

I. Cas d'utilisation en L^AT_EX

Documenter un package ou une classe se fait efficacement via des cas d'utilisation montrant à la fois du code et le résultat correspondant.¹

1. Codes « en ligne »

La macro `\tdoclatexin`² permet de taper du code en ligne via un usage similaire à `\verb` ou bien comme une macro standard (voir la gestion des accolades dans le dernier cas ci-dessous). Voici des exemples d'utilisation.

1: <code>\tdoclatexin \$a^b = c\$ </code> <code>\\</code>	1: $a^b = c$
2: <code>\tdoclatexin+\tdoclatexin \$a^b = c\$ +</code> <code>\\</code>	2: <code>\tdoclatexin \$a^b = c\$ </code>
3: <code>\tdoclatexin{\tdoclatexin \$a^b = c\$ }</code>	3: <code>\tdoclatexin \$a^b = c\$ </code>

Note.

La macro `\tdoclatexin` est utilisable dans une note de pied de page : voir ci-dessous.^a De plus, une couleur de fond est volontairement utilisée pour subtilement faire ressortir les codes `LaTeX`.

a. `$minted = TOP$` a été tapé `\tdoclatexin+$minted = TOP$+` dans cette note de bas de page..

2. Codes tapés directement

Exemple I.1 (Face à face). Afficher un code et son rendu côte à côte se fait comme suit où la macro `\tdoctcb` permet de juste taper `\tdoctcb{sbs}` au lieu de `listing side text` (`sbs` est pour « `s`-ide `b`.`y` `s`-ide » soit « côte à côte » en anglais, tandis que `tcb` est l'abréviation standard de `tcolorbox`).

```
\begin{tdoclatex}[\tdoctcb{sbs}]
  $A = B + C$
\end{tdoclatex}
```

Ceci produira ce qui suit.

$A = B + C$	$A = B + C$
-------------	-------------

Note.

L'argument optionnel de l'environnement `tdoclatex` doit être un style pour un `tcblisting`. Par exemple, on peut faire les modifications suivantes si besoin.^a

```
\begin{tdoclatex}[\tdoctcb{sbs},
  top      = 10pt,
  bottom   = 10pt]
  $A = B + C$
\end{tdoclatex}
```

Ceci produira ce qui suit.

$A = B + C$	$A = B + C$
-------------	-------------

a. Cette documentation utilise ceci pour obtenir des rendus corrects de codes L^AT_EX produisant des cadres ombrés : voir la section ??.

Exemple I.2 (À la suite). `\begin{tdoclatex}... \end{tdoclatex}` produit le résultat suivant où l'option par défaut correspondant à `\tdoctcb{std}`.³

$A = B + C$

$A = B + C$

Exemple I.3 (Juste le code). Via l'option `\tdoctcb{code}`, on aura juste le code comme ci-après.

1. La mise en forme des codes se fait via les packages `minted` et `tcolorbox`.
2. Le nom de la macro `\tdoclatexin` vient de « *in-line L^AT_EX* » soit « *L^AT_EX en ligne* » en anglais.
3. `std` fait référence au comportement « *standard* » de `tcolorbox` vis à vis de la librairie `minted`.

$$A = B + C$$

⚠ Avertissement.

Avec la mise en forme par défaut, si le code commence par un crochet ouvrant, il faudra indiquer explicitement l'option par défaut. Considérons le code suivant.

```
\begin{tdoclatex}[\tdoctcb{std}]
  [Étrange... Ou pas !]
\end{tdoclatex}
```

Ceci produira ce qui suit.

```
[Étrange... Ou pas !]
-----
[Étrange... Ou pas !]
```

Une autre méthode consiste à utiliser la primitive `\string`. Considérons le code suivant.

```
\begin{tdoclatex}
  \string[Étrange... Ou pas !]
\end{tdoclatex}
```

Ceci produira ce qui suit.

```
[Étrange... Ou pas !]
-----
[Étrange... Ou pas !]
```