excerpts

Generated by Doxygen 1.8.8

Fri Apr 7 2017 21:23:46

Contents

1 Namespace Index 1

1.1 Packages . . . 1

2 File Index 3

2.1 File List . . . 3

3 Namespace Documentation 5

3.1 excerpts Namespace Reference . . . 5

3.2 excerpts.command_line Namespace Reference . . . 5

3.2.1 Function Documentation . . . 5

3.2.1.1 main . . . 5

3.2.1.2 make_parser . . . 6

3.3 excerpts.main Namespace Reference . . . 6

3.3.1 Function Documentation . . . 6

3.3.1.1 convert . . . 6

3.3.1.2 excerpt . . . 7

3.3.1.3 extract . . . 7

3.3.1.4 modify_path . . . 8

3.4 excerpts.op Namespace Reference . . . 8

3.4.1 Function Documentation . . . 8

3.4.1.1 is_tool . . . 8

3.4.1.2 pandoc . . . 9

3.5 excerpts.ui Namespace Reference . . . 9

3.5.1 Function Documentation . . . 9

3.5.1.1 excerpts . . . 10

4 File Documentation 11

4.1 excerpts/__init__.py File Reference . . . 11

4.2 excerpts/command_line.py File Reference . . . 11

4.3 excerpts/main.py File Reference . . . 11

4.4 excerpts/op.py File Reference . . . 12

4.5 excerpts/ui.py File Reference . . . 12

Index 13

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

excerpts 5

excerpts.command_line 5

excerpts.main 6

excerpts.op 8

excerpts.ui 9

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

excerpts/__init__.py 11
excerpts/command_line.py 11
excerpts/main.py 11
excerpts/op.py 12
excerpts/ui.py 12

Chapter 3

Namespace Documentation

3.1 excerpts Namespace Reference

Namespaces

- command_line
- main
- op
- ui

3.2 excerpts.command_line Namespace Reference

Functions

- def make_parser

  Use a Custom Parser Function.
- def main

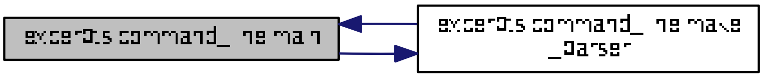
  A User Command Line Interface.

3.2.1 Function Documentation

3.2.1.1 def excerpts.command_line.main ( )

A User Command Line Interface.

Here is the call graph for this function:

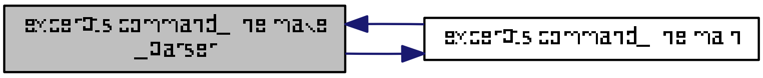


3.2.1.2 def excerpts.command_line.make_parser ( )

Use a Custom Parser Function.

To keep main() tiny.

Here is the call graph for this function:



3.3 excerpts.main Namespace Reference

Functions

- def extract

  Extract Matching Lines.
- def convert

  Convert Lines to Markdown.
- def excerpt

  Extract and Convert Matching Lines.
- def modify_path

  Modify a Path.

3.3.1 Function Documentation

3.3.1.1 def excerpts.main.convert (*lines*, *comment_character*, *magic_character*, *allow_pep8* = `True`)

Convert Lines to Markdown.

Remove whitespace and magic characters from lines and output valid markdown.

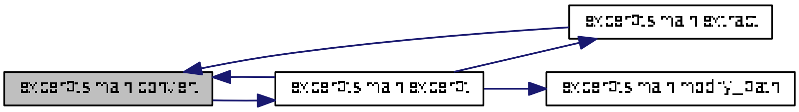
Parameters

lines	The lines to be converted.
comment_↩ character	The comment character of the language.
magic_character	The magic character marking lines as excerpts.
allow_pep8	Allow for a leading comment character and space to confrom to PEP 8 block comments.

Returns

A list of strings containing the lines converted.

Here is the call graph for this function:



3.3.1.2 def excerpts.main.excerpt (*lines*, *comment_character*, *magic_character*, *allow_pep8* = `True`)

Extract and Convert Matching Lines.

Just a wrapper to extract() and convert().

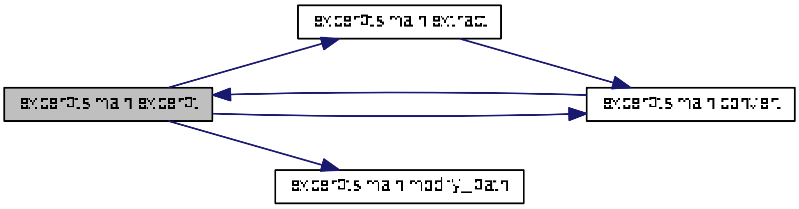
Parameters

lines	a list containing the code lines.
comment_↩ character	The comment character of the files language.
magic_character	The magic character marking lines as excerpts.
allow_pep8	Allow for a leading comment character and space to confrom to PEP 8 block comments.

Returns

A list of strings containing the lines extracted and converted.

Here is the call graph for this function:



3.3.1.3 def excerpts.main.extract (*lines*, *comment_character*, *magic_character*, *allow_pep8* = `True`)

Extract Matching Lines.

Extract all lines starting with a combination of comment_character and magic_character from a list.

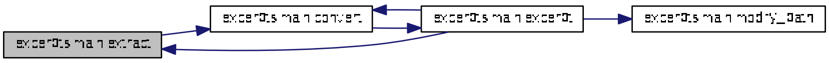
Parameters

lines	a list containing the code lines.
comment_↩ character	The comment character of the language.
magic_character	The magic character marking lines as excerpts.
allow_pep8	Allow for a leading comment character and space to confron to PEP 8 block comments.

Returns

A list of strings containing the lines extracted.

Here is the call graph for this function:



3.3.1.4 def excerpts.main.modify_path (*file_name*, *postfix* = `""`, *prefix* = `""`, *output_path* = `""`, *extension* = `None`)

Modify a Path.

Add a postfix and a prefix to the basename of a path and change it's extension.

Parameters

file_name	The file name to be modified.
postfix	Set the output file postfix.
prefix	Set the output file prefix.
extension	Set a new file extension.
output_path	Set a new file name or an output directory.

Returns

A string containing the modified path.

3.4 excerpts.op Namespace Reference

Functions

- def is_tool

  Test if a Program is Installed.
- def pandoc

  Run Pandoc on a File.

3.4.1 Function Documentation

3.4.1.1 def excerpts.op.is_tool (*name*)

Test if a Program is Installed.

Will raise an error if the programm is not found.

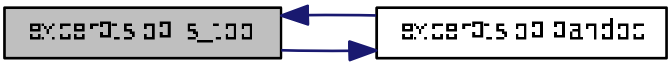
Parameters

name	The name of the program to be tested for.

Returns

True if the Program is installed.

Here is the call graph for this function:



3.4.1.2 def excerpts.op.pandoc (*file_name*, *compile_latex* = `False`, *formats* = `"tex"`)

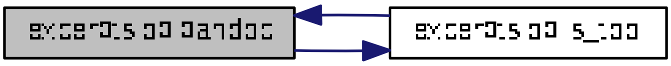
Run Pandoc on a File.

file_name The file on which to run pandoc. formats The pandoc output formats to be used. compile_latex Compile the LaTeX file?

Returns

0 if parsing was successful, 1 otherwise.

Here is the call graph for this function:



3.5 excerpts.ui Namespace Reference

Functions

- def excerpts

    Extract, Convert and Save Markdown Style Comments From a File.

3.5.1 Function Documentation

3.5.1.1 def excerpts.ui.excerpts (*file_name*, *comment_character* = `"#"`, *magic_character* = `"%"`, *allow_pep8* = `True`, *output_path* = `""`, *prefix* = `""`, *postfix* = `""`, *run_pandoc* = `True`, *compile_latex* = `False`, *pandoc_formats* = `"tex"`)

Extract, Convert and Save Markdown Style Comments From a File.

This is merely a wrapper to excerpt(), modify_path() and pandoc().

Parameters

file_name	The file from which the lines are to be extracted.
comment_↩ character	The comment character of the files language.
magic_character	The magic character marking lines as excerpts.
allow_pep8	Remove a leading single comment character and blank.
output_path	Set a new file name or an output directory.
postfix	Set the output file postfix.
prefix	Set the output file prefix.
run_pandoc	Run pandoc on the markdown file created?
compile_latex	Compile the LaTeX file?
pandoc_formats	The pandoc output formats to be used.

Returns

0 if output generation was successful, 1 otherwise.

Chapter 4

File Documentation

4.1 excerpts/__init__.py File Reference

Namespaces

- excerpts

4.2 excerpts/command_line.py File Reference

Namespaces

- excerpts.command_line

Functions

- def excerpts.command_line.make_parser

  Use a Custom Parser Function.
- def excerpts.command_line.main

  A User Command Line Interface.

4.3 excerpts/main.py File Reference

Namespaces

- excerpts.main

Functions

- def excerpts.main.extract

  Extract Matching Lines.
- def excerpts.main.convert

  Convert Lines to Markdown.
- def excerpts.main.excerpt

  Extract and Convert Matching Lines.
- def excerpts.main.modify_path

  Modify a Path.

4.4 excerpts/op.py File Reference

Namespaces

- excerpts.op

Functions

- def excerpts.op.is_tool

  Test if a Program is Installed.
- def excerpts.op.pandoc

  Run Pandoc on a File.

4.5 excerpts/ui.py File Reference

Namespaces

- excerpts.ui

Functions

- def excerpts.ui.excerpts

  Extract, Convert and Save Markdown Style Comments From a File.

Index

excerpts, 5