

The `subtext` package

Version 1.1

Palle Jørgensen

30th May 2019

1 Usage

1.1 Loading the package

To load the package type

```
\usepackage{subtext}
```

into your preamble.

1.2 Using the package

In math mode use the `_` character as usual to get the usual result.

```
\begin{equation}
\label{eq:1}
a_{long\ text}
\end{equation}
```

yielding

$$a_{longtext} \tag{1}$$

Replace curly braces with square brackets to get the subscript typeset using `amstext`'s `\text` command.

```
\begin{equation}
\label{eq:2}
a_{[long\ text]}
\end{equation}
```

yielding

$a_{\text{long text}}$ (2)

That's all!

1.3 Package options

Due to the behaviour of the `\amstext` command, the typeface of the text within the brackets is the same as the surrounding text. This may not always be desirable – I myself often find it somewhat annoying. Thus there are two package options

`upright` Makes the text within the brackets be in upright shape. The font family (sans serif, serif or others) and the series (e.g. boldface) is the same as the surrounding text. This is the authors favourite.

`normal` Makes the text within the brackets be in the documents `\normalfont` style.

2 License

The license of `subtext.sty` and related files is GNU GENERAL PUBLIC LICENSE.

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<https://fsf.org/>> Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed. <<https://www.gnu.org/licenses/why-not-lGPL.html>>.

View COPYING for the full license.

3 Source of **subtext.sty**

```
\ProvidesPackage{subtext}[2019/05/30 v. 1.1 LaTeX package for easy
text-style subscripts in math mode.]
\RequirePackage{amstext}
\def\subtext@font{}
\DeclareOption{upright}{\def\subtext@font{\upshape}}
\DeclareOption{normal}{\def\subtext@font{\normalfont}}
\ProcessOptions*
\let\saved@sub_%
\catcode`_ \active
\def\new@sub@text[#1]{\saved@sub{\text{\subtext@font #1}}}
\def\new@sub@other#1{\saved@sub{#1}}
\def\new@sub{%
  \@ifnextchar[%
    {\new@sub@text}{\new@sub@other}}
\protected\def_{\new@sub}
\endinput
```