

I. Specify packages, classes, macros or environments

Here's what you can type semantically.

<code>\tdoccls{myclass}</code>	<code>is for...</code>	<code>\\</code>	<code>myclass</code>	<code>is for...</code>
<code>\tdocpack{mypackage}</code>	<code>is for...</code>	<code>\\</code>	<code>mypackage</code>	<code>is for...</code>
<code>\tdocmacro{onemacro}</code>	<code>is for...</code>	<code>\\</code>	<code>\onemacro</code>	<code>is for...</code>
<code>\tdocenv{env}</code>	<code>produces...</code>	<code>\\</code>	<code>\begin{env} ... \end{env}</code>	<code>produces...</code>
<code>\tdocenv[[{opt1}<opt2>]]{env}</code>		<code>\\</code>	<code>\begin{env}[opt1]<opt2> ... \end{env}</code>	
Just <code>\tdocenv*{env}</code>	<code>...</code>	<code>\\</code>	<code>\begin{env}[opt1]<opt2> ... \end{env}</code>	
Finally <code>\tdocenv*[[{opt1}<opt2>]]{env}</code>	<code>...</code>		Just <code>env...</code>	
<code>% For copy and paste.</code>			Finally <code>env...</code>	

Remark I.1. Unlike `\tdoclatexin`, `\tdocenv` and `\tdocenv*` macros don't color the text they produce. In addition, `\tdocenv{monenv}` produces `\begin{monenv} ... \end{monenv}` with spaces to allow line breaks if required.

Warning.

The optional argument of the `\tdocenv` macro is copied and pasted ^a when rendering. This may sometimes require the use of protective braces, as in the example above.

^aRemember that almost anything is possible from now on.

II. Origin of a prefix or suffix

To explain the names chosen, there is nothing like indicating and explaining the short prefixes and suffixes used. This is easily done as follows.

<code>\tdocpre{sup}</code>	<code>relates to...</code>	<code>\\</code>	<code>sup</code>	<code>relates to...</code>
<code>\tdocprewhy{sup.erbe}</code>	<code>means...</code>	<code>\\</code>	<code>sup.erbe</code>	<code>means...</code>
<code>\emph{\tdocprewhy{sup.er} for...}</code>			<code>sup.er</code>	<code>for...</code>

Remark II.1. The choice of a full stop to split a word allows words with a hyphen to be used, as in `\tdocprewhy{bric.k-breaker}` which gives `bric.k-breaker`.