

The `alltt` environment*

Johannes Braams

1997/06/16

This file is maintained by the L^AT_EX Project team.
Bug reports can be opened (category `latex`) at
<https://latex-project.org/bugs.html>.

Abstract

This package defines the `alltt` environment, which is like the `verbatim` environment except that `\`, `{`, and `}` have their usual meanings.

Thus, other commands and environments can appear within an `alltt` environment.

1 Introduction

`alltt` Here are some things you may want to do in an `alltt` environment:

- Change fonts—e.g., by typing `{\em emphasized text\}`
- Insert text from a file `foo.tex` by typing `\input{foo}`. Beware that each `<return>` starts a new line, so if `foo.tex` ends with a `<return>` you can wind up with an extra blank line if you're not careful.
- Insert a math formula. Note that `$` just produces a dollar sign, so you'll have to type `\(...\)` or `\[...\]`. Also, `^` and `_` just produce their characters; use `\sp` or `\sb` for super- and subscripts, as in `\(x\sp{2}\)`.

2 The Implementation

```
1 (*package)
```

`alltt` The `alltt` environment is similar to the `verbatim` environment, except that `\`, `{` and `}` have their usual behaviour.

```
2 \begingroup
3 \lccode'\~='\'
4 \lowercase{\endgroup}
5 \newenvironment{alltt}{%
6   \trivlist
7   \item\relax
```

*This file has version number v2.0g, last revised 1997/06/16.

```

8      \if@minipage
9      \else
10         \vskip\parskip
11      \fi
12      \leftskip\@totalleftmargin
13      \rightskip\z@skip
14      \parindent\z@
15      \parfillskip\@flushglue
16      \parskip\z@skip
17      \@@par
18      \@tempwafalse
19      \def\par{%
20         \if@tempwa
21            \leavevmode\null\@@par\penalty\interlinepenalty
22         \else
23            \@tempwattrue
24            \ifhmode\@@par\penalty\interlinepenalty\fi
25         \fi}
26      \obeylines
27      \verbatim@font
28      \let\org@prime~%
29      \@noligs

30      \everymath\expandafter{\the\everymath
31         \catcode'\'=12 \let~\org@prime}
32      \everydisplay\expandafter{\the\everydisplay
33         \catcode'\'=12 \let~\org@prime}
34      \let\org@dosppecials\dosppecials
35      \g@remfrom@specials{\}
36      \g@remfrom@specials{\{ }
37      \g@remfrom@specials{\} }
38      \let\do\@makeoother
39      \dosppecials
40      \let\dosppecials\org@dosppecials
41      \frenchspacing\@vobeyspaces
42      \everypar \expandafter{\the\everypar \unpenalty}}
43 {\endtrivlist}}

```

`\g@remfrom@specials` In the old implementation of the `alltt` package a fixed `\dosppecials` was used. However nowadays the `\dosppecials` command might contain more special characters at run-time than as was defined in the format. Therefore we remove the necessary special character from `\dosppecials` at the start of the `alltt` environment. For this we need a macro. Remember that the list of special characters in `\dosppecials` contains the control sequence `\do` between the characters. We use that to check whether a character has to be removed.

The macro `\g@remfrom@specials` takes one argument, the character to be removed from the list.

```
44 \def\g@remfrom@specials#1{%
```

We build up a new list in `\@new@speicals`.

```
45 \def\@new@specials{}
```

The command `\@remove` compares its argument with the argument of `\g@remfrom@specials`.

```
46 \def\@remove##1{%
```

```
47 \ifx##1#1\else
```

When they are not the same the argument of `\@remove` is added (together with `\do`) to the new list.

```
48 \g@addto@macro\@new@specials{\do ##1}\fi}
```

Now we `\let \do` be equal to `\@remove` and execute `\dospecials`.

```
49 \let\do\@remove\dospecials
```

All that's left is to make `\dospecials` point to the new list.

```
50 \let\dospecials\@new@specials
```

```
51 }
```

```
52 </package>
```