



Full credit is given to the above companies including the Operating System (OS) that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'Glib::KeyFile.3pm'

\$ man Glib::KeyFile.3pm

Glib::KeyFile(3pm) User Contributed Perl Documentation Glib::KeyFile(3pm)

NAME

Glib::KeyFile - Parser for .ini-like files

SYNOPSIS

```
use Glib;

$data .= $_ while (<DATA>);

$f = Glib::KeyFile->new;
$f->load_from_data($data);
if ($f->has_group('Main') && $f->has_key('Main', 'someotherkey')) {
    $val = $f->get_integer('Main', 'someotherkey');
    print $val . "\n";
}

0;

__DATA__

# a comment

[MainSection]

somekey=somevalue

someotherkey=42

someboolkey=true
```

listkey=1;1;2;3;5;8;13;21

localekey=Good Morning

localekey[it]=Buon giorno

localekey[es]=Buenas dias

localekey[fr]=Bonjour

DESCRIPTION

Glib::KeyFile lets you parse, edit or create files containing groups of key-value pairs, which we call key files for lack of a better name. Several freedesktop.org specifications use key files now, e.g the Desktop Entry Specification and the Icon Theme Specification.

The syntax of key files is described in detail in the Desktop Entry Specification, here is a quick summary: Key files consists of groups of key-value pairs, interspersed with comments.

METHODS

keyfile = Glib::KeyFile->new

boolean = \$key_file->get_boolean (\$group_name, \$key)

? \$group_name (string)

? \$key (string)

Retrieves a boolean value from \$key inside \$group_name.

May croak with a Glib::Error in \$@ on failure.

list = \$key_file->get_boolean_list (\$group_name, \$key)

? \$group_name (string)

? \$key (string)

Retrieves a list of booleans from \$key inside \$group_name.

May croak with a Glib::Error in \$@ on failure.

`$key_file->set_boolean_list ($group_name, $key, ...)`

? `$group_name` (string)

? `$key` (string)

? ... (list) list of booleans

Sets a list of booleans in `$key` inside `$group_name`. If `$key` cannot be found then it is created. If `$group_name` cannot be found then it is created.

`$key_file->set_boolean ($group_name, $key, $value)`

? `$group_name` (string)

? `$key` (string)

? `$value` (boolean)

Sets a boolean value to `$key` inside `$group_name`. If `$key` is not found, it is created.

`string = $key_file->get_comment ($group_name=undef, $key=undef)`

? `$group_name` (string or undef)

? `$key` (string or undef)

Retrieves a comment above `$key` from `$group_name`. If `$key` is undef then `$comment` will be read from above `$group_name`. If both `$key` and `$group_name` are undef, then `$comment` will be read from above the first group in the file.

May croak with a Glib::Error in \$@ on failure.

`$key_file->set_comment ($group_name, $key, $comment)`

? \$group_name (string or undef)

? \$key (string or undef)

? \$comment (string)

Places a comment above \$key from \$group_name. If \$key is undef then \$comment will be written above \$group_name. If both \$key and \$group_name are undef, then \$comment will be written above the first group in the file.

May croak with a Glib::Error in \$@ on failure.

```
double = $key_file->get_double ($group_name, $key)
```

? \$group_name (string)

? \$key (string)

Retrieves a double value from \$key inside \$group_name.

May croak with a Glib::Error in \$@ on failure.

Since: glib 2.12

```
list = $key_file->get_double_list ($group_name, $key)
```

? \$group_name (string)

? \$key (string)

Retrieves a list of doubles from \$key inside \$group_name.

May croak with a Glib::Error in \$@ on failure.

Since: glib 2.12

`$key_file->set_double_list ($group_name, $key, ...)`

? `$group_name` (string)

? `$key` (string)

? ... (list) list of doubles

Sets a list of doubles in `$key` inside `$group_name`. If `$key` cannot be found then it is created. If `$group_name` cannot be found then it is created.

Since: glib 2.12

`$key_file->set_double ($group_name, $key, $value)`

? `$group_name` (string)

? `$key` (string)

? `$value` (double)

Sets a double value to `$key` inside `$group_name`. If `$key` is not found, it is created.

Since: glib 2.12

`list = $key_file->get_groups`

Returns the list of groups inside the `key_file`.

`boolean = $key_file->has_group ($group_name)`

? `$group_name` (string)

Checks whether `$group_name` is present in `$key_file`.

`boolean = $key_file->has_key ($group_name, $key)`

? \$group_name (string)

? \$key (string)

Checks whether \$group_name has \$key in it.

May croak with a Glib::Error in \$@ on failure.

integer = \$key_file->get_integer (\$group_name, \$key)

? \$group_name (string)

? \$key (string)

Retrieves an integer value from \$key inside \$group_name.

May croak with a Glib::Error in \$@ on failure.

list = \$key_file->get_integer_list (\$group_name, \$key)

? \$group_name (string)

? \$key (string)

Retrieves a list of integers from \$key inside \$group_name.

May croak with a Glib::Error in \$@ on failure.

\$key_file->set_integer_list (\$group_name, \$key, ...)

? \$group_name (string)

? \$key (string)

? ... (list) list of integers

Sets a list of doubles in \$key inside \$group_name. If \$key cannot be found then it is created. If \$group_name cannot be found then it is created.

`$key_file->set_integer ($group_name, $key, $value)`

? \$group_name (string)

? \$key (string)

? \$value (integer)

Sets an integer value to \$key inside \$group_name. If \$key is not found, it is created.

`list = $key_file->get_keys ($group_name)`

? \$group_name (string)

Returns the list of keys inside a group of the key file.

May croak with a `Glib::Error` in \$@ on failure.

`$key_file->set_list_separator ($separator)`

? \$separator (string)

Sets the list separator character.

`boolean = $key_file->load_from_data ($buf, $flags)`

? \$buf (scalar)

? \$flags (`Glib::KeyFileFlags`)

Parses a string containing a key file structure.

May croak with a `Glib::Error` in \$@ on failure.

`boolean = $key_file->load_from_data_dirs ($file, $flags)`

`(boolean, scalar) = $key_file->load_from_data_dirs ($file, $flags)`

? `$file` (string)

? `$flags` (Glib::KeyFileFlags)

Parses a key file, searching for it inside the data directories. In scalar context, it returns a boolean value (true on success, false otherwise); in array context, it returns a boolean value and the full path of the file.

May croak with a Glib::Error in `$@` on failure.

`boolean = $key_file->load_from_dirs ($file, $flags, @search_dirs)`

`(boolean, scalar) = $key_file->load_from_dirs ($file, $flags, @search_dirs)`

? `$file` (string)

? `$flags` (Glib::KeyFileFlags)

? ... (list)

Parses a key file, searching for it inside the specified directories. In scalar context, it returns a boolean value (true on success, false otherwise); in array context, it returns a boolean value and the full path of the file.

May croak with a Glib::Error in `$@` on failure.

Since: glib 2.14

`boolean = $key_file->load_from_file ($file, $flags)`

? `$file` (string)

? `$flags` (Glib::KeyFileFlags)

Parses a key file.

May croak with a Glib::Error in \$@ on failure.

`string = $key_file->get_locale_string ($group_name, $key, $locale=undef)`

? \$group_name (string)

? \$key (string)

? \$locale (string or undef)

Returns the value associated with \$key under \$group_name translated in the given \$locale

if available. If \$locale is undef then the current locale is assumed.

May croak with a Glib::Error in \$@ on failure.

`list = $key_file->get_locale_string_list ($group_name, $key, $locale)`

? \$group_name (string)

? \$key (string)

? \$locale (string)

May croak with a Glib::Error in \$@ on failure.

`$key_file->set_locale_string_list ($group_name, $key, $locale, ...)`

? \$group_name (string)

? \$key (string)

? \$locale (string)

? ... (list)

Associates a list of string values for \$key and \$locale under \$group_name. If the translation for \$key cannot be found then it is created.

\$key_file->set_locale_string (\$group_name, \$key, \$locale, \$string)

? \$group_name (string)

? \$key (string)

? \$locale (string)

? \$string (string)

\$key_file->remove_comment (\$group_name=undef, \$key=undef)

? \$group_name (string or undef)

? \$key (string or undef)

Removes a comment from a group in a key file. If \$key is undef, the comment will be removed from above \$group_name. If both \$key and \$group_name are undef, the comment will be removed from the top of the key file.

May croak with a Glib::Error in \$@ on failure.

\$key_file->remove_group (\$group_name)

? \$group_name (string)

Removes a group from a key file.

May croak with a Glib::Error in \$@ on failure.

\$key_file->remove_key (\$group_name, \$key)

? \$group_name (string)

? \$key (string)

Removes a key from \$group_name.

May croak with a Glib::Error in \$@ on failure.

string = \$key_file->get_start_group

Returns the first group inside a key file.

string = \$key_file->get_string (\$group_name, \$key)

? \$group_name (string)

? \$key (string)

Retrieves a string value from \$key inside \$group_name.

May croak with a Glib::Error in \$@ on failure.

list = \$key_file->get_string_list (\$group_name, \$key)

? \$group_name (string)

? \$key (string)

Retrieves a list of strings from \$key inside \$group_name.

May croak with a Glib::Error in \$@ on failure.

\$key_file->set_string_list (\$group_name, \$key, ...)

? \$group_name (string)

? \$key (string)

? ... (list) list of strings

Sets a list of strings in \$key inside \$group_name. The strings will be escaped if contain special characters. If \$key cannot be found then it is created. If \$group_name cannot be found then it is created.

\$key_file->set_string (\$group_name, \$key, \$value)

? \$group_name (string)

? \$key (string)

? \$value (string)

Sets a string value to \$key inside \$group_name. The string will be escaped if it contains special characters. If \$key is not found, it is created.

string = \$key_file->to_data

Returns the key file as a string.

May croak with a Glib::Error in \$@ on failure.

string = \$key_file->get_value (\$group_name, \$key)

? \$group_name (string)

? \$key (string)

Retrieves the literal value of \$key inside \$group_name.

May croak with a Glib::Error in \$@ on failure.

\$key_file->set_value (\$group_name, \$key, \$value)

? \$group_name (string)

? \$key (string)

? \$value (string)

Sets the literal value of \$key inside \$group_name. If \$key cannot be found, it is created. If \$group_name cannot be found, it is created.

ENUMS AND FLAGS

flags Glib::KeyFileFlags

? 'none' / 'G_KEY_FILE_NONE'

? 'keep-comments' / 'G_KEY_FILE_KEEP_COMMENTS'

? 'keep-translations' / 'G_KEY_FILE_KEEP_TRANSLATIONS'

SEE ALSO

Glib

COPYRIGHT

Copyright (C) 2003-2011 by the gtk2-perl team.

This software is licensed under the LGPL. See Glib for a full notice.