

The `visualtoks` Package, version 1.1b

plante

September 1, 2025

In The `TEX`book, Knuth demonstrates the concept of tokens with the following example:

For example, if the normal conventions of plain `TEX` are in force, the text ‘`{\hskip 36 pt}`’ is converted into a list of eight tokens:

`{`₁ `\hskip` _{3₁₂} `36`_{6₁₂} _{␣₁₀} `pt`_{p₁₁} `}`₂

The subscripts here are the category codes, as listed earlier: 1 for “beginning of group,” 12 for “other character,” and so on. The `\hskip` doesn’t get a subscript, because it represents a control sequence token instead of a character token. Notice that the space after `\hskip` does not get into the token list, because it follows a control word. (p. 38)

The same style of token display is used several times in the `TEX`book. It would be useful to be able to generate the display automatically for an arbitrary list of tokens, for pedagogical or debugging purposes. This package provides the `\visualtoks` command which does exactly that.

Usage

Usage: `\visualtoks{<token list>}`.

This package may be loaded by `\input{visualtoks}` (plain `TEX` and other formats) or `\usepackage{visualtoks}` (`LATEX`). It depends on the `tokmap` package.

The horizontal separation between displayed tokens may be configured by the dimension register `\visualtokssep`. The default value is 1em.

`<token list>` must be balanced with respect to explicit braces, and must not contain the token `\tokmap@nil`. It is assumed that `{` and `}` are the only characters with category codes 1 (beginning of group) and 2 (end of group) respectively.

An *anomalous* control sequence is one that differs in shape from the control sequence with the same name constructed by `\csname`. Anomalous control sequences are marked with a star next to their box.

Samples

- `\visualexpr{\def \macro#1{abc #1\egroup}}` gives
`def macro #6 112 {1 a11 b11 c11 \l10 #6 112 egroup }2.`
- `\visualexpr{\halign{&##\hfil\cr}\par}` gives
`$3 $3 halign {1 &4 #6 #6 hfil cr}2 $3 $3 par.`
- Unbalanced `\if...` tokens:
`\visualexpr{\ifnum\iffalse{\fi} = 0\else}` gives
`ifnum iffalse {1 fi '12 }2 \l10 =12 \l10 012 else.`
- To demonstrate how TeX tokenizes consecutive spaces:
`\makeatletter \edef\temp{\l\l\l\l\l}{\@spaces}`
`\expandafter\visualexpr\expandafter{\temp}` gives
`{1 \l10 }2 {1 \l10 \l10 \l10 \l10 }2.`
- To demonstrate the `\lowercase` technique:
`\begingroup`
`\lccode'&=' $ \lccode'#=' $ \lccode'^=' $ \lccode'_=' $`
`\lccode'_=' $ \lccode'A=' $ \lccode'?=' $ \lccode'~=' $`
`\lowercase\endgroup\def\temp{&##^_A?~}`
`\expandafter\visualexpr\expandafter{\temp}` gives
`$3 $4 $6 $7 $8 $10 $11 $12 $13.`
- To show anomalous tokens:
`\font\tenrm=cmr10 \tenrm`
`\expandafter\visualexpr\expandafter{\the\font \tenrm}` gives
`tenrm* tenrm.`

License

This package is copyright © 2025 plante, and released under the L^AT_EX Project Public License (LPPL) 1.3c.

Repository

The upstream repository of this package may be found at

<https://github.com/plante3/visualexpr/tree/main>.