle : `\qty`



- 
1. Sortie le 17 mai 2021
  2. Mais toutefois encore acceptée

Code	Résultat
<code>\qty{10}{m}</code>	10 m
<code>\qty{12.3}{cm^2}</code>	12,3 cm <sup>2</sup>
<code>\qty{10}{kg.m.s^{-2}}</code>	10 kg m s <sup>-2</sup>
<code>\qty{1.453}{g/cm^3}</code>	1,453 g/cm <sup>3</sup>



### Attention!

Marque du produit de 2 unités : au choix?

Marque	Exemple	Accepté?
Espace <sup>1</sup>	10 kg m s <sup>-2</sup>	✓
Point centré	10 kg · m · s <sup>-2</sup>	✓
Point ordinaire	10 kg . m . s <sup>-2</sup>	✗!
Autre	10 kg  m  s <sup>-2</sup>	✗!

Cf. le *Système international d'unités* (SI) <sup>2 3</sup>

1. Par défaut avec `siunitx`
2. Brochure qui définit et présente le SI, éditée par le BIPM
3. Notamment le § « Symboles des unités »

### Remarque

Marque du produit à modifier ? Option `inter-unit-product`

Syntaxe (seule alternative acceptable!)

```
\ssetup{inter-unit-product=\ensuremath{}}\cdot{}}
```

Dans :

Syntaxe (écriture d'une quantité)

`\qty[⟨options⟩]{⟨nombre⟩}{⟨unité(s)⟩}`

`⟨nombre⟩` subit l'effet de `\num`. P. ex. :

- Notation scientifique :

Code	Résultat
<code>\num{3.5e4}</code>	$3,5 \times 10^4$
<code>\qty{3.5e4}{kg}</code>	$3,5 \times 10^4 \text{ kg}$

- Incertitude (avec l'option `separate-uncertainty`) :

Code	Résultat
<code>\num{7.82(1)}</code>	$7,82 \pm 0,01$
<code>\qty{7.82(1)}{kg}</code>	$(7,82 \pm 0,01) \text{ kg}$

## Attention!

Le 1<sup>er</sup> argument de `\qty` ne contiendra donc pas `\num` :

Correct : `\qty{<nombre>}{<unité>}`

Incorrect : ~~`\qty{\num{<nombre>}}{<unité>}`~~

## Remarque

Écriture des unités : commandes **parfois nécessaires**

Code	Résultat
<code>\qty{10}{\celsius}</code>	10 °C
<code>\qty{5}{\micro\gram}</code>	5 µg
<code>\qty{12.3}{\ohm}</code>	12,3 Ω

## Remarque

Écriture des unités : commandes **toujours possibles**

Code	Résultat
<code>\qty{10}{\meter}</code>	10 m
<code>\qty{12.3}{\centi\meter\squared}</code>	12,3 cm <sup>2</sup>
<code>\qty{10}{\kilo\gram\metre\per\second\squared}</code>	10 kg m s <sup>-2</sup>
<code>\qty{1.453}{\gram\per\cubic\centi\metre}</code>	1,453 g cm <sup>-3</sup>

## Attention!

Unités saisies *via* des commandes :

- plus long et compliqué
- mais, comme on va le voir :
  - pas si ~~long~~ ni ~~compliqué~~
  - très avantageux

## Attention!

Méthode à utiliser **en cas de doute** sur l'unité

## Exemple

### Code source

```
1 \begin{itemize}
2 \item \qty{2}{km} ou \qty{2}{Km} ? \qty{2}{\kilo\meter} !
3 \item \qty{3}{bq} ou \qty{3}{Bq} ? \qty{3}{\becquerel} !
4 \end{itemize}
```

### Résultat

- 2 km ou 2 Km? 2 km!
- 3 bq ou 3 Bq? 3 Bq!



**Documents plus homogènes** : on n'écrira pas « kilomètre »

- tantôt « km »
- tantôt « Km »

## Attention!

La plupart des unités disposent d'abréviations

## Exemple

### Code source

- 1 `\qty{3}{\kilo\meter}` en `\qty{2}{\micro\second}`
- 2
- 3 `\qty{3}{\km}` en `\qty{2}{\us}`

### Résultat

3 km en 2  $\mu$ s

3 km en 2  $\mu$ s

Unité	Abréviation	Symbole	Unité	Abréviation	Symbole
femtogram	<code>\fg</code>	fg	millihertz	<code>\mHz</code>	mHz
picogram	<code>\pg</code>	pg	hertz	<code>\Hz</code>	Hz
nanogram	<code>\ng</code>	ng	kilohertz	<code>\kHz</code>	kHz
microgram	<code>\ug</code>	μg	megahertz	<code>\MHz</code>	MHz
milligram	<code>\mg</code>	mg	gigahertz	<code>\GHz</code>	GHz
gram	<code>\g</code>	g	terahertz	<code>\THz</code>	THz
kilogram	<code>\kg</code>	kg	millinewton	<code>\mN</code>	mN
picometre	<code>\pm</code>	pm	kilonewton	<code>\kN</code>	kN
nanometre	<code>\nm</code>	nm	meganeutron	<code>\MN</code>	MN
micrometre	<code>\um</code>	μm	pascal	<code>\Pa</code>	Pa
millimetre	<code>\mm</code>	mm	kilopascal	<code>\kPa</code>	kPa
centimetre	<code>\cm</code>	cm	megapascal	<code>\MPa</code>	MPa
decimetre	<code>\dm</code>	dm	gigapascal	<code>\GPa</code>	GPa
metre	<code>\m</code>	m	milliohm	<code>\mohm</code>	mΩ
kilometre	<code>\km</code>	km	kilohm	<code>\kohm</code>	kΩ
attosecond	<code>\as</code>	as	megohm	<code>\Mohm</code>	MΩ
femtosecond	<code>\fs</code>	fs	picovolt	<code>\pV</code>	pV
picosecond	<code>\ps</code>	ps	nanovolt	<code>\nV</code>	nV
nanosecond	<code>\ns</code>	ns	microvolt	<code>\uV</code>	μV
microsecond	<code>\us</code>	μs	millivolt	<code>\mV</code>	mV
millisecond	<code>\ms</code>	ms	volt	<code>\V</code>	V
second	<code>\s</code>	s	kilovolt	<code>\kV</code>	kV
femtomole	<code>\fmoL</code>	fmol	watt	<code>\W</code>	W
picomole	<code>\pmoL</code>	pmol	microwatt	<code>\uW</code>	μW
nanomole	<code>\nmoL</code>	nmol	milliwatt	<code>\mW</code>	mW
micromole	<code>\umoL</code>	μmol	kilowatt	<code>\kW</code>	kW
millimole	<code>\mmoL</code>	mmol	megawatt	<code>\MW</code>	MW
mole	<code>\mol</code>	mol	gigawatt	<code>\GW</code>	GW
kilomole	<code>\kmoL</code>	kmol	joule	<code>\J</code>	J

Unité	Abréviation	Symbole	Unité	Abréviation	Symbole
picoampere	<code>\pA</code>	pA	kilojoule	<code>\kJ</code>	kJ
nanoampere	<code>\nA</code>	nA	electronvolt	<code>\eV</code>	eV
microampere	<code>\uA</code>	μA	millielectronvolt	<code>\meV</code>	meV
milliampere	<code>\mA</code>	mA	kiloelectronvolt	<code>\keV</code>	keV
ampere	<code>\A</code>	A	megaelectronvolt	<code>\MeV</code>	MeV
kiloampere	<code>\kA</code>	kA	gigaelectronvolt	<code>\GeV</code>	GeV
microlitre	<code>\uL</code>	μL	teraelectronvolt	<code>\TeV</code>	TeV
millilitre	<code>\mL</code>	mL	kilowatt hour	<code>\kWh</code>	kWh
litre	<code>\L</code>	L	farad	<code>\F</code>	F
hectolitre	<code>\hL</code>	hL	femtofarad	<code>\fF</code>	fF
microliter	<code>\uL</code>	μL	picofarad	<code>\pF</code>	pF
milliliter	<code>\mL</code>	mL	kelvin	<code>\K</code>	K
liter	<code>\L</code>	L	decibel	<code>\dB</code>	dB
hectoliter	<code>\hL</code>	hL			

Conférence

LaTeX n° 4

D. Brouzé

Tableaux

Nombres,  
angles,  
quantités

Nombres

Angles

Quantités

Listes, produits et  
pages

Unités seules

Tableaux de  
nombres

Unités binaires

Raffinements

Listings

## Attention!

Unités saisies *via* des commandes : bien sûr abrégables

## Syntaxe

```
\DeclareSIUnit[<options>]{<unité>}{<symbole>}
```

## Remarque

`\DeclareSIUnit` : analogue à `\NewDocumentCommand`<sup>1</sup>

---

1. Mais spécialisée dans les nouvelles unités

## Exemple

### Code source

```
\DeclareSIUnit{\kmh}{\km\per\hour}
```

1 Je cours à `\qty{12}{\kmh}`.

### Résultat

Je cours à 12 km h<sup>-1</sup>.

## Attention!

Avec les **macros personnelles**, on peut :

- **gagner** encore plus de **temps**
- et surtout **séparer fond et forme**

## Exemple

### Code source

```
\DeclareSIUnit{\kmh}{\km\per\hour}
```

```
1 \begin{enumerate}
2 \item Je cours à \qty{12}{\kmh}. % (Pff, long !)
3 \item Puis je marche à \qty{5}{\kmh}. % (Pff, long !)
4 \item Et enfin je dors à \qty{0}{\kmh}. % (Pff, long !)
5 \end{enumerate}
```

### Résultat

- ① Je cours à 12 km h<sup>-1</sup>.
- ② Puis je marche à 5 km h<sup>-1</sup>.
- ③ Et enfin je dors à 0 km h<sup>-1</sup>.

## Exemple

### Code source

```
\DeclareSIUnit{\kmh}{\km\per\hour}
\NewDocumentCommand{\speed}{ m }{\qty{#1}{\kmh}}
```

```
1 \begin{enumerate}
2 \item Je cours à \speed{12}. % (Ah... court !)
3 \item Puis je marche à \speed{5}. % (Ah... court !)
4 \item Et enfin je dors à \speed{0}. % (Ah... court !)
5 \end{enumerate}
```

### Résultat

- ① Je cours à 12 km h<sup>-1</sup>.
- ② Puis je marche à 5 km h<sup>-1</sup>.
- ③ Et enfin je dors à 0 km h<sup>-1</sup>.



### Exemple

#### Code source

```
\NewDocumentCommand{\puis}{ m }{\qty{#1}{\kW}}
\NewDocumentCommand{\tmp}{ m }{\qty{#1}{\celsius}}
```

- 1 Grâce à ma chaudière d'une puissance de `\puis{500}`,
- 2 il fait toujours `\tmp{140}` chez moi !

#### Résultat

Grâce à ma chaudière d'une puissance de 500 kW, il fait toujours 140 °C chez moi!

## Syntaxe (« par »)

`\per`

Format de « par » modifiable au moyen de l'option `per-mode` :

## Syntaxe

`\sisetup{per-mode=<mode>}`

où `<mode>` peut valoir :

`reciprocal`<sup>1</sup> : puissance (-1)

`symbol` : « slash »

`fraction` : « vraie » fraction

---

1. Valeur par défaut

## Exemple

### Code source

```
\DeclareSIUnit{\kmh}{\km\per\hour}
\NewDocumentCommand{\speed}{ m }{\qty{#1}{\kmh}}
```

- ```
1 \begin{enumerate}
2 \item Je cours à \speed{12}.
3 \item \sisetup{per-mode=symbol} Je marche à \speed{5}.
4 \item \sisetup{per-mode=fraction} Je dors à \speed{0}.
5 \end{enumerate}
```

### Résultat

- ① Je cours à  $12 \text{ km h}^{-1}$ .
- ② Je marche à  $5 \text{ km/h}$ .
- ③ Je dors à  $0 \frac{\text{km}}{\text{h}}$ .

## Remarque

Exemple précédent = mauvais exemple car `\ssetup{...}` :

- à utiliser **en préambule** (**exclusivement**<sup>1</sup>)
- de façon à avoir un **document homogène**

## Attention (rappel)!

De manière générale, `\ssetup` : uniquement **en préambule**

---

1. Ou presque

## Syntaxe (carré)

`\square` $\langle$ *unité* $\rangle$  % avant  
 $\langle$ *unité* $\rangle$ `\squared` % après

## Syntaxe (cube)

`\cubic` $\langle$ *unité* $\rangle$  % avant  
 $\langle$ *unité* $\rangle$ `\cubed` % après

## Syntaxe (autre)

`\raiseto` $\{\langle$ *nombre* $\rangle\}$  $\langle$ *unité* $\rangle$  % avant  
 $\langle$ *unité* $\rangle$ `\tothe` $\{\langle$ *nombre* $\rangle\}$  % après

Code

Résultat

`$\$ \backslash qty{1}{\square\cm}$`

`$= \backslash qty{1}{\cm\squared}$`

$1 \text{ cm}^2 = 1 \text{ cm}^2$

`$\$ \backslash qty{1}{\cubic\cm}$`

`$= \backslash qty{1}{\cm\cubed}$`

$1 \text{ cm}^3 = 1 \text{ cm}^3$

`$\$ \backslash qty{1}{\raiseto{4}\cm} = \backslash qty{1}{\cm\tothe{4}}$`

$1 \text{ cm}^4 = 1 \text{ cm}^4$

Remarque (en passant)

`$\backslash qty$`  peut être employée en mode mathématique

## Attention!

Unités *via* des commandes  
⇒ Puissances *via* des commandes <sup>1</sup>

|        |                                             |                                         |
|--------|---------------------------------------------|-----------------------------------------|
| OK     | <code>\qty{1}{\g\cubic\metre}</code>        | $1 \text{ g m}^3$                       |
| OK     | <code>\qty{1}{\g\metre\cubed}</code>        | $1 \text{ g m}^3$                       |
| Pas OK | <del><code>\qty{1}{\g\metre^3}</code></del> | <del><math>1 \text{ g m}^3</math></del> |

1. Pas « manuellement » au moyen du caractère réservé <sup>^</sup>

## 2 Écriture des nombres, angles, unités et quantités

- Nombres
- Angles
- Écriture des quantités
- **Listes, produits et plages de nombres et de quantités**
- Écriture des unités seulement
- Tableaux de nombres
- Unités binaires
- Raffinements



## Syntaxe (écriture d'une liste de nombres et de quantités)

```
\numlist[⟨options⟩]{⟨nb1⟩; ... ;⟨nbn⟩}
```

```
\qtylist[⟨options⟩]{⟨nb1⟩; ... ;⟨nbn⟩}{⟨unité⟩}
```

## Exemple

### Code source

```
1 \begin{itemize}
2 \item \numlist{10;30;45}
3 \item \qtylist{10;30;45}{\metre}
4 \end{itemize}
```

### Résultat

- 10, 30 et 45
- 10 m, 30 m et 45 m

## Syntaxe (écriture d'un produit de nombres et de quantités)

`\numproduct` [*options*] { $\langle nb_1 \rangle x \dots x \langle nb_n \rangle$ }

`\qtyproduct` [*options*] { $\langle nb_1 \rangle x \dots x \langle nb_n \rangle$ } {*unité*}

## Exemple

### Code source

```
1 \begin{itemize}
2 \item \numproduct{2x2x2}
3 \item $\qtyproduct{2x2x2}{\metre}=\qty{8}{\metre\cubed}$
4 \end{itemize}
```

### Résultat

- $2 \times 2 \times 2$
- $2 \text{ m} \times 2 \text{ m} \times 2 \text{ m} = 8 \text{ m}^3$

Attention!

Séparateur pour les produits : **lettre** « x », pas « x »

## Syntaxe (écriture d'une plage de nombres et de quantités)

`\numrange`[*<options>*]{*<nb début>*}{*<nb fin>*}

`\qtyrange`[*<options>*]{*<nb début>*}{*<nb fin>*}{*<unité>*}

## Exemple

### Code source

```
1 \begin{itemize}
2 \item \numrange{10}{45}
3 \item \qtyrange{10}{45}{\metre}
4 \end{itemize}
```

### Résultat

- 10 à 45
- 10 m à 45 m

## 2 Écriture des nombres, angles, unités et quantités

- Nombres
- Angles
- Écriture des quantités
- Listes, produits et pages de nombres et de quantités
- **Écriture des unités seulement**
- Tableaux de nombres
- Unités binaires
- Raffinements

## Syntaxe (écriture d'une unité seule)

```
\unit[⟨options⟩]{⟨unité⟩}
```

## Exemple

### Code source

```
1 Distance exprimée :
2 \begin{itemize}
3 \item en \unit{\um},
4 \item pas en \unit{\kg},
5 \item ni en \unit{\celsius}.
6 \end{itemize}
```

### Résultat

Distance exprimée :

- en  $\mu\text{m}$ ,
- pas en kg,
- ni en  $^{\circ}\text{C}$ .

## Attention (ancienne syntaxe)!

À partir de `siunitx` v. 3.0.0<sup>1</sup>, syntaxe :

obsolète : `\si`<sup>2</sup>

nouvelle : `\unit`

- 
1. Sortie le 17 mai 2021
  2. Mais toutefois encore acceptée

## Attention!

Ne pas confondre les commandes pour l'écriture des :

**quantités :** `\qty{⟨nombre⟩}{⟨unité⟩}`

**unités :** `\unit{⟨unité⟩}`



## Remarque

Les multiples des unités sont des puissances de 10

| Préfixe | Valeur     | Symbole | Macro               |
|---------|------------|---------|---------------------|
| deci    | $10^{-1}$  | d       | <code>\deci</code>  |
| centi   | $10^{-2}$  | c       | <code>\centi</code> |
| milli   | $10^{-3}$  | m       | <code>\milli</code> |
| micro   | $10^{-6}$  | μ       | <code>\micro</code> |
| nano    | $10^{-9}$  | n       | <code>\nano</code>  |
| pico    | $10^{-12}$ | p       | <code>\pico</code>  |
| femto   | $10^{-15}$ | f       | <code>\femto</code> |
| atto    | $10^{-18}$ | a       | <code>\atto</code>  |
| zepto   | $10^{-21}$ | z       | <code>\zepto</code> |
| yocto   | $10^{-24}$ | y       | <code>\yocto</code> |

| Préfixe | Valeur    | Symbole | Macro               |
|---------|-----------|---------|---------------------|
| déca    | $10^1$    | da      | <code>\deca</code>  |
| hecto   | $10^2$    | h       | <code>\hecto</code> |
| kilo    | $10^3$    | k       | <code>\kilo</code>  |
| méga    | $10^6$    | M       | <code>\mega</code>  |
| giga    | $10^9$    | G       | <code>\giga</code>  |
| téra    | $10^{12}$ | T       | <code>\tera</code>  |
| péta    | $10^{15}$ | P       | <code>\peta</code>  |
| exa     | $10^{18}$ | E       | <code>\exa</code>   |
| zetta   | $10^{21}$ | Z       | <code>\zetta</code> |
| yotta   | $10^{24}$ | Y       | <code>\yotta</code> |

## 2 Écriture des nombres, angles, unités et quantités

- Nombres
- Angles
- Écriture des quantités
- Listes, produits et plages de nombres et de quantités
- Écriture des unités seulement
- **Tableaux de nombres**
- Unités binaires
- Raffinements

## Exemple

### Code source

```

1 \begin{tabular}{|r|}
2   \hline
3   Valeurs  \\
4   \hline
5   3876     \\
6   2.345   \\
7   3423.4  \\
8   56.7835 \\
9   -90.473 \\
10  \hline
11 \end{tabular}

```

### Résultat

| Valeurs |
|---------|
| 3876    |
| 2.345   |
| 3423.4  |
| 56.7835 |
| -90.473 |

## Exemple

### Code source

```

1 \begin{tabular}{|c|}
2   \hline
3   Valeurs \\
4   \hline
5   3876      \\
6   2.345    \\
7   3423.4   \\
8   56.7835  \\
9   -90.473  \\
10  \hline
11 \end{tabular}

```

### Résultat

| Valeurs |
|---------|
| 3876    |
| 2.345   |
| 3423.4  |
| 56.7835 |
| -90.473 |

## Exemple

### Code source

```

1 \begin{tabular}{|l|}
2   \hline
3   Valeurs \\
4   \hline
5   3876      \\
6   2.345    \\
7   3423.4   \\
8   56.7835  \\
9   -90.473  \\
10  \hline
11 \end{tabular}

```

### Résultat

| Valeurs |
|---------|
| 3876    |
| 2.345   |
| 3423.4  |
| 56.7835 |
| -90.473 |

## Exemple

### Code source

```

\usepackage{siunitx}

1 \begin{tabular}{|S|}
2   \hline
3   {Valeurs} \\
4   \hline
5   3876      \\
6   2.345     \\
7   3423.4    \\
8   56.7835  \\
9   -90.473  \\
10  \hline
11 \end{tabular}

```

### Résultat

| Valeurs |
|---------|
| 3876    |
| 2,345   |
| 3423,4  |
| 56,7835 |
| -90,473 |



## Remarque

- **S** : nouveau spécificateur de colonne de tableaux<sup>1</sup>
- Dans une colonne de type **S**, **nombres** automatiquement :
  - alignés verticalement sur leur **séparateur décimal**
  - en argument de la commande `\num`

## Exemple (`\num` appliquée aux entrées des colonnes de type S)

### Code source

```

\usepackage{siunitx}
\sisetup{separate-uncertainty}

1 \begin{tabular}{|S|}\hline
2   1.23e4   \\\hline
3   1.23(4)  \\\hline
4 \end{tabular}

```

### Résultat

|                    |
|--------------------|
| $1,23 \times 10^4$ |
| $1,23 \pm 0,04$    |

1. Fourni par `siunitx` donc  $\implies$  `siunitx` chargé

## Exemple

### Code source

```

\usepackage{siunitx}

1 \sisetup{negative-color=red}
2 \begin{tabular}{|S|}
3   \hline
4   {Valeurs} \\
5   \hline
6   3876      \\
7   2.345     \\
8   3423.4   \\
9   56.7835  \\
10  -90.473  \\
11  \hline
12 \end{tabular}

```

### Résultat

| Valeurs |
|---------|
| 3876    |
| 2,345   |
| 3423,4  |
| 56,7835 |
| -90,473 |

### Code source

```
\usepackage{siunitx}
```

```
1 \begin{tabular}{|S|}
2   \hline
3   {Valeurs} \\
4   \hline
5   3876      \\
6   [...]
7 \end{tabular}
```

### Attention!

Entrées de tableau  $\neq$  nombres : à « protéger » par des `{}`

## 2 Écriture des nombres, angles, unités et quantités

- Nombres
- Angles
- Écriture des quantités
- Listes, produits et plages de nombres et de quantités
- Écriture des unités seulement
- Tableaux de nombres
- **Unités binaires**
- Raffinements

**bit** : chiffre binaire<sup>1</sup>, noté « bit »<sup>2</sup>

**byte** : 1 B = 8 bit<sup>3</sup>, noté « B »

## Remarque

Capacité de mémoire informatique : en général exprimée

- en *bytes*
- pas ~~en bits~~

## Attention!

Unités binaires avec `siunitx`  $\implies$  option `binary-units`

1. C'est-à-dire valant 0 ou 1
2. Abréviation de *binary digit*
3. Sauf cas exceptionnels

### Définition (octet)

Un **octet** :

- est **noté** « o »
- **désigne** 8 bits : 1 o = 8 bit = 1 B

### Remarque

L'octet n'est pas ~~prévu~~ par **siunitx**<sup>1</sup>

Mais on peut déclarer une nouvelle unité pour lui :

### Code source

```
\DeclareSIUnit{\octet}{o}
```

1. Car n'est pas une unité du si

## Exemple

### Code source

```
\DeclareSIUnit{\octet}{o}
\sisetup{binary-units,per-mode=symbol}
```

- 1 Le débit d'une ligne ADSL, annoncé à `\og{100}~mégas\fg{}`,
- 2 est en fait de `\qty{100}{\mega\bit\per\second}` donc de
- 3 `\frac{1}{8}\times\qty{100}{\mega\octet\per\second}`,
- 4 c'est-à-dire de `\qty{12.5}{\mega\octet\per\second}`.

### Résultat

Le débit d'une ligne ADSL, annoncé à « 100 mégas », est en fait de 100 Mbit/s donc de  $\frac{1}{8} \times 100 \text{ Mo/s}$ , c'est-à-dire de 12,5 Mo/s.

Multiples des unités binaires : à exprimer <sup>1</sup> en puissances

- non pas de ~~10~~
- mais de 2

| Préfixe       | Valeur   | Symbole | Macro              |
|---------------|----------|---------|--------------------|
| kilo binaire  | $2^{10}$ | Ki      | <code>\kibi</code> |
| méga binaire  | $2^{20}$ | Mi      | <code>\mebi</code> |
| giga binaire  | $2^{30}$ | Gi      | <code>\gibi</code> |
| téra binaire  | $2^{40}$ | Ti      | <code>\tebi</code> |
| péta binaire  | $2^{50}$ | Pi      | <code>\pebi</code> |
| exa binaire   | $2^{60}$ | Ei      | <code>\exbi</code> |
| zetta binaire | $2^{70}$ | Zi      | <code>\zebi</code> |
| yotta binaire | $2^{80}$ | Yi      | <code>\yobi</code> |



## Exemple

### Code source

```
\ssetup{binary-units}
```

```
1 $\qty{1}{\mebi\byte}\neq\qty{1}{\mega\byte}$ car :
2 \begin{align*}
3 \qty{1}{\mebi\byte} &= \qty{1048576}{\byte} \\
4 \qty{1}{\mega\byte} &= \qty{1000000}{\byte} \\
5 \end{align*}
```

### Résultat

1 MiB ≠ 1 MB car :

$$1 \text{ MiB} = 1\,048\,576 \text{ B}$$

$$1 \text{ MB} = 1\,000\,000 \text{ B}$$

## 2 Écriture des nombres, angles, unités et quantités



### ● Raffinements

## Remarque

Options suivantes de `\num` :

- passées localement
- peuvent l'être (semi-)globalement *via* `\sisetup{...}`

## Exemple

### Code source

```
1 \num{12345} et \num[group-digits=false]{12345}
```

### Résultat

12 345 et 12345

## Exemple

### Code source

```

1 \begin{itemize}
2 \item \num{-3} et \num[negative-color=red]{-3}
3 \item \num{-3} et \num[negative-color=blue]{-3}
4 \item \num{-4e3} et \num[negative-color=blue]{-4e3}
5 \item \num{4e-3} et \num[negative-color=blue]{4e-3}
6 \end{itemize}

```

### Résultat

- -3 et -3
- -3 et -3
- $-4 \times 10^3$  et  $-4 \times 10^3$
- $4 \times 10^{-3}$  et  $4 \times 10^{-3}$

## Exemple

### Code source

1 `\num{1e10}` et `\num[exponent-product=\cdot]{1e10}`

### Résultat

$1 \times 10^{10}$  et  $1 \cdot 10^{10}$

### Exemple

#### Code source

```

1 \setup{round-precision=2}
2 \begin{itemize}
3 \item \num{1.234} et \num[round-mode=places]{1.234}
4 \item \num{5.4} et \num[round-mode=places]{5.4}
5 \item \num{-1.234} et \num[round-mode=places]{-1.234}
6 \item \num{1.23(4)} et \num[round-mode=places]{1.23(4)}
7 \end{itemize}

```

#### Résultat

- 1,234 et 1,23
- 5,4 et 5,40
- -1,234 et -1,23
- 1,23(4) et 1,23(4)

## Exemple

### Code source

```

1 \sisetup{round-precision=2}
2 \begin{itemize}
3 \item \num{1.234} et \num[round-mode=figures]{1.234}
4 \item \num{5.4} et \num[round-mode=figures]{5.4}
5 \item \num{-1.234} et \num[round-mode=figures]{-1.234}
6 \item \num{1.23(4)} et \num[round-mode=figures]{1.23(4)}
7 \end{itemize}

```

### Résultat

- 1,234 et 1,2
- 5,4 et 5,4
- -1,234 et -1,2
- 1,23(4) et 1,23(4)



## Exemple

### Code source

```

1 \begin{itemize}
2 \item \num{1.} et \num[add-decimal-zero=false]{1.}
3 \item \num{.5} et \num[add-integer-zero=false]{.5}
4 \end{itemize}

```

### Résultat

- 1 et 1
- 0,5 et 0,5

## Exemple

### Code source

```

1 \begin{itemize}
2 \item \num{1} et \num[explicit-sign=+]{1}
3 \item \num{1e10} et \num[explicit-sign=+]{1e10}
4 \end{itemize}

```

### Résultat

- 1 et +1
- $1 \times 10^{10}$  et  $+1 \times 10^{+10}$

## Conférence L<sup>A</sup>T<sub>E</sub>X n° 4

D. BRIZOUÉ

Tableaux

Nombres,  
angles,  
quantités

Listings

Insertion de listings

Fragments de code

Fonctionnalités  
diverses

Accents dans les  
listings

Usage avancé

Exemples de  
quelques  
fonctionnalités

- 1 Construction de tableaux
- 2 Écriture des nombres, angles, unités et quantités
- 3 Mise en forme de listings informatiques

Parmi les packages les plus aboutis : *listings*<sup>1</sup>

Attention!

*listings* : à charger *après* *babel*

Remarque

Dans la suite, *listings* supposé chargé<sup>2</sup>

Code source

```
\usepackage{listings} % (!) Après `babel`
```

1. D'autres (p. ex. *minted*) sont plus puissants mais moins immédiats
2. De même pour *xcolor*, le cas échéant

```
1 # Calcul de la factorielle en langage Python
2 def factorielle(x):
3     if x < 2:
4         return 1
5     else:
6         return x * factorielle(x-1)
7 str(5) + "!_=_ " + str(factorielle(5))
```

- 1 Mise en relief automatique
  - des mots-clés
  - des commentaires
- 2 Reconnaissance des chaînes de caractères <sup>1</sup>
- 3 Possibilité de numérotation des lignes
- 4 Possibilité de légende et de cadre
- 5 Nombreux langages et dialectes reconnus :
  - orientés math. : Gnuplot, Mathematica, MuPad, R, S, SAS
  - orientés sciences : Matlab, octave, Scilab
  - orientés programmation : bash, C, C++, Fortran, Java, Pascal, Python, Perl, SQL, XML, etc.
  - généralistes : L<sup>A</sup>T<sub>E</sub>X, Ada, HTML, PHP, etc.
- 6 Personnalisation très poussée

---

1. Autrement dit, des *strings*

## 3 Mise en forme de listings informatiques

- Insertion de listings
- Fragments de code
- Fonctionnalités diverses
- Accents dans les listings
- Usage avancé
- Exemples de quelques fonctionnalités

## 3 Mise en forme de listings informatiques

- **Insertion de listings**
  - **Insertion directe**
  - Insertion par importation de fichier
- Fragments de code
- Fonctionnalités diverses
- Accents dans les listings
- Usage avancé
- Exemples de quelques fonctionnalités



## Syntaxe (insertion directe d'un listing)

```
\begin{lstlisting}  
<listing informatique>  
\end{lstlisting}
```

## Syntaxe (insertion directe d'un listing)

```
\begin{lstlisting}[<options>  
<listing informatique>  
\end{lstlisting}
```

### Exemple

#### Code source

```
1 \begin{lstlisting}[language=Python]
2 # Calcul de la factorielle
3 def factorielle(x):
4     if x < 2:
5         return 1
6     else:
7         return x * factorielle(x-1)
8 str(5) + "! = " + str(factorielle(5))
9 \end{lstlisting}
```

#### Résultat

```
1 # Calcul de la factorielle
2 def factorielle (x):
3     if x < 2:
4         return 1
5     else:
6         return x * factorielle (x-1)
7 str(5) + "! = " + str( factorielle (5))
```

Automatiquement :

mots clés du langage : en relief (gras)

commentaires : en relief (italique)

espaces<sup>1</sup> : mis en évidence ( $\sqcup$ )

caractères réservés de L<sup>A</sup>T<sub>E</sub>X : pas interprétés

retours à la ligne : restitués tels quels

---

1. Dans les chaînes de caractères

Des *options* peuvent être :

- passées en argument optionnel de `lstlisting`

## Syntaxe

```
\begin{lstlisting}[options]
```

```
...
```

```
\end{lstlisting}
```

- déclarées de façon (semi-)globale<sup>1</sup> au moyen de :

## Syntaxe

```
\lstset{options}
```

---

1. Mieux!

Ces *options* permettent de **spécifier** :

- le **langage** sous-jacent
- la **mise en forme** :
  - générale
  - des mots clés
  - des commentaires
  - des chaînes de caractères
  - etc.
- etc.

### Exemple

#### Code source

```

1 \lstset{language=Python}
2 \begin{lstlisting}
3 # Calcul de la factorielle
4 def factorielle(x):
5     if x < 2:
6         return 1
7     else:
8         return x * factorielle(x-1)
9 str(5) + "! = " + str(factorielle(5))
10 \end{lstlisting}

```

#### Résultat

```

1 # Calcul de la factorielle
2 def factorielle (x):
3     if x < 2:
4         return 1
5     else:
6         return x * factorielle (x-1)
7 str(5) + "! = " + str( factorielle (5))

```

### Exemple

#### Code source

```

1 \lstset{language=Python,basicstyle=\ttfamily}
2 \begin{lstlisting}
3 # Calcul de la factorielle
4 def factorielle(x):
5     if x < 2:
6         return 1
7     else:
8         return x * factorielle(x-1)
9 str(5) + "! = " + str(factorielle(5))
10 \end{lstlisting}

```

#### Résultat

```

1 # Calcul de la factorielle
2 def factorielle(x):
3     if x < 2:
4         return 1
5     else:
6         return x * factorielle(x-1)
7 str(5) + "! = " + str(factorielle(5))

```

## Attention!

Option `basicstyle=\ttfamily` très fortement conseillée

Dans la suite, on suppose systématiquement chargé

## Code source

```
1 \lstset{language=Python,basicstyle=\ttfamily}
```

## Remarque

D'autres exemples de mise en forme sont vus plus loin



## 3 Mise en forme de listings informatiques

- Insertion de listings
  - Insertion directe
  - Insertion par importation de fichier
- Fragments de code
- Fonctionnalités diverses
- Accents dans les listings
- Usage avancé
- Exemples de quelques fonctionnalités

Autre moyen d'insérer un listing :

- 1 **stocker** son code dans un *⟨fichier annexe⟩*
- 2 **importer** ce *⟨fichier annexe⟩* au moyen de <sup>1</sup> :

Syntaxe

```
\lstinputlisting[⟨options⟩]{⟨fichier annexe⟩}
```

Attention!

Méthode à privilégier pour éviter les soucis de maintenance

---

1. Analogue à `\input`, mais spécialisée dans l'importation de listings

Dans la suite, on considère un **fichier annexe** :

- nommé **facto.py**
- **stockant** le **code** Python précédent :

```
# Calcul de la factorielle en langage Python
def factorielle(x):
    if x < 2:
        return 1
    else:
        return x * factorielle(x-1)
str(5) + "! = " + str(factorielle(5))
```

- avec toujours les réglages :

Code source

```
1 \lstset{language=Python,basicstyle=\ttfamily}
```

On peut alors utiliser :

## Exemple

### Code source

```
1 \lstinputlisting{facto.py}
```

### Résultat

```
1 # Calcul de la factorielle en langage Python
2 def factorielle(x):
3     if x < 2:
4         return 1
5     else:
6         return x * factorielle(x-1)
7 str(5) + "!_=_ " + str(factorielle(5))
```

### Code source

```

1 \lstset{keywordstyle = \color{magenta},
2     keywordstyle = [2]\color{orange},
3     commentstyle = \color{gray}\itshape,
4     stringstyle = \color{cyan}}
5 \lstinputlisting{facto.py}

```

### Résultat

```

1 # Calcul de la factorielle en langage Python
2 def factorielle(x):
3     if x < 2:
4         return 1
5     else:
6         return x * factorielle(x-1)
7 str(5) + "!_=_=" + str(factorielle(5))

```

### Attention!

Mise en forme d'un listing à configurer par le biais :

- soit de **bascules**
- soit de commandes à **argument unique**, p. ex. :

#### Code source

```
1 \lstset{keywordstyle=\fbox}
```

#### Code source

OU

```
1 \lstset{keywordstyle=\colorbox{orange}}
```

## 3 Mise en forme de listings informatiques

- Insertion de listings
- **Fragments de code**
- Fonctionnalités diverses
- Accents dans les listings
- Usage avancé
- Exemples de quelques fonctionnalités

## Remarque

Il est souvent nécessaire :

- d'insérer des **fragments de code** informatique<sup>1</sup>
- **dans** des paragraphes de **texte ordinaire**

## Exemple

### Résultat

Le mot clé **def** définit une fonction et le mot clé **return** retourne une valeur. Les commentaires sont introduits par le caractère #.

---

1. Mis en forme



Ceci peut se faire au moyen de la commande `\linline` :

## Syntaxe

```
\linline⟨caractère⟩⟨frag. code⟩⟨caractère⟩
```

où *⟨caractère⟩* est quelconque à condition d'être

- **unique**
- **identique avant/après** le *⟨frag. code⟩*
- **absent** du *⟨frag. code⟩*
- **≠** des caractères réservés de T<sub>E</sub>X

## Remarque

Des *⟨options⟩* peuvent être spécifiées :

```
\linline[⟨options⟩]⟨carac.⟩⟨frag. code⟩⟨carac.⟩
```

## Exemple

### Code source

```
1 Le mot clé \lstinline|def| définit une fonction et  
2 le mot clé \lstinline|return| retourne une valeur.  
3 Les commentaires sont introduits par le caractère  
4 \lstinline|#|.
```

### Résultat

Le mot clé **def** définit une fonction et le mot clé **return** retourne une valeur. Les commentaires sont introduits par le caractère #.

## Exemple

### Code source

- 1 Le mot clé `\linline|def|` définit une fonction et
- 2 le mot clé `\linline|return|` retourne une valeur.
- 3 Les commentaires sont introduits par le caractère
- 4 `\linline[commentstyle={}]|#|`.

### Résultat

Le mot clé **def** définit une fonction et le mot clé **return** retourne une valeur. Les commentaires sont introduits par le caractère #.

### Remarque

Caractères délimitant le fragment : **variables**

### Exemple

#### Code source

```
1 Les mots clés  
2 \linline|return| et  
3 \linline!return!  
4 sont les mêmes.
```

#### Résultat

Les mots clés **return** et **return** sont les mêmes.

### Remarque

Pour insérer des fragments de code :

- systématiquement commande `\lstinline`?
- Non : cf. section « **Usage avancé** »<sup>1</sup>

---

1. Disponible le cas échéant dans le *cours plus complet sur les listings* 198

## 3 Mise en forme de listings informatiques

- Insertion de listings
- Fragments de code
- **Fonctionnalités diverses**
- Accents dans les listings
- Usage avancé
- Exemples de quelques fonctionnalités

## Code source

```

1 \lstset{numbers=left,frame=single}%
2 %
3 Voici un listing encadré,           joli.
4 \lstinputlisting{facto.py}

```

## Résultat

```

Voici un listing encadré, joli.
1 # Calcul de la factorielle en langage Python
2 def factorielle(x):
3     if x < 2:
4         return 1
5     else:
6         return x * factorielle(x-1)
7 str(5) + "!_=_ " + str(factorielle(5))

```

## Code source

```

1 \lstset{numbers=left,frame=single,%
2     framesep=2pt,aboveskip=1ex}%
3 Voici un listing encadré, encore plus joli.
4 \lstinputlisting{facto.py}

```

## Résultat

Voici un listing encadré, encore plus joli.

```

1 # Calcul de la factorielle en langage Python
2 def factorielle(x):
3     if x < 2:
4         return 1
5     else:
6         return x * factorielle(x-1)
7 str(5) + "!_=" + str(factorielle(5))

```



## Exemple

### Code source (extraction)

```

1 \lstset{basicstyle=\color{white}\ttfamily,numbers=left,
2   numberstyle=\color{red},backgroundcolor=\color{orange}}
3 %
4 Le code complet de notre programme est le suivant :
5 \lstinputlisting{facto.py} % tout
6 %
7 \lstset{numbers=none}
8 %
9 En ligne 1, on y introduit un commentaire :
10 \lstinputlisting[linerange={1-1}]{facto.py} % ligne 1
11 En lignes 2 à 6, on y définit une fonction calculant
12 la factorielle :
13 \lstinputlisting[linerange={2-6}]{facto.py} % lignes 2-6

```

## Exemple (suite)

### Résultat

Le code complet de notre programme est le suivant :

```

1 # Calcul de la factorielle en langage Python
2 def factorielle(x):
3     if x < 2:
4         return 1
5     else:
6         return x * factorielle(x-1)
7 str(5) + "!_=_ " + str(factorielle(5))

```

En ligne 1, on y introduit un commentaire :

```
# Calcul de la factorielle en langage Python
```

En lignes 2 à 6, on y définit une fonction calculant la factorielle :

```

def factorielle(x):
    if x < 2:
        return 1
    else:
        return x * factorielle(x-1)

```

## 3 Mise en forme de listings informatiques

- Insertion de listings
- Fragments de code
- Fonctionnalités diverses
- **Accents dans les listings**
- Usage avancé
- Exemples de quelques fonctionnalités

## Caractères accentués dans les listings :

- parfois inévitables
- notamment dans les **commentaires**

### Exemple

#### Code source

```

1 \begin{lstlisting}
2 # Calcul de la factorielle par récursivité
3 def factorielle(x):
4     if x < 2:
5         return 1
6     else:
7         return x * factorielle(x-1)
8 str(5) + "! = " + str(factorielle(5))
9 \end{lstlisting}

```

### Attention!

Pour des listings :

- comportant des caractères accentués
- avec un codage d'entrée en UTF-8<sup>1</sup>

nécessaire<sup>2</sup> :

- 1 package listings → package listingsutf8
- 2 option inputencoding=utf8/latin1
- 3 se contenter de \lstinputlisting<sup>3</sup>

- 
1. Recommandé
  2. Avec L<sup>A</sup>T<sub>E</sub>X et pdfL<sup>A</sup>T<sub>E</sub>X (pas avec X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X et LuaL<sup>A</sup>T<sub>E</sub>X)
  3. \lstinline et environnement lstlisting inutilisables

### Syntaxe

```
\usepackage{listingsutf8}
```

```
\lstset{%
```

```
  inputencoding=utf8/latin1,
```

```
  <autres options>
```

```
}
```

```
\lstinputlisting{<fichier contenant des accents>}
```

## 3 Mise en forme de listings informatiques



### ● Usage avancé



## Remarque

Fragments de code : saisie systématique de `\lstinline`

- évitable
- en ne saisissant que le *caractère* délimiteur

Pour cela, choisir un *caractère* délimiteur et l'activer :

## Syntaxe

```
\lstMakeShortInline[options]caractère
```

Alors *fragment de code* insérable au moyen de `seulemt` :

## Syntaxe

```
caractèrefragment de codecaractère
```



## Exemple

### Code source

```

1 \lstMakeShortInline|
2 Le mot clé |def| définit une fonction.
  
```

### Résultat

Le mot clé **def** définit une fonction.

## Remarque

Au besoin, le `<caractère>` activé peut être « désactivé »<sup>1</sup>

## Syntaxe (désactivation du `<caractère>`)

```
\lstDeleteShortInline<caractère>
```

---

1. P. ex. si présent dans un nouveau fragment de code à considérer

### Exemple

#### Code source

```

1 \begin{itemize}
2 \item \lstMakeShortInline|           % activation de |
3   Le mot clé |def| définit une fonction.
4 \item \lstDeleteShortInline|       % désactivation de |
5   Le caractère | n'est plus actif : |def|.
6 \item \lstMakeShortInline!         % activation de !
7   Le mot clé !def! définit une fonction.
8 \end{itemize}

```

#### Résultat

- Le mot clé **def** définit une fonction.
- Le caractère | n'est plus actif : |def|.
- Le mot clé **def** définit une fonction.

## 3 Mise en forme de listings informatiques

- Insertion de listings
- Fragments de code
- Fonctionnalités diverses
- Accents dans les listings
- Usage avancé
- Exemples de quelques fonctionnalités

## Code source (exemples de quelques fonctionnalités)

```
\usepackage{graphicx,xcolor,listings}
\everymath{\color{blue}}
\lstset{language=C++,basicstyle=\ttfamily,keywordstyle=\color{magenta},framerule=0.7pt,numbers=left,
stepnumber=2,commentstyle=\color{gray},frame=tb,rulecolor=\color{green},escapechar=",mathescape=true}
```

```
1 \begin{lstlisting}[caption=Calcul de la factorielle en C++,label=facto]
2 double fact(int n)
3 {
4     double f = 1;
5     while (n >= 1) f *= n--;
6     return f;"\label{coucou}"
7 }
8 // OK, mais comment calculer la factorielle de, p. ex.,  $\frac{2}{3}$  ?
9 // Pour ce faire, on dispose heureusement de la fonction
10 //  $\Gamma: z \mapsto \int_0^{\infty} t^{z-1} e^{-t} dt$ ,  $\mathrm{d}t$ .
11 // Mais ma parole, il y a un
12 // "\par\centering\includegraphics[height=2cm]{tiger}\par"
13 // dans ce listing !
14 // "\rmfamily\fbbox{Coucou les amis, je suis dans \LaTeX{!}}"
```

```
15 \end{lstlisting}
16 Le listing-\ref{facto} est instructif, surtout sa ligne-\ref{coucou}.
```

Listing 1 - Calcul de la factorielle en C++

```
1 double fact(int n)
2 {
3     double f = 1;
4     while (n >= 1) f *= n--;
5     return f;
6 }
7 // OK, mais comment calculer la factorielle de, p. ex.,  $\frac{2}{3}$  ?
8 // Pour ce faire, on dispose heureusement de la fonction
9 //  $\Gamma: z \mapsto \int_0^{\infty} t^{z-1} e^{-t} dt$ .
```

```
// Mais ma parole, il y a un
11 //
```



```
13 // dans ce listing !
14 // Coucou les amis, je suis dans \LaTeX{!}
```

Le listing 1 est instructif, surtout sa ligne 5.