

1. Codes importés

Pour les codes suivants, on considère un fichier de chemin relatif `examples-listing-xyz.tex`, et ayant le contenu suivant.

```
% Juste une démo.  
$x y z = 1$
```

La macro `\tdoclatexinput`, présentée ci-dessous, attend le chemin d'un fichier et propose les mêmes options que l'environnement `\begin{tdoclatex} ... \end{tdoclatex}`.

Exemple 1..1 (Face à face).

```
\tdoclatexinput[sbs]{examples/listing/xyz.tex}
```

Ceci produit la mise en forme suivante.

<pre>% Juste une démo. \$x y z = 1\$</pre>	$xyz = 1$
--	-----------

Exemple 1..2 (À la suite).

```
\tdoclatexinput{examples/listing/xyz.tex}
```

Ceci produit la mise en forme suivante où l'option employée par défaut est `std`.

```
% Juste une démo.  
$x y z = 1$
```

 $xyz = 1$

Exemple 1..3 (Juste le code).

```
\tdoclatexinput[code]{examples/listing/xyz.tex}
```

Ceci produit la mise en forme suivante.

```
% Juste une démo.  
$x y z = 1$
```

2. Codes importés et mis en situation

Exemple 2..1 (Mise en situation). Ce qui suit s'obtient via `\tdoclatexshow{examples-listing-xyz.tex}`.

————— Début du rendu dans cette doc. —————

```
% Juste une démo.  
$x y z = 1$
```

Ceci donne :

————— Début du rendu réel —————

 $xyz = 1$

————— Fin du rendu réel —————

————— Fin du rendu dans cette doc. —————

Note.

Les textes par défaut tiennent compte de la langue choisie lors du chargement du package `tutodoc`.

Exemple 2..2 (Changer le texte explicatif). Via la clé `explain`, on peut utiliser un texte personnalisé. Ainsi, `\tdoclatexshow[explain = Voici le rendu réel.]{examples-listing-xyz.tex}` produira ce qui suit.

■ Début du rendu dans cette doc. ■

```
% Juste une démo.  
$x y z = 1$
```

Voici le rendu réel.

■ Début du rendu réel ■

$xyz = 1$

■ Fin du rendu réel ■

■ Fin du rendu dans cette doc. ■

Exemple 2..3 (Les options disponibles). En plus du texte explicatif, il est aussi possible d'utiliser toutes les options de `\begin{tdocshowcase} ... \end{tdocshowcase}`, voir ?? page ??. Voici un exemple illustrant ceci.

```
\tdoclatexshow[explain = Ce qui vient est coloré...,  
               before  = Rendu ci-après.,  
               after   = Rendu fini.,  
               color   = orange]  
{examples/listing/xyz.tex}
```

Ceci va produire ce qui suit.

■ Début du rendu dans cette doc. ■

```
% Juste une démo.  
$x y z = 1$
```

Ce qui vient est coloré...

■ Rendu ci-après. ■

$xyz = 1$

■ Rendu fini. ■

■ Fin du rendu dans cette doc. ■