The \LaTeX\ \texttt{dtxdescribe} Package

v1.04 — 2022/02/01

© 2016–2022 Brian Dunn
bd@BDTechConcepts.com

Describe additional object types in \texttt{dtx} source files.

Abstract

The \texttt{doc} package includes tools for describing macros and environments in \LaTeX\ source \texttt{.dtex} format. The \texttt{dtxdescribe} package adds additional tools for describing booleans, lengths, counters, hooks, keys, packages, classes, options, files, commands, arguments, and other objects. \texttt{dtxdescribe} also works with the regular document classes, for those who do not wish to use the \texttt{ltxdoc} class and \texttt{.dtex} files.

Each described item is given a margin tag similar to \texttt{\DescribeEnv}, and is listed in the index by itself and also by category. Each item may be sorted further by an optional class. All index entries except code lines are hyperlinked.

The \texttt{dtxexample} environment is provided for typesetting example code and its results. Contents are displayed verbatim along with a caption and cross-referencing. They are then \texttt{\input} and executed, and the result is shown.

Environments are also provided for displaying verbatim or formatted source code, user-interface displays, and sidebars with titles.

Macros are provided for formatting the names of inline \LaTeX\ objects such as packages and booleans, as well as program and file names, file types, internet objects, the names of certain programs, a number of logos, and inline dashes and slashes.

\texttt{dtxdescribe} works with the \texttt{ltxdoc} class, but also works with the standard classes as well, except that the \texttt{macro} and \texttt{environment} environments are not supported. Either \texttt{makeidx} or \texttt{splitidx} may be loaded by the user. \texttt{makeidx} will be used by default.

\texttt{dtxdescribe} works with \texttt{pdflatex}, \texttt{XeLaTeX}, and \texttt{LuaLaTeX}, and perhaps other engines as well.
Contents

1 Introduction 5

2 Using dtxdescribe 6

3 The macros, and the dtxexample environment 7
   3.1 Macros and environments .......................... 7
   3.2 Arguments ........................................ 7
   3.3 Booleans, lengths, counters, hooks, keys ........ 8
   3.4 Packages, classes, options ........................ 8
   3.5 Files, programs, commands ....................... 9
   3.6 Other source objects ............................ 9
   3.7 In a description environment ..................... 9
   3.8 Defaults .......................................... 10
   3.9 Nesting ............................................ 10
   3.10 \margintag, \watchout .......................... 11
   3.11 dtxexample environment .......................... 11
   3.12 noindmacro and noindenvironment environments ... 11
   3.13 sourceverb, sourcedisplay, UIdisplay, docsidebar .... 12
   3.14 Formatted objects ............................... 13
      3.14.1 \LaTeX{} objects ............................ 13
      3.14.2 Programs and commands ..................... 13
      3.14.3 File types ................................. 14
      3.14.4 Internet .................................... 14
      3.14.5 Specific programs .......................... 14
      3.14.6 Acronyms, brand names, trademarks .......... 15
   3.15 Logos ........................................... 15
   3.16 Dashes and slashes .............................. 15
4 Examples

5 Usage notes

6 Code

6.1 Required packages .................................. 31
6.2 Gobbling comment characters ....................... 33
6.3 Vertical spacing .................................... 33
6.4 \texttt{dtxdoc} emulation .............................. 34
6.5 \texttt{doc} emulation ................................ 34
6.6 Support macros .................................... 35
6.7 \texttt{\textbackslash DescribeMacro} and \texttt{\textbackslash DescribeEnvironment} ..... 39
6.8 New \texttt{\textbackslash Describe...} macros .......... 40
6.9 \texttt{\textbackslash DescribeDefault} .................... 45
6.10 \texttt{\textbackslash ItemDescribeMacro}, etc. ....... 45
6.11 \texttt{\textbackslash margintag}, \texttt{\textbackslash watchout} ...... 48
6.12 Nesting ........................................... 49
6.13 The \texttt{dtxexample} environment ................ 50
6.14 \texttt{noindmacro} and \texttt{noindentenvironment} ... 52
6.15 \texttt{sourcedisplay}, \texttt{\textbackslash UIdisplay}, \texttt{docsidebar} .......... 52
6.16 Formatted objects ................................ 54
6.16.1 \texttt{\LaTeX} objects ............................. 54
6.16.2 Programs and commands ......................... 55
6.16.3 File types ..................................... 56
6.16.4 Internet ....................................... 57
6.16.5 Specific programs ............................... 58
6.16.6 Acronyms, brand names, trademarks ............ 59
6.17 Logos ............................................. 59
6.18 Dashes and slashes ................................ 60
7 Compiling dtxdescribe

Change History and Index

List of Examples

1 Macros . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17
2 Environment . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18
3 Second Environment . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18
4 Booleans and Counters . . . . . . . . . . . . . . . . . . . . . . . . . . . 19
5 Lengths . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 19
6 Packages, Classes, and Options . . . . . . . . . . . . . . . . . . . . . . 20
7 Files, Commands, and Programs . . . . . . . . . . . . . . . . . . . . . . 20
8 Keys . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 21
9 Arguments . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22
10 Object . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23
11 Other . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23
12 Description environments . . . . . . . . . . . . . . . . . . . . . . . . . 24
13 Nesting . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
14 dtxexample . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 26
15 fsourceverb . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 27
16 sourcedisplay . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 27
17 UIDisplay . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 28
18 docsidebar . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 29

List of Figures

1 A Figure . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 26
1 Introduction

The doc package provides \DescribeMacro and \DescribeEnv to help document new macros and environments. Each generates a heading in the documentation, to which \marg, \oarg, and \parg may be added to identify arguments to be passed to the new object. Their names are added to the margin, and index entries are added, as well as group of entries for environments.

dtxdescribe extends this concept to include a number of additional objects, such as booleans and keys. To help identify what is being described in the margin, small tags are added to the name, such as “Env”, “Bool”, or “Key”. These new objects are also listed in the index with the same tag shown after their names, and also by group. Optional classes may be used to further categories index entries.

Modifications have been made to interact with hyperref to provide hyper links for regular index entries as well as the new \Describe entries.

Additional macros are provided to generate colored margin tags and warnings, and a new dtxexample environment demonstrates code examples.

This documentation and its index show examples of these macros in use.

While the index may appear to be overkill for a small package, keep in mind that it includes a number of fictional entries from the examples. Extensive cross-referencing can be useful for larger works. And, of course, you need not cross-reference everything!
2 Using dtxdescribe

Place \usepackage{dtxdescribe} in the .dtx file's driver section:

```latex
\%<*driver>
\documentclass{ltxdoc}
...
\usepackage{lmodern}
...
\usepackage{dtxdescribe}
...
\usepackage[packagename] % the name of your new package
\usepackage{hyperref}
\usepackage[cleveref]
...
\%</driver>
```

Various objects inside the dtx file may be described with new macros such as \DescribeBoolean, \DescribeLength, \DescribeCounter, similar to the already-familiar \DescribeMacro and \DescribeEnv.

Optional “classes” may be assigned to the objects being described, including the new versions of \DescribeMacro and \DescribeEnv. These classes are printed in the margin tag and index entry for each item, and also generate additional index entries sorted by class. This is especially useful for key/value sets, where several sets may appear in the same document.

- inside a float: The margin tag is not printed if the \Describe macros are used inside a float such as a table, but the index entries are still made.
- \margintag[text]: \margintag{text} may be used to place a colored tag in the margin to summarize paragraph contents or draw attention to an index destination.
- \watchout[optional text]: \watchout[optional text] may be used to place a red warning sign in the margin, along with optional text.

The dtxexample environment may be used to typeset and execute small pieces of \LaTeX code as examples of its use. Optional cross-referencing notes may be used to refer to any example float being generated.
3 The macros, and the dtexample environment

3.1 Macros and environments

These are only provided by the \texttt{bdoc} class and \texttt{doc} package to document a .\texttt{dt} file, where comments are used by \texttt{docstrip} to disable these environments in the resulting .\texttt{sty} file. When using the regular document classes, the macro and environment environments would localize any definitions, and \texttt{DescribeMacro} and \texttt{DescribeEnv} should be used instead.

\texttt{\DescribeMacro \[\langle class\rangle\] \{\langle name\rangle\}}

The preexisting macro from the \texttt{doc} package is redefined to create hyperlinked index entries, and include an optional class. A margin tag is created and an index entry is made. When the optional class is used, it is displayed in front of the margin tag, and is used to group an index entry by macro name and another index entry by class. An example would be to describe the float creation and caption setup for a new class of float, such as the \texttt{dtexample} float and the example “photograph” float both found in the index for this document. See example 1 on page 17 for examples.

\texttt{\DescribeEnv \[\langle class\rangle\] \{\langle environment name\rangle\}}

The preexisting macro from the \texttt{doc} package is redefined to create hyperlinked index entries, include an optional class, and also to place an ‘Env’ tag in front of the name in the margin. See example 2 on page 18.

3.2 Arguments

The \texttt{\Describe.\ldots} macros may be followed by \texttt{\marg}, \texttt{\oarg}, and \texttt{\parg} to describe arguments passed to the macros.

\texttt{\marg \{(\langle text\rangle)\}}

Shows a mandatory argument for a macro or environment.

The results looks like \{(\texttt{mandatory})\}.

\texttt{\oarg \{(\langle text\rangle)\}}

Shows an optional argument for a macro or environment.

The results looks like \{(\texttt{optional})\}.

\texttt{\parg \{(\langle text\rangle)\}}

Used for “picture” arguments, such as coordinates.

The result looks like \{(\texttt{coordinate})\}.
May be used to describe actions taken when given certain macro arguments. These will be given an ‘Arg’ margin tag and will appear in the index. The class may be used to categorize arguments by their macro or environment name. See example 9 on page 22.

3.3 Booleans, lengths, counters, hooks, keys

See example 4 on page 19.

\DescribeBoolean \[⟨class⟩\] {⟨name⟩}

Describes a boolean. Given a ‘Bool’ tag in the margin and index.

\DescribeLength \[⟨class⟩\] {⟨name⟩}

Describes a length. Given a ‘Len’ tag in the margin and index.

\DescribeCounter \[⟨class⟩\] {⟨name⟩}

Describes a counter. Given a ‘Ctr’ tag in the margin and index.

\DescribeHook \[⟨class⟩\] {⟨name⟩}

Describes a hook. Given a ‘Hook’ tag in the margin and index. The class may be used to categorize hooks by package. Example:

\DescribeHook[LaTeX]{para/begin}

\DescribeKey \[⟨class⟩\] {⟨name⟩}

Describes a key. Given a ‘Key’ tag in the margin and index. The class may be used to categorize keys by their kev/value group. See example 8 on page 21.

3.4 Packages, classes, options

\DescribePackage \[⟨class⟩\] {⟨name⟩}

Describes a package. Given a ‘Pkg’ tag in the margin and index.

\DescribeClass \[⟨class⟩\] {⟨name⟩}

Describes a \LaTeX{} class. Given a ‘Cls’ tag in the margin and index.

\DescribeOption \[⟨class⟩\] {⟨name⟩}

Describes a \LaTeX{} package or class option. Given an ‘Opt’ tag in the margin and index.
3.5 Files, programs, commands

\DescribeFile\langle class\rangle \langle name\rangle

Describes an operating-system file. Given a 'File' tag in the margin and index. The filename may have underscores.

\DescribeProgram\langle class\rangle \langle name\rangle

Describes an operating-system program. Given a 'Prog' tag in the margin and index. The program name may have underscores.

\DescribeCommand\langle class\rangle \langle name\rangle

Describes an operating-system command. Given a 'Cmd' tag in the margin and index. The command name may have underscores.

3.6 Other source objects

\DescribeObject\langle class\rangle \langle name\rangle

Describes an arbitrary programming object, such as a color definition or caption setup. A margin tag and index entry are created with \texttt{family} type. When a class is used, it is pre-pended to the margin tag, appended to the index entry, and a second index entry is created grouped by class. If a macro name is to be described, use \DescribeMacro instead. See example 10 on page 23.

\DescribeOther\langle class\rangle \langle name\rangle

Describes an arbitrary non-programming object, such as a license agreement or credits. A margin tag and index entry are created in roman type. When a class is used, it is pre-pended to the margin tag, appended to the index entry, and a second index entry is created grouped by class. See example 11 on page 23.

3.7 In a description environment

To describe an object using a description environment, use the following. See example 12 on page 24.

\ItemDescribeMacro\langle class\rangle \langle name\rangle A description.

\ItemDescribeEnv\langle class\rangle \langle name\rangle A description.

\ItemDescribeArgument\langle class\rangle \langle argument\rangle A description.

\ItemDescribeBoolean\langle class\rangle \langle name\rangle A description.

\ItemDescribeLength\langle class\rangle \langle name\rangle A description.
3.8 Defaults

\DescribeDefault[\textit{value}]

Shows the default value of a \Describe... item, such as displayed here. Place this macro immediately after the \Describe... macro and any arguments, but before the text description.

\DescribeDefaultcolor[\textcolor{green!50!black}]

The color of the margin tag used to show the default value. This is used by \textcolor to create the margin tag.

3.9 Nesting

\shownesting[\textit{fraction of \linewidth}][\textit{container name}][\textit{contents}]

It may be useful to show which objects contain which other objects. \shownesting shows a box enclosing a name for the container, and the container’s contents. \shownesting be nested, showing boxes inside other boxes, which displays how each environment and macro is fit together inside each other.

The optional argument is the fraction of \linewidth to use for the box, from [0] to [1]. The default is [1]. Each \shownesting starts its own paragraph, unless the star * is used, in which case the \shownesting* appears inline with previous text. To place two \shownesting boxes side-by-side, use optional arguments to specify less than full \linewidth for each box, and use \shownesting* for the second box to place it inline.

See example 13 on page 25 for an example.
3.10 \margintag, \watchout

\margintag

{⟨text⟩}

Creates a colored margin tag. May be used to identify the topic of a paragraph or the destination of an arbitrary index entry.

\margintagcolor

Default: blue!70!black

\watchout

⚠️ \watchout[example]

The color of the \margintag.

\watchoutcolor

Default: red!50!black

3.11 dtxexample environment

* [{⟨Notes/cross-references⟩}] {⟨caption & label⟩}

The dtxexample environment is useful for demonstrating a piece of \LaTeX code. The example is a simulated float with its own caption and optional label, along with optional notes and/or cross-referencing commands. The contents of the dtxexample environment are printed verbatim, then loaded and executed as \LaTeX code, showing the results just below the printed code. In the case of float commands, the floats are generated as expected somewhere nearby, and should be given their own labels. References to the float's labels may be placed in the optional argument to the dtxexample environment, and will be printed below the code.

The unstarred version places the code inside a minipage, forbidding a page break in the middle of the code listing. The starred version does not use a minipage. This is required when the code is too large to fit on a single page.

See example 14 for a demonstration of how dtxexample works.

\dtxexamplecodename

The text name of the code section.

\dtxxmplersresultname

The text name of the result section.

3.12 noindmacro and noindentenvironment environments

⚠️ .dtx only

These are like macro and environment, but not indexed. These only make sense if using the ltxdoc class and doc package to document a .dtx file, where comments are used by docstrip to disable these environments in the resulting .sty file. When using the regular document classes, noindmacro and noindentenvironment environments should not be used, as they would localize any definitions. \DescribeMacro and \DescribeEnv should be used instead.
\begin{macro}{\macroname} \oarg{optional} \marg{mandatory} 
...
\end{macro}

with

\begin{noindmacro}{\macroname} \oarg{optional} \marg{mandatory} 
...
\end{noindmacro}

and similarly for noindenvironment.

\section*{sourceverb, sourcedisplay, UIDisplay, docsidebar}

\begin{itemize}
\item \textbf{sourceverb}
\begin{description}
\item[Default:] \texttt{gobble=2,\tabsize=4,\xleftmargin=2em}
\end{description}
Displays source code verbatim. Uses optional fancyverb keys. Includes \texttt{gobble=2} to absorb the leading \texttt{%} and space character of a dtx file source format. Because this is a verbatim environment, it \textit{cannot} be used inside a macro.

\item \textbf{fsourceverb}
\begin{description}
\item[Default:] \texttt{gobble=2,\tabsize=4,\xleftmargin=2em,frame=lines}
\end{description}
Displays source code verbatim inside a frame. A label may be included using the \texttt{label} key. Because this is a verbatim environment, it \textit{cannot} be used inside a macro. See example 15 on page 27.

\item \textbf{sourcedisplay}
Display source code with manual formatting. \texttt{\textcolor}, \texttt{\textbf}, and \texttt{\emph} may be used to highlight text. Macros must be escaped with \texttt{\cs}, characters such as \texttt{\{} must be produced with \texttt{\{\}, etc. \texttt{\\} must be used to force a new line. \texttt{\fquad}, \texttt{\fqqquad}, and \\texttt{\fqqqqquad} may be used to force indenting. Because this is \textit{not} a verbatim environment, it \textit{can} be used inside a macro. See example 16 on page 27.

\item \texttt{\fquad}
Single-level indent inside a sourcedisplay.

\item \texttt{\fqquad}
Double-level indent inside a sourcedisplay.

\item \texttt{\fqqquad}
Triple-level indent inside a sourcedisplay.

\item \textbf{UIDisplay}
Displays a user interface, such as a dialog box entry or a menu selection. See example 17 on page 28. Also see the \texttt{\UI} macro.

\item \textbf{userentry}
\begin{description}
\item[Default:] \texttt{\textbf{Enter ⇒}}
\end{description}
Typeset something for the user to enter. Also see the \texttt{\cmds\userentry} macro.

\item \textbf{userentryname}
Text to tell the user to enter the following item. Change with \texttt{\renewcommand}.
Env docsiderbar

\[\langle title \rangle\]
Creates a sidebar within the document. See example 18 on page 29.

3.14 Formatted objects

Macros to format references to various kinds of objects.

This dtxdescribe package documentation uses erewhon, cabin, and inconsolata, along with metalogox, to demonstrate the following font effects.

3.14.1 \LaTeX objects

\pkg{\langle packagename \rangle}
Prints as packagename. Also for a classname.

\cs{\langle csname \rangle}
Prints as \csname.

\env{\langle environment \rangle}
Prints as environment.

\marg{\langle argument \rangle}
Prints (\langle arg \rangle). Mandatory argument.

\oarg{\langle argument \rangle}
Prints [\langle arg \rangle]. Optional argument.

\parg{\langle argument \rangle}
Prints (\langle arg \rangle). Picture-mode argument.

\ctr{\langle counter \rangle}
Prints as counter.

\bool{\langle boolean \rangle}
Prints as boolean.

\optn{\langle option \rangle}
Prints as option, for example to a macro, package, class.

\TOC
TOC: Table of contents.

\LOF
LOF: List of figures.

\LOT
LOT: List of tables.

3.14.2 Programs and commands

\progcode
Prints as inline program code: Escape underscores and other special characters such as [, %, $.

\prog
Prints as grep, make: A program name. Underscores allowed.

\filenm
Prints as file_name: Underscores allowed.

\UI
Prints as General user-interface text. What the user sees on the display. Also see the UIDisplay environment.

\cmds
Prints as Commands to be entered: What the user enters. Escape underscores and other special characters such as [, %, $. Also see the \userentry macro.
3.14.3 File types

\ODT  
ODT OpenDocument Format word processing document

\SVG  
SVG image format

\PNG  
PNG image format

\GIF  
GIF image format

\JPG  
JPG image format

\EPS  
EPS image format

\PDF  
PDF image format

\DVI  
DVI image format

3.14.4 Internet

\UTF  
UTF: Unicode

\URL  
URL: Uniform Resource Locator

\element  
\{\langle element name\rangle\}  Prints as <element>, an HTML/CSS element

\attribute  
\{\langle attribute name\rangle\}  Prints as attribute, an HTML/CSS attribute. pdf\TeX and \LaTeX\ only. Not for Lua\TeX.

\attrib  
\{\langle attribute name\rangle\}  Prints as attribute, an HTML/CSS attribute. pdf\TeX, \LaTeX, or Lua\TeX.

\HTML  
HTML: Hypertext Markup Language

\HTMLfive  
HTML5: Old-style figure if font supports

\CSS  
CSS: Cascading Style Sheet

\CSSthree  
css3: Old-style figure if font supports

\EPUB  
EPUB: E-book file format

3.14.5 Specific programs

\tikz  
Tikz: Package logo

\CTAN  
CTAN: Comprehensive \TeX Archive Network

\TDS  
TDS: \TeX Directory Structure

\MathML  
MathML: Mathematical Markup Language
\MathJax: Math on the web.

### 3.14.6 Acronyms, brand names, trademarks

\brand \{\langle\text{name}\rangle\}\ brandname, company name
\acro \{\langle\text{acronym}\rangle\}\ acro: acronym
\supregistered Superscript trademark symbol®

### 3.15 Logos

Several additional logos are provided. Also see the metalogo and metalogox packages.

\dviTeX dvi \TeX
\dviLaTeX dvi \LaTeX
\pdfTeX pdf \TeX
\pdfLaTeX pdf \LaTeX
\LuaTeX Lua \TeX
\LuaLaTeX Lua \LaTeX
\XeTeX xe \TeX, with reversed E if graphics is loaded.
\XeLaTeX xe \LaTeX, with reversed E if graphics is loaded.
\AmS \AmS
\LyX LyX
\BibTeX Bib \TeX
\MakeIndex \MakeIndex
\ConTeXt Con \TeX
\MiKTeX MiK \TeX

### 3.16 Dashes and slashes

\thinskips A breakable thin skip.
\endash An endash: –
An emdash: —

A thin space which allows a line break.

A very thin space which allows a line break.

An unbreakeable thin space, emdash, and breakable thin space: A—B

An unbreakeable thin space, endash, and breakable thin space: A–B

An unbreakable very thin space, a slash, and a breakable very thin space:

<table>
<thead>
<tr>
<th>Command</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>A--B</td>
<td>A–B (not breakable)</td>
</tr>
<tr>
<td>A \dash B</td>
<td>A–B (only breakable before the B)</td>
</tr>
<tr>
<td>A -- B</td>
<td>A–B (breakable before or after the dash)</td>
</tr>
<tr>
<td>A----B</td>
<td>A—B (not breakable)</td>
</tr>
<tr>
<td>A \Dash B</td>
<td>A—B (only breakable before the B)</td>
</tr>
<tr>
<td>A --- B</td>
<td>A—B (breakable before or after the dash)</td>
</tr>
<tr>
<td>A/B</td>
<td>A/B (not breakable)</td>
</tr>
<tr>
<td>A \Slash B</td>
<td>A/B (only breakable before the B)</td>
</tr>
<tr>
<td>A / B</td>
<td>A/B (breakable before or after the slash)</td>
</tr>
<tr>
<td>A~/~B</td>
<td>A/B (not breakable)</td>
</tr>
</tbody>
</table>
4 Examples

Example 1: Macros

Code:

\DescribeMacro{\mymacro}{\oarg{optional}}{\marg{mandatory}}
A typical macro definition.

\DescribeMacro[photograph]{\DeclareFloatingEnvironment}
Create a photograph float. \bigskip

\DescribeMacro[photograph]{\captionsetup}
Caption settings for a photograph float.

\DescribeMacro[photograph]{\cnameref}
\pkg{cleveref} name for the photograph float.

Result:

\mymacro
\[⟨\text{optional}⟩\] {⟨\text{mandatory}⟩} A typical macro definition.

\[\text{Create a photograph float.}\]

\DeclareFloatingEnvironment
\captionsetup
\cnameref
\pkg{cleveref} name for the photograph float.

The optional class is used to label and group tags and index entries. See this document’s index entries for examples of this “photograph” class and the \dtxexample class of macros.

The re-defined \DescribeMacro, \DescribeEnv, and all the following macros create hyperlinked index entries, along with regular uses of \index.
Example 2: Environment

Code:
\DescribeEnv{myenvironment} \marg{argument} Short description.

Result:
\{⟨argument⟩\} Short description.

The re-defined \DescribeEnv adds an ‘Env’ tag to the margin, and adds “(environment)” to its own index entry. Note that environments and all the other new objects defined by this package each receives two index entries, one by name, and one grouped with others of its kind.

Example 2 shows descriptive text on the same line as the \DescribeEnvironment. For macros and environments with many arguments after the name, it may be better to place any additional text in a following paragraph.

Example 3: Second Environment

Code:
\DescribeEnv[kindofenvironment]{otherenvironment} \oarg{opt args} \parg{coordinates} A description.

Result:
[⟨opt args⟩] ⟨coordinates⟩ A description.

The otherenvironment will be indexed by itself and also with myenvironment under the index entry “environments”, and also under the class kindofenvironment.
Example 4: Booleans and Counters

Code:

\ DescribeBoolean[examples]{sampleboolean} Some description.
\ DescribeCounter[examples]{samplecounter} Some description.

Result:

Some description.

Some description.

Most of the new \ Describe macros behave like the new \ DescribeEnv, placing a tag in the margin, an index entry by name, and another index entry by group.

Example 5: Lengths

Code:

\ DescribeLength[photograph]{\ photowidth} Some description.

Result:

Some description.

Lengths have a leading backslash, but are otherwise described the same as the rest of the objects.
Example 6: Packages, Classes, and Options

Code:

\DescribePackage[examples]{samplepackage}
   About a \LaTeX\ package.

\DescribeClass[examples]{sample_class}
   About a \LaTeX\ class.

\DescribeOption[examples]{sampleoption}
   About an option for a package or class.

Result:

Pkg [examples] samplepackage
   About a \LaTeX\ package.

Cls [examples] sample_class
   About a \LaTeX\ class.

Opt [examples] sampleoption
   About an option for a package or class.

Example 7: Files, Commands, and Programs

Code:

\DescribeFile[bigfiles]{really_big_file.txt} Some description.

\DescribeFile[bigfiles]{another_big_file.txt} Some description.

\DescribeFile{lone_file.txt} Some description.

\DescribeCommand{OS_command} An operating-system command.

\DescribeProgram{program_name} An operating-system program.

Result:

File  [bigfiles] really_big_file.txt
   Some description.

File  [bigfiles] another_big_file.txt
   Some description.

File  lone_file.txt
   Some description.

Cmd   OS_command
   An operating-system command.

Prog  program_name
   An operating-system program.

Filenames, program names, and command names may have underscores, such as tested here. A class is used to group “bigfiles” together in the index.
Example 8: Keys

Code:

\DescribeKey[groupofkeys]{firstkey} About the first key of the \textit{groupofkeys} set.

\DescribeKey[groupofkeys]{secondkey} About the second key of \textit{groupofkeys}.

\DescribeKey[examples]{samplekey} About some key of \textit{otherkeys}.

\DescribeKey[examples]{sampl ethrekey} About another key of \textit{otherkeys}.

\DescribeKey{lonekey} A key without a class.

Result:

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>firstkey</td>
<td>About the first key of the groupofkeys set.</td>
</tr>
<tr>
<td>secondkey</td>
<td>About the second key of groupofkeys.</td>
</tr>
<tr>
<td>samplekey</td>
<td>About some key of otherkeys.</td>
</tr>
<tr>
<td>sampl ethrekey</td>
<td>About another key of otherkeys.</td>
</tr>
<tr>
<td>lonekey</td>
<td>A key without a class.</td>
</tr>
</tbody>
</table>

See the index key groups.
Example 9: Arguments

Code:

\DescribeArgument[figure]{[H]}
What happens when a figure is [H]ere.

\DescribeArgument[figure]{[M]}
What happens when a figure is in the [M]argin.

\DescribeArgument[\cs{mymacro}]{bold}
What happens when \cs{mymacro} is given the |bold| argument.

Result:

| Arg[figure] | [H] | What happens when a figure is [H]ere. |
| Arg[figure] | [M] | What happens when a figure is in the [M]argin. |
| Arg[\mymacro] | bold | What happens when \mymacro is given the bold argument. |

Arguments behave like keys, and may have an optional class to identify their macro or environment, and group their entries in the index.

⚠️ **macro names** Note the need to use \cs{mymacro} for the macro’s name.
Example 10: Object

**Code:**

\DescribeObject[color]{somecolor}
  The color of something.

\DescribeObject[color]{othercolor}
  The other color.

\DescribeObject{randomobject} About some random object.

**Result:**

<table>
<thead>
<tr>
<th>[color] somecolor</th>
<th>The color of something.</th>
</tr>
</thead>
<tbody>
<tr>
<td>[color] othercolor</td>
<td>The other color.</td>
</tr>
<tr>
<td>randomobject</td>
<td>About some random object.</td>
</tr>
</tbody>
</table>

Describes an arbitrary programming object, using `ttfamily` text.

Example 11: Other

**Code:**

\DescribeOther{license agreement}
The following is the fictional license agreement:

\DescribeOther{Before \emph{myenvironment}}
  Actions to be done `\cs{BeforeBeginEnvironment}`.

\DescribeOther[otherclass]{Other Item} About the other item.
\DescribeOther[otherclass]{Additional Item} About the add'l item.

**Result:**

<table>
<thead>
<tr>
<th>license agreement</th>
<th>The following is the fictional license agreement:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before myenvironment</td>
<td>Actions to be done <code>\cs{BeforeBeginEnvironment}</code>.</td>
</tr>
<tr>
<td>Other Item</td>
<td>About the other item.</td>
</tr>
<tr>
<td>Additional Item</td>
<td>About the add'l item.</td>
</tr>
</tbody>
</table>

Describes an arbitrary non-programming object, using roman text.
**Example 12: Description environments**

*Code:*

\begin{description}
\ItemDescribeMacro{\macroname} Describe the macro.
\ItemDescribeBoolean{booleanname} Describe the boolean.
\ItemDescribeLength{\lengthname} Describe the length.
\ItemDescribeKey{keyname} Describe the key.
\ItemDescribePackage{package_name} Describe the package.
\ItemDescribeClass{class_name} Describe the class.
\ItemDescribeFile{file_name} Describe the file.
\ItemDescribeProgram{program_name} Describe the program.
\end{description}

*Result:*

<table>
<thead>
<tr>
<th>[descexamples] \macroname</th>
<th>\macroname: Describe the macro.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bool [descexamples]</td>
<td>booleanname: Describe the boolean.</td>
</tr>
<tr>
<td>Len [descexamples]</td>
<td>\lengthname: Describe the length.</td>
</tr>
<tr>
<td>Key [descexamples]</td>
<td>keyname: Describe the key.</td>
</tr>
<tr>
<td>Pkg [descexamples]</td>
<td>package_name: Describe the package.</td>
</tr>
<tr>
<td>Cls [descexamples]</td>
<td>class_name: Describe the class.</td>
</tr>
<tr>
<td>Prog [descexamples]</td>
<td>program_name: Describe the program.</td>
</tr>
<tr>
<td>Cmd [descexamples]</td>
<td>command_name: Describe the class.</td>
</tr>
</tbody>
</table>

Uses a description environment to describe objects.
Example 13: Nesting

Code:

```latex
\shownesting{\env{environmentA}}{
  \shownesting{\cs{macroB}}{
    \shownesting{\env{environmentC}}{
      The contents.
    }
  }
}
\shownesting{\env{sidebyside}}{
  \shownesting[.35]{minipage}{
    Left contents.
  } \hfill \cs{hfill} \hfill 
  \shownesting*[.35]{minipage}{
    Right contents.
  }
}
```

Result:

```
<table>
<thead>
<tr>
<th>environmentA</th>
</tr>
</thead>
<tbody>
<tr>
<td>\macroB</td>
</tr>
<tr>
<td>\shownesting{\env{environmentC}}{</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>sidebyside</th>
</tr>
</thead>
<tbody>
<tr>
<td>\shownesting[.35]{minipage}{</td>
</tr>
<tr>
<td>\cs{hfill} \hfill</td>
</tr>
<tr>
<td>\shownesting*[.35]{minipage}{</td>
</tr>
</tbody>
</table>

Note the use of the optional arguments to select less than full \textwidth, and the starred form for the second box to place it inline with the \hfill text.
Contents of the figure.

Figure 1: A Figure

Example 14: \texttt{dtxexample}

\textit{Code:}

\begin{verbatim}
\begin{figure}
  \centering \fbox{Contents of the figure.}
  \caption{A Figure} \label{fig:afigure}
\end{figure}
\end{verbatim}

\textit{Result:}

\textit{See fig. 1}

Example 14, typeset above, was created with the following code:

\begin{verbatim}
\begin{dtxexample}[See \cref{fig:afigure}]
  \begin{figure}
    \centering \fbox{Contents of the figure.}
    \caption{A Figure} \label{fig:afigure}
  \end{figure}
\end{dtxexample}
\end{verbatim}

When the example was created:

1. The “float” of type example was created, with the caption “\texttt{dtxexample}” and the label \texttt{ex:dtxexample}, which points to example 14.
2. The code was displayed verbatim.
3. The code was written to the file \texttt{dtxexample_cut.tex}.
4. The code was \texttt{\input} from \texttt{dtxexample_cut.tex}.
5. Executing the code created the figure with caption “A Figure” and label \texttt{fig:afigure}, which points to fig. 1.
6. The cross-reference to the figure was shown on the optional display line by the optional argument to \texttt{dtxexample}.
7. The starred form of \texttt{dtxexample} was used to create the closing rule below the code, since a float was being generated and nothing followed the code inline. An unstarred version would have created an extra rule.
Example 15: fsourceverb

Code:

% \begin{fsourceverb}[label=An fsourceverb example]
% \newcommand{fdosomething}[1][whattodo]{
%    doing #1
% }
% \end{fsourceverb}

Result:

\begin{fsourceverb}[label=An fsourceverb example]
\newcommand{fdosomething}[1][whattodo]{
    doing #1
}
\end{fsourceverb}

(The leading \% characters would be present in the dtx source.)

Example 16: sourcedisplay

Code:

\begin{sourcedisplay}
\cs{newcommand}\{dosomething\}[1][\textcolor{red}{whattodo}]{\\ \fquad \textcolor{blue}{doing \textcolor{red}{\#1}}\}\\}
\end{sourcedisplay}

Result:

\begin{sourcedisplay}
\cs{newcommand}\{dosomething\}[1][\textcolor{red}{whattodo}]{\\ \fquad \textcolor{blue}{doing \textcolor{red}{\#1}}\}\\}
\end{sourcedisplay}
Example 17: UIdisplay

Code:

Select:
\begin{UIdisplay}
  \textsf{Preferences $\to$ Plugins $\to$ Files $\to$ HTML}
\end{UIdisplay}

For the field
\begin{UIdisplay}
Title heading:
\end{UIdisplay}
\userentry{H1}

Result:

Select:

Preferences $\to$ Plugins $\to$ Files $\to$ HTML

For the field

Title heading:

Enter ⇒ H1
Example 18: docsidebar

*Code:*

Main text.

More main text.

\begin{docsidebar}[A title]
An aside, which may help explain something incidental to the main text.
\end{docsidebar}

Additional main text.

*Result:*

Main text.

More main text.

\begin{docsidebar}[A title]
An aside, which may help explain something incidental to the main text.
\end{docsidebar}

Additional main text.
5 Usage notes

Placement of \Describe macros: Typically \LaTeX macro and environment definitions are enclosed in macro and environment environments at their place in the source code. \DescribeMacro and \DescribeEnv would be used elsewhere in the manual to describe how to use the code. \DescribeBoolean and such might be at their place in the source code, unless they are worthy of discussion for the end-user, in which case they should be in the “User's Manual” section of the document. It may be useful to use \DeclareBoolean and friends both at the code location and also in the User's Manual section.

Extra spaces: When placing multiple uses of \Describe, \index, \margintag, and \watchout macros together, care must be taken to avoid extra space in the printed text where these macros occur. A trailing percent character may be used to avoid the extra space:

\text{text text text\%} \text{<-- avoids extra space}
\margintag{A comment.}
\index{An entry}
\index{Another entry}
more inline text

Unwanted vertical space: Other environments nested inside a \docsidebar may produce excessive vertical space. It may be required to insert

\vspace*{-\baselineskip}

\margintag placement: To have the margin tag appear next to the first line of a paragraph, place the \margintag or \watchout somewhere after the first few words in the paragraph. The \margintag may be on its own line, and the rest of the paragraph may follow on the next line. If too many words are printed before the \margintag, the words may wrap to the next line before the tag occurs.

Margin tag overlap: To keep margin tags in proper alignment, use a new paragraph or multiple lines between \margintag, \watchout, or \Declare macros

missing tags

\Describe inside floats: When these macros are used inside a float, the margin tag is suppressed (there is no margin in a float), but the index entries are still created.

\footnote{Future versions may include \DeclareBoolean for use at the point where the boolean is defined, creating an index entry with a code line number, and \DescribeBoolean with a page number index entry for the related discussion in the User's Manual portion of the document.}
6 Code

6.1 Required packages

One of several index programs must be provided. One of several index programs must be provided.

\AtBeginDocument{
\@ifpackageloaded{makeidx}{{
  \@ifpackageloaded{splitidx}{{
    \RequirePackage{makeidx}
    \makeindex
  }}
}}}

v2.6 or later for \BeforeBeginEnvironment, \AfterEndEnvironment

\RequirePackage{etoolbox}[2011/01/03]%

Used for the examples.

\RequirePackage{xparse}

Used for \shownesting.

\RequirePackage{calc}

Used for the examples.

\RequirePackage{xcolor}
\definecolor{myurlcolor}{rgb}{0,0,.7}
\definecolor{mylinkcolor}{rgb}{.7,0,0}

Used for the examples.

\RequirePackage{caption}

Used for the examples.

\RequirePackage{newfloat}

Used for the examples.

\RequirePackage{fancyvrb}
\StrSubstitute for \DescribeFile.

\RequirePackage{xstring}

If hyperref is loaded, disable some macros in PDF bookmarks:
\AtBeginDocument{
  \ifpackageloaded{hyperref}{
    \pdfstringdefDisableCommands{
      \def\quad{ }\n      \def\\{ }\n      \def\pkg#1{#1}\n      \def\ctr#1{#1}\n      \def\bool#1{#1}\n      \def\optn#1{#1}\n      \def\env#1{#1}\n      \def\cs#1{\textbackslash#1}\n      \def\,{}\n      \def\LuaLaTeX{LuaLaTeX}\n      \def\XeLaTeX{XeLaTeX}\n      \def\TeX{TeX}\n      \def\LaTeX{LaTeX}\n      \def\LaTeXe{LaTeX2e}\n      \def\LuaTeX{LuaTeX}\n      \def\XeTeX{XeTeX}\n      \def\AmS{AMS}\n      \def\Dash{ --- }\n      \def\dash{ -- }\n      \def\Slash{/}\n      \def\prog#1{\detokenize{#1}}\n      \def\progcode#1{#1}\n      \def\filenm#1{\detokenize{#1}}\n      \def\brand#1{#1}\n      \def\acro#1{#1}\n      \def\ODT{ODT}\n      \def\SVG{SVG}\n      \def\PNG{PNG}\n      \def\GIF{GIF}\n      \def\JPG{JPG}\n      \def\EPS{EPS}\n      \def\PDF{PDF}\n      \def\DVI{DVI}\n      \def\UTF{UTF}\n      \def\URL{URL}\n      \def\element#1{#1}\n      \def\attribute#1{#1}\n      \def\attrib#1{#1}\n      \def\HTML{HTML}\n      \def\HTMLfive{HTML5}\n      \def\CSS{CSS}\n      \def\CSSthree{CSS3}\n      \def\EPUB{EPUB}\n      \def\TOC{TOC}\n      \def\LOF{LOF}\n    }
  }
  \def\quad{ }\n  \def\\{ }\n  \def\pkg{#1}\n  \def\ctr{#1}\n  \def\bool{#1}\n  \def\optn{#1}\n  \def\env{#1}\n  \def\cs{\textbackslash #1}\n  \def\,{}\n  \def\LuaLaTeX{LuaLaTeX}\n  \def\XeLaTeX{XeLaTeX}\n  \def\TeX{TeX}\n  \def\LaTeX{LaTeX}\n  \def\LaTeXe{LaTeX2e}\n  \def\LuaTeX{LuaTeX}\n  \def\XeTeX{XeTeX}\n  \def\AmS{AMS}\n  \def\Dash{ --- }\n  \def\dash{ -- }\n  \def\Slash{/}\n  \def\prog{\detokenize{#1}}\n  \def\progcode{#1}\n  \def\filenm{\detokenize{#1}}\n  \def\brand{#1}\n  \def\acro{#1}\n  \def\ODT{ODT}\n  \def\SVG{SVG}\n  \def\PNG{PNG}\n  \def\GIF{GIF}\n  \def\JPG{JPG}\n  \def\EPS{EPS}\n  \def\PDF{PDF}\n  \def\DVI{DVI}\n  \def\UTF{UTF}\n  \def\URL{URL}\n  \def\element{#1}\n  \def\attribute{#1}\n  \def\attrib{#1}\n  \def\HTML{HTML}\n  \def\HTMLfive{HTML5}\n  \def\CSS{CSS}\n  \def\CSSthree{CSS3}\n  \def\EPUB{EPUB}\n  \def\TOC{TOC}\n  \def\LOF{LOF}\n
}
If \texttt{hyperref} is not loaded, emulate \texttt{\hyperpage} here.

\begin{verbatim}
\begin{Verbatim}
\newcommand*{\hyperpage}{1}\end{Verbatim}
\end{verbatim}

\section{6.2 Gobbling comment characters}

\texttt{DTXD@gobble} The \texttt{.dtx} format uses leading percent characters for code to be in the documentation only. Other classes do not.

\begin{verbatim}
\@ifpackageloaded{doc}{
  \newcommand*{\DTXD@gobble}{2}
}{
  \newcommand*{\DTXD@gobble}{0}
}
\end{verbatim}

\section{6.3 Vertical spacing}

\begin{verbatim}
\setlength{\marginparsep}{1em}
\setlength{\marginparpush}{.7ex}
\setlength{\parindent}{0em}
\setlength{\parskip}{2ex}
\end{verbatim}

\texttt{Len \IndexMin} From \texttt{ltxdoc}.

\begin{verbatim}
\ifdef{\IndexMin}{
  \setlength{\IndexMin}{40ex}}
\end{verbatim}
6.4 \texttt{ltxdoc} emulation

If the \texttt{ltxdoc} class is not used, some of its macros are replicated here.

\begin{verbatim}
\ifclassloaded{ltxdoc}{}
  \def\cmd#1{\cs{\expandafter\cmd@to@cs\string#1}}
  \def\cmd@to@cs#1#2{\char\number'#2\relax}
  \providecommand\marg[1]{\texttt{\char'\{}meta{#1}{\ttfamily\char'\}}}
  \providecommand\oarg[1]{\texttt{[}meta{#1}{\ttfamily\]}}
  \providecommand\parg[1]{\texttt{(}meta{#1}{\ttfamily\)}}
  \providecommand\url{\texttt}
\end{verbatim}

6.5 \texttt{doc} emulation

If the \texttt{doc} class is not used, some of its macros are replicated here.

\begin{verbatim}
\AtBeginDocument{
  \ifpackageloaded{doc}{}
    \newenvironment*[macro]{\%}{\PackageError{dtxdescribe}{The 'macro' environment is only available when using the doc package with a .dtx source file}{This environment only makes sense for .dtx source.}}
    \newenvironment*[environment]{\%}{\PackageError{dtxdescribe}{The 'environment' environment is only available when using the doc package with a .dtx source file}{This environment only makes sense for .dtx source.}}
    \def\MacroFont{\fontencoding{encodingdefault}
      \fontfamily{ttdefault}
      \fontseries{mddefault}
      \fontshape{updefault}
      \small}\%
    \ifdefined{actualchar}{\def\actualchar{@}}
    \ifdefined{quotechar}{\def\quotechar{"}}
    \ifdefined{levelchar}{\def\levelchar{!}}
    \ifdefined{encapchar}{\def\encapchar{|}}
    \ifdefined{verbatimchar}{\def\verbatimchar{+}}
    \setlength{marginparpush}{8pt} \setlength{marginparwidth}{8pc}
  \reversemarginpar
\end{verbatim}
6.6 Support macros

\PrintEnvName {⟨name⟩} Prints an environment name.

\DTXD@printtype {⟨text⟩} Used to print the object class in the margin:

\usage {⟨text⟩} Allow hyperlinks in the "usage" index entries:

\DTXD@origwrindex Used to bypass hyperref index modifications.

\DTXD@margintag {⟨class⟩} {⟨name⟩} {⟨margin tag⟩} Creates the margin tag for the object being described.

The class is used to sub-categories keys into their key/value groups.
\texttt{\DTXD@index} \{(\textit{class})\} \{(\textit{name})\} \{(\textit{margin tag})\} \{(\textit{index tag})\} \{(\textit{main/usage})\}

Creates the index entries for the object being described, where name has no backslash or underscore.

The \texttt{class} is used to sub-categories keys into their key/value groups. \texttt{main} prints code lines in the index, and \texttt{usage} prints page numbers.

\texttt{\DTXD@index} \{(\textit{class})\} \{(\textit{name})\} \{(\textit{margin tag})\} \{(\textit{index tag})\} \{(\textit{main/usage})\} \{\%

The \texttt{makeindex} program allows each index entry to call a macro by appending a vertical bar and a macro name to each entry. \texttt{hyperref} adds a call by \texttt{\hyperpage} to each index entry, by appending the phrase \texttt{\hyperpage} to the entry in the .idx file. The \texttt{doc} package uses the same mechanism to distinguish between code line entries (|\texttt{main}) and references to the use of a macro (|\texttt{usage}). The problem is that \texttt{makeindex} can only handle one macro call, but \texttt{hyperref} tries to append its \texttt{\hyperpage} to the already-existing |\texttt{usage} or |\texttt{main}.

The solution used for \texttt{dtxdescribe} is to allow \texttt{hyperref} to modify all regular index entries, but use the original definition of \texttt{\@wrindex} for the \texttt{\Describe} macros, before \texttt{hyperref} modified it. Then, the \texttt{\usage} macro, defined above, manually adds the hyperlink.

Below, \texttt{\@bsphack} and \texttt{\@esphack} seem to be required for \texttt{\@wrindex} to work. \texttt{\ignorespaces} is used in addition because \texttt{\Declare} and \texttt{\index} entries often come in groups.

\texttt{\@bsphack}%
\texttt{\begingroup%}
\texttt{\DTXD@origwrindex}%

Index by name:

Write the name, the formatted name, the index tag, and the class:
Index by tag and class:
Write the tag and class as a group, under which is the name and the formatted name.

Possibly index by class and name:

\DTXD@margintagindex \{\langle class\rangle\} \{\langle name\rangle\} \{\langle margin tag\rangle\} \{\langle index tag\rangle\} \{\langle main/usage\rangle\}

Creates the margin tag and the index entries. The class is used to sub-categories keys into their key/value groups.

\DTXD@macroname \{\langle control sequence\rangle\}

Given a control sequence such as \name, prints its name without the backslash.
From: http://tex.stackexchange.com/questions/42318/removing-a-backslash-from-a-character-sequence

\begin{verbatim}
\begingroup\lccode'\|='\\
\lowercase{\endgroup\def\removebs#1{\if#1|\else#1\fi}}
\newcommand*{\DTXD@macroname}[1]{\expandafter\removebs\string#1}
\DTXD@verbatimcmd{\name}
\DTXD@cmdmargintagindex{\class}{\name}{\margin tag}{\index tag}{\main/usage}
\DTXD@origwrindex{\ifblank{#1}{}{#1\actualchar[#1]:\levelchar}{\DTXD@macroname[#2]\actualchar\DTXD@verbatimcmd[#2] \index tag \DTXD@verbatimcmd{\name} \#4% index tag
\encapchar #5}}%
\end{verbatim}

While printing to the index file, prints the \name verbatim. From \SpecialIndex in the doc package.

\newcommand*{\DTXD@cmdmargintagindex}[5]{%}

Create a margin tag with the name of the macro:

\ifundefined{@captype}{% not float?
\leavevmode
\marginpar{%
\hbadness=10000%
\hfuzz=5em%
\DTXD@printtype{#3% margin tag
\ifblank{#1}{}{ \[#1\]}% class
}\cmd{#2}% name
}% marginpar
}% not float?

Create an index entry sorted by the name without its leading backslash, followed by the macro name with the backslash, and the tag. Prepend with the class if given.

Write (class):>name=csname (indextag)|usage

\begingroup\lccode'\|='\\
\lowercase{\endgroup\def\removebs#1{\if#1|\else#1\fi}}
\newcommand*{\DTXD@macroname}[1]{\expandafter\removebs\string#1}
Create an index entry grouped by the tag, then printed and sorted by the macro name with the backslash, and the tag.

Write `index:tag:(class):csname|usage`

\begingroup\DTXD@origwrindex{\DTXD@verbatimcmd{#2}\[\DTXD@macroname{#2}\]#5}%
\@esphack\ignorespaces}

6.7 `\DescribeMacro` and `\DescribeEnvironment`

Redefined to allow hyperlinked index entries and an optional class:

\providecommand*{\DescribeMacro}{\renewcommand*{\DescribeMacro}[2][\]{\@bsphack\ifundefined{@captype}{\leavevmode\marginpar{\hbadness=10000\hfuzz=5em\raggedleft\ifblank{#1}{}{{\scriptsize\textsf{[#1]}} }% class % cmd{#2}% name %}}% not float?%}}% marginpar%\}% not float?

Write the index sorted by the name without the backslash, followed by the actual name with the backslash. Append the class if given.

Write `name=csname>(class)|usage`

\begingroup\DTXD@origwrindex{\DH@macroname{#2}\actualchar\DTXD@verbatimcmd{#2}\ifblank{#1}{}{\levelchar[#1]}}% class % encapchar usage%}
Only if a class was given:

\begin{macrocode}
\begingroup
\DTXD@origwrindex{#1\actualchar[#1]:\levelchar\DTXD@macroname{#2}\actualchar\DTXD@verbatimcmd{#2}\encapchar usage}%
\@esphack\ignorespaces\endgroup
\end{macrocode}

\DescribeEnv \[⟨class⟩] \{⟨environment name⟩\}

Redefined to allow hyperlinked index entries:

\providecommand*{\DescribeEnv}{\DTXD@margintagindex{#1}{#2}{Env}{environment}{usage}}

\DTX@filename Stores the filename with a sanitized underscore.

\newcommand*{\DTXD@filename}{}

\DTXD@filemarginparindex \{⟨class⟩\} \{⟨name⟩\} \{⟨margin tag⟩\} \{⟨index tag⟩\} \{⟨main/usage⟩\}

The name may have underscores.

\newcommand*{\DTXD@filemarginparindex}[5]{\StrSubstitute{\DTXD@filename}{\detokenize{_}}{\detokenize{\_}}{\DTXD@filename}}

Create a detokenized version of the filename...

\renewcommand{\DTXD@filename}{\detokenize(#2)}

... then replace any underscores with a detokenized _, which will print as an underscore when read back from the index file:

\StrSubstitute{\DTXD@filename}{\detokenize(_)}{\detokenize(\_)}{\DTXD@filename}
The original filename is printed in the margin. Any underscore characters have already been disabled by the `\catcode` change.

\DTXD@margintag{#1}{#2}{#3}%

The detokenized and sanitized version is sent to the index file:

\DTXD@index{#1}{\DTXD@filename}{#3}{#4}{#5}%

End the group with the disabled underscore, and clean up the extra space from the `\catcode` command:

\endgroup%
\ignorespaces%
}

\DTXD@DescribeFile
⟨class⟩ {⟨name⟩}

The name may have underscores.

\newcommand*{\DTXD@DescribeFile}[2][]{%
\DTXD@filemarginparindex(#1)(#2)(File){file}(usage)
}

\DescribeFile {⟨name⟩}

The underscore character is temporarily disabled, then the name is passed directly to \DTXD@DescribeFile.

\newcommand*{\DescribeFile}{%
\begingroup\catcode\_=12 \DTXD@DescribeFile%
}

\DTXD@DescribeProgram
⟨class⟩ {⟨name⟩}

The name may have underscores.

\newcommand*{\DTXD@DescribeProgram}[2][]{%
\DTXD@filemarginparindex(#1)(#2)(Prog){program}(usage)
}

\DescribeProgram {⟨name⟩}

The underscore character is temporarily disabled, then the name is passed directly to \DTXD@DescribeProgram.

\newcommand*{\DescribeProgram}{%
\begingroup\catcode\_=12 \DTXD@DescribeProgram%
}
\DTXD@DescribeCommand \[⟨class⟩\] \{⟨name⟩\}

The name may have underscores.

\newcommand*{\DTXD@DescribeCommand}[2][]{% 
\DTXD@filemarginparindex{#1}{#2}{Cmd}{command}{usage}%
\}

\DescribeCommand \{⟨name⟩\}

The underscore character is temporarily disabled, then the name is passed directly to \DTXD@DescribeCommand.

\newcommand*{\DescribeCommand}{% 
\begingroup\catcode'\_=12 \DTXD@DescribeCommand%
\}

\DTXD@DescribePackage \[⟨class⟩\] \{⟨name⟩\} The name may have underscores.

\newcommand*{\DTXD@DescribePackage}[2][]{% 
\DTXD@filemarginparindex{#1}{#2}{Pkg}{package}{usage}%
\}

\DescribePackage \{⟨name⟩\}

The underscore character is temporarily disabled, then the name is passed directly to \DTXD@DescribePackage.

\newcommand*{\DescribePackage}{% 
\begingroup\catcode'\_=12 \DTXD@DescribePackage%
\}

\DTXD@DescribeClass \[⟨class⟩\] \{⟨name⟩\}

The name may have underscores.

\newcommand*{\DTXD@DescribeClass}[2][]{% 
\DTXD@filemarginparindex{#1}{#2}{Cls}{class}{usage}%
\}

\DescribeClass \{⟨name⟩\}

The underscore character is temporarily disabled, then the name is passed directly to \DTXD@DescribeClass.

\newcommand*{\DescribeClass}{% 
\begingroup\catcode'\_=12 \DTXD@DescribeClass%
\}
\DescribeOption \[[\textit{class}]\] \{\textit{name}\}

\newcommand*{\DescribeOption}[2]{\DTXD@margintagindex{#1}{#2}{Opt}{option}{usage}}

The class may be used to categorize arguments by their macro or environment name.

\DescribeArgument \[[\textit{class}]\] \{\textit{name}\}

\newcommand*{\DescribeArgument}[2]{\DTXD@margintagindex{#1}{#2}{Arg}{argument}{usage}}

The class may be used to categorize arguments by their macro or environment name.

\DescribeBoolean \[[\textit{class}]\] \{\textit{name}\}

\newcommand*{\DescribeBoolean}[2]{\DTXD@margintagindex{#1}{#2}{Bool}{boolean}{usage}}

\DescribeLength \[[\textit{class}]\] \{\textit{name}\}

\newcommand*{\DescribeLength}[2]{\DTXD@cmdmargintagindex{#1}{#2}{Len}{length}{usage}}

\DescribeCounter \[[\textit{class}]\] \{\textit{name}\}

\newcommand*{\DescribeCounter}[2]{\DTXD@margintagindex{#1}{#2}{Ctr}{counter}{usage}}

\DescribeHook \[[\textit{class}]\] \{\textit{name}\}

\newcommand*{\DescribeHook}[2]{\DTXD@margintagindex{#1}{#2}{Hook}{hook}{usage}}

\DescribeKey \[[\textit{class}]\] \{\textit{name}\}

The class may be used to categorize keys by their kev/value group.

\newcommand*{\DescribeKey}[2]{\DTXD@margintagindex{#1}{#2}{Key}{key}{usage}}

\DescribeObject \[[\textit{class}]\] \{\textit{name}\}

May be used to describe an arbitrary piece of code. Creates a margin tag and index entries with \texttt{ttfamily}.

\newcommand*{\DescribeObject}[2]{%}
% @ifundefined{@captype}{% not float?
% \@bsphack%
\DescribeOther \[\langle \text{class} \rangle \] \{\langle \text{name} \rangle \}

May be used to describe an arbitrary non-programming object. Creates a margin tag and index entries with roman type.
The color of the margin tag used to show the default value.

\newcommand{\DescribeDefaultcolor}{green!50!black}

\DescribeDefault \langle value \rangle

Creates a colored margin tag showing the booleandefault value.

\newcommand{\DescribeDefault}[1]{\margintag{\footnotesize\textcolor{\DescribeDefaultcolor}{Default: \texttt{#1}}}}

The following are for use inside a description.

\ItemDescribeMacro \langle class \rangle \langle name \rangle

\newcommand{\ItemDescribeMacro}[2][{}]{\item[\texttt{\texttt{#2}}:\]}\DescribeMacro[#1]{#2}}

\ItemDescribeEnv \langle class \rangle \langle name \rangle

\newcommand{\ItemDescribeEnv}[2][{}]{\item[\texttt{\texttt{#2}}:\]}\DescribeEnv[#1]{#2}}
\setlength{\parskip}{1.5ex}
\DescribeEnv[#1]{#2}

\ItemDescribeArgument \[⟨class⟩\] {⟨argument⟩}
\newcommand{\ItemDescribeArgument}[2][% 
\item[\texttt{#2}]:% \setlength{\parskip}{1.5ex}% \DescribeArgument[#1]{#2}%
%
\ItemDescribeBoolean \[⟨class⟩\] {⟨name⟩}
\newcommand{\ItemDescribeBoolean}[2][% 
\item[\texttt{#2}]:% \setlength{\parskip}{1.5ex}% \DescribeBoolean[#1]{#2}%
%
\ItemDescribeLength \[⟨class⟩\] {⟨name⟩}
\newcommand{\ItemDescribeLength}[2][% 
\item[\texttt{#2}]:% \setlength{\parskip}{1.5ex}% \DescribeLength[#1]{#2}%
%
\ItemDescribeCounter \[⟨class⟩\] {⟨name⟩}
\newcommand{\ItemDescribeCounter}[2][% 
\item[\texttt{#2}]:% \setlength{\parskip}{1.5ex}% \DescribeCounter[#1]{#2}%
%
\ItemDescribeHook \[⟨class⟩\] {⟨name⟩}
\newcommand{\ItemDescribeHook}[2][% 
\item[\texttt{#2}]:% \setlength{\parskip}{1.5ex}% \DescribeHook[#1]{#2}%
%
\ItemDescribeKey \[⟨class⟩\] {⟨name⟩}
\newcommand{\ItemDescribeKey}[2][% 
\item[\texttt{#2}]:% \setlength{\parskip}{1.5ex}% \DescribeKey[#1]{#2}%
\ItemDescribePackage \[\langle \text{class} \rangle \] \{\langle \text{name} \rangle \}

\newcommand{\DTXD@ItemDescribePackage}[2][]{% 
\item\[\texttt{#2}:\]%  
\setlength{\parskip}{1.5ex}%  
\DescribePackage[#1]{#2}%  
\endgroup%}

\ItemDescribeClass \[\langle \text{class} \rangle \] \{\langle \text{name} \rangle \}

\newcommand{\DTXD@ItemDescribeClass}[2][]{% 
\item\[\texttt{#2}:\]%  
\setlength{\parskip}{1.5ex}%  
\DescribeClass[#1]{#2}%  
\endgroup%}

\ItemDescribeOption \[\langle \text{class} \rangle \] \{\langle \text{name} \rangle \}

\newcommand{\DTXD@ItemDescribeOption}[2][]{% 
\item\[\texttt{#2}:\]%  
\setlength{\parskip}{1.5ex}%  
\DescribeOption[#1]{#2}%}

\ItemDescribeFile \[\langle \text{class} \rangle \] \{\langle \text{name} \rangle \}

\newcommand{\DTXD@ItemDescribeFile}[2][]{% 
\item\[\texttt{#2}:\]%  
\setlength{\parskip}{1.5ex}%  
\DescribeFile[#1]{#2}%  
\endgroup%}

\ItemDescribeProgram \[\langle \text{class} \rangle \] \{\langle \text{name} \rangle \}

\newcommand{\DTXD@ItemDescribeProgram}[2][]{% 
\item\[\texttt{#2}:\]%}
\ItemDescribeCommand \{⟨class⟩\} \{⟨name⟩\}

\ItemDescribeObject \{⟨class⟩\} \{⟨name⟩\}

\ItemDescribeOther \{⟨class⟩\} \{⟨name⟩\}

6.11 \margintag, \watchout

\margintagcolor The color of the \margintag.

\margintag \{⟨text⟩\}

Prints a colored margin tag.
The color of the \watchout.

\newcommand*{\watchoutcolor}{red!50!black}

\watchout{⟨text⟩}

Prints a warning sign and optional text.

\newcommand{\watchout}[1][\textcolor{\watchoutcolor}{\warningsign
normalsize\quad#1}]{}%\ignorespaces%

6.12 Nesting

Shows a box enclosing a label for the container, and the container’s contents. May be nested.

\shownesting{⟨fraction of \linewidth⟩}{⟨container⟩}{⟨contents⟩}
6.13 The dtxexample environment

Also see example 14 on page 26.

Used to store then \input example code.

The color of the middle rule in the dtxexample.

\definecolor{DTXD@examplerulecolor}{rgb}{.9,.9,.9}

The text name of the code section.

\newcommand*{\dtxexamplecodename}{Code:}

The text name of the result section.

\newcommand*{\dtxexampleresultname}{Result:}

Env dtxexample *

\begin{notes/cross-references}{caption & label}

Reads the code listing as a verbatim input using the fancybox package, then displays the code listing as a verbatim output, and also executes the code and displays the result. A title caption is specified, along with optional cross-referencing commands or notes to refer to the results. The unstarred version places the code inside a minipage, forbidding a page break in the middle of the code listing. The starred version does not use a minipage. This is required when the code is too large to fit on a single page.

\NewDocumentEnvironment{dtxexample}{s+O{}m}

Copy the environment's contents to the file dtxexample_cut.tex:

\VerbatimOut[gobble=\DTXD@gobble,tabsize=4]{dtxexample_cut.tex}% start dtxexample

When the environment closes:

\endVerbatimOut
\addvspace{\bigskipamount}

If unstarred, typeset the example in a minipage:

\IfBooleanTF{#1}{\vspace{\bigskipamount}}{\minipage{\linewidth}}%
Emulated a float of type “example”:
\captionsetup{type=dtxdexample}
\hrule
\medskip
\caption{#3}

Typeset the contents as verbatim:
\textcolor{DTXD@examplerulecolor}{\smallskip\hrule}
\smallskip
\textsize\itshape\dtxexamplecodename
\VerbatimInput*[tabsize=4]{dtxexample_cut.tex}
\unskip
\textcolor{DTXD@examplerulecolor}{\hrule}
\smallskip
\textsize\itshape\dtxexampleresultname

Possible add the optional cross-references or notes:
\ifstrempty{#2}
\{}
\{{\itshape\small #2}}

If unstarrred, close the \minipage.
\IfBooleanTF{#1}{}{\endminipage}

Outside of the environment’s scope, input the example to generate its output and labels:
\AfterEndEnvironment{dtxexample}
\%

Execute the code:
\par\unskip\input{dtxexample_cut.tex}\%

Closing rule::
\medskip\hrule\%

A new float type for the examples.
\DeclareFloatingEnvironment

\DeclareFloatingEnvironment[}
\fileext=lox, 
\listname={List of Examples}, 
\name=Example, 
\placement=hbp 
{dtxdexample}
Caption setup for the examples.

\captionsetup*[dtxexample]{
  format=hang,
  font=bf,
  justification=raggedright,
  singlelinecheck=false,
  skip=0pt,
  position=top,
}

Name for cleveref.

\AtBeginDocument{
  \ifpackageloaded{cleveref}{\crefname{dtxexample}{example}{examples}}{}\}

6.14 noindmacro and noindenvironment

Similar to macro and environment, but not indexed.

Env noindmacro \langle name\rangle

\newenvironment{noindmacro}[1]{
  \setlength{\parskip}{\marginparpush}
  \leavevmode\par\DTXD@margintag{}{\cmd{#1}}{}
}{\unskip}

Env noindenvironment \langle name\rangle

\newenvironment{noindenvironment}[1]{
  \setlength{\parskip}{\marginparpush}
  \leavevmode\par\DTXD@margintag{}{#1}{Env}
}{\unskip}

6.15 sourcedisplay, UIDisplay, docsidebar

For use in a sourcedisplay:

\fquad Forces a quad indent.

\newcommand*{\fquad}{\hspace*{1em}}
\fquad Forces a double-quad indent.
\fqqquad Forces a triple-quad indent.

\newcommand*{\fqquad}{\hspace*{2em}}
\newcommand*{\fqqquad}{\hspace*{3em}}

Env sourceverb To typeset a block of source code, verbatim.

\DefineVerbatimEnvironment{sourceverb}{Verbatim}{gobble=\DTXD@gobble,tabsize=4,xleftmargin=2em}
\BeforeBeginEnvironment{sourceverb}{\vspace*{-.5\parskip}}

Env fsourceverb To typeset a framed block of source code, verbatim.

\DefineVerbatimEnvironment{fsourceverb}{Verbatim}{gobble=\DTXD@gobble,tabsize=4,xleftmargin=2em,frame=lines}
\BeforeBeginEnvironment{fsourceverb}{\vspace*{-.5\parskip}}

Env sourcedisplay To typeset a block of source code, allowing direct formatting.

\DefineVerbatimEnvironment{sourcedisplay}{Verbatim}{gobble=\DTXD@gobble,tabsize=4,xleftmargin=2em,frame=lines}
\BeforeBeginEnvironment{sourcedisplay}{\vspace*{-.5\parskip}}

Env UIdisplay To typeset a user interface display.

\DefineVerbatimEnvironment{UIdisplay}{Verbatim}{gobble=\DTXD@gobble,tabsize=4,xleftmargin=2em,frame=lines}
\BeforeBeginEnvironment{UIdisplay}{\vspace*{-.5\parskip}}

\userentryname Text to tell the user to enter the following item.
\userentry \{ ⟨text to enter⟩ \}

Typesets text to be entered by the users.

\begin{minipage}{\linewidth-2em}
\footnotesize
\userentryname \quad \cmds{#1}
\end{minipage}
\par

\begin{env}
\docsidebar \To typeset a sidebar in the documentation.
\end{env}

\pkg \{ ⟨name⟩ \}

Also useable for class names.

\cs \{ ⟨csname⟩ \}

From \texttt{lbxdoc}.

\env \{ ⟨name⟩ \}

\texttt{\char`\\#1}
\marg \{\langle argument\rangle\} From ltxdoc.

\oarg \[\langle argument\rangle\] From ltxdoc.

\parg \(\langle argument\rangle\) From ltxdoc.

\ctr \{\langle name\rangle\}  

\bool \{\langle name\rangle\} 

\optn \{\langle name\rangle\} 

\TOC 

\LOF 

\LOT 

6.16.2 Programs and commands

\cmds \{\langle commands to print\rangle\} No processing is provided for special characters.
\progcde \texttt{(code to print)} \textit{No processing is provided for special characters.}

\singlespace
\begin{verbatim}
678 \robustcmd*{\progcde}[1]{\mbox{\texttt{#1}}}
\end{verbatim}

\progc 	exttt{(program name)} \textit{Underscores are allowed.}

\singlespace
\begin{verbatim}
679 \newcommand*{\DTXD@progc}[1]{%
680 \mbox{\texttt{\textsf{\textsl{\detokenize{#1}}}}}%
681 \endgroup%
682 }%
683 \begin{verbatim}
684 \robustcmd*{\progc}[1]{%
685 \begingroup%
686 \catcode`\_=12%
687 \DTXD@progc%
688 }
\end{verbatim}
\end{verbatim}

\filenn \texttt{(file name)} \textit{Underscores are allowed.}

\singlespace
\begin{verbatim}
689 \newcommand*{\DTXD@filenn}[1]{%
690 \mbox{\texttt{\detokenize{#1}}}%
691 \endgroup%
692 }%
693 \begin{verbatim}
694 \robustcmd*{\filenn}[1]{%
695 \begingroup%
696 \catcode`\_=12%
697 \DTXD@filenn%
698 }
\end{verbatim}
\end{verbatim}

\textbf{\UI} \textit{General user-interface text.}

\singlespace
\begin{verbatim}
699 \robustcmd*{\UI}[1]{\textbf{\texttt{#1}}}
\end{verbatim}

\section{File types}

\textbf{\ODT} \acro{ODT}

\singlespace
\begin{verbatim}
700 \robustcmd*{\ODT}{\acro{ODT}}
\end{verbatim}

\textbf{\SVG} \acro{SVG}

\singlespace
\begin{verbatim}
701 \robustcmd*{\SVG}{\acro{SVG}}
\end{verbatim}

\textbf{\PNG} \acro{PNG}

\singlespace
\begin{verbatim}
702 \robustcmd*{\PNG}{\acro{PNG}}
\end{verbatim}
Each of these is "provided", and any prior meaning will be unchanged. In particular, \LaTeX{} uses \attribute, so its meaning is unchanged if using \LaTeX{.}
6.16.5 Specific programs

\tikz

719 \providerobustcmd*{\tikz}{Ti\textit{k}z}

\CTAN

720 \providerobustcmd*{\CTAN}{\acro{CTAN}}

\TDS

721 \providerobustcmd*{\TDS}{\acro{TDS}}

\MathML

722 \providerobustcmd*{\MathML}{Math\acro{ML}}

\MathJax

723 \providerobustcmd*{\MathJax}{\brand{MathJax}}
6.16.6 Acronyms, brand names, trademarks

\brand\{⟨name⟩\}

724 \providerobustcmd*{\brand}[1]{\textsc{#1}}

\acro\{⟨acronym⟩\}

725 \providerobustcmd*{\acro}[1]{\textsc{\lowercase{#1}}}

\supregistered Superscript trademark symbol.

726 \providerobustcmd*{\supregistered}{\textsuperscript{\textregistered}}

6.17 Logos

\dviTeX DVI\TeX

727 \providerobustcmd*{\dviTeX}{\mbox{DVI\TeX}}

\dviLaTeX DVI\LaTeX

728 \providerobustcmd*{\dviLaTeX}{\mbox{DVI\LaTeX}}

\pdfTeX PDF\TeX

729 \providerobustcmd*{\pdfTeX}{\mbox{PDF\TeX}}

\pdfLaTeX PDF\LaTeX

730 \providerobustcmd*{\pdfLaTeX}{\mbox{PDF\LaTeX}}

\LuaTeX Lua\TeX

731 \providerobustcmd*{\LuaTeX}{\mbox{Lua\TeX}}

\LuaLaTeX Lua\LaTeX

732 \providerobustcmd*{\LuaLaTeX}{\mbox{Lua\LaTeX}}

\XeTeX X\TeX, X\LaTeX

733 \providerobustcmd*{\XeTeXrevE}{\hspace{-.1667em}\raisebox{-.5ex}{E}\hspace{-.125em}}

734 \hspace{-.1667em}\raisebox{-.5ex}{E}\hspace{-.125em}}
\AtBeginDocument{\ifpackageloaded{graphics}{\renewrobustcmd*{\XeTeXrevE}{\hspace{-.1667em}\raisebox{-.5ex}{\reflectbox{E}}\hspace{-.125em}}}}

\begin{tabular}{ll}
\texttt{\AmS} & $\mathcal{A}\mathcal{M}\mathcal{S}$
\end{tabular}

\begin{tabular}{ll}
\texttt{\LyX} & LyX
\end{tabular}

\begin{tabular}{ll}
\texttt{\BibTeX} & Bib\TeX
\end{tabular}

\begin{tabular}{ll}
\texttt{\MakeIndex} & MakeIndex
\end{tabular}

\begin{tabular}{ll}
\texttt{\ConTeXt} & Con\TeX
\end{tabular}

\begin{tabular}{ll}
\texttt{\MiKTeX} & MiK\TeX
\end{tabular}

\section{6.18 Dashes and slashes}

\begin{tabular}{ll}
\texttt{\thinspace} & A breakable thin skip.
\end{tabular}

\begin{tabular}{ll}
\texttt{\endash} & An endash: –
\end{tabular}
\emdash An emdash: —

\thinbrspace A thin space which allows a line break.

\Dash An unbreakable thin space, emdash, and breakable thin space.

\dash An unbreakable thin space, endash, and breakable thin space.

\Slash An unbreakable very thin space, a slash, and a breakable thin space.
7 Compiling dtxdescribe

To compile the dtxdescribe package:

Enter ⇒ \texttt{pdflatex dtxdescribe.ins}

To compile the dtxdescribe documentation

Enter ⇒ \texttt{pdflatex dtxdescribe.dtx}

(Several times)

Enter ⇒ \texttt{makeindex -s gglo.ist -o dtxdescribe.gls dtxdescribe.glo}

Enter ⇒ \texttt{makeindex -s gind.ist dtxdescribe}

Enter ⇒ \texttt{pdflatex dtxdescribe.dtx}

(Several times)
Change History and Index

Change History

v0.10
General: 2016/12/08 Initial ver . . . . . 1

v0.11
\DTXD@cmdmargintagindex: Index
tag no longer plural. .............. 39
\DTXD@index: Index tag no longer
plural. .......................... 37
\watchout: Changed to \raggedleft. 49
General: 2018/03/30 ............... 1

v1.00
\DTXD@cmdmargintagindex: Sans tag
font. .............................. 38
\DTXD@filemarginparindex: Fix: File
class. ............................ 41
\DTXD@printtype: Sans tag font. . 35
\DescribeClass: Fix: Allow
underscore ........................ 42
\DescribeDefault: Added. ......... 45
\DescribeDefaultcolor: Added. ... 45
\DescribeMacro: Sans tag font. . 39
\DescribePackage: Fix: Allow
underscore ........................ 42
\ItemDescribeArgument: Added. ... 46
\ItemDescribeBoolean: Added. ..... 46
\ItemDescribeClass: Added. ...... 47
\ItemDescribeCommand: Added. ... 48
\ItemDescribeCounter: Added. ... 46
\ItemDescribeEnv: Added. ....... 45
\ItemDescribeFile: Added. ...... 47
\ItemDescribeKey: Added. ...... 46
\ItemDescribeLength: Added. .... 46
\ItemDescribeMacro: Added. .... 45
\ItemDescribeObject: Added. .... 48
\ItemDescribeOption: Added. ..... 47
\ItemDescribeOther: Added. ..... 48
\ItemDescribePackage: Added. .... 47
\ItemDescribeProgram: Added. ... 47
\dtxexamplecodename: Added. ... 50
\dtxexampleresultname: Added. ... 50
\fqqquad: Added. ................. 53
\fqquad: Added. .................. 53
\fquad: Added. .................. 52
\margintag: Uses \margintacolor. 48
\margintacolor: Added. .......... 48
\userentry: Added. ............... 54
\userentryname: Added. .......... 53
\watchoutcolor: Added. .......... 49
General: 2019/01/11 .............. 1
Added formatted objects. ......... 54
Added logos. ..................... 59
Cut file name changed to
dtxexample_cut.tex ............. 50
\UIdisplay: Added. ............... 53
docsidebar: Added. ............. 54
noindentenvironment: Added. .... 52
noindmacro: Added. ............. 52
sourcedisplay: Added. .......... 53
sourceverb: Added. ............. 53
fsourceverb: Added. ............ 53

v1.01
\DTXD@cmdmargintagindex: Put
margin tag class in brackets. .... 38
\DTXD@margintag: Put margin tag
class in brackets. ................ 35
\DescribeMacro: Put margin tag
class in brackets. ............... 39
\DescribeObject: Put margin tag
class in brackets. ............... 43
\DescribeOther: Put margin tag
class in brackets. ............... 44
General: 2019/03/22 .............. 1
Sanitize PDF bookmarks. ........... 31
\UIdisplay: Reduced width. ...... 53
\sourcedisplay: Reduced width. ... 53

v1.02
\DescribeEnv: Fix if not ltxdoc class. 40
\DescribeMacro: Fix if not ltxdoc
class. ............................ 39
\DescribeObject: \raggedleft
margin par. ...................... 43
\DescribeOther: \ignorespaces. ... 43
\AddIgnorespaces. ............... 43
\PrintEnvName: Fix if not ltxdoc
class. ............................ 35
\usage: Fix if not ltxdoc class. ... 35
General: 2019/07/16 .............. 1
Index

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

Symbols

[H] (argument) [figure] 22
[M] (argument) [figure] 22
[mymacro]: bold (argument) 22
\acro 15, 725
Additional Item [otherclass] 23
\AmS 15, 745
another_big_file.txt (file) [bigfiles] 20
argument:
    [\mymacro]: bold (argument) 22
    [figure]:
        [H] 22
        [M] 22
\attrib 14, 711
\attribute 14, 711
\bool 13, 672
  boolean:
    [descexamples]:
      booleanname 24
    [examples]:
      sampleboolean 19
      booleanname (boolean) [descexamples] 24
\brand 15, 724
\calc (package) 31
\caption (package) 31
\captionsetup [dtxexample] 52
\photograph 17
\class:
    [descexamples]:
      class_name 24
    [examples]:
      sample_class 20
      class_name (class) [descexamples] 24
\cmds 13, 677
\cnameref
  [photograph] 17
  [color]:
    DTXD@examplerulecolor 50
    othercolor 23

B

Before myenvironment 23
\BibTeX 15, 750
[bigfiles]:
    another_big_file.txt (file) 20
    really_big_file.txt (file) 20
    bold (argument) [\mymacro] 22
somecolor .......................... 23 \DTX@filename ........................ 285
command: ........................... 24 \DTX@cmdmarginparexampleindex ... 214
\desceexamples: .................... 24 \DTX@DescribeClass ................. 319
command_name ..................... 24 \DTX@DescribeCommand ............ 307
OS_command ....................... 20 \DTX@DescribeFile .................. 295
command_name (command) \desceexamples: ........................ 24
\ConText .......................... 15, 752
\DTX@DescribePackage .......... 313
\DTX@DescribeProgram .......... 301
counter: ........................... [dtxexample]:
\desceexamples: .................... 52
[example]: ........................ 19
\crefname                        301
\cs ................................ 13, 663
\css ................................ 14, 716
\CSSthree ........................ 14, 717
\CTAN .............................. 14, 720
\ctr ................................ 13, 671
dtxexample: ........................ 50
\cs ................................ 16, 763
\dash ................................ 16, 764
\DeclareFloatingEnvironment .... 51
dtxexample (environment) ...... 11, 549
dtxexample_cut.tex (file) ...... 50
\desceexamples: .................... 51
[example]: ........................ 17
\photograph: ........................ 11
\desceexamples: .................... 24
boolean (boolean) ................ 24
\dviLaTeX .......................... 15, 728
dviTeX .............................. 15, 727
\cs ................................ 13, 664
dviexample ......................... 319
\endash ................................ 16, 756
\endash ................................ 15, 755
\desceexamples: .................... 16
keyname (key) ...................... 24
\env ................................ 13, 664
evironment (environment) ...... 7
\desceexamples: .................... 24
command_name (command) ........ 24
evironment: (environment) ...... 11
\desceexamples: .................... 24
file_name (file) ................... 24
\otherenvironment ............... 18
\desceexamples: .................... 24
keyname (key) ...................... 24
docsidetax ......................... 13
\desceexamples: .................... 24
\lengthname (length) .......... 24
\endash ................................ 13
\endash ................................ 12
\desceexamples: .................... 24
package_name (package) .... 24
\noindentenvironment .......... 12
\desceexamples: .................... 24
program_name (program) .... 24
\noindmacro ....................... 12
\desceexamples: .................... 7
\DescribeArgument ................ 327
\sourceverb ...................... 12
\desceexamples: .................... 8
\DescribeBoolean ................. 8, 329
\sourceverb ...................... 12
\desceexamples: .................... 8
\DescribeClass ................... 8, 322
\docsidetax ....................... 13
\desceexamples: .................... 8
\DescribeCommand ................. 9, 310
\dtxexample ...................... 11
\desceexamples: .................... 8
\DescribeCounter ................. 8, 333
dtxexample ...................... 11
\desceexamples: .................... 10
\DescribeDefault ................ 10, 401
environment ...................... 7
\desceexamples: .................... 10
\DescribeDefaultcolor .......... 10, 400
fsourceverb ...................... 12
\desceexamples: .................... 7
\DescribeEnv ...................... 7, 282
macro .............................. 7
\desceexamples: .................... 9
\DescribeFile ..................... 9, 298
myenvironment .................... 18
\desceexamples: .................... 8
\DescribeHook ..................... 8, 335
\noindentenvironment .......... 12
\desceexamples: .................... 8
\DescribeKey ...................... 8, 337
\noindmacro ....................... 12
\desceexamples: .................... 8
\DescribeLength ................. 8, 331
\sourceverb ...................... 12
\desceexamples: .................... 7
\DescribeMacro ................... 7, 245
\sourceverb ...................... 12
\desceexamples: .................... 9
\DescribeObject ................. 9, 339
\uidisplay ....................... 12
\desceexamples: .................... 8
\DescribeOption ................. 8, 325
\docsidetax ....................... 649
\desceexamples: .................... 9
\DescribeOther ................. 9, 374
\docsidetax ....................... 649
\desceexamples: .................... 8
\DescribePackage ................. 8, 316
\dtxexample ...................... 549
\desceexamples: .................... 9
\DescribeProgram ............... 9, 304
\fsourceverb .................... 615
docsidetax (environment) ...... 13, 649
\noindentenvironment .......... 603

65
<table>
<thead>
<tr>
<th>Command</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>\UTF</td>
<td>14, 708</td>
</tr>
<tr>
<td>W</td>
<td></td>
</tr>
<tr>
<td>\warning</td>
<td>76</td>
</tr>
<tr>
<td>\watchout</td>
<td>11, 522</td>
</tr>
<tr>
<td>\watchoutcolor</td>
<td>11, 521</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Package</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>xcolor (package)</td>
<td>31</td>
</tr>
<tr>
<td>\XeLaTeX</td>
<td>15, 733</td>
</tr>
<tr>
<td>\XeTeX</td>
<td>15, 733</td>
</tr>
<tr>
<td>xparse (package)</td>
<td>31</td>
</tr>
<tr>
<td>xstring (package)</td>
<td>31</td>
</tr>
</tbody>
</table>