

USER DOCUMENTATION OF TGFParse.TGFParse V0.0.1-beta

Table of content :

- 1) Object [parser] as class(TGFParse.TGFParse)
 - 1.1) Function : parser.__init__
 - 1.2) Function : parser.create
 - 1.3) Function : parser.read
 - 1.4) Function : parser.write
 - 1.5) Object [parser.about] as class(TGFParse.TGFParseAbout)
 - 1.5.1) Function : parser.about.get_name
 - 1.5.2) Function : parser.about.set_name
 - 1.5.3) Function : parser.about.get_author
 - 1.5.4) Function : parser.about.set_author
 - 1.5.5) Function : parser.about.get_version
 - 1.5.6) Function : parser.about.set_version
 - 1.5.7) Function : parser.about.get_description
 - 1.5.8) Function : parser.about.set_description
 - 1.6) Object [parser.GUI] as class(TGFParse.TGFParseGUI)
 - 1.6.1) Function : parser.GUI.insert_pyp_file
 - 1.6.2) Function : parser.GUI.insert_glade_file
 - 1.7) Object [parser.i18n] as class(TGFParse.TGFParseI18N)
 - 1.7.1) Function : parser.i18n.add_language
 - 1.7.2) Function : parser.i18n.delete_language
 - 1.7.3) Function : parser.i18n.get_languages_list
 - 1.7.4) Function : parser.i18n.add_message
 - 1.7.5) Function : parser.i18n.delete_message
 - 1.7.6) Function : parser.i18n.get_messages_list
 - 1.7.7) Function : parser.i18n.get_message_value_dict
 - 1.7.8) Function : parser.i18n.set_message_value
 - 1.8) Object [parser.icons] as class(TGFParse.TGFParseIcons)
 - 1.8.1) Function : parser.icons.insert
 - 1.8.2) Function : parser.icons.remove
 - 1.9) Object [parser.libs] as class(TGFParse.TGFParseLibs)
 - 1.9.1) Function : parser.libs.insert
 - 1.9.2) Function : parser.libs.remove
 - 1.10) Object [parser.scripts] as class(TGFParse.TGFParseScripts)
 - 1.10.1) Function : parser.scripts.insert
 - 1.11) Object [parser.settings] as class(TGFParse.TGFParseSettings)
 - 1.11.1) Function : parser.settings.get_generals_list
 - 1.11.2) Function : parser.settings.get_general_value
 - 1.11.3) Function : parser.settings.set_general_value
 - 1.11.4) Function : parser.settings.get_parameters_list
 - 1.11.5) Function : parser.settings.add_parameter
 - 1.11.6) Function : parser.settings.delete_parameter
 - 1.11.7) Function : parser.settings.get_parameter_value
 - 1.11.8) Function : parser.settings.set_parameter_value
 - 1.12) Object [parser.sounds] as class(TGFParse.TGFParseSounds)
 - 1.12.1) Function : parser.sounds.insert
 - 1.12.2) Function : parser.sounds.remove

1) Object [parser] as class(TGFParse.TGFParse)

Class which parse a TGF file.

Global variables of this class:

"tgf_path" as string : Path of the TGF file.
"version" as string : Version of this class.

Objects in this class:

"about" as instance of class(TGFParseAbout) :
Object which manages the 'about' xml file.
"GUI" as instance of class(TGFParseGUI) :
Object which manages the GUI files.
"i18n" as instance of class(TGFParseI18N) :
Object which manages the internationalization files
"icons" as instance of class(TGFParseIcons) :
Object which manages the 'png' icons.

"libs" as instance of class(TGFParseLibs) :
 Object which manages the added python libraries.
 "scripts" as instance of class(TGFParseScripts) :
 Object which manages the script files of the TGF file.
 "settings" as instance of class(TGFParseSettings) :
 Object which manages the 'settings' xml file.
 "sounds" as instance of class(TGFParseSounds) :
 Object which manages the wav files.

Defines:

"lang_country_list" as list of strings:
 This list contains the supported languages by the TGF files.
 ['de_ALL', 'en_GB', 'en_US', 'ar_ALL', 'da_ALL', 'es_ALL',
 'fr_ALL', 'it_ALL', 'nl_BE', 'nl_NL', 'no_ALL', 'pt_ALL',
 'sv_ALL']
 "lang_ab_to_lang_country" as dictionary:
 Is a tux api - TGF translation of the languages name.
 {'D' : 'de_ALL',
 'GB' : 'en_GB',
 'US' : 'en_US',
 'AR' : 'ar_ALL',
 'DK' : 'da_ALL',
 'E' : 'es_ALL',
 'F' : 'fr_ALL',
 'I' : 'it_ALL',
 'B' : 'nl_BE',
 'NL' : 'nl_NL',
 'N' : 'no_ALL',
 'P' : 'pt_ALL',
 'S' : 'sv_ALL'}

1.1) Function : parser. init

Constructor

Parameters:

'tgf_path' as string : Path of the TGF file.

1.2) Function : parser.create

Create a new TGF file.

1.3) Function : parser.read

Read a TGF file.

1.4) Function : parser.write

Write the TGF file.

1.5) Object [parser.about] as class(TGFParser.TGFParseAbout)

Class which manages the 'about' file of the TGF file.

Global variables of this class:

"struct" as dict : Dictionary of 'about' values

1.5.1) Function : parser.about.get_name

Get the value of 'gadget_name'.

Returns:

A string.

1.5.2) Function : parser.about.set_name

Set the value of 'gadget_name'

Parameters:

"name" as string : Name of the gadget

Returns:

"True"

1.5.3) Function : parser.about.get_author

Get the value of 'gadget_author'

Returns:

A string.

1.5.4) Function : parser.about.set_author

Set the value of 'gadget_author'

Parameters:

"author" as string : Author of the gadget

Returns:

"True"

1.5.5) Function : parser.about.get_version

Get the value of 'gadget_version'

Returns:

A string.

1.5.6) Function : parser.about.set_version

Set the value of 'gadget_version'

Parameters:

"version" as string : Version of the gadget

Returns:

"True"

1.5.7) Function : parser.about.get_description

Get the value of 'gadget_description'

Returns:

A string.

1.5.8) Function : parser.about.set_description

Set the value of 'gadget_description'

Parameters:

"description" as string : Description of the gadget

Returns:

"True"

1.6) Object [parser.GUI] as class(TGFParser.TGFParseGUI)

Class which manages the GUI files of the TGF file.

Global variables of this class:

"struct" as dict : Dictionary of GUI '.pyp' code

1.6.1) Function : parser.GUI.insert pyp file

Insert a pyp file.

Parameters:

"pyp_path" as string : Path of the 'pyp' source file
"gui_name" as string : Name of the GUI

Returns:

A boolean which indicates if the file has been inserted

1.6.2) Function : parser.GUI.insert glade file

Insert a glade file.

Parameters:

"glade_path" as string : Path of the 'glade' source file
"gui_name" as string : Name of the GUI

Returns:

A boolean which indicates if the file has been inserted

1.7) Object [parser.i18n] as class(TGFParser.TGFParseI18N)

Class which manages the internationalization files of the TGF file.

Global variables of this class:

"languages_struct" as dict : Dictionary of languages

1.7.1) Function : parser.i18n.add language

Add a language.

Parameters:

"lang_country" as string : Language acronym

Returns:

A boolean which indicates if the language has been added.

1.7.2) Function : parser.i18n.delete language

Remove a language.

Parameters:

"lang_country" as string : Language acronym

Returns:

A boolean which indicates if the language has been deleted.

1.7.3) Function : parser.i18n.get languages list

Get the list of the languages supported.

Returns:

the list of languages acronym.

1.7.4) Function : parser.i18n.add message

Add a new message.

Parameters:

"msg_name" as string : Name of the message
"default_value" as string : Value of the message

Returns:

A boolean which indicates if the message has been added.

Comments:

The value of message for all languages is set with the "default_value" value.

We recommend to using a default message in english.

1.7.5) Function : parser.i18n.delete_message

Delete a message.

Parameters:

"msg_name" as string : Name of the message

Returns:

A boolean which indicates if the message has been deleted.

1.7.6) Function : parser.i18n.get_messages_list

Get the list of the messages.

Returns:

The list of the messages name.

1.7.7) Function : parser.i18n.get_message_value_dict

Get the translations of a message.

Parameters:

"msg_name" as string : Name of the message

Returns:

A dictionary of values by language acronym.

1.7.8) Function : parser.i18n.set_message_value

Set the value of a message for the specified language.

Parameters:

"msg_name" as string : Name of the message

"msg_value" as string : Value of the message

"lang_country" as string : Language acronym

Returns:

A boolean which indicates if the value of the message has been set.

1.8) Object [parser.icons] as class(TGFParseIcons)

Class which manages the 'png' icons in the TGF file.

Global variables of this class:

"struct" as dict : Dictionary of '.png' icons

1.8.1) Function : parser.icons.insert

Insert a 'png' icon in the TGF file.

Parameters:

"path" as string : Path of the 'png' icon

Returns:

A boolean which indicates if the icon has been

inserted.

1.8.2) Function : *parser.icons.remove*

Remove an icon from the TGF file

Parameters:

"icon_name" as string : Name of the icon

Returns:

A boolean which indicates if the icon has been removed.

1.9) Object [*parser.libs*] as class(*TGFParser.TGFParseLibs*)

Class which manages the added libraries in the TGF file.

Global variables of this class:

"struct" as dict : Dictionary of '.py' libraries

1.9.1) Function : *parser.libs.insert*

Insert a python library in the TGF file.

Parameters:

"path" as string : Path a the python file

Returns:

A boolean which indicates if the library has been inserted.

1.9.2) Function : *parser.libs.remove*

Remove an library from the TGF file

Parameters:

"lib_name" as string : Name of the 'py' library

Returns:

A boolean which indicates if the icon has been removed.

1.10) Object [*parser.scripts*] as class(*TGFParser.TGFParseScripts*)

Class which manages the script files of the TGF file.

Global variables of this class:

"struct" as dict : Dictionary of '.pyp' script codes

1.10.1) Function : *parser.scripts.insert*

Insert a pyp script file.

Parameters:

"pyp_path" as string : Path of the 'pyp' source file

Returns:

A boolean which indicates if the file has been inserted.

1.11) Object [*parser.settings*] as class(*TGFParser.TGFParseSettings*)

Class which manages the 'settings' file of the TGF file.

Global variables of this class:

"struct" as dict : Dictionary of settings

1.11.1) Function : parser.settings.get_generals_list

Get the list of the general settings.

Returns:
The list of general settings.

1.11.2) Function : parser.settings.get_general_value

Get the value of a general setting.

Parameters:
"name" as string : Name of the general item

Returns:
The value of the general item.

1.11.3) Function : parser.settings.set_general_value

Set the value of a general setting.

Parameters:
"name" as string : Name of the general item
"value" as untyped : Value of the general item

Returns:
A boolean which indicates value has been set.

1.11.4) Function : parser.settings.get_parameters_list

Get the list of the parameters.

Returns:
The list of parameters.

1.11.5) Function : parser.settings.add_parameter

Add a new parameter.

Parameters:
"name" as string : Name of the parameter
"value" as untyped : Value of the parameter

Returns:
A boolean which indicates if the parameter has been added.

1.11.6) Function : parser.settings.delete_parameter

Delete a parameter.

Parameters:
"name" as string : Name of the parameter

Returns:
A boolean which indicates if the parameter has been deleted.

1.11.7) Function : parser.settings.get_parameter_value

Get the value of a parameter.

Parameters:
"name" as string : Name of the parameter

Returns:
The value of the parameter.

1.11.8) Function : *parser.settings.set_parameter_value*

Set the value of a parameter.

Parameters:
"name" as string : Name of the parameter
"value" as untyped : Value of the parameter

Returns:
A boolean which indicates if value has been set.

1.12) Object [*parser.sounds*] as class(*TGFParser.TGFParseSounds*)

Class which manages the wav files in the TGF file.

Global variables of this class:
"struct" as dict : Dictionary of wav files

1.12.1) Function : *parser.sounds.insert*

Insert a wav file in the TGF file.

Parameters:
"path" as string : Path of the wav file

Returns:
A boolean which indicates if the wav file has been inserted.

1.12.2) Function : *parser.sounds.remove*

Remove a wav file from the TGF file

Parameters:
"wav_name" as string : Name of the wav file

Returns:
A boolean which indicates if the wav file has been removed.