

1. Imported codes

For the following codes, consider a file with the relative path `examples-listing-xyz.tex`, and with the following contents.

```
% Just one demo.
$x y z = 1$
```

The `\tdoclatexinput` macro, shown below, expects the path of a file and offers the same options as the `\begin{tdoclatex} ... \end{tdoclatex}` environment.

Example 1 (Side by side).

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

This produces the following layout.

<pre>% Just one demo. \$x y z = 1\$</pre>	$xyz = 1$
---	-----------

Example 2 (Following).

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

This produces the following formatting where the default option is `std`.

```
% Just one demo.
$x y z = 1$
```

 $xyz = 1$

Example 3 (Just the code).

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

This produces the following layout.

```
% Just one demo.
$x y z = 1$
```

2. Imported codes put into practice

Example 1 (Showcase). The following comes from `\tdoclatexshow{examples-listing-xyz.tex}`.

Start of the rendering in this doc.

```
% Just one demo.
$x y z = 1$
```

This gives :

Start of the real output

$xyz = 1$

End of the real output

End of rendering in this doc.

Note.

The default texts take into account the language chosen when loading the package `tutodoc`.

Example 2 (Changing the explanatory text). Using the key `explain`, you can use custom text. Thus, `tdoclatexshow[explain = Here is the actual rendering.]{examples-listing-xyz.tex}` will produce the following.

Start of the rendering in this doc.

```
% Just one demo.  
$x y z = 1$
```

Here is the actual rendering.

Start of the real output

$xyz = 1$

End of the real output

End of rendering in this doc.

Example 3 (The options available). In addition to the explanatory text, it is also possible to use all the options of `\begin{tdocshowcase} ... \end{tdocshowcase}`, see ?? page ??. Here is an example to illustrate this.

```
\tdoclatexshow[explain = What comes next is colourful...,  
               before  = Rendering below.,  
               after   = Finished rendering.,  
               color   = orange]  
{examples/listing/xyz.tex}
```

This will produce the following.

Start of the rendering in this doc.

```
% Just one demo.  
$x y z = 1$
```

What comes next is colourful...

Rendering below.

$xyz = 1$

Finished rendering.

End of rendering in this doc.