

I. A real-life rendering

It is sometimes useful to render code directly in the documentation. This type of rendering must be dissociable from the explanatory text.

1. With a coloured stripe

Example 1 (With default text). *It can be useful to show a real rendering directly in a document ¹. This is done via `\begin{tdocshowcase} ... \end{tdocshowcase}` as follows.*

```
\begin{tdocshowcase}
  \bfseries A bit of code \LaTeX.

  \bigskip

  \emph{\large End of the awful demo.}
\end{tdocshowcase}
```

The result is the following rendering ².

Start of the real output

A bit of code **\LaTeX** .

End of the awful demo.

End of the real output

Remark. See the section ?? on page ?? to easily obtain code followed by its actual rendering as in the previous example.

Note. The explanatory texts adapt to the language chosen when *tutodoc* is loaded.

Example 2 (Change the default colour and/or text).

```
\begin{tdocshowcase}[before = My beginning,
                      after  = My end,
                      color   = red]
  Bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla...
\end{tdocshowcase}
```

This will produce the following.

My beginning

Bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla...

My end

Note. You will have noticed that we don't obtain a pure red: behind the scenes, the expandable macros `\tdocbackcolor` and `\tdocdarkcolor` are used to create the background and title colours respectively from the ones proposed in `\begin{tdocshowcase} ... \end{tdocshowcase}`. These macros have a single argument, the chosen colour, and accept the following codes.

```
\NewExpandableDocumentCommand{\tdocbackcolor}{m}{\#1!5}
\NewExpandableDocumentCommand{\tdocdarkcolor}{m}{\#1!50!black}
```

Warning. With the default settings, if the code to be formatted begins with an opening bracket, use `\string` as in the following example.

¹Typically when making a demo.

²Behind the scenes, the strip is created effortlessly using the `clrstrip` package.

```
\begin{tdocshowcase}
  \string[This works...]
\end{tdocshowcase}
```

This will produce the following.

Start of the real output

[This works...]

End of the real output

Note. Behind the scenes, the `\tdocruler` macro is used.

```
\tdocruler{Un pseudo-titre décoré}{red}
```

Un pseudo-titre décoré