

TeXlib

Generated by Doxygen 1.8.13

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	Class Documentation	5
3.1	TeX Class Reference	5
3.1.1	Detailed Description	6
3.1.2	Constructor & Destructor Documentation	6
3.1.2.1	TeX() [1/2]	6
3.1.2.2	TeX() [2/2]	6
3.1.2.3	~TeX()	6
3.1.3	Member Function Documentation	6
3.1.3.1	close()	7
3.1.3.2	do_not_cancel() [1/2]	7
3.1.3.3	do_not_cancel() [2/2]	7
3.1.3.4	exists()	7
3.1.3.5	get_fullpath_ext()	7
3.1.3.6	get_name()	7
3.1.3.7	get_path()	8
3.1.3.8	open() [1/2]	8
3.1.3.9	open() [2/2]	8
3.1.3.10	open_rewritemode()	8
3.1.3.11	operator<<()	8
3.1.3.12	rmfiles()	9
3.1.3.13	set_image_density()	9
3.1.3.14	to()	9
3.1.3.15	to_pdf()	9
3.2	TeX::TeXException Class Reference	10
3.2.1	Detailed Description	10
3.2.2	Constructor & Destructor Documentation	10
3.2.2.1	TeXException()	10
3.2.3	Member Function Documentation	10
3.2.3.1	what()	10

Index	11
-----------------------	----

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

exception	
TeX::TeXException	10
TeX	5

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

TeX	5
TeX::TeXException	10

Chapter 3

Class Documentation

3.1 TeX Class Reference

```
#include <texlib.h>
```

Classes

- class [TeXException](#)

Public Member Functions

- [TeX](#) (bool show_shell=false)
- [TeX](#) (std::string filename, bool show_shell=false)
- [~TeX](#) ()
- template<typename string_convertable >
std::ostream & [operator<<](#) (const string_convertable to_be_written)
- void [to_pdf](#) ()
- void [to_dvi](#) ()
- std::string [to](#) (std::string ext, std::string middle_ext="pdf")
- void [set_image_density](#) (const int density)
- int [get_image_density](#) () const
- bool [open](#) ()
- bool [open_rewritemode](#) ()
- void [open](#) (std::string filename)
- void [close](#) ()
- bool [exists](#) ()
- template<typename T = std::string>
void [do_not_cancel](#) (T extension)
- template<typename T = std::string, typename... Types>
void [do_not_cancel](#) (T extension, Types... others)
- void [rmfiles](#) ()
- std::string [get_name](#) () const
- path [get_path](#) () const
- std::string [get_fullpath_ext](#) (std::string extension) const

3.1.1 Detailed Description

This class is a [TeX](#) quick-compiler: it basically converts small portions of tex code to pdf or png. It is not designed to handle a proper tex file (reason for which it the methods to write and compile are implemented in the same class), even though, with little modification, it can be used as such.

```
{TeX}
```

3.1.2 Constructor & Destructor Documentation

3.1.2.1 TeX() [1/2]

```
TeX::TeX (
    bool show_shell = false )
```

Default creator

Parameters

<i>show_shell</i>	if true shows shell information when executing shell commands (LaTeX compilation)
-------------------	---

3.1.2.2 TeX() [2/2]

```
TeX::TeX (
    std::string filename,
    bool show_shell = false )
```

Parameters

<i>filename</i>	TeX file to be opened
-----------------	---------------------------------------

3.1.2.3 ~TeX()

```
TeX::~TeX ( )
```

Destructor Removes the last file created and the compilation temporary file. If certain extensions are not to be removed they should be passed to the function `do_not_cancel(ext)`.

3.1.3 Member Function Documentation

3.1.3.1 close()

```
void TeX::close ( ) [inline]
```

Closes the current opened file if any.

3.1.3.2 do_not_cancel() [1/2]

```
template<typename T = std::string>
void TeX::do_not_cancel (
    T extension ) [inline]
```

The two following functions work thanks to variadic templates. They specify which extension of the compiled tex file are not to be removed in the exe path

3.1.3.3 do_not_cancel() [2/2]

```
template<typename T = std::string, typename... Types>
void TeX::do_not_cancel (
    T extension,
    Types... others ) [inline]
```

See above for explanation. The function is invoked like this: `do_not_cancel(ext1, ext2, ext3, ..., extn);`

3.1.3.4 exists()

```
bool TeX::exists ( ) [inline]
```

Returns true if the file stored in `_texname` exists in the exe path.

3.1.3.5 get_fullpath_ext()

```
std::string TeX::get_fullpath_ext (
    std::string extension ) const
```

returns filename with the full path and the extension

Parameters

<i>extension</i>	the extension to be added at the end of the file.
------------------	---

3.1.3.6 get_name()

```
std::string TeX::get_name ( ) const
```

Access method for the name of the .tex file without extension.

3.1.3.7 get_path()

```
path TeX::get_path ( ) const
```

returns the path where the tex_file is stored

3.1.3.8 open() [1/2]

```
bool TeX::open ( )
```

Opens the file stored in _texname. It returns

- false If no file constructor or opene(filename) has been called ever before or if any error occurs when opening the file
- true If it actually opens a file and everything goes well.

No exception is thrown in this function, for it returns false easily. Exceptions must be handled with an if statement when calling the function.

3.1.3.9 open() [2/2]

```
void TeX::open (
    std::string filename )
```

Opens a new file with a specified file name

Parameters

<i>filename</i>	the name or path of the file that's going to be opened
-----------------	--

If there's problem at opening the file exception is thrown here.

3.1.3.10 open_rewritemode()

```
bool TeX::open_rewritemode ( )
```

Opens the file explicitly with ios::trunc. Returns meaning same as in [open\(\)](#).

3.1.3.11 operator<<()

```
template<typename string_convertable >
std::ostream& TeX::operator<< (
    const string_convertable to_be_written ) [inline]
```

The default << operator to write on file.

Parameters

<i>string</i>	what will be written on file. Accepts anything that can be converted to a string.
---------------	---

3.1.3.12 rmfiles()

```
void TeX::rmfiles ( )
```

Removes the temporary files. Let A be the set of all possible extensions and B the set of extension defined by the user; then all files of the kind `emptyname.ext`, with `ext` in A/B will be removed

3.1.3.13 set_image_density()

```
void TeX::set_image_density (
    const int density )
```

Access methods for image density. The image density is intended to be the pixel density. By default it is 600.

3.1.3.14 to()

```
string TeX::to (
    std::string ext,
    std::string middle_ext = "pdf" )
```

The central function of the library

Parameters

<i>ext</i>	the extension to which one wants to convert the TeX. The allowed extensions can be found https://www.imagemagick.org/script/formats.php
<i>middle_ext</i>	the extension to which the TeX code should be compiled to: either PDF or DVI. Experimentally I can see that PDF is quicker, contrarily to expectations.

Returns an eventual warning message. It makes no sense to throw exceptions for simple warnings.

3.1.3.15 to_pdf()

```
void TeX::to_pdf ( )
```

Most basic conversion. Converts the tex input to a pdf document

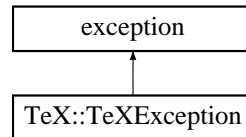
The documentation for this class was generated from the following files:

- C:/Users/nicoc_000/Desktop/Grafia/Grafia/Source/texlib.h
- C:/Users/nicoc_000/Desktop/Grafia/Grafia/Source/texlib.cpp

3.2 TeX::TeXException Class Reference

```
#include <texlib.h>
```

Inheritance diagram for TeX::TeXException:



Public Member Functions

- `template<typename T >`
`TeXException (T what)`
- `virtual const char * what () const throw ()`

3.2.1 Detailed Description

standard exception for errors happening in this library

3.2.2 Constructor & Destructor Documentation

3.2.2.1 TeXException()

```
template<typename T >
TeX::TeXException::TeXException (
    T what ) [inline]
```

sets the message error for the exception considered

3.2.3 Member Function Documentation

3.2.3.1 what()

```
virtual const char* TeX::TeXException::what ( ) const throw ) [inline], [virtual]
```

returns the message error for the exception considered

The documentation for this class was generated from the following file:

- C:/Users/nicoc_000/Desktop/Grafia/Grafia/Source/texlib.h

Index

- ~TeX
 - TeX, [6](#)
- close
 - TeX, [6](#)
- do_not_cancel
 - TeX, [7](#)
- exists
 - TeX, [7](#)
- get_fullpath_ext
 - TeX, [7](#)
- get_name
 - TeX, [7](#)
- get_path
 - TeX, [7](#)
- open
 - TeX, [8](#)
- open_rewritemode
 - TeX, [8](#)
- operator<<
 - TeX, [8](#)
- rmfiles
 - TeX, [9](#)
- set_image_density
 - TeX, [9](#)
- TeX::TeXException, [10](#)
 - TeXException, [10](#)
 - what, [10](#)
- TeXException
 - TeX::TeXException, [10](#)
- TeX, [5](#)
 - ~TeX, [6](#)
 - close, [6](#)
 - do_not_cancel, [7](#)
 - exists, [7](#)
 - get_fullpath_ext, [7](#)
 - get_name, [7](#)
 - get_path, [7](#)
 - open, [8](#)
 - open_rewritemode, [8](#)
 - operator<<, [8](#)
 - rmfiles, [9](#)
 - set_image_density, [9](#)
 - TeX, [6](#)
 - to, [9](#)
 - to_pdf, [9](#)
 - to
 - TeX, [9](#)
 - to_pdf
 - TeX, [9](#)
 - what
 - TeX::TeXException, [10](#)