NAME
pygettext - Python equivalent of xgettext(1)

SYNOPSIS
pygettext [*OPTIONS*] *INPUTFILE* ...

DESCRIPTION
pygettext is deprecated. The current version of xgettext supports many languages, including Python.

pygettext uses Python's standard tokenize module to scan Python source code, generating .pot files identical to what GNU xgettext generates for C and C++ code. From there, the standard GNU tools can be used.

pygettext searches only for _() by default, even though GNU xgettext recognizes the following keywords: gettext, dgettext, dcgettext, and gettext_noop. See the **-k**/**--keyword** flag below for how to augment this.

OPTIONS
-a, **--extract-all**
>Extract all strings.

-d, **--default-domain**=*NAME*
>Rename the default output file from messages.pot to name.pot.

-E, **--escape**
>Replace non-ASCII characters with octal escape sequences.

-D, **--docstrings**
>Extract module, class, method, and function docstrings. These do not need to be wrapped in _() markers, and in fact cannot be for Python to consider them docstrings. (See also the **-X** option).

-h, **--help**
>Print this help message and exit.

-k, **--keyword**=*WORD*
>Keywords to look for in addition to the default set, which are: _

>You can have multiple **-k** flags on the command line.

-K, **--no-default-keywords**
>Disable the default set of keywords (see above). Any keywords explicitly added with the **-k**/**--keyword** option are still recognized.

--no-location
>Do not write filename/lineno location comments.

-n, **--add-location**
>Write filename/lineno location comments indicating where each extracted string is found in the source. These lines appear before each msgid. The style of comments is controlled by the **-S**/**--style** option. This is the default.

-o, **--output**=*FILENAME*
>Rename the default output file from messages.pot to FILENAME. If FILENAME is '-' then the output is sent to standard out.

-p, **--output-dir**=*DIR*
>Output files will be placed in directory DIR.

-S, **--style**=*STYLENAME*
>Specify which style to use for location comments. Two styles are supported:

>• Solaris # File: filename, line: line-number

>• GNU #: filename:line

>The style name is case insensitive. GNU style is the default.

-v, **--verbose**
>Print the names of the files being processed.

-V, **--version**
>Print the version of pygettext and exit.

-w, **--width**=*COLUMNS*
>Set width of output to columns.

-x, **--exclude-file**=*FILENAME*
>Specify a file that contains a list of strings that are not be extracted from the input files. Each string to be excluded must appear on a line by itself in the file.

-X, **--no-docstrings**=*FILENAME*
>Specify a file that contains a list of files (one per line) that should not have their docstrings extracted. This is only useful in conjunction with the **-D** option above.

If 'INPUTFILE' is -, standard input is read.

BUGS

pygettext attempts to be option and feature compatible with GNU xgettext where ever possible. However some options are still missing or are not fully implemented. Also, xgettext's use of command line switches with option arguments is broken, and in these cases, pygettext just defines additional switches.

AUTHOR

pygettext is written by Barry Warsaw <barry@zope.com>.

Joonas Paalasmaa <joonas.paalasmaa@iki.fi> put this manual page together based on "pygettext --help".