The doctools package

Matthias Pospiech
matthias@pospiech.eu

0.1 from 2014/06/27

Contents

1 Documentation

This package is a collections of tools for the documentation of LaTeX code either in a normal document or within a dtx file.

1.1 Options

This package has three options

`applyLayout`	true/false (default: false)
`loadHyperref`	true/false (default: false)
`createIndexEntries`	true/false (default: true)

The option `applyLayout` should only be used in dtx files for which the layout of this package shall be applied. For normal documents setting this option to true could disturb the layout.

The selection of the other two options (`loadHyperref`, `createIndexEntries`) should follow the following principles, which are mostly related to the fact that doc.sty does not support package `hyperref` (prohibits display of index entries) and requires the package `hypdoc` instead.

Package `doctools` is typically loaded before most other packages and in the case of a dtx file possible the only package which is necessary to load.

Selection of options inside a dtx file

In a dtx file index entries are created by the package `doc.sty` which have a format the requires the use of the format file `gind.ist` for makeindex:

```
makeindex.exe -s gind.ist filename.idx
```

The index entries by `doc.sty` are created if `\PageIndex` or `\CodelineIndex` are defined except if `\CodelineNumbered` is defined. With the last definition no index is created, but the code lines are still numbered. In this case `hyperref` has no influence on the index creation since the index is never created:

```
\CodelineNumbered
\usepackage[
  loadHyperref=true,%
  createIndexEntries=false% (has no influence)
]{doctools} %
```

If in contrast the index entries of `doc.sty` are activated using `\PageIndex` or `\CodelineIndex`, then `hyperref` should not be loaded because it prohibits the display of the index entries created by `doc.sty`. Furthermore the index entries

of `doctools` should be disabled because they use a different format and would be displayed wrong.

```
\PageIndex
\usepackage[
  loadHyperref=false,% (would prohibit doc.sty index entries)
  createIndexEntries=false% (would be displayed wrong)
]{doctools} %
\usepackage{hypdoc}
```

If the index entries of `doc.sty` are supposed to be displayed with hyperlinks the package `hypdoc` must be loaded instead[1]. If the option `loadHyperref` is used all styles are applied although `hyperref` is not loaded, because it is already loaded by the package `hypdoc`. Take care of the loading order: `hypdoc` must be loaded before `doctools`.

```
\PageIndex
\usepackage{hypdoc} % load before doctools !
\usepackage[
  loadHyperref=true,% apply hyperref styles
  createIndexEntries=false% (would be displayed wrong)
]{doctools} %
```

However, if only the index entries of `doctools.sty` are supposed to be displayed in the index, this can be realized by loading `hyperref`. In this case the format file for `makeindex` must not be specified.

```
\PageIndex % creates index, but entries are not displayed
\usepackage[
  loadHyperref=true,% (prohibit doc.sty index entries)
  createIndexEntries=true%
]{doctools} %
```

Selection of options in a normal LaTeX file

In a normal LaTeX file that does not load `doc.sty` the package options may be selected according to the needs of the user. If both index entries shall be created and `hyperref` can be safely loaded one could load the package with these options:

```
\usepackage[
  loadHyperref=true,%
  createIndexEntries=true%
]{doctools} %
```

[1] This issue is explained very well in this thread: http://tex.stackexchange.com/questions/87670/disable-index-creation-in-dtx-file

1.2 Commands provided by doc.sty and ltxdoc

For the complete reference refer to `doc.pdf` and `ltxdoc.pdf`. Here only the most relevant commands for the code in the documentation part are described.

`\DescribeMacro` {⟨*macro*⟩}

Prints out the macro name in the margin and adds the command to the index. It primary usage is to indicate the part of the documentation where the usage of this macro is described.

`\DescribeEnv` {⟨*environment*⟩}

Is used analogues to `\DescribeMacro`.

`\marg` {⟨*argument*⟩}

Mandatory argument, printed as {⟨*argument*⟩}

`\oarg` {⟨*argument*⟩}

Optional argument, printed as [⟨*argument*⟩]

`\meta` Both `\marg` and `\oarg` use `\meta` to print out the argument as ⟨*argument*⟩.

If `ltxdoc` is not used both `\marg` and `\oarg` are instead defined by `doctools`.

1.3 Commands provided by doctools.sty

`\bs` Shortcut for `\textbackslash`.

`\command` {⟨*cmd*⟩}

Prints out the argument `\command{foo}` as `\foo`.

`\cs` {⟨*cmd*⟩}

Shortcut for `\command`. Also defined by `ltxdoc`.

`\arg` {⟨*cmd*⟩}

Prints out an argument in curled brackets without the use of angle brackets as in `\marg` or `\oarg`. Thus prints `\arg{foo}` as {`foo`}.

`\environment` {⟨*environment*⟩}

Prints out an environment name as `environment`.

`\env` {⟨*environment*⟩}

Shortcut for `\environment`

`\package` {⟨*package*⟩}

Prints out a package name as `package`.

`\ltxclass` {⟨*LaTeX documentclass*⟩}

Prints out a class name as `class`.

`\option` {⟨*option*⟩}

Prints out an option as `option`.

`\parameter` {⟨*parameter*⟩}

Prints out an option as `parameter`.

`\person` {⟨*person*⟩}

Print out name of a person, for example for the acknowledge of their help during the development of some code.

Example: This package was written by Matthias Pospiech.

`\AfterLastParam` Used in conjuction with `\DescribeMacro` and the printout of arguments using `\marg` or `\oarg`. `\AfterLastParam` is inserted at the end to begin a new line.

Example: `\DescribeMacro{foo}\marg{argument} \AfterLastParam`

`\Default` `\Default` is used to print out the default value of a command.

`\Example` `\Example` is used to print out an example for the usage of a command. See documentation of `\AfterLastParam` for an example.

`\latex` New lower case command for printing out LaTeX.

`Optionlist` Table to print out all possible options of a command. See the following code for an example.

```
\begin{Optionlist}
 top     &  placement is on top  \\
 bottom  &  placement is at the bottom\\
\end{Optionlist}
```

Output:

`top`	placement is on top
`bottom`	placement is at the bottom

1.4 Printing of LaTeX code

This package provides two `listings` styles for the printing of LaTeX code: *lstDemoStyleLaTeXCode* and *lstDocStyleLaTeXCode*. The first is meant for printing of code examples, the latter only to be used inside a dtx file for the documentation of the package code.

For convenience these style are provided with the following environments.

`latexcode` Listings environment for code examples. The output of this style is shown in the

example below. The lines are not numbered, so that the code can be directly copied from the PDF document.

```
\begin{latexcode}
% comment
Example Code \ldots
\end{latexcode}
```

In the case of a dtx file this must be wrapped in special dtx code as shown in the following code.

```
% \iffalse
%<*example>
% \fi
\begin{latexcode}
Example Code \ldots
\end{latexcode}
% \iffalse
%</example>
% \fi
```

`macrocode` is in contrast an environment for the code in a dtx file, which is used for the creation of a latex package or class. In a dtx file therefore the special comment and indentation of the environment must be taken care of. The following example is displayed with this style.

```
%    \begin{macrocode}
% definition of foobar
\newcommand{\foobar}{foo-bar}
%    \end{macrocode}
```

`\printCodeFromFile` `[`⟨*first line*⟩`]` `{`⟨*last line*⟩`}` `{`⟨*file name*⟩`}`

This command is very helpful to display parts of a file sequentially. If the first line is not given the last line value is incremented by one and used as the first line.

Example: `\printCodeFromFile[3]{6}{LaTeXTemplate.tex}`

1.5 Label Link system for files and their usage

The system explained in this section allows to adds a hyperlink to sections where a file is discussed and adds the usage page to the index. The code is very close to the label/ref system as shown in the following example:

```
\section{about links in pdf}
The code is in \file{preamble/hyperref.tex}
\dots
\section{preamble/hyperref.tex}
\labelfile{preamble/hyperref.tex}
```

`\labelfile` {⟨*filepath/filename.tex*⟩}

Creates a label that is used by `\file` for linking to the occurrence of `\labelfile`. Furthermore an index entry is created with a new sub-index for every path. In the case of `\labelfile{preamble/hyperref.tex}` this would look like:

```
Files
   preamble
      hyperref.tex
```

`\file` {⟨*filepath/filename.tex*⟩}

Prints out the filename in typewriter font and links to the usage if an equivalent `\labelfile` is given.

2 Implementation

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{doctools}[2012/12/01 v0.1 commands and packages for
documenting LaTeX Code]
%
```

2.1 Define keys

```
%%% === Define Keys ===================================
\RequirePackage{kvoptions-patch}
\RequirePackage{kvoptions}  % options
\RequirePackage{pdftexcmds} % string comparison
\SetupKeyvalOptions{family=doctools,prefix=doctools@}
%
```

Define default option for style key: *stacked*

```
\DeclareBoolOption[false]{loadHyperref}
\DeclareBoolOption[true]{createIndexEntries}
\DeclareBoolOption[false]{applyLayout}
\ProcessKeyvalOptions{doctools}
%
```

2.2 Preamble

Packages

Most package are loaded at the beginning of the document to avoid clashes with other packages and make a correct loading order possible. Since the packages are only available at the end of the preamble, all commands defined by this package are also only available at the end of the preamble.

```
%%% ---- Packages ----
%%% Programming
\usepackage{etoolbox}
\usepackage{xstring}
\usepackage{kvsetkeys}
%%% Font packages
\ifdoctools@applyLayout
\usepackage{cmap}
\usepackage[T1]{fontenc} % T1 Schrift Encoding
%\usepackage{lmodern} % Font not loaded, because this can lead
%                     % to incompatibilities with other math fonts
\usepackage{textcomp}
\fi % end \ifdoctools@applyLayout
%%% listings (must be loaded before \AtBeginDocument)
\@ifpackageloaded{listings}{}{\RequirePackage{listings}}
%%% load all further packages and all commands
%%% at the beginning of the document and thus
%%% after all other packages
\PassOptionsToPackage{table}{xcolor}
\AtBeginDocument{%
%%
%%% Additional packages
\usepackage{xspace}
\@ifpackageloaded{xcolor}{}
 {\usepackage{xcolor}}
%%% listings
\colorlet{doc@stringcolor}{green!40!black!100}
\colorlet{doc@commentcolor}{green!50!black!100}
\colorlet{doc@numbercolor}{white!50!black!100}
\definecolor{doc@keywordcolor}{rgb}{0,0.35,1.0}
\colorlet{doc@demo@backcolor}{white}
\definecolor{doc@rulecolor}{rgb}{0.5,0.5,0.5}
%
\lstdefinestyle{lstStyleDefault}{
%%% appearance
   ,basicstyle=\small\ttfamily % Standardschrift
%%%  Space and placement
   ,floatplacement=tbp    % is used as float place specifier
   ,aboveskip=\medskipamount % define the space above and
   ,belowskip=\medskipamount % below displayed listings.
   ,lineskip=0pt          % specifies additional space between lines in
listings.
   ,boxpos=c              % c,b,t
%%% The printed range
   ,showlines=false       % prints empty lines at the end of listings
%%% characters
   ,extendedchars=true   % allows or prohibits extended characters
                         % in listings, that means (national)
                         % characters of codes 128-255.
   ,upquote=true         % determines printing of quotes
```

```
   ,tabsize=2,           % chars of tab
   ,showtabs=false       % do not show tabs
   ,showspaces=false     % do not show spaces
   ,showstringspaces=false % do not show blank spaces in string
%%% Line numbers
   ,numbers=none         % left, right, none
%%% Captions
   ,numberbychapter=true %
   ,captionpos=b         % t,b
   ,abovecaptionskip=\smallskipamount % the vertical space respectively above
   ,belowcaptionskip=\smallskipamount % or below each caption
%%% Margins and line shape
   ,linewidth=\linewidth % defines the base line width for listings.
   ,xleftmargin=0pt      % extra margins
   ,xrightmargin=0pt     %
   ,resetmargins=false   % indention from list environments like enumerate
                         % or itemize is reset, i.e. not used.
   ,breaklines=true      % line breaking of long lines.
   ,breakatwhitespace=false % allows line breaks only at white space.
   ,breakindent=0pt     % is the indention of the second, third, ...
                         % line of broken lines.
   ,breakautoindent=true % apply intendation
   ,columns=flexible     %
   ,keepspaces=true      %
}

\lstset{style=lstStyleDefault}

\lstdefinestyle{lstDocStyleBase}{
%%% base style
   ,style=lstStyleDefault
%%% appearance
   ,commentstyle=\slshape
%%% Line numbers
   ,numbers=left         % left, right, none
   ,stepnumber=1         % seperation between numbers
   ,numberfirstline=false % number first line always
   ,numberstyle=\tiny\color{doc@numbercolor}    % style of numbers
   ,numbersep=5pt        % distance to text
   ,numberblanklines=true %
%%% language
   ,language = [LaTeX]TeX
%%% commands
   % LaTeX programming
   ,moretexcs={setlength,usepackage,newcommand,renewcommand,providecommand,RequirePackage,SelectInputMappings,ifpdftex,ifpdfoutput,AtBeginEnvironment,ProvidesPackage},
   % other commands
   ,moretexcs={maketitle,text,includegraphics,chapter,section,subsection,
```

```
    subsubsection,paragraph,textmu,enquote,blockquote,ding,mathds,ifcsdef,
Bra,Ket,Braket,subcaption,lettrine,mdfsetup,captionof,listoffigures,
listoftables,tableofcontents,appendix}
  % tables
  ,moretexcs={newcolumntype,rowfont,taburowcolors,rowcolor,rowcolors,
bottomrule,
    toprule,midrule,}
  % hyperref
  ,moretexcs={hypersetup}
  % glossaries
  ,moretexcs={gls,printglossary,glsadd,newglossaryentry,newacronym}
  % Koma
  ,moretexcs={mainmatter,frontmatter,geometry,KOMAoptions,setkomafont,
addtokomafont}
  % SI, unit
  ,moretexcs={si,SI,sisetup,unit,unitfrac,micro}
  % biblatex package
  ,moretexcs={newblock,ExecuteBibliographyOptions,addbibresource}
  % math packages
  ,moretexcs={operatorname,frac,sfrac,dfrac,DeclareMathOperator,mathcal,
underset}
  % demo package
  ,moretexcs={democodefile,package,cs,command,env,DemoError,PrintDemo}
  % tablestyles
  ,moretexcs={theadstart,tbody,tsubheadstart,tsubhead,tend}
  % code section package
  ,moretexcs={DefineCodeSection,SetCodeSection,BeginCodeSection,
    EndCodeSection}
  % template tools package
  ,moretexcs={IfDefined,IfUndefined,IfElseDefined,IfElseUndefined,
IfMultDefined,IfNotDraft,IfNotDraftElse,IfDraft,IfPackageLoaded,
IfElsePackageLoaded,IfPackageNotLoaded,IfPackagesLoaded,IfPackagesNotLoaded
,ExecuteAfterPackage,ExecuteBeforePackage,IfTikzLibraryLoaded,
IfColumntypeDefined,IfColumntypesDefined,IfColorDefined,IfColorsDefined,
IfMathVersionDefined,SetTemplateDefinition,UseDefinition,IfFileExists,
iflanguage}
  % tablestyles
  ,moretexcs={setuptablefontsize,tablefontsize,setuptablestyle,tablestyle,
 setuptablecolor,tablecolor,disablealternatecolors,  tablealtcolored,
tbegin,tbody,tend,thead, theadstart,tsubheadstart,tsubhead,theadrow,
tsubheadrow,resettablestyle,theadend,tsubheadend,tableitemize,
PreserveBackslash}
  % todonotes
  ,moretexcs={todo,missingfigure}
  % listings
  ,moretexcs={lstloadlanguages,lstdefinestyle,lstset}
  % index
  ,moretexcs={indexsetup}
  % glossaries
  ,moretexcs={newglossarystyle,glossarystyle,deftranslation,newglossary}
```

```
   % tikz
   ,moretexcs={usetikzlibrary}
   % color
   ,moretexcs={definecolor,colorlet}
   % caption
   ,moretexcs={captionsetup,DeclareCaptionStyle}
   % floatrow
   ,moretexcs={floatsetup}
   % doc.sty
   ,moretexcs={EnableCrossrefs,DisableCrossrefs,PageIndex,CodelineIndex,
CodelineNumbered}
   % refereces
   ,moretexcs={cref,Cref,vref,eqnref,figref,tabref,secref,chapref}
}

\lstdefinestyle{lstDemoStyleLaTeXCode}{ %
%%% base style
   ,style=lstDocStyleBase
%%% Line numbers
   ,numbers=none         % left, right, none
%%% colors
   ,stringstyle=\color{doc@stringcolor}
   ,keywordstyle=\color{doc@keywordcolor}
   ,commentstyle=\color{doc@commentcolor}
   ,backgroundcolor=\color{doc@demo@backcolor}
   ,rulecolor=\color{doc@rulecolor}
%%% frame
   ,frame=single         % none, leftline, topline, bottomline, lines
                         % single, shadowbox
   ,framesep=3pt
   ,rulesep=2pt          % control the space between frame and listing
                         % and between double rules.
   ,framerule=0.4pt      % controls the width of the rules.
}

\colorlet{doc@code@backcolor}{gray!5}
\colorlet{doc@code@keywordcolor}{black}
\colorlet{doc@code@commentcolor}{black!60}

\lstdefinestyle{lstDocStyleLaTeXCode}{%
%%% base style
   ,style=lstDocStyleBase
%%% colors
   ,stringstyle=\color{doc@stringcolor}
   ,keywordstyle=\color{doc@code@keywordcolor} %
   ,commentstyle=\color{doc@code@commentcolor}
   ,backgroundcolor=\color{doc@code@backcolor}
   ,rulecolor=\color{doc@rulecolor}
%%% frame
   ,frame=none         % none, leftline, topline, bottomline, lines
```

```
                          % single, shadowbox
   ,framesep=3pt
   ,rulesep=2pt           % control the space between frame and listing
                          % and between double rules.
   ,framerule=0.4pt       % controls the width of the rules.
%%% numbers
   ,firstnumber=last
}

\lstloadlanguages{[LaTeX]TeX}
%%% hyperref
\ifdoctools@loadHyperref
\makeatletter
\@ifpackageloaded{hyperref}{}
   % load hyperref only if package
   % hypdoc is not loaded, which
   % loads hyperref itself
   {\usepackage[
     ,backref=page%
     ,pagebackref=false%
     ,hyperindex=true%
     ,hyperfootnotes=false%
     ,bookmarks=true%
     ,pdfpagelabels=true%
   ]{hyperref}}
\makeatother
%
\usepackage[]{bookmark}
%
\definecolor{pdfanchorcolor}{named}{black}
\definecolor{pdfmenucolor}{named}{red}
\definecolor{pdfruncolor}{named}{cyan}
\definecolor{pdfurlcolor}{rgb}{0,0,0.6}
\definecolor{pdffilecolor}{rgb}{0.7,0,0}
\definecolor{pdflinkcolor}{rgb}{0,0,0.6}
\definecolor{pdfcitecolor}{rgb}{0,0,0.6}
%
\hypersetup{
   ,draft=false, % all hypertext options are turned off
   ,final=true   % all hypertext options are turned on
   ,debug=false  % extra diagnostic messages are printed in the log file
   ,hypertexnames=true % use guessable names for links
   ,naturalnames=false % use LATEX-computed names for links
   ,setpagesize=true   % sets page size by special driver commands
   ,raiselinks=true    % forces commands to reflect the real height of the
link
   ,breaklinks=true    % Allows link text to break across lines
   ,pageanchor=true    % Determines whether every page is given an implicit
   ,plainpages=false   % Forces page anchors to be named by the arabic
```

```
   ,linktocpage=true   % make page number, not text, be link on TOC, LOF
and LOT
   ,colorlinks=true    % Colors the text of links and anchors.
   ,linkcolor  =pdflinkcolor   % Color for normal internal links.
   ,anchorcolor=pdfanchorcolor % Color for anchor text.
   ,citecolor  =pdfcitecolor   % Color for bibliographical citations in
text.
   ,filecolor  =pdffilecolor   % Color for URLs which open local files.
   ,menucolor  =pdfmenucolor   % Color for Acrobat menu items.
   ,runcolor   =pdfruncolor    % Color for run links (launch annotations).
   ,urlcolor   =pdfurlcolor    % color magenta Color for linked URLs.
   ,bookmarksopen=true     % If Acrobat bookmarks are requested, show them
   ,bookmarksopenlevel=2   % level (\maxdimen) to which bookmarks are open
   ,bookmarksnumbered=true %
   ,bookmarkstype=toc      %
   ,pdfpagemode=UseOutlines %
   ,pdfstartpage=1         % Determines on which page the PDF file is
opened.
   ,pdfstartview=FitH      % Set the startup page view
   ,pdfremotestartview=Fit % Set the startup page view of remote PDF files
   ,pdfcenterwindow=false  %
   ,pdffitwindow=false     % resize document window to fit document size
   ,pdfnewwindow=false     % make links that open another PDF file
   ,pdfdisplaydoctitle=true  % display document title instead of file name
} % end: hypersetup
%
\fi % end \ifdoctools@loadHyperref
%
```

color setup

```
%%% color setup
% table colors
\ifcsdef{colorlet}{%
\colorlet{tablebodycolor}{white!100}
\colorlet{tablerowcolor}{gray!10}
\colorlet{tableheadcolor}{gray!25}
}{}
%
```

Document layout / variables

```
%%% Set document layout / variables
\ifdoctools@applyLayout%
  \setlength{\parindent}{0pt}
  \setlength{\parskip}{0.5\baselineskip}
  \setcounter{secnumdepth}{2}
  \setcounter{tocdepth}{2}
```

```
\fi % end \ifdoctools@applyLayout%
%
```

2.3 Internal Variables

```
%%% ---- Internal Variables ----
%% Font for Index Headings
\newcommand{\doctools@indexHeadFont}[1]{\textsc{#1}}
%
```

2.4 Commands

`\bs` Shortcut for `\textbackslash`.

```
%%% ---- Commands ----
%% \bs
\newcommand{\bs}{\textbackslash}
%
```

`\command` Prints the argument with backslash in typewriter font. This is meant to be used for LaTeX commands. Additionally the command is added to the index.

```
%% \command
\newcommand{\command}[1]{%
\texttt{\textbackslash{}#1}\relax%
\ifdoctools@createIndexEntries%
  \index{\doctools@indexHeadFont{Command}!\textbackslash{}#1}%
\fi%
}%
%
```

`\cs` Shortcut for `\command`

```
%% \cs (shortcut for \command)
%% \cs might be defined by ltxdoc, therefore it needs to be deleted
%% before it can be redefined.
\ifcsdef{cs}{\csundef{cs}}{}%
\let\cs\command%
%
```

`\environment` Formats an environment in typewriter font and adds it to the index.

```
%% \environment
\ifcsdef{environment}{\csundef{environment}}{}
\newcommand{\environment}[1]{%
\texttt{#1}%
\ifdoctools@createIndexEntries
```

```
  \index{\doctools@indexHeadFont{Environment}!#1}%
\fi
}%
%
```

`\env` Shortcut for `\environment`

```
%% \env
\newcommand{\env}[1]{\environment{#1}}
%
```

`\package` Formats an package in typewriter font and adds it to the index.

```
%% \package
\newcommand{\package}[1]{%
\texttt{#1}%
\ifdoctools@createIndexEntries
\index{\doctools@indexHeadFont{Package}!#1}%
\fi
}%
%
```

`\ltxclass` Formats a LaTeX class in typewriter font and adds it to the index.

```
%% \ltxclass
\newcommand{\ltxclass}[1]{%
\texttt{#1}%
\ifdoctools@createIndexEntries
\index{\doctools@indexHeadFont{Class}!#1}%
\fi
}%
%
```

`\marg` Formats a mandatory argument of a command. If `\meta` is defined this is reused.

```
%% \marg
\ifcsdef{marg}{}{% if not defined
\ifcsdef{meta}%
  {\newcommand\marg[1]{\texttt{\{}\meta{#1}\texttt{\}}}}%
  {\newcommand\marg[1]{%
\texttt{\{}%
$\langle${\normalfont\slshape#1}$\rangle$%
\texttt{\}}}}%
}%
%
```

`\oarg` Formats a optional argument of a command. If `\meta` is defined this is reused.

```
%% \oarg
\ifcsdef{oarg}{}{% if not defined
\ifcsdef{meta}%
  {\newcommand\oarg[1]{\texttt{[}\meta{#1}\texttt{]}}}%
  {\newcommand\oarg[1]{%
\texttt{[}%
$\langle${\normalfont\slshape#1}$\rangle$%
\texttt{]}}}%
}%
%
```

`\arg` Formats an argument of a command without extra angle brackets in curled brackets in monospaced font.

```
%% \arg
\ifcsdef{arg}{%
  \csundef{arg}%
  \newcommand\arg[1]{\{\texttt{#1}\}}%
}{}%
%
```

`\option` Formats an option in typewriter font and adds it to the index.

```
%% \option
\newcommand{\option}[1]{%
\texttt{#1}%
\ifdoctools@createIndexEntries%
\index{option!#1}%
\fi
}%
%
```

`\parameter` Formats a parameter in typewriter font.

```
%% \parameter
\newcommand{\parameter}[1]{%
\texttt{#1}%
}%
%
```

`\person` Print out name of a person, for example for the acknowledge of their help during the development of some code.

```
%% \person
\newcommand{\person}[1]{\textsc{#1}}
%
```

`\AfterLastParam` Used in conjuction with `\DescribeMacro` and the printout of arguments using `\marg` or `\oarg`. `\AfterLastParam` is inserted at the end to begin a new line.

```
%% \AfterLastParam
\newcommand{\AfterLastParam}{\par\noindent}
%
```

`\Default` `\Default` is used to print out the default value of a command.

```
%% \Default
\newcommand{\Default}[1]{\par Default: \texttt{#1} \par}
%
```

`\Example` `\Example` is used to print out an example for the usage of a command.

```
%% \Example
\newcommand{\Example}[1]{\par Example: \texttt{#1} \par}
%
```

2.5 Option list

`Optionlist` Environment

```
\newenvironment{Optionlist}{%
\begin{flushleft}%
\vspace{-2\parskip}
%%  Style  changes
\small\renewcommand{\arraystretch}{1.4}%
%%  table
\begin{tabular} {>{\ttfamily}l<{\normalfont}p{0.7\textwidth}}%
}{% end of environment
\end{tabular}%
\end{flushleft}%
}%
%
```

2.6 LaTeX engine names

`\latex` New lower case command for printing out LaTeX

```
%%% ---- engine names ----
%% \latex
% ensure that space is added after \latex
\newcommand{\latex}{\LaTeX\xspace}
%
```

2.7 doc.sty modifications

The following commands from doc.sty are changed by introducing colors to the commands

```
%%% ---- doc.sty modifications ----
% define color for Macro and Environment names
\colorlet{doctools@ColorCodeNames}{blue!50!black}
%%% Overwrite font for \meta
\def\meta@font@select{\normalfont\slshape} % original: \itshape
%
\ifcsdef{PrintMacroName}
   {\def\PrintMacroName#1{\strut \MacroFont %
   \color{doctools@ColorCodeNames}\string #1\ }}{}
%
\newcounter{MacroName} % hyperref uses \theH<counter>
\providecommand*{\theHMacroName}{\theMacroName}
%
%% create label
%   \renewcommand*{\theHMacroName}{#1}%
%   \ifcsdef{phantomsection}{\phantomsection}{}%
%   \@bsphack%
%   \refstepcounter{MacroName}%
%   \label{doc:desc:#1}%
\ifcsdef{PrintDescribeMacro}
   {\def\PrintDescribeMacro#1{%
   \strut \MacroFont %
   \color{doctools@ColorCodeNames} \string #1\ }}{}
\ifcsdef{PrintDescribeEnv}
   {\def\PrintDescribeEnv#1{\strut \MacroFont %
   \color{doctools@ColorCodeNames} #1\ }}{}
\ifcsdef{PrintEnvName}
   {\def\PrintEnvName#1{\strut \MacroFont %
   \color{doctools@ColorCodeNames} #1\ }}{}
%
%%% disable the index preamble if index entries are generated by this
package
\ifdoctools@createIndexEntries
\ifcsdef{index@prologue}
     {\def\index@prologue{\section*{Index}\markboth{Index}{Index}}}
     {}
\fi
%
%
```

2.8 Listings environments

```
%%% ---- listings environments for LaTeX code
```

```
%%% Overwriting the environment macrocode for printing the code in a dtx
file.
\ifcsdef{macrocode}{\csundef{macrocode}}{}
\lstnewenvironment{macrocode}{\lstset{style=lstDocStyleLaTeXCode}}{}
%
```

```
%%% environment code examples.
\lstnewenvironment{latexcode}{\lstset{style=lstDemoStyleLaTeXCode}}{}
%
```

2.9 Print LaTeX code from external file using listings

```
% ------------------------------------------------
% \printCodeFromFile
% ------------------------------------------------
\newcounter{lstFirstLine}
\newcounter{lstLastLine}
\setcounter{lstLastLine}{0}
\setcounter{lstFirstLine}{0}
%
\newcommand{\printCodeFromFile}[3][]{%
\ifstrempty{#1}{}{%
  \setcounter{lstFirstLine}{#1}%
}%
\setcounter{lstLastLine}{#2}%
%
\lstinputlisting[%
  firstnumber=\thelstFirstLine,%
  firstline=\thelstFirstLine,%
  lastline=\thelstLastLine,%
  nolol=true,
  style=lstDemoStyleLaTeXCode]%
  {#3}%
%
%% set counter to lastLine + 1
\setcounter{lstFirstLine}{\thelstLastLine}
\addtocounter{lstFirstLine}{1}
}
%
```

2.10 References and links to files

```
% ------------------------------------------------
% \file and \labelfile
% ------------------------------------------------
% Code copied from http://tex.stackexchange.com/
%  questions/65639/how-to-create-my-on-ref-label-system/
% with later modifications.
% Thanks to Heiko Oberdiek for providing this answer !
```

```
% ------------------------------------------------
%% Command for printing the filename
\ifcsdef{urlstyle}{}{\usepackage{url}}
\DeclareUrlCommand{\PrintFileName}{\urlstyle{tt}}
%%
\newcounter{file}
%% hyperref uses \theH<counter>
\providecommand*{\theHfile}{\thefile}
%% code from tex.stackexchange with the
%% the help from Heiko Oberdiek.
\newcommand*{\labelfile}[1]{%
  %% convert all "/" to "/," (comma list) and save in \IndexFileA
  \StrSubstitute{#1}{/}{/,}[\IndexFileA]%
  %% define \IndexFileB as empty (used for the output string)
  \let\IndexFileB\@empty
  %% parse and convert \IndexFileA using \@AddFileEntry
  \expandafter
  \comma@parse@normalized\expandafter{\IndexFileA}\@AddFileEntry
  %% create label and print to index
  \@bsphack
  \renewcommand*{\theHfile}{#1}%
  \refstepcounter{file}%
  \ifcsdef{phantomsection}{\phantomsection}{}
  \label{file:#1}%
  \ifdoctools@createIndexEntries
  \index{\textsc{Files}!\IndexFileB}%
  \fi
  \@esphack
}
%% add entries of comma list to the index.
\newcommand*{\@AddFileEntry}[1]{%
  \ifx\IndexFileB\@empty
    %% add first entry to index with lexEntry@{printEntry}
    \def\IndexFileB{#1@#1}%
  \else
    \expandafter\def\expandafter\IndexFileB\expandafter{%
    %% add following entries to index with
    %% previousEntries!lexEntry@{printEntry}
\IndexFileB!%
#1@#1%
    }%
  \fi
}
%% Print out filename and create a link if the label exists.
\newcommand*{\file}[1]{%
  \ifcsdef{hyperref}%
    {\hyperref[file:#1]{\PrintFileName{#1}}}%
    {\PrintFileName{#1}}%
}%
%%
```

```
} % end of \AtBeginDocument

%
```

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the definition; numbers in roman refer to the pages where the entry is used.